

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя
(повне найменування вищого навчального закладу)

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

Кафедра комп'ютерних наук
(повна назва кафедри)

ПОЯСНЮВАЛЬНА ЗАПИСКА

до дипломного проекту (роботи)

Магістр

(освітній рівень)

на тему: Проект програмної системи для потокового обміну даними через
інтеграцію мереж мобільного зв'язку та WAN

Виконав: студент (ка) VI курсу, групи СНмн-61

напряму підготовки (спеціальності) 122

Комп'ютерні науки (освітньо-наукова програма)

(шифр і назва напряму підготовки, спеціальності)

Бугай В.П.

(підпис)

(прізвище та ініціали)

Керівник

(підпис)

Харченко О.Г.

(прізвище та ініціали)

Нормоконтроль

(підпис)

Мацюк О.В.

(прізвище та ініціали)

Рецензент

(підпис)

Загородна Н.В.

(прізвище та ініціали)

м. Тернопіль – 2020

АНОТАЦІЯ

Проект програмної системи для потокового обміну даними через інтеграцію мереж мобільного зв'язку та WAN // Дипломна робота освітнього рівня «Магістр» // Бугай Віктор Петрович// Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем та програмної інженерії, кафедра комп'ютерних наук, група СНмн – 61 // Тернопіль, 2020 // с. – 121, рис. – 19, табл.– 10, слайдів – 8, додат. – 4, бібліогр. – 24 .

Ключові слова: ГЛОБАЛЬНА МЕРЕЖА, ТРАНСЛЮВАННЯ, МОБІЛЬНИЙ ПРИСТРІЙ, ПОТІК, ПРОГРАМНИЙ ІНТЕРФЕЙС, КОМУНІКАЦІЯ, ПРОТОКОЛ, ПЕРЕДАЧА ДАНИХ.

Метою дипломної роботи є аналіз комунікаційних можливостей глобальних мереж та створення програмного комплексу з метою реалізації технології інтеграції мобільних мереж. Ціллю набуття навичок програмування мережевих додатків за допомогою програмних інтерфейсів і дослідження можливостей комунікації та передачі даних в мережі.

Об'єктом дослідження є глобальна мережева Інтернет, та її використання для комунікаційної взаємодії з іншими мережами. Предметом дослідження виступає методологія використання та передачі даних за допомогою мереж та програмних можливостей. Методи, які використовуються в даній роботі – програмні мережеві інтерфейси, які реалізуються за допомогою мови програмування високого рівня Java.

ЗМІСТ

ВСТУП.....	8
РОЗДІЛ 1 ІНТЕГРАЦІЯ ТЕХНОЛОГІЙ МОБІЛЬНИХ КОМІНІКАЦІЙ ГЛОБАЛЬНИМИ МЕРЕЖАМИ.....	3 11
1.1 Огляд мережевих технологій комунікації.....	11
1.2 Актуальність мобільних мереж для сучасного рівня вимог до комунікацій.....	16
РОЗДІЛ 2 МЕТОДИ ОРГАНІЗАЦІЇ ІНТЕГРУВАННЯ СИСТЕМ ЗВ'ЯЗКУ НА БАЗІ МЕРЕЖЕВИХ, МОБІЛЬНИХ ТА ПРОГРАМНИХ МОЖЛИВОСТЕЙ.....	28
2.1 Загальна структура та можливості технологій організації комунікаційних зв'язків.....	28
2.2 Особливості використання комунікаційних технологій в середовищі Android.....	34
2.3 Мережеві комунікаційні можливості та методи їх взаємодії.....	39
2.4 Вибір програмних продуктів та засобів.....	48
РОЗДІЛ 3 ПРОГРАМНА РЕАЛІЗАЦІЯ ТЕХНОЛОГІЇ ЗА ДОПОМОГОЮ КЛІЄНТ-СЕРВЕРНОЇ ВЗАЄМОДІЇ ЧЕРЕЗ ГЛОБАЛЬНУ МЕРЕЖУ.....	58
3.1 Програмування інтерфейсів системних та апаратних компонентів....	58
3.2 Реалізація потокового з'єднання.....	64
3.3 Розробка мережевої структури.....	67
3.4 Програмний комплекс для інтеграції мобільного зв'язку з мережею Інтернет.....	72
РОЗДІЛ 4 ОСНОВИ РОЗРОБКИ ЗАСТОСУНКІВ ПІД ANDROID.....	76
4.1 Установка середовища розробки Android Studio.....	77

4.2 Створення першого Android проекту	79
4.3 Запуск програми на емуляторі.....	82
4.4 Структура проекту Android.....	84
4.5 Внесення змін в програмний код та дизайн	85
РОЗІДЛ 5 Обґрунтування економічної ефективності	87
5.1 Визначення стадій технологічного процесу та загальної тривалості проведення НДР	87
5.2 Визначення витрат на оплату праці та відрахувань на соціальні заходи.....	88
5.3 Розрахунок матеріальних витрат.....	90
5.4 Розрахунок витрат на електроенергію	91
5.5 Розрахунок суми амортизаційних відрахувань.....	92
5.6 Обчислення накладних витрат.....	93
5.7 Складання кошторису витрат та визначення собівартості НДР	93
5.8 Розрахунок ціни проекту	94
5.9 Визначення економічної ефективності і терміну окупності капітальних вкладень.....	95
РОЗІДЛ 6 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ	97
6.1 Функціональні заходи у сфері державного регулювання та контролю захисту населення і територій.....	97
6.2 Підвищення стійкості роботи комп'ютеризованих систем в умовах дії ЕМІ ядерних вибухів.....	100
6.3 Загальні вимоги законодавства з охорони праці в галузі інформаційних технологій.....	105
6.4 Аналіз умов праці розробника програмного забезпечення	106

6.4.1 Загальна характеристика умов праці	106
6.4.2 Повітряне середовище	107
6.4.3 Освітлення	109
РОЗІДЛ 7 ЕКОЛОГІЯ	113
7.1 Радіоактивне забруднення довкілля та його моніторинг	113
7.2 Статистичне групування в екології	115
ВИСНОВКИ	118
ПЕРЕЛІК ДЖЕРЕЛ	120
ДОДАТКИ	

ВСТУП

Актуальність теми дипломної роботи характеризується бурхливим розвитком комунікаційних можливостей глобальних мереж. Оскільки спілкування в глобальній мережі стрімко розвивається, а мобільні мережі не втрачають актуальність та утримуються на ринку, постає завдання в об'єднанні мереж для створення можливостей зручної комунікації в одному середовищі. Збільшується використання мережі зі стеком протоколів TCP / IP та UDP. Розвиваються нові технології та мобільні пристрої та виникає потреба в комунікації цих пристроїв з різною архітектурою і операційними системами. Постійно реалізуються і з'являються нові ідеї, це тягне за собою появу нових можливостей мереж і тому новизною системи є використання новітніх програмних методів та алгоритмів мережевих технологій для створення нових середовищ комунікації, які матимуть змогу функціонувати на персональному комп'ютері або сервері та на мобільному пристрої, не залежачи від типу мережі. Важливим є питання абонентського доступу до комунікаційних ресурсів і дослідження їх взаємодії.

Метою дипломної роботи є аналіз комунікаційних можливостей глобальних мереж та створення програмного комплексу з метою реалізації технології інтеграції мобільних мереж, дослідження її основних аспектів між пристроями з різною архітектурою, набуття навичок з мережевого програмування, а саме організації мережевих протоколів передачі потокових даних, створення архітектури клієнт-серверних додатків та використання мережевих інтерфейсів на мовах програмування високого рівня. Завданням технології є розвиток комунікаційних зв'язків в мережі та інтеграція різних комунікаційних середовищ. Економічним значенням дослідження є збільшення якості, зручності та збільшення масштабів комунікаційних зв'язків підприємства в цілому.

Об'єктом дослідження є глобальна мережева Інтернет, та її використання для комунікаційної взаємодії з іншими мережами. Тобто, створення

комунікаційного середовища в мережі Інтернет за допомогою інтеграції в неї мобільних мереж, які за допомогою мережевих та програмних можливостей, в комплексі використовуються для спілкування користувачів.

Предметом дослідження є мережева методологія використання та передачі даних, які за допомогою глобальних мереж отримуються зі спеціальних комунікаційних пристроїв або з абонентських пристроїв для побутового спілкування. Об'єкт і предмет дослідження використовуються як сукупність і джерело інформації та ресурсів абонентів, сучасних підприємств, провідних компаній та державних установ.

Методологія дослідження представляє собою програмний комплекс в вигляді клієнт-серверного додатку, розробленого на мові високого рівня Java. Головними інструментами є мережеві можливості зв'язку персональних комп'ютерів та мобільних пристроїв, програмна взаємодія між різними операційними системами такими як: Android та Windows. Використовуються середовища програмування Eclipse та плагін Android Development Tools. Реалізуються програмні інтерфейси сокети, потоки вводу, виводу, перетворення типів даних та використання мережевого з'єднання за допомогою протоколу передачі.

Інформаційним забезпеченням проекту є наукова література про мережеві технології, розвиток програмних засобів, проблеми використання глобальних мереж, організації протоколів передачі. Також використовується учбова література про можливості мов програмування, наукові статі з реалізаціями класів і методів мови Java, дані про розвиток і тестування сучасних кросплатформних додатків та відгуки про проблеми сучасного використання глобальних мереж в комунікації.

Практична значимість проекту підштовхує і виражає подальший розвиток мережевих технологій та можливість глобальної комунікації в різних мережах. Використання програмного комплексу можливе в цілях об'єднання безлічі комунікаційних пристроїв мобільних мереж в один сегмент мережі Інтернет. Також головною з можливостей є переадресація дзвінків з мобільного пристрою

на зручній пристрій користувача. Головною перспективою є об'єднання взаємодії користувачів для можливостей спілкування ведення бізнесу та інших побутових цілей.

Дипломна робота складається з трьох основних та чотирьох додаткових розділів. У першому розділі розглянуті досягнення та проблеми мобільних мереж з точки зору комунікаційних можливостей на фоні розвитку глобальних мереж. Представлена теоретична стратегія вирішення безлічі проблем в спілкуванні між мережами.

Другий розділ описує можливості мережевих технологій в створенні комунікаційних зв'язків та передачі даних. Також досліджується взаємодія мобільних пристроїв з глобальною мережею в операційній системі Android.

У третьому розділі представлений один із напрямків реалізації технології на прикладі програмного комплексу, методологія його функціонування та архітектури.

Додаткові розділи містять матеріал відповідно до завдань консультантів з цих розділів.

РОЗДІЛ 1

ІНТЕГРАЦІЯ ТЕХНОЛОГІЙ МОБІЛЬНИХ КОМІНІКАЦІЙ З ГЛОБАЛЬНИМИ МЕРЕЖАМИ

1.1 Огляд мережевих технологій комунікації

На сьогодні Інтернет, як всесвітня мережа, виступає в якості сучасного засобу зв'язку. Покриття мереж з кожним роком росте і технології для більш швидкої комунікації також. В Україні на сьогодні, за даними Київського міжнародного інституту соціології, на 2016 рік 62% дорослого населення України користується Інтернетом. Частка користувачів 18-39 років досягла 91% і продовжує рости надалі. Рівень зростання актуальності мережі в процентах відносно років в Україні наведено на рис. 1.1.

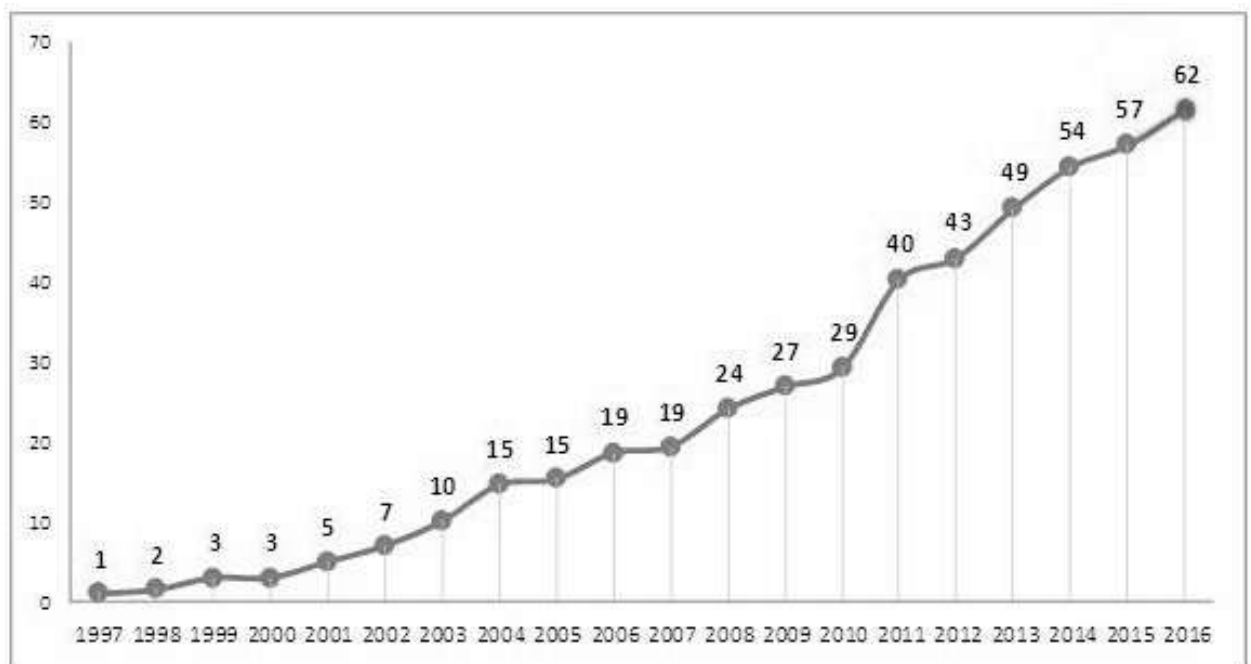


Рисунок 1.1 – Відносний графік розвитку відсотку користувачів мережі Інтернет дорослого населення України до років

За останніми даними статистики провідних аналітичних компаній в світі налічується близько двохсот мільйонів комп'ютерів і велика їх частина об'єднана

в різноманітні інформаційні мережі, серед яких найбільше поширення отримав Інтернет, який на сьогоднішній день є одним із найшвидших засобів зв'язку і дає безмежні можливості в спілкуванні по всьому світу та відкриває більш розвинені можливості надалі[8].

Така популярність глобальних мереж, а саме Інтернет, обумовлюється багатьма причинами, наприклад прагнення до прискорення передачі інформації між абонентами, можливість приймати електронну пошту і спілкуватися за допомогою Інтернет пейджерів, оприлюднювати певну інформацію для соціуму і нарешті, користуватися сучасною телекомунікаційною системою, яка дозволяє людям з допомогою спеціального IP-телефону спілкуватися за мінімальними тарифами з усім світом.

А саме головне, що на сьогодні не має меж комунікаційних можливостей в мережі Інтернет, все лиш впирається в технології, які функціонують на сьогодні і обладнання. Інтернет на сьогодні представляє собою всесвітню комп'ютерну мережу, яка доступна користувачам всього світу. Кількість користувачів з кожним роком зростає практично на сто відсотків, і ця цифра зайвий раз підтверджує слова фахівців ІТ індустрії, які говорять про те, що в найближчому майбутньому на землі не залишиться людей, позбавлених можливості користуватися всесвітньою павутиною і безліч технологій будуть взаємодіяти з нею.

Ще зовсім недавно мережа використовувалася лише для отримання електронної пошти та передачі, отримання невеликих файлів[2], але стрімкий розвиток сучасних технологій зробив Інтернет справжнім інформаційним гігантом, який поступово замінює нам інші джерела інформації та методи комунікації, які ще два десятиліття тому здавалися непорушними. Великим досягненням є те, що всі без винятку, користувачі мережі Інтернет мають можливість використовувати її інформаційні ресурси, які щодня поповнюються і оновлюються, тим самим дозволяють нам отримувати найактуальніші відомості по будь-якого питання і передавати їх іншим.

Крім того, всевітня павутина багатьом замінила звичний телефонний зв'язок, який в більшості випадків невиправдано дорогий і небезпечний для тих, хто під час розмов оперує важливою інформацією[8]. Розвиток IP-телефонії вже зараз дозволяє здійснювати дзвінки в будь-яку точку світу за досить демократичними тарифами. Крім цього, такий вид зв'язку дозволяє нам робити абсолютно безкоштовні відеодзвінки, що, звичайно ж, просто неможливо при використанні звичайного виду зв'язку, який, схоже, починає переборювати себе в тому вигляді, в якому він представлений сьогодні і адаптується під нові пристрої.

При бурхливому розвитку Інтернет технологій, можливість отримувати інформацію де завгодно змінили життя всього суспільства і основні аспекти ведення бізнесу зокрема, що дозволяє використовувати комунікаційні можливості глобальних мереж, як в побуті, так і в комерційній діяльності. З кожним днем з'являються все нові, і нові можливості як для кожної людини зокрема, так і для всіх сфер діяльності. Сьогодні вже нікого не здивуєш тим, що за квитками в кіно, театр або на концерт не потрібно стояти в черзі до каси, а можна купити їх по Інтернету.

Сучасні Інтернет технології змінили докорінно життя суспільства. Трохи більше десятиліття тому мобільні телефони були рідкістю і доступні не багатьом людям. Але вже сьогодні ніхто не уявляє собі життя без мобільного зв'язку. Без комп'ютерів і Інтернету неможлива навчання в школі і ведення великого бізнесу. Життя в сучасному місті тече в високому ритмі, для того щоб все встигати, потрібно робити все одночасно. Поряд з цим більшість сучасних людей проводять багато часу в Інтернеті, спілкуються в соціальних мережах, переглядають пошту, відео або просто інформацію, викладену на сайтах.

Сучасні пристрої дають можливість робити це в будь-якому місці, якщо з'явилася вільна хвилина. При цьому вони компактні і з легкістю поміщаються в сумочці або навіть кишені. Кожне покоління портативних пристроїв, таких як планшет і смартфон, є все більш функціональним. За допомогою цих пристроїв можна користуватися багатьма перевагами, які надає світова павутина.

Гаджети, які оснащені потужними процесорами, помітно розширюють функціональність пристроїв. Їм є величезна різноманітність сервісів, служб і додатків. Для багатьох наприклад, важливою функцією є якісна відео- та фотозйомка, для оприлюднення певних даних.

Ця можливість так полюбилася багатьом користувачам, що викликала справжній ажіотаж навколо відеосервісів. Сьогодні тільки за допомогою планшета або смартфона можна зняти фільм, відредагувати його і надати для перегляду глядачам в мережі. Такі чудові можливості пробудили вражаючий ріст популярності сучасних гаджетів.

Великі компанії і маленькі фірми повсюдно використовують у своїй роботі нові Інтернет технології. Наявність власного сайту необхідно для ведення сучасного бізнесу. Причому краще, якщо це буде не простий сайт-візитка, а повноцінний портал, який володіє різноманітною інформацією і можливістю інтерактивного зв'язку з клієнтами.

Розвиток Інтернет технологій дозволяє сучасним користувачам мережеских ресурсів пропускати через себе величезну кількість інформації. Основи Інтернет технологій призвели до зміни структури економіки, а розвиток телекомунікаційних та інформаційних технологій послужили сприятливим середовищем для виникнення електронного бізнесу, що базується на принципі мережевої економіки [1].

Це підприємницька діяльність, яка здійснюється за допомогою цифрових зв'язків і здійснюється тільки в Інтернет мережах. Від індустріальної економіки, яка носить національний характер, мережева економіка відрізняється глобальними масштабами, що значно розширює коло замовників, постачальників, партнерів і конкурентів. На даний час підприємницька діяльність в мережі стала доступною всім. Наприклад, прямі продажі через Інтернет-магазини без участі посередників являють собою нову модель ведення бізнесу завдяки реалізації новим можливостям комунікації. Розвиток інформаційної індустрії залежить від безлічі обставин, ключовими є обсяг капіталовкладень в телекомунікації.

Глобальні комп'ютерні мережі (WAN), які належать до створених за територіальним принципом комп'ютерних мереж, призначені для надання інформаційних послуг великій кількості користувачів, що розміщені на цій території на постійній чи тимчасовій основі. Без заперечень найвідомішою WAN сьогодні є всесвітня мережа Інтернет. Типово, глобальні мережі є публічними (громадськими). Інколи WAN будуються на приватній основі, як мережі крупних корпорацій. Компанії, які підтримують роботу мереж, – це оператори мережі. Компанії, котрі забезпечують надання послуг з використанням комунікаційних мереж, називаються сервіс-провайдерами.

У комп'ютерних мережах для обміну даними використовуються наступні види комутації:

- комутація каналів (при передачі аудіоінформації по телефонних лініях зв'язку);
- комутація повідомлень (для пересилання повідомлень електронної пошти);
- комутація пакетів (для обміну даними, в т.ч. і для обміну мультимедіа).

Інтернет об'єднує різні локальні комп'ютерні мережі. Для пересилання повідомлень між ними використовуються канали громадських телекомунікацій. Майже всі сервіси в мережі Інтернет реалізовані на основі клієнт-серверної архітектури. Міжсерверна комунікація здійснюється по магістралях, до яких належать виділені лінії провідного зв'язку, оптичні канали, канали супутникового зв'язку.

Сервіси мережі Інтернет [10]:

- електронна пошта;
- комп'ютерна телефонія;
- обмін файлами FTP;
- віддалений доступ через термінали;
- довідкові та пошукові сервіси.

Перелік послуг і розвиток можливостей створюють певне підґрунтя для функціонування та розвитку нових технологій для задоволення потреб комунікації користувачів. На сьогодні мережа Інтернет починає затьмарювати інші види комунікації, завдяки свої багатогранності і універсальності. Гнучкість технологій дозволяє використовувати технології Інтернет для задоволення унікальних потреб різних компаній, наприклад створення певних конференцій з по заданих географічних масштабах або спираючись на певні пристрої. Весь цей розвиток змушує інші технології комунікації адаптуватися і об'єднуватися з глобальною мережею.

1.2 Актуальність мобільних мереж для сучасного рівня вимог до комунікацій

Серед багатьох мереж мобільного зв'язку на сьогодні найпоширенішою є стільникова мережа. Територія, покрита такою мережею, поділена на області (соти), в кожній з яких зв'язок забезпечує окрема базова станція. Безперервність зв'язку забезпечується частковим перекриттям чарунок. Комутуюче обладнання реалізує визначення локації абонента та безперервність зв'язку при переміщенні його між зонами різних базових станцій [1].

Тенденції розвитку мобільної телефонії по при збільшенню інших технологій займають значне місце на ринку комунікацій. Завдяки угодам операторів, та методам об'єднань з іншими мережами мобільна телефонія не тільки установилась, а і доволі обґрунтовано розвивається. Цьому також сприяє стрімкий розвиток мереж. Компанія J'son&PartnersConsulting оприлюднила результати дослідження світового ринку інфраструктури мобільного зв'язку. За підсумками третього кварталу 2013 року число активних SIM-карт в світовому масштабі склало 6,6 млрд. За прогнозами, до 2017 року їх кількість перевищить 8,4 млрд.

Триває швидкий розвиток мереж третього (3G) і четвертого (4G / LTE) покоління: очікується, що до кінця 2017 року світові доходи від мобільної обміну

даними досягнуть \$ 200 млрд. При цьому мобільний трафік зростає значно швидше виручки операторів, а вартість передачі одиниці об'єму даних постійно знижується. У 2015-2017 рр. очікується комерціалізація гетерогенних мереж, які дадуть можливість плавного (для абонента) роумінгу в мережі і управління мобільністю між макростільниками, малими сотами і точками доступу Wi-Fi. Вітчизняні оператори продовжують активно інвестувати в інфраструктуру мобільного зв'язку, в тому числі в розгортання мереж LTE.

Ще одним каталізатором зростання ринку LTE-обладнання може стати впровадження принципу технологічної нейтральності. Зараз принцип технологічної нейтральності діє відносно частот в діапазонах 890-915 МГц та 935-960 МГц, які можуть бути задіяні телекомунікаційними компаніями для розгортання 3G-сервісів, а також смуг в діапазонах 1710-1885 МГц і 1805-1880 МГц, що виділяються під 4G .

В цілому, за прогнозами, глобальні витрати операторів на бездротову інфраструктуру будуть залишатися відносно стабільними до 2017-го; пік витрат припадає на 2013-2015 рр. – Близько \$ 53 млрд. Найвищі темпи зростання ринку інфраструктури бездротових мереж між 2012 і 2017 рр. очікуються в Азіатсько-Тихоокеанському регіоні.

В даний час телефон це чи не головна пристрій, за допомогою якого людина кожен день. Необхідність в телефоні може виникнути в будь-який момент: на роботі, вдома, в кафе і т.д. Найчастіше потрібно зателефонувати в інше місто або країну, а це коштує чималих грошей. Особливо ця проблема актуальна для великої компанії, яка постійно на зв'язку з іноземними партнерами. В цьому випадку витрати на телефонний зв'язок можуть становити тисячі доларів на місяць. І тут конкурентом мобільної телефонії стає IP-телефонія.

Інтернет-телефонія працює не за такою схемою, як стандартна. У другому випадку телефонна станція виступає головним пристроєм зв'язку, а голосові сигнали передаються через телефонну лінію. Головні відмінності традиційної мережі – невелика пропускну здатність і виділена лінія. Мова, як відомо, є

аналоговим сигналом, тому одна гілка підходить для обмеженої кількості з'єднань. Тому для досягнення безперебійної зв'язку, потрібне створення великої кількості гілок, що відчутно позначається на витратах. Механізм роботи Інтернет-телефонії досить відрізняється. Інтернет тут виступає головним джерелом зв'язку. З'єднання реалізується за допомогою перетворення голосів абонентів в цифровий сигнал в спеціальних портах. Це означає, що мова певним чином зашифрована і перетворюється в електронні пакети.

Пакети по мережі Інтернет передаються до одержувача, потім вони знову перетворюються в звуковий сигнал і вже в цьому виді сигнал використовується абонентом. Взагалі реалізація цього методу досить трудомістка. Важливо, щоб механізм спрацьовував протягом пари секунд, в іншому випадку розмови в реальному часі вийде. З цієї причини обладнання є дуже дорогим і з цієї причини провайдерів IP-телефонії на вітчизняному ринку дуже мало. Але всі фахівці сходяться в одностайній думці, що IP-телефонія це зв'язок майбутнього. Адже економія у сучасних компаній найчастіше стоїть на першому місці.

Слід сказати, що інтернет-телефонія і IP-телефонія це різні речі. Різниця полягає в наступному. Інтернет-телефонія це лише один із напрямів IP-телефонії. Вона має на увазі використання в якості ліній передач стандартних каналів Інтернет, тобто тих, з якими звичайні Інтернет-користувачі знайомі вже давно. Якщо ж вести мову конкретно про IP-телефонії, то лінією передачі телефонних сигналів є цифрові канали. Перевага таких каналів полягає в тому, що вони можуть забезпечити зв'язок пристойної якості. Але дуже часто можна зіткнутися з простоями, тому що немає необхідності у використанні каналів безперервно і постійно.

З економічної точки зору це часто необґрунтовано, тому що створювати такі канали зв'язку означає вкладати великі гроші. Крім того оплачувати виділений канал під силу не кожній компанії. Зате Інтернет-телефонія користується вже готовою мережею Інтернет, а це означає, що витрати будуть значно нижче, ніж в першому випадку. Що важливо, простої в такій мережі виключені. Але Інтернет-телефонія, звичайно, не позбавлена недоліків – якість

розмови не завжди на гідному рівні, тому що канали всесвітньої мережі знаходяться на різному рівні завантаження, пропускна здатність у них різна, що багато в чому визначається часом доби, цим стандартна телефонія перемагає і в деякі компанії беруть це за основу в використанні.

Щоб оцінити якість зв'язку, вдаються до трьох головних параметрах:

- якість мовного сигналу;
- час очікування;
- кількість електронних пакетів, які не досягли кінцевого користувача.

Все прогресивне і нове швидко підхоплюється компаніями в усьому світі, а старі технології забуваються. Фахівці вважають, що розвиток IP-телефонії буде набирати темпи найближчим часом. Приміром, уже зараз можна скористатися програмним шлюзом для зв'язку між мобільними GSM-мережами та корпоративної АТС вашої компанії. А незабаром, можна буде втілювати глобальний роумінг із застосуванням IP-зв'язку для мобільних операторів. А це означає, що мобільні технології вийдуть на зовсім інший етап розвитку. І варто зазначити, що вартість хвилини розмови по мобільному телефону значно зменшиться.

З іншого боку, оператори мобільного зв'язку втрачають мільйони через те, що абоненти здійснюють все менше дзвінків. Дослідження виявили, що 26% споживачів телекомунікаційних послуг користуватимуться мобільним телефоном для дзвінків в середньому не більше одного разу на тиждень. Решта обсягу спілкування припадає на месенджери.

Аналітики компанії J'son&Partners Consulting, які спеціалізуються на дослідженні ринку комунікаційних послуг підрахували, що в 2014 році втрати багатьох операторів мобільного зв'язку пострадянських країн за рахунок переходу абонентів на Інтернет-сервіси склали 1-2 млрд. доларів (2% річної виручки). У світі сукупні втрати досягли 14 млрд. доларів.

Deloitte прогнозує, надалі здійснювати голосові виклики і обмінюватися повідомленнями в месенджерах користувачі будуть ще активніше. Так, якщо в

2011 році такий спосіб спілкування воліло 11% абонентів, у 2015 році – вже 22%, а в 2016 році їх вже 26%.

У той же час тривалість дзвінків зростає – за рахунок більш дешевих тарифів, пакетів з передоплаченим лімітом розмов та послуг. Так звані смартфони спочатку були розроблені в основному для послуг, не пов'язаних з голосовим спілкуванням. Саме тому частка часу, затраченого на реалізацію цих послуг значно більша. Наприклад у Великобританії та США таке споживання в три рази перевищує звичайне спілкування телефоном. Тобто споживачі користуються соціальними мережами, пишуть повідомлення, обмінюються електронною поштою, ніж дзвонять один одному. Мобільні оператори сьогодні фактично стали каналом для обміну даними і розвивають інфраструктуру, при цьому сервісам та програмному забезпеченню приділяють недостатньо уваги. Через це битву комунікаційних сервісів вони вже програли.

Втім оператори мобільного зв'язку просто так не здаються: телекомунікаційні компанії в усьому світі проспали появу месенджерів, тому що не інвестували вчасно кошти в розробку нових технологій, аналізують аналітичні компанії ринку комунікацій. Компаніям, які хочуть залишатися в лідерах, необхідно інвестувати в розвиток нових технологій, створювати власні месенджери і думати над можливістю брати плату за користування своєю інфраструктурою. І тому оператори створюють аналогічні сервіси. У МТС є месенджер MTS Connect. Крім того, в кінці минулого року декілька операторів об'єдналися в альянс з метою стандартизувати IP-інтерфейси інтернет-сервісів, в тому числі месенджерів, а також щоб привернути увагу користувачів.

