

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ТЕХНОЛОГІЙ ТА ДИЗАЙНУ  
КАФЕДРА КОМП'ЮТЕРНИХ НАУК

## Дипломна магістерська робота

на тему: Розробка QR кода для веб- додатку  
«Ідентифікації студентів»

Виконав: студент групи МгІТ 1-20  
спеціальності 122 Комп'ютерні науки  
освітньої програми Комп'ютерні науки

*Дмитро ЄГОРОВ*

Керівник: к.т.н., доц. Тетяна АСТІСТОВА

Рецензент: д.т.н., проф. Володимир ЩЕРБАНЬ

Київ 2021

# КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ТЕХНОЛОГІЙ ТА ДИЗАЙНУ

## КАФЕДРА КОМП'ЮТЕРНИХ НАУК

Спеціальність 122 Комп'ютерні науки

Освітня програма Комп'ютерні науки

### ЗАТВЕРДЖУЮ

Завідувач кафедри КНТ

проф.Щербань В.Ю.

“ \_\_\_\_\_ ” \_\_\_\_\_ 2021 року

## З А В Д А Н Н Я

### НА ДИПЛОМНУ МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТУ

*Єгорову Дмитру Сергійовичу*

- 1. Тема роботи** Розробка QR кода для веб -додатку «Ідентифікації студентів», науковий керівник роботи *Астімова Тетяна Іванівна, к.т.н., доцент*, затверджені наказом вищого навчального закладу від “04” 10 2021 року №286
- 2. Строк подання студентом роботи** 17.12.2021р.
- 3. Вихідні дані до роботи** Розробка кафедри комп'ютерних наук. *Впровадження технології «Система ідентифікації студентів» для відзначення присутності студентів на парах в рамках існуючого навчального закладу*
- 4. Зміст дипломної роботи:** Вступ; Розділ 1. Теоретичні дослідження та використані технології; Розділ 2. Архітектура проекту та компоненти; Розділ 3. Програмне забезпечення;
- 5. Консультанти розділів дипломної магістерської роботи**

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Вступ	<i>Тетяна АСТІСТОВА к.т.н., доц.</i>		
Розділ 1	<i>Тетяна АСТІСТОВА к.т.н., доц.</i>		
Розділ 2	<i>Тетяна АСТІСТОВА к.т.н., доц.</i>		
Розділ 3	<i>Тетяна АСТІСТОВА к.т.н., доц.</i>		

- 6. Дата видачі завдання** 10.2021 р.

## КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломної магістерської роботи	Терміни виконання етапів	Примітка про виконання
1	Вступ	5.10.2021	
2	Розділ 1 Теоретичні дослідження та використані технології	25.10.2021	
3	Розділ 2 Архітектура проекту та компоненти	30.10.2021	
4	Розділ 3 Програмне забезпечення	10.11.2021	
5	Висновки	10.11.2021	
6	Оформлення дипломної магістерської роботи (чистовий варіант)	15.11.2021	
7	Здача дипломної магістерської роботи на кафедру для рецензування (за 14 днів до захисту)	20.11.2021 до 10.12.21	
8	Перевірка дипломної магістерської роботи на наявність ознак плагіату (за 10 днів до захисту)	25.11.2021 до 14.12.21	
9	Подання дипломної магістерської роботи на затвердження завідувачу кафедри (за 7 днів до захисту)	30.11.2021 до 17.12.21	

Студент

\_\_\_\_\_

*Дмитро Єгорov*

Науковий керівник роботи

\_\_\_\_\_

Тетяна АСТІСТОВА

Директор НМЦУПФ

\_\_\_\_\_

Олена ГРИГОРЕВСЬКА

## АНОТАЦІЯ

**Єгоров Д.С. Розробка QR кода для веб- додатку «Ідентифікації студентів» – Рукопис.**

Дипломна магістерська робота за спеціальністю 122 - «Комп'ютерні науки». – Київський національний університет технологій та дизайну, Київ, 2021 рік.

Дипломна робота присвячена спрощенню та вдосконаленню існуючої системи ідентифікації присутності студентів на парах. Впродовж роботи було впроваджено WEB-додаток, мобільний-додаток та створено базу даних для взаємодії студенту з викладачем.

Для досягнення результату були проаналізовані та обрані WEB та мобільні додатки, інтернет-сервер, генератор QR коду.

Дипломний проект демонструє повну взаємодію між студентами та викладачем, що дає комплексне уявлення про технології обміну даними та сучасної візуалізації даних.

*Ключові слова: QR код, система ідентифікації, WEB-додаток, мобільний додаток.*

## ANNOTATION

**DMYTRO YEHOROV Development of QR-code for the web application «Identification Student».**

Diploma master's degree work on speciality 122 are «Computer Science». It is the Kiev a national university of technologies and design, Kiev, 2021 year.

Thesis is devoted to the simplification and improvement of the existing system of identifying the presence of students in pairs. During the work, a WEB-application, a mobile-application were introduced and a database for student-teacher interaction was created.

To achieve the result, WEB and mobile applications, an Internet server, and a QR code generator were developed.

The diploma project demonstrates the full interaction between students and teachers, which gives a comprehensive understanding of data exchange technologies and modern data visualization.

*Keywords: QR code, identification system, WEB-application, mobile application.*

## ЗМІСТ

<b>ВСТУП</b> .....	7
<b>Розділ 1 Теоретичні дослідження та використані технології</b> .....	10
1.1 Поняття про WEB-технології.....	10
1.2 Мови програмування.....	11
1.3 Бібліотека React.JS для написання WEB додатку.....	13
1.4 Бібліотека для управління сховищем Redux.....	17
1.5 Бібліотека ReactNative для написання мобільних додатків.....	22
1.6 Концепція QR коду.....	26
1.7 Висновок до першого розділу.....	34
<b>Розділ 2 Архітектура проекту та компоненти</b> .....	35
2.1 Головна мета та архітектура додатку.....	35
2.2 Розробка UI/UX для WEB ті мобільних додатків.....	35
2.3 Використовування AJAX для створення запитів на сервер.....	36
2.4 Середовище для розробки WEB-версії додатку.....	40
2.5 Середовище для розробки мобільної версії додатку.....	43
2.6 Висновок до 2 розділу.....	49
<b>Розділ 3 Програмне забезпечення</b> .....	50
3.1 Архітектура додатку.....	50
3.2 Аналіз інтерфейсу користувача.....	52
3.3 Прототипування інтерфейсу користувача.....	53
3.4 API запити.....	57
3.5 Форма авторизації до системи.....	58
3.6 Сторінка профілю викладача.....	59
3.7 Сторінка профілю студента.....	61
3.8 Висновок до 3 розділу.....	62
<b>ВИСНОВКИ</b> .....	63
<b>СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ</b> .....	64

## ВСТУП

**Актуальність.** У зв'язку з пандемією коронавірусу тема QR кодів є дуже актуальною. В першу чергу, це безконтактність, яка дозволяє забезпечити додаткову безпеку здоров'я. По друге, це більш точніший спосіб відстежування та запису даних. Також ця система дуже гарно вписується до тенденції розвитку діджиталізації у нашій країні.

**Мета дослідження.** Розробити новий підхід по вдосконаленню існуючої системи ідентифікації присутності студентів на парах

створення WEB та мобільних додатків для вдосконалення та поліпшення існуючої системи відзначення присутності студентів під час пар у навчальному закладі. Завдання - розробити WEB інтерфейси та впровадити систему сканування QR кодів у додатках для відстеження, аналізу та збору статистики відвідувань пар студентами університетів.

**Об'єкт дослідження.** Об'єктом дослідження є існуюча система визначення присутності студентів на парах засобами голосового обліку та ручним записом до журналу відвідувань викладача.

**Предмет дослідження.** Розробка WEB та мобільного додатка для вдосконалення та поліпшення існуючої системи відзначення присутності студентів під час пар у навчальному закладі. Завдання - розробити WEB інтерфейси та впровадити систему сканування QR кодів у додатках для відстеження, аналізу та збору статистики відвідувань пар студентами університетів

**Методи дослідження.** Для реалізації поставленої задачі були проаналізовані WEB та мобільні додатки, інтернет-сервери, генератор QR коду, програмне забезпечення та використання літерно-цифрового кодування. Стек обраних технологій для реалізації: Frontend розробка із використанням UX/UI/IA-проекування, мови розмітки, мови програмування, реактивності, клієнт-серверної взаємодії.

Дослідницька оригінальність і теоретичне значення результатів, що отримуються. Оригінальністю даного дослідження є виробництво програмного комплексу, що об'єднає у собі сучасні технології сканування даних та представлення їх у додатку з простим інтерфейсом та легким у використанні.

**Наукова новизна та практичне значення отриманих результатів.** Використання `Qrcode.react` для швидкого генерування QR кодів та шифрування потрібних даних в них. Використовуючи бібліотеку `Qrcode.react` можна постійно генерувати нові коди в онлайн режимі та шифрувати в них дані про дату, час, назву поточної пари, ім'я викладача, назву групи та інформацію про студентів. Усі дані передаються швидко та безперешкодно, дозволяючи викладачу миттєво моніторити необхідну інформацію о студентах.

Дана технологія дає можливість викладачу зберегти час при перевірці наявності студентів на парах, бачити час прибуття студента на пару, аналізувати відсоток відвідувань занять кожним студентом.

**Результати дослідження.** Використання новітніх WEB-технологій для забезпечення гарно оптимізованої кросплатформності такої важкої системи є дуже важливим, так як дає перспективу використання додатку на більшості комп'ютерних та мобільних приладів.

Надійність та робоча стійкість є дуже вагомими, тому для реалізації додатку обрано найсучасні та новітні фреймворки й бібліотеки. В ході R&D (дослідження та розробки) були обрані переваги, що надає технологія `React.js`, `Redux`, та `ReactNative`.

Система контролю відвідувань складається з трьох частин – написання клієнтського WEB-інтерфейсу, клієнтського мобільного інтерфейсу та WEB-інтерфейсу для адміністраторів. Також обробка даних які надходять з серверу на всіх трьох додатках. Також перехід від паперових журналів забезпечить збереження нашої природи та допоможе сучасній проблемі вирубці лісів, у масштабах великої країни ця ідея спроможе допомогти зберегти значну частину дерев.



**Теоретична цінність.** Теоретична цінність роботи полягає в здійсненню аналізу затраченого часу на облік присутніх студентів на парі, аналізу випадків невірного визначення присутності студентів із-за людських помилок. Було запропоноване автоматизоване та сучасне вирішення цих проблем.

**Практична цінність.** Практична значимість полягає у виявленні методів спрощення існуючої системи навчання, а саме відзначення присутності студентів на парах у навчальних закладах, також ця розробка дозволяє зменшити витрати на це відзначення та автоматизувати цей процес.

**Апробація проєкту.** Цей проєкт зайняв друге місце на конкурсі стартапів КНУТД .

Результати роботи були апробовані на міжнародній конференції та в статті.

1. Астісова Т.І. , Розробка системи ідентифікації студентів / Т.І. Астісова, Д.С. Егоров //Тези V Міжнародної науково-практичної конференції «Мехатронні системи: інновації та інжиніринг – «MSIE-2021» К. КНУТД , 4 листопада 2021р. - С. 157-158

2. Астісова Т. І., Егоров Д.С., Розробка QR-коду для веб додатку «Система ідентифікації студентів»//Т.І. Астісова, Д.С.Егоров// Інформаційні технології в науці, виробництві та підприємстві: зб. наук. праць молодих вчених, аспірантів, магістрів кафедри комп'ютерних наук та технологій. – К. : Освіта України, 2021 р. . – С. 214 – 217

## Розділ 1 ТЕОРЕТИЧНІ ДОСЛІДЖЕННЯ ТА ВИКОРИСТАНІ ТЕХНОЛОГІЇ

### 1.1 Поняття про WEB-технології

Веб-технологія — це створення та використання механізмів, які дозволяють різним комп'ютерам і пристроям обмінюватися ресурсами та розповсюджувати їх. Веб-технології є будівельними блоками інфраструктури будь-якої ефективної комп'ютерної мережі.



Рис. 1.1. Веб-Технологія

Яка користь від веб-технологій?

Комп'ютери та інші мережеві пристрої потребують зв'язку.

Механізм повинен давати можливість комп'ютеру зв'язуватися з іншим комп'ютером у тій самій мережі чи іншій мережі.

Механізм повинен гарантувати, що повідомлення переміщається від відправника до одержувача, дозволяючи одержувачу отримати повідомлення, надіслати зворотний зв'язок і підтвердити отримання або невдачу зв'язку.

Різноманітні веб-технології є життєво важливими для функціонування та успіху багатьох підприємств.

До них належать програми онлайн-планування зустрічей, веб-сайти та спосіб спілкування клієнтів із представниками. Крім того, веб-технологія дає можливість підприємствам збирати дані про своїх клієнтів, щоб додатково налаштувати свої послуги.

Веб-технології розвиваються за допомогою мов розмітки.

Мови розмітки, такі як HTML, CSS і XML, є частиною веб-технологій.

Ці мови розповідають комп'ютерам у тексті, як форматувати, розташовувати та стилізувати веб-сторінки та програми.

Два типи мов розмітки включають процедурну розмітку та описову розмітку.

До додаткових типів мов належать CGI та HTTP.

## **1.2 Мови програмування**

Мови програмування включають Perl, C#, Java та Visual Basic .NET. Ці мови використовуються веб-розробниками для створення веб-сайтів і програм. Кожна мова має плюси і мінуси, і більшість розробників знають кілька різних типів, які допомагають їм досягати своїх цілей.

HTML: Основа будь-якого веб-сайту.

HTML (мова розмітки гіпертексту) — це клей, який скріплює кожен веб-сайт. Як будувати будинок, ви завжди будете міцний фундамент. Для будь-якого сайту HTML є основою. HTML є мовою з відкритим вихідним кодом (тобто нікому не належить), яку легко вивчити, і для початку її використання не потрібні химерні (або дорогі!) пакети. Все, що вам потрібно, це щось для введення тексту, наприклад, Блокнот Windows, а також багато часу та терпіння.

HTML працює в системі «тегів», де кожен тег впливає на вміст, розміщений у цьому тегу;

<TAG>На що впливає тег</TAG>.

CSS (каскадні таблиці стилів)

CSS є відносно новою мовою, розробленою для розширення обмежених властивостей стилю HTML.

Легкий у освоєнні та застосуванні, CSS – це чудовий спосіб керувати стилем вашого сайту, таким як стилі тексту, як-от розмір, колір та шрифт.

CSS також може бути розміщений всередині сторінки HTML або в окремих файлах.

### Java Script

JavaScript — це об'єктно-орієнтована мова програмування, розроблена для того, щоб зробити веб-розробку простішою та привабливішою. У більшості випадків JavaScript використовується для створення адаптивних інтерактивних елементів для веб-сторінок, які покращують роботу користувачів.

Такі речі, як меню, анімація, відеопрогравачі, інтерактивні карти та навіть прості ігри в браузері можна створювати швидко й легко за допомогою JavaScript.

Справжня перевага розміщення всіх властивостей стилю для всього сайту в одному файлі CSS полягає в тому, що ви можете редагувати цей файл, щоб внести зміни на весь сайт, замість того, щоб переглядати кожен файл HTML по одному. З цієї причини це, мабуть, найкорисніша веб-технологія і, безумовно, одна з моїх улюблених.

