

Міністерство освіти і науки України  
Кам'янець-Подільський національний університет імені Івана Огієнка  
Фізико-математичний факультет  
Кафедра комп'ютерних наук

Дипломна робота  
з теми: “Розробка та реалізація програмно-апаратної частини та користувачького інтерфейсу мобільного застосунку «Інтерактивна карта»”

Виконав: студент 2 курсу ступеня вищої освіти  
магістр групи KN1-M20  
спеціальності 122 Комп'ютерні науки  
Олішинський Владислав Борисович  
Керівник: Понеділок В.В., кандидат технічних  
наук, старший викладач кафедри комп'ютерних  
наук  
Рецензент:

Кам'янець-Подільський – 2021 р.

## Зміст

ВСТУП.....	4
РОЗДІЛ 1 ОПИС ІСНУЮЧИХ ТЕХНОЛОГІЙ ДЛЯ МОБІЛЬНОЇ РОЗРОБКИ.....	7
1.1 Нативні технології для розробки мобільних додатків.....	9
1.1.1 Платформа Android .....	9
1.1.1.1 Java .....	11
1.1.1.2 Kotlin.....	13
1.1.2 Платформа iOS.....	17
1.1.2.1 Objective-C.....	18
1.1.2.2 Swift.....	20
1.2 Кросплатформові технології для розробки мобільних додатків .....	23
1.2.1 React Native .....	23
1.2.2 Flutter.....	26
1.2.3 Cordova .....	28
РОЗДІЛ 2 ОПИС ПРОГРАМНО-АПАРАТНОЇ ЧАСТИНИ ЗАСТОСУНКІВ .....	30
2.1 Web- та HTTP-сервери .....	30
2.2 Технології реалізації HTTP-серверів.....	32
2.2.1 NodeJS.....	32
2.2.2 PHP (Laravel).....	34
РОЗДІЛ 3 РЕАЛІЗАЦІЯ МОБІЛЬНОГО ЗАСТОСУНКУ «ІНТЕРАКТИВНА КАРТА» ТА ЙОГО ПРОГРАМНО-АПАРАТНОЇ ЧАСТИНИ .....	37
3.1 Вибір технології для розробки користувацького інтерфейсу мобільного застосунку .....	37
3.2 Вибір технології для серверної розробки .....	37
3.3 Проектування архітектури.....	38
3.4 Налаштування середовища розробки та структури проєктів .....	39

	3
3.5 Розробка користувацького інтерфейсу мобільного додатку .....	43
3.6 Розробка програмно-апаратної частини застосунку (API).....	49
ВИСНОВКИ.....	53

## ВСТУП

Ринок мобільних додатків росте неабиякими темпами. Так, у 2020 році загальні доходи їх розробників перевищили 600 мільярдів доларів, навіть незважаючи на пандемію. Більше того, коронавірус прискорив розвиток деяких сегментів мобільних додатків, а деяким дало друге дихання. В останні декілька років можна спостерігати наступні тенденції, пов'язані з мобільними додатками:

- **Збільшення кількості мобільних додатків.** Зараз мати свій додаток – правило хорошого тону для організації. Якщо раніше це було характерно, в основному, для сфери послуг, то тепер свої мобільні додатки мають навіть державні установи, де інновації. Інша справа, що мало зробити додаток, потрібно, щоб воно було простим у використанні, надійним і володіло наглядним інтерфейсом. У протилежному випадку доцільність його розробки та впровадження ставиться під сумнів.
- **Ріст сегменту мобільної комерції.** На початку пандемії коронавірусу багато країн закрили торгові центри та ввели обмеження на роботу магазинів. Тому люди стали частіше замовляти продукти харчування, побутові речі, одяг, взуття, електроніку через свої гаджети. Так, за статистикою, за 2020 рік вдвічі збільшилось число встановлень додатків технічних сервісів, що пропонують послуги доставки. Беззаперечно, пандемія рано чи пізно відступить, але споживачі вряд чи відмовляться від можливості доставки до дверей, так як це значно економить час.
- **Управління голосом, штучний інтелект, машинне навчання.** Людина, в середньому, блокує і розблоковує свій смартфон 70-80 разів у день. Так як ми користуємося смартфонами не тільки вдома, але і, наприклад, в громадському транспорті, то на них осідає безліч бактерій. Тому люди намагаються зайвий раз не торкатися до гаджету. Саме цим і обумовлений розвиток системи голосового управління. Машинне навчання та штучний інтелект з'явилися в мобільних додатках досить давно. Однак, нерідко значення даних функцій применшується. Насправді, перспективи в даному

сегменті є, і вони дійсно грандіозні. Крім запису звичок власника гаджета (адрес доставки), в перспективі можливі такі розробки як, наприклад, аналітичні додатки, які здатні вибудовувати інвестиційну стратегію з урахуванням багатьох факторів.