Зараз ринок мобільних мереж збагачується великою кількістю можливостей і сервісів. При виборі конкретного оператора всі надані їм послуги потрібно уточнювати, які можуть істотно відрізнитися. У кожній країні є безліч операторів зв'язку, які пропонують послуги и тарифи. Одні працюють в межах своєї держави, інші знайшли вихід на міжнародний ринок. Зазвичай всі намагаються розширити свої діапазони, але деякі працюють в певною цільовою політикою. У абонентів є вибір, кому віддати перевагу.

І в залежності від особистих потреб, враховуючи умови послуг та сервісів, тарифи, можна підібрати найбільш вигідний варіант. Користувачі мережі можуть, проаналізувавши свої витрати на зв'язок, кількість і тривалість здійснених дзвінків, відправлених повідомлень, необхідність використовувати мобільний Інтернет, вибрати відповідний тариф. Оператори в боротьбі за своїх споживачів послуг розробляють найрізноманітніші пропозиції тарифних планів:

- безліміт – оплачується фіксована вартість, час розмов не обмежується;
- безкоштовне спілкування в мережі, певну кількість безкоштовних повідомлень і трафіку інтернету в місяць;
- вигідні тарифи на інших операторів, за кордон;
- з платою за з'єднання і без плати;
- з абонплатою і без неї;
- доплата за вхідні дзвінки і багато інших.

Мобільні мережі мають неалий багаж технологій і стандартів для якісного зв'язку. Це їх головний плюс і його надалі розвивають. Телекомунікаційним стандартом нового п'ятого покоління мереж мобільного зв'язку буде стандарт 5G. Найбільші інвестиції в стандарт п'ятого покоління вкладає компанія "Huawei". Передбачається, що стандарт 5G буде забезпечувати швидкість обміну даними рівню 10 Гбіт / с.

Збільшення швидкості обміну даними для мобільних мереж наступного покоління необхідно, оскільки в майбутньому до мережі будуть підключатися пристрої без втручання людини і кількість абонентів мережі різко зросте. Також в 5G буде реалізована технологія device-to-device (D2D), що дозволить абонентам мережі та їх пристроям спілкуватися між собою безпосередньо, без проходження даних через базову станцію, яка обслуговує дану соту. Така передача даних буде можлива при невеликих Відстанях між абонентами однієї стільниці. Через базову станцію буде проходити тільки сигналізація, для здійснення тарифікації з'єднання.

Розвиток полягає в поступовому поліпшенні технології четвертого покоління мобільних мереж – 4G, яка не дарма називається "довготривалою еволюцією". Першим кроком у розвитку є використання технології LTE Advanced (просунута LTE), що дозволить суттєво підвищити обміну даними: до 10 Гбіт / с. Але на цьому програма розвитку покоління 4G не закінчується: планується створення таких технологій, як LTE for TV Broadcast (для ТВ мовлення), LTE for Car (для автомобілів), LTE for Airplanes (для літаків). В цілому, в ході планування поступового переходу до 5G мереж відбудеться поліпшення таких основних параметрів:

- 1) збільшення ємності мереж;
- 2) збільшення швидкості обміну даними;
- 3) зменшення часу затримки обміну даними.

З поліпшенням технологій також відбудеться зростання пристроїв, які будуть взаємодіяти з мережами. Кількість таких пристроїв збільшиться в десятки і сотні разів у порівнянні з сьогоднішнім днем. Відповідно, в десятки тисяч разів зросте і передається по мережах трафік. Також зросте кількість мереж, які будуть взаємодіяти один з одним, створюючи і організовуючи роботу нових мереж, без втручання людини.

Надалі спроститься архітектура мереж, шляхом переходу від складної й розгалуженої ієрархічної системи до більш простої і оптимізованої "хмарної" ієрархії. Нові тенденції і плани розвитку забезпечують мобільним мережам значні перспективи на світовому ринку комунікаційних послуг і звертають на себе увагу, як на одного зі значних технологій спілкування.

1.3 Методи взаємодії і об'єднання технологій мобільного зв'язку в глобальній мережі для розвитку та інтеграції

Самим загальним методом організації комунікаційного середовища завдяки кількості сервісів і технологій є Інтернет-телефонія, одне з найбільш зручне и перспективних напрямків в області передачі голосу через Глобальну

мережу. Інтернет-телефонія може бути використана для спілкування через сайт, наприклад з відвідувачами Інтернет-магазину. Або для організації конференцій в мережі між декількома користувачами. У нашому випадку веб-додаток використовується для організації телефонних дзвінків. Весь аудіопотік проходить через сервер, який виступає обробляючою одиницею системи, та забезпечує перекодування і подальшу взаємодію з протоколами. Таким чином організація забезпечує безперебійний зв'язок користувачу з абонентом міського або мобільного телефону.

Головним завданням методики інтеграції в глобальній мережі є використання даної технології як незалежного методу, каналу, джерела чи ресурсу. В мережі Інтернет досить високі можливості для зберігання якості передачі аудіо потоків з мобільних мереж. Увага приділяється організації зручності та меншої громіздкості систем, для задоволення потреб користувачів.

Для інтеграції даних з мережі GSM, потрібно розглянути її структуру та особливості. Мережа складається з декількох функціональних одиниць, чії функції і інтерфейси специфіковані. Вона може бути розділена на 3 крупних частини: мобільний телефон, який знаходиться у абонента, базова станція, яка контролює радіозв'язок з мобільним телефоном, і мережева підсистема, основною частиною якої є центр комутації, що виконує комутацію викликів між мобільними абонентами або між мобільними абонентами і абонентами фіксованих мереж. Центр комутації також здійснює управління при переміщенні абонентів.

Мобільний телефон повинен містити термінал (безпосередньо телефонного апарату) і смарт-карти SIM. SIM забезпечує персональну мобільність абонента, а саме те, що абонент має доступ до послуг зв'язку незалежно від терміналу. Вставивши SIM-карту в інший термінал стандарту GSM, абонент здатний приймати і робити дзвінки з цього терміналу, користуватися всіма іншими послугами зв'язку.

Підсистема базової станції складається з двох частин: з базової станції, що представляє собою приймально-передавач, і контролера базової станції. Базова

станція визначає соту і взаємодіє відповідно до протоколів радіозв'язку з мобільним телефоном. У крупних перенаселених районах, де буде використовуватися потенційно велика кількість базових станцій, неодмінними вимогами є висока продуктивність, надійність, портативність і мінімальна ціна.

Головним джерелом, потрібним для використання в глобальній мережі є центральна мережева підсистема. Основним її модулем є центр комутації. Він діє аналогічно вузлу комутації інших мереж, і додатково забезпечує функціональність, необхідну для взаємодії з мобільним абонентом: реєстрацію, перевірку справжності, оновлення розташування, хендовери і маршрутизацію викликів абоненту в роумінгу.

Ці послуги забезпечуються в сполученні з іншими функціональними одиницями, все разом утворюють мережеву підсистему. Оскільки радіоефір – це обмежений ресурс, що розділяється усіма абонентами, повинен бути визначений алгоритм розподілу частотних смуг серед багатьох абонентів. Такий метод, обраний GSM, являє собою поєднання множинний доступ із тимчасовим і частотним поділом. Така система включає ділення по частотах смуги шириною в 25 МГц на 124 смуги з несучими частотами, розділеними 200 кГц. Одна або більше несучих частот прив'язані до кожної базової станції.

Завданням методики інтегрування є використання потокової інформації з іншої мережі. Розроблена система повинна відповідати вимогам і можливостям з'єднання та обробки даних із каналу GSM. Розглянувши загальну структуру мобільних мереж, можна зробити висновок, що на сьогодні можливості мережевих технологій повністю задовольняють та мають змогу використовувати більш старі та в дечому досконаліші канали передачі речової інформації. Це відбувається завдяки розвитку нових протоколів обміну даними, та пристроїв в яких і розроблюються технології доступу до потоків зв'язку.

В глобальній мережі можливо реалізувати з'єднання з іншими потоками в мережі за допомогою декількох архітектур. З точки зору розробника, два варіанти реалізації передачі аудіо (з ретрансляцією через сервер і без) не сильно відрізняються. В обох випадках потрібен зовнішній сервер. Однак, його роль

можна зменшити і використовували лиш при встановленні з'єднання. Весь голосовий трафік буде йти безпосередньо від клієнта до клієнта.

Однією з архітектур є мережі типу peer-to-peer (або P2P), які працюють згідно принципу рівноправності усіх вузлів мережі. Тобто кожен вузол одночасно користується послугами інших вузлів і є клієнтом та надає іншим вузлам свої ресурси і є сервером.

Архітектура клієнт-сервер широко застосовується у комунікаційному середовищі. Сервер в цьому випадку бере на себе головну роль структури самої системи, виконуючи роль взаємодії з зовнішнім потоком і взаємодією між клієнтами в цілому. Таким чином є базою для безлічі сервісів всієї системи.

Враховуючи те, що при інтегруванні потоку з мобільної мережі він вже має підтримку серверів на стороні мережі GSM, а тому сегмент серверу в архітектурі P2P самої глобальної мережі можна відкинути, але це потягне за собою свої особливості технології та методу з'єднання. Одна з ключових проблем при реалізації голосового зв'язку «один на один» – сигналізація користувачів про вступників дзвінках. Якщо користувач є ініціатором дзвінка, він знає, в який момент йому ініціювати передачу і прийом голосового трафіку. Що стосується користувача, якому телефонують, потрібно реалізувати спосіб сповіщення його про це. Як вирішувати цю задачу – питання конкретного застосування:

1. Варіант використовувати асинхронні запити, що виконуються періодично. Наприклад, 1 раз в секунду. У відповіді на запит повинна міститися ознака, що є вхідний дзвінок і треба прийняти рішення, чи відповідати на нього. Потім, налаштувати вхідний і вихідний аудіопотоки.

2. Варіант Comet-архітектура, коли клієнт тримає постійне підключення до сервера і отримує відповідь тільки тоді, коли відбулося деяке подія. В даному випадку, вхідний дзвінок.

Обидва варіанти мають на увазі використання основних засобів веб-розробки на стороні сервера. Хоча, в принципі, для цього завдання можна пристосувати і медіа-сервер.

У випадку з голосовим зв'язком досить просто реалізувати схему, яка оптимально взаємодіятиме з зовнішнім потоком даних без його зміну, спираючись на мережевий розвиток [4]. Знову ж, можлива реалізація як через сервер, так і використовуючи P2P. Основна ідея об'єднання полягає в тому, що кожен зі співрозмовників публікує вихідний аудіопотік і підписується на вхідний. Дані передаються з використанням протоколу мережевого рівня і охоплюються.

В технології рівень обробки та доступу до даних реалізуємо за допомогою програмних можливостей передачі так званих поточкових даних. Такий тип послуги надає можливість обмінюватись мультимедійною звуковою та/чи відеоінформацією в режимі реального часу. Це так званий streaming.

Актуального використання такого сервісу набуло внаслідок політики провайдерів в області авторського права, коли користувач не може завантажити музику чи фільм, а лише прослуховувати чи переглядати контент без збереження цього файлу на своєму комп'ютері. Для мобільних мереж використання такої послуги було проблемним, оскільки вимагало наявності широкого швидкісного каналу зв'язку. Такі можливості з'явилися в впровадженні стандарту 3G.

Звичайно, що апаратне забезпечення клієнта повинно мати можливість виконати опрацювання прийнятого потоку даних в режимі реального часу. Мається на увазі буферизація та виправлення помилок при передачі з наступним перенаправленням даних на медіапрогравач. Буферизація згладжує нерівномірності в надходженні пакетів потоку. Ще однією перевагою поточкового відтворення медіаконтенту є відсутність вимог до обсягу постійної пам'яті для зберігання даних, що є дуже важливим для портативних невеликих мобільних пристроїв.

Крім того, потік може надходити з мережі Інтернет, що є важливою складовою для інтегрування. Від мережі оператора стільникового та мобільного зв'язку потрібно лише забезпечити передачу потоку з заданою якістю обслуговування. Для послуги streaming в стандартах 3G і 4G спочатку

передбачаються механізми, які не допускають великі затримки і високий відсоток втрати даних.

Таким чином мережі мобільного зв'язку мають всі ознаки для якісного інтегрування в глобальну мережу[1]. А остання, в свою чергу має потрібні архітектури та технології обробки даних з зовнішніх мереж та реалізує безліч перспектив розвитку на сьогодні.

РОЗДІЛ 2

МЕТОДИ ОРГАНІЗАЦІЇ ІНТЕГРУВАННЯ СИСТЕМ ЗВ'ЯЗКУ НА БАЗІ МЕРЕЖЕВИХ, МОБІЛЬНИХ ТА ПРОГРАМНИХ МОЖЛИВОСТЕЙ

2.1 Загальна структура та можливості технологій організації комунікаційних зв'язків

На сьогодні, технологій передачі HD-відео і звуку досить багато, під HD-відео і звуком мається на увазі не просто файли музики і кіно, але робота програм у реальному часі і високій роздільній здатності для створення можливостей спілкування, взаємодії та обміну даними між користувачами. Одна з найбільш доступних є технологія передачі відео і звуку по Wi-Fi. Поверх Wi-Fi можна використовувати інші безпроводні технології. Але для цього потрібно розуміти загальні переваги і недоліки, обумовлені таким використанням Wi-Fi. При нових потребах використання техніки більшість комп'ютерів (та інших пристроїв) вже оснащені модулями безпроводного зв'язку, тобто вже не потрібен окремий передавач, все, що потрібно для трансляції, вже є.

Головна перевага це простір для розвитку нових можливостей, оскільки використовувати технологію можна не тільки для локального вирішення завдання, в даному випадку розглядаючи комунікаційний простір, а й для отримання доступу до глобальної мережі. Глобальна поширеність звертає на себе увагу найбільших учасників IT-індустрії, як Qualcomm, Intel, Android, Apple, Cavium Networks, використовуючи Wi-Fi можливості і розвиваючи свої технології.

Але існують і недоліки, наприклад використання бездротової передачі при комунікаційному зв'язку займає частину ефіру для доступу в мережу. Для того, щоб потрібний сигнал був переданий в рамках пропускної спроможності безпроводної мережі, його потрібно стиснути відповідним кодеком (як правило, h.264), хоча таке стискання спричиняє незначну втрату якості.

В швидкоплинному ІТ-світі при створенні зв'язку або приєднанні інших ліній між користувачами, містити сервер на своєму ПК і змушувати додатковими, зайвими рухами підключати до нього пристрої-клієнти, зовсім не раціонально в плані ресурсів і зручності використання. Алгоритм інтегрування повинен базуватись на потоковому з'єднанні між клієнтами, головну роботу виконує сервер, який і є своєрідною точкою передачі, але керування підключення повинен брати на себе клієнт. Отже, головна і опорна, для бездротової потокової передачі

структура – розмежування клієнтських і серверних обов'язків. Це дасть змогу використовувати клієнт, як джерело для іншої мережі, або навіть лінії зв'язку. Для функціонування наведеної структури, для отримання даних з пристрою, на сервері повинен функціонувати протокол обміну даними, а клієнт (пристрій який створює з'єднання, або використовує іншу лінію зв'язку – джерело), повинен ініціювати підключення до даного серверу.

Глобальною структурою комунікаційних технологій є організація та створення процесу обміну інформації. Обмін даними здійснюється по каналах передачі, котрі працюють на основі різних фізичних принципів. Це звукові хвилі при безпосередньому спілкуванні, при використанні телефонного зв'язку здійснюється перетворення акустичних коливань у електричний сигнал і навпаки. Обчислювальна техніка використовує кабельні з'єднання на основі мідних провідників, оптоволоконні канали зв'язку, радіоканали та оптичні випромінювання різних частот. Звичайно, що кожна з цих технологій має свої особливості при розробці та використанні.

Загальна схема обміну даними по мережі показана на рисунку 2.1. Тут видно джерело інформації (відправник), канал передачі даних та вузол призначення (отримувач). Вимоги до якості з'єднання визначаються специфікою послуги, яку потребує використати споживач.



Рисунок 2.1 – Загальна схема передачі інформації

При взаємному обміні даними (двосторонній зв'язок) по каналу зв'язку як відправник, так і отримувач мають мати можливість як відправляти дані, так і приймати, і, відповідно, змінювати свою роль під час сеансу зв'язку з передавача на прийом і навпаки. Основною характеристикою каналу зв'язку є його пропускна здатність (ширина каналу – альтернативний термін). Ця характеристика може бути представлена, як обсяг даних, які може канал пропустити за одиницю часу. Інколи цей параметр називають також швидкістю каналу зв'язку.

При розробці різних додатків для комунікаційних з'єднань, часто стоять завдання щодо обміну даними з мобільного додатку на сервер або інший пристрій. Це можуть бути різні файли, такі як фотографія або відео, аудіо, дані по датчикам або ж будь-яка інша інформація. При такому завданні найчастіше будуються архітектури, при яких є певний заздалегідь розмежований клієнт під конкретну типологію з'єднання. Адже необхідно знати, куди ми будемо відправляти дані і хто їх буде приймати. Крім вибору, хто буде сервером, а хто клієнтом, необхідно ще знати (або визначити), в якому вигляді будуть надходити дані і як їх можна інтерпретувати.

Основні інструменти при трансляванні:

- джерело (в даному випадку мобільний пристрій);
- сервер (отримує потік для обробки);
- клієнт (приймає потік джерела і обробляє для сервера).

Щоб сервер міг зв'язуватися з клієнтом, йому потрібно джерело. коли з'єднання з джерелом встановлено, сервер буде приймати і передавати дані

клієнтам, коли вони будуть підключатися. Джерело виконує головну функцію інтегрування потоків зв'язку і при з'єднанні з сервером використовує клієнт як контрольний пункт передачі. Існує і інший напрямок, коли для прийому від клієнта, він повинен використовувати дані джерела, тоді з'єднання з сервером відбувається лиш після ідентифікації клієнта, як завершеного – з'єданого з джерелом.

Вся інформація, розміщена в глобальній мережі, є більш доступною порівняно з іншими засобами масової комунікації, Це пояснюється тим, що доступ до неї має будь-яка людина, котра може підключитись до мережі Інтернет. Ця інформація регулярно оновлюється, на неї в загальному випадку не накладаються обмеження по обсягу, вона може бути візуалізована різними способами. І швидкість, зручність доступу до неї постійно зростає. Слід зауважити, що мережа (локальна чи глобальна) є лише інструментом забезпечення надання послуг, а сам термін "Інтернет" вказує всього лиш на можливий стек технологій, котрі будуть доступні при обміні даними в рамках глобальної мережі. Безпосередньо сам зв'язок (комунікація) здійснюється за допомогою конкретних служб (сервісів), які доступні споживачеві завдяки мережі Інтернет. Найпоширенішими сервісами (службами) є:

- надання з'єднань;
- передача повідомлень і блоків даних;
- організація і управління діалогом партнерів;
- електронна і голосова пошта;
- проведення сеансів;
- зберігання даних;
- відео-сервіс.

Кожна з цих послуг та неперераховані у цьому списку також надаються відповідною службою. Таким чином Інтернет не є засобом масової інформації, а лише платформою для технологій створення каналів зв'язку для передачі та

обміну даними (повідомленнями) різної природи та походження. Основою обміну даними є їх цифрове кодування .

На даний момент існує кілька класифікаційних основ для структурованого опису комунікації в мережі, однак найбільш поширена класифікація базується на кількості осіб, які беруть участь в цьому спілкуванні. Залежно від цього виділяють наступні комунікаційні типи:

- від людини – до комп'ютера;
- від одного (людини) – до одного;
- від багатьох – до одного;
- від одного – до багатьох;
- від багатьох – до багатьох.

Однак це узагальнена класифікація, на практиці їх існує набагато більша кількість, прив'язаних конкретно до певної типології та конкретного завдання. При цьому комунікативні взаємодії в Інтернет-середовищі можуть оцінюватися по самим різним параметрам:

- відповідно до географічної ареалу учасників комунікації (локальні мережі, Інтранет, національний сегмент глобальних телекомунікаційних мереж загального користування і т.д.);
- за типом використовуваних засобів комунікації;
- за тематикою взаємодій;
- за характером інформаційного наповнення (особисті, офіційні, рекламні тощо);
- за формою звернення (персоніфіковані або знеособлені).

Також, в спільнотах міжособистісної взаємодії, комунікації можуть здійснюватися як в монохронічному режимі, при якому доступ до повідомлень мають тільки відправник і обмежена кількість акторів (як приклад можна привести електронною та голосовою поштою, SMS-повідомлення та ін.), так і в поліхронічному режимі. В останньому випадку, доступ до повідомлення отримує необмежене коло користувачів.

Головним інструментом мережевої взаємодії для вирішення завдання комунікації в реальному часі є можливість організації потокової обміну даними. Розглядаючи мережеве комунікаційне середовище на фоні технологій мобільного спілкування потокова передача є основою, як розробки автономних систем зв'язку так і системою інтегрування інших технологій в одну. Обробка поточкових даних є кращою для більшості сценаріїв використання, що мають на увазі безперервне формування нових динамічних даних.

Обробка може бути застосована в більшості галузевих сегментів і випадків використання, що мають на увазі обробку крупних даних. Зазвичай компанії починають з простих завдань, наприклад з'єднання декількох користувачів, а потім обробки даних в комунікаційних групах. Потім ці завдання трансформуються в більш складну обробку, яка відбувається в режимі, близькому до реального часу. З плином часу в цей процес також додаються комплексні поточкові алгоритми обробки подій.

Різним типам завдань комунікації потрібно мати різні особливості процесу з'єднання та обміну інформацією і це завдання вирішується за допомогою використання мережевих протоколів.

Протоколи типу "точко-точка" (P2P) можуть служити для обміну вже записаними даними мультимедіа. Тоді сервер буде менше навантажений, проте вузьким місцем є пропускна здатність мережі між окремими вузлами, котрі здійснюють обмін даними.

Для забезпечення працездатності технології комунікації додатки для прийому-обміну даними реалізують по клієнт-серверної технології, при якій сервер збирає потік і, при отриманні запиту від клієнта, відправляє цей потік йому. Клієнт же отримує дані, обробляє їх і перетворює в звук. Якщо клієнт отримує дані недостатньо швидко, їх відтворення не буде гладким, тобто буде обриватися або завмирати.

Можлива і зворотна ситуація, коли додаток отримує дані швидше, ніж відтворює. Для подолання цієї проблеми додатки, які здійснюють потокову передачу медіа-даних, зберігають отримані пакети даних в пам'яті

(буферизуються), внаслідок чого забезпечується програвання файлу з постійною швидкістю. Компресія даних є компромісом між якістю переданих даних і швидкістю самого комп'ютера, а також швидкістю доступу в Інтернет. При організації обміну даними в масштабі реального часу потрібна наявність спеціального (потокowego) сервера.

2.2 Особливості використання комунікаційних технологій в середовищі Android

Згідно проведеного багаторічного спостереження встановлено, що Android краще себе проявляє в роботі з рядом сервісів порівняно з основним конкурентом від Apple – IOS. Це використання служб Google, веб-серфінг та інші. Android також є відкритою платформою.

Аудіо та телефонія в середовищі Android віртуалізовані в просторі користувача. Загальна ідея такої віртуалізації полягає в реалізації проксі (сервера і клієнта) на рівні апаратно незалежного інтерфейсу в стеці Android. Проксі-сервер розміщується в окремому контейнері з оригінальним програмним стеком оперативної системи для виконання запитів клієнтів на фізичних пристроях. Проксі-клієнт розміщується в кожному контейнері з Android-користувачем.

Типовим сценарієм використання аудіо для комунікації в Android могло б стати прослуховування в одній з ОС і відтворення звуку в іншій, запущеної в іншому. Також потрібно реалізувати взаємодію з сервісом телефонії, щоб під час розмови по телефону аудіо пристрій записував і передавав в стільникову мережу голос користувача і відтворювало одержуваний з мережі голос співрозмовника.

Взаємодію з користувачем виконує апаратний звуковий інтерфейс Audio Hardware Interface, про що нам говорить Guide платформа Android розробника. Крім того, він є самим низькорівневим і простим апаратно незалежним аудіо інтерфейсом.

По суті, він вдає із себе потік для запису звуку на відтворення. Крім звукового потоку, відтвореного Android, в його реалізації можна розвинути для

широкого використання. Є можливість використання інтерфейсів вищого рівня. Наприклад, підмінити частина аудіо фреймворка Android. Але аналіз показав, що аудіо фреймворк значно складніше, ніж Audio Hardware Interface. Тому проксі-клієнт у випадку з використанням аудіо пристрою є реалізацією Audio Hardware Interface.

Основний обов'язок клієнта – відправляти звуковий потік Android пристроя і іншу корисну інформацію до певного сервера, який буде відтворювати звукові потоки всіх Android. Звукові потоки приймають та відтворюються в стислому вигляді, тому їх передача з використанням копіювання в сервер може давати помітні накладні витрати. Їх ми уникаємо завдяки використанню між-контейнерній роздільній пам'яті для передачі звукових потоків. Інформаційні та керуючі повідомлення передаються з використанням механізму між-контейнерній передачі повідомлень.

Головне завдання сервера в середовищі Android – міксувати одержувані від клієнтів звукові потоки і відтворювати результуючий звуковий потік на фізичному аудіопристрої. Щоб забезпечити максимальну переносимість сервера, міксування і відтворення звуку повинні здійснюватися незалежно від пристрою. Інтерфейси аудіофреймворка Android є апаратно незалежним.

Крім того, в аудіо фреймворку вже закладені можливості для міксування звукових потоків. Аудіофреймворк Android не дає доступ до свого мікшеру ззовні, тому для його використання потрібно або впровадитися в аудіофреймворк, або використовувати мікшер неявно. Розглянута архітектура трансляції звуку наведена на рис. 2.2.

Мобільні пристрої, як правило, мають безліч пристроїв виведення звуку. Наприклад, динамік, трубка, навушники, Bluetooth-гарнітура. Запущені ОС одночасно можуть вимагати відтворення своїх звукових потоків на різних пристроях виводу. У деяких випадках їх вимоги будуть несумісні. Наприклад, не має сенсу програвати один звук на великій динаміці, а інший в навушниках або трубці. Також буде дивно, якщо під час дзвінка в трубці буде чути не тільки голос співрозмовника, але і грала до дзвінка музика.

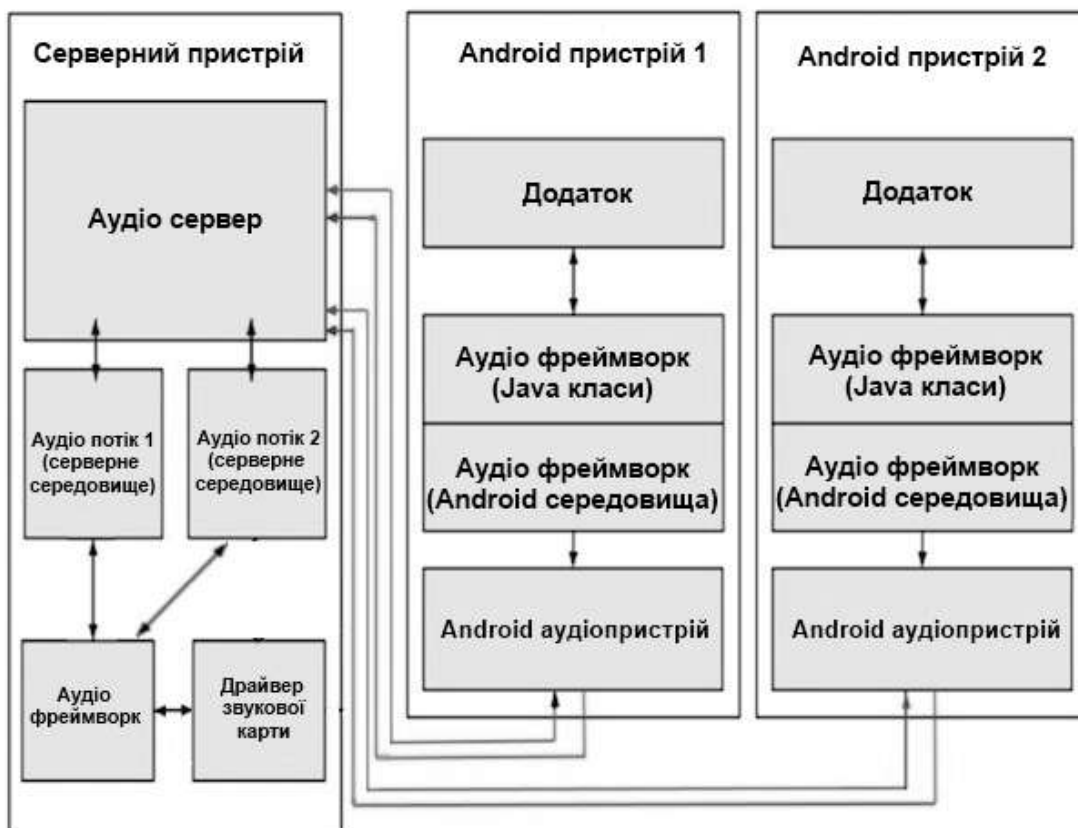


Рисунок 2.2 – Глобальна архітектура процесу трансляції звуку між сервером та Android-пристроями

Тому була розроблена політика роутингу звукових потоків на різні пристрої виведення, що реалізує стратегію «найменшого подиву» користувача. В рамках даної стратегії, зокрема, іноді потрібно імітувати відтворення звуку Android. Для цього клієнт переводиться в режим фіктивного програвання, в якому синхронні функції блокуються на час відтвореного ділянки звукового потоку, завдяки чому Android думає, що відтворив звук, хоча фактично він не використовував сервер для відтворення.

На стороні клієнту, що є реалізацією Audio Hardware Interface, ми можемо отримати інформацію про те, чи йде в даний момент розмова по телефону чи ні. Якщо розмова йде, то потрібно налаштувати аудіо підсистему для отримання і відтворення голосу і взаємодії з сервісом телефонії. Фактично для цього досить перевести реалізацію Audio Hardware Interface, що поставляється з пристроєм, в режим дзвінка за допомогою виклику методу SetMode (MODE_IN_CALL) і

встановити роутинг звуку в трубку, інакше голос співрозмовника буде відтворюватися поточним пристроєм виведення звуку. Таким чином, за реалізацію взаємодії з сервісом телефонії відповідає постачальник пристрою.

