

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра комп'ютерних наук
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр

(назва освітнього ступеня)

на тему: Розробка системи підтримки прийняття рішень на основі аналітичних показників

Виконав: студент IV курсу, групи СНЗ-41

спеціальності 122 Комп'ютерні науки
(шифр і назва спеціальності)

Будівський В.П.
(прізвище та ініціали)

(підпис)

Керівник

Гром'як Р.С.
(прізвище та ініціали)

(підпис)

Нормоконтроль

Шимчук Г.В.
(прізвище та ініціали)

(підпис)

Завідувач кафедри

Боднарчук І.О.
(прізвище та ініціали)

(підпис)

Рецензент

Карпінський М.П.
(прізвище та ініціали)

(підпис)

Тернопіль
2026

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра комп'ютерних наук
(повна назва кафедри)

ЗАТВЕРДЖУЮ

Завідувач кафедри

Боднарчук І.О.
(підпис) (прізвище та ініціали)

« 14 » травня 2026 р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

на здобуття освітнього ступеня Бакалавр
(назва освітнього ступеня)

за спеціальністю 122 Комп'ютерні науки
(шифр і назва спеціальності)

Студенту Будівському Володимирі Петровичу
(прізвище, ім'я, по батькові)

1. Тема роботи Розробка системи підтримки прийняття рішень на основі аналітичних показників

Керівник роботи Гром'як Роман Сильвестрович, к.ф.м наук, доцент кафедри КН
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від « 14 » травня 2026 року № 4/9-238

2. Термін подання студентом завершеної роботи 18 червня 2026р.

3. Вихідні дані до роботи Розробка системи підтримки прийняття рішень на основі аналітичних показників

4. Зміст роботи (перелік питань, які потрібно розробити)

Вступ. 1 Аналіз предметної області енергоспоживання 1.1 Теоретичні основи систем підтримки прийняття рішень 1.2 Аналіз предметної області та процесів прийняття рішень 1.3 Аналітичні показники як основа прийняття рішень 1.4 Огляд існуючих програмних рішень і технологій 1.5 Постановка задачі дослідження та формування вимог до системи 1.6 Висновок до першого розділу. 2 Проектна частина. Проектування системи підтримки прийняття рішень 2.1 Обґрунтування вибору архітектури системи 2.2 Проектування інформаційної моделі та структури бази даних 2.3 Розробка алгоритму обробки аналітичних показників 2.4 Проектування модуля підтримки прийняття рішень 2.5 Проектування інтерфейсу користувача та засобів візуалізації 2.6 Висновок до другого розділу

3. Практична частина. Практичні аспекти реалізації системи підтримки прийняття рішень

3.1 Вибір програмних засобів і технологій реалізації 3.2 Реалізація основних модулів системи 3.3 Тестування та перевірка працездатності системи 3.4 Аналіз результатів роботи розробленої системи 3.5 Оцінювання ефективності та напрями вдосконалення системи 3.6 Висновки до третього розділу 4. Безпека життєдіяльності, основи охорони праці.

Висновки. Перелік джерел. Додатки. 5. Перелік графічного матеріалу

1 Титульна сторінка. 2 Тема, Мета, Об'єкт, Предмет дослідження. 3 Завдання дослідження.

4 Актуальність дослідження. 5 Система підтримки прийняття рішень. 6 Архітектура системи підтримки прийняття рішень. 7 ER-діаграма бази даних. 8 Алгоритм обробки аналітичних показників. 9 Схема формування рекомендації на основі аналітичних показників

10 Макет інформаційної панелі користувача. 11 Структура програмних модулів системи

12 Приклад результатів тестування роботи системи. 13 Візуалізація результатів аналітичного оцінювання. 14 Головна інформаційна панель системи. 15 Сторінка введення даних

16 Висновки. 17 Завершальний.

АНОТАЦІЯ

Розробка системи підтримки прийняття рішень на основі аналітичних показників // Кваліфікаційна робота освітнього рівня «Бакалавр» // Будівський Володимир Петрович // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем і програмної інженерії, кафедра комп'ютерних наук, група СНз-41 // Тернопіль, 2026 // С.62 , рис. – 9, табл. – 9, кресл. – 17, додат. – 4, бібліогр. – 30.

Ключові слова: система підтримки прийняття рішень, аналітичні показники, нормалізація даних, база даних, веб-додаток, візуалізація.

Кваліфікаційна робота присвячена розробці системи підтримки прийняття рішень на основі аналітичних показників.

У першому розділі кваліфікаційної роботи розглянуто теоретичні основи систем підтримки прийняття рішень, проаналізовано предметну область, аналітичні показники та сформовано вимоги до системи.

У другому розділі виконано проєктування архітектури системи, інформаційної моделі бази даних, алгоритму обробки показників, модуля рекомендацій та інтерфейсу користувача.

У третьому розділі обґрунтовано вибір програмних засобів, описано реалізацію основних модулів системи, проведено тестування та проаналізовано результати її роботи.

У четвертому розділі висвітлено особливості безпечної роботи а також вимоги безпеки для з комп'ютерними системами.

Об'єкт дослідження: процес підтримки прийняття управлінських рішень у структурному підрозділі організації на основі аналізу кількісних та якісних показників.

Предмет дослідження: методи, алгоритми та програмні засоби обробки аналітичних показників у системі підтримки прийняття рішень.

ANNOTATION

Development of a Decision Support System Based on Analytical Indicators// Qualification work of the educational level «Bachelor» // Budivskyi Volodymyr Petrovych // Ternopil Ivan Pulyu National Technical University, Computer and Information Systems and Software Engineering Faculty, Computer Sciences Department, group SNz-41 // Ternopil, 2026 // P.62, fig. – 9, tabl. – 9, chair. – 17, annexes. – 4, references – 30.

Keywords: decision support system, analytical indicators, data normalisation, database, web application, visualisation.

The qualification work is devoted to the development of a decision support system based on analytical indicators.

The first section of the thesis examines the theoretical foundations of decision support systems, analyses the subject area and analytical indicators, and formulates the system requirements.

The second section outlines the design of the system architecture, the database information model, the indicator processing algorithm, the recommendation module and the user interface.

In the third section justifies the choice of software tools, describes the implementation of the system's main modules, conducts testing and analyses the results of its operation.

In the fourth section covers the principles of safe working practices and the safety requirements for working with computer systems.

Research object: the process of supporting managerial decision-making within a structural unit of an organisation based on the analysis of quantitative and qualitative indicators.

Research subject: methods, algorithms and software tools for processing analytical indicators in a decision-support system.

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

СППР — система підтримки прийняття рішень.

BI (англ. Business Intelligence) — бізнес-аналітика, сукупність методів і програмних засобів для збору, аналізу та візуалізації даних.

KPI (англ. Key Performance Indicator) — ключовий показник ефективності.

БД — база даних.

СУБД — система управління базами даних.

SQL (англ. Structured Query Language) — мова структурованих запитів для роботи з реляційними базами даних.

API (англ. Application Programming Interface) — програмний інтерфейс взаємодії між програмними компонентами.

UI (англ. User Interface) — користувацький інтерфейс.

UX (англ. User Experience) — досвід користувача під час взаємодії із системою.

CSV (англ. Comma-Separated Values) — текстовий формат подання табличних даних.

JSON (англ. JavaScript Object Notation) — текстовий формат обміну структурованими даними.

ORM (англ. Object-Relational Mapping) — технологія об'єктно-реляційного відображення для роботи з базою даних через об'єктну модель.

HTML (англ. HyperText Markup Language) — мова розмітки вебсторінок.

CSS (англ. Cascading Style Sheets) — мова стилізації вебсторінок.

JavaScript — мова програмування для реалізації інтерактивної поведінки вебінтерфейсу.

ЗМІСТ

ВСТУП	7
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	11
1.1 Теоретичні основи систем підтримки прийняття рішень	11
1.2 Аналіз предметної області та процесів прийняття рішень.....	14
1.3 Аналітичні показники як основа прийняття рішень	17
1.4 Огляд існуючих програмних рішень і технологій	20
1.5 Постановка задачі дослідження та формування вимог до системи ...	22
1.6 Висновок до першого розділу	24
РОЗДІЛ 2. ПРОЕКТНА ЧАСТИНА. ПРОЄКТУВАННЯ СИСТЕМИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ	25
2.1 Обґрунтування вибору архітектури системи.....	25
2.2 Проєктування інформаційної моделі та структури бази даних.....	27
2.3 Розробка алгоритму обробки аналітичних показників.....	30
2.4 Проєктування модуля підтримки прийняття рішень	32
2.5 Проєктування інтерфейсу користувача та засобів візуалізації.....	34
2.6 Висновок до другого розділу	36
РОЗДІЛ 3. ПРАКТИЧНА ЧАСТИНА. ПРАКТИЧНІ АСПЕКТИ РЕАЛІЗАЦІЇ СИСТЕМИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ.....	37
3.1 Вибір програмних засобів і технологій реалізації	37
3.2 Реалізація основних модулів системи	38
3.3 Тестування та перевірка працездатності системи.....	40
3.4 Аналіз результатів роботи розробленої системи	42
3.5 Оцінювання ефективності та напрями вдосконалення системи.....	44
3.6 Висновки до третього розділу	45
РОЗДІЛ 4. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ	46
4.1 Питання щодо безпеки життєдіяльності	46
4.2 Питання з основ охорони праці	49
Висновок до четвертого розділу	53
ВИСНОВКИ.....	54
ПЕРЕЛІК ДЖЕРЕЛ	56
ДОДАТКИ	

ВСТУП

Актуальність теми. Сучасні організації функціонують в умовах постійного зростання обсягів даних, ускладнення управлінських процесів і необхідності швидкого реагування на зміни внутрішнього та зовнішнього середовища. Ефективність управління значною мірою залежить від здатності своєчасно отримувати, аналізувати й інтерпретувати інформацію. Дані, які накопичуються в інформаційних системах організації, мають практичну цінність лише тоді, коли вони перетворюються на аналітичні висновки, показники ефективності та обґрунтовані рекомендації.

Одним із важливих напрямів розвитку сучасних інформаційних технологій є створення систем підтримки прийняття рішень. Такі системи поєднують засоби збирання, зберігання, обробки, аналізу та візуалізації даних і призначені для допомоги користувачу у виборі найбільш доцільного варіанта дій. На відміну від звичайних інформаційних систем, які переважно забезпечують зберігання й пошук інформації, системи підтримки прийняття рішень орієнтовані на аналітичну обробку даних, порівняння альтернатив, оцінювання стану об'єкта та формування рекомендацій [1].

Актуальність теми бакалаврської кваліфікаційної роботи зумовлена тим, що значна кількість управлінських рішень у структурних підрозділах організацій досі приймається на основі розрізаних таблиць, ручного аналізу, суб'єктивної оцінки окремих показників або фрагментарної звітності. За таких умов існує ризик неповного врахування важливих факторів, помилок у розрахунках, дублювання інформації та несвоєчасного виявлення проблемних напрямів діяльності. Особливої ваги набуває потреба у програмних рішеннях, які дозволяють об'єднати різні показники в єдину систему оцінювання, виконати їх нормалізацію, врахувати вагомість кожного критерію та сформулювати зрозумілу для користувача рекомендацію.

Аналітичні показники є одним із ключових інструментів обґрунтування управлінських рішень. Вони дають змогу кількісно або якісно описати стан

об'єкта, порівняти фактичні результати з очікуваними, оцінити ефективність діяльності та виявити відхилення від бажаного стану. Однак використання окремих показників без їх системного опрацювання не завжди забезпечує достатню якість управлінського висновку. Тому для підтримки прийняття рішень важливим є не лише розрахунок окремих показників, а й формування узагальненої інтегральної оцінки, яка враховує значущість кожного критерію [2].

Системи підтримки прийняття рішень можуть застосовуватися у виробництві, освіті, фінансах, логістиці, медицині, державному управлінні, бізнесі та інших сферах, де необхідно аналізувати велику кількість показників і приймати рішення на основі даних [3]. У межах цієї роботи доцільним є розгляд узагальненої предметної області — підтримки управлінських рішень у структурному підрозділі організації. Такий підхід дозволяє зосередитися на основних принципах побудови системи, не обмежуючи її використання лише однією вузькою галуззю.

Проблема, яка розглядається в роботі, полягає у недостатній автоматизації процесу обробки аналітичних показників і формування рекомендацій для прийняття рішень. У багатьох випадках користувач має справу з набором розрізнених даних, які потрібно вручну перевіряти, порівнювати, обчислювати, інтерпретувати та подавати у вигляді звітів. Це збільшує час підготовки управлінського рішення, знижує прозорість оцінювання та підвищує ймовірність помилки.

Особливістю запропонованого підходу є використання інтегральної оцінки, що формується на основі нормалізованих значень показників і вагових коефіцієнтів. Така оцінка дозволяє привести різномірні показники до єдиної шкали, врахувати їх відносну важливість і отримати узагальнений числовий результат. Надалі цей результат може бути інтерпретований через систему порогових значень, наприклад як низький, задовільний, достатній або високий рівень стану об'єкта аналізу.

Значну роль у таких системах відіграє візуалізація даних. Графіки, діаграми, інформаційні панелі й таблиці дозволяють швидше сприймати результати аналізу, виявляти відхилення, порівнювати показники між собою та оцінювати загальну ситуацію. Візуалізація не замінює аналітичну обробку, але робить її результати доступнішими для користувача [4]. Саме тому в межах роботи передбачається розроблення веборієнтованого користувацького інтерфейсу з інформаційною панеллю, блоком аналітичних показників, графічним поданням результатів і текстовими рекомендаціями.

Мета і задачі дослідження. Метою бакалаврської кваліфікаційної роботи є розробка системи підтримки прийняття рішень на основі аналітичних показників, яка забезпечує автоматизовану обробку вхідних даних, розрахунок часткових і інтегральних оцінок, візуалізацію результатів та формування рекомендацій для підвищення обґрунтованості управлінських рішень.

Для досягнення поставленої мети необхідно виконати такі завдання: проаналізувати теоретичні основи СППР; дослідити предметну область; розглянути аналітичні показники як основу прийняття рішень; провести огляд існуючих програмних рішень; сформулювати вимоги до системи; обґрунтувати її архітектуру; спроектувати інформаційну модель і базу даних; розробити алгоритм обробки показників; спроектувати модуль рекомендацій; розробити інтерфейс; реалізувати основні програмні модулі; провести тестування та оцінити ефективність запропонованого рішення.

Об'єктом дослідження є процес підтримки прийняття управлінських рішень у структурному підрозділі організації на основі аналізу кількісних та якісних показників.

