

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет імені Петра Могили
Факультет комп'ютерних наук
Кафедра комп'ютерної інженерії

ДОПУЩЕНО ДО ЗАХИСТУ

Завідувач кафедри,
д-р техн. наук, проф.

_____ І. М. Журавська

« __ » _____ 2023 р.

КВАЛІФІКАЦІЙНА МАГІСТЕРСЬКА РОБОТА
СИСТЕМА ВІДДАЛЕНОГО КЕРУВАННЯ
ПІДКЛЮЧЕННЯМ БАНКІВСЬКОГО
ОБЛАДНАННЯ ЧЕРЕЗ GSM-МОДУЛІ

Спеціальність «Комп'ютерна інженерія»
123 – КМР.1 – 605.21710206

Студентка

_____ А. А. Винар
підпис

« ____ » _____ 2023 р.

Керівник д-р техн. наук, проф.

_____ І. М. Журавська
підпис

« ____ » _____ 2023 р.

Миколаїв – 2023

ЗАВДАННЯ

на виконання магістерської роботи

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет імені Петра Могили
Факультет комп'ютерних наук
Кафедра комп'ютерної інженерії

ЗАТВЕРДЖУЮ

Зав. кафедри _____ І. М. Журавська

«_____» _____ 2023 р.

ЗАВДАННЯ

на виконання кваліфікаційної магістерської роботи

Видано студентці групи 605м факультету комп'ютерних наук

Винар Анастасії Анатоліївни
(прізвище, ім'я, по батькові студента)

1. Тема кваліфікаційної роботи

Система віддаленого керування підключенням банківського обладнання через GSM-модулі

Затверджена наказом по ЧНУ ім. Петра Могили від 03.11.2022 № 201.

2. Строк представлення кваліфікаційної роботи «_____» _____ 20__ р.

3. Очікуваний результат роботи та початкові дані, якщо такі потрібні

Очікуваним результатом роботи є мобільний застосунок для віддаленого керування підключенням банківського обладнання через GSM-модулі. Вхідними даними роботи є архітектура мережі банкоматів, характеристики обладнання (стандарти підключення, кількість та MAC-адреси хостів).

4. Перелік питань, що підлягають розробці

1) огляд сучасних підходів та систем віддаленого керування хостами комп'ютерної мережі, аналіз переваг та недоліків існуючих систем;

2) обґрунтування вибору методів, моделей, апаратного забезпечення та програмного середовища;

3) моделювання процесу віддаленого відключення банківського обладнання;

- 4) розробка мобільного застосунку;
5) перевірка працездатності, тестування та апробація розробленого ПЗ, аналіз результатів тестування.

5. Перелік графічних матеріалів:

слайди презентації

6. Завдання до спеціальної частини:

Проведення оцінки умов праці фахівців підприємства ЧНУ ім.Петра Могили, а саме аналіз виробничого приміщення, оцінка природнього освітлення. Визначення рівня електричної та пожежної безпеки підприємства.

7. Консультанти:

Консультант	Кафедра (організація)	Частина роботи
Д-р біол. наук, професорка Григор'єва Л.І.	Кафедра екології Медичного інституту ЧНУ ім. Петра Могили	Спеціальна частина з охорони праці

Керівник роботи

д-р техн. наук, професор Журавська Ірина Миколаївна
(посада, прізвище, ім'я, по батькові)

(підпис)

Завдання прийнято до виконання

Винар Анастасія Анатоліївна
(прізвище, ім'я, по батькові студента)

(підпис)

Дата видачі завдання «____» _____ 2022 р.

КАЛЕНДАРНИЙ ПЛАН
виконання кваліфікаційної магістерської роботи

Тема: Система віддаленого керування підключенням банківського обладнання через GSM-модулі

№	Найменування роботи	Початок	Закінчення	Примітки
1.	Розробка та затвердження завдання на виконання КМР	01.09.2022	15.09.2022	<i>Виконано</i>
2.	Огляд літератури за темою роботи	15.09.2022	01.10.2022	<i>Виконано</i>
3.	Складання календарного плану КМР	01.10.2022	03.10.2022	<i>Виконано</i>
4.	Аналіз предметної області	05.10.2022	25.10.2022	<i>Виконано</i>
5.	Обґрунтування вибору методів, моделей, апаратного забезпечення та програмного середовища	25.10.2022	10.11.2022	<i>Виконано</i>
6.	Моделювання процесу віддаленого відключення банківського обладнання	10.11.2022	15.11.2022	<i>Виконано</i>
7.	Розробка мобільного застосунку	15.11.2022	20.12.2022	<i>Виконано</i>
8.	Перевірка працездатності, тестування та апробація розробленого ПЗ	20.12.2022	05.01.2023	<i>Виконано</i>
9.	Аналіз результатів тестування	05.01.2023	10.01.2023	<i>Виконано</i>
10.	Відгук керівника КМР	10.01.2023	13.01.2023	<i>Виконано</i>
11.	Оформлення КМР та презентації	13.01.2023	05.02.2023	<i>Виконано</i>
12.	Попередній захист	07.02.2023	08.02.2023	<i>Виконано</i>
13.	Рецензування	08.02.2023	11.02.2023	<i>Виконано</i>
14.	Завершення оформлення КМР та презентації	12.02.2023	18.02.2023	<i>Виконано</i>
15.	Захист кваліфікаційної роботи	24.02.2023	27.02.2023	<i>Виконано</i>

Розробила здобувачка ВО Винар Анастасія Анатоліївна _____
(прізвище, ім'я, по батькові) (підпис)
«__» _____ 20__ р.

Керівник роботи зав. каф. КІ Журавська Ірина Миколаївна _____
(підпис)
«__» _____ 20__ р.

АНОТАЦІЯ

до кваліфікаційної магістерської роботи
«Система віддаленого керування підключенням банківського обладнання
через GSM-модулі»

Студентка гр. 605м Винар Анастасія Анатоліївна

Керівник: д-р техн. наук, професор Журавська Ірина Миколаївна

Актуальність теми кваліфікаційної роботи обумовлена необхідністю дотримання режим світломаскування цивільних та виробничих об'єктів шляхом віддаленого відключення банківського обладнання та відновлення електроживлення при завершенні аварійних ситуацій.

Об'єкт дослідження – процес керування дистанційним відключенням мережевого обладнання та модель справедливого обслуговування черги.

Предмет дослідження – система віддаленого керування підключенням банківського обладнання через GSM-модулі з мобільного пристрою.

У даній роботі поставлено за **мету** створення застосунку для керування банкоматами на відстані, через вбудовані або під'єднані до них GSM-модулі.

Для досягнення мети необхідно виконати такі **завдання**:

- огляд існуючих систем віддаленого керування;
- моделювання системи, що розробляється, та вибір GSM-розетки;
- створення програмного забезпечення для дистанційного керування банкоматами через мережу GSM;
- тестування створеного застосунку.

Кваліфікаційна робота містить перелік скорочень, вступ, чотири розділи, висновок, перелік джерел посилання та два додатки.

Вступ містить основні обґрунтування актуальності розробки обраної теми, об'єкт, предмет дослідження, мету та завдання, які необхідно виконати для досягнення поставленої мети.

У розділі 1 було розглянуто принцип дії стандарту GSM та створюваних на його основі модулів, за допомогою яких здійснюється віддалене керування пристроями, підключеними до корпоративної банківської мережі.

У розділі 2 було розглянуто апаратне та програмне забезпечення системи та обрано GSM-модуль та середовище розробки та мову програмування.

У третьому розділі було розглянуто життєвий цикл розробки програмного забезпечення для дистанційного керування GSM-розеткою.

У четвертому розділі описано етапи розробки мобільного застосунку для віддаленого вимикання/вмикання банкоматів через GSM-модулі, протестовано додавання нового обладнання до системи.

У висновку описано результати виконання кваліфікаційної роботи.

Додатки містять код програмного забезпечення та інформацію про апробацію кваліфікаційної магістерської роботи.

В спеціальній частині з охорони праці розглянуто забезпечення вимог охорони праці на робочому місці.

Кваліфікаційна магістерська робота містить 68 сторінок (без додатків), 24 рисунки, 24 джерела посилання, 2 додатки.

ABSTRACT

of the Master's Thesis

«The system of remote control of the connection of banking equipment through GSM modules»

Student of group 605: Vynar Anastasiia Anatoliivna

Supervisor: D.Sc. (Techn.), Professor Zhuravska I. M.

The **relevance of the topic** of the qualification work is due to the need to support the mode of light masking of civil and industrial facilities by remote disconnection of banking equipment and restoration of power supply at the end of emergency situations.

The **object of the study** is the process of managing the remote shutdown of network equipment and the model of fair queue service.

The **subject of the study** is a system of remote control of the connection of banking equipment through GSM modules from a mobile device.

In this work, **the goal** is to create an application for controlling ATMs remotely, through built-in or connected GSM modules.

To achieve the goal, it is necessary to complete **the following tasks**:

- review of existing remote control systems;
- simulation of the system under development and selection of a GSM socket;
- creation of software for remote control of ATMs via the GSM network;
- testing the created application.

The thesis contains a list of abbreviations, an introduction, four sections, a conclusion, a list of reference sources and two appendices.

The introduction contains the main justifications for the relevance of the development of the chosen topic, object, subject of research, goal and task, which are necessary to achieve the goal.

Section 1 established the principle of operation of the GSM standard and the modules created on its basis, which are used to remote control devices connected to the corporate banking network.

In section 2, the hardware and software of the systems were designed and the GSM module and language development and programming environment were selected.

In the third section, the life cycle of software development for remote control of a GSM socket was important.

The fourth section describes the stages of development of mobile equipment for remote switching off/on of ATMs via GSM modules, the addition of new equipment to the system is tested.

The conclusion describes the results of the qualification work.

Appendices according to the software code and information on the approval of the qualifying master's thesis.

In the special part on labor protection issues, ensuring labor protection requirements at the workplace.

The qualifying master's thesis contains 68 pages (without appendices), 24 figures, 24 reference sources, 2 appendices.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ	4
ВСТУП	5
1 ЗАСОБИ ВІДДАЛЕНОГО КЕРУВАННЯ БАНКІВСЬКИМ ОБЛАДНАННЯМ	8
1.1 Архітектура мереж віддаленого керування банківським обладнанням	8
1.2 GSM-модуль та принцип його роботи	11
1.3 Використання GSM-модулів	15
Висновки до розділу 1	17
2 ОБГРУНТУВАННЯ ВИБОРУ МЕТОДІВ, МОДЕЛЕЙ, АПАРАТНОГО ЗАБЕЗПЕЧЕННЯ ТА ПРОГРАМНОГО СЕРЕДОВИЩА РОЗРОБКИ	18
2.1 Забезпечення принципів QoS в розгалуженій системі банкоматів	18
2.2 Аналітична модель справедливого обслуговування черги банкоматів	25
2.3 Аналіз апаратного забезпечення підключення банкоматів до комп'ютерної мережі	34
2.4 Аналіз програмного забезпечення розробки застосунків для віддаленого керування банкоматами	37
Висновки до розділу 2	38
3 СТВОРЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ СИСТЕМИ	40
3.1 Етапи розробки мобільного застосунку	40
3.2 Планування та аналіз	44
3.3 Дизайн	45
3.4 Розробка	53
Висновки до розділу 3	55
4 ТЕСТУВАННЯ СИСТЕМИ ВІДДАЛЕНОГО КЕРУВАННЯ	56
4.1 Тестування	56
4.2 Розгортання	60
4.3 Технічне обслуговування	61

Висновки до розділу 4	63
ВИСНОВКИ	64
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	66
ДОДАТОК А Коди програм	69
А1 Код програми для обробки результату SMS-повідомлення	69
А2 Код програми для налаштування бази даних	70
ДОДАТОК Б Публікації за темою роботи	71

ПЕРЕЛІК СКОРОЧЕНЬ

ATM	– Automated Teller Machine
BSC	– Base Station Controller
EDGE	– Enhanced Data Rates for GSM Evolution
GGSN	– Gateway GPRS Support Node
GPRS	– General Packet Radio Service
GSM	– Global System for Mobile Communications
GTP	– GPRS Tunnelling Protocol
HLR	– Home Location Register
IDE	– Integrated Development Environment
IoT	– Internet of Thing
MS	– Mobile Station
MVP	– Minimum Viable Product
PDP	– Packet Data Protocol
PS	– Packet Switched
PQ	– Priority Queuing
SDLC	– Software Development Lifecycle
SMS	– Short Message Service

ВСТУП

З перших днів повномасштабного вторгнення росії в Україну в багатьох регіонах задля перешкоджанню орієнтування авіації та безпілотників противника, а також роботи диверсійно-розвідувальних сил, був введений режим світломаскування. В зв'язку з цим, ввечері та вночі (після 21-ї години) освітлення має бути вимкнуте – в будинках, на вулицях, в значущих об'єктах – у т. ч. на рекламних білбордах та на банкоматах [1].

Восени українська енергетична система серйозно постраждала від російських терактів. Блекаут в Україні змінив підхід до використання електроенергії, змусивши задуматися людей про свідоме використання енергії та її економію. Крім того, після відновлення електропостачання через планові відключення електроенергії, оператор має дистанційно, за допомогою програмних засобів, підключити банківське обладнання до електромережі [2].

Однією з проблем в цьому випадку є банкомати, які працюють цілодобово. Кількість банківського обладнання у місті унеможлиблює їх відключення вручну. Тож має сенс створення системи дистанційного керування підключенням банківського обладнання. Для цього є доцільним використання GSM-розетки.

GSM-розетка – це тип пристрою дистанційного керування, який дозволяє керувати електричними приладами через мережу GSM. По суті, це розетка, яку можна дистанційно вмикати або вимикати за допомогою мобільного телефону. Пристрій зазвичай має вбудований модем GSM, який «спілкується» з програмою для мобільного телефону або веб-сайтом через мобільну мережу для надсилання та отримання команд керування. З розеткою GSM, зручність та ефективність обслуговування та керування банкоматом зростає.

Отже, створювана система дозволяє співробітникам або технічному спеціалісту дистанційно керувати банківським обладнанням (в даному випадку банкоматами) та контролювати його за допомогою модулів GSM (Global System for Mobile Communications).

Для віддаленого відключення в якості керуючого елемента використовується GSM-розетка – невеликий електроприлад зі вбудованою електронною платою і GSM-модулем, який має слот для SIM-карти, вхідну вилку й електричний роз'єм для підключення іншої апаратури [3].

GSM-розетка має можливість дистанційного керування через дзвінок або SMS-повідомлення, але це не завжди є зручним, оскільки зовнішній шум може завадити «вводу» команди [4].

Актуальність роботи, що виконується, підкреслюється важливістю збереження світломаскування в містах, а також заощадження електроенергії.

Об'єкт дослідження – процес керування дистанційним відключенням мережевого обладнання та модель справедливого обслуговування черги на відключення.

Предмет дослідження – система віддаленого керування підключенням банківського обладнання через GSM-модулі з мобільного пристрою.

У даній роботі поставлено за **мету** створення застосунку, який забезпечує зручний спосіб керування банківськими пристроями на відстані, та керування налаштуваннями та конфігураціями, використовуючи вбудовані або під'єднані до них GSM-модулі.

Для досягнення мети необхідно виконати такі **завдання**:

- огляд існуючих систем віддаленого керування та програмних застосунків;
- моделювання системи, що розробляється, та вибір GSM-розетки;
- створення програмного забезпечення для дистанційного керування банкоматами через мережу GSM;
- тестування створеного застосунку.

Важливо ретельно спланувати та протестувати систему, щоб переконатися, що вона функціонує належним чином і відповідає вимогам цільового середовища.

Практична значимість розробленої системи полягає у тому, що вона дозволяє дистанційно керувати GSM-розеткою через застосунок, оскільки існуюче управління через SMS-повідомлення чи дзвінок є не завжди зручним, так як зовнішній шум може завадити «вводу» команди.

Апробація результатів кваліфікаційної роботи відбулася під час Всеукраїнської науково-практичної конференції молодих вчених, аспірантів і студентів «Інформаційні технології та інженерія» (Миколаїв, 07–10 лютого 2023 р.).

Публікації. Основні положення та результати магістерської роботи опубліковані у збірнику матеріалів Всеукраїнської науково-практичної конференції «Інформаційні технології та інженерія» [5].

1 ЗАСОБИ ВІДДАЛЕНОГО КЕРУВАННЯ БАНКІВСЬКИМ ОБЛАДНАННЯМ

1.1 Архітектура мереж віддаленого керування банківським обладнанням

У сучасному світі стрімко збільшується кількість банків / банківських послуг / банківського обладнання, яке за локалізацією може бути класифіковано таким чином:

1) банкомат на місці: у цьому випадку банкомати розташовані або у відділенні, або поблизу від відділень, щоб клієнти могли уникнути черги, які присутні у відділенні, і заощадити свій час для завершення транзакції;

2) зовнішні банкомати: такі банкомати розташовані за межами банку, як-от на залізничних станціях, у торгових центрах, АЗС та аеропортах і т. п. Зазвичай це відбувається на автономній основі, де немає фізичної присутності філії. Це робиться банками для того, щоб охопити все більше клієнтів у географічних місцях, де немає філій, але клієнти все ще можуть користуватися їхніми послугами;

3) банкомат на робочому місці: такі, що розташовані в будівлях організації та зазвичай обслуговують лише працівників організації.

4) мобільний банкомат, також відомий як банкомат на колесах. У цьому випадку банкомати переміщуються в різні місця, щоб клієнти могли користуватися їхніми послугами.

З огляду на вищезазначене зрозуміло, що треба приділяти велику увагу технічній організації мережевої взаємодії між віддаленими банкоматами та банківським відділенням, яке забезпечує його функціонування. При цьому необхідно чітко розмежувати необхідні моніторингові та сервісні функції для кожного модулю банкомата (англ. АТМ □ Automated Teller Machine), як наведено на рис. 1.1.

дистанційного доступу та управління обладнанням. Для організації дистанційного керування за допомогою GSM-розетки можна виконати наступні кроки:

Архітектура мережі: архітектура мережі повинна бути розроблена таким чином, щоб забезпечити безпечний доступ до обладнання для віддаленого керування.

