**Анотація**

Магістерська кваліфікаційна робота містить опис розробки мобільної системи для допомоги незрячим особам орієнтуватись в приміщеннях.

Проект включає розробку веб клієнту, серверного застосунку, бази даних та мобільного застосунку з використанням Bluetooth модуля ESP32.

З врахуванням існуючих аналогів систем для незрячих осіб, появилась необхідність в розробці нового рішення з іншим підходом до взаємодії незрячого користувача з навколишнім світом шляхом озвучування йому інформації, що сприятиме йому в орієнтуванні.

Існуючі підходи акцентують свою увагу на подачі звукових сигналів лише для перенаправлення людини в певному напрямку, заборони або дозволу на рух чи інші вказівки.

Метою є створення системи, яка базується на взаємодії Bluetooth пристроїв з смартфоном для отримання та озвучування тексту в залежності від відстані до давачів на основі величини сигналу від Bluetooth давача.

Об’єктом дослідження є процес взаємодії телефону особи з Bluetooth давачем для орієнтування незрячих людей в приміщенні на основі сили сигналу давача.

Предметом дослідження є алгоритм визначення кращого джерела з допомогою RSSI (Received Signal Strength Indication).

Обсяг роботи 68 сторінок. Обсяг таблиць – 10. Обсяг зображень - 19.

**Abstract**

The master's qualification paper contains a description of the development of a mobile system to help blind people navigate indoors.

The project involves developing a web client, a server application, a database, and a mobile application with using Bluetooth module ESP32 to work. Taking into account the existing analogues of systems for blind people, there was a need to develop a new solution with a different approach to the interaction of the blind user with the world by voice-telling him information that would guide him in orientation.

Existing approaches emphasize the supply of audible signals only for redirection of a person in a particular direction, a prohibition or permission to move or other instructions.

The goal is to create a system based on the interaction of Bluetooth devices with a smartphone for receiving and voice typing depending on the distance to the sensors based on the magnitude of the signal from the Bluetooth sensor.

The object of the study is the process of contacting a person's phone with a Bluetooth sensor to orient the blind people in the room based on the signal strength of the sensor.

The subject of the study is the algorithm for determining the best source using RSSI (Received Signal Strength Indication).

The scope of work is 68 pages. The size of the tables is 10. The size of the images is 19.

**ЗМІСТ**

[РОЗДІЛ 1. АНАЛІЗ ІСНУЮЧИХ СХЕМ ТА МЕТОДІВ ВЗАЄМОДІЇ З BLUETOOTH ДАВАЧАМИ. РОЗРОБКА ЗАСОБІВ ТА ВИКОРИСТАННЯ ЕЛЕМЕНТІВ У МОБІЛЬНІЙ СИСТЕМІ 8](#_Toc531477793)

[1.1. Опис проблеми мобільної системи і засобів для її вирішення 8](#_Toc531477794)

[1.2. Огляд існуючих аналогів систем для допомоги незрячим особам 9](#_Toc531477795)

[1.3. Принцип роботи Bluetooth давача 13](#_Toc531477796)

[1.3.1. Апаратне забезпечення давача 14](#_Toc531477797)

[1.4. Алгоритм визначення величини сигналу Bluetooth давача 15](#_Toc531477798)

[1.4.1. RSSI 15](#_Toc531477799)

[Висновки до розділу 1 16](#_Toc531477800)

[РОЗДІЛ 2. АРХІТЕТКУРА МОБІЛЬНОЇ СИСТЕМИ 18](#_Toc531477801)

[2.1. Аналіз моб платформ. та технологій для розробки мобільної системи 18](#_Toc531477802)

[2.2. Інструменти розробки мобільної системи 19](#_Toc531477803)

[2.3. Особливості використання компонентів мобільної системи 20](#_Toc531477804)

[2.4. Середовище функціонування та вимоги інтерфейсів до системи 20](#_Toc531477805)

[2.5. Принцип взаємодії Bluetooth давача з смартфоном 22](#_Toc531477806)

[2.6. Взаємодія смартфону та серверу мобільної системи 22](#_Toc531477807)

[2.7. Взаємодія серверу з базою даних та веб клієнтом у мобільній системі 23](#_Toc531477808)

[2.8. Проектування графічного інтерфейсу користувача 24](#_Toc531477809)

[2.9. Проектування архітектури мобільної системи 26](#_Toc531477810)

[2.9.1. Проектування основних компонент і модулів мобільної системи 26](#_Toc531477811)

[2.9.2. Проектування класів системи 26](#_Toc531477812)

[Висновки до розділу 2 27](#_Toc531477813)

[РОЗДІЛ 3. РЕАЛІЗАЦІЯ МОБІЛЬНОЇ СИСТЕМИ 28](#_Toc531477814)

[3.1. Архітектура бази даних та проектування серверної частини мобільної системи 28](#_Toc531477815)

[3.2. Проектування Веб клієнта для мобільної системи 31](#_Toc531477816)

[3.3. Проектування мобільного клієнту для мобільної системи 32](#_Toc531477817)

[3.4. Реалізація взаємодії Bluetooth давача з смартфоном 34](#_Toc531477818)

[3.5. Результати розробки мобільної системи для орієнтації незрячих осіб в приміщеннях. 35](#_Toc531477819)

[Висновки до розділу 3 37](#_Toc531477820)

[РОЗДІЛ 4. ПРОВЕДЕННЯ ДОСЛІДЖЕННЯ МОБІЛЬНОЇ СИСТЕМИ ТА АНАЛІЗ ОТРИМАНИХ ДАНИХ 38](#_Toc531477821)

[4.1. Опис процесу дослідження мобільної системи 38](#_Toc531477822)

[4.2. Дослідження швидкості обміну даними між компонентами мобільної системи 39](#_Toc531477823)

[4.3. Результати взаємодії смартфону та Bluetooth давача 41](#_Toc531477824)

[Висновки до розділу 4 44](#_Toc531477825)

[РОЗДІЛ 5. ЕКОНОМІЧНА ХАРАКТЕРИСТИКА ПРОГРАМНОГО ПРОДУКТУ 46](#_Toc531477826)

[5.1. Економічна характеристика проектного рішення 46](#_Toc531477827)

[5.2. Розрахунок витрат на розробку та впровадження проектного рішення 46](#_Toc531477828)

[5.3. Визначення комплексного показника якості 50](#_Toc531477829)

[5.4. Визначення експлуатаційних витрат 52](#_Toc531477830)

[5.5. Розрахунок ціни споживання проектного рішення 54](#_Toc531477831)

[5.6. Визначення показників економічної ефективності 55](#_Toc531477832)

[Висновки до розділу 5. 57](#_Toc531477833)

[ЗАГАЛЬНІ ВИСНОВКИ 58](#_Toc531477834)

[СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ 59](#_Toc531477835)

[ДОДАТКИ 62](#_Toc531477836)

[Додаток А. Діаграма розгортання 62](#_Toc531477837)

[Додаток Б. Діаграма послідовностей 63](#_Toc531477838)

[Додаток В. Інструкція користувача 64](#_Toc531477839)

[B.1. Налаштування серверу та баз даних мобільної системи. 64](#_Toc531477840)

[B.2. Налаштування смартфон клієнту мобільної системи 64](#_Toc531477841)

[B.3. Робота з Bluetooth давачем 64](#_Toc531477842)

[Додаток Г. Код отримання Bluetooth пристроїв та даних про давачі 65](#_Toc531477843)

**ВСТУП**

**Вступ.** На сьогоднішній день відомо зовсім не багато систем чи приладів, що допомагали б незрячим особам орієнтуватись в просторі, рухатись за маршрутом тощо.

Доступні аналоги таких систем мають свій ряд переваг та недоліків. Зокрема, деякі системи з «маячками» мають постійне зовнішнє джерело живлення, та не потребують стороннього вмішування, але дорогі у покупці та монтажі. Вони не помітні особам без вад зору, що в свою чергу не викликає роздратування, оскільки людина може бачити світ навколо та не потребує циклічних сигналів навколо для орієнтації. Тому виникає потреба у створенні відокремленого та більш локалізованого під визначену категорію користувачів давача, що працюватиме непомітно та беззвучно для людей, що бачать, але будуть передавати інформацію для незрячих людей.

Основна ідея розробці мобільної системи, яка б дозволяла зчитувати з Bluetooth модулів, закріплених на дверях інформацію місцезнаходження незрячої людини та іншу інформацію (інформація про лабораторію, вихід з приміщення, поверх приміщення), яка була б корисна незрячим людям для орієнтації в приміщеннях. Завдання у визначенні з поміж багатьох пристроїв, що перекривають один одного, найближчого (потрібного) для обміну даними. Ідея може бути застосована не лише в приміщеннях, а у будь якому іншому середовищі перебування незрячих людей. Функціональність системи можна застосувати і у інших областях. Мобільна система доповнюватиметься веб інтерфейсом для додавання інформації про нові давачі, їх розміщення на карті, інструкцією для їх використання та сторінкою з пропозиціями щодо удосконалення системи. Усю інформацію оброблятиме сервер у взаємодії з базою даних розміщеними на хмарі.

# **РОЗДІЛ 1. АНАЛІЗ ІСНУЮЧИХ СХЕМ ТА МЕТОДІВ ВЗАЄМОДІЇ З BLUETOOTH ДАВАЧАМИ. РОЗРОБКА ЗАСОБІВ ТА ВИКОРИСТАННЯ ЕЛЕМЕНТІВ У МОБІЛЬНІЙ СИСТЕМІ**

## **Опис проблеми мобільної системи і засобів для її вирішення**

Одним із способів вирішення проблеми орієнтації незрячих осіб в приміщенні є розміщення в місцях приміщення [1] чи на дверях Bluetooth давачів [2]. У порівнянні з RFID мітками , за даними [3-4] Bluetooth давачі можуть працювати на більшій відстані та підтримуються всіма сучасними смартфонами. Давачі дають можливість разом з назвою та адресою передавати силу сигналу, що за [5] є чи не єдиним способом визначення сили сигналу. Давачі дають можливість окрім сили сигналу, визначити відстань до них [6].

Bluetooth давачі мають швидко взаємодіяти з смартфоном, та озвучити текст ще до того як користувач вийде з зони дії давача, тоді інформація буде не дійсна, а в гіршому випадку хибна. Вони повинні за можливості бути конфігурований на обмежену зону дії задля уникнення перекриття сигналів між поверхами приміщення або ж задля уникнення перекриття сигналів одне одного, бо в такому випадку зменшується точність та коректність розпізнання найближчого давача.

Мобільна система повинна мати веб клієнт для управління давачами, адже незрячі люди ймовірно не зможуть самостійно налаштувати її. Крім того, слід забезпечити такій системі гнучкість, на випадок ініціативи розробки застосунків від сторонніх клієнтів під інші платформи, шляхом створення для системи відкритого API з вичерпною документацією.

Сервіс як і база даних для зберігання даних мають бути у постійному доступі, тому повинні бути розміщені на хмарі для постійної роботи.

Веб клієнт для зручності користування повинен бути гнучкий та підлаштовуватись під будь який пристрій користувача, у випадку якщо користувач додаватиме новий давач чи переглядатиме існуючий, наприклад, з мобільного пристрою чи планшета.

Для прикладу, користувач вирішив розмістити давачі в корпусі університету. Він заходить на веб клієнт та вводить необхідні дані, що зберігаються у базі даних. Припустимо незрячий студент/викладач заходить в університет чи підходить до входу. Смартфон, знаходячись в стані пошуку пристроїв навколо, знаходить неподалік Bluetooth давач, заздалегідь розміщений користувачем. Отримуючи MAC адресу давача, смартфон здійснює запит до хмарного серверу та отримує текст для озвучування, наприклад: «Ви знаходитесь біля входу в 5 корпус Національного університету «Львівська політехніка», перед входом знаходяться 8 сходинок та вхід, що складається з двох секцій дверей». Заходячи на певний поверх, телефон розпізнає кілька давачів навколо. Визначивши силу сигналу кожного, смартфон здійснює з`єднання з тим, сила сигналу якого найбільша та озвучує текст. Базова система для озвучування тексту непогано справляться з озвучуванням тексту англійською, навіть одночасно з українською.

## **Огляд існуючих аналогів систем для допомоги незрячим особам**

За один з прикладів можна взяти OpenWorld, проект описаний у [7], який активно розвивається в Україні. OpenWorld - це соціальний проект, створений для того, щоб допомогти незрячим людям та людям з вадами зору за допомогою оптимізації і поліпшення інфраструктури міста, в якому вони живуть.

Розробники створили спеціальні маячки, які встановлені на міському транспорті, світлофорах та інших важливих місцях, кожен зможе встановити такий маячок на дверях власного дому, кав’ярні та будь-якому місці, де маячок може стати корисним.

Людина, з вадами зору, може отримати перелік маячків навколо місця її знаходження за допомогою спеціального мобільного додатку та отримати додаткову інформацію, прокласти маршрут та дізнатись точне положення об’єкту у просторі (звукове повідомлення з боку маячка) [8].

Система OpenWorld складається з трьох основних складових:

1. Bluetooth-маячки, обладнані динаміком, кріпляться на міський транспорт і об'єкти міської інфраструктури.
2. Спеціально розроблений мобільний додаток, за допомогою якого незряча людина має можливість знайти маячки навколо та керувати ними. Додаток постійно сканує наявність маячків поблизу. Щойно людина потрапляє в зону дії маячка, той відображається в додатку у вигляді пункту списку.

Для більш детальної інформації незряча людина зможе підключитися до маячка і отримати усі необхідні відомості.

Наприклад, якщо незряча людина обирає зі списку доступних їй маячків аптеку, то мобільний додаток OpenWorld дозволяє їй отримати інформацію про години роботи, адресу та іншу інформацію. У випадку оснащення транспорту маячками OpenWorld, незряча людина може дізнатися номер маршруту та його напрямок.