Предметом дослідження є методи, алгоритми та програмні засоби обробки аналітичних показників у системі підтримки прийняття рішень.

У роботі використовуються методи аналізу наукових і технічних джерел, системного аналізу, моделювання даних, нормалізації показників, багатокритеріального оцінювання, алгоритмізації, проектування баз даних, програмної інженерії, візуалізації даних і тестування програмного

забезпечення. Використання цих методів дозволяє комплексно розглянути задачу: від теоретичного обґрунтування систем підтримки прийняття рішень до практичної реалізації програмного рішення.

Практичне значення роботи полягає у створенні програмної системи, яка може бути використана для автоматизації процесу аналізу аналітичних показників і підтримки прийняття управлінських рішень. Система може бути адаптована до різних предметних областей, у яких рішення приймаються на основі сукупності формалізованих показників. Розроблене рішення дає змогу зменшити час опрацювання даних, підвищити прозорість оцінювання, забезпечити зручне подання результатів і сформулювати рекомендації на основі формалізованих правил.

РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

У першому розділі бакалаврської кваліфікаційної роботи розглянуто теоретичні та прикладні передумови розробки системи підтримки прийняття рішень на основі аналітичних показників. Основну увагу приділено сутності систем підтримки прийняття рішень, їх місцю в інформаційному середовищі організації, ролі аналітичних показників у процесі оцінювання стану об'єкта та формуванню вимог до майбутньої програмної системи.

1.1 Теоретичні основи систем підтримки прийняття рішень

Системи підтримки прийняття рішень є одним із важливих класів інформаційних систем, призначених для допомоги користувачам у процесі аналізу даних, оцінювання альтернатив і вибору доцільного варіанта дій. На відміну від традиційних інформаційних систем, які переважно орієнтовані на зберігання, пошук і обробку операційних даних, СППР спрямовані на аналітичну інтерпретацію інформації та формування основи для обґрунтованого управлінського рішення [1].

Під системою підтримки прийняття рішень доцільно розуміти інтерактивну комп'ютерну систему, яка використовує дані, моделі, алгоритми, правила оцінювання та засоби візуалізації для допомоги особі, що приймає рішення. Така система не замінює користувача і не повинна повністю автоматизувати управлінське рішення. Її головне призначення полягає у зменшенні інформаційної невизначеності, структуруванні даних, виявленні закономірностей, порівнянні варіантів і поданні результатів аналізу у зручній формі.

У практичному аспекті СППР застосовуються в тих випадках, коли рішення не може бути прийняте лише на основі одного показника або простого правила. Зазвичай користувачу потрібно врахувати сукупність критеріїв,

порівняти різні аспекти діяльності, оцінити ризики, виявити проблемні напрями та визначити можливі дії. Саме тому СППР є корисними для задач управління, планування, контролю, моніторингу, оцінювання ефективності та прогнозування [2].

Основними функціями систем підтримки прийняття рішень є збирання та інтеграція даних із різних джерел, зберігання структурованої інформації у базі даних, попередня перевірка й очищення даних, розрахунок аналітичних показників, порівняння фактичних значень із нормативними або цільовими, формування часткових і узагальнених оцінок, використання моделей або правил оцінювання, візуалізація результатів аналізу, формування аналітичних висновків і збереження історії результатів.

Залежно від підходу до обробки інформації системи підтримки прийняття рішень можуть бути класифіковані за кількома ознаками. За типом основного ресурсу, який використовується для прийняття рішення, виділяють системи, орієнтовані на дані, моделі, знання, документи або комунікацію. Системи, орієнтовані на дані, використовують бази даних, сховища даних, запити, звіти та аналітичні панелі. Модельно-орієнтовані системи застосовують математичні, оптимізаційні, статистичні або імітаційні моделі. Системи, орієнтовані на знання, використовують правила, експертні оцінки та логічні механізми виведення [3].

У межах цієї бакалаврської роботи основну увагу приділено системі підтримки прийняття рішень, яка поєднує ознаки системи, орієнтованої на дані, та системи, орієнтованої на правила. З одного боку, вона працює з аналітичними показниками, їх значеннями, ваговими коефіцієнтами та результатами розрахунків. З іншого боку, система використовує порогові значення та правила інтерпретації інтегральної оцінки для формування рекомендацій користувачу.

Окремо слід розмежувати поняття систем підтримки прийняття рішень, звичайних інформаційних систем і систем бізнес-аналітики. Звичайна інформаційна система переважно забезпечує введення, зберігання, облік і

пошук даних. Система бізнес-аналітики зосереджена на звітності, інформаційних панелях, інтерактивній візуалізації та аналізі історичних даних. СППР, у свою чергу, поєднує аналітику з логікою оцінювання та формуванням рекомендацій [4].

Узагальнене місце системи підтримки прийняття рішень в інформаційному середовищі організації доцільно подати на рисунку 1.1.



Рисунок 1.1 – Місце системи підтримки прийняття рішень в інформаційному середовищі організації

Отже, система підтримки прийняття рішень у межах цієї роботи розглядається як веборієнтована інформаційна система, що забезпечує обробку аналітичних показників, формування інтегральної оцінки та надання рекомендації користувачу.

1.2 Аналіз предметної області та процесів прийняття рішень

Предметною областю бакалаврської кваліфікаційної роботи є підтримка управлінських рішень у структурному підрозділі організації на основі аналізу ключових аналітичних показників. Такий підхід дозволяє розглядати систему як інструмент, який може бути адаптований до різних типів організацій: освітніх, виробничих, сервісних, адміністративних або комерційних. Основною умовою використання системи є наявність набору показників, за якими можна оцінювати стан об'єкта аналізу.

У межах обраної предметної області структурний підрозділ організації розглядається як об'єкт управління, діяльність якого може бути описана за допомогою кількісних і якісних показників. До таких показників можуть належати рівень виконання плану, продуктивність роботи, якість виконання завдань, своєчасність виконання, рівень використання ресурсів, кількість помилок або відхилень, рівень задоволеності користувачів чи замовників, динаміка зміни результативності. Кожен із цих показників відображає окремий аспект діяльності, але для прийняття рішення необхідно оцінювати їх у сукупності.

Процес прийняття управлінського рішення зазвичай включає кілька етапів. Спочатку виникає потреба в оцінюванні певної ситуації або стану об'єкта. Далі здійснюється збирання даних, їх перевірка, аналіз і порівняння з очікуваними або нормативними значеннями. Після цього користувач визначає, чи є ситуація прийнятною, чи потребує вона коригування. Якщо виявлено відхилення або недостатній рівень ефективності, необхідно сформулювати рекомендацію щодо подальших дій.

У традиційному підході значна частина цих операцій виконується вручну. Дані можуть зберігатися у різних таблицях, документах або звітах. Користувач змушений самостійно порівнювати показники, виконувати розрахунки, визначати вагомість критеріїв і робити висновки. Такий процес є трудомістким і залежить від досвіду особи, яка виконує аналіз. Крім того, ручна обробка

даних підвищує ризик арифметичних помилок, дублювання інформації, неправильного трактування показників і несвоєчасного виявлення проблем.

Автоматизація цього процесу за допомогою системи підтримки прийняття рішень дозволяє формалізувати порядок аналізу. Система забезпечує єдиний підхід до зберігання даних, розрахунку показників, нормалізації, застосування вагових коефіцієнтів і формування інтегральної оцінки. Це робить процес оцінювання більш прозорим, повторюваним і контрольованим.

Основними користувачами системи можуть бути керівник структурного підрозділу, аналітик, відповідальний працівник або адміністратор системи. Керівник зацікавлений у швидкому отриманні узагальненої інформації про стан підрозділу та рекомендацій щодо подальших дій. Аналітик виконує введення, перевірку й інтерпретацію даних. Адміністратор забезпечує налаштування системи, керування користувачами, показниками та параметрами оцінювання.

Основною проблемою предметної області є не відсутність даних, а складність їх системної інтерпретації. У багатьох організаціях показники існують у вигляді окремих звітів, таблиць або статистичних зведень. Вони можуть бути корисними для локального аналізу, але не завжди дозволяють швидко сформулювати загальну оцінку стану об'єкта. Різні показники можуть мати різні одиниці вимірювання, шкали, напрями оптимізації та значущість для прийняття рішення.

Отже, предметна область характеризується наявністю різнорідних даних, потребою у комплексному оцінюванні, необхідністю формалізації правил прийняття рішень і потребою в наочному поданні результатів. Це обґрунтовує доцільність розробки системи підтримки прийняття рішень, яка поєднує базу даних, аналітичний модуль, алгоритм інтегрального оцінювання, модуль рекомендацій і засоби візуалізації.

Таблиця 1.1 – Основні інформаційні потоки системи підтримки прийняття рішень

№	Інформаційний потік	Джерело даних	Отримувач даних	Зміст інформації	Призначення
1	Введення значень показників	Користувач, файл CSV/XLSX, внутрішні звіти	Модуль введення даних	Фактичні значення показників за певний період	Формування початкового набору даних для аналізу
2	Передавання показників до бази даних	Модуль введення даних	База даних	Назви показників, значення, одиниці вимірювання, періоди	Централізоване зберігання аналітичної інформації
3	Отримання даних для аналізу	База даних	Аналітичний модуль	Значення показників, ваги, межі нормалізації	Підготовка даних до розрахунку оцінок
4	Розрахунок часткових оцінок	Аналітичний модуль	Модуль підтримки прийняття рішень	Нормалізовані та зважені значення показників	Формування основи для інтегральної оцінки
5	Формування інтегральної оцінки	Аналітичний модуль	Модуль рекомендацій	Узагальнене числове значення стану об'єкта	Визначення рівня ефективності або проблемності
6	Формування рекомендації	Модуль рекомендацій	Користувацький інтерфейс	Рівень стану, пояснення, рекомендація	Підтримка прийняття управлінського рішення
7	Візуалізація результатів	Аналітичний модуль, база даних	Користувач	Графіки, таблиці, діаграми, інформаційні картки	Наочне подання результатів аналізу
8	Збереження історії результатів	Модуль розрахунків	База даних	Попередні оцінки, рекомендації, дата аналізу	Порівняння результатів у часі та контроль динаміки

1.3 Аналітичні показники як основа прийняття рішень

Аналітичні показники є основним інформаційним ресурсом системи підтримки прийняття рішень. Вони відображають кількісні або якісні характеристики об'єкта аналізу та дозволяють оцінити його стан, ефективність, результативність або проблемність. Без чітко визначеної системи показників процес прийняття рішень стає менш структурованим, оскільки користувач не має формалізованої основи для порівняння, оцінювання та вибору дій [5].

Показник у контексті цієї роботи розглядається як вимірювана характеристика діяльності структурного підрозділу, що має назву, значення, одиницю вимірювання, допустимі межі, напрям оптимізації та ваговий коефіцієнт. Саме така структура дозволяє використовувати показники не лише для звітності, а й для автоматизованого оцінювання.

Аналітичні показники можуть бути класифіковані за кількома ознаками. За способом вираження вони поділяються на кількісні та якісні. Кількісні показники мають числове значення, наприклад кількість виконаних завдань, відсоток виконання плану, обсяг витрат або кількість помилок. Якісні показники можуть описувати рівень задоволеності, якість виконання, відповідність вимогам або експертну оцінку. Для використання в інформаційній системі якісні показники доцільно переводити у числову шкалу.

За характером обчислення показники можуть бути абсолютними, відносними та інтегральними. Абсолютні показники відображають безпосереднє значення певної величини. Відносні показники виражають співвідношення між величинами. Інтегральні показники формуються на основі кількох часткових показників і відображають узагальнену оцінку стану об'єкта.

За напрямом оптимізації показники можна поділити на стимулятори та дестимулятори. Стимуляторами є показники, зростання яких позитивно впливає на загальну оцінку. До них належать рівень виконання плану, продуктивність, якість, своєчасність, задоволеність користувачів. Дестимуляторами є

показники, зростання яких погіршує загальну оцінку. До них належать витрати ресурсів, кількість помилок, кількість відхилень або тривалість затримок.

Ключові показники ефективності є окремою групою аналітичних показників, які використовуються для оцінювання досягнення цілей організації або підрозділу. КРІ повинні бути пов'язані з конкретними управлінськими цілями, бути вимірюваними, зрозумілими для користувачів, доступними для регулярного розрахунку та придатними для порівняння в часі [6]. Для системи підтримки прийняття рішень важливо, щоб кожен КРІ мав чітко визначене джерело даних і спосіб інтерпретації.

Для розроблюваної системи доцільно використовувати обмежений, але змістовний набір показників. Надмірна кількість показників може ускладнити інтерпретацію результатів і перевантажити користувача. Водночас занадто малий набір показників може не відображати реальний стан об'єкта. Тому в межах бакалаврської роботи передбачено 5–8 основних показників, які характеризують різні аспекти діяльності структурного підрозділу.

Значення вагових коефіцієнтів у таблиці 1.2 є орієнтовними й можуть бути змінені залежно від предметної області та управлінських пріоритетів. Головна умова полягає в тому, що сума вагових коефіцієнтів повинна дорівнювати одиниці. Використання вагових коефіцієнтів може ґрунтуватися на експертній оцінці, управлінських пріоритетах або методах багатокритеріального аналізу. У подальшому для формальнішого визначення ваг може застосовуватися, наприклад, метод аналізу ієрархій [27].

Однією з ключових процедур у системі підтримки прийняття рішень є нормалізація показників. Вона необхідна тому, що різні показники можуть мати різні одиниці вимірювання та діапазони значень. Наприклад, рівень виконання плану може вимірюватися у відсотках, кількість помилок — у штуках, витрати ресурсів — у грошових одиницях, а якість виконання — за експертною шкалою. Без нормалізації такі значення неможливо коректно об'єднати в одну інтегральну оцінку.