Найбільш цікавою і складною виявляється технологія диспетчеризації телефонних дзвінків між контейнерами. Програмний стек управління обладнанням для доступу до мобільних мереж на Android містить наступні компоненти:

1. API пакету `com.android.internal.telephony`.
2. Запити користувача додатків в апаратурі доступу до мобільних мереж.
3. Бібліотека доступу до обладнання.
4. Драйвер GSM-модему.

Також слід відзначити, що реалізація драйверів GSM-модему істотно розрізняється на різних пристроях, тому існування універсального рішення інтегрування цього пристрою в ядрі неможливо. Крім того, бібліотека радіо інтерфейсу використовує недокументований протокол для взаємодії з GSM-модемом, що є перешкодою для його інтегрування навіть для конкретної моделі смартфона.

Серверна частина реалізовується як однопоточковий додаток, який завантажує реалізацію інтерфейсів і за допомогою механізму міжконтейнерної взаємодії приймає повідомлення від клієнтів, що працюють в призначених для користувача контейнерах. Але, сервер повинен обробляти ситуацію, коли з різних клієнтів приходять запити і повинен з іншого боку мати можливість використання великої кількості потоків. Обробка цих потоків повинна проводитись в зборі даних, тобто потоку аудіо і передачі його в головний транспортний потік для відтворення.

Для кожного типу запитів, що надходять на `OnRequest`, в Android визначена політика обробки. В даний час визначені три політики:

- безумовне блокування запиту;
- безумовна передача запиту бібліотеці. політика поширюється на всі запити, що не змінюють стан бібліотеки і GSM-модему;

- передача запиту від активного контейнера бібліотеці. служить для запитів, пов'язаних з відправкою SMS та вчиненням дзвінків.

Для кожного типу асинхронних повідомлень також визначені політики його маршрутизації:

- безумовне блокування повідомлення;
- маршрутизація в активний контейнер. політика встановлена для повідомлень, пов'язаних з прийняттям SMS і дзвінків;
- маршрутизація в усі контейнери. для службових повідомлень «заліза» – наприклад, про зміну рівня сигналу мережі.

Android використовує підсистему EVDEV для збору подій введення ядра. Для кожного процесу, цей драйвер створює чергу подій, що надходять з ядра підсистеми введення.

Доступ до потоків, що інкапсулюють телефонні розмови, програмно відкритий і API платформи Android дозволяють це робити. Проте такий доступ відключений в багатьох пристроях незалежно від використовуваних технологій, що пов'язано з законодавчими обмеженнями. Тобто такий доступ згідно до законів конкретної країни може бути незаконним. Тому виробники повинні дотримуватись законодавства. Є країни, де такий доступ не обмежений на рівні законодавства.

Стосовно нашої держави, то в нас в Конституції прописане право кожної людини збирати будь-яку інформацію. Тобто кожен може записувати власні телефонні розмови, попередивши співрозмовника про запис. А от вже надання цих записів іншим особам може розцінюватись як порушення конституційних прав співрозмовника в частині статті 32 Конституції України. Тому і доступ до цього потоку не відкритий повністю, для уникнення порушення конфіденційності при розробках додатків, які використовують потік співрозмовника в неправомірних цілях.

Доступ до потоків звуку з розмовами здійснюється через спеціальний драйвер. При його відсутності у ядрі операційної системи можливість доступу до розмов відключена. Навіть, коли на пристрої встановлені системні бібліотеки,

вони абсолютно не мають ніякої користі без цього драйвера. Проте, оскільки ОС Android побудована на основі іншої ОС Linux, то користувач може отримати права суперкористувача (рута) на цій системі і скопіювати власне (кастомне) ядро вже з потрібними драйверами та бібліотеками. Тоді, природним чином, можливість запису телефонних розмов з'явиться на такому пристрої. Виробники сильно не схвалюють таку "модернізацію" смартфонів і знімають часто гарантію. Інколи в ядрі драйвер присутній, потрібно його лише включити, отримавши права рута на цьому пристрої.

Якщо на телефоні доступ до лінії через ядро неможливий, то є шанс, що існують спеціальні бібліотеки з підтримкою та отриманням інформації з потоку розмови. Але при розробці більш глобальної технології, з використанням ліцензування додатку та підтвердження його офіційного статусу можливо отримати доступ без сторонніх методів. Така технологія буде зобов'язуватись використовувати дані без порушення закону і по призначенню.

2.3 Мережеві комунікаційні можливості та методи їх взаємодії

Інтеграція мереж володіє величезними потенційними можливостями для комунікації споживача з найрізноманітнішими службами та ресурсами мережі Інтернет. Для абстрагування від особливостей роботи обладнання, ліній передачі даних, апаратних та програмних платформ служить стек протоколів, котрий реалізовано по шаровому принципу. На кожному шарі реалізовано свій функціонал. На кожному шарі повідомлення опрацьовується окремо, до нього добавляється інформація про кожний шар. Потім пакет даних передається на адресу отримувача.

Одними з основних протоколів є мережевий протокол IP. Котрий забезпечує адресацію в мережі та між мережами (в глобальному масштабі). Цей протокол добавляє до кожного пакету адреси відправника та отримувача. Самі ж пакети передаються за допомогою протоколу TCP. Він гарантує цілісність повідомлення та доставку кожного пакету даних з наступним їх складанням у

правильній послідовності. Оскільки ці протоколи тісно пов'язані між собою та з іншими, то стек протоколів дістав назву TCP/IP.

Адреси для цього стеку протоколів мають свій спеціальний формат відповідно до стандарту IP v4 або IP v6. Також кожному мережевому інтерфейсові може присвоюватись доменне ім'я, з коорим людині легше працювати, ніж з числовою адресою.

Використання мережевих технологій необов'язково реалізовується в рамках всесвітньої інформаційної мережі. Технології, що застосовуються в глобальній мережі, можуть масштабуватись на локальні мережі різного обсягу для створення потужних корпоративних інформаційних систем. Це так званий Інтранет.

Інтранет – це комп'ютерна мережа для обміну інформацією, інструментами співпраці, операційними системами та іншими обчислювальними послугами в організації, зазвичай, за винятком доступу сторонніх осіб. Термін використовується на відміну від публічних мереж, таких як Інтернет, але використовує більшість тих же технологій, що базуються на Internet Protocol Suite.

Внутрішня мережа компанії може стати важливим центром внутрішньої комунікації та співпраці та забезпечити єдину вихідну точку доступу до внутрішніх та зовнішніх ресурсів. У своїй найпростішій формі створена інтрамережа з технологіями для локальних мереж (локальних мереж) та широких мереж (WAN). Багато сучасних Інтранет мають пошукові системи, профілі користувачів, блоги, мобільні додатки із сповіщеннями та події, що планують свою інфраструктуру.

Глобальна мережа незалежно чи вона публічна, як Інтернет, чи закрита, як Інтранет, повинна забезпечувати обмін пакетами даних, які на рівні користувача мають найрізноманітнішу природу і тип: файли, мультимедіа, IP-телефонія локальна та міжнародна, банківські дані тощо.

Основні типи даних, якими здійснюється обмін через глобальні мережі, зображено на рис. 2.3.

Голосовий трафік телефонних розмов характеризується невисокою інтенсивністю. Тому вимоги до якості передачі звуку є мінімальними. Отже, навіть найпростіша техніка для реалізації цих вимог справляється зі своїми завданнями.

На сьогодні широко використовується новий термін – Інтранет, котрий застосовується тоді, коли технології корпоративна мережа будується за принципами функціонування мережі Інтернет. Всі ці мережі структурно володіють певними загальними компонентами, функціями та характеристиками. Їх сукупність забезпечує повне використання можливостей для доступу та обробки даних в різних мережах. В тому числі:

- сервери (сервер) – надають свої ресурси мережевим користувачам;
- клієнти (клієнт) – здійснюють доступ до мережевих ресурсів, що надаються сервером;
- середовище – спосіб з'єднання пристроїв в мережі;
- спільно використовувані дані – файли, що надаються серверами по мережі;
- спільно використовувані периферійні пристрої, наприклад принтери, жорсткий диск пам'яті комп'ютера, тощо;
- ресурси, що надаються серверами;
- ресурси – файли, інші елементи, які використовуються в мережі.

Незважаючи на певні подібності, мережі поділяються на два типи:

- однорангові (рівний-рівному);
- на основі сервера (сервер на базі).

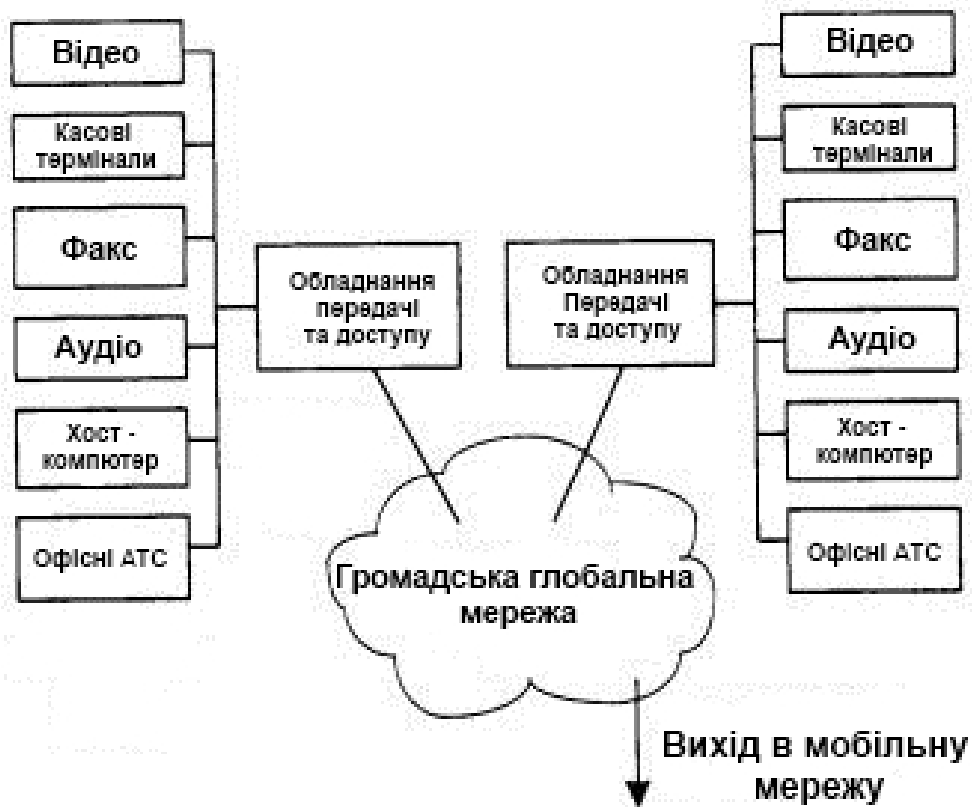


Рисунок 2.3 – Найпоширеніші види інформації/даних для обміну в глобальних комп'ютерних мережах

Відмінності в архітектурі мереж (однорангові чи з виділеним сервером) мають визначальне значення при розробці програмного забезпечення для надання чи споживання сервісів у цих мережах.

З іншого боку вибір типу мережі для підприємства – також процес, котрий керується різними факторами: розміром компанії, вимогами до безпеки, специфіки бізнес-процесів та самих видів діяльності, необхідності оперативного менеджменту, обсягом даних, що планується передавати по мережі, обсягом бюджету, вимогами користувачів/працівників компанії тощо.

Всі ці аспекти потрібно враховувати при розробці технології, яка буде використовувати певні мережеві особливості.

Незалежно від типу мережі та її масштабу доступ до певного сервісу здійснюється клієнтом шляхом постановки запиту та отримання відповіді від

сервера у вигляді затребуваного ресурсу чи повідомлення з кодом помилки відповідно до використовуваного протоколу.

Взаємодія додатків в програмному продукті, побудованому за архітектурою клієнт-сервер, відбувається через мережу. Ця мережа може бути як локальною, так і глобальною. Але незалежно від особливостей з'єднання та топології мережі для клієнта та сервера це з'єднання реалізує принцип прозорості. Це означає, що мережеві протоколи повністю приховують від сервера та клієнта всі особливості мережі і надають каналові зв'язку повну незалежність. Через це програмісти зосереджуються лише на особливостях прикладної області, а не на реалізації кожен раз мережевих протоколів.

На стороні клієнта його програмне забезпечення містить як правило елементи інтерфейсу користувача, що забезпечують виконання операцій вводу/виводу та візуального представлення даних. На серверному рівні виділяють інколи окремо так званий програмний шар (шар прикладної логіки). На цьому рівні розробники реалізують серверне програмне забезпечення, що реалізує особливості бізнес-логіки предметної області.

Цей компонент програмного продукту як правило напряму взаємодіє з ще одним шаром (рівнем) клієнт-серверної частини. Це рівень даних чи сервер баз даних. Часто прикладна логіка та сервер БД фізично розміщені на одному комп'ютері для мінімізації мережевого трафіку.

Залежно від того, на якій програмній платформі розгортається розроблюваний програмний продукт, він може взаємодіяти з організованим доступом до каталогів і файлів у мережі для ОС UNIX/Linux, з системою аутентифікації та авторизації базової ОС, використовувати сервіси домені чи Active Directory для ОС Windows.

Якщо потреби технології потребують рівноправності всіх учасників мережі то слід використовувати однорангові топології мереж. У архітектур однорангових мереж всі її вузли здатні виконувати роль як сервера, так і клієнта. Інколи одночасно. Іншими словами така мережа не містить виділеного окремого сервера.

Оскільки кожен комп'ютер однорангової мережі є одночасно і клієнтом, і сервером, то необхідності в потужному центральному сервері немає. Однорангові мережі зазвичай дешевші в реалізації і проектуванні за мережі на основі сервера. У такій мережі вимоги до ефективності роботи її вузлів та особливо до захисту типowo нижчі, ніж для мереж з виділеним сервером. Виділені для серверів комп'ютери працюють виключно в якості серверів, але не клієнтів або робочих станцій.

Коли абсолютно всі вузли мережі виконують однакові функції або автоматично можуть змінювати набір своїх функцій в залежності від оточуючих умов. Іноді при з'єднанні безлічі користувачів в мережі, наприклад мобільних пристроїв, виникає проблема великої кількості обробка через підвищений енергоспоживання при величезній кількості підключень.

Щоб вирішити цю проблему використовуються такі технології як TCP Relay. Ця технологія дозволяє використати більшість P2P систем, які використовують UDP протокол, з величезною кількістю одночасних підключень, при яких можна вибрати собі вузол який буде виконувати функції отримання запитів з мережі по UDP і пересилання їх на кінцевий пристрій по TCP через одне і теж з'єднання. На сьогодні, подібний механізм уже був реалізований в технології Skype, до його купівлі компанією MS ці функції працювали, пізніше політика Skype змінилась і їх замінюють сервера MS.

Сьогодні все більший інтерес користувачів і організацій пов'язаний з комунікацією та проведенням конференцій, вебінарів, відеоконференцій та відео трансляцій. Тому корисно розглянути можливості таких сервісів. Аудіо, відео трансляції з кожним днем набирають все більшої популярності. Мовлення потокове по своїй природі це доставка мультимедійних потоків інформації (відео, аудіо) віддаленим користувачам, в реальному режимі часу.

Найчастіше користувачі мають можливість задавати питання організаторам онлайн трансляції через звичайний чат. Якщо розглянути саму структуру цієї методики, то це спосіб обміну даними пакетами. Кожен пакет може використовуватись незалежно від інших, тобто до завершення передачі

файлу цілком. Потокове мовлення може створюватися як компаніями, так і звичайними користувачами. В Інтернеті можна знайти досить багато сервісів, які дозволяють будь-якому користувачеві досить легко створити свою власну трансляцію.

На рисунку 2.4 представлена узагальнена комунікаційна схема потокового мовлення та транслявання в глобальних та локальних мережах.



Рисунок 2.4 – Схема потокового мовлення та транслявання в глобальних та локальних мережах

Концепт потокової обміну даними такий: файли піддаються стисканню і розбиваються на пакети, які окремо передаються по мережі. Розмір пакетів залежить від ширини каналу (пропускної здатності) ділянки мережі між клієнтом і сервером. Після наповнення буфера встановленим обсягом даних, клієнтське програмне забезпечення починає їх відтворення та паралельно приймає та опрацьовує наступні пакети. Тобто основна роль буфера полягає у забезпеченні плавного та, основне, безперервного відтворення сигналу. Для цього, виникають значні вимоги до пристроїв, відповідальних за цю частину обробки сигналу.

Розглянемо потокову передачу даних на прикладі трансляції з камери. Для того щоб організувати відправку зображення з камери, необхідно впершу чергу,

захопити і кодувати відео сигнал з камери кодером. Далі кодер відправляє потік на медіа сервер, до якого вже підключаються віддалені користувачі і запитують трансляцію для перегляду. Таким чином для організації онлайн відеотрансляції необхідно три основні компоненти:

- відеокодеки (wirecast telestream, media flash encoder, microsoft expression encoder pro);
- медіасервер (Microsoft IIS Media Services, Wowza Streaming Server, Adobe Flash Server, RealNetworks Helix Universal Server, Red5 (безкоштовний), Erlyvideo (безкоштовний));
- плеєр (відеокодеки: H.264/263, VP6; плеєр VLC, HTML5 відеоплеєр, аудіокодеки).

На практиці результати роботи додатків для потокової комунікації залежать від швидкодії пристроїв і від швидкості підключення до мережі, тому якість звуку / відео – це завжди компроміс. Інтенсивність потоку (кількість даних) прямо впливає на якість його відтворення.

Характеристики потокового мовлення:

- протоколи обміну даними (RTMP, RTSP / RTP, HTTP, MPEG-TS (UDP));
- дозвіл аудіо, відео тощо.
- кодек;
- затримка відтворення;
- бітрейт потоку.

Бітрейт потоку може бути один для всіх користувачів, наприклад, 500Кбіт / сек, але краще використовувати технології мультибітрейтного або адаптивного мовлення, коли бітрейт буде змінюватися в залежності від різних умов або пропускної здатності мережі. На сьогоднішній день майже всі нові сервери підтримують технології перемикання бітрейтів. Розмір потоку можна дізнатися у властивостях файлу, однак багато кодеків використовують

динамічно зміну чий бітрейт, тому навіть вказаному значенню іноді не слід вірити.

Для використання доступні такі варіанти відтворення віддалених медіа-файлів:

- комп'ютер може бути здатний працювати мережевими файлами. Вони будуть доступні, як правило, через SMB, що працює на верхніх шарах стеку протоколів TCP/IP;

- для відтворення служить медіа-сервер з підтримкою відповідних протоколів.

В програмному середовищі механізмом, який реалізує передачу даних в мережі використовуємо мережевий інтерфейс сокетів. Він є механізмом обміну даними між процесами, при цьому ці процеси можуть знаходитися на різних пристроях, які з'єднані в одну мережу.

Створюючи модуль, потрібно описати логіку підключення. І відразу додати в нього поля для імені сервера і номера порту, а також самого сокету, через який ми буде виконуватися спілкування між пристроями. Для мінімальної роботи необхідно хоча б три методи: відкрити з'єднання, відправити дані і закрити з'єднання. При відкритті з'єднання, потрібно переконатися, що раніше не було відкрито сокет, щоб не засмічувати ресурси, і якщо він відкритий – закрити його. Кожен процес може створити і слухати сокет та прив'язати його до встановленого порту ОС.

Процес, який слухає певний порт, типово знаходиться в циклі очікування і «прокидається» при появі нового з'єднання. Програмне створення об'єкту сокету означає представлення комунікаційного вузла доступним для підключень зовні з мережі. Для ідентифікації сокету його необхідно ініціалізувати іменем в рамках поточного мережевого домену. Крім того всі канали обміну даним володіють властивостями із ідентифікаторами (адресами) джерела та одержувача даних, а також такими параметрами, як використовувані протоколи, адреси в локальній мережі, ідентифікаторами локальних процесів, адресою та ідентифікатором віддаленого процесу.

Потокові сокети забезпечують надійність своєї двосторонньої системою комунікації. Якщо відправляємо в сокет два елементи в порядку "1, 2", вони і "співрозмовнику" прийдуть в тому ж порядку – "1, 2". Крім того, забезпечується захист від помилок. Сокетна передача використовує протокол під назвою "Протокол управління передачею", інакше – "TCP". TCP гарантує, що ваші дані передаються послідовно і без помилок.

Тип дейтаграмних сокетів, призначених для обміну даними у вигляді окремих повідомлень (дейтаграмм), також використовують IP для роутинга, але не використовують TCP; вони використовують User Datagram Protocol. У порівнянні з поточковим сокетом, обмін даними відбувається швидше, але є ненадійним: повідомлення можуть губитися в дорозі, дублюватися і змінювати порядок. UDP як правило використовується або там, де стек TCP недоступний, або там, де один-другий пропущений пакет не призводить до суттєвих порушень. Дейтаграмний сокет допускає передачу повідомлення кільком одержувачам, використовуючи багато адресну передачу і трансляцію, передачу мовлення.

2.4 Вибір програмних продуктів та засобів

Ознайомившись з мережевими технологіями, для повного аналізу, розробки та дослідження технології було потрібно її реалізувати в сучасному комунікаційному середовищі. Саме на стороні клієнтської частини користувачі, в середовищі мобільних мереж, була вибрана платформа Android. Android – це операційна система для цілого ряду різноманітних пристроїв, в т.ч. смартфонів, планшетів, телевізорів, мультимедіа програвачів та інших пристроїв. Вона має певні особливості і прогнозовано прогресує в багатьох технологіях. Обійти всіх інших конкурентів у Android є всі шанси, оскільки темпи його розвитку доволі високі. У кожній новій версії з'являються все нові функції і можливості.

Android є відкритою і безкоштовною платформою, тому користується зараз великою популярністю у виробників електроніки. Платформа легко модифікується під потреби конкретного виробника. Володіючи гарною якістю,

універсальністю і сумісністю з різними пристроями, Android платформа має відмінні перспективи на ринку.

Додатки для ОС Android є фактично програмами на мові Java, але для спеціальної віртуальної Java машини. Формат установочних файлів (пакетів) –

*.apk. Для роботи над додатками є ряд бібліотек:

- Bionic (бібліотека стандартних функцій);
- мультимедійні бібліотеки на базі PacketVideo OpenCORE;
- SGL (рушій двомірної графіки);
- OpenGL ES 1.0 ES 2.0 (рушій тривимірної графіки);
- поверхневий Manager (забезпечує для додатків доступ до 2D / 3D);
- WebKit (рушій для веб-браузера; обробляє HTML, JavaScript);
- FreeType (рушій роботи зі шрифтами);
- SQLite (портативна СКБД);
- SSL.

Застосунки Android на додачу до прав ОС Linux повинні відповідати додатковим правилам:

- контент-провайдери – обмін даними між додатками;
- менеджер ресурсів – доступ до таких ресурсів, як файли xml, png, jpeg;
- повідомлення Manager – доступ до контролю стану;
- керівник напрямку – управління активними додатками.

В якості мови для розробки програмного продукту була обрана мова програмування Java, яка реалізує об'єктно-орієнтовану парадигму. Одна із сильних сторін Java полягає в безпроблемній мережевий роботі. Дизайнери мережевої бібліотеки Java зробили її досить простий для читання і запису файлів, за винятком випадку, коли "файл" існує на віддаленому сервері і віддалена машина може вирішувати що їй робити з інформацією, яку ми запитуємо або відправляємо.

Програмна модель, яку ми використовуємо, фактично, це обгортка мережевого з'єднання ("сокет") з об'єктом потоку, так що в кінцевому рахунку використовуємо ті ж виклики методів, які ми використовуємо для інших потоків. Крім того, вбудована багатопоточність Java виключно зручна, коли маємо справу з такою мережевий можливістю, як обробка безлічі з'єднань одночасно. Мова Java підтримує стек протоколів TCP/IP за рахунок розширення підтримки стандартного інтерфейсу потоків вводу/виводу. Також добавляються можливості для побудови об'єктів вводу/виводу при роботі в мережі.

Також підтримується адресація в Інтернет за допомогою класу `InetAddress`. Для адресації в мережі використовуються сервісні функції, що працюють зі звичайними символічними іменами. Ці функції на основі виклику сервісу DNS перетворюють символічні імена в 32-бітові IP-адреси.

Як інтерфейс мережевого з'єднання в мові Java застосуємо технологію обміну даними через сокети. Сокети можна використовувати для комунікації програм, розміщених на будь-яких вузлах Інтернет.

Об'єкти класу `Socket`, на відміну від класу `DatagramSocket`, реалізують високонадійні канали, що з'єднують клієнт із сервером. Відразу після створення об'єкта класу `Socket` як правило встановлюється з'єднання між клієнтом і сервером для обміну повідомленнями. Для створення сокетів існує ряд різних конструкторів:

- `Socket (String хост, Int-порт)` встановлює з'єднання між локальним вузлом та віддаленим сервісом на основі доменного імені і порту для віддаленого вузла.

- `Socket (InetAddress адреса, порт Int)` робить те саме, але тут використовується числове значення IP-адреси, яка передається в об'єкт типу `InetAddress`.

З об'єкта `Socket` в будь-який час можна отримати інформацію про IP-адресу та номер порту, до якого створений сокет підключений. Для цього можна скористатись одним з наступних методів:

- `getInetAddress` повертає IP-адресу у вигляді об'єкта типу `InetAddress`;

- `getPort` повертає номер порта для підключеного віддаленого вузла.
- `getLocalPort` повертає значення порта на локальному вузлі.

Після того, як об'єкт `Socket` створений, з його допомогою отримують доступ до потоків вводу/виводу, асоційованих з цим портом. Ці потоки нічим не відрізняються від звичайних потоків вводу/виводу. Методи, що повертають ідентифікатори для роботи з такими потоками, ми розглянемо далі:

- `getInputStream` – повертає посилання на потік вводу.
- `getOutputStream` – повертає посилання на потік виводу.
- `close` – закриває всі потоки вводу/виводу, асоційовані з сокетом.

Мова `Java` також має підтримку сокетів сервера. Для створення серверів в Інтернет в програмах створюють об'єкти типу `ServerSocket`. Цей об'єкт реєструє себе в системі і стає готовий для виконання запитів підключення від клієнтів. Один з методів цього класу блокує підпроцес до тих пір, поки який-небудь клієнт не встановить з'єднання з відповідного порту.

Після того, як з'єднання встановлено, метод приймає дані і повертає викликавши звичайний підпроцес. Два конструктора класу `ServerSocket` дозволяють задати, за яким портом ми хочемо з'єднуватися з клієнтами, і (необов'язковий параметр) як довго потрібно чекати, поки цей порт не звільниться.

Стосовно можливих видів сокетів, то їх можна застосувати два. Це залежно від протоколу (`TCP` чи `UDP`) будуть потокові та дейтаграмні сокети. Все залежить від ключових аспектів і напрямку технології. З використанням потокових сокетів розробниками створюються канали обміну повідомленнями між двома додатками з можливістю потокового обміну даними. Кожен з них, природно, працює у вигляді окремого потоку, які обговорювались вище.

Принцип організації обміну повідомленнями між працюючими програмами на мові `Java` на основі потокових сокетів реалізовано по повній аналогії роботи з потоками, як із звичайними файлами. Для підключення декількох процесів до одного сервера ми можемо створювати в серверному додатку окремі потоки для кожного клієнтського додатка або застосувати

дейтаграмні сокети. Як було розглянуто в попередньому пункті, саме дейтаграмні сокети на основі протоколу UDP дозволяють реалізувати передачу даних відразу від усіх вузлів. Така функція використовується рідко і тому часто заблокована з міркувань безпеки.

В такому випадку кожен клієнтський процес може виконувати обмін даними з декількома серверними процесами і навпаки. Природа дейтаграмних сокетів така, що вони не гарантують доставку переданих. Поточкові сокети таку гарантію надають. Це ніяк не означає, що одні з них гірші, а інші кращі. Просто природа обміну повідомленнями без встановлення віртуального каналу і з встановленням такого каналу різні. Отже, застосування поточкових сокетів надає гарантоване та безперебійне транслявання мультимедійного контенту при наявності фізичного з'єднання між вузлами мережі.

Після підключення сокету до сервера ми можемо почати отримувати вхідні і вихідні потоки на сервер повідомлень. Вхідні потоки використовуються для зчитування даних з сервера. Вихідні ж використовуються для запису даних на сервер.

Поточкові дані в нашому випадку формуються великою кількістю джерел, які зазвичай відправляють записи даних одночасно і невеликими обсягами. Ці дані повинні бути оброблені послідовно або по кожній із частин, або на безперервній основі тимчасового вікна, після чого їх можна використовувати в різних задачах перед представленням користувачу. Інформація, отримана в результаті такого аналізу, дозволяє компаніям розібратися в багатьох аспектах своєї діяльності, наприклад у використанні сервісів (для задач обліку / виставлення рахунків), активності серверів, навігації по веб-сайтам, геолокації пристроїв, людей або товарів, і в результаті швидко реагувати на мінливі умови.

Наприклад, компанії можуть відслідковувати зміни суспільного настрою щодо своїх торгових марок і продуктів за рахунок постійного аналізу потоків даних, одержуваних з соціальних медіа, і в разі необхідності забезпечувати своєчасну реакцію. В нашому ж випадку ми будемо працювати з живим

комунікаційним потоком, який в свою чергу використовується для задоволення потреб користувачі або ж в певних професійних цілях.

Обробка поточкових даних є кращою для більшості сценаріїв використання, що мають на увазі безперервне формування нових динамічних даних. Обробка поточкових даних може бути застосована в більшості галузевих сегментів і випадків використання, що мають на увазі обробку крупних даних. Зазвичай використовуючи такий принцип компанії починають з простих завдань, наприклад зі збору даних системних журналів, або з елементарних обчислень, наприклад з поновлення мінімумів і максимумів.

Потім ці завдання трансформуються в більш складну обробку, яка відбувається в режимі, близькому до реального часу. В результаті цих програм доводиться виконувати більш складні форми аналізу даних, наприклад застосовувати алгоритми машинного навчання і досягати глибшого розуміння ситуації на підставі даних. З плином часу в цей процес також додаються комплексні поточкові алгоритми обробки.

Процес обробки даних вимагає використання двох рівнів: рівня сховища і рівня обробки. Рівень сховища повинен підтримувати черговість записів і сувору несуперечливість для забезпечення швидких, економічних і відтворюваних операцій запису і читання крупних потоків даних. Рівень обробки відповідає за споживання даних, розташованих на рівні сховища, виконання обчислень з використанням цих даних і повідомлення рівня сховища про те, які дані можна видалити за непотрібністю. Крім того, необхідно передбачити масштабованість, надійність даних і відмовостійкість як на рівні сховища, так і на рівні обробки. В результаті з'явилося безліч платформ, які надають необхідну інфраструктуру для створення додатків обробки поточкових даних.