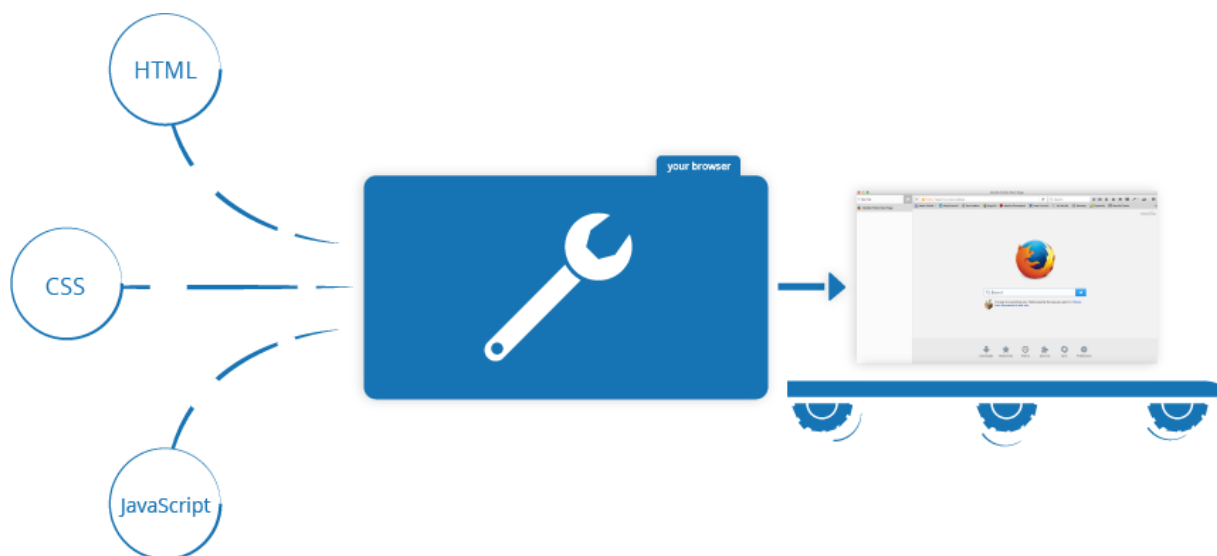


Рис. 1.2. JavaScript

Коли ви завантажуєте сторінку в браузері, ви запускаєте ваш код (HTML, CSS і JavaScript) всередині середовища (всередині вкладки браузера). Це ніби фабрика бере сировину (якийсь код) і видає продукцію (веб-сторінку).

Код JavaScript виконується JavaScript-движком браузера, після того, як код HTML і CSS був оброблений і сформований на веб-сторінку. Це гарантує, що структура та стиль сторінки вже сформовані на момент запуску JavaScript.

Це добре, оскільки часто використання JavaScript полягає в динамічній зміні HTML і CSS з метою оновлення інтерфейсу користувача за допомогою Document Object Model API (як згадувалося вище). Якби запуск JavaScript здійснювався насамперед завантаження HTML і CSS, це призвело б до виникнення помилок.

### **1.3 Бібліотека ReactJS для написання WEB додатку**

React або ReactJS — це дуже популярна форма передньої бібліотеки JavaScript, яка широко використовується для створення інтерфейсів користувача. Це особливо корисно для розробки односторінкових програм. ReactJS був розроблений інженером-програмістом Facebook на ім'я Джордан Волке і привів до значних удосконалень в інтерфейсі Facebook і зараз використовується кількома відомими компаніями, включаючи Instagram, New York Times, Apple, PayPal, Airbnb і Netflix.

React дозволяє безболісно створювати інтерактивні інтерактивні інтерфейси. Створюйте прості представлення для кожного стану у вашій програмі, і React ефективно оновлюватиме й відобразатиме потрібні компоненти, коли ваші дані змінюються.

Декларативні уявлення роблять ваш код більш передбачуваним і легшим для налагодження.

Створюйте інкапсульовані компоненти, які керують своїм власним станом, а потім компонуєте їх, щоб створити складні користувацькі інтерфейси.

Оскільки логіка компонентів написана на JavaScript, а не на шаблонах, ви можете легко передавати розширені дані через свою програму і зберігати стан поза межами DOM.

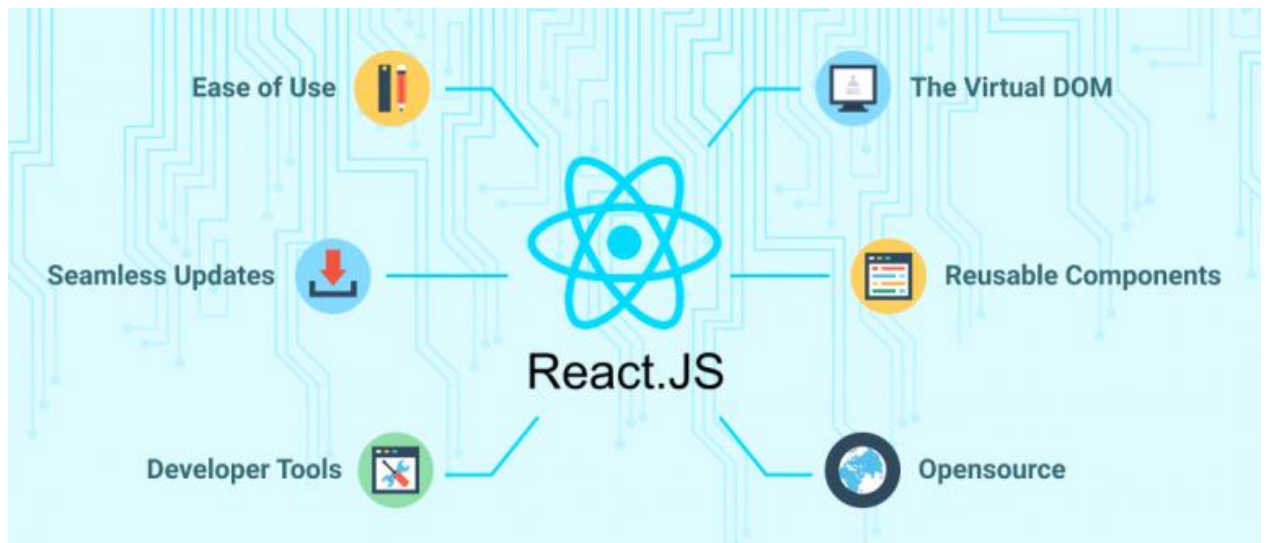


Рис. 1.3. ReactJS

ReactJS пропонує кілька переваг для розробників:

- Простота створення динамічних веб-додатків;
- Допомогає заощадити час, допомагаючи повторно використовувати компоненти, розроблені для інших додатків;
- Полегшує налаштування машиночитаних кодів;
- Менша ймовірність помилок під час кодування через меншу кількість задіяного коду;
- Віртуальний DOM забезпечує кращу продуктивність;
- Забезпечує зручну для SEO структуру;
- Легко навчатися та впроваджувати в проекти;
- Забезпечує гнучкість бібліотеки JavaScript;
- Легко тестувати та налагоджувати коди за допомогою вбудованих інструментів.

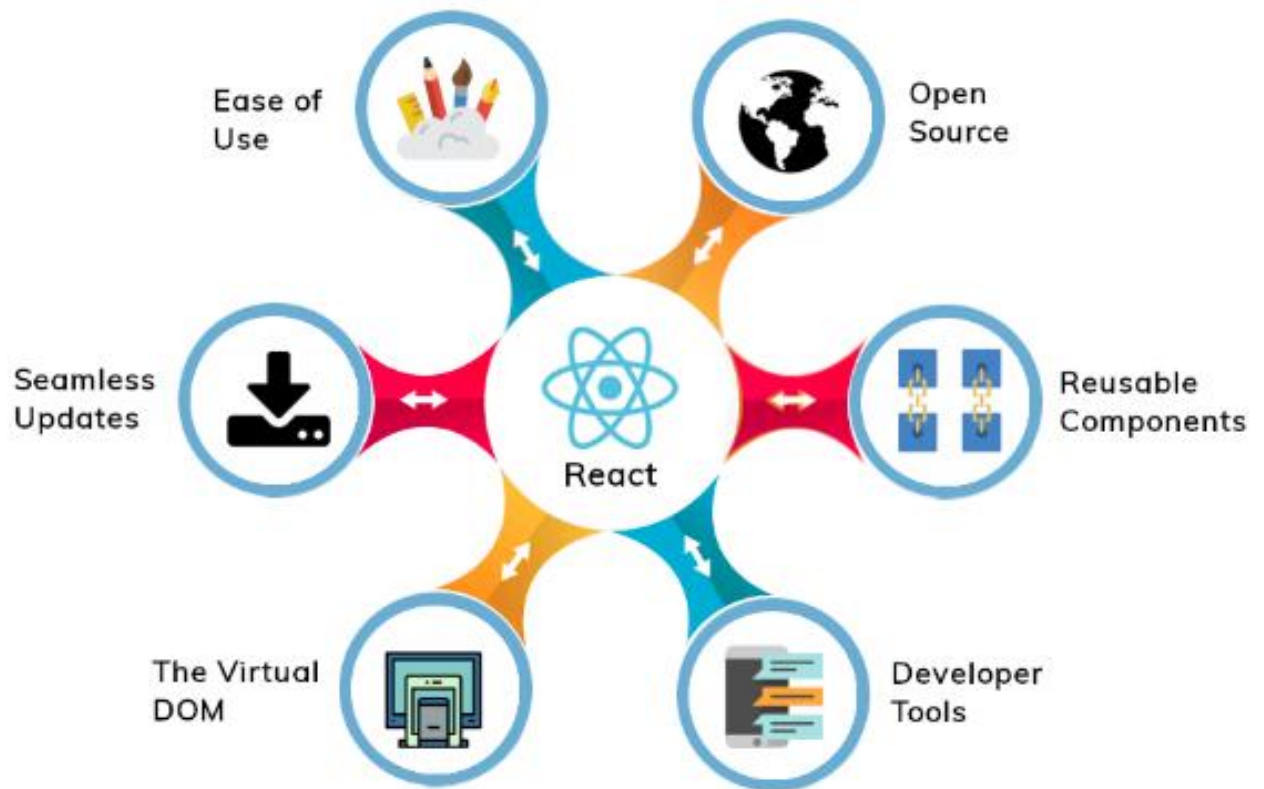


Рис. 1.4. Переваги ReactJS

Життєвий цикл компонента ReactJS описує різні етапи існування компонента:

- Ініціалізація – на цьому етапі компонент будується разом із заданими Props;
- Монтування – це процес, який відбувається, коли компонент вставляється в DOM;
- Оновлення – на цьому етапі компонент оновлюється і програма перефарбовується;
- Демонтування – цей етап допомагає видалити небажані компоненти з DOM.

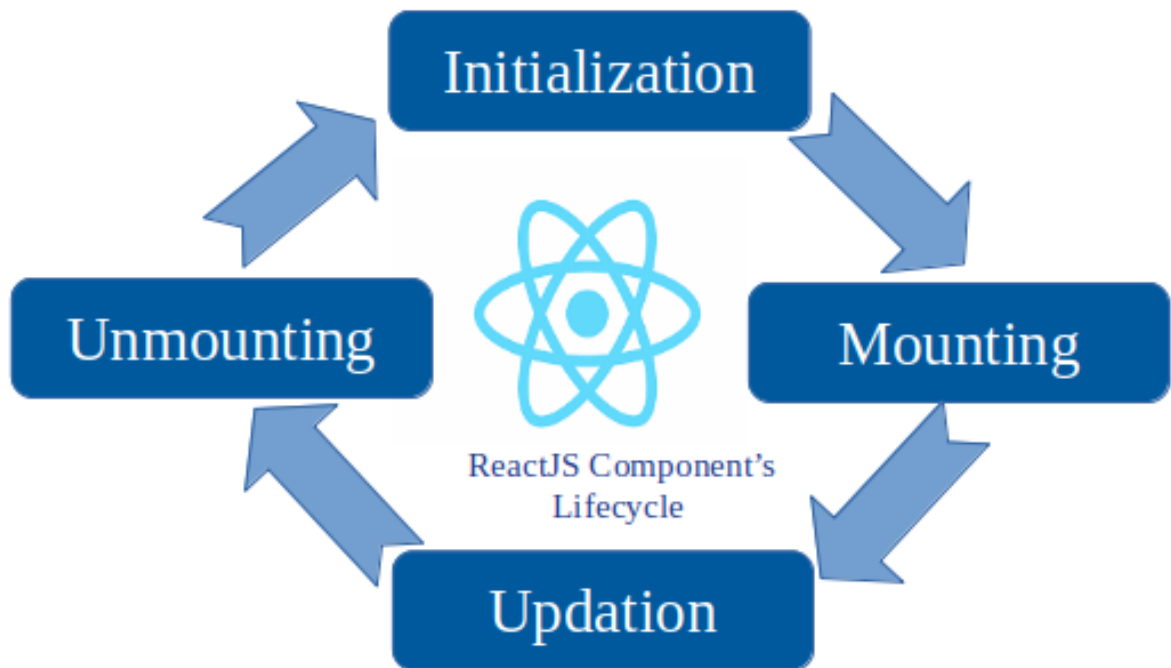


Рис. 1.5. Життєвий цикл компонента ReactJS

Також ми можемо використовувати хуки які є однією з останніх функцій React, яка була представлена у версії 16.8. Вони є зворотно сумісні. Вони не містять жодних критичних змін. Написання хука дуже корисно, коли ви бажаєте написати компонент функції і хочете додати до нього якийсь стан. Це можна легко зробити, використовуючи Hook всередині функціонального компонента.

Для оптимізації продуктивності React додатку будемо запобігати таких речей:

- Використання Memo();
- React. PureComponent;
- Використання ESLint-plugin-React;
- Налаштування CSS;
- Використання стиснення Brotli;



Обробники подій React вирішують проблеми сумісності за допомогою кросбраузерної оболонки React – Synthetic Event. Синтетичні події чудові, оскільки мають інтерфейс, подібний до нативних подій.

Тому саме цю бібліотеку JavaScript було обрано для створення WEB-додатку, на основі саме таких відмінностей React:

- Архітектура;
- DOM;
- Розробник;
- Відображати;
- Прив'язка даних;
- Налаштування;
- Використання;
- Продуктивність.

#### **1.4 Бібліотека для управління сховищем Redux**

Redux — популярна бібліотека JavaScript для керування станом вашої програми.

Стан програми — це як глобальний об'єкт, який містить інформацію, яку ви використовуєте для різних цілей пізніше в додатку (наприклад, прийняття рішень про те, які компоненти відображати і коли, відтворення збережених даних тощо).

Прикладом, з яким ми часто стикаємося, є відображення індикатора завантаження під час завантаження сторінки. У цьому випадку, якби ми використовували сховище лише для цієї мети, об'єкт стану зберігав би логічне поле незалежно від того, завантажено сторінку, і ми використовували б це поле для перемикання відображення індикатора завантаження.

Інший приклад: якби ми створили програму для соціальних мереж, ми б зберігали кілька об'єктів і масивів у стані нашої програми — поточну інформацію про користувача, статус HTTP, тост-повідомлення, сповіщення,

які буде отримувати користувач, і якщо вони будуть відтворені в декількох місцях у програмі — поточні публікації та історії з профілів, за якими користувач стежить тощо.

Великі програми мають великі стани додатків, і керувати ними стає все більш незручним у міру зростання вашої програми. Крім того, у вас можуть бути компоненти, які використовують ті самі дані, але розміщуються випадковим чином у дереві DOM.

Ось чому нам потрібні бібліотеки державного управління, такі як Redux. Один з шаблонів, яким дотримується Redux, називається «Єдине джерело правди», що означає, що у нас є лише одне місце (так зване Store), де ми зберігаємо єдиний стан для всієї програми.

