

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ІВАНА ПУЛЮЯ**

Кафедра комп'ютерних наук

**М Е Т О Д И Ч Н І В К А З І В К И
до виконання**

**ДИПЛОМНОЇ РОБОТИ
ОКР «МАГІСТР»**

**для студентів спеціальності
8.05010101 – Інформаційні управляючі системи та
технології**

ТЕРНОПІЛЬ

2014

УДК 681.3
ББК 32.97
М54

Укладачі:

О. В. Маєвський, О. В. Мацюк, М. В. Приймак, Г. В. Шимчук

Рецензент

М. М. Касянчук, доцент ТНЕУ

Відповідальний за випуск

М. В. Приймак, д.т.н., професор, зав.каф. КН

Затверджено на засіданні кафедри комп'ютерних наук.

Протокол № 4 від 31 жовтня 2013 р.

Схвалено та рекомендовано до друку методичною комісією
факультету комп'ютерно-інформаційних систем і програмної інженерії
Тернопільського національного технічного університету імені Івана Пулюя.

Протокол № 4 від 29 листопада 2013 р.

М54 **Методичні** вказівки до виконання дипломної роботи ОКР “Магістр”
для студентів спеціальності 8.05010101– Інформаційні управляючі системи
та технології / Укладачі : . – Тернопіль : Вид-во ТНТУ імені Івана Пулюя
,2014. – 196 с.

ЗМІСТ

Вступ.....	5
1 Мета та завдання виконання кваліфікаційної роботи	8
1.1 Мета та завдання виконання кваліфікаційної роботи	8
1.2 Загальні вимоги до змісту та структури роботи.....	10
1.2.1 Структура магістерської роботи.....	11
1.2.2 Вимоги до змісту магістерської кваліфікаційної роботи	12
1.3 Зміст та вимоги до вступу магістерської кваліфікаційної роботи	15
1.4 Зміст та вимоги до основної частини роботи	17
1.4.1 Зміст та вимоги до розділу «Аналітичний огляд літературних та інших джерел»	20
2 Системний аналіз та обґрунтування проблеми	23
2.1 Системний аналіз об'єкту дослідження та предметної області.....	23
2.1.1 Особливості імітаційних моделей прийняття рішень	35
2.1.2 Особливості створення нової системи	44
2.2 Постановка та обґрунтування проблеми.....	46
3 Методи проектування інтелектуальних систем	52
3.1 Вибір та обґрунтування методів проектування інтелектуальних систем..	52
3.2 Вибір та обґрунтування методів проектування бази даних та знань	57
3.2.1 Методологія концептуального проектування реляційних баз даних .	59
3.2.2 Методи логічного проектування баз даних реляційного типу	69
4 Засоби проектування інтелектуальної системи.....	87
4.1 Вибір та обґрунтування систем і технологій управління базами даних....	87
4.2 Вибір та обґрунтування засобів проектування програмних систем	94
4.3 Вимоги до реалізації	96
5 Практична реалізація	102
5.1 Описання реалізації завдання.....	102
5.2 Аналіз отриманих результатів	104
5.2.1 Верифікація та валідація даних	105

5.2.2 Тестування продуктивності (performance testing)	109
5.2.3 Застосування формальних підходів до верифікації та валідації результатів.....	111
5.3 Аналіз контрольних прикладів	112
5.3.1 Особливості валідації інтелектуальних систем	113
6 Оформлення та порядок захисту магістерської роботи	115
6.1 Організація дипломного проектування.....	115
6.1.1 Керівництво магістерською роботою.....	115
6.1.2 Організація консультацій.	116
6.1.3 Етапи виконання магістерської роботи.....	116
6.1.4 Контроль за ходом розробки дипломної роботи.....	118
6.2 Обов'язки осіб за організацію та проведення дипломного проектування	119
6.2.1 Обов'язки керівників дипломних робіт	119
6.2.2 Обов'язки консультантів	120
6.2.3 Обов'язки студента-дипломника.....	120
6.2.4 Обов'язки нормоконтролера	121
6.3 Оформлення матеріалів магістерських робіт	122
6.3.1 Загальні вимоги	122
6.3.2 Вимоги до оформлення пояснювальної записки	124
6.3.3 Вимоги до оформлення графічної документації.....	133
6.4 Перелік документів, які подаються в ДЕК при захисті магістерських кваліфікаційних робіт	133
6.5 Підготовка до захисту та захист магістерської роботи	134
6.6 Подання магістерської роботи	136
Перелік рекомендованих джерел	140
Додатки	

ВСТУП

Магістр – це освітньо-кваліфікаційний рівень вищої освіти, що передбачає здобуття особою повної вищої освіти з відповідної спеціальності на базі освітньо-кваліфікаційного рівня «бакалавр» (базова вища освіта) або освітньо-кваліфікаційного рівня «спеціаліст» (повна вища освіта), а також вищої освіти, здобутої до запровадження в Україні ступеневої вищої освіти.

Підготовка магістрів у системі вищої освіти спрямована на створення умов для творчого розвитку обдарованої особистості і підготовку фахівців за одним із функціональних напрямів діяльності: науково-дослідним (творчим), науково-педагогічним, управлінським.

Особа, яка здобула освітньо-кваліфікаційний рівень «магістр» (надалі - магістр), повинна володіти поглибленими знаннями з обраної спеціальності, вміннями інноваційного характеру, навичками науково-дослідної (творчої), або науково-педагогічної, або управлінської діяльності, набуті певний досвід використання одержаних знань і вміти продукувати (створювати) елементи нових знань для вирішення завдань у відповідній сфері професійної діяльності.

Навчатися за програмою «магістр» можуть особи, які мають відповідну вищу освіту і виявили схильність до науково-дослідної (творчої), або до науково-педагогічної, або до управлінської діяльності, що підтверджується рекомендацією ради факультету вищого навчального закладу.

Згідно з Дублінськими дескрипторами (*Dublin Descriptors*), які були розроблені відповідно до ідей Болонського процесу, випускники магістратури:

- мають продемонструвати знання та вміння на рівні, що забезпечують можливість аналізувати, оцінювати і порівнювати альтернативи, генерувати оригінальні ідеї у відповідній галузі знань;
- можуть застосовувати свої знання і володіють компетенціями, які дозволяють вирішувати завдання у новому, широкому (міждисциплінарному) контексті у відповідній галузі знань;
- спроможні інтегрувати знання, вирішувати складні завдання в умовах неповної інформації з урахуванням соціальної та етичної відповідальності за прийняті рішення;
- володіють методами проведення сучасних експериментів і можуть давати науково обґрунтовану інтерпретацію отриманим результатам;
- можуть чітко, аргументовано доводити до аудиторії фахівців наукову інформацію та свої висновки;
- мають компетенції самоосвіти та саморегулювання.

З урахуванням Дублінських дескрипторів можна сформулювати вимоги до випускників магістратури.

Магістр має:

- перевести одержувані знання в інноваційні технології, перетворюючи нові знання в конкретні пропозиції, демонструючи творчість та гнучкість у застосуванні знань, досвіду і методів;

- методологічними знаннями, уміти аналізувати, оцінювати і порівнювати альтернативи, що стосуються проблеми, уміти генерувати та оптимізувати нові рішення; •

- компетенціями проведення наукових досліджень на сучасному рівні, виконання натурних та імітаційних експериментів, давати обґрунтовану інтерпретацію отриманим результатам;

- узагальнювати і готувати до публікації результати наукових досліджень;

- вплив рішень, що приймаються, на навколишнє середовище і соціум, мати професійну й етичну відповідальність за сталий розвиток суспільства;

- готовим до побудови та викладання навчальних курсів залежно від інтересів аудиторії, слухачів з урахуванням їх вікових, професійних та інших особливостей;

- мотивацію та розуміти необхідність навчання протягом всього життя, володіти досвідом самостійного одержання знань і підвищення кваліфікації;

- здатності роботи в міждисциплінарних командах, уміння адаптуватися до змін, сприяти соціальній згуртованості;

- здатності взаємодії і посередництва, розвинену міжкультурну толерантність і досвід міжкультурної взаємодії;

- відповідний рівень усної та письмової грамотності рідною, і принаймні, однією іноземною мовою (переважно англійською), уміти ефективно спілкуватися з колегами, представляти результати своєї роботи у вигляді звітів, статей, рефератів, використовуючи сучасні засоби презентації;

- підприємницький образ мислення і діяльності;

- цінностями, необхідними для того, щоб жити в умовах складного демократичного суспільства, бути його відповідальним громадянином, мати необхідні соціальні компетенції.

Магістерська підготовка реалізує освітньо-професійні програми, які базуються на проведенні наукових досліджень і орієнтовані на підготовку

фахівців для науково-дослідної і педагогічної діяльності, для роботи у галузі наукоємних технологій.

Магістерська дисертаційна робота (надалі дипломна робота ОКР «Магістр» - дипломна робота), що виконується на завершальному етапі навчання, є обов'язковою для присвоєння кваліфікації «магістр» і повинна передбачати проведення наукових досліджень (творчих розробок) з проблем відповідної галузі.

Цінність дипломної роботи визначається її науково-дослідною значимістю, а також логічністю, доведеністю, зрозумілим і послідовним викладом матеріалу.

Для магістра важливо знати не тільки основні положення, що характеризують кваліфікаційну роботу, але і мати поняття про методологію та організацію науково-дослідної роботи на всіх етапах виконання роботи.

Дані методичні вказівки спрямовані на підвищення якості дипломної роботи (ДР) і призначені для студентів денної форми навчання за спеціальністю 8.05010101 „Інформаційні управляючі системи та технології”, а також для керівників і консультантів ДР. У них регламентується загальний порядок проведення ДР по вказаному напрямку, розглянуті питання вибору тем ДР, організації проектування, описані структура і зміст ДР, дані рекомендації до змісту окремих частин пояснювальної записки, приведені основні вимоги до оформлення матеріалів ДР у відповідності зі стандартами ЄСКД і ДСТУ, викладений порядок представлення і захисту ДР.

В основу методичних вказівок покладені вимоги вищої школи до ДР і закон про вищу освіту.

1 МЕТА ТА ЗАВДАННЯ ВИКОНАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ

Магістратура – навчальний рівень, що спрямований на підвищення ступеня фундаментальної та методологічної підготовленості студентів і формування готовності випускників до самостійного проведення наукових досліджень. У зв'язку з цим провідна роль у навчальному процесі належить такій формі організації навчально-дослідної діяльності як самостійне виконання магістрантом науково-дослідної роботи за конкретною темою. Заключний етап навчання в магістратурі присвячений підготовці та захисту магістерської кваліфікаційної роботи.

Магістерська кваліфікаційна робота є самостійним науковим дослідженням, яке має внутрішню цілісність і відображає хід та результати розроблення вибраної теми. Вона повинна відповідати сучасному рівню розвитку науки і техніки, а її тема повинна бути актуальною.

До дипломного проектування допускається студент, який пройшов повний курс навчання за відповідною освітньо-професійною програмою підготовки (ОПП) та склав усі передбачені навчальним планом заліки та екзамени, тобто виконав усі вимоги навчального плану з напряму підготовки або спеціальності. Права та обов'язки студента-дипломника наведені у додатку А.

1.1 Мета та завдання виконання кваліфікаційної роботи

Магістерська кваліфікаційна робота відрізняється від кваліфікаційної роботи бакалавра ґрунтовнішим теоретичним і практичним опрацюванням проблеми, від дипломного проекту спеціаліста ґрунтовнішою науковою спрямованістю, але на відміну від кандидатської дисертації випускна робота магістра не містити чітко виділеної наукової новизни дослідження.

Метою магістерської кваліфікаційної роботи є:

- виявлення вмінь автора планувати експериментальні дослідження, проводити їх, здійснювати опрацювання експериментальних даних і проводити аналіз отриманих результатів;
- формування у магістра навиків роботи з технічною та довідковою літературою та іншими інформаційними джерелами;
- формування та виявлення вмінь автора аргументовано подавати свої думки технічно грамотною мовою та публічно їх захищати;
- виявлення вмінь автора складати математичні моделі та вирішувати їх зі застосуванням комп'ютерних технологій.

Випускна кваліфікаційна робота виконується на завершальному етапі навчання студентів за фахом «Інформаційні управляючі системи та технології» та передбачає:

- систематизацію, закріплення, розширення теоретичних і практичних знань за спеціальністю «Інформаційні управляючі системи та технології» та застосування їх при вирішенні конкретних наукових, технічних, економічних, виробничих та інших завдань;

- розвиток навичок самостійної роботи та оволодіння методикою створення та досліджень інтелектуальних інформаційних систем, методів та систем підтримання та прийняття рішень, систем баз даних та знань, реалізації імітаційних експериментів на моделях таких систем, пов'язаних з метою кваліфікаційної роботи ;

- тема дипломної роботи визначається кафедрою «Комп'ютерні науки» за пропозицією керівника теми та погодженням зі студентом-магістрантом. Тематика робіт повинна відповідати сучасному стану та перспективним напрямам розвитку спеціальності «Інформаційні управляючі системи та технології», бути пов'язаною з вирішенням практичних завдань підприємств та організацій, тематикою наукової роботи кафедри «Комп'ютерні науки» та з напрямками діяльності підприємств і установ-партнерів і відповідати вимогам кваліфікаційної характеристики спеціаліста за фахом «Інформаційні управляючі системи та технології».

Тематика робіт повинна передбачати можливість самостійного розв'язання інженерних і наукових задач за спеціальністю «Інформаційні управляючі системи та технології» та виявлення елементів самостійності дослідницької роботи.

Тема є не лише назвою магістерської роботи, а й наміченим результатом дослідження, скерованим на вирішення конкретної проблеми.

Отже, мета дослідження полягає у вирішенні конкретної наукової проблеми шляхом вдосконалення вибраної сфери діяльності конкретного об'єкта.

Назва теми має бути якомога коротшою, точною і відповідати змісту магістерської роботи. Не треба вживати в заголовку роботи невизначені формулювання, наприклад: «Аналіз деяких питань...», а також штампованих формулювань типу: «До питання про...», «До вивчення...», «Матеріали до...».

За темою магістерської роботи студент-магістрант у співавторстві з керівником або самостійно повинен підготувати тези доповіді та виступити на студентській науково-практичній конференції університету.

Вимоги до оформлення тез для друку на Web-сторінці ТНТУ (<http://tntu.edu.ua>).

1.2 Загальні вимоги до змісту та структури роботи

Знаючи типову структуру дослідження, магістрант без особливих зусиль може розробити типовий план магістерської кваліфікаційної роботи. Це можна зробити ще до початку безпосереднього дослідження, ґрунтуючись на назві теми. Процес розроблення плану має за суттю механічний характер. Розділи типової структури треба конкретизувати стосовно до теми магістерської роботи.

У процесі роботи над магістерською роботою розвивайте, уточнюйте, деталізуйте її план. В ідеалі треба прийти до такого докладного і детально розробленого плану, який би дав змогу оформити матеріал, починаючи з будь-якого розділу і підрозділу майже в остаточному вигляді.

Після вибору теми і складання першого варіанта плану (змісту) магістерської роботи, необхідно скласти укрупнений план роботи над нею. Складіть жорсткий план-графік написання розділів роботи в цілому. Не затримуйтеся на першому розділі, якнайшвидше завершіть роботу, бо тільки тоді можна буде оцінити дослідження загалом і усунути недоліки.

В укрупненому плані вкажіть основні розділи роботи і детальніше викладіть матеріал частин (підрозділів, пунктів), які зрозумілі й відомі з самого початку роботи. У ньому не потрібна жорстка фіксація змісту й обсягів розділів. У міру виконання дослідження частина розділів буде збільшуватися за обсягом, а деякі виявляться малозначущими або навіть непотрібними. Відзначте у початковому плані календарні терміни найближчих очевидних робіт. Виокремте серед них найважливіші. Відзначте у плані побажання до подальшої роботи.

Магістерська кваліфікаційна робота виконується на ґрунті аналітичного вивчення літератури за фахом та індивідуального завдання і складається з пояснювальної записки і графічного матеріалу. Графічний матеріал до магістерської кваліфікаційної роботи виконується у вигляді презентацій. Окрім того, графічний матеріал може бути виконаний комп'ютерним способом та роздрукований на плотері.

До складу пояснювальної записки входить основна (описова) частина та додатки. У додатки виносяться роздруки програмної реалізації, вхідних і вихідних документів системи, копія графічного матеріалу у зменшеному вигляді. За потреби, за погодженням з керівником в додатки можуть бути винесені й інші матеріали. При виконанні роботи групою студентів (комплексна кваліфікаційна робота) кожен з них виконує власне індивідуальне завдання та окремо оформляє результати роботи.

Загальний обсяг пояснювальної записки магістерської роботи має становити 80-120 сторінок тексту, оформленого згідно з вимогами розділу 6.

1.2.1 Структура магістерської роботи

Магістерська робота повинна містити:

- титульний аркуш;
- індивідуальне завдання;
- анотація;
- анотація іноземною мовою;
- перелік умовних позначень, символів, одиниць, скорочень і термінів;
- зміст;
- вступ;
- основна частина;
- висновки;
- перелік джерел;
- додатки.

Пояснювальна записка до ДР повинна коротко і чітко розкривати творчий задум ДР (таблиця 1.1), містити техніко-економічне обґрунтування, вибір методів розрахунку, розділи спеціальної частини, питання охорони праці.

Таблиця 6.1 – Структура ДР

Структура ДР	Кількість сторінок
Титульний аркуш (додаток В)	
Завдання на дипломну роботу (додаток Д)	2
Анотація (додаток З)	1-2
Анотація іноземною мовою	3
Перелік умовних позначень, символів, одиниць, скорочень і термінів	1-2
Зміст	1-2
Вступ	4-7
Основна частина	65-85
Висновки	1-2
Перелік джерел	1-2
Додатки	

1.2.2 Вимоги до змісту магістерської кваліфікаційної роботи

Титульний аркуш.

Титульний аркуш у вигляді стандартного бланка (додаток В), на якому повинні бути всі необхідні підписи (керівника, рецензента) заповнюється студентом.

Індивідуальне завдання.

Індивідуальне завдання на магістерську роботу містить дані про тему роботи; термін здачі студентом закінченої роботи; вихідні відомості; перелік питань, як належить розробити; перелік графічного матеріалу; перелік консультантів із розділів роботи; календарний план. Завдання підписується керівником, магістром, консультантами і затверджується завідувачем кафедри (додаток Д).

Анотація.

Анотація виконується українською мовою, об'ємом 1 сторінка і містить загальну характеристику поставленого завдання, виконаної роботи та описання отриманих результатів (додаток З).

Викладення матеріалу в анотації повинно бути стислим і точним. Належить використовувати синтаксичні конструкції, притаманні мові ділових документів, уникати складних граматичних зворотів. Необхідно використовувати стандартизовану термінологію, уникати маловідомих термінів і символів.

Після анотації наводять ключові слова. Ключовим словом називається слово або стійке словосполучення із тексту анотації, яке з погляду інформаційного пошуку несе смислове навантаження. Сукупність ключових слів повинна відображувати поза контекстом основний зміст роботи. Загальна кількість ключових слів повинна бути не меншою трьох і не більшою двадцяти.

Ключові слова подають у називному відмінку, друкують в рядок, через кому.

Анотація іноземною мовою.

Після анотації українською мовою розміщують анотацію іноземною мовою (3 сторінки). Анотація виконується іноземною мовою, яку магістр вивчав протягом навчання в університеті (англійською, французькою, німецькою або іспанською) і містить загальну характеристику поставленого завдання, виконаної роботи та описання отриманих результатів.

Викладення матеріалу в анотації повинно бути стислим і точним. Належить використовувати синтаксичні конструкції, притаманні мові ділових документів, уникати складних граматичних зворотів. Необхідно використовувати стандартизовану термінологію, уникати маловідомих термінів і символів.

Після кожної анотації наводять ключові слова відповідною мовою. Ключовим словом називається слово або стійке словосполучення із тексту анотації, яке з погляду інформаційного пошуку несе смислове навантаження. Сукупність ключових слів повинна відображувати поза контекстом основний зміст роботи. Загальна кількість ключових слів повинна бути не меншою трьох і не більшою десяти.

Ключові слова подають у називному відмінку, друкують в рядок, через кому.

Перелік умовних позначень, символів, одиниць, скорочень і термінів.

Якщо в магістерській роботі вжита специфічна термінологія, а також використано маловідомі скорочення, нові символи, позначення і таке інше, то її перелік може бути поданий в роботі у вигляді окремого списку, який розміщують перед змістом. Якщо в анотації роботи використано скорочення, то даний розділ розміщують перед анотацією.

Перелік треба друкувати двома колонками, в яких зліва за абеткою наводять, наприклад, скорочення, справа – їх детальну розшифровку.

Якщо в роботі спеціальні терміни, скорочення, символи, позначення і таке інше повторюються менше трьох разів, перелік не складають, а їх розшифровку наводять у тексті при першому згадуванні.

Зміст.

Зміст подають на початку роботи. Він містить найменування та номери початкових сторінок усіх розділів, підрозділів та пунктів (якщо вони мають заголовки), зокрема вступу, висновків до розділів, загальних висновків, додатків, перелік джерел та ін. Зміст виконується за допомогою відповідних засобів текстового процесора.

Вступ.

Розкриває сутність і стан наукової проблеми (задачі) та її значущість, підстави і віхідні дані для розробки теми, обґрунтування необхідності проведення дослідження.

Далі подають загальну характеристику роботи в рекомендованій нижче послідовності (див. п.1.3).

Рекомендований обсяг вступу 4 -7 сторінок.

Основна частина.

Основна частина магістерської кваліфікаційної роботи складається з розділів, підрозділів, пунктів, підпунктів. Кожний розділ починають з нової сторінки. У кінці кожного розділу формулюють висновки із стислим викладенням наведених у розділі наукових і практичних результатів, що дає змогу вивільнити загальні висновки від другорядних подробиць.

У розділах основної частини подають:

- огляд літератури за темою і вибір напрямів досліджень;
- виклад загальної методики й основних методів досліджень;
- експериментальну частину і методику досліджень;
- відомості про проведені теоретичні і (або) експериментальні дослідження;
- аналіз і узагальнення результатів досліджень.

В огляді літератури здобувач окреслює основні етапи розвитку наукової думки за своєю проблемою. Стисло, критично висвітлюючи роботи попередників, магістр повинен назвати ті питання, що залишились невирішеними і, отже, визначити своє місце у розв'язанні проблеми. Бажано закінчити цей розділ коротким резюме стосовно необхідності проведення досліджень у даній галузі. Загальний обсяг огляду літератури не повинен перевищувати 10-12 % обсягу основної частини роботи.

У другому розділі, як правило, обґрунтовують вибір напрямку досліджень, наводять методи вирішення задач і їх порівняльні оцінки, розробляють загальну методику проведення дисертаційних досліджень. У теоретичних роботах розкривають методи розрахунків, гіпотези, що розглядають, в експериментальних - принципи дії і характеристики розробленої апаратури, оцінки похибок вимірювань.

У наступних розділах з вичерпною повнотою викладають результати власних досліджень автора з висвітленням того нового, що він вносить у розробку проблеми. Здобувач повинен давати оцінку повноти вирішення поставлених задач, оцінку достовірності одержаних результатів (характеристик, параметрів), їх порівняння з аналогічними результатами вітчизняних і зарубіжних праць, обґрунтування потреби додаткових досліджень, негативні результати, які обумовлюють необхідність припинення подальших досліджень.

Виклад матеріалу підпорядковують одній провідній ідеї, чітко визначеній автором.

До кожного розділу наводяться короткі висновки про пророблену роботу.

Висновки.

Викладають найважливіші наукові та практичні результати, одержані в роботі, які повинні містити формулювання розв'язаної наукової проблеми (задачі), її значення для науки і практики. Далі формулюють висновки та рекомендації щодо наукового та практичного використання здобутих результатів. У першому пункті висновків коротко оцінюють стан питання. Далі у висновках розкривають методи вирішення поставленої в кваліфікаційній роботі наукової проблеми (задачі), їх практичний аналіз, порівняння з відомими розв'язаннями.

У висновках необхідно наголосити на якісних і кількісних показниках здобутих результатів, обґрунтувати достовірність результатів, викласти рекомендації щодо їх використання.

Список використаних джерел

Список використаних джерел треба розміщувати одним із таких способів: у порядку появи посилань у тексті (найбільш зручний для користування і рекомендований при написанні дисертацій), в алфавітному порядку прізвищ перших авторів або заголовків у хронологічному порядку.

Бібліографічний опис джерел складають відповідно до чинних стандартів із бібліотечної та видавничої справи. Зокрема потрібну інформацію можна одержати із таких міждержавних і державних стандартів ДСТУ ГОСТ 7.1:2006 «Система стандартів з інформації, бібліотечної та видавничої справи. Бібліографічний запис. Бібліографічний опис. Загальні вимоги та правила складання (ГОСТ 7.1-2003, IDT)», ДСТУ 3582-97 «Інформація та документація Скорочення слів в українській мові у бібліографічному описі. Загальні вимоги та правила», ГОСТ 7.12-93 «СИБИД. Библиографическая запись Сокращение слов на русском языке. Общие требования и правила», ГОСТ 7.11-78 «СИБИД. Сокращение слов и словосочетаний на иностранных европейских языках в библиографическом описании».

Додатки.

До додатків за необхідності доцільно включати допоміжний матеріал:

- копії поданих тез на студентську конференцію;
- проміжні математичні доведення, формули та розрахунки;
- таблиці допоміжних цифрових даних;
- протоколи й акти випробувань, впровадження, розрахунки економічного ефекту;
- інструкції та методики, опис алгоритмів і програм вирішення задач на ЕОМ, розроблених у магістерській роботі;
- допоміжні ілюстрації (роздрук слайдів презентацій, роздруки текстів програмного забезпечення, результати розв'язування контрольних прикладів, графіки, діаграми, ілюстрації, екранні копії, форми документів, у разі необхідності інші матеріали).

1.3 Зміст та вимоги до вступу магістерської кваліфікаційної роботи

У вступі необхідно лаконічно сформулювати суть і стан наукової проблеми (завдання) та її значущість, підстави і вихідні дані для розроблення теми, обґрунтування необхідності виконання дослідження.

Потім подається загальна характеристика магістерської роботи у такій послідовності.

Актуальність теми.

Шляхом критичного аналізу та порівняння з відомими рішеннями проблеми (наукової задачі) обґрунтовують актуальність та доцільність роботи для розвитку відповідної галузі науки чи виробництва, особливо на користь України.

Висвітлення актуальності не повинно бути багатослівним. Досить кількома реченнями висловити головне - суть проблеми або наукового завдання.

Мета і задачі дослідження.

Формулюють мету роботи і задачі, які необхідно розв'язати для досягнення поставленої мети. Не треба формулювати мету як «Дослідження...», «Вивчення...», тому що ці слова вказують на засіб досягнення мети, а не на саму мету.

Об'єкт дослідження.

Об'єкт дослідження - це процес або явище, що породжує проблемну ситуацію і обране для вивчення.

Предмет дослідження.

Предмет дослідження міститься в межах об'єкта. Об'єкт і предмет дослідження як категорії наукового процесу співвідносяться між собою як загальне і часткове. В об'єкті виділяється та його частина, яка є предметом дослідження. Саме на нього скерована основна увага магістранта, оскільки предмет дослідження визначає тему магістерської роботи, яка подається на титульному аркуші у вигляді назви.

Наукова новизна одержаних результатів.

Подають коротку анотацію нових наукових положень (рішень), запропонованих магістрантом особисто. Необхідно показати відмінність одержаних результатів від відомих раніше, описати ступінь новизни (вперше одержано, удосконалено, отримало подальший розвиток).

Кожне наукове положення чітко формулюють, відокремлюючи його основну суть і зосереджуючи особливу увагу на рівні досягнутої при цьому новизни. Сформульоване наукове положення має читатися і сприйматися легко і однозначно (без нагромодження дрібних і таких, що затемнюють його суть деталей і уточнень). У жодному випадку не можна вдаватися до викладу наукового положення у вигляді анотації, коли просто констатують, що в магістерській кваліфікаційній роботі зроблено те й те, а сутності і новизни положення із написаного виявити неможливо. Подання наукових положень у

вигляді анотацій є найпоширенішою помилкою магістрантів під час викладення загальної характеристики роботи.

До цього пункту не можна вводити опис нових прикладних (практичних) результатів, отриманих у вигляді способів, пристроїв, методик, схем, алгоритмів і т. ін. Треба завжди розмежовувати одержані наукові положення і нові прикладні результати, що впливають з теоретичного доробку магістранта.

Усі наукові положення з урахуванням досягнутого ними рівня новизни є теоретичною основою (фундаментом) вирішеного в магістерській кваліфікаційній роботі наукового завдання або наукової проблеми. Насамперед за це магістрантові присуджується кваліфікація магістра з інформаційних управляючих систем та технологій.

Практичне значення одержаних результатів.

У магістерській роботі, що має теоретичне значення, треба подати відомості про наукове використання результатів досліджень чи рекомендації щодо їх використання, а в кваліфікаційній роботі, що має прикладне значення, відомості про практичне застосування одержаних результатів чи рекомендації щодо їх використання. Зазначаючи практичну цінність одержаних результатів, необхідно подати інформацію щодо ступеня готовності до використання чи масштабів використання. Необхідно дати короткі відомості щодо впровадження результатів досліджень із зазначенням назв організацій, в яких здійснена реалізація, форм реалізації та реквізитів відповідних документів.

Апробація результатів магістерської роботи.

Вказується, на яких наукових з'їздах, семінарах, конференціях, симпозіумах, нарадах оприлюднені результати досліджень, введені до магістерської роботи.

1.4 Зміст та вимоги до основної частини роботи

Ґрунтуючись на багаторічному досвіді наукової роботи науково-педагогічних працівників кафедри «Комп'ютерні науки» пов'язаному з вирішенням практичних завдань підприємств та організацій, тематикою та з напрямками діяльності підприємств і установ-партнерів, а також, необхідності відповідати вимогам кваліфікаційної характеристики спеціаліста за фахом «Інформаційні управляючі системи та технології», основна частина магістерської роботи повинна містити:

- аналітичний огляд літературних та інших джерел;
- системний аналіз та обґрунтування проблеми;
- методи та засоби вирішення проблеми;

- практична реалізація;
- спеціальна частина (4-7 ст.);
- обґрунтування економічної ефективності (8-13 ст.);
- охорона праці та безпека в надзвичайних ситуаціях (7-10 ст.);
- екологія (4-7 ст.).

Аналітичний огляд літературних та інших джерел.

Аналітичний огляд літературних та інших джерел - електронні публікації, матеріали Internet тощо (не має перевищувати 10-12% обсягу основної частини).

Аналітичний огляд має містити основні відомості про найновіші та відомі підходи, методи, засоби, алгоритми розв'язання аналогічних задач, які публікуються в сучасній літературі, мати критичний характер і завершуватись висновком про актуальність постановки та розв'язання задачі кваліфікаційної роботи; огляд літературних джерел супроводжується посиланнями на відповідну позицію у списку використаної літератури, використання авторського матеріалу без посилання на відповідні публікації не допускається; у випадку цитування цитата береться в лапки та вказується джерело і номери сторінок, на яких розташована цитата.

Системний аналіз та обґрунтування проблеми.

Системний аналіз об'єкта дослідження та предметної області.

У цьому підрозділі виконується аналіз мети функціонування системи, конкретизація її до окремих критеріїв якості, основних варіантів досягнення мети та наявних ресурсів; аналізується система «азів» - як є: визначається система та зовнішнє середовище, обирається критерій декомпозиції та аналізуються елементи системи, основні бізнес- процеси, потоки даних; як зображальні засоби використовуються структурні та (або) об'єктні нотації; визначаються основні альтернативи побудови системи та в загальному обґрунтовується вибір варіанта побудови системи; висновки розкривають основні можливості та варіанти побудови проекрованої системи з обґрунтуванням доцільності обраного для остаточної реалізації (10-15%).

Постановка та обґрунтування проблеми.

Постановка задачі передбачає описання множини вимог до системи (підсистеми) прийняття (підтримання прийняття) рішень, що реалізуватиметься як конкретизації результатів системного аналізу, вимоги до інтелектуальної складової, вхідних та вихідних даних, форми їх подання та динамічних характеристик, множину бізнес-процесів, що мають бути реалізовані; за необхідності наводиться формальна чи узагальнена модель функціонування системи «to be» – як повинно бути, вимоги до алгоритмів чи технологічних процесів опрацювання інформації (8-10%).

Методи та засоби вирішення проблеми.

Дослідження методів та засобів вирішення проблеми складається з двох частин:

- Вибір та обґрунтування методів вирішення проблеми.
- Вибір та обґрунтування засобів вирішення проблеми.

У розділі «Методи та засоби вирішення проблеми» магістрант розробляє такі питання (8-10%):

1) Методи і засоби представлення знань в системі прийняття рішень, яка розробляється:

- специфіка та особливості розроблюваної системи прийняття рішень;

- вимоги до представлення правил та фактів в системі;
- вибір форми представлення знань;
- порівняння та описання переваг вибраного методу над іншими.

2) Описання механізмів логічного виведення в системі:

- специфіка логічного виведення рішень в системі;
- вимоги до порядку та способів прийняття рішень;
- обґрунтування вибору та переваг методу логічного виведення рішень в системі;

- описання способів та засобів побудови бази знань системи.

3) Технології та засоби побудови баз даних:

- особливості бази даних системи, що розробляється;
- кількісні та якісні характеристики бази даних;
- технологічні вимоги до середовища опрацювання даних (сумісність з іншими системами, мови програмування, інтерфейси, продуктивність, мережні можливості, обсяги даних, тощо);

- характеристика вибраної системи (технології) управління базами даних, її переваги перед альтернативними варіантами.

4) Описання алгоритмів прийняття рішень застосованих в системі, яка розробляється:

- аналіз існуючих в інших системах чи стандартних алгоритмів прийняття рішень з висновками про можливість їх застосування в розробці;

- описання та обґрунтування алгоритмів прийняття рішень запропонованих для застосування в системі;

- довести* 'гор.авилі.їїасті запропонованого алгоритму та коректності результатів його виконання.

Практична реалізація.

Описання реалізації завдання.

Підрозділ має містити описання структури бази даних, бази знань (правил), механізмів логічного виведення й отримання чи підтримання процесів прийняття рішень, структури, складу, змісту та функцій розробленого програмного забезпечення (модулів, підпрограм, функцій, форм, макросів, запитів тощо) та процесів їхнього сумісного функціонування (15-20%).

Аналіз отриманих результатів.

Контрольний приклад має підтвердити працездатність розроблення та відповідність результатів функціонування системи поставленому завданню.

Спеціальна частина.

Розділ виконується за завданням консультанта та під його керівництвом.

Обґрунтування економічної ефективності.

Розділ виконується за завданням консультанта з економічних питань, під його керівництвом і має на меті підтвердження економічної ефективності виконання та впровадження розроблення.

Охорона праці та безпека в надзвичайних ситуаціях.

Розділ виконується за завданням консультантів та під їх керівництвом і визначає основні засади безпечної експлуатації розробленої магістрантом системи прийняття рішень.

Екологія.

Розділ виконується за завданням консультанта та під його керівництвом.

1.4.1 Зміст та вимоги до розділу «Аналітичний огляд літературних та інших джерел»

В аналітичному огляді літератури магістрант окреслює основні етапи розвитку на укової думки щодо вирішуваної проблеми. Стисло, критично висвітлюючи роботи по передників, магістрант повинен виділити ті питання, що залишились нерозкритими і, отже, визначити певне місце у вирішенні проблеми. Бажано закінчити цей розділ коротким резюме стосовно необхідності здійснення досліджень у цій галузі. Загальний обсяг огляду літератури не повинен перевищувати 20% обсягу основної частини магістерської роботи.

Підбір літератури треба починати відразу після обрання теми магістерської кваліфікаційної роботи.

Підбираючи літературу, треба насамперед використовувати сучасні інформаційні технології та мережі для пошуку інформації, а також звертатися до предметно-тематичних каталогів і бібліографічних довідників бібліотек. Вивчення літератури за обраною темою доцільно починати із загальних робіт, щоб отримати уявлення про основні питання, з якими межує обрана тема, а

потім вже шукати новий матеріал. При вивченні літератури бажано дотримуватися таких рекомендацій:

- починати з літератури, яка розкриває теоретичні аспекти питання, що вивчається, - монографій і журнальних статей. Після цього використовувати інструктивні матеріали (використовуються лише інструктивні матеріали останніх видань);

- детальне вивчення магістрантом літературних джерел полягає в їх конспектуванні та систематизації; характер конспектів визначається можливістю використання того чи іншого матеріалу в магістерській кваліфікаційній роботі (витяги, цитати, короткий виклад змісту літературного джерела або характеристика фактичного матеріалу; систематизацію отримуваної інформації здійснювати за основними розділами роботи, що передбачені планом);

- при вивченні літератури не прагніть засвоїти всю інформацію, а відбирайте лише ту, яка безпосередньо стосується теми роботи; критерієм оцінки прочитаного є можливість його практичного використання в роботі;

- вивчаючи літературні джерела, ретельно стежте за оформленням витягів, щоб надалі було легко ними користуватися;

- не турбуйтеся, якщо частина отриманих даних виявиться зайвою, дуже рідко всі вони використовуються повністю;

- прагніть орієнтуватися на останні дані з відповідної проблеми, спиратися на найавторитетніші джерела, точно вказувати, звідки взяті матеріали;

- при відборі фактів з літературних джерел їх потрібно критично осмислювати.

Тематична періодика.

З кожної зі значущих наукових дисциплін видаються відомі наукові часописи, які мають бути відомі магістранту. Існує практика, за якою в кожному номері зазвичай надається інформація про нові дисертації.

Крім того, в «Бюлетені ВАК» регулярно публікуються огляди дисертацій за 2-3 роки за різними спеціальностями - варто пошукати в бібліотеках і це видання (електронна версія «Бюлетеня ВАК» не містить останніх номерів).

Якщо відповідній науковій дисципліні присвячені значні Інтернет-сайти (розсилки новин, форуми і т.ін.), то можна пошукати зведення про дисертації й там. Проте інформацію на таких ресурсах не можна назвати всеохоплювальною, і завжди є небезпека, що найважливішу для магістранта дисертацію на сайті не згадали.

Електронні каталоги бібліотек.

Використання електронних каталогів бібліотек - найшвидший і продуктивний спосіб з'ясувати список дисертацій за спеціальністю, що цікавить. Чим більша бібліотека, електронним каталогом якої ви користуєтеся, тим точніші будуть результати. Тому варто зосередити свої пошукові зусилля, наприклад, на каталозі Російської державної бібліотеки (РДБ) - aleph.rsl.ru. Електронні каталоги РДБ охоплюють бібліографічні описи дисертацій (з 1995 року) і авторефератів дисертацій (з 1987 року). Тому, зайшовши за вказаною адресою, необхідно обрати в навігації розділ «Бази даних», у ньому - каталог авторефератів, а потім перейти в «Пошук». Далі в рядок пошуку треба ввести шифр спеціальності, наприклад, «051708» без крапок (і без лапок). У результаті ви отримаєте великий список (декілька сотень авторефератів) за всі роки, з яким можна ознайомитися.

Відомості про спеціальність у бібліографічних записах цього каталогу належать до заголовка, тому при складному пошуку вводити шифр треба в полі «заголовок». Щоб звузити перелік, наприклад, дізнатися про дисертації за певною спеціальністю за конкретний рік, треба перейти до «пошуку за поєднанням полів», де в полі «заголовок» ввести шифр, а в полі «рік» - необхідний рік. Отже, для пошуку дисертацій за спеціальністю, що цікавить, у бібліотеках треба проглядати «Літопис авторефератів» і тематичну періодику (включаючи «Бюлетень ВАК»). Якщо треба здійснити такий пошук в Інтернеті, необхідно використовувати каталог РДБ.

Інформацію про дисертаційні дослідження та роботи, які виконуються у провідних університетах світу, можна знайти на сайтах цих вузів. Можна також використати інформацію наукових часописів, що мають високі рейтинги, визначені Філадельфійським інститутом наукової інформації (181).

2 СИСТЕМНИЙ АНАЛІЗ ТА ОБҐРУНТУВАННЯ ПРОБЛЕМИ

Розділ «Системний аналіз та обґрунтування проблеми» складається з двох підрозділів. Підрозділ «Системний аналіз об'єкта дослідження та предметної області» є одним з основних, оскільки за його результатами надалі ставиться та розв'язується основне завдання магістерської кваліфікаційної роботи. Виконання підрозділу «Постановка та обґрунтування проблеми» дає змогу зрозуміти мету та суть поставленого завдання, визначити обсяг роботи, побудувати концептуальну модель системи та намітити можливі проблеми і шляхи до їх вирішення.

2.1 Системний аналіз об'єкту дослідження та предметної області

Системний аналіз (СА) спрямований на розв'язання складних проблем. Проблема виникає тоді, коли є розходження між бажаним та дійсним, тобто це абстрактна категорія, що відображає розуміння людьми мотивів своєї діяльності. Проблеми породжуються та розв'язуються людьми, а тому поняття «проблема» має людські риси сприйняття, що породжує такі труднощі:

- неясність розуміння проблеми;
- складність постановки проблем на віддалену перспективу;
- складність класифікації проблем і, як наслідок, вибір неадекватних засобів їх розв'язання;
- спотворена оцінка проблем (близькі, але дрібні проблеми затуляють великі, але віддалені);
- неправильна оцінка значимості проблем внаслідок вузькопрофесійної точки зору;
- змішування цілей, яких необхідно досягти, з засобами їх досягнення.

Метою застосування системного аналізу до конкретної проблеми є підвищення ступеня обґрунтованості рішення, що приймається, та подолання перелічених вище труднощів.

Для СА важливими є такі методологічні принципи:

- органічна єдність суб'єктивного та об'єктивного;
- структурність системи, що визначає цілісність та стійкість характеристик системи;
- динамізм системи;
- міждисциплінарний характер системних досліджень;

– органічна єдність формального та неформального при здійсненні СА.

На відміну від достатньо широкої системної методології системний аналіз обмежують дві такі особливості:

– системні аналітики вивчають лише штучно створені системи, в яких людині належить надзвичайно важлива, а в багатьох випадках і вирішальна роль;

– головна задача СА - прийняття рішень і управління.

Системний аналіз - це методологія дослідження таких властивостей та відношень в об'єктах, які важко спостерігаються та важко розуміються за допомогою представлення цих об'єктів у вигляді цілеспрямованих систем та вивчення властивостей цих систем та взаємних відношень як відношень між цілями та засобами їх реалізації.

Проблеми розрізняються за ступенем їх структурованості:

– добре структуровані та сформульовані кількісно;
– слабо структуровані, в яких зустрічаються як кількісні, так і якісні оцінки;

– неструктуровані, якісні проблеми.

Перший тип проблем не потребує СА, оскільки існує потужний апарат математичного моделювання та точні кількісні методи розв'язання. Основною областю застосування методів СА є слабо структуровані проблеми.

Потреба в СА виникає в тому випадку, коли виникають такі ситуації:

– для розв'язання нової проблеми за допомогою СА її формують, визначають, що і про що потрібно взяти і хто повинен знати;

– розв'язання проблеми передбачає координацію цілей з множиною засобів їх досягнення;

– проблема має розгалужені зв'язки, що викликають віддалені наслідки в різних галузях, і прийняття рішення в таких випадках потребує врахування сукупної ефективності та повних затрат;

– існують варіанти розв'язання проблеми або досягнення взаємопов'язаного комплексу цілей, які важко порівняти;

– створюються нові складні системи;

– здійснюється вдосконалення, реконструювання виробництва, необхідна реінженерія бізнес-процесів;

– для створення інформаційних систем та комп'ютеризованих систем керування;

– коли важливі рішення повинні прийматися за наявності невизначеності і ризику та (або) на достатньо віддалену перспективу.

Тому в розділі системного аналізу необхідно обґрунтувати потребу застосування системного аналізу, виходячи з тематики магістерської роботи.

Різноманітність задач, цілей дослідження, об'єктів дослідження природно приводить до існування різних методик СА, які ґрунтуються на єдиній методології. Це пояснюється тим, що існують певні об'єктивні закони людської діяльності, які виявляються при розв'язанні різних проблем. Ці закономірності, виявлені шляхом узагальнення досвіду та теоретичних досліджень, і становлять основу методології СА. Тому методики розв'язування системних задач, розроблені для різних конкретних випадків, схожі.

Якщо виникає проблема, треба взаємодіяти з особами, які залучені до побудови системи чи до оцінювання її функціонування. В інших випадках проблема полягає в тому, як визначити та модифікувати призначення системи, спроектувати систему, яка прагне до бажаного стану та пристосовується до оточення чи пристосовує оточення до себе, як врахувати різноманітні дії навколишнього середовища на систему. Ці аспекти є скоріше проблемами реконструювання, аніж «розв'язання проблем».

Чіткіше сформулювати вимоги до системи можна шляхом застосування та інтерпретації відповідно до її особливостей основних положень, або принципів системного підходу, які є досить загальними твердженнями, що узагальнюють досвід роботи людини зі складними системами. Це такі принципи [34, 105]:

- Принцип остаточної мети: глобальна мета системи має абсолютний пріоритет;
- Принцип єдності: сумісний розгляд системи і як цілого, і як сукупності компонент (елементів, підсистем, системотворчих відношень);
- Принцип зв'язності: довільна компонента системи розглядається сумісно з її зв'язками з оточенням;
- Принцип модульності: в системі доцільно в багатьох випадках реалізувати декомпозицію на складові (модулі) різного ступеня загальності та розглядати її як сукупність модулів та зв'язків між ними;
- Принцип ієрархії: в більшості випадків в системі доцільно реалізувати ієрархічну побудову та (або) впорядкування (можливий не повний порядок) її складових за важливістю;
- Принцип функціональності: структура системи та її функції повинні розглядатися сумісно з пріоритетом функції над структурою;
- Принцип розвитку: необхідно враховувати змінність системи, її здатність до розвитку, розширення, заміни складових, накопичення інформації;

– Принцип децентралізації: в управлінні системою співвідношення між централізацією та децентралізацією визначається призначенням та метою системи;

– Принцип невизначеності: невизначеності та випадковості повинні братися до уваги при визначенні стратегії та тактики розвитку системи.

Принципи системного підходу є загальними положеннями, що відображають абстраговані від конкретного змісту прикладних проблем відношення. Тому цілком доречним є запитання: «Як застосувати такі знання?» Для конкретної системи чи проблемної ситуації принципи системного підходу повинні бути конкретизовані, тобто насамперед необхідно відповісти на запитання: «Що означає той чи інший принцип в цій предметній області та в цій конкретній ситуації?»

У магістерській роботі необхідно наповнити принципи системного аналізу конкретним змістом, виходячи з тезаурусу предметної області дослідження. Це дає змогу у випадку складних систем краще побачити суттєві особливості проблеми, врахувати важливі взаємні зв'язки. У багатьох випадках інтерпретацією системних принципів в конкретних умовах можна піднятися на новий рівень розуміння системи загалом, вийти за межі розгляду її «зсередини». Така інтерпретація може приводити до висновків про відсутність умов для застосування деяких принципів або їх незначного впливу в конкретних умовах.

Системний аналіз ґрунтується на коректному використанні таких понять, як – система оточення, мета, функції та ін. У роботі необхідно, виходячи з загальних визначень основних понять системного аналізу, інтерпретувати їх з погляду досліджуваної проблеми – тобто визначити, що є системою, оточенням, елементами, функціями стосовно досліджуваної предметної області. Існують дві групи визначень системи.

Першу групу утворюють визначення, які не виділяють поняття її цілісності, наприклад, таке: «Система - це множина об'єктів разом з відношеннями між об'єктами та між їхніми атрибутами (властивостями)» [84]. Історія визначень такого типу зрозуміла і походить від природничих наук, в яких дослідник йшов шляхом від простого до складного - поділяв систему на елементи, розглядав властивості окремих частин і способи їх взаємодії, отримуючи так уявлення про систему як про сукупність взаємопов'язаних елементів.

Пізніше цей підхід використовувався в загальній теорії систем, для здійснення чіткої формалізації. Система визначалася як множина, на якій визначене задане відношення з фіксованими властивостями [84].

Друга група визначень передбачає цілісність як важливу властивість системи. Дійсно, якщо в результаті детального вивчення системи знайдена властивість, яку не можна поставити у відповідність жодному з її елементів, то визначення першої групи виявляється недійсним, і потрібно «довизначати» систему. В цьому сенсі система - це комплекс взаємопов'язаних елементів, що утворюють цілісність. Система утворює особливу єдність з середовищем та є елементом «надсистеми». Своєю чергою, її елементи системи можна розглядати як системи, якщо визначити інший критерій декомпозиції [1].