Таблиця 1.2 – Приклад набору аналітичних показників для підтримки прийняття рішень

№	Назва показника	Тип показника	Напрямок оптимізації	Джерело даних	Орієнтовна вага	Інтерпретація
1	Рівень виконання плану	Кількісний, відносний	Максимізація	Планові та фактичні звіти	0,20	Відображає частку виконаних завдань або планових показників
2	Продуктивність роботи	Кількісний	Максимізація	Внутрішня статистика діяльності	0,15	Характеризує обсяг виконаної роботи за період
3	Якість виконання завдань	Кількісний/експертний	Максимізація	Оцінки контролю якості, акти перевірки	0,20	Показує відповідність результатів встановленим вимогам
4	Своєчасність виконання	Кількісний, відносний	Максимізація	Дані про строки виконання завдань	0,15	Відображає частку завдань, виконаних у встановлені терміни
5	Рівень витрат ресурсів	Кількісний	Мінімізація	Дані обліку ресурсів	0,10	Характеризує обсяг ресурсів, використаних для досягнення результату
6	Кількість помилок або відхилень	Кількісний	Мінімізація	Журнали помилок, звіти контролю	0,10	Відображає кількість недоліків або відхилень від вимог
7	Рівень задоволеності користувачів	Кількісний/експертний	Максимізація	Анкетування, оцінки користувачів	0,05	Показує сприйняття результатів діяльності користувачами або замовниками
8	Динаміка результативності	Кількісний	Максимізація	Історичні дані за періоди	0,05	Відображає покращення або погіршення результатів у часі

Отже, аналітичні показники є основою розроблюваної системи підтримки прийняття рішень. Їх правильний вибір, формалізація, нормалізація, зважування та агрегування забезпечують можливість переходу від розрізнених даних до узагальненої оцінки й рекомендації.

1.4 Огляд існуючих програмних рішень і технологій

Ринок сучасних програмних засобів аналітики містить значну кількість рішень, призначених для збирання, обробки, візуалізації та аналізу даних. Найпоширенішими серед них є BI-платформи, інформаційні панелі, системи звітності та спеціалізовані аналітичні застосунки. Вони дозволяють користувачам працювати з наборами даних, будувати інтерактивні звіти, виявляти тенденції та приймати рішення на основі фактичної інформації [7].

Microsoft Power BI є одним із поширених інструментів бізнес-аналітики, який дозволяє підключатися до різних джерел даних, створювати моделі, будувати інтерактивні інформаційні панелі та публікувати звіти. Його перевагами є інтеграція з екосистемою Microsoft, широкі можливості візуалізації, наявність засобів для розрахунків і підтримка роботи з різними форматами даних [8].

Tableau є платформою для візуального аналізу даних. Вона відома широкими можливостями інтерактивної візуалізації, зручністю побудови графіків і здатністю працювати з різними джерелами даних. Tableau часто використовується для дослідження даних, створення аналітичних панелей і презентації результатів [23]. Qlik Sense є BI-платформою, яка орієнтована на асоціативну модель аналізу даних. Вона дозволяє користувачам досліджувати взаємозв'язки між даними, створювати інтерактивні візуалізації та здійснювати аналітичний пошук [24].

Google Looker Studio є інструментом для створення інтерактивних звітів та інформаційних панелей. Він дозволяє підключати різні джерела даних, будувати візуалізації та надавати доступ до звітів іншим користувачам [25]. Metabase є відкритим інструментом для аналітики та побудови інформаційних панелей. Він дозволяє створювати запити до баз даних, будувати графіки й ділитися результатами з користувачами [26].

Таблиця 1.3 – Порівняльна характеристика існуючих програмних рішень для аналітики та підтримки прийняття рішень

Програмне рішення	Основне призначення	Переваги	Обмеження	Доцільність для теми роботи
Microsoft Power BI	Бізнес-аналітика, звіти, інтерактивні панелі	Інтеграція з Microsoft, широкі можливості візуалізації	Ліцензійні обмеження, потреба у навичках моделювання даних	Доцільний як приклад ВІ-рішення, але не повністю відповідає потребі у власному модулі рекомендацій
Tableau	Візуальний аналіз даних, інтерактивні панелі	Потужна візуалізація, зручність аналізу	Вартість, складність адміністрування	Корисний для порівняння можливостей візуалізації
Qlik Sense	Асоціативна бізнес-аналітика	Гнучкий аналіз зв'язків у даних, інтерактивність	Складність налаштування, ліцензування	Доцільний як приклад корпоративної аналітичної платформи
Google Looker Studio	Створення звітів та інформаційних панелей	Простота використання, доступність, інтеграція з сервісами Google	Обмеження складної логіки обробки	Доцільний для базових звітів, але недостатній для повної СППР
Metabase	Відкрита система звітності та аналітичних панелей	Відкритість, простота підключення до БД	Обмежена підтримка спеціалізованих алгоритмів прийняття рішень	Корисний як приклад відкритого аналітичного рішення
Власна система	Підтримка прийняття рішень на основі показників	Повна адаптація до задачі, власний алгоритм оцінювання, модуль рекомендацій	Потребує розробки, тестування й підтримки	Найбільш доцільний варіант для бакалаврської роботи

Проведений огляд показує, що готові ВІ-рішення мають значні переваги для побудови звітів, інформаційних панелей і візуального аналізу даних. Однак більшість таких інструментів орієнтована переважно на відображення й дослідження даних, а не на реалізацію спеціалізованої логіки підтримки

прийняття рішень. У межах цієї бакалаврської роботи важливо не лише показати аналітичні дані, а й реалізувати алгоритм нормалізації, зважування, інтегрального оцінювання та формування рекомендацій.

1.5 Постановка задачі дослідження та формування вимог до системи

На основі аналізу теоретичних основ систем підтримки прийняття рішень, предметної області, аналітичних показників та існуючих програмних рішень можна сформулювати задачу дослідження. Вона полягає у розробці веборієнтованої системи підтримки прийняття рішень, яка забезпечує введення або імпорт аналітичних даних, їх зберігання, обробку, нормалізацію, розрахунок інтегральної оцінки, візуалізацію результатів і формування рекомендацій для користувача.

Основна задача розробки може бути сформульована так: необхідно створити програмну систему, яка на основі значень аналітичних показників виконує їх нормалізацію, враховує вагові коефіцієнти, обчислює інтегральну оцінку стану об'єкта аналізу, визначає рівень стану відповідно до порогових значень і формує рекомендацію для користувача.

Функціональні вимоги описують те, які дії повинна виконувати система. Нефункціональні вимоги визначають якісні характеристики системи: зручність, надійність, безпечність, продуктивність, супроводжуваність і розширюваність. Обидва типи вимог є важливими, оскільки система має не лише виконувати потрібні розрахунки, а й бути придатною для реального використання.

Таблиця 1.3 – Порівняльна характеристика існуючих програмних рішень для аналітики та підтримки прийняття рішень

№	Тип вимоги	Назва вимоги	Зміст вимоги	Критерій виконання
1	Функціональна	Введення даних	Система повинна дозволити користувачу вводити значення аналітичних показників вручну	Користувач може додати значення показника через форму інтерфейсу
2	Функціональна	Імпорт даних	Система повинна підтримувати завантаження даних із файлу CSV або XLSX	Дані з файлу успішно зчитуються та передаються на перевірку
3	Функціональна	Керування показниками	Система повинна дозволити додавати, редагувати й видаляти показники	Адміністратор може змінити назву, вагу, межі та напрям оптимізації показника
4	Функціональна	Перевірка даних	Система повинна перевіряти коректність введених значень	Некоректні або порожні значення не допускаються до розрахунку
5	Функціональна	Нормалізація показників	Система повинна приводити різні показники до єдиної шкали	Для кожного показника формується нормалізоване значення від 0 до 1
6	Функціональна	Розрахунок інтегральної оцінки	Система повинна обчислювати інтегральну оцінку на основі нормалізованих значень і ваг	Користувач отримує підсумкову оцінку стану об'єкта
7	Функціональна	Формування рекомендації	Система повинна визначати рівень стану та формувати текстову рекомендацію	Для кожного рівня оцінки відображається відповідна рекомендація
8	Функціональна	Візуалізація результатів	Система повинна подавати результати у вигляді графіків, таблиць і діаграм	Користувач бачить інформаційну панель із результатами аналізу
9	Нефункціональна	Зручність використання	Інтерфейс має бути зрозумілим і логічно структурованим	Основні дії доступні користувачу без складного навчання
10	Нефункціональна	Супроводжуваність	Код системи має бути структурованим і придатним до доопрацювання	Модулі системи мають чітко розділені функції

Сформовані вимоги є основою для подальшого проектування архітектури, бази даних, алгоритмів обробки показників і користувацького інтерфейсу. Саме

ці вимоги будуть використані у проєктній частині для визначення структури майбутньої системи та її основних функціональних компонентів.

1.6 Висновок до першого розділу

У першому розділі розглянуто теоретичні основи систем підтримки прийняття рішень, визначено їх призначення, функції, класифікацію та роль в інформаційному середовищі організації. Показано, що СППР відрізняються від звичайних інформаційних систем тим, що не лише зберігають і відображають дані, а й забезпечують їх аналітичну обробку, інтерпретацію та використання для формування рекомендацій.

Проаналізовано предметну область, пов'язану з підтримкою управлінських рішень у структурному підрозділі організації. Визначено основні інформаційні потоки, користувачів системи, типові управлінські ситуації та проблеми ручного аналізу даних. Обґрунтовано, що для ефективного прийняття рішень необхідно використовувати формалізовану систему аналітичних показників, яка включає нормалізацію, вагові коефіцієнти, інтегральну оцінку та порогові правила інтерпретації.

Проведено огляд існуючих програмних рішень для аналітики та підтримки прийняття рішень. Встановлено, що готові рішення мають широкі можливості візуалізації й аналізу даних, однак для цілей бакалаврської роботи доцільною є розробка власної системи, яка дозволить реалізувати спеціалізований алгоритм обробки показників і формування рекомендацій. На основі проведеного аналізу сформульовано функціональні та нефункціональні вимоги до майбутньої системи.

РОЗДІЛ 2. ПРОЕКТНА ЧАСТИНА. ПРОЄКТУВАННЯ СИСТЕМИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ

У другому розділі бакалаврської кваліфікаційної роботи розглянуто проєктування системи підтримки прийняття рішень на основі аналітичних показників. Якщо в аналітичній частині було обґрунтовано актуальність, предметну область, показники та вимоги до системи, то проєктна частина зосереджується на побудові логічної моделі майбутнього програмного рішення.

2.1 Обґрунтування вибору архітектури системи

Архітектура системи підтримки прийняття рішень визначає склад її основних компонентів, їх функції, взаємозв'язки та порядок обміну даними. Для розроблюваної системи доцільно використати веборієнтовану клієнт-серверну архітектуру, яка забезпечує розділення користувацького інтерфейсу, серверної логіки, аналітичного модуля та бази даних. Такий підхід дозволяє структурувати програмне рішення, розмежувати відповідальність між модулями та забезпечити можливість розширення системи [9].

Веборієнтована архітектура передбачає, що користувач взаємодіє із системою через браузер. Клієнтська частина відповідає за подання інформації, введення даних, відображення таблиць, графіків, діаграм і рекомендацій. Серверна частина приймає запити користувача, перевіряє дані, викликає відповідні алгоритми обробки, взаємодіє з базою даних і повертає результати для відображення. База даних забезпечує зберігання користувачів, аналітичних показників, значень показників, результатів розрахунків і сформованих рекомендацій.

У межах розроблюваної системи доцільно виділити такі основні компоненти: користувацький інтерфейс, серверну частину, модуль роботи з даними, базу даних, аналітичний модуль, модуль підтримки прийняття рішень,

модуль візуалізації та модуль збереження історії результатів. Модульний підхід дозволяє розробляти та тестувати окремі частини системи незалежно, а також спрощує її подальше вдосконалення [10].

Загальна логіка взаємодії компонентів полягає в тому, що користувач вводить або імпортує значення аналітичних показників через вебінтерфейс. Серверна частина приймає ці дані, виконує первинну перевірку та передає їх до модуля роботи з даними. Після цього дані зберігаються у базі даних або передаються до аналітичного модуля для розрахунків. Аналітичний модуль нормалізує показники, враховує вагові коефіцієнти та формує інтегральну оцінку. Далі модуль підтримки прийняття рішень інтерпретує отриманий результат і формує рекомендацію.

Для реалізації системи доцільно використати логічну модель, близьку до архітектурного підходу MVC, де модель відповідає за дані, представлення — за інтерфейс, а контролер — за обробку запитів і координацію роботи системи. Такий підхід дозволяє відокремити бізнес-логіку від інтерфейсу й роботи з базою даних, що спрощує супровід і тестування програмного рішення [11].

Архітектуру системи доцільно подати у вигляді рисунка 2.1.

Узагальнена архітектура системи підтримки прийняття рішень

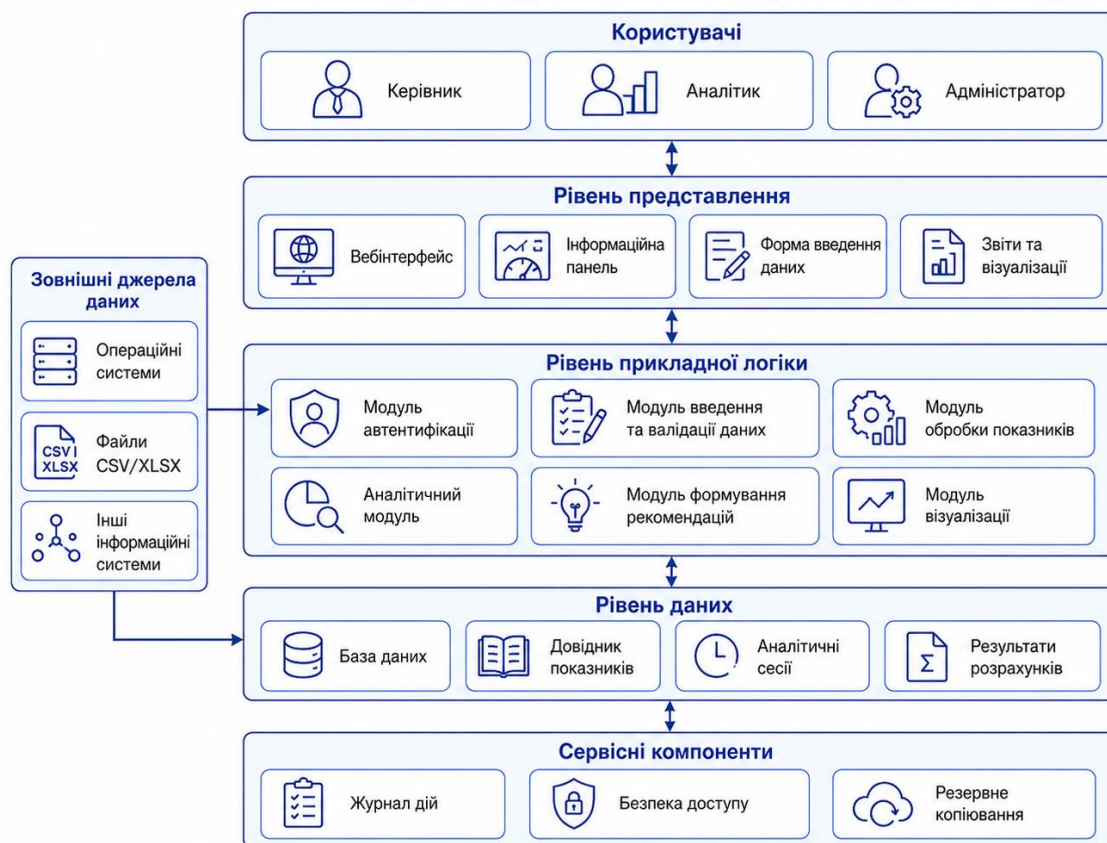


Рисунок 2.1 – Узагальнена архітектура системи підтримки прийняття рішень

2.2 Проектування інформаційної моделі та структури бази даних

Інформаційна модель системи визначає, які дані зберігаються в системі, як вони структуровані, які зв'язки існують між ними та яким чином ці дані використовуються для обробки аналітичних показників і формування рекомендацій. Для системи підтримки прийняття рішень інформаційна модель має особливе значення, оскільки коректність розрахунків безпосередньо залежить від якості, повноти та узгодженості даних.