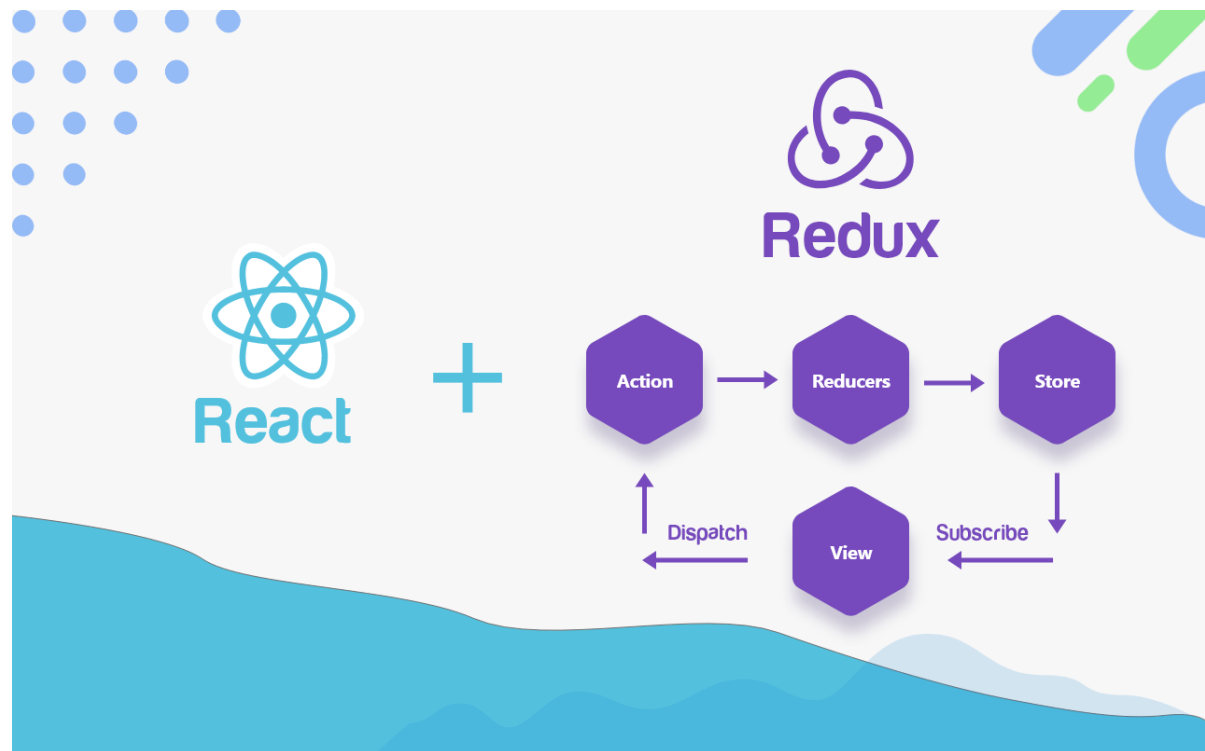


Рис. 1.6. Redux

Іншими словами, один додаток — один магазин — одна держава.

Однак компоненти в React або інших фреймворках також можуть містити власний внутрішній стан. Як правило, ви не хочете, щоб буквально все було в стані програми.

Інший шаблон, якому дотримується Redux, називається «незмінністю». І ви досить часто зустрінете цей термін в інших фреймворках і бібліотеках.

Коротше кажучи, незмінність означає, що ми не змінюємо об'єкт стану та його властивості безпосередньо. Замість цього ми створюємо новий об'єкт, перераховуємо новий стан програми та оновлюємо його за допомогою нашого щойно створеного об'єкта. Ми хочемо залишити наш старий державний об'єкт недоторканим.

### Три будівельні блоки Redux

Redux складається з 3 основних частин:

1. Actions
2. Reducers
3. Store

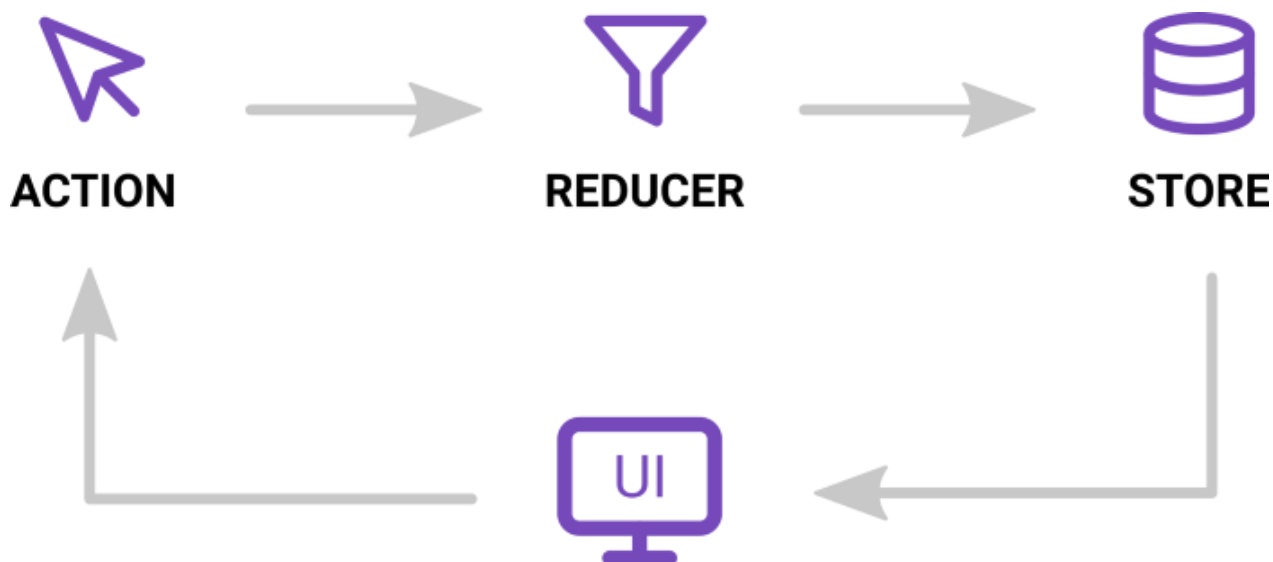


Рис. 1.7. Три основні частини Redux

Store - утримувати стан програми.

Сховище насправді є об'єктом, а не класом, хоча спочатку він може здаватися таким. Він також містить кілька додаткових речей, крім стану вашої програми (наприклад, функції та інші об'єкти).

Хоча теоретично можна створити кілька магазинів, це суперечить шаблону, якого слідує Redux.

Пам'ятайте, ми створюємо лише один магазин для кожної програми!

Ми можемо підписатися, щоб слухати події, коли оновлюється магазин. У додатку без React ми можемо використовувати цю підписку для оновлення інтерфейсу, наприклад, як ми будемо робити у нашій програмі.

Стан у Redux у формі об'єкта JavaScript і часто називають «деревом станів». Ви можете помістити в нього будь-які значення, які хочете зберегти, і ви можете вкладати їх стільки, скільки вам потрібно.

Actions — це прості об'єкти JavaScript, які описують ЩО сталося, але не описують, ЯК змінюється стан програми.

Ми просто відправляємо (відсилаємо) їх у наш store, коли хочемо оновити стан нашої програми.

Важливо пам'ятати, що Redux вимагає, щоб наші об'єкти дії містили поле типу. Це поле використовується для опису дії, яку ми надсилаємо, і зазвичай це має бути константа, яку ви експортуєте з файлу.

Усі інші поля в об'єкті дії є необов'язковими і вирішувати вам.

Ще один термін, з яким ви часто зустрічаєтеся, – це творці дій.

В основному це функції, які генерують і повертають прості об'єкти JavaScript.

Вони використовуються для того, щоб ми могли «вставляти» динамічні дані в наші дії (або іншими словами генерувати дії з динамічними даними).

```
{ type: ADD_NOTE, title: 'Some Title', content: 'This is an action object' }
```

Рис. 1.8. Приклад запити в Store

Ще один термін, з яким ми часто зустрічаємось, – це Action Creators.

В основному це функції, які генерують і повертають прості об'єкти JavaScript.

Вони використовуються для того, щоб ми могли «вставляти» динамічні дані в наші дії (або іншими словами генерувати дії з динамічними даними).

Іншими словами, редуктор обчислить новий стан нашої програми на основі дії (і її типу), яку ми відправили.

У реальному світі ваші редуктори, швидше за все, будуть дуже складними. Щоб впоратися зі складністю редукторів, ми розбиваємо їх на кілька простіших редукторів, а потім об'єднуємо їх із допоміжною функцією Redux, яка називається `combineReducers`.

Основний редуктор умовно називається «Root Reducer».

#### Потік даних

Хоча спочатку це виглядає дещо складним, потік даних у Redux насправді досить простий.

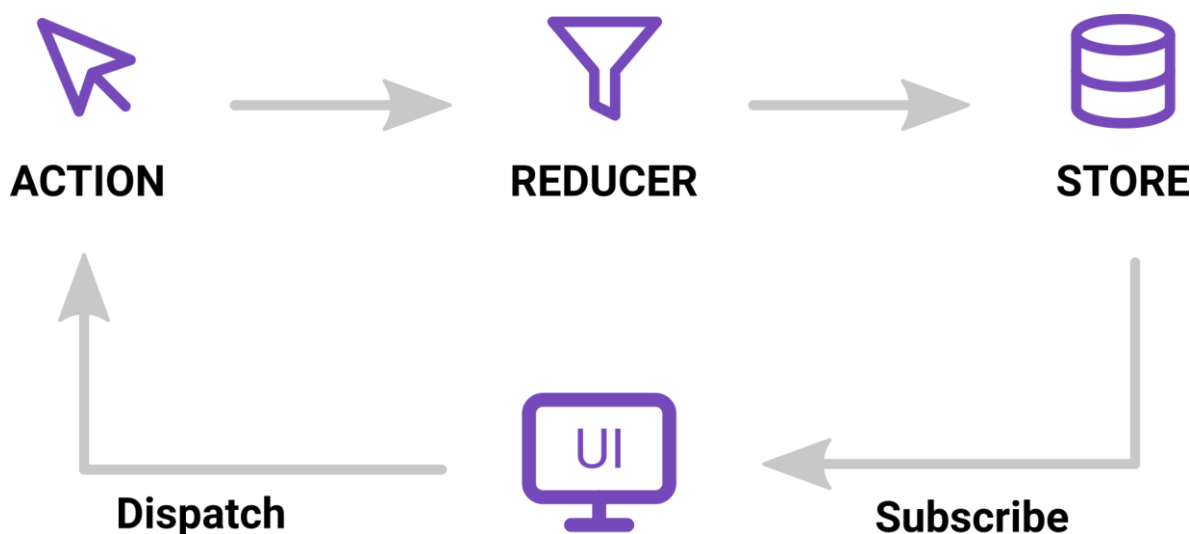


Рис. 1.9. Потік даних у Redux

Скажімо, користувач ініціює подію (наприклад, натискає кнопку «Додати примітку»), і стан програми оновлюється (тобто нова примітка вставляється в стан програми). Ось що відбувається під капотом:

1. Функція обробника натискання кнопки надсилає дію до магазину за допомогою методу `store.dispatch()`.
2. Redux передає відправлену дію в редуктор
3. Сховище зберігає новий стан, повернутий редуктором
4. Оскільки ми підписалися на магазин, буде викликана функція, яку ми надали, і вона відповідно оновить інтерфейс користувача (тобто додасть нову примітку до списку нотаток)

## 1.5 Бібліотека React Native для написання мобільних додатків

React Native — це фреймворк, створений Facebook, який дозволяє одночасно розробляти нативні мобільні додатки для Android та iOS.

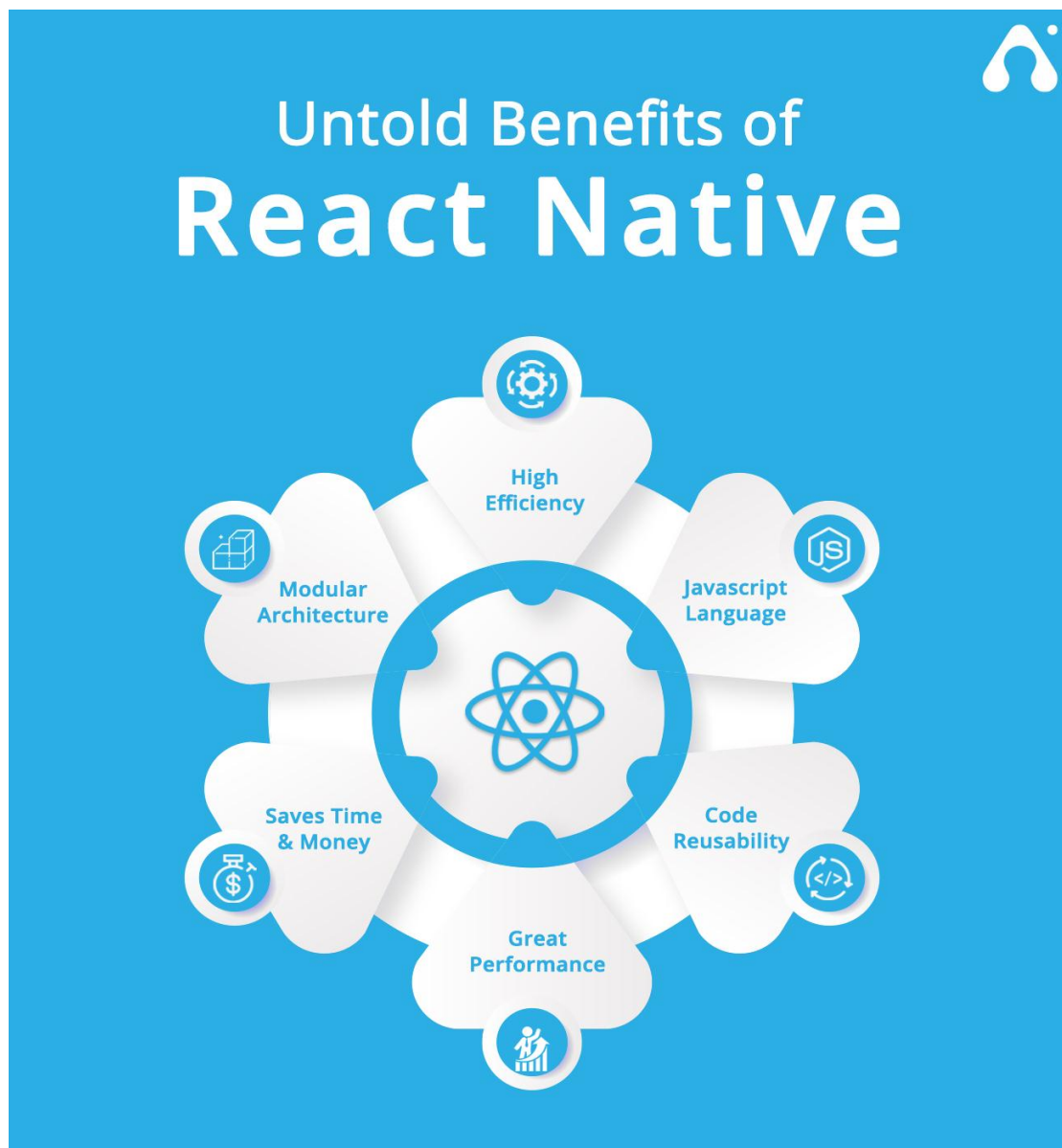


Рис. 1.10. React Native

Він надає ряд компонентів для створення користувацького інтерфейсу, які перетворюються на власні елементи керування, що відповідають даній платформі. Наприклад, коли ми хочемо відобразити рядок тексту, ми будемо використовувати компонент `Text` у React Native. Тоді фреймворк гарантуватиме, що програма Android використовує рідний елемент керування `TextView`, а iOS — елемент керування `UITextView`.

Початок фреймворка React Native датується 2012–2015 роками, коли творці Facebook хотіли випустити його версію мобільного додатка. Спочатку це намагалися зробити за допомогою технологій розробки веб-програм. Однак їх виявилось недостатньо, враховуючи зростання популярності мобільних додатків і важливість користувацького досвіду. Забезпечення хорошого UX для Facebook у мобільній версії стало однією з їхніх головних цілей.

Поворотним моментом стало відкриття способу створення вбудованих елементів керування iOS за допомогою JavaScript. Потім був організований хакатон. Це дозволило розвинути цю ідею та розробити прототип React Native. У 2015 році, Facebook випустив першу версію React Native і показав, що Ads Manager був першим додатком, повністю написаним на React Native. Незабаром з'явилися інші додатки.