Точне розташування об’єкта в інфраструктурі міста незряча людина може отримати, натиснувши на клавішу “Викликати”, одразу після цього пролунає звуковий сигнал із маячка. Для людей із вадами зору саме звук є орієнтиром у просторі та за допомогою нього вони визначають напрямок руху. Додатковою функцією мобільного додатку OpenWorld є клавіша “Зателефонувати”, яка використовується для зв’язку з адміністратором закладу за телефоном, вказаним при реєстрації маячка. Важливою функцією є можливість пошуку маячків, зареєстрованих в базі даних OpenWorld, та прокладання маршруту у Google Maps (Рис. 1.1.).

|  |  |  |
| --- | --- | --- |
| https://lh4.googleusercontent.com/L27GWPU4uVcOAU_o9Yu5sDci0unO4jQaJ6rpOuAcKzzOhO-RUHi7YkEUsSbeZTf6kuJcv1rlxZuUY7eQ4cabjr0EnI6_hW3iMps0OcQQ6m26nwKF2lOQJDHMwLZaMgQUpTg0y5Yo | https://lh3.googleusercontent.com/5yET_DZ7Bh--OXBAoiVU1JSygoRNswGg3aUy2GUpqrLk6Kb11KkGjiYAEAuCsoiqf2cbyfS2pNuCifbWqa09P6Y6CSqEa_QKi-u20rJR7wJraQGSHu2m_-hVNQw9PtL-Sm09Ax3R | https://lh5.googleusercontent.com/LzFwqGC1_-jQue504_hIdUNVSIvbyxinhbXExs9Na6y_ECNARQ-E63ETaD1umvCM3HMKl_pgl3L_qzlIr9rrsfFohllT5ENbfoWIkFYjnGdWgvU0ObHDRGwkAvqXEVRwcSA5l7Cc |

Рис. 1.1. Інтерфейс мобільного застосунку

1. Сайт OpenWorld, над яким працює команда розробників програмного забезпечення, дає можливість створення персональної сторінки для кожного користувача. Завдяки цьому можна додавати нові маячки, зручно керувати ними, проглядати статистику та поточну інформацію про стан маячка.

На даному етапі функціонал сайту OpenWorld дозволяє додавати маячки, а уся інформація синхронізується з мобільним додатком.

Отже, підводячи підсумки, можна виділити такі основні переваги:

* можливість прокладання маршруту
* додаткова Інформація про місце розміщення маячка
* можливість викликати адміністратора закладу, у якому розміщений маячок
* можливість виклику маяка натиском кнопки

Недоліком системи є:

* звукове сповіщення від маячка, яке все ж чути людям без вад з зором
* надлишкова кількість дій для роботи з маяками
* висока вартість запуску системи. Проект не набрав необхідної суми для повноцінного запуску.

Ще одним яскравим аналогом є проект, розроблений вінницькими активістами - Mobilis.

Принцип дії дуже простий. Маяк монтують, наприклад, над дверима будівлі. Коли до них наближається людина з вадами зору, апарат подає сигнал. Тож незрячий знає, де розташовано вхід/вихід.

Маячок активується автоматично за умови, що користувач має на своєму смартфоні спеціальний додаток (поки є лише для Android, IOS версія у розробці) або невеликий пульт, що працює на батарейках.

Маяк споживає 10 кВт/год на рік. Витримує температури від -40°С до +80°С. Має захист від дощу. Радіус його дії — 10-20 м.

Наразі вже є кілька різноманітних моделей маячків: для світлофора (при активації називає колір), для транспорту (називає номер маршруту), для системи «розумне» місто. Повний перелік маяків:

* “Mobilis SB” це система наступного покоління, маяк якої активується тільки у разі потреби (якщо до неї наближається користувач системи – людина із обмеженим зором).
* **Mobilis SB B.02 Звуковий маяк для будівель, що активується через смартфон**
* **Mobilis SA A.01 Програма користувача для активації маяків**
* **Mobilis SA D.01 Пульт-активатор для користувача**
* **Mobilis SA I.01 Пульт-активатор для користувача для зв’язки ключів**
* **Mobilis SB T.01 Звуковий маяк для міського транспорту**
* **Mobilis SB L.01 Звуковий маяк для світлофорів**
* **Mobilis SB E.01 Звуковий маяк для інтеграції в систему «розумне місто» (вбудований)**
* **Mobilis SM M.01 Програма адміністратора для керування маяками**

Інтерфейс додатка адаптовано для користувачів із вадами зору. Програму оптимізовано для мінімального використання ресурсів смартфону, щоб економити заряд акумулятора. Окрім цього, програма вголос інформує про стан роботи системи.

Вартість одного маячка — до 1700 грн.

В підсумку, основні переваги системи такі:

* багато способів використання, за рахунок великої кількості маяків та пультів
* великий радіус дії
* широкий робочий діапазон температур для роботи
* оптимізований для мінімального використання ресурсів смартфону

До недоліків належить:

* висока вартість маячка
* система відтворює зву всім навколо, а не лише незрячій особі
* багато різних пультів, в залежності від застосування маячка

## **Принцип роботи Bluetooth давача**

Працюючи на частоті 2.4 [ГГц](https://uk.wikipedia.org/wiki/%D0%93%D0%B5%D1%80%D1%86), давач дає змогу встановлювати зв'язок у межах 10 або 100 [метрів](https://uk.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D1%80). Різниця у відстані, безумовно, велика, однак з'єднання в межах 10 [метрів](https://uk.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D1%80) дає змогу зберегти низьке енергоспоживання, компактний розмір і досить невисоку вартість компонентів. Враховуючи, що такої відстані цілком достатньо, а для невеликих приміщень, достатньо і кількох метрів, такий спосіб взаємодії є оптимальним. Так, малопотужний передавач споживає всього 0.3 [мА](https://uk.wikipedia.org/wiki/%D0%90%D0%BC%D0%BF%D0%B5%D1%80) в режимі standby і в середньому 30 [мА](https://uk.wikipedia.org/wiki/%D0%90%D0%BC%D0%BF%D0%B5%D1%80) під час обміну інформацією. У стандарті Bluetooth передбачене [шифрування](https://uk.wikipedia.org/wiki/%D0%A8%D0%B8%D1%84%D1%80%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F) даних, що передаються з використанням ключа ефективної довжини від 8 до 128 [біт](https://uk.wikipedia.org/wiki/%D0%91%D1%96%D1%82) і можливістю вибору односторонньої або двосторонньої [автентифікації](https://uk.wikipedia.org/wiki/%D0%90%D0%B2%D1%82%D0%B5%D0%BD%D1%82%D0%B8%D1%84%D1%96%D0%BA%D0%B0%D1%86%D1%96%D1%8F). Додатково, до шифрування на рівні протоколу, може бути використано шифрування на програмному рівні. Так, як в мобільній системі шифрування не використовуватиметься та відсутня автентифікація, то ці алгоритми не застосовувались у системі ні а рівні протоколу, ні на програмному рівні.

У мобільній системі при розробці використовувався Bluetooth модуль ESP32, документація на який подана у [9]. Смартфон користувача не здійснюватиме обмін даними з давачем та не створюватиметься пара. З великою ймовірністю смартфон користувача не з`єднуватиметься з давачем повторно або на постійній основі, а тому не доцільно зберігати давач в смартфоні, особливо, якщо таких давачів розміщено навколо користувача у великій кількості.

За замовчуванням при «опитуванні» смартфоном пристроїв навколо, Bluetooth давачі, що відгукуються за опитування, одночасно сповіщають свою MAC адресу, величину сигналу та ім'я. Цих даних достатньо, щоб порівняти величини сигналів давачів між собою, а за отриманою адресою найближчого здійснити запит до сервера та отримати відповідний текст.

## **Апаратне забезпечення давача**

**Модуль розробника LuaNode32 побудований на мікромодулі ESP-WROOM-32 - новому мініатюрному високопродуктивному модулі з поєднання у собі Wi-Fi + BT + BLE від компанії Espressif, призначеним для широкого спектра застосувань, починаючи від мережевих давачів до найскладніших програм, наприклад, таких як кодування голосу, потокова передача музики та MP3 кодування тощо. На модулі зібрана вся необхідна мінімальна периферія, достатня для швидкого і комфортного старту роботи з ESP-WROOM-32. Він виконаний на базі популярного двоядерного чіпсета ESP32, із змінною тактовою частотою від 80 МГц до 240 МГц, можливістю індивідуального управління і живлення.**

**Модуль розроблений для переносної і автономної електроніки та додатків інтернет-речей, виконаний в мініатюрному корпусі 25,5 мм x 18 мм, має на борту Flash пам'ять, кварц 40 МГц і PCB антену, що забезпечує відмінні RF характеристики.**

**ESP-WROOM-32 має багату периферію, що включає в себе такі інтерфейси як UART, SPI, I²C, I²S, роз'єм для SD карти, інфрачервоний порт, інтерфейс для підключення ємнісної сенсорної панелі.**

**Однією з особливостей модуля є наднизьким споживання і гнучкий вибір «сплячих» режимів, що дозволяють отримати цифри до 20мкА (deep sleep mode).**

**Модуль підтримує весь стек протоколів стандартів WiFi 802.11n і BT4.2, забезпечуючи даний функціонал через інтерфейси SPI/SDIO або I²C/UART.**

**Відмінні особливості:**

* Повне програмування через USB
* Швидкість Wi-Fi: 802.11 b/g/n до 150 Мбіт/с
* Мінімальна чутливість -98 dBm
* Широкий діапазон робочих температур: -40°C…+125°C
* Енергоспоживання до 20мкА (deep sleep mode).
* Безпровідне оновлення.

**Технічні характеристики:**

* Напруга живлення: 5В
* Частотний діапазон: ГГц 2.4 ~ 2.5
* Bluetooth протоколи: v4.2 BR/EDR і BLE specification
* Передатчик: Class-1, class-2 и class-3 AFH
* Аудіо: CVSD и SBC
* Апаратні засоби та інтерфейси: SD, UART, SPI, SDIO, I²C, LED PWM, Motor PWM, I²S, I²C, IR
* Живлення мікромодуля: В 2.2 ~ 3.6
* Діапазон робочих температур: -40°C ~ +85°C
* Відстань між контактними ніжками: 25.5 мм

З більш детальними даними про апаратне забезпечення давача можна дізнатись у офіційній документації.

## **Алгоритм визначення величини сигналу Bluetooth давача**

## **RSSI**

Показник рівня сигналу, RSSI (англ. Received signal strength indicator) - повна потужність прийнятого приймачем сигналу. Вимірюється приймачем за логарифмічною шкалою в дБм (dBm, децибел щодо 1 мілівата).

Як правило (зокрема, в стільникових телефонах і інших GSM-пристроях), потужність сигналу вимірюється після перетворення його частоти з основної в проміжну, але до посилення. У пристроях, що не перетворюють частоту сигнал з основної в проміжну, потужність сигналу вимірюється на основній частоті .

Значення RSSI погано корелює з якістю сигналу, але може використовуватися для його приблизної оцінки. Більш точну оцінку можна отримати за допомогою параметра індикатор якості сигналу.

Для пристроїв, що працюють за стандартами Wi-Fi і Bluetooth 4.0, RSSI є єдиним параметром, що дозволяє виміряти відстань від пристрою до базової станції або маяка. Рівняння для обчислення відстані (за межами ближньої зони передавача) має такий вигляд:

де:

*d* - відстань від пристрою до передавача, м;

*d0* - відстань від пристрою до точки, на якій виконувалось вимірювання потужності сигналу *P0* пристрою, м (вибрана одинична (калібрована) відстань, наприклад, 1 м) ;

*lg* - десятковий логарифм;

*P0* - потужність сигналу пристрою, виміряна на одиничному відстані *d0* від пристрою, dBm;

n - коефіцієнт втрат потужності сигналу при розповсюдженні в середовищі, безрозмірна величина (для повітря n = 2; збільшується при наявності перешкод);

*Pd* - RSSI, dBm.

Дане рівняння випливає з формули передачі Фріїса для поширення радіосигналу в вільному просторі.

Оскільки описаний вище алгоритм отримання величини сигналу є єдиним для Wifi та Bluetooth пристроїв, він використовуватиметься як основний. Для порівняння, як спосіб визначити рівень сигналу розглядався CINR (англ. Carrier to Interference + Noise Ratio) - відношення рівня сигналу до рівня шуму, існує синонім SINR (англ. Signal Interference + Noise Ratio). Так як він використовується в телекомунікаціях для визначення якості сигналу, наприклад: GSM, CDMA, WiMAX, то був відкинутий через несумісність з архітектурою давача.

## **Висновки до розділу 1**

На основі проведеного аналізу RFID міток та Bluetooth модулів різних версій можна сказати, що перевагу отримують Bluetooth модулі. Причина вибору полягає у широкому розповсюдженні та простоті використання модулів, а саме їх поєднання зі смартфонами для обміну даними. З поміж RFID та Bluetooth, останній є більш популярний у використанні, володіє більшим набором документації, більш надійний та дає можливість здійснити обмін даними на велику відстань. Для обраного варіанту легше здійснити аналіз та потужність сигналу модуля для подальшого вибору модуля для підключення.

Було проаналізовано принципи розміщення пристроїв в приміщенні та оцінено кращу топологію, враховано проаналізовані топології у [10]. Оцінено варіанти отримання величини сигналу з [11] та локалізації пристрою з [12]. Так як в кінцевому випадку було обрано Bluetooth, то доцільним було проаналізувати його версії та дослідити його поведінку при визначенні величини сигналу [13]. Тематиці Bluetooth та методам отримання величини сигналу та локалізації пристрою було надано багато уваги, так як це найбільш популярний пристрій для передачі даних, та підтримується майже всіма пристроями у сьогоденні.

# **РОЗДІЛ 2. АРХІТЕТКУРА МОБІЛЬНОЇ СИСТЕМИ**

## **Аналіз моб платформ. та технологій для розробки мобільної системи**

При розробці мобільної системи в якості операційної системи для розробки було обрано Android. Дана операційна система показала себе як більш лояльна до розробників, оскільки є відкритою. Обліковий запис розробника IOS платний, а операційна система Android взагалі його не потребує.

В якості давача для взаємодії з смартфоном у Розділі 1 розглядався RFID та Bluetooth. Останній показав себе краще, оскільки присутній у всіх моделях смартфонів на базі операційної системи Android за останні роки. RFID, на жаль, не має необхідних засобів визначення величини сигналу чи відстані до нього, не підтримується достатньою кількістю смартфонів, щоб скласти конкуренцію Bluetooth. У RFID радіочастотне розпізнавання здійснюється за допомогою закріплених за об'єктом спеціальних міток, що несуть ідентифікаційну та іншу [інформацію](https://uk.wikipedia.org/wiki/%D0%86%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%86%D1%96%D1%8F). Цей метод вже став основою побудови сучасних безконтактних [інформаційних систем](https://uk.wikipedia.org/wiki/%D0%86%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%86%D1%96%D0%B9%D0%BD%D0%B0_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0).