Конфігурація обладнання: банківське обладнання, таке як банкомати, має бути налаштовано таким чином, щоб забезпечити віддалений доступ і керування. Це може передбачати налаштування протоколів віддаленого доступу, налаштування обладнання з необхідними програмними та апаратними компонентами та встановлення GSM-розетки.

Програмне забезпечення для моніторингу та керування: слід запровадити систему програмного забезпечення для моніторингу та керування для моніторингу продуктивності банківського обладнання та надання можливостей дистанційного керування. Це програмне забезпечення повинно включати у себе моніторинг продуктивності в реальному часі та сповіщення, а також виконувати інші завдання віддаленого керування.

Контроль доступу: доступ до банківського обладнання слід контролювати, щоб гарантувати, що лише авторизований персонал має доступ до обладнання для віддаленого керування. Це може включати налаштування облікових записів користувачів і дозволів, і впровадження заходів безпеки, таких як двофакторна автентифікація.

Технічне обслуговування та підтримка: необхідно створити програму технічного обслуговування та підтримки, щоб забезпечити регулярне технічне обслуговування та підтримку банківського обладнання. Це може включати планування регулярних оновлень програмного забезпечення та обслуговування обладнання, а також надання підтримки при збоях обладнання та інших проблемах.

Безпека. Безпека системи віддаленого керування має бути головним пріоритетом, а для захисту конфіденційної інформації слід застосовувати такі заходи, як шифрування та безпечне зберігання даних. Необхідно також проводити регулярні перевірки безпеки, щоб виявити та усунути будь-які вразливі місця.

Реалізуючи ці кроки та використовуючи GSM-розетку, дистанційне керування мережею банківського обладнання може бути ефективно організовано із одночасним забезпеченням безпечного та зручного способу дистанційного доступу.

На рис. 1.2 продемонстровано схему зв'язку смартфона з банківським обладнанням.

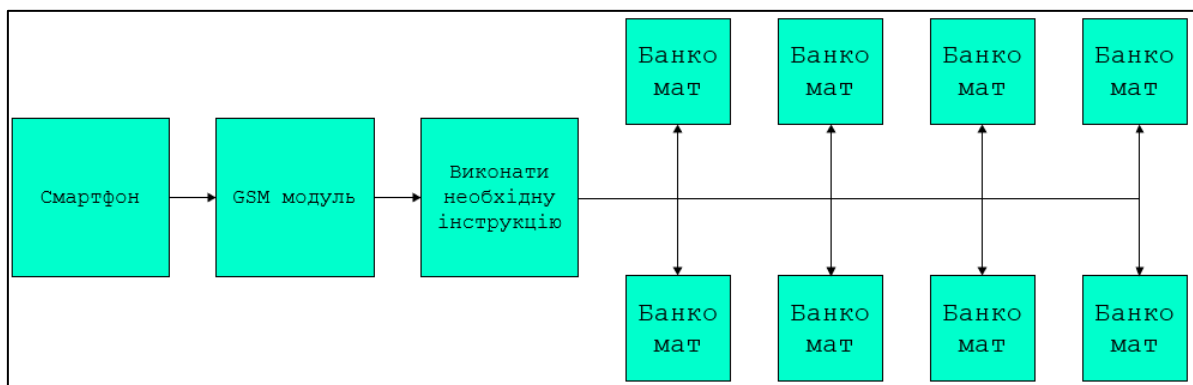


Рисунок 1.2 – Блок-схема системи

1.2 GSM-модуль та принцип його роботи

Стандарт GSM (від назви групи Groupe Special Mobile, пізніше перейменований на Global System for Mobile Communications) – глобальний цифровий стандарт для мобільного стільникового зв'язку другого покоління, є цифровим та забезпечує високу якість та конфіденційність зв'язку.

На основі стандарту було створено модуль GSM – апаратний пристрій, який об'єднує модем GSM з мікроконтролером (рис. 1.3), дозволяючи обмінюватися даними між пристроєм і мобільним телефоном через SMS (загальна пакетна радіопередача).

Електрична схема «розумної» розетки включає електромагнітне реле, здатне проводити ввімкнення і вимикання живлення на контактах розетки, включаючи і вимикаючи при цьому прилад, який підключений до розетки.

Драйвер реле використовується для ізоляції як керуючого, так і керованого пристрою. Реле – це електромагнітний пристрій, який складається з соленоїда, рухомих контактів (перемикача) і відновної пружини; споживає порівняно велику кількість енергії.

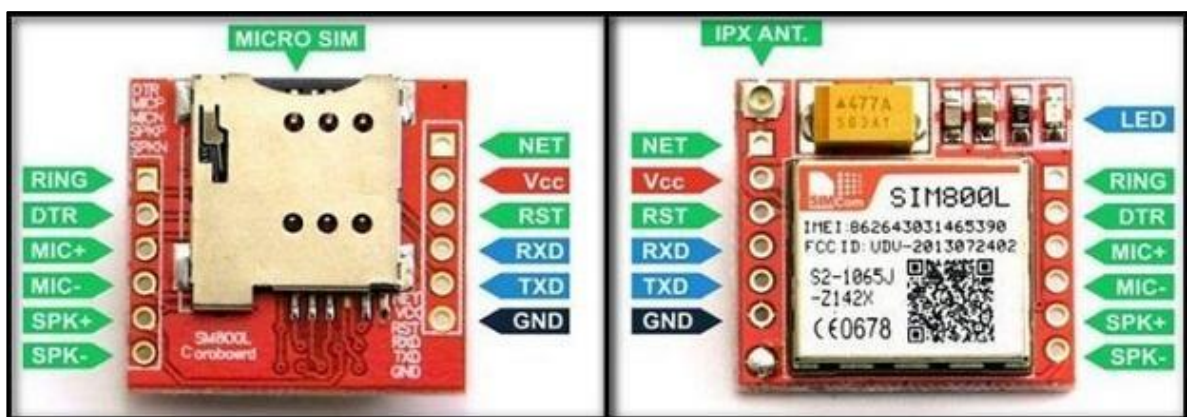


Рисунок 1.3 – Модуль GSM SIM800L

Принцип роботи модуля GSM базується на концепції стільникових мереж. Стільникова мережа складається з багатьох невеликих стільників або базових станцій, кожна з яких охоплює певну географічну область, які підключені до центральної мережі.

Коли мобільний телефон або модуль GSM увімкнено, він реєструється на найближчій базовій станції та встановлює з'єднання з центральною мережею. Це з'єднання підтримується, коли пристрій переміщується з однієї комірки в іншу, при цьому з'єднання передається від однієї базової станції до іншої в процесі, відомому як «хендовер».

Дані та голосовий зв'язок між пристроєм і центральною мережею передаються за допомогою радіочастот. Модуль GSM використовує множинний доступ з тимчасовим поділом каналів (TDMA) і множинний доступ з частотним поділом каналів (FDMA), щоб керувати доступними

радіочастотами та гарантувати, що кілька пристроїв можуть використовувати одну частоту в різний час.

Що стосується передачі даних, модуль GSM може підтримувати різні служби передачі даних, такі як «Служба коротких повідомлень (SMS)», «Служба пакетної радіопередачі (GPRS)» і «Покращена швидкість передачі даних для GSM Evolution (EDGE)». Ці послуги передачі даних дозволяють модулю GSM передавати та отримувати дані по повітрю, забезпечувати необхідне підключення для застосунків IoT (Інтернет речей) та інших застосунків, що інтенсивно працюють з даними.

Сервіс SMS (англ. SMS, Short Message Service) був включений в стандарти з самого початку. Пізніше його перенесли на такі бездротові технології, як CDMA та TDMA.

Служба коротких повідомлень (SMS) популярна серед користувачів мобільних телефонів як дешевий і зручний спосіб спілкування. Таким чином, технологія SMS є загальною для всіх постачальників послуг мобільного зв'язку. Оскільки використання технології SMS є дешевим, зручним і гнучким способом передачі даних, дослідники намагаються застосувати цю технологію в багатьох сферах, які не охоплюються постачальниками послуг.

SMS-повідомлення зараз є одним із найважливіших джерел доходу операторів бездротового зв'язку.

- SMS-повідомлення можна надсилати та читати в будь-який час;
- SMS-повідомлення можна надіслати на офлайн-мобільний телефон;
- SMS-повідомлення менш заважають, але ви все ще можете залишатися на зв'язку;
- SMS підтримуються мобільними телефонами 100 % GSM, і вони можуть бути надіслані між різними операторами бездротового зв'язку.

Стандарти GSM і SMS були спочатку розроблені ETSI. Одне SMS-повідомлення може містити в собі максимум 140 байт (1120 біт) даних, тому одне SMS-повідомлення може містити до: 160 символів, якщо

використовується 7-бітне кодування символів, або 70 символів, якщо використовується 16-бітне кодування символів Unicode UCS2 (текстові SMS-повідомлення, що містять нелатинські символи, як-от китайські, мають використовувати 16/32-бітне кодування символів.)

Після надсилання повідомлення, його отримує SMS-центр (англ. SMSC), який потім має передати його на відповідний мобільний пристрій. Щоб зробити це, SMSC надсилає SMS-запит до Home Location Register (HLR), щоб знайти клієнта в роумінгу.

Як тільки HLR отримує запит, він відповідає на SMSC із статусом абонента: неактивний або активний. Якщо відповідь «неактивна», SMSC буде зберігати повідомлення протягом певного часу. Коли абонент отримує доступ до свого пристрою, HLR надсилає SMS-сповіщення до SMSC, і SMSC намагатиметься доставити. SMSC передає повідомлення у форматі доставки коротких повідомлень. Система пінгує пристрій, і якщо він відповідає, то повідомлення буде доставлено. SMSC отримує підтвердження того, що повідомлення отримано кінцевим користувачем, потім класифікує повідомлення як «надіслане» і більше не намагатиметься надіслати. SMS забезпечує механізм для передачі коротких повідомлень «до» та «з» бездротової мережі пристроїв. Сервіс використовує SMSC, який діє як система зберігання та пересилання коротких повідомлень.

Підсумовуючи, принцип роботи модуля GSM передбачає встановлення з'єднання зі стільниковою мережею, використання радіочастот для передачі та отримання даних і голосового зв'язку, і управління доступними радіочастотами для забезпечення ефективного використання мережі.

У системі дистанційного керування модуль GSM використовується для надсилання та отримання команд керування з мобільного телефону, який потім може керувати пристроями, підключеними до модуля. Цей тип системи забезпечує зручний спосіб дистанційного керування та моніторингу пристроїв, що робить його корисним для широкого спектру програм.

1.3 Використання GSM-модулів

GSM-розетка – це пристрій, яким можна дистанційно керувати за допомогою модуля GSM; дозволяє комп'ютеру чи іншому пристрою зв'язуватися з мережею для надсилання та отримання даних та голосових викликів (рис. 1.4).



Рисунок 1.4 □ Керування через GSM-мережу

Модулі GSM широко використовуються в різних сферах, таких як:

- телекомунікації: модулі GSM використовуються для мобільного зв'язку та забезпечують передачу голосу та даних для мобільних телефонів;
- IoT (Інтернет речей): модулі GSM використовуються в пристроях IoT для підключення та зв'язку між пристроями та Інтернетом;
- автоматизація: модулі GSM використовуються в системах промислової автоматизації для забезпечення можливостей дистанційного керування та моніторингу;

- відстеження транспортних засобів: модулі GSM використовуються в системах GPS-відстеження для забезпечення реальної інформації про місцезнаходження транспортних засобів;
- системи безпеки: модулі GSM використовуються в системах безпеки, таких як охоронна сигналізація, для забезпечення дистанційного моніторингу та контролю;
- охорона здоров'я: модулі GSM використовуються в медичних пристроях, таких як монітори глюкози, для передачі даних пацієнтів постачальникам медичних послуг;
- управління енергією. модулі GSM використовуються в інтелектуальних системах керування енергією для моніторингу та контролю використання енергії в режимі реального часу;
- сільське господарство: модулі GSM використовуються для моніторингу погодних умов, вологості ґрунту та інших факторів.

Це деякі сфери, де зазвичай використовуються модулі GSM, але програми не обмежуються цим, і їх можна використовувати також у багатьох інших сферах. GSM-розетки часто використовуються для домашньої автоматизації: дистанційного керування освітленням, вентиляційною системою, кондиціонером та іншими електричними приладами, а також для моніторингу та контролю споживання електроенергії побутовими приладами.

Впроваджує вжиток GSM-модулів також аграрний бізнес для контролю та моніторингу іригаційних систем та іншого обладнання, що використовується в сільському господарстві, що оптимізує споживання води та підвищує врожайність.

У промисловості GSM-модулі можуть бути використані для дистанційного керування насосами, двигунами. Це може допомогти скоротити час простою, підвищити ефективність і підвищити безпеку.

Загалом, використання розеток з вбудованим GSM-модулем є дуже гнучким і адаптовним, що робить його популярним та практичним вибором

для дистанційного керування та моніторингу в широкому діапазоні сфер застосування.

Висновки до розділу 1

У розділі 1 було розглянуто принцип дії стандарту GSM та створюваних на його основі модулів, за допомогою яких надсилаються команди керування з мобільного телефону та, таким чином, здійснюється керування пристроями, підключеними до модуля. Система дистанційного керування банківським обладнанням забезпечує практичний спосіб моніторингу пристроїв, що робить її корисною для широкого спектру сфер діяльності, таких як домашня автоматизація, системи безпеки та промислові системи керування.

Але існує низка невирішених питань щодо керування розгалуженою банківською мережею з чисельними одиницями обладнання, для вирішення яких необхідні додаткові дослідження та технічні пропозиції.

2 ОБГРУНТУВАННЯ ВИБОРУ МЕТОДІВ, МОДЕЛЕЙ, АПАРАТНОГО ЗАБЕЗПЕЧЕННЯ ТА ПРОГРАМНОГО СЕРЕДОВИЩА РОЗРОБКИ

2.1 Забезпечення принципів QoS в розгалуженій системі банкоматів

QoS (англ. Quality of Service – «якість обслуговування») – технологія надання різним класам трафіку різних пріоритетів в обслуговуванні, концепція в телекомунікаціях і комп'ютерних мережах. Це стосується здатності контролювати та керувати продуктивністю мережі з метою забезпечення певного рівня якості послуг для користувачів мережі. Мета QoS полягає в тому, щоб критично важливі програми отримали пріоритет над менш чутливими до часу програмами, з точки зору втрати пакетів даних, затримок у передачі пакетів та нерівномірності цих затримок.

Розвиток Інтернету речей (IoT) призвів до конвергенції гетерогенних мережевих систем, що працюють за допомогою різних протоколів, і, як наслідок, до появи протокольної-незалежної обробки пакетів на основі програмованої площини даних (PDP) для IoT.

Різні програми та послуги, що базуються на пакетній передачі даних у мобільних мережах, що реалізуються за допомогою технологій GPRS/EDGE, пред'являють істотно відмінні вимоги до таких параметрів передачі як: ймовірність помилки при передачі, швидкість передачі (максимальна та середня), тимчасові затримки під час передачі даних, і т. д.

У цілому технологію GPRS абстрактно можна уявити як блочну будівлю, що складається з кількох незалежних блоків, «з'єднаних» між собою з допомогою певних інтерфейсів. Всередині кожного окремого блоку властивий свій дизайн, тобто, рівень якості, який він може надати – QoS, «нав'язаний» йому його ж технологією реалізації.

Для більшості випадків якість зв'язку визначається чотирма параметрами:

- смуга пропускання (Bandwidth), що визначає номінальну пропускну здатність середовища передачі інформації, визначає ширину каналу в bit/s (bps), kbit/s (kbps), Mbit/s (Mbps), Gbit/s (Gbps);
- затримка при передачі пакета (Delay), вимірюється в мілісекундах;
- коливання (тремтіння) затримки під час передачі пакетів □ джиттер (Jitter);
- втрата пакетів (Packet loss). Визначає кількість пакетів, втрачених у мережі під час передачі.

Контролюючи пропускну здатність, доступну для різних типів трафіку, QoS може допомогти забезпечити отримання критично важливих програм ресурсами, необхідними для ефективної роботи, тоді як некритичні програми не споживають надмірну кількість мережевих ресурсів. Це може допомогти підвищити загальну продуктивність і надійність мережі.

Оскільки система, з якою відбувається взаємодія в даній роботі, має в своєму складі не одну одиницю банківського обладнання, а декілька, виникає питання черги, яка має бути розвантажена.

Коли передача даних стикається з проблемою «вузького місця» або «пляшкового горла» (англ. «bottle neck») для прийому та відправлення пакетів даних, тоді використовуються різні методи забезпечення якості обслуговування. Найчастіше використовується метод FIFO (First In □ First Out): перший прийшов □ перший пішов. При інтенсивному трафіку це створює затори, які вирішуються простим способом: всі пакети, черги FIFO (на вхід або на вихід), що не увійшли до буфера, ігноруються і, відповідно, втрачаються безповоротно (рис. 2.1). Найрозумніший метод □ використовувати «розумну» чергу, в якій пріоритет пакетів залежить від типу сервісу □ ToS (Type of Service).

Звичайні користувачі найчастіше стикаються з терміном QoS у домашніх маршрутизаторах із підтримкою QoS.