#### Код в JavaScript

Той факт, що весь ваш проект написаний на JavaScript, є найбільш привабливим аспектом React Native. Для розробника це сильна перевага. Протягом кількох років JavaScript стабільно вважався однією з найбільш широко використовуваних і популярних мов програмування. Це означає, що якщо ви знаєте JavaScript, ви зможете швидко опанувати React Native. Веб-розробники можуть використовувати свої наявні навички JavaScript для створення мобільних додатків.

#### Використовує React

React Native використовує React для створення нативних мобільних додатків, як видно з назв. Це друга найважливіша перевага використання React Native для мобільної розробки. React досі залишається найбільш широко використовуваною бібліотекою JavaScript для створення інтерфейсних програм. Вибирати React Native для розробки мобільних додатків цілком природно, якщо ви вже є розробником React або маєте команду розробників

React. На цьому етапі крива навчання надзвичайно мала. Це також означає, що React Native включає всі новітні функції React.

### За підтримки Facebook

Незважаючи на те, що React Native був відкритим у 2015 році, Facebook продовжує підтримувати його. Група розробників працює над покращенням React Native. Минулого року команда React Native опублікувала движок Hermes Engine, який перебудував основні розділи React Native. Hermes — це двигун JavaScript з відкритим кодом для Android, який спеціалізується на запуску програм React Native. Це значно покращує швидкість React Native на Android.

Усі ці ініціативи надходять від команди Facebook, враховуючи високий рівень довіри спільноти. Багато продуктів Facebook також використовують React Native. React Native використовується, серед іншого, для кодування Marketplace у додатку Facebook.

### React Native Debugger

React Native Debugger — це фантастичний окремий інструмент для налагодження програм React Native. Він також поставляється з інструментами React Inspector і Redux Dev з коробки. Цей інструмент забезпечує значно кращий досвід налагодження, ніж стандартне налагодження Chrome. Якщо ви створюєте програми React Native.

### Ехро

Ехро значно допоміг у вдосконаленні та зручності використання React Native для розробників. Ехро дозволяє створювати програми React Native, не торкаючись жодного нативного коду. Це оболонка React Native, яка дозволяє створювати програми, не турбуючись про рідний код або інструменти, такі як Xcode та Android Studio. Це ідеально підходить для тих, хто ніколи раніше не працював з мобільним пристроєм. Ехро надає багато API, які допоможуть у розробці проекту.



Ехро — це найшвидший спосіб створити додаток React Native, але при цьому він пропонує фантастичний досвід розробки.

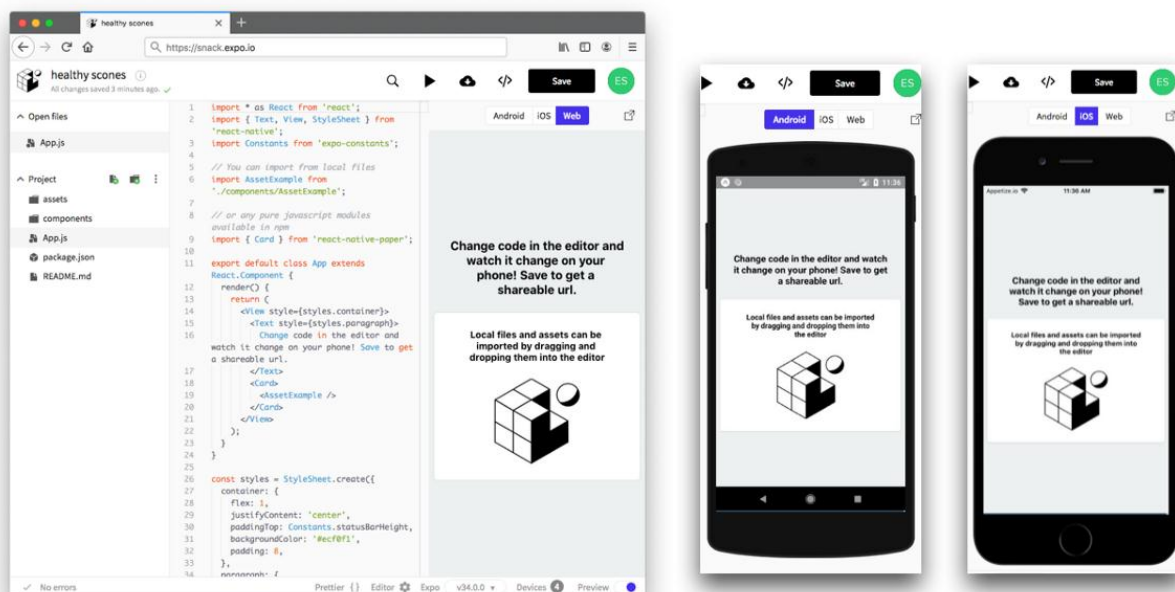


Рис. 1.11. Ехро

### Швидке оновлення

Швидке оновлення — це функція React Native, яка дає вам майже миттєвий зворотній зв'язок, коли ваші компоненти React оновлюються. Він увімкнений за замовчуванням, але ви можете вимкнути його в меню розробника React Native, перемкнувши «Увімкнути швидке оновлення». Більшість змін мають бути видимі протягом секунди або двох, якщо ввімкнено швидке оновлення. Це значно прискорює процес розробки.

### Code Once

Кросплатформні можливості React Native є основною перевагою фреймворка. Щоб створювати мобільні програми за допомогою React Native, вам не потрібно знати Objective-C, Swift, Java або Kotlin. Ви можете розробити кросплатформну програму за допомогою Javascript і JSX. Майже весь код використовується на iOS та Android. Зі скромними коригуваннями для покращення кінцевого продукту на обох платформах.

Для міжплатформної підтримки одного програмного забезпечення, вам не потрібно мати кілька команд або кодових баз.

Натомість у нас є єдина команда та кодова база, яка працює як на iOS, так і на Android.

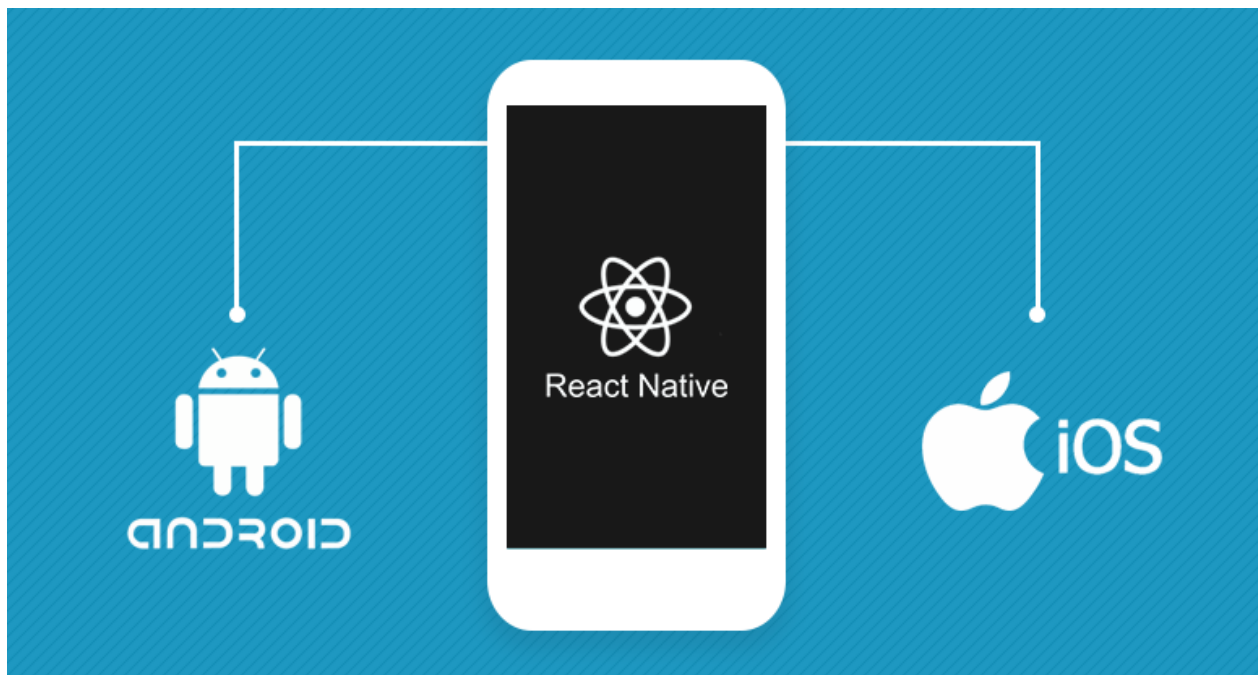


Рис. 1.12. React Native для ios та android

Крім того, створені нами програми є нативними, а не онлайн-програмами. Основні віджети React Native є повністю рідними компонентами, що забезпечують безперебійну роботу для користувача. У порівнянні з деякими його конкурентами, які створюють веб-перегляди, це має значну різницю.

### 1.6 Концепція QR коду

Що таке QR-код?

QR-код є різновидом штрих-коду. Скануючи його, ви отримуєте доступ до закодованої в ньому інформації.

У стандартних штрих-кодах інформація кодується в ширині і відстані між вертикальними лініями. У QR-кодах інформація закодована в розташуванні квадратів. У будь-якому випадку дані перетворюються на машиночитане

розташування візуальних елементів. А після сканування оптичним скануючим пристроєм дані повертаються до початкового вигляду.

Але три речі роблять QR-коди особливо особливими. Кількість даних, які вони можуть зберігати, швидкість їх зчитування та те, що практично всі наші телефони можуть миттєво й легко їх сканувати.



Рис. 1.13. QR код

Як працюють QR-коди?

QR-код — це штрих-код, який можна сканувати, закодований даними. Закодований означає перетворений у певну форму. У разі QR-кодів числові та буквено-цифрові символи, байти та кандзі перетворюються в унікальне двовимірне розташування квадратів. Коли оптичний сканер проходить над цими квадратами, він перетворює їх розташування назад у вихідну форму цих даних.



Рис. 1.14. Сканування QR коду

Ось приклад. Ми створюємо індивідуальні цифрові меню ресторанних столів на основі QR-коду для готельного бізнесу. Тому у нас є демонстраційне цифрове меню, щоб показати зацікавленим сторонам. Щоб продемонструвати, що ми можемо зробити, ми закодували URL-адресу демонстраційного меню в QR-код нижче.

У QR-коді розташування квадратів — або модулів даних, як їх називають — насправді є нашою URL-адресою. Його щойно було переведено з буквено-цифрового рядка URL-адреси в набір квадратів. Таким чином ви переходите від посилання до QR-коду. Потім сканер QR-коду переведе його назад на вихідну URL-адресу.

#### Частини QR-коду

Анатомія QR-коду здебільшого актуальна для тих, хто думає створити QR-код. Однак пам'ятайте про ризики безпеки QR-коду, пов'язані з безкоштовними онлайн-сервісами.

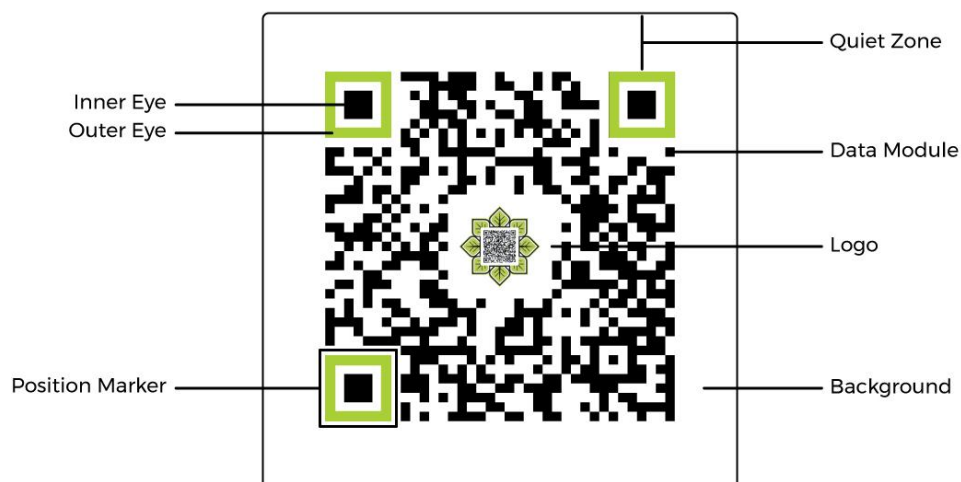


Рис. 1.15. Частини QR коду

Найважливішими частинами QR-коду є:

- Модуль даних. Це стандартна одиниця QR-коду. Зазвичай це чорний квадрат на білому тлі. Хоча кольори та контрастність можуть бути різними, чорне на білому є найоптимальнішим при створенні власного QR-коду. Розташування цих чорних квадратів, або модулів даних, становить більшість QR-коду.
- Маркер позиції. На кожному QR-коді є три маркери позиції. Вони складаються з внутрішнього і зовнішнього ока, що дозволяють сканерам і камерам швидко і точно визначати місцезнаходження модулів даних і напрямок сканування.
- Тиха зона. Це порожня область з усіх боків матриці модуля даних, яка містить усі модулі даних і маркери положення. Це дозволяє сканерам і зчитувачам оптично розташовувати, де починається і закінчується QR-код.

Яка інформація міститься в QR-кодi?

Існує три типи інформації, яку зберігає QR-код: розмір, рівень виправлення помилок і тип даних.

#### Розмір

QR-код може складатися з максимум 177 рядків і 177 стовпців, що становить 31 329 модулів даних. Однак більшість QR-кодів не такі великі.

Розмір QR-коду відповідає його версії. Найменший QR-код, який може бути, — це 21 рядок на 21 стовець, що є версією 1. 25x25 — це версія 2, і далі.

Вищезгаданий найбільший можливий QR-код, 177x177, є версією 40.

Також варто зазначити, що мінімальний розмір QR-коду зазвичай залежить від відстані сканування, а не від розміру даних.

#### Рівні виправлення помилок

Закодований у QR-код є одним із чотирьох рівнів виправлення помилок QR-коду. Чим вищий рівень корекції, тим більше пошкоджень може зазнати QR-код, при цьому його можна сканувати. Це як збережена резервна копія QR-коду. Чим нижчий рівень корекції, тим більше місця залишається для розміру та даних.

#### Тип даних

QR-коди можуть зберігати до 7089 цифрових символів або 2953 буквено-цифрових символів. Вони також можуть зберігати байти та кандзі, але вони рідше використовуються. Ці числа припускають найнижчий рівень виправлення помилок.

На практиці це означає, що використання QR-коду включає все, що використовує цифри, літери, розділові знаки та символи для спілкування. Візитки, QR-коди на столах у ресторанах, аутентифікація, реєстрація в готелях, вхід на веб-сайти, безконтактні платежі, цифрові винні карти, використання QR-коду в їжі тощо.

Для контексту кількість символів, яку може вмістити стандартний одновимірний штрих-код, становить від 20 до 100 символів. Ця здатність QR-кодів зберігати таку велику кількість інформації — і швидко надавати її — робить їх набагато кориснішими інструментами, ніж стандартні штрих-коди. Практично в кожній галузі.

Але чим більше символів ви зберігаєте в QR-коді, тим більшим і складнішим має бути QR-код, чи не так? Так і ні. Це різниця між статичними QR-кодами та динамічними QR-кодами.

Як працюють динамічні QR-коди?

Динамічний QR-код – це QR-код, закодована інформація якого може бути змінена після створення. Це тому, що інформація, закодована в динамічному QR-коді, є короткою URL-адресою переспрямування. Це означає URL-адресу, яка переспрямовує на іншу URL-адресу, або те, що називається цільовою URL-адресою.

