

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМ. В. ДАЛЯ
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ЕЛЕКТРОНІКИ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК ТА ІНЖЕНЕРІЇ

До захисту допускається
Завідувач кафедри
_____ І. С. Скарга-Бандурова
« ____ » _____ 2020 р.

ДИПЛОМНИЙ ПРОЕКТ (РОБОТА) БАКАЛАВРА
ПОЯСНЮВАЛЬНА ЗАПИСКА

НА ТЕМУ:

Інформаційно-комп'ютерна система обслуговування Android-додатків

Освітній ступінь “бакалавр”

Спеціальність 123 Комп'ютерна інженерія
(шифр і назва спеціальності)

Керівник проекту:

_____ (підпис)

Є. В. Щербаков

_____ (ініціали, прізвище)

Консультант з охорони праці:

_____ (підпис)

Я. О. Критська

_____ (ініціали, прізвище)

Здобувач вищої освіти:

_____ (підпис)

Д. О. Зінов'єв

_____ (ініціали, прізвище)

Група:

_____ КІ-16д

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВОЛОДИМИРА ДАЛЯ

Факультет Інформаційних технологій та електроніки
Кафедра Комп'ютерних наук та інженерії
Освітній ступінь бакалавр
Спеціальність 123 Комп'ютерна інженерія
(шифр і назва)

ЗАТВЕРДЖУЮ:

Т.в.о. завідувача кафедри КНІ
С.О. Сафонова
« » 2020 р.

З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ) БАКАЛАВРА

(прізвище, ім'я, по батькові)

1. Тема роботи Інформаційно-комп'ютерна система обслуговування
Android-додатків

керівник проекту (роботи) Щербаков Є. В., доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від " " 2020 р. №

2. Термін подання здобувач вищої освіти роботи

3. Вихідні дані до роботи технічне завдання, РНР, додатки

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1) Аналіз предметної області

2) Вибір та обґрунтування програмних засобів розробки системи

3) Розробка системи

4) Охорона праці

5) Висновки

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	Критська Я.О., ст. викл.		

7. Дата видачі завдання _____

Керівник _____

(підпис)

Завдання прийняв до виконання _____

(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту (роботи)	Примітка
1	Пошук літературних джерел	14.05.20-15.05.20	
2	Вивчення літературних джерел	16.05.20-19.05.20	
3	Розробка технічного завдання	20.05.20-22.05.20	
4	Проектування дизайну	23.05.20-26.05.20	
5	Розробка системи	27.05.20-06.06.20	
6	Розробка заходів по безпечному веденню робіт	07.06.20-10.06.20	
7	Оформлення пояснювальної записки	11.06.20-15.06.20	

Здобувач вищої освіти _____

(підпис)

Д. О. Зінов'єв _____

(ініціали, прізвище)

Керівник _____

Є. В. Щербаков _____

РЕФЕРАТ

Пояснювальна записка до дипломного проекту (роботи) бакалавра: 79 с., 29 рис., 5 табл., 23 бібліографічних джерел, 1 додаток.

Об'єкт розробки: інформаційно-комп'ютерна система обслуговування Android-додатків

Мета роботи: розробка інформаційно-комп'ютерної системи для обслуговування Android-додатків

В проекті виконано:

1. Аналіз актуальності створення такої системи.
2. Порівняння існуючих рішень у мережі інтернет.
3. Розглянуто різні засоби та компоненти, за допомогою яких буде розроблятися система.
4. Описано створення та налаштування серверної частини системи.
5. Розроблено систему для обслуговування Android-додатків у мережі інтернет.

Практичне значення, галузь застосування роботи: електронна комерція.

ВЕБ-ДОДАТОК, СЕРВІС, ІНТЕРНЕТ, ІНТЕРФЕЙС, АНДРОЇД-ДОДАТКИ

Умови одержання дипломного проекту: СНУ ім. В. Даля, пр. Центральний 59-А, м. Сєверодонецьк, 93400

Зміст

ВСТУП	8
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	9
1.1 Огляд предметної області	9
1.2 Актуальність розробки сайту для Android-додатків.....	9
1.3 Аналіз існуючих аналогів	10
1.4 Мережа Інтернет.....	12
1.4.1 Протоколи у мережі Інтернет	13
1.4.2 DNS у мережі інтернет	14
1.5 Веб-сайт	16
1.5.1 Різновиди сайтів	18
1.6 Веб-додаток	19
1.7 Сервер	20
1.8 SEO.....	24
1.9 Висновки та постановка завдання	24
2 ВИБІР ЗАСОБІВ РОЗРОБКИ.....	25
2.1 Мова HTML	25
2.1.1 Мова розмітки SVG	27
2.2 Таблиці стилів CSS.....	29
2.3 Мова JavaScript.....	31
2.4 Технологія AJAX.....	33
2.5 Фреймворк Bootstrap	34
2.6 Бібліотека JQuery.....	35
2.7 Мова PHP	36
2.7.1 Скрипти PHP	37
2.8 СУБД MySQL	37
2.9 Висновки до другого розділу.....	38
3 РОЗРОБКА СИСТЕМИ.....	23
3.1 Підготовка сервера	39

3.1.1	Установка та налаштування Apache	41
3.1.2	Установка та налаштування MySQL	43
3.1.3	Установка та налаштування PHP	44
3.1.4	Підключення SSL сертифікату та домену	45
3.2	Структура системи	46
3.3	Головна сторінка	46
3.4	Сторінка додатка	50
3.5	Адаптованість сторінок.....	53
3.6	Висновки до третього розділу	54
4	ОХОРОНА ПРАЦІ	55
4.1	Загальні питання з охорони праці.....	55
4.1.1	Правові та організаційні основи охорони праці	46
4.2	Аналіз стану умов праці.....	57
4.2.1	Вимоги до приміщень.....	57
4.2.2	Вимоги до організації місця праці.....	57
4.3	Виробнича санітарія.....	59
4.3.1	Аналіз небезпечних та шкідливих факторів при виробництві (експлуатації) виробу	59
4.3.2	Пожежна безпека	60
4.3.3	Електробезпека	61
4.4	Гігієнічні вимоги до параметрів виробничого середовища	61
4.4.1	Мікроклімат	61
4.4.2	Освітлення.....	62
4.5	Вентилювання.....	64
4.6	Заходи з організації виробничого середовища та попередження виникнення надзвичайних ситуацій.....	64
4.6.1	Розрахунок захисного заземлення (забезпечення електробезпеки будівлі).....	65
4.7	Висновки до четвертого розділу.....	69
	ВИСНОВКИ	70

ПЕРЕЛІК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	71
ДОДАТОК А.....	74
ДОДАТОК Б	80

ВСТУП

В сучасному світі інформаційних технологій інтернет та Android-додатки стали невід'ємною частиною людського життя. Android-додатки — термін, що позначає програму, створену за допомогою певної мови, призначеної для програмування сучасних смартфонів на базі операційної системи Android.

З поширенням та популяризацією смартфонів з цією ОС в усьому світі з'явилась неймовірно велика кількість додатків для цих мобільних девайсів. На теперішній час, це одна з самих перспективних платформ, для яких розробляють додатки більшість великих компаній та корпорацій. Це показує наскільки великий ринок у операційної системи Android. Навіть Apple створює свої додатки для цієї операційної системи, щоб їх товарами, наприклад airpods, могли користуватися власники девайсів на Android.

Але постає питання, як поширювати свої додатки, особливо починаючим програмістам, або інді-розробникам. Наприклад, всім відомий Плей Маркет вимагає внесок у розмірі 30 доларів. Не всі готові витратити кошти на свій додаток, особливо якщо вони не впевнені в ньому, або додаток націлено на вузький круг користувачів. Тому ці розробники звертаються до сторонніх ресурсів, де поширюють свій додаток. Додатки великих компаній або відомих розробників також поширюються через неофіційні ресурси, особливо це актуально тому, що велика кількість людей звикла використовувати пошукові системи та завантажувати файли з сайтів.

Тому, метою цього проекту є інформаційно-комп'ютерна система обслуговування Android-додатків, за допомогою якої будуть поширюватися Android-додатки різних напрямів від різних розробників. Таким чином, користувач матиме можливість завантажити їх безпосередньо через браузер.

Розроблювана система повинна надавати простий та швидкий доступ до завантаження додатків з усіх девайсів. Це означає що він повинен бути повністю адаптивним під будь-який пристрій та написаний на сучасному HTML5 з використанням CSS, JS та PHP.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Огляд предметної області

Зараз веб-технології неймовірно розвинені і можуть надавати безліч можливостей. Наприклад, для користувача - можливість завантаження потрібної йому гри або програми, а для розробника - платформу для розповсюдження його проекту.

Однак, для створення такого ресурсу потрібно врахувати технічну можливість впровадження: кількість необхідного місця для зберігання файлів, на кого він націлений, на яких пристроях буде використовуватися, актуальність інформації та інтуїтивно зрозумілий інтерфейс для людей різного віку, щоб навіть не впевнені користувачі ПК могли користуватися ресурсом.

У всякому разі власний ресурс - це складна і багатофункціональна платформа, в якій потрібно чітко розуміти, для чого він створюється і хто буде його споживати. Правильно створений сайт - запорука успішного та прибуткового проекту, бо фахово створений сайт це не тільки красивий дизайн, а також правильна SEO-оптимізація з урахуванням конкурентів та оптимізований код як веб-сторінок так і окремих скриптів. По запевненням компанії Google, на ранжування сайтів в пошукових системах також впливає його оптимізація, тобто швидкість завантаження сторінок та їх адаптація під різні пристрої.

1.2 Актуальність розробки сайту для Android-додатків

На даний момент Android-додатки використовуються у всіх сферах життя, вони навіть можуть замінити фізичні перемикачі світла або пульти дистанційного управління. Вони стали настільки невід'ємною частиною нашого життя, що навіть використовуються у сучасних авто, щоб запускати двигун та керувати їм. Прикладом можуть слугувати останні моделі Tesla та Ferrari. Однак, для розповсюдження цих додатків повинна існувати спеціальна платформа.

Тому створення такого веб-додатку дуже актуально, так як більшість існуючих сайтів потребують реєстрації. Наприклад, найвідоміший сайт 4PDA, щоб завантажити додаток, обов'язково потребує акаунт на цьому сайті. Інші потребують не тільки реєстрації, а й завантаження їх Android-додатка для подальшого інсталювання потрібних програм та ігор на смартфон або планшет користувача. До них відноситься найбільш відомі Google Play та Aptoide.

Отже, потрібно розробити веб-сайт для завантаження Android-додатків з наступним функціоналом:

- можливість завантажувати додатки;
- отримання інформації про додаток;
- сортування за категоріями;
- можливість зв'язку з адміністрацією.

З основних переваг, що може надати розроблений сервіс, потрібно виділити:

- адаптованість до мобільних пристроїв;
- інтуїтивно зрозумілий каталог додатків;
- сортування за категоріями;
- простий спосіб завантаження додатків.

1.3 Аналіз існуючих аналогів

Самим популярним сервісом для Android-додатків є Google Play Market (рис. 1.1), в якому знаходиться величезна кількість додатків. Однак він має декілька великих недоліків:

- обов'язкова реєстрація;
- неможливість завантажувати додатки через комп'ютер або браузер у смартфоні;
- для додавання свого додатку потрібно внести суму у розмірі 30\$;
- якщо на телефоні відсутні сервіси Google Play, то Play Market не буде працювати.

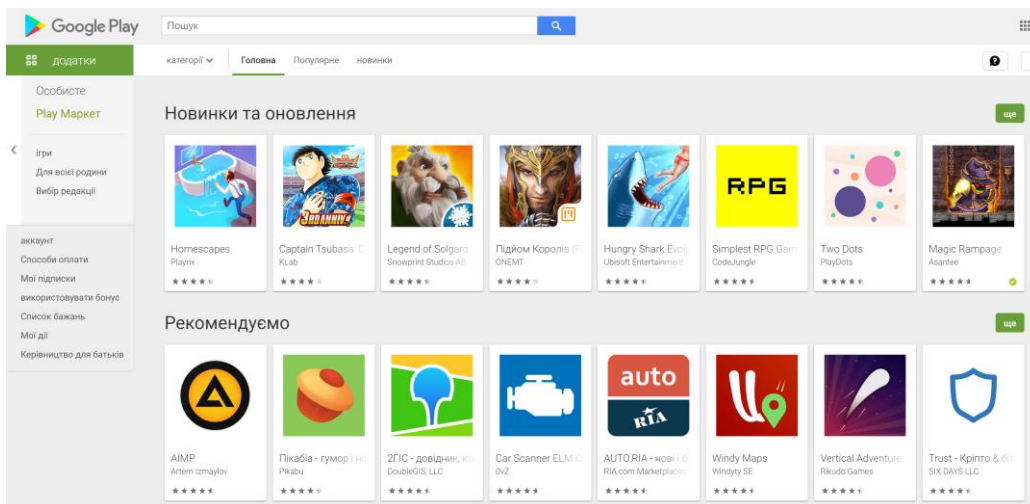


Рисунок 1.1 – Головна сторінка Google Play Market

Іншим аналогом є веб-сайт 4PDA, один із популярних сторонніх сайтів для завантаження додатків, який виконаний у форматі форуму (рис. 1.2), що робить сайт не дуже зручним, так як:

- ускладнено пошук по категоріях;
- немає можливості завантаження без реєстрації на форумі;
- зовсім не сучасний дизайн сайту.

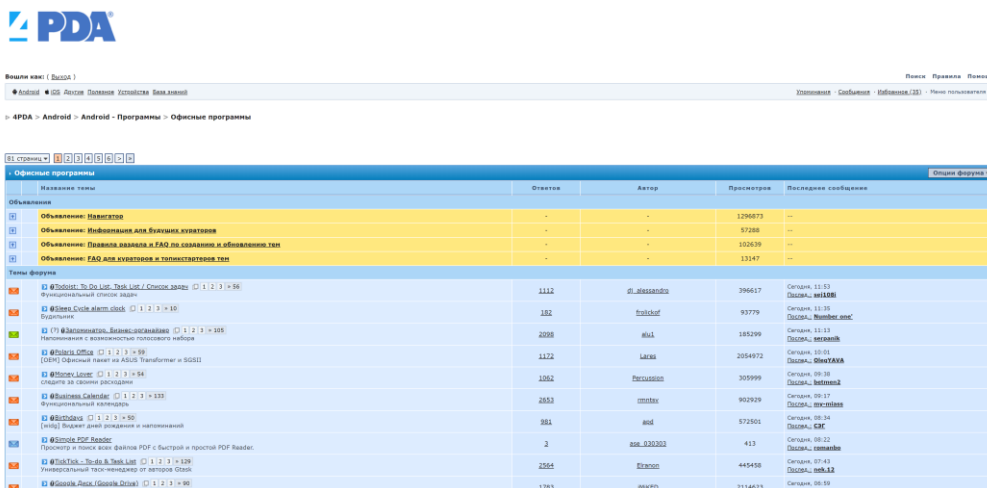


Рисунок 1.2 – Одна з веб-сторінок сайту 4PDA

Хоча це дві основні платформи для того, щоб ділитися та завантажувати ігри та програми для смартфонів на базі ОС Android, у них все одно є перелічені вище недоліки.

1.4 Мережа Інтернет

Сучасний світ неможливо уявити без мережі Інтернет, яка невпинно охоплює різноманітні сфери людського життя. “Інтернет – всесвітня система сполучених комп’ютерних мереж, що базуються на комплекті Інтернет-протоколів” [1].

Спочатку, інтернет був приватною мережею яку створили та використовували американські військові, а тільки наприкінці 1991 року, він став доступним для усіх сфер діяльності.

Інтернет містить дуже багато інформації яка зберігається в різних місцях. Зрозуміло, що неможливо тримати увесь інтернет на одному комп’ютері, чи навіть великому сервері. Оскільки тоді, цей сервер повинен мати колосальні об’єми пам’яті, також, якщо цей сервер вийде з ладу, увесь інтернет перестане працювати. Тому майже кожен сайт чи онлайн сервіс має свій власний сервер. А великі компанії, як Google, Yandex, YouTube та інші, використовують цілі кластери супер потужних комп’ютерів для обробки та зберігання інформації.

Однак, коли ми використовуємо інтернет, наприклад читаємо статті на сайті з новинами, то ми не підключаємось напряму до цього сайту, з’єднання йде через декількох провайдерів та провайдера DNS.

Зазвичай використовується DNS який надав ваш локальний провайдер до якого ви підключений і який надає вам доступ у мережу інтернет. Однак провайдер, який надає вам послуги та зовнішню ір адресу це тільки одна частина в ланцюзі. Тобто, коли ви підключаємось до Wifi роутера чи модема, далі слідує цілий ланцюг підключень. Після вашого маршрутизатора ми переходимо у локальну мережу нашого постачальника послуг, той в свою чергу зазвичай підключений до більшого провайдера котрого зазвичай називають UpLink, і тільки він вже надає запити до провайдера, в чий діапазон входить ір адреса до якої ви підключаєтесь, чи робите відправку пакетів будь-якого протоколу.

Інтернет надає безліч онлайн-сервісів, наприклад такі як:

- Web сервіси – завдяки яким можна отримувати будь-яку інформацію.

- Поштові – сервіси для організованої відправки та отримання електронних повідомлень.
- Медіа-сервіси – використовуються для перегляду фото та відео матеріалу.

1.4.1 Протоколи у мережі Інтернет

Уся інформація у мережі передається за допомогою протоколів, які розділяють трафік на пакети. "Протокол - це набір правил і угод, що використовуються при передачі даних" [2], а пакет свого роду блок даних, який передається по локальній чи глобальній мережі. Однак є й мережі які не підтримують пакетний режим, наприклад "телекомунікаційний зв'язок точка-точка", в цьому випадку в цьому випадку інформація передається як послідовність байтів.

При використанні пакетної передачі даних існує можливість фрагментації пакета, тобто генерація двох пакетів з одного, яка трапляється через перевищення MTU. MTU являє собою максимальний розмір корисного блоку даних. Однак фрагментація передбачена тільки в протоколі IP (Internet Protocol), тому в інших випадках, коли мережевий адаптер побачить кадр більший його MTU, то цей кадр ігнорується.

Найбільш поширеними протоколами є:

- HTTP – протокол для передачі інформації у WWW;
- HTTPS – захищений протокол для передачі інформації у WWW;
- FTP – протокол який використовується для передачі будь-яких файлів.
- POP3 – протокол прийняття повідомлень електронної пошти.
- SMTP – протокол для відправки електронних листів;
- TELNET – протокол для віддаленого керування
- SSH – більш надійний зашифрований протокол віддаленого доступу.

Для доступу до сайтів використовується протокол HTTP, він розшифровується і перекладається як "протокол передачі гіпертексту" і є найбільш використовуваним протоколом передачі даних в інтернеті, який застосовується у браузерях чи будь-яких додатках WebView.

HTTP-протокол є клієнт-серверною технологією, це дає зрозуміти що існують клієнти, які підключаються та надають запит, та сервер, комп'ютер який надає дані в відповідь на запит.

Інші протоколи використовують розробники сайту, чи його адміністратори сайту для налаштування сайту чи серверу на якому встановлено сайт. Наприклад, протоколи SSH та TELNET можна використовувати для налаштування серверу, чи оновлення компонентів ОС, на якому знаходиться сервер. А протокол FTP для завантаження нових файлів чи сторінок для сайту.

1.4.2 DNS у мережі інтернет

DNS розшифровується як "доменна система імен" що являє собою розподілену базу даних яка використовується для отримання інформації о доменах. За допомогою DNS ми маємо можливість перетворювати ім'я хостів у ір адреса.

Це було створено для спрощення запам'ятовування та вводу адресів у мережі інтернет. Так як використання більш простого імені замість числового адреси хоста дуже зрозуміло для людського сприйняття. Зараз кожний сайт має свій домен, тобто ім'я хосту, яке якраз отримується через провайдерів DNS. Нижче зображено принцип роботи DNS (рис. 1.3).