- **Розвиток застосунків для віддаленої роботи.** Як відомо, одним із наслідків пандемії Covid-19 став перехід більшості галузей на віддалений режим роботи. Навіть зараз більшість працівників не хочуть повертатися до традиційного режиму роботи в офісі, а віддають перевагу віддаленій роботі з дому. Саме цей фактор неабияк вплинув на розвиток засобів віддаленої комунікації таких як, наприклад, Zoom, Google Meet та ін. Ріст популярності таких сервісів також вплинув на розвиток мобільних застосунків для віддаленої комунікації, так, в багатьох платформ з'явилися мобільні версії застосунків, а уже існуючі значно покращили якість продукту.
- **Нові програми для здоров'я та фітнесу.** Наявні зараз застосунки для здоров'я та фітнесу з легкістю здатні замінити блокноти, зі списком вправ, та щоденники з раціоном та порядком харчування. Їхній функціонал дозволяє повністю відслідковувати вашу активність як у повсякденному житті, так і при занятті спортом, що дозволяє значно краще бачити прогрес у тому чи іншому напрямку.

В рамках даної роботи виконується дослідження і розробка програмно-апаратної частини сервера та користувацького інтерфейсу мобільного додатку, який дозволить користувачам знаходити та додавати цікаві місця на карту у вигляді маркерів, щоб інші користувачі могли їх переглядати в режимі реального часу.

В період пандемії більшість працівників перейшли на віддалений режим роботи з дому. На протязі всього карантинного періоду люди були «закриті» у своїх будинках, і після послаблення карантинних мір, за допомогою розробленого додатку, можна буде дізнатися, що нового і цікавого з'явилось в його оточенні.

Окрім вище описаного даному додатку можна буде знайти безліч інших застосувань, починаючи від простих відміток цікавих місць, подій, зустрічей і

закінчуючи складною картою міста, з відображенням державних структур, підприємств, відомств, їхнього опису та усією необхідною інформацією.

Об'єктом дослідження в даній роботі виступає дослідження процесу розробки мобільного застосунку та серверу для обробки запитів.

Предметом дослідження в даній роботі є розробка кросплатформового мобільного застосунку та його програмно-апаратної частини.

Метою даної роботи є здобуття теоретичних та практичних навичок в розробці користувацького інтерфейсу кросплатформового мобільного додатку та розробці програмно-апаратної частини додатку.

Згідно сформованого змісту у дипломній роботі ставляться такі завдання:

- Дослідити нативні технології для розробки мобільних додатків під такі платформи як Android та iOS;
- Дослідити такі мови програмування як Java, Kotlin, Objective-C, Swift. Визначити їхні плюси та мінуси, області застосування та популярність використання на сьогоднішній день;
- Дати визначення кросплатформової розробки, її плюси та мінуси;
- Дослідити технології для кросплатформової розробки, а саме React Native, Flutter та Cordova;
- Дати визначення web- та HTTP-серверу, сфери їх застосування;
- Дослідити такі технології для реалізації веб-серверів, як NodeJS та PHP (Laravel);
- Вибрати технології для розробки користувацького інтерфейсу мобільного застосунку та веб-сервера;
- Описати архітектуру для веб-сервера;
- Налаштувати середовище для розробки;
- Розробити користувацький інтерфейс мобільного застосунку «Інтерактивна карта»;
- Розробити веб-сервер для мобільного застосунку;

## ВИСНОВКИ

В ході виконання були досліджені та описані існуючі технології для розробки мобільних застосунків, досліджені найпопулярніші технології для нативної розробки для платформ Android та iOS, а саме мови Java і Kotlin для Android та Objective-C і Swift для iOS. Описано переваги та недоліки кожної з них, актуальність їх використання на сьогоднішній день. За результатами дослідження, можна сказати, що нові технології, а саме Kotlin та Swift з щодня набирають все більше популярності, в тому числі і за рахунок того, що розробляють, просувають і підтримують їх такі компанії як Google та Apple відповідно.

Було описані технології для розробки кросплатформових мобільних застосунків, а саме React Native, Flutter та Cordova, які на сьогодні є одними з найбільш популярних серед кросплатформових рішень. Кожна з них має свої переваги та недоліки. Спільним недоліком для всіх технологій є низька, в порівнянні з нативними технологіями, швидкість роботи розроблених застосунків, оскільки всі вони працюють у своєму оточенні, яке взаємодіє з нативною частиною пристрою. React Native є другою за популярністю серед представлених, розроблена вона була Meta Platforms (Facebook) у 2015 р., та активно просувається самою Meta та спільнотою. Flutter розробляється та підтримується компанією Google, і також активно просувається, що дозволило підняти дану технологію на перше місце за популярністю. Cordova серед представлених технологій є менш популярною, але до її переваг можна віднести легкість в розробці, оскільки для написання використовується HTML, CSS та JavaScript.

