

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ПОЛІСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет інформаційних технологій, обліку та фінансів

Кафедра комп'ютерних технологій

і моделювання систем

Кваліфікаційна робота

На правах рукопису

Дубина Вадим Петрович

УДК 004.05:658.8

**КВАЛІФІКАЦІЙНА РОБОТА**

Інформаційна система підтримки роботи логістичної компанії

126 «Інформаційні системи та технології»

Подається на здобуття освітнього ступеня бакалавр

Кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

---

(підпис, ініціали та прізвище здобувача вищої освіти)

Керівник роботи

Маєвський Олександр Володимирович

д.т.н., доцент.

**Висновок кафедри** \_\_\_\_\_

за результатами попереднього захисту: \_\_\_\_\_

Протокол засідання кафедри \_\_\_\_\_

№ \_\_-\_\_ від «\_\_\_\_» \_\_\_\_\_ 20 \_\_\_\_ р.

Завідувач кафедри \_\_\_\_\_

\_\_\_\_\_

(науковий ступінь, вчене звання)

\_\_\_\_\_

(підпис)

\_\_\_\_\_

(прізвище, ім'я, по батьковій)

«\_\_\_\_» \_\_\_\_\_ 20 \_\_\_\_ р.

### **Результати захисту кваліфікаційної роботи**

Здобувач вищої освіти \_\_\_\_\_ захистив (ла)

(прізвище, ім'я, по батьковій)

Кваліфікаційну роботу з оцінкою:

сума балів за 100-бальною шкалою \_\_\_\_\_

за шкалою ECTS \_\_\_\_\_

за національною шкалою \_\_\_\_\_

Секретар ЕК

\_\_\_\_\_

(науковий ступінь, вчене звання)

\_\_\_\_\_

(підпис)

\_\_\_\_\_

(прізвище, ім'я, по батьковій)

## АНОТАЦІЯ

Дубина В.П. Інформаційна система контролю руху автотранспорту – Кваліфікаційна робота на правах рукопису.

Кваліфікаційна робота на здобуття освітнього ступеня магістра за спеціальністю 126 – Інформаційні системи та технології. – Поліський національний університет, Житомир, 2023.

Дипломна робота присвячена розробці інформаційної системи моніторингу громадського транспорту. У роботі досліджується поточний стан транспортної системи міста та аналізуються існуючі підходи до моніторингу транспорту.

Запропоновано архітектуру системи та методику збору та обробки даних про рух транспорту. Особлива увага приділяється розробці алгоритмів аналізу даних, які дозволяють визначати місцеположення транспорту на даний час та прогнозувати час прибуття транспорту до зупинки. Результати дослідження можуть бути використані для покращення ефективності транспортної системи та забезпечення комфортних умов пересування громадян.

Ключові слова: транспорт, інформаційна система, аналіз даних, місцеположення.

## SUMMARY

Dubyna V.P. Information system for monitoring public transport movement – Qualification work in manuscript form.

Qualification work for the degree of Master in the specialty 126 – Information Systems and Technologies. – Polissia National University, Zhytomyr, 2023.

The thesis is dedicated to the development of an information system for monitoring public transport. The current state of the city's transport system is studied, and existing approaches to transport monitoring are analyzed.

The system architecture and methodology for collecting and processing data on transport movement are proposed. Special attention is paid to the development of algorithms for data analysis, which allow determining the current location of transport and predicting the arrival time of transport at the stop. The research results can be used to improve the efficiency of the transport system and provide comfortable conditions for citizens' movement.

**Keywords:** transportation, information system, data analysis, location.

## ЗМІСТ

<b>ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ .....</b>	<b>5</b>
<b>ВСТУП .....</b>	<b>6</b>
<b>РОЗДІЛ 1 ПРЕДСТАВЛЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ПІДТРИМКИ РОБОТИ ЛОГІСТИЧНОЇ КОМПАНІЇ.....</b>	<b>7</b>
<b>1.1 Аналіз предметної області дослідження системи підтримки роботи логістичної компанії.....</b>	<b>7</b>
<b>1.2 Функціональні вимоги системи.....</b>	<b>8</b>
<b>Висновки до розділу 1 .....</b>	<b>10</b>
<b>РОЗДІЛ 2 РОЗРОБКА СИСТЕМИ ПІДТРИМКИ РОБОТИ ЛОГІСТИЧНОЇ КОМПАНІЇ.....</b>	<b>11</b>
<b>2.1 Структура інформаційної системи.....</b>	<b>11</b>
<b>2.2 Архітектура інформаційної системи.....</b>	<b>15</b>
<b>Висновки до розділу 2 .....</b>	<b>17</b>
<b>РОЗДІЛ 3 РОБОТА З ДОДАТКОМ.....</b>	<b>18</b>
<b>3.1 Системні вимоги та процес встановлення.....</b>	<b>18</b>
<b>3.2 Опис інтерфейсу та роботи з додатком.....</b>	<b>18</b>
<b>Висновки до розділу 3 .....</b>	<b>20</b>
<b>ВИСНОВКИ .....</b>	<b>21</b>
<b>СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ .....</b>	<b>22</b>

## ВСТУП

У сучасних містах транспортна система є важливою складовою життя громадян, оскільки забезпечує швидке та комфортне пересування між різними точками міста. Проте, часто транспортна система стикається з проблемами, такими як затори, недостатній рівень безпеки та відсутність інформації про розклад руху транспорту. Цю проблему було розглянуто мною у роботах [11] та [12] У зв'язку з цим, необхідно розвивати нові технології, що допоможуть вирішити ці проблеми.