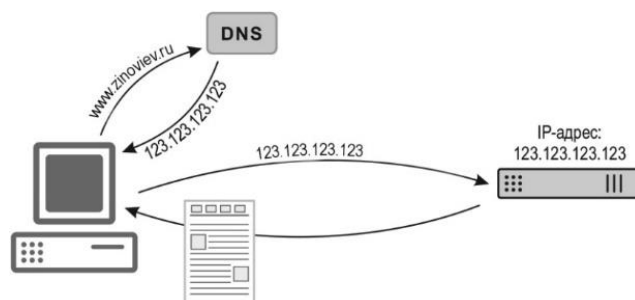


Рисунок 1.3 – Принцип роботи DNS

Роботу DNS-сервера можна розділити на кілька кроків:

1. Користувач вводить URL адресу сайту.
2. Браузер надає запит до DNS провайдера.
3. DNS-сервер шукає ір адресу яка прив'язана до домену.
4. DNS-сервер надає ір сайту.
5. Веб-браузер надсилає запит на IP веб-серверу сайту.
6. Веб-сервер надсилає всі необхідні файли для відображення веб-сайту.
7. Веб-браузер формує сторінку та виводить на екран.

В більшості випадків до одного домену завжди прив'язане одне доменне ім'я. Однак буває, що один сайт може мати декілька доменних імен, або два сайти мали один домен загальний домен. Декілька доменів, зазвичай використовують великі компанії які мають величезний сайт, або багато сервісів. Більшість великих сайтів мають один домен і два сайти, один сайт використовується для новин, а інший це форум, який використовується для обговорення цих новин, і оба сайту зв'язані гіперсилками в меню. Це зроблено задля того, щоб зменшити навантаження на один сервер через велику кількість користувачів. А наприклад відомі пошукові системи Yandex та Google, вони мають багато доменів які ведуть на один сайт, однак різняться в тім, що в залежності від домена буде використовуватися різна мова.

Наприклад, декілька доменів що має пошукова система Google:

- google.com – всесвітний домен;
- google.com.ua – український домен;
- google.us – домен що використовують в США;
- google.ru – російський домен;
- google.co.th – домен, що використовується в Тайланді.

Комп'ютер який використовується для функціонування сайту, називають веб-сервер, а компанії які надають такі сервера називають хостинг провайдерами.

Раніше коли сервера були не такі потужні, на одному сервері зберігався один сайт, однак все змінилося і тепер на одному сервері можуть знаходитись багато сайтів, навіть більше десяти.

В ролі провайдера DNS може виступати локальний провайдер, який надає користувачу доступ в інтернет, так и сторонній провайдер наприклад Google, усім відомий 8.8.8.8 або CloudFlare, новий, але швидкий DNS, який майже вдвічі швидший за інші.

В залежності від обраного провайдера ви будете отримувати різні послуги, а також матимете різну швидкість отримання ір адреси із домена. Наприклад, якщо вам важлива швидкість ви можете обрати CloudFlare, якщо потрібна стабільність то Google DNS, якщо потрібен більш захищений то Yandex DNS, який блокує доступ до шкідливих сайтів і окремих мереж. Також існують ті які блокують нав'язливу рекламу в інтернеті, до них відноситься AdGuard DNS. Однак блокування шкідливих доменів, або блокування реклами через DNS не дуже ефективно, оскільки блокуються усі ресурси які знаходяться на цій ір адресі.

1.5 Веб-сайт

Сайт – це сукупність логічно пов'язаних між собою HTML або PHP сторінок, до яких можуть бути приєднані скрипти на мові JavaScript та таблиці стилів CSS, в більш складні сайти входять також бази даних MySQL в яких зазвичай зберігається інформація про користувачів, або матеріали на сайті [3].

Сам сайт та всі потрібні для його функціонування файли чи пакети встановлюються на сервер. Сервер - це теж комп'ютер, однак він відрізняється тим, що пристосований до цілодобової безперервної роботи та зазвичай має більш високу обчислювальну потужність для забезпечення швидкої обробки інформації або побудови сторінок, таких як PHP. І в більшості випадків має величезне сховище для зберігання інформації. Він як і любий персональний комп'ютер, складається з: материнської плати, оперативної пам'яті, процесора та

завичай з декількома SSD, однак більш застарілі серверні системи використовують звичайні жорсткі диски (HDD).

Також сучасні сервери на яких тримаються веб сайти мають високу пропускну здатність, бо в них використовуються гігабітні мережеві адаптери, які забезпечують можливість використовувати ресурси сайту відразу для великої кількості користувачів.

Раніше, веб-сайти являли собою декілька зв'язаних статичних сторінок між собою та з посиланнями на файли. З розвитком технологій і поширенням інтернету, збільшувалося і кількість локальних та зовнішніх сторінок на сайтах, а також файли і скриптів які використовувалися на цих сторінках. І тепер сайт це не просто сторінка з текстом, це сукупність стилей, скриптів, БД та інших файлів по типу шрифтів які використовують на сайті (рис. 1.4).

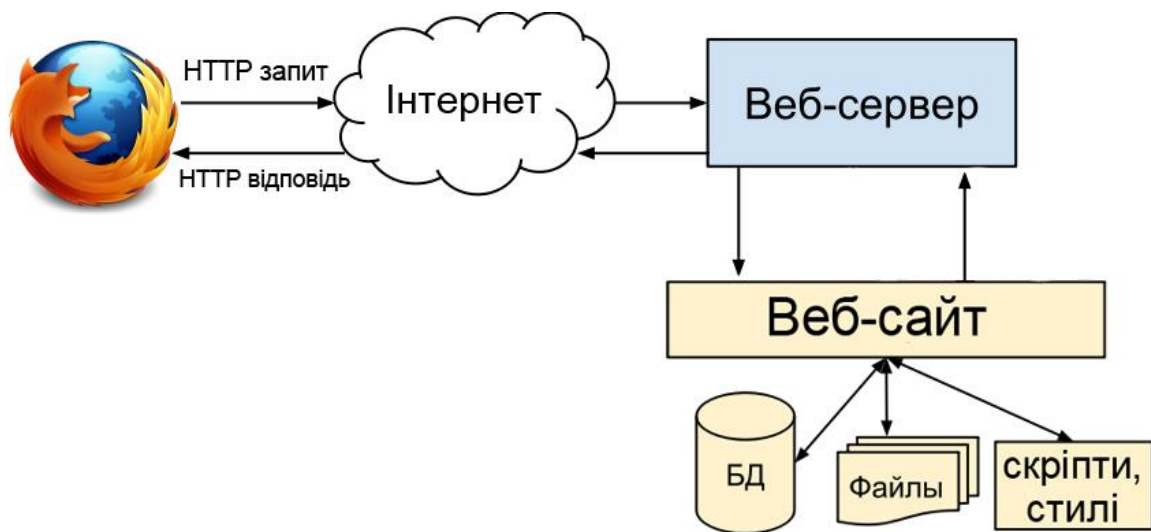


Рисунок 1.4 – Структура запиту на сервер сайту

І з часом, на сайтах ставало з'являтися все більше скриптів, стилей, зображень і навіть відео, це все було пов'язано з зростом пропускнуої можливості інтернету. На сьогодні, пропускна здатність досягає гігабайтів на секунди, та практично всі сайти мають красивий дизайн з використанням великих скриптів, використання котрих раніше було просто неможливо. Зараз комп'ютери з легкістю відкривають відеофайли великого розширення напряму у браузері.

Інтернет став настільки швидким, а сервера та комп'ютери потужними, що можна передивлятися фільми, чи грати в ігри навіть не завантажуючи їх на ПК [4].

1.5.1 Різновиди сайтів

Кожен сайт належить до певної категорії, наприклад більшість сайтів можна поділити на наступні категорії:

1. Комерційний – це сайт який належить якійсь компанії. Його створено для клієнтів або для робітників сайту, які можуть використовувати певні сервіси компанії.
2. Односторінковий сайт, він же Лендинг – сайт що має одну сторінку на якій знаходиться уся потрібна інформація, починаючи від інформації і закінчуючи формою обраного зв'язку. Зазвичай такі сайти використовуються для просування товару чи якоїсь програми.
3. Візитка – зазвичай спрощена версія односторінкового сайту. Такі сайти являються рекламою як для працівників, які шукають роботу, так і для компаній.
4. Інтернет-магазин – це сервіс, де клієнти можуть придбати товари, які пропонуються власниками сайту, чи тими хто публікує товари.
5. Особистий сайт – це сайт, який належить одній певній людині. Він може бути як простим, так і складним з різною кількістю послуг чи можливостей.
6. Інформаційний – новинні портали чи інші постачальники інформації.

За складністю сайти ділять на дві категорії - статичні та динамічні:

1. Статистичний – звичайний набір HTML-сторінок, який може знаходитись як на локальній машині так і на сервері, зумовлено тим, що сайту не потрібна обробка баз даних чи PHP. Такі сайти мають дуже малий розмір, і інформація редагується через сам код сторінки. Великим

недоліком слугує – неможливість гарячої та динамічної зміни інформації на таких сторінках безпосередньо з самого сайту.

2. Динамічний – це складний сайт який складається з різних скриптів чи PHP який має доступ до бази даних.

1.6 Веб-додаток

Веб-додаток це теж сайт, тільки різниться в тім, що там сторінки самого сайту знаходяться у несформованому вигляді, або частково несформованому [5]. Сторінка формується тільки тоді, коли її запрошує клієнт, тобто намагається відкрити її через свій браузер. А так як сторінка формується в залежності від дій клієнта, то такі сторінки можна назвати динамічними.

Перевагою таких сторінок є те, що інформація може формуватися безпосередньо з будь-якої доступної бази даних на сервері, тобто сторінка виводить завжди з актуальною інформацією, наприклад як курс Біткоїна на сайті blockchain.com (рис. 1.5). Також, якщо потрібно щоб інформація була актуальною не тільки при завантаженні сторінки, а і під час знаходження на ній клієнта, то можна використовувати простий скрипт для оновлення сторінки або фрагменту сторінки через певні інтервали часу.



Рисунок 1.5 – Завжди актуальний графік курсу з авто-оновленням

Більш простим прикладом може слугувати сайт aliexpress.ru, який при пошуці видає інформацію про наявність обраних товарів, або головна сторінка цього сайту, де список товарів формується в залежності від того які товари переглядав користувач до цього. Наприклад, я переглядав павербанки та годинник для стола, тому на рисунку 1.6 можна побачити, що після завантаження головної сторінки у розділі рекомендацій з'явилося багато товарів з цих категорій.

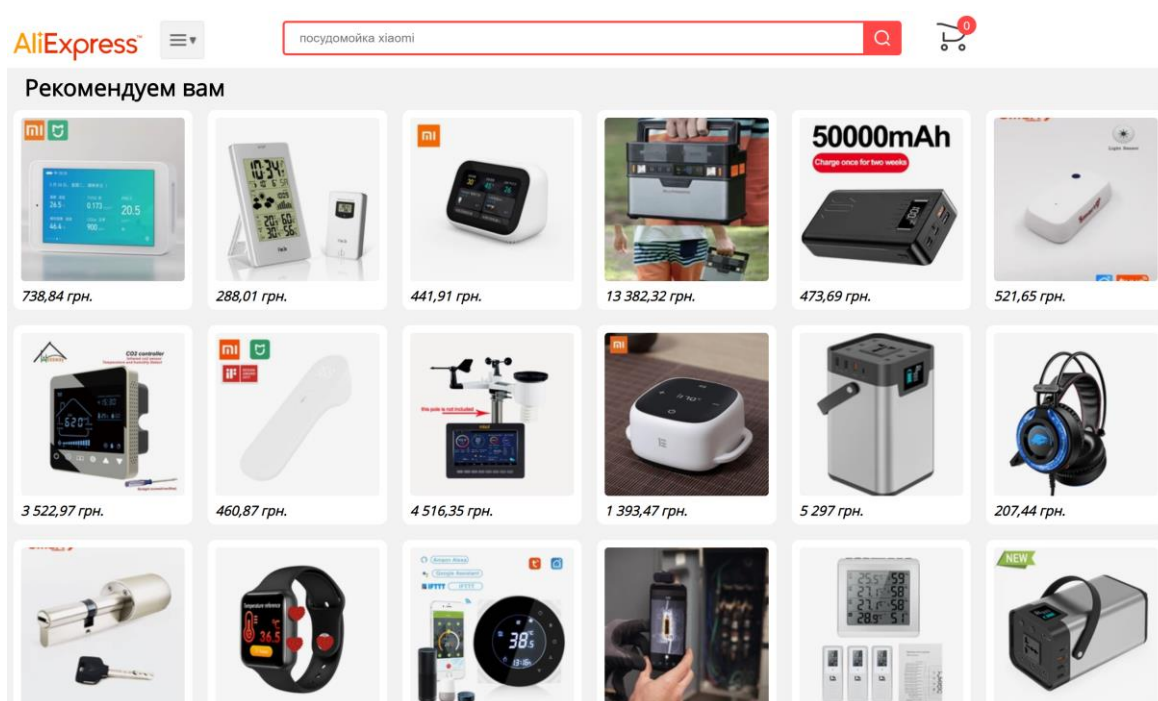


Рисунок 1.6 – Головна сторінка з актуальними рекомендаціями

Хоча сторінка і називається динамічною, в ній може бути багато статичних елементів, таких як логотип сайту, чи якась інформація яку не потрібно змінювати.

1.7 Сервер

Для належного функціонування будь-якого сайту чи сервісу, він повинен знаходитись на сервері. Сервера забезпечують цілодобову роботу і відмінну

обчислювальну потужність, яка використовується тільки для роботи сервісу або сайту. На таких комп'ютерах ніколи не встановлюється зайве програмне забезпечення, яке не знадобиться для коректної роботи.

Хоча сервер в якомусь сенсі теж комп'ютер, однак в ньому використовуються спеціальні серверні материнські плати та процесори, які пристосовані до цілодобової роботи та мають вдвічі більше ядер та L кешу для швидкої обробки інформації. Якщо порівняти сучасні десктопні та серверні процесори (табл. 1.1), то можна побачити різницю у кеші та ядрах з потоками. [6,7]

Таблиця 1.1 — Порівняння десктопного та серверного ЦП

	Xeon E5-2697	Core i7-8700K
Кількість ядер	12	6
Кількість потоків	24	12
Частота ЦП	3.5 ГГц	4.7 ГГц
Кеш 1 рівня	64К	64К
Кеш 2 рівня	256К	256К
Кеш 3 рівня	30 Мб	12 Мб

Однак не тільки процесор відрізняється, це стосується і оперативної пам'яті. Використовується спеціальна серверна пам'ять в якій вбудовано контролер ЕСС, котрий використовує надлишкове кодування даних. Уся інформація в цій ОЗУ забезпечується додатковими службовими бітами, які при пошкодженні даних дозволяють знайти і виправити проблему. На звичайному комп'ютері в такому немає необхідності, але якщо це сервер, на якому знаходиться банківська система або обробляється будь-яка важлива інформація, це може призвести до втрати клієнтів або навіть грошових коштів.

Постійна пам'ять на серверах також відрізняється, хоча більшість хостингів та компаній намагаються використовувати SSD, однак це дуже дорого, особливо якщо потрібні величезні обсяги пам'яті. Тоді використовують звичайні HDD

тільки з більшою швидкістю обертання шпинделя, яке досягає аж 15.000 об/хв. Коли звичайний жорсткий диск має швидкість 5.000-7.000 об/хв. Через те, що швидкість обертання дуже велика, то і збільшується кількість виділеного тепла.

Так як сервери виділяють багато тепла, особливо якщо на цьому сервері встановлено не один, а зразу декілька процесорів, дисків та ОЗУ для ЦП. То йому потрібно спеціальне охолодження, одним чи декількома кулерами не обійтись. В таких серверах використовують величезні радіатори які обдуваються великою кількістю кулерів, і навіть кожен жорсткий диск обдувається одним чи кількома кулерами. На рис. 1.7 можна бачити одну з варіацій охолоджувальної системи для сервера.



Рисунок 1.7 – Фото охолоджувальної системи сервера

Сервери мають багато призначень, тому їх можна поділити на декілька основних типів:

- Веб-сервер. Зазвичай використовується для обслуговування сайтів, клієнтами до яких є веб-браузер користувача або додаток на базі WebView.
- Проксі-сервер. Сервер який виступає в ролі посередника між клієнтом та ресурсом до якого підключається клієнт.
- Поштовий сервер. Сервер який обробляє та зберігає пошту відправлену використовуючи протоколи POP3 та SMTP. Поштові сервери не потребують швидкого доступу до інформації, або великої пропускної

здатності, тому такі сервера не такі потужні як наприклад ті які використовують в відео-сервісах таких як YouTube, де для перегляду 4K відео потрібно мати швидкий SSD та велику пропускну здатність.

— Сервери баз даних. Використовуються для обробки запитів від користувачів. Може бістро зберігати, формувати і виводити дані.

1.8 SEO

SEO – це оптимізація сторінок сайту для підвищення трафіку через пошукові системи. Головною метою SEO є збільшення кількості відвідувачів веб-ресурсу.

Зараз, коли інтернет щодня поповнюється новими сайтами, SEO - налаштування дуже важливі для майбутнього розвитку. Той сайт, у якого добре налаштовані ключові слова та зроблена оптимізація сайту під різні девайси, той і буде першим у пошукових системах. Для кожного веб-ресурсу SEO індивідуальна, не існує універсальної формули успішного сайту. Тут потрібно брати до уваги безліч факторів. Пошукові системи завжди роблять так, щоб користувач завжди бачив найкращі та найактуальніші результати пошуку, а оскільки основний трафік на сайт йде якраз з пошуку, то потрібно дуже ретельно опрацювати SEO для свого сайту.

1.9 Висновки та постановка завдання

На сьогодні, сфери використання Android-додатків та веб-сайтів дуже великі і розширюються з кожним днем. Створення проекту в цій галузі дуже актуально, і для цього потрібно опрацьовувати не тільки програмну, а й апаратну частину проекту з урахуванням оптимізації у сфері SEO.

1. Проектована система повинна бути орієнтована на користувачів операційної системи Android та бути оптимізованою для доступу з усіх типів пристроїв.
2. Користувач без реєстрації повинен отримати можливість:
 - пошуку;
 - завантаження додатка з сервера;
 - перегляд технічної інформації та опису додатка;
 - сортування за категоріями;
 - зворотний зв'язок з адміністрацією сайту.

3. Зареєстрований користувач повинен отримати всі можливості першого та додатково:

- додавання свого додатка на сайт та його зберігання на сервері;
- редагування свого додатка;
- залишати відгуки;
- робити запит для оновлення версії додатка.

4. Адміністрація сайту повинна мати:

- можливість редагування будь-якої сторінки;
- видалення додатка з сервера, де він зберігається;
- блокування користувача;
- повний контроль над налаштуванням сайту та сервера.

Такий підхід зможе максимально забезпечити користувача потрібною інформацією та можливостями, а користувачі, які хочуть розширити свої можливості в системі, можуть зареєструватися і додавати свої програми та ігри в систему.

2 ВИБІР ЗАСОБІВ РОЗРОБКИ

2.1 Мова HTML

Мова розмітки гіпертексту - є основою більшості веб-сайтів та веб-сервісів [8]. Сьогодні, у будь-якому проекті, якщо цей проект має відношення до інтернету, то скоріш за все там використовується мова HTML. Нижче, на рисунку 2.1, зображено приклад коду розмітки гіпертексту.

Товар	Ціна	Кол-во
Гайка	20р	50
Болт	30р	80

Товар	Ціна	Кол-во
Гайка	20р	50
Болт	30р	80

Рисунок 2.1 – Приклад використання мови HTML

Її розповсюджене використання пов'язано з тим, що ця мова вважається універсальною для створення інтернет-сторінок. Зазвичай, такі сторінки мають одне з двох розширень .htm або .html, їх можна створювати як у звичайному блокноті, так і у спеціалізованому програмному забезпеченні. Наприклад, до такого ПО відносяться програми: Sublime Text, Aptana Studio, Adobe Dreamweaver та багато інших.