Середовище – це сукупність всіх об'єктів, зміна яких впливає на систему, а також об'єктів, – що змінюються під дією системи [84]. Весь наш світ можна розглядати як гігантську систему, але ми не досліджуємо Всесвіт практично кожного разу, коли виникає проблема. Тому певна система є підсистемою Всесвіту, а Всесвіт лише в найширшому сенсі можна називати середовищем цієї системи, а в абсолютній більшості середовище – це все те, що взаємодіє з системою, тобто теж певна підсистема Всесвіту.

Взаємні зв'язки між системою та зовнішнім середовищем, а також подальшу деталізацію доцільно реалізувати за допомогою структурних та (або) об'єктних нотацій. Так, в загальному взаємодія системи з елементами зовнішнього середовища з достатньою повнотою може бути відображеною в межах нотації DFD (діаграм потоків даних) за допомогою контекстної діаграми, а деталізація - шляхом побудови ієрархії DFD наступних рівнів, а також інших засобів стандартів IDEF,

Мета - це одна з найскладніших та разом з тим найдавніших категорій. Вона так чи інакше присутня в свідомості людини, яка здійснює довільний вид діяльності, і переноситься нею на багато природних та штучних систем. Пізнання мети допомагає зрозуміти сутність систем, що досліджуються.

Мета відображає призначення системи, яке не є детерміністично фіксованим; воно може розвиватися в часі і не обов'язково однаково. Призначення - це біхевіористичне поняття, і в соціальних системах поведінка людей є потужною силою, яка не завжди є достатньо зрозуміла, але яка є такою, що значною мірою формує наш світогляд. Існує певний різновид принципу невизначеності при спробі визначити систему, вивчаючи деякі прояви, що ототожнюються з системою. Тому ми вимушені модифікувати наші визначення залежно від того, як дослідження стають все конкретнішими.

Мета конкретизується за допомогою цілей. Одним зі способів розкриття внутрішньої структури мети є побудова дерева цілей. З метою аналізу мети функціонування системи в роботі необхідно побудувати дерево цілей системи (проблеми). Цілі в часовому аспекті поділяються на тактичні цілі (objectives), макроцілі (goals) та ідеали [84].

Тактичні цілі – це бажані результати, досягнення яких відбувається за визначений і порівняно короткий період часу. Макроцілі досягаються за довший час і вимагають для цього досягнення хоча б однієї тактичної цілі. Ідеали – це такі цілі, яких ніколи не досягають, але до яких система постійно наближається, реалізуючи деякі тактичні та макроцілі.

Категорія «мета» пройшла довгий шлях розвитку від найпростіших форм до складних структурно-функціональних представлень. Мета відображає те, що може чи повинно виникнути, прообраз майбутнього, стан, якого бажано досягти. Мета має декілька аспектів. Пізнавальний аспект мети відповідає прогнозу майбутнього, а конструктивний – можливим способам переходу до бажаного майбутнього чи плану дій. У тих випадках, коли мета відносно проста, усвідомлення мети передбачає і спосіб її досягнення, а у випадку складної мети – план набуває самостійного значення як елемент постановки мети. План встановлює послідовність етапів досягнення мети, визначаються засоби та методи, строки дій.

За наявністю інформації про способи досягнення цілей розрізняють такі класи цілей:

- Функціональна ціль - це ціль, спосіб досягнення якої відомий системі, що вже досягала цієї цілі. Функціональні цілі повторюються в часі та просторі. Прикладами такого типу цілей є результати виконання виробничих операцій, що періодично повторюються, стандартні функції управління та ін.

- Ціль-аналог - це образ, отриманий в результаті дії іншої системи, але який жодного разу не досягався системою, що розглядається, а якщо і досягався, то за інших умов зовнішнього середовища.

- Ціль розвитку, або нова ціль - це ціль, яка ніколи і ніким раніше не досягалася. Така ціль по суті пов'язана з утворенням нових систем.

Ці типи цілей пов'язані одна з іншою. Ціль розвитку за умови її успішного досягнення однією з систем перетворюється на ціль-аналог для всіх інших систем, а для цієї системи стає функціональною ціллю за умови незмінних зовнішніх умов та ціллю-аналогом за умови змінених зовнішніх умов.

При формуванні конкретних структур цілей доцільно використовувати такі два прийоми формування верхніх рівнів дерева цілей.

Перший ґрунтується на концепції про відповідність складових двох шкал розвитку системи: просторової (цілі власне системи; цілі, що визначаються взаємними відношеннями з найближчим середовищем; цілі, що ініціюються віддаленим середовищем) та шкали часу (поточні оперативні цілі, цілі найближчої перспективи, цілі віддаленої перспективи). Відповідність між цими шкалами відображено на рисунку 2.1.



Рисунок 2.1 – Відповідність шкал простору та часу

Підцілі верхніх рівнів дерева визначаються діагоналлю матриці, тобто залежно від конкретних умов є можливість обмежувати дерево за рахунок вилучення, наприклад, відразу всієї гілки підцілей віддаленої перспективи та віддаленого середовища, не змінюючи при цьому інших гілок дерева.

Другий прийом стосується випадку аналізу нової, неясної проблеми і полягає в тому, що складові верхнього рівня структури дерева визначаються відповідями на запитання (рисунок 2.2).

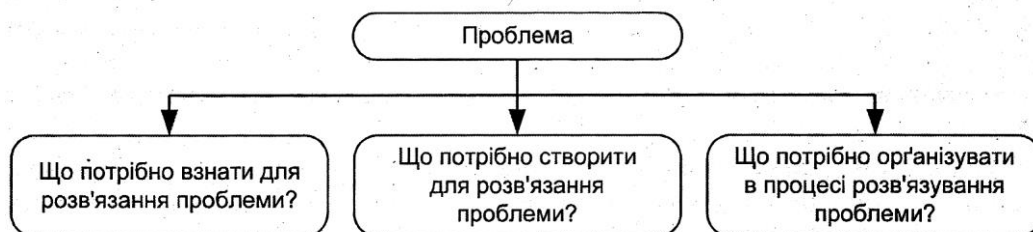


Рисунок 2.2 – Визначення складових дерева за допомогою запитань

Окрім того, доцільно формувати структуру дерева цілей шляхом уявлення його у вигляді піраміди з послідовним обходженням всіх її граней, повертаючись на новому витку до вже структурованих раніше гілок з врахуванням нового бачення проблеми.

Для побудови дерева цілей доцільно також використовувати загальні ознаки структурування для різних рівнів системи.

Принципи та ознаки структуризації, що пропонуються в різних методологіях, ґрунтуються на досвіді формування структур цілей. Розкриваючи одну й ту саму глобальну мету, різні аналітики можуть отримати різні ієрархічні структури, і це є природним, оскільки системний аналіз ґрунтується не лише на формальних процедурах та прийомах, але й на використанні інтуїції та досвіду фахівців.

Декомпозиція – це поділ системи на частини з метою зробити зручнішими певні операції з цією системою. Найважливішим стимулом і суттю

декомпозиції є спрощення системи, надміру складної для розгляду загалом. Таке спрощення може фактично приводити до заміни системи якоюсь іншою, в певному сенсі відповідною первинній – це здійснюється введенням гіпотез та послабленням або відкиданням деяких зв'язків та елементів в системі, або відповідати реальній системі і полегшувати роботу з нею – в цьому випадку (строга декомпозиція) необхідно розробляти спеціальні процедури узгодження та координації частин.

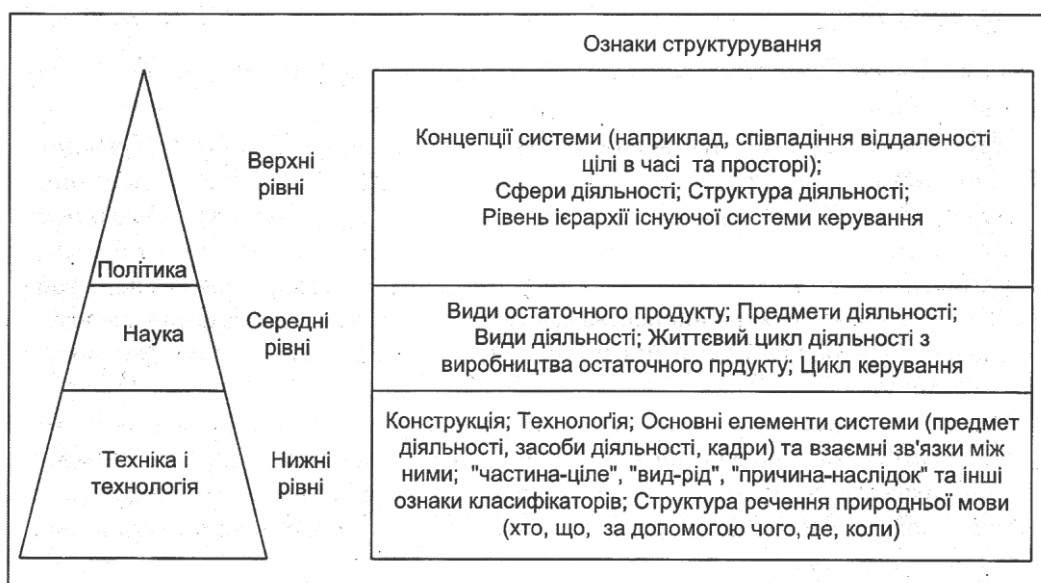


Рисунок 2.3 – Побудова дерева цілей за ознаками структурування

У роботі необхідно визначити ознаки, за якими реалізується декомпозиція проблеми, та які часткові проблеми отримуються, які з них розв'язуються в роботі, а які ні і чому.

Елементом називається деякий об'єкт (матеріальний, енергетичний, інформаційний), що має низку важливих властивостей, але внутрішня будова якого безвідносна до мети дослідження, тобто елемент не піддається подальшій декомпозиції при обраному рівні розгляду системи [143].

У роботі необхідно пояснити, що є елементами системи (проблеми), і чому вони не піддаються подальшій декомпозиції.

Функція системи – це все те, що виконує система або може виконувати згідно із своїм призначенням. Множина функцій системи є перетворенням призначення системи в дії. Функцію елемента зручно розглядати як сукупність її станів у просторі та часі. У разі взаємодії функцій доволі часто виникає нова властивість (властивості), які не виявляються в кожному окремому елементі системи. Наприклад, достатньо складно зрозуміти функцію окремої частини автомобіля ізольовано від нього, однак вона стає зрозумілою, коли ми бачимо її

у взаємодії з іншими, тобто коли автомобіль рухається. Одна й та сама функція може здійснюватися декількома шляхами.

У розділі системного аналізу необхідно визначити функції системи та розглянути взаємні зв'язки між ними.

Поняття структури є одним з основних в системному аналізі.

Структура - це множина частин або форм (елементів), які знаходяться у взаємодії та специфічному порядку необхідному для реалізації функцій. Тобто функція є первинною щодо структури. Властивістю структури є можливість існування протягом певного часу за допомогою зв'язуючого пристосування для збереження елементів (частин) та їх відношень приблизно в одній й тій самій послідовності, реагуючи при цьому на діє середовища.

Структура системи зберігається та збагачується через її функціональні трансформації, водночас структура полегшує ці перетворення. В організаціях та в ширшій соціальній структурі існують зв'язуючі сили, що підтримують форму структури. З практичного погляду представлення структури бажано спростити, щоб ідентифікувати її елементи та взаємні зв'язки між ними. Структура системи може бути охарактеризована за типами зв'язків, які в ній є або які в ній переважають. Найпростішими зв'язками є паралельне послідовне з'єднання та обернений зв'язок. Обернений зв'язок виконує регулювальну роль в системі.

Однією з найпростіших моделей структури є мережа. Мережева модель може використовуватися як для описання фактичних структурних зв'язків між елементами системи (транспортна мережа), або ж є абстрактним функціональним представленням взаємодій між підсистемами (ієрархічне представлення різних рівнів регулювання та управління в організації).

Часто оновлення структури є тимчасовою «тактичною» реакцією на дії зовнішнього середовища для того, щоб виграти час, необхідний для формування стратегічної реакції. Наприклад, фірми децентралізують з більшою ймовірністю свою структуру у випадку швидкозмінних зовнішніх умов як реакцію на них (сильна конкуренція, зміна економічних умов, поява на ринку нової продукції). За відносно статичних умов оточення більше вірогідно, що утвориться жорсткіша централізована ієрархічна структура. Структура оновлюється завжди для спрощення руху нових потоків, які виконують функції, що орієнтуються на нові цілі та макроцілі. Тому організація є самоорганізованою системою.

У процесі створення системи та планування її діяльності врешті-решт розглядаються компроміси між роботою великої, добре об'єднаної ієрархічної системи з доволі простим призначенням та багатоелементною багатоцільовою мережею достатньо малих систем з розподіленими і нечіткими взаємними зв'язками.

Ієрархія – це структура з підпорядкованістю, тобто з нерівноправними зв'язками – де в одному напрямку виявляють набагато більший вплив, аніж в оберненому. В більшості випадків прямий зв'язок – це керування і керуюча інформація, обернений – інформація про виконання та відхилення. На практиці розглядаються два основні типи ієрархічні структур – деревоподібна та ієрархічна.

Деревоподібна структура є найпростішою для аналізу та реалізації. В майже всіх випадках в ній виділяються ієрархічні рівні – групи елементів, що знаходяться на однаковій відстані (вимірюваної як кількість ребер) від головного елемента (кореня дерева.) Структури цього типу є надзвичайно поширеними (ієрархія проектування складної програмної системи, ієрархія цілей в складній організаційній системі, ієрархія за ознакою керованості процесів в живому організмі, ієрархія в зграї тварин).

Ромбоподібна структура приводить до множинної (частковий випадок - подвійної) підпорядкованості, належності елементів нижнього рівня. Приклади – участь одного технічного елемента в роботі більш ніж одного вузла, блока, використання одних і тих самих даних або результатів вимірювань в різних завданнях.

Будь-яка ієрархія звужує можливості та гнучкість системи. Елементи нижнього рівня обмежуються домінуванням верхнього рівня, вони здатні впливати на це домінування лише частково та зазвичай з певною затримкою. Однак введення ієрархії різко спрощує створення та функціонування системи. Недарма ієрархія того чи іншого ступеня існує практично у всіх складних природних системах. Негативні наслідки ієрархії багато в чому долаються шляхом зменшення жорсткості підпорядкування, можливістю самостійно реагувати на деякі дії без жорсткої регламентації згори.

Тобто структура є стійкими взаємними зв'язками між елементами системи, які забезпечують її цілісність. Якщо розглядати поняття «структура» у взаємному зв'язку з поняттям «мета», то під структурою треба розуміти спосіб досягнення мети [6].

У магістерській роботі необхідно визначити структуру конкретної системи (проблеми) та описати її.

Зв'язок (потік) - це важливий з погляду трактування системи обмін речовиною, енергією, інформацією між елементами та зовнішнім середовищем і елементами системи.

Функції системи реалізуються через потоки енергії, людей, матеріальні та інформаційні. Структуру можна розглядати також як множину обмежень на потоки в просторі та часі. Структура ініціює потоки, спрямовуючи їх вздовж певних шляхів (каналів), перетворює їх з певною затримкою в часі (час

перетворення), в певних випадках припускає регулювання та обернений зв'язок. Структура може змінюватися в часі самостійно, а також під впливом потоків, впливає на потоки і є системою в межах системи. Потоки, які є необхідними для збереження первісної структури, називаються підтримувальними, а ті, що є результатами дії системи та її структури – потоками продукції [34].

Комунікаційний потік в ієрархічній системі може бути закріплений жорстко за певними «каналами», межі яких не визначені в чітких фізичних термінах, однак вплив цих «каналів» може бути дуже реальним. Так, традиції організації, норми поведінки та неписані правила утворюють такі «канали», тобто організаційна структура накладає обмеження на комунікаційні потоки та сприяє ефективній роботі системи. В кожній структурі існує певна ієрархія потоків (потоки між елементами, підсистемами, системою та зовнішнім середовищем).

Інформаційні потоки в окладних штучних системах мають особливе значення:

- по-перше, інформаційні потоки та інформаційні зв'язки в багатьох випадках є домінуючими, визначальними в системі;
- по-друге, вони зазвичай супроводжують і інші – матеріальні, енергетичні та людські дії цих потоків фіксуються і у вигляді інформації.

Інформація в системі вивчається як з погляду її отримання, зберігання, передавання, перетворення, фільтрації, так і з погляду її вимірювання. Інформаційним потокам відповідають певні структурні схеми (наприклад, діаграми потоків даних та ін.), які мають спільні риси: вказані джерела та споживачі інформації, обсяг, форми представлення, напрямки передавання, місця і вид зберігання та ін. Ці структурні схеми (інформаційні моделі системи) використовуються для аналізу та мінімізації потоків даних та зменшення їхнього обсягу, виявлення як дублювання інформації, так і дублювання шляхів її передавання та ін. Поняття інформації має високий ступінь універсальності, і в загальному сенсі функціонування системи можна розглядати як перетворення вхідної інформації у вихідну шляхом прийняття певних рішень в системі.

Стан системи – це зафіксовані значення характеристик системи, важливі для цілей дослідження. Зміна будь-якої з цих характеристик означатиме перехід системи до іншого стану. Тобто отримуємо набір станів, який ще не є процесом.

Процес – це набір станів системи, що відповідає впорядкованій неперервній або дискретній зміні деякого параметра, що визначає характеристики чи властивості системи. В більшості випадків таким параметром є час.

Процес зміни станів системи в часі відображає динаміку системи.

Процеси в системі мають різноманітне значення. Так, процес проектування інформаційної системи як рух від системного аналізу через декілька проміжних етапів (технічне завдання, технічне та робоче проектування, впровадження, супровід) є основною функцією системи – розробницької фірми. У цьому випадку необхідно враховувати також і внутрішні процеси. Тобто процеси описуються як залежності виходів від входів в модулях різного ступеня узагальнення або різного рівня ієрархії. При цьому принципово не важливо, сприяє чи перешкоджає загалом той чи інший процес реалізації системою своїх функцій.

Треба розрізняти складність та «важкість». Проблема чи річ може бути «важкою» до розв’язання, але це ще не обов’язково означатиме, що вона буде складною (головоломки переважно є важкими до розв’язання, але не складними). Різниця між цими поняттями полягає в тому, що складна проблема зазвичай має велику кількість розв’язань, і ці розв’язання мають багато призначень, що не в останню чергу викликане слабкою структурованістю та різноманітністю складних проблем та мети їх розв’язання.

Складність є взаємодією та взаємною залежністю, причому взаємні залежності складових системи є симетричними зі змінною інтенсивністю (економіка залежить від енергетики та енергоресурсів і, навпаки, наявність енергоресурсів залежить від політики і навпаки; політика залежить від сили і навпаки, а сила, своєю чергою, залежить від військової могутності та економічної стабільності). Для складних об’єктів властива множинність системних образів, тобто один і той самий об’єкт допускає різні системні моделі, що відповідають різним його сторонам та різному ступеню глибини проникнення в об’єкт [63].

У розділі системного аналізу необхідно вказати, до якого типу належить проблема – простих чи складних, – і подати відповідне обґрунтування.

Одним з найважливіших методів отримання нових знань про досліджувану систему є системне моделювання. Модель (від лат. *modulus* - міра) - це заміщувач об’єкта дослідження, що знаходиться з ним у такій відповідності, яка дає змогу отримати нове знання про цей об’єкт.

Оскільки модель ґрунтується на аналогії, то вона губить сенс засобу пізнання як у випадку тотожності моделі та об’єкта дослідження, так і у випадку дуже великих відмінностей між ними. Тобто моделювання пов’язане зі спрощенням, огрубленням прототипу, абстрагуванням від низки його властивостей, ознак, сторін. Надмірно спрощена модель, проте, може привести до невідповідності з досліджуваним об’єктом, що унеможливить дослідження його за допомогою такої моделі. З іншого боку, врахування в моделі якомога

більшої кількості властивостей досліджуваного об'єкта ускладнює процес дослідження.

Будь-яка модель охоплює суттєве в певному сенсі, тобто завжди представляє об'єкт лише щодо деяких його властивостей. Окремі властивості об'єкта вивчають, відмовляючись від дослідження інших його властивостей. Водночас багато властивостей, від яких доводиться абстрагуватись під час моделювання, ще невідомі, і врешті-решт не виключено, що саме вони можуть виявитися важливими, тобто модель буде некоректною. Говорять, що модель відповідає (адекватна) об'єкту дослідження, якщо результати моделювання є основою для прогнозування процесів в реальному об'єкті, що досліджується.

Моделювання – це метод опосередкованого пізнання за допомогою штучних або природних систем, які зберігають деякі особливості об'єкта дослідження, тобто заміщають його, що дає можливість отримати нове знання про об'єкт-оригінал. У системному аналізі моделі є дуже важливим компонентом дослідження та проектування нової системи, і зазвичай використовується множина моделей для забезпечення якісного дослідження системи.

Бажано вказати, які типи моделей передбачається розглянути в подальших розділах та чому.

2.1.1 Особливості імітаційних моделей прийняття рішень

Імітаційні моделі складних систем є найпоширенішими внаслідок своєї універсальності, можливості проведення експериментів, планування різноманітних змін. У процесі проведення експериментів на імітаційній моделі можна вносити такі зміни:

- в структурі моделі (ввести нові елементи та зв'язки, вилучити інші);
- моделей поведінки, параметрів моделей;
- параметрів та законів розподілу випадкових факторів;
- значень та зміни в часі зовнішніх (екзогенних) змінних.

Ці змінні використовуються для вибору варіантів під час оптимізації системи. Імітаційна модель - це, з одного боку, зовнішнє описання системи, що діє за принципом «чорної скрині», і в цій якості використовується для проведення експериментів. З іншого боку, елементи імітаційних моделей - це внутрішні описання систем, що можуть функціонувати за принципом аксіоматичних моделей.

Але треба пам'ятати, що імітаційні моделі створюються на зовсім іншому концептуальному ґрунті - вони не продукт чи об'єкт якоїсь математичної теорії або «чорна скриня» - вони ґрунтуються на сукупних знаннях експертів з цієї проблеми.

У процесі опрацювання концептуальної схеми моделі системи як її елементи використовують моделі інших класів. Моделі компонент описують локальні механізми поведінки, мають конкретну природу та в багатьох випадках можуть бути отримані у межах відповідних конкретних дисциплін.

Структура імітаційної моделі відображає систему уявлень експертів про проблему загалом та методи її дослідження, призначення моделі та її цінність, функціонально-цільові причинно-наслідкові зв'язки між елементами та компонентами системи, відношення системи до зовнішнього середовища.

Імітаційні моделі дають змогу дослідити загальносистемні властивості, поведінку системи в особливих ситуаціях, знайти кращі значення параметрів системи, які до початку дослідження були вільними, прогнозувати поведінку системи в часі. Алгоритмічна структура імітаційних моделей сприяє реалізації різноманітних схем ієрархічного підпорядкування та координації між елементами моделі.

Тобто імітаційні моделі дають змогу дослідити поведінку великих систем, які ш можливо досліджувати за допомогою інших методів.

Разом з тим не можна не усвідомлювати недосконалості імітаційних моделей. По перше, імітаційні моделі будуються на основі знань дослідника, який має певну мету тобто існує елемент суб'єктивності як в процесі побудови моделі, так і під час проведення експериментів на ній. Крім того, обов'язковим етапом є інтерпретація отриманих результатів, тому що імітаційні моделі не є моделями «пояснювального» типу.

Аналітичний метод має величезне наукове та практичне значення (розкладення функцій в ряди, диференціальне та інтегральне числення - в математиці; аналізатори спектрів дослідження атомів — у фізиці; анатомія - в медицині; конвеєрна технологія - у виробництві). Однак дуже важливою є цілісність системи, що особливо підкреслювалося Р. Акофом. Цілісність при аналізі порушується, при розчленуванні системи втрачаються не лише важливі властивості системи («розібраний автомобіль не поїде»), але й зникають і істотні властивості частин системи («відділеним кермом автомобілем не керують, відділене око не бачить»). Тому за Акофом результатом аналізу є лише відкриття структури знання того, як система працює, а не розуміння того, чому і навіщо вона це робить [4].

«Синтетичне мислення вимагає пояснення поведінки системи. Воно істотно відрізняється від аналізу. На першому кроці аналізу річ, що підлягає поясненню, розділяється на частини; в синтетичному мисленні вона повинна розглядатися як частина великого цілого. На другому кроці аналізу пояснюються складові частини; в синтетичному мисленні пояснюється ціле, до складу якого входить річ. На останньому кроці аналізу знання про частини

агрегується в знання про ціле; в синтетичному мисленні розуміння цілого, не має в своєму складі річ, дезінтегрується для пояснення частин. Цього досягають відкриттям їх ролей та функцій загалом. Синтетичне мислення відкриває не структуру, функцію; воно відкриває, чому система працює так, а не те, як вона це робить» [3, 4].

Аналітичним методом можна досягти найвищих результатів за умови, що ціле вдається розділити на незалежні одна від однієї частини, оскільки розглядаючи їх окремо, можна правильно визначити їхній внесок для досягнення загального ефекту. Одні випадки, коли система є сумою складових, зустрічаються надзвичайно рідко. Зазвичай внесок частини до загальносистемного ефекту залежить і від внесків інших складових. Внаслідок цього автономно частина може функціонувати якнайкраще, але загалом ефект функціонування буде далеко не найвищим.

Отже, під час аналізу «неадитивних» систем важливо досліджувати не окремі складові, а їхню взаємодію, що є значно складнішим завданням. Ідеалом, остаточною метою аналітичного методу є встановлення причинно-наслідкових зв'язків між явищами. Дещо вважається пізнаним і повністю зрозумілим лише тоді, якщо відома його причина (сукупність умов, необхідних та достатніх для реалізації наслідку). Однак така ситуація в багатьох випадках є недосяжною, і навіть в тих випадках, коли є причинно-наслідковий опис, все інше необхідно вилучити. Для причинно-наслідкового опису не існує поняття навколишнього середовища, оскільки для наслідку нічого, окрім причини, не потрібно (приклад, закон всесвітнього тяжіння, справедливий тоді, коли відсутні всі сили, окрім сили тяжіння).

У випадку складних систем вилучення на перший погляд «непотрібних» чи «нецікавих» взаємодій може істотно порушити адекватність моделі і є в багатьох випадках принципово неможливим. Для описання таких ситуацій використовується дві можливості: або відобразити «безпричинну» компоненту через «об'єктивну випадковість» чи «суб'єктивну невизначеність» (чи їх комбінацію), або виходити з того, що відношення «причина-наслідок» є не єдино можливим способом для описання взаємодії. В останньому випадку застосовується модель «продуцент - продукт» за якою для отримання продукту продуцент є необхідною, але не достатньою умовою (для дуба жолудь є продуцентом, оскільки окрім нього, потрібні й інші умови, щоби виріс дуб). Тобто у відношенні «продуцент-продукт» для отримання продукту необхідні й інші умови, окрім продуцента, які й утворюють навколишнє середовище. Причинно-наслідковий зв'язок у цьому випадку є граничним ідеальним випадком зв'язку «продуцент-продукт», до якого можна наблизитися, але досягти якого не завжди можливо.

Аналіз і синтез доповнюють, але не замінюють один одного. Системне мислення об'єднує обидва ці методи.

Із застосуванням як аналітичного, так і синтетичного підходів отримують обов'язкові кроки, за якими необхідно розкласти ціле на складові (здійснити декомпозицію) або об'єднати складові в ціле (здійснити агрегування).

Для моделей інформаційних систем характерні такі особливості.

1. Двоїстість природи мети. З одного боку, це мета функціонування власне системи, що моделюється, яка в багатьох випадках описується у вигляді множини критеріїв, що відображають різні її аспекти. З іншого боку, це мета створення моделі (одноразове використання, періодичне використання, використання в контурі управління), що безпосередньо впливає на критерії оцінки адекватності моделі, точності та достовірності модельних результатів.

2. Складність моделі, яку дуже приблизно можна оцінити на основі загальної кількості елементів певних типів та взаємозв'язків між ними. Крім того, різноманітність елементів та зв'язків між ними для забезпечення виконання одного з основних кібернетичних принципів – принципу необхідної різноманітності – приводить до побудови моделі у вигляді ієрархічної структури.

3. Невизначеність інформаційної системи виявляється і в моделі. Оскільки в багатьох випадках ця невизначеність не має характеру стохастичності, необхідно вводити до таких моделей підсистеми експертного типу, елементи штучного інтелекту та передбачати можливість маніпулювання з нечіткими множинами. Так, для прогнозування курсу валют з успіхом використовуються моделі перцептронного типу.

4. Особливістю моделей, які працюють в контурі управління, є адаптованість, тобто здатність цілеспрямовано функціонувати в умовах нестационарного середовища, що досягається реалізацією вищенаведеного системного принципу адаптації.

Характерною особливістю правильно побудованої моделі є те, що вона відбиває лише ті аспекти реальної системи, які цікавлять дослідника, тобто є в певному сенсі мінімальною.

Системний аналіз необхідно реалізовувати за певною послідовністю кроків, що визначається тією чи іншою методологією.

Методологія – це базове начало системного аналізу. Вона містить визначення використовуваних понять, принципи системного підходу, а також постановку та загальну характеристику основних проблем організації системних досліджень. Визначення в методології подаються зазвичай на словесно-інтуїтивному рівні і мають властивість конструктивності.

Методологія реалізується з використанням структурного чи об'єктного підходу. У процесі СА доцільно використовувати відповідні CASE-засоби, як, наприклад, All Fusion Suite чи Rational Rose.

Більшість CASE-засобів проектування інформаційних систем ґрунтується на парадигмі методологія-метод-нотація-засіб.

Методологія визначає керівні положення для оцінки та обрання проекту ІС, що розробляється, кроки проектування, їх послідовність, правила розподілу та призначення методів.

Метод – це систематична процедура або техніка генерації описань компонентів ІС (наприклад, проектування потоків та структур даних).

Нотації призначені для опису структури системи, елементів даних, етапів опрацювання та містять графи, діаграми, таблиці, блок-схеми, формальні та природні мови.

Засоби – це інструментарій для підтримки та посилення методів. Ці інструменти підтримують роботу користувачів в процесі створення та редагування Графічного проект} в інтерактивному режимі, вони сприяють організації проекту в вигляді ієрархії рівної абстракції, реалізують перевірки компонентів на відповідність.

Методологія для побудови та розв'язання системних проблем сама повинна бути системною, а саме [123]:

- примушувати осіб, що приймають рішення (ОПР), пояснювати взаємодію елементів в системі, розуміючи й те, що деякі елементи можуть діяти незалежно від інших;

- виявляти та пояснювати тенденції до більшої спеціалізації та зменшення зв'язності елементів системи;

- ідентифікувати та впорядковувати домінуючі елементи перед описанням системи як єдиного цілого, орієнтувати на використання обмежених ресурсів насамперед для управління домінуючими елементами;

- при збиранні релевантної інформації про систему використовувати творчі здібності ОПР для визначення бажаного призначення та структури системи, ідентифікації складових частин та формування альтернативних стратегій втручання;

- орієнтувати на остаточний результат, полегшуючи порівняння альтернатив з метою вибору найприйнятнішої;

- містити механізм оберненого зв'язку з метою аналізу від'ємної ентропії, еволюції та стійкості; як відкрита система методологія повинна використовувати інформацію зовнішнього середовища для перевірки правильності управління системою та модифікації її призначення або давати змогу імітувати реакції зовнішнього середовища.

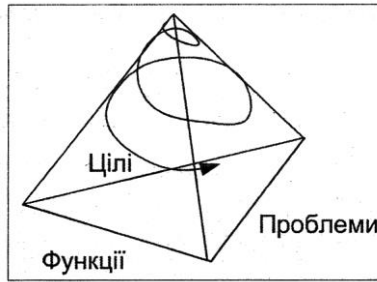


Рисунок 2.4 – Характер руху у разі структуризації в системному дослідженні

Незалежно від конкретних етапів та підетапів системного дослідження існують основні структурні елементи, основна послідовність яких «мета - способи досягнення мети - ресурси» (рисунок 2.4). Мета зазвичай структурується у вигляді дерева (мультидерева) цілей, і в більшості випадків в процесі структуризації реалізується спіральний рух гранями піраміди «цілі - функції - проблеми». У загальному випадку для побудови системи та керування нею необхідна реалізація таких етапів:

- визначення проблеми;
- ідентифікація призначення системи;
- ідентифікація змінних та взаємних зв'язків між ними;
- ідентифікація функцій та структури системи;
- ідентифікація оточення (середовища існування) системи;
- генерація та визначення пріоритетів альтернативних потоків;
- оцінювання ресурсів, необхідних для реалізації кожного з можливих варіантів;
- визначення наявності ресурсів;
- оцінка ефективності варіантів та вибір прийнятної альтернативи, яка може бути реалізована з врахуванням існуючих обмежень;
- реалізація (впровадження) обраної альтернативи та корегування плану.

На ґрунті основних наведених понять та термінів, що використовуються при дослідженні складних систем, будуються загальні методики використання цих понять у вигляді методологій системного дослідження, що, своєю чергою, містять певну послідовність кроків. Ця абстрактна схема є деякою послідовністю орієнтуючих дій, тобто дослідження тієї чи іншої конкретної системи може відрізнитися певним чином від наведеної послідовності кроків. Різні автори схильні по-різному визначати послідовність та назви етапів системного дослідження, але за цим для всіх них характерним є єдність поглядів та принципова єдність підходів до поділу СА на етапи [143].

Розглянемо детальніше дві методології (хоча в роботі не забороняється використовувати й інші) – Ю. І. Черняка та методологію, орієнтовану на дослідження існуючих систем та виявлення проблем.

Згідно з методологією Ю. І. Черняка необхідно діяти у такій послідовності кроків (таблиця 2.1).

Таблиця 2.1. Послідовність кроків за методологією Ю.І. Черняка

I. Аналіз проблеми
Чи існує проблема? Точне формулювання проблеми; Аналіз логічної структури проблеми; Розвиток проблеми (минуле і майбутнє); Зовнішні зв'язки проблеми (з іншими проблемами); Принципова розв'язальність проблеми
II. Визначення системи
Специфікація задачі; Визначення позиції спостерігача; Визначення об'єкта; Виділення елементів (визначення критеріїв декомпозиції); Визначення підсистем; Визначення середовища
III. Аналіз структури системи
Визначення рівнів ієрархії; Визначення аспектів та мов; Визначення процесів та функцій; Визначення та специфікація процесів управління та інформаційних каналів; Специфікація підсистем; Специфікація процесів та функцій (рутинних та цільових)
IV. Формулювання мети системи
Визначення цілей та вимог надсистеми; Визначення цілей та обмежень середовища; Формулювання спільної мети; Визначення критеріїв; Декомпозиція цілей та критеріїв за підсистемами; Композиція загального критерію з критеріїв підсистем
V. Декомпозиція мети, виявлення потреби в ресурсах та процесах
Формулювання цілей верхнього рівня; Формулювання цілей поточних процесів; Формулювання цілей ефективності; Формулювання цілей розвитку; Формулювання зовнішніх цілей та обмежень; Виявлення потреб в ресурсах та процесах
VI. Виявлення ресурсів та процесів, композиція цілей
Оцінювання існуючої технології та потужностей; Оцінювання поточного стану ресурсів; Оцінювання поточних та запланованих проектів; Оцінка можливості взаємодії з іншими системами; Оцінювання соціальних факторів; Композиція цілей
VII. Прогноз та аналіз майбутніх умов
Аналіз стійких тенденцій розвитку системи; Прогноз розвитку та змін середовища; Передбачення появи нових факторів, які сильно впливають на розвиток системи; Аналіз ресурсів майбутнього; Комплексний аналіз взаємодії факторів майбутнього розвитку; Аналіз можливих зсувів цілей та критеріїв
VIII. Оцінювання цілей та засобів

Розрахунок оцінок за критеріями; Оцінювання взаємної залежності цілей; Оцінювання відносної важливості цілей; Оцінювання дефіцитності та вартості ресурсів; Оцінювання впливу зовнішніх факторів; Оцінювання комплексних розрахункових показників
IX. Відбір варіантів
Аналіз цілей на сумісність та входження; Перевірка цілей на повноту; Відкидання надлишкових цілей; Планування варіантів досягнення окремих цілей; Оцінка та порівняння варіантів; Суміщення комплексу взаємопов'язаних варіантів
X. Діагноз існуючої системи
Моделювання технологічного та економічного процесу; Розрахунок потенційної та фактичної потужності; Аналіз втрат потужності; Виявлення недоліків в організації виробництва та керування; Виявлення та аналіз заходів з вдосконалення організації
XI. Побудова комплексної програми розвитку
Формулювання заходів, проектів та програм; Визначення черговості цілей та заходів з їх досягнення; Розподіл сфер діяльності; Розподіл сфер компетенції; Розроблення комплексного плану заходів за умов дії обмежень в часі та за ресурсами; Розподіл за відповідальними організаціями, керівниками та виконавцями
XII. Проектування організації для досягнення мети
Призначення Цілей організації; Формулювання функції організації; Проектування організаційної структури; Проектування інформаційних механізмів; Проектування режимів роботи; Проектування механізмів матеріального та морального стимулювання

Варіант методології СА, запропонований Ю. І. Черняком, є певною мірою синтетичним і враховує всі етапи інших класифікацій, побудований за принципом логічної послідовності етапів системного дослідження, які пов'язані з основними положеннями теорії систем, по суті максимальною класифікацією. Розглянемо коротко особливості, укрупнених етапів.

Питання про те, чи існує взагалі проблема, є надзвичайно важливим, оскільки застосування величезних зусиль до розв'язання неіснуючих проблем є доволі звичним випадком. Це пояснюється тим, що існують проблеми-привиди, які можуть маскувати інші, дійсно важливі та актуальні проблеми. Окрім того, правильне та достатньо точне формулювання проблеми є першим та необхідним етапом будь-якого системного дослідження.

Для побудови системи проблему необхідно піддати декомпозиції, тобто розкласти на комплекс чітко сформульованих завдань. Позиція спостерігача (аналітика) в принципі визначає критерій розв'язання проблеми. Визначення об'єкта в деяких випадках може становити основну складність дослідження.

Довільність в процесі виділення підсистем та процесів, що реалізуються в системі, може привести до того, що системне дослідження буде невдалим. У складних системах з участю людей структурні співвідношення не лише не є очевидними, але й заховані за відношеннями адміністративної підпорядкованості. Рутинні цілі та процеси у багатьох випадках затіняють цілі та процеси розвитку, а тому потрібно їх розрізняти.

Сформулювати мету організації неможливо за допомогою вивчення «суспільної думки», це складна логічна процедура, що реалізується в межах системного аналізу та теорії систем і вимагає знання досліджуваної предметної області.

У складних системах мета у більшості випадків є віддаленою від конкретних засобів її досягнення, і «наближення» реалізується узгодженням мети з засобами її досягнення шляхом декомпозиції цілей. Це є однією з найважливіших складових системного аналізу і реалізується за допомогою методу дерева цілей.

СА застосовується насамперед до розв'язання стратегічних проблем, а тому повинен враховувати майбутнє. Майбутні відкриття, ресурси, система майбутніх соціальних цінностей радикально можуть вплинути на функціонування системи, а тому прогнозування є однією з найважливіших та найскладніших складових СА.

Велика кількість діючих в СА факторів (соціальні, політичні, моральні, етичні та ін.) не піддається кількісному вимірюванню, а враховується експертним шляхом. Оскільки СА переважно оперує слабо структурованими чи неструктурованими даними, отримання та опрацювання експертних оцінок є необхідним етапом СА.

Корегування цілей з метою виявлення основних дає змогу звужити множину варіантів та привести їх у відповідність з наявними ресурсними обмеженнями.

У більшості випадків СА скерований на покращення функціонування існуючих систем, а тому виявлення важливих проблем управління та тактичних цілей є предметом діагностичного дослідження та аналізу системи керування.

Формування комплексних програм за цією методологією має на меті «переклад» результатів СА мовою конкретних заходів, що повинні бути реалізовані для досягнення поставленої мети. У певних випадках може виявитися необхідність проектування додаткових структур управління або перепроєктування існуючих.

Методологія системного дослідження, орієнтована насамперед на дослідження існуючих систем та виявлення проблем, передбачає такі кроки [34].

Крок I. Формування загальних уявлень про систему.

1) Виявлення призначення, мети, головних цілей, функцій, властивостей системи. Формування (вибір) основних предметних понять, що використовуються в системі.

2) Виявлення основних складових (модулів) системи та їх функцій; зрозуміння єдності цих складових у межах системи.

3) Виявлення основних процесів у системі, їхнього значення, умов перебігу, етапності, стрибків, змін стану та інших особливостей у функціонуванні системи, виокремлення основних керуючих факторів.

4) Виявлення основних елементів оточення системи (не-системи), з якими пов'язана система, що вивчається, характеру зв'язків системи з елементами оточення.

5) Виявлення невизначеностей та випадковостей в ситуаціях визначального впливу їх на систему.

Крок II. Формування поглиблених уявлень про систему.

6) Виявлення розгалуженої структури, ієрархії, формування уявлень про систему як про сукупність модулів, пов'язаних входами-виходами.

7) Виявлення всіх елементів та зв'язків, важливих для цілей розгляду, їх співвіднесення до ієрархії системи, ранжування елементів та зв'язків за важливістю.

8) Врахування змін та невизначеностей в системі.

9) Дослідження функцій та процесів в системі з метою управління ними. Формування управлінь та процедур прийняття рішень. Формування системи управління на основі окремих керуючих дій.

Крок III. Моделювання системи як етап її дослідження.

10) Побудова сукупності моделей для описання системи.

Крок IV. Супровід системи.

11) Накопичення досвіду роботи з системою та її моделлю, уточнення інформації про систему, вдосконалення моделей.

12) Оцінка граничних можливостей системи, дослідження відмов, виходів з ладу, відхилень від норми.

13) Розширення функцій (властивостей) системи, зміна вимог до неї, нове коло задач, нові умови роботи, введення системи елементом до системи вищого рівня.

2.1.2 Особливості створення нової системи

Ця ж методологія може застосовуватися і в процесі створення нової системи. Послідовність етапів залишається такою самою, але змінюється їх спрямованість - по суті вони будуть етапами «попереднього проектування», після чого власне виконується етап створення нової системи. Особливості

процесів проектування та перепроєктування інформаційних систем розглядаються в наступному розділі.

До формулювання проблеми залучається декілька ОНР, альтернативи та переваги яких можуть бути множинними та конфліктними, а ймовірність того, що певна альтернатива приведе до певного результату, або невідома, або невизначена. Основна властивість слабо структурованої проблеми не виявляється з переліку фактів чи цифр і не виникає з переваг декількох ОНР, а тому визначення та структуризування проблеми полягає не лише в униканні неправильних рішень, але й у формулюванні правильних запитань, щоб розв'язати власне потрібну проблему. В СА існує декілька методів, які використовуються при структуризації проблем, як, наприклад, метод аналізу ієрархій.

При реалізації розділу системного аналізу потрібно використовувати методи та інструменти системного аналізу.

Це дає змогу проаналізувати шляхи вирішення проблеми (чи альтернативи побудови системи) в загальному і обґрунтувати вибір однієї з них для подальшого детального розгляду.

При цьому можуть використовуватися такі методи, як метод аналітичної ієрархії (МАІ), функціонального аналізу та його різновидів, формування експертних висновків, комбінаторно-морфологічний аналіз та ін.

Одним з найпоширеніших є МАІ, використання якого рекомендується в розділі системного аналізу магістерської роботи, хоча за потреби можуть використовуватися й інші з відповідним обґрунтуванням.

Метод аналізу ієрархій (МАІ) - це систематична процедура, що ґрунтується на ієрархічному поданні елементів, які визначають суть проблеми [5, 123]. Проблема піддається декомпозиції на простіші складові з оцінкою децидента, відносного ступеня взаємодії елементів отримуваної ієрархічної структури. В методі використовуються жорсткі оцінки в шкалі відношень. МАІ будується на принципі ідентичності та декомпозиції і містить процедури синтезу множинних тверджень, отримання пріоритетності критеріїв та знаходження альтернативних рішень.

Принцип ідентичності та декомпозиції передбачає структуризування проблем у вигляді ієрархії або мережі як першого етапу МАІ.

Можна виділити ряд модифікацій МАІ, що визначаються характером зв'язків між критеріями й альтернативами, розташованими на найнижчому рівні ієрархії, а також методом порівняння альтернатив.

За характером зв'язків між критеріями й альтернативами визначається два типи ієрархій. До першого типу належать такі, у яких кожен критерій, що має зв'язок з альтернативами, зв'язаний із усіма альтернативами, що

розглядаються (тип ієрархій з однаковими числом і функціональним складом альтернатив під критеріями). До другого типу ієрархій належать такі, у яких кожен критерій, що має зв'язок з альтернативами, зв'язаний не з усіма альтернативами, що розглядаються (тип ієрархій з різними числом і функціональним складом альтернатив під критеріями).

У МАІ є три методи порівняння альтернатив: попарне порівняння; порівняння альтернатив щодо стандартів і порівняння альтернатив копіюванням.

Закінчується розділ СА висновками, в яких лаконічно викладаються первісне завдання та отримані результати.

2.2 Постановка та обґрунтування проблеми

Розділ «Постановка та обґрунтування проблеми» є одним із найважливіших у магістерській роботі. Виконання цього підрозділу дає змогу зрозуміти мету та суть поставленого завдання, визначити обсяг роботи, побудувати концептуальну модель системи та окреслити можливі проблеми і шляхи їхнього вирішення. При виконанні цього розділу необхідно керуватися вимогами ГОСТ 24.204-80. Требования к содержанию документа «Описание постановки задачи» (додаток Н).

Так, при визначенні мети та обсягу роботи важливо не ставити перед собою невирішальних завдань або таких, що вимагають багаторічних досліджень і які не можна виконати протягом часу, відведеного на виконання магістерської роботи. У випадку, якщо тематику магістерської роботи планується розвивати у подальших дослідженнях, необхідно, крім загальної проблеми, також чітко визначити завдання, поставлені для магістерської роботи, та вказати, яку частину більш загальної проблеми вони вирішують.

Побудована в результаті виконання цього розділу концептуальна модель системи дає змогу краще зрозуміти поставлене завдання, окреслити можливі шляхи його вирішення. У результаті обговорення концептуальної моделі з керівником роботи та студентами на семінарах можна вчасно отримати критичні зауваження та поради, що покращить якість роботи. Загалом помилки, допущені під час концептуального проектування, є найвагомішими. Серйозні прорахунки на цьому етапі призводять до необхідності відкинути вже зроблене та розпочати роботу наново.

У розділі «Постановка та обґрунтування проблеми» магістерської роботи розробляють такі питання:

- мета розроблення;
- призначення системи;

- місце застосування системи;
- обґрунтування розроблення та впровадження системи;
- очікувані ефекти від впровадження системи;
- розроблення концептуальної моделі системи;
- опис вхідних даних;
- опис вихідних даних;
- опис функцій та структури системи;
- опис вимог до системи;
- додаткові формальні або узагальнені моделі системи.

Мета розроблення.

У підрозділі «Мета розроблення» визначають основні результати, які мають бути отримані в процесі виконання роботи. Результатом роботи є дієздатна, готова до застосування інтелектуальна система (підсистема) прийняття рішень, експертна система, інша система, що функціонує із застосуванням методів та засобів штучного інтелекту та застосовується у певній галузі. Якщо у магістерській роботі розробляють нові алгоритми, методи, підходи та окремі вирішення, то необхідно вказати, в чому є наукова новизна запропонованих вирішень і чим вони відрізняються від наявних. Інтелектуальна система, яка є результатом виконання магістерської роботи, повинна реалізовувати запропоновані вирішення (алгоритми, методи, підходи) та демонструвати їхні дієвість та ефективність.

Опис призначення системи містить перелік задач та функцій, які виконує система, а також перелік акторів (осіб, підсистем), які взаємодіють з нею. Задачі, функції та акторів треба подати у структурованому вигляді. Якщо задачі та функції залежні, то треба вказати це та визначити тип залежності. Якщо існує декілька підтипів акторів, треба вказати відмінності між ними. Для опису функцій системи, крім простих табличних форм, рекомендується використовувати діаграми прецедентів (use-case).

Місце застосування системи.

У підрозділі «Місце застосування системи» визначають:

- галузь, в якій пропонується застосувати систему прийняття рішень;
- вид діяльності, до якого вона застосовується;
- конкретні підприємства, установи, компанії, організації, підрозділи, в яких пропонується впровадження системи.