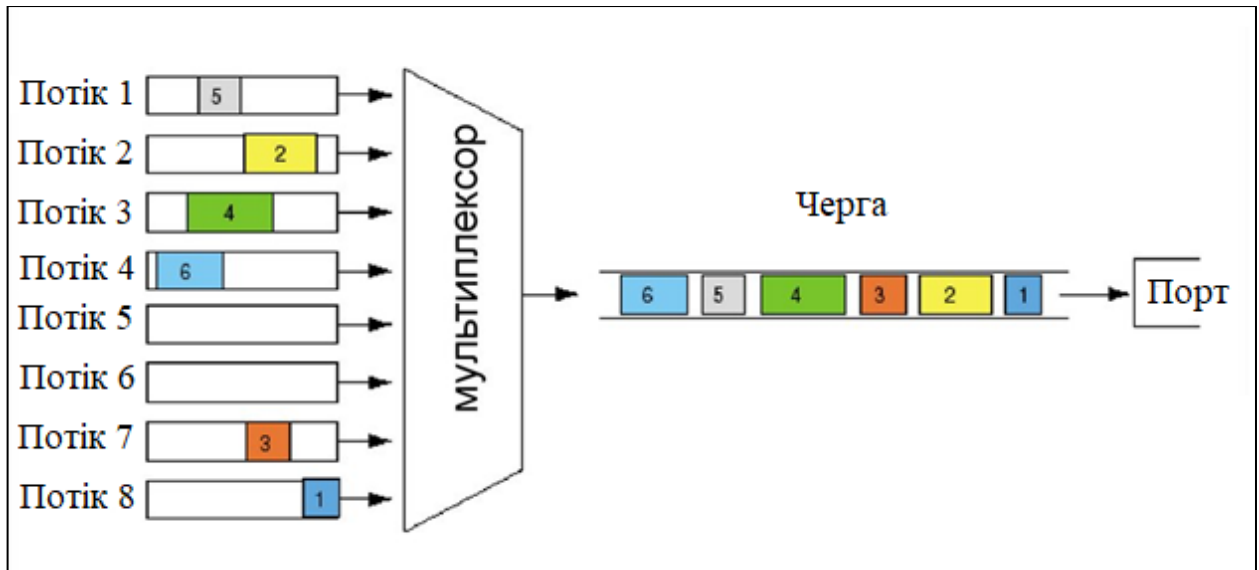


Рисунок 2.1 □ Метод обслуговування черги «First In □ First Out»

Слід зазначити, що розробка і вдосконалення методів класифікації якості, що надається, не стоїть на місці і з новими специфікаціями профілі QoS оновлюються новими, все більш дискретними характеристиками, але ключові параметри класів і атрибутів QoS були закладені в декількох основних специфікаціях:

- *Reliability class*. Специфікаціями GSM визначено 5 основних класів надійності для GPRS, при цьому надійність даних визначається з точки зору ймовірності: втрати даних (probability of data loss), дублювання доставлених даних (Probability of Duplicate Data Delivery), спотворення даних (probability of corrupted data). Залежно від класу надійності значення цих параметрів можуть становити від 10^{-9} до 10^{-4} . Клас надійності визначає вимоги до мережевих протоколів різних рівнів. Для забезпечення вимог щодо класу надійності встановлюють відповідну комбінацію режимів передачі (Mode) на рівнях GTP, LLC і RLC. Передачі сигналізації, тобто, службових даних (GMM/SM) та SMS використовують 3-й клас надійності, але не факт, що всі

PLMN-мережі операторів можуть підтримувати цей клас при переміщенні абонента між мережами;

Частково цей клас пов'язані з необхідністю передачі: від 1 (Class A) – застосунки, «чутливі» до появи помилок, до 5 – для non-real-time застосунків, які можуть компенсувати появу помилок;

– *Delay class*. У специфікації GSM Rec.02.60 виділяють чотири класи з надання можливих затримок у передачі даних, проте на мережі оператора можна реалізувати лише частину з них, наприклад одним із можливих варіантів є надання т.зв. «best effort delay class» (класу з найменшою затримкою за заданих умов), що відповідає Class 4.

При цьому загальні тимчасові затримки через всю опорну пакетну мережу GPRS як і раніше класифіковані в специфікаціях, тому що при «переходах» пакетів даних, наприклад, від радіоінтерфейсу до опорної мережі можуть з'явитися свої часові затримки. Вимоги до максимальної затримки залежать від довжини пакетів, наприклад, при передачі коротких пакетів вимоги до затримки дещо жорсткіші;

– *Precedence class*. У цьому атрибуті визначені три основних класи, що визначають важливість пакетів, що проходять, на підставі якої пакети з меншою важливістю будуть відкинуті у разі підвищеного навантаження або обмеженості ресурсів;

– *Peak Throughput class*. Цей клас визначає максимальну пропускну спроможність (bytes/s), яку очікує отримати користувач за весь час сесії. Однак, немає жодних гарантій, що така пропускну спроможність буде досягнута за вказаний проміжок часу, оскільки цей фактор буде залежати як від можливостей мобільної станції (MS), так і від радіоресурсів, що виділяються. У цьому атрибуті визначено 9 основних класів (рис. 2.2), починаючи від Class 1 – 1000 bytes/s (8 kbit/s) до Class 9 – 256000 bytes/s (2048 kbit/s). При цьому ємність пропускну здатності, що надається, подвоюється від одного класу до іншого;

– *Mean Throughput class*. Пропускнну здатність оцінюють на тих самих інтерфейсах, що і пікову.

Цей атрибут визначає ресурси, що виділяються під швидкість передачі даних, що очікує отримати з боку кінцевого пристрою під час активації лінка. Загалом виділяють 19 класів, від Class 1 до Class 18 – що надає швидкість передачі на рівні 111 kbit/s, при цьому 18 класу будуть виділені максимальні ресурси, щоб забезпечити потрібну швидкість. Можливий також «best effort» клас, що забезпечує максимально можливу швидкість, йому надано номер 31.

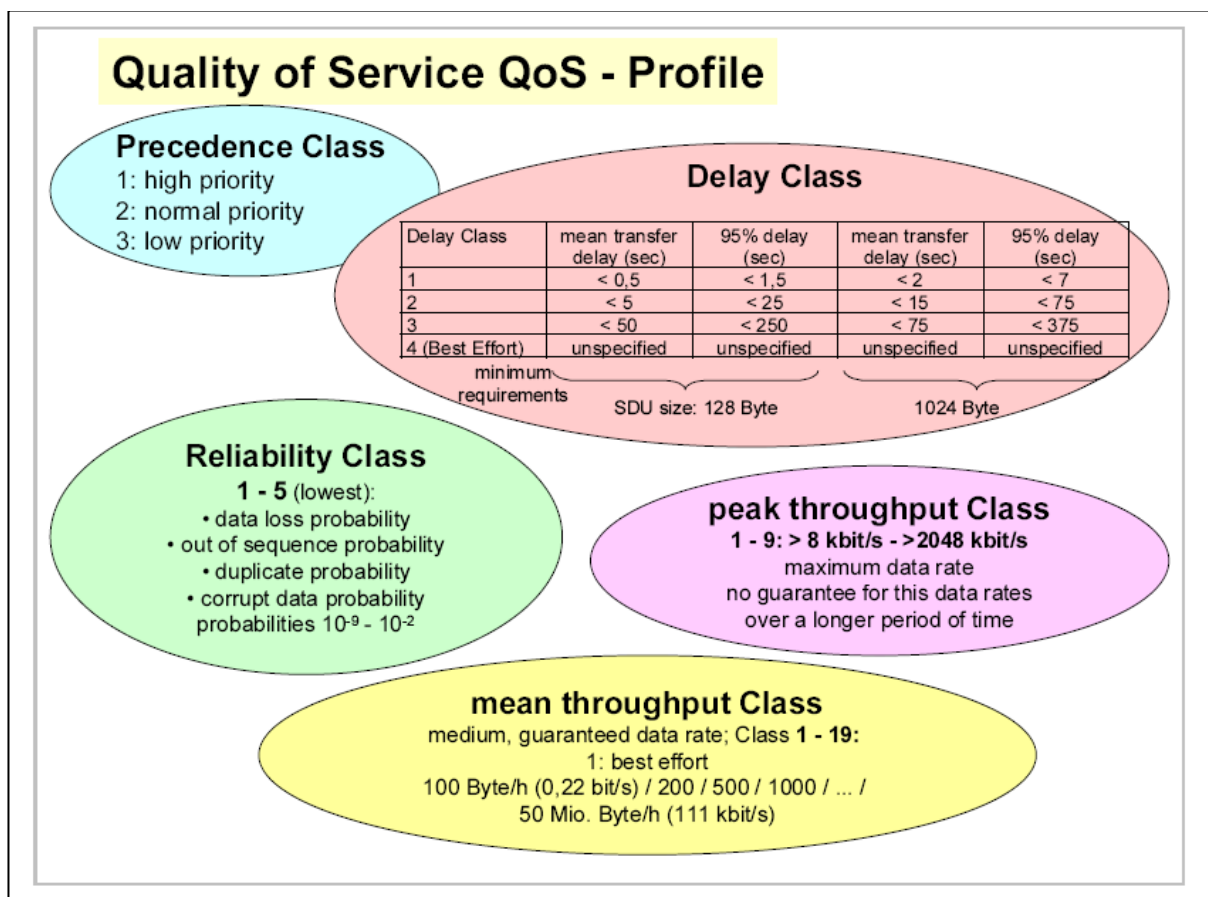


Рисунок 2.2 – Класи QoS схематично

На продуктивність системи можуть в основному впливати три фактори:

- 1) простір пам'яті;
- 2) затримка розвантаження/видалення;
- 3) обробка пакетів затримки потоку, коли він обробляється в ядрі та шляхах.

Завдяки численним експериментальним вимірюванням виявлено, що затримка обробки пакетів потоку в різних шляхах даних може бути емпірично змодельована як:

$$T_i = \{ \square_2, C_i \leq C_i^{kr} \square_3, C_i > C_i^{kr} \square_1, \quad \text{в системі} \quad , \quad (2.1)$$

де T_i , C_i та C_i^{kr} – затримка обробки пакетів, швидкість надсилання та пропускна здатність потоку даних ядра відповідно.

Крім цього, існує питання про витрати ресурсів акумулятора під час використання пакетних сервісів за допомогою технологій GPRS/EDGE.

Доступ до пакетної мережі оператора з боку абонента відбувається через процедуру GPRS Attach (або Combined Attach – одночасне підключення як до голосових, і до пакетних сервісів), тобто, абонент повинен пройти авторизацію та аутентифікацію на мережі оператора. При цьому, якщо абонент не починає активувати PDP Context-ів, то він знаходиться в так званому IDLE-стані. У цьому стані абонент поки що не може приймати або надсилати дані через пакетну мережу. Щоб почати відправляти/приймати дані, йому необхідно активувати хоча б один PDP Context.

Виходячи з усього вищесказаного, щоб визначити рівні використання акумулятора пристрою абонента, мають бути розглянуті дві наступні ситуації:

- коли абонент знаходиться у GPRS Attach, але не активував жодного PDP Context (GPRS Attach, no PDP Context activated);
- коли абонент перебуває в GPRS Attach, та активував хоча б один PDP Context (GPRS Attach, one PDP Context activated).

У першому випадку абонент не передає і не приймає жодних пакетних даних, а лише може оновлювати своє місцезнаходження в пакетній мережі оператора. Обладнання може передавати у мережу лише службові дані (наприклад, безперервний пінг, який повідомляє про те, що обладнання на

зв'язку). В такому разі споживання акумулятора терміналу буде мінімальним. Витрата акумулятора мало змінюється при постійному знаходженні абонента в GPRS Attach, оскільки мобільний термінал практично не використовує радіомережу та не передає активних даних оператору.

З іншого боку, при використанні пакетних даних досить зростає споживання акумуляторної батареї апарата в порівнянні з використанням голосових сервісів.

Це в першу чергу обумовлено тим, що під пакетні послуги на базовій станції може бути виділено кілька тайм-слотів (TS), які можуть бути розподілені між різним обладнанням.

Існує максимально 8 тайм-слотів на одному TRX-стілнику, але при цьому вони динамічно поділяються між сервісами з можливим визначенням пріоритетів (рис. 2.3).

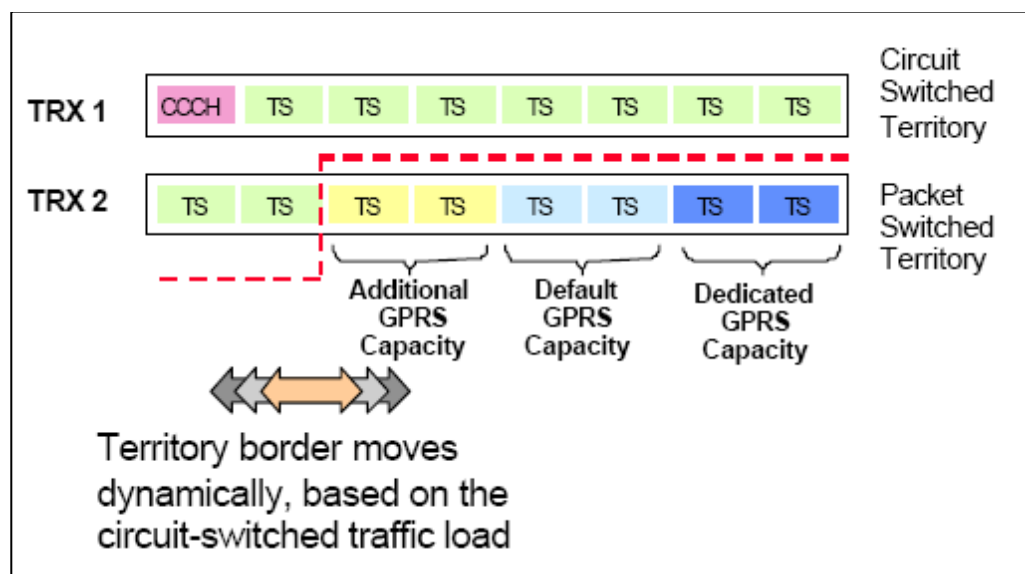


Рисунок 2.3 – Схема виділення радіоресурсів з боку базової станції під пакетні дані

Тож, чим більше TS виділиться одному абоненту на одному TRX, тим більша швидкість йому буде надана. На швидкість також впливатиме схема кодування потоку даних наведена в табл. 2.1.

Таблиця 2.1 – Основні схеми кодування для технології GPRS

Channel Coding	CS1	CS2	CS3	CS4
Single TS date rate, kbit/s	9,0 5	13, 40	15, 60	21, 40
8 TS date rate, kbit/s	72, 00	107 ,20	124 ,80	171 ,20

Таким чином, використовуючи пакетні дані абоненту може бути виділено більше TS на стороні базової станції, що, у свою чергу, призведе до збільшення потужності приймача на мобільному терміналі абонента і як наслідок збільшить витрату енергозапасу акумулятора.

Саме тому під час використання пакетних сервісів на основі технологій GPRS/EDGE значно збільшується витрата батареї мобільного терміналу.

2.2 Аналітична модель справедливого обслуговування черги банкоматів

Однією з головних завдань при побудові сучасних мультисервісних мереж з пакетною комутацією є забезпечення необхідної якості обслуговування різних видів трафіку (мовленнєвий зв'язок, передача даних, потокове відео або інше мультимедійне навантаження). Цим обумовлена важливість теоретичного аналізу процесів передачі неоднорідних потоків інформації з урахуванням топології проектованої або діючої мережі в умовах реальної або прогнозованого інформаційного навантаження.

Для забезпечення якісного надання користувачам послуг зв'язку необхідно на постійній основі проводити моніторинг якості роботи мережевої інфраструктури, що забезпечує передачу трафіку в інтересах користувача. Оскільки маршрутизація пакетів у пакетній мережі зв'язку

носить імовірнісний характер, то для передачі пакетного трафіку в інтересах користувача необхідно проводити загальну оцінку якості роботи мережі по передачі трафіку між точками підключення користувацьких мережевих пристроїв та точками підключення або виходу із мережі в напрямку сервісних платформ, до яких користувач звертається для отримання необхідних йому послуг. В інтересах оператора щодо забезпечення якості роботи мережі на належному рівні проводити оцінку якості функціонування окремих сегментів та елементів мережі з передачі трафіку користувача. Кореляція змін у показниках якості роботи сегментів та елементів мережі зі змінами в показниках якості, які характеризують роботу мережі з передачі користувацького трафіка, дозволить операторам виявити найбільш важливі і навантажені сегменти та елементи мережі для проведення найбільш ефективних заходів по забезпеченню підтримки якості роботи мережі на найвищому рівні. При проведенні оцінки якості роботи мережі для забезпечення узгодженості отриманих результатів операторам необхідно використовувати єдині методи та засоби для проведення такої оцінки.

Пакетні мережі використовуються для передачі пакетів даних між двома або більше підключеними пристроями. Індикатори якості використовуються для вимірювання продуктивності пакетних мереж, і їх можна використовувати для оцінки як якості передачі, так і загальної продуктивності мережі.

Нижче проаналізовано пакетні мережі за показниками якості.

Втрата пакетів відбувається, коли один або кілька пакетів втрачаються під час передачі. Це може бути викликано перевантаженням мережі, апаратним збоєм або іншими факторами. Високий рівень втрати пакетів може погіршити продуктивність мережі, що призведе до зниження пропускної здатності даних і збільшення затримки.

Коефіцієнт втрат IP-пакетів – це показник, який використовується для вимірювання відсотка IP-пакетів, втрачених під час передачі через мережу. Цей показник важливий для оцінки якості та надійності мережевого зв'язку.

Коефіцієнт втрат IP-пакетів обчислюється шляхом ділення кількості втрачених пакетів на загальну кількість переданих пакетів, а потім помножується результат на 100, щоб отримати відсоток.

Затримка передавання IP-пакетів, також відома як мережева затримка або наскрізна затримка, – це час, потрібний IP-пакету для проходження від пристрою-джерела до пристрою призначення через мережу.

Затримку передачі IP-пакетів можна розділити на кілька компонентів:

- затримка обробки: це час, потрібний вихідному пристрою для обробки пакета та підготовки його до передачі. Це може включати такі завдання, як інкапсуляція пакета у кадрі канального рівня, обчислення контрольних сум, і виконання інших функцій, що стосуються протоколу;

- затримка в черзі: це час, який пакет проводить в очікуванні в черзі мережі, перш ніж його можна буде передати. Ця затримка може бути викликана перевантаженням мережі або іншими факторами, які призводять до великого обсягу трафіку;

- затримка передавання: це час, потрібний для передачі пакету через фізичне середовище, таке як кабель або бездротове з'єднання. Затримка передачі визначається довжиною пакета і швидкістю передачі мережевого середовища;

- затримка розповсюдження: це час, який потрібен пакету для проходження від пристрою джерела до пристрою призначення через фізичне середовище. Затримка розповсюдження визначається відстанню між пристроями та швидкістю світла в середовищі передачі.

Варіація затримки IP-пакетів (також відома як «джиттер» пакетів) є показником мінливості часу, необхідного IP-пакетам для проходження від пристрою джерела до пристрою призначення через мережу.

Варіація затримки пакетів зазвичай виражається як різниця між максимальною та мінімальною затримками послідовних IP-пакетів або як дисперсія значень затримки протягом певного інтервалу часу.