Також існує оновлена, та доповнена мова розмітки під назвою XHTML, яка створена на базі XML та має всі можливості звичайної мови HTML. Однак, хоча це і відносно нова мова, на жаль її розвиток було припинено, тому по цей день використовується звичайний HTML.

Для відкриття та відображення HTML файлів на екранах використовуються програми, які називаються браузерами. Під браузером мається на увазі будь-яке універсальне програмне забезпечення, яке створено для перегляду інтернет-сторінок та переходу з однієї гіпертекстової сторінки на іншу.

Розмір сторінок HTML не дуже великий, тому добре вони підходять навіть для не дуже швидкого інтернет-з'єднання. Це пов'язано з тим, що сама сторінка містить в собі тільки текст, а точніше текст з розміткою та посиланнями на інші потрібні файли, такі як зображення, шрифти чи інші скрипти. Для зображень зазвичай використовується формати JPEG, GIF, PNG та сучасний WebP, який використовує Google. WebP – це формат який ділиться на два типи. Перший lossless, тобто він майже не відрізняється за якістю від оригіналу. І він на 26% менший за PNG. А також lossy, тобто зображення з невеликою втратою якості. Він менший за JPEG аж на 34%, та на відміну від JPG, в ньому можна використовувати картинки з прозорим фоном, тобто даний формат підтримує альфа-канал [9].

Хоча HTML вважається самою простою мовою, однак це не зовсім вірно, бо сторінки теж можна поділити на складні та прості, це залежить від того, що хоче замовник чи як їх уявив дизайнер, і тільки при наявності макету розробник пише HTML-сторінку мовою веб-розмітки. На рис. 2.2 зображено структуру одного з html-елементів.

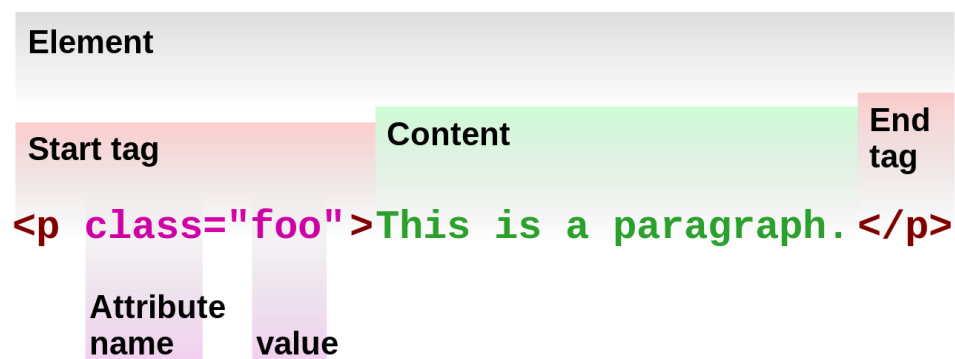


Рисунок 2.2 – Структура одного html-елемента

Як можна побачити, елемент складається з декількох частин. На початку йде початковий тег, його ще називають відкриваючим, в ньому міститься тип елемента, тобто його ім'я, яке указує на те що повинно виконуватися с контентом всередині елемента. Також в початковий тег входять додаткові параметри і, якщо потрібно, то і значення для них. Після відкриваючого тега йде сам контент, тобто до чого застосовується тег. В кінці йде закриваючий тег, який на відмінно від того що йде спочатку, не може містити в собі будь-які параметри.

Чим складніше і більш функціональний дизайн, тим більше буде елементів в HTML-коду, а також буде підключено більше зовнішніх елементів, таких як JavaScript та CSS, які також можуть бути вписані напряму, використовуючи тег, наприклад `<script></script>`, для написання JS коду безпосередньо у HTML-документі.

На цей час, останньою версією є HTML 5, який було створено для кращої сумісності з технологіями мультимедіа. Якраз починаючи з п'ятої версії було впроваджено теги `<audio>`, `<canvas>` та `<video>`. А головне з'явилась можливість використання SVG зображень.

2.1.1 Мова розмітки SVG

SVG – це векторна графіка, яка є масштабованою і написана на основі XML [10]. Файли з її використанням мають формат `.svg` та можуть містити в собі не тільки статичне зображення, а й анімоване або інтерактивне. Також замість просто зображень файл може містити описи сценаріїв.

Через те, що SVG являє собою векторну графіку, це дуже добре позначається на адаптивності зображень на сайті, тим самим робить зображення на сайті красивими і чіткими у будь-якому розширенні, на будь-якому девайсі. На рис. 2.3 зображено відмінність векторного та растрового зображень.

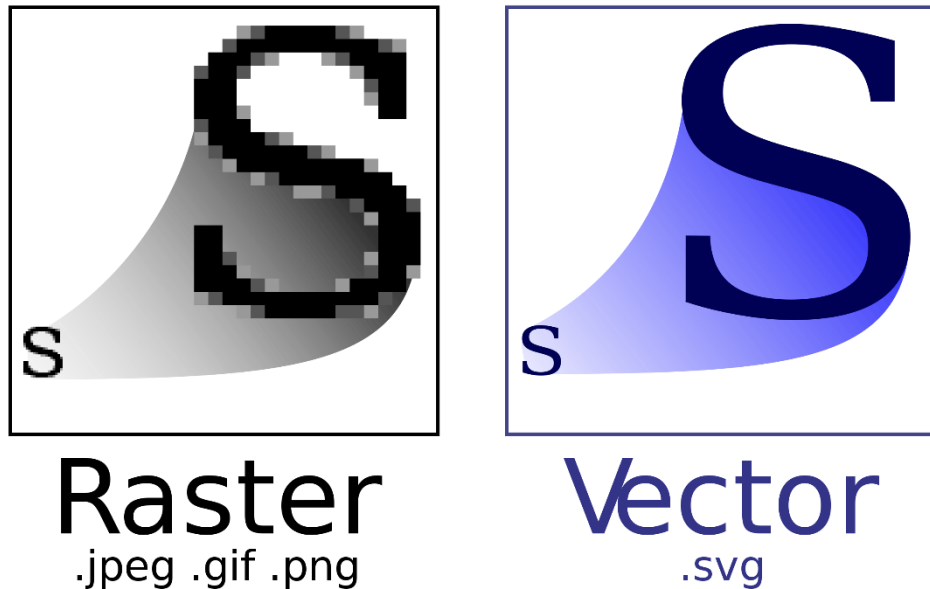


Рисунок 2.2 – Відмінність векторного та растрового зображень

Як ви можете бачити, векторні зображення завжди залишаються чіткими, незважаючи на збільшення розміру, а растрові, які звикли всі використовувати, втрачає чіткість та пікселізується.

Основними перевагами SVG є:

- Можливість використовувати усі функції XML. Завдяки цьому можливе використання в великій кількості середовищ, які мають підтримку або сумісність з XML, а тому мається і повна підтримка Юнікода.
- Сумісність з каскадними таблицями стилів. За допомогою CSS можна напряму взаємодіяти з SVG.
- Просте інтегрування в HTML-документ. Зовнішні SVG-елементи можна інтегрувати у веб-сторінку, використовуючи тег `<object>`.
- Відкритий стандарт. Тому на використання цього формату не потрібна ліцензія або спеціальний SDK.
- Написання анімацій через SMIL. Завдяки цьому можна робити динамічні сцени на сайті, і при цьому не використовувати відео або gif-файли, які дуже важкі для слабких девайсів або повільного інтернету.
- Пряма індексація. Так як SVG складається з тексту, тому вони легко індексуються пошуковими системами.

- Сумісність с растровими файлами, в середині можна використовувати не тільки векторні зображення або написані зображення, а й звичайні готові растрові.
 - Масштабованість. Через те, що написане зображення є векторним, то воно легко масштабується до будь-яких розмірів без втрати якості.
 - Не потребує стороннього ПО. Створювати SVG-файли можна у звичайному текстовому редакторі та відкривати у будь-якому сучасному браузері.
- Однак є і недоліки:
- Немає можливості працювати або створювати 3D-об'єкти.
 - Чим більше невеликих деталей у зображенні, тим більше файл SVG.
 - Присутні усі недоліки мови розмітки XML.

2.2 Таблиці стилів CSS

CSS – це каскадні таблиці, які задають стиль документу за допомогою мови розмітки, тобто їх можна назвати мовою стилю для HTML документів [11]. Хоча CSS найчастіше використовують разом із HTML або XHTML, однак їх можна застосовувати у будь-якому документі на основі XML. На рис. 2.3 зображено приклад коду CSS.

```
<style>
p.center {
    text-align: center;
    color: red;
}
</style>
</head>
<body>

<h1 class="center">Стиль не діє</h1>
<p class="center">Стиль діє</p>
<p>Стиль не діє</p>

</body>
```

Рисунок 2.3 – Приклад коду CSS

Якщо основною справою для HTML є надання інформації про типи даних у документі, то CSS надає інформацію про візуальну частину документу. Через CSS можна вказувати:

- розмір;
- місце знаходження елемента;
- прив'язку до іншого елемента;
- колір;
- шрифт;
- фон;
- відступ;
- адаптивність;

Хоча більшість стилів застосовуються до одного елемента, однак їх можна застосовувати зразу для декількох елементів, або одразу для всього сайту, як зображено на рис. 2.4, де стиль застосовано для декількох елементів одразу.

```
h1, h2, p {  
    text-align: center;  
    color: red;  
}
```

Рисунок 2.4 – Приклад коду CSS

Як ми бачимо, теги можна групувати, тим самим надавати різним елементам один і той же параметр, що дуже зручно для написання та зменшує об'єм файлу. Особливо, якщо урахувати те, що з кожним написаним рядком зростає об'єм файлу, і тому сторінка сайту завантажується довше при низькій швидкості інтернету.

CSS можна поділити на зовнішні та внутрішні, це залежить від того, де пишеться сам код. Якщо стилі пишуться на самій сторінці, де вони будуть

використовуватись, то це будуть внутрішні стилі. Вони пишуться з використанням тегу `style`, який відкривається конструкцією `<style>` та зачиняється `</style>`. Зовнішні CSS пишуться в окремому файлі, та зазвичай мають формат `.css`, а потім на цей файл посилаються в коді HTML-сторінки, наприклад `<link href="/cssfile.css" rel="stylesheet">`.

Використання зовнішніх CSS дуже полегшило створення красивих дизайнів для сайтів, та зробило їх завантаження ще швидшим. Бо зовнішні таблиці стилів можна використовувати на будь якій сторінці з якої йде посилання на них.

2.3 Мова JavaScript

На теперішній час, у більшості сучасних сайтів використовуються різні динамічні та інтерактивні об'єкти, такі як слайд-шоу або анімоване меню. Все це, та багато іншого, пишеться з використанням мови програмування JavaScript, щоб сайт був більш функціональним та інтерактивним [12].

Дана мова дуже схожа на мову C, це пов'язано з тим, що при створенні JS було взято велику кількість функцій з других мов, однак спочатку її робили як спрощену версію Java, яка була би зрозуміло більшості користувачам, навіть тим, хто не дуже добре розуміє програмування.

Всі JS-скрипти виконуються на комп'ютері користувача. Тим самим дозволяє асинхронно обмінюватися даними з сервером, та динамічно змінювати інформацію та будь-які об'єкти на сторінці у користувача. Однак це не означає, що на серверах не можливо виконувати JS, наприклад, Node.js надає таку можливість.

JavaScript використовується не тільки на сайтах, а й в різних веб-додатках, написаних на React або AngularJS. Це відноситься й до мобільних додатків, наприклад, JS використовується у React Native та Cordova. Використовувати мову JavaScript можливо навіть в середині документів PDF, тому це дуже практична та універсальна мова.

Як і CSS, JavaScript скрипти можна підключати двома способами. Перший, це пряме підключення, тобто написання скрипта на сторінці HTML використовуючи тег `<script>`, як показано на рис. 2.5.

```
<script>  
  alert("Hello, World!");  
</script>
```

Рисунок 2.5 – Приклад використання JS на сторінці HTML

Однак, перший спосіб не дуже зручний, бо він пишеться безпосередньо на сторінці HTML, тим самим збільшує розмір файлу, а також робить код сторінки дуже великим і складним для редагування. А головне, якщо потрібно використати цей же скрипт на іншій сторінці, то його потрібно повністю скопіювати на цю сторінку. Тому JS-скрипти краще підключати як зовнішні файли, що показано на рис. 2.6.

```
<script src="text/javascript">//Шлях_до_файла_зі_скриптом  
</script>
```

Рисунок 2.6 – Приклад зовнішнього підключення JS

Для більш швидкого завантаження сторінки, якщо скрипт підтримує, або його можна виконувати у будь-який час, не ламаючи хід завантаження сторінки, то до нього рекомендовано використовувати параметр `async`, який дозволяє виконувати скрипт асинхронно. Тобто, браузер не перестав завантажувати сторінку і виконувати подальші скрипти, навіть якщо виконання першого не закінчилося. Бо зазвичай, при завантаженні веб-сторінки, якщо браузер розпочинає завантаження та виконання скрипта, він очікує його завершення, і тільки потім починає завантажувати інші.

Можливості мови. JavaScript вважається безпечною мовою, що пов'язано з тим, що будь-який скрипт написаний на ній, немає доступу до процесів інших програм, або низькорівневого доступу до пам'яті. До основних можливостей можна віднести:

- додавання ефектів та анімацій;
- реагування на події;
- стеження за переміщенням миші;
- перевірка введення даних в полях;
- створення і зчитування cookie;
- визначення браузера;
- змінення вмісту HTML-елементів;
- додавання нових тегів;
- змінювання стилів.

Все перераховане вище тільки основні та часто використовувані можливості, насправді їх набагато більше, все залежить від розробника проекту.

2.4 Технологія AJAX

AJAX – це асинхронний JS та XML, який використовується при створенні веб-додатків, де потрібно оновлювати лише певну частину, або певний блок сторінки. Це дуже практично, бо не потрібно постійно перезавантажувати одну й ту саму сторінку, щоб оновити інформацію. З технологією AJAX ми маємо можливість фонові відправки запитів до сервера і отримання оновленого блока інформації.

Ця технологія дуже розповсюджена, так як вона дозволяє створити більш зручний інтерфейс для зворотного зв'язку та іншої взаємодії користувача з сайтом. Ця технологія використовується у таких великих проектах як Gmail, Google Maps та інших. На рис. 2.6 зображено принцип роботи сервісу Email, який в якості основи використовує AJAX.

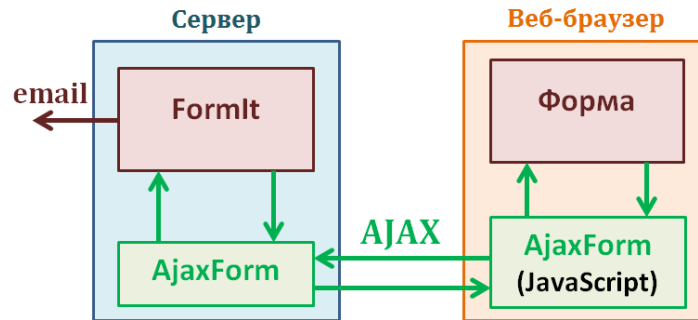


Рисунок 2.6 – Принцип роботи Email з AJAX

Послідовність дій при використанні моделі запитів AJAX наступна:

1. Користувач вводить URL адресу сайту.
2. Браузер надає запит до сервера.
3. Сервер відправляє сторінку.
4. Користувач тисне на елемент на сайті.
5. Веб-браузер надсилає новий запит.
6. Сервер відправляє лише ті частини сторінки, які потребують зміни.

Реалізація моделі запитів AJAX дозволяє економити трафік та дозволяє швидше завантажувати інформацію в порівнянні з моделлю звичайного запиту:

1. Користувач вводить URL адресу сайту.
2. Браузер надає запит до сервера.
3. Сервер відправляє сторінку.
4. Користувач тисне на елемент на сайті.
5. Веб-браузер надсилає новий запит.
6. Сервер знову відправляє всю сторінку.

Звичайний запит завжди повністю завантажує всі файли сторінки.

2.5 Фреймворк Bootstrap

Сучасні сервери дуже потужні, і можуть обробляти велику кількість інформації, однак з підвищення популярності ресурсу, збільшується кількість

користувачів, що може спричинити багато незручностей. Тому зараз усі фреймворки намагаються писати так, щоб більша частина формувалася на стороні клієнту. Тому фреймворк Bootstrap не виключення.

Bootstrap - це відкритий набір великої кількості інструментів, які можна використовувати при створенні сайту [13]. Він містить в собі веб-форми, блоки навігації та JavaScript-додатки. Через те, що Bootstrap - клієнтський фреймворк, який використовує сучасні та актуальні CSS з HTML, то потрібно брати до уваги, що користувач повинен мати сучасну версію браузера, щоб отримати повний функціонал. Зараз цей фреймворк настільки популярний що його використовують такі сервіси як Twitter і навіть сайт NASA.

Структура цього фреймворку складається переважно з наборів таблиць стилів, основними компонентами котрих є:

- Grid — готові, оформлені шаблони сітка;
- Template — адаптивні сторінки;
- Typography — визначення шрифту для цитат та форми коду;
- Media — управління медіафайлами, зображенням або відео;
- Table — управління, сортування та форматування таблиць;
- Form — оформлення форм та подій.

2.6 Бібліотека JQuery

Функціонал і можливості JavaScript дуже великі, однак навіть у неї є недоліки та обмеження. І тому було створено jQuery, яка спрощує роботу з JS та розширює її можливості, додаючи велику кількість наборів функцій для взаємодії з будь-яким елементом DOM. На рис. 2.7 зображено принцип того, як працює jQuery.

```

$('#test') //знаходимо елемент з id="test"
  .text('Клікни по мені') //встановлюємо текст елемента рівним "Клікни по мені"
  .addClass('myAlert') //додаємо клас "myAlert"
  .css('color','red') //встановлюємо колір тексту червоним
  .attr('alert','Привіт, світе!') // додаємо атрибут "alert" із значенням "Привіт, світе!"
  .bind( // додаємо в обробник події click функцію, яка відкриє модальне
    'click', // вікно із текстом, що вказаний в атрибуті "alert" ("Привіт, світе!")
    function(){alert($('#this').attr('alert'))}
  );

```

Рисунок 2.7 – Принцип роботи jQuery

При використанні бібліотеки ми маємо можливість, не тільки взаємодіяти з DOM та його атрибутами, а й використовувати API jQuery для взаємодії з AJAX, про котру було написано в пункті 2.4.

Підключення jQuery здійснюється за допомогою тега <script>, та зазвичай у частині <head> </head>, сама бібліотека працює з будь-яким браузером, тому не потрібно хвилюватися про сумісність з застарілими браузерами чи девайсами, на які неможливо встановити актуальні версії браузерів.

Основні можливості бібліотеки:

- адресація DOM з CSS;
- інтерфейси для Ajax додатків;
- візуальні ефекти;
- написання JavaScript-плагінів;
- створення подій;
- перехід по дереву DOM.

2.7 Мова PHP

PHP – це високорівнева мова сценаріїв, яка частіш за все застосовується у веб-додатках [14]. Мова настільки поширена, що використовується більшістю хостинг-провайдерів. А також, вона є однією із популярніших мов для створення динамічних сайтів, насамперед це пов'язано з тим, що вона є відкритим проектом і її можуть використовувати як для комерційних сайтів так і для персональних. А також через наявність великої кількості доповнень які розширюють функціонал PHP.

Зазвичай при написанні сайтів, пишуться веб-сторінки з урахуванням на те, що вони будуть оброблятися браузером, тобто вони являються клієнтською частиною, однак PHP-сторінки належать до серверної частини. До основних можливостей PHP можна віднести:

- Робота з POST та GET.
- Сумісність з великою кількістю різних баз даних, та з сервісним ПО.
- Запрограмовані сценарії посилання на HTTP-заголовки.
- Взаємодія з HTTP-авторизацією.
- Взаємодія з cookies та сесіями.
- Обробка завантажених файлів на сервер.
- Робота з XForms.