На цільовій URL-адресі міститься вміст. Це може бути цифрова винна карта, навчальний план, контактна інформація, запрошення тощо. Цей вміст можна повністю змінити, не змінюючи QR-код. Крім того, саму цільову URL-адресу можна повністю змінити, не змінюючи QR-код. Оскільки в QR-коді закодовано URL-адресу переспрямування, а не цільову URL-адресу.

Це налаштування також дозволяє динамічно відстежувати сканування та використання QR-коду, що має вирішальне значення для будь-якої форми маркетингу QR-коду.

Використання динамічного та статичного QR-коду є одним із варіантів, які компанія робить під час створення QR-коду. І, враховуючи переваги використання динамічних QR-кодів, багато хто задається питанням, як працюють динамічні QR-коди. Це справедливе запитання, адже вони досить вражаючі. Вони також досить прості.

Як технічно працюють QR-коди

Хоча навчитися сканувати QR-код легко для вас, QR-код і сканер виконують трохи більше роботи.

Сканер QR-коду починається в нижньому правому куті QR-коду. Потім він переміщається вгору на два модулі даних одночасно, доки не дійде до маркера першої позиції. Потім він переміщує два модулі даних ліворуч і опускається вниз. Він повторює цей зигзагоподібний процес справа наліво, вгору-вниз, доки не буде охоплено кожен модуль даних.

Ось базовий шестиетапний опис того, як працює процес сканування.

1. Наведіть телефон на QR-код.
2. Сканер QR-коду в камері вашого телефону розпізнає три маркери позицій у QR-коді. Завдяки достатній тихій зоні ваш сканер тепер знає, де знаходяться краї QR-коду.
3. Сканер починається в нижньому правому куті, де він зустрічається з індикатором режиму. Ці чотири модулі даних вказують, до якого типу даних (числових, буквено-цифрових, байтових або кандзі) належать решта закодованих даних.
4. Далі сканер зустрічає індикатор кількості символів, який є наступними 8 модулями даних після індикатора режиму. Вони вказують, скільки символів становить загальна кількість закодованих даних.
5. Знаючи тип даних і довжину символів, сканер продовжує свій зигзагоподібний шлях уздовж модулів даних, поки не отримає всю закодовану інформацію та не досягне індикатора кінця.
6. Після зчитування кінцевого символу сканер продовжує свій шлях до модулів даних корекції помилок. У цих закодованих модулях є один із чотирьох рівнів виправлення помилок. Або скільки закодованих даних QR-коду зберігається у разі пошкодження коду.

Це допоможе вам уявити, як працює QR-код:



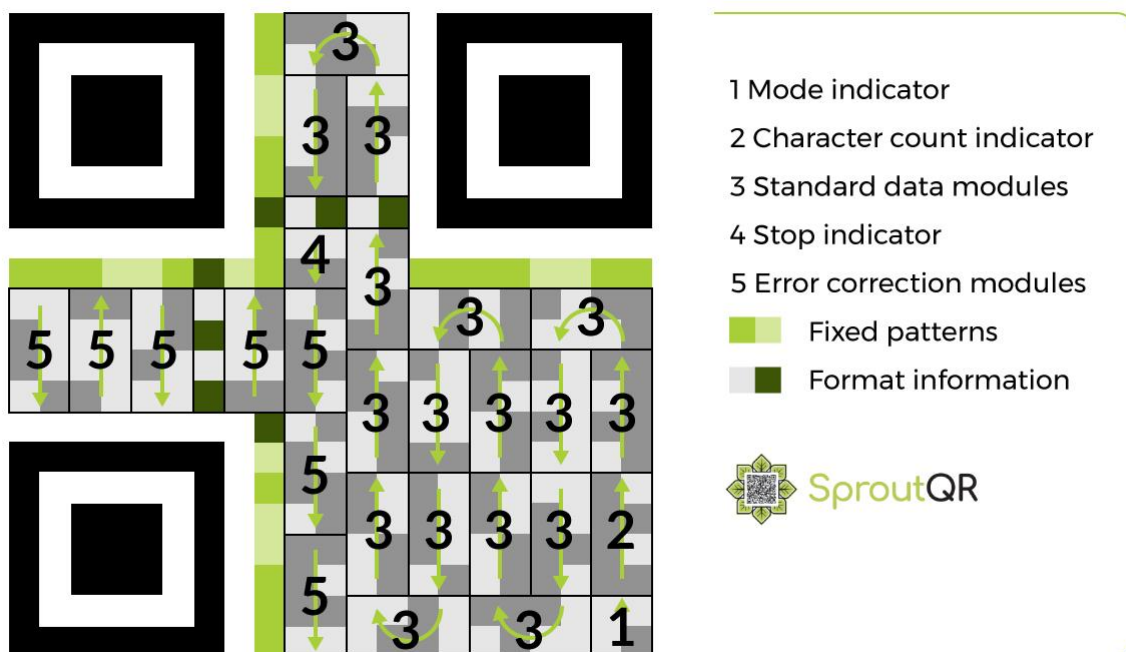


Рис. 1.16. Як працює QR код

### І ось як працюють QR-коди

Звичайно, у принципах кодування QR-кодів є більше нюансів, які ми не розглянули, але це не суть.

QR-коди є стрибком уперед від стандартних штрих-кодів UPC. Просто подивіться на історію QR-кодів. Це тому, що їх процес кодування та сканування дозволяє зберігати набагато більші набори даних, швидше сканування та можливість редагування після створення. Ось чому вони є таким чудовим варіантом для барів або ресторанів. Незалежно від того, чи хочуть ці компанії створити PDF-меню з QR-кодом або повністю адаптивне меню HTML (в обох випадках BinWise може допомогти)

### **1.7 Висновок до першого розділу**

Головний задум технічної сторони проєкту – застосовувати новітніші кросплатформні технології, що дозволять дуже гарно використовувати додатки та вільно взаємодіяти з даними користувачів та слідкувати викладачам за оновленням інформації у реальному часі, що оновились в системі. Завдяки технологіям сканування qr кодів заощаджити багато часу на ручному способі ідентифікації студентів. QR це з найліпший засіб на сьогоднішній день передавати та відображати ті чи інші дані.

## **Розділ 2 АРХІТЕКТУРА ПРОЕКТУ ТА КОМПОНЕНТИ**

### **2.1 Головна мета та архітектура додатку**

Використовуючи усі обрані нами технології та маючи доступні ресурси потрібно побудувати WEB та мобільний додатки які будуть запобігати усім сучасним стандартам та вимогам для діджиталізації у сфері освіти.

На даний момент як у вищих так і в середніх навчальних закладах використовуються паперові журнали. Викладачі не завжди мають достатньо часу під час занять, щоб внести дані відвідувань. Наш програмний продукт пропонує змінити давно застарілі журнали запису відвідувань на швидку та ефективну систему ідентифікації по QR коду, звідки викладач відразу отримує інформацію про присутніх студентів.

Технічна реалізація нашого стартапу буде створюватися на основі сучасних комп'ютерних технологій таких як реляційні бази даних та мови запити Java-script. Архітектура системи ідентифікації буде складатися з веб додатку та мобільних додатків на платформі Android/IOS.

Ця ідея дозволить викладачам економити час який можна використати для надання нових знань та також це дозволяє більш ефективно оцінювати присутність студентів на заняттях.

### **2.2 Розробка UI/UX для WEB та мобільних додатків**

Дизайн інтерфейсу користувача (UI) — це процес, для створення інтерфейсів у програмному забезпеченні або комп'ютеризованих пристроях, зосереджуючись на зовнішньому вигляді чи стилі. Ідея створити інтерфейси, які користувачі знайдуть простими у використанні та приємними. Дизайн інтерфейсу користувача відноситься до графічних інтерфейсів користувача та інших форм, наприклад, інтерфейсів, керованих голосом.

Візуальний вигляд інтерфейсу користувача створюється за допомогою графічного програмного забезпечення, такого як Figma, Sketch або Adobe XD. Знання програм Adobe Creative Cloud, таких як Illustrator та Photoshop та

використання інструментів для створення анімації. Для створення каркасів і прототипів використовуються Axure, UXPin, Invision тощо.

Щоб інтерфейс користувача виглядав привабливим, необхідні навички візуального спілкування та естетичне бачення. Крім того, потрібно знати про поточні тенденції в UI/UX, щоб дизайн виглядав сучасним.

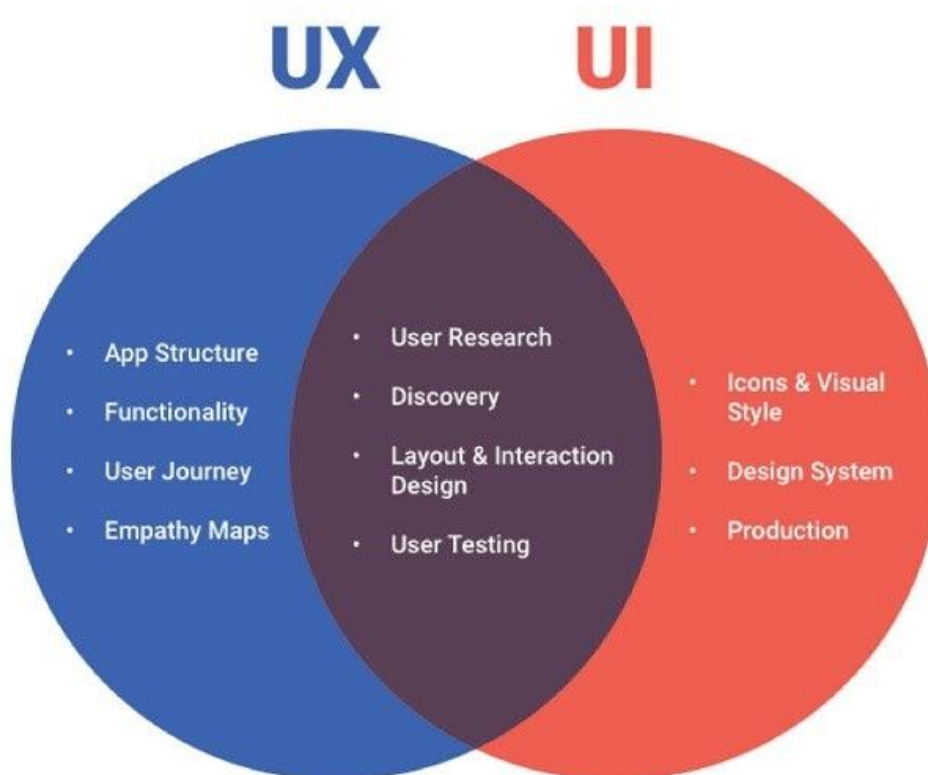


Рис. 2.1. UI/UX

### 2.3 Використовування AJAX для створення запитів на сервер

Аjax - це аббревіатура від Asynchronous Javascript та XML. Він використовується для зв'язку з сервером без оновлення веб-сторінки і, таким чином, підвищує зручність використання та продуктивність.

По-перше, давайте розберемося, що насправді означає асинхронність. Є два типи запитів: синхронні та асинхронні. Синхронні запити - це запити, які слідуєть послідовно, тобто. якщо один процес продовжується і в той же час інший процес хоче бути виконаний, це не буде дозволено, що означає, що виконуватиметься тільки один процес за раз. Це недобре, тому що в цьому типі більшу частину часу ЦП залишається в режимі очікування, наприклад,

під час операцій введення-виведення в процесі, які на порядок повільніше, ніж ЦП, що обробляє інструкції. Таким чином, щоб повністю використовувати ЦП та інші ресурси, використовуйте асинхронні запити. Чому тут є слово javascript. Насправді запити виконуються за допомогою функцій javascript. Тепер термін XML, який використовується для створення об'єкта XMLHttpRequest .

Таким чином, короткий виклад наведеного вище пояснення полягає в тому, що Ajax дозволяє оновлювати веб-сторінки асинхронно, обмінюючись невеликими обсягами даних із сервером за лаштунками. Тепер обговоримо основну частину та її реалізацію. Для реалізації Ajax потрібно знати лише об'єкт XMLHttpRequest. Тепер, що цей об'єкт являє собою насправді. Це об'єкт, який використовується для обміну даними із сервером за лаштунками. Спробуйте згадати парадигму ООП, яка свідчить, що об'єкт взаємодіє за допомогою викликаних методів. Той самий випадок застосуємо і тут. Зазвичай цей об'єкт створюється та використовується для виклику методів, що забезпечують ефективну взаємодію. Усі сучасні браузері підтримують об'єкт XMLHttpRequest.

Базовий синтаксис: нижче наведено синтаксис створення об'єкта:

```
req = new XMLHttpRequest();
```

Є два типи методів open() та send(). Використання цих методів пояснюється нижче.

```
req.open("GET", "abc.php", true);  
req.send();
```

Два наведені вище рядки описують два методи. req означає запит, це в основному посилальна змінна. Параметр GET, як завжди, є одним із двох типів методів відправки запиту. Також використовуйте POST залежно від того, чи надсилаєте дані через метод POST або GET. Другий параметр – це ім'я файлу, який фактично обробляє запити. Третій параметр істинна, він повідомляє, чи обробляються запити асинхронно або синхронно. За

вмовчанням це true, що означає, що запити є асинхронними. Метод open() готує запит перед його надсиланням на сервер. Метод send використовується для надсилання запиту на сервер.

Надсилання параметра через отримання або POST-запит. Синтаксис наведено нижче:

```
req.open("GET", "abc.php?x=25", true);  
req.send();
```

У наведених вище рядках коду зазначений запит у вигляді URL-адреси, за якою слідує ? за яким слідує ім'я змінної, потім =, а потім відповідне значення. Під час надсилання двох або більше змінних використовуйте знак амперсанда (&) між двома змінними. Зазначений метод, як показано, застосовується для запиту GET. Надішліть дані через POST, а потім надішліть їх методом надсилання, як показано нижче.

```
req.send("name=johndoe&marks=99");
```

Будь-яка дія, що виконується при натисканні кнопки, наведенні курсору на елементи, завантаженні сторінки і т. д., називається подіями. JavaScript може виявляти події. Тому зв'яжіть код конкретної події з його дією, яка може бути реалізована за допомогою JavaScript. Це переважно обробники подій.

Обробники подій - це в основному функції, написані на JavaScript, які діють або запускаються, коли користувач запускає подію. При надсиланні запиту за допомогою методу send зазвичай отримують відповідь від сервера пізніше. Але час отримання відповіді невідомий. Тож відстежуйте це.

Тому, щоб відслідковувати подію onreadystatechange відповіді, яка пов'язується з обробником події (функцією), яка буде виконана, коли прийде відповідь.

Під час надсилання запиту до сервера ми виконуємо дії на основі відповіді. Подія onreadystatechange запускається щоразу, коли змінюється readyState. Отже, що насправді являє собою стан готовності і коли насправді

станеться подія `onreadystatechange` і скільки разів вона відбудеться між запитом і відповіддю?

Об'єкт `XMLHttpRequest` має властивість, яка називається `readyState`, значення якого змінюється під час повного циклу запит-відповідь, тобто коли запит підготовлено, надіслано, дозволено, оброблено і коли надходить відповідь. Ось чому це називається `onreadystatechange`.

`onreadystatechange` зберігає функцію (або ім'я функції), яка буде викликатися автоматично кожного разу при зміні якості `readyState`.

`ReadyState` містить різні значення від 0 до 4.

1. запит не ініціалізований
2. з'єднання з сервером встановлено
3. запит прийнято
4. запит на обробку
5. запит завершено і готова відповідь

`XMLHttpRequest` також має властивість під назвою `status`. Статус має такі значення:

- 200: "ОК"
- 404 Сторінка не знайдена

Тепер завжди пам'ятайте, що коли `readyState` дорівнює 4, а `status` дорівнює 200, відповідь готова. Все, що описано вище, реалізовано у кодуванні, як зазначено нижче.

```
<p id = "dis"></p>
```

```
req.onreadystatechange = function(){
if(req.readyState == 4 && req.status == 200){
    document.getElementById("dis").innerHTML = req.responseText;
}
}
```

У наведеному вище кодї, якщо умова істинна (тобто відповідь готова) потїм відображається результат.