База даних розробленої системи повинна забезпечувати зберігання даних користувачів системи, ролей і прав доступу, довідника аналітичних показників, значень аналітичних показників за певні періоди, параметрів нормалізації показників, вагових коефіцієнтів, результатів розрахунку

часткових оцінок, інтегральних оцінок, сформованих рекомендацій та історії виконаних аналітичних сесій.

Логічну структуру бази даних доцільно побудувати за реляційною моделлю. Реляційний підхід є придатним для цієї системи, оскільки дані мають чітку табличну структуру, між сутностями існують визначені зв'язки, а для роботи з ними можна використовувати SQL-запити. Крім того, реляційна модель забезпечує цілісність даних, можливість встановлення первинних і зовнішніх ключів, а також зручність подальшого розширення структури [12].

Основними сутностями системи є користувач, роль користувача, аналітичний показник, значення показника, аналітична сесія, результат розрахунку, інтегральна оцінка та рекомендація. Центральною сутністю інформаційної моделі є аналітична сесія. Вона об'єднує набір значень показників, результати їх обробки, інтегральну оцінку та сформовану рекомендацію. Такий підхід дозволяє зберігати історію оцінювань і порівнювати результати за різні періоди.

ER-діаграму бази даних доцільно подати у вигляді рисунка 2.2.

ER-діаграма бази даних системи підтримки прийняття рішень

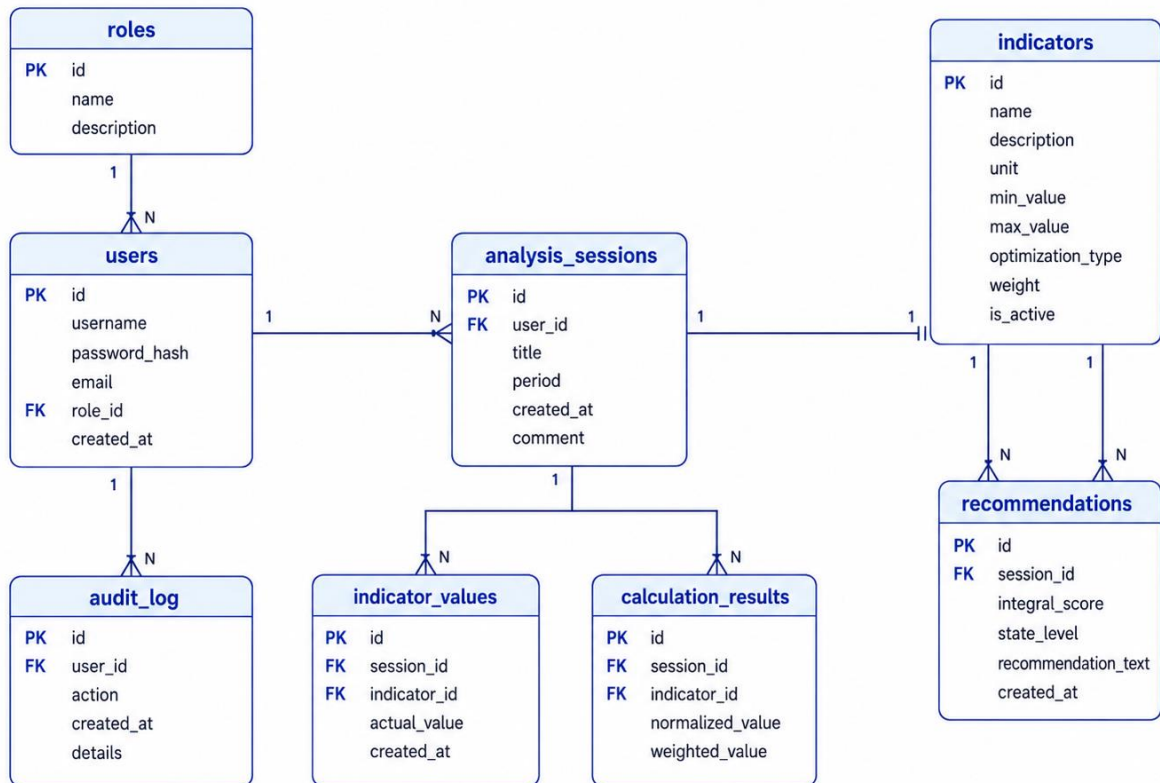


Рисунок 2.2 – ER-діаграма бази даних системи підтримки прийняття рішень

Для забезпечення цілісності даних у базі необхідно використовувати первинні та зовнішні ключі. Первинний ключ однозначно ідентифікує кожен запис у таблиці. Зовнішній ключ забезпечує зв'язок між таблицями. Наприклад, поле `indicator_id` у таблиці `indicator_values` повинно посилатися на існуючий запис у таблиці `indicators`. Це запобігає появі значень для неіснуючих показників.

Таблиця 2.1 – Опис основних таблиць бази даних

№	Назва таблиці	Призначення	Основні поля	Зв'язки
1	roles	Зберігання ролей користувачів	id, name, description	Пов'язана з таблицею users
2	users	Зберігання облікових записів користувачів	id, username, password_hash, email, role_id, created_at	Має зовнішній ключ role_id; пов'язана з analysis_sessions
3	indicators	Зберігання довідника аналітичних показників	id, name, unit, min_value, max_value, optimization_type, weight, is_active	Пов'язана з indicator_values і calculation_results
4	analysis_sessions	Зберігання окремих запусків оцінювання	id, user_id, title, period, created_at, comment	Пов'язана з users, indicator_values, calculation_results, recommendations
5	indicator_values	Зберігання фактичних значень показників	id, session_id, indicator_id, actual_value, created_at	Пов'язана з analysis_sessions і indicators
6	calculation_results	Зберігання результатів обробки окремих показників	id, session_id, indicator_id, normalized_value, weighted_value	Пов'язана з analysis_sessions та indicators
7	recommendations	Зберігання підсумкових оцінок і рекомендацій	id, session_id, integral_score, state_level, recommendation_text, created_at	Пов'язана з analysis_sessions

2.3 Розробка алгоритму обробки аналітичних показників

Алгоритм обробки аналітичних показників є центральним елементом розроблюваної системи підтримки прийняття рішень. Саме він забезпечує перехід від вхідних значень показників до узагальненої інтегральної оцінки та рекомендації. Алгоритм повинен бути формалізованим, зрозумілим, повторюваним і придатним для програмної реалізації.

Основна складність полягає в тому, що показники можуть мати різні одиниці вимірювання, різні діапазони значень і різний напрям оптимізації. Наприклад, рівень виконання плану потрібно максимізувати, а кількість помилок — мінімізувати. Тому перед об'єднанням показників у спільну оцінку необхідно привести їх до єдиної шкали. Для цього використовується нормалізація.

Алгоритм обробки аналітичних показників складається з таких етапів: отримання вхідних даних, перевірка коректності даних, визначення параметрів показників, нормалізація значень, розрахунок часткових оцінок, застосування вагових коефіцієнтів, формування інтегральної оцінки, визначення рівня стану об'єкта, формування рекомендації, збереження результатів і підготовка даних до візуалізації.

Для показників, які потрібно максимізувати, нормалізоване значення розраховується за формулою:

$$S_i = (X_i - X_{\min}) / (X_{\max} - X_{\min}), \quad (2.1)$$

де S_i — нормалізоване значення i -го показника; X_i — фактичне значення i -го показника; X_{\min} — мінімальне значення показника; X_{\max} — максимальне значення показника.

Для показників, які потрібно мінімізувати, використовується формула:

$$S_i = (X_{\max} - X_i) / (X_{\max} - X_{\min}) \quad (2.2)$$

Інтегральна оцінка формується як сума зважених оцінок:

$$I = \sum w_i \cdot S_i, \quad i = 1 \dots n, \quad (2.3)$$

де I — інтегральна оцінка стану об'єкта аналізу; w_i — ваговий коефіцієнт i -го показника; S_i — нормалізоване значення i -го показника; n — кількість активних показників.

Алгоритм обробки аналітичних показників доцільно подати у вигляді рисунка 2.3.

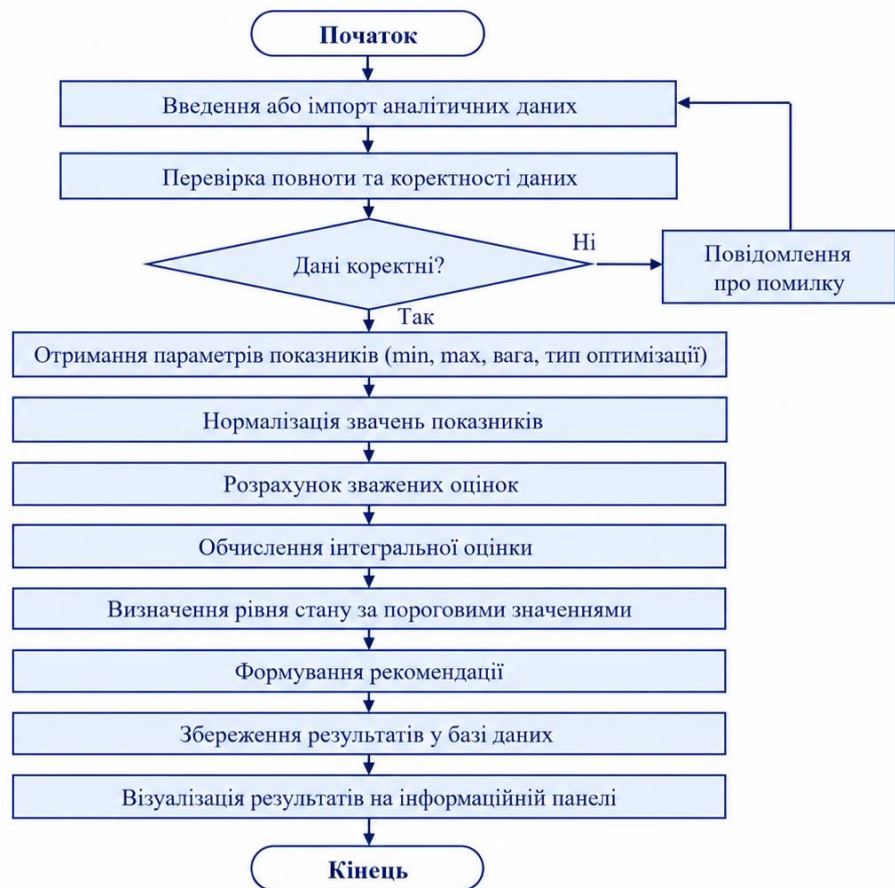


Рисунок 2.3 – Алгоритм обробки аналітичних показників у системі

У повному вигляді алгоритм передбачає створення аналітичної сесії, введення або імпорт значень показників, перевірку повноти та коректності даних, отримання параметрів показників із бази даних, розрахунок нормалізованих і зважених значень, формування інтегральної оцінки, визначення рівня стану, створення рекомендації, збереження результатів і відображення їх на інформаційній панелі.

2.4 Проєктування модуля підтримки прийняття рішень

Модуль підтримки прийняття рішень є функціональним компонентом системи, який перетворює результати аналітичних розрахунків на зрозумілу для користувача рекомендацію. Його завдання полягає не лише у відображенні

інтегральної оцінки, а й в інтерпретації отриманого результату відповідно до визначених правил.

У розроблюваній системі модуль підтримки прийняття рішень працює на основі інтегральної оцінки, нормалізованих значень окремих показників, зважених значень, порогових рівнів стану, переліку показників із найнижчими оцінками та правил формування рекомендацій.

Основою роботи модуля є правила прийняття рішень. У межах бакалаврської роботи доцільно використати порогову модель, оскільки вона є зрозумілою, формалізованою та придатною для програмної реалізації. Така модель передбачає, що інтегральна оцінка порівнюється з визначеними інтервалами, кожному з яких відповідає певний рівень стану та рекомендація.

Схему формування рекомендації доцільно подати у вигляді рисунка 2.4.