Низька варіація затримки пакетів вказує на те, що пакети надходять на пристрій призначення з відносно постійною затримкою, тоді як висока варіація затримки пакетів вказує на коливання затримки пакетів.

На зміну затримки пакетів впливає кілька факторів, зокрема перевантаження мережі, черги і використання кількох мережевих шляхів або протоколів маршрутизації. У перевантаженій мережі пакети можуть мати більшу затримку в черзі, що може збільшити варіацію затримки пакетів. Подібним чином використання кількох мережевих шляхів або протоколів маршрутизації може призвести до того, що пакети будуть проходити різними маршрутами з різними затримками поширення, що також може збільшити варіацію затримки пакетів.

Коефіцієнт помилок IP-пакетів – це показник, що вимірює відсоток IP-пакетів, які містять помилки під час передачі через мережу; розраховується шляхом ділення кількості пакетів з помилками на загальну кількість переданих пакетів, а потім множення результату на 100, щоб отримати відсоток.

Коди виправлення помилок можна використовувати для виявлення та виправлення помилок у пакетах даних шляхом додавання додаткових бітів до даних, які можна використовувати для виявлення та виправлення помилок.

Регенерацію сигналу можна використовувати для посилення та регенерації мережевих сигналів для подолання шуму та перешкод.

Коефіцієнт перевпорядкування IP-пакетів – це показник, який вимірює відсоток IP-пакетів, отриманих не в порядку на пристрої призначення. Перевпорядкування пакетів може відбуватися, коли пакети проходять різними шляхами в мережі або коли вони буферизуються або затримуються в різних точках мережі.

Коефіцієнт перевпорядкування IP-пакетів обчислюється шляхом ділення кількості пакетів, отриманих не за порядком, на загальну кількість отриманих пакетів, а потім множення результату на 100, щоб отримати відсоток.

Запобігти перевпорядкуванню пакетів можна за допомогою методів упорядкування пакетів і міток часу, щоб забезпечити отримання пакетів у правильному порядку на пристрої призначення. Коди виправлення помилок можна використовувати для виявлення та виправлення помилок у пакетах даних, що може допомогти запобігти зміні порядку пакетів.

Коефіцієнт дублікатів IP-пакетів – це показник, який вимірює відсоток IP-пакетів, отриманих на пристрої призначення більше одного разу. Дублювання пакетів може статися, коли пакети буферизуються або затримуються в різних точках мережі або коли вони передаються кілька разів через помилки мережі, або перевантаження.

Коефіцієнт дублікатів IP-пакетів обчислюється шляхом ділення кількості дублікатів пакетів на загальну кількість отриманих пакетів, а потім множення результату на 100, щоб отримати відсоток.

Контроль перевантажень можна використовувати для виявлення та реагування на перевантаження мережі шляхом коригування швидкості трафіку та мережевих ресурсів, що може допомогти зменшити дублювання пакетів.

Найбільші проблеми мережі пов'язані з розподілом мережевих ресурсів, таких як пропускна здатність каналу та буфери для різних користувачів. Обмежена кількість ресурсів повинна бути розподілена між різними потоками трафіку таким чином, щоб розширити використання ресурсів мережі та продуктивність.

Різні види послуг можуть бути досягнуті, коли поведінка маршрутизаторів з точки зору обробки пакетів контролюється. Щоб керувати

тим, які пакети відкидаються або передаються, можна використовувати різні методи постановки в чергу.

Пріоритетна черга (англ. Priority Queuing, PQ) використовується для типу трафіку, який не може обробляти втрату пакетів і чутливий до затримки. PQ створює кілька черг, і кожна черга має рівень пріоритету. Черга з вищим пріоритетом керується раніше черги з нижчим пріоритетом. PQ має чотири відомі черги: низький, нормальний, середній і високий пріоритет.

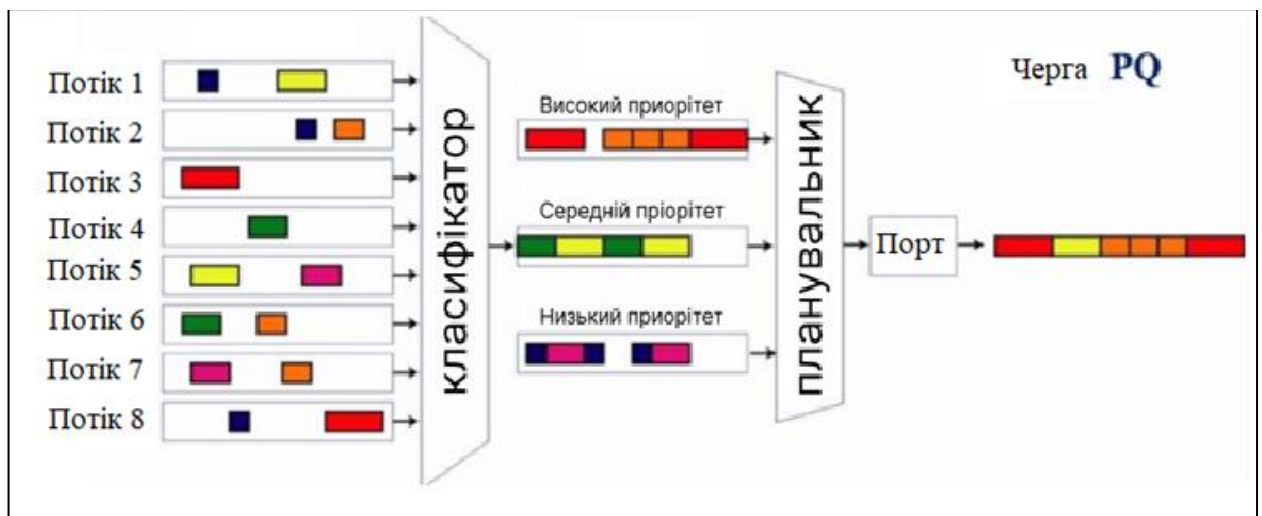


Рисунок 2.4 – Пріоритетна черга (PQ)

Пріоритетна черга скидає всі свої дії, коли пакети надходять у старшу чергу для передачі цих пакетів. Цей алгоритм дуже ефективний для важливого трафіку, але може призвести до втрати черги. Проблема виникає з пакетами, які мають нижчий пріоритет у черзі.

Нижче наведено формули для визначення певних параметрів мережі для визначення черги згідно з моделлю справедливого обслуговування.

Середня затримка передачі пакетів:

$$W_k = \frac{\sum_{i=1}^K \rho_i}{(1-\rho''_k)(1-\rho''_{k-1})} \quad (2.2)$$

Сумарна інтенсивність навантаження потоків від першого до k -го пріоритету включно:

$$\rho''_i = \sum_{k=1}^i \rho_k \quad (2.3)$$

$$\sum_{k=1}^K (r_k) = C' \dots \quad (2.4)$$

Мінімальна величина швидкості обслуговування (рис. 2.5):

$$r_k = \frac{w_k}{\sum_{i=1}^K w_i} \quad (2.5)$$

Згідно з (2.5):

$$\begin{aligned} r_1 &= \frac{3}{3+1+1} 10 = 6 & (8,6) &= 6 \\ r_2 &= \frac{1}{3+1+1} 10 = 2 & (6,2) &= 2 \\ r_3 &= \frac{1}{3+1+1} 10 = 2 & (2,2) &= 2 \end{aligned}$$

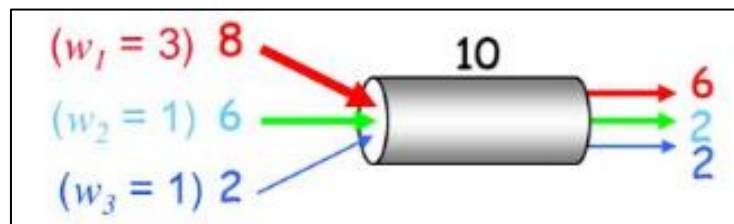


Рисунок 2.5 – Розрахунок мінімальної величини швидкості обслуговування

Потенційно обслуговуване навантаження окремого потоку:

$$\rho'_i = \frac{\lambda_i}{r_i} \quad (2.6)$$

Еквівалентна межа відносно затримки обслуговування:

$$\psi_k = B_k / r_k, \quad (2.7)$$

де K – число потоків ($K = 8$).

Множина всіх потоків:

$$K \in \{i\}_{i=1, \dots, K}. \quad (2.8)$$

Множина потоків за виключенням k -го:

$$J \in \{i\}_{i=1, \dots, K, i \neq k}. \quad (2.9)$$

Нижня межа затримки передачі пакетів:

$$W'_k = \frac{\frac{1}{2} \sum_{i=1}^K \rho'_i}{-\rho_k}. \quad (2.10)$$

Верхня межа затримки передачі пакетів:

$$W''_k = \frac{\frac{1}{2} \sum_{i=1}^K \dots + \dots}{1 - \rho_k} \quad (2.11)$$

Розрахований час закінчення передачі для $K = 8$ потоків наведено на рис. 2.6.

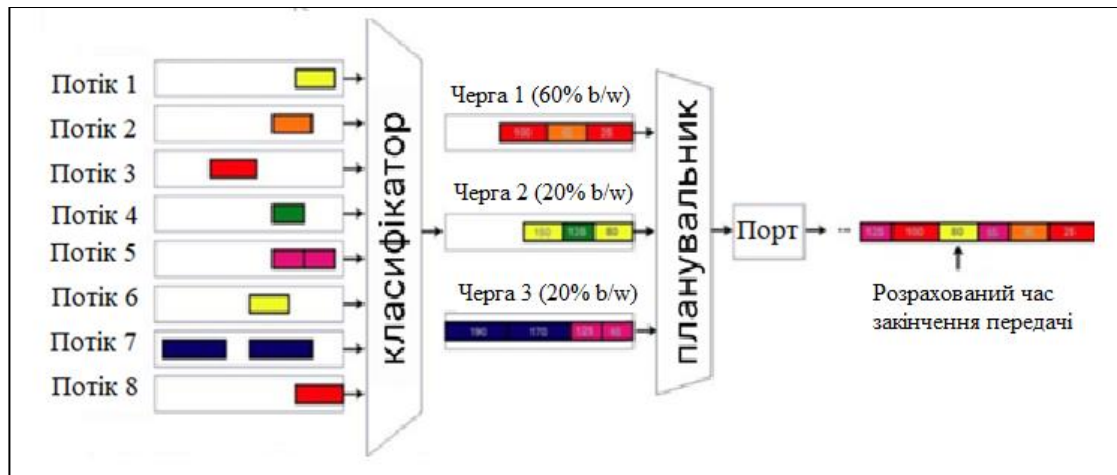


Рисунок 2.6 – Розрахований час закінчення передачі

Наведені вище формули допомагають вирахувати середню затримку передачі пакетів, швидкість обслуговування, чергу відключення банкоматів у різних районах міста залежно від навантаження мережі.

Пріоритетна диспетчеризація, при якій кожен пакет має пріоритет: високопріоритетні пакети відправляються раніше ніж низькопріоритетні; останні розміщуються у буфері. Пакети з однаковим пріоритетом відправляються за принципом FIFO.

Зважене справедливе обслуговування – більш кращий варіант. Якщо присвоїти високопріоритетній черзі більшу вагу (наприклад, 3), високопріоритетні пакети будуть в основному проходити по швидкій лінії (так як пакетів з високим пріоритетом відносно небагато), але одночасно з цим частина низькопріоритетних пакетів теж буде передаватися.

Алгоритм диспетчеризації пакетів функціонує на основі порівняння двох віртуальних часів: системного потенціалу та потокового потенціалу.

Пакетні мережі використовуються для передачі пакетів даних між двома або більше підключеними пристроями. Індикатори якості використовуються для вимірювання продуктивності пакетних мереж, і їх можна використовувати для оцінки як якості передачі, так і загальної продуктивності мережі.

В якості прикладу для перевірки передачі пакетних даних до розетки через MAC-адресу використано утиліту ping, яка перевіряє доступність хоста в мережі. Зазначена утиліта доступна практично для всіх операційних систем, які мають мережеві можливості, включаючи більшість вбудованих програм для адміністрування мережі. Для розетки з MAC-адресою fe80::86d7:7e79:3c01:7940%6 середнє значення луна-пакету становить 5 мс (рис. 2.7, а), а для розетки з MAC-адресою fe80::9b07:ea9e:39d9:4245%13 середнє значення луна-пакету становить 11 мс (рис. 2.7, б):

```
C:\Users\Nast>ping fe80::86d7:7e79:3c01:7940%6
Обмен пакетами с fe80::86d7:7e79:3c01:7940%6 по с 32 байтами данных:
Ответ от fe80::86d7:7e79:3c01:7940%6: время<1мс
Ответ от fe80::86d7:7e79:3c01:7940%6: время<12мс
Ответ от fe80::86d7:7e79:3c01:7940%6: время<16мс
Ответ от fe80::86d7:7e79:3c01:7940%6: время<1мс
Статистика Ping для fe80::86d7:7e79:3c01:7940%6:
  Пакетов: отправлено = 4, получено = 4, потеряно = 0
  (0% потерь)
Приблизительное время приема-передачи в мс:
  Минимальное = 1 мсек, Максимальное = 16 мсек, Среднее = 5 мсек
C:\Users\Nast>
```

а)

```
C:\Users\Nast>ping fe80::9b07:ea9e:39d9:4245%13
Обмен пакетами с fe80::9b07:ea9e:39d9:4245%13 по с 32 байтами данных:
Ответ от fe80::9b07:ea9e:39d9:4245%13: время<18мс
Ответ от fe80::9b07:ea9e:39d9:4245%13: время<23мс
Ответ от fe80::9b07:ea9e:39d9:4245%13: время<26мс
Ответ от fe80::9b07:ea9e:39d9:4245%13: время<14мс
Статистика Ping для fe80::9b07:ea9e:39d9:4245%13:
  Пакетов: отправлено = 4, получено = 4, потеряно = 0
```

(0% потерь)
 Приблизительное время приема-передачи в мс:
 Минимальное = 14мсек, Максимальное = 26 мсек, Среднее = 21 мсек
 C:\Users\Nast>

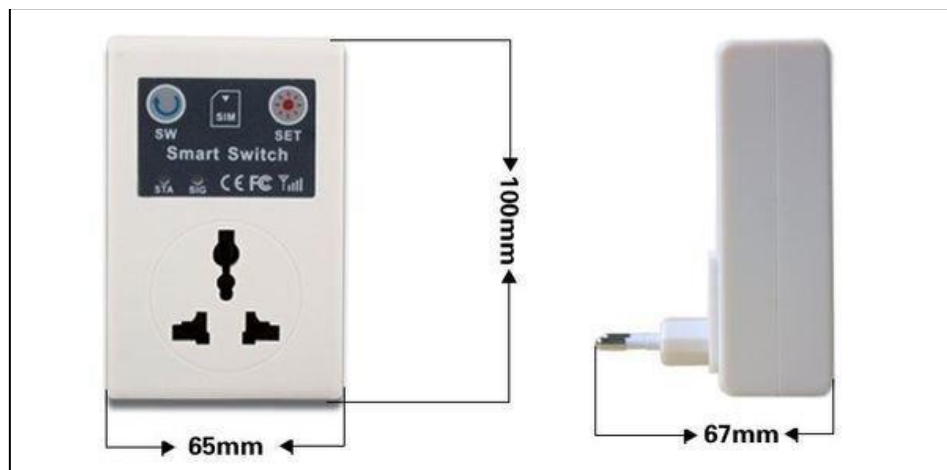
б)

Рисунок 2.7 – Перевірка доступності хостів (банкоматів) у мережі

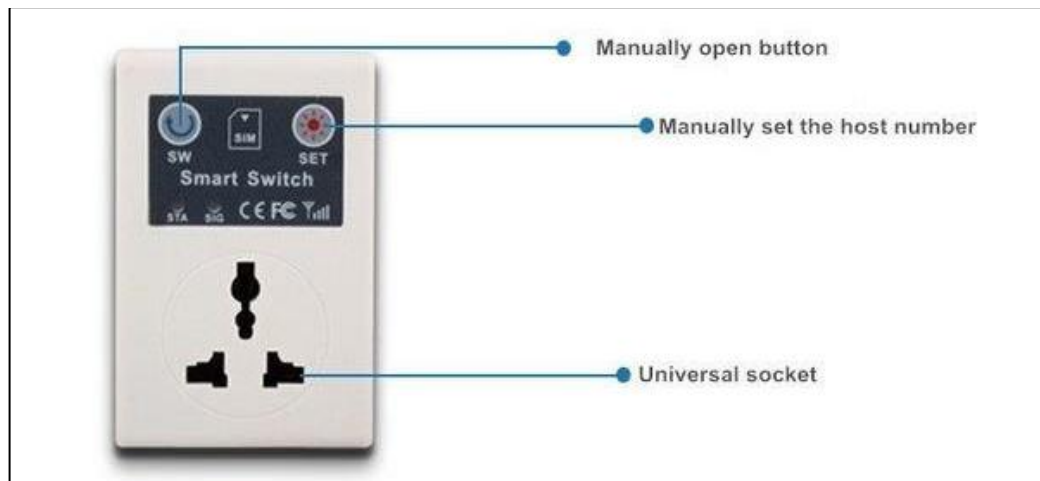
Згідно з вищезазначеною перевіркою, першими у черзі на підключення/відключення будуть ті хости, які мають менший пінг у мс, тобто, перший за зверненням (рис. 2.7, а).

2.3 Аналіз апаратного забезпечення підключення банкоматів до комп'ютерної мережі

SC1-GSM – це GSM-розетка дистанційного керування на базі GSM, запропонована KONLEN, яка дозволяє користувачам дистанційно керувати електроприладами та іншим обладнанням через мобільну мережу (рис. 2.8).



а)



б)

Рисунок 2.8 □ Характеристики GSM-розетки: а □ розміри;
б □ функціональні модулі

Після встановлення SIM-карти в пристрій та підключення до електромережі, розетка починає працювати через 30 секунд та може бути з'єднана з будь-яким пристроєм, яким потрібно керувати (рис. 2.9).