Bluetooth — [технологія](https://uk.wikipedia.org/wiki/%D0%A2%D0%B5%D1%85%D0%BD%D0%BE%D0%BB%D0%BE%D0%B3%D1%96%D1%8F) бездротового зв'язку, створена у [1998](https://uk.wikipedia.org/wiki/1998) році групою компаній: [Ericsson](https://uk.wikipedia.org/wiki/Ericsson), [IBM](https://uk.wikipedia.org/wiki/IBM), [Intel](https://uk.wikipedia.org/wiki/Intel), [Nokia](https://uk.wikipedia.org/wiki/Nokia), [Toshiba](https://uk.wikipedia.org/wiki/Toshiba).

Сьогодні розробки в області Bluetooth ведуться групою *Bluetooth SIG* ([англ.](https://uk.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D1%96%D0%B9%D1%81%D1%8C%D0%BA%D0%B0_%D0%BC%D0%BE%D0%B2%D0%B0) *Special Interest Group*), до якої входять Lucent, [Microsoft](https://uk.wikipedia.org/wiki/Microsoft) та інші компанії, чия діяльність пов'язана з мережними технологіями. Основне призначення *Bluetooth* — забезпечення економного (з точки зору споживаного [струму](https://uk.wikipedia.org/wiki/%D0%95%D0%BB%D0%B5%D0%BA%D1%82%D1%80%D0%B8%D1%87%D0%BD%D0%B8%D0%B9_%D1%81%D1%82%D1%80%D1%83%D0%BC)) і дешевого радіозв'язку між різноманітними типами електронних пристроїв, таких як мобільні телефони та аксесуари до них, портативні та настільні комп'ютери, принтери та інші. Причому, велике значення приділяється компактності електронних компонентів, що дає можливість застосовувати Bluetooth у малогабаритних пристроях розміром з наручний [годинник](https://uk.wikipedia.org/wiki/%D0%93%D0%BE%D0%B4%D0%B8%D0%BD%D0%BD%D0%B8%D0%BA).

Bluetooth дозволяє з допомогою смартфону визначати його величину сигналу, тобто RSSI. Яскравим прикладом використання Bluetooth в приміщеннях разом з RSSI є Синтра від SAMSUNG.

При контролі місця розташування виробів з металу ефективнішими виявилися маячки Bluetooth. Маячок поміщається в поліетиленовий пакет, потім кріпиться до вантажного ящика. Ідентифікатор Bluetooth маячка внесений до програми Синтра і збережений за допомогою хмарних технологій.

Другим елемент програми Синтра є Samsung з операційною системою Android від OC 5.1. Пристрої Android прикріплені до стін приміщення на території більше 1300 м2. Вони захоплюють сигнал, визначають значення RSSI, а потім відсилають дані в Синтра систему. На практиці працювати з програмою Синтра нескладно. Для пошуку потрібної деталі у мобільному пристрої робочий вносить потрібний ідентифікатор, після чого приступає до пошуку за принципом дії датчика Гейгера. Планшет визначає відстань до об'єкту, оцінюючи величину RSSI, показує працівнику, як далеко він знаходиться від цілі.

Другий спосіб використання системи Синтра - пошук за допомогою Wi-Fi. При цьому варіант використання портативних та стаціонарних смартфонів взаємодіють з маячком за програмою Синтра, в якій зберігається повна інформація про місце розташування продукту.

В якості хмарного сервісу було обрано Azure. Дане сховище володіє дуже великою кількістю різноманітних технологій, в тому числі хостинг веб застосунків та баз даних, що і стало ключовою причиною для вибору даного сховища. В якості середовища розробки було обрано Visual Studio та мову C# відповідно, так як і мова програмування, середовище та Azure від компанії Microsoft, то у поєднанні технологій немає жодних проблем.

## **Інструменти розробки мобільної системи**

Для програмування Bluetooth давача використовувалось середовище Arduino IDE.

У Android смартфоні для створення застосунку було використано середовище Android Studio, а для з’єднання з сервером було використано бібліотеку Retrofit [14] - типобезпечний HTTP-клієнт для Android і Java. Він є незамінним інструментом для роботи з API в клієнт-серверних додатках. Передача даних здійснюється за допомогою HTTP запитів. Формат передачі даних – JSON.

Для побудови ASP.NET MVC Web API використовувалось середовище Microsoft Visual Studio. Для розміщення серверу та веб застосунку на хмарі було використано Azure сховище. Це ж сховище було використано для розміщення SQL бази даних. Базу даних було розроблено у середовищі SQL Management Studio.

Для написання веб клієнта було використано мову розмітки HTML, Java Script та ajax для обміну даними між сервером та веб клієнтом та CSS для створення стилів на клієнтській частині. Для коректного відображення веб клієнту на різних платформах та розширеннях було використано бібліотеку Bootstrap.

## **Особливості використання компонентів мобільної системи**

Для з'єднання бази даних до проекту використовувався SQL Server.

При першому запуску буде створено дані таблиці, а вже при наступних запусках даний код не буде виконаний, що і передбачається логікою програми.

Так як система соціального характеру та безкоштовна, не міститиме персональних даних, немає сенсу застосовувати до неї шифрування чи інші засоби захисту.

Для зберігання даних можна обійтись лише однією таблицею у базі даних, а початкова її версія виглядатиме так:

* Назва: Sensors\_Data.
* Поля:
  + Sensor\_ID – унікальний ідентифікатор сенсора.
  + SensorValue – поле для зберігання текстових даних сенсора.

## **Середовище функціонування та вимоги інтерфейсів до системи**

**Середовище функціонування:**

**Апаратні вимоги:**

* 10 Mb вільного простору на диску;
* Частота процесора 1 GHz;
* Оперативна пам’ять 512 Mb.

**Системні вимоги:**

* Операційна система Android 4.4.4 або вище;
* Доступ до мережі.

**Вимоги зовнішніх інтерфейсів:**

**Користувацькі інтерфейси:**

* Інтерфейс користувача має відповідати вимогам «Android Human Interface Guidelines».

Програмні інтерфейси:

* Android SDK;
* Бібліотека Retrofit;
* комп’ютер під керуванням ОС Windows 7 чи вище;
* USB кабель тип B;
* середовище розробки Arduino IDE, Android Studio та MS Visual Studio*.*

**Апаратні інтерфейси:**

* Bluetooth давач (використаний при розробці - ESP32);

**Інші нефункціональні вимоги:**

**Вимоги продуктивності:**

Передача та отримання даних залежатиме від об’єму отриманих/переданих даних та від швидкості передачі даних.

**Вимоги безпеки:**

У зв’язку з соціальною тематикою проекту, відсутністю секретних даних, даних, що вимагають захисту, протоколи безпеки не задіюватимуться.

**Інші вимоги**

Дозволи необхідні застосунку для роботи:

* Доступ до файлової системи
* Переглядати з’єднання wifi
* Переглядати мережеві з’єднання
* Доступ до мережі

## **Принцип взаємодії Bluetooth давача з смартфоном**

Деталі взаємодії та застосування компонентів системи [15] можна побачити на діаграмі послідовностей у додатку Б.

Смартфон на базі операційної системи Android контактуватиме зі всіма пристроями навколо [16] (Рис. 2.1). Той Bluetooth пристрій, сигнал RSSI котрого найбільший буде цільовим для з’єднання [17]. Він повертатиме смартфону унікальний ключ, за яким в свою чергу здійснюватимуться подальші операції.

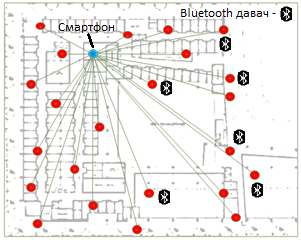


Рис. 2.1. Схема взаємодії смартфону та давачів

Маючи ключ від давача, можемо створити з’єднання з сервером, та отримати від нього текстові дані для озвучування користувачу.

Сервер у свою чергу мусить звернутись до бази даних, перш ніж повернути якийсь результат користувачу.

Існує клієнт, який у свою чергу зможе додавати дані у базу з допомогою сервера.

Все що залишається це встановити відповідний ключ на давач та розмістити його у приміщенні. Всю іншу роботи зробить смартфон.

## **Взаємодія смартфону та серверу мобільної системи**

Обмін даними між сервером та мобільною системою відбувається за допомогою http протоколу та запитів у мережі.

Сервер працює за такими типами запитів: GET та POST.

Смартфон клієнт при відправці запитів, отримає відповідь про статус відправки, успіх або невдача. Відповідно до статусу, опрацьовується певна логіка.

Для прикладу, при відправці запиту а отримання тексту за ключем давача, при успішному відправленні запиту та отриманні відповіді відтворюється отриманий текст, при помилці відбувається відтворення відповідного повідомлення.

Інтерфейс з запитом на смартфоні показано на Рис. 2.2.

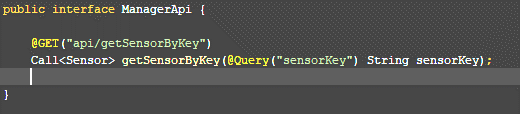


Рис. 2.2. Інтерфейс з запитом до серверу

## **Взаємодія серверу з базою даних та веб клієнтом у мобільній системі**

Сервер, розміщений у хмарному сховищі, взаємодіє з базою даних для збереження даних та їх отримання. З’єднання відбувається за допомогою файла конфігурацій Web.config, в якому знаходяться дані для з’єднання з базою даних.

З’єднання з веб клієнтом відбувається з допомогою згаданих у обміні давача і смартфону http запитів. У веб клієнті кількість таких запитів більша. Підтримувані запити можна переглянути в розділі з документацією на веб клієнті та на рис 2.3.



Рис. 2.3. Підтримувані запити з веб клієнту до серверу

## **Проектування графічного інтерфейсу користувача**

Враховуючи вимоги до програмного продукту, було розроблено макет інтерфейсу системи, який включає всі потрібні елементи для функціонування системи (Рис. 2.4.). Він ключає в себе кращі сторони і практики, які були використані в продуктах аналогах і виправлені недоліки.

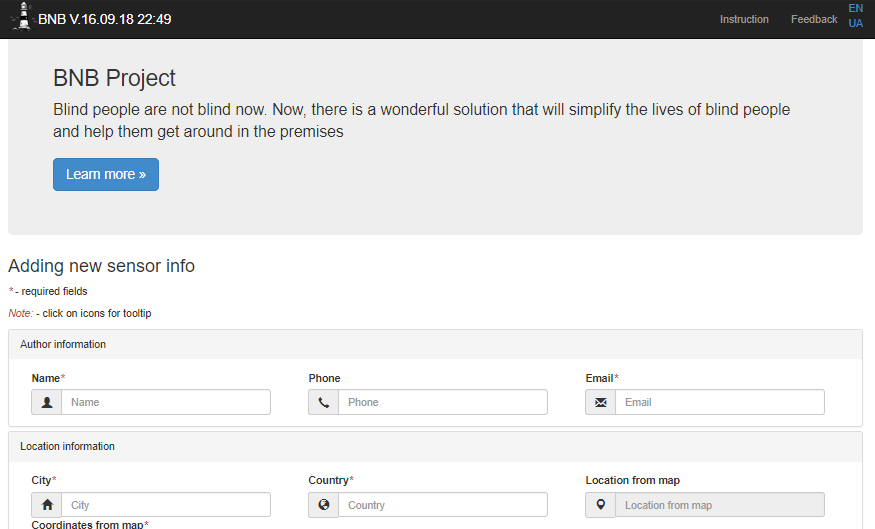


Рис. 2.4. Макет для функціонування системи

Користувачу повинна бути доступна робоча область, яка дозволятиме бачити на карті існуючі маркери (Рис. 2.5.).

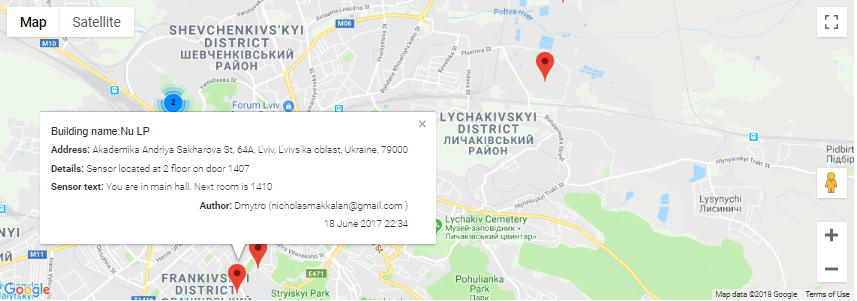


Рис. 2.5. Карта з існуючими маркерами та інформацією про сенсори

Повинна бути надана можливість додавати нові маркери, тобто дані про нові сенсори (Рис. 2.6.).

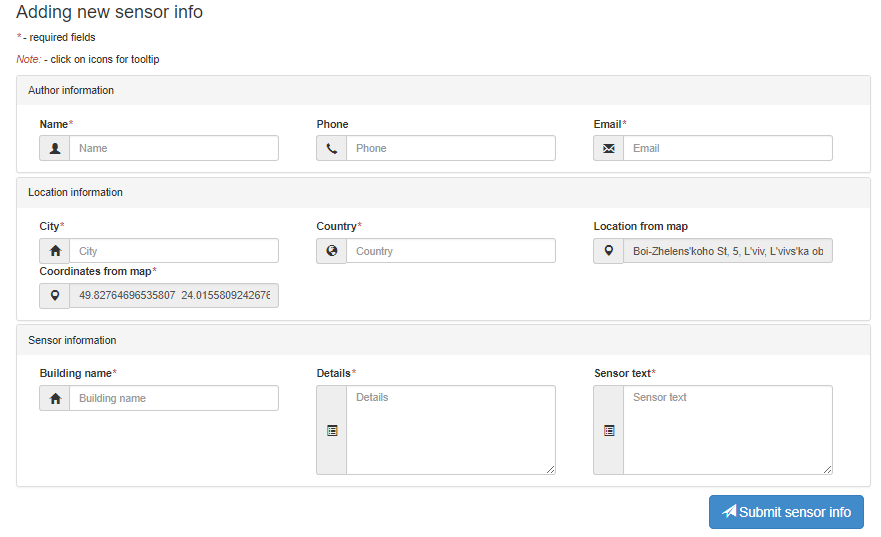


Рис. 2.6. Приклад додавання інформації про новий сенсор

Інструкція з використання ключа для сенсора існуватиме в окремому розділі. Для кращого функціонування, удосконалення системи було зроблено сторінку відгуків. (Рис. 2.7.).