Обґрунтування розроблення та впровадження системи.

У підрозділі «Обґрунтування розроблення та впровадження системи» треба розглянути декілька пунктів, поданих нижче.

1. Визначається наявність проблеми, яка потребує вирішення за рахунок впровадження системи прийняття рішень, та потреби її вирішення.

Детальний аналіз проблеми здійснюється в контексті обраної галузі. Проблему визначають серед інших, споріднених проблем з метою визначення її місця та важливості. У роботі показують, як вирішення цієї проблеми вплине на вирішення інших проблем аналізують, чи вирішення цієї проблеми не призведе до виникнення інших, передбачуваних проблем. Якщо в результаті аналізу передбачено виникнення нових проблем треба рекомендувати методи та засоби для їх вирішення. Структуру проблем рекомендується відобразити з використанням відповідних діаграм мови SysML.

2. Обґрунтовується неможливість чи проблемність вирішення проблеми за рахунок наявних засобів та доцільність розроблення нового продукту.

У цьому пункті наводиться характеристика наявних засобів, які використовують і обраній галузі для вирішення таких завдань. Перед порівнянням продуктів треба обрати параметри системи (кількісні та якісні), які будуть використовуватися для порівняння. Конкретний набір параметрів визначається проблемою, яка вирішується в магістерській роботі. До цього набору також треба ввести найважливіші загальносистемні характеристики (наприклад, продуктивність, складність, вартість тощо). Існуючі продукти порівнюють для всіх обраних параметрів та обґрунтовують, чому жоден з них повнок мірою не зможе вирішити визначену проблему.

3. Описується новизна розроблення та його переваги порівняно з існуючими засобами.

У цьому пункті чітко визначають новизну запропонованого розроблення та переваги порівняно з існуючими засобами, користуючись критеріями, визначеними у попередньому пункті. Якщо очікується, що впровадження системи приведе до покращання одних, але погіршення інших параметрів або можливостей системи, то впровадження системи додатково обґрунтовується, посиляючись на відносну важливість параметрів або пропонуючи заходи для вирішення можливих проблем.

4. Подається оцінка часових та фінансових витрат на розроблення.

Оцінка часових витрат обґрунтовується, враховуючи складність завдання. Для оцінки складають укрупнений календарний план із зазначенням етапів (підзадач) та орієнтовного часу, необхідного на виконання кожного етапу (розв'язування підзадачі). Загальний план виконання роботи подають у вигляді діаграми Ганта (наприклад, користуючись засобами MS Project). У плані робіт вказують необхідну кількість ресурсів для розв'язування окремих підзадач, та ролі (кваліфікацію) цих ресурсів. Фінансові витрати оцінюють

відповідно до розробленого календарного плану, необхідного обладнання і інших видатків на розроблення. Структура видатків подається у вигляді розгорнуті калькуляції видатків.

Очікувані ефекти від впровадження системи.

У підрозділі «Очікувані ефекти від впровадження системи» описуються:

- прогнозований економічний ефект від застосування системи прийняття рішень, зокрема наявність чи відсутність такого ефекту, джерела і фактори, за рахунок яких передбачається ефект, попереднє оцінення розмірів економічного ефекту;

- оцінення ефекту від зростання продуктивності праці, підвищення якості прийняття рішень, зменшення часових витрат на впровадження системи прийняття рішень;

- організаційний ефект - скорочення кількості персоналу, задіяного в процесах підготовки, формування та прийняття рішень чи зміни його функцій, зміна організаційної структури, скорочення кількості операцій, які виконуються працівниками;

- технологічний ефект за рахунок вивільнення чи зменшення технічних, програмних та інших ресурсів, а також їх ефективнішого використання, освоєння нових технологій;

- інші ефекти - ергономічні, психологічні, рекламні, соціальні.

Розроблення концептуальної моделі системи.

У підрозділі «Розроблення концептуальної моделі системи» детальніше визначають вимоги та обмеження, вхідні та вихідні дані, структуру та порядок роботи системи. При цьому спираються на попередній аналіз функцій системи, поданий у підрозділі «Призначення системи».

Опис вхідних даних містить:

- перелік видів вхідної інформації (документи, складові баз даних, вхідні повідомлення, сигнали, Web-сайти тощо), їхню характеристику, опис, класифікацію;

- структура елементів вхідних даних - назва вхідного документа (елемента, повідомлення), перелік одиниць даних, їхні назви, типи, розмірність, точність, обмеження, інші вимоги;

- від кого і як часто надходять вхідні дані, додаткові умови та обмеження, які накладає джерело вхідних даних (наприклад, шифрування або дотримання певних ліцензійних угод).

Перелік вхідної інформації подають структуровано з обов'язковим вказанням, яким стандартам (API) та яким версіям цих стандартів відповідають вхідні сигнали та повідомлення. Перелік вхідної інформації подають як частину вимог до системи.

Опис вихідних даних передбачає:

- перелік вихідних документів, повідомлень, сигналів, їх опис, характеристику, класифікацію, спосіб формування та передавання;
- опис структури вихідних даних - назву, склад, перелік складових одиниць, їхні назви та зміст;
- призначення вихідної інформації;
- періодичність видачі вихідних даних.

Опис вихідних даних подають структуровано, зазначаючи, яким стандартам (API) та яким версіям цих стандартів відповідають вихідні сигнали та повідомлення.

У пункті «Опис функцій та структури системи» детальніше визначають складові компоненти системи та пропонують порядок їхньої взаємодії. Рекомендується починати цей розділ з побудови контекстної діаграми системи (наприклад, з використанням SysML), в якій визначають основну функцію системи, з чим система взаємодіє і яка вхідна та вихідна інформація передається під час взаємодії.

Функції системи детальніше визначаються у діаграмі класів мови UML або діаграмі блоків мови SysML. Ця діаграма подає головні сутності та складові частини системи і залежності між ними. У концептуальній моделі визначають функції кожного блоку, зв'язки та залежності з інттиими блоками, а також загальну структуру системи у вигляді діаграми блоків.

Для опису динамічних особливостей поведінки системи використовують діаграми станів та діяльностей SysML.

У пункті «Опис вимог до системи» структуровано визначають вимоги до системи. Вимоги поділяють на функціональні та додаткові. Функціональні вимоги визначають способи використання системи. Власне відповідність готового продукту функціональним вимогам визначає успіх проекту. Вимоги подають структуровано, у табличній формі або у формі діаграми вимог SysML. У роботі посилаються на відповідні блоки системи, які відповідають за реалізацію кожної вимоги.

Додаткові вимоги до системи прийняття рішень, яка розробляється, передбачають перелік додаткових умов та обмежень щодо функціонування системи, зокрема вони містять:

- вимоги до технічного, програмного та мережного середовища, в якому реалізується система;
- вимоги до результатів роботи системи прийняття рішення, регламенту функціонування, способів відображення, передавання та збереження;

– вимоги до сумісності та способів взаємодії й обміну інформацією з іншими системами;

– ергономічні вимоги до системи – зручність супроводу та обслуговування системи, раціональне компонування технічних елементів і елементів інтерфейсу, комфортність засобів керування системою, естетичне оформлення;

– вимоги до безпеки і захисту інформації та інших компонентів системи від несанкціонованого доступу, втрати, знищення, пошкодження;

– екологічні вимоги - обмеження на споживання електроенергії та тепловиділення, використання найновіших апаратних і програмних вирішень для контролю екологічних параметрів;

– організаційні та регламентні вимоги - вимоги до персоналу, його складу та кваліфікації, графіків експлуатації системи, прав та повноважень для роботи з системою тощо.

У пункті *«Додаткові формальні та узагальнені моделі системи»* за потреби наводять формальну математичну постанову задачі та будують відповідну математичну модель. Тут також пропонують математичні методи та підходи, які використовуватимуть для розв'язання поставленої задачі та детально і всебічно обґрунтовують їхній вибір.

З МЕТОДИ ПРОЕКТУВАННЯ ІНТЕЛЕКТУАЛЬНИХ СИСТЕМ

У окремому підрозділі розділу роботи «Методи та засоби вирішення проблеми» магістрант досліджує питання вибору та обґрунтування методів проектування як інтелектуальної системи в цілому так і кожної її складової частини. В цьому розділі наведено обґрунтування вибору методів проектування інтелектуальних систем. Для проектування бази даних та знань запропоновано методологію концептуального проектування реліційних баз даних. Приведено опис технологій процедурно-орієнтованого програмування, структурного програмування, модульне програмування, а також сутність об'єктно-орієнтованого підходу. Для вибору та обґрунтування методів вирішення проблеми зі штучного інтелекту наведено методи аналізу предметної області для проектування семантичних зв'язків та їхнього семантичного наповнення, моделі подання знань, постановку задачі побудови індуктивної інтелектуальної системи, а також застосування нейтонних мереж прямого поширення. Розглянуто онтології як ядро бази знань інтелектуальних систем.

3.1 Вибір та обґрунтування методів проектування інтелектуальних систем

Мова UML – загально-цільова мова візуального моделювання, що розроблена для специфікації, візуалізації, проектування й документування компонентів програмного забезпечення, бізнес-процесів та інших систем [26, 135]. Мова UML одночасно є простим і потужним засобом моделювання, що може бути ефективно використаним для побудови концептуальних, логічних та графічних моделей складних систем будь-якого цільового призначення. Ця мова увібрала в себе найкращі якості методів програмної інженерії, які з успіхом використовувалися протягом останніх років під час моделювання складних систем.

Мова UML ґрунтується на множині базових понять, які можуть бути вивчені й застосовані більшістю програмістів та розробників, знайомих з методами об'єктно-орієнтованого аналізу й проектування. При цьому базові поняття можуть комбінуватися й розширюватися тобто, що фахівці об'єктного моделювання одержують можливість самостійно розробляти моделі складних систем у будь-яких предметних областях.

Мова UML призначена для вирішення таких завдань:

1. Надати в розпорядження користувачів виразну мову візуального моделювання, що легко сприймається та спеціально призначену для

розроблення й документування моделей складних систем будь-якого цільового призначення.

Мова йде про те, що важливим фактором подальшого розвитку й повсюдного використання методології об'єктно-орієнтованого аналізу та проектування (ООАП) є інтуїтивна ясність і зрозумілість основних конструкцій відповідної мови моделювання. Мова UML містить у собі не тільки абстрактні конструкції для подання метамodelей систем, але й цілий ряд конкретних понять, що мають цілком певну семантику. Це дає змогу мові UML одночасно досягти не тільки універсальності подання моделей, але й можливості опису досить тонких деталей реалізації цих моделей стосовно конкретних систем.

2. Розширення й спеціалізація базових понять для точнішого подання моделей систем у конкретній предметній області.

Хоча мова UML є формальною мовою - специфікації та формальність її опису відрізняється від синтаксису традиційних формально-логічних мов та відомих мов програмування. Мова UML може бути пристосована для конкретних предметних областей. Це можливе тому що в самому описі мови UML закладений механізм розширення базових понять, що є самостійним елементом мови й має власний опис у формі правил розширення.

3. Опис мови UML підтримує специфікацію моделей, щоб не залежати від конкретних мов програмування та інструментальних засобів проектування програмних систем.

Мова йде про те, що жодна з конструкцій мови UML не повинна залежати від особливостей її реалізації у відомих мовах програмування.

4. Опис мови UML містить в собі семантичний базис для розуміння загальних особливостей ООАП.

Говорячи про цю особливість, мають на увазі самодостатність мови UML для розуміння не тільки його базових конструкцій, але що не менш важливо - розуміння загальних принципів ООАП. У цьому зв'язку необхідно відзначити, що оскільки мова UML не є мовою програмування, а служить засобом для рішення завдань об'єктно-орієнтованого моделювання систем, опис мови UML повинне по можливості містити в собі всі необхідні поняття для ООАП. Без цієї властивості мова UML може виявитися непотрібною більшістю користувачів, які не знайомі із проблематикою ООАП складних систем.

5. Заохочувати розвиток ринку об'єктних інструментальних засобів.

Більше 800 провідних виробників програмних і апаратних засобів, зусилля яких зосереджені в рамках OMG, бачать перспективи розвитку сучасних інформаційних технологій і основу свого комерційного успіху в широкому просуванні на ринок інструментальних засобів, що підтримують

об'єктні технології. Говорячи саме про об'єктні технології, розроблювані з OMG мають на увазі, насамперед, сукупність технологічних рішень CORBA і UML. Із цього погляду мові UML приділяється роль базового засобу для опису й документування різних об'єктних компонентів CORBA.

6. Сприяти поширенню об'єктних технологій і відповідних понять ООАП.

Це завдання тісно пов'язане з попереднім й має з ним багато загального. Зусилля досить великої групи розроблювачів були спрямовані на інтеграцію в рамках мови UML багатьох відомих технік візуального моделювання, які успішно зарекомендували себе на практиці. Хоча це привело до ускладнення мови UML у порівнянні з відомими нотаціями структурного системного аналізу, однак такою перевагою є висока гнучкість і візуальних можливостей.

7. Інтегрувати в себе новітні й найкращі досягнення практики ООАП.

Мова UML безупинно вдосконалюється розробниками, і основою цієї роботи є його подальша інтеграція із сучасними модельними технологіями. При цьому різні методи системного моделювання одержують своє прикладне осмислення в рамках ООАП. Надалі ці методи можуть бути включені до складу мови UML.

Із загального погляду опис мови UML складається із двох взаємодіючих частин, таких як:

- семантика мови UML, що являє собою деяку мета-модель, яка визначає абстрактний синтаксис і семантику понять об'єктного моделювання мовою UML;

- нотація мови UML, що являє собою графічну нотацію для візуального подання семантики мови UML.

У рамках мови UML всі подання про модель складної системи фіксуються у вигляді спеціальних графічних конструкцій, що одержали назву діаграм. У термінах мови UML визначені такі види діаграм:

- діаграма варіантів використання (use case diagram);
- діаграма класів (class diagram);
- діаграми поведінки (behavior diagrams);
- діаграма станів (statechart diagram);
- діаграма діяльності (activity diagram);
- діаграма взаємодії (interaction diagrams);
- діаграма послідовності (sequence diagram);
- діаграма кооперації (collaboration diagram);
- діаграма реалізації (implementation diagrams);
- діаграма компонентів (component diagram);

- діаграма розміщення (deployment diagram).

Розроблення діаграми варіантів використання має такі цілі:

- Визначити загальні межі і контекст модельованої предметної області на початкових етапах проектування системи.
- Сформулювати загальні вимоги до функціональної поведінки проектованої системи.
- Розробити початкову концептуальну модель системи для її подальшої деталізації у формі логічних і фізичних моделей.
- Підготувати початкову документацію для взаємодії розробників системи з її замовниками та користувачами.

Суть даної діаграми полягає в наступному: проектована система представляється у вигляді множини сутності або акторів, що взаємодіють із системою за допомогою так званих варіантів використання. При цьому актором (actor) або дійовою особою називається будь-яка сутність, що взаємодіє з системою ззовні. Це може бути людина, технічний пристрій, програма або будь-яка інша система, яка може служити джерелом дії на модельовану систему так, як визначить сам розробник. У свою чергу, варіант використання (use case) служить для опису сервісів, які система надає акторові. Іншими словами, кожен варіант використання визначає деякий набір дій, що здійснюється системою під час діалогу з актором. При цьому нічого не мовиться про те, яким чином буде реалізовано взаємодію акторів з системою.

Діаграма класів (class diagram) служить для подання статичної структури моделі системи в термінології класів об'єктно-орієнтованого програмування. Діаграма класів може відображати, зокрема, різні взаємозв'язки між окремою сутністю предметної області, такими як об'єкти і підсистеми, а також описує їх внутрішню структуру і типи відношень. На даній діаграмі не вказується інформація про тимчасові аспекти функціонування системи. З цього погляду діаграма класів є подальшим розвитком концептуальної моделі проектованої системи.

Призначення діаграми станів - опис можливих послідовностей станів та переходів, які в сукупності характеризують поведінку елементів моделі протягом його життєвого циклу. Діаграма станів представляє динамічну поведінку сутностей, на основі специфікації їх реакції на сприйняття деяких конкретних подій. Системи, які реагують на зовнішні дії від інших систем або від користувачів, іноді називають реактивними. Якщо такі дії ініціюються в довільні випадкові моменти часу, то говорять про асинхронну поведінку моделі.

Для моделювання процесу виконання операцій в мові UML використовуються так звані діаграми діяльності. Вживана в них графічна

нотація багато в чому схожа на нотацію діаграми станів, оскільки на діаграмах діяльності також присутні позначення станів і переходів. Відмінність полягає в семантиці станів, які використовуються для подання діяльностей. Кожне позначення на діаграмі діяльності відповідає виконанню деякої елементарної операції, а перехід в наступний стан спрацьовує тільки при завершенні цієї операції в попередньому стані. Графічно діаграма діяльності представляється у формі графа діяльності, вершинами якого є стани дій, а дугами - переходи від одного стану дій до іншого.

Для моделювання взаємодії об'єктів у мові UML використовуються діаграми взаємодії. Говорячи про ці діаграми, мають на увазі два аспекти взаємодії. По-перше, взаємодію об'єктів можна розглядати в часі, і тоді для подання тимчасових особливостей передачі й прийому повідомлень між об'єктами використовуються діаграми послідовності.

Особливості взаємодії елементів модельованої системи можуть бути подані на діаграмах послідовності і кооперації. Якщо перша служить для візуалізації тимчасових аспектів взаємодії, то діаграма кооперації призначена для специфікації структурних аспектів взаємодії. Головна особливість діаграми кооперації полягає в можливості графічно подати не тільки послідовність взаємодії, але і всі структурні відносини між об'єктами, що беруть участь в цій взаємодії.

Діаграма компонентів, на відміну від раніше розглянутих діаграм, описує особливості фізичного подання системи. Діаграма компонентів дає змогу визначити архітектуру системи, що розробляється, встановивши залежності між програмними компонентами, в ролі яких може виступати початковий, бінарний і виконавчий код. У багатьох середовищах розроблення модуль або компонент відповідає файлу. Пунктирні стрілки, що сполучають модулі, показують відношення взаємозалежності, аналогічні тим, які мають місце при компіляції початкових текстів програм. Основними графічними елементами діаграми компонентів є компоненти, інтерфейси і залежності між ними.

Діаграма компонентів розробляється для таких цілей:

- Візуалізація загальної структури початкового коду програмної системи.
- Специфікація варіанту програмної системи.
- Забезпечення багатократного використання окремих фрагментів програмного коду.
- Подання концептуальної і фізичної схем баз даних.

Діаграма розміщення застосовується для подання загальної конфігурації і топології розподіленої програмної системи і містить розподіл компонентів за

окремими вузлами системи. Крім того, діаграма розміщення показує наявність фізичних з'єднань – маршрутів передачі інформації між апаратними пристроями, що задіяні в реалізації системи.

3.2 Вибір та обґрунтування методів проектування бази даних та знань

У цьому розділі описані основні етапи життєвого циклу застосування бази даних. Одним з них є проектування бази даних. Воно починається після закінчення аналізу всіх вимог до проекту, що висувуються з боку підприємства-замовника.

У цьому розділі розглядається методологія, запропонована для проектування бази даних, що входить у загальний життєвий цикл застосування, що використовує бази даних реляційного типу [64, 65]. Запропонована методологія є поетапним керівництвом, що охоплює основні етапи проектування баз даних, а саме: побудова інфологічної та даталогічної моделей, концептуальне, логічне і фізичне проектування. Етап інфологічного моделювання можна поділити на два – концептуальне та логічне. Нижче описано призначення кожного з цих етапів.

- Концептуальне проектування - створення концептуального подання бази даних, що включає визначення типів найважливіших сутностей і зв'язків, що існують між ними, та атрибутів.

- Логічне проектування - перетворення концептуального уявлення в логічну структуру бази даних, включаючи проектування взаємозв'язків.

- Даталогічне проектування - ухвалення рішення про те, як логічна модель буде реалізована (за допомогою таблиць) у базі даних, що створюється з використанням вибраної СУБД.

Найважливіші чинники успішного проектування бази даних.

Наведемо деякі рекомендації, які треба виконувати для успішного проектування бази даних.

- Підтримуйте постійний і активний зв'язок з майбутніми користувачами застосування.

- При проведенні процедур моделювання даних дотримуйтеся обґрунтованої методології.

- Застосовуйте підходи, що передбачають створення застосувань управління даними.

- Створюйте модель даних з урахуванням вимог підтримки їхньої структурної цілісності й узгодженості.

– У процесі реалізації методології моделювання даних застосовуйте методи розроблення концептуальної моделі, нормалізації та перевірки цілісності транзакцій.

– Для подання моделі даних якнайширше використовуйте схеми.

– Для опису додаткових семантичних вимог до даних використовуйте засоби мови проектування баз даних (Database Design Language - DBDL).

– На додаток до схем моделей даних і конструкцій DBDL розробіть словник опису даних.

– Повертайтеся до вже виконаних раніше етапів, якщо це потрібно для досягнення оптимальних результатів.

Викладені вище рекомендації враховані в запропонованій методології проектування баз даних.

Концептуальне і логічне проектування бази даних включає *три* основні етапи. Завданням *першого* етапу є розбиття проекту на групу відносно невеликих (і простіших) завдань виходячи з уявлень про предметну область застосування, властивих кожному з типів кінцевих користувачів. Результатом виконання цього етапу є створення локальних концептуальних моделей даних, що є повним і точним віддзеркаленням уявлень про предметну область застосування окремих типів користувачів.

На *другому* етапі локальні концептуальні моделі даних перетворюються в локальні логічні моделі даних (для реляційної моделі даних), що складаються з ER-діаграми, реляційної схеми і супровідної документації. Потім коректність логічних моделей даних перевіряється за допомогою правил нормалізації. Нормалізація є ефективним засобом, що дає змогу переконатися в структурній узгодженості, логічній цілісності і мінімальній надлишковості прийнятої моделі даних. Додатково модель даних перевіряється на можливість проведення транзакцій, які виконуватимуться користувачами створюваного застосування. Всі ці перевірки дають змогу отримати необхідну впевненість в тому, що прийнята модель даних є цілком прийнятною. Після закінчення етапу кожна локальна логічна модель даних при необхідності може застосовуватися для підготовки прототипів реалізацій бази даних, призначених для окремих типів користувачів застосування.

На *третьому* етапі виконується об'єднання локальних логічних моделей даних (що відображають уявлення про предметну область окремих типів користувачів) в єдину глобальну логічну модель даних всього підприємства (узагальнювальну уявлення про предметну область всіх типів користувачів). Глобальна логічна модель перевіряється за таким же принципом, як і локальні моделі; це дає змогу переконатися в тому, що її структура є правильною і

підтримує всі необхідні транзакції. Цей етап зазвичай потрібний, якщо застосування складається більше ніж з одного уявлення.

У пропонованій методології проектування істотна роль відводиться кінцевим користувачам, які постійно залучаються розробниками для ознайомлення і перевірки створюваних моделей даних і супровідної документації. Проектування баз даних, зазвичай, є циклічним процесом, що має конкретну точку початку, практично не має кінця та включає необмежене число циклів поліпшень і доопрацювань.

Запропоновану методологію необхідно розглядати як загальну схему, яка дає змогу підвищити ефективність роботи з проектування баз даних.

3.2.1 Методологія концептуального проектування реляційних баз даних

Етап 1. Створення локальної концептуальної моделі даних на основі уявлення про предметну область кожного з типів користувачів.

Мета. Створення локальної концептуальної моделі даних на основі уявлення про предметну область кожного окремого типу користувачів.

На першому етапі проектування бази даних має бути розроблена концептуальна модель даних для кожного уявлення, що охоплює предметну область; така модель даних називається локальною концептуальною моделлю даних для цього уявлення. У процесі аналізу треба виявити всі уявлення користувача, які потрібні для застосування, що розробляється, і передбачити можливість об'єднання деяких уявлень для створення узагальненого уявлення, позначеного відповідним ідентифікатором, залежно від ступеня перекриття окремих уявлень.

Кожна локальна концептуальна модель даних складається з таких компонентів:

- типи сутностей;
- типи зв'язків;
- атрибути і домени атрибутів;
- первинні ключі;
- альтернативні ключі;
- обмеження цілісності.

Концептуальна модель даних доповнюється документацією, що створюється в процесі розроблення цієї моделі, включаючи словник даних. Докладні відомості про супровідну документацію, яка може бути підготовлена на різних етапах, наведені при описанні цих етапів. На першому етапі розроблення виконується:

- 1) визначення типів сутностей;
- 2) визначення типів зв'язків;

- 3) визначення атрибутів і їхньої належності до типів сутностей і зв'язків;
- 4) визначення доменів атрибутів;
- 5) визначення атрибутів, що є потенційними і первинними ключами;
- 6) обґрунтування необхідності використання понять розширеного моделювання (необов'язковий етап);
- 7) перевірка моделі на відсутність надлишковості;
- 8) перевірка відповідності локальної концептуальної моделі конкретним призначенням для користувача транзакціям;
- 9) обговорення локальних концептуальних моделей даних з кінцевими користувачами.

Етап 1.1. Визначення типів сутностей

Мета. *Визначення основних типів сутностей, яка потрібна для конкретного уявлення.* Перший етап створення локальної концептуальної моделі даних полягає у визначенні основних об'єктів, які можуть цікавити користувача. Ці об'єкти є типами сутностей, що входять у модель. Один з методів ідентифікації сутностей полягає у вивченні специфікацій з виконання конкретних функцій користувача. З цих специфікацій треба визначити всі використовувані в них іменники або поєднання іменника і прикметника (наприклад, «табельний номер», «прізвище працівника», «номер об'єкту нерухомості», «адреса об'єкту нерухомості», «орендна плата», «кількість кімнат»). Потім серед них вибираються найбільш значущі об'єкти (категорії працівників, напрями діяльності) або важливі концепції, виключаються всі іменники, які просто визначають інші об'єкти.

Альтернативний спосіб ідентифікації сутностей полягає в пошуку об'єктів, які існують незалежно від інших. При цьому способом істотну допомогу можуть надати користувачі, для яких створюється застосування.

У деяких випадках виділення сутностей буває ускладненим незадовільним способом їх подання в специфікаціях. Користувачі, висловлюючи свої думки, часто використовують не однозначні визначення, а приклади або аналогії.

Процес проектування ще більше ускладнюється у зв'язку з тим, що користувачі часто вживають синоніми або омоніми. Синонімами називаються слова, схожі за сенсом, але різні за звучанням і написанням, наприклад «відділення» і «філія». Омоніми - це слова, однакові за написанням і звучанням, але різні за смисловими значеннями, які вони мають, причому реальне значення у кожному конкретному випадку можна встановити тільки за контекстом. Так, слово «програма» може позначати курс навчання, майбутню серію послідовних подій, план майбутньої роботи і навіть розклад телепередач.

Далеко не завжди очевидно те, чим є певний об'єкт - сутністю, зв'язком або атрибутом. Аналіз є суб'єктивним процесом, тому різні розробники можуть розглядати різноманітні, але цілком допустимі інтерпретації одного і того самого факту. Вибір варіанту в значній мірі залежить від здорового глузду і досвіду виконавця. Розробники баз даних повинні обмежити предметну область рамками того погляду на світ і категорії, що існують в ньому, які задаються контекстом створюваного застосування. Тому іноді неможливо встановити унікальну множину сутностей, яка однозначно визначається із заданої специфікації вимог. Проте серія ітеративних процедур аналізу всього комплексу специфікацій проекту, безумовно, дасть змогу визначити множину сутностей, необхідну для задоволення вимог до системи.

Документування типів сутностей. Після виділення кожній сутності присвоюється певне осмислене ім'я, яке обов'язково має бути зрозуміле користувачам. Вибране ім'я і опис сутності вміщується у словник даних. Якщо це можливо, встановлюють і вносять до документації дані про очікувану кількість екземплярів кожної сутності. Якщо сутність відома користувачам під різними іменами, всі додаткові імена рекомендується визначити як синоніми або псевдоніми і також внести у словник даних.

Етап 1.2. Визначення типів зв'язків.

Мета. *Визначення найважливіших типів зв'язків, що існують між сутностями, ви діленими на попередньому етапі.*

Після виділення сутностей наступним етапом розроблення стає встановлення всіх зв'язків, що існують між ними. Одним із методів визначення сутностей є вибірка всіх іменників, присутніх у специфікаціях вимог користувачів. І в цьому випадку для виявлення зв'язків треба виконати граматичний аналіз специфікації вимог. Аналогічний підхід можна використовувати і при визначенні існуючих зв'язків, проте в цьому випадку вибираються всі вирази, в яких містяться дієслова.

Той факт, що текст специфікацій містить інформацію про деякі зв'язки, дає змогу припустити, що ці зв'язки є вельми важливими для предметної області. Тому вони обов'язково мають бути відображені у створюваній моделі.

Проектувальників цікавлять тільки ті зв'язки між сутностями, які необхідні для задоволення вимог до проекту.

У більшості випадків зв'язки є двосторонніми, іншими словами, зв'язки існують тільки між двома сутностями. Треба ретельно перевіряти наявність у проекті складних зв'язків, які об'єднують понад дві сутності різних типів, а також рекурсивних зв'язків, що існують між сутностями одного і того самого типу.

Особливу увагу треба приділяти перевірці того, чи були виділені всі зв'язки, явно або неявно присутні у специфікаціях вимог користувачів.

Кожну з можливих пар сутностей корисно перевірити на наявність між ними деякому зв'язку, проте в складних системах, що включають сотні типів сутностей, це завдання може виявитися надзвичайно трудомістким. Але відмовлятися взагалі від виконання подібних перевірок безглуздо, оскільки відповідальність за сумні наслідки цієї відмови доведеться нести як аналітикам, так і проектувальникам. Так або інакше, всі пропущені і зв'язки будуть обов'язково виявлені пізніше, при здійсненні перевірки можливості виконання транзакцій, необхідних користувачам (етап 1.8).

Застосування діаграм «сутність-зв'язок» (ER-діаграм). Робота проектувальника істотно спрощується, якщо є можливість вивчити структуру складної системи за допомогою схеми, а не аналізувати докладні текстові описи специфікацій вимог користувачів. Для подання сутностей і зв'язків між ними зазвичай використовуються діаграми і «сутність-зв'язок» (ER-діаграми). Рекомендується передбачити найширше використання ER-діаграм на всіх етапах проектування бази даних.

Це дозволить завжди мати під рукою образ модельованої частини предметної області. Для побудови ER-діаграм можна застосувати об'єктно-орієнтовану систему позначень, що ґрунтується на використанні мови *UML* (*Unified Modeling Language* - універсальна мова моделювання), але інші системи позначень дозволяють досягти такого самого результату.

Визначення обмежень кратності, які розповсюджуються на типи зв'язків. Встановивши зв'язки, які мають місце в створюваній моделі, треба визначити кратність кожного з них. Якщо відомі конкретні значення кратності або навіть верхня чи нижня межа цих значень, то цю інформацію обов'язково треба зафіксувати в документації.

Обмеження кратності служать для перевірки якості даних. Ці обмеження є тими визначеннями властивостей екземплярів сутностей, які перевіряються при зміні даних в базі з метою виявлення того, чи приведуть такі зміни до порушення встановлених правил. Модель, яка включає обмеження кратності, наочніше відображає семантику зв'язків і тому набагато краще характеризує задану предметну область.

Перевірка відсутності дефектів типу «розгалуження» і типу «розрив». Після виявлення необхідних зв'язків треба перевірити, чи служить кожен зв'язок в моделі справжнім відображенням залежностей «реального світу». Крім того, треба переконатися, що немає в моделі не виявлених дефектів типу «розгалуження» і типу «розрив».

Перевірка дотримання вимоги про участь кожної сутності щонайменше в одному зв'язку. Як правило, в моделі не повинно бути сутностей, ізольованої від всіх інших сутностей, оскільки після прив'язки ізольованої сутностей до відношення на етапі 2.2 неможливо буде перейти до цього відношення за допомогою засобів доступу до бази даних. Відомим виключенням з цього правила є база даних з єдиним відношенням. Але якщо в базі даних з декількома відношеннями буде виявлена ізольована сутність, треба перевірити, чи присутня така сутність де-небудь в моделі, можливо, під іншим ім'ям. Якщо ця перевірка не дасть результатів, треба знову вивчити вимоги, щоб визначити, чи не був пропущений якийсь зв'язок. Якщо пропущені зв'язки не будуть виявлені, знову звернутися до користувачів і визначити разом з ними, як саме використовується така ізольована сутність.

Документування типів зв'язків. Після визначення окремих типів зв'язків їм присвоюються осмислені імена, які мають бути зрозумілі користувачам. Рекомендується поміщувати у словник даних розгорнений опис кожного зв'язку, що включає зведення про обмеження кратності.

Етап 1.3. Визначення атрибутів і їх приналежності до типів сутностей і зв'язків.

Мета. Пов'язання атрибутів з відповідними типами сутностей або зв'язків.

На цьому етапі треба виявити всі дані, що описують сутності і зв'язки, виділені в створюваній моделі бази даних. Можна скористатися тим самим методом, який застосовувався для ідентифікації сутностей. Для цього виберемо всі іменники і фрази, що їх містять, присутні в специфікаціях вимог користувачів. Вибраний іменник подає атрибут в тому випадку, якщо він описує властивість, якість, ідентифікатор або характеристику деякої сутності або зв'язку.

Після виявлення сутності (х) або зв'язку (у) для отримання необхідних відомостей про атрибути найпростіше скористатися специфікацією вимог і спробувати знайти відповідь на запитання: «Яку інформацію треба зберігати про х або у?» Відповідь на це запитання треба також включити в текст специфікації. Але в деяких випадках треба звернутися до користувачів, щоб вони уточнили свої вимоги. На жаль, користувачі при подальших зверненнях до них часто дають відповіді, що містять додаткові вимоги, тому кожна отримана відповідь користувача підлягає найстрогішому аналізу.

Прості і складені атрибути. Важливо відзначити, що кожен атрибут може бути або простим, або складеним (композиційним, груповим). Складені атрибути є набором простих атрибутів. На цьому етапі важливо визначити всі прості атрибути, які мають бути подані в концептуальній моделі бази даних,

включаючи і ті, які згодом будуть використані для створення складених атрибутів.

Однозначні і багатозначні атрибути. Атрибути можуть поділятися не тільки на прості або складені, але і розглядатися як однозначні або багатозначні. Найчастіше зустрічаються однозначні атрибути, але при певних обставинах можуть також зустрітися і багатозначні атрибути; іншими словами, атрибути, які включають декілька значень, відповідних одному екземпляру сутностей. Як показано при описі етапу 2.2, багатозначні атрибути зрештою перетворюються у відношення, тому обидва підходи приводять до отримання одного і того самого кінцевого результату.

Похідні атрибути. Атрибути, значення яких можуть бути встановлені за допомогою значень інших атрибутів, називаються похідними або обчислюваними. Дуже часто подібні атрибути взагалі не відображаються в концептуальній моделі даних. Але в деяких випадках похідний атрибут, що застосовується в моделі даних, може не відображати зміни значень одного або декількох атрибутів, з яких він обчислюється, при їхньому видаленні або модифікації. У цьому випадку похідний атрибут має явно подаватися в моделі даних, що дозволить попередити небажану втрату інформації. Проте, якщо похідний атрибут поданий в моделі даних, треба неодмінно вказати, що він є похідним. Залежно від того, як застосовується такий атрибут, нове значення похідного атрибуту може обчислюватися або при кожному зверненні до нього, або тільки при зміні значень атрибутів, що використовуються для його обчислення.

Потенційні проблеми. При визначенні використовуваних у деякому уявленні сутностей, зв'язків і атрибутів дуже часто виявляється, що на попередніх етапах одна або декілька сутностей, зв'язки і атрибути було пропущено. У цьому випадку треба повертатися до вже виконаних етапів і документально оформити виявлену сутність, зв'язок і атрибути, після чого ще раз проаналізувати зв'язки, в яких вони беруть участь.

Оскільки звичайна кількість атрибутів значно перевищує кількість сутностей і зв'язків, може виявитися корисним спочатку підготувати список всіх атрибутів, що використовуються в специфікаціях вимог користувачів. У міру пов'язання чергового атрибута з деякою сутністю або зв'язком він викреслюється зі списку. Подібний метод дає змогу гарантувати, що кожен з атрибутів буде пов'язаний із сутністю або зв'язком тільки одного типу. Коли зі списку буде викреслений останній атрибут, всі ідентифіковані в моделі атрибути будуть пов'язаними з деякою сутністю або зв'язком.

Треба пам'ятати, що в деяких випадках створюється враження, що деякі атрибути повинні відноситися до сутностей або зв'язків декількох різних типів. Подібна ситуація виникає в таких випадках.

1) Виявлено декілька сутностей, які можуть бути подані у вигляді однієї сутності. З іншого боку, може бути так, що ці сутності мають багато спільних атрибутів, але характеризуються також атрибутами або зв'язками, унікальними для кожної із сутностей. У подібних випадках треба вирішити, чи варто узагальнити ці сутності і подати у вигляді однієї сутності або зберегти їх як спеціалізовані сутності, що подають різні функції.

2) Виявлено новий зв'язок між типами сутностей. Треба встановити належність атрибута тільки до однієї сутності (яка називається батьківською) і переконатися в тому, що такий зв'язок був вже виявлений на етапі 1.2. Якщо ці умови не дотримуються, то треба відновити проектну документацію і внести до неї відомості про новий виявлений зв'язок.

Етап 1.4. Визначення доменів атрибутів.

Мета. Визначення доменів для всіх атрибутів, присутніх у концептуальній моделі оаних.

Завдання цього етапу полягає у визначенні доменів атрибутів для всіх атрибутів, присутніх у моделі. Доменом називається деяка множина значень, елементи якої вибираються для присвоєння значень одному або декільком атрибутам.

Повністю розроблена модель даних має включати домени для кожного з атрибутів, що містяться в ній. Домени мають містити такі дані:

- множину допустимих значень для атрибута;
- відомості про розмір і формат кожного з атрибутів.

У доменах може бути вказана й інша додаткова інформація, наприклад, відомості про допустимі операції із значеннями атрибутів, а також дані про те, які атрибути можна використовувати для порівняння з іншими атрибутами або при побудові комбінацій з декількох атрибутів. Проте методологія визначення характеристик доменів атрибутів для СУБД все ще є предметом досліджень.

Документування доменів атрибутів. Після визначення доменів атрибутів їхні іменна і характеристики вміщуються у словник даних. Одночасно оновлюються записи словника даних, що відносяться до атрибутів, - у них заносяться імена, призначених кожному атрибуту доменів, замість позначення типів даних та інформації про розмірність.

Етап 1.5. Визначення атрибутів, що є потенційними і первинними ключами.

Мета. Визначення всіх потенційних ключів для кожного типу сутностей і, якщо таких ключів буде декілька, вибір серед них первинного ключа.

На цьому етапі для кожної сутності встановлюється потенційний ключ (або ключі) після чого здійснюється вибір первинного ключа. Потенційним ключем називається атрибут або мінімальний набір атрибутів заданої сутності, що дає змогу однозначне ідентифікувати кожен її екземпляр. Для деякої сутності можлива наявність декількох потенційних ключів. У цьому випадку серед них треба вибрати один ключ, який називатиметься первинним ключем. Решта всіх потенційних ключів називатиметься альтернативними ключами.

При виборі первинного ключа серед декількох потенційних рекомендується:

- використовувати потенційний ключ з мінімальним набором атрибутів;
- використовувати той потенційний ключ, ймовірність зміни значень якого мінімальна;
- використовувати потенційний ключ, значення якого мають мінімальну довжину (л разі текстових атрибутів);
- використовувати потенційний ключ, значення якого мають найменшу максимальну довжину (у разі числових атрибутів);
- зупинити свій вибір на потенційному ключі, з яким буде найпростіше працювати (: погляду користувача).

У процесі визначення первинного ключа встановлюється, чи є певна сутність сильною або слабкою. Якщо вибрати первинний ключ для певної сутності виявилось можливим, то таку сутність прийнято називати сильною. І навпаки, якщо вибрати первинний ключ для певної сутності неможливо, то її називають слабкою. Первинний ключ для слабкої сутностей можна визначити тільки після відображення цієї слабкої сутностей її зв'язку з сутністю-власником на відношення, в якому згаданий зв'язок моделюється шляхом введення в таке відношення відповідного зовнішнього ключа. Процедура відображення сутностей і їхніх зв'язків у відношення буде розглянута при обговоренні етапу 2.2. Тому визначення первинних ключів для слабких сутностей може бути виконано тільки після досягнення вказаного етапу розроблення.

Документування первинних і альтернативних ключів. Після вибору первинних і альтернативних ключів сутностей (якщо такі визначені) зведення про них треба вмістити у словник даних.

Етап 1.6. Обґрунтування необхідності використання понять розширеного моделювання (необов'язковий етап).

Мета. Розглянути необхідність використання таких розширених понять моделювання, як уточнення/узагальнення, агрегація і композиція.

На цьому етапі передбачена можливість продовжити розроблення ER-моделі за допомогою розширених понять моделювання, а саме: уточнення/узагальнення, агрегація і композиція. Якщо вирішено здійснити уточнення, то у процесі розроблення треба виявити відмінності між сутностями шляхом визначення одного або декількох підкласів суперкласу сутностей. Підхід, що вимагає узагальнення, передбачає необхідність виявити загальні особливості різних сутностей для визначення узагальнювальної сутності суперкласу. Агрегація може застосовуватися для позначення зв'язку «has-a» (включає) або «is-part-of» (входить до складу) між типами сутностей; у такому зв'язку одна сутність є «ціле», а інша - її «частина».

Композиція (особливий тип агрегації) може застосовуватися для визначення взаємозв'язку між типами сутностей, який обумовлює строгу належність і термін існування між «цілим» і «частиною».

Неможливо дати точні рекомендації щодо того, чи повинні застосовуватися при розробленні ER-моделі розширені поняття моделювання, оскільки таке рішення часто є суб'єктивним і залежить від конкретних особливостей моделювання предметної області. Як зручне емпіричне правило можна рекомендувати, щоб при розгляді необхідності використання таких понять спочатку спробувати подати сутності і їхні зв'язки на ER-діаграмі з максимально можливою точністю. Отже, про необхідність використання розширених понять моделювання можна буде судити на підставі того, наскільки зручною для читання є ER-діаграма і чи дає змогу вона повністю змодельовувати сутності і зв'язки.

Етап 1.7. Перевірка моделі на відсутність надлишковості.

Мета. Перевірка на відсутність будь-якої надлишковості даних у моделі.

На цьому етапі локальна концептуальна модель даних перевіряється з конкретною ієтою: виявити-наявність в ній надлишковості даних та усунути цей недолік, якщо він буде виявлений. На цьому етапі виконуються такі операції.

1) Повторне дослідження зв'язків «один-до-одного» (1:1).

2) Видалення зайвих зв'язків.

Повторне дослідження зв'язків «один-до-одного» (1:1). Можливо, що в процесі визначення сутностей було виявлено дві сутності, які відповідають у цій організації одному і тому самому концептуальному об'єкту. У такому разі ці дві сутності мають бути об'єднані. Якщо для них визначені різні первинні ключі, то як первинний повинен вибиратися тільки один з них, а інший повинен використовуватися як альтернативний ключ. Видалення зайвих зв'язків. Зв'язок є зайвим, якщо подана в ньому інформація може бути отримана за допомогою

інших зв'язків. Розробник прагне створити мінімальну модель даних, а оскільки надлишкові зв'язки не потрібні, вони повинні бути видалені. На ER-діаграмі надлишкові зв'язки можна відносно легко виявити, оскільки вони проявляється в тому, що між двома сутностями є декілька шляхів. Але така ситуація не завжди означає, що один із зв'язків є зайвим, оскільки вони можуть відображати різні асоціації між сутностями.

При оцінці надлишковості важливо також враховувати, в який час виявляються ці зв'язки. Тому при оцінці надлишковості треба визначити призначення кожного зв'язку між сутностями.

Після закінчення цього етапу треба спростити локальну концептуальну модель даним шляхом видалення всієї властивої їй надлишковості.

Етап 1.8. Перевірка відповідності локальній концептуальній моделі конкретним користувацьким транзакціям.

Мета. Переконатися в тому, що локальна концептуальна модель підтримує транзакції, необхідні для певного уявлення.

На цьому етапі вже є локальна концептуальна модель даних, яка відповідає конкретному уявленню в певній предметній області. Призначення цього етапу полягає в перевірці моделі для визначення того, чи підтримує ця модель всі транзакції, необхідні для конкретного зображення. Для цього треба виконати всі необхідні операції вручну за допомогою цієї моделі. Якщо всі транзакції вдалося виконати, то перевірка відповідності концептуальної моделі даних необхідним транзакціям вважається успішною. Але якщо неможливо виконати вручну всі транзакції, це означає, що модель даних містить дефекти, які треба усунути. Тоді, ймовірно, в моделі даних не врахована яка-небудь сутність зв'язку або атрибуту.

Розглянемо два можливі способи перевірки того, що локальна концептуальна моделі даних підтримує необхідні транзакції.

1. Опис транзакції. При використанні цього способу перевіряється, чи надає моделі всю інформацію (сутності, зв'язки і їхні атрибути), необхідну для кожної транзакції. Для цього має бути складений опис вимог кожної транзакції.

2. Перевірка із застосуванням шляхів виконання транзакцій. Другий спосіб перевірки відповідності моделі даних необхідним транзакціям передбачає схематичне зображення шляху проходження кожної транзакції безпосередньо на ER-діаграмі. Безумовно, що при збільшенні кількості транзакцій ця діаграма ускладнюється, тому для зручності читання треба зобразити шляхи виконання транзакцій на декількох діаграмах.

Останній спосіб дає змогу проектувальникові подати візуально області моделі, які не потрібні для виконання транзакцій, а також області, які є необхідними для виконання транзакцій. Тому проектувальник має можливість

безпосередньо ознайомитися з тими, яку підтримку подає така модель даних при виконанні необхідних транзакцій. А якщо моделі є області, які, мабуть, не використовуються в яких-небудь транзакціях, то можна ще раз перевірити, чи повинна ця інформація подаватися в моделі даних. З іншого боку якщо деякі області моделі не дозволяють надати правильний шлях виконання транзакції, то, можливо, доведеться ще раз переконатися в тому, чи не пропущена яка-небудь важлива сутність, зв'язки або атрибути.

На перший погляд може здатися, що при використанні цього способу доводиться виконувати великий обсяг складної роботи для перевірки кожної транзакції, яку має підтримувати певне уявлення, і таке припущення не позбавлене підстави. Тому може виникнути спокуса пропустити цей етап. Проте вкрай важливо виконати ці перевірки саме зараз, а не відкладати їх на пізніший термін, коли усунення будь-яких помилок в моделі даних стане складнішим і дорожчим.

Етап 1.9. Обговорення локальних концептуальних моделей даних з кінцевими користувачами.

Мета. Обговорення локальних концептуальних моделей даних з кінцевими користувачами з метою підтвердження того, що така модель повністю відповідає специфікації вимог користувацького уявлення.

Перш ніж завершити перший етап розроблення, треба обговорити створені локальні концептуальні моделі даних з кінцевими користувачами.

Концептуальна модель даних має бути подана ER-діаграмою і супровідною документацією, що містить опис розробленої моделі даних. Якщо в запропонованій моделі будуть виявлені якісь невідповідності, треба внести до неї необхідні зміни (швидше за все, для цього треба буде повторно виконати один або декілька попередніх етапів розроблення) Цей процес повинен продовжуватися доти, поки користувач не підтвердить, що запропонована йому модель повністю відповідає заданій предметній області.

3.2.2 Методи логічного проектування баз даних реляційного типу

Логічне проектування баз даних - процес конструювання загальної інформаційної моделі на основі окремих моделей даних користувачів, яка є незалежною від особливостей реально використовуваної СУБД й інших фізичних умов.

Етапами логічного проектування баз даних є:

- етап 2 - побудова і перевірка локальної логічної моделі даних для окремих зображень кожного з типів користувачів;
- етап 3 - створення і перевірка глобальної логічної моделі даних.

Етап 2. Побудова і перевірка локальної логічної моделі даних для окремих уявлень кожного з типів користувачів.

Мета. Побудова логічної моделі даних на основі концептуальної моделі даних, що відображає уявлення окремого користувача про предметну область, перевірка отриманої моделі за допомогою методів нормалізації та контролю виконання транзакцій.