Також було досліджено програмно-апаратну частину застосунків, з чого вона складається, як саме працює, та якими способами реалізовується. Описані web- та HTTP-сервери та мови, які використовуються для написання власних HTTP-серверів, а саме середовище виконання NodeJS та PHP фреймворк Laravel,

які є на сьогоднішній день одними з найпопулярніших технологій для реалізації серверної частини застосунків. За результатами дослідження, можна сказати, що NodeJS чудово підходить для рішень низької та середньої важкості, хоча доволі часто використовується і у великих проєктах, оскільки NodeJS не дуже добре справляється з важкими обчисленнями. Laravel у свою чергу чудово підходить для складних великих проєктів, але через високу важкість налаштувань не підходить для простіших рішень.

В останньому розділі даної роботи описаний процес розробки мобільного застосунку «Інтерактивна карта» на технології React Native. Основне застосування даного застосунку в розміщені на карті міток, які можуть переглядати інші користувачі, з метою ознайомлення та зустрічей. Функціонал застосунку включає в себе реєстрацію та авторизацію, додавання міток та переглядання міток, коментування міток, перегляд та редагування профілю.

Авторизація виконується в два етапи:

1. перевірка відповідності емейлу та паролю, які передаються в Firebase, де перевіряються на існування та відповідність один одному, і в разі успіху, авторизують користувача та повертають інформацію про користувача.
2. Далі виконується запит на сервер застосунку, який перевіряє існування користувача в базі даних та повертає його у відповідь.

Наступною частиною останнього розділу була розробка серверної частини для мобільного застосунку. Був описаний та аргументований вибір технологічного стеку для розробки, та в підсумку був обраний наступний набір технологій

- архітектурний стиль REST API;
- мова програмування JavaScript та платформа NodeJS як основу, поверх якої буде працювати фреймворк Express.js;
- база даних MongoDB;
- технологія автентифікації – Firebase Auth.



Вибраний стек для розробки носить назву MERN (Mongo-Express-React/React Native-NodeJS). А вибрана архітектура для серверної розробки слідує паттерну MVC (Model-View-Controller).

Розробка на Express.js складається з декількох таких складових етапів, опис роутів, на які буде звертатися клієнт, та написання функцій обробки роутів – контролерів. Виконавши ці два завдання, ми отримаємо готовий до роботи сервер, до якого можна звертатися.

В подальшому розроблений додаток можна розширити додатковим функціоналом, що збільшить коло його застосувань. Також його можна використати як каркас для створення нових, або доопрацювання даного функціоналу, з метою отримання нового додатку з поточним функціоналом.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. *Application Fundamentals [Електронний ресурс]: Android Developers.* – Режим доступу: <https://developer.android.com/guide/components/fundamentals>.
2. *Java [Електронний ресурс]: Вікіпедія. Вільна енциклопедія.* – Режим доступу: <https://uk.wikipedia.org/wiki/Java>.
3. *Number of available apps in the Apple App Store from 2008 to 2021 [Електронний ресурс]: Statista.* – Режим доступу: <https://www.statista.com/statistics/268251/number-of-apps-in-the-itunes-app-store-since-2008>
4. *iOS version history [Електронний ресурс]: Wikipedia the free encyclopedia.* – Режим доступу: [https://en.wikipedia.org/wiki/IOS\\_version\\_history](https://en.wikipedia.org/wiki/IOS_version_history)
5. *Andrew J. Wagner. Learning Swift. Build a solid foundation in Swift to develop smart and robust iOS and OS X applications.* – Packt Publishing, 2015. – 266 p.
6. *React Native [Електронний ресурс]: GitHub* – Режим доступу: <https://github.com/facebook/react-native/releases>
7. *Документація Flutter [Електронний ресурс]– Режим доступу: <https://api.flutter.dev/flutter/foundation/foundation-library.html>*
8. *Документація Apache Cordova [Електронний ресурс] – Режим доступу: <https://cordova.apache.org/docs/en/latest/guide/overview/index.html>*
9. *Документація NodeJS [Електронний ресурс] – Режим доступу: <https://nodejs.dev>*
10. *The 10 Best PHP Frameworks To Use in 2022 [Електронний ресурс]: Trio* – Режим доступу: <https://trio.dev/blog/php-frameworks>
11. *Что такое PHP? [Електронний ресурс] Php.su* – Режим доступу: <http://www.php.su/php/?php>
12. *Laravel source code [Електронний ресурс] Github* – Режим доступу: <https://github.com/laravel/laravel>

13. Sinatra *[Электронный ресурс]* – Режим доступа: <http://sinatrarb.com>
14. MVC *[Электронный ресурс]* – Режим доступа:  
<https://ru.wikipedia.org/wiki/Model-View-Controller>