Інформаційна система моніторингу громадського транспорту є однією з таких технологій. Вона дає можливість в режимі реального часу відслідковувати місцезнаходження транспортних засобів та дізнаватись точний час їх прибуття до зупинок. Це дозволяє зменшити час очікування на зупинках та забезпечити комфортні умови пересування громадян.

У даній дипломній роботі буде розглянуто розробку інформаційної системи моніторингу громадського транспорту. У роботі буде проведено аналіз існуючих підходів до моніторингу транспорту та розроблено архітектуру системи та методику збору та обробки даних. Окрема увага буде приділена розробці алгоритмів аналізу даних для визначення місцезнаходження транспорту та прогнозування часу прибуття до зупинок.

Отримані результати можуть бути використані для покращення ефективності транспортної системи та забезпечення комфортних умов пересування громадян.

## **РОЗДІЛ 1 ПРЕДСТАВЛЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ПІДТРИМКИ РОБОТИ ЛОГІСТИЧНОЇ КОМПАНІЇ**

### **1.1 Аналіз предметної області дослідження системи підтримки роботи логістичної компанії**

Аналіз предметної області дослідження для дипломної роботи "інформаційна система підтримки роботи логістичної компанії" проведено з метою оцінки поточного стану транспортної системи міста та існуючих підходів до моніторингу транспорту.

В результаті аналізу встановлено, що велика кількість людей, стикаються з проблемами перевантаження громадського транспорту, неефективним розкладом руху та неякісним обслуговуванням пасажирів. У зв'язку з цим, розробка інформаційної системи підтримки роботи логістичної компанії є актуальною задачею.

Для вирішення цієї задачі досліджено існуючі підходи та методики моніторингу транспорту, такі як GPS-трекінг, встановлення датчиків руху транспорту та використання систем контролю проїзду. Було встановлено, що ці методики дозволяють отримувати інформацію про рух транспорту, але не завжди надійно визначають місцезнаходження транспорту та не забезпечують достатньої точності для прогнозування часу прибуття транспорту до зупинки.

У зв'язку з цим, було запропоновано розробити систему моніторингу транспорту, яка базуватиметься на використанні GPS-даних. Це дозволить забезпечити достатню точність для визначення місцезнаходження транспорту та прогнозування часу прибуття до зупинки, що покращить якість обслуговування пасажирів та ефективність транспортної системи міста.

Також в області дослідження активно використовуються різноманітні технології для збору та обробки даних про рух транспорту. Наприклад, системи геолокації та GPS-трекінгу, які дозволяють точно визначати місцезнаходження транспортних засобів у режимі реального часу. Також використовуються технології радіочастотної ідентифікації (RFID) та системи машинного зору для автоматичного розпізнавання номерних знаків.

У контексті розробки інформаційної системи, необхідно розглянути існуючі підходи до збору та обробки даних про рух транспорту та визначити їх переваги та недоліки. Також необхідно розробити власну архітектуру системи та відповідні алгоритми для аналізу та обробки даних, які дозволять визначати місцезнаходження транспорту та прогнозувати час прибуття до зупинки.

Для забезпечення моніторингу транспорту необхідно збирати та аналізувати велику кількість даних. Основними джерелами інформації є GPS-навігаційні приймачі, які встановлюються на транспортних засобах, а також системи контролю за рухом транспорту на зупинках.

Отримані дані необхідно обробляти та аналізувати з використанням спеціальних алгоритмів, щоб визначити місцезнаходження транспорту на даний момент, швидкість руху, час прибуття до наступної зупинки тощо. Для цього можна використовувати методи математичного моделювання, статистичний аналіз, машинне навчання тощо.

Після отримання та обробки даних інформація повинна бути візуалізована в зручному для користувача форматі, наприклад, на інтерактивній мапі міста. Це дозволить користувачам моніторити рух транспорту в режимі реального часу, отримувати інформацію про час прибуття транспорту до конкретної зупинки, місцезнаходження транспорту, існуючі маршрути тощо.

Така інформаційна система моніторингу транспорту може бути використана для покращення ефективності транспортної системи, зменшення часу очікування на зупинках, скорочення витрат на паливо та зниження викидів в атмосферу. Також вона може бути корисною для пасажирів, які зможуть планувати свій маршрут заздалегідь та отримувати актуальну інформацію про рух транспорту.