На цьому етапі продовжується роботу з локальними концептуальними моделями даних, створеними на попередньому етапі. Завдання полягає в доопрацюванні цих моделей з метою видалення з них всіх елементів, що ускладнюють реалізацію цих моделей у середовищі реляційних СУБД. У результаті виконання цих дій структури концептуальних моделей даних будуть змінені так, щоб цілком відповідати вимогам реляційної моделі організації баз даних. Тому нові моделі коректніше називати логічними моделями даних. Далі створені логічні моделі даних будуть перевірені з використанням правил нормалізації й піддані контролю на можливість виконання всіх необхідних користувачам транзакцій так, як це зазначено в специфікаціях на створюваний додаток. Згодом перевірені логічні моделі даних можна буде використовувати як прототипи, якщо це буде треба.

У результаті виконання цього етапу будуть створені коректні, повні й точні моделі поглядів користувачів. Це дасть нам підґрунтя для виконання наступного етапу, що полягає в об'єднанні окремих локальних логічних моделей даних у єдину глобальну модель даних.

Цей етап включає таке.

Етап 2.1. Перетворення локальної концептуальної моделі даних у локальну логічну модель.

Етап 2.2. Визначення набору відношень, виходячи зі структури локальної логічної моделі даних.

Етап 2.3. Перевірка моделі за допомогою правил нормалізації.

Етап 2.4. Перевірка моделі у відношенні транзакцій користувачів.

Етап 2.5. Створення діаграм «сутність-зв'язок».

Етап 2.6. Визначення вимог підтримки цілісності даних.

Етап 2.7. Обговорення з кінцевими користувачами розроблених локальних логічних моделей даних.

Етап 2.1. Перетворення локальної концептуальної моделі даних і локальну логічну модель.

Мета. Доопрацювання локальних концептуальних моделей з метою видалення з них небажаних елементів і перетворення отриманих моделей у локальні логічні моделі даних.

У результаті виконання першого етапу ми одержимо набір локальних концептуальних моделей даних, що відображають уявлення окремих користувачів про предмети області. Однак ці моделі даних можуть містити деякі структури даних, реалізація яких у звичайних типах СУБД буде ускладнена. На цьому етапі подібні структури даних перетворюються в таку форму, що не викликає труднощів при їхній реалізації в середовищі наявних СУБД. Може виникнути зауваження, що ці дії не є елементом логічного проектування баз даних. Однак запропонована процедура змушує розробляти ретельніше обмірковуваний зміст кожного елемента даних, що позитивно позначається на точності відображення в моделі особливостей того чи іншого об'єкту. На цьому етапі виконуються такі дії:

- видалення зв'язків типу $M:N$;
- видалення складних зв'язків;
- видалення рекурсивних зв'язків;
- видалення зв'язків з атрибутами;
- видалення множинних атрибутів;
- повторний огляд зв'язків типу $1:1$;
- видалення надлишкових зв'язків.

Видалення зв'язків типу $M:N$. Якщо в концептуальній моделі присутні зв'язку типу $M:N$ («багато-до-багатьох»), то їх варто усунути шляхом визначення деякої проміжної сутності. Зв'язок типу $M:N$ замінюється двома зв'язками типу $1:M$, що встановлюються зі створеною проміжною сутністю.

Видалення складних зв'язків. Складним називається зв'язок, що існує між трьома і більше типами сутностей. Якщо в концептуальній моделі присутній складний зв'язок, його варто усунути за допомогою проміжної сутності. Складний зв'язок замінюється необхідною кількістю бінарних зв'язків типу $1:M$, що встановлюються зі створеною проміжною сутністю.

Видалення рекурсивних зв'язків. Рекурсивними називаються такі зв'язки, у яких сутність деякого типу взаємодіє сама із собою. Якщо концептуальна модель містить рекурсивні зв'язки, вони мають бути усунуті за допомогою визначення деякої проміжної сутності.

Видалення рекурсивного зв'язку типу $M:N$ виконується так само, як і бінарних і зв'язків типу $M:N$, описаних вище.

Видалення зв'язків з атрибутами. Якщо в концептуальній моделі присутні зв'язки, що мають власні атрибути, вони мають бути перетворені шляхом створення нової сутності.

Видалення множинних атрибутів. Множинними називають атрибути, що можуть мати одночасно кілька значень для того самого екземпляра сутності. Якщо в концептуальній моделі присутній множинний атрибут, його варто

перетворити шляхом визначення нової сутності. Наприклад, для відображення ситуації, коли те саме відділення компанії має кілька телефонних номерів, у концептуальній моделі був визначений множинний атрибут Tel_No, що відноситься до сутності Branch. Цей множинний атрибут варто видалити, визначивши нову сутність Telephone, що має єдиний простий атрибут Tel_No, і створивши новий зв'язок типу 1:M.

Повторний огляд зв'язків типу 1:1. У процесі визначення сутностей могли бути створені дві різні сутності, що насправді відображають той самий об'єкт у предметній області. У подібному випадку варто об'єднати ці дві сутності в одну. Якщо первинні ключі поєднуваних сутностей різні, вибирається один з них як первинний, а інший вказується як альтернативний ключ.

Видалення надлишкових зв'язків. Зв'язок є надлишковим, якщо та сама інформація може бути отримана не тільки через нього, але і за допомогою іншого зв'язку.

Завжди варто прагнути створювати мінімальні моделі даних, і тому, якщо надлишковий зв'язок не є очевидно необхідним, його варто видаляти. Установити, що між двома сутностями є більше одного зв'язку, досить просто. Однак з цього ще не випливає, що один із двох зв'язків обов'язково є надлишковим, оскільки обидва вони можуть відображати різні об'єднання, що реально існують в організації.

При усуненні надлишковості доступу велике значення мають тимчасові показники. Суть полягає в тому, що при усуненні надлишковості дуже важливо досліджувати значення кожного зі зв'язків, що існують між сутностями.

По завершенні цього етапу отримаємо спрощену локальну концептуальну модель тих, з якої вилучені всі структури, реалізація яких у середовищі реляційних СУБД є складною.

Тому на цьому етапі поліпшену локальну концептуальну модель називають локальною логічною моделлю даних.

Етап 2.2. Визначення набору відношень, виходячи зі структури локальної логічної моделі даних.

Мета. Визначення набору відношень на основі локальної логічної моделі даних.

На цьому етапі на основі створених локальних логічних моделей даних визначаємо множину відношень, необхідну для подання сутностей і зв'язків, що містяться в зображеннях окремих користувачів про предметну область.

Для опису структури створюваних відношень можна скористатися мовою DBDL (Database Definition Language), що широко використовується у реляційних СУБД. Опис відношення мовою DBDL починається з присвоєння

йому імені, за яким слідує вміщений у дужки список імен його простих полів. Потім указується первинний ключ відношення і всі його альтернативні і/чи зовнішні ключі. Після опису зовнішніх ключів може вказуватися посилавний первинний ключ, якщо такий є.

Розглянемо, як відношення, що відображають сутності і зв'язки між ними, можуть бути створені виходячи з наявних у логічній моделі різних припустимих структур даних.

Зв'язки, які сутність має з іншими типами сутностей, подаються за допомогою механізму первинних і зовнішніх ключів. Для ухвалення рішення про те, де взяти і куди помістити значення атрибута(ів) зовнішнього ключа, попередньо варто встановити, яка із сутностей, що беруть участь у зв'язку, є батьківською, а яка - дочірньою. Батьківською вважається сутність, що передає копію набору значень свого первинного ключа у відношення, що відображає дочірню сутність, де ці значення будуть відігравати роль зовнішнього ключа.

Сильні типи сутностей. Для кожної сильної (регулярної) сутності в локальній моделі даних створюється відношення, що включає всі прості атрибути цієї сутності. У випадку складених атрибутів (наприклад, адреси) у відношення включаються тільки складові їхніх простих атрибутів (такі, як вулиця, місто, поштовий індекс).

Слабкі типи сутностей. Для кожної слабкої сутності, що присутня у логічній моделі, створюється відношення, що включає всі прості атрибути цієї сутності. Додатково у відношення включається атрибут зовнішнього ключа, що відповідає первинному ключу сутності-власника. Первинний ключ слабкої сутності частково чи повністю виводиться із ключа сутності-власника.

Отже, до кінця цього етапу ми повинні визначити будь-які первинні і потенційні ключі, що можуть знадобитися для побудови множини відношень, необхідної для повного подання вихідної логічної моделі даних.

Бінарні зв'язки типу «один-до-одного» (1:1). Для кожного присутнього в логічній моделі даних бінарного зв'язку типу 1:1, встановленого між сутностями E1 і E2, ми повинні переслати атрибути первинного ключа сутності E1 у відношення, що подає сутність E2. Ці атрибути будуть використовуватися в ньому як зовнішній ключ. Визначення батьківської та дочірньої сутностей залежить від обмежень участі, накладених на екземпляри відношення E1 і E2. Сутність, що частково бере участь у зв'язку, визначається як батьківська, а та сутність, що бере участь у зв'язку повністю (тотально), визначається як дочірня. Як уже зазначалося вище, копія набору значень первинного ключа батьківської сутності міститься у відношенні, що відображає дочірню сутність. Відзначимо, що в тому випадку, коли обидва види сутностей беруть участь у зв'язку типу 1:1 або тотально, або частково, вибір батьківської та дочірньої

сутностей може виконуватися довільно. Понад це, якщо обидві сутності беруть участь у зв'язку тотально, можна (на вибір) або подати цей зв'язок за допомогою пари первинного і зовнішнього ключів (як описувалося вище), або злити атрибути обох сутностей у єдине відношення. Злиття в єдине відношення переважніше в тому випадку, якщо ці сутності не приймають участі в інших типах зв'язків.

Бінарні зв'язки типу «один-до-багатьох» (1:M). Для кожного бінарного зв'язку типу 1:M, встановленого в логічній моделі даних між сутностями E1 і E2, необхідно помістити копію атрибутів первинного ключа сутності E1 у відношення, що відображає сутність E2, де вони будуть відігравати роль зовнішнього ключа. Сутність, що подає «єдиничну» сторону зв'язку визначається як батьківська, а сутність, що подає «множинну» сторону, - як дочірня. Як і в попередньому випадку, для подання цього зв'язку необхідно скопіювати первинний ключ батьківської сутності у відношення, що відображає дочірню сутність, де цей ключ описується як зовнішній.

Зв'язку типу «суперклас/підклас». Для кожного присутнього в логічній моделі даних зв'язку типу «суперклас/підклас» сутність суперкласу необхідно визначити як батьківську, а сутність підкласу - як дочірню. Існують різні варіанти подання подібних зв'язків у вигляді одного чи декількох відношень. Вибір найпридатнішого варіанта залежить від обмежень участі і перетинання, накладених на учасників зв'язку типу «суперклас/підклас».

Найдоцільнішим рішенням для зв'язку типу «суперклас/підклас» з тотальними і непересічними членами є подання кожного з підкласів у вигляді окремого відношення, що містить копію первинного ключа суперкласу. Однак на остаточне рішення можуть вплинути й інші фактори - зокрема те, включені чи ні окремі підкласи в різні зв'язки.

Документування створених відношень і атрибутів зовнішніх ключів. Документування складу відношень, створених на базі кожної з логічних моделей даних, здійснюється мовою DBDL. Відзначимо, що синтаксис мови DBDL може бути розширений з метою вказання обмежень цілісності для зовнішніх ключів (етап 2.6) Крім того, варто оновити вміст словника даних для відображення відомостей про будь-які нові ключові атрибути, визначені на цьому етапі.

Етап 2.3. Перевірка моделі за допомогою правил нормалізації.

Мета. Перевірка локальної логічної моделі даних з використанням технології нормалізації.

Нормалізація використовується для поліпшення моделі даних, задоволення нею обмежень, вилучення небажаного дублювання даних. Нормалізація гарантує, що отримана в результаті її застосування модель даних

буде найкраще відображати особливості використання інформації користувачами, не міститиме протиріч, матиме мінімальну надлишковість і максимальну стійкість.

Нормалізація є процедурою прийняття рішень про те, які саме атрибути мають бути об'єднані для подання сутностей кожного типу. В одній з фундаментальних концепцій теорії нормалізації стверджується, що атрибути мають бути згруповані у відношенні відповідно до наявних між ними логічних зв'язків. У деяких випадках мають місце твердження, що нормалізація розроблюваних баз даних не дає змоги досягти максимальної продуктивності при їхньому опрацюванні. На це можна відповісти такими зауваженнями.

- Нормалізація проекту дає змогу організувати розміщення даних відповідно до їхніх функціональних залежностей. Тому ця процедура повинна виконуватися між етапами концептуального і фізичного проектування.

- На етапі логічного проектування не ставиться за завдання досягти остаточного вигляду проекту. Мета цього етапу полягає в наданні проектувальнику ґрунтовнішого розуміння природи і призначення даних, що використовуються в предметній області. Якщо до застосування подаються специфічні вимоги щодо його продуктивності, то вони повинні враховуватися на етапі фізичного проектування. У цьому випадку одним з підходів є денормалізація визначених таблиць. Однак це не означає, що час на їхню нормалізацію було витрачено, оскільки для правильного виконання нормалізації проектувальник повинен глибоко вивчити семантику й особливості використання даних.

- Нормалізація проекту дає змогу підвищити його стійкість і позбутися від аномалій відношення.

- За останніх кілька років потужність комп'ютерів істотно зросла. Тому в деяких випадках може бути ділком обірунтованим рішення реалізувати проект, що дає змогу спростити роботу з даними за рахунок виконання деякого обсягу додаткового опрацювання.

- Виконання нормалізації вимагає від розроблювача повного розуміння призначення кожного атрибута, який присутній у створюваній базі даних.

- У результаті застосування нормалізації розроблювач одержує дуже гнучкий проект бази даних, що дає змогу легко вносити в неї необхідні розширення.

На попередньому етапі був створений набір відношень, що реалізують локальні логічні моделі даних. На цьому етапі проаналізуємо коректність об'єднання атрибутів у кожному з відношень. Іншими словами, наше завдання полягає в перевірці коректності складу кожного зі створених відношень за

допомогою застосування до них процедур нормалізації. Процес нормалізації включає такі чотири основні етапи:

- приведення до першої нормальної форми (1НФ), що дає змогу видалити з відношення повторювані групи атрибутів;
- приведення до другої нормальної форми (2НФ), що дає змогу усунути часткову залежність атрибутів від первинного ключа;
- приведення до третьої нормальної форми (3НФ), що дає змогу усунути транзитивну залежність атрибутів від первинного ключа;
- приведення до нормальної форми Бойса-Кодда (НФБК), що дає змогу видалити функціональних залежностей аномалії, що залишилися.

Метою виконання цих етапів є одержання гарантій того, що кожне з відношень, створених на підставі логічної моделі даних, відповідає, принаймні, вимогам НФБК. Якщо будуть знайдені відношення, що не відповідають вимогам НФБК, це може вказувати на те, що частина логічної моделі даних невірна або перетворення логічної моделі в множину відношень виконано некоректно. При необхідності буде потрібно перебудувати модель даних і переконатися, що вона вірно відображає модельовану частину інформаційної структури предметної області.

Етап 2.4. Перевірка моделі на виконання транзакцій користувачів.

Мета. Переконатися в тому, що локальна логічна модель даних дає змогу виконати всі транзакції, що передбачені користувачем.

Метою виконання цього етапу є перевірка локальної логічної моделі даних на можливість виконання всіх транзакцій, що передбачені користувачем. Перелік транзакцій визначається у відповідності зі специфікаціями, що описують дії, виконувані певним користувачем. Використовуючи ER-діаграми, словник даних і встановлені зв'язки між первинними і зовнішніми ключами, зазначені в описі відношень, спробуємо виконати всі необхідні операції доступу до даних вручну. Якщо вдасться подібним чином знайти спосіб виконання всіх необхідних транзакцій, то на цьому перевірка логічної моделі даних буде завершена. Однак, якщо яку-небудь із транзакцій виконати вручну не вдасться, то побудована модель даних є неадекватною і містить помилки, які треба усунути. Ймовірніше, помилка буде пов'язана із пропущенням у моделі даних сутності, зв'язку або атрибута.

Розглянемо два можливих підходи, завдяки яким ми зможемо переконатися, що локальна логічна модель даних дає змогу виконати всі необхідні транзакції. Перший підхід передбачає виконання перевірки того, що така логічна модель надає всю інформацію (сутності, зв'язки і їхні атрибути), необхідну для виконання кожної із транзакцій. Практично це реалізується при описі вимог, щодо виконання кожної із транзакцій.

Другий підхід до перевірки моделі даних на відповідність необхідним транзакціям полягає в нанесенні безпосередньо на ER-діаграми всіх шляхів, що будуть потрібні для виконання кожної із транзакцій. Цей підхід дає змогу візуально виділити ті області моделі, що не використовуються для виконання транзакцій, а також ті області, що істотніші з погляду виконання транзакцій. Ми маємо зручний засіб прямого аналізу підтримки, яку модель даних забезпечує для виконання необхідних транзакцій користувача. Якщо на діаграмі є області, що не використовуються в жодній із транзакцій, виникає питання про доцільність подання цієї інформації в моделі даних. У той же час, якщо в моделі присутні області, що не дають змогу знайти придатний метод виконання деякої транзакції, необхідно буде здійснити аналіз того, які з обов'язкових для виконання транзакції сутності чи зв'язки були пропущені при складанні моделі.

Етап 2.5. Створення діаграм «сутність-зв'язок».

Мета. Створення остаточного варіанту діаграм «сутність-зв'язок» (ER-діаграм), що є локальним логічним поданням даних, що використовуються окремими користувачами додатка.

Тепер все готово для того, щоб створити остаточні варіанти ER-діаграм для окремих уявлень кожного з користувачів. Дані на цих діаграмах були перевірені із застосуванням методів нормалізації, а також проконтрольовані щодо можливості виконання всіх необхідних транзакцій.

Етап 2.6. Визначення вимог підтримки цілісності даних.

Мета. Визначення обмежень, що накладаються вимогою збереження цілісності даних на уявлення користувачів.

Обмеження цілісності даних є такими обмеженнями, що вводяться з метою запобігання внесення в базу суперечливих даних. На цьому етапі ми займаємося проектуванням тільки на верхньому рівні, де розгляд питань цілісності даних є обов'язковою умовою, не пов'язаною з аспектами конкретної реалізації. Повне і точне відображення уявлення користувача ми зможемо одержати тільки після визначення обмежень, необхідних з погляду збереження цілісності даних. При необхідності на основі локальної логічної моделі можна навіть створити попередній фізичний проект бази даних, що згодом може послужити прототипом системи для основного користувача.

Розглянемо п'ять типів обмежень цілісності даних:

- обов'язкові дані;
- обмеження для доменів атрибутів;
- цілісність сутностей;
- посилавна цілісність;
- вимоги певного підприємства.

Обов'язкові дані. Деякі атрибути завжди повинні містити одне з дозволених значень. Тобто, ці атрибути не можуть мати порожнього значення. Так, кожен працівник повинен займати ту чи іншу посаду (наприклад, «керівника» чи «референта»).

Ці обмеження мають фіксуватися при внесенні даних про атрибут у словник даних (етап 1.3).

Обмеження для доменів атрибутів. Кожен атрибут має домен, що є множиною його допустимих значень. Наприклад, атрибут «стать» може містити одне з двох можливих значень - «ч» або «ж», тому його домен складається з двох символічних рядків довжиною в один символ, що містять вказані значення.

Такі обмеження встановлюються при визначенні доменів атрибутів, що відображені у моделі даних (етап 1.4).

Цілісність сутностей. Первинний ключ будь-якої сутності не може містити порожнього значення. Подібні обмеження мусять враховуватися при визначенні первинних ключів для сутностей кожного типу (етап 1.5).

Посилальна цілісність. Зовнішній ключ зв'язує кожен рядок дочірнього відношення з тим рядком батьківського відношення, що містить це саме значення відповідного потенційного ключа. Поняття посилальної цілісності означає, що якщо зовнішній ключ містить деяке значення, то воно обов'язково повинно бути присутнім у потенційному ключі одного з рядків батьківського відношення.

Існує кілька важливих моментів, пов'язаних з використанням зовнішніх ключів. По-перше, варто проаналізувати, чи припустиме використання в зовнішніх ключах порожніх значень. У загальному випадку, якщо участь дочірнього відношення в зв'язку з тотальним, то рекомендується заборонити використання порожніх значень у відповідному зовнішньому ключі. У той же час, якщо участь дочірнього відношення в зв'язку є частковим, то зазначення порожніх значень в атрибуті зовнішнього ключа має бути дозволено.

Наступна проблема пов'язана з організацією підтримки посилальної цілісності. Реалізація цієї підтримки здійснюється за допомогою завдання обмежень існування, що визначають умови, при яких може додаватися, оновлюватися або видалятися кожне значення потенційного чи зовнішнього ключа.

Випадок 1. Додавання нового рядка в дочірнє відношення. Для забезпечення посилальної цілісності необхідно переконатися, що значення атрибута зовнішнього ключа нового рядка відношення дорівнює порожньому значенню або деякому конкретному значенню, що міститься в одному з рядків відношення.

Випадок 2. Видалення рядка з дочірнього відношення. При видаленні рядка з дочірнього відношення ніяких порушень посилальної цілісності не відбувається.

Випадок 3. Відновлення зовнішнього ключа в рядку дочірнього відношення. Для збереження посилальної цілісності необхідно переконатися, що атрибут в оновленому рядку відношення містить або порожнє значення, або деяке конкретне значення, що міститься в одному з рядків відношення.

Випадок 4. Додавання рядка в батьківське відношення. Додавання рядка в батьківське відношення не може викликати порушення посилальної цілісності. Доданий рядок просто стає батьківським об'єктом, що не має дочірніх об'єктів.

Випадок 5. Видалення рядка з батьківського відношення. При видаленні рядка з батьківського відношення посилальна цілісність буде порушена в тому випадку, якщо в дочірньому відношенні будуть існувати рядки, що посилаються на вилучений рядок батьківського відношення. Наприклад, посилальна цілісність буде порушена, якщо вилучений працівник відповідав за один чи більше об'єктів нерухомості. У цьому випадку може бути використана одна із таких стратегій.

NO ACTION. Видалення рядка з батьківського відношення забороняється, якщо в дочірньому відношенні існує хоча б один рядок, що посилається на нього.

CASCADE. При видаленні рядка з батьківського відношення автоматично видаляються всі рядки дочірнього відношення, що посилаються на нього. Якщо рядки дочірнього відношення, що видаляються, виступають як батьківська сторона в деякому іншому зв'язку, то операція видалення застосовується до всіх рядків дочірнього відношення цього зв'язку, і т.д.. Тобто, видалення рядка батьківського відношення автоматично розповсюджується на будь-які дочірні відношення.

SET NULL. При видаленні рядка з батьківського відношення у всіх рядках дочірнього відношення, що посилаються на нього, в атрибут зовнішнього ключа записується порожнє значення. Отже, видалення рядків з батьківського відношення викликає занесення порожнього значення у відповідний атрибут рядків дочірнього відношення. Ця стратегія може використовуватися тільки в тих випадках, коли в атрибут зовнішнього ключа дочірнього відношення дозволяється заносити порожні значення.

SET DEFAULT. При видаленні рядка з батьківського відношення в атрибут зовнішнього ключа всіх рядків дочірнього відношення, що посилаються на нього, автоматично записується значення, зазначене для цього атрибута як значення за замовчуванням. Отже, видалення рядка з батьківського

відношення викликає запис прийнятого за замовчуванням значення в атрибут зовнішнього ключа всіх рядків дочірнього відношення, що посилаються на вилучений рядок. Ця стратегія застосовується тільки в тих випадках, коли атрибуту зовнішнього ключа дочірнього відношення присвоюється деяке значення, прийняте за замовчуванням, і NO CHECK. При видаленні рядка з батьківського відношення ніяких дій щодо збереження посилальної цілісності даних не відбувається.

Випадок 6. Відновлення первинного ключа в рядку батьківського відношення. Якщо значення первинного ключа деякого рядка батьківського відношення буде оновлено, порушення посилальної цілісності буде мати місце в тому випадку, якщо у дочірньому відношенні існують рядки, що посилаються на вихідне значення первинного ключа. Для збереження посилальної цілісності може використовуватися кожна з описаних вище стратегій. При використанні стратегії CASCADE відновлення значення первинного ключа в рядку батьківського відношення буде відображено в будь-якому рядку дочірнього відношення, що посилається на даний рядок (каскадним чином).

Вимоги певного підприємства. Необхідно проаналізувати обмеження, що називаються обмеженнями підприємства (або бізнес-правилами). Наприклад, відновлення сутностей може регламентуватися прийнятими на підприємстві правилами, що описують методи виконання транзакцій, пов'язаних з подібними відновленнями.

Документування всіх обмежень цілісності даних. Збережіть дані про всі встановлені обмеження цілісності даних у словнику даних. Вони будуть потрібні на етапі фізичної реалізації бази даних.

Етап 2.7. Обговорення з кінцевими користувачами розроблених локальних логічних моделей даних.

Мета. Переконатися, що створені локальні моделі даних точно відображають бачення користувачів предметної області.

На цей момент робота над локальними моделями даних, що відображають уявленн конкретних користувачів, має бути закінчена і повністю відображена в документації. Однак, перш ніж другий етап розроблення буде вважатися цілком завершеним, необхідно обговорити з користувачами створені логічні моделі даних і всю супровідну документацію.

Додатковим методом перевірки локальних логічних моделей даних є використання діаграм потоків даних.

Взаємозв'язок між логічними моделями даних і діаграмами потоків даних. Логічна модель даних відображає структуру даних підприємства, які підлягають зберіганню. Діаграма потоків даних (Data Flow Diagram - DFD) відображає переміщення даних у межах підприємства та збереження їх у

сховищі інформації. Всі атрибути, що зберігаються на підприємстві, мають бути оголошені в одному з типів сутностей і, ймовірно знайти своє місце в потоках даних, що переміщаються в межах підприємства. Якщо обидві ці технології використовуються для моделювання вимог користувачів, кожна них може застосовуватися для контролю погодженості та повноти іншої. Правила, що визначають співвідношення між цими двома технологіями, такі:

- кожне сховище даних має подавати всю множину типів сутностей;
- атрибути в потоках даних повинні належати до сутності того чи іншого типу.

Етап 3. *Створення і перевірка глобальної логічної моделі даних.*

Мета. *Об'єднання окремих локальних логічних моделей даних у єдину глобальну логічну модель даних, яка відображає ту частину предметної області, що охоплюється додатком.*

На цьому етапі фази логічного проектування баз даних будується глобальна логічна модель даних, яка створюється за допомогою злиття окремих логічних моделей даних, що відображають уявлення кожного користувача. Після завершення об'єднання локальних моделей необхідно перевірити правильність отриманої глобальної моделі як щодо правил нормалізації, так і щодо можливості виконання транзакцій, передбачених специфікаціями на функції окремих користувачів. Перевірка виконується з використанням тих самих методів, що застосовувалися при виконанні етапів 2.3 і 2.4. Однак виконання нормалізації буде потрібно тільки в тому випадку, якщо в процесі злиття були внесені зміни до складу окремих типів сутностей. Аналогічно, перевірка можливості виконання транзакцій виконується тільки для тих частин моделі, що були піддані змінам у ході злиття. У великих системах подібний підхід дає змогу істотно скоротити обсяг необхідних повторних перевірок.

Хоча передбачається, що кожна окрема логічна модель даних є коректною, повною і несуперчливою, будь-яка з них відображає лише сприйняття системи окремим користувачем або групою користувачів. Кожна модель відображає не функції підприємства, а лише уявлення про функції підприємства окремих його працівників. Тому кожна з цих моделей є неповною. А це означає, що між окремими моделями в повному наборі уявлень можуть існувати несумісності і взаємне перекриття. Отже, при злитті локальних моделей даних у єдину глобальну модель необхідно докладати зусиль для усунення конфліктів між окремими поданнями і брати до уваги їхнє можливе перекриття.

Процедуру злиття можна вважати найважливішою в ході логічного проектування бази даних, оскільки за її допомогою створюється уявлення, що

не залежить від будь-яких конкретних користувачів, бізнес-процесів або додатків. Цей етап передбачає виконання таких дій.

Етап 3.1. Злиття локальних логічних моделей даних у єдину глобальну модель даних.

Етап 3.2. Перевірка глобальної логічної моделі даних.

Етап 3.3. Перевірка можливостей розширення моделі в майбутньому.

Етап 3.4. Створення остаточного варіанта діаграми «сутність-зв'язок».

Етап 3.5. Обговорення глобальної логічної моделі даних з користувачами.

Етап 3.1. Злиття локальних логічних моделей даних у єдину глобальну модель даних.

Мета. Об'єднати окремі локальні логічні моделі даних у єдину глобальну логічну модель даних підприємства.

У невеликих системах, що нараховують два-три користувацькі зображення з незначною кількістю типів сутностей і зв'язків, задача порівняння локальних моделей із наступним злиттям і усуненням будь-яких можливих протиріч є відносно нескладною. Однак у великих системах треба використовувати системний підхід. Нижче запропоновано один з подібних підходів, який можна використовувати для виконання злиття локальних моделей і вирішення будь-яких проблем, що при цьому виникають. Запропонований підхід передбачає виконання таких дій.

- 1) Аналіз імен сутностей та їхніх первинних ключів.
- 2) Аналіз імен зв'язків.
- 3) Злиття спільних сутностей з окремих локальних моделей.
- 4) Включення (без злиття) сутностей, унікальних для кожного локального зображення.
- 5) Злиття спільних зв'язків з окремих локальних моделей.
- 6) Включення (без злиття) зв'язків, унікальних для кожного локального зображення.
- 7) Перевірка наявності пропущених сутностей і зв'язків.
- 8) Перевірка коректності зовнішніх ключів.
- 9) Перевірка дотримання обмежень цілісності.
- 10) Виконання креслення глобальної логічної моделі даних.
- 11) Оновлення документації.

Імовірно, найпростіший метод злиття декількох локальних моделей даних у єдину модель полягає в злитті двох локальних моделей в одну загальну модель, з наступним додаванням до неї третьої локальної моделі (і т.д.) Процес додавання до попереднього результату чергової локальної моделі буде продовжуватися доти, поки всі моделі не будуть злиті в єдину глобальну

модель. Цей підхід можна вважати простішим, ніж спроба злити всі локальні моделі за одну операцію.

Дуже важливо відзначити, що перш ніж приступати до створення глобального подання інформаційної моделі підприємства, варто переконатися, що кожна з локальних логічних моделей, що зливаються, була створена відповідно до послідовності дій, передбачених першим і другим етапами розроблення, що визначені цією методологією проектування баз даних.

Аналіз імен сутностей і їхніх первинних ключів. Може виявитися корисним по передній аналіз імен сутностей локальних моделей даних, – ці відомості можна знайти в словнику даних. Проблеми мають місце у випадках:

- якщо дві чи більше сутностей мають те саме ім'я, але насправді відрізняються один від одної;
- якщо дві чи більше сутностей ідентичні, але мають різні імена.

Для виявлення можливих проблем варто порівняти між собою властивості сутностей кожного типу. Зокрема, знайти еквівалентні сутності з різними іменами допоможе порівняння їхніх первинних ключів.

Аналіз імен зв'язків. Виконувані дії аналогічні описаним на попередньому кроці. Злиття спільних сутностей з окремих локальних моделей. Тепер варто проаналізувати імена і вміст сутностей кожного типу у логічних моделях, що зливаються. Зазвичай ця процедура включає такі дії:

- злиття сутностей з однаковими іменами і первинними ключами;
- злиття сутностей з однаковими іменами, але з різними первинними ключами;
- злиття сутностей з різними іменами, що мають однакові чи різні первинні ключі.

Злиття сутностей з однаковими іменами і первинними ключами. Як правило сутності з тим самим первинним ключем подають той самий об'єкт реального світу отже, повинні бути злиті. Об'єднана сутність буде включати всі атрибути сутностей, що зливаються, за винятком тих, що дублюються.

Злиття сутностей з однаковими іменами, але з різними первинними ключами. У деяких ситуаціях можуть бути виявлені дві сутності з тим самим ім'ям, у яких використовуються різні первинні ключі, але вони мають однакові потенційні ключі. У цьому випадку сутності зливаються аналогічно до того, як це було зроблено в попередньому варіанті. Додатково треба вибрати в сутності первинний ключ, оголосивши всі інші ключі альтернативними.

Злиття сутностей з різними іменами, що мають однакові чи різні первинні ключі. У деяких випадках можна знайти сутності, що мають різні імена, але призначені для однієї і тієї ж мети. Подібні еквівалентні сутності можна розпізнати за їхніми іменами, що будуть вказувати на їхнє подібне

призначення, за їх змістом і за їх первинним ключами. Крім того, їх можна розпізнати за участю у визначених зв'язках. Типови прикладом подібної ситуації є наявність у моделях сутностей з назвами ПЕРСОНАЛ ПРАЦІВНИК, що за суттю, є еквівалентними і мають бути злиті в єдину сутність.

Включення (без злиття) сутностей, унікальних для кожного локального зображення. На попередньому етапі були виділені всі сутності, що описують подібні об'єкти. Всі інші сутності просто включаються в глобальну модель без внесення яких-небудь змін.

Злиття спільних зв'язків з окремих локальних моделей. На цьому етапі аналізуються імена і призначення кожного зі зв'язків у всіх зображеннях окремих користувачів, ніж поєднувати зв'язки, дуже важливо вирішити будь-які конфлікти, що можуть мати місце між ними, - наприклад, щодо обмеження участі чи кардинальності. Виконуванні на цьому етапі дії включають злиття зв'язків з однаковими іменами і призначенням, після чого може знадобитися виконати злиття зв'язків з різними іменами, але які мають те саме призначення.

Включення (без злиття) зв'язків, унікальних для кожного локального зображення. На попередньому етапі були виявлені і злиті всі зв'язки, що мали подібне призначення (за визначенням, ці зв'язки повинні існувати між тими самими сутностями, які повинні бути об'єднані одна з одною). Усі зв'язки, що залишилися, включаються в глобальну модель без будь-яких змін.

Перевірка на наявність пропущених сутностей і зв'язків. Імовірно, однією з трудомісткіших задач при створенні глобальної моделі даних є задача виявлення пропущених сутностей і зв'язків між елементами уявлень різних користувачів. Якщо на підприємстві існує корпоративна модель даних, вона може використовуватися для виявлення сутностей і зв'язків між елементами уявлень різних користувачів, яких немає в жодному з локальних уявлень. У той же час, при здійсненні опитувань користувачів конкретного уявлення як превентивний захід варто попросити їх приділити деяку увагу сутностям і зв'язкам, що, на їхню думку, можуть існувати в інших уявленнях. Крім того, при аналізі атрибутів сутностей кожного типу можна спробувати виділити посилання на сутності, що належать іншим користувальницьким уявленням. Досить часто виявляється, що атрибут, пов'язаний з тією чи іншою сутністю в уявленні одного користувача, відповідає первинному ключу, альтернативному ключу чи навіть простому, не ключовому атрибуту деякої сутності з іншого уявлення.

Перевірка коректності зовнішніх ключів. На цьому кроці може здійснюватися злиття різних сутностей і зв'язків, зміна первинних ключів і визначення нових зв'язків, треба переконатися, що зовнішні ключі в дочірніх

сутностях як і раніше є коректними і, в разі потреби, внести у модель усі необхідні зміни.

Перевірка дотримання обмежень цілісності. Треба переконатися, що встановлені для глобальної логічної моделі обмеження цілісності даних не вступають у протиріччя з тими обмеженнями, що були встановлені для кожного із зображень користувача. Будь-які конфлікти варто усувати за допомогою консультацій з користувачами.

Виконання креслення глобальної логічної моделі даних. На цьому кроці рисується остаточний варіант ER-діаграми, що подає глобальну логічну модель даних, отриману в результаті злиття всіх локальних моделей.

Оновлення документації. Оновлення документації виконується з метою відображення будь-яких змін, внесених у процесі створення глобальної логічної моделі даних з множини окремих користувальницьких уявлень. Дуже важливо, щоб документація завжди підтримувалася в актуальному стані і точно відображала поточний стан моделі даних. Якщо згодом у модель будуть вноситися зміни, або в ході фізичної реалізації бази даних або в процесі її супроводу, усі внесені зміни також повинні негайно фіксуватися в документації. Застаріла документація часто є джерелом прикрих помилок і є причиною прийняття невірних рішень.

Етап 3.2. Перевірка глобальної логічної моделі даних.

Мета. *Перевірка глобальної логічної моделі даних за допомогою методів нормалізації і контроль можливості виконання необхідних транзакцій.*

На цьому етапі виконуються дії, аналогічні до тих, що виконувалися на етапах 2.2...2.4 при перевірці кожної з локальних логічних моделей даних.

Етап 3.3. Перевірка можливостей розширення моделі у майбутньому.

Мета. Визначення ймовірності внесення яких-небудь істотних змін у створену модель даних у недалекому майбутньому й оцінка того, наскільки модель пристосована для цього.

Дуже важливо, щоб створена глобальна модель була легкою для розширення. Які модель зможе підтримувати тільки поточні вимоги, то час її існування буде дуже обмеженим, а для реалізації підтримки нових вимог, що змінюються, треба буде докладати значних зусиль. Необхідно так побудувати модель, щоб вона була легкою для розширення і давала змогу реалізувати підтримку нових вимог з мінімальними змінами в робі користувачів.

Отже, має сенс виконати перевірку створеної моделі з погляду ефективності реалізації нових вимог, які можуть виникнути у майбутньому. Однак не треба вносити в модель будь-які зміни, поки вони не будуть схвалені користувачами.

Етап 3.4. Створення остаточного варіанта діаграми «сутність-зв'язок».

Мета. Створення остаточного варіанта діаграми «сутність-зв'язок», що відображає глобальну логічну модель даних предметної області.

Завершивши всі перевірки створеної глобальної логічної моделі, можна приступити до підготовки остаточного варіанта ER-діаграми. Ця діаграма повинна подавати глобальну логічну модель даних тієї частини предметної області, яка моделюється в такій додатку. Документація, що описує цю модель, (включаючи схему відношень і словни даних) має бути оновлена і підготовлена в повному обсязі.

Етап 3.5. Обговорення з користувачами глобальної логічної моделі дані

Мета. Переконатися, що створена глобальна логічна модель даних адекватно відображає частину інформаційної структури підприємства, що моделюється.

Глобальна логічна модель даних підприємства до цього моменту має бути повніс завершеною й перевіреною. Сама модель і документація до неї надаються для перегляду й аналізу кінцевим користувачам, які повинні переконатися, що вона точно відображає структуру і функціонування підприємства.

4 ЗАСОБИ ПРОЕКТУВАННЯ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ

Обґрунтування вибору програмних, системних та допоміжних засобів, які використовуються в роботі, для розроблення прикладного програмного забезпечення. При розробленні застосування магістр керується вимогами до платформи, на якій реалізується застосування (сумісність, продуктивність, інтерфейси, технологічні можливості), обґрунтовує вибір операційної системи та середовища реалізації, переваги перед альтернативними варіантами. Характеристика вибраного операційного середовища подається з погляду функціонування системи прийняття рішень, яка розробляється.

4.1 Вибір та обґрунтування систем і технологій управління базами даних

База даних є однією з фундаментальних складових в архітектурі інтелектуальних систем прийняття рішень. Від способів, методів та принципів її побудови залежать склад, структура, методи і засоби реалізації інших компонентів системи. Тому вибір та обґрунтування системи чи технології управління базами даних є початковим кроком у вирішенні проблеми організації середовища розроблення та реалізації всього проекту системи. Головною метою, що досягається вирішенням цієї проблеми є визначення та вибір такої системи або технології управління базами даних, яка найбільш відповідає завданням та вимогам системи, яка розробляється, та найефективніше взаємодіє з іншим?: складовими та елементами системи. При цьому система управління базами даних має задовольняти вимогам трьох учасників процесу побудови системи прийняття рішень розробника, замовника та користувача. Вирішальним елементом у процесах обґрунтування вибору СУБД є як її функціональні, експлуатаційні, технологічні, економічні та інші характеристики, так і прогнозовані параметри бази даних системи, яка розробляється. Найсуттєвішими при визначенні та обґрунтуванні вибору цієї чи іншої системи або технології для побудови інформаційного забезпечення інтелектуальної системи прийняття рішень є, зокрема, такі чинники.

1. Простір застосування системи, яка розробляється, визначається потенційною кількістю її користувачів, кількістю та складністю завдань, які розв'язує система, обсяги значень, якими оперує система. За цими показниками інтелектуальні системи прийняття рішень можна класифікувати тобто.

– *Персональні системи.* Це комплекс засобів для забезпечення потреб одного користувача. Середовище реалізації таких систем обмежується засобами персонального комп'ютера.

– *Системи для робочих груп.* Такі системи створюються для спільної роботи невеликих об'єднань користувачів, які вирішують однотипні завдання. Як правило, вони реалізуються в середовищах локальних мереж невеликого масштабу.

– *Офісні системи.* Системи такого типу забезпечують діяльність кількісно обмеженого набору користувачів, що сумісно вирішують завдання пов'язані з різними сторонами діяльності підприємства, організації, установи тощо.

– *Корпоративні системи.* Це великі програмно-інформаційні комплекси, призначені для вирішення значної кількості різнопланових завдань, пов'язаних з усіма сторонами діяльності середніх і великих бізнес-структур, чи інших утворень значного масштабу (місто, регіон, галузь, державні служби тощо).

– *Web-системи.* Такі системи призначені для широкого доступу великої кількості користувачів у середовищі мереж Internet/Intranet/Extranet. Користувачі таких систем є незалежними один від одного, кожен з них вирішує засобами системи набір однотипних проблем для забезпечення своїх потреб.

2. Обсяг баз даних. Обсяг бази даних, визначається кількістю її складових та кількістю одиниць даних, значення яких підлягають зберіганню у базі даних. Кожна із систем управління базами даних має свої кількісні обмеження на вміст даних та кількість об'єктів, які застосовуються для їхнього зображення. Такі параметри закладаються розробниками самих СУБД і залежать від значної кількості факторів, зокрема від вимог та потреб потенційного споживача цього продукту. Орієнтовними показниками для класифікації СУБД за обсягами баз даних, доступними для реалізації в їхньому середовищах, є такі.

– Дуже малими вважають бази даних, що мають в своєму складі до 10 таблиць, до 1000 записів у таблиці.

– До категорії невеликих баз даних відносять такі, що мають до 20 таблиць, до 10000 записів у таблиці.

– Середній обсяг бази даних допускає підтримку в ній до 50 таблиць обсягом 10000-50000 записів у кожній.

– Бази даних великих обсяг складаються з понад 50 таблиць, кожна з яких може містити понад 50000 записів у своєму інформаційному наповненні.

3. Масштаб використання бази даних визначає кількість потенційних користувачів, що отримують одночасний чи розподілений в часі доступ до її інформаційного ресурсу. За цією характеристикою розрізняють бази даних таких масштабів.

- *Вбудована база даних* - організується засобами інтегрованих середовищ розроблення. Такі бази даних не є самостійними елементами системи, не мають власних засобів доступу, керування, адміністрування тощо.

- *Персональна база даних* - засіб організації інформаційних ресурсів систем, орієнтованих на потреби одного користувача.

- *Бази даних групового застосування*. Такі бази даних розробляються для систем прийняття рішень, що забезпечують колективну роботу невеликої кількості (одиниці, десятки) користувачів, що спільно вирішують множину пов'язаних задач в одній предметній області.

- *Корпоративні бази даних*. Це бази даних великого обсягу для зберігання інформаційного ресурсу множини різнопланових систем, що вирішують завдання різного змісту для забезпечення функціонування великомасштабних структур (корпорацій, концернів, регіонів, галузей тощо) з доступом значної кількості (десятки, сотні) різноманітних користувачів.

4. Середовище і платформа. Платформа визначається, виходячи з вимог до проекту системи прийняття рішень, особливостей середовища застосування, складу задач та функцій, вимог щодо інтеграції та взаємодії з іншими системами, економічних і фінансових вимог до проекту тощо. Платформа визначає не лише середовище реалізації баз даних, але й набір засобів та технологій для їх створення, підтримки, розроблення засобів застосування і порядку використання. Основними елементами платформи є апаратне забезпечення та операційна система. Платформи для організації баз даних можна поділити на такі категорії.

- *Клієнтська платформа*. Технічна база для організації клієнтської платформи реалізується на основі одного комп'ютера і може бути утворена на основі:

- комп'ютерів архітектури PC;
- комп'ютерів архітектури Apple;
- комп'ютерів архітектури Pocket PC;
- спеціалізованих комп'ютерних засобів.

Операційними системами для реалізації середовища створення і підтримки баз даних можуть виступати версії таких ОС:

- Microsoft Windows для PC;
- Linux;

- Unix;
- Mac Os;
- IBM OS/2;
- Windows для PDA тощо.
- *Серверна платформа.* Технічне забезпечення в такому разі

реалізується на основ таких комп'ютерних засобів:

- серверів архітектури PC;
- серверів архітектури MainFrame;
- серверів архітектури RISK;
- серверів архітектури SUN;
- спеціалізованих серверів.

Операційне середовище платформи реалізується на основі таких ОС:

- Windows Server різноманітних версій;
- Linux Server, Free BSD Server різноманітних версій;
- Unix;
- IBM OS/2 Server;
- Novell;
- спеціалізовані серверні операційні системи.

Вибір платформи для реалізації бази даних системи прийняття рішень визначає набір потенційних систем та засобів управління базами даних, що функціонують у вибраному середовищі.

5. Архітектура застосувань. Кожна система управління базами даних передбачає наперед визначений спосіб організації та взаємодії елементів бази даних та середовища їхнього опрацювання, що подається у вигляді архітектурних засад, на які орієнтовна СУБД. За цими принципами можна виділити такі види архітектури.

– Однокористувацька - застосовується для реалізації персональних проектів і передбачає створення локальних баз даних, що експлуатуються в монопольному нерозподілюваному режимі.

– Групова архітектура з файловим розподілом ресурсів передбачає паралельний доступ до ресурсів бази даних незначної кількості (одиниці, десятки) користувачів без застосування сервера баз даних. Як правило, бази даних, що працюють за такою архітектурою реалізуються такими самими засобами, що й однокористувацькі, а функції розподіленого доступу забезпечуються засобами операційної системи.

– Клієнт-серверна архітектура з розподілом баз даних передбачає застосування спеціалізованих серверних СУБД їхньою особливістю є можливість паралельного доступу до баз даних значної кількості користувачів

(десятки, сотні), що забезпечується сумісним функціонуванням як власних засобів СУБД, так і засобів операційної системи. Сервери баз даних надають не лише можливості забезпечення потреб користувачів в доступі до даних, але виконують і такі функції як підтримка безпеки і захисту баз даних, адміністрування баз даних, виконання запитів і серверних процедур. В залежності від обсягу функцій розрізняють такі типи архітектури "клієнт-сервер":

- з «тонким» клієнтом - основні функції системи зосереджуються на серверній складовій, клієнтська частина призначається лише для взаємодії з користувачем та відображення результатів;

- з «тонким» сервером - основні функції опрацювання даних та розв'язання задач виконуються клієнтським програмним забезпеченням, функції сервера баз даних зводяться до організації та розподілу доступу користувачів до баз даних;

- з рівномірним розподілом функцій між клієнтською та серверною частинами.

6. Функціональні можливості СУБД. У загальному випадку до завдань та функцій систем управління базами даних відносять досить широкий перелік завдань, вирішення яких вони забезпечують. Залежно від вимог, мети створення, області та масштабів застосування конкретної системи чи технології, версії випуску, варіанту комплектування тощо склад її функціональних можливостей може бути розширено чи обмежено. У загальному випадку, при виборі СУБД необхідно врахувати такі фактори.

- Наявність та можливості засобів створення та підтримки баз даних, способи їхнього застосування, рівень інтерфейсу побудови баз даних та його функціональність.

- Функції та можливості засобів адміністрування баз даних.

- Додаткові сервіси та утиліти адміністратора, проектувальника та користувача баз даних.

- Рівень підтримки і версія мови SQL, засоби формування та виконання запитів.

- Наявність та можливості засобів розроблення застосувань - вбудовані інтерпретатори чи компілятори мов програмування, інтерфейс з мовами програмування, бібліотеки API, інтеграція з середовищами розроблення, спільні платформи (наприклад, .NET) СУБД і засобів створення програм.

- Функції аналітичного застосування баз даних, зокрема, наявність засобів OLAP та їхні можливості, можливості побудови динамічних звітів,

засоби створення нестандартних запитів, наявність та можливості засобів Business Intelligence.

- Імпорт, експорт даних, інтеграція з іншими середовищами.

- Підтримка XML як в середовищі бази даних, так і на рівні інтерфейсу обміну та інтеграції з іншими системами, можливості опрацювання, перетворення XML-документів засобами СУБД та формування XML-запитів, підтримка XPAth, XQuery тощо.