Рисунок 2.9 □ Керування розеткою за допомогою різних пристроїв

GSM-розетка приймає та надсилає SMS-повідомлення, а також відповідає на голосові дзвінки. Якщо контроль буде здійснюватися за допомогою SMS, то номер картки, яка використовується в смарт-розетці, заздалегідь має бути внесений до телефонної книги. Також GSM-розетка передбачає керування голосом – вона включає або вимикає прилад, підключений до неї.

Згідно з технічними характеристиками, розетка має керування через SMS-повідомлення, а також дзвінки.

Але, оскільки процедура реєстрації займає багато часу, спеціаліст стикається з низкою проблем при обслуговуванні GSM-модуля. Зокрема, якщо для керування надана ціла система банківського обладнання, тобто, наприклад, не один банкомат, а декілька. На рис. 2.10 наведена структурна схема наданої системи.

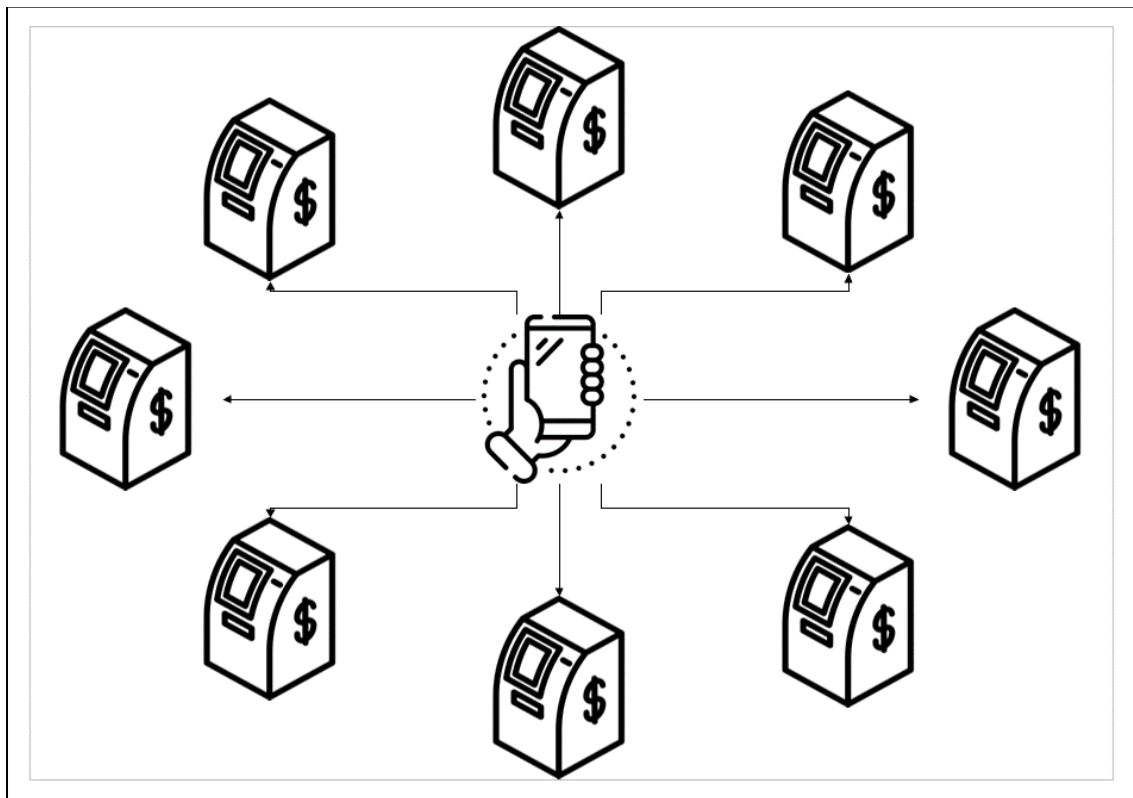


Рисунок 2.10 □ Схема підключеного банківського обладнання

Налаштування доступними способами займає велику кількість часу, так як потрібно надіслати повідомлення на кожен модуль, вказуючи у повідомленні достатньо довгу послідовність символів, яка являється певною командою.

У випадку з дзвінками фахівець стискається з ще більшими труднощами. Окрім того, що спеціаліст мусить дозвонитися на кожен номер, він також має кожен раз прослуховувати команди, які розташовані за нумерацією від одного до восьми.

Тож, дана процедура займає достатньо довгий проміжок часу. Тому було вирішено створити застосунок, який дозволяє керувати підключенням системи банківського обладнання.

2.4 Аналіз програмного забезпечення розробки застосунків для віддаленого керування банкоматами

Для виконання поставленої задачі було вирішено створити застосунок на смартфон на базі операційної системи (ОС) Android.

Android — це операційна система з відкритим кодом, яка використовується переважно для мобільних пристроїв, таких як смартфони та планшети. Він базується на ядрі Linux і вперше був розроблений компанією Android Inc., яку пізніше придбала Google. Android розроблено для роботи на багатьох пристроях і має велику екосистему програм, з мільйонами програм, доступних для завантаження через магазин Google Play. Це найпоширеніша мобільна операційна система у світі. За різними підрахунками за 2020 рік цією операційною системою користуються близько 85 % власників смартфонів, а загальна кількість користувачів смартфонів на ОС Android оцінюється більш ніж у 2,5 млрд людей по всьому світу.

Android постійно еволюціонує, та разом з операційною системою еволюціонують засоби та інструменти для розробки.

Android Studio – це інтегроване середовище розробки (англ. Integrated Development Environment, IDE) для розробки програм на операційну систему Android (рис. 2.11). Воно засноване на IntelliJ IDEA від JetBrains і забезпечує уніфіковане середовище для створення програм.

Середовище має вбудовані інструменти для проєктування, тестування та налагодження, а також пропонує підтримку Google Cloud Platform, дозволяючи легко інтегрувати хмарні служби у застосунок. Android Studio надає можливість розробляти програми для телефонів, планшетів, Android TV і Android Wear.

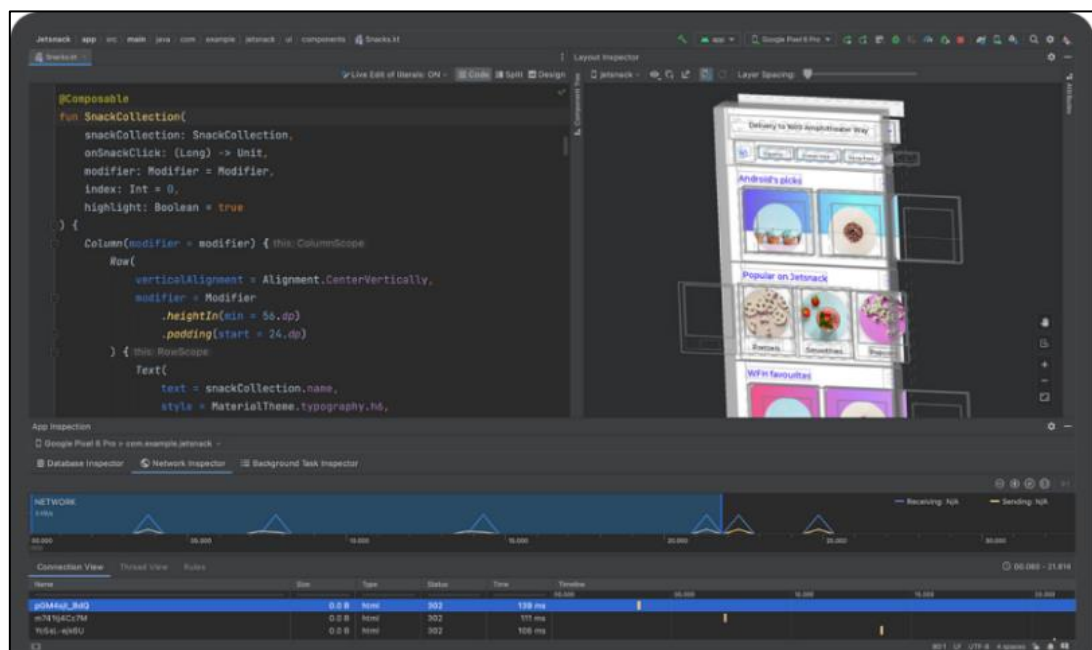


Рисунок 2.11 □ Інтерфейс Android Studio

Варто зазначити, що розробляти програми під Android можна за допомогою різних фреймворків та мов програмування. Так, як мови програмування можуть застосовуватися Java, Kotlin, Dart (фреймворк Flutter), C++, Python, C# (платформа Xamarin) тощо. При виконанні даної роботи було використано мову програмування Java, як найбільш поширену та використовувану.

Висновки до розділу 2

У розділі 2 було розглянуто основні принципи QoS, можуть допомогти підвищити загальну продуктивність і надійність мережі. Контролюючи пропускну здатність, доступну для різних типів трафіку, QoS допомагає забезпечити отримання критично важливих програм ресурсами, необхідними для ефективної роботи, тоді як некритичні програми не споживають надмірну кількість мережевих ресурсів.

Оскільки система, з якою відбувається взаємодія в даній роботі, має в своєму складі не одну одиницю банківського обладнання, а декілька, виникає питання черги, яка має бути розвантажена.

Крім цього, існує питання про витрати ресурсів акумулятора під час використання пакетних сервісів з технологіями GPRS/EDGE, при використанні яких збільшується витрата батареї мобільного терміналу. Також розглянуто типи пакетних мереж та їх індикатори якості, які використовуються для вимірювання продуктивності.

В розділі також було розглянуто апаратне та програмне забезпечення, яке є доцільним для виконуваної роботи. Як вихідні дані для роботи було використано GSM-розетку та мобільний телефон.

GSM-розетка приймає та надсилає SMS-повідомлення, а також відповідає на голосові дзвінки, що дозволяє дистанційно керувати підключеним банківським обладнанням.

Оскільки подібне керування не є зручним, то має сенс створення застосунку, який допомагає спеціалісту у налаштуванні та керуванні обладнанням. Для цього обрано Android Studio – інтегроване середовище розробки та мова програмування Java.

Також розглянуто основні етапи розробки мобільного застосунку, такі як планування та аналіз, дизайн, розробка, тестування, розгортання і технічне обслуговування.

3 СТВОРЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ СИСТЕМИ

3.1 Етапи розробки мобільного застосунку

SDLC – життєвий цикл розробки програмного забезпечення, який являє собою серію етапів, що описують процес створення програмного забезпечення. Метою SDLC є забезпечення структурованої та ефективної розробки програмного забезпечення, задоволення вимог замовника та забезпечення високоякісного програмного забезпечення (рис. 3.1).

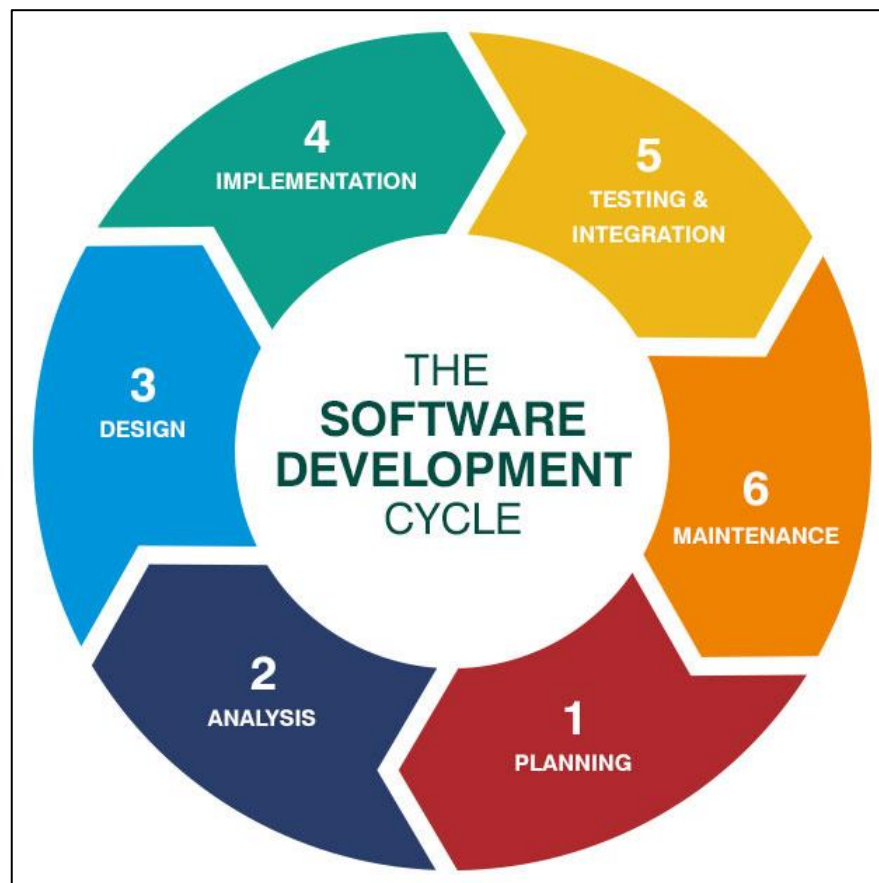


Рисунок 3.1 – Цикл розробки програмного застосунку

Незважаючи на те, що в життєвому циклі розробки програмного забезпечення не існує єдиного остаточного переліку етапів, все ж варто окреслити ті, які найчастіше є частиною різних методологій і структур:

- планування та аналіз;

- дизайн;
- розробка;
- тестування;
- розгортання;
- технічне обслуговування.

Процеси *планування та аналізу* часто об'єднують в одну фазу життєвого циклу розробки програмного забезпечення. Але вони так само часто поділяються на дві окремі фази.

Під час першої фази аналізується економічне обґрунтування розробки програмного забезпечення.

Серед питань, які зазвичай ставляться на цьому етапі, є:

- яку проблему вирішить програмне забезпечення? Чого потенційно варте вирішення цієї проблеми?
- наскільки велику базу потенційних користувачів має програмне забезпечення?
- чи існує інше програмне забезпечення, яке вирішує ту саму проблему?
- як це програмне забезпечення покращить це?
- скільки може коштувати розробка програмного забезпечення для ефективного вирішення проблеми?
- чи вирішення проблеми достатньо цінне, щоб запропонувати прийнятну віддачу від інвестицій за таку вартість?
- чи є у нас ресурси для покриття цих витрат?

Припущення, що містяться у відповідях на ці запитання, слід перевірити, бажано за допомогою таких методів, як фокус-групи користувачів, або інші інструменти дослідження ринку, включаючи такі речі, як онлайн-опитування, розробка технічного прототипу тощо.

Обсяг проєкту або початкова оцінка обсягу в гнучкому підході буде обговорюватися тут на основі початкових вимог, як і планування ресурсів, необхідних для реалізації проєкту, а потім їх впровадження.

Після перевірки потреби в програмному забезпеченні та визначення його початкового обсягу, вимог і очікуваних ресурсів на етапі аналізу та планування розпочинається етап *дизайну*.

Саме тут приймається рішення щодо високорівневого дизайну програмної системи. Буде вибрано технічний стек, наприклад, які мови, фреймворки та бібліотеки використовуватимуться, а також інші деталі.

Обсяг і функції MVP, який буде випущений першим, також слід визначити тут, оскільки це буде інформувати про наступний етап – розробку.

На етапі *розробки* програмного забезпечення продукт кодується та збирається розробниками. У гнучкому підході, який використовують більшість значущих проєктів розробки програмного забезпечення, перша фаза розробки створить MVP.

У водоспадному підході те, що розглядається як готовий продукт, документується в технічних деталях від початку до кінця на етапі проєктування, а потім створюється від початку до кінця на етапі розробки.

Під час етапу *тестування* програмне забезпечення, розроблене на попередньому етапі, перевіряється на відповідність вимогам, визначеним на етапі планування та аналізу, а також на продуктивність і помилки. Мета етапу полягає в тому, щоб переконатися, що програмний продукт розроблено за призначенням, як з точки зору взаємодії з користувачем, так і ресурсів, необхідних для безперебійної роботи та відповідності вимогам доступності та безпеки.

Тестування програмного забезпечення зазвичай передбачає як ручне тестування шляхів користувача та досвіду, так і автоматизоване тестування на основі інструментів і сценаріїв. Якщо виявлені помилки або інші дефекти, тестувальники повинні задокументувати проблему та повідомити групу

розробників, яка створить нову версію програмного забезпечення, у якій проблеми вирішено. Цю версію програмного забезпечення буде перевірено знову, якщо оновлення призвели до небажаних наслідків і створили нові помилки чи дефекти.

На етапі *розгортання* програма переміщується із середовища розробки та тестування в живе середовище, де вона доступна для користувачів. Розгортання може бути відносно простим процесом або надзвичайно складним, залежно від характеру програмного забезпечення. Наприклад, якщо новий програмний продукт включає базу даних, яку використовуватимуть інші системи, інтеграція їх усіх може представляти значне завдання.

Вирішальним кроком на етапі розгортання також є переконання, що програмне забезпечення працює точно так само в реальному середовищі, як і в середовищах розробки та тестування.

Етап *технічного обслуговування* включає перевірку того, що програмне забезпечення продовжує працювати відповідно до його призначення з точки зору функцій, продуктивності та доступності. Користувачі та поточне тестування можуть виявити помилки та інші дефекти або проблеми безпеки, не знайдені під час фази тестування, які слід виправити.

Вони можуть бути незначними або вимагати запуску нового етапу розробки.

Життєвий цикл розробки програмного забезпечення може і адаптується групами розробників програмного забезпечення на основі філософії, методології та структури, які вони приймають під час створення певного програмного продукту або між організаціями.

SDLC — це інструмент у системі керування проектами, який має відповідати потребам конкретного проекту, команди, яка над ним працює, та інших ключових зацікавлених сторін, залучених до процесу. Змінюються назви фаз, їх порядок або якщо вони є різними або злитими одна в одну.

3.2 Планування та аналіз

Створення застосунків для мікроконтролерів абсолютно відрізняється від будь-якої іншої роботи з розробки комп'ютерів та електроніки. У більшості інших застосунків доступна низка підсистем та інтерфейсів, але це не стосується мікроконтролера, де необхідно виконувати наступні обов'язки:

- розподіл електроенергії;
- синхронізація системи;
- дизайн інтерфейсу та підключення;
- системне програмування;
- прикладне програмування;
- програмування пристрою.