Синтаксис PHP дуже схожий с синтаксисом мови C, однак деякі елементи позичені з мови Perl. Також, дуже зручним є те, що у відмінність від інших мов тут не потрібно вказувати тип, коли об'являєш змінну.

2.7.1 Скрипти PHP

При використанні мови PHP не обов'язково створювати всю сторінку, використовуючи її синтаксис. Дуже часто розробники використовують движок PHP тільки для виконання певних скриптів, наприклад, за допомогою PHP можна брати інформацію з інших сайтів і ресурсів за допомогою запитів від сервера, тобто якщо потрібен курс валюти на сайті, то можна написати скрипт без використання JS для отримання будь-якої інформації з обраного ресурсу, і ця інформація буде завжди актуальною.

2.8 СУБД MySQL

Сьогодні кожен веб-проект, не залежно від його величини, скоріш за все використовує базу даних, бо навіть для простої на перший погляд функції

реєстрації, потрібно формувати БД де буде зберігатися інформація про користувачів.

Більшість баз даних являють собою таблиці, в яких може зберігатися майже будь-який тип інформації. Також БД підтримують шифрування для захисту інформації користувачів, таких як пароль, номер банківської карти, тощо. До версії 3.22 максимальний об'єм БД складав 4 ГБ, однак надалі це обмеження було знято.

MySQL настільки поширена, що його портували майже на усі доступні платформи, декілька з яких приведено нижче:

- FreeBSD,
- macOS,
- SunOS,
- Windows,
- WinCE,
- BSDi,
- Solaris,
- Linux.

Для взаємодії з БД, було створено спеціальне API, яке сумісне з великою кількістю мов програмування, тому такі поширені мови як C, C++, Java, Perl, Python та PHP з легкістю можуть взаємодіяти з базами даних MySQL.

2.9 Висновки до другого розділу

У цьому розділі було розглянуто усі засоби, які будуть використовуватись при створенні системи, а також потрібне програмне забезпечення та додаткові компоненти, які будуть підключені до створеної системи. Наприклад, разом із мовою JavaScript будуть використані засоби AJAX, JQuery та Bootstrap.

3 РОЗРОБКА СИСТЕМИ

3.1 Підготовка сервера

Створення та тестування системи буде проводитись на комп'ютері Raspberry Pi 4 (RPi), який складається з однієї плати, на якій розміщені всі інші елементи одноплатного комп'ютера [15], який зображений на рис. 3.1.

Дана версія містить 4 гігабайта оперативної пам'яті, якої цілком достатньо для більшості веб-проектів.

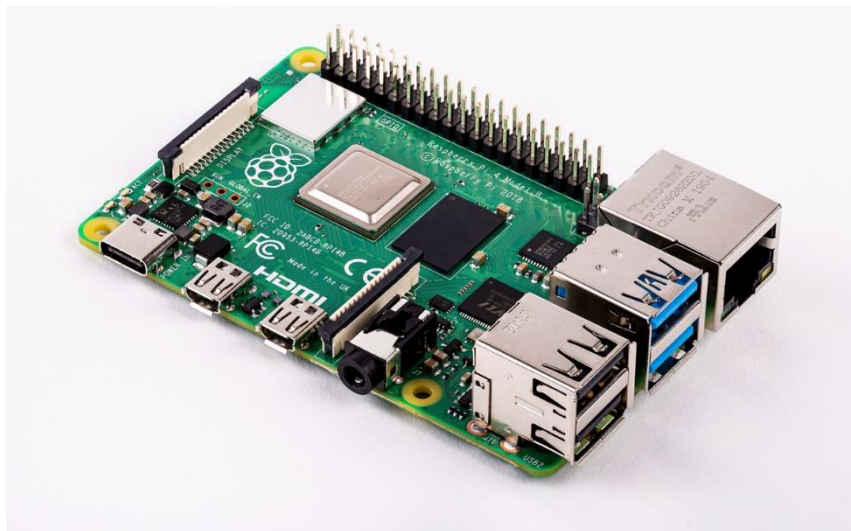


Рисунок 3.1 – Фотографія одноплатного комп'ютера Raspberry Pi 4

Повні характеристики комп'ютера, який буде використовуватись як сервер:

- процесор Broadcom BCM2711 з частотою 1.5 ГГц;
- ОЗУ 4 Гігабайта LPDDR4-3200;
- гігабітний інтернет-порт.

Використання локального сервера дозволить тимчасово використовувати сервер, не орендуючи хостинг, а при наявності приватного хостингу, безпечно

проводити тестування нових функцій та оновлень, перед тим як додати їх на основний сервер.

Операційною системою, на якій буде працювати система буде Ubuntu Server, яка призначена для підтримки веб-серверів різного призначення.

Ubuntu – операційна система, яка являється дистрибутивом Linux, заснованим на Debian GNU. Серед усіх Linux-дистрибутивів, він являється найбільш популярним дистрибутивом ОС, як для веб-серверів, так і для домашніх комп'ютерів. Дана система відноситься до легких, так як для її роботи потрібно 512 мегабайтів ОЗУ, тому вона підходить для нашого сервера на Raspberry Pi. Однак, слід зазначити, що на орендованих серверах також зустрічаються і інші серверні ОС, однак вони всі також використовують ядро Linux, тому налаштування будуть майже ідентичні. Часто хостинги створюють системи, які вже мають всі модулі, уже налаштованими.

Перше, що потрібно зробити після встановлення серверної ОС, це оновити всі її компоненти. Робиться це для гарантування максимальної безпеки, так як в нових версіях виправлено більшість вразливостей, та оптимізована швидкодія бібліотек, а також реалізовані новітні функції. Для цього потрібно в терміналі сервера виконати команди `apt update` та `apt full-upgrade`. Також, усі команди виконуються через `sudo`, так як нам потрібні права адміністратора, коли ми використовуємо non-root користувача. На рис. 3.2 зображено, як виглядає робота терміналу Ubuntu Server.

```

20 16:45:28 +0000. DataSource DataSourceNoCloud [seed=/var/lib/cloud/seed/nocloud-net] [dsmode=net].
Up 30.95 seconds
[ OK ] Finished Execute cloud user/final scripts.
[ OK ] Reached target Cloud-init target.

diplomsnu login: danil
Password:
Welcome to Ubuntu 20.04 LTS (GNU/Linux 5.4.0-33-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Fri May 29 16:46:11 UTC 2020

System load:  0.71          Processes:            127
Usage of /:   36.2% of 14.70GB   Users logged in:     0
Memory usage: 5%           IPv4 address for enp0s3:
Swap usage:   0%

22 updates can be installed immediately.
0 of these updates are security updates.
To see these additional updates run: apt list --upgradable

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

danil@diplomsnu:~$ apt update && apt full-upgrade

```

Рисунок 3.2 – Термінал Ubuntu Server

Після оновлення компонентів ОС переходимо до встановлення пакетів, потрібних для функціонування веб-сервера.

В першу чергу встановимо пакет, який називають «LAMP». Ця назва являється акронімом набору ПО, який розшифровується як «Linux, Apache, MySQL, PHP».

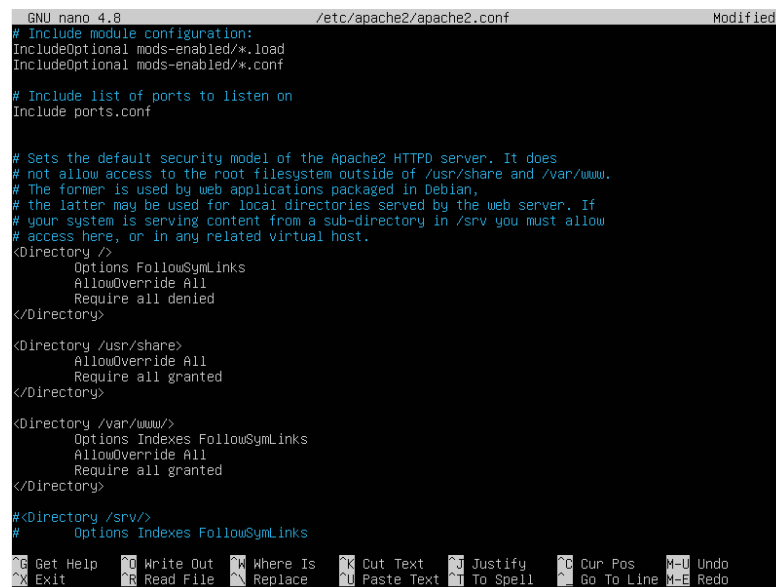
3.1.1 Установка та налаштування Apache

Першим встановлюємо Apache, для чого використовуємо команду «`sudo apt install apache2`». Після цього в фаєрволі, використовуючи команду «`sudo ufw allow in "Apache Full"`», дозволяємо з'єднання, використовуючи порти 80 та 443.

Після інсталювання переходимо до більш детального налаштування. Спочатку виконуємо команду «`sudo a2enmod rewrite`», тим самим активуємо модуль `mod_rewrite`, який призначений для конвертування URL, яке дозволяє використовувати файли PHP навіть тоді, якщо вони будуть мати інше розширення, а також кешувати сторінки, для котрих не потрібна динамічне оновлення даних, щоб сервер постійно не формував одну й ту саму PHP-сторінку, бо на слабких серверах, чи дуже завантажених, це може погано вплинути на швидкодію.

Також є модуль ssl, однак на локальній машині нам він не потрібний, його ми встановимо коли будемо все переносити на орендований хостинг-сервер.

Якщо у майбутньому ми будемо використовувати файли htaccess, то рекомендується у файлі apache2.conf, який знаходиться у папці /etc/apache2/, встановити значення AllowOverride в All. Для цього ми використаємо редактор nano. Виконавши команду «nano /etc/apache2/apache2.conf», ми зможемо редагувати цей файл, як показано на рис. 3.3



```
GNU nano 4.8 /etc/apache2/apache2.conf Modified
# Include module configuration:
IncludeOptional mods-enabled/*.load
IncludeOptional mods-enabled/*.conf

# Include list of ports to listen on
Include ports.conf

# Sets the default security model of the Apache2 HTTPD server. It does
# not allow access to the root filesystem outside of /usr/share and /var/www.
# The former is used by web applications packaged in Debian,
# the latter may be used for local directories served by the web server. If
# your system is serving content from a sub-directory in /srv you must allow
# access here, or in any related virtual host.
<Directory />
    Options FollowSymLinks
    AllowOverride All
    Require all denied
</Directory>

<Directory /usr/share>
    AllowOverride All
    Require all granted
</Directory>

<Directory /var/www/>
    Options Indexes FollowSymLinks
    AllowOverride All
    Require all granted
</Directory>

#<Directory /srv/>
#     Options Indexes FollowSymLinks
```

Рисунок 3.3 – Файл apache2.conf у редакторі nano

Для перевірки роботи сервера та apache, вводимо ір-адресу сервера у браузері. В даному випадку сервер встановлено на RPi, який під'єднано до загального маршрутизатора, тому ми можемо використовувати його локальну ір-адресу. При завантаженні сторінки маємо побачити стандартну сторінку, як вона показана нижче на рис. 3.4

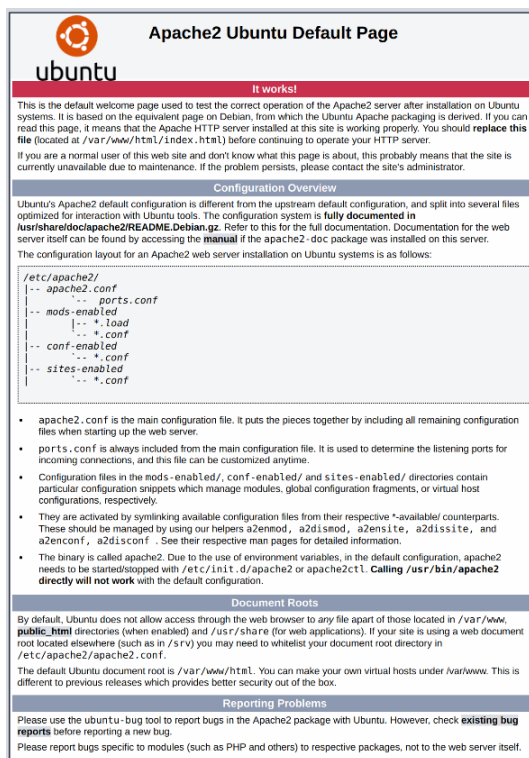


Рисунок 3.4 – Стандартна сторінка Apache

3.1.2 Установка та налаштування MySQL

Для роботи с базами даних потрібно встановити СУБД, використовуючи систему Ubuntu. Ми маємо можливість обрати та встановити MariaDB або усім відому MySQL, яку ми і будемо використовувати.

Командою «`sudo apt install mysql-server`» ми встановлюємо MySQL. Після цього потрібно налаштувати параметри безпеки, це є обов'язковим пунктом, якщо ми будемо використовувати БД від імені root, для чого вводимо команду «`sudo mysql_secure_installation`». Після цього з'явиться запит на встановлення «VALIDATE PASSWORD PLUGIN», на який, якщо ми відповімо «Yes», то всі паролі, які не відповідають правилам складного пароля, будуть відхилятися. Щоб мати можливість вказувати будь-який пароль, ми відповідаємо «No». Наступним потрібно ввести пароль для root-доступу до БД, а потім на запит видалення анонімних користувачів відповісти «Yes», як показано на рис. 3.5

```

Last login: Sat May 30 12:29:36 UTC 2020 on tty1
daniil@diplomsnu:~$ sudo mysql_secure_installation
[sudo] password for daniil:

Securing the MySQL server deployment.

Connecting to MySQL using a blank password.

VALIDATE PASSWORD COMPONENT can be used to test passwords
and improve security. It checks the strength of password
and allows the users to set only those passwords which are
secure enough. Would you like to setup VALIDATE PASSWORD component?

Press y|Y for Yes, any other key for No: n
Please set the password for root here.

New password:

Re-enter new password:
By default, a MySQL installation has an anonymous user,
allowing anyone to log into MySQL without having to have
a user account created for them. This is intended only for
testing, and to make the installation go a bit smoother.
You should remove them before moving into a production
environment.

Remove anonymous users? (Press y|Y for Yes, any other key for No) : yes_

```

Рисунок 3.5 – Налаштування безпеки MySQL

Після початкового налаштування MySQL вже можна створювати таблиці БД або приступити до налаштування PHP, однак перед цим нам потрібно дозволити підключатись до БД, використовуючи не тільки auth_socket, а також звичайний пароль, для чого потрібно увійти у MySQL та ввести команду «ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'tempparol'», яка дозволить root-користувачу приєднатися до бази даних за допомогою пароля.

Для створення бази даних нам потрібно увійти у MySQL, для цього використовується команда «mysql» з параметрами -u root -p «Наш пароль».

Коли ми увійшли, ми можемо розпочати створення БД, для цього виконуємо команду «CREATE DATABASE AndroidBS», тим самим ми створимо БД під назвою «AndroidBS».

Для створення таблиць обираємо нашу БД командою «USE AndroidBS», та створюємо таблицю User командою:

```

«CREATE TABLE instructor (
  ID CHAR(5),
  name VARCHAR(20) NOT NULL,
  login VARCHAR(20) NOT NULL,
  level VARCHAR(20),

```

```
password NUMERIC(8,2),  
);»
```

3.1.3 Установка та налаштування PHP

Після налаштування Apache та MySQL ми можемо приступити до створення сайту. Однак хоча сайт не буде повністю створений на PHP, нам все одно потрібно його встановити, щоб мати можливість використовувати скрипти на цій мові. Для цього встановимо три пакети: перший - це пакет php останньої версії; другий - це бібліотека для сумісної роботи PHP та Apache; третій - надає підтримку MySQL у PHP. Для інсталювання разом всіх пакетів виконуємо команду: «`sudo apt install php libapache2-mod-php php-mysql`».

Якщо потрібно, то можна встановити додаткові компоненти PHP, такі як «php-cli» та інші.

На цьому налаштування сервера завершено, і ми можемо приступити до створення нашої системи.

3.1.4 Підключення SSL-сертифікату та домену

Сучасний інтернет дуже небезпечний, тому пошукові системи намагаються надавати ресурси, які мають свій сертифікат SSL, який надає шифрування для трафіка HTTP, перетворюючи його у HTTPS.

Сертифікат видається спеціальними організаціями на фізичне ім'я або ім'я компанії і тільки один сертифікат на один домен.

Для нашого сервісу ми будемо використовувати сертифікат від Comodo, який називається Essential SSL. Він діє один рік, а далі його знову потрібно подовжувати. На рис. 3.6 показано, як виглядає інформація про сертифікат.

Сертифікат для androidbs.ru	
деталі	
Дата активації	2019-09-05 00:00:00
дата закінчення	2020-09-20 00:00:00
статус сертифіката	активний
статус перевірки	ok
ID замовлення в ЦС	
Операційна система	linux
хеш алгоритм	sha2
CSR	-----BEGIN CERTIFICATE REQUEST----- MIICZzCCAdMCAGAwgZoxCzAJBgNVBAYMAVBRMmEwYDVOQIDApMfWdhbnNlYXNl MRYuFAYDVQQGEA1TZkZlcm9kZS9HbWVhYDVOQIDApMfWdhbnNlYXNlCzAJ
параметри	
домен	androidbs.ru
додаткові домени	
спосіб перевірки	androidbs.ru () <input type="button" value="Наступні дії"/>
Розмір ключа	2048
організація	AndroidBS
Відділ	IT
Область / Region	
Населений пункт	
Країна	UA
Емалі адреса для підтвердження	
Контакт власника	Danylo Zinoviev, AndroidBS <input type="button" value="перейти"/>

Рисунок 3.6 – Інформація про SSL-сертифікат

Після отримання сертифікату, нам надається декілька сформованих ключів, які ми повинні встановити на сервер, а потім зробити переадресацію з HTTP на HTTPS.

Якщо в нас відсутній домен, то ми не можемо підключити SSL, однак користувачі все одно можуть приєднатися до нашого сайту, використовуючи нашу зовнішню ір-адресу з портом 80. Хоча вони матимуть можливість використовувати наш сервіс, однак це буде дуже незручно, бо система доменів робить запам'ятовування адреси простіше, і пошукові системи ставлять такі сайти на першу сторінку.

3.2 Структура системи

Структура розробляємої системи ієрархічна, тому потрібно врахувати зручність використання її. Створення великої кількості розділів може заплутувати користувача, не дуже важливі розділи краще перенести у підрозділи. Тому структура системи буде наступна (рис. 3.7).

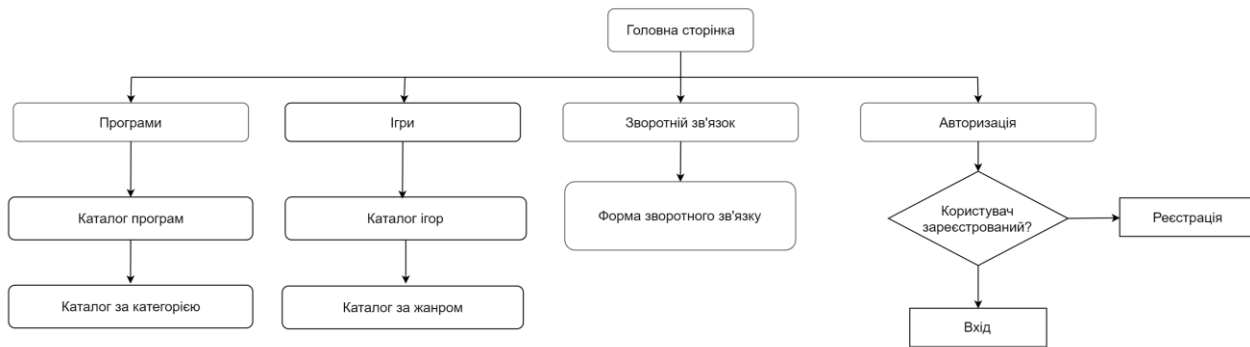


Рисунок 3.7 – Структура розробленої системи

3.3 Головна сторінка

Головна сторінка сайту (рис. 3.7), це сторінка, на яку потрапляє користувач, коли вводить домен сайту у браузері, чи коли шукає сайт за його назвою у пошукових системах.