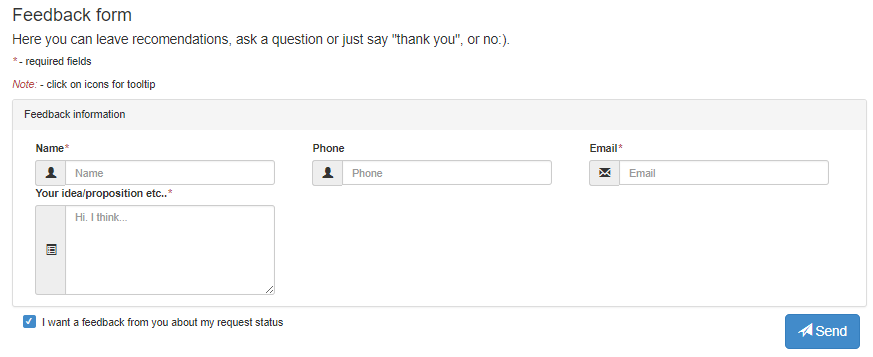


Рис. 2.7. Сторінка відгуку користувача

## [**Проектування архітектури мобільної системи**](#_Toc525168679)

## **Проектування основних компонент і модулів мобільної системи**

Оскільки система мобільна, то основний обмін даними здійснюватиметься між мобільними пристроями та сервером. Тому основними складовими будуть сервер та мобільний застосунок. До складових системи відноситимуться веб інтерфейс, сенсор та база даних.

Сервер мобільної системи слугуватиме логічним центром. Він прийматиме дані з новими давачами, повертатиме існуючі дані про давачі та дані за фільтрами, оброблятиме відгуки користувачів.

Веб інтерфейс дозволятиме зрячим користувачам додавати нові давачі та відгуки про роботу системи для її подальшого удосконалення. Користувачі зможуть побачити на карті вже розміщені давачі, скористатись відкритим API для побудови власних систем.

База даних міститиме всі дані з веб інтерфейсу та дані з мобільного пристрою, наприклад логування, дані про збої і тд.

Сенсор міститиме унікальний ключ за яким завантажуватиметься текст для озвучування незрячим користувачам.

Мобільний застосунок слугуватиме засобом для завантаження даних та озвучування його користувачу.

## **Проектування класів системи**

Для реалізації функціональних можливостей системи було розроблено гнучку ієрархію класів. Вона дозволяє легко модифікувати систему додаючи новий функціонал і модулі.

Основним класом системи являється ValuesController та HomeController клас. Даний клас дозволяє обробляти всі запити отримані зі смартфону чи веб інтерфейсу. Методи містять опис для оформлення документації для інших розробників.

WebApiConfig файл конфігурації, у ньому обрано формат json як основний формат передачі даних між клієнтом та сервером.

У проекті створено AccountController для легшої подальшої реалізації авторизації та автентифікації користувачів, якщо така потреба виникне.

## **Висновки до розділу 2**

Серед популярних на сьогоднішній день мобільних операційних систем за розподілом по доступності, складності розробки та вартості, Android має переваги, тому обраний як цільова платформа для розробки. Запропоновані технології для розробки та порядок використання компонентів у мобільній системі сприятиме їх гнучкості при подальшій розробці та взаємодії між компонентами. Популярність технологій розробки та архітектура, дозволяє розміщати систему на хмарі та успішно використовувати її на практиці.

# **РОЗДІЛ 3. РЕАЛІЗАЦІЯ МОБІЛЬНОЇ СИСТЕМИ**

## **Архітектура бази даних та проектування серверної частини мобільної системи**

База даних мобільної системи є головним сховищем, та розміщається на Microsoft Azure. Структура бази примітивно проста, адже не містить даних про користувачів, паролів чи інших особистих даних.

Головна інформація, яку необхідно зберігати перш за все, це дані про сам давач. В якості унікального ключа для давача було використано його MAC адресу. Наступним кроком додається інформація про локацію та автора. Автор може не вказувати інформацію про себе, а саме пошту, ім`я та телефон, так як система не шифрує та не захищає ці дані. В такому випадку користувач не отримає зворотній дзвінок на відгук або електронного листа з інструкціями.

Схема бази даних мобільної системи зображена на Рис. 3.1.

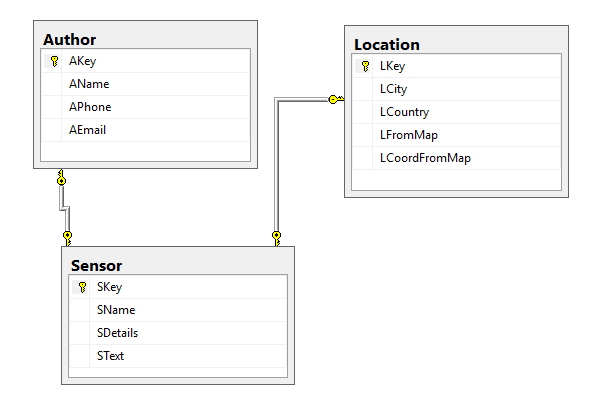


Рис. 3.1. Схема бази даних мобільної системи

Як уже описувалось раніше автору (таблиця Author) достатньо вказати лише ім`я, телефон та пошту (на Рис. 3.1. поля AName, APhone та AEmail відповідно). У таблиці, що містить дані про локацію Bluetooth давача (таблиця Location) містяться дані про місто, країну, повну адресу з карти та координати з карти (на Рис. 3.1. поля LCity, LCountry, LFromMap та LCoordFromMap відповідно). Таблиця, що містить дані про давачі (таблиця Sensor), зберігає в собі ім`я давача, деталі про нього, наприклад дані про розміщення та текст, який відтворюватиметься при взаємодії з давачем (на Рис. 3.1. поля SName, SDetails та SText).

У базу були додані додаткові поля, що містять дату додавання давача та дату останньої зміни інформації про давач, хоча відповідний функціонал, а саме модифікацію даних ще не розроблено, така конфігурація буде корисною при подальшій розробці системи.

При отриманні даних про новий давач, дані додаються в такому порядку:

1. Першою збережеться інформація про сам давач.
2. Для вже існуючого давача з закріпленим за ним ключем збережеться його локація.
3. Останньою збережеться інформація про автора.

Варто зауважити, що операції виконуються транзакцією, якщо в ході збереження порушиться цілісність або відбудеться збій, додавання давача не відбудеться. Користувач отримає відповідне повідомлення, про це в розділі з результатами розробки.

Серверна частина мобільної системи розроблена на мові C# [18] у середовищі Microsoft Visual Studio. Сервер розроблений як ASP.NET MVC Web API проект. Структура проекту представлена на Рис.3.2.

В якості файлу конфігурацій проекту слугує файл Web.Config. Він містить основні конфігурації, в тому числі для взаємодії з базою даних.

Серед багатьох файлів проекту варто звернути увагу на такі папки, як App\_Data. Вона містить в собі \*.xml файл для відображення документації з методами, підтримуваними сервером. App\_Start, папка з файлами конфігурації серверу, фільтрами, конфігураціями формату передачі даних та інше. Основна логіка, що відповідає за формування, збереження та відтворення документації відкритого API користувачу міститься у папці Areas.

Логіка, що відповідає на запити з смартфон клієнту, веб клієнту та сторонніх платформ обробляється контролерами у папці Controllers. При потребі у додаванні методів захисту, авторизації та автентифікації у майбутньому, додано AccountController. Він дозволяє користувачам здійснювати реєстрацію, входити у свій обліковий запис, змінювати пароль і тд. Моделі, що передаються між сервером та клієнтами, та між сервером і базою даних, містяться у папці Models.

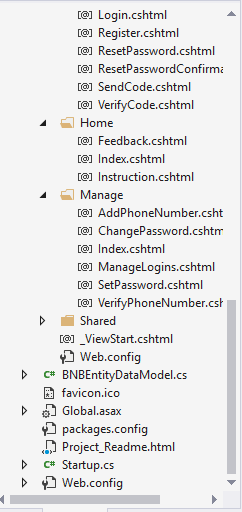
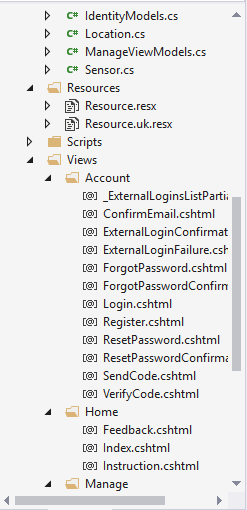
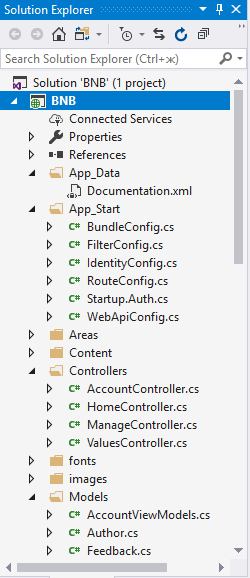


Рис. 3.2. Структура проекту серверу

На веб клієнті розміщено 2 кнопки з перемиканням мови інтерфейсу. Ресурси для цього функціоналу розміщені у папці Resources. Кількість підтримуваних мов відповідає кількості файлів у цій папці. Кожна нова мова отримує назву за шаблоном Resource.<шифр мови>.resx.

Всі вікна/сторінки веб клієнту розподілені по папках у головній папці з назвою Views. Назви Файлів, відповідають назвам сторінок та розподілені за категоріями по папках. Останнім файлом, вартим уваги є packages.json, який містить інформацію про пакети, що використовуються у проекті та їх версії. Пакети завантажуються у випадку їх відсутності чи розгортання серверу у новому середовищі чи а новому пристрої.

Ключові фрагменти коду деяких файлів наведено у додатку Д.

## **Проектування Веб клієнта для мобільної системи**

Веб клієнт мобільної системи розроблений з допомогою мови розмітки HTML, JS та CSS у середовищі Microsoft Visual Studio.

Клієнт розміщений у цьому ж проекті, що і сервер, що спрощує обмін даними та розробку оскільки все знаходиться в одному рішенні. Структура проекту представлена на Рис. 3.3.

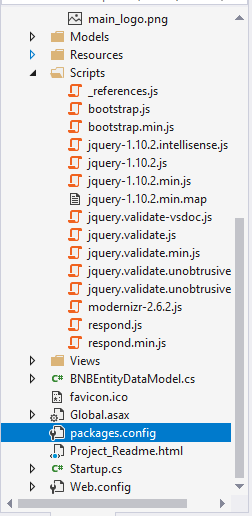
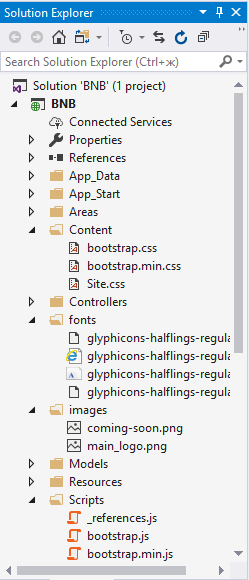


Рис. 3.3. Структура проекту веб клієнта

Для перетворення сторінок під розширення смартфону користувача було використано бубліотеку bootstrap, вона розміщена у папці content та містить в собі набір стилів, що перетворює та правильно розміщує компоненти під пристрій, що відкрив сторінку.

Шрифти для сторінок розміщені у папці fonts. За бажанням, можна додати новий шрифт до проекту.

Зображення, що використовувались у веб клієнті розміщені у папці images. У ній містяться зображення, що відображаються у випаду якщо сторінка ще не розроблена, відсутня та головна іконка.

JavaScript бібліотеки розміщені у папці scripts. З їх допомогою здійснюється взаємодія з картами google, надсилання даних з веб клієнта до серверу та навпаки.

## **Проектування мобільного клієнту для мобільної системи**

Мобільний клієнт розроблено на мові програмування java з використанням середовища Android Studio. Застосунок для смартфону розроблявся на версії операційної системи Android 5.0.1. На цій же версії тестувався. Застосунок може бути успішно встановлений на більш свіжі версії операційної системи без явних перешкод чи проблем в установці та користуванні. Повна структура проекту описана на Рис. 3.4.

Першим та головним файлом у проекті є маніфест застосунку з відповідною назвою – AndroidManifest.xml. Цей файл містить перелік сторінок застосунку та перелік необхідних дозволів, дозвіл на які буде запитано у користувача при установці. До таких дозволів належать такі як:

* Доступ до Bluetooth
* Доступ до інтернету
* Дозвіл на використання Wifi
* Доступ до облікового запису
* Доступ до внутрішнього сховища

Відповідні дозволи необхідні для запиту у користувача дозволу на увімкнення Bluetooth, здійснення запитів до інтернету за даними про давач, збереження налаштувань локально, запуск фонового сервісу та використання wifi у випадку його наявності.

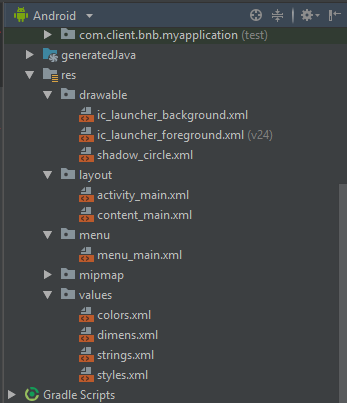
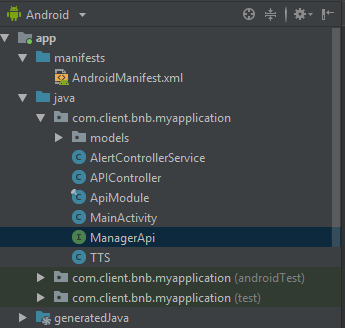


Рис. 3.4. Структура проекту мобільного клієнту

Після опису маніфесту, розміщено вихідний код застосунку. Для моделей створено окрему папку з назвою models. Вона містить такі ж моделі, що і сервер, для обміну даними та їх перетворення та манімулювання ними у коді.

AlertControllerService, призначений для створення та фонової роботи сервісу, який запускається після відкриття застосунку та до його завершення користувачем.