- Можливості організації та функціонування СУБД у складі Web-систем, наявність можливостей і засобів публікації баз даних у мережі Internet/Intranet.

7. Експлуатаційні характеристики системи управління базами даних. Це набір її властивостей та параметрів, які визначають можливості та ефективність їхнього застосування в складі системи прийняття рішень, що розробляється. Основними з них є, зокрема, такі показники.

- Простота встановлення, впровадження, налаштування та застосування СУБД. Цей показник визначає рівень кваліфікації і вимоги до персоналу, необхідні для виконання таких функцій, а також співвідношення цих показників з аналогічними вимогами щодо інших складових системи прийняття рішень.

- Стійкість СУБД- це показник, який визначає її можливості щодо захисту даних від руйнування та втрат відновлення при збоях, засоби захисту від зовнішнього впливу та втручань.

- Підтримання механізму транзакцій - забезпечує можливості ефективної та стійкої роботи в умовах доступу до даних великої кількості користувачів та забезпечує високий рівень стійкості та надійності роботи СУБД.

- Розподілений режим роботи є необхідним для СУБД, які застосовують для побудови систем, що призначені для забезпечення потреб груп чи колективів користувачів, а також систем, орієнтованих на Web-застосування.

- Засоби безпеки доступу є необхідними у багатокористувацьких системах. Ці засоби забезпечують ідентифікацію та авторизацію користувачів, встановлення прав привілеїв та повноважень на виконання ними дій та операцій над об'єктами бази даних, планування та регламентування порядку доступу користувачів до баз даних.

- Засоби збереження/відновлення баз даних забезпечують можливості адміністратора підвищувати надійність бази даних, а також швидко відновлювати роботу системи в разі збоїв, руйнування чи втрати даних.

– Масштабованість баз даних — це можливість збільшувати обсяги значень у таблицях, кількість об'єктів бази даних, перелік задач та завдань, вирішення яких вона забезпечує без перебудови логічної та фізичної структури.

8. Умови придбання. За умовами придбання розрізняють такі способи отримання необхідного набору засобів СУБД.

– *Вільне розповсюдження* - до такої категорії належать продукти, що безоплатно поширюються на умовах open source та GNU GPL. Засоби такого типу без-будь яких обмежень можна використовувати, змінювати, поширювати з некомерційною метою.

– *Умовно вільне* поширення передбачає можливості безкоштовного отримання та використання програмного продукту з дотриманням певних умов. Такими умовами як правило є заборона змін (засоби типу as is), продажу, поширення без зазначення авторства, комерційне використання тощо.

– *Отримання безкоштовних або тестових версій* ліцензованих продуктів. Таку практику застосовують провідні виробники СУБД, зокрема, Oracle, Microsoft, Borland тощо. Основна мета такого поширення є інформування користувачів та стимулювання їх до застосування своїх продуктів. Такі версії СУБД відрізняються від комерційних, як правило, дещо обмеженим набором функцій чи засобів, підтриманням роботи меншої кількості користувачів, обмеженнями обсягів баз даних, обмеженнями термінів використання, обмеженнями на оновлення тощо. Такі версії можна досить ефективно і легально застосовувати у невеликих за обсягами, чи некомерційних проектах.

– *Поширення у складі інтегрованих пакетів*. Такі системи та технології управління базами даних поширюють як елементи технологічних комплексів призначених, для розроблення та впровадження застосувань. СУБД, зокрема, входять до складу таких інтегрованих засобів як MS Office, Open Office, Star Office, .Net Visual Studio, окремі елементи створення та підтримання баз даних такі, як Borland Database Engine, входять до складу середовищ розроблення застосувань таких як Delphi, C++ Builder тощо.

– *Отримання на умовах ліцензування* передбачає придбання ліцензії виробника на СУБД відповідної конфігурації з подальшим взаємним дотриманням умов ліцензії як споживачем так і виробником. Як правило основними перевагами придбання ліцензії на використання програмних засобів, і зокрема СУБД, є легальність, можливість застосування в комерційних проектах та поширення розроблень на основі цих СУБД, сервіс та підтримання виробником, періодичне оновлення, пільгове отримання нових версій.

9. Вартість. Вартість системи управління базами даних має бути сумірною з вартістю проекту та збалансованою з вартісними показниками

інших його складових. За вартісними характеристиками СУБД можна поділити на декілька категорій.

– *Безкоштовні СУБД*. Такі продукти поширюються як засоби категорії Freeware, Open Source, продукти за ліцензією GPL, а також як спеціальні вільні версії комерційних ліцензованих продуктів. Останні відрізняються від комерційних версій, як правило, функціональними обмеженнями, кількістю користувачів, відсутністю деяких складових, але в деяких нескладних застосуваннях можуть повноцінно виконувати покладені на них функції.

– *Бюджетні продукти* - це системи невеликої потужності та функціональних можливостей, призначені для реалізації проектів локального рівня або масштабу невеликих офісних застосувань. Вартість таких засобів не перевищує порядку 1000 -5000 грн.

– *Бізнес продукт* - це системи управління базами даних для проектів середнього масштабу. Системи такого типу орієнтовані на створення мережових вирішень для локальних мереж або невеликих Web-проектів. Вартість таких систем може коливатись в межах від 5000 до 10000 грн.

– *Корпоративні продукти* – це високотехнологічні багатофункціональні СУБД, що призначені для створення систем прийняття рішень великого масштабу, обслуговування великої кількості користувачів, забезпечення вирішення широкого кола проблем функціонування складних бізнесових, державних, регіональних та інших структур або великих Web-систем, порталів та холдингів. Вартість таких систем управління базами даних може сягати від 10000 грн до сотень тисяч грн.

4.2 Вибір та обґрунтування засобів проектування програмних систем

Сучасні CASE-засоби забезпечують підтримку різноманітних технологій проектування інформаційних систем (ІС) - від простого аналізу і документування до повномасштабних засобів автоматизації, що покривають весь життєвий цикл (ЖЦ) програмного забезпечення (ПЗ). Сучасний ринок програмних засобів нараховує близько 300 різних CASE-засобів [28, 29, 128]. Повний комплекс CASE-засобів, що забезпечує підтримку життєвого циклу ПЗ, містить такі компоненти:

– *репозиторій*, який є основою CASE-засобу. Він забезпечує збереження версій проекту і його окремих компонентів, синхронізацію

надходження інформації від команди розробників, контроль метаданих на повноту і несуперечність;

- *графічні засоби аналізу і проектування*, що забезпечують створення і редагування ієрархічно зв'язаних діаграм (потоків даних, «сутність-зв'язок» та ін.), які утворюють модель ІС;

- *засоби розроблення додатків*, включаючи мови 4GL і генератори кодів;

- *засоби конфігураційного керування*;

- *засоби документування*;

- *засоби тестування*;

- *засоби керування проектом*;

- *засоби реінжинірингу*.

Всі сучасні CASE-засоби можуть бути класифіковані за:

- типами;

- категоріями;

- застосовуваним методологіям і моделям систем і БД;

- ступенем інтегрованості із СУБД;

- доступними платформами.

Класифікація за типами відображає функціональну орієнтацію CASE-засобів на ті або інші процеси ЖЦ. Класифікація за типами в основному збігається з компонентним складом CASE-засобів і включає такі основні типи:

- *засоби аналізу* (Upper CASE), призначені для побудови й аналізу моделей предметної області (Design/IDEF, BPwin);

- *засоби аналізу і проектування* (Middle CASE), що підтримують найбільш розповсюджені методології проектування і використовуються для створення проектних специфікацій (Vantage Team Builder, Designer/2000, Silverrun, PRO-IV, CASE.Аналітик). Результатом застосування таких засобів є специфікації компонентів і інтерфейсів системи, архітектури системи, алгоритмів і структур даних;

- *засоби проектування баз даних*, що забезпечують моделювання даних і генерування схем баз даних (як правило, мовою SQL) для найбільш розповсюджених СУБД. До них відносяться ERwin, S-Designor і DataBase Designer (ORACLE) Засоби проектування баз даних є також у складі CASE-засобів Vantage Team Builder, Designer/2000, Silverrun і PRO-IV;

- *засоби розроблення додатків*. До них відносяться засоби 4GL (Uniface, JAM, PowerBuilder, Developer/2000, New Era, SQLWindows, Delphi та ін.) і генератори кодів, що входять до складу Vantage Team Builder, PRO-IV та ін.;

– *засоби реінжинірингу*, що забезпечують аналіз програмних кодів і схем баз даних і формування на їхній основі різних моделей і проектних специфікацій (відновлення моделі проекту за текстами програм).

Засоби аналізу схем БД і формування ERD входять до складу Vantage Team Builder, PRO-IV, Silverrun, Designer/2000, ERwin і S-Designor. В області аналізу програмних кодів найбільшого поширення здобули об'єктно-орієнтовані CASE-засоби, що забезпечують реінжиніринг програм мовою C++ (Rational Rose, Object Team, Borland Together). Допоміжні типи включають:

- засоби планування і керування проектом (SE Companion, Microsoft Project та ін.);
- засоби конфігураційного керування (PVCS, SCCS та ін.);
- засоби тестування (Quality Works та ін.)

Класифікація за категоріями визначає ступінь інтегрованості CASE-засобів за виконуваними функціями і включає у себе:

- локальні засоби, що вирішують невеликі автономні задачі (tools);
- набір частково інтегрованих засобів, що охоплюють більшість етапів життєвого циклу ІС (toolkit);
- цілком інтегровані засоби, що підтримують весь ЖЦ ІС і пов'язані загальним репозиторієм.

На сьогоднішній день для проектування програмного забезпечення в основному використовуються такі CASE-засоби [28]:

- Silverrun;
- Vantage Team Builder;
- Oracle Designer;
- ERwin+BPwin;
- S-Designor;
- CASE.Аналітик;
- Rational Rose;
- Borland Together.

4.3 Вимоги до реалізації

Описання програмної реалізації проекту повинно відповідати вимогам ГОСТ 19.402- 78 «ЕСПД. Описание программы» (додаток М). Цей розділ може складатися з таких підрозділів:

- алгоритм розв'язування задачі;
- вибір технології програмування;
- вибір мови програмування;

- проектування структури програми;
- проектування інтерфейсу програми;
- розроблення програми та її опис;
- інструкція користувачеві;
- контрольний приклад та аналіз комп'ютерної реалізації програми.

Алгоритм розв'язування задачі.

Алгоритм розв'язування задачі повинен задавати скінченну послідовність дискретних дій перетворення вхідних даних для досягнення результату згідно із сформульованою метою. Алгоритм може бути виконаний у вигляді покрокового мовного опису або у вигляді графічної схеми. В цьому розділі бажано навести узагальнений покроковий мовний опис алгоритму, на основі якого провести планування структури програми та розробити детальні графічні схеми її основних блоків. Для об'єктно-орієнтованих програм наводяться схеми взаємодії класів. Графічна схема алгоритму (програми, класів) повинна бути описана. В описі вказуються прийняті позначення та призначення основних ланок графічної схеми.

Вибір технології програмування.

Після розроблення узагальненого алгоритму необхідно побудувати модель проектування та реалізації програми. Кожну програму можна побудувати за допомогою різних методів програмування. Розроблення програми будь-яким методом повинно базуватися на абстракціях, які найбільш точно відповідають постановці конкретної задачі. Реалізація програми повинна являти собою сукупність завершених модулів, які можна використати для побудови інших програм. В практиці програмування найбільшого поширення здобули методи процедурного та об'єктно-орієнтованого програмування.

Методи процедурного програмування базуються на моделі побудови програми, як сукупності функцій. Функціональна декомпозиція програми визначає функції як абстрактні операції в термінах задачі, а не в деталях їх реалізації. За допомогою глобального алгоритму ці функції дозволяють розв'язувати конкретну задачу програмування. Механізм фактичних-формальних параметрів функцій та незалежність від глобальних змінних програми дає можливість використання таких функцій як абстрактних операцій в інших задачах. Файлова організація програми та засоби структурного програмування роблять реалізацію функцій зрозумілою та очевидною.

Об'єктно-орієнтоване програмування (ООП) базується на абстракціях даних. Модель абстракції даних полягає в інкапсуляції даних та операцій над ними в окремий класовий тип. Доступ до даних можливий лише за допомогою інкапсульованих операцій. Класовий тип є автономно завершеним і дозволяє повне або часткове успадкування для інших класів. ООП концентрується на

суті задачі, для якої розробляється програма. Задача будується як сукупність об'єктів, які взаємодіють між собою за допомогою повідомлень. Мета ООП полягає у визначенні найбільш вдалих об'єктових типів. Об'єктовий тип служить модулем, який може бути використаним для розв'язування інших подібних задач.

Ні одна із моделей програмування не дозволяє оптимально вирішити всі проблеми, які виникають при проектуванні програми. Тому методи програмування необхідно використовувати гнучко, враховуючи їх особливості та можливості для розв'язування конкретного типу задач.

Вибір технології програмування тісно пов'язаний із можливостями мов програмування.

Вибір мови програмування.

Мови програмування діляться на два великі класи: машинно-орієнтовані (асемблери) та алгоритмічні (машинно-незалежні мови високого рівня).

Машинно-орієнтовані мови (автокоди, асемблери, макроасемблери) тісно зв'язані з системою машинних команд конкретного процесора. Написання програми на машинній або машинно-орієнтованій мові є трудомістким процесом і вимагає високої кваліфікації програміста. Поряд з тим, потрібно відмітити, що програми на машинно-орієнтованій мові розуміються комп'ютером точно і однозначно, займають менший об'єм ОП та виконуються швидше, ніж відповідні програми на мовах високого рівня.

Алгоритмічні мови - це мови, наближені до розмовної мови зі своїми строгими правилами побудови. Основний критерій побудови мови - це однозначність його розуміння людиною та виконавчою системою. Алгоритмічні мови використовують деякі слова розмовної мови і загальноприйняті математичні символи. Це спрощує розуміння змісту дій та описів у мовних конструкціях, операторах та виразах. Основна перевага алгоритмічних мов - зручність для програміста. Програма на алгоритмічній мові пишеться швидше та легше, є більш компактною і наглядною. В ній легше проглядаються логічні зв'язки. Така програма, як правило, містить менше помилок, простіша для відлагодження, є мобільною.

У світі реально використовується декілька сотень алгоритмічних мов та їх діалектів, призначених для описання алгоритмів опрацювання даних у різних галузях науки, техніки та виробництва. Тільки біля 10 з них завоювали позиції універсальних мов, що знайшли широке розповсюдження і області застосування (наприклад, Pascal, Object Pascal, C, C++, C#, Java, Fortran, Basic, Visual Basic та ін.) Базова множина кожної мови обумовлюється її стандартом.

Мови програмування характеризуються алфавітом, синтаксисом та семантикою.

Алфавіт - це набір символів, які можуть бути використані при написанні програми.

Синтаксис - визначає правила побудови із символів алфавіту спеціальних конструкцій - операторів, за допомогою яких можна складати різноманітні алгоритми розв'язку задач.

Семантика - це система правил інтерпретації (розуміння) цих конструкцій.

В основі мови програмування лежить поняття оператора. Оператор - це самостійна одиниця мови, що описує зміст етапу алгоритмічного процесу. Оператор складається із зарезервованих у мові ключових слів та слів і виразів, що вводяться програмістом. Програма складається із послідовності операторів. Оператори виконуються один за одним в порядку слідування. Зміна послідовності виконання операторів може здійснюватися операторами переходу, циклу або виходу із програми.

Проектування структури програми.

Програма - це запис алгоритму засобами мови програмування. Програма складається із окремих операторів (команд), директив компілятора (псевдокоманд) та коментарів.

Проектування структури програми в основному визначається вибраною технологією та мовою програмування, можливостями середовища програмування і здійснюється після розроблення алгоритму розв'язування задачі. В цьому розділі необхідно визначити склад файлової системи проекту програми, вимоги до їх структури та змісту. Проектування структури програми базується на визначенні структур класів, операцій, даних та методів їх опрацювання, семантики потоків повідомлень. Спроектовану структуру програми бажано зобразити у вигляді графічної схеми. При потребі здійснюється побудова UML-діаграм відношень між класами, послідовностей взаємодії між об'єктами, кооперації об'єктів, структури та взаємодії компонентів, розгортання компонентів та ін.

Проектування інтерфейсу програми.

Після планування структури програми здійснюється проектування елементів користувацького діалогу з програмою - екранних форм, вікон, меню, блоків діалогу тощо. Проектування інтерфейсу програми здійснюється виходячи з ергономічних характеристик, простоти та компактності відображення інформації на екрані, зручності доступу до основних органів керування режимами роботи програми, забезпечення надійності роботи з програмою. Приводяться ескізи екранних форм та обґрунтування розміщення елементів діалогу на екрані. Описуються інструментальні засоби автоматизованого проектування екранних форм та їх підключення до програми.

Розроблення програми та її опис.

Після планування структури програми та розроблення її екранних форм і елементів діалогу здійснюється деталізація структури складових частин програми, які в цілому повинні виконувати розв'язування поставленої задачі та забезпечувати взаємодію з користувачем через спроектований інтерфейс екранних форм. Розроблення складових частин програми базується на вибраній технології програмування і вимагає знання можливостей мови програмування, складу та призначення бібліотечних функцій інструментального середовища програмування.

Текст програми повинен бути документований. Кожна програма на початку повинна містити інформацію про тему роботи, прізвище автора та дату створення. Оголошення класу повинно містити дані про його призначення та про призначення його елементів даних і методів. Крім того, на початку кожної підпрограми розміщується таблиця із специфікацією на використанні ідентифікатори по наступному зразку: позначення ідентифікатора, призначення у підпрограмі. Кожен логічно завершений фрагмент програми повинен мати коментар про його призначення.

Опис програми повинен бути проведений згідно такої схеми:

- назва програми;
- призначення програми;
- мови програмування, на яких написана програма;
- логічна, структура програми;
- вхідні дані;
- вихідні дані;
- програмні засоби, необхідні для реалізації програми;
- технічні засоби, необхідні для реалізації програми.

У залежності від особливостей програми дозволяється вводити додаткові або об'єднувати вказані розділи.

У розділі «Призначення програми» вказуються класи задач, які можуть бути розв'язані за допомогою розробленої програми, та відомості про функціональні обмеження на застосування.

У розділі «Мови програмування» проводиться коротка характеристика мов програмування, на яких написана програма.

У розділі «Логічна структура програми» описуються методи та алгоритми розв'язку задачі, структура програми з описом функцій її складових частин, зв'язків між ними та зв'язків програми з іншими програмами. Оскільки методи та алгоритми розв'язку задачі можуть бути описані в попередніх розділах роботи, то їх повторний опис не потрібен. В цьому розділі бажано навести узагальнену схему взаємодії підпрограм та спосіб обміну інформацією

між ними (через загальні області пам'яті, глобальні змінні, список фактичних-формальних параметрів, дискові файли, командні рядки субпроцесів та ін).

У розділах «Вхідні дані» та «Вихідні дані» вказуються способи організації даних, їх формати, оголошення, способи кодування та носії.

У розділі «Програмні засоби» перераховується програмне забезпечення, необхідне для функціонування програми - операційна система, засоби автоматизації програмування, інтегровані середовища, редактори текстів, компілятори, редактори зв'язків, завантажувачі коду програми, відлагоджувальники та ін.

У розділі «Технічні засоби» описуються структура та характеристики обчислювальної системи, необхідні для виконання програми.

Інструкція користувачеві.

В інструкції користувачеві наводять детальну послідовність дій по запуску програми на виконання та описують всі можливі режими керування роботою програми. Ілюстрування режимів роботи програми здійснюється за допомогою зображення екранних форм, вікон, меню, блоків діалогу, форм документів і т.п.

Контрольний приклад та аналіз результатів комп'ютерної реалізації програми.

Здійснюється формулювання умови прикладу, який був використаний для реалізації та відлагодження програми. Розробляються структури і формати вхідних та вихідних даних.

Проводиться аналіз результатів, отриманих в ході реалізації програми для різних тестових наборів вхідних даних. Аналіз здійснюється по кількісних та якісних характеристиках роботи програми: точності обчислення, часу розв'язування задачі, необхідному об'ємі оперативної та зовнішньої пам'яті, надійності роботи програми при зміні вхідних даних, ергономічних характеристиках спроектованого програмного інтерфейсу.

5 ПРАКТИЧНА РЕАЛІЗАЦІЯ

Розглянуто реалізацію вирішення поставленого завдання, а саме описання: складу та структури бази даних (бази знань); складу, структури, змісту і функцій розробленого програмного забезпечення та процесів їхнього сумісного функціонування; описання реалізації механізмів логічного виведення і отримання рішень; описання технологічного процесу прийняття рішень. Виконання розділу «Аналіз отриманих результатів» дає змогу оцінити практичні аспекти застосування запропонованих вирішень і зробити висновок про успіх (або неуспіх) у досягненні мети магістерської роботи. Розглянуто особливості валідації інтелектуальних систем і створення набіру контрольних прикладів для тестування таких систем.

5.1 Описання реалізації завдання

До складу цього розділу входять:

- 1) Описання складу та структури бази даних (бази знань):
 - формати, імена та середовище зберігання баз даних;
 - перелік таблиць бази даних, їхніх назв та сутностей, які вони представляють;
 - перелік полів таблиць зі зазначенням їхніх властивостей, типів, обсягів, обмежень та змісту характеристик, які вони представляють;
 - описання обмежень таблиць, правил, повноважень на виконання операцій над ними;
 - описання зв'язків в базі даних, їхній зміст, типи, обмеження, правила цілісності;
 - обґрунтування та характеристика процедур нормалізації бази даних до потрібної нормальної форми;
 - описання системи захисту та збереження даних, процедур копіювання/відновлення, дублювання, реплікації тощо;
 - кількісні та якісні показники бази даних розробленої системи прийняття рішень.
- 2) Описання складу, структури, змісту і функцій розробленого програмного забезпечення (модулів, підпрограм, функцій, форм, макросів, запитів тощо) та процесів їхнього сумісного функціонування:
 - перелік програмних одиниць, які входять до складу розробленого програмного забезпечення із зазначенням їхніх імен, характеристик та призначення;

- описання кожної програмної одиниці за схемою:
- ідентифікатор;
- призначення;
- мова реалізації;
- перелік функцій;
- вхідні дані;
- вихідні дані;
- способи виклику;
- особливості застосування;
- структурна схема зв'язків та взаємодії елементів програмного забезпечення;
- способи, регламент, графік запуску та завершення роботи системи;
- опрацювання аварійних ситуацій у функціонуванні системи - причини та способи аварійного припинення роботи, відновлення роботи системи після збоїв;
- у додатках до магістерської роботи подаються тексти програмних одиниць, які супроводжуються необхідними для їхнього розуміння поясненнями та коментарями.

3) Описання реалізації механізмів логічного виведення і отримання рішень:

- перелік та характеристика засобів, за допомогою яких реалізуються процеси, логічного виведення в прийнятті рішень;
- перелік програмних одиниць, які реалізують прийняття рішення/логічне виведення;
- способи ініціалізації процесів прийняття рішення/логічного виведення та передавання вхідних даних;
- форма подання результатів прийняття рішення/логічного виведення.

4) Описання технологічного процесу прийняття рішень:

У цьому пункті описується послідовність етапів, кроків, дій інтелектуальної системи прийняття рішень та її користувачів в процесі функціонування та розв'язання задач, визначених призначенням системи. Разом з текстом розробляється графічна схема технологічного процесу, яка має супроводжуватись необхідними поясненнями.

Магістрант розробляє такі питання:

- характеристика джерел вхідної інформації системи, їхнє описання та способи передачі вхідних даних від джерела до системи;
- порядок та процедури опрацювання вхідних даних;

- описання потоків даних в процесах формування та прийняття рішень;
- перелік дій, етапів та процедур, що реалізуються в процесах прийняття рішень;
- способи формування вихідних даних системи;
- описання отримувачів вихідної інформації (користувачі, клієнти, особи, інші системи, засоби накопичення та зберігання даних, технічні пристрої тощо) системи прийняття рішень;
- способи та формати передачі вихідної інформації отримувачам;
- способи і процедури використання вихідної інформації.

5.2 Аналіз отриманих результатів

Метою виконання розділу «Аналіз отриманих результатів» є демонстрація на базі отриманих результатів того, що запропоновані у магістерській роботі рішення дійсно мають визначені у попередніх розділах переваги та особливості. Виконання цього розділу дає змогу оцінити практичні аспекти застосування запропонованих вирішень та зробити висновок про успіх (або неуспіх) у досягненні мети магістерської роботи.

Виконання розділу «Аналіз отриманих результатів» полягає у застосуванні двох взаємодоповнюючих видів діяльності, які використовуються як у наукових дослідженнях, так і в розробленні програмних продуктів - верифікації та валідації.

Верифікація полягає у перевірці, що у проектованій системі правильно реалізовані алгоритми, що вона працює для всіх наборів вхідних даних, у ній немає програмних дефектів, система зручна у користуванні. Отже, верифікація полягає у перевірці того, що система спроектована правильно щодо до загальних вимог з проектування аналогічних систем.

Валідація полягає у перевірці того, що в результаті проектування була створена така система, яка була задумана проектувальниками. При цьому перевіряють чи були реалізовані всі початкові вимоги до системи і чи дійсно система настільки ефективна на практиці, як передбачалося. У науковій діяльності валідація полягає у порівнянні параметрів ефективності роботи створеної системи з системами - аналогами. Отже, валідація полягає у перевірці того, що в результаті створена правильна система відповідно до мети проектування.

Для здійснення верифікації та валідації системи використовують різні методи та засоби. Для верифікації застосовують загальноприйнятий набір

засобів тестування програмних та інформаційних систем. Тільки правильно верифіковану систему можна вважати закінченою та придатною до використання. Конкретний набір методів та підхід до валідації залежить від типу розробленої системи та прогнозованих її переваг порівняно з наявними системами-аналогами. Як правило, кожне твердження про перевагу розробленої системи порівняно з аналогом має бути ретельно валідовано та доведено на практиці. Загалом, виконання усіх вимог до системи та усі її переваги відносно систем-аналогів мають бути валідовані.

Нижче описано деякі підходи до верифікації та валідації й визначено сферу дії та обмеження щодо їхнього застосування.

5.2.1 Верифікація та валідація даних

Важливою частиною валідації та верифікації довільної моделі є верифікація вхідних даних. Набори даних, що використовують для моделювання, повинні відповідати меті та задачам моделювання та охоплювати усі аспекти функціонування реальної системи, що досліджуються в процесі моделювання. Крім того, повинні бути враховані обмеження на набори значень вхідних даних, що відповідають реальним умовам функціонування системи. Якщо для моделювання обрати нереальні або недостовірні вхідні дані, то й результату моделювання довіряти не можна.

Валідація даних - це процес, який гарантує, що система працює з правильними, та потрібними даними. Для проведення валідації використовують правила валідації, що перевіряють правильність або значимість даних, що надходять до системи.

Найпростіше правило валідації перевіряє, чи введені дані належать до дозволеного набору символів. Наприклад, якщо вводять телефонний номер, то дозволено водити цифри, знаки - та + а також дужки (). Складніше правило валідації перевірятиме, чи правильно введено код країни.

Для валідації даних використовують такі методи:

- перевірка формату введених даних, наприклад, перевіряють, що дату введено у форматі DD/MM/YY;
- перевірка типів даних, наприклад, якщо дозволено вводити тільки числові дані, а помилково було введено букву О замість нуля;
- перевірка того, що дані належать до дозволених діапазонів значень, наприклад, що місяць може бути числом від 1 до 12;
- перевірка чи введене значення не перевищує максимального та не менше за мінімальне;
- перевірка, чи вхідні дані наявні взагалі;

– перевірка з використанням надлишкових символів, наприклад, остання цифра в номері ISBN підраховується як сума по модулю 11 інших цифр;

– перевірка сум, наприклад, якщо у бухгалтерському документі треба перевірити наявність всіх рядків (складових загальної суми), то досить перерахувати загальну суму та порівняти зі збереженим окремо значенням;

– перевірка правопису;

– перевірка цілісності даних, наприклад, якщо до клієнта звертаються як «Пан», то його стать повинна бути «Ч».

Завдання валідації даних є складовою частиною (а коректність даних - необхідною умовою) усіх підходів до валідації, описаних нижче.

Натурне моделювання.

Натурне моделювання доцільно виконувати, якщо система, розроблена у магістерській роботі, впроваджена у виробництво й існує можливість здійснення з нею експериментів з метою визначення характеристик її роботи. В цьому випадку можна спланувати експерименти зі збору метрик, що характеризують функціонування системи, та порівняти їх з такими самими метриками інших аналогічних систем.

Підхід до організації таких експериментів, необхідні етапи та документи що потрібно створити, в значній мірі схожі на описані нижче у розділі «Тестування продуктивності». Відмінності полягають у виборі метрик та способі опрацювання результатів експериментів.

Імітаційне моделювання.

У випадку, коли немає можливості здійснити натурні експерименти з системою, альтернативним підходом до валідації системи є метод імітаційного моделювання. Цей метод застосовують і тоді, коли система є складною і не може бути описана відносно простими математичними моделями.

Центральними елементами, які використовують для валідації системи на моделях, є самі моделі. Будь-яка модель відображає тільки певні аспекти функціонування реальної системи, які вважаються важливими для досягнення мети моделювання. У випадку, якщо потрібно валідувати різні функціональні характеристики системи, може викинути необхідність у створенні декількох моделей, кожна - для валідації певної характеристики.

Зрозуміло, що необхідною умовою для валідації системи з використанням моделей є правильність самих моделей. Тому кожную модель, яку використовують для валідації, треба, своєю чергою, валідувати.

Валідувати модель - це перевірити, що комп'ютерна модель для передбаченої сфери: використання має точність достатню для досягнення мети моделювання. Отже, кожен модель розробляється з певною метою, і якщо

метою моделі є перевірити декілька тверджень, то і валідувати модель потрібно відносно кожного з цих тверджень. Звичайно, вихідні параметри моделювання (тобто такі дані, які використовуються для перевірю: тверджень) повинні бути визначені та для них необхідно визначити потрібну точність. Якщо потрібні параметри є ймовірнісними, то первинне значення для валідації набувають такі характеристики, як математичне сподівання та дисперсія.

Верифікувати модель - це перевірити, що відповідна комп'ютерна програма моделювання створена правильно.

Під час створення як концептуальної моделі, так і імітаційної комп'ютерної моделі необхідно забезпечити валідність даних моделювання.

Залежності між об'єктом моделювання, концептуальною та комп'ютерною (імітаційною і моделлю, що відображають процес створення та валідації моделі, подані на рисунку 5.1.

Для валідації моделей застосовують низку методів.

- *Анімація* - поведінка моделі відображається графічно під час процесу моделювання. Наприклад - відображається рух пакетів в інформаційній мережі, або рух складової частин продукту.

- *Порівняння з іншими моделями*. Модель валідують шляхом порівняння вихідних даних моделі з вихідними даними інших, валідних моделей.

- *Тестування «вироджених» ситуацій*. Наприклад, чи середня довжина черги на обслуговування зростатиме, якщо інтенсивність надходження завдань перевищить перепускную здатність сервісу.

- *Валідація подій*. Перевірка, чи усім подіям у реальній системі відповідають події моделі.

- *Тестування граничних значень*. Модель повинна давати правильний результат у малоімовірних ситуаціях, наприклад, якщо ціна товару дорівнює нулю, то її плата за нього повинна бути нульовою.

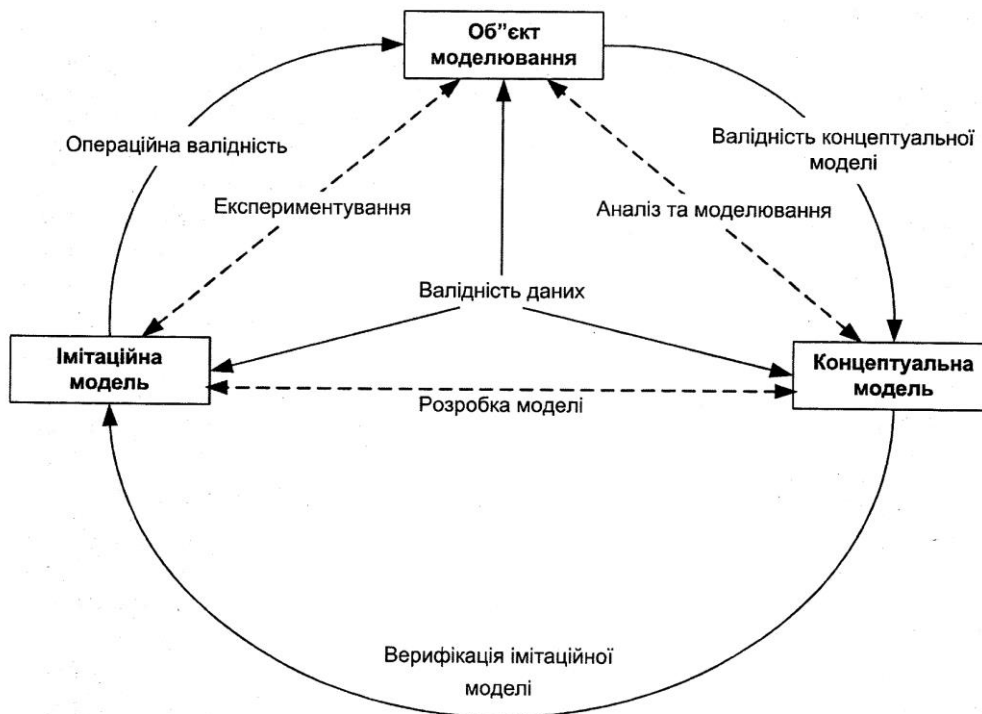


Рисунок 5.1 – Концептуальна схема процесу побудови імітаційної моделі

– *Рецензування експертів.* Модель переглядають практики-експерти у даній області та дають своє заключення щодо валідності моделі.

– *Використання історичних даних.* Якщо існують історичні дані з експлуатації системи, яка моделюється, то результати моделювання повинні відповідати цим історичним даним.

– *Внутрішня валідність.* Якщо модель є стохастичною, то її виконують декілька раз для одного і того самого набору параметрів і порівнюють результати. Якщо взаємні відхилення результатів перевищує визначену межу, то валідність моделі ставлять під сумнів.

– *Аналіз чутливості параметрів.* Вхідні дані або внутрішні параметри моделі змінюють з метою визначення впливу змін на вихідні дані. Для визначених тобто чутливих параметрів підвищують вимоги щодо точності вимірювань

– *Валідація по прогнозам.* Модель використовують для вироблення прогнозів щодо поведінки реальної системи в певних умовах. Модель валідують шляхом дослідження поведінки реальної системи у зазначених умовах.

– *Трасування.* Поведінку та параметри окремих частин моделі відстежують протягом моделювання з метою перевірки достатньої точності та логіки моделі.

Операційна валідність - це перевірка того, що вихідні дані моделювання мають достатню точність.

5.2.2 Тестування продуктивності (performance testing)

Тестування продуктивності є необхідним, якщо однією з основних переваг вирішень, запропонованих у роботі є підвищення продуктивності роботи системи (наприклад, зменшення часу реакції, або збільшення перепускної здатності). Тестувати продуктивність доцільно й тоді, коли існує обґрунтоване припущення, що застосування запропонованих у роботі методів може знизити загальну продуктивність системи.

Загалом тестування продуктивності дає змогу розв'язати такі задачі магістерської роботи:

- визначити як швидко працює розроблена система в умовах середнього навантаження, тобто перевірити, що швидкість роботи системи є прийнятною;
- порівняти продуктивність роботи системи з системами-аналогами в аналогічних умовах;
- перевірити дотримання вимог щодо надійності, масштабованості, використання ресурсів;
- обґрунтувати вибір таких компонент системи, як СУБД або серверна ОС;
- визначити обмеження щодо застосування системи. Для цього необхідно провести стрес-тестування;
- визначити вузькі місця системи та розробити рекомендації щодо покращання продуктивності.

Для тестування продуктивності системи створюють тестову систему. Як правило, до такої системи входять декілька машин, що створюють навантаження, відповідні сервера та комп'ютер-координатор тестування, який збирає тестові метрики.

Загалом тестування продуктивності вимагає виконання етапів, які розглянемо нижче.

Планування тестування продуктивності.

На цьому етапі визначають:

- структуру тестової системи, специфікують, які комп'ютери та програмне забезпечення використовуватимуть для тестування;
- сценарії тестування з детальною специфікацією кроків та профілі віртуальних користувачів, що беруть участь у тестуванні; наприклад, для тестування корпоративно: системи обліку робочого часу треба задіяти сценарії для працівника що заповнює свій табель, та окремий сценарій для начальника

відділу, що перевіряє години, вказані працівниками його відділу, і визначити кількість віртуальних працівників обох типів, що одночасно працюватимуть в системі (наприклад, 100/5);

- метрики тестування, які потрібно вимірювати, місце вимірювання, частоту вимірювання;

- визначають попередні критерії прийнятності результатів тестування (наприклад максимально допустимий час завантаження Web-сторінки).

У результаті цього етапу складають план тестування, який обговорюють з керівником магістерської роботи та планують виділення відповідних ресурсів для його проведення.

Підготовка до тестування.

На цьому етапі вирішують такі завдання:

- готують командні скрипти та програми що реалізують тестові сценарії;

- налаштовують тестове середовище – інсталюють необхідні програми, перевіряють їхню працездатність;

- виконують пробне тестування з метою перевірки розроблених скриптів і виявлення можливих конфігураційних проблем.

Проведення тестування.

На цьому етапі виконують розроблені скрипти і здійснюють тестування згідно з планом. Тестування доцільно виконати декілька разів з метою визначення стабільності та повторюваності результатів тестування. Результати тестування зберігаються у файлах результатів.

Опрацювання результатів тестування.

Під час опрацювання результатів тестування працюють з файлами результатів. Як правило, таке опрацювання полягає у підрахунку сумарних, кумулятивних показників, зіставлені подій, що відбувалися у системі одночасно. Для цього можна використати спеціальні інструментальні засоби та утиліти, призначені для опрацювання лог-файлів. На основі аналізу результатів і кумулятивних метрик приймають рішення щодо:

- прийнятності/неприйнятності продуктивності системи;
- вузьких місць у системі;
- планування нових тестувань для уточнення результатів, з врахуванням знайдених вузьких місць;
- поведінки системи для діапазонів значень даних, які не використовувалися в тестуванні;
- способів використання системи, виходячи з результатів тестування.

За результатами тестування формують звіт, який є частиною магістерської роботи.

5.2.3 Застосування формальних підходів до верифікації та валідації результатів

У магістерській роботі застосувати формальні підходи для верифікації та валідації доцільно тоді, коли розроблена система має певні нові властивості у поведінці або самим студентом запропоновані нові алгоритми і треба довести їхню правильність.

Формальні підходи використовують тоді, коли спроектована система може бути описана моделлю зі скінченною та відносно невеликою кількістю станів. Найчастіше формальну верифікацію проводять для розроблених протоколів (зокрема - криптографічних), а також програмного забезпечення або алгоритмів. Формальні підходи дають змогу валідувати програмну систему, наприклад, перевірити, що система має певну властивість чи її поведінка відповідає специфікації.

Формальна верифікація програмного забезпечення - це доведення коректності запропонованих алгоритмів на основі певної формальної специфікації з використанням формальних математичних методів.

Для виконання формальної верифікації використовують математичну модель системи. Для побудови такої моделі застосовують апарат скінчених автоматів, мережі Петрі, гібридні та часові автомати, алгебри процесів, формальні семантики мов програмування, такі як операційна семантика, нотаційна семантика, аксіоматична семантика та логіка Хоара.

Існують два основні підходи до формальної верифікації. Перший з них полягає у вичерпному дослідженні формальної моделі. При цьому зазвичай визначають усі стани в моделі та умови переходу між ними. У випадку великої кількості станів (або нескінченної їх кількості) цей метод також можна інколи застосувати, якщо проводити групування станів або використовувати абстракції станів та операцій.

Другий підхід полягає у проведенні формального логічного виведення. При цьому використовують формальні підходи розроблені для автоматичного доведення теорем і відповідне програмне забезпечення. Властивості, які треба довести часто описують з застосуванням апарату часової логіки.

5.3 Аналіз контрольних прикладів

У випадку коли результатом виконання магістерської роботи є створення програмного продукту з певними новими функціональними характеристиками, валідація виконується шляхом перевірки наявності зазначених функціональних характеристик під час тестового виконання програми. Для такого тестування створюють набір контрольних прикладів.

За основу побудови контрольних прикладів беруть початкові вимоги до системи, що відображають функціональні вимоги до системи та наведені, наприклад, у відповідних діаграмах прецедентів. Для кожної функціональної вимоги яку потрібно перетестувати розробляють окремий тестовий сценарій та документують його. До опису тестового сценарію входить:

- покроковий опис алгоритму його виконання;
- опис вхідних, вихідних та проміжних даних;
- додаткові умови, які мають виконуватися перед початком виконання сценарію;
- очікувані результати, які повинні бути отримані на кожному кроці виконання;
- додаткові процедури для перевірки стану системи у критичних моментах виконання сценарію.

Опис тестового сценарію має бути повним, щоб тестувальник, не знайомий із системою, міг виконати тестування тільки на основі опису.

Опис тестових сценаріїв вводять до плану тестування, який містить:

- загальну інформацію про об'єкт тестування;
- опис підходу та стратегії тестування, спеціальних методів тестування, якщо вони використовуються;
- апаратне та програмне забезпечення, які використовують для тестування;
- додаткові припущення та обмеження;
- опис тестових сценаріїв.

Результати тестування відображають у протоколі тестування. Для більшої наочності, крім текстових даних, протокол тестування може містити скріншоти важливих результатів або знайдених дефектів.

На основі отриманих результатів тестування формують звіт, в якому:

- визначають успіх/неуспіх тестування для кожного тестового сценарію;
- у разі неуспіхів у виконанні сценаріїв аналізують їхні причини;

- зазначають інші знайдені дефекти, які не стосуються безпосередньо тестових сценаріїв;
- у висновках в цілому рекомендують (не рекомендують) тестований продукт до використання.

5.3.1 Особливості валідації інтелектуальних систем

Верифікація та валідація інтелектуальних систем мають свої особливості.

Для валідації роботи інтелектуальної системи можна використати тест Тьюринга. При цьому розробляють набір тестових сценаріїв, які відображають певні реальні ситуації використання спроектованої системи. Система успішно проходить тестування, якщо рішення, які вона приймає, не відрізняються від рішень, які прийняла б людина-експерт у подібній ситуації.

Для систем, що використовують знання, подані у вигляді правил, верифікація системи часто поєднується з процесом уточнення правил. Під час верифікації системи правил перевіряють:

- цілісність бази правил; перевіряють, чи немає правил, які протирічать одне одному;
- повнота; для всіх можливих вхідних даних генерується правильна відповідь;
- ациклічність; доводять, що у всіх можливих послідовностях застосування правил жодне правило не використовується двічі;
- відсутність надлишковості; база знань надлишкова, якщо деякі знання використовуються по декілька разів, або в ній існують знання, які не використовуються.

Тестування неформальних та суб'єктивних переваг.

У деяких випадках переваги системи є якісними і не формалізуються. При цьому на об'єктивну оцінку впливають суб'єктивні фактори та уподобання користувачів системи. Насамперед це стосується ергономічних переваг, які часто сприймаються користувачем суб'єктивно, наприклад - зручність інтерфейсу користувача або якість дизайну веб-сайта. Але і в цьому випадку можна провести валідацію системи і запропонувати використання таких методів:

- отримати відгук експертів з ергономіки (або Web-дизайну) про зручність інтерфейсу;
- інтерфейс пропонується декільком експертам з ергономіки, які обґрунтовано оцінюють його переваги та недоліки і виставляють рейтингову оцінку;

– отримати відгуки користувачів; серед випадкових користувачів системи проводять рейтингове голосування.

6 ОФОРМЛЕННЯ ТА ПОРЯДОК ЗАХИСТУ МАГІСТЕРСЬКОЇ РОБОТИ

Магістерська робота виконується під керівництвом висококваліфікованих спеціалістів кафедри, що мають вчений ступінь або вчене звання. Розглянуто вимоги до оформлення пояснювальної записки магістерської кваліфікаційної роботи, які ґрунтуються на Державному стандарті України ДСТУ 3008-95 «Документація. Звіти у сфері науки і техніки. Структура і правила оформлення». В розділі описано процедур; попереднього захисту та захисту магістерської роботи в ДЕК.

6.1 Організація дипломного проектування

6.1.1 Керівництво магістерською роботою

Кафедра не менш як за 30 днів до стажування з фаху (дослідницької практики) ознайомлює студентів з тематикою магістерських робіт, проводить розподіл тем серед студентів, визначає керівників магістерських робіт, узгоджує списки консультантів ДР з інших кафедр.

Безпосереднє керівництво магістерською роботою здійснює керівник. Керівниками призначаються досвідчені викладачі кафедри комп'ютерних наук, які ведуть наукові дослідження або займаються науковою тематикою кафедри КН. Також за рекомендацією кафедри КН на певні види магістерських робіт поряд із керівником може призначатися консультант з числа професорів кафедри.

Консультантами із змістовних частин роботи призначаються наукові співробітники та висококваліфіковані фахівці університету, які затверджуються наказом ректора по університету.

За одним керівником закріплюється не більше п'яти студентів-дипломників одночасно.

Керівник магістерської роботи:

- готує та видає студенту завдання на магістерську роботу;
- надає методичну допомогу у вирішенні тих чи інших питань;
- надає рекомендації дипломнику щодо опрацювання необхідної літератури, нормативних і довідкових матеріалів, наукових видань за темою ДР;
- застерігає від прийняття некваліфікованих хибних рішень;
- вказує студенту напрямки пошуку і т.п.;
- допомагає дипломнику скласти, затверджує та контролює реалізацію календарного плану виконання ДР;

- здійснює загальне керівництво ДР;
- готує відгук.

У разі суттєвих порушень, які можуть призвести до зриву встановлених термінів надання ДР до Державної атестаційної комісії, інформує керівництво кафедри для прийняття відповідних заходів, у тому числі й рішення про недопущення до захисту.

Магістерська робота виконується студентом самостійно і він несе **повну відповідальність** за прийняті рішення, якість та терміни виконання роботи.

6.1.2 Організація консультацій.

Для успішної розробки ДР для студентів кафедрою організовуються консультації, розклад проведення яких доводиться до студента на початку виконання ДР.

Керівник ДР допомагає в організації для студентів консультацій із співробітниками інших кафедр, якщо під час виконання ДР в цьому виникає потреба. Консультації проводяться так, що консультант не підміняє ними студента у вирішенні питань ДР.

Студент зобов'язаний з'являтися на консультації за графіком, маючи при собі всі необхідні матеріали.

6.1.3 Етапи виконання магістерської роботи

Згідно з навчальним планом підготовки магістрів кафедри дипломне проектування виконується в три етапи:

- стажування з фаху (3 тижні, дослідницька практика – 4 тижні);
- саме дипломне проектування (6 тижнів);
- захист дипломної роботи (ДР) в Державній Екзаменаційній Комісії (ДЕК) (2 тижні).

Стажування з фаху (Дослідницька практика).

До початку стажування з фаху (орієнтовно за 1÷1,5 місяці) за студентами закріплюються теми та керівники ДР. Вони розглядаються і ухвалюються на засіданні кафедри та затверджуються наказом ректора по університету. Корекція або зміна теми ДР допускається, **як виняток**, після проходження студентом практики та захисту звіту за її результатами, упродовж не більше ніж двох тижнів.

Практика завершується складанням та захистом звіту про проходження практики. У разі необхідності зміни теми чи заміни керівника магістерської роботи оформлюється лист на ім'я завідуючого кафедрою з обов'язковим обґрунтуванням заміни. Згідно листа готується наказ про зміну.

За матеріалами проведених робіт студент готує тези доповіді, які представляє на студентських конференціях.

Дипломне проектування.

Протягом перших двох тижнів дипломного проектування студент складає календарний план-графік виконання магістерської роботи, що узгоджується та затверджується керівником роботи. Студент виконує роботи по підготовці роботи згідно з план-графіком. Керівник та консультанти проводять консультації, здійснюють нормоконтроль, організують перегляди ДР згідно з розкладом консультацій та план-графіком виконання ДР. Виконання окремих етапів магістерської роботи на завершальному етапі помічається підписом керівника і консультантів з відповідних розділів на завданні до роботи.