Від того, як виглядає головна сторінка, залежить, буде користувач використовувати наш веб-додаток у майбутньому, чи ні. Тому він повинен бути інтуїтивно зрозумілий, та містити тільки необхідний мінімум інформації.

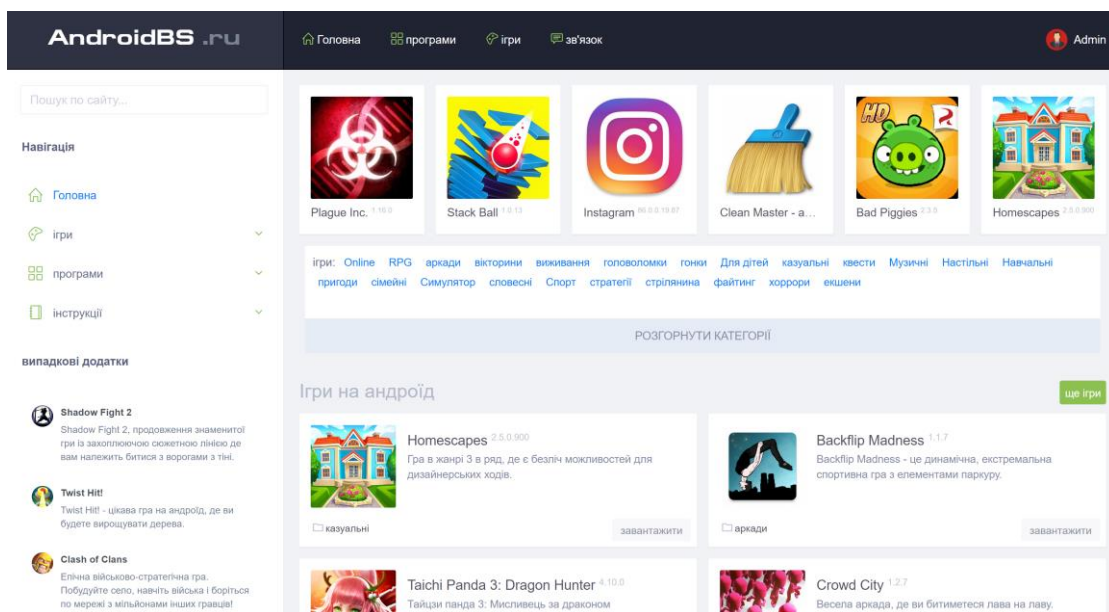


Рисунок 3.7 – Головна сторінка сайту

На головній сторінці користувача розташовані основне меню, піктограми найпопулярніших додатків, а також випадне меню з жанрами та категоріями ПО, яке приведено на рис. 3.8.

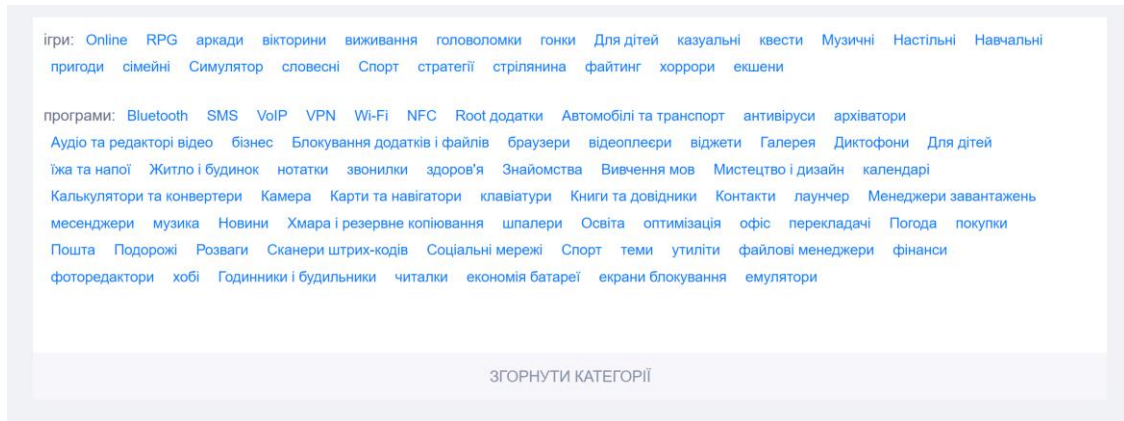


Рисунок 3.8 – Випадне меню

В цьому меню є поділ на категорії та жанри і воно розташовано тільки на головній сторінці та на сторінках зі списком додатків.

Для створення такого меню потрібно мати підключену бібліотеку jQuery. Тому у розділі <head> ми підключаємо її (була обрана полегшена версія jquery-1.12.4.min). Для цього, записуємо:

```
<script src="/src/jquery-1.12.4.min.js"></script>
```

Так як ми обрали полегшену версію, в ній відсутні деякі функції, тому їх ми підключаємо окремо через інші бібліотеки. Це пов'язано з тим, що ми будемо використовувати тільки потрібні нам функції, тим самим зменшимо загальний розмір сторінки, що дуже важливо для мобільних девайсів, які використовують 3G- та 2G-інтернет.

Меню складається з трьох частин: CSS частина, JavaScript та HTML частина.

Функція JS «moreContent» застосовується до обраного елемента div, наприклад:

```
$('#[menu1]').moreContent();
```

У HTML цей елемент, до якого застосовується функція виглядає так:

```
<div menu1>
    <p>тут знаходиться наше меню</p>
</div>
```

Інших змін в HTML-кодi нам робити не потрібно. Залишається тільки написати повний код JS, а також застосувати таблиці стилів CSS до вже створених елементів HTML.

JS частина меню має такий вигляд:

```
$(document).ready(function(){
$(a[href^="#"], *[data-href^="#"]).on('click', function(e){
e.preventDefault();
var t = 200;
var d = $(this).attr('data-href') ? $(this).attr('data-href') : $(this).attr('href');
$('html,body').stop().animate({ scrollTop: $(d).offset().top - 14 }, t);
});
});
$(document).ready(function() {
$('#Menu1').moreContent({
duration: 1000,
btn: 'Развернуть категории',
btnActive: 'Свернуть категории',
btnClass: 'btn2-default'
});
});
</script>
```

Щоб скрипт виконувався коректно, ми використовуємо функцію jQuery `$(document).ready(function() {})`, яка запускає скрипт тільки після повного формування DOM. Також, призначимо дію `click` до атрибута `href`, який буде викликати метод `preventDefault()`, що використовується для відміни дій браузера.

А атрибут `data-href` буде використовуватися для зберігання інформації в HTML коді, яка буде надаватися через функцію `attr('data-href')`.

Функції `duration`, `btn`, `btnActive` та `btnClass` надають підключені бібліотеки. За допомогою цих функцій, ми можемо робити налаштування нашого розгортання.

Повний лістинг коду, який включає в собі CSS, HTML та JS-частину, розміщено в додатку А.

На головній сторінці присутнє також і основне, звичайне меню, яке знаходиться на усіх сторінка. Це меню зображено на рис. 3.9.

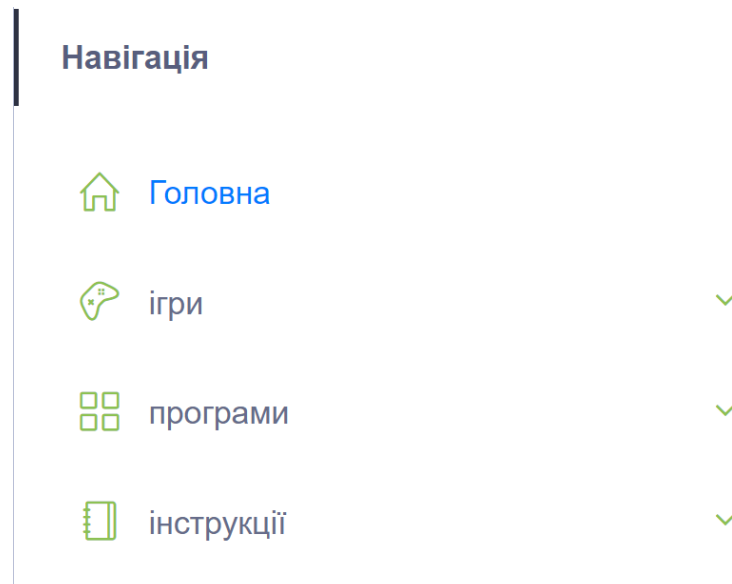


Рисунок 3.9 – Основне меню сайту

Незалежно від того, зареєстрований користувач чи ні, він має доступ до усіх категорій з додатками, та має можливість завантажувати будь-який додаток напряму з сайту, якщо додаток було завантажено на сервер.

Меню використовує стандартні HTML теги, тому його код виглядає так:

```
<div class="mnu">
<ul id="accordion" class="accordion" style="margin: 0;">
<li>
<a href="/" rel="nofollow"><div class="link"><i class="icon-home"></i>
Главная</div></a>
</li>
<li>
<div class="link"><i class="icon-game-controller"></i> Игры<i class="fa fa-
angle-down"></i></div>
<ul class="submenu">
<li><a href="/apps/igry_na_android/online/2">Online</a></li>
```

```

<li><a href="/apps/igry_na_android/rpg/3">RPG</a></li>
<li><a href="/apps/igry_na_android/arkady/4">Аркады</a></li>
<li><a href="/apps/igry_na_android/viktoriny/5">Викторины</a></li>
<li><a href="/apps/igry_na_android/vyzhivanie/6">Выживание</a></li>
<li><a href="/apps/igry_na_android/golovolomki/7">Головоломки</a></li>
<li><a href="/apps/igry_na_android/gonki/8">Гонки</a></li>
</ul>
</li>
<li>
<div class="link"><i class="icon-notebook"></i> Инструкции<i class="fa fa-angle-down"></i></div>
<ul class="submenu">
<li><a href="/">Установка приложений</a></li>
<li><a href="/index/pravila-publikacii/0-4">Добавление приложений на сайт</a></li>
</ul>
</li>
<li>
</li>
</ul>
</div>

```

3.4 Сторінка додатку

Після того як користувач обрав додаток, який він хоче завантажити, здійснюється перехід до сторінки обраного додатку, приклад якої зображено на рисунку 3.10

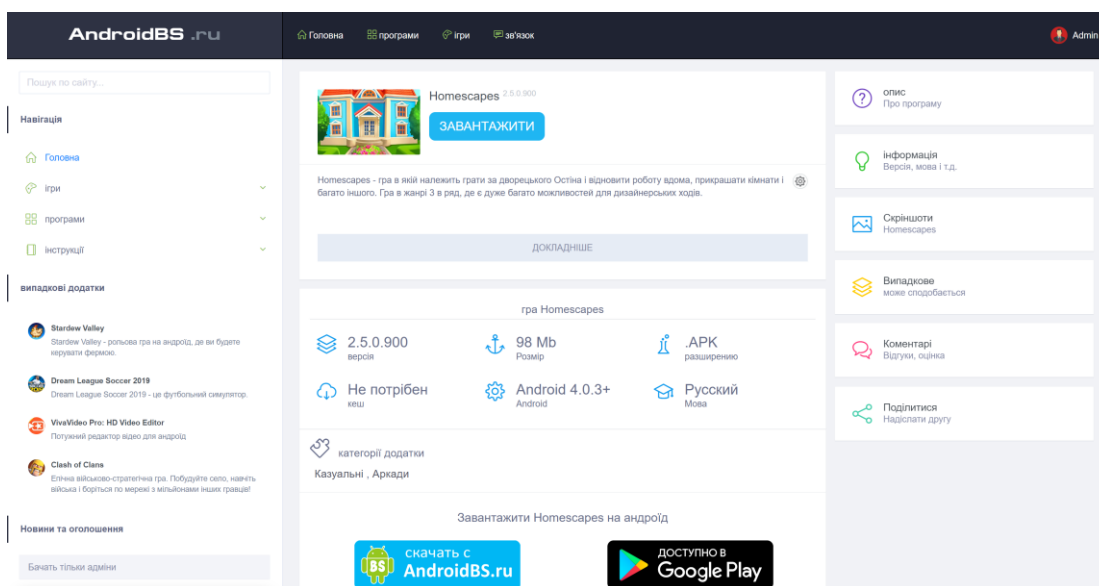


Рисунок 3.10 – Сторінка додатка

На сторінці додатка користувач може прочитати опис додатка, а також подивитись інформацію про файл, наприклад:

- розмір;
- версія;
- мова;
- версія андроїд;
- потрібність кешу;
- категорія;
- наявність у Play Market.

Крім того, на сторінці є скріншоти, відгуки, а також можливість поділитись сторінкою для завантаження додатка з сайту (рис. 3.11)

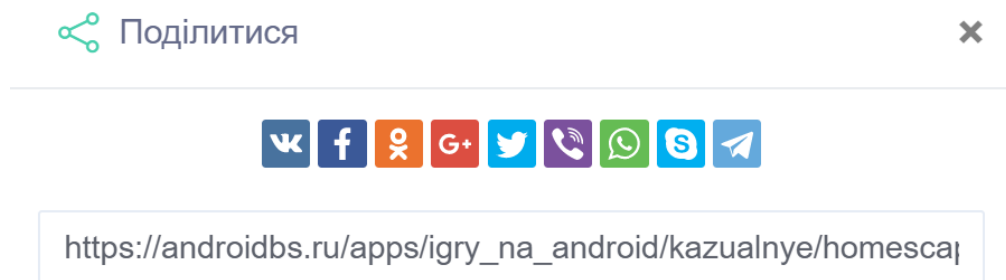


Рисунок 3.11 – Форма «Поділитись»

Якщо при завантаженні додатка на сайт не був написаний опис додатку, але була вказано посилання на сторінку в Google Play Market, то використовуючи скрипт на мові PHP (лістинг у додатку А), ми завантажуюмо опис додатку з Google Play Market.

Для цього ми використаємо функцію `file_get_contents` у PHP, яка поміщає все це у контейнер `konteiner1`. Це буде мати вигляд:

```
$content = file_get_contents('http://primer.saita/stranica');
```

Для перевірки того, що потрібна інформація завантажується, можливо вивести її, використовуючи `echo`, як показано нижче:

```
<?php
    $konteiner1 = file_get_contents('http://primer.saita/stranica');
    echo $konteiner1;
?>
```

Наступною дією ми вказуємо перший рядок коду з сайту, з якого ми беремо інформацію, для чого ми використовуємо функцію `strpos`.

```
$pos = strpos($konteiner1, 'текст або код зі сторінки іншого сайту');
```

А за допомогою функції `substr`, ми будемо видаляти усю інформацію до вказаної частини кода. Після цього, знову використовуючи `strpos`, ми вказуємо останній рядок коду, а `substr` видаляє всю інформацію після неї.

Якщо при завантаженні інформації з іншого сайту виникли проблеми з кодуванням, то в початок скрипта вставляємо «`header('Content-Type: text/html; charset=windows-1251');`»

Зліва на сторінці додатку є прикріплене меню (рис. 3.12), яке прокручується разом із сторінкою

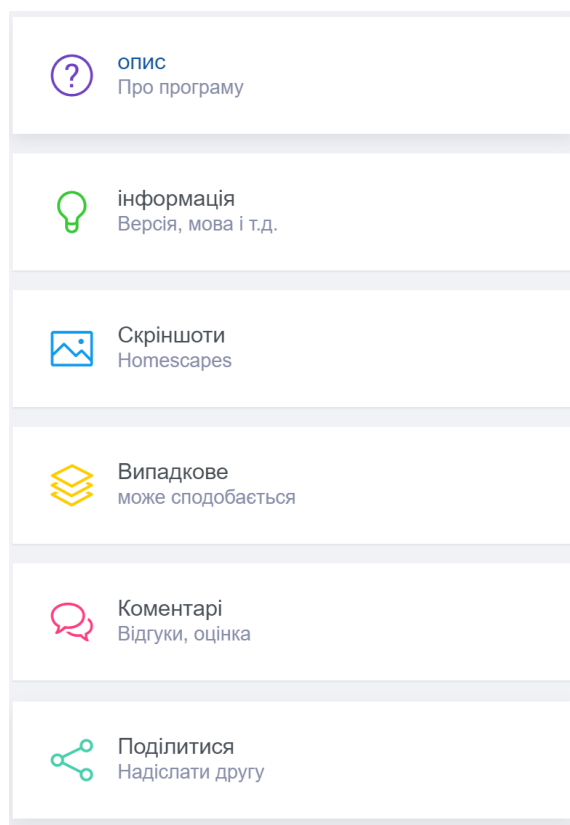


Рисунок 3.12 – Прикріплене меню

В даному меню використовується система id для div-елементів, створених у HTML, що дозволяє переміщатися, а точніше, скролити сторінку сайту, не перезавантажуючи її. Тобто сайт був поділений на кілька розділів за допомогою div, кожному div-елементу було присвоєно свій id, на цей id посилається кнопка у меню через href, у якому вказано адресу поточної сторінки та в кінці додано #id.

3.5 Адаптивність сторінок

Так як наш сервіс будуть використовувати і на смартфонах, ми повинні подбати про адаптивність до різних мобільних пристроїв. Для цього ми використаємо стилі CSS та фреймворк Bootstrap, за допомогою яких ми встановлюємо, як буде виглядати сайт при різних розширеннях.

До потрібних div-елементів ми додаємо класи, які будуть відповідати за певні розширення, які будуть вказані у CSS Bootstrap. Тоді мобільна версія головної сторінки матиме вигляд, як показано на рис. 3.13

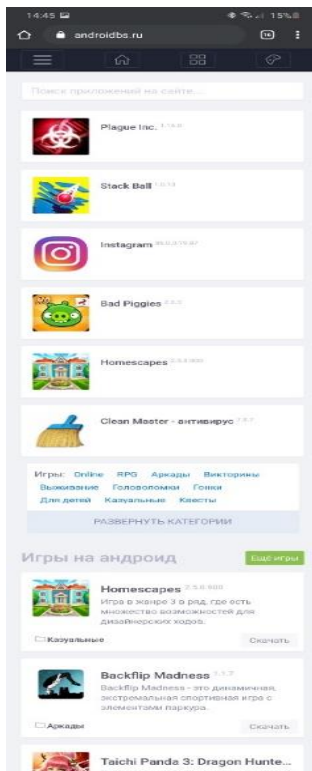


Рисунок 3.12 – Адаптована мобільна версія

3.6 Висновки до третього розділу

У третьому розділі було описано процес створення та налаштування інформаційно-комп'ютерної системи для обслуговування Android-додатків, який містить у собі:

- налаштування серверного комп'ютера;
- опис інтерфейсу основних сторінок;
- опис різних видів меню.
- опис інтерфейсу сторінки з інформацією об андроїд додатку;
- показано адаптивну версію для мобільних пристроїв

Також у розділі наведені ключові фрагменти коду та їх детальний опис. Повні версії коду можна знайти у додатку А.

4 ОХОРОНА ПРАЦІ

В даному розділі проведено аналіз потенційно небезпечних або шкідливих виробничих факторів, причин пожеж та інших надзвичайних ситуацій. Розглянуті заходи, які дозволяють забезпечити гігієну праці і виробничу санітарію. На підставі аналізу розроблені заходи з техніки безпеки та рекомендації з пожежної профілактики.

Завданням даної дипломної роботи бакалавра було створення веб-додатку з використанням мов JavaScript та PHP, і як результат було створено повністю робочий веб-додаток. Так як, в процесі використовувався комп'ютер, то аналіз потенційно небезпечних і шкідливих виробничих чинників виконується для персонального комп'ютера на якому буде розроблятися й тестуватися серверна та клієнтська частина веб-додатку.

4.1 Загальні питання з охорони праці

Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці. В законі України «Про охорону праці» [16] визначається, що охорона праці - це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних і лікувально-профілактичних заходів та засобів, спрямованих на збереження життя, здоров'я і працездатності людини у процесі трудової діяльності.