ApiController містить методи для перевірки з’єднання з інтернетом, відправки запитів у інтернет та інші методи, що необхідні для взаємодії з сервером.

ApiMudule призначений для створення з’єднання з сервером та в подальшому взаємодії з ним.

Відповідальним за головне вікно застосунку є MainActivity. Він є вхідною точкою для застосунку, відкривається першим після запуску застосунку користувачем.

Усі запити що відправляються до серверу містяться у ManagerApi інтерфейсі.

За синтез мовлення відповідає клас TTS.

Що стосується іконок, зображень, стрічок кольорів та опису вікон застосунку знаходиться у папці res. Згідно Android material design, у застсунку мають бути присутні три базові кольори, які знаходяться за шляхом res/values/ colors.xml. Там знаходяться стилі та стрічки для перекладу інтерфейсу на різні мови у випадку виникнення такої потреби.

Середовище дозволяє компілювати проект безпосередньо на Android смартфон у випадку його наявності, увімкнувши режим розробника та підєднавши смартфон до компютера. У випадку відсутносі смартфону, існує можливість використання симулятора.

## **Реалізація взаємодії Bluetooth давача з смартфоном**

Для взаємодії Bluetooth давача з смартфоном незрячого користувача, перш за все необхідно увімкнути Bluetooth на смартфоні та сам давач. На давачі Bluetooth буде увімкнено автоматично, одразу після подачі живлення на нього. Авач одразу виступаючи у ролі веденого пристрою буде оповіщати пристрої навколо про свою присутність. Разом з сигналом, надсилатиметься ім’я давача та його мак адреса.

Android смартфон після увімкнення Bluetooth та запуску застосунку користувачем розпочне пошук Bluetooth пристроїв навколо. Разом з даними давача буде отримано силу сигналу знайденого давача. При пошуку пристроїв, ті які належать до мобільної системи буду додані до списку. Належність давача до списку визначається ім’ям проекту. Після завершення пошуку, у списку пристроїв визначається давач з максимальною силою сигналу. Наступним кроком, з давача з найбільшим сигналом, тобто розташованим найближче, отримується MAC адреса, яка передається одразу ж при розпізнанні пристроїв. За цією адресою отримується текст для озвучування, шляхом надсилання до серверу http запиту з MAC адресом давача.

Отриманий текст озвучується незрячому користувачу, а програма повторює описані кроки знову.

Цей підхід має ряд переваг та недоліків. Наприклад до переваг відноситься швидкість обміну. Смартфон не створює пару з давачем та не обмінюється з ним даними. Це є одночасно і недоліком, оскільки смартфон отримує дані з інтернету з адресою давача, а не здійснює обмін даними безпосередньо з давачем. Код взаємодії давача з смартфоном можна переглянути у додатку Г.

## **Результати розробки мобільної системи для орієнтації незрячих осіб в приміщеннях.**

Після опису всіх компонентів мобільної системи, її можна об’єднати в одне ціле та продемонструвати описаний, реалізований функціонал. Загально схему елементів можна побачити у додатку А. Починаючи роботу з мобільною системою, користувач в першу чергу додає інформацію про Bluetooth давач та його MAC адресу. Це продемонстровано на Рис. 3.5. Після додавання та збереження давача веб клієнт дає можливість користувачу переглянути карту з розміщеними давачами (Рис. 3.6.), в тому числі про давач, який він щойно додав. Для цього треба оновити сторінку після додавання інформації про Bluetooth давач.

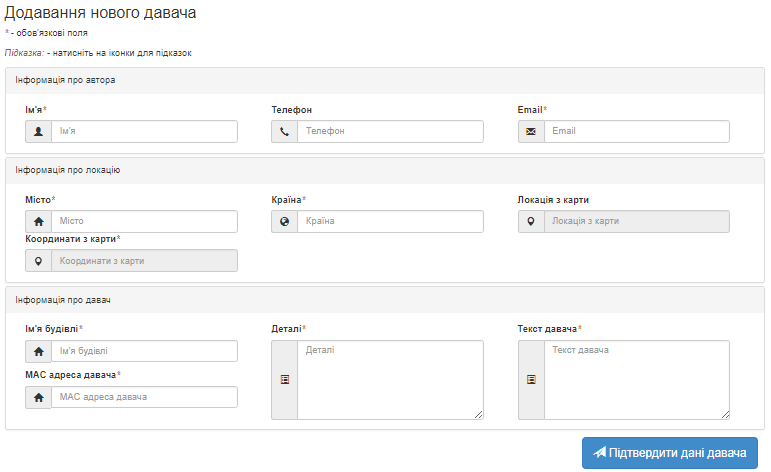


Рис. 3.5. Інтерфейс додавання нового давача

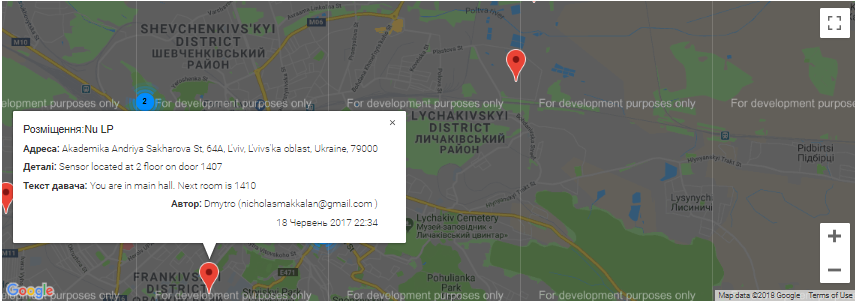


Рис. 3.6. Карта з існуючими давачами та описом

Маючи Bluetooth давач, необхідно просто увімкнути його та розмістити у описаному на веб клієнті місці. Для уникнення надлишкових кроків з встановленням середовища для давача та його програмування, залишаємо його існуючу назву, тобто “ESP32”. Фото давача, який було використано при розробці, продемонстровано на Рис. 3.7. Для роботу підійде будь який інший Bluetooth давач.



Рис. 3.7. Вигляд Bluetooth давача.

Смартфон застосунок запущений незрячим користувачем, виконає усю іншу роботу. Інтерфейс простий та зрозумілий (Рис. 3.8.), а сам застосунок може бути закритий після запуску, фоновий сервіс продовжить роботу. Для його завершення необхідно запустити застосунок та натиснути кнопку зупинки, або ж вбити процес в диспетчері застосунків операційної системи.

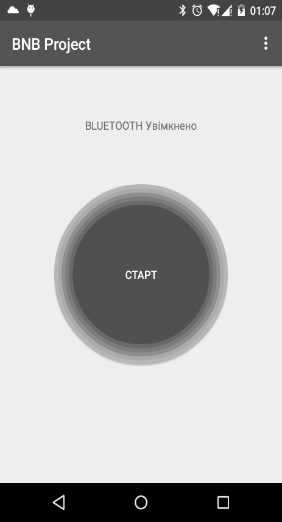


Рис. 3.8. Інтерфейс мобільного застосунку

Вся взаємодія здійснюється з сервером, який у свою чергу взаємодіє з базою даних. І сервер і база даних вже описані. Інструкції щодо розміщення серверу на хмарі Azure та розміщення на ній бази даних існує в інтернеті у вигляді відео уроків та іншої документації.

## **Висновки до розділу 3**

Архітектура бази даних мобільної системи виявилась доволі гнучкою та ефективною. Структури бази достатньо, щоб надати користувачу достатньо даних про Bluetooth давачі та успішно зберігати їх у хмарі.

Сервер показав себе як зручний застосунок за рахунок популярності та легкості розміщення на хмарі та справно працює з отриманням, збереженням та поверненням результату клієнту.

Мобільний застосунок успішно взаємодіє з сервером та озвучує текст користувачеві, Необхідно задати текст однією мовою, так як відтворення на двох мовах не завжди коректне. Необхідно в подальшому зберігати дані локально для збереження трафіку.

Веб клієнт інформативно демонструє користувачеві у вигляді карти наявні давачі з описом. Та містить форму для відгуку, щоб ефективно удосконалювати системи, ґрунтуючись на відгуках користувачів.

Результати роботи у підсумку подані у вигляді зображень. Сервер розміщений на хмарі та успішно працює. З інструкцією користувача можна ознайомитись у додатку В.

# **РОЗДІЛ 4. ПРОВЕДЕННЯ ДОСЛІДЖЕННЯ МОБІЛЬНОЇ СИСТЕМИ ТА АНАЛІЗ ОТРИМАНИХ ДАНИХ**

## **Опис процесу дослідження мобільної системи**

Дослідження мобільної системи здійснювалось окремо для кожної складової мобільної системи.

Оскільки смартфон клієнт та веб клієнт працюють за протоколом http, дослідження швидкості здійснювалось шляхом надсилання http запитів до серверу. Для тестування було використано 2 розширення для браузера. Це RestMan та Rested та веб браузер Opera. Дослідження у такий спосіб дуже просте та надає всю необхідну інформацію, наприклад: час виконання запиту, статус виконання, формат запиту та відповідь від серверу. Задля уникнення ручного формування моделі для надсилання, для додавання нового давача та відгуку заповнювалась форма на веб клієнтів та натискались відповідні кнопки надсилання та підтвердження. Швидкість опрацювання запиту та отримання відповіді відслідковувалась у Network Monitor меню у браузері.

Тестування з’єднання з базою даних виконувати не доцільно, оскільки база даних розміщена на одному сховищі з сервером там залежить від обраної та оплаченої користувачем потужності віддаленої машини.

Дослідження обміну даними з Bluetooth давачем здійснювалось шляхом вимірювання часу пошуку пристрою. Наступим вимірювалась швидкість пошуку пристрою з найбільший рівнем сигналу.

Вимірювалась відстань, за якої давач попадатиме в поле зору смартфон.

Ще одним критерієм була сила сигналу давача, якщо присутня перешкода. Вимірювання проводились на двох давачах для більшої кількості даних та точності вимірювання.

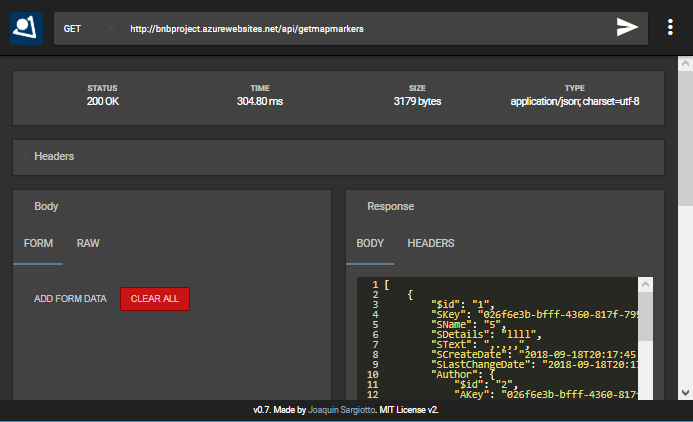
Такі дослідження системи дадуть точне уявлення про прогалини системи, вкажуть на складові системи, які варто удосконалити та допоможуть точніше визначити на які саме аспекти мобільної системи варто звернути увагу.

Таким чином в майбутньому існуватиме можливість незалежно один від одного удосконалювати певний елемент мобільної системи

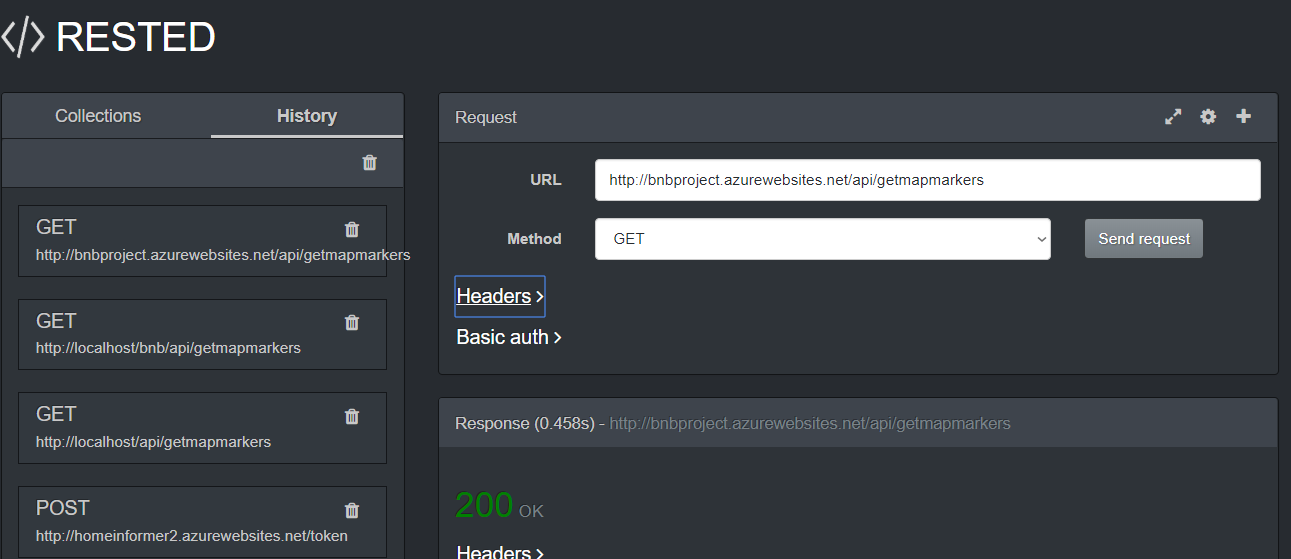
## **Дослідження швидкості обміну даними між компонентами мобільної системи**

Початок роботи здійснюється з додавання нового давача, то ж доцільно першим протестувати обмін даними між веб клієнтом та сервером.

При відкритті сайту, користувачу доступна карта з існуючими давачами. Перший запит відправляється для отримання даних про давачі. Для точнішого результату, в подальшому виконуватимемо тести 7 разів. Тести виконуватимуться двома розширеннями (Рис. 4.1.) для браузера та з допомогою меню розробника (Рис. 4.2.).