Таким чином, виконання роботи складається із шести етапів:

I етап – аналіз завдання на магістерську роботу, підбір бібліографічних матеріалів, необхідних для її розробки, техніко-економічний аналіз;

II етап – вибір напрямку роботи, постановка задачі;

III етап – виконання роботи — розробка основних розділів ДР;

IV етап – виконання роботи — розробка додаткових розділів ДР;

V етап – оформлення ДР;

VI етап - захист ДР.

Виконана у відповідності із завданням і в повному обсязі ДР, підписаний виконавцем, керівником та консультантами, подається на нормоконтроль. Після нормоконтролю всі текстові та графічні матеріали **роботи разом з письмовим відгуком керівника подаються у незшитому вигляді** для попереднього захисту.

На останньому тижні етапу дипломного проектування (за 3-7 днів до початку основних захистів) проводиться попередній захист магістерської роботи.

Дата попереднього захисту студенту – дипломнику призначається рішенням кафедри згідно графіка основного захисту. Проте дату попереднього захисту студента-дипломника необхідно завчасно погодити з керівником роботи.

Попередній захист роботи на кафедрі проводиться студентом — автором ДР перед комісією, що призначається завідувачем кафедрою, з обов'язковою присутністю керівника ДР. Виявлені при попередньому захисті ДР неточності і помилки студент зобов'язаний виправити і результати представити керівникові ДР. Якщо ж при огляді встановлено, що робота в будь-якій частині потребує суттєвої доробки, то визначається обсяг доробки і встановлюється термін подання виправленої ДР на повторний попередній захист. Рішення комісії, як

про допуск ДР до захисту, так і, у випадку необхідності, доопрацювання ДР, заноситься в протокол засідання комісії. На основі рішення комісії кафедра видає студенту направлення на рецензію ДР.

Рецензент зобов'язаний в дводенний термін дати розгорнуту рецензію на подану йому роботу і оцінку виконаній ДР ("відмінно", "добре", "задовільно", "незадовільно").

При позитивній рецензії робота подається на підпис завідувачу кафедрою. На основі поданих матеріалів завідувач кафедрою вирішує питання про допуск ДР до захисту в ДЕК. До захисту може бути допущена робота.

Захист дипломної роботи.

На етапі захисту дипломної роботи студент отримує відгук керівника та рецензію від рецензента, подає проект секретарю ДЕК (не пізніше ніж за добу до захисту) та захищає роботу на засіданні ДЕК. Рекомендації до складання відгуку керівника приведені у додатку Е, а рецензії – у додатку Ж.

Рецензент призначається завідувачем кафедри КН відповідно до наказу ректора ТНТУ імені Івана Пулюя, при цьому можуть бути враховані пропозиції керівника та/або студента-автора ДР.

Рецензія, що не відповідає встановленій формі, може бути ДЕКом відхилена з направленням роботи на нове рецензування. Студент має право ознайомитися з рецензією до захисту. Негативна рецензія не є підставою відхилення дипломної роботи до захисту.

6.1.4 Контроль за ходом розробки дипломної роботи

Студент зобов'язаний щоденно працювати над ДР, суворо дотримуватися графіка розробки, з'являтися на консультації, звітуватись перед керівником ДР за пророблену роботу з представленням виконаних матеріалів. Якщо студент не з'являється на консультації, нерегулярно звітується перед керівником ДР, або систематично порушує графік розробки ДР, то за представленням керівника ДР він несе адміністративну відповідальність, аж до недопущення його до захисту ДР.

6.2 Обов'язки осіб за організацію та проведення дипломного проектування

6.2.1 Обов'язки керівників дипломних робіт

Керівниками призначаються досвідчені викладачі кафедри комп'ютерних наук, які ведуть наукові дослідження або займаються науковою тематикою кафедри КН.

Керівник роботи в процесі своєї роботи систематично вивчає становище і перспективи розвитку науки і техніки своєї спеціальності. Керує науковою роботою студентів, підготовляючи їх до дипломного проектування.

Своєчасно і якісно розробляє тематику роботи з урахуванням наукової роботи студентів, зроблених раніше висновків та рекомендацій ДЕК.

Розробляє конкретні і достатньо повні завдання на дипломне проектування, рекомендує необхідну літературу і дає методичні рекомендації.

Проводить бесіди із студентами по узгодженню завдань дипломного проектування, структури магістерської роботи і етапів роботи над нею. Рекомендує методики виконання розрахунків, проведення експериментів і моделювання (якщо такі передбачені завданням на дипломне проектування). Подає допомогу студентам в складанні, календарних планів виконання дипломних проектів.

Здійснює безпосереднє і систематичне керівництво розробкою всіх питань дипломної роботи, розвиваючи при цьому у студента навички самостійної роботи, творчі здібності і ініціативу. Основною формою керівництва дипломним проектуванням являється індивідуальна консультація. Керівник докладно консультує студента по питанням, які виходять за межі учбових дисциплін рекомендує доступну для студента літературу. Питання, які доступно викладені в літературі, керівник рекомендує студенту проробити самостійно.

Здійснює систематичну перевірку відповідності ходу роботи студента календарному плану виконання магістерської роботи, розглядає виконану частину роботи, звертає увагу на недостатню глибину опрацьованих питань. Перевіряє розділи пояснювальної записки дипломного проекту в чорновому (першій редакції) вигляді по мірі їх підготовки з метою недопущення грубих помилок, які можуть привести до невиконання завдання, надлишкове збільшеного обсягу, зниженого наукового рівня проекту. Контролює відповідність змісту і оформленню магістерської роботи завдання, вимогам стандартів ДСТУ і керівним документам по дипломному проектуванні.

Встановлює відповідність роботи індивідуальному завданню на дипломне проектування;

Відображує хід виконання магістерських робіт в графіку контролю дипломного проектування на кафедрі. Всі позначки, які виникають, в ході контролю роботи студента-дипломника, керівник вносить в чорновий варіант дипломного проекту.

Підписує пояснювальну записку до магістерської роботи та графічну частину після досконалої перевірки. Складає відгук з характеристикою роботи студента над магістерською роботою.

6.2.2 Обов'язки консультантів

По узгодженню із відповідними завідувачами кафедр для консультування студентів по окремим питанням роботи завідувач кафедри комп'ютерних наук може запрошувати викладачів із числа професорсько-викладацького складу інших кафедр. Узгодження питань, які виносяться консультантами на розгляд в магістерських роботах, зі студентами-дипломниками повинно бути закінчено до початку роботи ДЕК.

Методична підготовка є інтегруючою ланкою між психолого-педагогічною і інженерною підготовками. Її мета – сформувати у студента-дипломника уміння проектувати дидактичні системи, іншими словами – уміння будувати і реалізовувати власні дидактичні проекти.

В своїй роботі по аналізу економічної ефективності роботи викладачі-консультанти виходять із того, що оцінка прийняття технічних рішень повинна мати місце на всіх станах дипломного проектування.

В учбово-методичних матеріалах, доступних для студентів-дипломників відображається сукупність варіантів задач, які можуть бути включені в завдання на дипломне проектування. При цьому студенту надається право вибору однієї із цих задач, а також право рішення задачі, яка не увійшла в загальний перелік, але відповідає даній спеціалізації і темі дипломного проекту.

6.2.3 Обов'язки студента-дипломника

Студенту представляється право вибору теми магістерської роботи з урахуванням рекомендацій професорсько-викладацького складу профілюючої кафедри. Студент може запропонувати свою тему з необхідним обумовленням її розробки для підприємства.

Якщо студент не проявив необхідної ініціативи по вибору тем в указаний термін, завідувач кафедрою закріплює тему магістерської роботи за студентами на свій погляд, але з урахуванням індивідуальних особливостей та

ступенем його підготовки. Повний перелік обов'язків студента-дипломника наведено в додатку А.

6.2.4 Обов'язки нормоконтролера

Нормоконтролеру магістерських робіт представляється пояснювальна записка і графічна частина.

Основна мета нормоконтролю магістерських робіт – підвищення якості підготовки спеціалістів по профілю підготовки кафедри комп'ютерних наук.

Основні обов'язки нормоконтролера:

- перевірка в магістерських роботах дотримання норм і вимог, установлених в стандартній та додатковій нормативно-технічній документації;
- перевірка правильності оформлення пояснювальної записки;
- перевірка в розроблених об'єктах дипломного проектування високого рівня стандартизації, уніфікації і типізації на основі типових проектів і проектних рішень;
- перевірка зовнішнього вигляду проектної документації на акуратність;
- проведення аналізу виявлених при нормоконтролі помилок;
- інформування дипломників і керівників магістерських робіт про виявлені помилки.

При перевірці магістерських робіт перевіряються:

- відповідність позначень установлених системі позначень конструкторських документів;
- комплектність документації;
- правильність виконання основних надписів;
- правильність застосованих скорочених слів;
- наявність і правильність посилань на стандарти;
- правильність оформлення таблиць, схем, ілюстрацій, додатків;
- відповідність одиниць вимірювання ГОСТ 8417-81 Метрологія.

Одиниці фізичних величин;

- відповідність стандартів до текстових конструкторських документів (ГОСТ 2.105-79. ГОСТ 2.106-68);
- відповідальність показників і розрахункових величин нормативним даним;
- відповідність виконання креслень вимогам стандарту на формати, масштаби, зображення (види, розміри, перерізи), конструкторських документів.

Нормоконтроль являється останнім етапом розробки документації магістерської роботи і здійснюється перед попереднім захистом на кафедрі.

6.3 Оформлення матеріалів магістерських робіт

6.3.1 Загальні вимоги

Всі документи, що входять у ДР, повинні бути виконані відповідно до вимог ЄСКД та ДСТУ.

В таблиці 6.1 приведений перелік основних державних стандартів, якими необхідно керуватися при виконанні та оформленні ДР.

Таблиця 6.1 - Державні стандарти, що використовуються при виконанні ДР

Документ	Стандарт
ПЗ	ДСТУ 3008-95. Документація. Звіти в сфері науки і техніки. ГОСТ 2.105-79 ЕСКД. Общие требования к текстовым документам. ГОСТ 2.106-68 ЕСКД. Текстовые документы. ГОСТ 2.106-96 ЕСКД. Текстовые документы. ГОСТ 19.106-78 ЕСКД. Требования к программным документам, выполненным печатным способом. ГОСТ 2.104-2006. Единая система конструкторской документации. Основные надписи. ГОСТ 19.404-79 ЕСКД. Пояснительная записка. Требования к содержанию и оформлению. ДСТУ ГОСТ 7.1:2006 Система стандартів з інформації, бібліотечної та видавничої справ. БІБЛОГРАФІЧНИЙ ЗАПИС. БІБЛОГРАФІЧНИЙ ОПИС. Загальні вимоги й правила складання.
Специфікація	ГОСТ 2.108-69. Спецификация. ГОСТ 19.202-78 ЕСКД. Спецификация. Требования к содержанию и оформлению.
Креслення	ГОСТ 2.109-73 ЕСКД. Основные требования к чертежам. ГОСТ 2.119-73 ЕСКД. Эскизный проект.

	ГОСТ 2.120-73 ЕСКД. Технический проект. ГОСТ 2.417-91 ЕСКД. Правила выполнения чертежей печатных плат ГОСТ 2.104-2006. Единая система конструкторской документации. Основные надписи
Схеми	ГОСТ 2.701-2008 ЕСКД. Схемы. Виды и типы. Общие требования к выполнению. ГОСТ 2.702-75 ЕСКД. Правила выполнения электрических схем. ГОСТ 2.708-81 ЕСКД. Правила выполнения электрических схем цифровой вычислительной техники. ГОСТ 2.710-81 ЕСКД. Обозначения буквенно-цифровые в электрических схемах. ГОСТ 2.711-82 ЕСКД. Схемы деления изделия на основные части ГОСТ 19.701-90 ЕСКД. Схема алгоритмов и программ. Обозначения условно-графические
Програми і програмні документи	ГОСТ 19.101-77 ЕСКД. Виды программ и программных документов. ГОСТ 19.105-78 ЕСКД. Общие требования к программным документам ГОСТ 19.401-78 ЕСКД. Текст программы ГОСТ 19.402-78 ЕСКД. Описание программы

Пояснювальна записка магістерської роботи оформляється у відповідності до вимог ДСТУ 3008-95. Документація. Звіти в сфері науки і техніки.

Позначення технологічних документів, що входять до ДР, виконуються згідно з ГОСТ 3.1201-74.

Позначення програм і програмних документів повинні відповідати ГОСТ 19.103-77.

Зміст, розміщення і розмір граф основних надписів, а також розміри рамок на кресленнях і схемах повинні відповідати формі 1, в текстових документах – формам 2 і 2а за ГОСТ 2.104-68 та ГОСТ 2.104-2006.

Код класифікаційної характеристики надають виробу або конструкторському документу за класифікатором виробів та конструкторських документів машинобудування і приладобудування (класифікатор ЄСКД). Код класифікаційної характеристики має таку структуру:

XXXX	XX	XXX	XX	XXX	XX
ДРКН	13	214	11	001	ПЗ
Код організації розробника вибирають за модифікатором (тип роботи)	Останні дві цифри року захисту	Останні три цифри номеру залікової книжки	Порядковий номер теми ДП в наказі	Порядковий номер розробки від 001 до 999	Код документа

Примітка: Після коду організації розробника і порядкового номеру розробки ставиться пропуск одного знаку, розділення інших структурних одиниць проводиться крапкою. (Шрифт – Times New Roman, кегль 20 пт).

6.3.2 Вимоги до оформлення пояснювальної записки

Пояснювальна записка, креслення, плакати та інші матеріали оформляються в одному примірнику.

Пояснювальна записка виконується на листах формату А4 за формами відповідно до вимог діючих ГОСТів та ДСТУ (додатки И, К). Текст магістерської роботи друкують (пишуть), залишаючи поля таких розмірів:

- ліве - не менше 25 мм;
- праве - не менше 10 мм;
- верхнє - не менше 20 мм;
- нижнє - не менше 25 мм.

При оформленні текстової частини на листах з рамкою, відступи від тексту до рамки: зліва і справа – не менше 5 мм, зверху і знизу – не менше 10 мм.

Пояснювальна записка ДР виконується машинним (за допомогою комп'ютерної техніки) способом на одному боці аркуша. Також дозволяється виконувати пояснювальну записку рукописним способом у чорному кольорі.

Пояснювальна записка повинна починатися з титульного аркуша встановленого зразка (додаток В), далі розміщують завдання на ДР (додаток Д), реферат (додаток З), перелік умовних позначень, символів, одиниць, скорочень і термінів (за необхідністю), зміст, основний текст, перелік використаних джерел, використаних при виконанні роботи та додатки.

Текстовий матеріал, при виконанні комп'ютерним способом, друкується на одному боці аркушів формату А4 через 1,5 міжрядкового інтервалу, текст

вирівнюється по ширині аркуша (текстовий редактор сумісний з Word for Windows версія 7.0 або більш пізня. Шрифт – Times New Roman, кегль 14 пт).

При наведенні ключових слів в анотації пояснювальної записки магістерської роботи (додаток 3) використовується шрифт – Times New Roman, кегль 13 пт, ефекти – усі великі літери, наприклад, РИТМІЧНІСТЬ, ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ.

При виконанні текстового матеріалу (додатки Б, В, Д, Е, Ж) рукописним способом текст повинен бути виконаний креслярським шрифтом згідно з ГОСТом 2.304-81 з висотою букв і цифр не менше 2,5 мм. Цифри і букви необхідно писати чітко, виконаними в **чорному кольорі**.

Помилки й графічні неточності допускається виправляти заклеюванням, підчищуванням або замальовуванням білою фарбою з наступним внесенням виправленого тексту (графіки).

Пошкодження листів текстових документів, забруднення, неповністю знищені сліди попереднього тексту **не допускається**.

При вписуванні слів, формул, знаків у надрукований текст вони мають бути чорного кольору; щільність вписаного тексту має максимально наближуватись до щільності основного зображення.

Виправлення мають бути чорного кольору.

Титульний лист і завдання на дипломну роботу виконують на стандартних бланках.

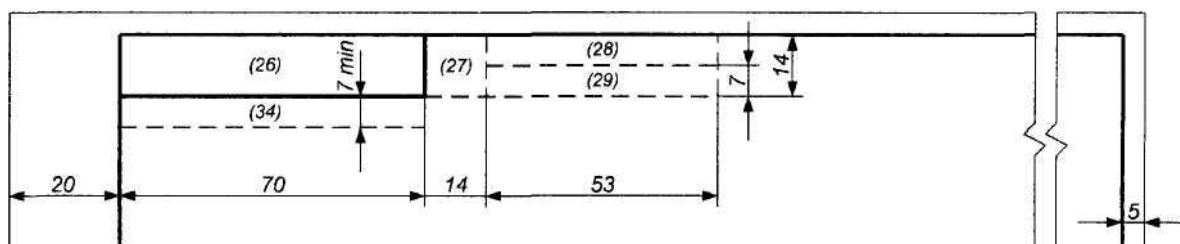
Всі аркуші (сторінки) пояснювальної записки нумерують.

Нумерацію листів ПЗ починають із титульного аркуша, на якому номер не проставляється. Аркуш, розміщений після завдання на дипломне проектування, нумерується цифрою 3.

Кожну структурну частину роботи починають з нової сторінки. При оформленні текстової частини на листах з рамкою розміри рамок для основних частин мають відповідати формі 2, а для наступних листів – формі 2а (див. рисунок 6.1). До загального обсягу роботи не входять додатки, таблиці та рисунки, які повністю займають площу сторінки. Але всі сторінки зазначених елементів підлягають нумерації на загальних засадах.

Форма 1

Основний напис і додаткові графи для креслень і схем



					(2)			
Зм	Арк.	№ докум.	Підпис	Дата	(1)	Літ.	Маса	Масш.
Розроб.								
Перевір.								
Консульт.								
Затверд.	(11)	(12)	(13)		(3)	Арк. (7) Аркушів (8)		
Н. контр.								
Реценз.						(9)		

Форма 2

Основний напис для текстових конструкторських документів
(перший або заголовний листок)

					(2)			
Зм	Арк.	№ докум.	Підпис	Дата	(4)	Літ.	Арк.	Аркушів
Розроб.								
Перевір.								
Консульт.	(11)	(12)	(13)					
Н. контр.					(1)	(9)		
Затверд.								

Форма 2а

Основний напис для креслень (схем) та текстових конструкторських
документів(наступні листи)

					(2)	Арк.
Зм	Арк.	№ докум.	Підпис	Дата		(7)

При виконанні текстової частини на листах з рамкою, номер сторінки проставляють у відповідній графі рамки. При цьому на першій сторінці кожного розділу поміщають кутовий штамп за формою 2, розміри і правила заповнення якого регламентовані ГОСТ 2.104-68 та ГОСТ 2.104-2006. В графах 1, 4 вказують назву розділу та назву графічного плакату. В графі 2 наводять код класифікаційної характеристики. Графи 3, 5, 6 заповнюють в разі потреби. Графу 8 на цих листах заповнюють, вказуючи кількість листів певного розділу ПЗ або графічного матеріалу, а в графі 7 показують порядковий номер листа, продовжуючи наскрізну нумерацію сторінок по всій записці ДР. В графах 11, 12, 13 прописують прізвище ініціали виконавця, консультантів тощо, проставляють підпис та дату відповідно. В графі 9 наводять інформацію про підприємство, підрозділ, наприклад: *ТНТУ, ФІС, КН, гр. СНм-51*. В графі 26 наводять код класифікаційної характеристики, позначення документу, повернуте на 180° для формату А4 і для форматів більше А4 при розташуванні основного напису вздовж довгої сторони листа і на 90° для форматів більше А4 при розташуванні основного напису вздовж короткої сторони листа (додаток Л).

В тексті пояснювальної записки мають бути обов'язковими посилання на використані літературні та інші джерела. Після згадки (після цитати) проставляють в квадратних дужках номер, під яким вона іде в бібліографічному списку і, у випадку необхідності, сторінки, наприклад: [9] або [9, с.92].

Перелік джерел інформації повинен бути відсортований в алфавітному порядку (за прізвищами авторів) (додатки И, К).

Роботи іноземних авторів подаються в переліку джерел в оригінальній транскрипції.

Документи, розміщення яких в основному тексті недоцільне (рисунок, програми розрахунків на ЕОМ, таблиці, які займають одну або більше аркушів А4 ПЗ), повинні бути оформлені у вигляді додатків до дипломного проекту. В основному тексті потрібно вказати посилання на ці додатки.

Додаток повинен мати заголовок, надрукований угорі малими літерами з першої великої симетрично відносно тексту сторінки. З правого боку рядка над заголовком малими літерами з першої великої друкується слово «Додаток __» і велика літера, що позначає додаток. Додатки необхідно позначати послідовно великими літерами української абетки, за винятком літер Г, Є, І, Ї,

Й, О, Ч, Ь, наприклад, додаток А, додаток Б і т.д. Один додаток позначається як додаток А.

Текст кожного додатка за необхідності може бути поділений на розділи і підрозділи, які нумерують у межах кожного додатка. В такому разі перед кожним номером ставлять позначення додатка (літеру) і крапку, наприклад, А.2 – другий розділ додатка А.

Ілюстрації, таблиці і формули, розміщені в додатках, нумерують у межах кожного додатка, наприклад: рис. Д.1.2. – другий рисунок першого розділу додатка Д; формула (В.1) – перша формула додатка В.

У тексті пояснювальної записки не рекомендується вживати звороти із займенниками першої особи, наприклад: "Я вважаю ...", "Ми вважаємо ...", тощо. Рекомендується вести виклад, не вживаючи займенників, наприклад: "Вважаємо ...", "... знаходимо ..." тощо.

Числа з розмірністю необхідно писати цифрами, а без розмірності – словами, наприклад: "Висота – 600 м", "... за другим варіантом ...". Не допускається у тексті записки використовувати число розмірності в одному рядку, а позначення розмірності – в іншому, таким чином – "600 м" мають бути в одному рядку.

Порядкові числівники, які йдуть один за одним, можуть бути подані цифрами з відмінковим закінченням, яке ставлять лише при останній цифрі, наприклад: 1-е; 7, 8, 9-й тощо.

Абзацний відступ повинен бути однаковим упродовж всього тексту записки дипломного проекту і дорівнювати 1,5 см абзацного відступу.

При використанні в пояснювальній записці списків, абзацний відступ становить 2 см абзацного відступу. При використанні нумерованого списку дозволяється в якості номерів використовувати арабські та римські цифри, кириличні та латинські букви. За умови використання маркованого списку використовуються наступні маркери: –; ●. Абзацний відступ багаторівневого списку становить 2 см абзацного відступу. По всій пояснювальній записці дипломного проекту використовується тільки один із вибраних студентом-дипломником маркерів.

Пояснювальну записку магістерської роботи розбивають на розділи і підрозділи, пункти і підпункти.

Розділи в межах усієї записки повинні мати порядкові номери, позначені арабськими цифрами без крапки.

Кожен розділ текстового документу рекомендується починати з нового аркуша (сторінки).

Підрозділи повинні мати нумерацію в межах розділу: номер підрозділу складається з номера розділу і підрозділу, розділених крапкою, наприклад, 2.3.

Це означає: третій підрозділ другого розділу. В кінці порядкового номера розділу, підрозділу і т.п. крапка не ставиться.

Номер пункту вміщує номер розділу, підрозділу і пункту, які розділені крапками, наприклад, 3.2.1 – перший пункт другого підрозділу третього розділу.

Структурні елементи “ЗМІСТ”, “ВСТУП”, “ВИСНОВОК”, “ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ” не нумерують, а їх назви правлять за заголовки структурних елементів.

Назви розділів повинні бути короткими і записуватись у вигляді заголовків великими буквами посередині рядка, наприклад: 1 ТЕОРЕТИЧНІ АСПЕКТИ ДОСЛІДЖЕННЯ. Переноси слів у заголовках не допускаються. Крапка в кінці заголовка не проставляється. Між назвами основних розділів, підрозділів другого рівня і основним текстом повинен бути пропущений рядок. Між назвами підрозділів третього рівня і основним текстом пропускається один рядок перед назвою підрозділу, а після назви – ні.

Заголовки підрозділів, пунктів і підпунктів слід починати з абзацного відступу і друкувати маленькими літерами, крім першої великої, не підкреслюючи, без крапки в кінці.

Якщо заголовок складається з двох і більше речень, їх розділяють крапкою.

Відстань між основами рядків заголовків, а також між двома заголовками приймають такою, як у тексті.

Не допускається розміщувати назву розділу, підрозділу, а також пункту й підпункту в нижній частині сторінки, якщо після неї розміщено тільки один рядок тексту.

При написанні розділів, підрозділів, пунктів і підпунктів використовується стиль шрифту напівжирний.

На основі розділів, підрозділів, пунктів і підпунктів формують аркуш “ЗМІСТ”.

Оформлення змісту здійснюється на основі вимог оформлення тексту пояснювальної записки. При наведенні заголовків 1 рівня абзацний відступ не роблять. Для наведення заголовків 2 рівня абзацний відступ становить 0,5 см, 3 рівня – 1 см. З протилежної сторони рядка заголовка в змісті проставляється номер сторінки, який розділяється багато крапкою, наприклад:

ЗАГОЛОВОК 1.....	1
Заголовок 2.....	2
Заголовок 3.....	3

Як правило нумерація змісту закінчується на 3 рівні заголовків.

Ілюстрації (креслення, рисунки, графіки, схеми, діаграми, фотознімки) слід розмішувати у записці дипломного проекту безпосередньо після тексту, де вони згадуються вперше, або на наступній сторінці. На всі ілюстрації повинні бути посилання у дипломному проекті.

Ілюстрації у тексті виконуються в графічному редакторі або тушшю або олівцем (рукописний спосіб).

Кількість ілюстрацій повинна бути достатньою для пояснення тексту, що викладається. Ілюстрації розміщуються відразу після посилання на них за текстом.

Фотознімки розміру меншого за формат А4, мають бути наклеєні на аркуші білого паперу формату А4.

Ілюстрація позначається словом "Рисунок__", яке разом з назвою ілюстрації розміщують після пояснювальних даних посередині рядка, наприклад, "Рисунок 3.1 – Схема розміщення". Ілюстрацію разом із її назвою відділяють від основного тексту записки рядком пропуску.

За необхідності під ілюстрацією розміщують пояснювальні дані (підрисунковий текст).

Перше посилання на ілюстрації подається за типом «на рисунку 3.1», повторно – «див. рис. 3.1».

Ілюстрації слід нумерувати арабськими цифрами порядковою нумерацією в межах розділу, за винятком ілюстрацій, наведених у додатках.

Номер ілюстрації складається з номера розділу і порядкового номера ілюстрації, відокремлених крапкою, наприклад, рисунок 3.2 – другий рисунок третього розділу.

Якщо ілюстрація не поміщається на одній сторінці, можна переносити її на інші сторінки, вміщуючи назву ілюстрації на першій сторінці, пояснювальні дані – на кожній сторінці, і під ними позначають: "Рисунок __, аркуш__".

Цифровий матеріал, як правило, оформляють у вигляді таблиць.

Горизонтальні та вертикальні лінії, які розмежовують рядки таблиці, а також лінії зліва, справа і знизу, що обмежують таблицю, можна не проводити, якщо їх відсутність не утруднює користування таблицею.

Таблицю слід розташовувати безпосередньо після тексту, у якому вона згадується вперше, або на наступній сторінці.

На всі таблиці мають бути посилання в тексті ПЗ. Перше посилання на таблицю має вигляд “У таблиці 2.2 наведено...”, повторно – «див. табл. 2.2».

Таблиці слід нумерувати арабськими цифрами порядковою нумерацією в межах розділу.

Номер таблиці складається з номера розділу і порядкового номера таблиці, відокремлених крапкою, наприклад, таблиця 2.1 – перша таблиця другого розділу.

Таблиця має назву, яку друкують малими літерами (крім першої великої) і вміщують над таблицею. Назва має бути стислою і відображати зміст таблиці. Назва проставляється після номера таблиці через тире. Назву таблиці відокремлюють від основного тексту рядком пропуску.

Якщо таблиця виходить за межі формату сторінки, її поділяють на частини, переносячи частину таблиці на наступну сторінку, повторюючи в кожній частині таблиці її заголовок і боковик.

При поділі таблиці на частини допускається її заголовок або боковик замінити відповідно номерами граф чи рядків, нумеруючи їх арабськими цифрами у першій частині таблиці.

Слово “Таблиця __” вказують один раз зліва з абзацного відступу над першою частиною таблиці, над іншими частинами справа пишуть: “Продовження таблиці __” із зазначенням номера таблиці.

Після таблиці проставляється рядок пропуску перед основним текстом ПЗ.

Елементи програмного коду (тексти програм, процедур чи функцій, скриптів, HTML або XML–коди) слід розміщувати у записці дипломного проекту безпосередньо після тексту, де вони згадуються вперше, або на наступній сторінці. Кількість елементів програмного коду повинна бути достатньою для пояснення тексту, що викладається.

Елементи програмного коду у тексті виконуються в текстовому редакторі або тушшю чи олівцем (рукописний спосіб). Елементи програмного коду, при виконанні комп’ютерним способом, друкуються на одному боці аркушів формату А4 через 1 (один) міжрядковий інтервал, текст вирівнюється по лівій стороні аркуша (текстовий редактор сумісний з Word for Windows версія 7.0 або більш пізня. Шрифт – Courier New, кегль 12 пт).

На всі елементи програмного коду мають бути посилання в тексті ПЗ. Посилання на елемент програмного коду має вигляд: “У лістингу 1.2 наведено...”.

Елементи програмного коду слід нумерувати арабськими цифрами порядковою нумерацією в межах розділу.

Номер елемента програмного коду складається з номера розділу і порядкового номера лістингу, відокремлених крапкою, наприклад, лістинг 1.2 – другий лістинг першого розділу.

Елемент програмного коду має назву, яку друкують малими літерами (крім першої великої) і вміщують над лістингом. Назва має бути стислою і

відображати зміст програмного коду. Назва проставляється після номера лістингу через тире. Назву лістингу відокремлюють від основного тексту ПЗ рядком пропуску.

Якщо програмний код виходить за межі формату сторінки, його поділяють на частини, переносячи частину лістингу на наступну сторінку, не повторюючи заголовок.

Слово “Лістинг ___” вказують один раз зліва з абзацного відступу над першою частиною програмного коду із зазначенням номера лістингу.

Після лістингу проставляється рядок пропуску перед основним текстом ПЗ.

Формули та рівняння розташовують безпосередньо після тексту, в якому вони згадуються, посередині сторінки.

Вище і нижче кожної формули або рівняння повинно бути залишено не менше одного вільного рядка.

Формули і рівняння у ПЗ (за винятком формул і рівнянь, наведених у додатках) слід нумерувати порядковою нумерацією в межах розділу.

Номер формули або рівняння складається з номера розділу і порядкового номера формули або рівняння, відокремлених крапкою, наприклад, (1.3) – третя формула першого розділу.

Номер формули або рівняння зазначають на рівні формули або рівняння в дужках у крайньому правому положенні на рядку.

Якщо формула або рівняння не вміщується в один рядок, його переносять в інший після знаків рівності (=), плюс (+), мінус (-), множення (x) і ділення (:)

Пояснення значень символів і числових коефіцієнтів, що входять до формули чи рівняння, слід наводити безпосередньо під формулою у послідовності, в якій вони наведені у формулі чи рівнянні.

Пояснення значення кожного символу та числового коефіцієнта слід давати з нового рядка. Перший рядок пояснення починають з абзацу словом “де” без двокрапки.

Написання формул, цифр, заголовків розділів і підрозділів, заповнення таблиць виконується тільки шрифтом (при оформленні рукописним способом).

Основний текст пояснювальної записки і додатки розділяють сторінкою пропуску, на якій по центру сторінки друкується слово “ДОДАТКИ” з наступними параметрами: шрифт – Times New Roman, кегль – 48 пт, стиль шрифту – напівжирний.

Специфікації, що входять у додатки до записки ДР, виконуються за формами відповідно до ГОСТу 2.108-68.

Інші конструкторські документи, що входять у додатки (відомість купованих виробів, методика та програма випробувань та ін.), виконуються за формами, вказаними у відповідних стандартах.

6.3.3 Вимоги до оформлення графічної документації

Допускається виконання креслень вручну, але не менше як один документ має бути виконано засобами САПР.

Креслення і схеми виконуються на креслярському папері (ватмані) формату А1.

Усі схеми повинні бути відображені в ПЗ. Спочатку потрібно описати схему в статичі (перерахувати компоненти, зв'язки та їх призначення), після цього описати схему в динаміці, тобто пояснити функціонування схеми (послідовність проходження і перетворення сигналу від входу до виходу). Аналогічно описують креслення загального вигляду, складальні креслення та ін.

Після самоконтролю роботи студентом і перевірки ДР (текстової і графічної частини) консультантами вони затверджуються керівником, після цього ДР проходить нормоконтроль, а після усунення всіх зауважень, затверджуються завідувачем кафедри.

Креслення і схеми, представлені дипломником до захисту є основними документами, на підставі яких ДЕК оцінює якість його загально-інженерної і проектно-конструкторської підготовки.

6.4 Перелік документів, які подаються в ДЕК при захисті магістерських кваліфікаційних робіт

Перед початком засідань ДЕК кафедра подає до комісії такі документи:

- наказ (витяг з наказу) вищого навчального закладу про затвердження складу ДЕК із спеціальності;
- розклад роботи комісії;
- наказ про затвердження списків студентів за навчальними групами, допущених до захисту кваліфікаційних робіт;
- зведену відомість, завірену деканом факультету, про виконання студентом усіх вимог навчального плану і одержаних ним оцінок з теоретичних дисциплін, курсових проектів (робіт), виробничих практик;
- залікову книжку студентів.

Перед захистом магістерських кваліфікаційних робіт студент подає секретареві ДЕК такі матеріали:

- виконану роботу з усіма підписами;
- копію роботи на електронних носіях (CD-R або CD-RW) з додатками; вихідні тексти розробленого програмного забезпечення та у вигляді виконуваного або інсталяційного модуля;

- письмовий відгук керівника, в якому розкривається уміння студента самостійно працювати з літературними джерелами, ступінь творчої роботи магістранта, рівень теоретичної та практичної підготовки, уміння розв'язувати фахові завдання, новизну, актуальність роботи та можливості практичного використання отриманих результатів, можливість магістранта самостійно працювати за фахом, висновок про можливість допустити студента до відкритого захисту і мотивована оцінка роботи загалом;

- письмову рецензію на роботу, в якій рецензент характеризує: відповідність роботи темі та завданню; кожен розділ роботи, ступінь використання магістрантом останніх досягнень в галузі інформаційних технологій та інтелектуальних систем підтримання прийняття рішень; новизну та актуальність; грамотність і ясність викладення, якість оформлення і повноту виконання індивідуального завдання, відповідність вимогам стандартів, якість графічної частини. У висновку рецензент дає загальну оцінку, відзначає переваги та недоліки роботи, оригінальні та вдалі вирішення і оцінює кваліфікаційну роботу за чотирибальною шкалою: «відмінно», «добре», «задовільно», «незадовільно». Рецензування кваліфікаційних робіт доручають висококваліфікованим спеціалістам виробничих, наукових і проектних організацій, працівникам та викладачам. Негативна рецензія не є підставою для відхилення роботи від захисту;

До державної комісії можуть подаватися й інші матеріали, які характеризують наукову та практичну цінність виконаної роботи, а саме: наукові статті за темою роботи: документи, що підтверджують практичне застосування роботи та ін.

6.5 Підготовка до захисту та захист магістерської роботи

Виконана робота, підписана студентом та консультантами, подається студентом керівникові. Після перевірки роботи керівник підписує записку та разом зі своїм письмовим відгуком подає завідувачу кафедри. Завідувач кафедри після ознайомлення з роботою та відгуком вирішує питання про допуск магістранта до захисту, підписує титульний аркуш і надсилає роботу на рецензію.

Якщо завідувач кафедри не вважає можливим допустити студента до захисту роботи, це питання розглядається на засіданні кафедри, на яке запрошують керівника роботи.

Рішення кафедри з цього питання оформляється протоколом, який подається деканові факультету.

Студент знайомиться з рецензією на свою роботу до початку засідання ДЕК, на якому проводиться її захист.

Завідувач профільною кафедрою вживає заходів, щоб публічний захист робіт відбувався у підготованих до цього приміщеннях в урочистій обстановці. Захист може відбуватись як у вищому навчальному закладі, так і у філіях кафедри, підприємствах, в закладах та організаціях, для яких тематика робіт, що захищаються, становить науково-теоретичний та (або) практичний інтерес.

Розклад роботи Державної екзаменаційної комісії, узгоджений з головою комісії, затверджується першим проректором та оголошується випускникам не пізніше ніж за місяць до захисту робіт.

На засіданні Державної комісії складається протокол. Тривалість засідання не повинна перевищувати шести академічних годин на день. Захист однієї роботи не має тривати понад 30 хв. Для розкриття змісту роботи студентові відводиться не більше ніж 15 хвилин.

Після доповіді членами ДЕК або екзаменаторами, запрошеними на захист, магістранту можуть бути поставлені питання за темою кваліфікаційної роботи або інші, які стосуються теми магістерської кваліфікаційної роботи. Після того, як магістрант відповість на поставлені запитання, зачитується відгук керівника та рецензента.

Рішення ДЕК про оцінку знань, виявлених під час захисту магістерської кваліфікаційної роботи, а також про присвоєння студентам-випускникам відповідного освітнього рівня (кваліфікації) та видачу дипломів (загального зразка чи з відзнакою), приймається на закритому засіданні комісії відкритим голосуванням звичайною більшістю голосів членів комісії, які брали участь у засіданні. За однакової кількості голосів голос голови комісії є вирішальним. Під час цього засідання члени ДЕК підписують протоколи засідань.

Студентам, які успішно склали державні іспити, а також захистили магістерську кваліфікаційну роботу, рішенням Державної екзаменаційної комісії присвоюється кваліфікація відповідно до отриманої спеціальності і видається диплом встановленого зразка.

Студентові, який має оцінки «відмінно» не менше ніж з 75% усіх дисциплін навчального плану, а з решти дисциплін та індивідуальних завдань - оцінки «добре», захистив магістерську кваліфікаційну роботу з оцінкою «відмінно», а також виявив себе у науковій (творчій) роботі, що

підтверджується рекомендацією кафедри, видається диплом з відзнакою, що записується у протоколі засідання комісії.

У тих випадках, коли захист роботи не відповідає кваліфікаційній характеристиці, Державна комісія приймає рішення про те, що студент не захистив магістерську кваліфікаційну роботу, тобто отримав незадовільну оцінку, що зазначається у протоколі засідання комісії.

Студент, який отримав незадовільну оцінку при захисті магістерської кваліфікаційної роботи, відраховується з вищого навчального закладу. Йому видається академічна довідка встановленого зразка.

Якщо студент не з'явився на засідання ДЕК, то в протоколі комісії зазначається, що він є не атестованим через неприбуття на засідання комісії.

Студенти, що не атестовані у затверджений термін, мають право на повторну атестацію в наступний термін роботи Державної комісії протягом трьох років після закінчення вищого навчального закладу.

У випадках, коли захист магістерської кваліфікаційної роботи визнається незадовільним, Державна комісія встановлює, чи може студент подати на повторний захист ту саму роботу з доопрацюванням, чи він повинен опрацювати нову тему, визначену кафедрою.

Студентам, які при повторному захисті отримали незадовільну оцінку, видається академічна довідка встановленого зразка.

Після захисту рішення ДЕК про надання студентам, які захистили магістерські кваліфікаційні роботи, відповідної кваліфікації, оголошується наказом, після чого в урочистій обстановці їм вручаються дипломи.

Узагальнені результати захисту магістерських кваліфікаційних робіт обговорюються на засіданні кафедри та на засіданні Вченої ради факультету.

6.6 Подання магістерської роботи

Виступи перед аудиторією здавна були важливою і непростою галуззю розумової праці. Кожен оратор знає, яку велику допомогу можуть надати всілякі наочні посібники, що ілюструють матеріал, стимулюють увагу і зацікавленість аудиторії, полегшують орієнтацію у складних проблемах. З появою персональних комп'ютерів виникли спеціальні засоби підготовки презентацій, якими можна оснастити виступ.

Презентація - це набір слайдів, де є текст, графічні об'єкти, рисунки, кнопки. Презентація може містити звук, відео та анімацію - три основні компоненти мультимедіа Її демонструють на екрані монітора комп'ютера чи на великому екрані у залі за допомогою проекційної панелі і проектора.

Створюють презентації для того, щоб ілюструвати доповіді, рекламні повідомлення, демонструвати проекти, презентувати конференції, захист дипломних проектів, курсових робіт та дисертацій, монографій.

До складу програмного комплексу Microsoft Office входить спеціальний засіб підготовки візуального супроводу будь-яких доповідей - програма PowerPoint. За її допомогою ми можемо підготувати до виступу слайди, які потім можна надрукувати на прозорих плівках, папері, або просто демонструвати на екрані комп'ютера.

PowerPoint створює файл презентацій, що містить набір слайдів і має розширення імені pps, ppt, або експортований у інші, більш зручні для конкретних випадків графічні формати wmf, gif тощо.

Перш за все перед створенням презентації треба сформулювати тему, план і тези. Якщо текст з ілюстраціями вже заготовлений в інших програмах, то їх можна швидко й просто перенести у PowerPoint, упорядкувавши для виступу.

Презентація магістерської кваліфікаційної роботи будується за такою схемою подання матеріалу:

- тема роботи; ім'я та прізвище студента; вчене звання, ім'я та прізвище керівника;
- актуальність теми роботи;
- мета і завдання дослідження;
- об'єкт дослідження;
- предмет дослідження;
- методи дослідження;
- наукова новизна одержаних результатів;
- практичне значення одержаних результатів;
- результати проведеного системного аналізу;
- постановка та обґрунтування проблеми;
- методи вирішення проблеми;
- засоби вирішення проблеми;
- результати практичної реалізації;
- висновок.

У слайдах треба сконцентрувати увагу саме на тих аспектах магістерської кваліфікаційної роботи подання, яких є особливо важливим.

У презентації, окрім тексту, необхідно використовувати різноманітні зображення, ілюстрації, таблиці й діаграми.

Діаграми - вельми важлива форма комунікації. Коли діаграми добре продумані і складені, вони дають змогу подати матеріал набагато швидше і простіше, ніж таблиці. Вибір типу діаграми в першу чергу залежить не від

даних або одиниці вимірювань, а від *ідеї*, від того, що *доповідач* хоче показати, на що він робить акцент. Діаграми - це наочна допомога, допоміжні матеріали, а зовсім не *заміна письмовому та усному слову*. Діаграми поділяють на п'ять груп за типами порівняння:

- *по компонентні*; таке порівняння показує розмір кожного компоненту у відсотках від цілого;
- *позиційні*; позиційне порівняння показує розташування об'єктів щодо один одного;
- *часові*, група часових порівнянь поділяється на сегменти: окремо гістограми, лінійні діаграми та комбінації цих двох типів; гістограма показує зміни в часі;
- *частотні*; частотне порівняння показує, скільки об'єктів потрапляє в ряд послідовних інтервалів;
- *кореляційні*, кореляційне порівняння показує наявність або відсутність залежності між змінними.

Написи на слайдах повинні бути розбірливими для тих, хто сидить далі за всіх від екрану. Допустимим є від 5 до 10% відхилення розбірливості залежно від яскравості мультимедійного проектора, контрастності, створеної затемненням кімнати, а також яскравості зображення, пов'язаної з відстанню від проектора до екрану.

Для того, щоб зробити написи розбірливими треба дотримуватися таких рекомендацій:

- заокруглювати цифри або відкидати знаки після коми, якщо вони не істотні для передачі основної ідеї - завжди можна використати точні числа в самому виступі;
- використовувати координатну сітку або шкалу замість чисел на кінцях стовпців і лінійок або усередині них;
- де можливо використовувати скорочення та аббревіатури, якщо це не вносить плутанину;
- редагувати написи: скорочувати 10 слів до чотирьох, чотири слова до трьох, три слова до двох;
- видалити виноски. Цю інформацію подають в усному повідомленні, якщо вона настільки важлива.

Треба зазначити, що ця таблиця і рекомендації не підходять для випадку, коли проблема немає простого рішення.

При розробленні презентації керуйтеся правилами:

- чим більше, тим краще;
- чим простіше, тим краще;

- чим більше уяви, тим краще;
- чим різноманітніше, тим краще.

Іноді поділ докладної діаграми на декілька не вирішує проблему розбірливості. У цьому випадку діаграму треба спростити до одного розбірливого слайду.

При формуванні слайдів презентації колір треба використовувати за призначенням, а не для краси. Значна частина діаграм, забезпечує бажаний рівень візуальної дії, будучи чорно-білими. Переклад презентації в чорно-білий формат - хороша перевірка її ефективності: якщо вона зрозуміла і в чорно-білого виконання, то все гаразд; якщо ні - то колір їй навряд чи допоможе. Ми живемо в кольоровому світі й сучасні технології дають змогу дуже легко створювати кольорові слайди.

Ось декілька порад з їхньої підготовки.

1) Для елементів діаграми використовують холодні кольори (синій і зелений), а для виділення конкретних елементів віддають перевагу жовтому і білому.

2) Використання кольору - це справа самого доповідача. Кольори використовуйте не просто для краси, а з певною метою, а саме:

- *щоб виділити*; наприклад, який-небудь компонент кругової діаграми, один сегмент гістограми або лінійної діаграми, одну лінію тренда, ряд цифр, заголовок;

- *щоб звернути увагу на лейтмотив презентації*; наприклад, позначення даних по вашій компанії одним і тим же кольором на всіх слайдах презентації;

- *щоб відокремити*; наприклад, фактичні дані від прогнозів, одну групу стовпців або лінійок від іншої, одну лінію тренда від іншої;

- *щоб символічно зобразити*; наприклад, червоним кольором - збитки, зеленим - прибуток. Червоний колір - зупинка, жовтий - увага, зелений - рух.

3) Для кращого сприйняття доповіді треба використовувати спеціальні ефекти.

ПЕРЕЛІК РЕКОМЕНДОВАНИХ ДЖЕРЕЛ

1. Bui Minh Duc. Real-Time Object Uniform Design Methodology with UML. - Springer, 2007.
2. Helmer Olaf. Analysis of the Future: The Delphi Method. - Santa Monica: Rand Corporation, 1969.
3. Акофф Р Системы, организации и междисциплинарные исследования / Акофф Р - М.: Прогресс, 1969. - 164 с. - (Исследования по общей теории систем).
4. Акофф Р О целеустремленных системах / Акофф Р., Эмери Ф. - М.: Сов. Радио, 1974. - 272 с.
5. Андрейчиков А. В. Анализ, синтез, планирование решений в экономике / А. В. Андрейчиков, О. Н. Андрейчикова. - М.: Финансы и статистика, 2000. - 368 с.
6. Анфилатов В. С. Системный анализ в управлении / Анфилатов В. С., Емельянов А. А., Кукушкин А. А. - М.: Финансы и статистика, 2002. - 368 с.
7. Атре Ш. Структурный подход к организации баз данных / Атре Ш. - М.: Финансы и статистика, 1983.-320 с.
8. Бекаревич Ю. Самоучитель Microsoft Access 2000 / Ю. Бекаревич, Н. Пушкина. - СПб.: БХВ-Санкт-Петербург, 1999. - 480 с.
9. Белкин А. Р. Принятие решений: комбинаторные модели аппроксимации информации / А. Р. Белкин, М. Ш. Левин. - М.: Наука, 1990. - 157 с.
10. Бемер С. MS Access / Бемер С.; [пер. с нем.]. - СПб.: ВHV-Санкт-Петербург, 1995. -448 с.
11. Берко А. Ю. Застосування баз даних: навч. посібник / А. Ю. Берко, О. М. Верес. - Львів: Ліга-Прес, 2007. - 208 с.
12. Берко А. Ю. Організація баз даних: практичний курс / А. Ю. Берко, О. М. Верес. Львів: Видавництво Національного університету «Львівська політехніка», 2003. 150 с.
13. Берко А. Ю., Верес О. М. Організація баз даних та знань: Методичні вказівки до виконання курсових робіт для студентів базового напрямку «Комп'ютерні науки» спеціальності «Інтелектуальні системи прийняття рішень» / А. Ю. Берко, О. М. Верес. - Львів: Видавництво Національного університету «Львівська політехніка», 2007. - 64 с.
14. Берко А. Ю. , Верес О.М. Проектування та застосування реляційних баз даних. Методичні вказівки до курсових робіт з дисципліни «Організація баз даних та знань» / А. Ю. Берко, О. М. Верес. - Львів: Видавничий центр ІППТ при НУ «Львівська політехніка», 2005; - 120 с.