При роботі з обчислювальною технікою змінюються фізичні і хімічні фактори навколишнього середовища: виникає статична електрика, електромагнітне випромінювання, змінюється температура і вологість, рівень вміст кисню і озону в повітрі. Неправильна організація робочого місця сприяє загальному і локальній напрузі м'язів шиї, тулуба, верхніх кінцівок, викривлення

хребта і розвитку остеохондрозу. На всіх підприємствах, в установах, організаціях повинні створюватися безпечні і нешкідливі умови праці. Забезпечення цих умов покладається на власника або уповноважену ним особу. Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці. Роботодавець повинен впроваджувати сучасні засоби техніки безпеки, які запобігають виробничому травматизмові, і забезпечувати санітарно-гігієнічні умови, що запобігають виникненню професійних захворювань працівників. Він не має права вимагати від працівника виконання роботи, поєднаної з явною небезпекою для життя, а також в умовах, що не відповідають законодавству про охорону праці. Працівник має право відмовитися від дорученої роботи, якщо створилася виробнича ситуація, небезпечна для його життя чи здоров'я або людей, які його оточують, і навколишнього середовища.

4.1.1 Правові та організаційні основи охорони праці

Державна політика в галузі охорони праці визначається відповідно до Конституції України Верховною Радою України і спрямована на створення належних, безпечних і здорових умов праці, запобігання нещасним випадкам та професійним захворюванням. Відповідно до статті 3 Закону України «Про охорону праці» [16] законодавство про охорону праці складається з Закону, Кодексу законів про працю України [17], Закону України "Про загальнообов'язкове державне соціальне страхування від нещасного випадку на виробництві та професійного захворювання, які спричинили втрату працездатності" [18] та прийнятих відповідно до них нормативно-правових актів, норм міжнародного договору.

4.2 Аналіз стану умов праці

Робота над створенням веб-додатку буде проходити в приміщенні житлового будинку, де проживає студент. Для роботи достатньо однієї людини, для якої надано робоче місце зі стаціонарним або портативним комп'ютером.

4.2.1 Вимоги до приміщень

Геометричні розміри приміщення зазначені в табл. 4.1.

Таблиця 4.1 – Розміри приміщення

Найменування	Значення
Довжина, м	4.3
Ширина, м	3
Висота, м	2
Площа, м ²	12.9
Об'єм, м ³	25.8

Згідно з ДСН 3.3.6.042-99 [19] розмір площі для одного робочого місця оператора персонального комп'ютера має бути не менше 6 кв. м, а об'єм — не менше 20 куб. м. Отже, можна зробити висновок, що дане приміщення цілком відповідає зазначеним нормам.

4.2.2 Вимоги до організації місця праці

При порівнянні відповідності характеристик робочого місця з нормативними основними вимогами до організації робочого місця за ДСанПіН

3.3.2.007-98 [20] (табл. 4.2) і відповідними фактичними значеннями для робочого місця, констатуємо повну відповідність.

Таблиця 4.2 – Характеристика робочого місця

Найменування параметра	Фактичне значення	Нормативне значення
Висота робочої поверхні, мм	800	680 ÷ 800
Висота простору для ніг, мм	750	не менше 600
Ширина простору для ніг, мм	550	не менше 500
Глибина простору для ніг, мм	680	не менше 650
Висота поверхні сидіння, мм	430	400 ÷ 500
Ширина сидіння, мм	410	не менше 400
Глибина сидіння, мм	400	не менше 400
Висота поверхні спинки, мм	500	не менше 300
Ширина опорної поверхні спинки, мм	470	не менше 380
Радіус кривизни спинки в горизонтальній площині, мм	400	400
Відстань від очей до екрану дисплея, мм	730	700 ÷ 800

Приміщення кабінету знаходиться у двокімнатній квартирі на третьому поверсі та має об'єм 25.8 м³, площу — 12.9 м².

Температура в приміщенні протягом року коливається у межах 20–25°C, відносна вологість — близько 50%. Вентилювання приміщення — природна неорганізована, опалення — індивідуальне. Також, застосовуються очисник повітря та контроль вологості у приміщенні.

Згідно ДБН В.2.5-28:2018 [21] таке приміщення, площею 12.9 м², відноситься до категорії "В".

4.3 Виробнича санітарія

На підставі аналізу небезпечних та шкідливих факторів при виробництві (експлуатації), пожежної безпеки можуть бути надалі вирішені питання необхідності забезпечення працюючих достатньою кількістю освітлення, вентиляції повітря, організації заземлення, тощо.

4.3.1 Аналіз небезпечних та шкідливих факторів при виробництві (експлуатації) виробу

Аналіз небезпечних та шкідливих виробничих факторів виконується у табличній формі (табл. 4.3). Роботу, пов'язану з ЕОП з ВДТ, у тому числі на тих, які мають робочі місця, обладнані ЕОМ з ВДТ і ПП, виконують із забезпеченням виконання НПАОП 0.00-7.15-18 [22], яке встановлюють вимоги безпеки до обладнання робочих місць, під час роботи з екранними пристроями. Основними робочими характеристиками персонального комп'ютера є:

- робоча напруга $U = +220 \text{ В} \pm 5\%$;
- робочий струм $I = 3.5 \text{ А}$;
- споживана потужність $P = 230 \text{ Вт}$.

Таблиця 4.3 – Аналіз небезпечних і шкідливих виробничих факторів

Небезпечні шкідливі виробничі фактори	Джерела факторів	Кількісна оцінка	Нормативні документи
1	2	3	4
Фізичні			
Підвищена або знижена вологість повітря	Експлуатація ЕОМ, принтерів, сканерів або серверного обладнання для роботи	3	[1]

Продовження таблиці 4.3

Підвищений рівень напруги електричної мережі, замикання якої може відбуватися через тіло людини	Експлуатація ЕОМ, принтерів, сканерів або серверного обладнання для роботи	2	[2]
Недостатнє освітлення робочої зони	Порушення гігієнічних параметрів виробничого середовища	1	[4]
Психофізіологічні:			
Нервово-психічне перевантаження	- формулювання теми; - виконання роботи; - оформлення записки.	4	[3]
Фізичні (статичне-сидіння)	Порушення умов праці організації робочого місця (безперервна робота)	2	[2] [3]

4.3.2 Пожежна безпека

Пожежна безпека при застосуванні ЕОМ забезпечується:

- 1) системою запобігання пожежі;
- 2) системою протипожежного захисту;
- 3) організаційно-технічними заходами.

Потенційними джерелами запалювання можуть бути:

- 1) іскри і дуги короткого замикання;
- 2) електрична іскра при замиканні і розмиканні ланцюгів;
- 3) перегріву від тривалого перевантаження;
- 4) відкритий вогонь і продукти горіння;
- 5) наявність речовин, нагрітих вище за температуру самозаймання;
- 6) розрядна статична електрика.

Причинами можливого загоряння і пожежі можуть бути:

- 1) несправність електроустановки;

- 2) конструктивні недоліки устаткування;
- 3) коротке замикання в електричних мережах;
- 4) запалювання горючих матеріалів, що знаходяться в безпосередній близькості від електроустановки.

4.3.3 Електробезпека

На робочому місці виконуються наступні вимоги електробезпеки: ПК, периферійні пристрої та устаткування для обслуговування, електропроводи і кабелі за виконанням та ступенем захисту відповідають класу зони за ПУЕ (правила улаштування електроустановок), мають апаратуру захисту від струму короткого замикання та інших аварійних режимів. Лінія електромережі для живлення ПК, периферійних пристроїв і устаткування для обслуговування, виконана як окрема групова три-провідна мережа, шляхом прокладання фазового, нульового робочого та нульового захисного провідників. Нульовий захисний провідник використовується для заземлення (занулення) електроприймачів. Штепсельні з'єднання та електророзетки крім контактів фазового та нульового робочого провідників мають спеціальні контакти для підключення нульового захисного провідника. Електромережа штепсельних розеток для живлення персональних ПК, укладено по підлозі поруч зі стінами відповідно до затвердженого плану розміщення обладнання та технічних характеристик обладнання. Металеві труби та гнучкі металеві рукави заземлені. Захисне заземлення включає в себе заземлюючих пристроїв і провідник, який з'єднує заземлюючий пристрій з обладнанням, яке заземлюється - заземлюючий провідник.

4.4 Гігієнічні вимоги до параметрів виробничого середовища

4.4.1 Мікроклімат

Мікроклімат робочих приміщень - це клімат внутрішнього середовища цих приміщень, що визначається діючою на організм людини з'єднанням температури, вологості, швидкості переміщення повітря. В даному приміщенні проводяться роботи, що виконуються сидячи і не потребують динамічного фізичного напруження, то для нього відповідає категорія робіт 1а. Отже оптимальні значення для температури, відносної вологості й рухливості повітря для зазначеного робочого місця відповідають ДСН 3.3.6.042-99 [19] і наведені в табл. 4.4:

Таблиця 4.4 – Норми мікроклімату робочої зони об'єкту

Період року	Категорія робіт	Температура С ⁰	Відносна вологість, %	Швидкість руху повітря, м/с
Холодна	Легка-1а	22-24	40-60	0,1
Тепла	Легка-1а	23-25	40-60	0,1

4.4.2 Освітлення

Розрахунок освітлення.

Для виробничих та адміністративних приміщень світловий коефіцієнт приймається не менше -1/8, в побутових – 1/10:

$$\sqrt{a^2 + b^2} \cdot S_b = \left(\frac{1}{8} \div \frac{1}{10}\right) \times S_n, \quad (4.1)$$

де S_b – площа віконних прорізів, м²;

S_n – площа підлоги, м².

$$S_n = a \cdot b = 4,3 \cdot 3 = 12,9 \text{ м}^2,$$

$$S_{\text{вік}} = 1/8 \cdot 12,9 = 1,612 \text{ м}^2.$$

Приймаємо вікно площею $S=1.612 \text{ м}^2$.

Розрахунок штучного освітлення виробляється по коефіцієнтах використання світлового потоку. Розрахунок кількості світильників n виробляється за формулою (4.2):

$$n = \frac{E \cdot S \cdot Z \cdot K}{F \cdot U \cdot M} \quad (4.2)$$

де n – кількість світильників;

E – нормована освітленість робочої поверхні, визначається нормами – 300 лк;

S – освітлювана площа, м^2 ($S = 13 \text{ м}^2$);

Z – поправочний коефіцієнт світильника (1.15 для ламп розжарювання та ДРЛ; 1.1 для люмінесцентних ламп), приймаємо рівним 1.1;

K – коефіцієнт запасу, що враховує зниження освітленості в процесі експлуатації – 1.5;

U – коефіцієнт використання, залежний від типу світильника, показника індексу приміщення і т.п., 0.575М ;

M – число люмінесцентних ламп в світильнику, 1 одиниця;

F – світловий потік - 5400лм (для ЛБ-80).

Згідно вимог ДБН В.2.5-28:2018 [21] зорова робота при експлуатації персонального комп'ютера відноситься до робіт середньої зорової точності (розряд IV, підрозряд «а»), освітленість робочого місця оператора обчислювальної техніки повинна бути 300 лк. В результаті підстановки числових значень у формулу (4.2) кількість світильників:

$$n = \frac{E \cdot S \cdot Z \cdot K}{F \cdot U \cdot M} = \frac{300 \cdot 13 \cdot 1.15 \cdot 1.5}{5400 \cdot 0.575 \cdot 1} = 2.16 \approx 2$$

Приймаємо освітлювальну установку, яка складається з 2-х світильників, оснащених лампами типу ЛБ (одна – 80 Вт) зі світловим потоком 5400 лм.

4.5 Вентилювання

У приміщенні, де знаходяться ЕОМ, повітрообмін реалізується за допомогою природної організованої вентиляції (вентиляційні шахти), тобто при V приміщення > 40 м³ на одного працюючого допускається природна вентиляція. Цей метод забезпечує притік потрібної кількості свіжого повітря, що визначається в СНіП. Також має здійснюватися провітрювання приміщення, в залежності від погодних умов, тривалість повинна бути не менше 10 хв. Найкращий обмін повітря здійснюється при наскрізному провітрюванні.

4.6 Заходи з організації виробничого середовища та попередження виникнення надзвичайних ситуацій

1) Заходи безпеки під час експлуатації персонального комп'ютера та периферійних пристроїв передбачають:

- правильне організування місця праці та дотримання оптимальних режимів праці та відпочинку під час роботи з ПК;

- експлуатацію сертифікованого обладнання;
- дотримання заходів електробезпеки;
- забезпечення оптимальних параметрів мікроклімату;

- забезпечення раціонального освітлення місця праці (освітленість робочого місця не перевищувала $2/3$ нормальної освітленості приміщення);

- облаштовуючи приміщення для роботи з ПК, потрібно передбачити припливно-витяжну вентиляцію або кондиціонування повітря;

2) Заходи безпеки під час експлуатації інших електричних приладів передбачають дотримання таких правил:

- постійно стежити за справним станом електромережі, розподільних щитків, вимикачів, штепсельних розеток, лампових патронів, а також мережевих кабелів живлення, за допомогою яких електроприлади під'єднують до електромережі;

- постійно стежити за справністю ізоляції електромережі та мережевих кабелів, не допускаючи їхньої експлуатації з пошкодженою ізоляцією;
- не тягнути за мережевий кабель, щоб витягти вилку з розетки;
- не закривати меблями, різноманітним інвентарем вимикачі, штепсельні розетки;

4.6.1 Розрахунок захисного заземлення (забезпечення електробезпеки будівлі)

Згідно з класифікацією приміщень за ступенем небезпеки ураження електричним струмом НПАОП 40.1-1.01-97 [23], приміщення в якому проводяться всі роботи відноситься до першого класу (без підвищеної небезпеки). Під час роботи використовуються електроустановки з напругою живлення 36 В, 220 В, та 360 В. Опір контуру заземлення повинен мати не більше 4 Ом.

Послідовність розрахунку.

1) Визначається необхідний опір штучних заземлювачів $R_{шт.з.}$:

$$R_{шт.з.} = \frac{R_{\delta.} \cdot R_{пр.з.}}{R_{пр.з.} - R_{\delta.}}, \quad (4.3)$$

де $R_{пр.з.}$ – опір природних заземлювачів;

R_{δ} – допустимий опір заземлення.

Якщо природні заземлювачі відсутні, то $R_{шт.з.} = R_{\delta}$.

Підставивши числові значення у формулу (4.3), отримуємо:

$$R_{шт.з.} = \frac{4 \cdot 40}{40 - 4} \approx 4 \text{ Ом}$$

2) Опір заземлення в значній мірі залежить від питомого опору ґрунту ρ , Ом·м. Приблизне значення питомого опору глини приймаємо $\rho=40$ Ом·м (табличне значення).

3) Розрахунковий питомий опір ґрунту, $R_{\text{розр}}$, Ом·м, визначається відповідно для вертикальних заземлювачів $R_{\text{розр.в}}$, і горизонтальних $R_{\text{розр.г}}$, Ом·м за формулою:

$$R_{\text{розр.}} = \psi \cdot \rho \quad (4.4)$$

де ψ – коефіцієнт сезонності для вертикальних заземлювачів і кліматичної зони з нормальною вологістю землі, приймається для вертикальних заземлювачів $\rho_{\text{розр.в}} = 1.7$ і горизонтальних $\rho_{\text{розр.г}} = 5.5$ Ом·м.

$$R_{\text{розр.в.}} = 1.7 \cdot 40 = 68 \text{ Ом/м}$$

$$R_{\text{розр.г.}} = 5.5 \cdot 40 = 220 \text{ Ом/м}$$

4) Розраховується опір розтікання струму вертикального заземлювача $R_{\text{в}}$, Ом, за (4.5).

$$R_{\text{в}} = \left(\ln \frac{2 \cdot l_{\text{в}}}{d_{\text{ст}}} + \frac{1}{2} \cdot \ln \frac{4 \cdot t + l_{\text{в}}}{4 \cdot t - l_{\text{в}}} \right) \quad (4.5)$$

де $l_{\text{в}}$ – довжина вертикального заземлювача (для труб - 2–3 м; $l_{\text{в}}=3$ м);

$d_{\text{ст}}$ – діаметр стержня (для труб - 0,03–0,05 м; $d_{\text{ст}}=0,05$ м);

t – відстань від поверхні землі до середини заземлювача, яка визначається за ф. (4.6);

$$t = h_B + \frac{1_B}{2} \quad (4.6)$$

де h_B – глибина закладання вертикальних заземлювачів (0,8 м); тоді

$$t = 0.8 + \frac{3}{2} = 2,3 \text{ м}$$

$$R_B = \frac{68}{2 \cdot \pi \cdot 3} \cdot \left(\ln \frac{2 \cdot 3}{0.05} + \frac{1}{2} \cdot \ln \frac{4 \cdot 2.3 + 3}{4 \cdot 2.3 - 3} \right) = 18.5 \text{ Ом}$$

5) Визначається теоретична кількість вертикальних заземлювачів n штук, без урахування коефіцієнта використання η_B :

$$n = \frac{2 \cdot R_B}{R_D} = \frac{2 \cdot 18.5}{4} = 9.25$$

(4.7)

6) Визначається необхідна кількість вертикальних заземлювачів з урахуванням коефіцієнта використання η_B , шт:

$$n_B = \frac{2 \cdot R}{R_D \cdot \eta_B} = \frac{2 \cdot 18.5}{4 \cdot 0.57} = 16.2 \approx 16$$

(4.8)

7) Визначається довжина з'єднувальної стрічки горизонтального заземлювача l_C ,

$$l_C = 1.05 \cdot L_B \cdot (n_B - 1)$$

(4.9)

де l_B – відстань між вертикальними заземлювачами, (прийняти за $l_B = 3\text{м}$);
 n_B – необхідна кількість вертикальних заземлювачів.

$$l_c = 1.05 \cdot 3 \cdot (16 - 1) \approx 48\text{м}$$

8) Визначається опір розтіканню струму горизонтального заземлювача R_Γ ,
 Ом:

$$R_\Gamma = \frac{\rho_{\text{розр.}\Gamma}}{2 \cdot \pi \cdot l_c} \cdot \ln \frac{2 \cdot l_c^2}{d_{\text{см}} \cdot h} \quad (4.10)$$

де $d_{\text{см}}$ – еквівалентний діаметр смуги шириною b , $d_{\text{см}} = 0,95b$, $b = 0,15\text{ м}$;

h_Γ – глибина закладання горизонтальних заземлювачів (0.5 м);

l_c – довжина з'єднувальної стрічки горизонтального заземлювача l_c , м

$$R_\Gamma = \frac{220}{2 \cdot \pi \cdot 48} \cdot \ln \frac{2 \cdot 48^2}{0.95 \cdot 0.15 \cdot 0.5} = 8.1 \text{ Ом}$$

9) Визначається коефіцієнт використання горизонтального заземлювача η_c .
 відповідно до необхідної кількості вертикальних заземлювачів n_B .

Коефіцієнт використання з'єднувальної смуги $\eta_c = 0,3$ (табличне значення).

10) Розраховується результуючий опір заземлювального електроду з
 урахуванням з'єднувальної смуги:

$$R_{\text{заг}} = \frac{R_B \cdot R_\Gamma}{R_B \cdot n_c \cdot R_\Gamma \cdot n_B \cdot n_B} \leq R_d, \quad (4.11)$$

Висновок: дане захисне заземлення буде забезпечувати електробезпеку будівлі, так як виконується умова: $R_{\text{заг}} < 4 \text{ Ом}$, а саме:

$$R_{\text{заг}} = \frac{18.5 \cdot 8.1}{18.5 \cdot 0.3 + 8.1 \cdot 16 \cdot 0.57} = 1.9 \leq R_{\text{д}}$$

4.7 Висновки до четвертого розділу

В результаті проведеної роботи було зроблено аналіз умов праці, шкідливих та небезпечних чинників, з якими стикається робітник. Було визначено параметри і певні характеристики приміщення для роботи над запропонованим проектом написаному в кваліфікаційній роботі, описано, які заходи потрібно зробити для того, щоб дане приміщення відповідало необхідним нормам і було комфортним і безпечним для робітника.