Здійснення введення – виведення за допомогою потоків має свої плюси, тому що потік приховує всі деталі низькорівневих процесів, що відбуваються з даними безпосередньо в пристрої вводу/виводу.

Основні абстрактні класи, від яких успадковують всі класи байтового вводу/виводу – `InputStream` і `OutputStream`. Кожен з цих абстрактних класів має

кілька реальних підкласів, які керують відмінностями між різними пристроями, такими як дискові файли, мережеві підключення і навіть буфери пам'яті. Абстрактні класи `InputStream` і `OutputStream` скасовують кілька ключових методів, які реалізує інші потокові класи. Два найбільш важливих – це зчитування і записи, які, відповідно, читають і пишуть байти даних. Обидва методи оновлені як абстрактні всередині `InputStream` `OutputStream`. У класи – спадкоємців вони перевизначаються.

Безліч операцій обміну даними між сегментами мережі доцільно виконувати по протоколу TCP / IP. У зв'язку з тим що всі взаємодії являють собою пересилання послідовності байт від клієнта до сервера або навпаки, необхідно розробити формат відправки пакетів даних.

Мережева структура технології повинна виражати ключові особливості її принципу роботи. Роль сегментів в ній залежить від того чи іншого завдання, яке виконує користувач подаючи запит, або створюючи з'єднання. Взагалі для розроблюваної технології можна використати як технологію клієнт-сервер, так і однорангову мережеву структуру. Кожна в свою чергу матиме свої особливості в залежності від напрямку використання технології в комунікації. Враховуючи що останнім часом архітектура клієнт-сервер отримала подальший розвиток. З'явилася архітектура, звана P2P або рівний до рівного (однорангова), обидві мають актуальне застосування на сьогодні.

Оскільки при P2P архітектурі все вузли мережі є рівноправними, тобто вони є і комунікаційними клієнтами і серверами одночасно. З будь-якого вузла мережі можна запросити і оновити дані, що зберігаються на будь-якому іншому сегменті мережі. При такому підході розподіл пристроїв на сервери і клієнти стає умовним, однак, оскільки програми-сервери вимагають чималих обчислювальних ресурсів, архітектура рівний до рівного вимагає установки в вузлах пристроїв з великою оперативною і дисковою пам'яттю, або обчислювальною спроможністю. Часто P2P мережі будуються на прикладному рівні стеку протоколів, як надбудови над TCP/IP. Замість TCP може також застосовуватись UDP.

Як і інші мережі, пірингові мережі складаються з окремих вузлів. Але їх особливістю є те, що кожен вузол не обов'язково одночасно взаємодіє зі всіма вузлами цієї мережі, а лише з їх частиною (підмножиною). Це обумовлюється обмеженими обчислювальними ресурсами мережі. Для забезпечення можливості отримати доступ до служб в однорангових мережах, клієнтське ПЗ має реалізовувати один з протоколів P2P. Пори рівноправний статус всіх вузлів пірингових (P2P) мереж кожен з цих вузлів може мати різні можливості та свої окремі функції. Наприклад, в багатьох однорангових мережах допускається наявність виділених серверів для реалізації, наприклад, авторизації користувачів, реплікації особистих даних чи виконання додаткових функцій (хешування).

Але в нашому випадку в зв'язку з інтегруванням зовнішнього потоку даних, який обробляється серверами на стороні клієнтські обов'язки в такому розподілі будуть визначитися лиш прийомом даних та створенням напрямку з'єднання.

Очевидно, що ріст числа підключень клієнтів у клієнт-серверній мережі спричиняє ріст навантажень на серверну частину. Рано чи пізно настає момент, коли обчислювальних потужностей сервера не вистарчає для обслуговування всіх запитів клієнтів. Порівняно з цією ситуацією пірингова мережа має переваги і стає продуктивнішою, оскільки одна і та ж задача внаслідок розпаралелювання може бути виконана швидше на більшій кількості вузлів мережі.

Перевагами однорангових мереж типу P2P є такі властивості:

- самоорганізованість,
- більша стійкість до відмов при відключенні вузлів мережі;
- спільне використання обчислювальних ресурсів;
- більша продуктивність при операціях з файлами за рахунок одночасного доступу до декількох джерел одного і того ж файлу;
- використання максимально всієї смуги каналу зв'язку;
- автоматичне балансування навантаження між вузлами.

У випадку клієнт-серверної архітектури мережевого додатку також є можливість організації розподіленої обробки шляхом поділу виконання операцій

на стороні клієнта (презентаційна логіка, валідація введених даних) та на стороні сервера (основна бізнес-логіка та робота з БД). Винесення невеликих за обсягом але значних по кількості операцій на сторону клієнта суттєво зменшує навантаження на сервер та зменшує також мережевий трафік. Як результат – зростає кількість клієнтів, які можуть одночасно отримати підключення до сервера.

Типово клієнт-серверна архітектура використовується для випадків створення програмних продуктів, коли виконується опрацювання даних, що зберігаються у централізованій БД чи для надання можливості доступу до ресурсів великій кількості користувачів. При цьому на сервері розміщується ядро для обробки і виконуються найбільш складні операції по введенню, зберіганню, модифікації, вилученню та первинної обробки даних.

Так виникає можливість архівування дзвінків, подальшої їх переадресації та інші пристрої, а також управління самим потоком даних. В плані простого використання ця перевага є не досить актуальною, оскільки при прийому потоку даних сервер відповідатиме лише за виконання завдань організацій командного центра користувача, якому напрямлена переадресація з інших його клієнтських пристроїв зв'язку.

Архітектура клієнт-сервер також дозволяє прискорити роботу додатків за рахунок мінімізації обсягу інформації, що передається по мережі. Зазвичай від клієнта до сервера передається запит на обробку даних. Виконання запиту здійснюється на сервері і назад клієнту передаються тільки ті дані, які відповідають критеріям запиту. Тобто, це знизить навантаження на клієнтську частину додатку і тим самим збільшить частку спроможності пристроїв на її використання. Клієнтська частина виконуватиме лиш транзитну функцію, передаючи дані в первинній структурі, а приймати в потрібній за запитом.

Програми-сервери зазвичай розробляються так, щоб максимально повно використовувати можливості конкретної обчислювальної платформи (комп'ютер + ОС) і забезпечити максимальну продуктивність якомога більшого числа одночасно працюючих пристроїв. Такі можливості дадуть змогу збільшити

універсальність технологій, шляхом обробки потоків даних з різних пристроїв і потоків даних різної природи, приймаючи дані, та повертаючи за запитом конкретного типу.

Ще досить важливими перевагами є розподіл функцій комунікаційної системи між декількома вузлами мережі. Це спрощує обслуговування проєктовано комунікаційної системи. Це також досягається шляхом задоволення комунікаційних потреб клієнтів через розподілення та розмежування навантаження на окремі вузли. Адміністрування прав доступу простіше здійснювати, коли засоби управління ними розміщені на сервері системи, Тді на основі аутентифікації клієнтів можна їх авторизувати з наданням відповідних привілеїв.

Тобто можливе створення певної групи клієнтів які підключаються до одного комунікаційного вузла та інших груп, які обробляє інший пристрій. Як було зауважено вище, така архітектура дозволяє об'єднати в одній мережі різні за апаратною та програмною платформами клієнтів. Але існують також деякі недоліки, вирішення яких залежить від особливостей застосовуваної технології. Наприклад, вихід з ладу сервере може спричинити збій в роботі всієї комунікаційної системи. Вирішенням цієї проблеми є підключення декількох серверних додатків на різних пристроях за потребою.

Отже, обидві мережеві архітектури мають, як переваги, так і недоліки. Відкриваючи цим особливості в залежності від середовища комунікації, принципів використання техніки та цілей застосування.

РОЗДІЛ 3

ПРОГРАМНА РЕАЛІЗАЦІЯ ТЕХНОЛОГІЇ ЗА ДОПОМОГОЮ КЛІЄНТ-СЕРВЕРНОЇ ВЗАЄМОДІЇ ЧЕРЕЗ ГЛОБАЛЬНУ МЕРЕЖУ

3.1 Програмування інтерфейсів системних та апаратних компонентів

Визначивши вимоги до технології можемо розробити мережеву архітектуру та програмну реалізацію. В зв'язку з обраними завданнями – реалізувати технологію інтегрування для переадресації мобільних дзвінків в середовищі Android користувачів через приймач персонального комп'ютера на операційній системі Windows. Клієнт-серверна архітектура повністю задовольнятиме вимоги та за своєю структурою дасть змогу повністю реалізувати можливості технології у цьому напрямку. Для прикладної реалізації розглянемо апаратні компоненти, які використовують обидві частини додатків.

Клієнтська частина повинна взаємодіяти з користувачами мобільних пристроїв, а отже функціонувати на обраній операційній системі Android. Додаток повинен містити основні функції доступу до ресурсів:

- доступ та використання мікрофону;
- відтворення звукового потоку;
- доступ і взаємодія з телефонним інтерфейсом;
- доступ та використання мережевих ресурсів.

Для можливості використання мережевих інтерфейсів в додатку потрібно оголосити дозвіл в файлі маніфесту `AndroidManifest.xml`. Файл маніфесту надає основну інформацію про програму системі.

Оголошуємо дозвіл на доступ додатку в мережу, роботу з нею, та доступ до системних додатків Wi-Fi пристрою:

```
android:name="android.permission.INTERNET"  
android:name="android.permission.ACCESS_NETWORK_STATE"  
android:name="android.permission.ACCESS_WIFI_STATE"  
android:name="android.permission.CHANGE_WIFI_STATE"
```

Основний тип даних, навколо яких функціонує програмний комплекс – голосові повідомлення. Загальні вимоги з використанням цих даних є їх відтворення та зчитування. Щоб приймати голосове мовлення користувача серверна частина, тобто головний приймач повинен відтворювати його та взаємодіяти з мікрофонним пристроєм для передачі голосових поточкових повідомлень клієнтам (слухачам). Для відтворення потоку який надходить використовуємо інтерфейс `SourceDataLine` та клас `AudioInputStream`:

```
AudioInputStream  ais = new AudioInputStream(baiss, format,  
receivePacket.getLength());  
SourceDataLine   sourceDataLine      =      (SourceDataLine)  
AudioSystem.getLine(dataLineInfo);  
sourceDataLine.open(format);  
sourceDataLine.start();
```

На сьогодні багато сучасних мобільних пристроїв обладнані безліччю функцій, які полегшують комунікацію та виконують інші потреби користувачів, але не варто забувати їх основне призначення – мобільний зв'язок. Технологія передбачає моніторинг вхідних дзвінків клієнтського додатку, який працюватиме в режимі станції. Моніторинг дзвінків реалізуємо за допомогою доступу до інтерфейсів такого системного телефонного додатку. Для вхідних викликів, використовуємо клас `TelephonyManager`, і його метод `listen`. Метод використовується для реєстрації слухача, який буде отримувати інформацію про

стан виклику з телефонного додатку, і інших подій, пов'язаних з телефонією. В нашому випадку, потрібна тільки інформація про стан викликів. Також, оголошуємо дозвіл в файлі маніфесту:

```
android.permission.READ_PHONE_STATE.
```

Після цього, створюємо клас слухач, успадкований від PhoneStateListener, і пере визначаємо:

```
private class MyCallListener extends PhoneStateListener {
    @Override
    public void onCallStateChanged (int state, String incomingNumber) {
        switch (state) {
            case TelephonyManager.CALL_STATE_RINGING:
                // Потрібна дія
                break;
        }
    }
}
```

Ключовим параметром аргументу state буде CALL_STATE_RINGING – системна константа, яка визначає, що здійснюється вхідний дзвінок. По замовчуванню параметр приймає значення CALL_STATE_OFFHOOK – вказує, що вхідного і вихідного дзвінку в системний додаток телефону не здійснюється. Після цього, за допомогою екземпляра TelephonyManager реєструємо слухача станів:

```
tm = (TelephonyManager) ctx.getSystemService
(Context.TELEPHONY_SERVICE);
tm.listen (callStateListener,
PhoneStateListener.LISTEN_CALL_STATE);
```

В результаті при отриманні вхідного дзвінка, контролююча служба (яка буде розглянута далі) виконує трансляцію цієї події на сервер, зображено на рис.3.1.

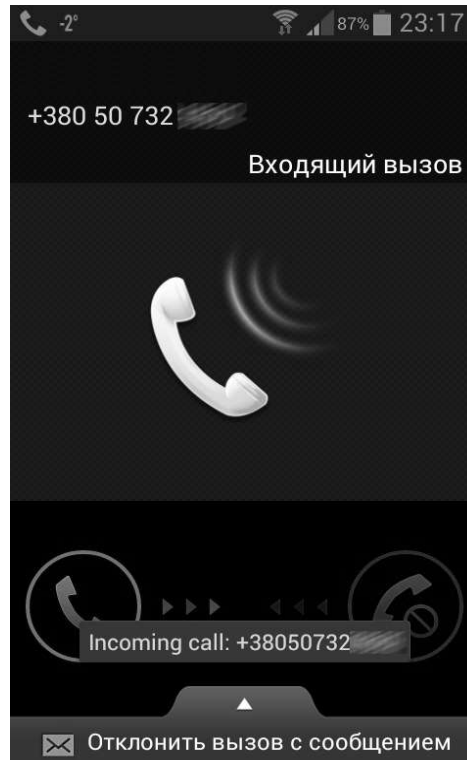


Рисунок 3.1 – Приклад роботи класу MyCallListener

Коли додаток більше не потребує отримання повідомлень, він повинен видалити реєстрацію слухача:

```
tm.listen (callStateListener, PhoneStateListener.LISTEN_NONE);
```

Взаємодія клієнта з сервером відбувається за рахунок передачі та прийому поточкових даних, можливості яких були розглянуті в минулих розділах. Крім аудіо потоків клієнт приймає контрольні команди від сервера для прийому або відхилення дзвінка. Тим самим розроблений додаток повинен взаємодіяти з системним додатком телефону.

Для використання створюємо новий намір (Intent) звернувшись до класу Intent() бібліотеки Java, за допомогою якого визначаємо кнопку керування

відтворенням після натискання на яку, відбувається намір з дією ACTION_MEDIA_BUTTON. Реалізація приймача отримує дію натиснутої кнопки, передану в оповіщенні. Об'єкт Intent включає в себе ключ EXTRA_KEY_EVENT, клас KeyEvent включає в себе список статичних констант KEYCODE_MEDIA_*. Використовуючи створені наміри в залежності від команди прийняття або відхилення дзвінка, приймаємо від сервера і виконуємо наміри з константами подій KeyEvent. ACTION_DOWN – відхилення або прийому KeyEvent.ACTION_UP відповідно:

```
IntentdownCall = newIntent(Intent.ACTION_MEDIA_BUTTON);
downCall.putExtra(Intent.EXTRA_KEY_EVENT,
newKeyEvent(KeyEvent.ACTION_DOWN,
KeyEvent.KEYCODE_HEADSETHOOK));
IntentupCall = newIntent(Intent.ACTION_MEDIA_BUTTON);
upCall.putExtra(Intent.EXTRA_KEY_EVENT,
newKeyEvent(KeyEvent.ACTION_UP,
KeyEvent.KEYCODE_HEADSETHOOK));
```

Весь процес в клієнтському додатку, який функціонує на Androidпристрої повинен виконуватися в фоновому режимі. Передача поточкових даних, сповіщення про вхідний дзвінок, прийом команд сервера та інші функції працюють за допомогою служби або сервісу (Service), в Android працюють як фонові процеси і представлені класом android.app.Service.

Вони не мають призначеного для користувача інтерфейсу і потрібні в тих випадках, коли не потрібно втручання користувача. Використовуємо сервіси, виконуючи мережеві запити до веб-сервера, обробляючи інформацію, запускаючи повідомлення і т.д. Створюємо клас CallTranslationService, який успадковує android.app.Service. Визначаємо методи onCreate і onDestroy – які спрацьовують при створенні і знищенні сервісу.

Сервіс, як і нові, створювані Activity додатку необхідно прописати в маніфесті:

```
<serviceandroid:name=".CallTranslationService" >
```

Приклад роботи служби зображена на рис. 3.2., запуск відбувається за допомогою кнопки Start і надходженню вхідного дзвінка, а знищення сервісу, тобто деактивація додатку за допомогою кнопки Stop.

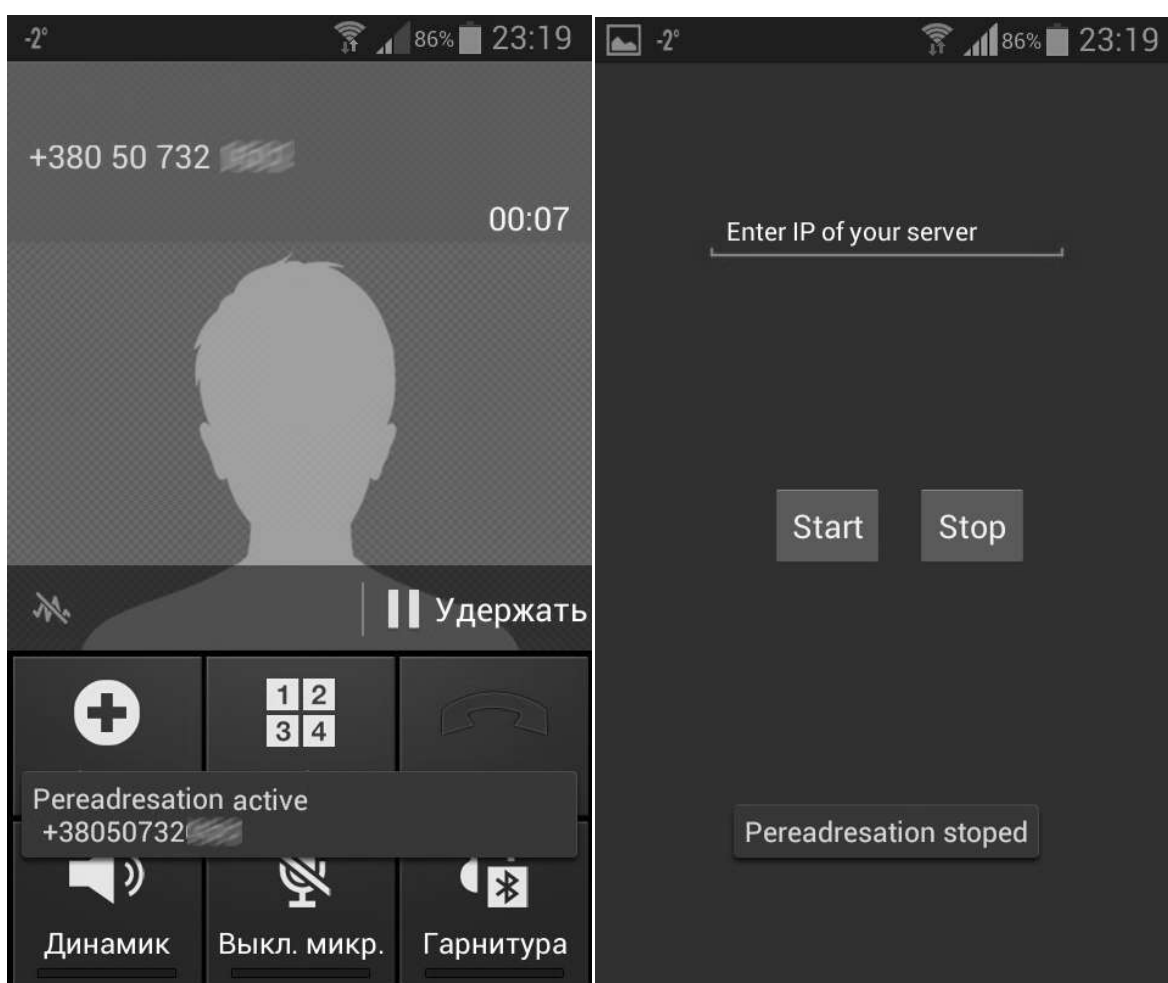


Рисунок 3.2 – Приклад роботи головної служби транслявання при створенні і припиненні переадресації

Керування службою відбувається в головному класі MainActivity.java. За допомогою створення наміру та передачі в нього потрібного нам сервісу – CallTranslationServi і використання методу startService для запуску:

```
Intent intent = new Intent(MainActivity.this,
CallRecordingService.class);
startService(intent);
finish();
або для припинення роботи служби:
stopService(intent);
```

Android дає службам вищий пріоритет, ніж бездіяльним активностям в додатку, тому ймовірність того, що вони будуть завершені через брак ресурсів, помітно зменшується. Використовуючи службу, ми можемо бути впевнені, що додаток буде працювати і реагувати на події, навіть якщо вони в неактивному стані.

Для роботи сервісів не потрібен окремий графічний інтерфейс, як у випадку з Activity, але вони як і раніше виконуються в головному потоці ходу програми. Щоб підвищити якість роботи додатку, потрібно перенести трудомісткі процеси (наприклад, мережеві запити) в фонові потоки, використовуючи класи Thread і AsyncTask. Алгоритм роботи головної служби для транслявання даних представлений у додатку Б.

3.2 Реалізація потокового з'єднання

Створення потокового з'єднання в розробленій технології між клієнтським і серверним додатками має на меті зчитування потокової аудіо інформації співрозмовника, обробка потоку для сприйнятливих умов передачі за допомогою мережі та прийому його для подальшого відтворення. Принцип роботи повинен працювати в двосторонньому напрямку, відмінність полягає в доступі до потоку

співрозмовника, який отримує клієнтський додаток на пристрої – приймачі від сторонніх пристроїв мобільного чи стільникового зв'язку.

Головна проблема доступу до потоку співрозмовника в ОС Android розглянута в минулих розділах. Її вирішенням є підтвердження статусу додатку, як офіційного.

Технологія обміну та транслявання аудіо потоків в розробленому додатку працює за принципом потокового мовлення розглянутою в минулих розділах, але адаптованого для вирішення поставлених завдань комунікації в даному програмному комплексі, який виражає можливий напрям технології інтегрування.

Для отримання поточних даних аудіо інформації для обробки з мікрофона та інших джерел відповідає клас `android.media.AudioRecord`. Для створення об'єкта цього потоку вказуємо необхідні параметри (аргументи):

- джерело звукового потоку `audiosource`;
- частота дискретизації в герцах `samplerate`;
- конфігурацію каналів `channelconfig`;
- кодек, або формат вхідних даних, для обробки `audioformat`;
- розмір внутрішнього буфера з якого зчитуємо аудіо потік `minbufsize`.
- визначаємо джерело для отримання поточних даних. Клас

`mediarecorder` представляє декілька варіантів джерел:

- `MediaRecorder.AudioSource.VOICE_CALL` — джерело голосового виклику всього дзвінка, вхідного і вихідного потоку;
- `MediaRecorder.AudioSource.VOICE_DOWNLINK` — джерело вхідного потоку голосового виклику;
- `MediaRecorder.AudioSource.VOICE_UPLINK` — джерело вихідного потоку голосового виклику;
- `MediaRecorder.AudioSource.MIC` – джерело звукового потоку з пристрою мікрофону.

В нашому випадку використовуємо:

```
MediaRecorder.AudioSource.VOICE_DOWNLINK.
```

Частоту дискретизації встановлюємо 44100 Гц, оскільки вона є загально прийнятою для роботи аудіо даними на багатьох пристроях:

```
intsampleRate = 44100;
```

Конфігурацію каналів може визначити двома параметрами: CHANNEL_IN_MONO або CHANNEL_IN_STEREO. В зв'язку з тим, що перший працює на більшій кількості пристроїв:

```
intchannelConfig = AudioFormat.CHANNEL_IN_MONO;
```

Кодек визначаємо:

```
intaudioFormat = AudioFormat.ENCODING_PCM_16BIT;
```

Розмір буфера потрібно задати, щоб розмір порції зчитування не був більшим чим сам розмір буфера. Для цього отримуємо мінімально допустиме значення за допомогою методу `getMinBufferSize()`. За допомогою цього статичного методу отримуємо мінімальний розмір внутрішнього буфера, при якому об'єкт `AudioRecord` зможе працювати.

Параметри мають той же сенс, що і для конструктора. Слід зауважити, що використання саме цієї величини для запису – не найкраща ідея. Якщо система буде ще чимось зайнята, то програма все одно може не встигати зчитувати всі дані поспіль. Визначмо розмір буфера задавши необхідні аргументи:


```
int minBufSize = AudioRecord.getMinBufferSize(sampleRate,
channelConfig, audioFormat);
byte[] buffer = new byte[minBufSize];
```

Після визначення всіх необхідних параметрів і створення об'єкту потрібного аудіо потоку, необхідно упакувати його для передачі по мережі на сервер.

Для цього зчитуємо створений об'єкт в буфер і упаковуємо в мережевий інтерфейс:

```
recorder = new AudioRecord(MediaRecorder.AudioSource.MIC,
sampleRate, channelConfig, audioFormat,
minBufSize);
recorder.startRecording();
minBufSize = recorder2.read(buffer, 0, buffer.length);
```

3.3 Розробка мережевої структури

Розглянута архітектура організації мережі клієнт – сервер для даного додатку вважається більш сприйнятливою ніж однорангова. Оскільки розробка додатку в такій організації дає змогу розглянути та дослідити технологію інтегрування в напрямку переадресування ліній мобільного зв'язку.

Отже, розмежуємо обов'язки клієнтського і серверного додатку. Клієнтський додаток виступає у ролі приймача для вхідного дзвінка, взаємодіє з зовнішніми пристроями приймаючи телефонний дзвінок. Після прийому створюється з'єднання і додаток вже виступає вузлом між вхідним потоком з зовні і сервером.

Основні функції клієнтської частини додатку в розробленій архітектурі:

- моніторинг вхідних дзвінків;
- створення запиту до сервера;
- прийом команд від сервера;

- реакція на вхідний дзвінок;
- трансляція аудіо потоку;
- прийом аудіо потоку.

Серверна частина свою чергу виконує роль загального приймача всієї системи. Після приєднання клієнта, серверний додаток реагує на вхідний дзвінок та відсилає певні команди клієнту, останній створює трансляцію і завданням сервера вже є забезпечення комунікаційного зв'язку для переадресованого потоку. Функції серверної частини:

- моніторинг запитів клієнта;
- прийом даних про вхідний дзвінок;
- відсилання управляючих команд клієнту;
- відтворення аудіо потоку;
- трансляція аудіо потоку.

Загальна схема функціонуючих модулів представлена на рисунку 3.3.

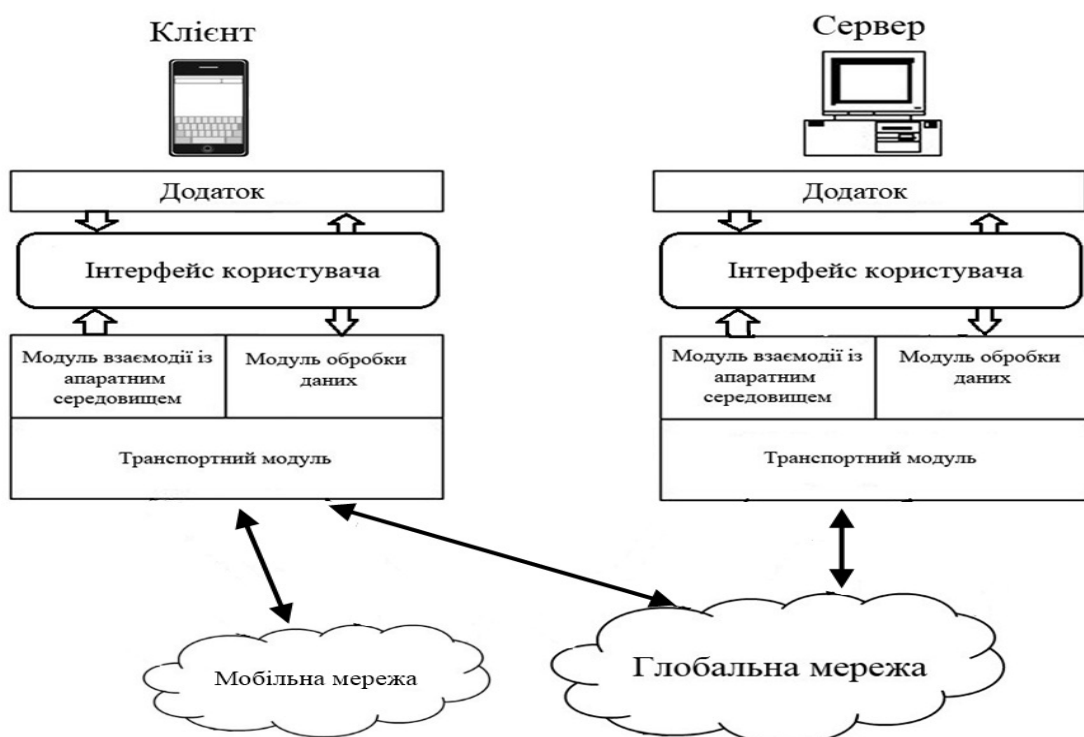


Рисунок 3.3 – Програмна архітектура додатків.

Для програмної реалізації обраної архітектури та організації швидкого обміну даними використовуємо мережеві інтерфейси Socket, описані в минулих розділах. Оскільки додаток забезпечує можливість відправлення ширококомовних повідомлень на основі протоколу UDP і складається з двох частин, які працюють одночасно і в ролі клієнта, і в ролі сервера. Використовуємо для цього класи стандартної бібліотеки Java DatagramSocket і DatagramPacket.

Як було сказано вище, дейтаграмні сокети внаслідок своєї простоти мають більшу продуктивність та можуть бути застосовані для ширококомовної пакетної розсилки даних. Тому для вирішення поставлених завдань в розроблюваному продукті обраємо цю технологію.

Для роботи з такими сокетами в додатку скористаємось об'єктами класу DatagramSocket. Крім того для передачі даних потрібно буде працювати з об'єктами класу DatagramPacket. У нього будемо записувати створений об'єкт поточкових даних, отриманий від мобільного пристрою. Кожен відправлений або одержаний пакет чи сокет і дейтаграми індивідуально адресуються і маршрутизуються. Множинні пакети, відправлені з однієї машини на іншу, може бути направлений по-різному, і можуть надходити в будь-якому порядку.