## **1.2 Функціональні вимоги системи**

Інформаційна система моніторингу громадського транспорту має



задовольняти наступні функціональні вимоги:

- Збір і збереження даних про рух транспорту в режимі реального часу;
- Аналіз даних про рух транспорту з метою визначення місцеположення транспорту на даний час і прогнозування часу прибуття до зупинки;
- Відображення інформації про рух транспорту на карті міста;
- Надання інформації про рух транспорту користувачам в режимі реального часу через мобільний додаток або веб-інтерфейс;
- Можливість моніторингу роботи системи та аналізу даних з попередніх періодів.

Для збору та збереження даних про рух транспорту в режимі реального часу можна використовувати різноманітні технології, такі як GPS-трекери, системи автоматичного визначення місцеположення на основі GSM-модулів та інші. Для збереження даних можна використовувати реляційну базу даних, наприклад, MySQL або PostgreSQL, які дозволять швидко та ефективно зберігати та оновлювати дані в режимі реального часу.

Для аналізу даних про рух транспорту з метою визначення місцеположення транспорту на даний час та прогнозування часу прибуття до зупинки можна використовувати різноманітні алгоритми машинного навчання, такі як алгоритми класифікації та кластеризації даних.

Для відображення інформації про рух транспорту на карті міста можна використовувати картографічні сервіси, такі як Google Maps або OpenStreetMap, та відповідні бібліотеки, наприклад, Leaflet. Також можна використовувати власні карти міста, якщо вони є у доступі.

Для надання інформації про рух транспорту користувачам в режимі реального часу через мобільний додаток або веб-інтерфейс можна розробити відповідні додатки або веб-сайти, що будуть отримувати дані про рух транспорту з бази даних та відображати їх користувачам. Для цього можна використовувати різноманітні технології, такі як React, Angular, Vue.js та Node.js.

## **Висновки до розділу 1**

На даний час інформаційна система моніторингу громадського транспорту є досить розвинутою галуззю, що дозволяє покращити якість надання послуг в сфері транспорту. Основною метою її створення є визначення місцезнаходження транспортних засобів та прогнозування часу прибуття до зупинок, що є важливим для пасажирів та організаторів транспортних послуг.

У процесі дослідження предметної області було виявлено, що на сьогоднішній день існує велика кількість різноманітних технологій, методів та інструментів для збору, збереження, аналізу та відображення даних про рух транспорту. Було проаналізовано різні підходи до збору даних, зокрема використання GPS, Wi-Fi та інших технологій.

Також було проаналізовано основні функціональні вимоги до інформаційної системи моніторингу громадського транспорту. Для реалізації цих вимог було розглянуто різні методи та інструменти.

## РОЗДІЛ 2 РОЗРОБКА СИСТЕМИ ПІДТРИМКИ РОБОТИ ЛОГСТИЧНОЇ КОМПАНІЇ

### 2.1 Структура інформаційної системи

Оскільки одним з основних аспектів цієї інформаційної системи є зручність, то її слід реалізувати у вигляді мобільного додатку. Також можна використовувати стандартну гугл-мапу для відображення інформації про маршрути і точки.

Для створення додатку на Android для відстеження автобусів за допомогою Google карт потрібно виконати наступні кроки:

Налаштувати середовища розробки: Встановимо середовище розробки Android Studio на свій комп'ютер та налаштуємо його для розробки додатків на мові Java.

Створити проект: Створімо новий проект у середовищі Android Studio. В проекті створімо активності для домашньої сторінки, екрану входу, екрану мапи, екрану контролю компанії та екрану контролю водія.