Створювана система віддаленого керування вирішує проблему налаштування та підключення\відключення банківського обладнання, використовуючи модулі GSM та оптимізуючи при цьому роботу технічного спеціаліста.

Використовуваний GSM-модуль може керуватися дистанційно за допомогою дзвінків або SMS-повідомлень. Але зовнішнє середовище, наприклад, шум, може завадити співробітнику коректно застосувати налаштування до системи.

Також слід пам'ятати, що кількість банкоматів в системі завелика для контролю їх безпосередньо на місці.

Отже, виходячи з вищенаведених проблем та важливістю збереження світломаскування в містах і заощадження електроенергії, є сенс створення застосунку, який пришвидшить дану процедуру.

Подібні застосунки, які створені для GSM-розеток, вирішують в основному суто цивільні проблеми, пов'язані, наприклад, з контролем побутової техніки.

У будь-якій мобільній програмі необхідно проектувати архітектуру коду. Погана архітектура веде до ускладнення внесення змін до нього, тестування, до зниження читабельності, тобто загального розуміння роботи програми, що у свою чергу збільшує кількість помилок і призводить до неправильної роботи програми.

Існує кілька архітектур, які використовуються для побудови взаємозв'язків між компонентами програми. Це MVC, MVP, MVVM, VIPER.

3.3 Дизайн

Дизайн інтерфейсу користувача є фактором, що впливає на три основні показники якості програмного продукту: його функціональність, естетику та продуктивність.

Функціональність є фактором, на який розробники застосунків часто звертають основну увагу. Вони намагаються створювати програми так, щоб користувачі могли виконувати свої завдання, і їм було зручно це робити. Функціональність важлива, проте це не єдиний показник, який має враховуватись у ході розробки.

Естетичний зовнішній вигляд самого застосунку та способу його уявлення дозволяє сформувати у споживача позитивну думку про програму. Однак естетичні характеристики дуже суб'єктивні та описати їх кількісно набагато важче, ніж функціональні вимоги чи показники продуктивності. Вся естетика програми часто зводиться до простого вибору: чи співвідносяться між собою використовувані кольори, чи передають елементи інтерфейсу їх призначення і зміст операцій, що подаються, що відчуває людина при використанні тих чи інших елементів управління та наскільки успішно він їх використовує.

Продуктивність, а також надійність, також впливають на перспективу застосування програми. Якщо програма добре виглядає, має просте і зручне управління, але, наприклад, регулярно «підвисає» або, ще гірше, падає з

критичною помилкою при некоректних діях користувача, у нього, ймовірно, буде мало шансів на тривалу експлуатацію. У свою чергу, швидка та стабільна робота програми можуть частково компенсувати його не найкращий дизайн чи відсутність яких-то вторинних функцій.

Створювана програма, в першу чергу, має задовольняти певну функціональність, описану спеціалістом системи, а також бути продуктивною і «безпечною». Безпека мобільних застосунків є одним з найважливіших факторів, що впливають на успіх продукту в довгостроковій перспективі. Як компанії, так і звичайні користувачі потребують безпечного робочого середовища. Саме тому розробники програмного забезпечення та компанії повинні приділяти велику увагу цьому фактору.

Окрім цього, дизайн інтерфейсу має бути user-friendly, тобто, «дружнім для користувача» – орієнтованим на максимальну зручність користувача.

Головна мета user-friendly інтерфейсу полягає в забезпеченні ідеального user experience – щоб після встановлення програми чи використання сайту залишилися лише позитивні емоції.

Дані численних досліджень веб-аналітиків показують, що майже половина аудиторії не дають другий шанс інтерфейсу, який не виправдав їх очікування або виявився надто складним для сприйняття. Якщо на шляху до вирішення завдання доводиться пробиратися через перешкоди, більшість людей намагатимуться знайти альтернативу.

UX (взаємодія з користувачем) і UI (інтерфейс користувача) є пов'язаними, але різними областями в дизайні цифрових продуктів. Обидва елементи мають вирішальне значення для продукту та тісно взаємодіють.

UX стосується взаємодії між користувачем і продуктом або послугою. Таким чином, дизайн користувацького досвіду враховує всі різні елементи, які формують цей досвід.

Розробка користувацького досвіду □ це перш за все спосіб розробки продуктів для людини.

UX-дизайн зосереджений на тому, що відчуває користувач, і наскільки легкі у виконанні бажані завдання. Також спостерігається і проводиться аналіз завдань, щоб побачити, як користувачі фактично виконують завдання.

Кінцева мета UX-дизайну – створити простий, ефективний, релевантний і всебічний приємний досвід для користувача.

У той час як користувальницький досвід є конгломератом завдань, зосереджених на оптимізації продукту для ефективного та приємного використання, дизайн інтерфейсу користувача є його доповненням; вигляд і відчуття, презентація та інтерактивність продукту.

На відміну від UX, UI користувача є суто цифровим терміном.

Інтерфейс користувача □ це точка взаємодії між користувачем і цифровим пристроєм або продуктом, як-от сенсорний екран на смартфоні або сенсорна панель, яку використовується, щоб вибрати, яку каву отримати з кавоварки.

Що стосується вебсайтів і програм, дизайн інтерфейсу користувача враховує зовнішній вигляд, відчуття та інтерактивність продукту, тобто, інтерфейс користувача продукту має бути максимально інтуїтивно зрозумілим, а це означає ретельний розгляд кожного візуального інтерактивного елемента, з яким може зіткнутися користувач.

Як і розробка користувацького досвіду, розробка користувацького інтерфейсу є багатогранною та складною роллю, яка відповідає за перенесення розробки, дослідження, вмісту та макета продукту в привабливий, орієнтуючий і чуйний досвід для користувачів.

Отже, коли йдеться про дизайн продукту, UX та UI доповнюють один одного, і на сучасному конкурентному ринку вкрай необхідно правильно підібрати обидва аспекти.

Після затвердження ідеї та обговорення архітектури мобільного застосунку, створюються мокапи (макети) – низько деталізований прототип, щоб визначити пріоритет та розташування елементів інтерфейсу на екрані,

передбачити для них зручне для доступу місцезнаходження. Для цього було використано програму Figma.

Figma – це кросплатформний графічний онлайн-редактор з можливістю розробки інтерфейсу та прототипування з додатковими офлайн-функціями, доступними в програмах для macOS і Windows. Набір функцій Figma зосереджений на дизайні інтерфейсу користувача та взаємодії з користувачем, з акцентом на співпрацю в реальному часі, використовуючи різноманітні редактори векторної графіки та інструменти для створення прототипів. Мобільна версія Figma для Android та iOS дозволяє переглядати прототипи Figma та взаємодіяти з ними в режимі реального часу на мобільних і планшетних пристроях.

На рис. 3.2–3.4 відображено створений дизайн для застосунку віддаленого керування банківським обладнанням.

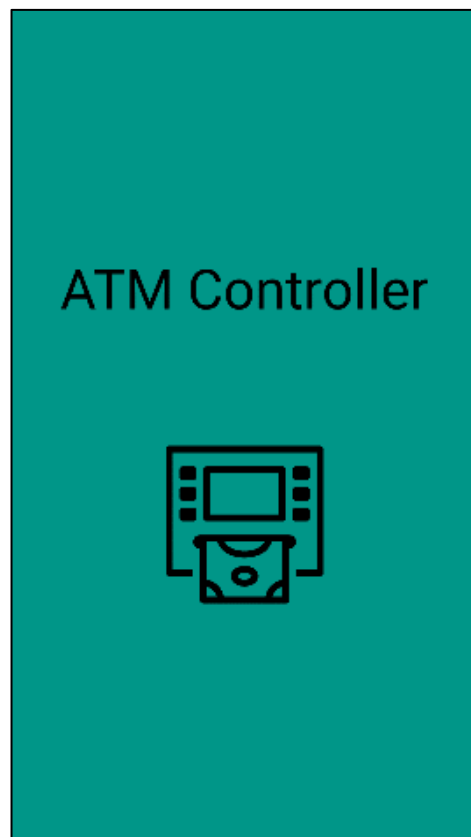


Рисунок 3.2 – Екран завантаження застосунку

Після завантаження застосунку, на головному екрані з'являється перша сторінка. Тут спеціаліст має обрати, з яким банківським обладнанням він буде взаємодіяти. Всі банкомати пронумеровані, а також буде додано адреси для зручності. Якщо комутатор GSM увімкнено та підключено до мобільної мережі, то банкомат з'являється у списку, що вказує на успішне підключення. Тож, користувач може вибрати або один банкомат, або декілька, або всі одразу. Знизу сторінки знаходиться кнопка Next, яка перенаправляє до наступної сторінки.

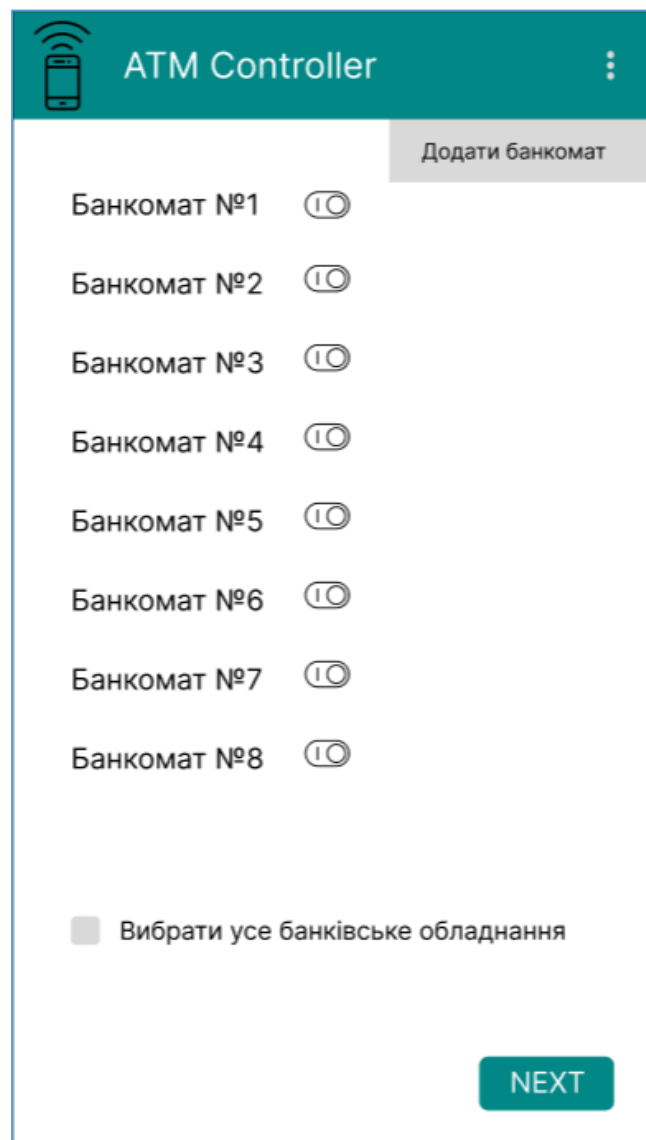
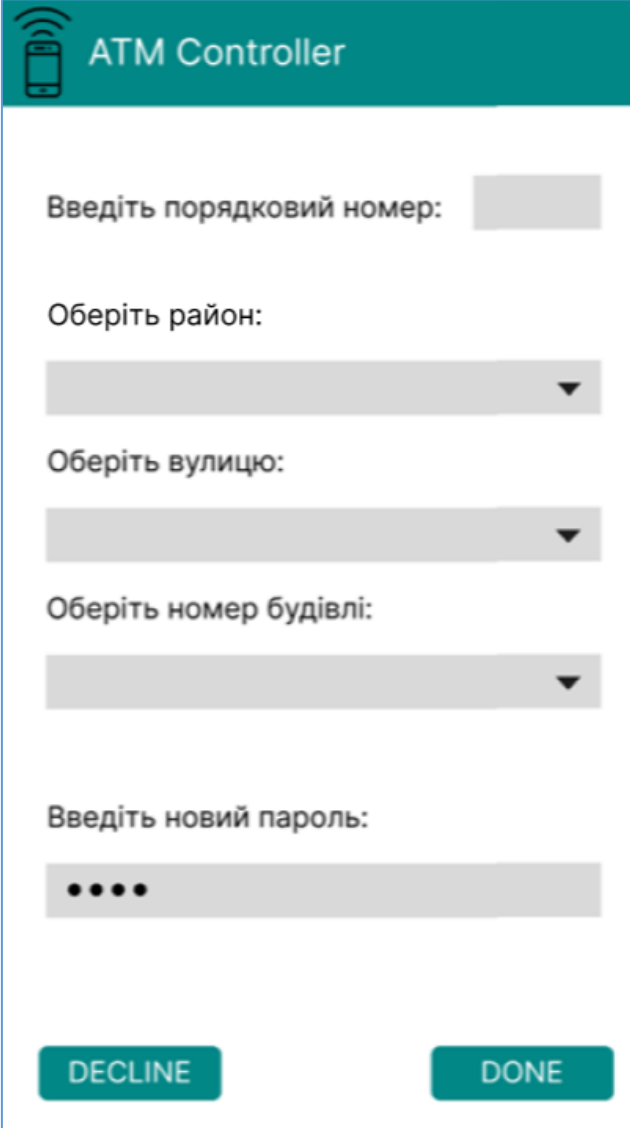


Рисунок 3.3 – Перша сторінка мобільного інтерфейсу

Також на сторінці наявна кнопка додавання нового банкомату (рис. 3.4). Кнопка перенаправляє на сторінку додавання нового обладнання в систему. Для цього обирається порядковий номер банкомату та вказується поточна адреса його розташування. В подальшому це дозволяє спеціалісту не плутатись в системі банкоматів.



The screenshot shows a mobile application interface titled "ATM Controller" with a mobile phone icon and signal waves. The interface contains the following elements:

- A text label "Введіть порядковий номер:" followed by a grey input field.
- A text label "Оберіть район:" followed by a grey dropdown menu with a downward arrow.
- A text label "Оберіть вулицю:" followed by a grey dropdown menu with a downward arrow.
- A text label "Оберіть номер будівлі:" followed by a grey dropdown menu with a downward arrow.
- A text label "Введіть новий пароль:" followed by a grey password input field with four black dots.
- At the bottom, there are two teal buttons: "DECLINE" on the left and "DONE" on the right.

Рисунок 3.4 – Сторінка додавання нового банкомату до системи

Щоб мати можливість використовувати SMS-контроль, має бути встановлений пароль користувача. Комутатор GSM поставляється з заводським паролем «0000», який потрібно змінити на власний пароль, який нелегко вгадати.

Після введення даних, натискається кнопка Done, яка зберігає інформацію у базу даних, або кнопка Decline, яка відмінює поточну дію. При успішному додаванні банкомату до системи, на поверхні вікна програми з'являється спливаюче повідомлення (Toast Notification):

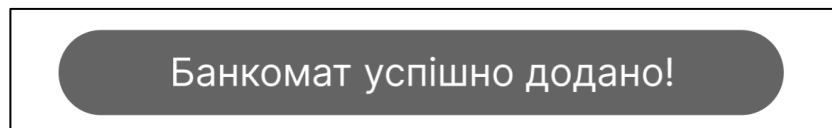


Рисунок 3.5 – Повідомлення про успішне оновлення системи

При цьому, поточна діяльність програми залишається працездатною для користувача. Протягом кількох секунд повідомлення плавно закривається.

На рис. 3.6 відображено наступну сторінку – вибір дії, яка буде виконана з банківським обладнанням. З випадаючого списку (Spinner) спеціаліст обирає ту дію, яку потрібно, з 8 наявних. Він може ввімкнути\вимкнути навантаження розетки, яка підключена до банківського обладнання, дізнатися стан розетки, редагувати телефонну книгу SIM-карти, яка вставлена в розетку, або перевірити баланс на ній.

Після вибору дії користувач має дві кнопки на вибір: або повернутися на сторінку вибору банкомату, натиснувши кнопку Previous, або завершити операцію, натиснувши кнопку Done. Після цього обрана дія буде виконана, про що застосунок повідомить спеціаліста.

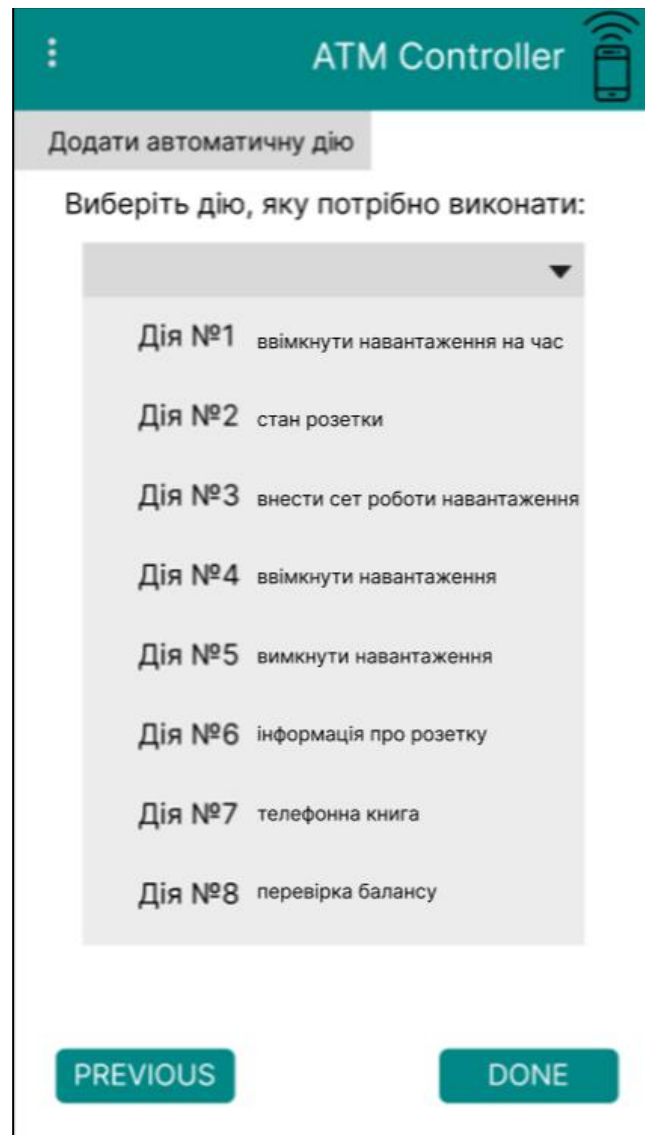


Рисунок 3.6 – Друга сторінка

Також, за аналогією з першою сторінкою, є кнопка додавання автоматичної дії.