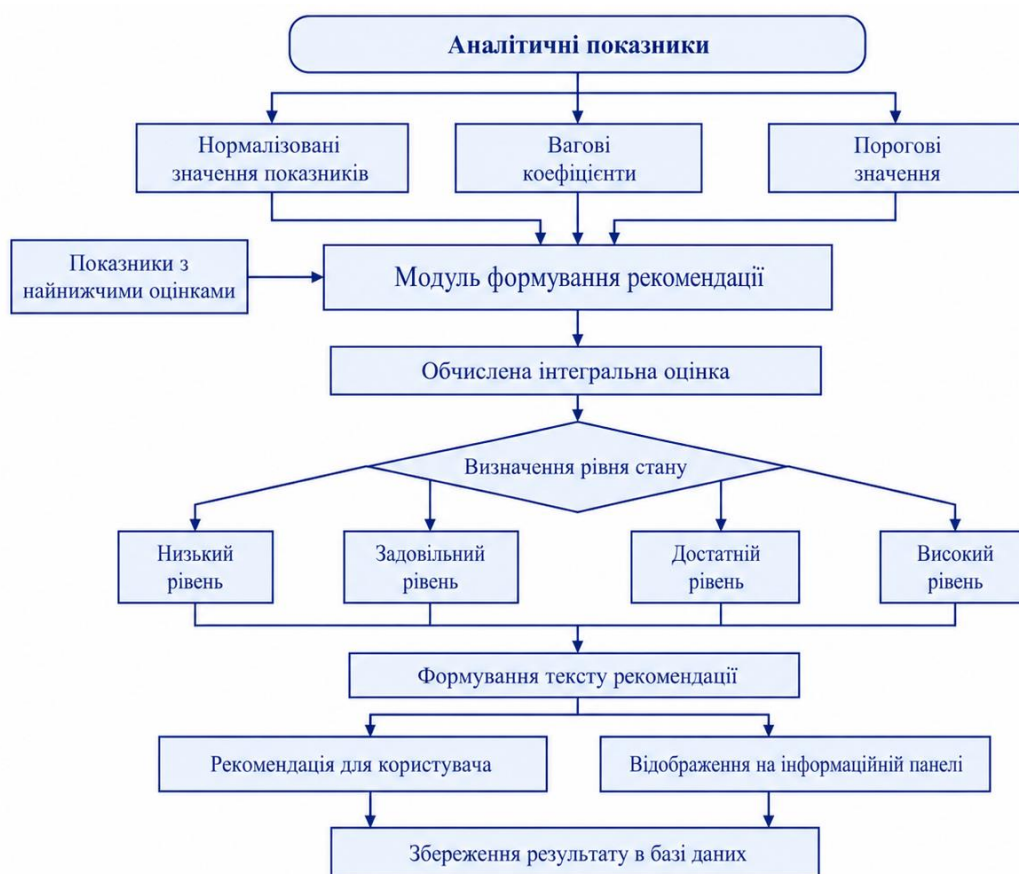


Рисунок 2.4 – Схема формування рекомендації на основі аналітичних показників

Таблиця 2.2 – Правила формування рекомендацій системи

№	Інтервал інтегральної оцінки	Рівень стану	Загальна характеристика	Рекомендація системи
1	0,00–0,39	Низький	Стан об'єкта є проблемним	Необхідно проаналізувати причини низьких результатів і розробити коригувальні заходи
2	0,40–0,59	Задовільний	Стан є частково прийнятним	Доцільно зосередити увагу на показниках із найнижчими оцінками
3	0,60–0,79	Достатній	Загальний стан є прийнятним	Рекомендовано підтримувати поточний рівень діяльності та покращити слабші показники
4	0,80–1,00	Високий	Стан об'єкта є позитивним	Рекомендовано підтримувати обрану стратегію і контролювати динаміку показників

Наведені правила є базовими й можуть бути доповнені більш деталізованою логікою. Наприклад, система може не лише визначати загальний рівень стану, а й аналізувати, які саме показники мають найнижчі нормалізовані значення. Це дозволить сформувавши рекомендацію не загального характеру, а з урахуванням конкретних слабких місць.

2.5 Проектування інтерфейсу користувача та засобів візуалізації

Користувацький інтерфейс є важливою складовою системи підтримки прийняття рішень, оскільки саме через нього користувач взаємодіє з даними, запускає розрахунки, переглядає результати та отримує рекомендації. Навіть коректно реалізований аналітичний алгоритм не буде ефективним, якщо результати його роботи подано незрозуміло або незручно.

Основними вимогами до інтерфейсу є зрозуміла структура сторінок, мінімальна кількість зайвих дій, логічне групування функцій, наочне подання ключових показників, швидкий доступ до інтегральної оцінки та рекомендації, можливість перегляду детальних результатів, адаптивність для різних розмірів екрана та використання таблиць, графіків і діаграм для пояснення результатів.

У системі доцільно передбачити сторінку входу, головну інформаційну панель, сторінку керування показниками, сторінку введення даних, сторінку результатів аналізу, сторінку історії аналітичних сесій та сторінку налаштувань. Головна інформаційна панель має бути центральним елементом інтерфейсу. Саме на ній користувач повинен швидко отримати відповідь на основне питання: який поточний стан об'єкта аналізу та які дії рекомендовано виконати.

Макет інформаційної панелі доцільно подати у вигляді рисунка 2.5.

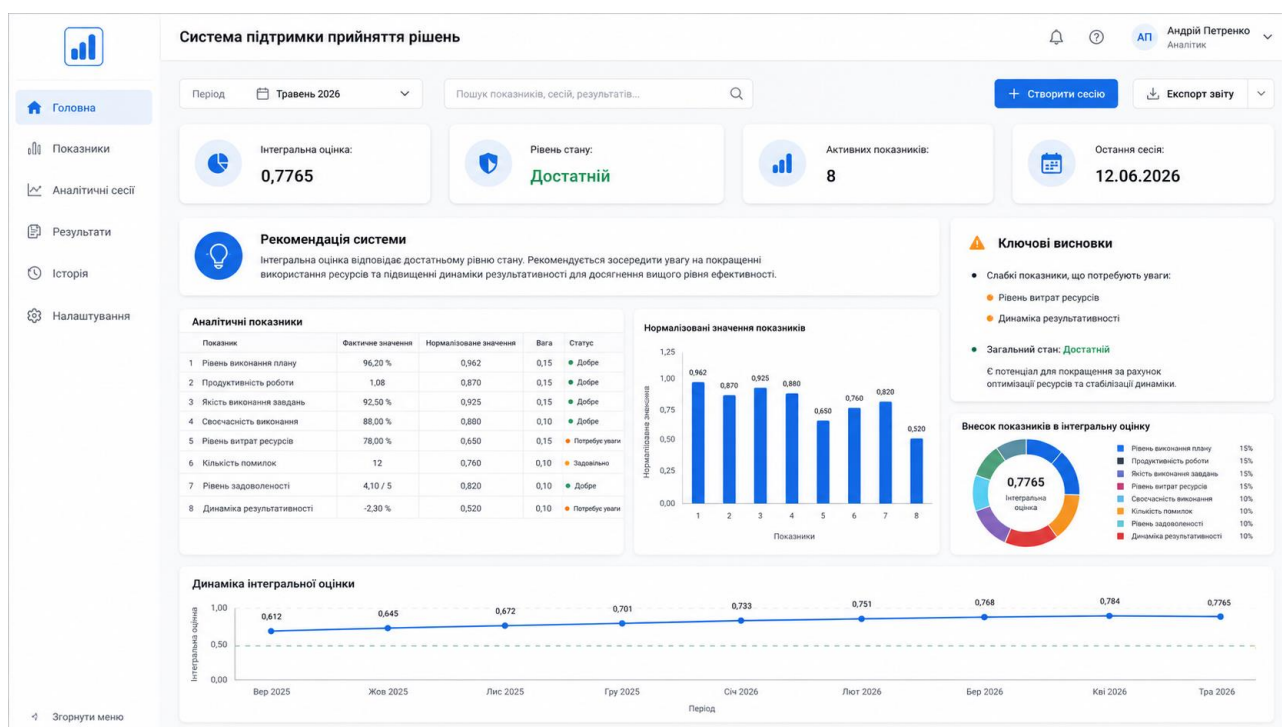


Рисунок 2.5 – Макет інформаційної панелі користувача системи підтримки прийняття рішень

Для візуалізації даних у системі доцільно використати кілька типів графічного подання: стовпчикову діаграму для порівняння нормалізованих оцінок окремих показників, кругову або кільцеву діаграму для відображення внеску показників у підсумкову оцінку, лінійний графік для відображення динаміки інтегральної оцінки за періодами та індикатор оцінки для наочного подання інтегральної оцінки від 0 до 1.

Сценарій взаємодії користувача із системою передбачає вхід до системи, створення нової аналітичної сесії, введення або імпорт значень показників, перевірку даних, запуск розрахунку, перегляд підсумкового результату, аналіз таблиці часткових показників, перегляд графіків і рекомендації, а також збереження результатів в історії.

2.6 Висновок до другого розділу

У другому розділі бакалаврської кваліфікаційної роботи було виконано проектування системи підтримки прийняття рішень на основі аналітичних показників. Обґрунтовано вибір веборієнтованої модульної архітектури, яка передбачає розділення системи на користувацький інтерфейс, серверну частину, базу даних, модуль обробки даних, аналітичний модуль, модуль формування рекомендацій і модуль візуалізації.

Спроектовано інформаційну модель і структуру бази даних системи. Визначено основні сутності: користувачі, ролі, аналітичні показники, значення показників, аналітичні сесії, результати розрахунків і рекомендації. Запропонована структура бази даних забезпечує збереження як початкових даних, так і результатів обробки.

Розроблено алгоритм обробки аналітичних показників, який включає введення даних, перевірку коректності, нормалізацію, застосування вагових коефіцієнтів, розрахунок інтегральної оцінки, визначення рівня стану та формування рекомендації. Описано логіку роботи модуля підтримки прийняття рішень на основі порогових правил і запропоновано структуру користувацького інтерфейсу.

РОЗДІЛ 3. ПРАКТИЧНА ЧАСТИНА. ПРАКТИЧНІ АСПЕКТИ РЕАЛІЗАЦІЇ СИСТЕМИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ

3.1 Вибір програмних засобів і технологій реалізації

Для реалізації системи доцільно обрати стек Python + Flask + SQLite + SQLAlchemy + Pandas + Chart.js + HTML/CSS/JavaScript + Bootstrap. Такий набір технологій є достатнім для створення веборієнтованої системи бакалаврського рівня, не потребує складної інфраструктури та дозволяє зосередитися на основній логіці роботи системи — обробці аналітичних показників і підтримці прийняття рішень.

Мова програмування Python є доцільною для реалізації серверної логіки та алгоритмів обробки даних. Вона має зрозумілий синтаксис, значну кількість бібліотек для роботи з даними, підтримує об'єктно-орієнтований і процедурний підходи, а також широко використовується у задачах аналізу даних, веброзробки та автоматизації обчислень [13]. Фреймворк Flask обрано для створення серверної частини вебзастосунку, оскільки він є легким, гнучким і придатним для невеликих прикладних систем [14].

Для взаємодії з базою даних доцільно використати SQLAlchemy, що забезпечує об'єктно-реляційне відображення і дозволяє працювати з таблицями через класи й об'єкти Python [15]. У базовій версії системи обрано SQLite, оскільки ця СУБД не потребує окремого серверного розгортання, зберігає дані у локальному файлі та забезпечує достатню функціональність для навчального й прикладного проекту [20].

Бібліотека Pandas може використовуватися для імпорту даних із CSV або XLSX-файлів, перевірки структури набору даних і підготовки даних до розрахунку [16; 22]. Для візуалізації результатів доцільно використати Chart.js, який інтегрується у вебсторінки та дозволяє будувати діаграми [17]. Користувацький інтерфейс реалізується з використанням HTML, CSS і JavaScript [21], а для адаптивного дизайну застосовується Bootstrap [18].

Таблиця 3.2 – Обґрунтування вибору програмних засобів реалізації системи

№	Технологія	Призначення в системі	Переваги використання	Обмеження
1	Python	Серверна логіка, алгоритми обробки показників	Простий синтаксис, широкі можливості аналізу даних	Потребує правильного структурування проєкту
2	Flask	Побудова серверної частини вебзастосунку	Легкість, гнучкість, простота маршрутизації	Для великих систем потребує додаткової архітектури
3	SQLite	Зберігання даних системи	Не потребує окремого сервера, проста у використанні	Обмежена масштабованість
4	SQLAlchemy	Робота з базою даних через об'єктну модель	Спрощує доступ до даних	Потребує розуміння ORM
5	Pandas	Обробка табличних даних	Зручна робота з таблицями імпортом	Надлишкова для дуже малих наборів даних
6	Chart.js	Візуалізація результатів аналізу	Легка інтеграція у вебінтерфейс	Працює переважно на клієнтській стороні
7	HTML, CSS, JavaScript	Створення користувацького інтерфейсу	Базові вебтехнології	Потребують акуратного оформлення
8	Bootstrap	Адаптивний дизайн інтерфейсу	Готові компоненти	Типовий вигляд без стилізації

3.2 Реалізація основних модулів системи

Реалізація системи підтримки прийняття рішень передбачає створення набору взаємопов'язаних програмних модулів, кожен із яких виконує окрему функцію. Такий підхід відповідає модульній архітектурі, описаній у проєктній частині, і дозволяє спростити розробку, тестування та подальше вдосконалення системи.

У базовій версії програмний проєкт може містити головний файл застосунку `app.py`, файл конфігурації `config.py`, каталог `models` для опису моделей бази даних, каталог `services` для бізнес-логіки, каталог `templates` для HTML-шаблонів і каталог `static` для CSS- та JavaScript-файлів. Структуру програмних модулів системи доцільно подати на рисунку 3.1.

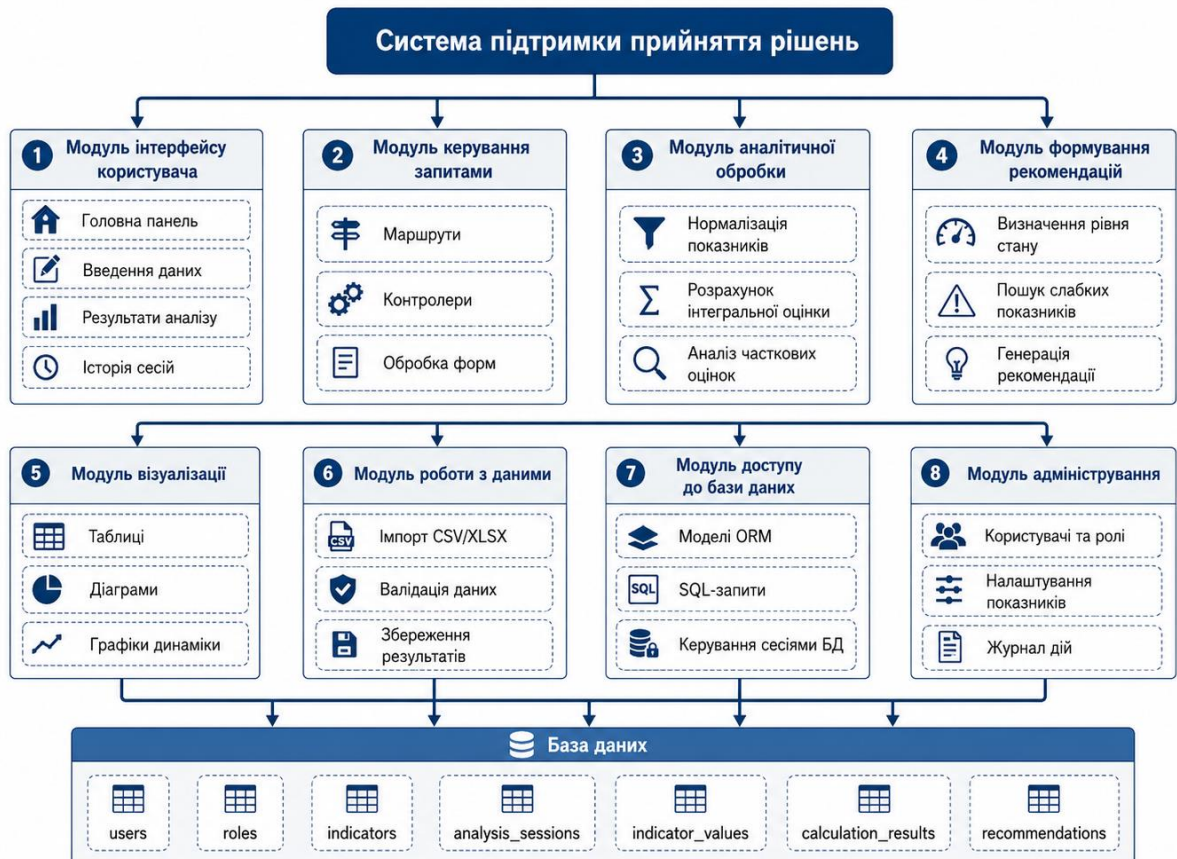


Рисунок 3.1 – Структура програмних модулів системи

Наведені фрагменти програмного коду ілюструють реалізацію ключових компонентів системи: моделей бази даних, введення значень показників, нормалізації, розрахунку інтегральної оцінки та формування рекомендацій. Вони не є повним лістингом програмного продукту, а демонструють основну логіку роботи системи. Додаткові фрагменти коду подано в додатку А.

```
class Indicator(db.Model):
    __tablename__ = "indicators"
    id = db.Column(db.Integer, primary_key=True)
    name = db.Column(db.String(120), nullable=False)
    unit = db.Column(db.String(30), nullable=False)
    min_value = db.Column(db.Float, nullable=False)
    max_value = db.Column(db.Float, nullable=False)
    optimization_type = db.Column(db.String(10), nullable=False)
    weight = db.Column(db.Float, nullable=False)
    is_active = db.Column(db.Boolean, default=True)
```

Модель Indicator відповідає таблиці аналітичних показників і містить назву показника, одиницю вимірювання, мінімальне та максимальне значення, напрям оптимізації, ваговий коефіцієнт і ознаку активності. Саме ці поля необхідні для подальшої нормалізації та інтегрального оцінювання.