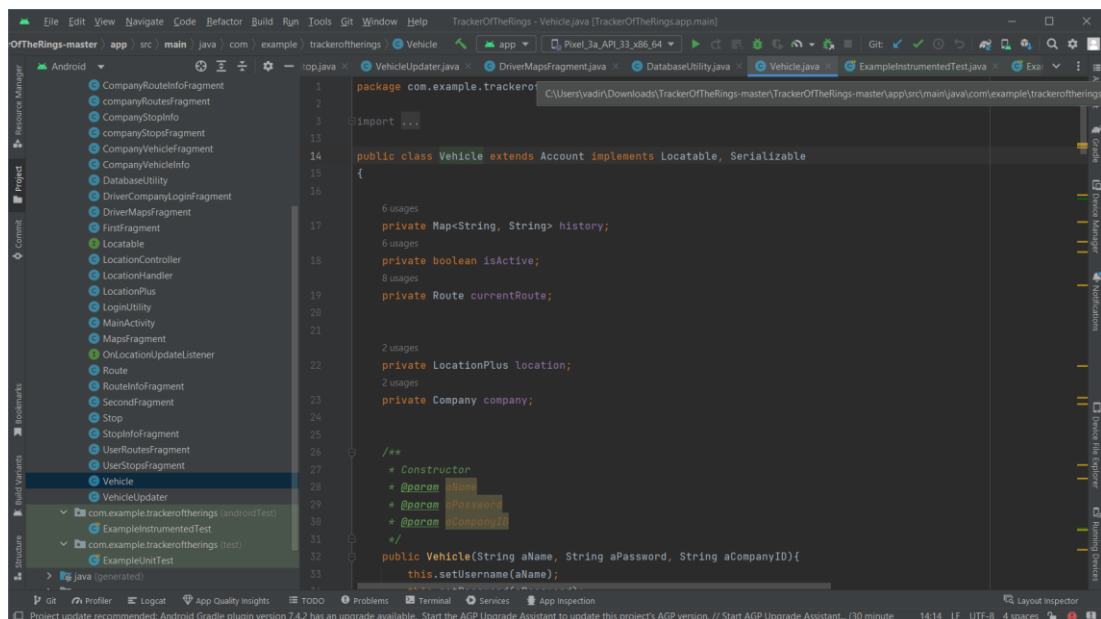


Рисунок 2.1 – Середовище Android Studio

Додати Google мапи: Додаймо пакет Google Maps API для Android до проекту та зареєструємо API-ключ у консолі Google Cloud. (Рис. 2.2)

```

<meta-data
    android:name="com.google.android.geo.API_KEY"
    android:value="{MAPS_API_KEY}" /><!-- API key

```

Рисунок 2.2 – Реєстрація API-ключа

Розробити мапи: Будемо використовувати Google Maps API для створення мапи та додавання на неї маркерів для водіїв та зупинок. Також додаймо можливість вибору маршруту.

Додати відстеження місцезнаходження водія: Використаємо Google Maps API для відстеження місцезнаходження водіїв та передачі цих даних компанії та користувачам.

```

package com.example.trackeroftherings;

import android.location.Location;

|
2 usages 2 implementations
public interface Locatable {

    2 implementations
    public LocationPlus getLocation();

    2 implementations
    public void setLocation(LocationPlus aLocation);

}

```

Рисунок 2.3 – реалізація надання об'єкту місцеположення та можливість його отримати

Додати авторизацію: Додаймо можливість авторизації водіїв та компаній через екрани входу.

Додати функціонал компанії: Додаймо можливість додавати, видаляти та редагувати зупинки та маршрути, додавати нові облікові записи водіїв та контролювати поточний парк водіїв.

Для реалізації функціоналу компанії в додатку для відстеження автобусів можна використати базу даних, де будуть зберігатися інформація про зупинки, маршрути та водіїв.



```

113 showBottomSheetDialog();
114 return binding.getRoot();
115 }
116
117 @SuppressWarnings("ResourceColor")
118 public void showBottomSheetDialog() {
119     BottomSheetDialog bottomSheet = new BottomSheetDialog(this.getContext());
120     bottomSheet.setContentView(R.layout.bottom_sheet_stop_route_vehicle_edit_info);
121     TextView text = bottomSheet.findViewById(R.id.info);
122     LinearLayout linear1 = bottomSheet.findViewById(R.id.linear);
123     text.setText("\n---VEHICLE NAME---\n" + vehicleToDisplay.getUsername());
124
125     try {
126         text.append("\n---ACTIVE ON ROUTE---\n" + vehicleToDisplay.getCurrentRoute().getName() + "\n");
127     } catch (NullPointerException e) {
128         text.append("\n---ACTIVE ON ROUTE---\n" + "NONE");
129     }
130
131     Button historyButton = new Button(this.getContext());
132     historyButton.setText("HISTORY");
133     historyButton.setId(0);
134     historyButton.setTextSize(20);
135     historyButton.setTextColor(Color.parseColor("#FFFFFF"));
136     historyButton.setBackgroundResource(R.color.teal_200);
137     historyButton.setDrawingCacheEnabled(true);
138     historyButton.setPadding(15, 10, 15, 10);
139     historyButton.setLayoutParams(new RelativeLayout.LayoutParams(ViewGroup.LayoutParams.WRAP_CONTENT, 100));
140     bottomSheet.findViewById(R.id.floatingActionButtonEdit).setOnClickListener(new View.OnClickListener() {
141         @Override
142         public void onClick(View v) {
143             bottomSheet.hide();
144             CompanyEditVehicle.setStatus(CompanyEditStatus.EDIT);
145             NavController.findNavController(CompanyVehicleInfo.this)
146                 .navigate(R.id.action_companyVehicleInfo_to_companyEditVehicle);
147         }
148     });
149     linear1.addView(historyButton);
150     bottomSheet.show();
151 }
152
153 @Override
154 public void onCreateView(@NonNull View view, @Nullable Bundle savedInstanceState) {
155     super.onCreate(savedInstanceState);
156     SupportMapFragment mapFragment =
157         (SupportMapFragment) getChildFragmentManager().findFragmentById(R.id.map);
158     if (mapFragment != null) {
159         mapFragment.getMapAsync(callback);
160     }
161 }
162 }