Автоматична дія – дія, яка при правильному налаштуванні має бути виконана без прямого втручання спеціаліста у певний час та дату. На рис. 3.7 відображено власне додавання дії.

ATM Controller

Виберіть банкомат(-и):

Банкомат №1 Банкомат №5

Банкомат №2 Банкомат №6

Банкомат №3 Банкомат №7

Банкомат №4 Банкомат №8

Вибрати усе обладнання

Виберіть дію, яку потрібно виконати:

Виберіть час та дату:

Дата Час

Повторювання:

DECLINE DONE

Рисунок 3.7 – Сторінка додаткових налаштувань

Користувач обирає потрібне банківське обладнання за номером від 1 до 8 або все одразу. Після вибору певної дії, залишається вибір певної дати та часу, в який дія буде виконуватися. Також доступне повторювання: один раз, щоденно, у будні дні, або, можна обрати певні дні.

3.4 Розробка

Розробка Android-застосунку для дистанційного керування банківським обладнанням через GSM-розетку є складним завданням, яке потребує досвіду як у розробці програмного забезпечення, так і в банківських системах.

Нижче наведено частину коду програми для Android, яка використовує SMS-повідомлення для дистанційного керування розеткою GSM:

1) спочатку додається дозвіл на надсилання SMS-повідомлень у файл AndroidManifest.xml:

```
<uses-permission android:name="android.permission.SEND_SMS"/>
```

2) створюється новий клас для обробки SMS-повідомлень:

```
public class SmsSender {
    private static final String SMS_SENT_ACTION = "com.example.app.SMS_SENT";
    private static final String SMS_DELIVERED_ACTION =
"com.example.app.SMS_DELIVERED";

    public static void sendSms(Context context, String phoneNumber, String message) {
        PendingIntent sentPendingIntent = PendingIntent.getBroadcast(context, 0, new
Intent(SMS_SENT_ACTION), 0);
        PendingIntent deliveredPendingIntent = PendingIntent.getBroadcast(context, 0,
new Intent(SMS_DELIVERED_ACTION), 0);

        SmsManager smsManager = SmsManager.getDefault();
        smsManager.sendTextMessage(phoneNumber, null, message, sentPendingIntent,
deliveredPendingIntent);
    }
}
```

3) у основній дії використовується клас SmsSender для надсилання SMS-повідомлень до сокета:

```
public class MainActivity extends AppCompatActivity {
    private static final String SOCKET_NUMBER = "+1234567890"; // Replace with the
phone number of the socket

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        Button onButton = findViewById(R.id.onButton);
        onButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                SmsSender.sendSms(MainActivity.this, SOCKET_NUMBER, "1");
            }
        });

        Button offButton = findViewById(R.id.offButton);
        offButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                SmsSender.sendSms(MainActivity.this, SOCKET_NUMBER, "0");
            }
        });
    }
}
```

```
    });  
  }  
}
```

4) додаються приймачі трансляції для намірів SMS_SENT_ACTION і SMS_DELIVERED_ACTION для обробки результату SMS-повідомлення – відповідний код наведено в додатку А.1.

5) в кінці реєструються приймачі трансляції в MainActivity:

```
public class MainActivity extends AppCompatActivity {  
    private static final String SOCKET_NUMBER = "+1234567890"; // Replace with the  
    phone number of the socket  
    private SmsSentReceiver smsSentReceiver = new SmsSentReceiver();  
    private SmsDeliveredReceiver smsDeliveredReceiver = new SmsDeliveredReceiver();
```

Висновки до розділу 3

У третьому розділі було розглянуто життєвий цикл розробки програмного забезпечення, який являє собою серію етапів, що описують процес створення застосунку – SDLC. Метою є забезпечення структурованої та ефективної розробки програмного забезпечення, задоволення вимог замовника та забезпечення високоякісного програмного забезпечення.

На етапі дизайну розглянуто його види, а саме UX та UI, та зроблено висновок, що у розроблюваному застосунку в пріоритеті UX; розглянуто мокапи сторінок з детальним роз'ясненням. На етапі розробки продемонстровано один з фрагментів програми, а саме наведено частину коду програми, яка використовує SMS-повідомлення для дистанційного керування розеткою GSM.

4 ТЕСТУВАННЯ СИСТЕМИ ВІДДАЛЕНОГО КЕРУВАННЯ

4.1 Тестування

Тестування Android-застосунку є обов'язковою частиною процесу розробки. Ось деякі кроки тестування, які мають бути виконані, щоб переконатися, що програма працює стабільно та безпечно:

- модульне тестування передбачає тестування окремих компонентів програми, таких як класи або методи, щоб переконатися, що вони функціонують належним чином. Це може допомогти виявити помилки на ранніх стадіях процесу розробки, що спростить їх виправлення;

- інтеграційне тестування передбачає перевірку того, як різні компоненти програми працюють разом. Це може допомогти виявити будь-які проблеми, які виникають під час інтеграції програми з банківським обладнанням або розеткою GSM;

- тестування прийнятності користувачами передбачає тестування реальними користувачами програми, щоб переконатися, що вона відповідає їхнім потребам і працює правильно. Це може надати цінний відгук і допомогти виявити будь-які проблеми, які могли не бути виявлені на попередніх етапах тестування;

- тестування безпеки передбачає перевірку програми на потенційні вразливості, такі як витік даних або несанкціонований доступ до конфіденційної інформації. Це може допомогти забезпечити безпеку програми та захист даних користувача;

- тестування продуктивності передбачає тестування продуктивності програми за різних умов, наприклад інтенсивний трафік або погане мережеве з'єднання. Це може допомогти виявити будь-які вузькі місця або області, де застосунок може потребувати оптимізації;

– регресійне тестування передбачає повторне тестування програми після внесення змін або виправлень, щоб переконатися, що зміни не викликають нових проблем або помилок.

Важливо провести ретельне тестування, щоб переконатися, що програма функціональна, стабільна та безпечна. Це може допомогти переконатися, що користувачі отримують позитивний досвід роботи з програмою та що вона відповідає їхнім потребам.

Нижче приведено частину коду для збереження інформації у базі даних, яка ідеально підходить для повторюваних або структурованих даних, таких як контактна інформація.

API, необхідні для використання бази даних на Android, доступні в пакеті `android.database.sqlite`.

Додавання нового банкомату до бази даних створеного застосунку вимагає інтеграції застосунку з базою та написання коду для додавання нових записів до бази.

```
public class ATM {
    private int id;
    private String district;
    private String street;
    private String number;
    private String password;

    public ATM(int id, String district, String street, String number, String
password) {
        this.name = name;
        this.location = location;
        this.ip = ip;
        this.port = port;
    }

    // getters and setters for each property
}
```

Далі потрібно налаштувати базу даних і створити таблицю для зберігання даних банкомату. Відповідний код наведено у додатку А.2.

```
public void addATM(ATM atm) {
    SQLiteDatabase db = this.getWritableDatabase();

    ContentValues values = new ContentValues();
    values.put(ATM_COLUMN_NAME, atm.getName());
    values.put(ATM_COLUMN_LOCATION, atm.getLocation());
    values.put(ATM_COLUMN_IP, atm.getIp());
    values.put(ATM_COLUMN_PORT, atm.getPort());

    db.insert(ATM_TABLE_NAME, null, values);
    db.close();
}
```

Після налаштування класу банкомату та бази даних написано код для додавання нового банкомату до бази даних.

Наведений вище код створює новий об'єкт `ContentValues` і додає до нього властивості банкомату. Потім він вставляє значення в таблицю `ATM` у базі даних.

За допомогою цього коду додається новий банкомат в базу даних застосунку для системи дистанційного керування підключенням банківського обладнання через модулі `GSM`.

При використанні коду, наведеного у додатку `A.2`, для додавання нового банкомату, програма викличе метод `addATM` і передасть об'єкт банкомату з відповідною інформацією про новий банкомат, такою як його назва, місцезнаходження, `IP`-адреса та номер порту.

Коли викликається метод `addATM`, він вставляє новий рядок у таблицю `atm_table` у базі даних програми зі значеннями для нового банкомату.

Ці значення зберігатимуться в стовпцях рядка, які відповідають властивостям банкомату, таким як ім'я, місцезнаходження, `IP`-адреса та порт.

Щойно новий банкомат буде додано до бази даних, програма може отримати його та використовувати за потреби. Наприклад, програма може відобразити список усіх банкоматів у базі даних, або дозволити користувачеві вибрати певний банкомат для підключення. Результат зображено на рис. 4.1.

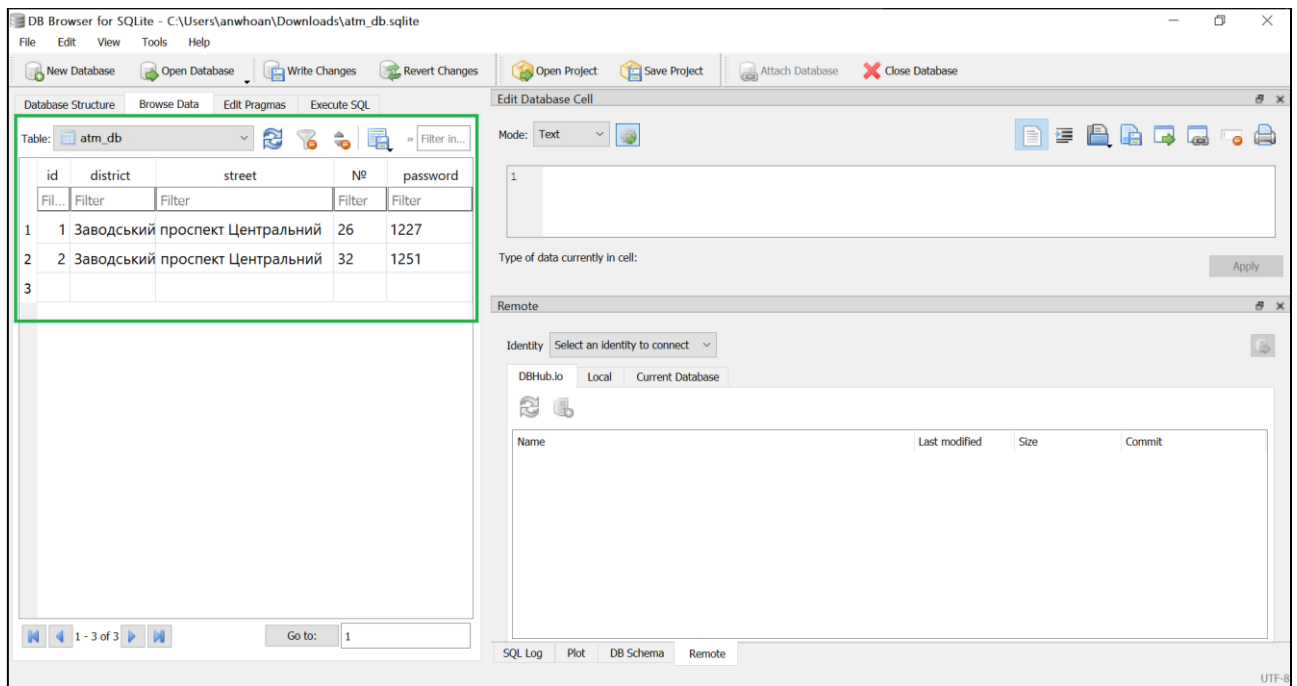


Рисунок 4.1 – Процес додавання банкомату до бази даних SQLite

На рисунку 4.1 зображено процес додавання банкоматів до бази даних. У виділеному фрагменті відображається створена таблиця `atm_table`, в якій вже наявні 2 банкомати. Таблиця має 4 поля – `id`, `district` (район), `street` (вулиця, на якій знаходиться підключене банківське обладнання), `№` (номер будинку), `password` (пароль).

На рис. 4.2 нижче показано власне результат додавання нових банкоматів до бази даних. Застосунок розгорнуто на емуляторі Android, який імітує пристрої Android на комп'ютері, для перевірки програми на різних пристроях і рівнях Android API без необхідності мати кожен фізичний пристрій. Таблиця з даними виводиться на першій сторінці застосунку, де користувач обирає, з яким обладнанням буде взаємодіяти надалі.



Рисунок 4.2 – Відображення результату додавання банкомату до бази даних SQLite

4.2 Розгортання

Розгортання застосунку включає кілька кроків, які забезпечують можливість встановлення та використання застосунку кінцевими користувачами. Ось основні кроки для розгортання програми Android:

- підготовка програми до розповсюдження передбачає створення випускної версії програми, яку можна встановити на пристрої користувачів, а

також створення набору приміток до випуску, які описують функції програми та будь-які зміни чи виправлення помилок;

- вибір каналу розповсюдження: через магазин Google Play, сторонні магазини застосунків, пряме розповсюдження електронною поштою чи обмін файлами. Канал розповсюдження має підходити для програми та цільової аудиторії;

- надсилання програми на перевірку. Якщо програма розповсюджується через сторонній магазин програм або магазин Google Play, потрібно буде надіслати програму на перевірку. Це передбачає надання інформації про програму та її функції, та перевірку програми на відповідність правилам і політикам магазину;

- опублікування програми зробить застосунок доступним для користувачів, які зможуть завантажити та встановити його на своїх пристроях;

- відстеження й оновлення програми відбувається після розгортання програми та включає в себе стеження за його використанням і відгуками користувачів, а також оновлення програми за потреби для усунення помилок або додавання нових функцій.

Оскільки застосунок на даному етапі створений і тестується як локальний інструмент і має конфіденційні дані, то завантаження його в магазин Google Play не є доцільним. Застосунок поширюється електронною поштою. За запитом спеціаліста застосунок перевіряється на наявність помилок або задля додавання нових необхідних для взаємодії з системою банківського обладнання функцій.

4.3 Технічне обслуговування

Технічне обслуговування має вирішальне значення для того, щоб застосунок продовжував коректно функціонувати та відповідати потребам своїх користувачів.

Відстеження того, як користувачі взаємодіють із програмою, важливо для виявлення будь-яких проблем або помилок, які можуть виникнути. Інструменти аналітики можуть бути використані, щоб відстежувати поведінку користувачів, визначати області, які потребують покращення, і визначати, як оптимізувати програму для кращої продуктивності.

Важливо регулярно оновлювати програму, щоб підтримувати її в актуальному стані з найновішими функціями, виправленнями безпеки та покращеннями продуктивності. Це також допомагає забезпечити сумісність програми з новими версіями операційної системи Android. Потрібно тестувати програму, щоб переконатися, що вона продовжує працювати належним чином і не містить помилок, використовуючи інструменти автоматизованого тестування, щоб спростити процес тестування та завчасно виявити проблеми.

Оскільки програма має справу з конфіденційною банківською інформацією, надзвичайно важливо стежити за безпекою та гарантувати захист від потенційних загроз. Це включає впровадження найновіших заходів безпеки, таких як шифрування, двофакторна автентифікація та безпечне кодування.

Надання своєчасної та ефективної підтримки клієнтів важливо для того, щоб користувачі могли отримати допомогу, коли вона їм потрібна. Підтримка може бути надана через різні канали, такі як електронна пошта, телефон або обмін повідомленнями в програмі.

Підсумовуючи, супровід застосунку передбачає моніторинг використання, реагування на відгуки користувачів, оновлення застосунку, тестування застосунку, моніторинг безпеки застосунку та надання клієнтської підтримки. Виконуючи ці дії, програма залишається надійною, безпечною та зручною для користувачів, і продовжує задовольняти потреби своїх користувачів.

Висновки до розділу 4

У даному розділі описано останні три етапи розробки мобільного застосунку для віддаленого вимикання/вмикання банкоматів через GSM-модулі: тестування, розгортання та технічне обслуговування. У відповідному розділі протестовано додавання нового обладнання до системи. В подальшому буде проводитися тестування застосунку напряду з банкоматами з допомогою спеціаліста.

Також розглянуто етапи розгортання мобільних застосунків. Оскільки створене програмне рішення націлене на керування локальним банківським обладнанням, то розповсюдження у спеціалізованих магазинах застосунків або на тематичних сайтах не є доцільним.

В якості технічного обслуговування відбувається оновлення застосунку за запитом користувачів, моніторинг безпеки даних застосунку та надання клієнтської підтримки.

ВИСНОВКИ

Створювана система дозволяє співробітникам або технічному спеціалісту дистанційно керувати банківським обладнанням (в даному випадку банкоматами) та контролювати його за допомогою модулів GSM (Global System for Mobile Communications).

Для віддаленого відключення в якості керуючого елемента використовується GSM-розетка – невеликий електроприлад зі вбудованою електронною платою і GSM-модулем для дистанційного керування через дзвінок або SMS-повідомлення.

Актуальність роботи, що виконується, підкреслюється важливістю збереження світломаскування в містах, а також заощадження електроенергії.

Згідно з поставленою метою у даній роботі створено застосунок, який забезпечує зручний спосіб керування банківськими пристроями на відстані, та керування налаштуваннями та конфігураціями, використовуючи вбудовані або під'єднані до них GSM-модулі.

У даній роботі було досліджено систему віддаленого керування підключенням банківського обладнання через GSM-модулі.

Зазначену мету досягнуто завдяки виконання наступних завдань:

- виконано огляд існуючих систем віддаленого керування та програмних застосунків;
- здійснено моделювання системи, що розробляється, та вибір GSM-розетки;
- створено програмне забезпечення для полегшення зв'язку між розеткою та телефоном, яке також дозволяє дистанційно керувати обладнанням через мережу GSM;
- протестовано створений мобільний застосунок.

В ході магістерської кваліфікаційної роботи було розглянуто та проведено аналіз створених систем віддаленого керування та їх призначення

та використання в різних галузях економіки: в охоронній сфері, промисловості та аграрному бізнесі. Проаналізовано особливості віддаленого керування розгалуженою мережею банківського обладнання.