Клас для створення сокетів DatagramSocket має два варіанти перевизначеного конструктора, варіанти яких наведені нижче:

```
publicDatagramSocket (intport);  
publicDatagramSocket ();
```

Перший варіант конструктора надає можливість розробникові задати конкретний номер порта для підключення. В той же час інший варіант дозволяє створити з'єднання з використання будь-якого вільного відкритого порту. Скористаємось тут другим варіантом конструктора та створимо об'єкт сокета:

```
publicstaticDatagramSocketsocket;  
privateintport = 50005;  
DatagramSocketsocket = newDatagramSocket ();
```

Канал обміну даними разом з потоками вводу/виводу створювати не потрібно. Для цього служать методи `send` і `receive` відповідно для відправки та отримання пакетів, задекларовані у класі `DatagramSocket`:

```
publicvoidsend (DatagramPacket p);  
publicvoidreceive (DatagramPacket p);
```

В додатку використовуємо один з цих методів залежно від потреби прийому чи передачі та в якості аргумента передаємо йому створений об'єкт класу `DatagramPacket`.

```
socket.send(packet);
```

Після того, як сокет більше не потрібен, для оптимального використання оперативної пам'яті всі об'єкти цього класу мають бути закриті. Сам об'єкт продовжує бути доступним, але після виклику методу `close` стає неактивним:

```
publicvoidclose ();
```

Для обміну даними через виклики методів `receive` та `send` мають бути вже підготовлені об'єкти класу `DatagramPacket`. Як і для будь-яких інших об'єктів, з цією метою викликаємо конструктор класу:

```
publicDatagramPacket (byte,int);
```

В якості аргументів цей конструктор отримує ідентифікатор масиву буферу, котрий буде містити відправлені чи прийняті дані, а також розмір цього масиву. Тут підхід дуже близький до роботи з масивами у мовах C/C++. Передача даних здійснюється викликом відповідного конструктора класу дейтаграмного

пакета даних, який в якості аргументів приймає буфер з даними, його розмір, об'єкт класу IP-адреси та число, що означає номер порту для підключення:

```
publicDatagramPacket (byte, int, InetAddress, int);
```

Таким чином сокет у нас відповідає тільки за встановлення каналу зв'язку. А характеристики підключення зберігаються в окремому об'єкті, що дозволяє організувати пул ідентичних підключень без дублювання програмного коду з повторним використанням об'єктів DatagramPacket.

Створюємо конструктор:

```
DatagramPacket packet;  
packet = new DatagramPacket(buffer, buffer.length, destination,  
port);
```

Таким чином упаковані дані використовуються клієнтом і відправляються до вузла призначення мережі, а там їх приймає сервер для подальшого відтворення.

Серверна сторона програмного комплексу функціонує за допомогою класу ServerSocket, яких трохи відрізняється від звичайного класу Socket. Головна відмінність полягає в тому, програма з його допомогою перебуває у стані очікування запиту від клієнта. При створенні такого сокету традиційно потрібно вказати порт підключення та після цього викликаємо його метод accept(). Саме цей метод класу серверного сокету переводить програму у стан очікування запиту на підключення до порту, номер якого вказаний при створенні об'єкту класу серверного сокету. Настає зацикловання процесу сервера до настання підключення клієнта. Коли клієнт успішно підключився до процесу сервера, створюється звичайний об'єкт типу Socket. І вже цей об'єкт служить для прийому аудіо- чи відеоданих від клієнта для подальшої трансляції.

3.4 Програмний комплекс для інтеграції мобільного зв'язку з мережею Інтернет

Розроблений програмний додаток дає можливість користувачеві переадресувати безліч дзвінків зі свого мобільного пристрою до додатку приймача. За допомогою якого можливо зв'язуватись з користувачами мобільних мереж не залежно від оператора, навіть переадресація можлива коли дзвінок надходить із стільникового телефону. Така можливість зручна для користувачі з безліччю мобільних пристроїв, навіть розмежованих територіально.

Всі дзвінки надходитимуть на сервер оснащений інтерфейсом користувача, для персонального прийому дзвінків, ведення бізнесу, зв'язування з пристроями старого покоління та тими, які знаходять не в зоні доступу до мережі Інтернет, але мають змогу зателефонувати вам по мережі GSM. Таким способом ви можете зв'язуватись з усіма зі свого універсального приймача на персональному комп'ютері. Узагальнивши, до визначних досягнень розробленої технології та основних її можливостей можна віднести:

- комунікаційний зв'язок, який потребує належності мережі лише на вашій стороні, співрозмовнику потрібна лише мережа GSM;
- незалежність від операторів в мобільній мережі;
- об'єднання пере адресацією безлічі різних пристроїв з підтримкою клієнтського додатку;
- якісний комунікаційний зв'язок за рахунок інтегрування мережі.

Програмний комплекс складається з двох додатків: клієнта і сервера. Сервер безперервно приймає пакети потокових даних через сокет дейтаграм. Кожна дейтаграма, отримана сервером вказує на запит клієнта. Сервер обробляє дані, які надійшли до нього відправляє відповідь і відтворює аудіо співрозмовника, відкриваючи канал для передачі клієнту.

Клієнтська програма виконує основні, але доволі не громіздкі функції, передаючи основний потік, заради якого функціонує весь комплекс. Інтерфейс клієнтської частини зображений на рис. 3.4 доволі простий для користування і має поле для вводу адреси сервера, та дві керуючі кнопки, які запускають основний сервіс, а тобто весь процес переадресації та зупиняють його.

Мережевий програмний комплекс, орієнтований на роботу клієнта з сервером складається з наступних архітектурних шарів:

- ядро програми, оброблює з'єднання і містить контейнери для поточкових даних, не взаємодіє з користувачем;
- інтерфейс користувача клієнтської частини, використовується для установки параметрів з'єднання з сервером;
- інтерфейс користувача серверної частини, використовується для прийому або відхилення дзвінків.
- компоненти повторного використання: бібліотеки, візуальні компоненти та інше;
- ресурси програми: буфер, необхідні бінарні файли.

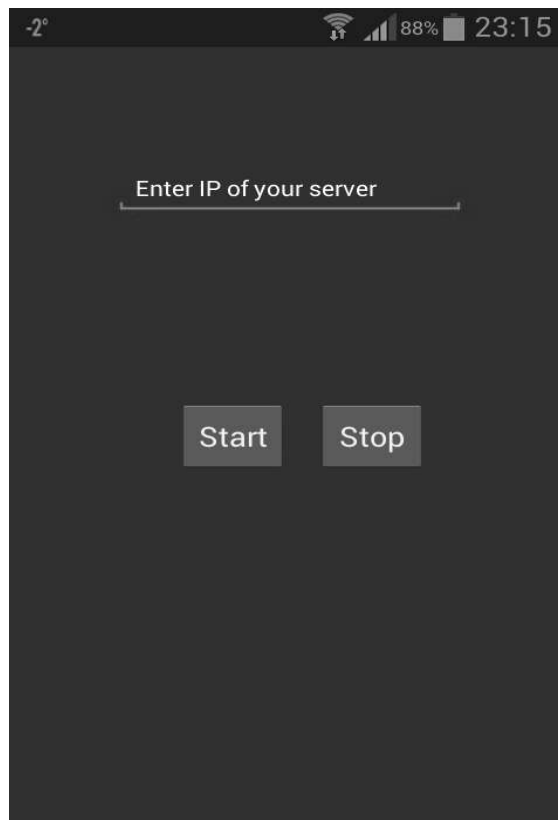


Рисунок 3.4 – Інтерфейс клієнтської частини додатку для переадресації

Найважливішою умовою побудови стійкої архітектури є відділення ядра системи від інтерфейсу, настільки, що б одне могло успішно функціонувати без іншого.

Ядро додатку, складається з наступних шарів:

- стартовий шар, що визначає робочий процес початку виконання програми;
- мережевий шар механізму транспортної взаємодії;
- шар єдиного набору команд для взаємодії клієнта з сервером;
- шар перевірки даних з мережі;
- шар передачі даних;
- модель даних для відображення бізнес-процесів;
- локальний кеш для роботи з отриманими ресурсами;
- шар робочих процесів, що реалізує бізнес-логіку процесів.

Загальна архітектура комунікаційної взаємодії пристроїв за технологією інтегрування мобільного зв'язку в глобальну мережу зображена на рисунку 3.5.



Рисунок 3.5 – Загальна архітектура програмного комплексу

Аналіз ключових особливостей актуальних мережевих комунікацій дозволяє говорити про їх істотні еволюційні зміни і визначити потреби користувачів на сьогодні. Розроблена технологія має багато перспектив на сьогодні, а що говорити про розвиток. Додаток може адаптуватись під іншу технологію організації мережі, наприклад рівний до рівного. Таким чином існує безліч можливостей та способів використання в комунікаційному середовищі.

Так звичайні користувачі можуть при проведенні часу за персональним комп'ютером бути активними в мобільній мережі навіть знаходячись віддалено від свого пристрою. З допомогою розробленого програмного засобу є можливість для користувача супроводжувати та автоматизувати виконання завдань з обслуговування декількох клієнтів та об'єднувати в цих клієнтів у конференції для оперативного менеджменту бізнес-процесами.

РОЗДІЛ 4

ОСНОВИ РОЗРОБКИ ЗАСТОСУНКІВ ПІД ANDROID

Виділимо кілька основних моментів, щоб показати, яким чином проходить процес розробки під Android.

- В Java файлах описується логіка програми – те, програма повинна виконувати.

- У XML файлах розробляються макети – зовнішній вигляд.

- Як тільки додаток буде написано, потрібно використовувати інструмент збірки для того, щоб скомпілювати всі файли і упакувати їх разом в .apk файл, який можна запускати на пристроях Android і / або опублікувати в Google Play.

- Всі утиліти і файли, які використовуються для створення програми під Android, об'єднані в інтегроване середовище розробки (IDE). IDE – це програма, яку використовують для редагування файлів коду, а також компіляції і запуску.

- Раніше стандартним IDE для розробки під Android була Eclipse, але зараз його замінило більш функціональне Android Studio – продукт компанії Google.

Звичайно ж, існують більш глибокі процеси, що відбуваються за лаштунками перерахованих вище кроків. Наприклад, просунуті користувачі захочуть дізнатися роль віртуальної машини Dalvik.

Ось що ми розглянемо в рамках даної частини роботи:

- Завантажимо і встановимо Android Studio.

- Запуск і тестування додатків на пристроях і емуляторах Android.

- Створення простого додатку на Android, який виводить напис «Hello World» на екран мобільного пристрою, як приклад.

4.1 Установка середовища розробки Android Studio

Почати читати документацію і писати код що б дізнатися, на що здатна платформа – це дійсно заманливо. І це треба робити. Однак, для початку роботи з платформою Android необхідно налаштувати середовище розробки.

Навіть якщо виконувати всі кроки правильно, може знадобитися усунути невелику проблему з налаштуванням середовища в залежності від конфігурації системи або версії продукту. Для цього можна використовувати пошукові сервіси. Особливо можна виділити ресурс [StackOverflow](https://stackoverflow.com).

Важливо не допустити, щоб будь-які підводні камені перешкоджали кінцевій меті – навчитися Android програмування. Відомо, що навіть професіонали іноді відчувають певні проблеми з налаштуванням робочого оточення. У таких випадках буває важливо знання командного рядка.

Поряд з тренуваннями в синтаксисі, важливо навчити себе мисленню успішного програміста, яке не сприйматиме повідомлення про помилку `file X not found` остаточним вироком.

Android Studio можна завантажити за посиланням: developer.android.com/studio/index.html (рис. 4.1).

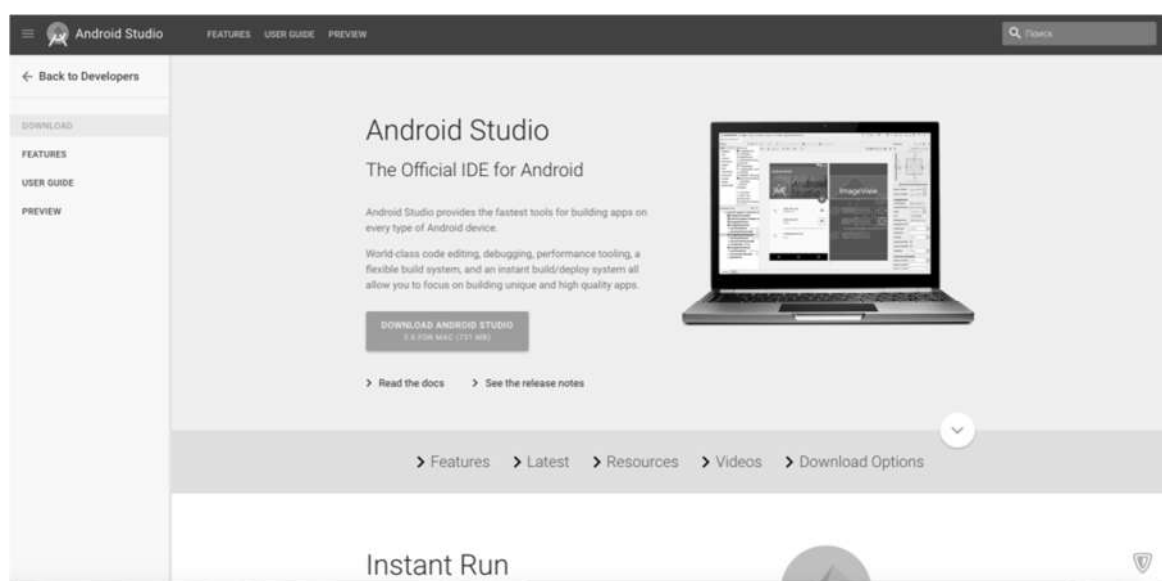


Рисунок 4.1 – Сторінка завантаження Android Studio

Після натискання на кнопки завантаження користувача попросять ознайомитися з правилами та умовами використання програмного продукту. Після уважного прочитання і прийняття умов ліцензії, починається завантаження. Після цього можна встановити Android Studio. На початковій сторінці завантаження містяться інструкції по установці під Mac і Windows.

Тепер після запуску Android Studio програма запитає, чи хоче користувач імпортувати свої налаштування. Зрозуміло, що для першого запуску імпортувати нічого.

Тепер відкриється завантажувальний екран в стилі Material Design. Після закінчення завантаження, користувач потрапить на екран вітання

Щоб уникнути проблем з версіями в подальшому, слід кнопку «Check for updates now» і, якщо необхідно, виконати всі інструкції для отримання останньої версії. Іноді Studio автоматично інформує про те, що є оновлення (рисунок 4.2).

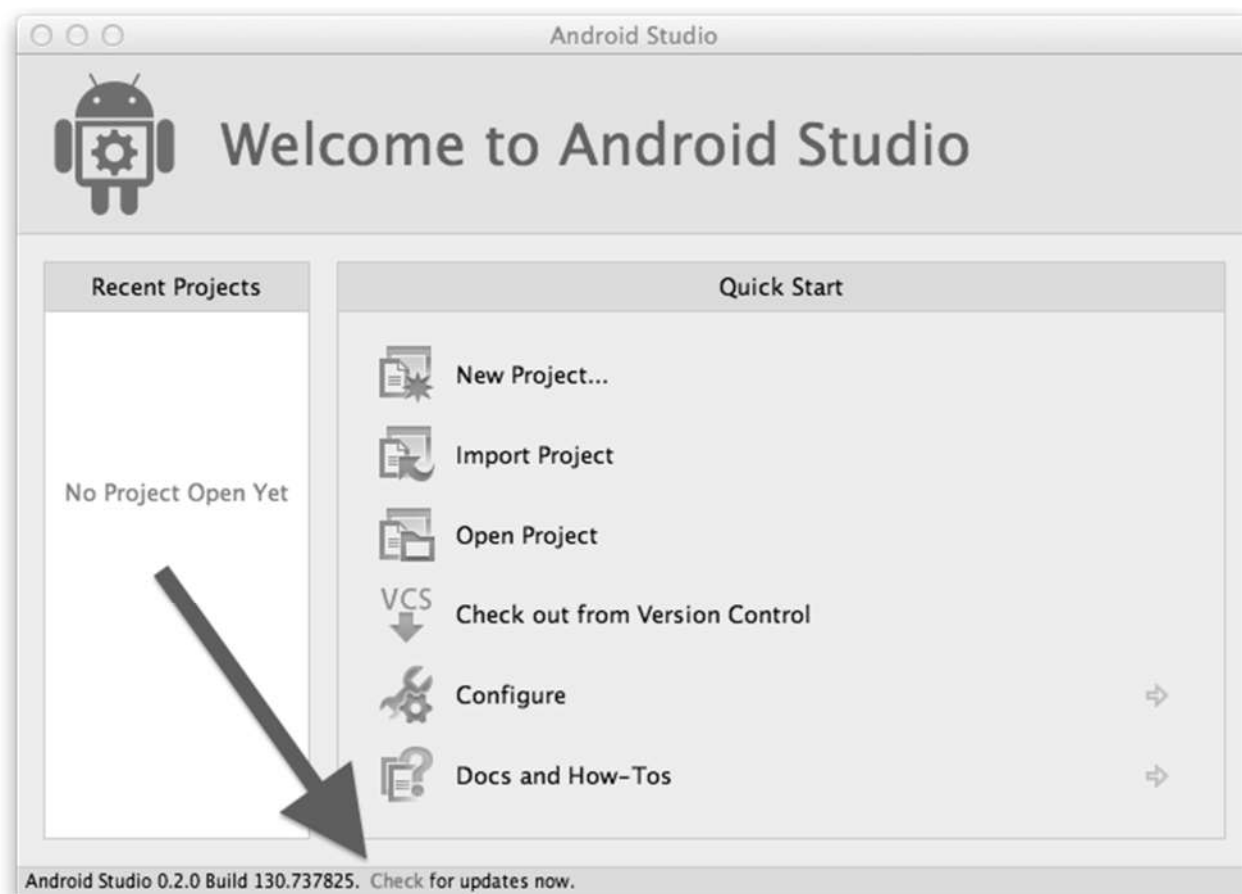


Рисунок 4.2 – Початковий екран Android Studio

В цьому випадку завжди слід вибирати Update and Restart.

4.2 Створення першого Android проекту

Прийшов час створити перший проект. Почнемо з простого. У програмістів прийнято називати першу програму «Hello World». Давайте дотримуватися цієї традиції, а потім зробимо кілька невеликих змін, щоб додаток використовував ваше ім'я для привітання. Android Studio має невеликий покроковий інструмент, який допоможе вам створити свій проект. Натисніть «New Project» на стартовому екрані (рисунок 4.3).



Рисунок 4.3 – Вибір типу проекту

Далі відкриється вікно, подібне наступного (рисунок 4.4):

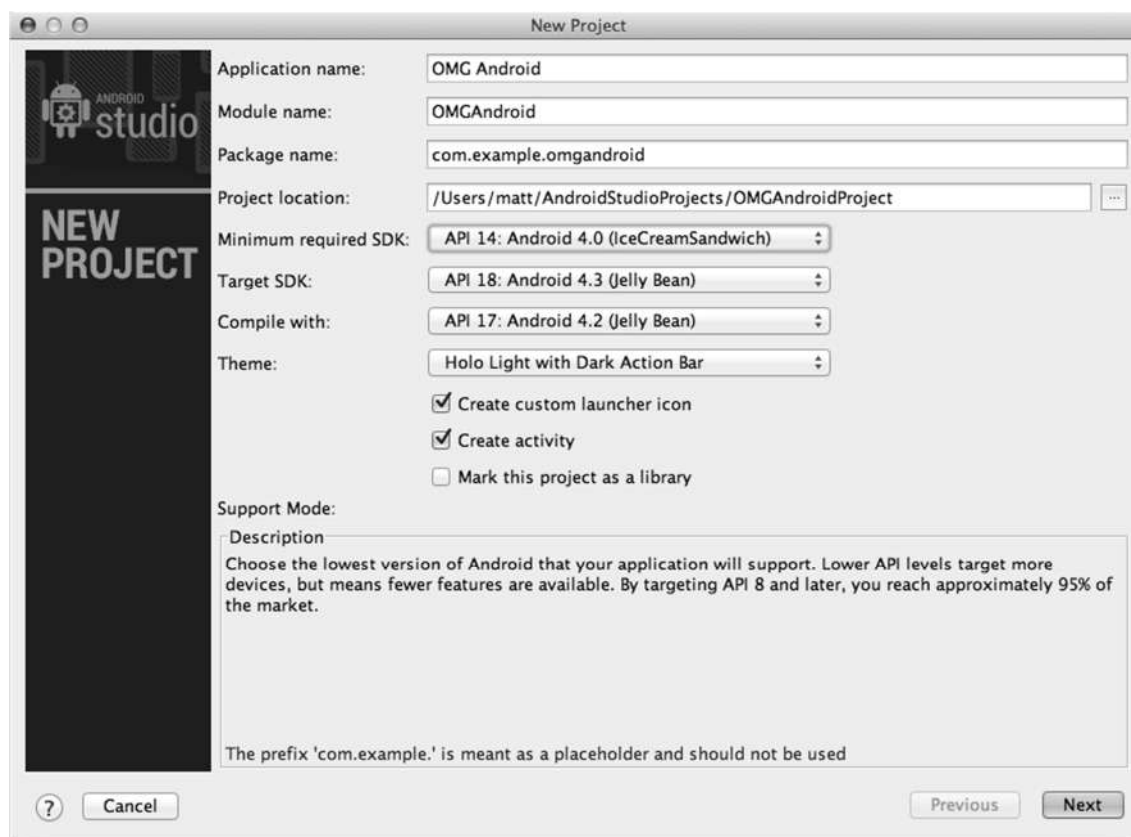


Рисунок 4.4 – Властивості новоствореного проекту

Можна встановити місце розташування проекту, вказавши будь-яку папку на жорсткому диску

Для розкриваються версій SDK зверніть увагу на розділ «Опис» в нижній частині діалогового вікна. У ньому пояснюється, для чого призначена кожна настройка.

Встановіть мінімальний необхідний SDK, як показано на знімку екрана. Це встановлює мінімальну версію Android, необхідну для запуску програми. Вибір цього значення для власних проектів – це питання балансування можливостей SDK, які ви хочете, і пристроїв, які будуть підтримуватися.

Для першого додатка я рекомендую використовувати версію API 14, яка є Android 4.0 Ice Cream Sandwich. Кожна програма буде мати різні вимоги, і ви можете вибрати щось інше, в залежності від ситуації.

Для отримання додаткової інформації про версії API і їх використанні, на сайті для розробників під Android існує спеціальна сторінка Dashboards <https://developer.android.com/about/dashboards/index.html>.

Після вибору версії, відкривається екран вибору стартового шаблону. Ви можете створити додаток, яке вже взаємодіє з API Google Maps і відображає карту. У нашому тестовому прикладі вибираємо Empty Activity і натискаємо кнопку «Next».

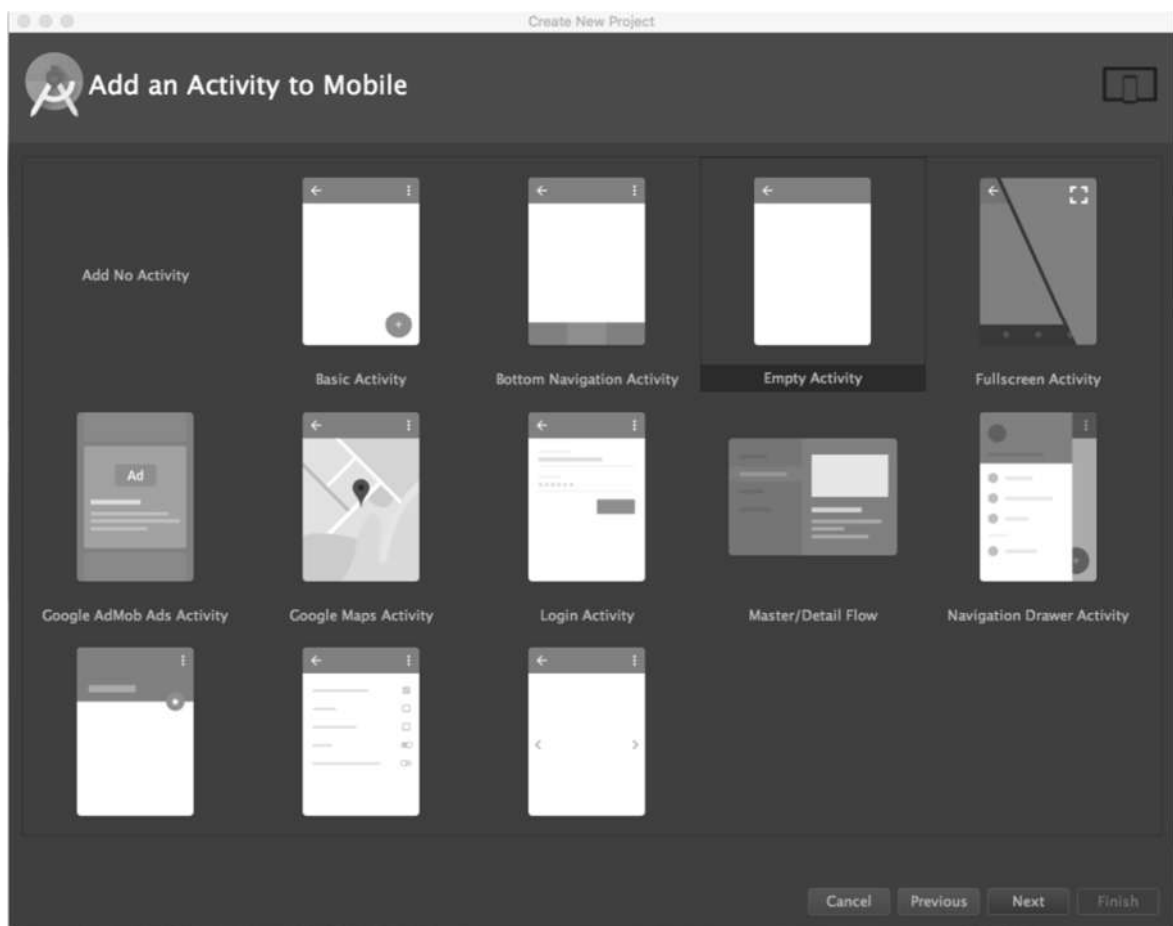


Рисунок 4.5 – Шаблони для нових проектів

І зараз на останньому кроці процесу створення програми слід звернути увагу на деякі речі. Тут користувач вперше стикається зі згадками про головні архітектурні компоненти усіх програм.

Activity Name – це перша, але не остання згадка слова Activity. В контексті Android, Activity зазвичай розглядається як «екран» в додатку. Цей елемент дуже

гнучкий. Коли Android Studio створює клас MainActivity, він успадковує його від класу Activity з пакета Android SDK. Ті, хто знайомий з об'єктно-орієнтованим програмуванням, розуміють цю концепцію, але для новачків, це, в основному, означає, що MainActivity буде настраюється версією Activity.

Layout Name – макет того, що буде показано користувачеві, визначений в спеціальному вигляді Android XML. Ви скоро навчитеся читати і редагувати ці файли.

Натискаємо Finish. Деякий час займе створення і завантаження проекту. Через деякий час Android Studio завершить білд проекту. Звичайно, проект поки порожній, але в ньому є все необхідне для запуску на Android-пристрої або емуляторі.

Після завантаження проекту файл макету можна переглядати у форматі XML. Перш ніж перейти до програмування під Android, давайте поговоримо про те, яким чином ми можемо запустити цю програму.

4.3 Запуск програми на емуляторі

Тепер настав час описати коротко емулятори. Android Studio поставляється з програмним забезпеченням, здатним емулювати Android-пристрій для запуску на ньому програм, перегляду веб-сайтів, налагодження і всього іншого (рисунок 4.6).

Цю можливість надає Android Virtual Device (AVD) Manager. За бажанням можна налаштувати кілька емуляторів, встановити розмір екрана і версію платформи для кожного нового емулятора. Цей функціонал дуже корисний, оскільки позбавляє розробників від необхідності купувати кілька пристроїв для тестування програм.

Натисніть на кнопку Run у вигляді зеленої стрілки.

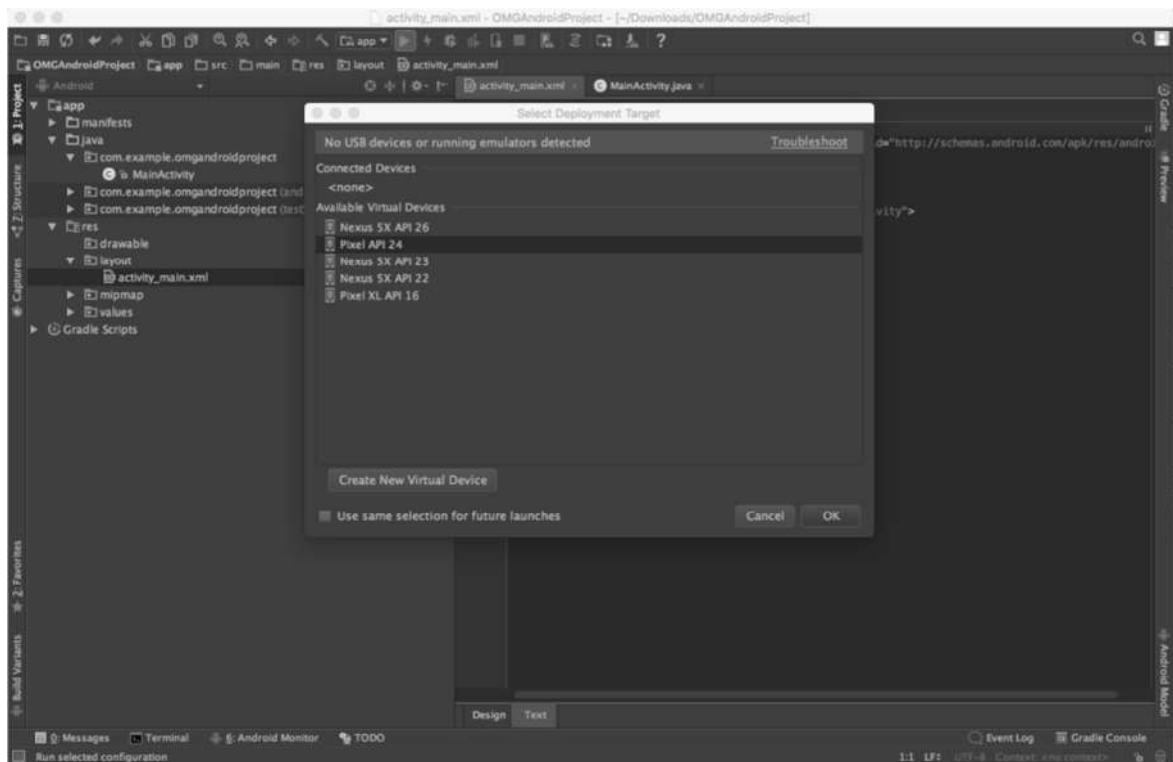


Рисунок 4.6 – Запуск проекту

Доведеться почекати деякий час, поки емулятор завантажиться і як тільки він буде готовий, ви побачите щось на зразок цього (рисунок 4.7):

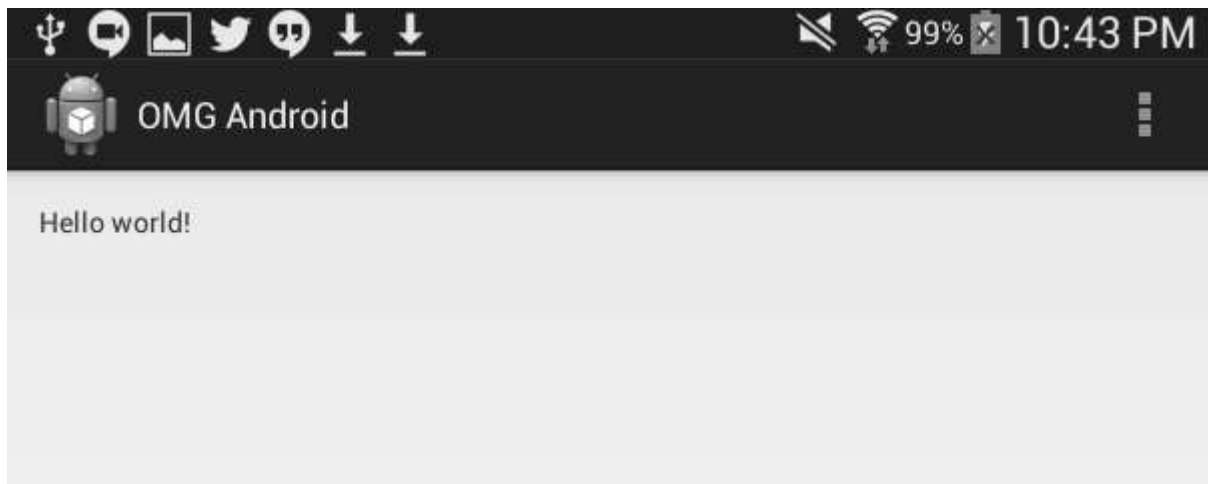


Рисунок 4.7 – Виконання простої програми в емуляторі

Щоб почати робити зміни і додавати цікаві функції, необхідно отримати робоче знання про те, що відбувається за лаштунками. Погляньте на розділ проекту Android Studio з файлами і папками в лівій частині екрана. Можливо,

вам знадобиться натиснути маленьку вкладку на краю (див. рис. 4.8), якщо на даний момент провідник проекту не відображається.