а)



б)

Рис. 4.1. Інтерфейс розширень: а) RestMan, б) Rested

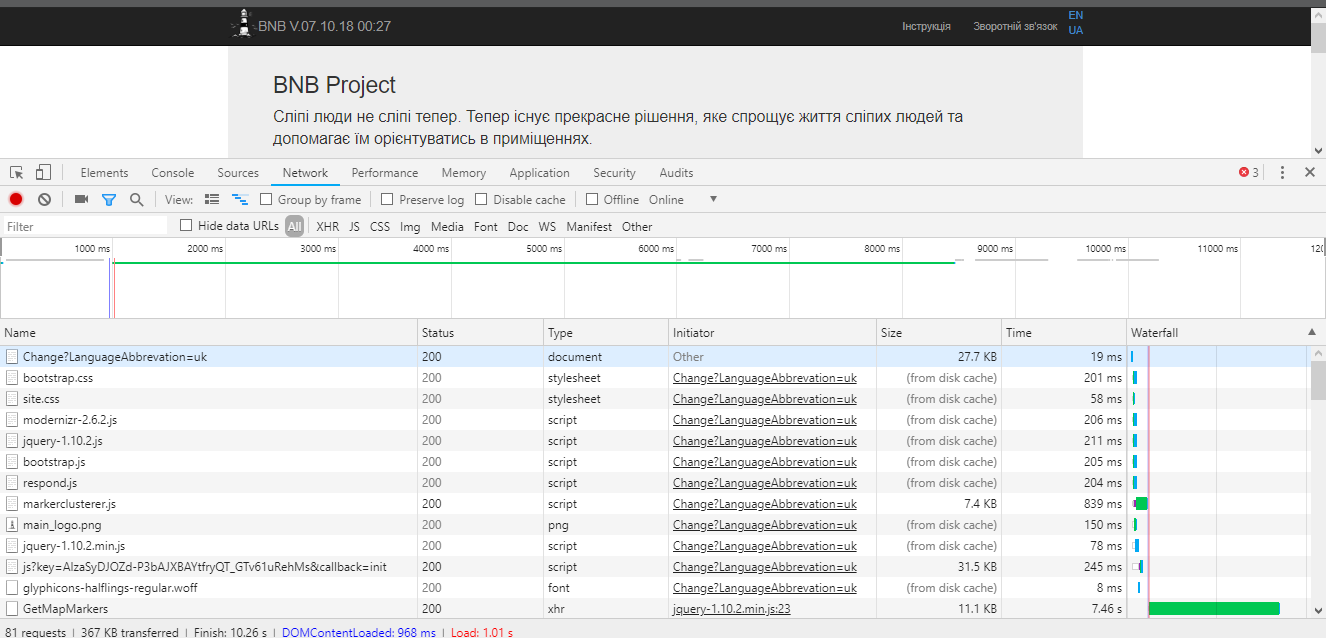


Рис. 4.2. Вигляд меню розробника у браузері

Інформація про застосунки була описана у розділі 4.1. Відправлення даних до серверу продемонстровано у таблиці 4.1. У таблиці описано шлях, за яким здійснювався запит, номер розширення, у випадку відправки даних, назва розширення з меню розробника у браузері та час виконання у мілісекундах.

*Таблиця 4.1.*

**Таблиця запитів до сервера та час їх виконання**

|  |  |  |  |
| --- | --- | --- | --- |
| api/getmapmarkers | | | api/addnewmarker |
| Розши-  рення | 1 | 2 | Network Monitor |
| № | Час,мс | | Час,мс |
| 1 | 1488.10 | 313.0 | 120.0 |
| 2 | 463.60 | 300.0 | 125.0 |
| 3 | 290.10 | 295.0 | 118.0 |
| 4 | 287.10 | 297.0 | 126.0 |
| 5 | 297.40 | 280.0 | 124.0 |
| 6 | 289.50 | 288.0 | 117.0 |
| 7 | 290.60 | 285.0 | 120.0 |
| Середній час | 529,07 | 294 | 121,4 |

В середньому, можна припустити, що час виконання запиту буде такий самий при зверненні до серверу з смартфон клієнту. Дані можуть погіршуватись у випадку низького рівня сигналу мережі або ж за рахунок відстані, бо смартфон клієнт не знаходиться на тому ж комп’ютері, що і веб клієнт.

## **Результати взаємодії смартфону та Bluetooth давача**

Для дослідження взаємодії Bluetooth давачів з смартфоном було використано вже описану версію давача ESP32. Тестування здійснювалось в кілька етапів. Було здійснено кілька видів досліджень, а саме:

* Здатність до роботи на різних відстанях від смартфону
* Зміна сили сигналу в залежності від присутності перешкод та їх виду

В якості перешкод було обрано найбільш ймовірні перешкоди в ході використання давача незрячим давачем. До таки перешкод було віднесено:

* Звичайні двері, матеріал – дерево, товщина – близько 2см
* Металеві двері, товщина – близько 5-10мм
* Цегляна стіна, товщина еквівалентна 25-30см
* Скло

В загальному вплив певних матеріалів, отриманих з інтернет джерел [19] можна зобразити у вигляді таблиці (Таблиця 4.2.)

*Таблиця 4.2.*

**Вплив перешкод на сигнал давача**

|  |  |
| --- | --- |
| **Матеріал** | **Рівень створюваних перешкод** |
| Дерево | Низький |
| Синтетичний матеріал | Низький |
| Скло | Низький |
| Вода | Середній |
| Цегла | Середній |
| Мрамор | Середній |
| Гіпс | Високий |
| Бетон | Високий |
| Куленепробивне скло | Високий |
| Метал | Дуже високий |

Перераховані нижче пристрої можуть створювати перешкоди для бездротового зв'язку по Wi-Fi і Bluetooth:

1. Мікрохвильові печі. Якщо поблизу комп'ютера, пристрої Bluetooth або базової станції Wi-Fi працює мікрохвильова піч, вона може генерувати перешкоди.
2. Системи супутникового телебачення. Коаксіальний кабель і роз'єми, які використовуються з деякими типами супутникових тарілок, можуть бути джерелами перешкод, через які можуть виникати радіочастотні перешкоди (так звана витік на високих частотах).
3. Джерела електроенергії. Деякі зовнішні джерела електричної напруги, такі як лінії електропередавач, електрифіковані залізничні колії і силові підстанції, можуть бути джерелами перешкод.
4. Радіотелефони, що працюють в діапазоні 2,4 або 5 ГГц. При прийомі викликів бездротовий радіотелефон, що працює на частоті 2,4 ГГц або 5 ГГц, може створювати перешкоди для бездротових пристроїв або мереж.
5. Бездротові високочастотні передавачі відеосигналу. Бездротові передавачі відеосигналу, що працюють в діапазоні 2,4 або 5 ГГц, можуть бути джерелами перешкод для бездротових пристроїв або мереж.
6. Бездротові динаміки. Бездротові динаміки, що працюють на частоті 2,4 ГГц або 5 ГГц, можуть створювати перешкоди для інших бездротових пристроїв або мереж.
7. Деякі зовнішні монітори і РК-екрани. Деякі монітори створюють гармонійні перешкоди, які особливо помітні в діапазоні між 11 і 14 каналами на частоті 2,4 ГГц. При використанні ноутбука із закритою кришкою і підключеним зовнішнім монітором перешкоди можуть бути досить відчутними. Спробуйте вирішити проблему, змінивши точку доступу. Перейдіть на частоту 5 ГГц або використовуйте один з перших десяти каналів в діапазоні 2,4 ГГц.
8. Кабелі з недостатнім екрануванням. Зовнішні жорсткі диски або інші пристрої з недостатньо екранованими кабелями можуть створювати перешкоди для бездротових пристроїв. Якщо при відключенні або вимкнення пристрою перешкоди зникають, замініть кабелі, за допомогою яких пристрій підключено до комп'ютера.
9. Інші бездротові пристрої. Заважати роботі пристроїв, підключених по мережі Wi-Fi або Bluetooth, можуть і інші бездротові прилади, що працюють в діапазоні 2,4 або 5 ГГц, зокрема мікрохвильові передавачі, бездротові камери, радіоняні і пристрої сусідів, підключені по Wi-Fi.

Як бачимо існує чимало пристроїв та способів створення перешкод для передачі сигналу, тому рівень сигналу може бути не завжди таким, як очікується

Варто зауважити, що на момент тестування, на давачі не здійснювались жодні налаштування. Це виконано задля отримання результатів при заводських налаштуваннях. Припускаємо, що робота з давачем здійснюватиметься після покупки та без жодних конфігурацій. Характеристика давача описана у розділі 1.3.1.

Вимірюючи силу сигналу (RSSI, dBm) на певній відстані, віддалятимемо давач на 0.5м кожного разу. Дослідження проведемо 7 раз. Для більшої точності на кожному проміжку замірятимемо дані 3 рази. Результати наведено у таблиці 4.32.

*Таблиця 4.3.*

**Результати тестування давача з різної відстані**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Відстань від смартфону, м | **№ дослідження** | | | Середнє значення, dBm |
| **1** | **2** | **3** |
| Сила сигналу, dBm | | |
| 1 | 57 | 59 | 58 | 58 |
| 1.5 | 67 | 70 | 68 | 68 |
| 2 | 70 | 65 | 71 | 68,6 |
| 2.5 | 69 | 68 | 69 | 68,6 |
| 3 | 65 | 68 | 67 | 66,6 |
| 3.5 | 72 | 74 | 73 | 73 |
| 4 | 73 | 76 | 76 | 75 |

Наступним проведемо дослід з використанням перешкод. Для реалістичності дослідження та ситуації, розміщатимемо смартфон на відстані 1.5м з однієї сторони перешкоди та на такій же відстані Bluetooth давач з іншої сторони перешкоди. Запишемо втрату сили сигналу (RSSI, dBm) від давача в порівнянні з 100% сили сигналу. Результати для різних видів перешкод подамо у таблиці 4.4. Як було описано раніше, в якості перешкод оберемо дерев’яні та металеві двері, стіну та скло. Відповідно у таблиці позначимо перешкоди як дерево, метал, стіна.

*Таблиця 4.4.*

**Результати тестування втрати сили сигналу з різними видами перешкод**

|  |  |  |  |
| --- | --- | --- | --- |
| Перешкода | | | |
| Дерево | Метал | Стіна | Скло |
| 10 | 18-23 | 15-20 | 4 |

## **Висновки до розділу 4**

Протестувавши складові компоненти мобільної системи можна зробити висновок, що час виконання запитів є оптимальним та не виконується надто довго. Великий час виконання першого запиту до сервера можна обґрунтувати його економією ресурсів, тому після отримання першого запиту, сервер переходить в активний режим та опрацьовує подальші запити.

Запити на давання нового давача є значно швидшими, оскільки на сервер передається модель малого розміру, що дозволяє надіслати запит швидше.

Відповідно до кількості даних, довжина запиту ростиме.

Не варто розраховувати на швидку відповідь від серверу при слабкому сигналі мережі. Такий запит виконуватиметься довше.

У результаті дослідження Bluetooth давача було отримано хороші результати та показники його роботи. Наглядно доказано, що такий спосіб взаємодії є хорошим для використання у більшості сучасних приміщень. Звісно не можна розраховувати на однакові якість роботи давача оскільки досліди не проведено на великій кількості перешкод та на більшій відстані, але даного результату достатньо щоб дана система мала шанс на існування та розвиток. З проведених досліджень та даних про перешкоди та їх види, при використанні слід врахувати описані перешкоди при розміщенні давачів. Попри те, при повсякденному використанні, відчутних недоліків чи непрацездатності системи не відчуватиметься. Аналізуючи інтернет джерела, при виникненні перешкод для сигналу, можна скористатись способами підвищення ефективності давача [20]. Для цього можна збільшити силу розповсюдження сигналу, що в свою чергу впливатиме на час роботи пристрою від одного заряду. З виконаних досліджень можна зробити висновок, що давачі краще розміщувати в таких місцях, де відбуватиметься прямий контакт з смартфоном, тобто в одному приміщення з користувачами, що знаходитимуться в ньому.

# РОЗДІЛ 5. ЕКОНОМІЧНА ХАРАКТЕРИСТИКА ПРОГРАМНОГО ПРОДУКТУ

## **Економічна характеристика проектного рішення**

Магістерська кваліфікаційна робота містить в собі опис розробки мобільної системи для допомоги незрячим особам орієнтуватись в приміщеннях.

Проект включає розробку веб клієнту, серверного застосунку, бази даних та мобільного застосунку. Для роботи необхідно мати Bluetooth модуль.

З врахуванням існуючих аналогів систем для незрячих осіб, появилась необхідність в розробці нового рішення з іншим підходом до взаємодії незрячого користувача з навколишнім світом шляхом озвучування йому інформації, що сприятиме йому в орієнтуванні.

Існуючі підходи акцентують свою увагу на подачі звукових сигналів лише для перенаправлення людини в певному напрямку, заборони або дозволу на рух чи інші вказівки.

Метою є створення системи, яка базується на взаємодії Bluetooth пристроїв з смартфоном для отримання та озвучування тексту в залежності від відстані до давачів на основі величини сигналу від Bluetooth давача.

## **Розрахунок витрат на розробку та впровадження проектного рішення**

1) Витрати на розробку і впровадження програмного засобу (К) визначаються як:

 (5.1)

де – витрати на розробку програмного засобу, грн.;

 – витрати на відлагодження і дослідну експлуатацію програмного засобу на ЕОМ, грн.

Витрати на розробку програмного засобу включають в себе:

1. витрати на оплату праці розробників ();
2. єдиний соціальний внесок ();
3. вартість додаткових виробів, що закуповуються ();
4. накладні витрати ();
5. інші витрати ().

Для проведення розрахунків витрат на оплату праці необхідно визначити категорії працівників, які приймають участь в процесі проектування, їх чисельність, середньоденну заробітну плату спеціаліста відповідної категорії та трудомісткість робіт у людино-днях (людино-годинах).

До цієї статті належать витрати на виплату основної та додаткової заробітної плати керівникам відділів, лабораторій, секторів і груп, науковим, інженерно-технічним працівникам та іншим працівникам, безпосередньо зайнятим розробкою програмного забезпечення за конкретною темою (технічним завданням), обчисленої за посадовими окладами, відрядними розцінками, тарифними ставками згідно з діючими в організації системами оплати праці, включаючи будь-які види грошових і матеріальних доплат.

Над даним проектом працюють: програміст з місячною заробітною платою 13000грн., тестувальник з заробітною платою 9000грн в місяць та керівник проекту – 15000грн. Сума для програміста та тестувальника обгрунтована технологіями для розробки. Програміст працює над розробкою серверної частини, бази даних, веб та смартфон клієнту та працює з давачами. Тестувальник тестує продукт повністю.