```

Рисунок 2.5 – Реалізація функції переглядання даних про водія

Для доступу до бази даних можна використовувати SQL або ORM бібліотеки, такі як Room або Realm. Також можна використовувати хмарні бази даних, на кшталт Firebase. Для забезпечення безпеки даних можна застосовувати різні методи, такі як шифрування, автентифікацію та авторизацію.

```

1 package com.example.trackeroftherings;
2
3 import android.widget.Toast;
4
5 import androidx.annotation.NonNull;
6
7 import com.google.firebase.database.DataSnapshot;
8 import com.google.firebase.database.DatabaseError;
9 import com.google.firebase.database.DatabaseReference;
10 import com.google.firebase.database.FirebaseDatabase;
11 import com.google.firebase.database.ValueEventListener;
12
13 import java.util.ArrayList;
14 import java.util.List;
15 import java.util.concurrent.TimeUnit;
16
17 public class DatabaseUtility {
18
19     //References
20     public static FirebaseDatabase database = FirebaseDatabase.getInstance();
21     public static DatabaseReference companiesReference = database.getReference("Companies");
22     public static DatabaseReference vehiclesReference = database.getReference("Vehicles");
23     public static DatabaseReference stopsReference = database.getReference("Stops");
24     public static DatabaseReference routesReference = database.getReference("Routes");
25
26     /**
27      * Add method for Companies

```

Рисунок 2.6 – Підключення бази даних Firebase

## 2.2 Архітектура інформаційної системи

Нашу систему можна умовно поділити на три складові

- База даних;
- Інтерфейс;
- Процеси інформаційної системи.

Поєднання цих елементів утворює повну роботу системи.

Створювати базу даних було вирішено в хмарному сервісі Firebase Database, оскільки він має ряд серйозних переваг перед аналогічними сервісами, насамперед це безпечність, робота на платформі Google, та усунення необхідності використовувати сервер для своєї бази даних. Зручною функцією сервісу є те, що в ньому можна обмежити доступ до даних в залежності від прав доступу користувачів. Також одним з вагомих плюсів це можливість користуватись сервісом безкоштовно, та підвищити план на більш ефективний, при необхідності.

Для збереження даних в базі даних потрібно створити модель даних. Модель даних - це схема бази даних, яка відображає структуру даних, які потрібно зберігати (рис 2.7).

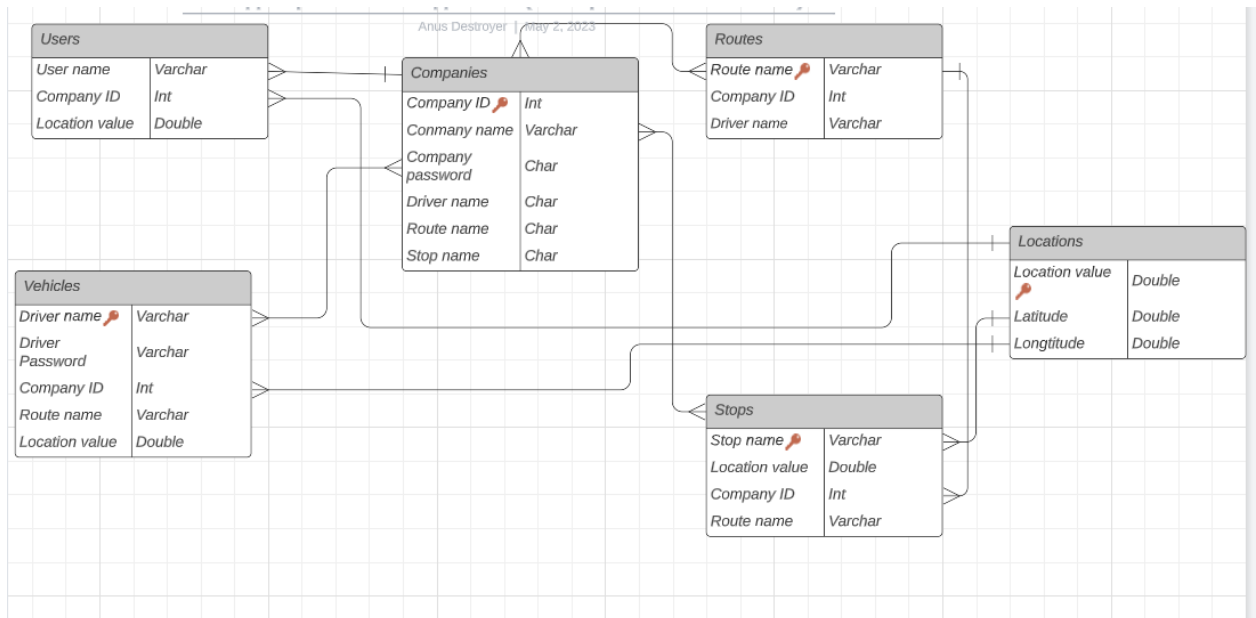


Рисунок 2.7 – Модель бази даних

Firestore SDK надає API для збереження та отримання даних з Firestore. Для збереження даних можна використовувати метод `setValue`, а для отримання - метод `addValueEventListener`.

Інтерфейс системи було реалізовано за допомогою інструментів Android Studio, він включає у себе елементи зображень, тексту та кнопок. Всі елементи інтерфейсу Android системи реалізуються на мові XML.