Ключовою функцією аналітичного модуля є нормалізація. Вона враховує напрям оптимізації показника: для показників на максимізацію більше значення вважається кращим, а для показників на мінімізацію — навпаки. Після нормалізації значення обмежується діапазоном від 0 до 1, що забезпечує коректне агрегування.

```
def normalize_value(actual, min_value, max_value,
optimization_type):
    if max_value == min_value:
        raise ValueError("Некоректні межі показника")
    if optimization_type == "max":
        normalized = (actual - min_value) / (max_value -
min_value)
    elif optimization_type == "min":
        normalized = (max_value - actual) / (max_value -
min_value)
    else:
        raise ValueError("Невідомий тип оптимізації")
    return max(0, min(1, normalized))
```

Розрахунок інтегральної оцінки виконується шляхом послідовної нормалізації всіх активних показників, множення їх на вагові коефіцієнти та підсумовування зважених значень. Модуль рекомендацій порівнює отриману оцінку з пороговими значеннями та формує рівень стану об'єкта.

3.3 Тестування та перевірка працездатності системи

Тестування є необхідним етапом розробки програмної системи, оскільки дозволяє перевірити правильність реалізації функцій, коректність алгоритмів, стабільність роботи інтерфейсу та відповідність системи сформованим вимогам. Для системи підтримки прийняття рішень особливо важливо

перевірити не лише технічну працездатність, а й правильність обробки аналітичних показників [19].

Метою тестування є перевірка того, що система коректно приймає вхідні дані, виявляє некоректні або порожні значення, правильно зберігає дані в базі даних, виконує нормалізацію показників, застосовує вагові коефіцієнти, обчислює інтегральну оцінку, визначає рівень стану об'єкта, формує рекомендацію, відображає результати у таблицях і графіках та зберігає історію аналітичних сесій.

Тестове середовище може бути організоване на локальному комп'ютері. Для перевірки роботи системи достатньо використати локальний сервер Flask, базу даних SQLite і тестовий набір аналітичних показників.

Таблиця 3.2 – Тестові сценарії перевірки працездатності системи

№	Назва тестового сценарію	Вхідні дані	Очікуваний результат	Статус
1	Вхід користувача до системи	Коректний логін і пароль	Перехід на головну панель	Виконано
2	Введення значень показників	Повний набір числових значень	Дані зберігаються у БД	Виконано
3	Перевірка порожнього поля	Одне поле не заповнене	Повідомлення про помилку	Виконано
4	Перевірка нечислового значення	У полі введено текст	Розрахунок блокується	Виконано
5	Показник на максимізацію	Значення в межах діапазону	Коректна нормалізація	Виконано
6	Показник на мінімізацію	Значення в межах діапазону	Коректна нормалізація	Виконано
7	Розрахунок інтегральної оцінки	Нормалізовані значення та ваги	Оцінка відповідає формулі	Виконано
8	Формування рекомендації	Інтегральна оцінка 0,7765	Рівень — достатній	Виконано

Приклад результатів тестування роботи системи доцільно подати на рисунку 3.2.

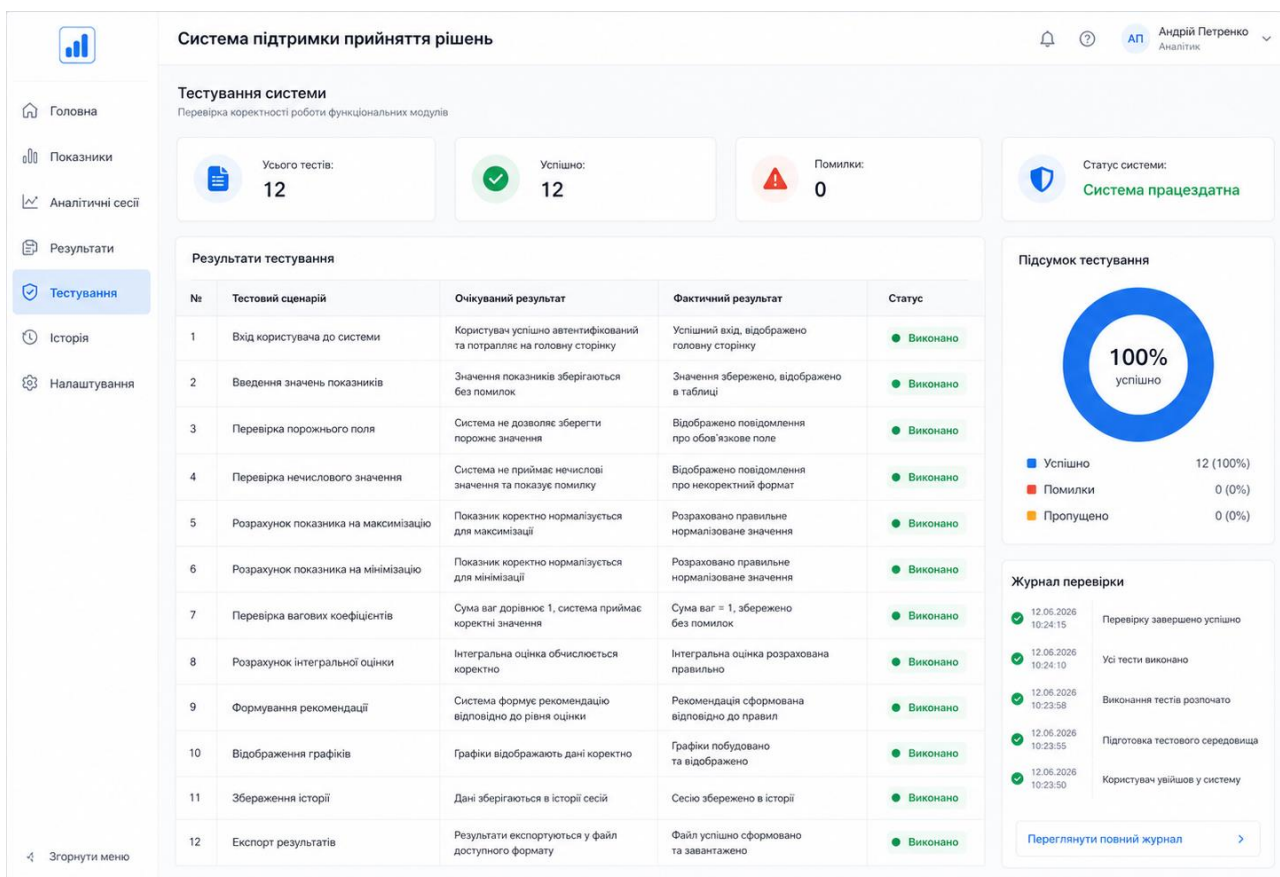


Рисунок 3.2 – Приклад результатів тестування роботи системи

3.4 Аналіз результатів роботи розробленої системи

Для перевірки роботи розробленої системи використано тестовий набір даних, який імітує оцінювання стану структурного підрозділу організації за вісьмома аналітичними показниками. Тестовий набір даних сформовано на основі переліку показників, визначених у таблиці 1.2. Для кожного показника задано фактичне значення, напрям оптимізації, ваговий коефіцієнт і межі нормалізації.

Таблиця 3.2 – Тестові сценарії перевірки працездатності системи

№	Показник	Фактичне значення	Напрямок	Вага
1	Рівень виконання плану, %	82	Макс.	0,20
2	Продуктивність роботи, ум. од.	96	Макс.	0,15
3	Якість виконання завдань, %	88	Макс.	0,20
4	Своєчасність виконання, %	76	Макс.	0,15
5	Рівень витрат ресурсів, ум. од.	95	Мін.	0,10
6	Кількість помилок або відхилень	5	Мін.	0,10
7	Рівень задоволеності користувачів	8	Макс.	0,05
8	Динаміка результативності, %	6	Макс.	0,05
	Підсумкова інтегральна оцінка			1,00

За результатами розрахунку інтегральна оцінка становить 0,7765. Відповідно до шкали інтерпретації, це значення належить до інтервалу 0,60–0,79, що відповідає достатньому рівню стану. Це означає, що загальний стан об'єкта аналізу є прийнятним, однак окремі показники потребують додаткової уваги.

Найнижчими нормалізованими значеннями у тестовому прикладі є рівень витрат ресурсів — 0,55, динаміка результативності — 0,65, кількість помилок або відхилень — 0,75 і своєчасність виконання — 0,76. Найбільший негативний вплив на підсумкову оцінку має показник витрат ресурсів, оскільки його нормалізоване значення є найнижчим.

На основі отриманої інтегральної оцінки система формує рекомендацію про підтримання поточного рівня діяльності з одночасною увагою до оптимізації використання ресурсів і підвищення динаміки результативності. Візуалізацію результатів аналітичного оцінювання доцільно подати на рисунку 3.3.

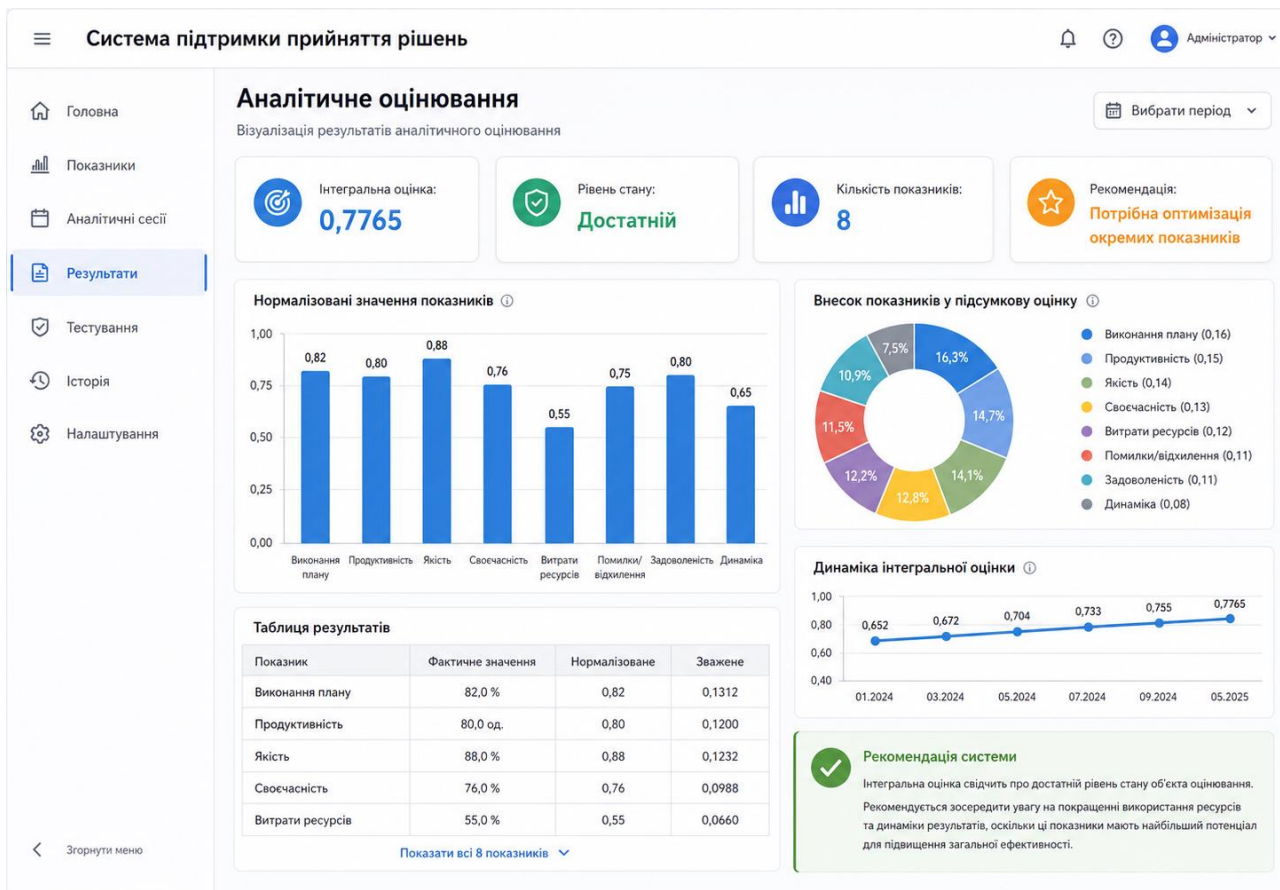


Рисунок 3.3 – Візуалізація результатів аналітичного оцінювання

3.5 Оцінювання ефективності та напрями вдосконалення системи

Ефективність розробленої системи визначається тим, наскільки вона спрощує роботу з показниками, підвищує прозорість оцінювання, зменшує час обробки даних і допомагає користувачу сформулювати обґрунтований висновок. До основних переваг можна віднести автоматизацію обробки показників, використання єдиної логіки оцінювання, підтримку показників на максимізацію та мінімізацію, урахування вагових коефіцієнтів, формування інтегральної оцінки, автоматичне визначення рівня стану та наочне подання результатів.