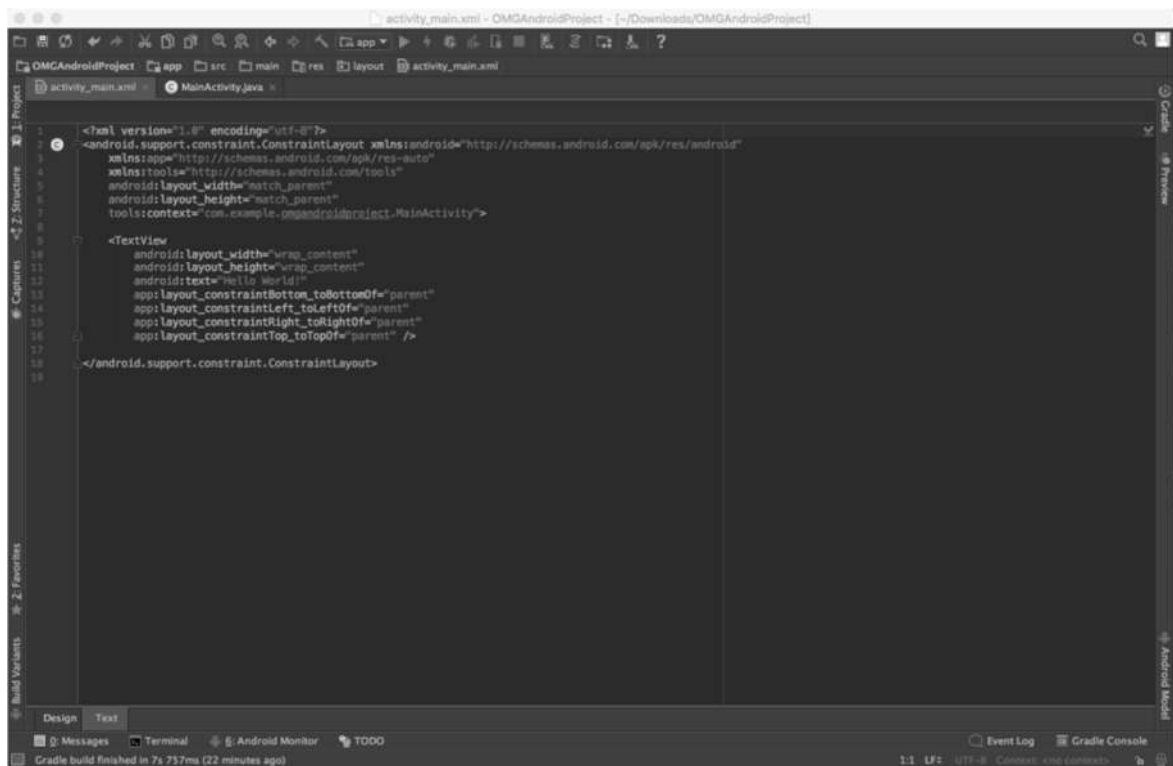


Рисунок 4.8 – Вікно редактора коду

Перегляньте протягом декількох хвилин структуру папок і двічі клікніть по файлах, щоб побачити їх вміст в головному вікні.

4.4 Структура проекту Android

У проєктах Android є кілька ключових елементів, і кожен з них повинен зіграти певну роль.

JAVA.

Це частина вашого коду, яка відповідає за логіку програми. Ваш код буде знаходитися в каталозі `src \ main \ java` в основній папці проєкту. Для вивчення Java можна порадити книгу Брюса Еккель «Філософія Java».

RESOURCES.

Недостатньо просто зробити Android додаток, він повинен бути ще і стильним. Ппрограма ніколи не буде виділятися, якщо у неї немає чітких значків і зображень, добре продуманих макетів і, можливо, навіть плавних анімацій.

При ініціалізації папка містить наступні папки:

- drawable, в якій зберігаються іконки. Зараз там лежить тільки стандартна іконка програми.
- layout з XML файлами, які представляють собою дизайни екранів.
- menu с XML файлами списків елементів, які будуть відображатися в панелі дій.
- values з XML файлами, що містять розміри, кольори, рядкові константи і стилі.

ANDROIDMANIFEST.XML.

Цей XML-файл інформує систему щодо вимог до обладнання та програмного забезпечення програми та містить його версію, ім'я та значок. Маніфест також містить інформацію про всі Activity в додатку.

4.5 Внесення змін в програмний код та дизайн

Перейдіть до res / values / strings.xml і двічі клацніть файл. Коли відкрити файл, то буде видно два стрічкових ресурси в XML.

Ці ресурси використовуються в різних місцях, але дуже зручно мати весь текст, який використовується в додатку, в одному файлі. Якщо потрібно перевести його, або якщо маркетолог попросить видалити всі зайві посилання, тут буде легко внести всі зміни.

Змініть рядок hello_world, яку програма відображає на екрані. Змініть її вміст на щось більш особисте, наприклад використовуйте ваше власне ім'я. Вийде щось на зразок:

```
<string name = "hello_world">> Matt is learning Android! </ string>
```

Натисніть Run. Додаток перезавантажиться і буде видно персональне повідомлення (рис. 4.9).

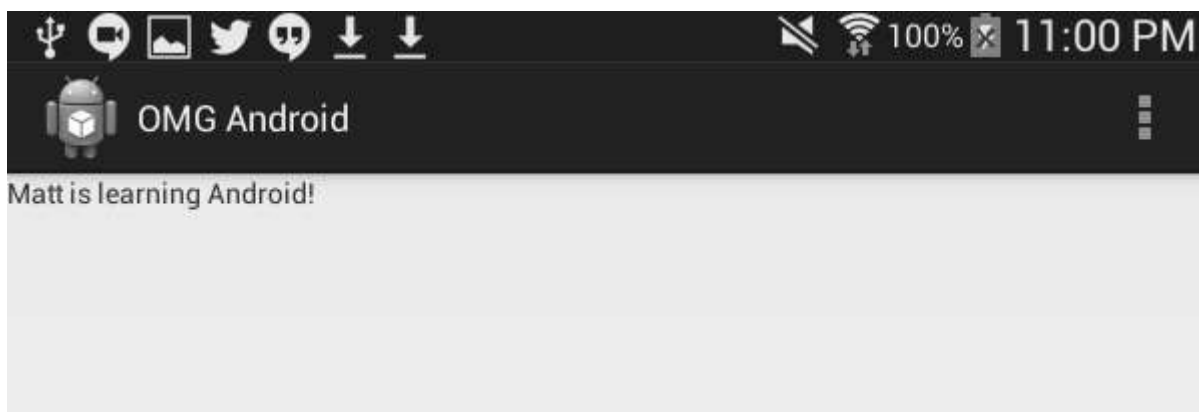


Рисунок 4.9 – Результат внесення змін в програмний код

Таким чином за допомогою цих простих кроків в розробці можна почати глибше вивчення програмування для ОС Android.

РОЗДІЛ 5

ОБҐРУНТУВАННЯ ЕКОНОМІЧНОЇ ЕФЕКТИВНОСТІ

Метою цього розділу дипломної роботи є здійснення економічних розрахунків, спрямованих на визначення економічної ефективності від розробки, а також прийняття рішення щодо подальшого розвитку і впровадження або ж недоцільність впровадження відповідної розробки. Передбачається, що описаний в роботі підхід буде імплементовано у вигляді програмного продукту для мобільної платформи. Розробка такого продукту вимагатиме певних затрат. Тому розрахуємо ці затрати. Для здійснення оцінки потрібно зробити розрахунки трудомісткості кожної операції.

5.1 Визначення стадій технологічного процесу та загальної тривалості проведення НДР

Витрати часу по окремих операціях технологічного процесу відображені в таблиці 5.1.

Таблиця 5.1 – Операції технологічного процесу та час їх виконання

№	Назва операції (стадії)	Викона- вець	Середній час виконання операції, год.
1.	Витрати праці на підготовку опису задачі	інженер	10
2.	Витрати праці на розробку проекту	інженер	15
3.	Витрати праці на розробку структури системи	інженер	10
4.	Витрати праці на створення системи	інженер	75
5.	Витрати праці на підготовку документації	інженер	15
6.	Витрати праці на тестування	інженер	40
Разом			165

Загальні затрати на дипломний проект становить 175 годин.

5.2 Визначення витрат на оплату праці та відрахувань на соціальні заходи

Відповідно до Закону України “Про оплату праці” заробітна плата – це “винагорода, обчислена, як правило, у грошовому виразі, яку власник або уповноважений ним орган виплачує працівникові за виконану ним роботу”.

Розмір заробітної плати залежить від складності та умов виконуваної роботи, професійно-ділових якостей працівника, результатів його праці та господарської діяльності підприємства. Заробітна плата складається з основної та додаткової оплати праці.

Основна заробітна плата нараховується на виконану роботу за тарифними ставками, відрядними розцінками чи посадовими окладами і не залежить від результатів господарської діяльності підприємства.

Додаткова заробітна плата – це складова заробітної плати працівників, до якої включають витрати на оплату праці, не пов’язані з виплатами за фактично відпрацьований час. Нараховують додаткову заробітну плату залежно від досягнутих і запланованих показників, умов виробництва, кваліфікації виконавців. Джерелом додаткової оплати праці є фонд матеріального стимулювання, який створюється за рахунок прибутку.

При розрахунку заробітної плати кількість робочих днів у місяці слід в середньому приймати – 24,5 дні/міс., або ж 196 год./міс. (тривалість робочого дня – 8 год.).

Місячний оклад кожного працівника слід враховувати згідно існуючих на даний час тарифних окладів. Згідно закону України «Про Державний бюджет України на 2020 рік», зокрема Статтею восьмою мінімальна заробітна плата у погодинному розмірі встановлена у розмірі 28,31 грн. Рекомендовані тарифні ставки: керівник дипломної роботи – 30,00...50,00 грн./год., інженер – 28,31...30,00 грн./год., консультант – 28,31...30,00 грн./год., технік – 28,31...40,00 грн./год., лаборант – 28,31...35,00 грн./год.

Основна заробітна плата розраховується за формулою:

$$Z_{осн.} = T_c \cdot K_z, \quad (5.1)$$

де T_c – тарифна ставка, грн.;

K_z – кількість відпрацьованих годин.

Оскільки всі види робіт в даному проекті виконує інженер, то основна заробітна плата буде розраховуватись тільки за однією формулою

$$Z_{осн.} = 30 \cdot 165 = 4950 \text{ грн.}$$

Додаткова заробітна плата становить 10–15 % від суми основної заробітної плати.

$$Z_{дод.} = Z_{осн.} \cdot K_{додл.}, \quad (5.2)$$

де $K_{додл.}$ – коефіцієнт додаткових виплат працівникам, 0,1–0,15 (візьмемо його рівним 0,15).

$$Z_{дод.} = 4950 \cdot 0,15 = 742,50 \text{ грн.}$$

Звідси загальні витрати на оплату праці ($B_{о.л.}$) визначаються за формулою:

$$B_{о.л.} = Z_{осн.} + Z_{дод.}, \quad (5.3)$$

$$B_{о.л.} = 4950 + 742,50 = 5692,50 \text{ грн.}$$

Крім того, слід визначити відрахування на соціальні заходи:

- 1) ЄСВ + ПДФО 22 %;
- 2) військовий збір – 1,5 %.

У сумі зазначені відрахування становлять 23,5 %.

Отже, сума відрахувань на соціальні заходи буде становити:

$$B_{c.z.} = \Phi_{оп} \cdot 0,235, \quad (5.4)$$

де $\Phi_{оп}$ – фонд оплати праці, грн.

$$B_{c.z.} = 5692,50 \cdot 0,235 = 1337,74 \text{ грн.}$$

Проведені розрахунки витрат на оплату праці зведемо у таблицю 5.2.

Таблиця 5.2 – Зведені розрахунки витрат на оплату праці

№ п/п	Категорія працівник- ків	Основна заробітна плата, грн.			Додаткова заробітна плата, грн.	Нарахув. на $\Phi_{оп}$, грн.	Всього витрати на оплату праці, грн. $6=3+4+5$
		Тарифна ставка, грн.	К-сть відпра- цьов. год.	Фактично нарах. з/пл., грн.			
1	інженер	30	165	4950	742,50	1337,74	7030,24

Загальні витрати на оплату праці становить 7030,24 грн.

5.3 Розрахунок матеріальних витрат

Матеріальні витрати визначаються як добуток кількості витрачених матеріалів та їх ціни:

$$M_{vi} = q_i \cdot p_i, \quad (5.5)$$

де: q_i – кількість витраченого матеріалу i -го виду;

p_i – ціна матеріалу i -го виду.

Звідси, загальні матеріальні витрати можна визначити:

$$Z_{м.в.} = \sum M_{Ві.} \quad (5.6)$$

Проведені розрахунки занесемо у таблицю 5.3. Для розробки ПЗ передбачається покупка Visual Studio Team Foundation Server CAL SNGL LicSAPk OLP NL UkrCAL 2017, вартість якого на сьогодні становить 19400 грн.

Таблиця 5.3 – Зведені розрахунки матеріальних витрат

Найменування матеріальних ресурсів	Одиниця виміру	Норма витрат	Ціна за одиницю, грн	Затрати матеріалів, грн	Транспортно-заготівельні витрати, грн	Загальна сума витрат на матеріали, грн
1. Основні матеріали						
Програмне забезпечення	комп.	1	19400,00	19400,00	–	19400,00
2. Допоміжні матеріали						
Папір формату А4	шт.	200	0,18	36	–	36
Разом:						19436,00

Загальні матеріальні затрати становлять 19436,00 гривень.

5.4 Розрахунок витрат на електроенергію

Затрати на електроенергію 1-ці обладнання визначаються за формулою:

$$Z_e = W \cdot T \cdot S, \quad (5.7)$$

де W – необхідна потужність, кВт;

T – кількість годин роботи обладнання;

S – вартість кіловат-години електроенергії.

Вартість кіловат-години електроенергії слід приймати згідно існуючих на даний час тарифів. Отже, 1 кВт з ПДВ коштує 2,42 грн.

Потужність комп'ютера для створення проекту – 550 Вт, кількість годин роботи обладнання згідно таблиці 5.1 – 165 годин.

Тоді, $z_e = 0,55 \cdot 175 \cdot 2,42 = 219,62$ грн.

5.5 Розрахунок суми амортизаційних відрахувань

Характерною особливістю застосування основних фондів у процесі виробництва є їх відновлення. Для відновлення засобів праці у натуральному виразі необхідне їх відшкодування у вартісній формі, яке здійснюється шляхом амортизації.

Амортизація – це процес перенесення вартості основних фондів на вартість новоствореної продукції з метою їх повного відновлення.

Комп'ютери та оргтехніка належать до четвертої групи основних фондів. Для цієї групи річна норма амортизації дорівнює 60 % (квартальна – 15 %).

Для визначення амортизаційних відрахувань застосовуємо формулу:

$$A = \frac{B_B \cdot H_A}{100\%}, \quad (5.8)$$

де A – амортизаційні відрахування за звітний період, грн.;

B_B – балансова вартість групи основних фондів на початок звітного періоду, грн.;

H_A – норма амортизації, %.

Для даного проекту засобом розробки є комп'ютер. Його сума становить 17400 грн. Отже, амортизаційні відрахування будуть рівні:

$$A = 17400 \cdot 5\% / 100\% = 870,00 \text{ грн.}$$

Оскільки робота виконувалась 165 годин, то амортизаційні відрахування будуть становити:

$$A = 870,00 \cdot 175 / 150 = 1015,00 \text{ грн.}$$

5.6 Обчислення накладних витрат

Накладні витрати пов'язані з обслуговуванням виробництва, утриманням апарату управління спілкою та створення необхідних умов праці.

В залежності від організаційно-правової форми діяльності господарюючого суб'єкта, накладні витрати можуть становити 20 – 60 % від суми основної та додаткової заробітної плати працівників.

$$H_B = B_{o.n.} \cdot 0,2 \dots 0,6, \quad (5.9)$$

де H_B – накладні витрати.

Отже, накладні витрати:

$$H_B = 5692,00 \cdot 0,2 = 1138,50 \text{ грн.}$$

5.7 Складання кошторису витрат та визначення собівартості НДР

Результати проведених вище розрахунків зведемо у таблицю 5.4.

Таблиця 5.4 – Кошторис витрат на НДР

Зміст витрат	Сума, грн.	В % до загальної суми
Витрати на оплату праці (основну і додаткову заробітну плату)	5692,50	19,7%
Відрахування на соціальні заходи	1337,74	4,6%
Матеріальні витрати	19436,00	67,4%
Витрати на електроенергію	219,62	0,8%
Амортизаційні відрахування	1015,00	3,5%
Накладні витрати	1138,50	3,9%
Собівартість	28839,35	100,0%

Собівартість (C_B) проекту розрахуємо за формулою:

$$C_B = B_{o.n.} + B_{c.z.} + Z_{m.v.} + Z_B + A + H_B. \quad (5.10)$$

Отже, собівартість проекту дорівнює:

$$C_B = 5692,50 + 1337,74 + 19436 + 219,62 + 1015,00 + 1138,50 = 28839,35 \text{ грн.}$$

5.8 Розрахунок ціни проекту

Ціну НДР можна визначити за формулою:

$$Ц = \frac{C_B \cdot (1 + P_{рен}) + K \cdot B_{н.і.}}{K} \cdot (1 + ПДВ), \quad (5.11)$$

де $P_{рен.}$ – рівень рентабельності, 30 %;

K – кількість замовлень, од. (встановлюється лише при розробці програмного продукту та мікропроцесорних систем);

$B_{н.і.}$ – вартість носія інформації, грн. (встановлюється лише при розробці програмного продукту);

$ПДВ$ – ставка податку на додану вартість, (20 %).

Оскільки розробка є прикладною, і використовуватиметься тільки для одного підприємства, то для розрахунку ціни не потрібно вказувати коефіцієнти K та $B_{н.і.}$, оскільки їх в даному випадку не потрібно.

Тоді, формула для обчислення ціни розробки буде мати вигляд:

$$Ц = C_B \cdot (1 + P_{рен}) \cdot (1 + ПДВ). \quad (5.12)$$

Звідси ціна на проект складе:

$$Ц = C_B \cdot (1 + 0,3)(1 + 0,2) = 44989,39 \text{ грн.}$$

5.9 Визначення економічної ефективності і терміну окупності капітальних вкладень

Ефективність виробництва – це узагальнене і повне відображення кінцевих результатів використання робочої сили, засобів та предметів праці на підприємстві за певний проміжок часу.

Економічна ефективність (E_p) полягає у відношенні результату виробництва до затрачених ресурсів:

$$E_p = \Pi / C_B, \quad (5.13)$$

де Π – прибуток;

C_B – собівартість.

Плановий прибуток ($\Pi_{пл}$) знаходимо за формулою:

$$\Pi_{пл} = \Pi - C_B. \quad (5.14)$$

Розраховуємо плановий прибуток:

$$\Pi_{пл} = 44989,39 - 28839,35 = 16150,04 \text{ грн.}$$

Отже, формула для визначення економічної ефективності набуде вигляду:

$$E_p = \frac{\Pi_{пл}}{C_B}. \quad (5.15)$$

$$\text{Тоді, } E_p = 16150,04 / 28839,35 = 0,56$$

Поряд із економічною ефективністю розраховують термін окупності капітальних вкладень (T_p):

$$T_p = 1 / E_p , \quad (5.16)$$

Термін окупності дорівнює:

$$T_p = 1 / 0,56 = 1,8 \text{ роки.}$$

В цьому розділі дипломної роботи було розраховано основні техніко-економічні показники проекту (див. таблицю 5.5).

Розраховане значення економічної ефективності становить 0,56 що є високим значенням.

Так само нормальним є термін окупності. Для даного продукту він становить 1,8 роки.

Таблиця 5.5 – Техніко-економічні показники НДР

№ п/п	Показник	Значення
1.	Собівартість, грн.	28839,35
2.	Плановий прибуток, грн.	16115,04
3.	Ціна, грн.	44989,39
4.	Економічна ефективність	0,56
5.	Термін окупності, рік	1,8

Отже, даний проект може бути впроваджений та мати подальший розвиток, оскільки він є економічно вигідним за всіма основними техніко-економічними показниками.

РОЗІДЛ 6

ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

6.1 Функціональні заходи у сфері державного регулювання та контролю захисту населення і територій

Функціональні підсистеми єдиної державної системи цивільного захисту створюються у відповідних сферах суспільного життя центральними органами виконавчої влади з метою захисту населення і територій від надзвичайних ситуацій у мирний час та в особливий період, забезпечення готовності підпорядкованих їм сил і засобів до дій, спрямованих на запобігання і реагування на надзвичайні ситуації.

Безпосереднє керівництво діяльністю функціональної підсистеми здійснюється керівником органу чи суб'єкта господарювання, що створив таку підсистему. Підсистема здійснює ряд заходів через органи державної влади.

Державна стандартизація.

Стандартизація становить основу проведення державної експертизи, контролю, нагляду, ліцензування видів діяльності, декларування безпеки промислових об'єктів і сертифікації для встановлення норм, правил та характеристик з метою забезпечення:

- безпеки продукції, робіт та послуг для довкілля, життя і здоров'я людей;
- якості продукції, робіт і послуг відповідно до рівня розвитку науки, техніки і технологій;
- єдності вимірів;
- безпеки об'єктів господарювання з урахуванням ризику природних і техногенних катастроф та інших надзвичайних ситуацій.

Державна експертиза.

Державна експертиза проектів і рішень щодо об'єктів виробничого та соціального призначення і процесів, що можуть спричинити надзвичайні

ситуації та вплинути на стан захисту населення і територій, організується і проводиться відповідно до законодавства спеціально уповноваженими центральними і місцевими органами виконавчої влади, виконавчими органами рад.

У разі потреби експертиза проектів і рішень щодо об'єктів виробничого та соціального призначення, окремих технологічних процесів, які можуть викликати надзвичайні ситуації або вплинути на стан захисту населення і територій від надзвичайних ситуацій, може проводитися об'єднаннями громадян, незалежними експертами та спеціалістами міжнародних експертних організацій у порядку, встановленому законодавством України.

Державний нагляд і контроль.

Державний нагляд і контроль організується з метою перевірки повноти і якості заходів щодо запобігання надзвичайним ситуаціям, забезпечення готовності органів управління, сил і засобів системи захисту населення і територій, посадових осіб до дій у разі виникнення цих ситуацій. Нагляд і контроль полягають у забезпеченні:

- безаварійного функціонування об'єктів економіки та зменшення масштабів можливих збитків у разі виникнення надзвичайних ситуацій природного та техногенного походження;

- готовності органів управління системи захисту населення і територій, підприємств та їх спеціалізованих аварійно-рятувальних формувань до виконання поставлених перед ними завдань;

- охорони довкілля та збереження природних ресурсів, додержання порядку і умов користування надрами з метою запобігання небезпечним екологічним і геологічним процесам;

- готовності технічних засобів і споруд інженерного захисту населення і територій від лісових, торф'яних та інших пожеж, повеней, підтоплень та інших небезпечних дій води;

- фінансових гарантій відшкодувань збитків здоров'ю та життю громадян, захисту майнових інтересів фізичних та юридичних осіб від збитків,

спричинених надзвичайними ситуаціями природного та техногенного походження.

Державний нагляд і контроль проводиться згідно із законодавством України спеціально уповноваженими центральними та місцевими органами виконавчої влади, виконавчими органами рад відповідно до завдань, покладених на систему захисту населення і територій.

Ліцензування окремих видів діяльності.

Ліцензування окремих видів діяльності здійснюється з метою проведення єдиної державної політики для забезпечення життєво важливих інтересів громадян, суспільства, держави. Ліцензуванню підлягає діяльність щодо забезпечення промислової, пожежної та транспортної безпеки, охорони довкілля та інших сфер.

Декларування безпеки промислових об'єктів.

Декларування безпеки промислових об'єктів здійснюється з метою забезпечення контролю за додержанням заходів безпеки на етапах їх введення в експлуатацію, експлуатації та виводу з експлуатації. Декларування передбачає:

- оцінку ризику виникнення на промислових об'єктах надзвичайних ситуацій з урахуванням визначення джерел загроз, умов розвитку і можливих наслідків надзвичайних ситуацій, у тому числі викидів у довкілля шкідливих речовин;

- оцінку готовності до експлуатації потенційного об'єкта відповідно до вимог промислової безпеки;

- аналіз достатності й ефективності вжитих заходів щодо запобігання, локалізації та ліквідації надзвичайної ситуації на промисловому об'єкті;

- розроблення заходів, спрямованих на зменшення масштабів і величини негативних наслідків від надзвичайних ситуацій.

Сертифікація.

Сертифікація організується і здійснюється з метою підтвердження відповідності продукції встановленим вимогам, включаючи контроль небезпечної продукції для життя та здоров'я людей, довкілля та майна.

Сертифікація може мати обов'язковий і добровільний характер.

Страхування.

Головною метою страхування є забезпечення економічної підтримки заходів щодо запобігання надзвичайним ситуаціям, які здійснюються центральними і місцевими органами виконавчої влади, виконавчими органами рад, підприємствами та організаціями незалежно від форм власності, і страхового покриття збитків у разі їх виникнення. Страхування організується і здійснюється на підставі договору про страхування, який є сукупністю видів страхування, що передбачають обов'язок страховика щодо страхових виплат у розмірі повної або часткової компенсації збитків, завданих об'єкту страхування.

Страховий захист населення і територій від надзвичайних ситуацій забезпечується обов'язковим і добровільним страхуванням.

6.2 Підвищення стійкості роботи комп'ютеризованих систем в умовах дії ЕМІ ядерних вибухів

Ядерні вибухи, особливо в атмосфері й більш високих шарах, призводять до виникнення потужних електромагнітних полів з довжиною хвиль від 1 до 1000 м і більше. Ці поля через короткочасне існування називають електромагнітним імпульсом (ЕМІ). ЕМІ виникає при ядерному вибусі у воєнний час, у мирний час – при випробуванні ядерної зброї або ядерних аваріях і катастрофах в атмосфері й космосі.

Гамма-промені, які випускаються із зони вибуху в напрямі поверхні землі, поглинаються в більш щільних шарах атмосфери, вибиваючи з атомів повітря швидкі електрони, які летять у напрямку гамма-променів зі швидкістю світла, а позитивні іони (залишки атомів) залишаються на місці. У результаті поділу і переміщення позитивних і негативних зарядів у цій області й у зоні вибуху, а також при взаємодії зарядів з геомагнітним полем Землі утворюються елементарні й результуючі електричні та магнітні поля ЕМІ, які досягають поверхні землі в зоні радіусом кількох сотень кілометрів. Виникають сильні

поперечні струми і утворюється подібність великої "плоскої антени", яка випромінює потужний ЕМІ з часом наростання порядна 10 нс і тривалістю більше 230 нс; зі смугою частот від 10 кГц до 100 МГц. Залежно від висоти ядерного вибуху за інших однакових умов змінюються характер, інтенсивність ЕМІ і дальність його поширення.

При наземному і низькому повітряному вибуху уражаюча дія ЕМІ спостерігається на відстані кількох кілометрів від центру вибуху. Під час ядерного вибуху на висотах від 3 до 25 км утворюється симетричне джерело генерації, але радіус поширення ЕМІ залишається обмеженим внаслідок сильного поглинання гамма-випромінювання в щільних шарах атмосфери.

Найбільшу уражаючу дію має ЕМІ, що виникає при екзоатмосферному вибуху (більше 40 км). Зі збільшенням висоти вибуху збільшується і район джерела генерації ЕМІ, досягаючи в діаметрі тисячі кілометрів і товщини 20 – 40 км. Так, під час вибуху на висоті 80 км ЕМІ буде поширюватися на площі радіусом 960 км, а під час вибуху на висоті 160 км – на площі радіусом 1400 км. Екзоатмосферний ЕМІ характеризується дуже малим часом наростання (декілька сот наносекунд), високою інтенсивністю електричного поля (більше 50 кВ/хв) і магнітного поля (близько 130 А/хв).

Розряд блискавки порівняно з ЕМІ має значно більшу тривалість зростання і спаду (5—300 мкс), створює дуже потужні поля (близько 100 кВ/хв), несе значно більшу енергію, але спектр частот становить близько 10 МГц, тоді як для ЕМІ він більше — 100 МГц. Пікове значення ЕМІ може досягти 50 000 Вт/хв, що дорівнює всій енергії яка випромінюється в радіочастотній частині спектра.