Середньоденна заробітна плата і-го розробника () обчислюється за формулою:

, (5.2)

де - основна місячна заробітна плата розробника і-ої спеціальності, грн.;

 – місячний фонд робочого часу, днів (24 дні – листопад 2018).

Таким чином – денна заробітня плата програміста:

ЗПд1=13000 / 24 = 541

Денна заробітна плата тестувальника:

ЗПд2 = 9000 / 24 = 375 грн;

Денна заробітна плата керівника проекту:

ЗПд3 = 15000 / 24 = 625 грн

Розрахунок витрат на оплату праці усіх розробників проекту обчислюємо за формулою:

, (5.3)

де – кількість розробників проекту і-ої спеціальності, чол.;

– час, витрачений на розробку проекту працівником і-ої спеціальності, дні;

– денна заробітна плата розробника і-ої спеціальності, грн.;

Розрахунок витрат на оплату праці розробників зводиться у таблицю 5.1.

*Таблиця 5.1.*

**Розрахунок витрат на оплату праці**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Спеціальність  розробника | Кількість осіб. | Час роботи, дні | Денна заробітна плата розробника, грн. | Витрати на оплату праці, грн. |
|
| Програміст | 1 | 32 | 541 | 17312 |
| Тестувальник | 1 | 17 | 375 | 6375 |
| Керівник | 1 | 32 | 625 | 20000 |
| Всього | 3 | - | - | 43687 |

2) Витрати на оплату праці працівникам призводять до виникнення зобов'язань підприємства за єдиним соціальним внеском (2-й клас ризику -22%)

Вф = 43687 \* 0,22 = 9611,14 грн.

3) Витрати на додаткові вироби, що закуповуються (Вд) (папір, накопичувачі, тощо) визначаються за їхніми фактичними цінами з врахуванням найменування, номенклатури та необхідної їх кількості в проекті. Вихідні дані та результати розрахунків оформляються у таблицю 5.2. Транспортно-заготівельні витрати (B*mp*) становлять 15% суми витрат на додаткові вироби, що закуповуються.

*Таблиця 5.2.*

**Розрахунок витрат на закуплені вироби**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Найменування купованих виробів | Марка, тип | Кількість штук | Ціна за одиницю, грн | Сума витрат, грн. | Сума витрат з урахуванням транспортно-заготівельних витрат, грн. |
| Папір | MAESTRO, А4,  500арк, клас C | 1 | 50 | 50 | 57,5 |
| Ручка | Ручка кулькова  Economix BOLIDE | 10 | 5,04 | 50,4 | 57.96 |
| Тонер | 1 | 1 | 150 | 150 | 172,5 |
| Всього | - | - | - | 250,4 | 287,96 |

Отже, витрати на додаткові вироби, що закуповуються (*Вд*) дорівнюють 287,96 грн.

4) Накладні витрати (*)* проектних організацій включають витрати на управління, загальногосподарські, невиробничі витрати. Вони становлять 24 % витрат на оплату праці:

Вн = 43687⋅ 0,24 = 10484,88 грн

5) Інші витрати (*)* – це витрати, які не враховані в попередніх статтях витрат. Вони розраховуються за встановленими відсотками 9% до витрат на оплату праці:

Він = 43687⋅0,09 =3931,83 грн

6) Витрати на розробку проектного рішення обчислюємо за формулою:

 (5.4)

Підставивши значення у формулу 5.4, отримаємо:

*К* = 43687+9311,14 +287,96 + 10487,88 + 3931,83 = 67705,81 (грн.).

Результати обрахунків зведені в табл. 5.3

*Таблиця 5.3.*

**Кошторис витрат на розробку проектного рішення**

|  |  |
| --- | --- |
| Назва елементів витрат | Сума витрат, грн |
| Витрати на розроблення проектного рішення, зокрема: |  |
| * витрати на оплату праці | 43687 |
| * відрахування у спеціальні державні фонди | 9311,14 |
| * відрахування на додаткові вироби, що закуповуються | 287,96 |
| * накладні витрати | 10487,88 |
| * інші витрати | 3931,83 |
| Разом | 67705,81 |

## **Визначення комплексного показника якості**

Для визначення П*я* доцільно використовувати систему показників технічного рівня і якості, яка містить в собі наступні групи, причому в кожній групі вказана в дужках мінімальна кількість показників:

1. показники призначення (функціональності) (3-4);
2. показники надійності (2-3);
3. зручність застосування (1-3);
4. супроводжуваність (1-2);
5. інші показники.

*Комплексний показник* якості проектованої системи визначаємо методом арифметичного середньозваженого з формули:

 (5.5)

де  - кількість одиничних показників (параметрів), прийнятих для оцінки якості проектованої системи;

- коефіцієнт вагомості кожного з параметрів щодо їхнього впливу на технічний рівень та якість проектованої системи (встановлюється експертним шляхом), причому:

 (5.6)

 *-* часткові показники якості, визначені порівнянням числових значень одиничних показників проектованої системи і аналога за формулами:

 або , (5.7)

де ,  - кількісні значення і-го одиничного показника якості відповідно проектованої системи і аналога.

З попередніх двох формул вибирається та, в якій збільшення відповідає покращенню показника якості проектованої системи. Результати розрахунку зведені у таблицю 5.4.

*Таблиця 5.4*

**Показники якості проектованої системи та аналога**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Найменування показника | | Числові значення | | Відносний показник (С) | Коефіцієнт вагомості (q) | C\*q |
| Аналог | Проектована система |
| Функціональна повнота | | 8 | 9 | 1,125 | 0,15 | 0,16875 |
| Правильність | | 8 | 8 | 1 | 0,1 | 0,1 |
| Сумісність | | 6 | 6 | 1 | 0,05 | 0,05 |
| Захищеність | | 9 | 7 | 1,286 | 0,05 | 0,0643 |
| Безвідмовність | | 10 | 9 | 1,11 | 0,1 | 0,111 |
| Відновлюваність | | 6 | 9 | 1,5 | 0,1 | 0,15 |
| Зрозумілість | | 6 | 9 | 1,5 | 0,1 | 0,15 |
| Оперативність | | 9 | 9 | 1 | 0,05 | 0,05 |
| Аналізованість | | 7 | 10 | 1,42 | 0,05 | 0,071 |
| Актуальність | | 7 | 8 | 1,14 | 0,1 | 0,114 |
| Ступінь новизни | | 6 | 10 | 1,67 | 0,15 | 0,2505 |
| Разом | - | | - | - | 1 | 1,28 |

Отже, комплексний показник якості (Пя) рівний 1,28.

## **Визначення експлуатаційних витрат**

При порівнянні програмних засобів в експлуатаційні витрати включають вартість підготовки даних () і вартість годин роботи ПК (). Одноразові експлуатаційні витрати визначаються за формулою:

 (5.8)

де - одноразові експлуатаційні витрати на проектне рішення (аналог), грн.;

- вартість підготовки даних для експлуатації проектного рішення (аналогу), грн.;

 - вартість машино-годин роботи ПК для проектного рішення (аналогу), грн.

Річні експлуатаційні витрати визначаються за формулою:

 (5.9)

де – експлуатаційні річні витрати проектного рішення, грн.;

 - періодичність експлуатації проектного рішення (аналогу), разів/рік.

Вартість підготовки даних для експлуатації проектного рішення (аналогу) () визначаються за формулою:

 (5.10)

де  – номери категорій персоналу, які беруть участь у підготовці даних;

– кількість співробітників і-ї категорії, чол.;

– трудомісткість роботи співробітників і-ї категорії, чол.;

– середньогодинна ставка робітника і-ї категорії з врахуванням сплати єдиного соціального внеску, грн./год.

Середньогодинна ставка оператора визначається за формулою:

 (5.11)

де – основна місячна зарплата працівника і-ї категорії, грн.;

– коефіцієнт, який враховує сплату єдиного соціального внеску (його ставка повинна відповідати значенню, визначеному у п. 2 економічної частини);

 – місячний фонд робочого часу, год.

Отже, для проектного рішення середньогодинна ставка становить:

ЗПг1 = 15000 *⋅*(1+0,24) / (24*⋅*8) = 96.83 грн.

ЗПг2 = 13000 *⋅*(1+0,24) / (24*⋅*8) = 83.95 грн.

ЗПг3 = 9000 *⋅*(1+0,24) / (24*⋅*8) = 58.12 грн.

Тоді одноразові експлуатаційні витрати на проектне рішення становлять:

*EП =*(1*⋅*32*⋅* 96.83) +(1*⋅*32*⋅* 83.95) + (1*⋅*17*⋅* 58.12) = 6773 грн.

A річні експлуатаційні витрати з урахуванням того, що періодичність експлуатації проектного рішення дорівнює 12 разів/рік, становлять:

*В(е)П* = 6773*⋅*12 = 81276 грн.

Над проектом-аналогом працює 1 керівник проекту, 4 розробники та 2 тестери. Їхні місячні заробітні ставки відповідно 9000 грн., 7000 грн. та 5000 грн. Тоді середньогодинна ставка для кожного з них становить:

ЗПг1 = 9000⋅ (1+0,24) / (24⋅ 8)= 58.12 грн.

ЗПг2 = 7000⋅ (1+0,24) / (24⋅ 8) = 45.20 грн.

ЗПг3 = 5000⋅ (1+0,24) / (24⋅ 8) = 32.29 грн.

Тоді одноразові експлуатаційні витрати на аналог становлять:

*EА* =(1*⋅*32*⋅*58.12)+(4*⋅*32*⋅*45.20)+(2*⋅*17⋅32.29) = 8743.3 грн.

A річні експлуатаційні витрати з урахуванням того, що періодичність експлуатації аналогу дорівнює 12 раз/рік, становлять:

*B(е)А*=8743.3 *⋅*12 = 104919.6 грн.

Вихідні дані та результати розрахунків витрат на підготовку даних для експлуатації на EOM зводяться у табл. 5.5.

*Таблиця 5.5*

**Розрахунок витрат на підготовку даних для роботи на EOM**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Категорія  персоналу | Чисельність співробітників і-ої категорії, люд., | | Час роботи співробітників і-ої категорії, год. | | | Середньогодинна ЗП співробітника і-ої категорії, грн. | | Витрати на підготовку даних, грн. |
| 1 | 2 | | 3 | | | 4 | | 5 |
| Проектне рішення | | | | | | | | |
| Керівник проекту | 1 | | 4 | | | 96.83 | | 387,32 |
| Розробник | 1 | | 6 | | | 83.95 | | 503,7 |
| Тестувальник | 1 | | 4 | | | 58.12 | | 232,48 |
| Всього | - | | - | | | - | | 1123,5 |
| Аналог | | | | | | | | |
| Керівник проекту | | 1 | | 4 | 58.12 | | 232,48 | |
| Розробник | | 4 | | 24 | 45.20 | | 1084,8 | |
| Тестувальник | | 2 | | 8 | 32.29 | | 258,32 | |
| Всього | | - | | - | - | | 1575,6 | |

Витрати на підготовку даних для розроблюваного програмного продукту є суттєво меншими ніж витрати на підготовку даних для програми аналога.

## **Розрахунок ціни споживання проектного рішення**

Ціна споживання () – це витрати на придбання і експлуатацію проектного рішення за весь строк його служби:

 (5.12)

де – ціна придбання проектного рішення, грн.;

– теперішня вартість витрат на експлуатацію проектного рішення (за весь час його експлуатації), грн.:

, (5.13)

де – норматив рентабельності (приймаємо 25%);

– витрати на прив'язку та освоєння проектного рішення на конкретному об’єкті, грн.;

*K0=* 1500 грн.;

– витрати на доукомплектування технічних засобів на об'єкті, грн.

*Кк* = 0 грн.;

*СПДВ -*ставка податку на додану вартість (20 %).

*ЦП* =67705,81 *⋅* (1 + 25/100)*⋅* (1 + 0,2) + 1500 + 0 = 103058,71 грн.

Згідно ринкових цін, ціна аналога становить ЦА=327770,00 грн.

Теперішня вартість витрат на експлуатацію проектного рішення розраховується за формулою:

 (5.14)

де  - річні експлуатаційні витрати в t-ому році, грн.;

 - строк служби проектного рішення, 3 роки;

 - річна ставка проценту банків(15%).

*B(E)NPV =* 81276 / (1 + 0,15)1 +81276 / (1 + 0,15)2 + 81276 / (1+ 0,15)3 =

= 185602,16 грн.

Таким чином ціна споживання проектного рішення становить:

*ЦС(П)* = 103058,715 + 185602,16 = 288660,87 грн.

Аналогічно визначається ціна споживання для аналогу. Визначимо теперішню вартість витрат на експлуатацію аналогу. Термін експлуатації аналогу становить 3 років, тоді за формулою 5.15:

*B(E)NPV =* 327770,00 / (1 + 0,15)1 + 327770,00 / (1 + 0,15)2 + 327770,00 /

/ (1 + 0,15)3 = 748383,32 грн.

Таким чином ціна споживання аналогу становить:

*ЦС(А)* = 327770,00 +748383,32 = 1076153,32 грн.

## **Визначення показників економічної ефективності**

1. Показник конкурентоспроможності:

 (5.15)

ККС =1076153,32⋅1,28 / 288660,875 = 4,77

1. Економічний ефект в сфері експлуатації:

ЕЕКС = В(Е)А - B(Е)П (5.16)

ЕЕКС= 104919.6 – 81276= 23646.6 грн.

1. Економічний ефект в сфері проектування:

*EПР*= *ЦА - ЦП*  (5.17)

EПР = 327770,00 - 103058,71 = 224711,28 грн.

1. Додатковий економічний ефект в сфері експлуатації (3 роки):

 (5.18)

ЕЕКС Д*=* 23646,6⋅ (1,150+1,151+1,152) = 94370,62 грн*.*

1. Додатковий економічний ефект в сфері проектування:

 (5.19)

*EПР Д* =224711,285⋅ 1,15 = 258417,97 грн.

1. Термін окупності витрат на проектування рішення:

*ТОК =* (5.20)

ТОК = 67705,8/ 23646,6 = 2.86 року aбo 2 роки і 9 місяців.

Результуючі показники економічної ефективності зводяться у табл. 5.6.