Практична ефективність системи проявляється передусім у зменшенні трудомісткості аналізу. Користувачу не потрібно самостійно виконувати всі розрахунки, перевіряти формули та будувати графіки. Система автоматично виконує ці операції та подає результат у структурованому вигляді.

Водночас система має обмеження: вона залежить від якості вхідних даних, правильності вибору показників і коректності вагових коефіцієнтів. Базова версія використовує порогові правила формування рекомендацій і не враховує складних залежностей між показниками. Подальший розвиток може включати інтеграцію із зовнішніми джерелами даних, розширення набору показників, аналіз динаміки, експорт звітів, покращення інтерфейсу та використання методів машинного навчання.

3.6 Висновки до третього розділу

У третьому розділі розглянуто практичну реалізацію системи підтримки прийняття рішень на основі аналітичних показників. Обґрунтовано вибір програмних засобів і технологій реалізації, зокрема Python, Flask, SQLite, SQLAlchemy, Pandas, Chart.js, HTML, CSS, JavaScript і Bootstrap. Обраний стек є достатнім для створення веборієнтованої системи бакалаврського рівня.

Описано реалізацію основних модулів системи: модуля роботи з базою даних, модуля введення даних, модуля нормалізації показників, модуля розрахунку інтегральної оцінки, модуля формування рекомендацій і користувачького інтерфейсу. Проведено тестування системи на контрольному наборі даних. За результатами тестування встановлено, що система коректно виконує введення, перевірку й обробку даних, розраховує інтегральну оцінку, визначає рівень стану та формує рекомендацію.

РОЗДІЛ 4. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

4.1 Питання щодо безпеки життєдіяльності

Безпека життєдіяльності є важливою складовою підготовки та виконання бакалаврської кваліфікаційної роботи, оскільки будь-яка професійна діяльність людини пов'язана з впливом зовнішніх і внутрішніх факторів, які можуть створювати загрозу для здоров'я, працездатності та життя. У межах теми «Розробка системи підтримки прийняття рішень на основі аналітичних показників» основна діяльність виконавця пов'язана з роботою за персональним комп'ютером, аналізом інформації, програмуванням, тестуванням вебзастосунку, обробкою даних і підготовкою текстової документації. Тому питання безпеки життєдіяльності доцільно розглядати з позиції безпечної організації робочого місця, запобігання негативному впливу технічних засобів, збереження працездатності користувача та правильних дій у разі виникнення надзвичайних ситуацій.

Під час виконання роботи основними потенційними небезпечними та шкідливими факторами є тривале перебування в сидячому положенні, значне зорове навантаження, монотонність роботи, психоемоційне напруження, можливе перевантаження уваги, вплив електричного обладнання, ризик пожежі внаслідок несправності електромережі або комп'ютерної техніки, а також небезпеки, пов'язані з аварійними чи надзвичайними ситуаціями. Оскільки розроблення системи підтримки прийняття рішень передбачає тривалу роботу з програмним кодом, таблицями, схемами, базою даних і текстовими матеріалами, особливого значення набуває раціональна організація робочого режиму.

Робоче місце користувача комп'ютерної техніки повинно бути організоване таким чином, щоб зменшити фізичне та психофізіологічне навантаження. Монітор необхідно розміщувати на відстані, яка забезпечує комфортне сприйняття інформації без надмірного напруження зору. Верхня

межа екрана бажано має бути приблизно на рівні очей або трохи нижче, що дозволяє уникнути перенапруження шийного відділу хребта. Крісло повинно забезпечувати підтримку спини, а положення рук під час роботи з клавіатурою і мишею має бути природним, без надмірного напруження кистей. Важливим є також достатнє освітлення робочої зони, відсутність відблисків на екрані, правильне розміщення джерел світла та регулярне провітрювання приміщення.

Для зменшення негативного впливу тривалої роботи за комп'ютером необхідно дотримуватися раціонального режиму праці та відпочинку. Під час виконання програмної частини роботи доцільно робити короткі перерви після тривалих періодів концентрації, виконувати вправи для очей, змінювати положення тіла, уникати надмірного безперервного сидіння. Такі заходи сприяють зниженню втоми, підтриманню уваги та запобіганню помилкам, які можуть виникати внаслідок перенавантаження. Для розробника програмного забезпечення це має не лише санітарно-гігієнічне, а й практичне значення, оскільки втома може призвести до помилок у коді, некоректної роботи алгоритмів або неправильного аналізу результатів тестування.

Окрему увагу слід приділити електробезпеці. Робота з комп'ютером, монітором, мережевим обладнанням, зарядними пристроями та периферійними засобами передбачає використання електричної енергії. Для безпечної експлуатації обладнання необхідно використовувати справні розетки, кабелі та блоки живлення, не допускати перевантаження електромережі, не використовувати пошкоджені дроти, не розміщувати рідину поруч із технікою та не виконувати самостійний ремонт обладнання без відповідної підготовки. Перед початком роботи варто переконатися у справності техніки, відсутності запаху перегріву, іскріння або нестабільної роботи пристроїв. У разі виявлення ознак несправності обладнання потрібно негайно припинити його використання та від'єднати від мережі.

Важливим елементом безпеки життєдіяльності є пожежна безпека. У приміщенні, де використовується комп'ютерна техніка, пожежна небезпека може виникати через коротке замикання, перегрів електроприладів,

несправність подовжувачів, накопичення пилю в системних блоках або недотримання правил експлуатації електрообладнання. Для запобігання пожежі необхідно підтримувати справний стан електромережі, не залишати обладнання без нагляду у разі його нестабільної роботи, не накривати вентиляційні отвори ноутбуків або системних блоків, не використовувати саморобні електричні з'єднання. У приміщенні мають бути вільними проходи до виходу, а користувач повинен знати порядок дій у разі пожежі.

У разі виникнення пожежі або задимлення насамперед необхідно припинити роботу, за можливості відключити електроживлення, повідомити відповідальних осіб і викликати аварійні служби. Якщо загоряння є незначним і його можна безпечно ліквідувати, допускається використання відповідного вогнегасника. Однак у разі загрози життю пріоритетом є евакуація з приміщення. Під час евакуації потрібно діяти організовано, не користуватися ліфтами, не створювати паніки, допомогти іншим особам у разі потреби та прямувати до безпечної зони.

Безпека життєдіяльності також охоплює готовність до дій у надзвичайних ситуаціях природного, техногенного або соціального характеру. До таких ситуацій можуть належати відключення електроенергії, аварії інженерних мереж, пожежі, загрози обстрілів, сигнали повітряної тривоги, надзвичайні погодні явища або інші обставини, які порушують нормальні умови роботи. У таких випадках важливо зберігати спокій, дотримуватися офіційних інструкцій, припинити роботу з обладнанням, за можливості зберегти важливі дані та перейти до безпечного місця.

Для роботи над програмною системою важливим є також резервне збереження даних. Втрата програмного коду, тексту роботи, бази даних або результатів тестування може бути наслідком технічної несправності, аварійного вимкнення електроенергії, помилки користувача або пошкодження носія інформації. Тому доцільно використовувати регулярне резервне копіювання, збереження версій файлів, хмарні сховища або зовнішні носії. Це стосується не лише інформаційної безпеки, а й безпеки життєдіяльності в широкому

розумінні, оскільки дозволяє зменшити наслідки аварійних ситуацій і забезпечити безперервність роботи.

Таким чином, під час виконання бакалаврської кваліфікаційної роботи необхідно враховувати комплекс питань безпеки життєдіяльності: ергономіку робочого місця, режим праці та відпочинку, електробезпеку, пожежну безпеку, готовність до дій у надзвичайних ситуаціях і захист інформаційних результатів роботи. Дотримання цих вимог сприяє збереженню здоров'я виконавця, підвищенню ефективності роботи, зменшенню ризику аварійних ситуацій і забезпеченню безпечних умов під час розробки системи підтримки прийняття рішень.

4.2 Питання з основ охорони праці

Охорона праці є системою правових, організаційних, технічних, санітарно-гігієнічних і профілактичних заходів, спрямованих на збереження життя, здоров'я та працездатності людини в процесі трудової діяльності. У контексті бакалаврської кваліфікаційної роботи на тему «Розробка системи підтримки прийняття рішень на основі аналітичних показників» питання охорони праці пов'язані передусім із безпечною організацією роботи користувача персонального комп'ютера, умовами праці в офісному або навчальному приміщенні, запобіганням професійному перенавантаженню та дотриманням вимог безпечної експлуатації електронної техніки.

Процес розробки програмного забезпечення має специфічні особливості з погляду охорони праці. На перший погляд така діяльність не пов'язана з високим рівнем фізичної небезпеки, однак тривала інтелектуальна робота за комп'ютером створює комплекс шкідливих факторів. До них належать статичне навантаження на опорно-руховий апарат, зорове напруження, монотонність робочих операцій, підвищене навантаження на нервову систему, ризик емоційного виснаження, вплив електромагнітних полів у межах допустимих

рівнів, шум від обладнання, недостатнє або надмірне освітлення, а також ризики, пов'язані з електричним обладнанням.

Під час розробки системи підтримки прийняття рішень виконавець працює з текстовими редакторами, середовищами програмування, веббраузером, базою даних, графічними матеріалами та документацією. Така робота потребує високої концентрації уваги, точності та тривалого перебування перед екраном. Тому однією з головних вимог охорони праці є правильна організація робочого місця. Робочий стіл має забезпечувати достатній простір для розміщення монітора, клавіатури, миші, документів і допоміжних матеріалів. Необхідно уникати захаращення робочої поверхні, оскільки це може створювати додаткове навантаження та підвищувати ризик випадкового пошкодження обладнання.

Важливою умовою безпечної праці є правильне положення тіла під час роботи. Крісло повинно мати зручну спинку, бажано з можливістю регулювання висоти. Ступні мають розташовуватися на підлозі або спеціальній підставці, спина повинна мати опору, а передпліччя — перебувати у зручному положенні під час роботи з клавіатурою. Неправильна посадка може призводити до болю у спині, шиї, плечах, кистях, а також до загального зниження працездатності. Для профілактики таких порушень необхідно періодично змінювати положення тіла, робити короткі перерви, виконувати прості вправи для спини, шиї, рук і очей.

Освітлення робочого місця повинно бути достатнім, рівномірним і не створювати відблисків на екрані монітора. Недостатнє освітлення призводить до зорового перенапруження, а надмірно яскраве або неправильно спрямоване світло може викликати дискомфорт і втому очей. Бажано поєднувати природне та штучне освітлення, розміщуючи монітор так, щоб пряме світло не потрапляло на екран і не засліплювало користувача. Екран повинен мати достатню яскравість і контрастність, а розмір шрифту має забезпечувати комфортне читання тексту без нахилу голови вперед.

Мікроклімат у приміщенні також впливає на умови праці. Надмірна температура, сухість повітря, недостатня вентиляція або підвищений рівень шуму можуть знижувати працездатність і спричиняти втому. Приміщення, у якому виконується робота, повинно регулярно провітрюватися, а технічне обладнання не повинно створювати надмірного шумового навантаження. У разі використання ноутбука або системного блока необхідно стежити за належною вентиляцією пристроїв, не перекривати вентиляційні отвори та не допускати перегрівання обладнання.

Значну увагу слід приділяти електробезпеці. Усі електроприлади, що використовуються під час роботи, повинні бути справними. Не допускається використання пошкоджених кабелів, нестійких розеток, несправних подовжувачів або перевантаження електромережі великою кількістю пристроїв. Перед підключенням обладнання необхідно перевірити цілісність кабелів і роз'ємів. Забороняється торкатися електроприладів мокрими руками, розміщувати рідину поблизу комп'ютерної техніки, самостійно розбирати блоки живлення або інші електричні вузли без спеціальної підготовки. У разі появи запаху горілого, іскріння, перегрівання або нестабільної роботи пристрою необхідно припинити його експлуатацію.

Пожежна безпека є невід'ємною складовою охорони праці. Комп'ютерне обладнання, мережеві пристрої, зарядні блоки та подовжувачі можуть бути джерелом пожежної небезпеки у разі несправності або неправильної експлуатації. Для запобігання пожежі потрібно підтримувати порядок на робочому місці, не залишати ввімкнені несправні електроприлади без нагляду, не допускати накопичення пилу в техніці, уникати перевантаження розеток і забезпечувати доступ до засобів пожежогасіння. Працівник або студент повинен знати місце розташування первинних засобів пожежогасіння, евакуаційних виходів і порядок дій у разі виникнення пожежі.

Окремим аспектом охорони праці є психофізіологічна безпека. Робота над програмним продуктом часто супроводжується тривалим розумовим напруженням, необхідністю пошуку помилок, аналізом складної інформації та

роботою в умовах обмеженого часу. Такі фактори можуть призводити до втоми, зниження концентрації, дратівливості та помилок. Для запобігання цьому необхідно планувати робочий час, розподіляти складні задачі на менші етапи, робити перерви, уникати надмірного нічного навантаження та підтримувати баланс між роботою і відпочинком. З погляду якості програмної розробки це також важливо, оскільки перевтома може негативно впливати на уважність під час написання коду, налаштування бази даних або тестування системи.

Під час тестування розробленої системи необхідно дотримуватися безпечного режиму роботи з комп'ютерним обладнанням і програмним середовищем. Якщо тестування проводиться на локальному сервері, потрібно контролювати навантаження на пристрій, не запускати неперевірені файли, не використовувати сумнівні джерела даних і не підключати сторонні носії без перевірки. Це поєднує питання охорони праці з базовими аспектами інформаційної безпеки, оскільки порушення цілісності програмного середовища може призвести до втрати результатів роботи або пошкодження даних.

Загалом організація безпечних умов праці під час розробки системи підтримки прийняття рішень повинна включати раціональне планування робочого часу, правильне облаштування робочого місця, дотримання вимог електробезпеки, пожежної безпеки, ергономіки, мікроклімату та психофізіологічного комфорту. Виконання цих вимог дозволяє знизити ризик професійного перенавантаження, зберегти працездатність виконавця, забезпечити безперервність роботи над програмним продуктом і підвищити якість отриманих результатів.