Ефективність і безпека роботи різноманітних науково-виробничих комплексів (НВК) багато в чому досягається завдяки застосуванню засобів інформатизації в різних сферах їх діяльності. Основну загрозу для інформаційних ресурсів до недавнього часу представляли хакерські атаки і впровадження комп'ютерних вірусів, які здійснюються програмним шляхом. Проте наявність генераторів потужних електромагнітних випромінювань, здатних негативно впливати на електронне устаткування, істотно змінило

пріоритети в області інформаційної безпеки. Сьогодні, в розряд першочергових, висувається проблема захисту від навмисного електромагнітного впливу.

Джерела електромагнітних імпульсів розробляються в ряді країн з метою досягнення якісно нового рівня радіолокації, радіозв'язку, технологій вирішення інших технічних завдань. Принцип їх роботи допускає генерацію і випромінювання в навколишній простір не лише одиничних електромагнітних сигналів, але і цілих пакетів. Параметри випромінювання таких пристроїв роблять їх дуже небезпечними при дії на мікроелектронні системи. Відносна простота виготовлення і доступність придбання таких генераторів, а також компактність цих приладів дозволяють розцінювати їх в якості потенційних засобів навмисного впливу, що дозволяють локально створювати ефекти, подібні до електромагнітних випромінювань ядерного вибуху.

Існує широка номенклатура генераторів, що формують електромагнітні імпульси, які призначені для перевірки стійкості електронного устаткування різних об'єктів до електромагнітного впливу. Характер порушень в роботі безпосередньо залежить від параметрів і рівня стійкості устаткування до цього впливу. Порушення в основному носять тимчасовий характер, проявляються під час впливу і зберігаються впродовж деякого періоду після цього впливу, причому виявити факт навмисного електромагнітного впливу, як в цей період, так і надалі є не можливим.

Електронна інфраструктура НВК, ставши об'єктом електромагнітної атаки, може зазнати ряд деструктивних змін, що, у свою чергу, приведе до збоїв в роботі електронного устаткування і далі – до функціональних порушень видів діяльності, що ним забезпечується.

Систематизований перелік можливих наслідків для засобів інформатизації, використовуваних в основних сферах діяльності, властивих об'єктам НВК, приведений в табл. 6.1.

Оскільки завади, що мають меншу енергію, виникають частіше ніж завади, що мають велику енергію, найбільш частою реакцією електронних систем на дію

електромагнітних завад буде не руйнування пристрою, а порушення його роботи або короткочасний збій в роботі з наступним відновленням порушеної функції.

Таблиця 6.1 – Деструктивні ефекти в електронному устаткуванні при навмисному електромагнітному впливі (НЕМВ)

Вид системи	Вид ефекту в результаті дії
Засоби телекомунікації	<ul style="list-style-type: none"> - зависання і перезавантаження комп'ютерів, - значне зниження інформаційного трафіку, - збільшення кількості помилок.
Засоби зв'язку і навігації	<ul style="list-style-type: none"> - зменшення ефективної дальності зв'язку (від 2 до 10 разів), - неправдиві свідчення, або збої в засобах навігації.
Засоби безпеки	<ul style="list-style-type: none"> - збої в системах контролю і управління доступом, - блокування охоронно-пожежної сигналізації, - мимовільне включення устаткування пожежогасіння, - спотворення зображень з камер відео нагляду.

Імпульсні перенапруження, що виникають при ядерних вибухах, розрядах блискавок і при комутації в силових електроустановках, здатні пошкодити, або зруйнувати як електронні прилади, так і цілі системи.

У нормах будівництва громовідводів приймають зазвичай струм блискавки до 200 тисяч ампер при тривалості близько 1 мс, хоча практично струм блискавки рідко перевищує 20-30 кА. Температура каналу при головному розряді може перевищувати 25000° С. Довжина каналу блискавки може бути від 1 до 10 км, діаметр — декілька сантиметрів. При ударі блискавки в громовідвід електричний струм (імпульс у формі дзвона) поступає в землю і розтікається в ґрунті на всі боки до декількох десятків і навіть сотень метрів, причому через опір ґрунту цей струм створює на ньому падіння напруги. Оскільки найбільший опір чинять шари ґрунту, що лежать поблизу місця входження струму в землю, то саме тут

спостерігається найвища напруга. В міру віддалення від цієї точки опір проходженню струму зменшується, при цьому знижується і напруга.

Ще один шлях для проникнення завади від ЕМІ різної природи – протікання струмів по заземленому металевому корпусу і заземлених екранах кабелів, підключених до нього. Отже, забезпечити належний рівень захисту від електромагнітних завад електронної апаратури дуже непросто. Особливо складно це зробити на старих підстанціях, системи заземлення яких проектувалися для роботи з електромеханічним захистом, значно стійкішим до електромагнітних дій, ніж мікропроцесорні. А якщо врахувати, що небезпечні підйоми потенціалу в колах заземлення виникають також при аварійних коротких замиканнях в електричних мережах, то проблема стане ще складнішою.

В деяких випадках для запобігання такому підйому потенціалу в кола електронної апаратури контури заземлення силового устаткування і електронної апаратури роблять окремими. Проте на реально існуючих підстанціях виконати таке розділення нереально.

Таким чином тільки комплексне рішення проблеми дозволить уникнути впливу потужних електромагнітних завад. Це рішення повинне включати:

- використання мікропроцесорних реле захисту тільки на підстанціях, спроектованих і побудованих з урахуванням найсучасніших вимог до електромагнітної сумісності і розрахованих на експлуатацію високочутливої електронної апаратури;
- вдосконалення конструкції самих мікропроцесорних реле захисту;
- розміщення мікропроцесорних реле захисту в металевих шафах, спеціально призначених для захисту електронного устаткування і забезпечених фільтрами на усіх кабелях, що входять в шафу.

6.3 Загальні вимоги законодавства з охорони праці в галузі інформаційних технологій

В цьому розділі розглядаються норми та заходи з охорони праці й техніки безпеки для комп'ютерної лабораторії, які будуть скеровані на усунення потенційно шкідливих і небезпечних виробничих факторів, що при певних умовах можуть негативно вплинути на організм працівників в процесі використання результатів даної дипломної роботи [21].

Правову основу охорони праці становить Конституція України як за своїми юридичними особливостями, так і своїми принципами, тобто юридично вираженими об'єктивними закономірностями організації і функції соціально-економічної, політичної, духовної сфер суспільства, правового положення особи.

Конституційні норми, з одного боку, закладають суть безпеки (норми-принципи), а з іншого, – вказують на цілі подальшого розвитку і реалізацію правового забезпечення безпеки життєдіяльності (норми-програми, норми-завдання, норми-зобов'язання).

Реалізація і розвиток основних конституційних положень, які регламентують суспільні правовідносини, безпосередніми суб'єктами яких є особа і держава, здійснюється за допомогою як чинних фундаментальних нормативно-правових актів, так і спеціальних (Кодексів України про працю, Закону "Про охорону навколишнього природного середовища" та ін.)

Поруч з нормативними актами, які прийняті вищим законодавчим органом держави, для встановлення взаємозв'язків, усунення програм, а в ряді випадків і реалізації окремих правових норм або їх елементів, до правової бази безпеки життєдіяльності належать спеціальні акти, розроблені за дорученням виконавчих державних органів усіх рівнів (Кабінет Міністрів, Міністерства, Державні Комітети та ін.).

Так, наприклад, "Положення...", які розвивають Закон України Про охорону праці, діляться на звичайні "Положення" і "Типові положення". Тут держава розподілила питання своєї прерогативи стосовно розробки нормативних

актів і прерогативи своїх повноважень стосовно контролю, "правового простору" у вигляді нормативних актів підприємств.

З іншого боку, формуючи систему "Типових положень" держава на сьогоднішній день ліквідує прогалини в чинному законодавстві, узгоджує взаємозв'язки між суб'єктами правовідносин, створює юридичну базу для удосконалення і розвинення "правового поля" підприємств.

Кожний нормативно-правовий документ має свою структуру, яка визначає собою ідею систематизації відповідно зі своїм рівнем, метою та завданнями. Відповідно до цього в кожному нормативному акті є елементи, що відповідальні за зовнішній його зв'язок і створення передумов для відповідного розвинення за рахунок розробки нижчих нормативно-законодавчих актів. Сама структура нормативного акту формує відповідні внутрішні зв'язки.

Основними систематизуючими ланками нормативних актів безпеки життєдіяльності (які за ієрархією знаходяться нижче законів) є встановлення взаємовідносин в галузі виробництва, в межах дії небезпечного фактора (в тому числі і факторів довкілля), а також відносно управління основних технологій безпеки життєдіяльності (розслідування нещасних випадків, навчання, організації робіт та ін.).

Узагальнюючими ланками систематизації на рівні держави є національна ідея, взаємовідносини в суспільстві, соціально-економічне і політичне становище держави, можливості сприймання і використання законодавчих актів з боку споживачів та ін.

6.4 Аналіз умов праці розробника програмного забезпечення

6.4.1 Загальна характеристика умов праці

В приміщенні бухгалтерії підприємства є місця для роботи трьох чоловік. Розміри приміщення наведені у таблиці 6.2.

Згідно СН-245-71, на одного працюючого об'єм приміщення повинен складати не менше 19,5 м², а площа – не менше 6 м².

Число працюючих у приміщенні $N_p=3$.

Таким чином, на кожного працюючого виходить приблизно $22,6 \text{ м}^3$ і $6,6 \text{ м}^2$, отже, усі вимоги тут дотримані.

Таблиця 6.2 – Розміри приміщення

l	довжина	5 м
d	ширина	4 м
h	висота	3,4 м
S_0	площа	20 м^2
V_0	об'єм	68 м^3

Далі, відповідно до норм, повинні дотримуватися:

- ширина основних проходів, не менше: – 1200 мм
- ширина допоміжних проходів, не менше: – 700 мм
- відстань між двома столами, якщо між ними є стілець, не менше: – 1300 мм

У розглянутому приміщенні:

- ширина основних проходів: – 2000 мм
- відстань між двома комп'ютерами у ряді: – 1500 мм

Отже, норми виконуються.

6.4.2 Повітряне середовище

Шляхом провітрювання і центральної системи опалення у приміщенні бухгалтерії завжди підтримується:

- стабільна температура повітря, що становить $25 \text{ }^\circ\text{C}$;
- відносна вологість повітря 55 %.

При зниженні тиску погіршується відвід тепла від елементів ЕОМ, знижуються ізоляційні властивості повітря. Показники об'єму і площі приміщення на одного працюючого відповідають нормативним значенням.

Роботи, що проводяться в бухгалтерії відносяться до легких фізичних робіт групи 1а, відповідно до ГОСТ 12.1.005-88, тому що вони проходять сидячи і не вимагають фізичного навантаження, проводяться при нормальних метеорологічних умовах і не викликають забруднення одягу і рук. Витрати енергії не перевищують 172 Дж/с (155 Ккал/год).

У таблиці 6.3 і 6.4 наведені норми температури, відносної вологості і швидкості руху повітря на робочих місцях відповідно ГОСТ 12.1.005-88, що встановлює норми виробничого мікроклімату. Дані приведені для приміщень з незначним надлишком явного тепла (до 20 Ккал/год м³) для виконання легких робіт.

Таблиця 6.3 – Норми температури, відносної вологості і швидкості руху повітря на постійних робочих місцях

Період року	Норми	Температура повітря t, °С	Відносна вологість, %	Швидкість руху повітря, м/с
холодний	оптим.	20-22	30-60	менше ніж 0,2
	доп.	17-22	менше 75	менше ніж 0,3
теплий	оптим.	20-25	30-60	0,2-0,3
	доп.	менше ніж 28	менше ніж 80*	0,3-0,5
* - у теплий період року припустима відносна вологість повітря для всіх приміщень і категорій робіт має значення, приведені в таблиці 6.4				

Таблиця 6.4 - Відносна вологість повітря в теплий період року

Температура повітря,	2	2	2	2	2	<
Відносна вологість, %	>	6	6	7	7	7

Основними джерелами тепла в приміщенні є:

- сонячна радіація;
- система опалення;
- люди, що працюють у приміщенні;
- устаткування.

У таблиці 6.5 наведені дані вимірювання в приміщенні ОЦ у місяці грудні.

Таблиця 6.5 – Результати виміру параметрів мікроклімату в приміщенні бухгалтерії

Температура повітря t, °С	20 – 25
Відносна вологість, %	50 – 60
Швидкість руху повітря, м/с	0,2

Як видно з таблиці 6.5, у розглянутому приміщенні значення параметрів мікроклімату відповідають нормативним. Стабільність цих параметрів підтримується загальною системою утеплення і кондиціонування повітря.

6.4.3 Освітлення

У приміщенні бухгалтерії використовується природне і штучне освітлення. Природне освітлення здійснюється за допомогою двох вікон загальною площею $S=7,5 \text{ м}^2$, що забезпечує коефіцієнт природної освітленості $E=1,5\%$. Це відповідає СНиП І-4-79.

Штучне освітлення в бухгалтерії здійснюється системою загального рівномірного освітлення, що реалізована на основі люмінесцентних ламп типу ЛДЦ-40-1, які мають наступні параметри:

- висока світловіддача;
- тривалий термін служби;
- мала яскравість освітлювальної поверхні;
- близькість спеціального складу до природного освітлення.

Робота за монітором ПЕОМ по розряду зорових робіт відноситься до III типу (роботи високої точності з розміром об'єкта 0,2–0,4 мм). При загальному освітленні, освітленість робочого місця повинна складати від 200 до 400 лк.

При штучному освітленні нормуються наступні параметри:

- E (лк) – найменша припустима освітленість;
- M – показник дискомфорту;
- Kn (%) – коефіцієнт пульсації освітлення.

Перевіримо відповідність фактичних параметрів штучного освітлення в приміщенні нормам. Номінальний світловий потік лампи білого свічення ЛДЦ-40-1: $\Phi_l = 3120$ лм.

У приміщенні застосовуються світильники, у яких встановлені дві лампи. Висоту підвіски світильника визначимо по формулі:

$$h = H - h_C - h_P - h_{II}, \quad (6.1)$$

де:

H – висота приміщення, м.;

h_C – висота світильника, м.;

h_{II} – відстань від стелі до підвіски, м.;

h_P – висота робочої поверхні, м.;

Для розглянутого приміщення:

$H = 3,4$ м,

$h_C = 0,15$ м,

$h_{II} = 0$ м, (підвісу немає)

$h_P = 0,8$ м.

Звідси $h = 3,4 - 0,15 - 0,8 = 2,45$ м.

Світильники розташовані в 2 ряди. Висота підвіски світильників складає 2,45 метра відносно підлоги, відстань між рядами 1 м, відстань від ряду до стіни 1,5 метра. Приміщення має наступні габарити:

- довжина $A = 5$ метрів;
- ширина $B = 4$ метрів.

Визначимо освітленість у робочій точці. Для розрахунку загальної рівномірної освітленості при горизонтальній робочій поверхні використовуємо метод коефіцієнта використання світлового потоку.

Розрахункова формула для світлового потоку світильника має такий вигляд:

$$\Phi_{л} = \frac{E \cdot K_{з} \cdot S \cdot Z}{N \cdot n}, \quad (6.2)$$

де N - кількість світильників у приміщенні, $N = 3 \cdot 2 = 6$;

n - коефіцієнт використання світлового потоку;

$\Phi_{л}$ - світловий потік ламп;

$K_{з}$ - коефіцієнт запасу, $K_{з} = 1,5$;

Z - коефіцієнт нерівномірності;

S - площа приміщення;

E - освітленість, створювана усіма світильниками.

Звідси одержуємо формулу для розрахунку освітленості на робочому місці :

$$E = \frac{\Phi_{л} \cdot N \cdot n}{K_{з} \cdot S \cdot Z} \quad (6.3)$$

Коефіцієнт використання світлового потоку залежить від:

- ККД кривої розподілу сили світла світильника;
- коефіцієнта відбивання стелі $R_{л}$ і стін $R_{с}$;
- висоти підвісу світильників $h_{л}$;
- показника приміщення i :

$$i = \frac{A \cdot B}{h \cdot (A + B)} \quad (6.4)$$

$$i = (5 \cdot 4) / (2,45 \cdot (5 + 4)) = 0,408$$

Стеля і стіни пофарбовані в білий колір.

Приймаємо:

$$R_{л} = 50\%$$

$$R_c = 30\%.$$

Звідси:

$$n = 31\%.$$

$$E = \frac{(3120 \cdot 2) \cdot 6 \cdot 0,31}{20 \cdot 1,1 \cdot 1,5} = 352_{\text{лк}}$$

Так як по розряду зорової роботи робота за дисплеєм ПЕОМ відноситься до III типу (високої точності, розмір об'єкта 0,2-14 мм), то при загальному висвітленні освітленість робочого місця повинна складати від 200 до 400 Лк, рекомендована освітленість при роботі з дисплеєм ПЕОМ складає 200 Лк, а при сполученні роботи з документами 400 Лк. Фактична освітленість на робочому місці складає 352 Лк.

Таким чином для роботи з дисплеєм цілком достатньо існуючих джерел світла, однак робота з документами повинна вестися при природному освітленні, або за допомогою додаткових місцевих джерел освітлення.

РОЗДІЛ 7 ЕКОЛОГІЯ

7.1 Радіоактивне забруднення довкілля та його моніторинг

У складній радіаційній ситуації, що виникла після вибуху у 1986 р. четвертого енергоблоку Чорнобильської АЕС, виникла необхідність в інформації щодо просторово-територіального розподілу радіоактивного забруднення [23].

Радіоекологічний моніторинг на Україні здійснюється з метою комплексної оцінки впливу нової техногенної радіогеохімічної провінції, що виникла в результаті Чорнобильської катастрофи, на екологічну ситуацію в зоні сильного радіоактивного забруднення і в Україні в цілому.

У структурі державної системи радіометричного контролю виділяються три функціональних типи радіоекологічного моніторингу: базовий (стандартний), кризовий (оперативний) і науковий (Чорнобильська..., 1996) [23].

Головними завданнями радіоекологічного моніторингу вважають:

- спостереження і контроль за станом забрудненої радіонуклідами зони відчуження, окремих, особливо небезпечних компонентів довкілля, і розробка заходів щодо зниження їхньої небезпеки;
- вивчення стану об'єктів природного середовища, згідно з параметрами, що характеризують радіаційну ситуацію у забрудненій зоні і за її межами;
- виявлення тенденцій зміни рівнів забруднення довкілля в зв'язку з функціонуванням екологічно небезпечних радіаційноядерних об'єктів і під час реалізації реабілітаційних заходів на забруднених територіях;
- аналіз тенденцій зміни показників здоров'я населення, що проживає на забруднених радіонуклідами територіях;

– геоінформаційне забезпечення прогнозу радіаційної ситуації у зоні відчуження і в Україні в цілому.

Радіоекологічний моніторинг в Україні здійснюється згідно з його науково-обґрунтованими розділами, які формують основні напрями досліджень: моніторингу ландшафтно-геологічного середовища з метою одержання базової інформації щодо оцінки і прогнозу загальної радіаційної ситуації на забруднених радіонуклідами територіях та їхнього впливу на екологічну обстановку; моніторингу стану ґрунтового покриву, рослинного і тваринного світу; моніторингу поверхневих, ґрунтових та підземних водних екосистем; моніторингу природоохоронних територій; моніторингу локальних джерел існуючого і потенційного радіоактивного забруднення; медичного й санітарно-гігієнічного моніторингу.

Радіоекологічний моніторинг ґрунтується на інформації, здобутої за допомогою методів дозиметричного контролю, які використовують з метою вивчення радіаційної обстановки на об'єктах природно-техногенного середовища.

На основі здобутих даних радіоекологічного моніторингу водних екосистем показано, що формування радіаційної обстановки в центральній частині України значною мірою зумовлюється процесами виносу радіоактивних речовин з водозбірних площ у водну мережу Дніпра і каскад його водоймищ.

До Чорнобильської катастрофи довкола АЕС не існувало спеціальної мережі радіогідрогеологічного моніторингу. Тому у 1986 р. для спостережень за станом радіоактивного забруднення підземних вод на ЧАЕС використовували малопридатні для цієї мети сільські колодязі і діючі водозабірні свердловини.

Мережу спостереження радіогідрогеологічного моніторингу через її технічну складність і високу вартість створювали протягом багатьох років з урахуванням досвіду, який накопичувався, та результатів радіоекологічних досліджень. Крім того, сучасна моніторингова мережа містить у собі раніш законсервовані пункти радіоекологічного контролю. Їхнє залучення у режимні

спостереження зумовлюється подальшим розширенням програми радіоекологічного моніторингу.

Сучасна мережа радіаційно-ядерного моніторингу охоплює усі найнебезпечніші у радіаційному відношенні об'єкти України. Головним її завданням є контроль за технічним станом ядерно- і радіаційнонебезпечних об'єктів й реалізація заходів щодо зниження ризику, а також оцінка та прогноз радіаційної ситуації в екосистемах довкола цих об'єктів.

7.2 Статистичне групування в екології

У системі статистичних методів групування займають особливе місце. Це пояснюється тим, що на відміну від інших методів групування виконує дві функції: по-перше, розподіляє сукупності на однорідні групи, а по-друге, визначає межі і можливості застосування інших статистичних методів (середніх величин, кореляційно-регресійного тощо). Використання цих методів потребує однорідності статистичних сукупностей, що зумовлює статистичне групування як важливий етап обробки статистичних даних.

Групування – це статистичний метод розмежування складного масового явища на істотно різні групи однорідні в тому чи іншому розумінні з метою всебічної характеристики його стану, розвитку і взаємозв'язків [24].

Суть методу статистичних групувань полягає у тому, що складне масове явище розглядається не як єдине нероздільне ціле, а в ньому виділяються окремі групи одиниць із статистичними показниками, які дають кількісну характеристику якісно своєрідній частині одиниць усієї сукупності. Тобто кожна з одержаних груп об'єднує однорідні одиниці сукупності.

Принципове значення при побудові групувань має вибір групувальної ознаки, на основі якої виділяють різні типи, групи і підгрупи. За групувальну приймають найістотнішу ознаку яка може бути атрибутивною (якісною) або кількісною. Залежно від складності явища та мети дослідження групувальних

ознак може бути декілька. Якщо групування здійснюється за атрибутивною ознакою, то виділяють стільки груп, скільки є найменувань ознаки.

Групування за атрибутивною ознакою називають класифікацією або номенклатурою. Класифікація у статистиці – це систематизований розподіл явищ та об'єктів на певні групи, класи, розряди на підставі їх збігу або різниці. Різновидом класифікацій є товарні номенклатури як стандартизований перелік об'єктів та груп.

Структурні групування характеризують склад однорідної сукупності за будь-якою ознакою. З допомогою таких групувань аналізують структуру сукупності і структурні зрушення в розвитку екологічних явищ і процесів. Потреба в таких групуваннях виникає тому, що однорідність явищ, елементів, з яких складається статистична сукупність, ще не означає їх тотожності. У межах однорідної сукупності елементи відрізняються один від одного, числові значення властивих їм ознак варіюють.

Аналітичні групування спрямовані на виявлення зв'язку між окремими ознаками явища, що вивчається. Вони проводяться за факторною ознакою і в кожній групі визначається середня величина результативної ознаки. Взаємозв'язок проявляється у систематичній зміні результативної ознаки у зв'язку зі зміною факторної ознаки. При наявності зв'язку між ознаками середні групові систематично збільшуються (прямий зв'язок) або зменшуються (зворотний зв'язок).

Факторне групування – це групування, яке проводиться за факторною ознакою, тобто ознакою, яка впливає на інші ознаки. Групування за факторними ознаками дає змогу показати різноманітність виникаючих форм і ступінь впливу того чи іншого фактора на результативні показники. За допомогою факторних групувань встановлюються і вивчаються причинно-наслідкові зв'язки між ознаками однорідних явищ, виявляються фактори розвитку сукупності та ефект впливу фактора на результат (різниця між груповими середніми результативної ознаки). Зокрема, це стосується

аналітичних групувань, хоча аналітичні функції притаманні типологічним і структурним групуванням.

Результативне групування – це групування, яке проводиться за результативною ознакою, тобто ознакою, яка є залежною від факторних ознак. Групування за результативними ознаками дають можливість досить надійно виділити виробничі типи і дати в середньому характеристику їх особливостям. Але за ними не можна виділити всю різноманітність форм і показати ступінь впливу того чи іншого фактора на результат виробництва.

При статистичному групуванні велике пізнавальне значення має поєднання факторних і результативних ознак. У такому разі будуються комбінаційні групування за формою факторно-результативних або результативно-факторних. Тобто одна з групувальних ознак є факторною, друга – результативною. Вибір схеми "факторно-результативна" чи "результативно-факторна" залежить від мети дослідження, знання природи екологічних явищ і досконалого володіння методикою статистичних групувань.

Комбінаційне групування – це групування, яке проводиться за двома і більше групувальними ознаками. У комбінаційних групуваннях групи з однією ознакою поділяються на підгрупи за іншою ознакою.

Комбінаційне групування має більш широкі аналітичні можливості, ніж просте, його використовують переважно для вивчення взаємозв'язків між ознаками. Порядок комбінації ознак обґрунтовується економічно і може бути легко змінений (при необхідності). Якщо по кожній з ознак є підсумкова група, комбінаційне групування можна "згорнути" у будь-якому напрямі в просте.

ВИСНОВКИ

Отже, в міру розвитку і поширення Інтернету в сучасному суспільстві інтернет-середовище поступово перетворюється в універсальний комунікативний простір, який використовується, як механізми ведення бізнесу та побутового спілкування. Більш того, розвиток Інтернету призводить до посилення міжособистісних і міжгрупових комунікацій, але що ще більш важливо – створює можливість зміни потоку цих комунікацій і таким чином трансформує соціальну структуру суспільства.

Ці всі перспективи зрушують розвиток нових методів та технології комунікації в мережі. Виникають потреби в об'єднанні різних мереж та середовищ. Оскільки в наш час з'явилася величезна кількість способів обмінюватися голосовими і текстовими повідомленнями: класичні телефонія і електронна пошта, ICQ, Skype, ВКонтакте, Facebook, Viber, WhatsApp. З'являються незручності в зв'язуванні з великим колом осіб, для цього людина повинна мати купу акантів та номерів і своєчасно їх відслідковувати. Рішеннями таких незручностей стає інтегрування одних мереж з іншими, більш глобальними та продуктивними. Така тенденція підтримує та підкреслює особливості сервісів та технологій, які втрачають актуальність в комунікації.

Все прогресивне і нове швидко підхоплюється компаніями в усьому світі, а старі технології забуваються. Фахівці вважають, що розвиток сервісів телефонії в мережі буде набирати темпи найближчим часом. Приміром, уже зараз розвиваються програмні шлюзи для зв'язку між мобільними GSM-мережами та корпоративної АТС певної компанії. А незабаром, можна буде втілювати глобальний роумінг із застосуванням IP-зв'язку для мобільних операторів. А це означає, що мобільні технології вийдуть на зовсім інший етап розвитку і глобальна мережа займе центральне місце цього розвитку. Мобільним зв'язком зможуть користуватися абсолютна більшість людей. Економія на телефонних переговорах буде дуже відчутною.

У даній роботі були розглянуті базові принципи розробки мережеских додатків в Java. Хочеться відзначити, що технологія Java розвивається дуже динамічно. З моменту своєї появи мову Java опинилася на передньому рубежі програмування для Інтернет. Кожна наступна версія ще більше зміцнювала його позиції. На сьогоднішній день Java залишається основним і найкращим засобом розробки Web-додатків. Одна з причин успіху Java – його швидка мінливість. Ця мова швидко адаптується до змін у середовищі програмування і в підходах до програмування. І найголовніше, вона не просто слідує тенденціям, а допомагає їх створювати. Здатність пристосування Java до високої швидкості змін в комп'ютерному світі – основна складова причини того, що він залишається на передових рубежах проектування комп'ютерних мов.

У ході виконання роботи по даній темі було отримано уявлення про мережеску архітектуру передачі даних і про роль цих технологій. Були виконані експериментальні напрацювання клієнт-серверної системи, що базується на Java технологіях. Розроблений спеціальний алгоритм передачі даних з різних мереж між системами з різним операційним середовищем та архітектурою. Розроблена технологія дає змогу створити єдине комунікаційне середовище, яке не залежить від природи мереж, пристроїв комунікації та операторів. На прикладі розробленого додатку реалізована технологія виконує переадресацію безлічі телефонних дзвінків через глобальну мережу на єдиний пристрій користувача.

Дана робота може бути використана як основа для подальшого вивчення комунікаційних технологій, об'єднаних концепцією взаємодії користувачів за допомогою глобальних мереж.

ПЕРЕЛІК ДЖЕРЕЛ

1. Беллами Дж. Цифровая телефония. - М.: Радио и связь, 1986. - 245с.
2. Левин А. Android на планшетах и смартфонах; Питер - Москва, 2013. - 224 с.
3. Ванштейн Г. Интернет как фактор общественных трансформаций. - МЭиМО. - 2002. - 70с.
4. Протоколы Internet. С. Золотов. — СПб.: ВHV — Санкт-Петербург, 1998. - 340с.
5. Олифер В. Г., Олифер Н. А. Олифер. Компьютерные сети. Принципы, технологии, протоколы. – СПб: Издательство “Питер”, 1999. – 672с.
6. Гурін М.І. Робота на персональному комп'ютері: довід. посібник. - Мн.: Білорусь, 1994.-224с.
7. Дуглас Камер, Дэвид Л. Стивенс, Сети TCP/IP, том 3. Разработка приложений типа клиент/сервер, издательство «Вильямс», 2002. - 592с.
8. Коноплицкий С. Интернет як предмет соціологічного аналізу. Соціологія: теорія, методи, маркетинг. - 2005. – 188с.
9. Вязилов Е.Д.: Архитектура, методы и средства интернет-технологий. - М.: КРАСАНД, 2009. -215с.
10. Почепцов Г.Г. Теория коммуникации. - К.: Київський університет, 1999. - 308с.
11. Коммутация и маршрутизация IP/IPX трафика / М. В. Кульгин, АйТи. - М.: Компьютер-пресс, 1998. - 69с.
12. Голощапов Алексей Google Android. Программирование для мобильных устройств; БХВ-Петербург - Москва, 2012. - 448 с.
13. Кульгин М. В. Коммутация и маршрутизация IP/IPX – трафика. – М.: КомпьютерПресс, 1998. - 320с.
14. Медникс Зигард , Дорнин Лайрд , Мик Блэйк , Накамура Масуми Программирование под Android; Питер - Москва, 2013. - 560 с.