Процеси, які виконуються в системі, створюють її функціонал, і тому повинні мати чітку структуру. Вхідні та вихідні дані, та окремі блоки для реалізації цілей системи. (рис 2.8)

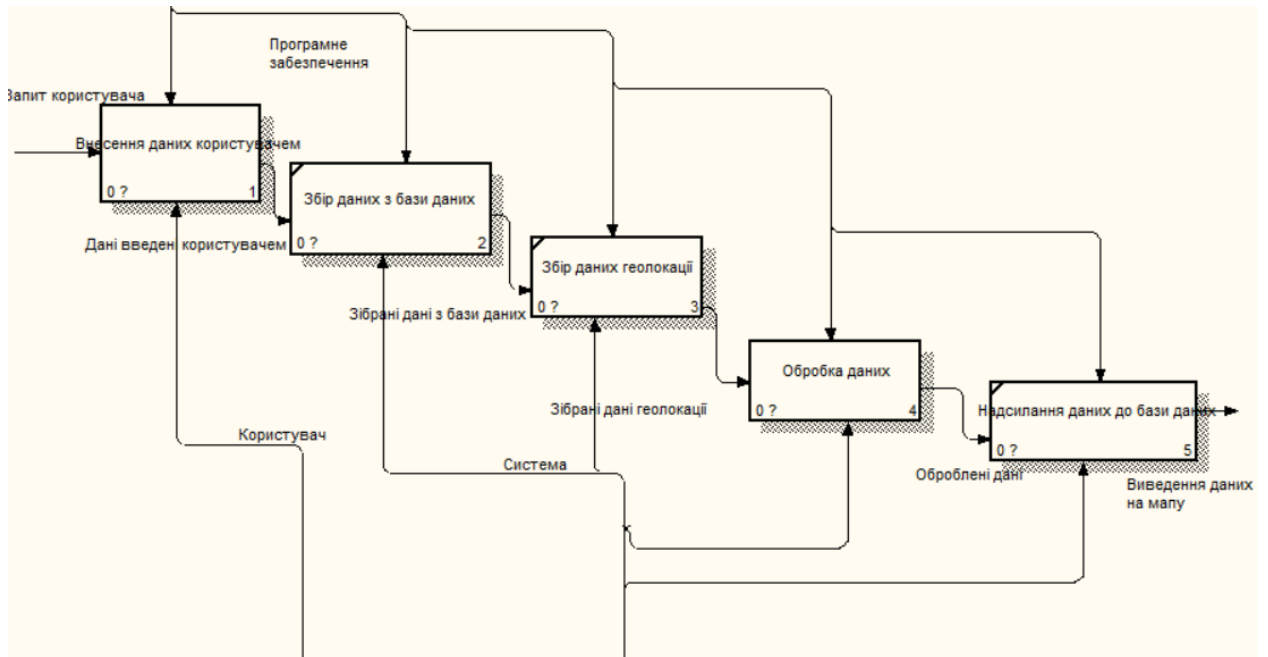


Рисунок 2.8 – Процеси системи

Для забезпечення правильної роботи програми, потрібно вистроїти послідовність виконання процесів (рис 2.9)