Таким чином, питання охорони праці у межах бакалаврської кваліфікаційної роботи мають прикладне значення, оскільки безпечні умови роботи безпосередньо впливають на ефективність розробки, якість програмного рішення та збереження здоров'я виконавця. Дотримання вимог охорони праці є необхідною умовою організації професійної діяльності у сфері комп'ютерних наук і розробки інформаційних систем.

Висновок до четвертого розділу

У процесі розробки системи підтримки прийняття рішень на основі аналітичних показників важливе значення має дотримання вимог безпеки життєдіяльності та охорони праці. Основні ризики під час виконання роботи пов'язані з тривалою роботою за комп'ютером, зоровим і психоемоційним навантаженням, статичним положенням тіла, використанням електричного обладнання та можливістю виникнення аварійних або надзвичайних ситуацій.

Забезпечення безпечних умов праці передбачає правильну організацію робочого місця, дотримання ергономічних вимог, раціональний режим праці та відпочинку, справність комп'ютерної техніки, виконання правил електробезпеки й пожежної безпеки. Дотримання цих вимог сприяє збереженню здоров'я виконавця, підвищенню працездатності, зменшенню ризику помилок під час програмної розробки та забезпеченню безперервності роботи над інформаційною системою.

Отже, питання безпеки життєдіяльності та охорони праці є невід'ємною складовою виконання бакалаврської кваліфікаційної роботи. Їх урахування дозволяє створити безпечні, раціональні та ефективні умови для розробки, тестування й подальшого використання системи підтримки прийняття рішень..

ВИСНОВКИ

У бакалаврській кваліфікаційній роботі було розглянуто задачу розробки системи підтримки прийняття рішень на основі аналітичних показників. Актуальність теми зумовлена потребою організацій у використанні програмних засобів, які дозволяють не лише зберігати й відображати дані, а й виконувати їх аналітичну обробку, формувати узагальнені оцінки та надавати користувачу рекомендації для прийняття обґрунтованих управлінських рішень.

У межах роботи досягнуто поставленої мети, яка полягала у розробці системи підтримки прийняття рішень на основі аналітичних показників. Запропонована система забезпечує введення або імпорт даних, їх зберігання, перевірку, нормалізацію, розрахунок часткових і зважених оцінок, формування інтегральної оцінки, визначення рівня стану об'єкта аналізу, побудову візуалізацій і формування текстової рекомендації для користувача.

У першому розділі досліджено теоретичні основи СППР, проаналізовано предметну область, визначено роль аналітичних показників і сформовано вимоги до системи. Встановлено, що прийняття рішень на основі різнорідних показників потребує формалізованого підходу до нормалізації, зважування та інтерпретації результатів.

У другому розділі виконано проектування системи: обґрунтовано веборієнтовану модульну архітектуру, спроектовано інформаційну модель і структуру бази даних, розроблено алгоритм обробки показників та правила формування рекомендацій. Запропонована модель забезпечує повний цикл аналітичної обробки — від введення даних до отримання інтегральної оцінки.

У третьому розділі обґрунтовано вибір програмних засобів реалізації та описано основні модулі системи. Для створення системи запропоновано стек Python, Flask, SQLite, SQLAlchemy, Pandas, Chart.js, HTML, CSS, JavaScript і Bootstrap. Проведене тестування підтвердило коректність роботи основних функцій: введення даних, нормалізації, розрахунку інтегральної оцінки та формування рекомендації.

У розділі «Безпека життєдіяльності, основи охорони праці» висвітлено особливості безпечної роботи а також вимоги безпеки для з комп'ютерними системами.

Практичне значення розробленої системи полягає в тому, що вона може бути використана як інструмент автоматизованого аналізу показників у структурному підрозділі організації. Система зменшує трудомісткість ручної обробки даних, підвищує прозорість оцінювання, допомагає виявляти проблемні напрями діяльності та забезпечує користувача зрозумілою рекомендацією.

ПЕРЕЛІК ДЖЕРЕЛ

- 1 Keen, P. G. W., & Scott Morton, M. S. (1978). *Decision Support Systems: An Organizational Perspective*. Addison-Wesley.
- 2 Power, D. J. (2002). *Decision Support Systems: Concepts and Resources for Managers*. Quorum Books.
- 3 Shim, J. P., Warkentin, M., Courtney, J. F., Power, D. J., Sharda, R., & Carlsson, C. (2002). Past, present, and future of decision support technology. *Decision Support Systems*, 33(2), 111–126. [https://doi.org/10.1016/S0167-9236\(01\)00139-7](https://doi.org/10.1016/S0167-9236(01)00139-7)
- 4 Arnott, D., & Pervan, G. (2005). A critical analysis of decision support systems research. *Journal of Information Technology*, 20(2), 67–87. <https://doi.org/10.1057/palgrave.jit.2000035>
- 5 Parmenter, D. (2015). *Key Performance Indicators: Developing, Implementing, and Using Winning KPIs* (3rd ed.). Wiley.
- 6 Kerzner, H. (2017). *Project Management Metrics, KPIs, and Dashboards: A Guide to Measuring and Monitoring Project Performance* (3rd ed.). Wiley.
- 7 Sharda, R., Delen, D., & Turban, E. (2020). *Analytics, Data Science, and Artificial Intelligence: Systems for Decision Support* (11th ed.). Pearson.
- 8 Microsoft. *Power BI documentation*. Microsoft Learn. <https://learn.microsoft.com/power-bi/>
- 9 Pressman, R. S., & Maxim, B. R. (2020). *Software Engineering: A Practitioner's Approach* (9th ed.). McGraw-Hill.
- 10 Sommerville, I. (2016). *Software Engineering* (10th ed.). Pearson.
- 11 Fowler, M. (2002). *Patterns of Enterprise Application Architecture*. Addison-Wesley.
- 12 Silberschatz, A., Korth, H. F., & Sudarshan, S. (2019). *Database System Concepts* (7th ed.). McGraw-Hill.
- 13 Python Software Foundation. *Python documentation*. <https://docs.python.org/>

- 14 Pallets. (n.d.). Flask documentation. <https://flask.palletsprojects.com/>
- 15 SQLAlchemy Authors and Contributors. SQLAlchemy documentation. <https://docs.sqlalchemy.org/>
- 16 The pandas development team. pandas documentation. <https://pandas.pydata.org/docs/>
- 17 Chart.js Contributors. Chart.js documentation. <https://www.chartjs.org/docs/>
- 18 Bootstrap Team. Bootstrap documentation. <https://getbootstrap.com/docs/>
- 19 Myers, G. J., Sandler, C., & Badgett, T. (2011). *The Art of Software Testing* (3rd ed.). Wiley.
- 20 SQLite Consortium. SQLite documentation. <https://www.sqlite.org>
- 21 MDN Web Docs Contributors. (n.d.). MDN Web Docs. <https://developer.mozilla.org/>
- 22 McKinney, W. (2022). *Python for Data Analysis: Data Wrangling with pandas, NumPy, and Jupyter* (3rd ed.). O'Reilly Media.
- 23 Tableau. Tableau Desktop and Web Authoring Help.
- 24 Qlik. (n.d.). Qlik Sense documentation. <https://help.qlik.com/>
- 25 Google. Looker Studio Help. Google Help.
- 26 Metabase. Metabase documentation. <https://www.metabase.com/docs/>
- 27 Saaty, T. L. (2008). Decision making with the analytic hierarchy process. *International Journal of Services Sciences*, 1(1), 83–98.
- 28 Portfolio project management / R. Nebesnyi, N. Kunanets, N. Veretennikova, R. Vaskiv, Z. Haladzhun, M. Graca. *ITPM*. 2024. P. 141–152.
- 29 Шимчук, Г. В., Назаревич, О. Б., Литвиненко, Я. В., Готович, В. А., Никитюк, В. В., Боднарчук, І. О. (2025). Грід-системи та технології хмарних обчислень.
- 30 Duda, O., Mykytyshyn, A., Mytnyk, M., & Stanko, A. Information technology sets formation and "TNTU Smart Campus" services network support. *Proceedings of the 3rd International Workshop on ITTAP 2023*, Ternopil, Ukraine, Opole, Poland, November 22–24, 2023.

ДОДАТКИ

Фрагменти програмного коду системи

Модель аналітичного показника

```
class Indicator(db.Model):
    __tablename__ = "indicators"
    id = db.Column(db.Integer, primary_key=True)
    name = db.Column(db.String(120), nullable=False)
    description = db.Column(db.Text)
    unit = db.Column(db.String(30), nullable=False)
    min_value = db.Column(db.Float, nullable=False)
    max_value = db.Column(db.Float, nullable=False)
    optimization_type = db.Column(db.String(10), nullable=False)
    weight = db.Column(db.Float, nullable=False)
    is_active = db.Column(db.Boolean, default=True)
```

Функція нормалізації показника

```
def normalize_value(actual, min_value, max_value,
                    optimization_type):
    if max_value == min_value:
        raise ValueError("Максимальне значення не може
дорівнювати мінімальному")
    if optimization_type == "max":
        normalized = (actual - min_value) / (max_value -
min_value)
    elif optimization_type == "min":
        normalized = (max_value - actual) / (max_value -
min_value)
    else:
        raise ValueError("Невідомий тип оптимізації показника")
    return max(0, min(1, normalized))
```

Розрахунок інтегральної оцінки

```
def calculate_integral_score(indicator_values):
    total_score = 0
    result_details = []
    for value in indicator_values:
        indicator = value.indicator
        normalized = normalize_value(
            actual=value.actual_value,
            min_value=indicator.min_value,
            max_value=indicator.max_value,
            optimization_type=indicator.optimization_type
        )
        weighted = normalized * indicator.weight
        total_score += weighted
    return round(total_score, 4), result_details
```

Структура бази даних системи

```
CREATE TABLE indicators (  
    id INTEGER PRIMARY KEY AUTOINCREMENT,  
    name VARCHAR(120) NOT NULL,  
    description TEXT,  
    unit VARCHAR(30) NOT NULL,  
    min_value REAL NOT NULL,  
    max_value REAL NOT NULL,  
    optimization_type VARCHAR(10) NOT NULL,  
    weight REAL NOT NULL,  
    is_active BOOLEAN DEFAULT 1  
);  
  
CREATE TABLE analysis_sessions (  
    id INTEGER PRIMARY KEY AUTOINCREMENT,  
    user_id INTEGER NOT NULL,  
    title VARCHAR(150) NOT NULL,  
    period VARCHAR(50),  
    created_at DATETIME DEFAULT CURRENT_TIMESTAMP,  
    comment TEXT  
);  
  
CREATE TABLE indicator_values (  
    id INTEGER PRIMARY KEY AUTOINCREMENT,  
    session_id INTEGER NOT NULL,  
    indicator_id INTEGER NOT NULL,  
    actual_value REAL NOT NULL,  
    created_at DATETIME DEFAULT CURRENT_TIMESTAMP  
);  
  
CREATE TABLE recommendations (  
    id INTEGER PRIMARY KEY AUTOINCREMENT,  
    session_id INTEGER NOT NULL,  
    integral_score REAL NOT NULL,  
    state_level VARCHAR(50) NOT NULL,  
    recommendation_text TEXT NOT NULL,  
    created_at DATETIME DEFAULT CURRENT_TIMESTAMP  
);
```

Тестові дані для перевірки роботи системи

№	Показник	Мін.	Макс.	Факт.	Напрям	Вага
1	Рівень виконання плану, %	0	100	82	Макс.	0,20
2	Продуктивність роботи, ум. од.	40	110	96	Макс.	0,15
3	Якість виконання завдань, %	0	100	88	Макс.	0,20
4	Своєчасність виконання, %	0	100	76	Макс.	0,15
5	Рівень витрат ресурсів, ум. од.	50	150	95	Мін.	0,10
6	Кількість помилок або відхилень	0	20	5	Мін.	0,10
7	Рівень задоволеності користувачів	0	10	8	Макс.	0,05
8	Динаміка результативності, %	-20	20	6	Макс.	0,05

Опис макетів інтерфейсу користувача

Головна інформаційна панель системи

Система підтримки прийняття рішень

Іваненко І. І.

Інтегральна оцінка

0,7765
Рівень: **Високий**

Низький Середній Високий Дуже високий

Рекомендація

Інтегральна оцінка має високий рівень. Стан об'єкта оцінювання є задовільним. Рекомендується підтримувати поточні показники та зосередити увагу на подальшому покращенні показників з нижчим внеском.

Аналітичні показники

№	Показник	Фактичне значення	Нормалізоване	Зважене	Вага, %
1	Фінансова стійкість	0,65	0,7222	0,1444	20
2	Рентабельність	0,48	0,5333	0,1067	20
3	Ліквідність	0,80	0,8889	0,1778	20
4	Ділова активність	0,60	0,6667	0,1333	20
5	Платоспроможність	0,90	1,0000	0,2000	20
Разом		-	-	0,7765	100

Нормалізовані значення показників

Внесок показників у підсумкову оцінку

Показник	Внесок
Фінансова стійкість	20,0%
Рентабельність	20,0%
Ліквідність	20,0%
Ділова активність	20,0%
Платоспроможність	20,0%
Разом	100%

Динаміка інтегральної оцінки

Сторінка введення даних

Система підтримки прийняття рішень

Іваненко І. І.

Нова аналітична сесія

Введіть дані показників для оцінювання діяльності об'єкта

Назва аналітичної сесії *
Оцінювання діяльності підрозділу за I квартал 2025

Період оцінювання *
I квартал 2025

Введення значень показників

№	Показник	Одиниця виміру	Значення
1	Рівень виконання плану, %	%	92,5
2	Продуктивність роботи, ум. од.	ум. од.	1,18
3	Якість виконання завдань, %	%	88,0
4	Свочасність виконання, %	%	90,3
5	Рівень витрат ресурсів, ум. од.	ум. од.	0,72
6	Кількість помилок або відхилень	шт.	
7	Рівень задоволеності користувачів	бал (1-10)	8,4
8	Динаміка результативності, %	%	7,2

Помилка: поле "Кількість помилок або відхилень" не може бути порожнім

Перевірити дані | Запустити розрахунок | Очистити форму

Інформація про сесію

Статус сесії: Чернетка
Створено: 12.05.2025 09:15
Оновлено: 12.05.2025 09:15
Активних показників: 8
Заповнено показників: 7 / 8

Підказки

- Усі показники мають бути заповнені для успішного розрахунку.
- Використовуйте крапку для десяткових значень (наприклад, 92.5).
- Допустимі значення відповідають встановленим діапазонам показників.

Детальніше про показники →

Перевірка даних

Діапазони значень: OK
Обов'язкові поля: Помилка
Типи даних: OK
Логічна узгодженість: OK