15. Практическая передача данных: Модемы, сети и протоколы / Ф. Дженнингс; пер. с англ. - М.: Мир, 1989.- 250с.
16. Программирование сокетов: [Электронный ресурс].- Режим доступа:<http://www.quizful.net/post/java-socket-programming>
17. Основы TCP/IP та UDP: [Электронный ресурс].- Режим доступа:http://www.network.xsp.ru/3_5.php
18. Принципы работы в GSM: [Электронный ресурс]. - Режим доступа: <http://www.iec.org/online/tutorials/gsm/www.iec.org>
- 19.Создание сетевых приложений: [Электронный ресурс]. - Режим доступа:<http://www.studfiles.ru/preview/1188507/>
- 20.Потоки ввода-вывода: [Электронный ресурс].- Режим доступа:<http://metanit.com/java/tutorial/6.1.php>
21. Основы охраны труда: учебник / А. С. Касьян, А. И. Касьян, С. П. Дмитрюк. – Дн-ськ: Журфонд, 2007. – 494 с.
22. Безпека життєдіяльності: Навч. посібник./ За ред. В.Г. Цапка. 4–те вид., перероб. і доп. – К.: Знання, 2006. – 397 с.
23. Основы екології та охорона навколишнього природного середовища [Текст]: навч. посіб. для студ. вищих навч. закладів / [Бедрій Я. І.; Джигирей В. С.; Кидисюк, А. І. та ін.]; за ред. В. С. Джигирей ; Український держ. лісотехнічний ун-т, Львівський електротехнікум зв'язку. - Л. : [б.в.], 1999. - 239 с. Альтернативна назва : Екологія та охорона природи.
24. Тарасова В.В. Екологічна статистика (з блочно-модульною формою контролю знань). Підручник. – К.: Центр учбової літератури, 2008. – 392 с.

ДОДАТКИ

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національна академія наук України

Тернопільський національний технічний університет імені Івана Пулюя (Україна)

Західний науковий центр НАН України

Університет імені П'єра і Марії Кюрі Сорбона Париж (Франція)

Вища школа промислової фізики і хімії міста Париж (Франція)

Технічний університет у Кошице (Словаччина)

Вільнюський технічний університет ім. Гедімінаса (Литва)

Шяуляйська державна колегія (Литва)

Жешувський політехнічний університет ім. Лукачевича (Польща)

Білоруський національний технічний університет (Республіка Білорусь)

Міжнародний університет цивільної авіації (Марокко)

Національний університет біоресурсів і природокористування України

Наукове товариство імені Шевченка

Науковий парк «Інноваційно-інвестиційний кластер Тернопілля»

Асоціація випускників ТНТУ

**Матеріали Міжнародної науково-технічної конференції
ФУНДАМЕНТАЛЬНІ ТА ПРИКЛАДНІ
ПРОБЛЕМИ СУЧАСНИХ ТЕХНОЛОГІЙ**

до 60 річчя з дня заснування

Тернопільського національного технічного університету

імені Івана Пулюя

та 175 річчя з дня народження Івана Пулюя

14–15 травня 2020 року



Тернопіль

2020

П.Д. Кривий, канд. техн. наук, доц., Н.М. Тимошенко, канд. фіз.-мат. наук, доц., В.О. Дзюра, канд. техн. наук, доц., В.Р. Кобельник, канд. техн. наук	132
УТОЧНЕНИЙ МЕТОД АПРІОРНО-ЕМПІРИЧНИХ ФУНКЦІЙ ВИЗНАЧЕННЯ ЗАКОНУ РОЗПОДІЛУ ТА ЙОГО ХАРАКТЕРИСТИК НА ОСНОВІ МАЛОЇ ВИБІРКИ	132
А. Курко, канд. техн. наук, доц., В. Карегін	134
МОДЕЛЮВАННЯ КІНЕМАТИКИ СФЕРИЧНОГО РУХУ ГЕОМЕТРИЧНИМИ ЗАСОБАМИ	134
В.П. Марценюк, д-р. техн. наук, проф., І.Є. Андрушак, д-р. техн. наук, проф., А.С. Сверстюк, канд. техн. наук, доц., В.О. Паничев.....	135
ПІДХІД ДО ІДЕНТИФІКАЦІЇ ПАРАМЕТРІВ SIR-МОДЕЛІВ ЗА РЕЗУЛЬТАТАМИ АНАЛІЗУ ПАНДЕМІЇ COVID-19 В ТЕРНОПІЛЬСЬКІЙ ОБЛАСТІ	135
Ю.Б. Паляниця канд. техн. наук, І.О. Гринюк, Б.Р. Савка	137
ДОСЛІДЖЕННЯ ОПОРУ ТКАНИН ТІЛА ЛЮДИНИ ЯК ЛІНІЙНОЇ ДИНАМІЧНОЇ СИСТЕМИ ТА СИНТЕЗ ЇЇ ПАРАМЕТРІВ ДЛЯ ЗАДАЧ ПРОЕКТУВАННЯ БІОМЕДИЧНОЇ ТЕХНІКИ В ГАЛУЗІ БІОМЕДИЧНОЇ ІНЖЕНЕРІЇ	137
Т.М. Приліпко, д-р. с.-г. наук, проф.	139
МОДЕЛЮВАННЯ МІКРОБІОЛОГІЧНИХ ТА БІОХІМІЧНИХ ПРОЦЕСІВ ПІД ЧАС ОБРОБКИ М'ЯСНИХ КОНСЕРВІВ ЗА УМОВИ ПАРОКОНТАКТНОГО НАГРІВУ ПРОДУКЦІЇ В ТАРІ	139
Т.М. Приліпко, д-р. с.-г. наук, проф., Н.В. Букалова, канд. вет. наук, доц., Л.М. Богатко, канд. вет. наук, доц., Лясота В.П., д-р. вет. наук, проф.....	143
МОДЕЛЮВАННЯ ЯКОСТІ МОЛОЧНОЇ СИРОВИНИ	143
Л. Романюк, канд. техн. наук, доц., І. Чихіра, канд. техн. наук, доц.....	147
МЕТОД DUBINS ДЛЯ АЕРОДИНАМІЧНОЇ МОДЕЛІ ГРУПИ БЕЗПЛОТНИХ ЛІТАЛЬНИХ АПАРАТІВ	147
С.І. Федак, канд. техн. наук, доц., О.П. Ясній, д-р. техн. наук, проф., І.С. Дідич	149
ВИЗНАЧЕННЯ НАПРУЖЕНЬ ІНІЦІАЦІЇ ПЕРЕРИВЧАСТОЇ ТЕКУЧОСТІ СПЛАВУ АМГ6	149
Секція: КОМП'ЮТЕРНО-ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА СИСТЕМИ ЗВ'ЯЗКУ	150
В.П. Бугай	150
ПРИНЦИПИ ПОБУДОВИ МЕТОДИКИ ОЦІНЮВАННЯ ЯКОСТІ АРХІТЕКТУРИ ПРОГРАМНИХ СИСТЕМ.....	150
В.П. Бугай, І.О. Боднарчук, канд. техн. наук, доц.	152
ОЦІНЮВАННЯ ПРОГРАМНОЇ АРХІТЕКТУРИ ПРИ ГНУЧКИХ МЕТОДАХ РОЗРОБКИ ПРОГРАМНИХ СИСТЕМ	152
О.В. Головацька, Я.В. Литвиненко, д-р. техн. наук, доц.....	154
РОЗВИТОК ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ В УКРАЇНІ	154

**Секція: КОМПЮТЕРНО-ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА СИСТЕМИ
ЗВ'ЯЗКУ**

Голови: д-р. техн. наук, проф. Б.І. Явоський, д-р. техн. наук, проф. М.В. Приймак,
д-р. техн. наук, проф. М.І. Паламар, д-р. техн. наук, проф. С.А. Лупенко

Вчений секретар: канд. техн. наук Р.З. Золотий

УДК 004.4

В.П. Бугай

Тернопільський національний технічний університет імені Івана Пулюя

**ПРИНЦИПИ ПОБУДОВИ МЕТОДИКИ ОЦІНЮВАННЯ ЯКОСТІ
АРХІТЕКТУРИ ПРОГРАМНИХ СИСТЕМ**

V. Buhai

**PRINCIPLES OF METHOD ELABORATION FOR SOFTWARE SYSTEMS
ARCHITECTURE QUALITY ASSESSMENT**

Для забезпечення якості при проектуванні програмних систем (ПС) важливим є процес управління якістю на стадіях життєвого циклу.

Одним з перших етапів в процесі проектування ПС є розробка архітектури. На сьогодні найпоширенішим способом для цього є використання патернів (шаблонів) проектування. В результаті матимемо декілька архітектур, серед яких потрібно вибрати оптимальний. Вибір здійснюється на основі критеріїв якості. Тому виникає задача вибору критеріїв, на основі яких можна буде оцінити якість архітектури.

Оцінювання якості архітектури – специфічний процес, оскільки це лише початкова стадія реалізації проекту і результатом її завершення буде лише концепція структурної побудови ПС. Спробуємо вибрати показники якості архітектури, проаналізувавши основні праці на цю тематику та узагальнивши їх. Далі, отримавши цей перелік, можна буде за допомогою методу QFD визначити, яким саме чином показники якості у використанні корелюють з показниками якості архітектури.

Почнемо з аналізу роботи Дж. Гарланда та Р. Ентоні. Тут для оцінки якості архітектури пропонується використовувати наступний перелік характеристик, підхарактеристик та атрибутів [0]: культурна сумісність, безпека в розумінні запобігання несанкціонованому доступу, цілісність даних, характеристики супроводжуваності, тестованість, зручність використання та ряд показників системи в процесі її роботи, характеристики процесів взаємодії з користувачем та безпека з розумінні співіснування з середовищем виконання.

Виконаємо аналогічний аналіз показників якості архітектури, які пропонуються корпорацією Microsoft [0]. Тут всі показники якості розбиті на чотири групи, при чому йдеться не про якість архітектури, а про якість архітектури та дизайну програмної системи. Спочатку перерахуємо всі характеристики і підхарактеристики якості системи згідно [1] по групах. Тут виділяються показники якості дизайну (концептуальна цілісність, зручність та простота обслуговування, можливість повторного використання), якість часу виконання (доступність, можливість взаємодії, керованість, продуктивність, масштабованість, безпека), якість системи (забезпечення технічною підтримкою, тестованість) та якість взаємодії з користувачем як характеристика зручності та простоти використання.

Для корпорації Microsoft можна відмітити, що в різних контекстах використовуються різні показники якості архітектури. Але можна зустріти зовсім інший перелік характеристик якості архітектури у [0]. Також в контексті оцінювання архітектури в [0] використовується стандарт якості ISO/IEC-9126 [0].

Звичайно, можна скористатись також іншими роботами, які стосуються розробки та оцінювання якості архітектури програмного забезпечення. Тут можна навести приклад роботи Л. Басса [0], де автор пропонує виконувати оцінювання архітектури на основі сценаріїв. Аналогічні підходи згадані і в [0]. Е. Брауде відображає свій підхід в роботі [0] до даної проблеми, де він оперує поняттями зчеплення та зв'язності, рядом комерційних характеристик, а також поняттям концептуальної цілісності архітектури. Саме це поняття якості архітектури (концептуальна цілісність) дуже часто згадується, коли мова йде про архітектуру програмних систем. Зокрема, колективом авторів в роботі [0] наголошується, що забезпечення концептуальної цілісності здійснюється через використання патернів (зразків) проектування.

Така велика кількість підходів до оцінювання якості приводить нас до висновку, що є потреба використання єдиного підходу до оцінювання якості архітектури на основі стандартів. Для цього можна скористатись як вже згаданим стандартом ISO/IEC-9126 [0], так і новим ISO/IEC-25010 [0]. Мова йде про формулювання вимог якості до архітектури в термінах згаданих стандартів. Зрозуміло, що не всі характеристики якості, взяті зі стандартів, мають безпосереднє відношення до архітектури. З іншого боку під час аналізу методик оцінювання архітектури зустрічаються спільні показники в різних джерелах, але вони відсутні у стандарті. Тому загальний перелік характеристик якості міститиме тільки ті з них, що стосуються архітектури, а відправним пунктом для оцінювання її якості є специфікація вимог користувача до проектованої ПС.

Отже, першим кроком виберемо з усіх запропонованих показників якості архітектури спільні для різних джерел та виразимо їх по можливості в термінах стандарту [0], а потім з використанням методу QFD спроекуємо вимоги користувача до ПС, сформульованих згідно стандарту [0], на отримані перед цим атрибути якості і, застосувавши порогове значення коефіцієнту "важливості" атрибутів, залишимо тільки ті, що мають найвагомий вплив на якість архітектури.

Література

1. Руководство Microsoft по проектированию архитектуры приложений, 2-е издание. Microsoft. – 2009. – 527 с.
2. Jeff Garland, Richard Anthony. Large-Scale Software Architecture: A Practical Guide using UML. John Wiley & Sons, Inc. New York, NY, USA © 2002.
3. Bass, Len. Software architecture in practice / Len Bass, Paul Clements, Rick Kazman – 2nd ed.
4. Брауде Э. Технология разработки программного обеспечения. – СПб.: Питер, 2004. – 655 с.: ил.
5. Приемы объектно-ориентированного проектирования. Паттерны проектирования : [пер. с англ.] / Э. Гамма [и др.]. – СПб. [и др.] : Питер., 2010. – 366 с.
6. Gabriel Morgan. Implementing System-Quality Attributes. url: [http://msdn.microsoft.com/en-us/library/bb402962.aspx]
7. Hui Zhu. Quality-Attribute Auditing: The What, Why, and How. url: [http://msdn.microsoft.com/en-us/library/bb508961.aspx]
8. ISO/IEC 9126. (1-4) Software Engineering-Product Quality. 2001-2004
9. ISO/IEC 25010:2011. Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – System and software quality models.

УДК 004.4

В.П. Бугай, І.О. Боднарчук, канд. техн.наук, доц.

Тернопільський національний технічний університет імені Івана Пулюя

ОЦІНЮВАННЯ ПРОГРАМНОЇ АРХІТЕКТУРИ ПРИ ГНУЧКИХ МЕТОДАХ РОЗРОБКИ ПРОГРАМНИХ СИСТЕМ

V. Buhai, O. Bodnarchuk, Ph.D., Assoc. Prof.

SOFTWARE SYSTEMS ARCHITECTURE ASSESSMENT FOR AGILE TECHNIQUE OF DEVELOPMENT

У зв'язку з широким впровадженням гнучких методів проектування програмних систем (Agile Software Development, ASD), стали достатньо актуальними проблеми забезпечення їх якості. Особливістю цих методів є те, що проект ділиться на завдання (спринти), в кожному з яких реалізується частина функціоналу і вони розробляються окремо, від формулювання вимог до розгортання. Причому, вимоги до програмної системи (ПС) можуть змінюватись як в процесі розробки, так і після розгортання, що потребує внесення змін в розроблювану компоненту. Дослідження і врахування впливу цих змін на якість ПС досить складне завдання і, як правило, воно не виконується в ASD. Використання технології TDD та рефакторинг дозволяють лише виявити помилки та дефекти, виправляти їх, а не контролювати якість [1].

Основними перевагами гнучких методів є те що вони пристосовані до врахування постійної зміни вимог, які аналізуються і уточнюються на початку кожної ітерації. За рахунок цього, а також проведенням неперервного тестування і інтеграції, зменшуються ризики проекту, збільшується його ефективність [1], [2].

Однак, гнучкі методи мають і ряд недоліків, з яких слід відмітити такі:

- оскільки вимоги в гнучких методах є «полегшеними» і керованими локально в межах кожної ітерації, можуть з'явитися проблеми з їх стабілізації та узгодження при інтеграції;

- можливі також проблеми з необхідністю створення "гнучкої" архітектури, яку потрібно постійно корегувати для врахування зміни вимог.

В традиційних методах проектування задача забезпечення якості ПС вирішується проведенням архітектурного проектування на початкових етапах розробки ПС. На етапі аналізу вимог до ПС формулюються вимоги до архітектури. Далі будуються варіанти архітектур. На останньому етапі проводиться оцінювання варіантів архітектур по множині критеріїв якості та вибір кращої з них, яка реалізується на наступних етапах проектування ПС. Одним з підходів до забезпечення якості ПС, розроблених в ASD, є поєднання гнучких та архітектурно орієнтованих методів проектування. З цього приводу є як позитивні оцінки, так і негативні [3], [4].

Для оцінювання такої можливості були проведені дослідження по застосуванню архітектурних технік в ASD командами розробників, які використовують SCRUM [2]. Результати показали, що команди розробників використовують в основному чотири принципи архітектурного проектування. Це big-up-front design, sprint zero, in sprints, separate architecture team. В практиці big-up-front design аналіз, синтез і оцінювання виконуються архітекторами до виконання спринтів, а не командою розробників. У практиці sprint zero проектування архітектури виконується у першому спринті командою розробників. Архітектура в подальшому реорганізується у межах спринтів. Оцінювання тут як правило не проводиться, тому що внесення необхідних змін в готовий продукт вимагає великих затрат. Застосування цієї практики можливе лише висококваліфікованою і досвідченою командою.

В separate architecture team підході існує окрема команда, яка розробляє архітектуру. В команду можуть входити представники різних команд розробників, в тому числі і архітектор. Архітектурна оцінка проводиться командою при випуску нової версії.

Як виходить з наведеного аналізу, елементи архітектурного проектування певним чином використовуються а ASD. Однак, для більш ефективного його застосування, з метою забезпечення якості, необхідно підвищити роль архітектора в ASD. Також потребують певної модифікації і архітектурні практики, для впровадження їх в ASD. Для цього в [4] пропонується проектувати архітектуру адаптованою до змін для полегшення внесення в неї цих змін у випадку уточнення вимог.

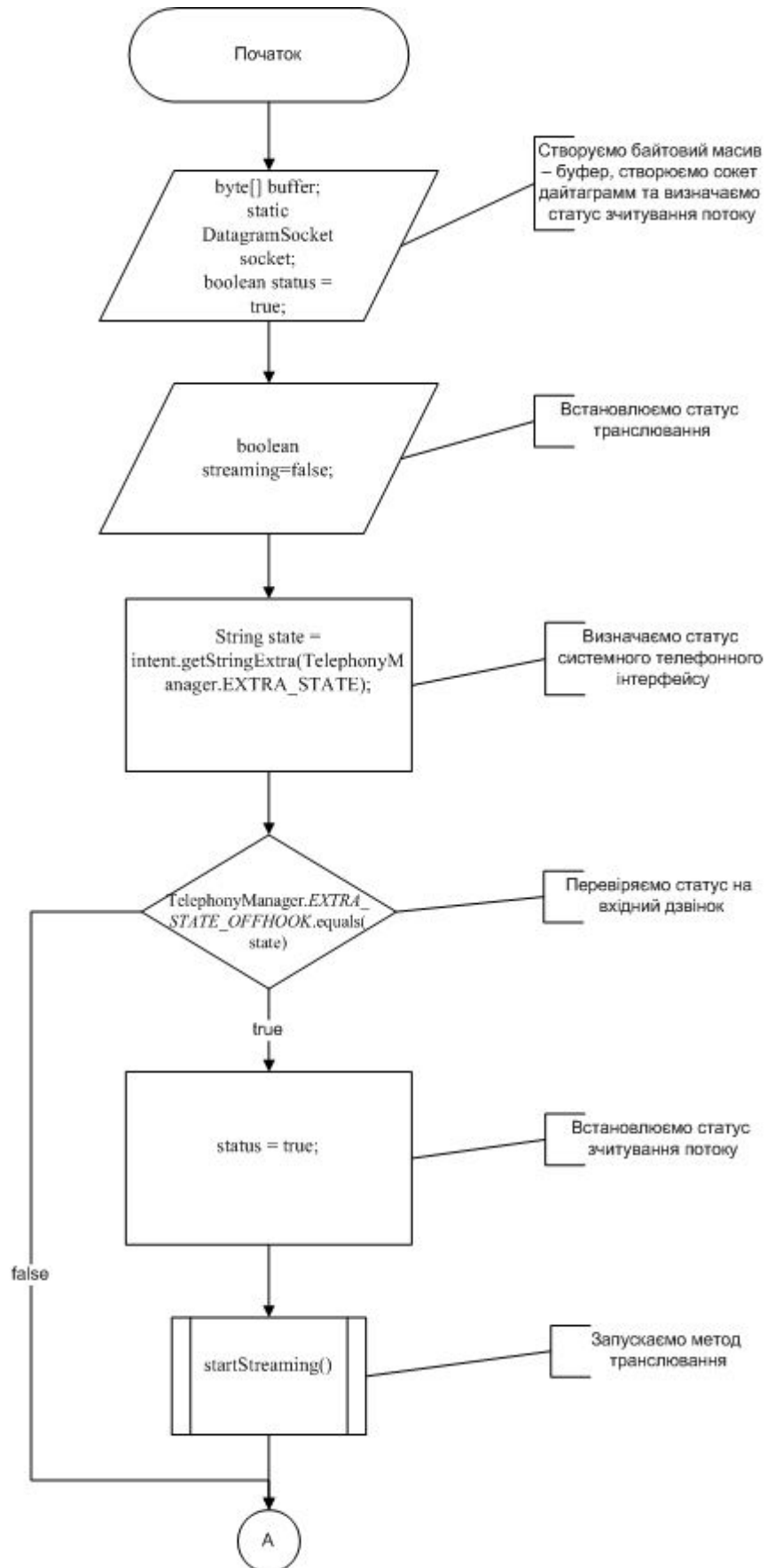
Поєднанню процесів архітектурного проектування з процесами гнучкої розробки присвячені також роботи [5], [6]. В них виділяються два етапи розробки, на першому з яких виконується архітектурне проектування і вибір базової архітектури, а на другому – реалізується процедура "гнучкої" зміни архітектури. Для забезпечення змінюваності архітектури пропонується на етапі архітектурного аналізу виявляти "точки змін" та будувати "профілі змін" на основі відповідей всіх, причетних до розробки ПС, на запитання контрольного списку. Але внесення змін в архітектуру в ASD і оцінювання відповідності її якості вимогам відбуваються шляхом аналізу відповідей експертів на запитання "контрольного списку", що є досить суб'єктивним і не точним. Також в них відсутня математична формалізація процесів оптимізації вибору архітектури та оперативного коригування критеріїв якості зміненої архітектури.

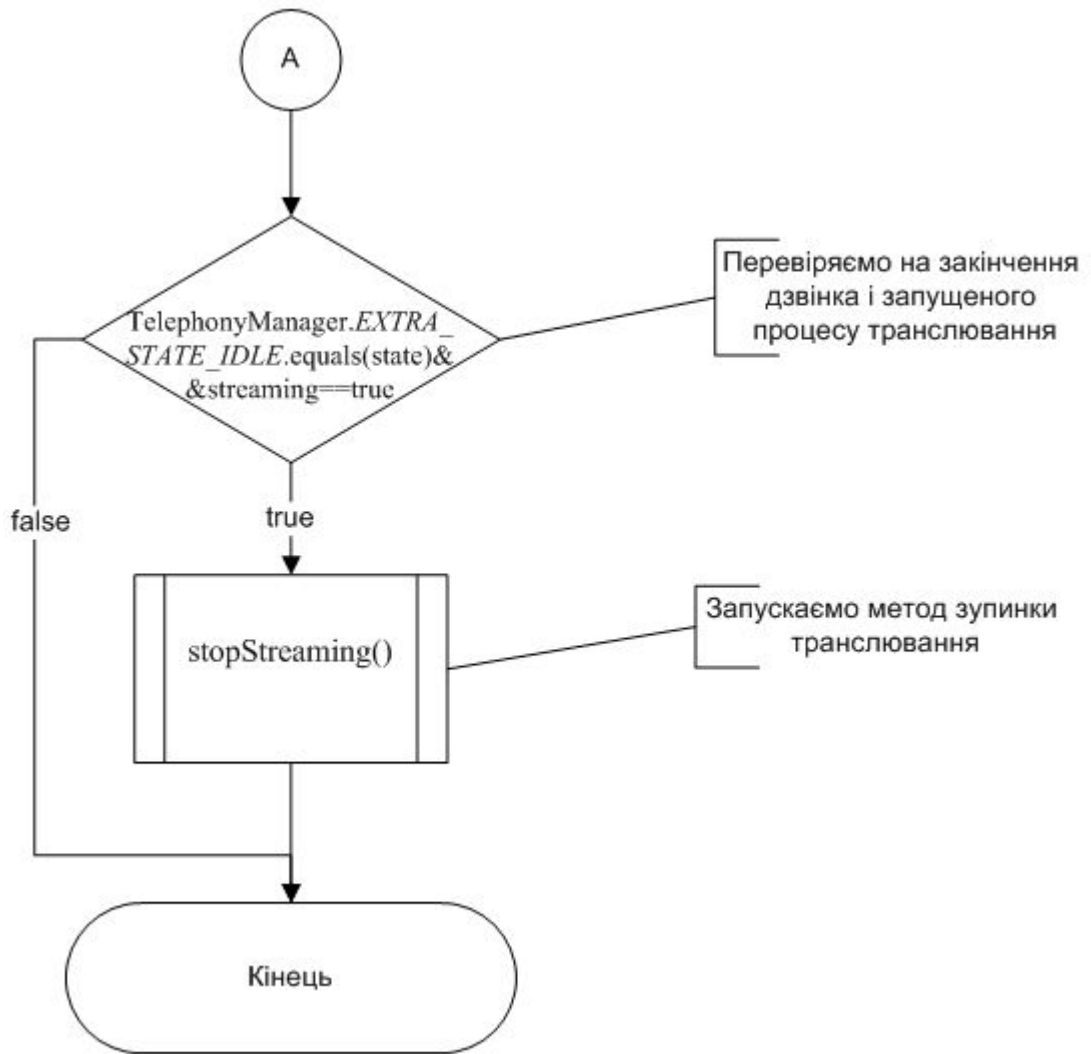
В даній роботі пропонується для поєднання архітектурного проектування і гнучких методів розробки контролювати якість зміненої архітектури обчисленням цільової функції, побудованої методом групового урахування аргументів (МГУА). Це дозволить оперативно отримати оцінки якості і прийняти рішення про продовження процедури ASD, або перехід на процедуру перепроєктування архітектури. Оцінювання та вибір архітектури з множини альтернатив на першому етапі виконується методом аналізу ієрархій (MAI), який менш трудомісткий порівняно з сценарними методами, добре формалізований і дозволяє автоматизувати процедури оцінювання.

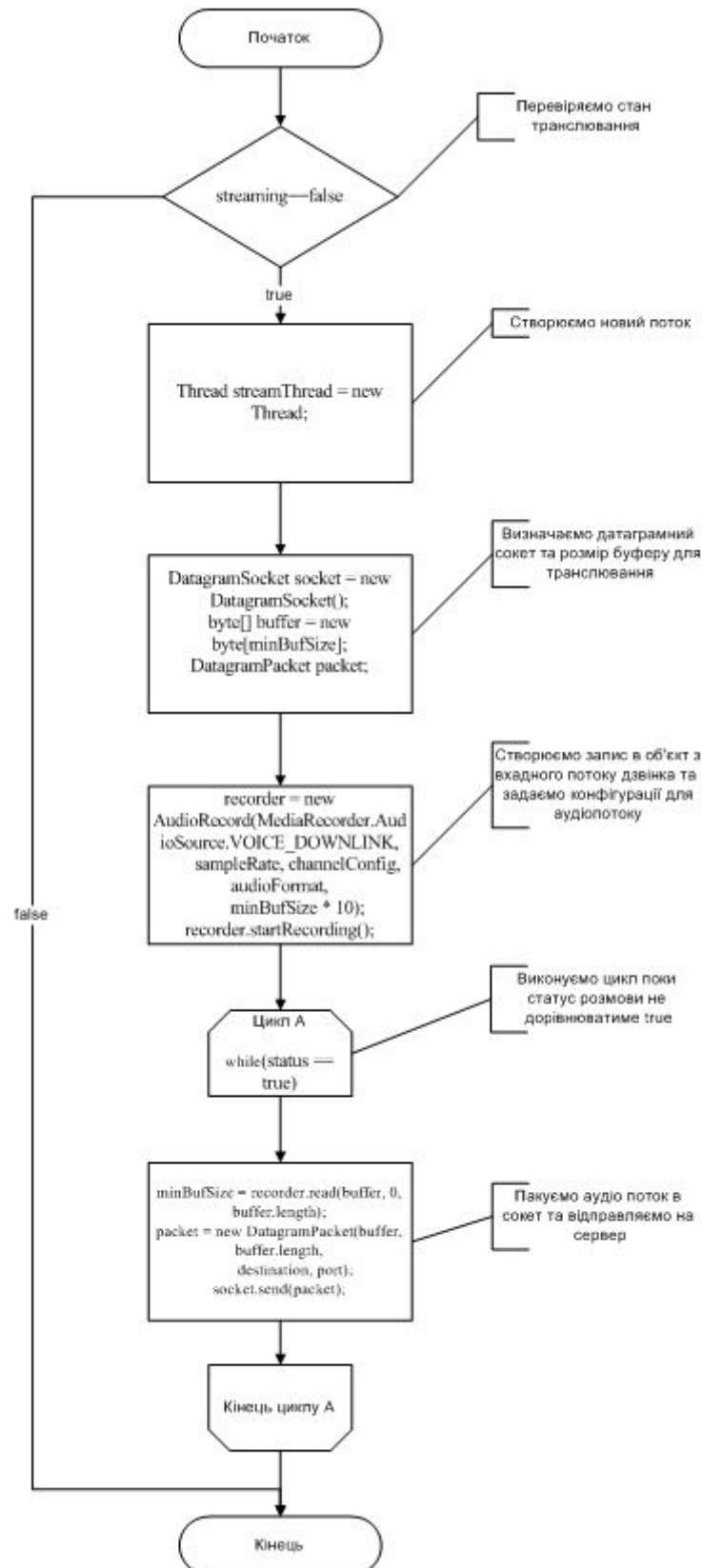
Література

1. Ioannis Stamelos, Panagiotis Stetsos editors / Agile Software Development Quality Assurance./ Information Science Reference, USA – 2007, – 257 p.
2. Moe, N.B., Dingsøyr, T., and Dybå, T., A teamwork model for understanding an agile team: A case study of a Scrum project, Information and Software Technology 52 (2010).
3. Abrahamsson P., Babar A.M., Krutchen P. Agile and Architecture- can the co- exist? IEEE Software 2010, 27(2): p. 16–22.
4. Mistrik I., Tang A., Bahsoon R. Software Architecture practices in Agile enterprise. Hershey: IGI Global, 2012. P. 230–49.
5. Babar A.M., Brown A., Mistrik I. Agile Software Architecture: aligning agile process and Software Architecture. Morgan Kaufman Elsevier Inc. 2014: – 410 p.
6. J. Coplien, G. BJORVING. Lean Architecture: for agile software development. // J Willey and Sons Ltd. U.K., 2010, – 351 p.

Алгоритм роботи головної служби для транслювання даних





Алгоритм блоку `startStreaming`, запуску транслявання

Алгоритм блоку stopStreaming, зупинки транслявання

