Міністерство освіти і науки України  
Харківський національний університет радіоелектроніки

Факультет навчально-науковий центр заочної форми навчання  
Кафедра програмної інженерії

СПЕЦИФІКАЦІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Веб-сервісу для відстеження руху міського транспорту

Харків

2025

# ЗМІСТ

[1 Вступ 4](#_Toc200296763)

[1.1 Огляд продукту 4](#_Toc200296764)

[1.2 Мета 4](#_Toc200296765)

[1.3 Межі 4](#_Toc200296766)

[1.4 Посилання 4](#_Toc200296767)

[1.5 Означення та абревіатури 5](#_Toc200296768)

[2 Загальний опис 6](#_Toc200296769)

[2.1 Перспективи розвитку 6](#_Toc200296770)

[2.2 Функції продукту 6](#_Toc200296771)

[2.3 Характеристики користувачів 6](#_Toc200296772)

[2.4 Загальні обмеження 7](#_Toc200296773)

[2.5 Припущення в залежності 7](#_Toc200296774)

[3 Специфікація вимог 9](#_Toc200296775)

[3.1 Вимоги до зовнішніх інтерфейсів 9](#_Toc200296776)

[3.1.1 Інтерфейс користувача 9](#_Toc200296777)

[3.1.2 Програмний інтерфейс 9](#_Toc200296778)

[3.1.4 Комунікаційний протокол 10](#_Toc200296779)

[3.1.5 Обмеження пам’яті 10](#_Toc200296780)

[3.1.6 Операції 10](#_Toc200296781)

[3.1.7 Функції продукту 11](#_Toc200296782)

[3.1.8 Припущення й залежності 11](#_Toc200296783)

[3.2 Властивості програмного продукту 11](#_Toc200296784)

[3.3 Атрибути програмного продукту 12](#_Toc200296785)

[3.3.1 Надійність 12](#_Toc200296786)

[3.3.2 Доступність 12](#_Toc200296787)

[3.3.3 Безпека 12](#_Toc200296788)

[3.3.4 Супроводжуваність 13](#_Toc200296789)

[3.3.5 Переносимість 13](#_Toc200296790)

[3.3.6 Продуктивність 13](#_Toc200296791)

[3.4 Вимоги бази даних 14](#_Toc200296792)

[3.5 Інші вимоги 15](#_Toc200296793)

[4 Додаткові матеріали 16](#_Toc200296794)

# 1 ВСТУП

## 1.1 Огляд продукту

Програмний застосунок для відстеження руху міського транспорту призначений для надання користувачам актуальної інформації про місцезнаходження, маршрути та розклад громадського транспорту у режимі реального часу. Система інтегрує дані з різних GTFS-систем (General Transit Feed Specification), що дозволяє об’єднувати інформацію про транспортні маршрути різних міст у єдиному інтерфейсі. Програмний застосунок складається з веб-додатку, бекенд-сервісу, бази даних та додаткових мікросервісів для обробки та агрегації даних.

## 1.2 Мета

Метою розробки є створення масштабованого програмного застосунку, який забезпечує зручний доступ до інформації про рух міського транспорту, дозволяє користувачам швидко будувати оптимальні маршрути з урахуванням пересадок та часу очікування, а також надає інструменти для міських адміністрацій та транспортних компаній для моніторингу ефективності транспортної мережі. Завдяки уніфікованому інтерфейсу та можливості підключення різних GTFS-джерел, система може бути адаптована для будь-якого міста без значних змін у базовій архітектурі

## 1.3 Межі

Програмний застосунок призначений для використання у веб-браузерах на сучасних платформах (Chrome, Firefox, Safari, Edge, Opera) з підтримкою WebSocket для оновлення інформації в реальному часі. Система не передбачає обов’язкової авторизації користувачів, не зберігає персональні дані та не реалізує функції бронювання чи оплати проїзду. Застосунок може інтегруватися з різноманітними GTFS-системами, але не підтримує автоматичну інтеграцію з іншими стандартами обміну даними про громадський транспорт без додаткових доопрацювань.

## 1.4 Посилання

GTFS Specification — https://developers.google.com/transit/gtfs;

PostgreSQL — <https://www.postgresql.org/>.

FastAPI — https://fastapi.tiangolo.com/;

React — https://react.dev/;

pgRouting — https://pgrouting.org/;

PostGIS — https://postgis.net/.