15. Берко А. Ю. Системы баз данных та знань. Книга 1. Організація баз даних: навч. посібник [для студ. вищ. навч. закл.] / Берко А. Ю., Верес О. М., Пасічник В.В. - Львів: «Магнолія 2006», 2008. - 456 с. - (Серія «Комп'ютинг»),
16. Берко А. Ю. Системы електронної конвент-комерції: Монографія / Берко А. Ю., Висоцька В.'А., Пасічник В. В. - Львів: Видавництво Національного університету «Львівська політехніка», 2009. -612 с.
17. Берко А. Ю., Верес О.М. Теоретичні основи баз даних: Конспект лекцій для студентів Інституту комп'ютерних наук та інформаційних технологій / А. Ю. Берко, О. М. Верес. - Львів: Видавництво Національного університету «Львівська політехніка», 2007. - 190 с.
18. Бишоп Дж. С# в кратком изложении / Дж. Бишоп, Н. Хорспул. - М.: Бином, 2005. - 472 с.
19. Боггс У. UML и Rational Rose / У. Боггс, М. Боггс. - М.: Изд. «ЛОРИ», 2008. - 600 с.
20. Бойко В. В. Проектирование баз данных информационных систем / В. В. Бойко, В. М. Савинков. - М.: Финансы и статистика, 1989. - 351 с.
21. Борисов А. Н. Диалоговые системы принятия решений на базе мини-ЭВМ: Информационное, математическое и программное обеспечение / Борисов А. Н., Вильюмс Э. Р. Сукур Л. Я. - Рига: Зинатне, 1986. - 999 с.
22. Буров Є. В. Комп'ютерні мережі / Буров Є.В. - Львів: БаК, 1999. - 468 с.
23. Буров Є. В. Комп'ютерні мережі /Буров С. В.-2-е вид., випр. і доповн. - Львів: БаК, 2003.-584 с.: іл.
24. Буров Є. В. Комп'ютерні мережі: підручник / Буров Є. В. - Львів: «Магнолія-плюс», 2007.-262 с.
25. Буч Г. Объектно-ориентированный анализ и проектирование с примерами приложений на С++ / Буч Г. - 2-е изд.; [пер. с англ.]. - М.: Издательство Бином; СПб.: Невский диалект, 1999. - 720 с.
26. Буч Г. Язык UML. Руководство пользователя / Буч Г., Рамбо Дж., Джекобсон А.; [пер. с англ.]. - М.: ДМК, 2000. - 496 с.
27. Вейскас Дж. Эффективная работа с СУБД Microsoft Access 7.0 для Windows 95 / Вейскас Дж. - СПб.: Питер, 1997. - 848 с.
28. Вендров А. М. CASE-технологии. Современные методы и средства проектирования информационных систем / Вендров А. М. - М: Финансы и статистика, 2000. - 176 с.
29. Вендров А. М. Проектирование программного обеспечения экономических информационных систем: учеб. / Вендров А. М. - М.: Финансы и статистика, 2000. - 256 с.

30. Вилкас Э. Н. Решения: теория, информация, моделирование / Э. Н. Вилкас, Е. З. Май-минас. - М.: Радио и связь, 1981. - 328 с.
31. Гамма Э. Приемы объектно-ориентированного проектирования. Паттерны проектирования / Э. Гамма, Р. Хелм, Р. Джонсон, Дж. Влиссидес; [пер. с англ.]. - СПб.: Питер, 2001. - 365 с.
32. Гарсиа-Молина Г. Системы баз данных: полный курс / Гарсиа-Молина Г., Ульман Дж., Уидом Дж.; [пер. с англ.]. - М.: Издательский дом «Вильямс», 2003. - 1088 с.: ил. - Парал. тит. англ.
33. Глобальні інформаційні системи та технології: моделі ефективного аналізу, опрацювання та захисту інформації: монографія / Пасічник В. В., Жежнич П. І., Краведь Р. Б., Пелешишин А. М., Тарасов Д. О. - Львів: Вид-во Нац. ун-ту «Львівська політехніка», 2006. - 348 с.
34. Губанов В. А. Введение в системный анализ / Губанов В. А., Захаров В. В., Коваленко А. Н. - Л.: Из-во ЛГУ, 1979. - 232 с.
35. Дейт К. Дж. Введение в системы баз данных / Дейт К. Дж. - 7-е изд.; [пер. с англ.]. М.: Издательский дом «Вильямс», 2001. — 1072 с.: ил. — Парал. тит. англ.
36. Дейт К. Дж. Введение в системы баз данных / Дейт К. Дж.; [пер. с англ.]. - К.: Диалектика, 1998. - 784 с.
37. Дейт К. Дж. Введение в системы баз данных / Дейт К. Дж. - 8-е изд.; [пер. с англ.]. М.: Издательский дом «Вильямс», 2005. — 1328 с.: ил. — Парал. тит. англ.
38. Дейт К. Дж. Руководство по реляционной СУБД БВ2 / Дейт К. Дж. - М.: Финансы и статистика, 1988. - 320 с.
39. Державні санітарні правила і норми робота з візуальними дисплейними терміналами електронно-обчислювальних машин: ДСанПН 3.3.2.007-98.
40. Джексон Г. Проектирование реляционных баз данных для использования с микро- ЭВМ / Джексон Г. - М.: Мир, 1991.-252 с.
41. Довідник офіційного опонента. Збірник нормативних документів та інформаційних матеріалів з питань експертизи дисертаційних досліджень / Упорядник Ю. І. Цеков; за редакцією Р. В. Бойка. - 2-е вид., випр. і доповн. - К: Редакція «Бюлетеня Вищої атестаційної комісії України», видавництво «Толока», 2008. - 64 с.
42. Долин П. А. Справочник по технике безопасности / Долин П. А. - М.: Энергоато-миздат, 1984. - 639 с.
43. Дюбуа Д. Теория возможностей: Приложения к представлению знаний в информатике / Д. Дюбуа; [пер.с фр. В. Б. Тарасова под ред. С. А. Орловского]. - М.: Радио и связь, 1990.-286 с.

44. Естественное и искусственное освещение (с изменениями). СНиП 11-4-79.
45. Заяць В. М. Логічне і функційне програмування: навч. посібник/ В. М. Заяць, М. М. Заяць. - Львів: Бескид Біт, 2006. - 352 с.
46. Згуровський М.З. Основи системного аналізу / М. З. Згуровський, Н. Д. Панкратова. К: ВНУ, 2007.-405 с.
47. Зелинский С. Э. Фильтры для экранов мониторов и ваше здоровье / С. Э. Зелинский // Компьютеры + Программы. - 1995. - №3. - С. 77—78.
48. Інтелектуальні системи, базовані на онтологіях: Монографія / Д. Г. Досин, В. В. Литвин, Ю. В. Нікольський, В. В. Пасічник. - Львів: Видавничий дім «Цивілізація». 2009.-414 с.
49. Карпова Т.С. Базы данных: модели, разработка, реализация / Карпова Т. С. - СПб.: Питер, 2002. - 304 с.: ил.
50. Кармайлл Э. Быстрая и качественная разработка программного обеспечения / Э. Кармайлл, Д. Хейвуд. - М.: Изд. дом «Вильямс», 2003. - 400с.
51. Катренко А. В. Дослідження операцій: підручник з грифом МОН / Катренко А. В. - Львів: «Магнолія-2006», 2007. - 480 с.
52. Катренко А. В. Дослідження операцій: підручник з грифом МОН / Катренко А. В. 3-є вид., виправ, і доп.; - Львів: «Магнолія-2006», 2009. - 352 с. - (Серія «Комп'ю- тинг»).
53. Катренко А. В. Системний аналіз об'єктів та процесів комп'ютери-зації: підручник з грифом МОН / Катренко А.В. - Львів: «Новий світ 2000», 2003. - 424 с.
54. Катренко А. В. Системний аналіз: підручник з грифом МОН / Катренко А. В. - Львів: «Магнолія-2006», 2009. - 352 с. - (Серія «Комп'ютинг»),
55. Катренко А. В. Теорія прийняття рішень: підручник з грифом МОН / Катренко А. В., Пасічник В. В., Пасько В. П. - К.: Видавнича група BVH, 2009. - 448 с.: ил.
56. Катренко Л. А. Охорона праці в галузі освіти / Л. А. Катренко, І. П. Пістун. - Суми: Універ. книга, 2001. - 344 с.
57. Квартани Т. Визуальное моделирование с помощью Rational Rose 2002 и UML / Т. Квартани. - М.: Изд. дом «Вильямс», 2003. - 192 с.
58. Кини Р. Л. Принятие решений при многих критериях: замещения и предпочтения / Р. Л. Кини, Х. Райфа. - М.: Радио и связь, 1981. - 560 с.
59. Кириллов В.В. Основы проектирования реляционных баз данных: Учебное пособие Электронный ресурс / Кириллов В. В. - Санкт- Петербургский Государственный институт точной механики и оптики, 2000. - Режим доступа до посіб.: <http://www.citforum.ru/database/dbguide/index.shtml>.

60. Клейнер Г. Б. Производственные функции / Клейнер Г. Б. - М.: ФиС, 1986. - 238 с.
61. Кнорринг Г. Я. Справочная книга для проектирования электрического освещения / Кнорринг Г. Я. - М.: Энергия, 1976. - 284с.
62. Козелецкий Ю. Психологическая теория решений / Козелецкий Ю.; [пер. с польск.]. М.: Прогресс, 1979. - 504 с.
63. Колесников Л. Основы теории системного подхода / Колесников Л. - К.: Наук, думка, 1988.- 176 с.
64. Конноли Т. Базы данных: проектирование, реализация и сопровождение. Теория и практика: учеб. пособие / Томас Конноли, Каролин Бегг, Анна Страчан. - 2-е изд.; [пер. с англ.]. - М.: Издательский дом «Вильямс», 2000. - 1120 с.: ил. - Парал. тит. англ.
65. Конноли Т. Базы данных: проектирование, реализация и сопровождение. Теория и практика: учеб. пособие / Томас Конноли, Каролин Бегг. - 3-е изд.; [пер. с англ.]. - М.: Издательский дом «Вильямс», 2003. - 1440 с.: ил. - Парал. тит. англ.
66. Крачтен Ф. Введение в Rational Unified Process / Крачтен Ф.; [пер. с англ.]. - М.: Вильямс, 2002. - 239 с.
67. Крёнке Д. Теория и практика построения баз данных / Крёнке Д. - 8-е изд. - СПб.: Питер, 2003. - 800 с.: ил. - (Серия «Классика computer science»).
68. Кузнецов С. Д. Будущие направления исследований в области баз данных: десять лет спустя. [Электронный ресурс] / Кузнецов С. Д. - Институт проблем программирования РАН, 1999. - Режим доступа: http://www.citforum.ru/database/articles/future_01.shtml.
69. Кузнецов С. Д. Крупные проблемы и текущие задачи исследований в области баз данных / [Электронный ресурс] / Кузнецов С. Д. - Институт проблем программирования РАН, 2003. - Режим доступа: <http://www.citforum.ru/database/articles/problems>.
70. Кузнецов С. Д. Основы современных баз данных: Информационно-аналитические материалы / [Электронный ресурс] / Кузнецов С. Д. - Центр информационных технологий, 2001. - Режим доступа: <http://www.citforum.ru/database/osbd/contents.shtml>.
71. Кузнецов С. Д. Три манифеста баз данных: ретроспектива и перспективы / [Электронный ресурс] / Кузнецов С. Д. - Институт проблем программирования РАН, 2003. - Режим доступа: <http://www.citforum.ru/database/articles/manifests/>.
72. Кухтенко А. И. Кибернетика и фундаментальные науки / Кухтенко А. И. - К: Наукова думка, 1987. - 140 с.

73. Ларичев О. И. Наука и искусство принятия решений / Ларичев О. И. - М.: Наука. 1979.-480 с.
74. Лемке Д. Microsoft Office Visio 2007 / Джуди Лемке. - Изд.: ЭКОМ Паблишерз. 2008.-368 с.
75. Леоненков А.В. Самоучитель UML/ А.В. Леоненков. - СПб.: БХВ-Петербург, 2001. -304 с.
76. Литвак Б. Г. Экспертная информация: Методы получения и анализа / Литвак Б. Г. - М.: Радио и связь, 1982. - 184 с.
77. Литвин В. В. Інтелектуальні системи. Моделі та методи побудови: навч.посіб. / Литвин В. В., Пасічник В. В., Яциптин Ю. В. - К.: Університет «Україна», 2007. - 534 с.
78. Литвин В. В. Інтелектуальні системи: підручник / Литвин В. В., Пасічник В. В.: Яцишин Ю. В. - Львів.: «Новий світ-2000», 2009. - 406 с. - (Серія «Комп'ютинг»),
79. Лопухин М. М. ПАТТЕРН - метод планирования и прогнозирования / Лопухин М. М. - М.: Советское радио, 1971. - 159 с.
80. Лотов А. В. Введение в экономико-математическое моделирование / Лотов А. В. - М.: Наука, 1984.-356 с.
81. Интрилигатор М. Математические методы оптимизации и экономическая теория Интрилигатор М. - М.: Прогресс, 1975. - 606 с.
82. Макаров И. М. Целевые комплексные программы / Макаров И. М., Соколов В. Б., Абрамов А. Л. - М.: Знание, 1980. - 135 с.: ил.
83. Маллинс К. С. Администрирование баз данных. Полное справочное руководство по методам и процедурам. Учебно-справочное издание / Крейг С.Маллинс. [пер. с англ.]. - М.: КУДИЦ-ОБРАЗ, 2003. - 752 с.
84. Мартин Дж. Планирование развития автоматизированных систем / Дж. Мартин. - М.: Финансы и статистика, 1984. - 196 с.
85. Мартиросова В. Г. Особливості праці користувачів ЕОМ / В. Г. Мартиросова // Охорона праці. - 1995. -№1.- С. 10—13.
86. Мейер Д. Теория реляционных баз данных / Мейер Д.; [пер. с англ.]. - М.: Мир, 1987.-608 с.
87. Месарович М. Теория многоуровневых иерархических систем / Месарович М., Мако Д., Такахара И. - М.: Мир, 1973. - 339 с.
88. Месарович М. Общая теория систем: Математические основы / М. Месарович, И. Такахара.-М. Мир, 1978.-311с.
89. Михеева В. Microsoft Access 2000 / В. Михеева , И. Харитоновна. - СПб.: БХВ - Санкт-Петербург, 1999. - 1088 с.
90. Моисеев Н. Н. Математические задачи системного анализа / Моисеев Н. Н. - М.: Наука, 1981.-488 с.

91. Научно-методические аспекты подготовки магистерских диссертаций: учеб. пособие. / С. И. Дворецкий, Е. И. Муратова, О. А. Корчагина, С. В. Осина. Тамбов: ТО- ГУП «Тамбовполиграфиздат», 2006. - 84 с.
92. Нікольський Ю. В. Дискретна математика: підручник з грифом МОН / Нікольський Ю. В., Щербина Ю. М., Пасічник В. В. - К.: Видавнича група BVH, 2007. - 368 с.
93. Нікольський Ю. В. Дискретна математика: підручник з грифом МОН / Нікольський Ю. В., Щербина Ю. М., Пасічник В. В. - 2-ге вид. - Львів: «Магнолія-плюс», 2007. - 608 с.
94. Нікольський Ю. В. Дискретна математика: підручник з грифом МОН / Нікольський Ю. В., Щербина Ю. М., Пасічник В. В. - Львів: «Магнолія-2006», 2009. - 432 с. - (Серія «Комп'ютинг»).
95. Новиков Ф. Microsoft Office 2000 в целом / Ф. Новиков, А. Яценко. - СПб.: БХВ- Санкт-Петербург 1999. - 728 с.
96. Нотон П. Полный справочник по Java / П. Нотон, Г. Шилдт. - К.: Диалектика, 1997. - 592 с.
97. Общие санитарно-гигиенические требования к воздуху рабочей зоны: ГОСТ 12.1.005-88. ССБТ.
98. Одрин М. В. Морфологический анализ систем / М. В. Одрин, С. С. Картавов. - К.: Наукова думка, 1977. - 183 с.
99. Опрацювання інформації. Символи та умовні позначення блок-схем даних, програм та систем, схем програмних мереж і системних ресурсів. ISO 5807-85.
100. Пасічник В. В. Організація баз даних та знань. / В. В. Пасічник, В. А. Резніченко. - К.: Видавнича група BVH, 2006. - 384 с.
101. Пасичник В. В. Реляционные модели баз данных / В. В. Пасичник, А. А. Стогний. - М.: ЦНИИАТОМИНФОРМ, 1983. - 268 с.
102. Пасічник В. В. Сховища даних: навч. посіб. / В. В. Пасічник, Н. Б. Шаховська. - Львів: «Магнолія-2006», 2008. - 496 с. - (Серія «Комп'ютинг»).
103. Пасічник В. В. Сховища та простори: Монографія / В. В. Пасічник, Н. Б. Шаховська. - Львів: Видавництво Національного університету «Львівська політехніка», 2009. - 244 с.
104. Пелешишин А. М. Розроблення комплексних Веб-сайтів за допомогою мови програмування Реді: навч. посібник / Пелешишин А. М., Жежнич П. І., Марковець О. В. - Львів: Вид-во Нац. ун-ту «Львівська політехніка», 2007. - 164 с.
105. Перегудов Ф. И. Введение в системный анализ / Ф. И. Перегудов, Ф. П. Тарасенко. М.: Высшая школа, 1989. - 367 с.

106. ПК и здоровье пользователей // ЧИП. - 1997. - №7. — С. 80—84.
107. Положення про магістратуру НТУУ «КПІ» / Уклад. В. П. Головенкін. За заг. ред. Ю. 1. Якименка. - К.: ВПК «Політехніка», 2007. - 36 с.
108. Правила охорони праці під час експлуатації електронно-обчислювальних машин: ДНАОП 0.00-1.31-99.
109. Проектирование реляционной базы данных: метод, указания к курсовому проектированию по курсу «Базы данных» / Московский государственный институт электроники и математики; [сост. Карпова И. П.]. - М., 2003. - 28 с.
110. Производственные здания промышленных предприятий. СНиП 2.09.02-85.
111. Противопожарные нормы проектирования зданий и сооружений. СНиП 2.01.02-85.
112. ПУЭ. Правила устройства электроустановок. - М.: Энергоиздат, 1982. - 89 с.
113. Пушников А. Ю. Введение в системы управления базами данных. Часть 1. Реляционная модель данных: Учебное пособие / [Электронный ресурс] / Пушников А.Ю. Уфа: Изд-е Башкирского ун-та. 1999. - 108 с. - ИБВК 5-7477-0350-1. - Режим доступа: <http://www.citforum.ru/database/dblearn/index.shtml>.
114. Пушников А. Ю. Введение в системы управления базами данных. Часть 2. Нормальные формы отношений и транзакции: Учебное пособие / [Электронный ресурс] / Пушников А.Ю. - Уфа: Изд-е Башкирского ун-та. 1999. - 138 с. - 18ВЫ 5-7477-0351-Х. - Режим доступа: <http://www.citforum.ru/database/dblearn/index.shtml>.
115. Рабочее место при выполнении работ сидя. Общие эргономические требования: ГОСТ 12.2.032 - 78. ССБТ.
116. Райордан Р. Основы реляционных баз данных / Райордан Р.; [пер. с англ.]. - М.: Издательско-торговый дом «Русская Редакция», 2001. - 384 с.: ил.
117. Райхман Э. П. Экспертные методы в оценке качества товаров / Э. П. Райхман, Г. Г. Азгальдов. - М.: Экономика, 1974. - 151 с.
118. Ревунков Г. И. Базы и банки данных и знаний: учеб. для вузов / Ревунков Г. И., Самохвалов Э. Н., Чистов В. В.; под ред. В. Н. Четверикова. — М.: «Высшая школа», 1992.-367 с.
119. Рейльян Я. Р. Аналитическая основа принятия управленческих решений / Рейльян Я. Р. - М.: «Финансы и статистика», 1989. - 206 с.
120. Роб П. Системы баз данных: проектирование, реализация и управление / Роб Питер, Коронел Карл. - 5-е изд., перераб. и доп.; [пер. с англ.]. - СПб.: БХВ-Петербург, 2004. -1040с.: ил.

121. Розен В. В. Цель - оптимальность - решение / Розен В. В. - М.: Радио и связь, 1982.- 168 с.
122. Руа Б. Классификация и выбор при наличии нескольких критериев: в сб.: Вопросы анализа и процедуры принятия решений / Руа Б. - М.: Мир, 1976. - С. 80—107.
123. Саати Т. Аналитическое планирование. Организация систем / Т. Саати, К. Кернс. - М.: Радио и связь, 1991. - 224 с.
124. Санитарные нормы микроклимата производственных помещений. СН 4088-86.
125. Системы управления базами данных и знаний: справочное издание; под. ред. А.Н. Наумова. - М.: Финансы и статистика, 1991. - 348 с.
126. Средства защиты в машиностроении: справочник / [С.В. Белов, А.Ф. Козьяков и др.]. - М.: Машиностроение, 1989. - 368 с.
127. Солнышков Ю. С. Обоснование решений (методологические вопросы) / Солнышков Ю. С. - М.: Экономика, 1980. - 168 с.
128. Sommerwill I. Инженерия программного обеспечения, 6-е изд. / И. Sommerwill; [пер. с англ.]. - М.: Изд. Дом «Вильямс», 2002. - 624 с.
129. Страуструп Б. Язык программирования C++/ Б. Страуструп. - М.; СПб.: Бином, Невский Диалект, 2004. - 1104 с.
130. Тиори Т. Проектирование структур баз данных: у 2 кн. / Т. Тиори, Дж. Фрай. - М.: Мир, 1985 - . Кн. 1.- 1985.-287 с. Кн. 2,- 1985.-320 с.
131. Трахтенгерц Э. А. Компьютерная поддержка принятия решений: научно-практ. изд. / Э. А. Трахтенгерц. - М.: СИНТЕГ, 1998. - 376с. - (Серия «Информатизация России на пороге XXI века»).
132. Ульман Дж. Базы данных на Паскале / Ульман Дж. - М.: Машиностроение, 1990. -386 с.
133. Ульман Дж. Основы систем баз данных/Ульман Дж.; [пер. с англ.]. - М.: Финансы и статистика, 1983. - 334 с.
134. Ульман Дж. Введение в системы баз данных / Ульман Джеффри Д., Уидом. Дженнифер. [пер. с англ.]. - М.: Издательство «Лори», 2000. - 374 с.
135. Фаулер М. UML в кратком изложении. Применение стандартного языка объектного моделирования / М. Фаулер , К. Скотт; [пер. с англ.]. - М.: Мир, 1999. — 191 с.
136. Федулов А. А. Введение в теорию статистически ненадежных решений / Федулов А. А., Федулов Ю. Т., Цыгичко В. Н. - М.: Радио и связь, 1979, - 276 с.
137. Фишберн П. Теория полезности для принятия решений / Фишберн П. - М.: Наука, 1978.-352 с.

138. Хаббард Дж. Автоматизированное проектирование баз данных / Дж. Хаббард. - М.: Мир, 1984. -294 с.
139. Хансен Г. Базы данных: разработка и управление / Гэри Хансен, Джэймс Хансен; [пер. с англ.]. - М.: ЗАО «Издательство БИНОМ», 1999. - 704 с.: ил.
140. Хомоненко А. Д. Базы данных: учебник для высших учебных заведений / Хомоненко А. Д., Цыганков В. М., Мальцев М. Г.; под ред. проф. А. Д. Хомоненко. - 2-е изд., доп. и перераб. - СПб.: КОРОНА принт, 2002. - 672 с.
141. Цикритизис Д. Модели данных / Д. Цикритизис, Ф. Лоховски. - М.: Финансы и статистика, 1985. - 344 с.
142. Черноруцкий И. Г. Методы принятия решений / И. Г. Черноруцкий. - СПб.: БХВ- Петербург, 2005. - 416 с.: ил.
143. Черняк Ю. Системный анализ в управлении экономикой / Черняк Ю. - М.: Экономика, 1975. - 265 с.
144. Четвериков В. Н. Базы и банки данных / Четвериков В. Н., Ревунков С. И., Самохвалов Е. Н. - М.: «Высшая школа», 1987. - 268 с.
145. Мулен Э. Кооперативное принятие решений: Аксиомы и модели / Э. Мулен. - М.: Мир, 1991.-463 с.
146. Юдин Д. Б. Число и мысль вып. 8. Сер. нар. ун-т. Естественно научн. фак. / Д. Б. Юдин , А. Д. Юдин. - М.: Знание, 1985. 192 с.
147. Шилдт Г. Самоучитель C++ / Г. Шилдт. - СПб.: БХВ-Петербург, 2005. — 688 с.
148. Шилдт Г. Искусство программирования на JAVA / Г. Шилдт, Д. Холмс. - М.: Изд. дом «Вильямс», 2005. - 336 с.
149. Шилдт Г. Полный справочник по C# / Г. Шилдт. - М.: Изд. дом «Вильямс», 2004. -752 с.
150. Шум. Общие требования безопасности: ГОСТ 12.1.003-83. ССБТ.

ДОДАТКИ

Права та обов'язки студента-дипломника

Студент-дипломник має право:

- вибирати тему магістерської роботи з числа запропонованих випусковою кафедрою або запропонувати власну тему з необхідним обґрунтуванням доцільності її розробки і можливості виконання. У всіх випадках він звертається з відповідною заявою на ім'я завідувача випускової кафедри.

- в разі потреби отримати окреме робоче місце для роботи над дипломним проектом у спеціальній аудиторії, обладнаній комп'ютерною технікою, необхідним наочним приладдям, довідковою літературою та стандартами, зразками фрагментів пояснювальної записки та графічного матеріалу, методичними вказівками щодо виконання та оформлення складових дипломної роботи та ін.;

- користуватися лабораторною та інформаційною базою кафедри, приладами, вимірювальною технікою тощо для проведення натурного експерименту, математичного моделювання або наукових досліджень за темою магістерської роботи;

- отримувати консультації керівника та консультантів магістерської роботи;

- попереднього (на кафедрі), основного (у ДЕК) захисту магістерської роботи.

Студент зобов'язаний:

- своєчасно вибрати тему магістерської роботи та отримати конкретні завдання від керівника роботи на відбір та опрацювання матеріалів, необхідних для дипломного проектування під час проведення стажування з фаху;

- на практиці, крім виконання її програми, максимально ознайомитися з практичною реалізацією питань організації та управління виробництвом (підприємством, фірмою тощо), охороною праці, вирішенням питань екології, техніко-економічних і спеціальних питань за темою магістерської роботи;

- після складання звіту і заліку за результатами стажування з фаху (дослідницької практики) отримати завдання на дипломне проектування у повному обсязі, з'ясувати зміст, особливості та вимоги до виконання окремих пунктів завдання;

– скласти та узгодити з керівником роботи календарний план-графік виконання дипломного проектування з урахуванням різної трудомісткості розділів, необхідності перевірки матеріалів керівником та консультантами, отримання відгуку керівника і рецензії та своєчасного надання повністю підготовленої і перевіреної та допущеної до захисту проекту не менш ніж за 5 днів до його захисту в ДЕК;

– регулярно, не менше одного разу на два тижні, інформувати керівника про стан виконання проекту відповідно до плану-графіку, надавати на його вимогу необхідні матеріали для перевірки;

– самостійно виконувати індивідуальну роботу або індивідуальну частину комплексної роботи;

– при розробці питань враховувати сучасні досягнення науки і техніки, використовувати передові методики наукових та експериментальних досліджень, приймати оптимальні рішення із застосуванням системного підходу;

– при проектуванні конкретних зразків техніки та розробці технологічних процесів виробництва, проведенні різного роду розрахунків та моделюванні використовувати сучасні комп'ютерні технології;

– відповідати за правильність прийнятих рішень обґрунтувань, розрахунків, якість оформлення пояснювальної записки та графічного матеріалу і відповідність їх вимогам методичних рекомендацій (вказівок) з дипломного проектування випускової кафедри існуючим нормативним документам та державним стандартам;

– дотримуватися календарного плану-графіку виконання дипломного проекту, встановлених правил поведінки в лабораторіях і кабінетах дипломного проектування, своєчасно та адекватно реагувати на зауваження та рекомендації керівника і консультантів ДП;

– у встановлений термін подати проект для перевірки керівнику та консультантам і після усунення їх зауважень повернути керівнику для отримання його відгуку;

– отримати всі необхідні підписи на титульному листі пояснювальної записки та кресленнях, а також резолюцію завідувача випускової кафедри про допуск до захисту;

– особисто подати ДП, допущений до захисту, рецензенту; на його вимогу надати необхідні пояснення з питань, які розроблялися в ДР;

– ознайомитися зі змістом відгуку керівника і рецензії та підготувати (у разі необхідності) аргументовані відповіді на їх зауваження при захисті ДР у ДЕК.

Вносити будь-які зміни або виправлення в ДР після отримання відгуку керівника та рецензії забороняється.

- у строк, визначений секретарем ДЕК, надати магістерську роботу до ДЕК;
- за згодою керівника роботи визначити дату та пройти попередній захист ДР на кафедрі;
- своєчасно прибути на захист ДР або попередити завідувача випускової кафедри та голову ДЕК (через секретаря ДЕК) про неможливість присутності на захисті із зазначенням причин цього та наступним наданням документів, які засвідчують поважність причин.

Зразок оформлення заяви на закріплення теми магістерської роботи

Зав. каф. комп'ютерних наук
д.т.н., професору Приймаку М.В.
студента групи СНм-51
Петренка Петра Петровича

ЗАЯВА

Прошу затвердити мені дипломну (дипломний проект, магістерську)
(відповідне підкреслити)
роботу на тему: Розробка інформаційної системи для інтервального аналізу
сигналів
та керівника роботи: Іваненка Івана Івановича

Мій № мобільного телефону 000 1234567
(або № домашнього телефону 0000 123456)
електронна пошта 1@1.com

“ 15 ” грудня 2013 р.

(підпис заявника)

* Примітка. При написанні заяви на виконання магістерської роботи пусті рядки бланку заповнюються рукописним способом, текст якого повинен бути виконаний креслярським шрифтом згідно з ГОСТ 2.304-81 з висотою букв і цифр не менше 2,5 мм. Цифри і букви необхідно писати чітко, виконаними в **чорному кольорі**.

Зразок оформлення титульного аркуша магістерської роботи

Форма № Н-9.02

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя
(повне найменування вищого навчального закладу)

Комп'ютерно-інформаційних систем і програмної інженерії
(назва факультету)

Комп'ютерних наук
(повна назва кафедри)

ПОЯСНЮВАЛЬНА ЗАПИСКА

до дипломного проекту (роботи)

магістр

(освітньо-кваліфікаційний рівень)

на тему: *Розробка інформаційної системи для імітаційного моделювання
світлових біосигналів методом ортогональних розкладів*

Виконав: студент (ка) 5 курсу, групи СНм-51
напряму підготовки (спеціальності)

8.05010101 "Інформаційні управляючі системи та технології"
(шифр і назва напряму підготовки, спеціальності)

Іваненко І.І.

(прізвище та ініціали)

Керівник Петренко П.П.
(прізвище та ініціали)

Рецензент Сидоренко С.С.
(прізвище та ініціали)

м. Тернопіль – 201 _

Рекомендації щодо розробки титульного аркуша дипломної роботи:

(назва факультету)

Тут вписується повна назва випускового факультету студента дипломника.

(повна назва кафедри)

Тут вписується повна назва випускової кафедри студента дипломника.

(освітньо-кваліфікаційний рівень)

Тут вписується назва освітньо-кваліфікаційного рівня студента випускника: бакалавр, спеціаліст, магістр.

На тему:

Назва теми повинна бути, за можливості, короткою, чітко і конкретно відображати мету та основний зміст дослідження і бути однаковою у наказі ректора про закріплення тем і керівників за студентами, завданні на ДР, титульному аркуші пояснювальної записки, кресленнях, документах ДЕК та в додатку до диплому.

Тематика робіт повинна відповідати сучасному стану та перспективним напрямам розвитку спеціальності «Інформаційні управляючі системи та технології», бути пов'язаною з вирішенням практичних завдань підприємств та організацій, тематикою наукової роботи кафедри «Комп'ютерні науки» та з напрямками діяльності підприємств і установ-партнерів і відповідати вимогам кваліфікаційної характеристики фахівця за спеціальністю «Інформаційні управляючі системи та технології».

Тематика робіт повинна передбачати можливість самостійного розв'язання інженерних і наукових задач за спеціальністю «Інформаційні управляючі системи та технології» та виявлення елементів самостійності дослідницької роботи.

Тема є не лише назвою магістерської роботи, а й наміченим результатом дослідження, скерованим на вирішення конкретної проблеми.

Назва теми комплексної ДР складається з назви загальної частини і, через крапку, з назви конкретної частини, яку відповідно до індивідуального завдання розробляє кожний студент.

Назва теми має бути якомога коротшою, точною і відповідати змісту магістерської роботи. Не треба вживати в заголовку роботи невизначені формулювання, наприклад: «Аналіз деяких питань...», а також штапованих формулювань типу: «До питання про...», «До вивчення...», «Матеріали до...».

У назві також не дозволяється використовувати скорочення (аббревіатури), крім загальноприйнятих.

Виконав: студент (ка) _____ курсу, групи _____

Наводиться курс та шифр групи студента-дипломника в називному відмінку.

напряму підготовки (спеціальності)

Наводиться шифр і повна назва напрямку підготовки, спеціальності студента-дипломника в називному відмінку.

(прізвище та ініціали)

Тут вписується прізвище та ініціали студента-дипломника в називному відмінку.

Керівник

Тут вписується прізвище та ініціали керівника студента-дипломника в називному відмінку.

Рецензент

Тут вписується прізвище та ініціали рецензента дипломної роботи, якого призначила випускова кафедра.

Навпроти кожного прізвища на титульній сторінці ставиться відповідний підпис.

Зразок оформлення завдання на магістерську роботу

Форма № Н-9.01

Міністерство освіти і науки України

Тернопільський національний технічний університет імені Івана Пулюя

(повне найменування вищого навчального закладу)

Факультет комп'ютерно-інформаційних систем і програмної інженеріїКафедра комп'ютерних наукОсвітньо-кваліфікаційний рівень магістр

Напрямок підготовки _____

(шифр і назва)

Спеціальність 8.05010101 "Інформаційні управляючі системи та технології"

(шифр і назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри _____

Приймак М.В.« 25 » лютого 201 3 р.

ЗАВДАННЯ НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ) СТУДЕНТУ

Петренко Петра Петровича

(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) Розробка інформаційної системи для автоматизації роботи кур'єрської служби з використанням систем геоопозиціонування

Керівник проекту (роботи) Кужель Сергій Олександрович, д.ф.-м.н., професор

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом по університету від « 26 » лютого 201 3 року № 279-72. Термін подання студентом проекту (роботи) 27 серпня 2013 року3. Вихідні дані до проекту (роботи) вибірки реалізації фотоплетизмосигналів

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)
Вступ. Аналітичний огляд літературних та інших джерел...

...Охорона праці та безпека в надзвичайних ситуаціях. Висновки

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)

Завдання. Структурна схема інформаційної системи для реєстрації та аналізу фотоплетизмосигналів. Структура проведення імітаційного моделювання фотоплетизмосигналу.

Алгоритм....

6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
<i>Екологія</i>	<i>к.т.н., доцент Лясота О.М.</i>		
<i>Охорона праці та безпека в надзвич. ситуаціях</i>	<i>ст. в. Яскілка В.Я.</i>		
<i>к.т.н., доцент Калушка В.П.</i>			
...			

7. Дата видачі завдання

11.02.2013

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Примітка
1	<i>Аналіз поставленого завдання. Підбір бібліографічних матеріалів необхідних для виконання роботи</i>	11.02-25.02.2013	<i>Виконано</i>
	...		
n	<i>Розробка розділу “Обґрунтування економічної ефективності”</i>	3.06-2.07.2013	<i>Виконано</i>
n+m	<i>Попередній захист роботи</i>	19.08-25.08.2013	<i>Виконано</i>
	<i>Захист роботи</i>	28.08.2013	

Студент _____
(підпис)

Петренко П.П.

(прізвище та ініціали)

Керівник проекту (роботи) _____
(підпис)

Кужель С.О.

(прізвище та ініціали)

Рекомендації щодо розробки завдання:

Затверджую

В даній області підпис ставить завідувач або заступник завідувача кафедри комп'ютерних наук. Дата затвердження встановлюється та, яка йде наступною після дати закінчення проходження стажування з фаху (дослідницької практик). В зразку використано терміни проходження стажування з фаху 11.02.13-25.02.13. В залежності від форми навчання терміни проходження стажування з фаху (дослідницької практики) можуть відрізнятися.

Завдання на дипломний проект (роботку) студента

В цьому полі вписується прізвище ім'я по батькові студента-дипломника виконавця дипломної роботи ОКР "Магістр" проекту в родовому відмінку.

Тема проекту (роботи), керівник проекту (роботи) та наказ по університету, яким вона затверджена

Назва теми повинна бути, за можливості, короткою, чітко і конкретно відображати мету та основний зміст дослідження і бути однаковою у наказі ректора про закріплення тем і керівників за студентами, завданні на ДР, титульному аркуші пояснювальної записки, кресленнях, документах ДЕК та в додатку до диплому.

Тематика робіт повинна відповідати сучасному стану та перспективним напрямам розвитку спеціальності «Інформаційні управляючі системи та технології», бути пов'язаною з вирішенням практичних завдань підприємств та організацій, тематикою наукової роботи кафедри «Комп'ютерні науки» та з напрямками діяльності підприємств і установ-партнерів і відповідати вимогам кваліфікаційної характеристики фахівця за спеціальністю «Інформаційні управляючі системи та технології».

Тематика робіт повинна передбачати можливість самостійного розв'язання інженерних і наукових задач за спеціальністю «Інформаційні управляючі системи та технології» та виявлення елементів самостійності дослідницької роботи.

Тема є не лише назвою магістерської роботи, а й наміченим результатом дослідження, скерованим на вирішення конкретної проблеми.

Назва теми комплексної ДР складається з назви загальної частини і, через крапку, з назви конкретної частини, яку відповідно до індивідуального завдання розробляє кожний студент.

Назва теми має бути якомога коротшою, точною і відповідати змісту магістерської роботи. Не треба вживати в заголовку роботи невизначені формулювання, наприклад: «Аналіз деяких питань...», а також штапованих формулювань типу: «До питання про...», «До вивчення...», «Матеріали до...».

У назві також не дозволяється використовувати скорочення (аббревіатури), крім загальноприйнятих.

Прізвище, ім'я та по батькові керівника вписується повністю

Дату та номер наказу виписується із наказу ректора університету про закріплення тем і керівників за студентами.

Термін подання студентом проекту (роботи)

Термін здачі студентом закінченої роботи встановлюється на основі графіка основного захисту (за один робочий день до захисту).

Вихідні дані до проекту (роботи)

Тут маються на увазі вхідні дані до магістерської роботи.

Зазначаються лише кількісні або (та) якісні показники (характеристики) об'єкта дослідження, яким він повинен відповідати після розробки системи в даній дипломній роботі; умови, в яких повинен функціонувати об'єкт проектування (часові, просторові, кліматичні, енергетичні, навантажувальні, екологічні, ергономічні); припустимі відхилення від нормативних значень показників або похибки (максимальні, мінімальні, середньоквадратичні) тощо. Вихідні дані до дипломної роботи проекту повинні визначати кількісні або (та) якісні показники щодо умов, засобів та методів, які характеризують спрямованість наукового дослідження, конкретизують методику розв'язання теоретичних проблем та проведення експерименту, якщо останнє не є предметом самостійного вибору студента в процесі виконання дипломної

роботи. **Залишати цей розділ завдання незаповненим або зазначати в ньому літературні джерела (крім тих, де надається опис і характеристика конкретного об'єкту дослідження) неприпустимо!**

Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

Зазначаються конкретні пункти змісту пояснювальної записки, що і є завданням з окремих частин роботи, послідовність та зміст яких визначають фактично програму дій дипломника та майбутню структуру пояснювальної записки.

Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)

Визначає креслення, діаграми, гістограми, малюнки, плакати тощо, які є обов'язковими для виконання в даній роботі.

Консультанти розділів проекту (роботи)

Зазначаються вчене звання, прізвище, ініціали та посада консультанта з додаткових питань дипломної роботи. Консультанти ставлять дату та свій підпис у відповідних графах таблиці.

Дата видачі завдання

В цьому полі приводиться дата, яка є першим днем проходження стажування з фаху (дослідницької практики). В зразку використано терміни проходження стажування з фаху 11.02.13-25.02.13. В залежності від форми навчання терміни проходження стажування з фаху (дослідницької практики) можуть відрізнятися.

Календарний план

Протягом перших двох тижнів дипломного проектування студент складає календарний план-графік виконання дипломної роботи, який узгоджується та затверджується керівником роботи.

Зразок відгуку керівника на магістерську роботу

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Кафедра _____

ВІДГУК
на дипломну роботу за освітньо-кваліфікаційним рівнем «магістр»

Студента _____

На тему: _____

Спеціальність і група _____

Обсяг дипломної роботи: графічного (ілюстративного) матеріалу – _____ аркушів формату А1 (слайдів); кількість сторінок пояснювальної записки – _____ аркушів формату А4; кількість сторінок додатків – _____ аркушів формату А4.

Висновок про відповідність дипломної роботи завданню _____

Актуальність теми _____

Наукова новизна, теоретичне та практичне значення роботи _____

Зауваження, побажання _____

Позитивні сторони роботи _____

Оцінка графічної частини (ілюстративного матеріалу) та пояснювальної записки: _____

Загальна оцінка дипломної роботи: _____

Відгук склав: _____

(вчений ступінь, посада, прізвище та ініціали керівника)

« _____ » _____ 201__ р. _____

(підпис керівника)

Відгук - це характеристика професійних та громадських якостей дипломника та його роботи в процесі дипломного проектування.

Відгук складається у довільній формі із зазначенням:

- головної цілі магістерської роботи, в інтересах або на замовлення якої організації він виконана (в рамках науково-дослідної роботи кафедри, підприємства, НДІ тощо);
- відповідності виконаної ДР завданню;
- ступеня самостійності при виконанні ДР;
- рівня підготовленості дипломника до прийняття сучасних рішень;
- умінь аналізувати необхідні літературні джерела, приймати правильні (інженерні, наукові) рішення, застосовувати сучасні системні та інформаційні технології, проводити фізичне або математичне моделювання, обробляти та аналізувати результати експерименту;
- найбільш важливих теоретичних і практичних результатів, апробації їх (участь у конференціях, семінарах, оформлення патентів, публікація в наукових журналах тощо);
- загальної оцінки виконаної ДР, відповідності якості підготовки дипломника вимогам ОКХ і можливості присвоєння йому відповідної кваліфікації;
- інші питання, які характеризують професійні якості дипломника.

Зразок рецензії на магістерську роботу

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

РЕЦЕНЗІЯ

на дипломну роботу за освітньо-кваліфікаційним рівнем «магістр»

Студента _____

На тему: _____

Напрямок підготовки і група _____

Обсяг дипломної роботи: графічного (ілюстративного) матеріалу – _____ аркушів формату А1 (слайдів); кількість сторінок пояснювальної записки – _____ аркушів формату А4; кількість сторінок додатків – _____ аркушів формату А4.

Короткий зміст дипломної роботи та отриманих результатів _____

Висновок про відповідність дипломної роботи завданню _____

Актуальність теми _____

Наукова новизна, теоретичне та практичне значення роботи _____

Короткий критичний огляд змісту кожного розділу роботи з виділенням найбільш важливих і значущих моментів, ступінь використання дипломником прогресивних методів досліджень, повнота викладення матеріалу, оцінка отриманих результатів: _____

Позитивні сторони роботи: _____

Зауваження, побажання: _____

Можливість впровадження отриманих результатів _____

Оцінка графічної частини (ілюстративного матеріалу) та пояснювальної записки: _____

Загальна оцінка дипломної роботи: _____

Рецензію склав: _____

(вчений ступінь, посада, місце роботи, прізвище, ім'я та по-батькові)

« ____ » _____ 201__ р. _____
(підпис рецензента)

Рецензія - це характеристика якості дипломної роботи.

Рецензія складається у довільній формі із зазначенням:

- відповідності ДР затвердженій темі та завданню;
- актуальності теми;
- реальності ДР (її виконання на замовлення підприємств, організацій, за науковою тематикою кафедри, НДІ тощо);
- глибини техніко-економічного обґрунтування прийняття рішень;
- ступеня використання сучасних досягнень науки, техніки, виробництва, інформаційних та інженерних технологій;
- оригінальності прийнятих рішень та отриманих результатів;
- правильності проведених розрахунків і конструкторсько-технологічних рішень;
- наявності і повноти експериментального (фізичного або математичного) підтвердження прийнятих рішень;
- якості виконання пояснювальної записки, відповідності креслень вимогам ДСТУ, ЕСКД;
- можливості впровадження результатів ДР;
- недоліків ДР;
- оцінки ДР за 4-бальною системою і можливості присвоєння дипломнику відповідної кваліфікації (формулювання згідно з діючими нормативними документами).

Рецензія не повинна дублювати відгук керівника, тому що відгук керівника – це в основному характеристика професійних та громадянських якостей дипломника та його магістерської роботи в процесі дипломного проектування, а рецензія – це характеристика якості безпосередньо дипломної роботи.

АНОТАЦІЯ

Розробка системи організації та управління персональною контактною інформацією // Дипломна робота ОКР “Магістр” // Рак Андрій Михайлович // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп’ютерно-інформаційних систем і програмної інженерії, кафедра комп’ютерних наук, група СНм-51 // Тернопіль, 2013 // С. ,рис. – , табл. – , кресл. – , додат. – , бібліогр. – .

Ключові слова: БАЗИ ДАНИХ, ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ, ІНТЕРНЕТ, ІНТЕРФЕЙС, ІНФОРМАЦІЙНО-ДОВІДКОВІ СИСТЕМИ, КАЛЕНДАР, КОНТАКТНІ ДАНІ, ОБЛАДНАННЯ, РОЗКЛАД, СЕРВЕР.

Дипломна робота присвячена розробці системи організації та управління персональною контактною інформацією.

В першому розділі дипломної роботи розглянуто інформаційні системи і технології в сучасному суспільстві, описано основні етапи розвитку інформаційних систем, наведено класифікацію інформаційних систем, розглянуто автоматизовані інформаційні системи для підприємств та організацій, описано принципи функціонування корпоративних систем управління взаємозв’язками з клієнтами, розглянуто соціальні медіа як сучасний стан еволюції засобів комунікації, висвітлено поняття соціального графу.

В другому розділі дипломної роботи сформовано вимоги до інформаційної системи, проведено вибір та обґрунтування технології розробки.

В третьому розділі дипломної роботи створено модель даних, розроблено відкриту архітектуру програмного засобу та інтерфейс модулів, спроектовано та розроблено інтерфейс користувача.

Об'єкт дослідження: системи управління інформацією про взаємовідносини з оточуючими.

Предмет дослідження: управління персональною контактною інформацією.

Мета роботи: проектування та розробка програмного продукту, аналогу CRM, для структурування, організації та управління персональною контактною інформацією.

Основні результати: досліджено концептуальні засади побудови системи управління контактною інформацією, сформовано вимоги до системи, обгрунтовано технологію розробки системи, розроблено модель даних, відкриту архітектуру програмного засобу, інтерфейс користувача.

ANNOTATION

Development of organization and management of personal contact information // Diploma work degree "Master" // Rak Andrew M. // Ternopil Ivan Pul'uj National Technical University, Department of Computer Information Systems and Software Engineering, Department of Computer Science // Ternopil, 2013 // P. , Fig. - , Table. - .

Every day we meet and communicate with many people, make new acquaintances, share your thoughts and experiences. Over time, some of the contacts that are important for us in the past are forgotten and lost, but they are still useful, as may be necessary to restore contact, so for them to remember and quickly find where necessary.

In connection with the development of Web 2.0 in almost every one of us has an account in a particular social network, and in most cases in one. We all want them to be associated with one person - directly to us. This knowledge will help to better organize relationships, manage time and achieve their goals. Therefore, always handy to have at hand a tool that would allow to structure and store all important details about such contacts, the history of relations between them, fast communication tools, to provide integration with social networks.

To solve this problem, called CRM – system (Contact Relationship Management). It is a broad term that covers concepts that are used to manage relationships with business partners, customers, friends, acquaintances, and your other contacts, including the collection, storage and analysis of information about them.

As part of the thesis just been tasked to develop a system for efficient storage of contacts and navigate between them, finding the channels of communication with them if necessary, and the collection and structuring of information related to them.