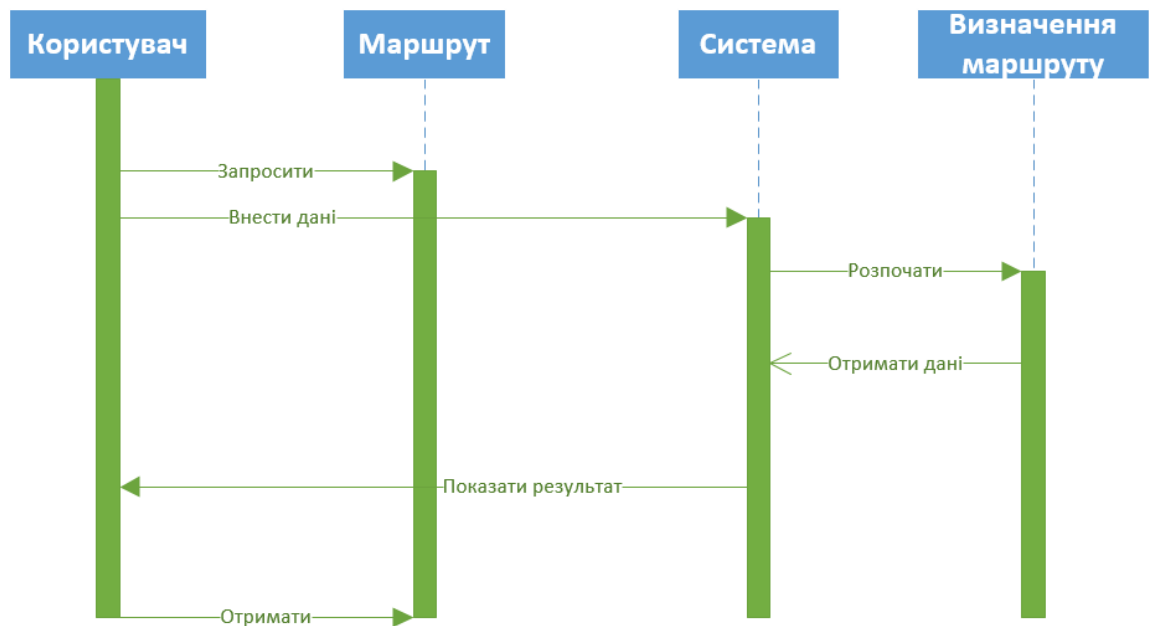


Рисунок 2.9 – Послідовність виконання процесів

## Висновки до розділу 2

Можна зробити висновок, що розроблена система має складну структуру, яка складається з багатьох компонентів, що взаємодіють між собою. Така структура дозволяє системі збирати великий обсяг даних про рух транспорту та їх аналіз.

Розроблена система має модульну архітектуру, що дає можливість розширювати та модифікувати її функціонал відповідно до потреб користувачів. Така архітектура дозволяє змінювати окремі модулі без необхідності зміни всієї системи.

Також система має достатньо високу швидкість обробки даних та може працювати в реальному часі. Це дозволяє користувачам системи миттєво отримувати необхідну інформацію про рух транспорту та ефективно управляти ним.

## **РОЗДІЛ 3 РОБОТА З ДОДАТКОМ**

### **3.1 Системні вимоги та процес встановлення**

Для того, щоб додаток міг функціонувати, потрібен пристрій з встановленою на ньому операційною системою Android 8.0 або вище, підключення до інтернет мережі, а також:

- процесор з архітектурою ARM або x86;
- не менше 1 Гб оперативної пам'яті;
- GPS-модуль;
- 20 Мб вільного місця

Для встановлення додатку потрібно завантажити APK-файл на пристрій, та відкрити його, після чого буде проведено процедуру встановлення додатку. Після того, як додаток буде встановлено, на робочому столі з'явиться іконка, через яку можна буде запустити застосунок.

### **3.2 Опис інтерфейсу та роботи з додатком**

При відкриванні додатку користувача зустрічає стартове меню, в якому можна побачити 3 кнопки: Driver, User та Company, а також кнопку «Homepage», яка повертає користувача до стартового меню з будь-якої сторінки. Натискання на кожну з кнопок (Driver, User та Company) переводить користувача у меню входу, однак в залежності від натиснутої кнопки, можна побачити різні поля для вводу, але загалом це Username, Password і Company ID

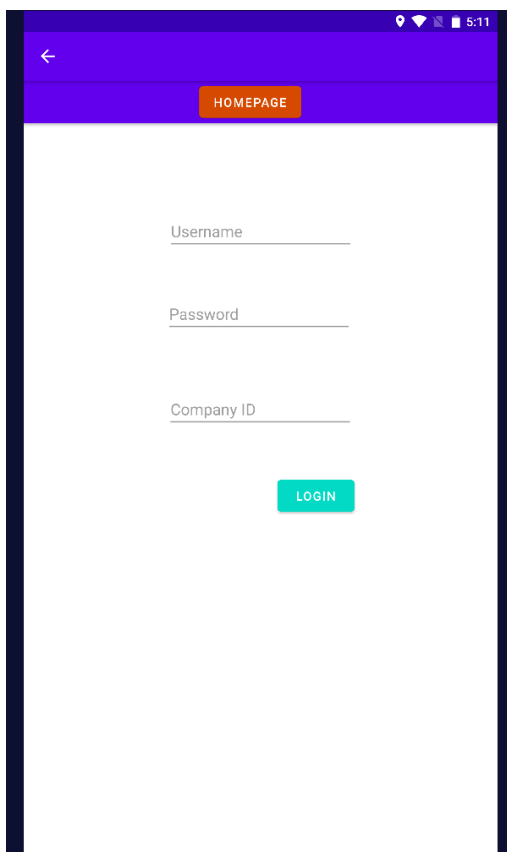


Рисунок 3.1 – Екран входу

Далі нас зустрічатиме Гугл-мапа, на якій буде відзначено вашу геолокацію.

Якщо ви увійшли в якості водія, то у вас буде можливість обрати маршрут, за яким буде курсувати транспорт. Після обрання маршруту на мапі з'являться маркери на зупинках маршруту, при приближенні до зупинки на відстань 50 метрів, інформація про знаходження водія на зупині додається у історію поїздки.