## 1.5 Означення та абревіатури

HTTP – Hypertext Transfer Protocol

JSON – JavaScript Object Notation

RPS – Requests per Second

API – Application Programming Interface

AWS – Amazon Web Services

WS – Web Sockets

GTFS – General Transit Feed Specification

RDBMS – Relational Database Management System

ORM – Object-Relational Mapping

AVL - Automated Vehicle Location

SIRI - Service Interface for Real Time Information

NeTEx - Network Timetable Exchange

SSR – Server-Side Rendering

Kafka – Розподілена потокова платформа для обміну повідомленнями

LocalStorage – Механізм веб-браузера для зберігання даних на стороні клієнта

FastAPI – Веб-фреймворк для Python

React – JavaScript-бібліотека для побудови інтерфейсів користувача

GTFS Static – Статична частина GTFS, що містить інформацію про маршрути, зупинки, розклади

GTFS Realtime – Розширення GTFS для передачі даних у реальному часі

# 2 ЗАГАЛЬНИЙ ОПИС

## 2.1 Перспективи розвитку

Програмний застосунок для відстеження руху міського транспорту розроблений з урахуванням можливості подальшого розширення функціональності. В майбутньому планується:

підтримка додаткових стандартів обміну даними (SIRI, NeTEx, TransXChange); для інтеграції з іншими транспортними системами.

розробка мобільних додатків для iOS та Android;

авторизація для персоналізації можливостей додатку;

інтеграція з платіжними системами для можливості придбання квитків онлайн;

розширення аналітики для міських адміністрацій та операторів транспорту;

підтримка мультимовності для забезпечення доступності користувачам з різних країн.

## 2.2 Функції продукту

Основні функції програмного застосунку:

відображення маршрутів громадського транспорту на інтерактивній мапі;

пошук оптимального маршруту між двома точками з урахуванням пересадок та часу очікування;

відстеження руху транспорту в реальному часі за допомогою GTFS Realtime;

відображення розкладу руху для вибраних зупинок та маршрутів;

фільтрація транспорту за типом (автобус, трамвай, метро тощо);

відображення інформації про зупинки (назва, координати, доступні маршрути);

інтеграція з різними GTFS-джерелами для підтримки багатьох міст;

інструменти для адміністраторів (моніторинг стану транспорту, аналіз завантаженості маршрутів).

## 2.3 Характеристики користувачів

Основні категорії користувачів системи:

звичайні користувачі**:**

* + пасажири громадського транспорту, які потребують актуальної інформації про маршрути та розклад;
  + не мають спеціальних знань, працюють через веб-інтерфейс;

адміністратори системи**:**

* + мають доступ до аналітичних інструментів та можуть налаштовувати джерела даних;

розробники**:**

* + програмісти, які інтегрують нові джерела даних або розширюють функціональність системи;
  + працюють з API та базою даних.

## 2.4 Загальні обмеження

**обмежена підтримка стандартів:**

* + наразі система підтримує лише GTFS (Static та Realtime), інші стандарти потребують доопрацювання;

**відсутність мобільних додатків:**

* + початкова версія доступна лише через веб-браузер;

**відсутність авторизації:**

* + користувачі не можуть зберігати свої налаштування;

**відсутність платіжних функцій:**

* + неможливо купити квиток або здійснити оплату через систему.

## 2.5 Припущення в залежності

Припущення, від яких залежить коректна робота системи:

наявність GTFS-джерел даних:

* + система працює лише за умови наявності GTFS-файлів (Static та Realtime) для відповідних міст;

стабільний інтернет-зв’язок:

* + для роботи веб-інтерфейсу та оновлення даних у реальному часі необхідний стабільний доступ до інтернету;

сумісність браузерів:

* + система розрахована на роботу у сучасних веб-браузерах з підтримкою WebSocket;

сумісність серверного ПЗ:

* + для роботи бекенду необхідні актуальні версії PostgreSQL, PostGIS, pgRouting, FastAPI;

відсутність значних змін у GTFS-стандарті:

* + система розрахована на сумісність з поточними версіями GTFS, значні зміни можуть вимагати оновлення коду;

відсутність DDoS-атак або значних перевантажень:

* + система не має вбудованих механізмів захисту від масових атак або перевантажень.

# 3 СПЕЦИФІКАЦІЯ ВИМОГ

## 3.1 Вимоги до зовнішніх інтерфейсів

### 3.1.1 Інтерфейс користувача

Веб-інтерфейс:

інтерактивна мапа з відображенням маршрутів, зупинок та рухомого складу;

пошук маршрутів між двома точками з можливістю вибору часу відправлення;

відображення розкладу руху на вибраних зупинках;

фільтрація транспорту за типом (автобус, трамвай, метро тощо);

відображення інформації про зупинки та маршрути;

інтуїтивно зрозумілий та адаптований до різних екранів інтерфей.

Сервер:

стандартні серверні рішення (x86-64 архітектура);

вимоги до процесора, оперативної пам’яті та сховища даних залежать від обсягу оброблюваної інформації та кількості користувачів.

Клієнт:

сучасний комп’ютер або мобільний пристрій з підтримкою сучасних веб-браузерів;

достатня оперативна пам’ять (мінімум 2 ГБ) та пропускна здатність інтернет-з’єднання.

### 3.1.2 Програмний інтерфейс

API:

RESTful API для взаємодії з бекендом;

WebSocket для оновлення даних у реальному часі;

підтримка JSON для обміну даними.

Інтеграція:

підтримка GTFS (Static та Realtime) для імпорту маршрутів, зупинок та даних про рух транспорту;

можливість додаткових інтеграцій через API сторонніх сервісів.