Приведені рекомендації щодо організації робочого місця, а також важливу інформацію щодо пожежної та електробезпеки. Були наведені розміри приміщення та значення температури, вологості й рухливості повітря, необхідна кількість ламп та інші параметри, значення яких впливає на умови праці робітника, а також – наведені інструкції з охорони праці, техніки безпеки при роботі на комп'ютері.

ВИСНОВКИ

Результатом дипломної роботи є повністю розроблена та впроваджена інформаційно-комп'ютерна система обслуговування Android-додатків, яка дозволяє незалежно від пристрою завантажувати додатки для ОС Android через будь який браузер, не встановлюючи спеціальне ПО.

Перед створенням системи, було проаналізовані вже існуючі аналоги, щоб уникнути помилок при створенні. Крім цього, було досліджено різні сучасні мови, доповнення, та фреймворки, які були застосовані при розробці системи.

Розроблена система має зручний, швидкий, та інтуїтивно зрозумілий інтерфейс, який здатний адаптуватись під різні пристрої та браузери. При створенні було застосовані актуальні версії бібліотек та фреймворків, які забезпечують безпеку системи та надають можливість використовувати нові функції.

На сьогоднішній день, пристрої які використовують мобільну систему Android займають більшу частину ринку мобільних пристроїв, тому веб-додатки, які допомагають користувачам цих пристроїв розширювати можливості своїх смартфонів за допомогою андроїд-додатків, дуже актуальні.

ПЕРЕЛІК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Інтернет [Електронний ресурс] // Вікіпедія. URL: <https://uk.wikipedia.org/wiki/Інтернет> (дата звернення 20.05.2020)
2. Протокол передачі даних [Електронний ресурс] // Вікіпедія. URL: https://ru.wikipedia.org/wiki/Протокол_передачи_данных (дата звернення 20.05.2020)
3. Сайт [Електронний ресурс] // Вікіпедія. URL: <https://ru.wikipedia.org/wiki/Сайт> (дата звернення 20.05.2020)
4. Облачные сервисы для игры [Електронний ресурс] // Хабр. URL: <https://habr.com/ru/post/454104/> (дата звернення 21.05.2020)
5. Веб-додаток [Електронний ресурс] // Вікіпедія URL: <https://uk.wikipedia.org/wiki/Веб-застосунок> (дата звернення: 25.05.2020).
6. Процессор Intel® Xeon® E5-2697 [Електронний ресурс] // Вікіпедія URL: <https://ark.intel.com/content/www/ru/ru/ark/products/75283/intel-xeon-processor-e5-2697-v2-30m-cache-2-70-ghz.html> (дата звернення: 30.05.2020).
7. Процессор Intel® Core™ i7-8700K [Електронний ресурс] // Вікіпедія URL: <https://ark.intel.com/content/www/ru/ru/ark/products/126684/intel-core-i7-8700k-processor-12m-cache-up-to-4-70-ghz.html> (дата звернення: 30.05.2020).
8. HTML [Електронний ресурс] // Вікіпедія URL: <https://ru.wikipedia.org/wiki/HTML> (дата звернення: 25.05.2020).
9. Альфа-канал [Електронний ресурс] // Вікіпедія URL: <https://ru.wikipedia.org/wiki/Альфа-канал> (дата звернення: 22.05.2020).
10. SVG [Електронний ресурс] // Вікіпедія URL: <https://ru.wikipedia.org/wiki/SVG> (дата звернення: 22.05.2020).
11. CSS [Електронний ресурс] // Вікіпедія URL: <https://uk.wikipedia.org/wiki/CSS> (дата звернення: 22.05.2020).
12. Дэвид Флэнаган . JavaScript. Подробное руководство. – Пер. с англ. – СПб: Символ_Плюс,2008. – 992 с.

13. Bootstrap [Електронний ресурс] // Вікіпедія URL: <https://uk.wikipedia.org/wiki/Bootstrap> (дата звернення: 30.05.2020).
14. PHP [Електронний ресурс] // Вікіпедія URL: <https://uk.wikipedia.org/wiki/PHP> (дата звернення: 30.05.2020).
15. Raspberry Pi [Електронний ресурс] // Вікіпедія URL: https://uk.wikipedia.org/wiki/Raspberry_Pi (дата звернення: 31.05.2020).
16. Закон України "Про охорону праці". Вводиться в дію Постановою ВР № 2695-ХІІ від 14.10.92, ВВР, 1992, № 49, ст.669. - Режим доступу: [www. URL: https://zakon.rada.gov.ua/laws/show/2694-12](http://www.zakon.rada.gov.ua/laws/show/2694-12)
17. Кодекс законів про працю України. Затверджується Законом № 322-VIII від 10.12.71 ВВР, 1971. Режим доступу: [www. URL: https://zakon.rada.gov.ua/laws/show/322-08](http://www.zakon.rada.gov.ua/laws/show/322-08)
18. Закон України "Про загальнообов'язкове державне соціальне страхування від нещасного випадку на виробництві та професійного захворювання, які спричинили втрату працездатності". Наказ від 21 грудня 2000 року N 2180-III. Режим доступу: [www. URL: https://dnaop.com/html/2065/doc-zakon-ukrajini-pro-zagalynoobovjazkove-derzhavne-socialynestrahuvannya-vid-neshhasnogo-vipadku-na-virobnictvi-ta-profesijnogo-z](http://www.dnaop.com/html/2065/doc-zakon-ukrajini-pro-zagalynoobovjazkove-derzhavne-socialynestrahuvannya-vid-neshhasnogo-vipadku-na-virobnictvi-ta-profesijnogo-z)
19. Санітарні норми мікроклімату виробничих приміщень ДСН 3.3.6.042-99. Постанова N 42 від 01.12.99. Режим доступу: [www. URL: https://zakon.rada.gov.ua/rada/show/va042282-99](http://www.zakon.rada.gov.ua/rada/show/va042282-99)
20. Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин ДСанПІН 3.3.2.007-98. Затверджено Постановою Головного державного санітарного лікаря України 10 грудня 1998 р. N 7. Режим доступу: [www. URL: https://zakon.rada.gov.ua/rada/show/v0007282-98](http://www.zakon.rada.gov.ua/rada/show/v0007282-98)
21. ДБН В.2.5-28:2018 «Природне і штучне освітлення». Режим доступу: [www. URL: http://www.minregion.gov.ua/wp-content/uploads/2018/12/V2528-1.pdf](http://www.minregion.gov.ua/wp-content/uploads/2018/12/V2528-1.pdf)
22. НПАОП 0.00-7.15-18 «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями». Зареєстровано в

Міністерстві юстиції України 25 квітня 2018 р. за № 508/31960. Режим доступу:
www. URL: <https://zakon.rada.gov.ua/laws/show/z0508-18>

23. НПАОП 40.1-1.01-97 «Правил безпечної експлуатації електроустановок». Режим доступу: www. URL: <https://zakon.rada.gov.ua/laws/show/z0011-98>

ДОДАТОК А

Лістинги програм

PHP скрипт для отримання опису з Google Play Market

```

<?php
    header('Content-Type: text/html; charset=windows-1251');
    $konteiner1 =
file_get_contents(https://play.google.com/store/apps/details?id=ИД_СТОРИНКИ
_ДОДАТКУ');
    $pos = strpos($konteiner1, '<span jsslot="">');
    $ konteiner1 = substr($konteiner1, $pos);
    $pos = strpos($konteiner1, '</span>');
    $ konteiner1 = substr($konteiner1, 0, $pos);
    echo $ konteiner1;
?>

```

Розгортаєме меню

```

<style>
    .tags-items ul {
        margin-top: 0;
        margin-bottom: 12px;
        padding-left: 50px;
        list-style: none;
        display: block;
        margin-right: 0px;
    }
    .tags-items ul li {
        display: inline-block;
        padding: 1px 0px;
        font-size: 14px;
    }

    .tagsname {
        font-size: 14px;
    }

    .tags-items {
        z-index: 1;
        text-align: left;
        margin-top: 9px;
        margin-bottom: 20px;
        position: relative;
        left: -40px
    }

```

```

.tags-items .a{
  padding: 2px 7px 2px 7px;
  display: block;
}
.tags-items .a.active,
.tags-items .a:hover {
  color: white;
  background: #1FB6F2;
}
.div_tags_menu_top .a:first-child {
  border-left: 0px !important
}
</style>

<script>
$(document).ready(function(){
$('a[href^="#"], *[data-href^="#"]').on('click', function(e){
e.preventDefault();
var t = 200;
var d = $(this).attr('data-href') ? $(this).attr('data-href') :
$(this).attr('href');
$('html,body').stop().animate({ scrollTop: $(d).offset().top - 14 }, t);
});
});
$(document).ready(function() {
$('.moreContent').moreContent({
duration: 1000,
btn: 'Развернуть категории',
btnActive: 'Свернуть категории',
btnClass: 'btn2-default'
});
});
</script>
<div class="col-lg-12 my">
  <div class="card mb15">
    <div class="moreContent" data-height="80">
      <div class="tags-items">
<ul>
Игры:
<li><a class="a" href="/apps/igry_na_android/online/2">Online</a></li>
<li><a class="a" href="/apps/igry_na_android/rpg/3">RPG</a></li>
<li><a class="a" href="/apps/igry_na_android/arkady/4">Аркады</a></li>
<li><a
class="a"
href="/apps/igry_na_android/viktoriny/5">Викторины</a></li>
<li><a
class="a"
href="/apps/igry_na_android/vyzhivanie/6">Выживание</a></li>
<li><a
class="a"
href="/apps/igry_na_android/golovolomki/7">Головоломки</a></li>
<li><a class="a" href="/apps/igry_na_android/gonki/8">Гонки</a></li>
<li><a
class="a"
href="/apps/igry_na_android/dlja_detej/9">Для
детей</a></li>

```

[Казуальные](/apps/igry_na_android/kazualnye/10)
[Квесты](/apps/igry_na_android/kvesty/11)
[Музыкальные](/apps/igry_na_android/muzykalnye/12)
[Настольные](/apps/igry_na_android/nastolnye/13)
[Обучающие](/apps/igry_na_android/obuchajushhie/14)
[Приключения](/apps/igry_na_android/prikljuchenija/15)
[Семейные](/apps/igry_na_android/semejnye/16)
[Симуляторы](/apps/igry_na_android/simuljatory/17)
[Словесные](/apps/igry_na_android/slovesnye/18)
[Спорт](/apps/igry_na_android/sport/19)
[Стратегии](/apps/igry_na_android/strategii/20)
[Стрелялки](/apps/igry_na_android/streljalki/21)
[Файтинги](/apps/igry_na_android/fajtingi/23)
[Хорроры](/apps/igry_na_android/khorroru/22)
[Экшены](/apps/igry_na_android/ehksheny/24)

Программы:

[Bluetooth](/apps/programmy_na_android/bluetooth/72)
[SMS](/apps/programmy_na_android/sms/55)
[VoIP](/apps/programmy_na_android/voip/56)
[VPN](/apps/programmy_na_android/vpn/75)
[Wi-Fi](/apps/programmy_na_android/wi-fi/76)
[NFC](/apps/programmy_na_android/nfc/89)
[Root приложения](/apps/programmy_na_android/root_prilozhenija/74)
[Автомобили и транспорт](/apps/programmy_na_android/avtomobili_i_transport/77)
[Антивирусы](/apps/programmy_na_android/antivirusy/78)
[Архиваторы](/apps/programmy_na_android/arkhivatory/79)
[Аудио и Видеоредакторы](/apps/programmy_na_android/audio_i_videoredactory/61)
[Бизнес](/apps/programmy_na_android/biznes/34)

```

<li><a class="a" href="/apps/programmy_na_android/blokirovka/80">Блокировка
приложений и файлов</a></li>
<li><a class="a"
href="/apps/programmy_na_android/brauzery/81">Браузеры</a></li>
<li><a class="a"
href="/apps/programmy_na_android/videopleery/62">Видеоплееры</a></li>
<li><a class="a"
href="/apps/programmy_na_android/vidzhety/67">Виджеты</a></li>
<li><a class="a"
href="/apps/programmy_na_android/galereja/63">Галерея</a></li>
<li><a class="a"
href="/apps/programmy_na_android/diktofony/26">Диктофоны</a></li>
<li><a class="a" href="/apps/programmy_na_android/dlja_detej/35">Для
детей</a></li>
<li><a class="a" href="/apps/programmy_na_android/eda_i_napitki/36">Еда и
напитки</a></li>
<li><a class="a" href="/apps/programmy_na_android/zhile_i_dom/37">Жилье и
дом</a></li>
<li><a class="a"
href="/apps/programmy_na_android/zametki/27">Заметки</a></li>
<li><a class="a"
href="/apps/programmy_na_android/zvonilki/57">Звонилки</a></li>
<li><a class="a"
href="/apps/programmy_na_android/zdorove/38">Здоровье</a></li>
<li><a class="a"
href="/apps/programmy_na_android/znakomstva/39">Знакомства</a></li>
<li><a class="a"
href="/apps/programmy_na_android/izuchenie_jazykov/40">Изучение
языков</a></li>
<li><a class="a"
href="/apps/programmy_na_android/iskusstvo_i_dizajn/41">Искусство и
дизайн</a></li>
<li><a class="a"
href="/apps/programmy_na_android/kalendari/28">Календари</a></li>
<li><a class="a"
href="/apps/programmy_na_android/kalkuljatory_i_konvertery/29">Калькуляторы
и конвертеры</a></li>
<li><a class="a"
href="/apps/programmy_na_android/kamera/64">Камера</a></li>
<li><a class="a"
href="/apps/programmy_na_android/karty_i_navigatory/42">Карты и
навигаторы</a></li>
<li><a class="a"
href="/apps/programmy_na_android/klaviatury/82">Клавиатуры</a></li>
<li><a class="a"
href="/apps/programmy_na_android/knigi_i_spravochniki/43">Книги и
справочники</a></li>
<li><a class="a"
href="/apps/programmy_na_android/kontakty/58">Контакты</a></li>
<li><a class="a"
href="/apps/programmy_na_android/launchery/68">Лаунчеры</a></li>

```



```
<li><a class="a"
href="/apps/programmy_na_android/menedzhery_zagruzok/83">Менеджеры
загрузок</a></li>
<li><a class="a"
href="/apps/programmy_na_android/messendzhery/59">Мессенджеры</a></li>
<li><a class="a"
href="/apps/programmy_na_android/muzyka/65">Музыка</a></li>
<li><a class="a"
href="/apps/programmy_na_android/novostnye_prilozhenija/44">Новости</a></li>
>
<li><a class="a"
href="/apps/programmy_na_android/oblako_i_rezervnoe_kopirovanie/31">Облако
и резервное копирование</a></li>
<li><a class="a" href="/apps/programmy_na_android/oboi/69">Обои</a></li>
<li><a class="a"
href="/apps/programmy_na_android/obrazovanie/45">Образование</a></li>
<li><a class="a"
href="/apps/programmy_na_android/optimizacija/84">Оптимизация</a></li>
<li><a class="a" href="/apps/programmy_na_android/ofis/30">Офис</a></li>
<li><a class="a"
href="/apps/programmy_na_android/perevodchiki/46">Переводчики</a></li>
<li><a class="a"
href="/apps/programmy_na_android/pogoda/47">Погода</a></li>
<li><a class="a"
href="/apps/programmy_na_android/pokupki/48">Покупки</a></li>
<li><a class="a" href="/apps/programmy_na_android/pochta/54">Почта</a></li>
<li><a class="a"
href="/apps/programmy_na_android/puteshestvija/49">Путешествия</a></li>
<li><a class="a"
href="/apps/programmy_na_android/razvlechenija/50">Развлечения</a></li>
<li><a class="a"
href="/apps/programmy_na_android/skanery_shtrikh_kodov/73">Сканеры штрих-
кодов</a></li>
<li><a class="a"
href="/apps/programmy_na_android/socialnye_seti/60">Социальные
сети</a></li>
<li><a class="a" href="/apps/programmy_na_android/sport/51">Спорт</a></li>
<li><a class="a" href="/apps/programmy_na_android/temy/70">Темы</a></li>
<li><a class="a"
href="/apps/programmy_na_android/utility/85">Утилиты</a></li>
<li><a class="a"
href="/apps/programmy_na_android/fajlovye_menedzhery/86">Файловые
менеджеры</a></li>
<li><a class="a"
href="/apps/programmy_na_android/finansy/32">Финансы</a></li>
<li><a class="a"
href="/apps/programmy_na_android/fotoredaktory/66">Фоторедакторы</a></li>
<li><a class="a" href="/apps/programmy_na_android/khobbi/52">Хобби</a></li>
<li><a class="a"
href="/apps/programmy_na_android/chasy_i_budilniki/33">Часы
будильники</a></li>
```

```
<li><a class="a"
href="/apps/programmy_na_android/chitalki/53">Читалки</a></li>
<li><a class="a"
href="/apps/programmy_na_android/ehkonomija_batarei/87">Экономия
батареи</a></li>
<li><a class="a"
href="/apps/programmy_na_android/ehkrany_blokirovki/71">Экраны
блокировки</a></li>
<li><a class="a"
href="/apps/programmy_na_android/ehmuljatory/88">Эмуляторы</a></li>
</ul>
</div></div></div></div>
```