*Таблиця 5.6*

**Показники економічної ефективності проектного рішення**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Найменування показників | Одиниці  вимірювання | Значення показників | | |
| Аналог | | Проектне  рішення |
| 1 | 2 | 3 | | 4 |
| 1. Капітальні вкладення | грн. | | - | 67705,8 |
| 2. Ціна придбання | грн. | | 327770,00 | 103058,71 |
| 3. Річні експлуатаційні витрати | грн. | | 104919.6 | 67217,64 |
| 4. Ціна споживання | грн. | | 1076153,32 | 288660,87 |
| 5. Економічний ефект в сфері експлуатації | грн. | | - | 23646.6 |
| 6. Додатковий економічний ефект в сфері експлуатації | грн. | | - | 94370,62 |
| 7. Економічний ефект в сфері проектування | грн. | | - | 94370,62 |
| 8. Додатковий економічний ефект в сфері проектування | грн. | | - | 258417,97 |
| 9. Термін окупності витрат на проектування рішення | місяці | | - | 2,86 (2 роки, 9 місяців) |
| 10. Коефіцієнт  конкурентоспроможності | - | | - | *4,77* |

## **Висновки до розділу 5.**

У цьому розділі дипломного проекту було проведено економічну оцінку проектного програмного продукту та його аналогу «OpenWorld». Після аналізу отриманих результатів можна зробити висновок, що дана система є конкурентоспроможною, оскільки коефіцієнт конкуренто-спроможності становить 2,17.

Отже, сума витрат на розробку проектного рішення рівна 67705,8 грн. Ціна розробленого продукту буде становити 103058,71 грн, що являється меншим, ніж ціна аналогу - 132000,00 грн. Відповідно ціна споживання продукту аналогу, - 1076153,32 грн. набагато перевищує ціну споживання розробленої системи - 288660,87 грн.

Отож, згідно проведених розрахунків можна стверджувати, що розроблювані засоби для проведення автоматизованого тестування програмного забезпечення вбудованих систем є кращою від її аналогу за економічними показниками.

Даний проект враховує підтримку актуальності свого функціоналу протягом 3 років, його підтримку та подальше розширення. Малий термін окупності витрат на проектування, який становить всього 2 роки та 9 місяців, прогнозує економічну вигідність продукту.

# **ЗАГАЛЬНІ ВИСНОВКИ**

Розроблена мобільна система допомагатиме орієнтуватись незрячим особам в приміщеннях. Озвучування тексту є індивідуальним для кожного користувача, озвучує дані лише йому та не відволікатиме оточуючих сторонніми сигналами чи повідомленнями. Система реалізована у вигляді веб та мобільного клієнту з використанням сучасних засобів та технологій розробки програмного забезпечення та передбачає можливість подальшого його оновлення. Перевагою такої системи є відкритість для сторонніх розробників, можливість взаємодіяти з створеними власноруч веб клієнтами, мобільними клієнтами чи іншими застосунками/платформами, що можуть працювати за http протоколом. Перевагою є те, що вартість давача коливається в діапазоні 70-150грн. Вартість може змінювати, якщо враховувати вартість виготовлення корпусу для давачів, батарейок, та оренди доменного імені та хостингу серверу та бази даних.

Виконані дослідження складових мобільної системи показали, що система конкурентно спроможна, дає хороші показники швидкості роботи та передачі даних. У майбутньому, за рахунок гнучкості така система може бути легко удосконалена шляхом створення локального сховища для зберігання даних про давачі, використання GPS, акселерометра тощо. Система може бути удосконалена як повністю, так і можуть бути удосконалені її компоненти без суттєвих змін чи погіршення умов чи якості роботи.

# **СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ**

1. Vicente Cantón Paterna, Anna Calveras Augé, Josep Paradells Aspas. A Bluetooth Low Energy Indoor Positioning System with Channel Diversity, Weighted Trilateration and Kalman Filtering: Sensors — Open Access Journal. Barcelona, Spain, 2017. P. 150-182.
2. Новак Д. Мобільна система для орієнтування незрячих осіб в приміщеннях університету: 10 науково практична конференція, 2018. Львів, Україна. 2018. С. 94-99.
3. Taylor Heinecke, Taylor Heinecke, Mark Wolfe The Role of Bluetooth Low Energy For Indoor Positioning Applications. USA. Montana State University. Computer Science Department Bozeman. 2016. P. 8-11.
4. Johan Larsson. Distance estimation and positioning based on Bluetooth low energy technology: Master of Science Thesis. Kista, 2015. P. 12-22.
5. Is RSSI a Reliable Parameter in Sensor Localization Algorithms – An Experimental Study / Ambili Thottam Parameswaran, Mohammad Iftekhar Husain, Shambhu Upadhyaya. Buffalo, USA: Department of Computer Science and Engineering State University of New York, 2012. P. 9-15.
6. OpenWorld: веб-сайт. URL:

[biggggidea.com/project/openworld/](https://biggggidea.com/project/openworld/) (дата звернення: 14.11.2018).

1. Hyunwook Park, Jaewon Noh and Sunghyun Cho. Three-dimensional positioning system using Bluetooth low-energy beacons: International Journal of Distributed Sensor Networks. 2016. Vol. 12. 2017. P. 1-11.
2. ESP32 Series Datasheet: веб-сайт. URL:

<https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf> (дата звернення: 15.11.2018).

1. Савочкин Д.А., Гимпилевич Ю. Оптимизация размещения антенн для систем пространственной двумерной RFID-локализации: Радіоелектроніка, інформатика, управління. 2015. № 2. Севастополь, 2015. С. 7-9.
2. Kannan Srinivasan, Philip Levis, Kannan Srinivasan. RSSI is Under Appreciated: Статті з продовжуючих та періодичних видань. Stanford, CA. Department of Electrical Engineering and ∗Department of Computer Science, Stanford University. 2006. P. 48-55.
3. Bluetooth positioning using RSSI and triangulation methods / Yapeng Wang, Xu Yang, Yutian Zhao. Conference Paper January, 2013. Las Vegas. P. 141-149.
4. Varun Almaula, David Cheng. Bluetooth Triangulator. San Diego La Jolla, CA. Department of Computer Science and Engineering University of California. 2010. P. 2-6.
5. Retrofit: веб-сайт. URL: square.github.io/retrofit/ (дата звернення: 12.11.2018).
6. Song Chai, Renbo An, Zhengzhong Du. An Indoor Positioning Algorithm Using Bluetooth Low Energy RSSI: International Conference on Advanced Material Science and Environmental Engineering (AMSEE 2016). Chengdu, China. 2016. P. 276-278.
7. Фальков Е. В., Романов А. Ю., Применение маячков Beacon и технологии Bluetooth Low Energy для построения систем навигации в зданиях: Национальный исследовательский университет «Высшая школа экономики». Москва, 2015. С. 62-65.
8. Stochastic geometry models of wireless networks: веб-сайт. URL:

<https://en.wikipedia.org/wiki/Stochastic_geometry_models_of_wireless_networks> (дата звернення: 15.11.2018).

1. Potential sources of Wi-Fi and Bluetooth interference: веб-сайт. URL: - <https://support.apple.com/en-euro/HT201542> (дата звернення 15.11.2018).
2. Троелсен Эндрю. Язык програмирования C# 5.0 и платформа .NET 4.5. 6-е изд. Пер. з англ. – М.: ООО «И.Д. Вильямс». 2013. – С. 47.
3. Alexander Lindemann, Bettina Schnor, Jan Sohre. Indoor Positioning: A Comparison of WiFi and Bluetooth Low Energy for Region Monitoring: In Proceedings of the 9th International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2016) - Volume 5: HEALTHINF. Potsdam, Germany. 2016. P. 314-321.
4. Abdur Rahim, Radka Dimitrova, Adolf Finger. Techniques for Bluetooth Performance Improvement: Semantic Scholar, 2010. Communications Laboratory Dresden University of Technology, Germany. 2010. P. 46-51.

# **ДОДАТКИ**

## **Додаток А. Діаграма розгортання**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Діаграма Пакетів | | | | | |
| Виконав | Новак Д.С. |  |  | Діаграма розгортання | Сторінка |
| Перевірив | Федасюк Д.В. |  |  | 1 з 1 |
| ІКНІ кафедра ПЗ ПЗІП-21 | | | |

## **Додаток Б. Діаграма послідовностей**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Діаграма взаємодії | | | | | |
| Виконав | Новак Д.С. |  |  | Діаграма послідовностей | Сторінка |
| Перевірив | Федасюк Д.В. |  |  | 1 з 1 |
| ІКНІ кафедра ПЗ ПЗІП-21 | | | |

## **Додаток В. Інструкція користувача**

### **Налаштування серверу та баз даних мобільної системи.**

Програмне забезпечення розроблено у середовищі Microsoft Visual Studio 2015 на мові програмування C# з використанням ASP.NET MVC WEB API для операційних систем сімейства Windows.

Для коректної роботи потрібна користувацька машина з процесором не менше 1,7 Ггц, оперативною пам'яттю не менше 1024 Мб і операційною системою Windows.

Дане програмне забезпечення не потребує додаткового встановлення. Для початку роботи з системою потрібно запустити виконавчий .sln файл проекту та запустити проект або ж просто налаштувати його у IIS. Всі інші необхідні бібліотеки встановляться автоматично при запуску проекту з середовища або ж використовуватимуться наявні з проекту.

Сервер, разом з базою даних після запуску та компіляції можна розміщати на власному сервері або ж на орендованому хостингу.

Для уникнення всіх кроків з розміщення власного серверу, можна скористатись існуючим, на якому розміщена документація для написання власних застосунків.

### **Налаштування смартфон клієнту мобільної системи**

Смартфон клієнт можна використати вже готовий інсталяційний файл формату \*.apk, встановивши його на смартфон. Увімкнути Bluetooth та натиснути кнопку старту. Для продовження розробки існуючого застосунку, необхідно встановити середовище розробки та відкрити проект у ньому.

### **Робота з Bluetooth давачем**

Для роботи з мобільною системою, необхідні давачі. Давачі можна придбати у інтернет магазинах або магазинах електроніки у локації користувача. Для роботи з давачем, необхідно з упаковки переписати MAC адресу у веб клієнт, при додаванні давача в базу. Потім розмістити давач у обраному місці та підключити його до джерела живлення.

## **Додаток Г. Код отримання Bluetooth пристроїв та даних про давачі**

Отримання Bluetooth пристрою та озвучування тексту користувачу

private final BroadcastReceiver mReceiver = new BroadcastReceiver() {

public void onReceive(Context context, Intent intent) {

String action = intent.getAction();  
 if (BluetoothDevice.*ACTION\_FOUND*.equals(action)) {  
 // Discovery has found a device. Get the BluetoothDevice  
 // object and its info from the Intent.  
 BluetoothDevice device = intent.getParcelableExtra(BluetoothDevice.*EXTRA\_DEVICE*);  
 String deviceName = device.getName();  
 String deviceHardwareAddress = device.getAddress(); // MAC address  
 int rssi = intent.getShortExtra(BluetoothDevice.*EXTRA\_RSSI*,Short.*MIN\_VALUE*);  
 Log.*i*("BLE--->","Name: " + deviceName + " MAC: " + deviceHardwareAddress + " RSSI: " + rssi);  
 }  
 }  
};  
  
public void SpeakText(String text){  
 String toSpeak = text;  
 Toast.*makeText*(getApplicationContext(), toSpeak,Toast.*LENGTH\_SHORT*).show();  
 t1.speak(toSpeak, TextToSpeech.*QUEUE\_FLUSH*, null);  
}

Отримання списку існуючих давачів для їх розміщення на карті

[System.Web.Http.Route("getallmarkers")]

[System.Web.Http.HttpGet]

[System.Web.Http.ActionName("Get all Records")]

public IEnumerable<Sensor> GetMapMarker()

{

BNBDB.Configuration.ProxyCreationEnabled = false;

var result = BNBDB.Sensor.Include(e =>e.Location).Include(e=>e.Author).ToList();

return result;

}

Додавання нового давача у базу даних

[System.Web.Http.Route("addnewsesor")]

[System.Web.Http.HttpGet]

[System.Web.Http.ActionName("Add new sensor")]

public JsonResult AddSensorInfo([FromBody]Sensor sensor)

{

Guid SKey = Guid.NewGuid();

sensor.SKey = SKey.ToString();

try

{

BNBDB.Sensor.Add(sensor);

BNBDB.Location.Add(sensor.Location);

BNBDB.Author.Add(sensor.Author);

BNBDB.SaveChanges();

return Json(new { success = SKey }, JsonRequestBehavior.AllowGet);

}

catch (Exception e)

{

return Json(new { error = e.Message }, JsonRequestBehavior.AllowGet);

}

}

Додавання нового відгуку

[System.Web.Http.Route("addfeedback")]

[System.Web.Http.HttpPost]

[System.Web.Http.ActionName("Add new feedback")]

public JsonResult AddUserFeedback([FromBody]Feedback data)

{

try

{

BNBDB.Feedback.Add(data);

BNBDB.SaveChanges();

return Json(new { success = "Feedback received" }, JsonRequestBehavior.AllowGet);

}

catch (Exception e)

{

return Json(new { error = e.Message }, JsonRequestBehavior.AllowGet);

}

}

Отримання всіх давачів за ключовими словами

[System.Web.Http.Route("getByKeyword")]

[System.Web.Http.HttpGet]

[System.Web.Http.ActionName("Get sensors by keyword")]

public IEnumerable<Sensor> GetMarkersByKeyWord([FromBody]String data)

{

BNBDB.Configuration.ProxyCreationEnabled = false;

var result = BNBDB.Sensor.Include(e => e.Location).Include(e => e.Author).

Where(s=>s.Location.LCountry.Contains(data) ||

s.Location.LCity.Contains(data) ||

s.Location.LFromMap.Contains(data)).ToList();

return result;

}

Отримання давача за ключем

[System.Web.Http.Route("getSensorByKey")]

[System.Web.Http.HttpGet]

[System.Web.Http.ActionName("Get sensor by key")]

public Sensor GetSensorByKey(String sensorKey)

{

BNBDB.Configuration.ProxyCreationEnabled = false;

var result = BNBDB.Sensor.Include(e => e.Location).Include(e => e.Author).

Where(s => s.SKey == sensorKey).SingleOrDefault();

return result;

}