Переваги:

- Швидкість збільшується, оскільки не потрібно повторно завантажувати сторінку.
- AJAX виконує асинхронні виклики веб-сервера, це означає, що клієнтські браузері уникають очікування надходження всіх даних перед початком рендерингу.
- За його допомогою можна успішно виконати перевірку форми.
- Використання смуги пропускання - заощаджує пам'ять, коли дані беруться з однієї сторінки.
- Більше інтерактивний.

Недолїки:

- Ajax залежить від Javascript. Якщо у браузері або ОС є якісь проблеми з Javascript, Ajax не буде підтримуватись.
- Ajax може бути проблематичним у пошукових системах, оскільки в більшості своїх частин він використовує Javascript.
- Вихідний код написаний на AJAX легко читається людиною. У Ajax будуть проблеми з безпекою.
- Налаштування складне.
- Проблема з кнопкою повернення в браузері під час використання сторінок з підтримкою AJAX.

#### **2.4 Середовище для розробки Web-версії додатку**

Visual Studio - це інтегроване середовище розробки (IDE), розроблене Microsoft для розробки графічного інтерфейсу користувача (GUI), консолі, веб-застосунків, веб-додатків, мобільних додатків, хмарних та веб-сервісів і т. д. За допомогою цього середовища IDE Ви можете створювати керований код та власний код. Він використовує різні платформи програмного забезпечення для розробки програмного забезпечення Microsoft, такі як магазин Windows,



Microsoft Silverlight, Windows API і т.д. Це не залежить від мови IDE, так як ви можете використовувати її для написання коду на C#, C++ , VB (Visual Basic), Python, JavaScript та багато інших мов. Він забезпечує підтримку 36 різних мов програмування. Він доступний як для Windows, так і MacOS.



Рис. 2.2. Visual Studio code

Visual Studio – це інструмент для розробників, доступний у платній та безкоштовній версіях. Це також сімейство пов'язаних інструментів, помічених торговою маркою Visual Studio. Я почну з розгляду IDE концепції. Програмні програми мають багато рівнів складності. Якщо ви хочете написати просту нотатку, підійде такий інструмент, як Блокнот. Але для більш складного редагування документів ви звернетеся до такого інструменту як Microsoft Word або Google Docs. Чому? Тому що вони містять набір інструментів, які допоможуть вам створити цікавіший документ. Наприклад, з нижнім колонтитулом, змістом і покажчиком. Він також містить граматику, інструменти перевірки орфографії та багато іншого. Те саме і при редагуванні фотографій. Ви можете працювати з простим інструментом, таким як Paint, або використовувати повнофункціональну альтернативу, таку як Adobe

Photoshop. У світі програмування ми називаємо програми, що містять набір комплексних інструментів, інтегрованого середовища розробки або IDE. Ось IDE Visual Studio. Він має безліч вікон, що закріплюються, які містять десятки інструментів для конкретних завдань. У крайньому лівому куті ви можете побачити елементи структури документа та джерела даних, розташовані навпроти межі програми. Також з лівого боку знаходиться оглядач сервера та панель інструментів. У правій частині екрана знаходиться оглядач рішень та вікна властивостей. У центрі знаходиться вікно конструктора XAML та вікна текстового редактора XAML та C#. У нижньому правому куті знаходиться вікно оглядача тестів. Я розповім про всі ці вікна та багато іншого в цій серії. Запитайте більшість людей, що, на їхню думку, робить програміст на роботі, і вони, безперечно, дадуть відповідь: «вони пишуть код». Одна з функцій, що роблять Visual Studio успішною, це докладна підтримка редакторів коду. Ці інструменти редагування намагаються полегшити написання та редагування коду. Visual Studio містить редактори всіх основних мов, таких як C++, C#, F#, Visual Basic, JavaScript. Також є редактори для транспіляторів JavaScript, такі як CoffeeScript та TypeScript. В даний час створення програм також вимагає роботи з файлами розмітки. У Visual Studio є редактори всіх основних мов розмітки, таких як HTML, XML, JSON, XAML, які є розміткою для робочого столу Windows, а також CSS. Кількість типів проектів, доступних у Visual Studio, вражає. Якщо ви можете уявити проект, орієнтований на технології Microsoft, ймовірно, для нього є шаблон, доступний у Visual Studio.

У стандартній комплектації Visual Studio містить десятки шаблонів проектів, що працюють різними мовами. Visual Studio повністю включає .NET framework у інструменти проекту. Ви можете легко завантажити нові шаблони та зразки проектів із діалогового вікна «Новий проект». Коли ви пишете програмне забезпечення, ви також будете робити помилки у вашому коді. Visual Studio має повний набір налагодження інструментів, інтегрованих

в IDE. Ви можете швидко усунути проблему через кілька хвилин після написання нового коду. Також є повний набір інструментів для тестування.

Давайте подивимося на кілька тестів, доступних у IDE. Багато розробників створюють модульні тести своїх класів одночасно з написанням коду. Сьогодні це проста практика. Це допомагає переконатися, що код працює так, як задумано. Є інші інструменти тестування. Інструменти продуктивності є першокласними, і мені подобається інтерфейс робочого столу та інструменти аналізу коду. У Visual Studio є багато інших помічників.

Існують інструменти для дослідження базових конструкцій коду та платформи .NET. Ви отримуете інструменти для моніторингу серверів та інших мережевих сервісів. Він поставляється з набором інструментів для створення коду. Напишіть кілька шаблонів і Visual Studio автоматично створить код. Вони дозволяють створювати інтерфейс користувача перетягуванням і незамінні для візуалізації і створення ваших красивих інтерфейсів. У Visual Studio є багато спеціальних інструментів для веб-розробників. Крім того, веб-команда і команда ASP.NET в Microsoft виконують чудову роботу, змінюючи і покращуючи способи створення та розгортання веб-додатків. Команда ASP.NET створює новий веб-стік, а це означає, що Visual Studio необхідно змінити, щоб пристосуватися до нової системи. Visual Studio 2015 – перша версія, що включає ці інструменти. У рамках цієї зміни Microsoft використовує багато поширених сторонніх веб-інструментів.

## **2.5 Середовище для розробки мобільної версії додатку**

### **Android Studio**

Як офіційне інтегроване середовище розробки або IDE операційної системи Android, Android Studio добре обладнана для швидкої розробки, забезпечуючи високоякісний вихід додатків на всіх пристроях.



Рис. 2.3. Android Studio

### Швидше кодування та швидка ітерація

Завдяки IntelliJ IDEA ця IDE забезпечує швидке завершення коду та миттєву оцінку робочого процесу. Існують певні функції Android Studio, такі як підтримка коду для змін і чудовий редактор коду для оптимізації виведення коду.

Android Studio дозволяє розробникам швидко вносити зміни, натискаючи код і полегшуючи швидкі зміни без повного перезапуску програми. Це забезпечує чудову гнучкість для внесення невеликих змін до програми, поки програма все ще працює.

Інтуїтивно зрозумілий редактор коду Android Studio є ключовою функцією, яка забезпечує одну з ключових переваг Android Studio, як-от більш швидке програмування. У той же час він забезпечує найсучасніший рефакторинг, завершення коду та аналіз коду.

### Швидкий і багатфункціональний емулятор

Android Studio постачається з емулятором, який допомагає запускати всю програму швидше, ніж фактичний пристрій. Емулятор, дозволяє тестувати додаток на кількох пристроях, включаючи телефони, планшети, Android Wear

і Android TV, може моделювати кілька різних апаратних функцій, як-от GPS, кілька сенсорних ввідів, датчики руху тощо.

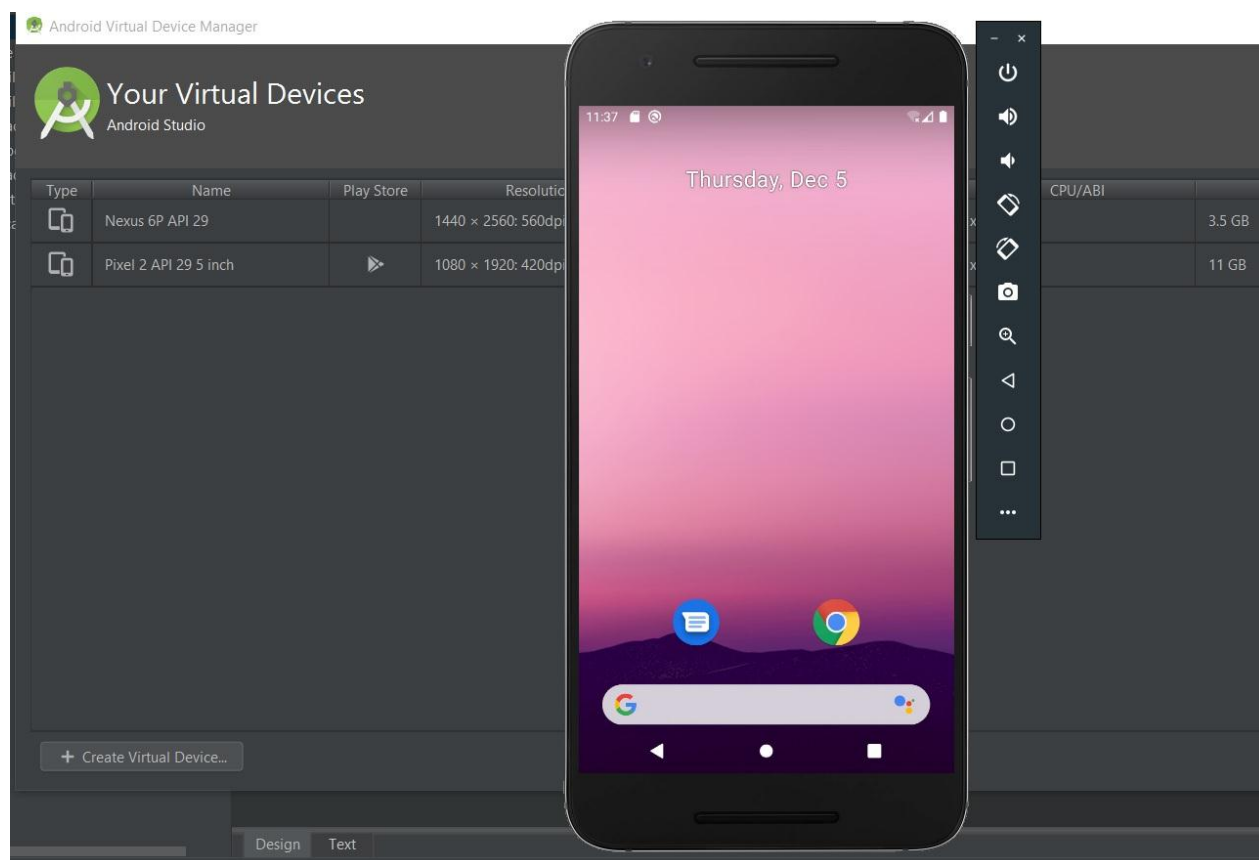


Рис. 2.4. Емулятор девайса

Завдяки Android Studio розробники можуть забезпечити підтримання найкращого стандарту кодування під час роботи над проектом. Android Studio в усьому проекті програми забезпечує надзвичайно ефективне кодування. Завдяки найкращій у галузі практики кодування, під вашим керівництвом Android Studio забезпечує оптимальний результат для проектів програм.

#### Надійні механізми тестування

Android Studio пропонує багато інструментів і фреймворків для тестування, які допомагають тестувати програми Android за допомогою функціональних інструментів тестування інтерфейсу користувача. Переваги Android Studio також показані з усіма видами розширених інструментів і фреймворків для будь-яких цілей. Тести, які проводяться цими

інструментами, можуть проводитися на реальних пристроях, емуляторах або надійних інтеграційних середовищах, а також за допомогою тестової лабораторії Firebase.

#### Підтримка Firebase та інтегрована хмара

Android Studio постачається з Firebase Assistant, який дозволяє підключати будь-яку програму до сервера Firebase, окрім додавання багатьох основних служб, таких як аналітика додатків, аутентифікація, повідомлення сповіщень та деякі інші. Android Studio також допомагає інтегрувати програму з платформою Google Cloud. Будь-яка компанія з розробки додатків для Android в Індії, яка створює корпоративні програми, вважає цю хмарну підтримку найважливішою.

#### Редактор макетів

Android Studio надає інструмент візуального редактора перетягування для роботи з файлами XML. Це допомагає легко створити абсолютно новий макет програми. Редактор макетів Android Studio, створений синхронно з ConstraintLayout API, дозволяє створювати макет, налаштований для різних розмірів екрана. Це забезпечує підтримку оптимізованих підходів до проектування на основі вимог пристрою.

#### Потужна система розвитку

Переваги Android Studio можна віднести до кількох розширених функцій, які допомагають автоматизувати процеси, керувати залежностями та раз і назавжди налаштовувати конфігурацію. Як локальні, так і розміщені бібліотеки можуть бути включені в проект, і існує величезна кількість налаштувань для проектів програми.

Android Studio пропонує надійну та уніфіковану платформу, що дозволяє створювати програми для всіх типів пристроїв Android. Інструмент також дозволяє вирішувати всі проблеми та забезпечувати оптимізований пристрій. Android пропонує найрозширенішу екосистему пристроїв і отримує

переваги оптимізованих для пристроїв атрибутів розробки завдяки Android Studio.

Cloud Test Lab, як інтегрований інструмент Android Studio, може проводити тестування всіх видів збоїв. Android Studio пропонує цю послугу, яка допомагає розробникам проводити тестування програми на кількох пристроях і налаштуваннях конфігурації. Він також забезпечує повне розширення процесів тестування для проведення різних тестів на різних пристроях Android.

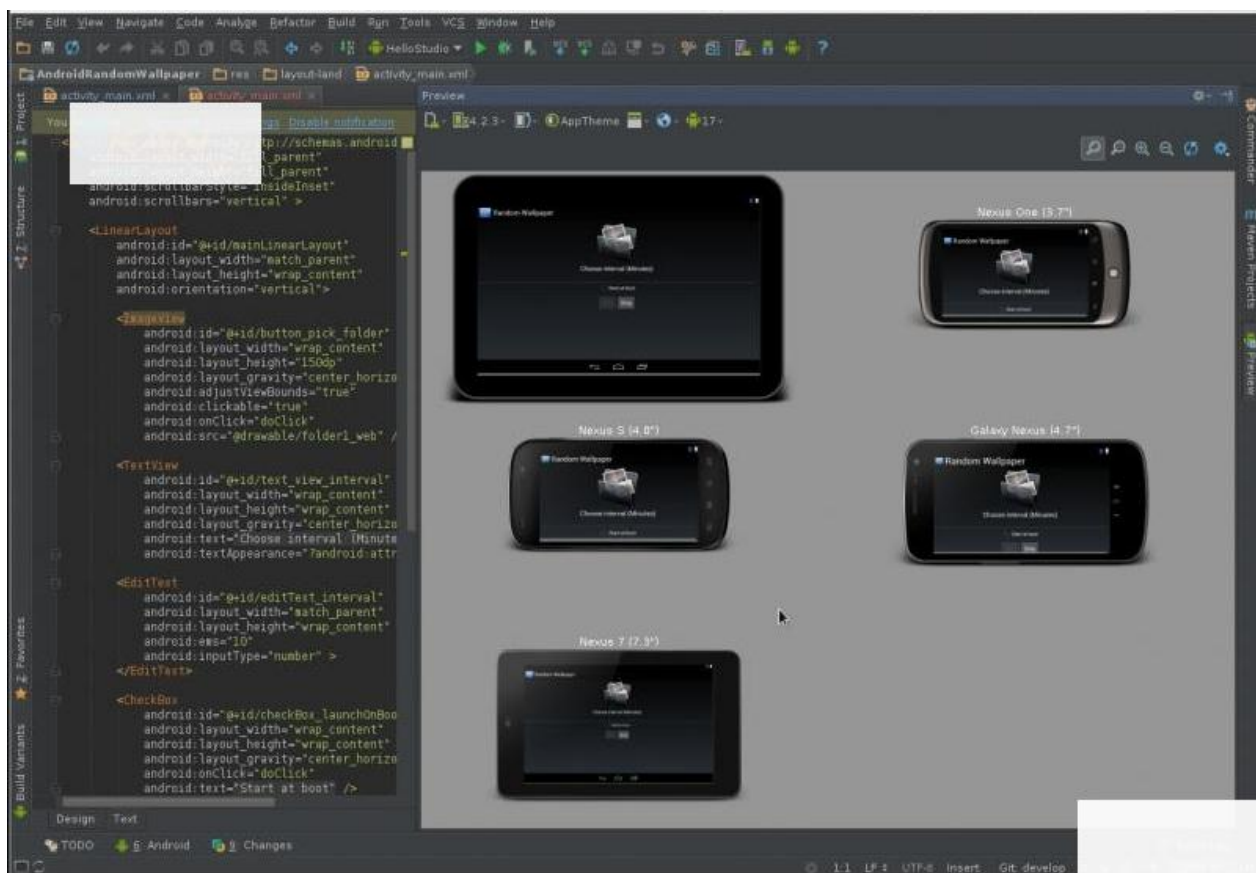


Рис. 2.5. Тестування різних типів пристроїв

### Надійна аналітика

Android Studio постачається з потужним інструментом статичного аналізу та фреймворками. Інструмент надає багато рішень для швидкого виправлення для різних категорій, включаючи безпеку додатків, продуктивність додатків та інші, лише одним клацанням миші. Завдяки такій детальній аналітиці, яка описує різноманітні точки даних, Android Studio надає додаткам більше вогневої потужності.