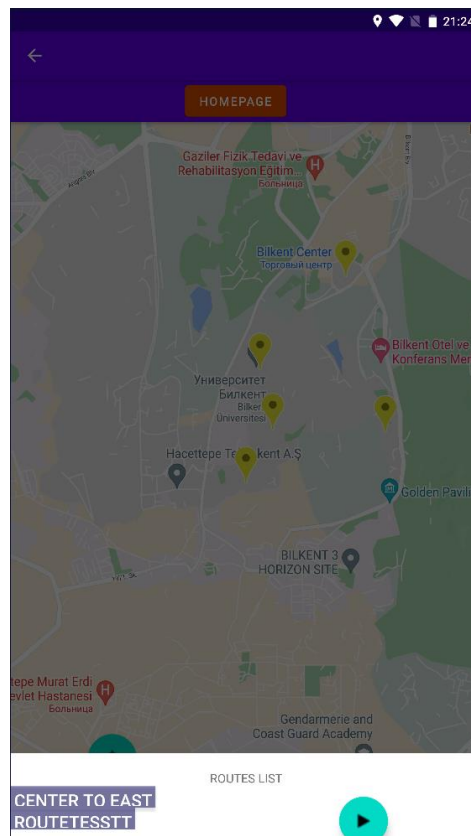


Рисунок 3.2 – Вибір маршруту водія та відзначення зупинок

При вході у профіль компанії, можна відстежити місцеположення водіїв, які на даний момент в дорозі, також можна переглянути можливі маршрути та зупинки і змінити інформацію про них.

Коли ви входите у ролі звичайного користувача, вам надається можливість переглядати маршрути та зупинки, а також місцеположення водія.

### Висновок до розділу 3

У даному розділі були описані системні вимоги та процес встановлення додатку на пристрій з операційною системою Android 8.0 або вище. Було також надано опис інтерфейсу та роботи з додатком, включаючи можливості для водіїв, користувачів та компаній. Додаток надає користувачам зручний спосіб відстежувати рух транспорту, переглядати маршрути та зупинки, а також місцеположення водіїв. Завдяки його функціональності можна забезпечити більш ефективно та зручно переміщення пасажирів та водіїв.

## ВИСНОВКИ

У результаті виконання дипломної роботи була розроблена інформаційна система моніторингу транспорту, яка дозволяє користувачам в режимі реального часу відстежувати місцезнаходження транспортних засобів та планувати свої поїздки. Система складається з трьох рівнів доступу: водії, звичайні користувачі та адміністратори компаній.

Основними функціями системи є відображення місцезнаходження транспорту на мапі, відстеження маршрутів, зупинок та історії поїздок. Крім того, водії можуть обирати маршрути, змінювати стан поїздок та надавати інформацію про розклади руху транспорту.

Під час розробки системи було використано сучасні технології, такі як Google Maps API, Firebase та Android-платформа, що дозволило досягти високої швидкості роботи та ефективного розподілення завдань між сервером та клієнтською частиною.

Таким чином, інформаційна система моніторингу транспорту є корисним інструментом для покращення якості обслуговування громадського транспорту та зручності пасажирів.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Lane, N. D., Miluzzo, E., Lu, H., Peebles, D., Choudhury, T., & Campbell, A. T. (2010). A survey of mobile phone sensing. *Communications Magazine, IEEE*, 48(9), 140-150.
2. Namiot, Dmitry, and Manfred Sneps-Sneppe. "On Open Source Mobile Sensing." *Internet of Things, Smart Spaces, and Next Generation Networks and Systems*. Springer International Publishing, 2014. 82-94.
3. Chennuru, S., Chen, P. W., Zhu, J., & Zhang, J. Y. (2012). Mobile Lifelogger Recording, Indexing, and Understanding a Mobile User's Life. In *Mobile Computing, Applications, and Services* (pp. 263- 281). Springer Berlin Heidelberg.
4. Sneps-Sneppe, M., & Namiot, D. (2015). Smart Socket for Activity Monitoring. arXiv preprint arXiv:1502.06904.
5. Namiot, D., & Schneps-Schneppe, M. (2013). Smart Cities Software from the developer's point of view. arXiv preprint arXiv:1303.7115.
6. Reddy, Sasank, et al. "MobiSense—mobile network services for coordinated Participatory Sensing." *Autonomous Decentralized Systems, 2009. ISADS'09. International Symposium on. IEEE, 2009*.
7. Opensignal <http://opensignal.com> Retrieved: Jul, 2015
8. WeatherSignal <http://weathersignal.com> Retrieved: Jul, 2015
9. Katevas, K., Haddadi, H., & Tokarchuk, L. (2014, September). Poster: Sensingkit: A multi-platform mobile sensing framework for large-scale experiments. In *Proceedings of the 20th annual international conference on Mobile computing and networking* (pp. 375-378). ACM.
10. Funf Open Sensing Framework <http://funf.org> Retrieved: Jul, 2015
11. Дубина В.П., Розробка та впровадження системи контролю руху автотранспорту. URL: [https://ontu.edu.ua/download/konfi/2023/Conference\\_abstract-IT-21-22-04-23.pdf](https://ontu.edu.ua/download/konfi/2023/Conference_abstract-IT-21-22-04-23.pdf)
12. Дубина В.П., Інформаційна система контролю руху автотранспорту. URL: [https://drive.google.com/file/d/1qBxHc7bTla--u6FbuPtHKucXxggF18ro/view?usp=share\\_link](https://drive.google.com/file/d/1qBxHc7bTla--u6FbuPtHKucXxggF18ro/view?usp=share_link)