Background includes the following tasks:

- the organization of storing contact data about customers, partners and suppliers from different sources;
- chat history collection, the agreements reached, the knowledge about contact its characteristics and prospects;
- ordered accounting, control and business relationship analysis.

In the first chapter of the thesis considers information systems and technology in modern society, describes the main stages of the development of information systems, information systems are classified, examined computerized information systems for businesses and organizations described principles of corporate governance of relationships with clients are considered social media as the current state of the evolution of communication highlights the concept of social graph.

In the second chapter of the thesis formed the requirements for an information system, made the choice and justification of technology development.

In the third chapter of the thesis a model of data, developed an open architecture software and interface modules, designed and developed user interface.

Object of study: management information systems of relationships with others.

Purpose of the study: Control personal contact information.

Objective: To design and develop the software counterpart CRM, to structure, organize and manage personal contact information.

Implementation CRM - system will be a useful tool for building the interface between management and relationships in the social graph, which provides quick access to existing and will attract new contacts through the establishment of clear and transparent processes of interaction with customers and partners, the formation mechanism loyalty contacts, control consistency work with employees, customers, partners and others.

To the solution was fully functional, it must contain a number of components:

- manage your contacts and related information;
- manage time and tasks (taymmenedzhmentu);
- integration with other systems by means of plug-ins;

– opportunities for new developments.

Main results: Investigates the principles of construction management contact information formed the system requirements, system development The technology, developed a data model, open architecture software, user interface.

The novelty of the work: the means of communications between users within the system, as well as associations and structuring of the relationship according to certain criteria and categories are solved question of finding necessary for us humans and all related information, which may be paid not only to the user, but also through the integration with other systems.

Keywords: DATABASE, INFORMATION TECHNOLOGY, INTERNET, INTERFACE, INFORMATION AND REFERENCE SYSTEMS, CALENDAR, CONTACTS DATA, EQUIPMENT SCHEDULE, SERVER.

Рекомендована література

1. Введення в дію нового стандарту з бібліографічного опису ДСТУ ГОСТ 7.1:2006. Основні відмінності від ГОСТ 7.1.-84 [Електронний ресурс] : нові правила бібліогр. опису / Кн. палата України. – Режим доступу: http://www.ukrbook.net/DSTU_pabl.htm

2. ГОСТ 19.101-77 ЕСКД. Виды программ и программных документов.

3. ГОСТ 19.105-78 ЕСКД. Общие требования к программным документам.

4. ГОСТ 19.401-78 ЕСКД. Текст программы.

5. ГОСТ 19.402-78 ЕСКД. Описание программы.

6. ГОСТ 19.701-90 ЕСКД. Схема алгоритмов и программ. Обозначения условно-графические.

7. ГОСТ 19.701-90. Единая система программной документации. Схемы алгоритмов, программ, данных и систем. Условные обозначения и правила выполнения. – М.: Изд-во стандартов, 1991. – 20 с.

8. ГОСТ 2.104-2006. Единая система конструкторской документации. Основные надписи

9. ГОСТ 2.105-79 ЕСКД. Общие требования к текстовым документам.

10. ГОСТ 2.106-68 ЕСКД. Текстовые документы.

11. ГОСТ 2.109-73 ЕСКД. Основные требования к чертежам.

12. ГОСТ 2.119-73 ЕСКД. Эскизный проект.

13. ГОСТ 2.120-73 ЕСКД. Технический проект.

14. ГОСТ 2.417-78 ЕСКД. Правила выполнения чертежей печатных плат

15. ГОСТ 2.702-75 ЕСКД. Правила выполнения электрических схем.

16. ГОСТ 2.710-81 ЕСКД. Обозначения буквенно-цифровые в электрических схемах.

17. ГОСТ 2.711-82 ЕСКД. Схемы деления изделия на основные части.
18. ГОСТ 34.003-90. Информационная технология. Комплекс стандартов и руководящих документов на автоматизированные системы. Автоматизированные системы. Термины и отделения. – М.: Изд-во стандартов, 1991. – 32 с.
19. ГОСТ 34.201-90. Информационная технология. Комплекс стандартов и руководящих документов на автоматизированные системы. Автоматизированные системы. Стадии создания. – М.: Изд-во стандартов, 1991. – 16 с.
20. ГОСТ 34.602-89. Информационная технология. Комплекс стандартов и руководящих документов на автоматизированные системы. Техническое задание на создание автоматизированной системы. – М.: Изд-во стандартов, 1990. – 28 с.
21. ГОСТ 19.106-78 ЕСКД. Требования к программным документам, выполненным печатным способом.
22. ГОСТ 19.202-78 ЕСКД. Спецификация. Требования к содержанию и оформлению.
23. ГОСТ 19.404-79 ЕСКД. Пояснительная записка. Требования к содержанию и оформлению.
24. ГОСТ 2.701-84 ЕСКД. Схемы. Виды и типы. Общие требования к выполнению.
25. ГОСТ 2.708-81 ЕСКД. Правила выполнения электрических схем цифровой вычислительной техники.
26. ДСТУ 3008-95. Документація. Звіти у сфері науки і техніки. Структура і правила оформлення. – К.: Держстандарт України, 1995. – 36 с.
27. ДСТУ ГОСТ 7.1-2003. Библиографическая запись. Библиографическое описание. Общие правила составления. – М.: ИПК Изд-во стандартов, 2004. – 48 с.
28. ДСТУ ГОСТ 7.1-2006. Бібліографічний запис. Бібліографічний опис. Загальні вимоги та правила складання: чинний з 2007-07-01. – К.:

Держспоживстандарт України, 2007. – 47 с. (Система стандартів з інформації, бібліотечної та видавничої справи) (Національний стандарт України).

29. Електронний фонд Національної бібліотеки імені В. І. Вернадського [Електронний ресурс]. – Режим доступу: <http://www.nbuv.gov.ua/>

30. Кушнарєнко Н. М. Наукова обробка документів : підручник / Н.М. Кушнарєнко, В.К. Удалова. – 2-ге вид., випр. і допов. – К.: Знання, 2004. – 331 с. – (Вища освіта ХХІ століття). – ISBN 966-8148-46-0.

31. Положення про організацію навчального процесу у вищих навчальних закладах. Наказ Міністерства освіти України № 161 від 2 червня 1993р. // Освіта України. Збірник нормативних актів. – Харків: Світ – Пресс, 1999 р. – С. 168–188.

32. Приклади оформлення бібліографічного опису у списку джерел, який наводять у дисертації, і списку опублікованих робіт, який наводять в авторефераті // Бюлетень ВАК України. – № 3. – 2008. – С. 9–13.

33. Український орфографічний словник: близько 170000 слів / за ред. В. М. Русанівського; [уклали: В. В. Чумак, І. В. Шевченко, Л. Л. Шевченко, Г. М. Ярун]; НАН України; Укр. мовно-інформ. фонд; Ін-т мовознав. ім. О. О. Потебні. – Вид. 6-те, переробл. і допов. – К.: Довіра, 2006. – 960 с. – ISBN 966-507-206-4. – (Словники України).

ГОСТ 7.1:2006 (на оформлення літературних джерел)

Згідно з Указом Держкомітету України з питань технічного регулювання і споживчої політики від 10.11.2006 р. № 322 з 1 липня 2007 року в Україні діє міждержавний стандарт ДСТУ 7.1:2006 “Система стандартів з інформації, бібліотечної та видавничої справи. Бібліографічний запис, бібліографічний опис. Загальні вимоги та правила складання”.

Нововведений стандарт відповідає ГОСТові 7.1-2003 і є базовим для складання бібліографічного опису всіх видів документів.

Цей стандарт вводиться замість п’яти попередніх стандартів опису нотних, картографічних та образотворчих видань, аудіовізуальних матеріалів, друкованих та електронних видів документів (ГОСТ 7.1-84, ГОСТ 7.16-79, ГОСТ 7.18-79, ГОСТ 7.34-81, ГОСТ 7.40-82).

Пропоновані рекомендації, розроблені на основі ДСТУ ГОСТ 7.1:2006, мають фрагментарний характер: розглядають лише суттєві відмінності між діючим і вищезгаданими стандартами.

Основні відмінності у новому стандарті стосуються зони назви і відомостей про відповідальність, зокрема, загального позначення матеріалу, зони видання, зони специфічних відомостей та деяких елементів інших зон опису.

За новими правилами для розрізнення граматичної і приписаної пунктуації (тобто розділових знаків між зонами бібліографічного опису та їх елементами) застосовують проміжок в один друкований знак до і після приписаного знака. Виняток становлять: крапка і кома – проміжки ставлять тільки після них, а також квадратні і круглі дужки, які виділяються проміжками лише ззовні, напр.:

У новому стандарті змінено правила вживання великої та малої літер. Їхнє вживання визначається не лише граматичними нормами, а й розділенням зон бібліографічного опису. Перші слова відомостей, що відносяться до зони назви та відомостей про відповідальність, записуються з малої літери, якщо вони не є власними назвами, першими словами назви чи цитатами. Винятком є загальне позначення матеріалу та будь-які назви в усіх зонах опису.

З великої літери розпочинається кожна зона опису, яка виділяється крапкою й тире. Перед елементами всередині зони зазначається відповідний їм приписаний знак. У разі повторення окремих елементів він також повторюється, за винятком знака “навскісна лінія”, який застосовується в аналітичному описі. В кінці бібліографічного опису ставиться крапка.

У ДСТУ ГОСТ 7.1:2006 розширився набір обов’язкових елементів бібліографічного опису. Статусу обов’язкових набули:

- перші відомості про відповідальність в усіх зонах (зоні назви та відомостей про відповідальність, зоні видання, серії);
- додаткові відомості про видання;
- ім’я видавця, розповсюджувача тощо;
- основна назва серії та підсерії;
- міжнародний стандартний номер серійного видання, що був наданий серії чи підсерії (ISSN);
- номер випуску серії чи підсерії;
- окремі примітки в описі певних видів документів (в описі електронних ресурсів – примітки про джерело основної назви).

Джерелом інформації для складання бібліографічного опису є документ у цілому. Головним джерелом інформації є елемент документа, який уміщує основні вихідні відомості – титульний аркуш, титульний екран, етикетка, наклейка тощо.

Необхідність застосування та набір факультативних елементів визначається установою, яка здійснює бібліографічні записи.

До зони назви і відомостей про відповідальність уперше введено новий факультативний елемент – *загальне позначення матеріалу, який доцільно подавати в описі документів різних видів*.

Якщо в конкретному інформаційному масиві переважають документи одного виду, загальне позначення матеріалу може бути випущеним.

Приклади бібліографічного запису

За назвою	1. З фольклорної криниці Франкового села. Народні пісні, перекази, спогади / записи Олекси Ошуркевича ; розшифр. і нотація мелодій Мирослава Стефанишина, Юрія Рибачака. – Луцьк : Волин. книга, 2007. – 167 с. : ноти. – ISBN 978-966-361-187-7. – ISBN 978-966-361-128-8.
	2. Володимир Винниченко – художник = Volodymyr Vynnychenko the Artist : альбом / [упорядкув. та комент. : Гальченко С., Маслянчук Т.]. - К. : Мистецтво, 2007. – 224 с. : іл. - Переднє слово і текст парал. англ., резюме і перелік іл. рос. та фр. – ISBN 978-966-577-073-2.
	3. “Дванадцятка”: Наймолодша львівська літературна богема 30-х років XX століття : антологія урбаністичної прози / авт. проекту Василь Габор. – Львів : Піраміда, 2006. – 344 с. : іл. – (Українська Літературна Спадщина). – До 750-ліття Львова. – Видавничий проект “Приватна колекція”. – ISBN 966-8522-70-2.
Збірник без загальної назви	4. Зразки процесуальних документів : (заяви, позовні заяви, скарги, клопотання) / уклад. : М. М. Лядецький, М. І. Хавронюк. Стратегія і тактика цивільного процесу : практ. посіб. / В. М. Кравчук. – К. : Атіка, 2007. – 352 с. – ISBN 966-326-076-9. 5. Новітня українська суспільна географія : хрестоматія для студ. геогр., екон. і фак. міжн. відн. / упоряд. Олег Шаблій. – Львів : ВЦ ЛНУ ім. І. Франка, 2007. – 1008 с. : іл. – Парал. тит. арк. англ. – ISBN 978-966-613-542-4. 6. Античная мифология : энциклопедия / [сост., ред. и предисл. К. Королева]. – М. ; СПб : Эксмо : Мидгард, 2005. – 768 с. : ил. – ISBN 5-699-07260-8.

Багаторівневий опис	7. Апокрифи і легенди з українських рукописів : [у 5 т.] / зібрав, упоряд. і пояснив Іван Франко ; [передм. Я. Мельник]. – Репр. вид. – Львів : [ЛНУ ім. І. Франка], 2006. – До 150-ліття від дня народження Івана Франка. – На обкл. : Апокрифи і легенди з українських рукописів. – ISBN 966-613-411-X. Т. 1 : Апокрифи старозавітні. – Репр. вид. 1896 р. – 2006. – 512 с. – ISBN 966-613-421-7. Т. 2 : Апокрифи новозавітні. А. Апокрифічні євангелія. – Репр. вид. 1899 р. – 2006. – 532 с. – ISBN 966-613-439-X.
Однорівневий опис	8. Збірник текстів з курсу “Педагогіка”. У 3 ч. Ч. 1. Дидактика : навч.-метод. посіб. / за заг. ред. Л. Ковальчук. – Львів : ВЦ ЛНУ ім. І. Франка, 2007. – 120 с. – ISBN 978-966-613-552-3. 9. Nierównosci społeczne a wzrost gospodarczy. Zesz. nr 10. Gospodarka oparta na wiedzy / [red. Michal Gabriel] ; Uniwersytet Rzeszowski, Katedra Teorii Ekonomii. – Rzeszów : [b. w.], 2007. – 626 s. – ISBN 978-83-7338-309-8.
1 автор	1. Барабаш Юрій Якович. Вибрані студії. Сковорода. Гоголь. Шевченко / Юрій Барабаш ; передм. Панченка Володимира. – К. : ВД “Києво-Могилян. акад.”, 2007. – 744 с. : покажч. імен с. 723-741. – (Б-ка Шевченків. комітету / [Загребельний П. А. (голова вид. ради), Дзюба І. М., Лубківський Р. М.]. – ISBN 966-01-0320-4 (серія). – ISBN 966-518-382-6. 2. Блум Гарольд. Західний канон: книги на тлі епох : пер. з англ. / Гарольд Блум ; [заг. ред. Ростислава Семківа]. – К. : Факт, 2007. – 720 с. – (“Висока полиця”). – ISBN 978-966-359-205-3. – ISBN 966-359-091-0 (серія).

	3. Вовк Володимир Михайлович. Математичні методи дослідження операцій в економіко-виробничих системах : монографія / В. М. Вовк. – Львів : ВЦ ЛНУ ім. І. Франка, 2007. – 584 с. – ISBN 979-966-613-532-5. 4. Еко Умберто. Роль читача : дослідження з семіотики текстів / Умберто Еко ; пер. з англ. Мар'яна Гірняк ; [наук. ред. Марія Зубрицька ; ред. Ірина Фаріон]. – Львів : Літопис, 2004. – 384 с. – ISBN 966-7007-107-3. 5. Кондратович Олександра Павлівна. Українські звичаї: Народини. Коса ж моя... / Олександра Кондратович. – Луцьк : Волин. обл. друк., 2007. – 240 с. : нот. – ISBN 978-966-361-220-1. 6. Шевченко Тарас. Усі твори в одному томі / Тарас Шевченко ; [вступ. ст. І. Франка, квітень 1914 р.]. – К. : Ірпінь, 2007. – 824 с. : іл. – (Літературні класики України) (Поетична
Аналітичний опис	7. Войтович Л. Доля і недоля міста Роздолу / Л. Войтович // Миколаївщина : зб. наук. ст. / Ін-т українознав. ім. І. Крип'якевича НАН України ; [редкол. : Л. Войтович (відп. ред.), О. Головка, М. Литвин та ін.]. – Львів, 2006. – Т. 3. – С. 177–223. – ISBN 966-02-1224-0. 8. Грицак Ярослав. Малий епізод до великої історії, або Ще раз про коло ідей “Руської Трійці” / Я. Грицак // Україна: культурна спадщина, національна свідомість, державність / НАН України, Ін-т українознав. ім. І. Крип'якевича ; [редкол. : Я. Ісаєвич (голова), О. Аркуша (відп. ред.), В. Горинь та ін.]. – Львів : Ін-т українознав. ім. І. Крип'якевича, 2001. – Вип. 9 : Ювілейний збірник на пошану Феодосія Стебля. – С. 139–143. – ISBN 966-02-2209-2. 9. Губерначук С. У гербах української шляхти / С. Губерначук // Українська культура. – 2008. – № 8. – С. 32–33. 10. Ісаєвич Я. Українське книгознавство: етапи розвитку / Ярослав Ісаєвич // Вісник Львівського університету. Серія : книгознавство, бібліотекознавство, інформаційні технології. – Львів : ЛНУ ім. І. Франка, 2006. – Вип. 1. – С. 7–19. – ISSN 0201-758X. – ISSN 0460-0509. 11. Мудрий М. Національно-політичні орієнтації в українському суспільстві Галичини австрійського періоду у

	висвітленні сучасної історіографії / М. Мудрий // Вісник Львівського університету. Серія історична. – Львів : ЛНУ ім. І. Франка, 2002. – Вип. 37, ч. 1 і Статті та повідомлення. – С. 465–500. 12. Hrytsak Y. Історія одного імені / Y. Hrytsak // States, Societies, Cultures : East and West : Essays in Honor of Jaroslaw Pelenski = Держави, суспільства, культури : Схід і Захід : зб. на пошану Ярослава Пеленського / National Academy of Sciences of Ukraine, European Research Institute, W. K. Lypynsky East European Research Institute ; ed. by : J. Duzinkiewicz (Editor-in-Chief), M. Popovych, V. Verstiuk, N. Jakovenko. – New York : Ross, 2004. – P. 351–368. – ISBN 0-88354-181-5.
2 автори	1. Крижановська Оксана Олегівна. Історія середніх віків. Вступ до історії західноєвропейського Середньовіччя : курс лекцій : навч. посіб. для студ. ВНЗ / О. О. Крижановська, О. П. Крижановський. – Вид. 2-ге, стер. – К. : Либідь, 2006. – 368 с. – ISBN 966-06-0412-2. 2. Попова Ирина Николаевна. Грамматика французского языка. Практический курс Le Français : учеб. для студ. вузов / И. Н. Попова, Ж. А. Казакова. – Изд. 12-е стер. – М. : Нестор Академик Пабlishерз, 2003. – 480 с. – ISBN – . 3. Савчин Володимир Павлович. Електронне перенесення в напівпровідникових структурах : навч. посіб. для студ. ВНЗ / В. П. Савчин, Р. Я. Шувар. – Львів : ВЦ ЛНУ ім. І. Франка, 2008. – 688 с. – ISBN 978-966-613-569-1.
Аналітичний опис	4. Каленюк І. Економічні часописи, технологічні уклади і прогнози майбутнього / Ирина Каленюк, Костянтин Корсак // Наук. світ. – 2008. – № 9. – С. 3–5. 5. Кондратюк К. Українське національне відродження ХІХ – початку ХХ століть у сучасній вітчизняній історіографії / Костянтин Кондратюк, Віктор Мандзяк // Українська історіографія на зламі ХХ і ХХІ століть: здобутки і

	проблеми : колект. моногр. / за ред. Л. Зашкільняка ; М-во освіти і науки України ; ЛНУ ім. І. Франка. – Львів : [ЛНУ ім. І. Франка], 2004. – 406 с. – ISBN 966-613-185-4.
3 автори	1. Воробель Яніна Максимівна. Англійська мова : навч. посіб. / Яніна Максимівна Воробель, Ольга Анатоліївна Шумська, Михайло Зенонович Гамкало. – Львів : ВЦ ЛНУ ім. І. Франка, 2006. – 106 с. : іл. – ISBN –. 2. Гайский Виталий Александрович. Применение функций Уолша в системах автоматизации научных исследований / В. А. Гайский, Н. Д. Егупов, Ю. П. Корнюшин. – К. : Наук. думка, 1993. – 212 с. – ISBN 5-12-003076-9. 3. Глинський Ярослав Миколайович. Паскаль. Turbo Paskal і Delphi : навч. посіб. / Глинський Я. М., Анохін В. Є., Ряжська В. А. – 4-те вид. – Львів : [Деол, СПД Глинський], 2003. – 144 с. – ISBN 966-7449-17-3.
<i>Аналітичний опис</i>	4. Шевченко О. Відлітають сірим шнуром... : [поет А. Могильний : некролог] / Олесь Шевченко, Василій Соловей, Станіслав Вишенський // Літ. Україна. – 2008. – 11 верес.
4 автори	1. Історія світової та української культури : підруч. для студ. ВНЗ / В. Греченко, І. Чорний, В. Кушнерук, В. Режко. – К. : Літера, [2005]. – 464 с. – ISBN 966-95287-8-X. 2. Український орфографічний словник : близько 170 000 слів / за ред. В. М. Русанівського ; [уклали : В. В. Чумак, І. В. Шевченко, Л. Л. Шевченко, Г. М. Ярун] ; НАН України ; Укр. мовно-інформ. фонд ; Ін-т мовознав. ім. О. О. Потебні. – Вид. 6-те, переробл. і допов. – К. : Довіра, 2006. – 960 с. – (Словники України). – ISBN 966-507-206-4.
<i>Аналітичний опис</i>	3. Використання техногенної сировини для виготовлення кераміки / І. С. Субота, Т. І. Булка, О. А. Шмельова, Р. А. Шугайло // Буд-во України. – 2008. – № 2. – С. 22–23. 4. Реальны ли перспективы энергетического развития Украины? / А. И. Амошина, В. В. Федоренко, Н. Г. Белопольский, Д. К. Турченко // Економіка та держава. – 2007. – № 10. – С. 4.
5 авторів і більше	1. Кириличні рукописні книги у фондах Львівської наукової бібліотеки ім. В. Стефаника НАН України : каталог. Т. 1. XI-XVI ст. / [уклад. : М. М. Кольбух (голов. ред.), Т. М.

	Гуцаленко, О. О. Дзьобан та ін.]. – Львів : Оріяна-Нова, 2007. – 522 с. : іл. – ISBN 978-966-02-2334-X. – ISBN 978-966-2128-05-5. 2. Новітня історія країн Західної Європи та Північної Америки, 1918-1945 рр. : навч. посіб. для студ. ВНЗ / Баран З. А., Кипаренко Г. М., Мовчан С. П. [та ін.] ; за ред. М. Швагуляка. – Львів : Афіша, 2005. – 288 с. – ISBN 966-325-052-6. 3. Правова статистика : підручник / [авт. кол. : Моїсєєв Є. М., Джу́жа О. М., Васи́левич В. В. та ін.]. – К. : Атіка, 2008. – 392 с. – ISBN 978-966-326-262-8. 4. Вища математика : навч. посіб. для студ. ВНЗ. У 2 ч. / Лиман Федір Миколайович, Власенко Віталій Федорович, Петренко Світлана Віталіївна та ін.]. – Суми : Університет. книга, 2006. – 624 с. – ISBN 966-680-230-9.
Аналітичний опис	5. Нейтронні дослідження взаємодії молекул поверхнево-активних речовин в неполярному розчиннику / В. І. Петренко, Л. А. Булавін, М. В. Авдєєв [та ін.] // Укр. фіз. журнал. – 2008. – № 3. – С. 229–234. – Резюме рос., англ. – Бібліогр. : с. 233.
Газета	1. Урядовий кур'єр : газ. центр. органів виконав. влади України / засн. Кабінет Міністрів України ; голов. ред. Алла Ковтун. – 1990– . – К. : Преса України, 2008– . – Виходить у віт., сер., четв., п'ятн. та суботу. 2008, 25 листоп., № 221 (3881).
Журнал	1. Вища школа : наук.-практ. журн. / засн. М-во освіти і науки України ; голов. ред. І. О. Вакарчук. – 2001– . – К. : Знання, 2008– . – Щомісяч. – ISSN 1682-2366. 2008, № 8-10.
Дисертація	1. Панчишин Тарас Володимирович. Інвестиційна діяльність в умовах ринкової трансформації економіки України : дис. ... канд. екон. наук : 08.01.01 / Панчишин Тарас Володимирович. – Львів, 2005. – 214 с. – Бібліогр. : с. 192–205. 2. Мацевко Ірина Іванівна. Неославізм у суспільно-політичному житті Галичини (1908–1914 рр.) : дис. ... канд. іст. наук : 07.00.02 / Мацевко Ірина Іванівна. – Львів, 2001. – 200 с. – Бібліогр. : с. 171–200.

Автореф.	1. Мацюк Г. П. Кодифікація української мови в галицьких граматиках першої половини XIX ст. : автореф. дис. на здобуття наук. ступеня д-ра філол. наук : спец. 10.02.01 “Українська мова” / Мацюк Галина Петрівна ; Ін-т мовознав. ім. О. О. Потебні НАН України. – К., 2002. – 32 с. 2. Авраменко О. В. Стан сильного душевного хвилювання: кримінально-правові та психологічні аспекти : автореф. дис. на здобуття наук. ступеня канд. юрид. наук : спец. 12.00.08 “Кримін. право та кримінологія; кримін.-викон. право” / Авраменко Олексій Володимирович ; Львів. нац. ун-т ім. Івана Франка. – Львів, 2008. – 19 с.
Рецензія (аналітичний опис)	1. Ющенко О. Перша авторська енциклопедія / Олекса Ющенко // Літ. Україна. – 2008. – 11 верес. – Рец. на кн. : Україна : енциклопед. словник / О. Сліпушко. – К. : Аконіт, 2008. – 324 с.
Інтерв'ю (аналітичний опис)	1. Павлюк І. Ігор Павлюк: “Письменник – це літописець ласки і болю” : [інтерв'ю з поетом, письм. Ігорем Павлюком] / спілкувався Сергій Соловійов // Березіль. – 2008. – № 7–8. – С. 102–106.
Нотні видання	1. Кирило Стеценко. Кантати і хори [Ноти] : [зб. хороших творів] / [упоряд. М. С. Юрченко ; вступ. ст. Л. О. Пархоменко ; худож. оформ. Ю. Г. Кудь]. – К. : Муз. Україна, 2008. – 324 с. : ноти : с. 11–320. – До 125-ти річчя від дня народж. класика укр. музики. – У зб. вперше друкується богослужбовий цикл “Панахида”. – ISMN M 707502-85-3. 2. Барвінський Василь. Прелюдії [Ноти] : для фортепяна = Wassyl Barwinskyj. Praeludien : für klavier / Василь Барвінський. – К. ; Ляйпціг : Укр. накладня, 1918. – 27 с. : ноти. – (Видавництво “Україна”). – Укр. муз. фест., Торонто = Ukrainian Music Festival, Toronto. 3. Педагогический репертуар [Ноты] : для фортепьяно : учеб. для детск. муз. школ : 5-й кл. – М. : Гос. муз. изд-во, 1958. – 23 с. : ноты. – Содерж. : Прощальный вальс ; Мазурка ; Хор волшебных дев / М. Глинка. Рондо / Д. Бортнянский. Две прелюдии / Л. Гурилев и др.
Картографічні видання	1. Історичний атлас Галичини [Карти] : карти, схеми, автентичні гравюри і печатки з праісторичних часів до XX

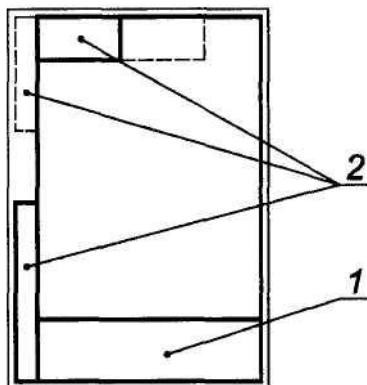
	ст. / уклад. Володимир Грабовецький. – Івано-Франківськ : [б. в.], 1996. – 49 с. : іл., карти, схеми. – ISBN –. 2. Львів [Карти] : панорама історичного центру : довідник-путівник / [Р. Атоян ; відп. за вип. І. Чумак]. – 3-тє вид., випр. та допов. – 1 : 80. – Львів : [Укрпол / М. Чумак], 2008. – 1 к. : кольор. мальов. ; 48х69/12 см. – Тексти : Хронологія Львова : етапи. Площа Ринок.
Електронні ресурси	1. Наш Львів [Електронний ресурс] = Lwow = Lemberg = Leopoldis : [фільм з субтитр.]. – 4 ГВ. – К. : Студія “Талан” : СТ “Укрмюзік”, 2007. – 1 електрон. опт. диск (DVD-ROM) ; 12 см. – Систем. вимоги: Pentium 324 MHz ; 256 Mb RAM ; 128 Mb Video ; від 4-х до 56-х DVD-ROM ; Windows 98/ME/XP/2000. – Назва з контейнера. 2. Атлас мира [Электронный ресурс] : мощные тематич. карты регионов, истор. справки и путеводители, экономич. обзоры, масштабирование любой точки планеты. – 80 Min / 700 MB. – [К.] : Компроект / ТОВ “Фортресс Паблішинг”, 2004. – 1 електрон. опт. диск (CD-ROM); 12 см. – (Master Soft ; Делаем знания доступными). – Систем. вимоги: Pentium 100 MHz ; 32 Mb RAM ; 8 Mb Video ; від 2-х до 32-х CD-ROM ; Windows 95/98/ME/ XP/2000. – Назва з контейнера. 3. Библиотека і доступність інформації у сучасному світі [Електронний ресурс] : електронні ресурси в науці, культурі та освіті : підсумки 10-ї Міжнар. конф. “Крим-2003” / Л. Й. Костенко, А. О. Чекмарьов, А. Г. Бровкін, І. А. Павлуша // Библиотечный вестник. – 2003. – № 4. – С. 43. – Режим доступа до журн. : http://www.nbuv.gov.ua/articles/2003/03klinko.htm 4. Компьютерные сети. Принципы, технологии, протоколы [Электронный ресурс] / В. Г. Олифер, Н. А. Олифер. – СПб. : Питер, 2002. – 672 с. – Режим доступа : URL : http://www.zipsites.ru/. – Загл. с экрана. 5. Электронный каталог ГПНТБ России [Электронный ресурс] ; база данных содержит сведения о всех видах лит., поступающей в фонд ГПНТБ России. – Электрон. дан. (5 файлов, 178 тыс. записей). – М., [199-]. – Режим доступа : URL : http://www.gpntb.ru/win/search/help/el-cat.html. – Загл. с экрана.

	6. Публікації. Нові правила бібліографічного опису [Електронний ресурс] / О. Устіннікова, П. Сенько, С. Юлдашева [та ін.] // ДСТУ ГОСТ 7.1:2006 в Україні ; Держ. наук. установа “Кн. палата України ім. Івана Федорова”. – Режим доступу : URL : http://www.ukrbook.net/DSTU.htm. – Загол. з екрану. 7. Andren C. 11 MBps Modulation Techniques / C. Andren // Информационный бюллетень Harris Semiconductor. — Режим доступа : http://www.intersil.com. — Дата доступа : 20.11.08. — Название с экрана. 8. Open Grid Forum (OGF): – Режим доступа: http://www.ogf.org. — Last access: 2008. – Title from the screen. 9. УкрМАРК: Національний формат представлення бібліографічних даних [Електронний ресурс]: (Проект) / НБУ ім. Вернадського, НПБ України, Наук. б-ка ім. М. Максимовича Київ. нац. ун-ту ім. Т.Г. Шевченка. – Електрон. дан. (13 файлів). – 2002-2003. – Режим доступу: www.nbuv.gov.ua/library/ukrmarc.html. – Назва з домашньої сторінки Інтернету. 10. Swanson E. Editing ISBD (SR): approach, scope, definitions [Electronic resource] // 68th IFLA Council and General Conference, August 18-24, 2002: Proceedings. – Mode of access: WWW.URL: www.ifla.org/IV/ifla68/papers/148-162e.pdf. - Last access: 2002. – Title from the screen. 11. Прокопенко Л. С. Бібліографічна секція Міжнародної федерації бібліотечних асоціацій та закладів як осередок дослідження національної бібліографії (1965–2002 рр.) [Електронний ресурс]: Автореф. дис. ... канд. іст. наук: 07.00.08/ Київ Нац. ун-т культури і мистецтв. – Електрон. дан. (1 файл). – К., 2004. – 18 с. – Режим доступу: www.nbuv.gov.ua/ard/2004/04plsdnb.zip. – Назва з екрана Примітка: „— Загол. з екрана” пишеться, якщо назва читається у верхньому рядку браузера після URL; якщо не читається, тоді можна замість цих слів нічого не писати.
Опис з помилкою	1. Ценгелевич [має бути Ценглевич] Каспер // Довідник з історії України (А-Я) / Ін-т іст. дослідж. Львів. нац. ун-ту ім. І. Франка ; вид-во “Генеза” ; [за заг. ред. І. Підкови і Р.

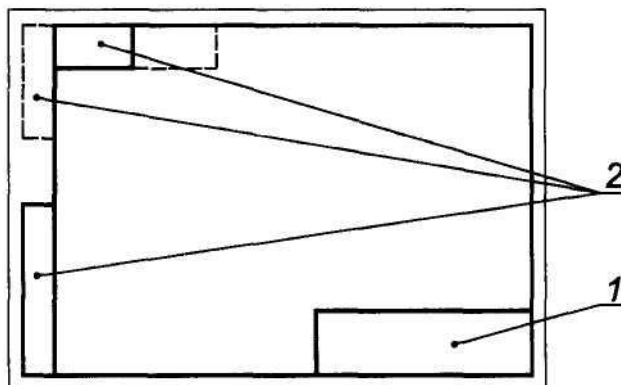
	Шуста]. – Вид. 2-ге, доопрац. і допов. – К. : Генеза, 2001. – С. 1033. – ISBN 966-504-179-7. 2. Чарнецький [має бути Чарноцький] Адам Зоріан // Довідник з історії України (А-Я) / Ін-т іст. дослідж. Львів. нац. ун-ту ім. І. Франка ; вид-во “Генеза” ; [за заг. ред. І. Підкови і Р. Шуста]. – Вид. 2-ге, доопрац. і допов. – К. : Генеза, 2001. – С. 1038–1039. – ISBN 966-504-179-7. 3. Чимський [має бути Чинський] Ян // Довідник з історії України (А-Я) / Ін-т іст. дослідж. Львів. нац. ун-ту ім. І. Франка ; вид-во “Генеза” ; [за заг. ред. І. Підкови і Р. Шуста]. – Вид 2-ге, доопрац. і допов. – К. : Генеза, 2001. – С. 1056–1057. – ISBN 966-504-179-7.
Авторські свідоцтва	1. А. с. 1007970 СССР, МКИЗ В 25 J 15/00. Устройство для захвата неориентированных деталей типа валов / В. С. Ваулин, В. Г. Кемайкин (СССР). – № 3360585/25–08 ; заявл. 23.11.81 ; опубл. 30.03.83, Бюл. № 12.
Патенти	1. Пат. 2187888 Российская Федерация, МПК7 Н 04 В 1/38, Н 04 J 13/00. Приемопередающее устройство / Чугаева В. И.; заявитель и патентообладатель Воронеж. науч.-исслед. ин-т связи. – № 2000131736/09 ; заявл. 18.12.00 ; опубл. 20.08.02, Бюл. № 23 (II ч.).
Стандарти	1. Графічні символи, що їх використовують на устаткуванні. Показчик та огляд (ISO 7000:2004, IDT) : ДСТУ ISO 7000:2004. – [Чинний від 2006–01–01]. – К. : Держспоживстандарт України, 2006. – IV, 231 с. : табл. – (Національний стандарт України). 2. Якість води. Словник термінів : ДСТУ ISO 6107-1:2004 – ДСТУ ISO 6107-9:2004. – [Чинний від 2005 – 04 – 01]. – К. : Держспоживстандарт України, 2006. – 181 с. : табл. – (Національні стандарти України). – Текст: нім., англ., фр., рос., укр. 3. Вимоги щодо безпечності контрольно-вимірювального та лабораторного електричного устаткування. Частина 2-020. Додаткові вимоги до лабораторних центрифуг (EN 61010-2-020:1994, IDT) : ДСТУ EN 61010-2-020:2005. – [Чинний від 2007–01–01]. – К. : Держспоживстандарт України, 2007. — IV, 18 с. : табл. — (Національний стандарт України).

Приклади розташування основного напису і додаткових граф до нього

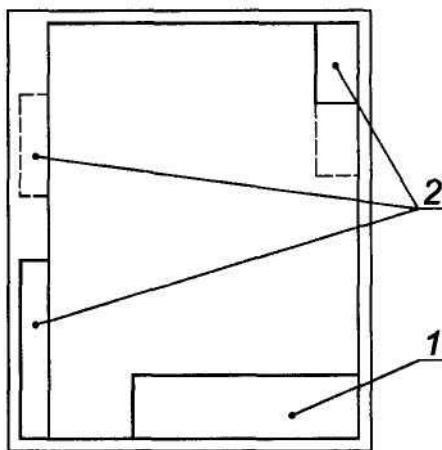
Для формату А4



Для форматів більше А4 при розташуванні основного напису вздовж довгої сторони листа



Для форматів більше А4 при розташуванні основного напису вздовж короткої сторони листа



1 – основний напис; 2 – додаткові графи

ГОСТ 19.402-78. ЕСПД. Описание программы

ГОСТ 19.402-78*

Группа Т55

Единая система программной документации

ОПИСАНИЕ ПРОГРАММЫ Unified system for program documentation . Program description СТ СЭВ 2092-80

с 01.01.80

1. Настоящий стандарт устанавливает состав и требования к содержанию программного документа «Описание программы», определенного ГОСТ 19.101-77.

Стандарт полностью соответствует СТ СЭВ 2092-80.

2. Структуру и оформление документа устанавливают в соответствии с ГОСТ 19.105-78. Составление информационной части (аннотации и содержания) является обязательным.

3. Описание программы должно содержать следующие разделы:

- общие сведения;
- функциональное назначение;
- описание логической структуры;
- используемые технические средства;
- вызов и загрузка;
- входные данные;
- выходные данные.

В зависимости от особенностей программы допускается вводить дополнительные разделы или объединять отдельные разделы.

4. В разделе «Общие сведения» должны быть указаны:

- обозначение и наименование программы;
- программное обеспечение, необходимое для функционирования программы;
- языки программирования, на которых написана программа.

5. В разделе «Функциональное назначение» должны быть указаны классы решаемых задач и (или) назначение программы и сведения о функциональных ограничениях на применение.

6. В разделе «Описание логической структуры» должны быть указаны:

- алгоритм программы;

- используемые методы;
- структура программы с описанием функций составных частей и связи между ними;
- связи программы с другими программами.

Описание логической структуры программы выполняют с учетом текста программы на исходном языке.

7. В разделе «Используемые технические средства» должны быть указаны тип электронных вычислительных машин и устройств, которые используются при работе программы.

8. В разделе «Вызов и загрузка» должны быть указаны:

- способ вызова программы с соответствующего носителя данных;
- входные точки в программу

Допускается указывать адреса загрузки, сведения об использовании оперативной памяти, объем программы.

9. В разделе «Входные данные» должны быть указаны:

- характер, организация и предварительная подготовка входных данных;
- формат, описание и способ кодирования входных данных.

10. В разделе «Выходные данные» должны быть указаны:

- характер и организация выходных данных;
- формат, описание и способ кодирования выходных данных.

11. Допускается содержание разделов иллюстрировать пояснительными примерами: таблицами, схемами, графиками.

12. В приложение к описанию программы допускается включать различные материалы которые нецелесообразно включать в разделы описания.

Изм. № 1 (ИУС 11-81).

ГОСТ 24.204-80. ТРЕБОВАНИЯ К СОДЕРЖАНИЮ ДОКУМЕНТА «ОПИСАНИЕ ПОСТАНОВКИ ЗАДАЧИ»

УДК 65.011.56:681.3:002:006.354

Группа Т52

ГОСУДАРСТВЕННЫЙ СТАНДАРТ СОЮЗА ССР

Система технической документации на АСУ	ГОСТ 24.204-80*
ТРЕБОВАНИЯ К СОДЕРЖАНИЮ ДОКУМЕНТА «ОПИСАНИЕ ПОСТАНОВКИ ЗАДАЧИ»	
System of technical documentation for computer control systems. Requirements for contents of document «Description of formulation of the task»	

Постановлением Государственного комитета СССР по стандартам от 14 мая 1980
г. № 2100 срок введения установлен

с 01.01.1981 г.

Настоящий стандарт распространяется на техническую документацию на автоматизированные системы управления (АСУ) всех видов, разрабатываемые для всех уровней управления (кроме общегосударственного), и устанавливает требования к содержанию документа «Описание постановки задачи».

1. ОБЩИЕ ПОЛОЖЕНИЯ

1.1. Документ «Описание постановки задачи» предназначен для описания характеристик комплекса задач (задачи), условий, необходимых для его решения, входной и выходной информации и совместно с «Техническим заданием» на создание АСУ определяет требования к видам обеспечения АСУ.

1.2. Содержание разделов должно охватывать все задачи комплекса.

1.3. В зависимости от назначения и специфических особенностей создаваемых АСУ допускается включать в документ дополнительные разделы и сведения, требования к содержанию которых не установлены настоящим стандартом.

1.4. Отсутствие проектных решений по разделу документа фиксируют в соответствующем разделе с необходимыми пояснениями.

(Введен дополнительно. Изм. 1).

2. ТРЕБОВАНИЯ К СОДЕРЖАНИЮ ДОКУМЕНТА

2.1. Документ должен содержать следующие разделы:

- характеристики комплекса задач;
- выходная информация;
- входная информация.

Примечание. При объединении документа «Описание постановки задачи» с документом «Описание алгоритма» последний помещают после раздела «Входная информация». Дублирование содержания разделов объединенного документа не допускается.

2.2. В разделе «Характеристика комплекса задач» следует приводить:

- цель, назначение, технико-экономическую (организационно-техническую) сущность комплекса задач и обоснование целесообразности его решения (в частности для задач оптимизации — критерий управления и ограничения);
- перечень объектов (технологических объектов управления, подразделений, предприятий и т. д.), при управлении которыми решают комплекс задач, при необходимости, — описание структуры объектов управления и перечень показателей характеризующих их состояние;
- описание назначения выходной информации;
- периодичность решения и ограничения по срокам выдачи выходной информации
- требования к организации сбора и передачи в обработку входной информации (с указанием сроков ее поступления), к порядку ее контроля и корректировки;
- условия, при которых прекращается решение комплекса задач автоматизированным способом;
- связи данного комплекса задач с другими комплексами (задачами) АСУ;
- должности лиц и (или) наименования подразделений, определяющих условия и временные характеристики конкретного решения

задачи (если они не определены общим алгоритмом функционирования системы);

- распределение функций между персоналом и техническими средствами пр; различных ситуациях решения комплекса задач.

2.3. Раздел «Выходная информация» должен содержать:

- перечень и описание выходных сообщений;
- перечень и описание имеющих самостоятельное смысловое значение структурных единиц информации выходных сообщений: показателей, реквизитов и их совокупностей, сигналов управления или ссылку на документы, содержащие эти данные

(Измененная редакция, Изм. № 1).

2.3.1. В описании по каждому выходному сообщению следует указывать:

- идентификатор;
- форму представления сообщения (документ, видеограмма, сигнал управления) и требования к ней;
- периодичность выдачи;
- сроки выдачи;
- получателей информации.

Состав описания допускается дополнять в зависимости от вида и особенностей сообщения.

2.3.2. В описании по каждой структурной единице информации следует указывать:

- наименование;
- идентификатор выходного сообщения, содержащего структурную единицу информации;
- требования к точности и надежности вычисления (при необходимости).

2.4. Раздел «Входная информация» должен содержать:

- перечень и описание входных сообщений;
- перечень и описание структурных единиц информации входных сообщений или ссылку на документы, содержащие эти данные.

(Измененная редакция, Изм. № 1).

2.4.1. В описании по каждому входному сообщению следует указывать:

- идентификатор;
- форму представления сообщения и частоту поступления.

2.4.2. В описании по каждой структурной единице информации входных сообщений следует указывать:

- наименование;
- требуемую точность ее числового значения (при необходимости);
- источник информации (документ, видеограмма, устройство, кодограмма, информационная база на машинных носителях и т. д.);
- идентификатор источника информации.

2.5. Иллюстрационный материал, таблицы или текст вспомогательного характера, а также документы, имеющие самостоятельные обозначения (чертежи форм документов, описания массивов информации, схемы и т. д.) допускается давать в виде приложений.

(Измененная редакция, Изм. № 1).

*Переиздание (май 1986 г.) с Изменением № 1, утвержденным в августе 1985 г. (ИУС11-85).

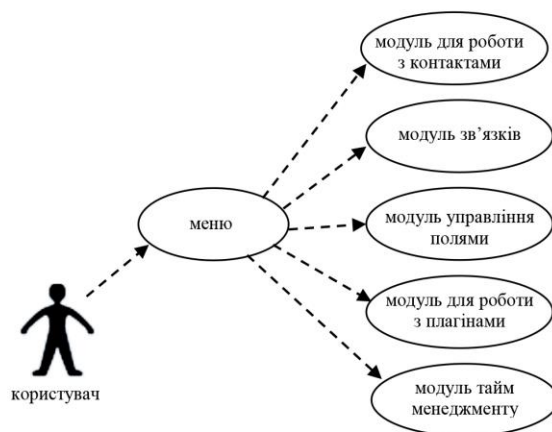
Зразки виконання графічної частини магістерської роботи

Тема: Розробка системи організації та управління персональною контактною інформацією

Мета роботи: Проектування та розробка програмного продукту, аналогу CRM, для структурування, організації та управління персональною контактною інформацією.

Завдання:

1. розглянути інформаційні системи і технології в сучасному суспільстві;
2. описати основні етапи розвитку інформаційних систем;
3. описати принципи функціонування корпоративних систем управління взаємозв'язками з клієнтами;
4. сформулювати вимоги до інформаційної системи;
5. провести вибір та обґрунтувати технології розробки;
6. спроектувати та розробити інтерфейс користувача.



UML-діаграма використання модуля роботи з контактами



ВИСНОВКИ

В ході виконання дипломної роботи ОКР “Магістр” зроблено наступне:

1. розглянуто інформаційні системи і технології в сучасному суспільстві;
2. описано основні етапи розвитку інформаційних систем ;
3. наведено класифікацію інформаційних систем;
4. розглянуто автоматизовані інформаційні системи для підприємств та організацій;
5. описано принципи функціонування корпоративних систем управління взаємозв'язками з клієнтами;
6. розглянуто соціальні медіа як сучасний стан еволюції засобів комунікації;
7. висвітлено поняття соціального графу.
8. сформовано вимоги до інформаційної системи;
9. проведено вибір та обґрунтування технології розробки.
10. створено модель даних;
11. розроблено відкриту архітектуру програмного засобу та інтерфейс модулів;
12. спроектовано та розроблено інтерфейс користувача.

[illegible]

[illegible]



Видавництво Тернопільського національного технічного університету ім. І. Пулюя

виготовляє підручники для вузів, методичну літературу, художні видання, надає редакційно-видавничі та поліграфічні послуги з набору тексту, розробки макетів і друку книги чи будь-якої іншої поліграфічної продукції (брошури, плакати, афіші, календарі).

КРИМ ТОГО, ВИДАВНИЦТВО ПРОПОНУЄ ТАКІ ПОСЛУГИ:

- дизайн візитівок, буклетів, вітальних листів;
- професійне вичитування і верстку;
- сканування та копіювання;
- чорно-білий і повноколірний друк.



м. Тернопіль,
вул. Гоголя, 8.
Тел.: 43-02-09.

e-mail.: vydavnytstvo@tu.edu.te.ua

Методичні вказівки

О. В. Маєвський, О. В. Мацюк, М. В. Приймак, Г. В. Шимчук

**МЕТОДИЧНІ ВКАЗІВКИ
до виконання**

**ДИПЛОМНОЇ РОБОТИ
ОКР «МАГІСТР»**

**для студентів спеціальності
8.05010101 – Інформаційні управляючі системи та
технології**

Редактор *Є. І. Гриценко*
Коректор *М. Д. Радик*
Комп'ютерне макетування *Р. Л. Федішин*

Формат 60х90/16. Ум. друк. арк. 12,79. Тираж 70 пр. Зам. № 2339.

Видавництво Тернопільського національного
технічного університету імені Івана Пулюя.
46001, м. Тернопіль, вул. Руська, 56.
Свідоцтво суб'єкта видавничої справи ДК № 4226 від 08.12.11.