### Індексування додатків

Android Studio допомагає користувачам легко знаходити програму в Пошуку Google завдяки API індексування додатків. Тепер Android Studio спрощує відображення програм у результатах пошуку, надаючи оптимізовану структуру URL-адрес прямо в коді програми. Розробники додатків також можуть додати ці атрибути URL-адреси прямо у файл `AndroidManifest.xml`, що допомагає краще індексувати за допомогою індексування додатків Google.

### Студія векторних активів

Android Studio дозволяє створювати нові графічні ресурси для різної щільності візуальних елементів. Завдяки цьому інструменту, який називається Vector Asset Studio, розробники можуть вибирати іконки, які підтримуються google material design. В іншому випадку розробники можуть імпортувати файли SVG або PSD для тієї ж мети. Vector Asset Studio також може генерувати файли растрового зображення для різних щільностей екранів, які підтримують попередні версії Android.



## **2.6 Висновки до другого розділу**

При розробці ми скористалися новітніми технологіями крос платформної розробки, що забезпечують надійність та перспективність працювати додаткам на будь-яких новітніх інсуючих пристроях, що мають камеру та вільний вихід до світової мережі Інтернет.

Реактивність додатків дає чудову можливість швидко оновити ті чи інші дані та покривають усі потреби юзера новішою інформацією в режимі дійсного часу. Компонентна побудова дозволяє дуже швидко кастомізувати сторінки додатоків під вимоги користувача та змінити окремі елементи у разі необхідності.

## Розділ 3. ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ

### 3.1 Архітектура додатку

Додаток складається з таких основних частин:

- Форма авторизації до системи
- Профіль студента
- Профіль викладача
- Сторінка відвідувань
- Панель, що знаходиться зверху містить кнопку відкриття та закриття головного меню, а також кнопку профілю. Натиснувши на кнопку профілю ми можемо спостерігати ім'я користувача та його електронну адресу
- Головне меню, що складається з наступних пунктів: “Список предметів”, “Календар відвідувань” та “Статистика”. Натиснувши на пункт меню відкриється відповідна сторінка.
- Сторінка сповіщень, куди буде надсилатися інформація про зміни у розкладі.
- Контентна область, що відображає поточну сторінку

Наш додаток буде складатися з WEB-сайту та мобільних додатків на системах Android та IOS. Android, це операційна система для смартфонів, планшетів і нетбуків. Операційна система Android заснована на модифікованому ядрі Linux і має безкоштовні програми та ігри. Під Android розробники пишуть програми на мові Java, керуючи пристроєм через розроблені Google бібліотеки.

Android - програми включають в себе java-додатки та бібліотеки, які запускаються віртуальною машиною Dalvik з JIT компілятором. Бібліотеки включають в себе систему управління, графіку OpenGL ES 2.0, движок WebKit, графічний движок SGL, SSL і бібліотеки Bionic.

Операційна система оновлюється і піддається удосконаленням і буде встановлюватися на багатьох моделях смартфонів, найбільш популярних як:

Samsung, Xiaomi, Huawei, Oppo.

iOS – це мобільна операційна система від Apple. Розроблена спочатку для iPhone, згодом також вдосконалена для використання на iPad. Apple не дозволяє роботу ОС на мобільних телефонах інших фірм. iOS є похідною від OS X, отже, є за своєю природою Unix-подібною операційною системою.

Функціонал мобільних додатків буде повторюватись з веб – версією програми, але буде вже реалізована реєстрація не тільки викладачів, але і студентів.

Кожен студент матиме свій унікальний ID за допомогою якого, студенту при скануванні QR-коду більше не потрібно буде заповнювати форму присутності на парі. Тобто, дані будуть відправлятися автоматично, що на багато підвищує зручність додатку.

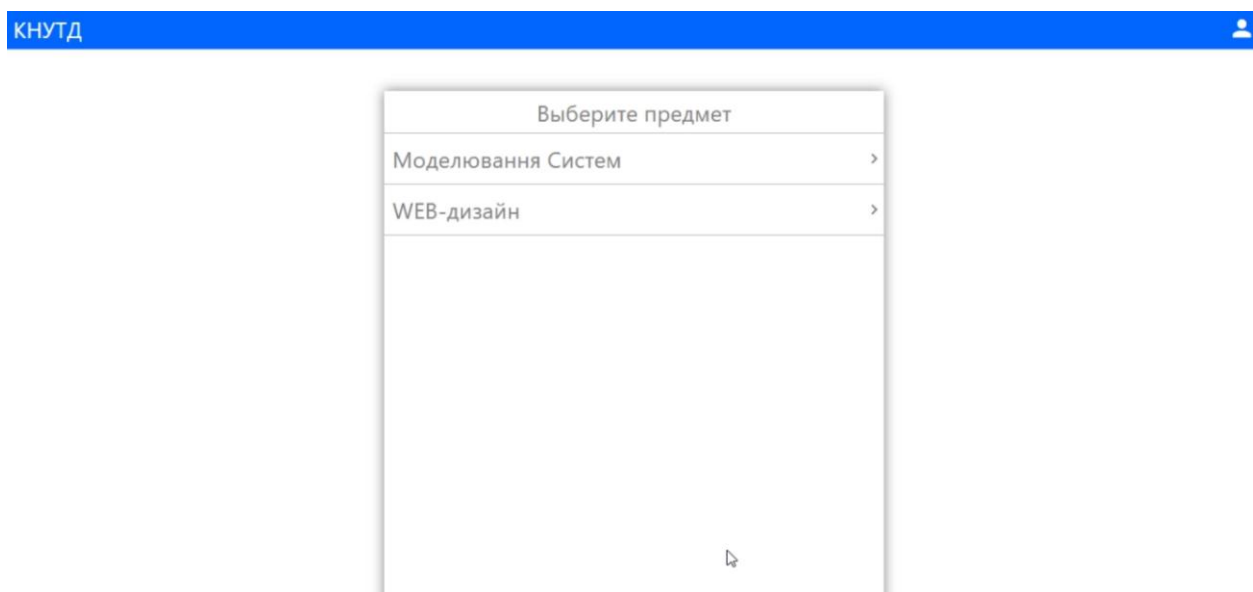


Рис. 3.1. Архітектура додатку

### 3.2 Аналіз інтерфейсу користувача

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

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

Естетичний зовнішній вигляд самого додатка та спосіб його представлення дозволяє сформувати у споживача позитивну думку про програму. Проте естетичні характеристики дуже суб'єктивні та описати їх кількісно набагато важче, ніж функціональні вимоги чи показники продуктивності. Вся естетика додатку часто зводиться до простого вибору: чи співвідносяться між собою використовувані кольори, чи передають елементи інтерфейсу їх призначення і сенс операцій, що відчувається при використанні тих чи інших елементів управління та наскільки успішно він їх використовує.

Продуктивність, а також надійність, також впливають на перспективу застосування програми. Якщо програма добре виглядає, має просте і зручне управління, але, наприклад, повільно промальовує екрани, регулярно «підвисає» на десяток-другий секунд або, ще гірше, падає з критичною помилкою при некоректних діях користувача, у нього, ймовірно, буде мало шансів на тривалу експлуатацію. У свою чергу, швидка та стабільна робота програми можуть частково компенсувати його не найстильніший дизайн чи відсутність якихось вторинних функцій.

Для забезпечення успішної роботи користувача від дизайнера інтерфейсу потрібно дотримуватися балансу між вище переліченими факторами протягом усього життєвого циклу розробки програми. Це досягається послідовним та ретельним опрацюванням деталей інтерактивної

взаємодії на кожному з етапів розробки інтерфейсу користувача які включають:

### 1. Проектування

- Функціональні вимоги: визначення мети розробки та вихідних вимог
- Аналіз користувачів: визначення потреб користувачів, розробка сценаріїв, оцінка відповідності сценаріїв очікуванням користувачів
- Концептуальне проектування: моделювання процесу, для якого розробляється програма
- Логічне проектування: визначення інформаційних потоків у додатку
- Фізичне проектування: вибір платформи, де буде реалізований проект та засоби розробки

### 2. Реалізація

- Прототипування: розробка паперових та/або інтерактивних макетів екранних форм
- Конструювання: створення програми з урахуванням можливості зміни його дизайну

#### 3.3 Прототипування інтерфейсу користувача

Прототипи стали незамінним етапом професійної розробки програмного забезпечення. Прототипи полегшують життя всім людям, залученим у процес створення того чи іншого проекту: клієнту, менеджерам, дизайнерам та розробникам. Прототип візуалізує кінцевий продукт і допомагає виявити серйозні проблеми на ранніх етапах, допомагаючи уникнути проблем на пізніх стадіях розробки.

Звичайно, первісні ескізи, виконані, наприклад, на папері і перші ідеї дизайнера, що втілюють, ще ніхто не скасовував. Але всю подальшу роботу, до якої будуть залучені всі члени робочої групи, найкраще проводити у

спеціальному програмному забезпеченні, яке підтримує редагування, керування версіями та спільну роботу.

Прототип програми буде створюватися в досить популярній графічній програмі Adobe Photoshop. Ця програма є досить зручною у використанні та швидкому освоєнні для побудови прототипу.

Починати прототипування ми почнемо з розробки лівого меню, так як цей функціонал є головним у додатку. Перше, що буде додано в макет це так звані StatusBar та Toolbar. Statusbar – це візуальний графічний елемент системи Android взаємодії користувача із системою. Toolbar - також відомий як панелі дій, є одним з найважливіших елементів дизайну в діяльності нашої програми, так як він забезпечує візуальну структуру та інтерактивні елементи, які знайомі користувачам.

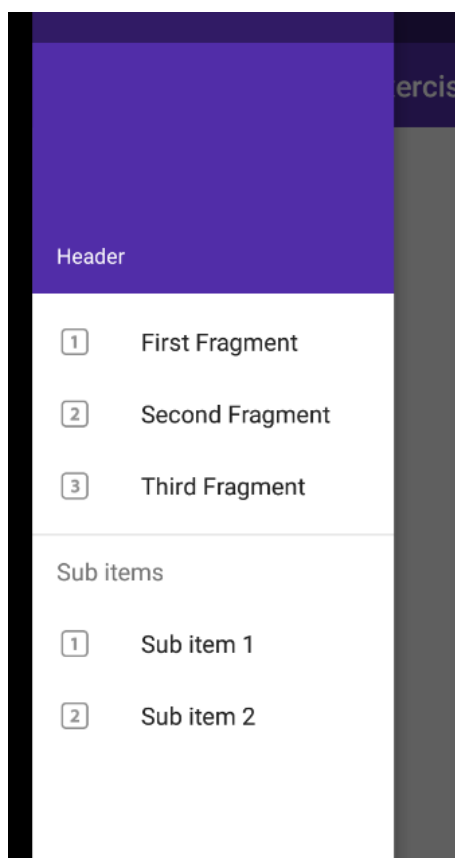
Для створення меню ми звернемося до Google Material Pattern, в якому описується найкращі рішення для побудови дизайну програми.

У рекомендаціях матеріального дизайну Google сказано, що об'єкти інтерфейсу користувача повинні відкидати тіні, як і об'єкти реального світу. Коли для представлення визначається властивість elevation, Android автоматично генерує тінь від цього уявлення. Чим більше значення elevation, тим чіткіше виражена тінь. У рекомендаціях матеріального дизайну наведені бажані значення elevation для різних екранних компонентів - наприклад, для діалогових вікон рекомендоване значення elevation дорівнює 24dp, а меню — 8dp.

Розробники додатків часто наводять кольори теми у відповідність до фірмовим стилем фірми. Якщо вам потрібно змінити колір теми, рекомендації матеріального дизайну Google щодо використання кольору радять вибрати колірну палітру, що складається з основного кольору (не більш ніж з трьома відтінками) та акцентного кольору. Основні кольори зазвичай використовуються для фарбування рядки стану та панелі програми в верхню

частину екрана; крім того, вони можуть використовуватися в графічному інтерфейсі.

Згідно з рекомендаціями дизайнерів Google, меню має містити елементи у вигляді списку з іконками та написами, так само мати заголовок з необхідною функціональністю. Ми вибрали так званий Navigation Drawer для реалізації навігації у додатку.



Значок документації називається "гамбургером" (Hamburger menu). Це офіційна позиція Google. При натисканні зліва вилізе навігаційна шторка. По висоті вона займає весь екран, включаючи системну область. Можете посунути шторку вперед-назад, щоб побачити, що верхня кромка шторки у системній області напівпрозора і не закриває системні значки. Подібна поведінка доступна на пристроях під Android 5 та вище. На старих пристроях шторка розташована під системною панеллю. Сама шторка складається з двох основних частин – у верхній частині знаходиться картинка та текст, а в нижній – меню зі значками. Меню у свою чергу розділено на дві групи. У

верхній частині значки можна вибрати і вибраний пункт залишиться виділеним. У нижній частині меню пункти не виділяються. Заберіть шторку назад і викличте її не натисканням на значок гамбургера, а рух пальця від краю екрану в центрі. Ми побачимо, що під час руху значок трансформується. На жаль, шторка закриває значок і незрозуміло, на що перетворюються три смужки. Наш Navigation Drawer містить заголовок з логотипом компанії в стилі Material Design, нижче йдуть список розділів програми, якими здійснюється навігація за розділами.

Спрототипований Navigation Drawer відповідає вимогам Material Design для Android і надає користувачеві програми зручні можливості для навігації по розділах також містить логотип організації, що підтримує належність додатка до цієї організації, яка займалася розробкою цього додатку для зручності користування користувача. Розробляючи прототип розділу сповіщення, ми застосували дизайн карток, який використовують багато програм, в Android вони називаються CardView. Новий компонент CardView з'явився в Android Lollipop, але завдяки бібліотеці сумісності доступний для старих пристроїв. Дозволяє створювати красиву картку з тінню і закругленими кутами, який служить контейнером інших компонентів. Звичайно ж, він застосовується не у всіх ситуаціях, але чудово підходить при відображенні якогось пункту, що складається з картинки, заголовка та невеликої вступної інформації.