ДОДАТОК Б ПРЕЗЕНТАЦІЯ

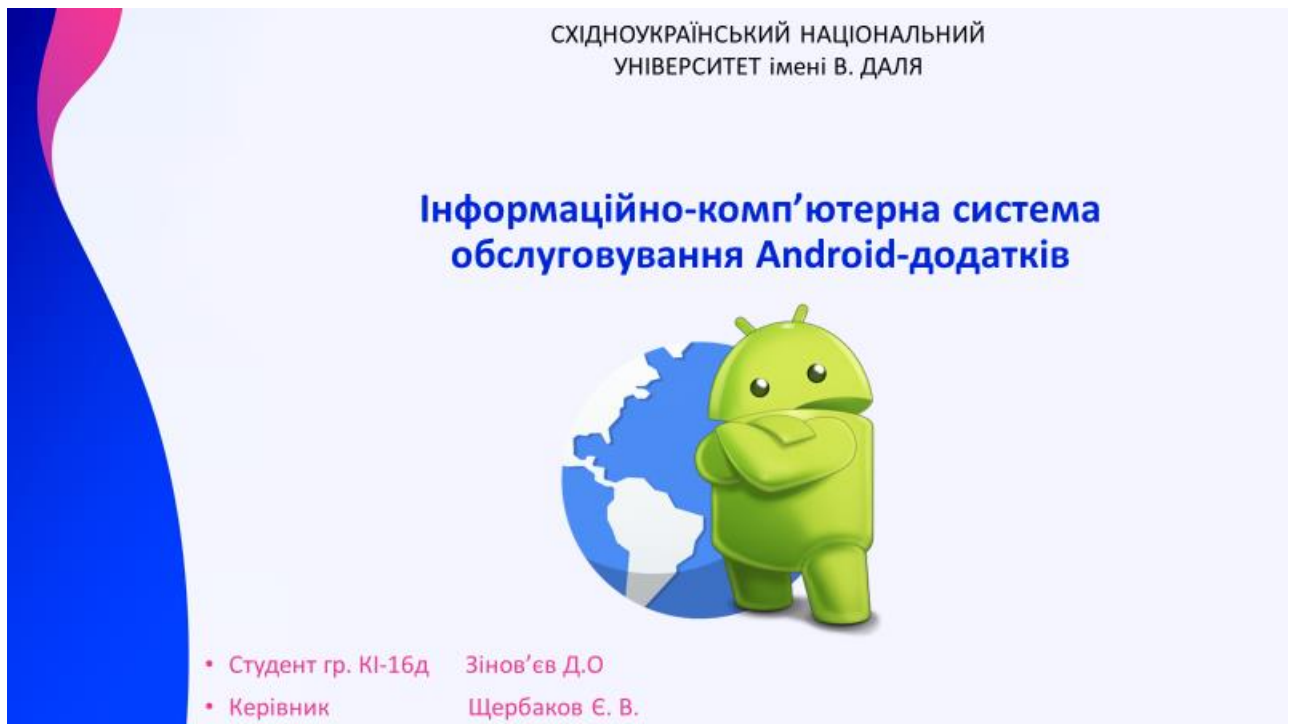


Рисунок Б.1 – Слайд №1

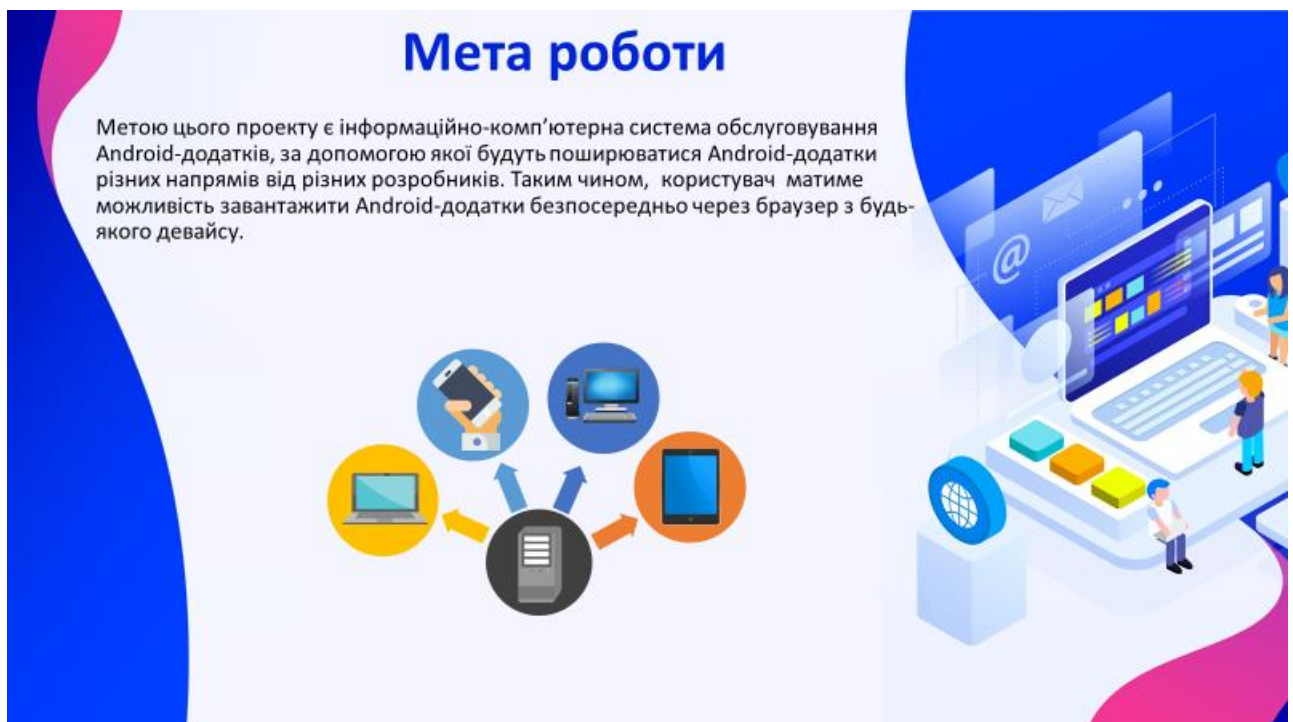


Рисунок Б.2 – Слайд №2

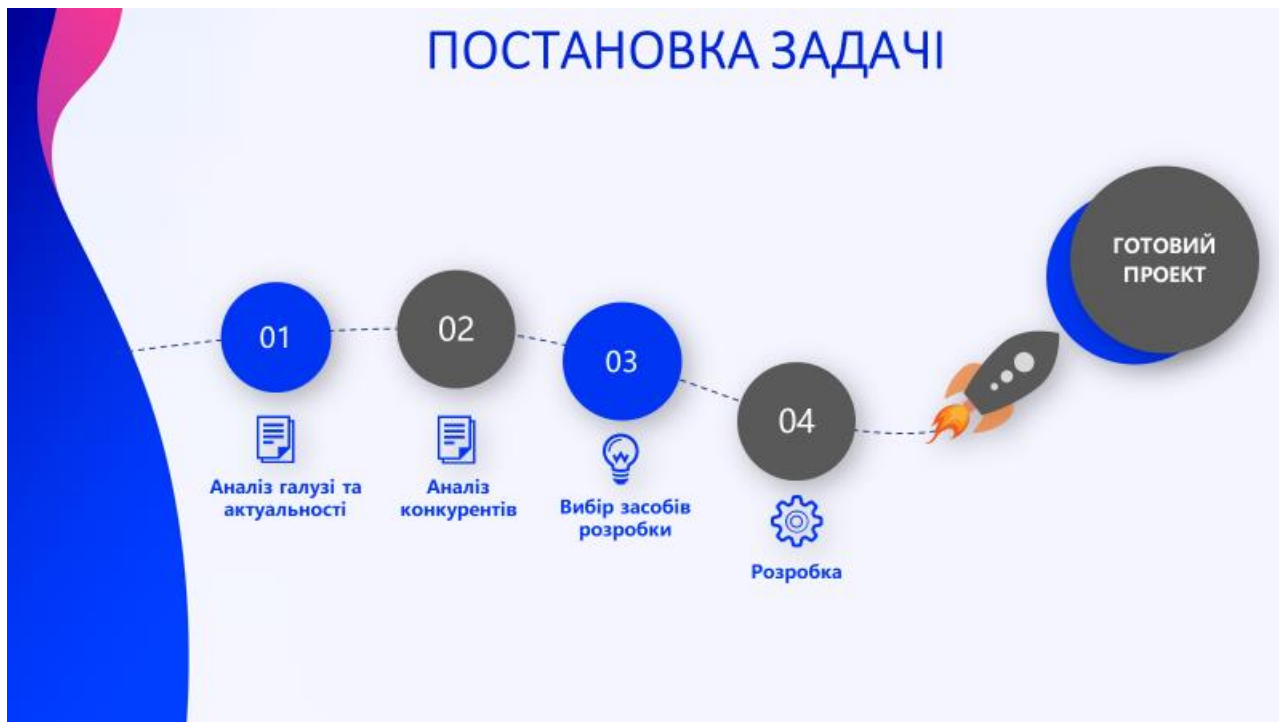


Рисунок Б.3 – Слайд №3



Рисунок Б.4 – Слайд №4

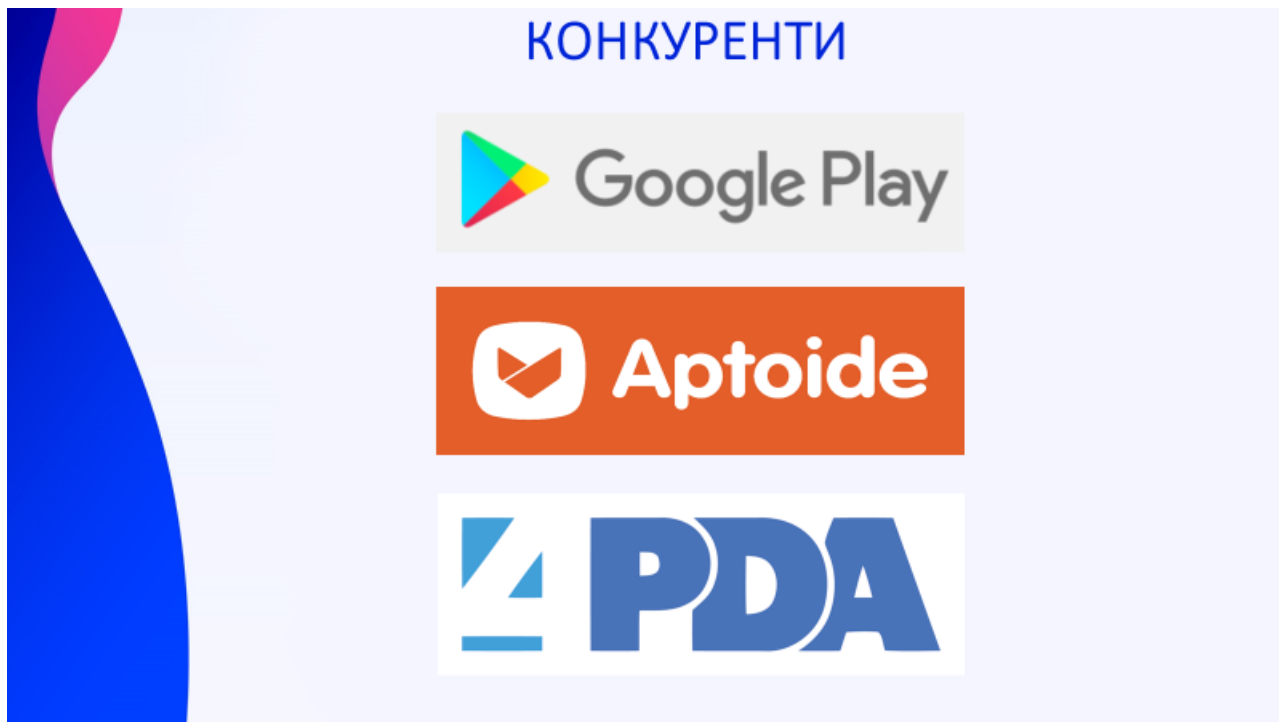


Рисунок Б.5 – Слайд №5



Рисунок Б.6 – Слайд №6



Рисунок Б.7 – Слайд №7



Рисунок Б.8 – Слайд №8

ВИКОРИСТАНІ ТЕХНОЛОГІЇ



MySQL
Вільна система керування реляційними базами даних.

jQuery
JavaScript-бібліотека з відкритим кодом. Бібліотека допомагає легко отримувати доступ до будь-якого елемента DOM, звертатися до атрибутів і вмісту елементів DOM.





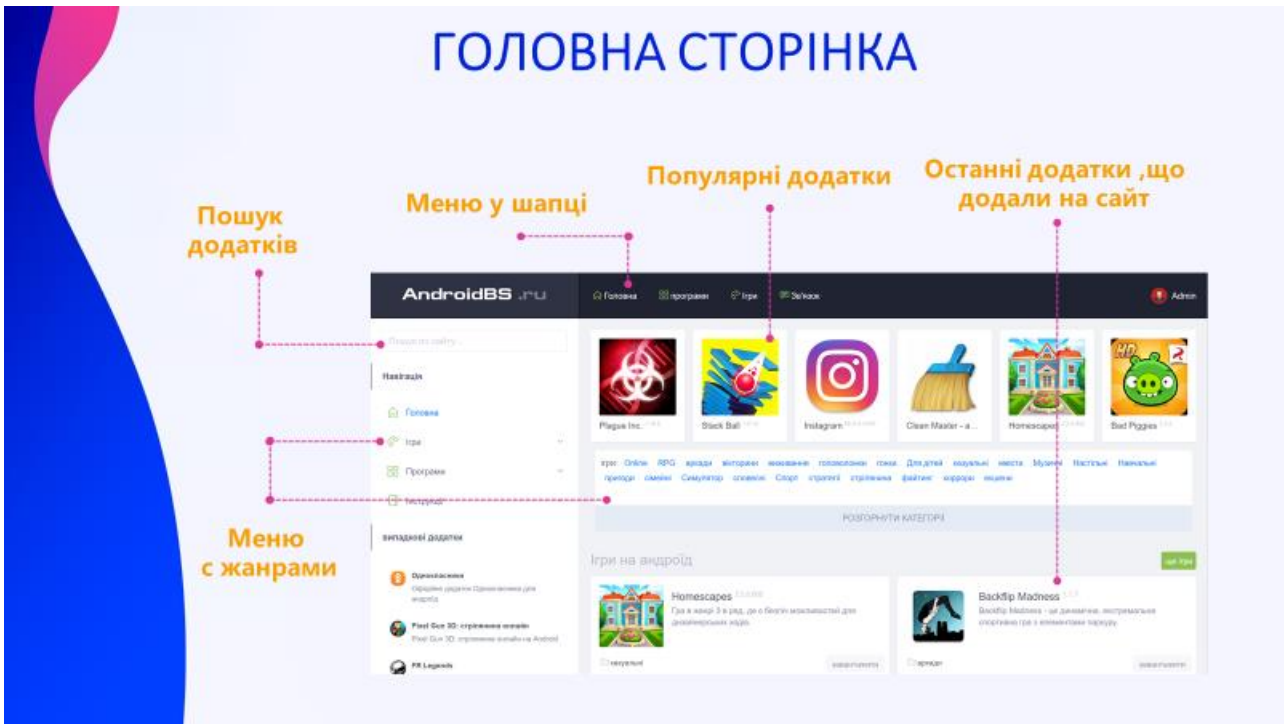
AJAX
Створює інтерфейси веб-застосунків, де веб сторінка, не перезавантажуючись, у фоновому режимі надсилає запити на сервер

Bootstrap
Клієнтський фреймворк та набір інструментів з відкритим кодом, призначений для створення веб-сайтів, які містять шаблони CSS та HTML



Рисунок Б.9 – Слайд №9

ГОЛОВНА СТОРІНКА



Пошук додатків

Меню у шапці

Популярні додатки

Останні додатки, що додали на сайт

Меню с жанрами

Рисунок Б.10 – Слайд №10

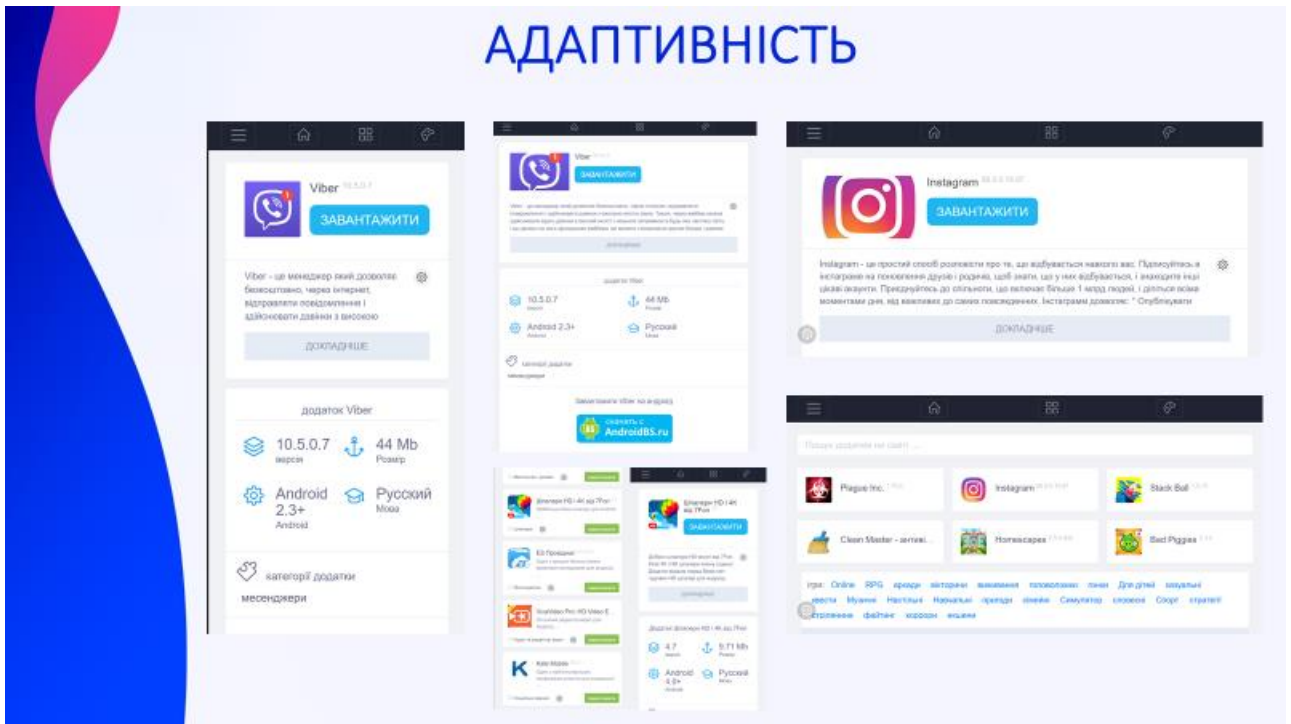


Рисунок Б.13 – Слайд №13

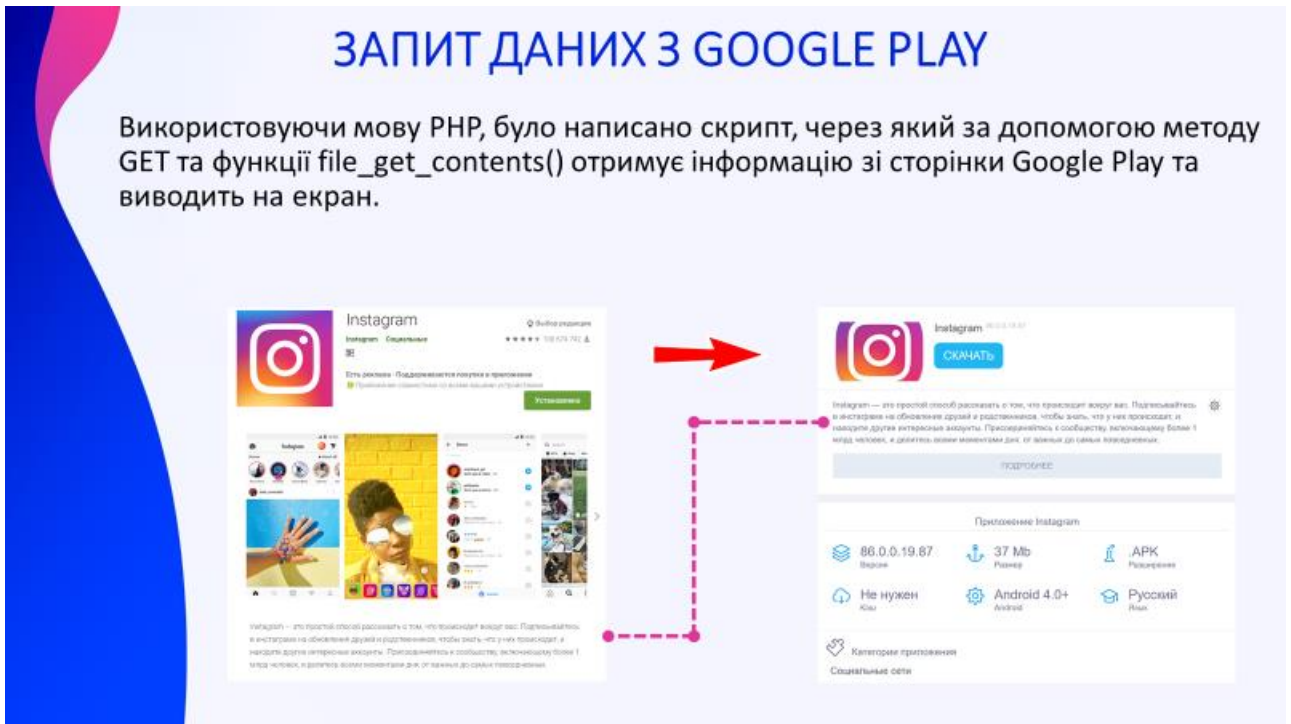


Рисунок Б.14 – Слайд №14

ПЕРСПЕКТИВИ РОЗВИТКУ

- Оптимізація веб-додатка для більш швидкого завантаження сторінок
- SEO оптимізація
- Розширення розміру місця на сервері
- Створення API для подальшої розробки додатків для завантаження з системи
- Наповнення системи різними додатками

Рисунок Б.15 – Слайд №15

ВИСНОВКИ

- Показана актуальність розробки сервісів розповсюдження андроїд-додатків
- Розглянуті сучасні технології розробки
- Проведений аналіз існуючих систем для обслуговування андроїд-додатків
- Розроблена власна система для обслуговування андроїд-додатків

Рисунок Б.16 – Слайд №16