### 3.1.4 Комунікаційний протокол

HTTP/HTTPS:

* + для обміну даними між клієнтом та сервером;

WebSocket:

* + для оновлення інформації про рух транспорту в реальному часі;

GTFS:

* + для імпорту та оновлення даних про маршрути, зупинки та розклад.

### 3.1.5 Обмеження пам’яті

Сервер:

мінімум 8 ГБ оперативної пам’яті для середнього навантаження;

рекомендовано 16 ГБ та більше для великих міст або високого трафіку.

Клієнт:

достатньо стандартної пам’яті сучасного браузера (LocalStorage для кешування даних).

### 3.1.6 Операції

Основні операції:

пошук маршрутів;

відображення інформації про зупинки та маршрути;

оновлення даних у реальному часі;

фільтрація транспорту за типом;

імпорт даних з GTFS-джерел.

Додаткові операції:

моніторинг стану транспорту для адміністраторів;

аналітика руху транспорту.

### 3.1.7 Функції продукту

відображення маршрутів та зупинок на мапі;

пошук оптимального маршруту з урахуванням пересадок та часу очікування;

відстеження руху транспорту в реальному часі;

відображення розкладу руху на вибраних зупинках;

фільтрація транспорту за типом;

імпорт даних з GTFS-джерел;

моніторинг стану транспорту для адміністраторів.

### 3.1.8 Припущення й залежності

наявність GTFS-джерел даних;

стабільний інтернет-зв’язок для роботи клієнта та сервера;

сумісність браузерів з WebSocket;

відсутність значних змін у GTFS-стандарті;

відсутність DDoS-атак або значних перевантажень системи.

## 3.2 Властивості програмного продукту

Масштабованість:

* + Система здатна обробляти дані для різних міст та адаптуватися до зростання кількості користувачів;

Модульність:

* + Архітектура дозволяє легко додавати нові функції та інтегрувати нові джерела даних;

Відкритість:

* + Підтримка відкритих стандартів (GTFS) та можливість інтеграції з іншими системами;

Кросс-платформенність:

* + Веб-інтерфейс доступний з будь-якої платформи з сучасним браузером.

## 3.3 Атрибути програмного продукту

### 3.3.1 Надійність

стабільна робота:

* + система працює без збоїв у штатному режимі;

відновлення після збоїв:

* + можливість швидкого відновлення роботи після апаратних або програмних збоїв.

### 3.3.2 Доступність

відкритий доступ:

* + система доступна для всіх користувачів через веб-інтерфейс без обов’язкової реєстрації;

адаптація до різних пристроїв:

* + інтерфейс адаптований для роботи на комп’ютерах, планшетах та смартфонах.

### 3.3.3 Безпека

захист даних:

* + система не зберігає персональні дані користувачів;

HTTPS:

* + всі дані передаються по захищеному протоколу;

обмежений доступ до адміністративних функцій:

* + адміністративні інструменти доступні лише авторизованим користувачам.

### 3.3.4 Супроводжуваність

документований код:

* + код системи добре документований для полегшення підтримки та розвитку;

модульна архітектура:

* + дозволяє легко вносити зміни та додавати нові функції;

типізація:

* + присутність типізації на бекенді і на фронтенді дозволяє краще розуміти структуру ПЗ і полегшує внесення змін у кодову базу.

### 3.3.5 Переносимість

підтримка різних операційних систем:

* + серверна частина працює на Linux та Windows завдяки Docker;

веб-клієнт:

* + доступний з будь-якої платформи з сучасним браузером.

### 3.3.6 Продуктивність

швидка обробка запитів:

* + оптимізовані алгоритми пошуку маршрутів та обробки даних;

ефективне використання ресурсів:

* + мінімальні вимоги до апаратних ресурсів для клієнта.

## 3.4 Вимоги бази даних

тип БД:

* + реляційна база даних PostgreSQL з розширеннями PostGIS та pgRouting;

структура даних:

* + таблиці для маршрутів, зупинок, розкладів, рухомого складу, історії руху;

інтеграція:

* + підтримка імпорту даних з GTFS-файлів

масштабування:

* + можливість збільшення обсягу даних без втрати продуктивності;

резервне копіювання:

* + регулярне резервне копіювання даних для забезпечення надійності.

## 3.5 Інші вимоги

тестування:

* + система повинна проходити регулярне тестування на коректність роботи та безпеку;

документація:

* + наявність інструкцій для користувачів, адміністраторів та розробників;

підтримка:

* + можливість технічної підтримки користувачів та адміністраторів.

# 4 ДОДАТКОВІ МАТЕРІАЛИ

GTFS-специфікація: https://developers.google.com/transit/gtfs;

документація PostgreSQL: https://www.postgresql.org/docs/;

документація PostGIS: https://postgis.net/documentation/;

документація pgRouting: https://docs.pgrouting.org/;

документація FastAPI: https://fastapi.tiangolo.com/;

документація React: https://react.dev/.