Багато популярних програм вже активно використовують CardView. У нашому випадку саме CardView підходить для створення списку розділу сповіщення, тому що у кожного сповіщення є заголовок, дата події, короткий опис. Для підкреслення заголовка використовувалась смуга чорного кольору з прозорістю 56 відсотків прозорості є найбільш прийнятним варіантом і є стандартним для даних заголовків.

Користувач бачить список елементів, та може прокручувати їх. Якщо він вибирає один із елементів списку, це може оновити ActionBar або тригери



докладний екран для вибору. Android надає ListView класу, який здатний відображати прокручувати список елементів. Ці елементи можуть бути будь-якого типу.

Подальше прототипування розділу напряму було направлено на вікно з інформацією про напрям, був обраний варіант з появою нового вікна. У цьому вікні відобразатиметься інформація доступна користувачеві. Спочатку має відобразитися картинка із зображенням заданим на віддаленому сервері, знизу буде докладна інформація у певному, зручному форматі. Для зручної навігації зверху-ліворуч знаходиться кнопка назад для закриття вікна з детальною інформацією. Користувач також має розуміти, що знаходиться у новому вікні для цього змінюється заголовок вікна на назву відкритого напрямку.

### **3.4 API запити**

Для роботи з API серверами було реалізовано відповідний клас. Всі запити до сервера API здійснюються через протокол https. Метод loadCities завантажує список студентів, звертаючись до базового API сервера. Всі інші запити надсилаються до API сервера, що відповідає вибраному для роботи курсу. Крім звернення до сервера API дані на мобільний пристрій можуть прийти з push-повідомленням. У повідомленні міститься тип даних, щоб на мобільному пристрої можна було зрозуміти, як їх обробити, наприклад orderPublished, та самі дані. Для роботи з даними було реалізовано клас NotificationManager. Програми, написані на React Native, не зберігають свого стану після вимкнення, тому при запуску програми загальне сховище буде порожнім. Перевага такої поведінки в тому, що програма займатиме мінімум пам'яті, необхідної для зберігання даних на пристрої. Однак є ряд даних, які повинні зберігатися на пристрої та бути доступними під час запуску програми без доступу до сервера. Наприклад, токен для автентифікації, дані користувача і т.д. Для таких даних React Native надає AsyncStorage – це сховище, до якого можна додавати, видаляти та вилучати дані, але, на відміну

від загального сховища, дані в AsyncStorage будуть збережені після вимкнення програми або телефону та будуть доступні при наступному запуску. Для роботи з AsyncStorage було реалізовано клас Storage.

### 3.5 Форма авторизації до системи

Для запуску додатку нам потрібно відкрити браузер який встановлений на нашому комп'ютері та перейти за посиланням на інтернет-сторінку. Перше, що нами буде помічено це конфігурація авторизації користувача в системі.

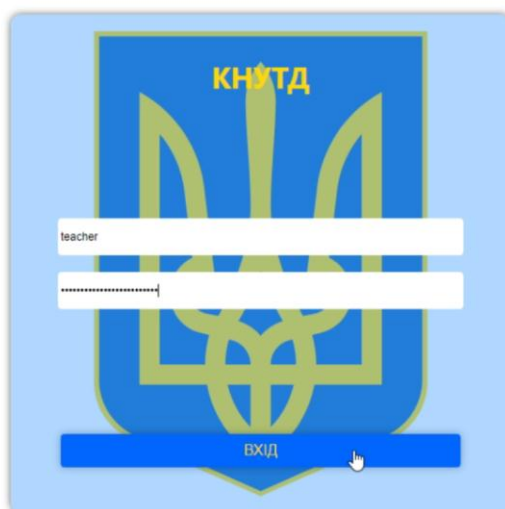


Рис. 3.2. Форма авторизації

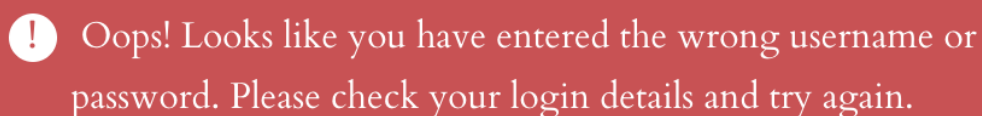
Форма авторизації має наступні елементи:

- Поле для введення електронної адреси викладача або студента
- Поле для введення паролю користувача
- Кнопка що відправляє запит на вхід у систему до сервера

Таким чином вставимо дані для входу у додаток у належні поля та натиснемо кнопку “Вхід”. Якщо користувач зробив усе правильно і він успішно пройшов реєстрацію та його дані введено коректно, сервер пере направить на наступну сторінку, якщо ж даних користувача немає в системі

чи вони введені неправильно буде виведено повідомлення про помилку. Також у нас буде відбуватися валідація

Валідація - це первинна перевірка введених даних, яка суттєво покращує зручність взаємодії з інтерфейсом; виявлення некоректних даних на стороні клієнта дозволяє користувачеві негайно їх виправити. Якщо ж перевірка відбувається лише на сервері, процес заповнення може бути більш трудомістким, тому що вимагає повторення одних і тих самих дій відправлення даних на сервер для отримання зворотної відповіді з повідомленням про те, що потрібно виправити. Наша валідація буде відбуватися по номеру телефону та електронному адресу. Якщо в базі даних не буде знайдено одного з цих пунктів користувач не зможе потрапити до профілю. В базі даних будуть зберігатися усі студенти та викладачі вузу та їхні данні. Завдяки цьому інші особи не матимуть змоги потрапити в додаток.



! Oops! Looks like you have entered the wrong username or password. Please check your login details and try again.

Рис. 3.3. Повідомлення про помилку

### 3.6 Сторінка профілю викладача

Перейшовши на свою сторінку викладач буде бачити стартову сторінку курсів які він викладає в університеті. Відкривши інформаційну сторінку обраного предмету ми можемо розділити її на три основні зони.

Перша зона містить усю основну інформацію по групі. Тут ми зможемо побачити назву предмета, групу для якої відбувається пара та початок та завершення заняття.

Друга зона являє собою список студентів, які належать до цієї групи. Зеленим кольором відображаються студенти котрі пройшли ідентифікацію за допомогою сканування QR-коду. Сірим кольором відображаються студенти які ще не пройшли ідентифікацію, яка можлива від початку до завершення пари. Червоним кольором позначено студентів які не прийшли на заняття.

Третя зона відповідає за генерування QR-коду у додатку. Ми будемо використовувати динамічні QR-коди для наших студентів. Динамічні QR-коди - це QR-коди, які можна змінювати за змістом і можуть зберігати більше даних навіть після того, як ви роздрукували їх на папері.

Цей тип QR-коду забезпечує більший доступ до моніторингу сканувань у реальному часі та розташування зроблених сканувань. Динамічні QR-коди мають 4 наступні функції:

- Це багатофункціональний QR-код, що дозволяє створювати каталоги з кількома URL-адресами.
- Редагується і дозволяє створювати / видаляти URL-адреси.
- Він використовує коротку динамічну URL-адресу для зберігання даних в Інтернеті.



Рис. 3.4. Профіль викладача

Також у викладача буде сторінка статистики де він зможе аналізувати кількість відвідувань студентами його пари.

### 3.7 Сторінка профілю студента

В профілі студента буде відображатися інформація про перелік занять які потрібно відвідувати, список групи, сторінка для сканування QR-коду та сторінка сповіщень. Також студент зможе заповнити особисту інформацію таку як електронна пошта, зображення профіллю, номер мобільного телефона та інше. У сповіщеннях буде відображатися інформація про зміни у розкладі. Наприклад якщо викладач захворів чи час проведення пари змінився, студенту прийде повідомлення про цю подію.

< Профиль

Общая информация

Смена пароля

ПЕРСОНАЛЬНЫЕ ДАННЫЕ

ИЗМЕНИТЬ

Имя

Введите ваше имя

Фамилия

Введите вашу фамилию

Email

Введите ваш email

Телефон

Введите ваш номер телефона

Дата рождения

МЕСТОПОЛОЖЕНИЕ

Страна

Украина

Область

Введите вашу область

Город

Введите ваш город

СОХРАНИТЬ

Рис. 3.4. Профіль студента

### **3.8 Висновки до розділу**

Користувацький інтерфейс наших додатків зручний та простий у використанні. Панель управління та кнопки розміщені зрозуміло. Дизайн розроблено, щоб не відволікати користувачів та дотримуючись одного стилю. Завдяки продуманому меню, що дозволяє легко здійснювати навігацію по додатку ми з легкістю знаходимо потрібну інформацію.

Висока швидкодія та захищеність, як показник того, що дане програмне забезпечення є кращим у своїй сфері застосування.

## ВИСНОВКИ

Протягом реалізація ідеї, були створені WEB та мобільні додатки – «Система ідентифікації студентів» на основі обраних нами технологій.

Обрана технологія сканування QR коду дозволяє дуже кардинально оновити та полегшити систему відзначення студентів котра використовується зараз у всіх державних навчальних закладах. Ця система дозволяє започаткувати діджиталізацію у сфері освіти, також завдяки реалізації цієї ідеї ми допомагаємо глобальній проблемі вирубці лісів завдяки переходу від паперових журналів до електронних гаджетів.

У майбутньому за допомогою цих додатків та сканування QR кодів також можна оновити велику кількість навчальних процесів та зробити їх більш зручними.

Додатки мають зручний та сучасний на сьогоднішній день дизайн, продуману конструкцію та структуру, гнучкі інтерфейси, що дозволяє використовувати на комп'ютерах, телефонах та планшетах з різними параметрами екранів і є простими у використанні. Побудова інтерфейсу цілеспрямована на спрощення діяльності у додатках та використання платформи за призначенням – ідентифікування присутності студентів на парах українських університетів. Розроблені програми максимально оптимізовані та добре сконфігуровані що дозволяє їм займати невеликий обсяг ресурсу девайса під час роботи з ним та невелику кількість простору на накопичувачі. Це дозволяє використовувати додатки без будь яких обмежень, на будь якому сучасному приладі, що має вихід до мережі інтернет.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Web Technologies: A Journey From HTML To Web 3.0 [Електронний ресурс] – Режим доступу до ресурсу:  
<https://www.goodcore.co.uk/blog/web-technologies/>.
2. Веб-технологии для разработчиков [Електронний ресурс] – Режим доступу до ресурсу: <https://developer.mozilla.org/ru/docs/Web>.
3. Веб-технологии для разработчиков [Електронний ресурс] – Режим доступу до ресурсу: <https://developer.mozilla.org/ru/docs/Web>.
4. React [Електронний ресурс] – Режим доступу до ресурсу:  
<https://ru.reactjs.org/>
5. React - The Complete Guide (incl Hooks, React Router, Redux) [Електронний ресурс] – Режим доступу до ресурсу:  
<https://www.udemy.com/course/react-the-complete-guide-incl-redux/>
6. Redux [Електронний ресурс] – Режим доступу до ресурсу:  
<https://www.udemy.com/course/react-the-complete-guide-incl-redux/>
7. React Native [Електронний ресурс] – Режим доступу до ресурсу:  
<https://reactnative.dev/>
8. What Is React Native? [Електронний ресурс] – Режим доступу до ресурсу: <https://www.netguru.com/glossary/react-native>
9. QR Code [Електронний ресурс] – Режим доступу до ресурсу:  
<https://www.qrcode.com/en/about/version.html>
10. Что такое QR-коды и как их сканировать [Електронний ресурс] – Режим доступу до ресурсу: <https://www.kaspersky.ru/resource-center/definitions/what-is-a-qr-code-how-to-scan>
11. QR-код [Електронний ресурс] – Режим доступу до ресурсу:  
<https://ru.wikipedia.org/wiki/QR-%D0%BA%D0%BE%D0%B4>
12. UX, UI, IA, IxD: определения четырех сложных терминов цифрового дизайна [Електронний ресурс] – Режим доступу до ресурсу:



- <https://www.blog.jazov.com/uiux-design/ux-ui-ia-ixd-opredeleniya-chetyrex-slozhnykh-terminov-cifrovogo-dizajna.html>
13. Что такое UX дизайн? [Электронный ресурс] – Режим доступа до ресурсу: <https://tilda.education/articles-what-is-ux-design>.
  14. Г. Буч, Д. Рамбо, I. Якобсон Мова UML. Керівництво користувача. - 2-ге вид. – Москва: ДМК, 2008. – 248 с.
  15. Expo quick start [Электронный ресурс] – Режим доступа до ресурсу: <https://docs.expo.dev/index.html>
  16. Android Studio - интегрированная среда разработки [Электронный ресурс] – Режим доступа до ресурсу: [https://ru.wikipedia.org/wiki/Android\\_Studio](https://ru.wikipedia.org/wiki/Android_Studio)
  17. Android Studio [Электронный ресурс] – Режим доступа до ресурсу: <https://developer.android.com/studio>
  18. The Complete Guide to Advanced React Component Patterns [Электронный ресурс] – Режим доступа до ресурсу: <https://www.udemy.com/course/the-complete-guide-to-advanced-react-patterns/>
  19. React Native - The Practical Guide [Электронный ресурс] – Режим доступа до ресурсу: <https://www.udemy.com/course/react-native-the-practical-guide/>
  20. Advanced CSS and Sass: Flexbox, Grid, Animations and More! [Электронный ресурс] – Режим доступа до ресурсу: <https://www.udemy.com/course/advanced-css-and-sass/>
  21. The Complete React Developer Course (w/ Hooks and Redux) [Электронный ресурс] – Режим доступа до ресурсу: <https://www.udemy.com/course/react-2nd-edition/>
  22. React - Mastering Test Driven Development [Электронный ресурс] – Режим доступа до ресурсу: <https://www.udemy.com/course/react-tdd/>

23. Mastering Regular Expressions in JavaScript [Электронный ресурс] – Режим доступа до ресурсу: <https://www.udemy.com/course/mastering-regular-expressions-in-javascript/>
24. Accelerated ES6 JavaScript Training [Электронный ресурс] – Режим доступа до ресурсу: <https://www.udemy.com/course/es6-bootcamp-next-generation-javascript/>
25. Web Design for Web Developers: Build Beautiful Websites! [Электронный ресурс] – Режим доступа до ресурсу: <https://www.udemy.com/course/web-design-secrets/>
26. The Complete Sass & SCSS Course: From Beginner to Advanced [Электронный ресурс] – Режим доступа до ресурсу: <https://www.udemy.com/course/sasscourse/>
27. Современный учебник JavaScript [Электронный ресурс] – Режим доступа до ресурсу: <https://learn.javascript.ru/>
28. JavaScript Учебные материалы [Электронный ресурс] – Режим доступа до ресурсу: <https://developer.mozilla.org/ru/docs/Web/JavaScript>
29. Основы CSS [Электронный ресурс] – Режим доступа до ресурсу: [https://developer.mozilla.org/ru/docs/Learn/Getting\\_started\\_with\\_the\\_web/CSS\\_basics](https://developer.mozilla.org/ru/docs/Learn/Getting_started_with_the_web/CSS_basics)
30. CSS по БЭМ [Электронный ресурс] – Режим доступа до ресурсу: <https://ru.bem.info/methodology/css/>
31. HTML [Электронный ресурс] – Режим доступа до ресурсу: <https://developer.mozilla.org/ru/docs/Web/HTML>
32. How QR Codes Work and Their History [Электронный ресурс] – Режим доступа до ресурсу: <https://www.qr-code-generator.com/blog/how-qr-codes-work-and-their-history/>
33. How QR Codes Work? [Электронный ресурс] – Режим доступа до ресурсу: <https://uqr.me/qr-code-marketing/how-qr-codes-work/>

34. Адаптивная вёрстка сайта [Электронный ресурс] – Режим доступа до ресурсу: <https://html5book.ru/adaptivnaya-vyorstka-sayta/>
35. AJAX [Электронный ресурс] – Режим доступа до ресурсу: <https://ru.wikipedia.org/wiki/AJAX>