Для даної системи була обрана GSM-розетка SC1-GSMV від KONLEN. Розетка має можливість дистанційного керування через дзвінки та SMS-повідомлення, але це не завжди є зручним, оскільки спеціаліст, який обслуговує систему має витратити багато часу для написання довгого коду в повідомленні або прослуховування всіх команд під час дозвону. Також це не є зручним, оскільки рівень шуму в зовнішньому середовищі може завадити введенню команд «голосом». Крім того, дана процедура ускладнюється кількістю банківського обладнання в системі, яке представляє собою не єдиний банкомат, а декілька – в даному випадку 8.

У розділі 3 відбувається розробка програмного вирішення. Для даної системи доцільним є створення застосунку на смартфон на базі операційної системи Android. Для розробки було використано середовище Android Studio.

Останнім завданням кваліфікаційної роботи було тестування застосунку, а також визначення практичного призначення даної системи. Зокрема, запропоновано ввести використання системи спеціалістами для керування банківського обладнання в місті.

Практична значимість розробленої системи полягає у тому, що вона дозволяє дистанційно керувати GSM-розеткою через застосунок, оскільки існуюче управління через SMS-повідомлення чи дзвінок є не завжди зручним, так як зовнішній шум може завадити «вводу» команди.

Апробація результатів кваліфікаційної магістерської роботи відбулася під час Всеукраїнської науково-практичної конференції молодих вчених, аспірантів і студентів «Інформаційні технології та інженерія»

За результатами роботи видано публікацію у збірнику тез доповідей [5].

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

- 1.Світломаскування: як працює та які регіони його дотримуються. URL: <https://suspilne.media/243237-svitlomaskuvanna-ak-pracue-ta-aki-regioni-jogo-dotrimuutsa/> (дата звернення: 17.01.2023).
- 2.Череп О. Г., Хмельковська Т. В. Оцінка розвитку функціонування дистанційного банківського обслуговування в Україні. Вісник Хмельницького національного університету. 2021. № 5, Том 2. С. 162–166. DOI: 10.31891/2307-5740-2021-298-5(2)-26.
- 3.Kukushkin A. Introduction to mobile network engineering: GSM, 3G-WCDMA, LTE and the Road to 5G. Wiley, 2018. 377 p.
- 4.Рівні гучності різних джерел шуму. URL: <https://www.acoustic.ua/directory/133> (дата звернення: 17.01.2023).
- 5.Винар А. А., Журавська І. М. Система віддаленого керування підключенням банківського обладнання через GSM-модулі. Інформаційні технології та інженерія : тези доп. Всеукр. наук.-практ. конф. молодих вчених, аспірантів і студентів. Миколаїв, 7–10 лютого 2023 р. Миколаїв : Чорном. нац. ун-т ім. Петра Могили, 2023. С. 61–63.
- 6.Seite für Elektriker und über Elektrizität. URL:<https://my-electricianexp.com> (дата звернення: 19.01.2023).
- 7.Remote Control Via GPRS/GSM SMS. URL: <https://www.instructables.com/Control-the-relays-via-GPRS-SMS/> (дата звернення: 19.01.2023).
- 8.GSM розетка дистанційне керування через мобільний телефон за допомогою дзвінка SMS-команд (KONLEN, SC1-GSM). URL: <https://rozetka.com.ua/135052910/p135052910/> (дата звернення: 04.01.2023).
- 9.Шапінський В. Електробезпека під час обслуговування та ремонту банкоматів. *Довідник спеціаліста з охорони праці*. 2021. № 1. URL: <https://esop.mcfp.ua/866112> (дата звернення: 04.01.2023).

10. Qubies Mobile System: An E-queuing Application. Project LIM YIT YEE (TP038344). Asia Pacific University, 2019. 332 p.
11. Giambene G. Queuing theory and telecommunications: Networks and applications. Springer International Publishing, 2022. 413 p.
12. Dattatreya G. R. Performance analysis of queuing and computer networks. Taylor & Francis Group, 2019. 472 p.
13. Rizk A. Adaptive queuing models for communication systems: queuing models for parallel systems and multimedia streaming. Technische Universität Darmstadt, 2019.
14. Ragsdale C. Spreadsheet modeling and decision analysis: A practical introduction to business analytics. Cengage Learning, 2021. 864 p.
15. Mpetla K., Selaotswe O. R. Application of queuing theory to analyse an ATM queuing system. *In book: Intelligent Computing & Optimization*. January 2022. DOI: 10.1007/978-3-030-93247-3_84.
16. Burodo M. S., Suleiman S., Shaba Ya. S. Queuing theory and ATM service optimization: Empirical evidence from First Bank Plc, Kaura Namoda Branch, Zamfara State. *American Journal of Operations Management and Information Systems* 2019. Vol. 4, Is. 3. P.80. DOI: 10.11648/j.ajomis.20190403.12.
17. Pradana J. A. Utility 1 server on queue service (Study: Bank account number conversion). November 2021. DOI: 10.20473/ajim.v2i2.30232.
18. Юрчук Н. П. Перспективи розвитку інформаційних технологій в банківській системі. *Ефективна економіка*. 2015. № 9. URL: <http://www.economy.nayka.com.ua/?op=1&z=4305>.
19. Інформаційні системи та технології в банківській сфері : метод. рек. для самот. роботи студентів / уклад. К. М. Азізова. Харків : ХНЕУ ім. С. Кузнеця, 2021. 42 с.
20. Блащук-Дев'яткіна Н., Петришин Х. Дистанційне банківське обслуговування: Теоретичні аспекти, сучасний стан та перспективи його

розвитку. *Молодий вчений*. 2022. № 9 (109). С. 122–128. DOI: 10.32839/2304-5809/2022-9-109-27.

21. ATMeye.iQ □ ATM Monitoring and Management System. URL: <https://atmeye.com/atm-monitoring/> (Last accessed: 19.01.2023).

22. What is the software development life cycle (SDLC) and why is it important? URL: <https://kruschecompany.com/what-is-a-software-development-life-cycle/#Design> (дата звернення: 05.02.2023).

23. Yablonski J. *Laws of UX: Using psychology to design better products & Services* paperback. 2020.

24. Kuang C., Fabricant R. *User friendly: how the hidden rules of design are changing the way we live, work, and play* hardcover. 2019.

ДОДАТОК А

Коди програм

A1 Код програми для обробки результату SMS-повідомлення

```
public class SmsSentReceiver extends BroadcastReceiver {
    @Override
    public void onReceive(Context context, Intent intent) {
        switch (getResultCode()) {
            case Activity.RESULT_OK:
                Toast.makeText(context, "SMS sent", Toast.LENGTH_SHORT).show();
                break;
            case SmsManager.RESULT_ERROR_GENERIC_FAILURE:
                Toast.makeText(context, "Generic failure",
Toast.LENGTH_SHORT).show();
                break;
            case SmsManager.RESULT_ERROR_NO_SERVICE:
                Toast.makeText(context, "No service", Toast.LENGTH_SHORT).show();
                break;
            case SmsManager.RESULT_ERROR_NULL_PDU:
                Toast.makeText(context, "Null PDU", Toast.LENGTH_SHORT).show();
                break;
            case SmsManager.RESULT_ERROR_RADIO_OFF:
                Toast.makeText(context, "Radio off", Toast.LENGTH_SHORT).show();
                break;
        }
    }
}

public class SmsDeliveredReceiver extends BroadcastReceiver {
    @Override
    public void onReceive(Context context, Intent intent) {
        switch (getResultCode()) {
            case Activity.RESULT_OK:
                Toast.makeText(context, "SMS delivered", Toast.LENGTH_SHORT).show();
                break;
            case Activity.RESULT_CANCELED:
                Toast.makeText(context, "SMS not delivered",
Toast.LENGTH_SHORT).show();
                break;
        }
    }
}
```

A2 Код програми для налаштування бази даних

```
public class DatabaseHelper extends SQLiteOpenHelper {
    private static final String DATABASE_NAME = "atm_db";
    private static final int DATABASE_VERSION = 1;
    private static final String ATM_TABLE_NAME = "atm_table";
    private static final String ATM_COLUMN_ID = "id";
    private static final String ATM_COLUMN_LOCATION = "location";
    private static final String ATM_COLUMN_IP = "ip";
    private static final String ATM_COLUMN_PORT = "port";

    public DatabaseHelper(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        String CREATE_ATM_TABLE = "CREATE TABLE " + ATM_TABLE_ID + " ("
            + ATM_COLUMN_ID + " TEXT,"
            + ATM_COLUMN_LOCATION + " TEXT,"
            + ATM_COLUMN_IP + " TEXT,"
            + ATM_COLUMN_PORT + " INTEGER)";
        db.execSQL(CREATE_ATM_TABLE);
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        // not implemented in this example
    }
}
```

ДОДАТОК Б

Публікації за темою роботи

ЗМІСТ	
Інформаційні системи та їх інтелектуалізація	
<i>Брадінич О. В., Степанчук Л. К.</i> Створення програмного застосунку для введення СЛІАР до діагностично домінуючого вигляду.....	8
<i>Бухатірова О. О., Обрухова К. О.</i> Моніторинг стабільності інтернет-конференції в умовах багатозадачності.....	10
<i>Гонтаренко С. С., Крайчик Я. М.</i> Процес розгортання кластерного застосунку в Kubernetes.....	12
<i>Ершардт Е. В., Козлов О. В.</i> Система первинної медичної діагностики на основі нечіткої логіки.....	14
<i>Жисляк І. О., Калідина І. О.</i> Інтелектуальні компоненти системи-емулятора для комп'ютерної гри в гольф.....	16
<i>Ковалюк М. О., Павлова О. О.</i> Інформаційна система для відтворення тривимірних об'єктів у доповненій реальності.....	18
<i>Коренікова Г. В., Зуй О. М., Стукалов С. А.</i> Моделювання квантового алгоритму Гровера.....	21
<i>Кравчук С. С., Павлова О. О.</i> Інформаційна система для інклюзивного доступу до громадських місць.....	22
<i>Мельниченко О. В.</i> Архітектура автоматизованої системи розпізнавання структурних об'єктів однієї природи в тривимірному просторі.....	24
<i>Освітницька А. В., Болобаш Н. М.</i> Розпізнавання рукописних цифр на основі згорткових нейронних мереж.....	26
<i>Швайко В. К., Фесік З. Ю.</i> Інформаційна система для вибору виду спорту на основі аналізу морфофункціональних показників людини.....	28
Машинне навчання та штучний інтелект	
<i>Башута А. Р.</i> Концепція інтерфейсу користувача на основі доповненої реальності.....	30
<i>Балик М. В.</i> Розробка бота гри «Змійка» з використанням нейронних мереж та навчання з підкріпленням.....	32

Міністерство освіти і науки України
Чорноморський національний університет
імені Петра Могили



«Інформаційні технології та інженерія»

*Всукраїнська науково-практична конференція
молодих вчених, аспірантів і студентів*

ТЕЗИ

7–10 лютого 2023 року

Миколай – 2023

Бурдин К. Ю., Лисенко Е. А. Телеграм-бот для розпізнавання українського тексту за допомогою нейронних мереж..... 34

Єрмолаєв О. А., Кулаковська І. В. Аналіз тональності коментарів з використанням машинного навчання 37

Жебко О. О. Інтелектуальна система прогнозування фінансових ризиків організації на основі ансамблевих методів 39

Ковалів О. П., Свіденко Є. В. Дослідження нейронних мереж для розпізнавання військової техніки на супутникових знімках 42

Кузьмін А. А., Павлова О. О. Застосування комп'ютерного зору для кіберфізичної системи розумної парковки..... 45

Ніценорук А. А., Данчук С. В., Коротких Ю. В., Ніценорук А. О. Метод виявлення шкідливого програмного забезпечення в операційній системі Android 47

Проворний О. В., Козлов О. В. Синтез та дослідження нейромережевої системи для розпізнавання облич співробітників підприємства 49

Пронін В. А., Калініна І. О. Інтелектуальна система прогнозування медичних витратків для страхового забезпечення на основі методів машинного навчання 53

Рудь В. О., Лисенко Е. А. Музичний сервіс із нейромережевою системою рекомендацій та кросплатформним клієнтом 54

Смаков Є. А., Козлов О. В. Інтелектуальна система для розширення мережі авіаційних маршрутів 57

Методи і засоби комп'ютерної інженерії

Авсієвчук В., Давидова Р. Security of smart parking subsystem 59

Винар А. А., Журавська І. М. Система віддаленого керування підключенням банківського обладнання через GSM-модулі **61**

Войтасюк А. М. Концепція створення засобів морської робототехніки в інтересах ВМС ЗС України 63

Гончаров Д. С. Стиснення інформації у системах моніторингу стану здоров'я: методи сімейства LZ 65

Данилова О. М., Горішина О. М., Бураченко І. С. Апаратно-програмний комплекс навігації безпілотного наземного транспорту в логістичній 67

Figure 2 - Smart Parking Software Architecture using Security Middleware

REFERENCES

1. Pavlova O., Korvalenko V., Novorushchenko T., Avsiyevchuk V. Neural network based image recognition method for smart parking. Comput. Syst. Inf. Technol. 3, 2021. pp. 49–55
2. Авсієвчук В., Кузьмін А. Дослідження вразливостей системи розумної парковки та способи їх усунення. Актуальні Проблеми Комп'ютерних Наук (АПКН-2022), Хмельницький, Україна, 18-19 листопада 2022. Хмельницький: ХНУ, 2022. С. 11-14.

УДК 004.031.43:336.74

Винар А. А., Журавська І. М.
Черноморський національний університет ім. Петра Могили,
Миколаїв, Україна

СИСТЕМА ВІДДАЛЕНОГО КЕРУВАННЯ ПІДКЛЮЧЕННЯМ БАНКІВСЬКОГО ОБЛАДНАННЯ ЧЕРЕЗ GSM-МОДУЛІ

З перших днів повномасштабного вторгнення росії в Україну в багатьох регіонах задіяли перешкоджанню орієнтування авіації та безпілотників противника, а також роботи диверсійно-розвідувальних сил, був введений режим світломаскування. В зв'язку з цим, ввечері та вночі освітлення має бути вимкнуте – в будинках, в значущих об'єктах і на вулицях [1].

Восени українська енергетична система серйозно постраждала від російських терактів. Балакут в Україні змінив підхід до використання електроенергії, змусивши задуматися людей про свідоме використання енергії та її економію. Крім того, після відновлення електропостачання через планові відключення електроенергії, оператор має диспетчерувати, за допомогою програмних засобів, підключити банківське обладнання до електромережі [2].

<p>Однією з проблем в цьому випадку є банкомати, які працюють цілодобово. Кількість банківського обладнання у місті унеможливило їх відключення вручну, тому метою дослідження було створення застосунок, за допомогою якого спеціаліст має можливість керувати роботою банківського обладнання, використовуючи існуючі або під'єднані до них GSM-модулі.</p> <p>Для віддаленого відключення в якості керуючого елемента використовується GSM-роутер – невеликий електроприлад зі вбудованою електронною платою і GSM-модулем, який має слот для SIM-карти, вхідну вишу й електричний роз'єм для підключення іншої апаратури [3].</p> <p>GSM-роутер має можливість дистанційного керування через дзвінок або SMS-повідомлення, але це не завжди є зручним, оскільки зовнішній шум може завадити вводу команд [4].</p> <p>Актуальність роботи, що виконується, підкреслюється важливістю збереження сигналмаскування в містах, а також заощадження електроенергії.</p> <p>Як вихідні дані для роботи було використано GSM-роутеру та мобільний телефон. В результаті розроблено бізнес-процес (рис. 1) та створено мобільний застосунок.</p>	<p>рпаче-та-аі-тегіопі-іего-догітімітсі/ (дата звернення: 17.01.2023).</p> <p>2. Череп О. Г., Хмельковська Т. В. Оцінка розвитку функціонування дистанційного банківського обслуговування в Україні. Вісник Хмельницького національного університету. 2021. № 5, Том 2. С. 162–166. DOI: 10.31891/2307-5740-2021-298-5(2)-26</p> <p>3. Kishishkin A. Introduction to mobile network engineering. GSM, 3G-WCDMA, LTE and the Road to 5G. Wiley, 2018. 377 p. URL: https://www.acoustic.ch/directory/133 (дата звернення: 17.01.2023).</p> <p>УДК 681.5</p> <p><i>Войтасик А. М.</i> <i>Черноморський національний університет ім. Петра Могили,</i> <i>Миколаїв, Україна</i></p> <p>КОНЦЕПЦІЯ СТВОРЕННЯ ЗАСОБІВ МОРСЬКОЇ РОБОТОТЕХНІКИ В ІНТЕРЕСАХ ВМС ЗС УКРАЇНИ</p> <p>Розвиток останніх подій в Україні вносить на порядок денний актуальне прикладне науково-технічне завдання розробки та побудови спеціальних засобів морської робототехніки (ЗМР) для прихованого переміщення, транспортування і розміщення на морському дні корисних приладів і обладнання в рамках інтересів Військово-Морських Сил Збройних Сил (ВМС ЗС) України [1, 2]. Серед основних тактико-технічних характеристик ЗМР можна відзначити:</p> <ul style="list-style-type: none"> – глибини занурення до 100 м; – автономність до 16 год; – вантажопідйомність до 80 кг; – можливість ручного і автоматичного керування; – дальність ходу до 120 км. <p>Згідно з цим основною метою є розробка концепції створення спеціалізованих ЗМР, випробування та дослідної експлуатації ВМС ЗС України.</p> <p>Процес створення спеціалізованого ЗМР в інтересах ВМС ЗС України можна реалізувати на базі підводних апаратів-роботів транспортного типу (ПАР-Т). Для цього варто виконати декілька етапів:</p> <ul style="list-style-type: none"> – розробити підводну технологію прихованого переміщення, транспортування і розміщення на морському дні корисних вантажів за допомогою ЗМР;
<p>Рисуюнок 1 – Схема дистанційного підключення банкомату</p> <p>Робота системи починається зі авторизації оператора в системі за допомогою логіна/пароллю або відбиття пальця чи зразка голосу користувача. Введення персональних параметрів доступу до банківської системи перевіряється на основі даних, що зберігаються в базі даних, і буде перевірено.</p> <p>ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ</p> <p>1. Світломаскування: як працює та як регіони його дотримуються. URL: https://svetrlina.media/243237-звітломаскування-аб-</p>	