

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
Факультет комп'ютерних наук і технологій
Кафедра комп'ютерних інформаційних технологій

ДОПУСТИТИ ДО ЗАХИСТУ

Завідувач кафедри

Аліна САВЧЕНКО

“ _____ ” _____ 2023 р.

КВАЛІФІКАЦІЙНА РОБОТА

(ДИПЛОМНИЙ ПРОЄКТ, ПОЯСНЮВАЛЬНА ЗАПИСКА)

ВИПУСКНИКА ОСВІТНЬОГО СТУПЕНЯ «БАКАЛАВР»

ЗА СПЕЦІАЛЬНІСТЮ 122 «КОМП'ЮТЕРНІ НАУКИ»

Тема: «WEB–сервіс продажу ігрових акаунтів»

Виконавець: Каленський Андрій Вячеславович

Керівник: к.т.н., доцент Райчев Ігор Едуардович

Нормоконтролер: Олександр ШЕВЧЕНКО

(підпис)

Київ 2023

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет комп'ютерних наук і технологій

Кафедра комп'ютерних інформаційних технологій

Освітній ступінь: «Бакалавр»

Галузь знань, спеціальність, освітньо–професійна програма:

12 «Інформаційні технології», 122 «Комп'ютерні науки», «Інформаційні управляючі системи та технології»

ЗАТВЕРДЖУЮ

Завідувач кафедри

Аліна САВЧЕНКО

“ _____ ” _____ 2023 р.

ЗАВДАННЯ

на виконання кваліфікаційної роботи

Каленського Андрія Вячеславовича

(прізвище, ім'я, по батькові випускника в родовому відмінку)

- 1. Тема кваліфікаційної роботи:** «WEB–сервіс продажу ігрових акаунтів» затверджена наказом ректора №623/ст від 01.05.2023 р.
- 2. Термін виконання роботи:** 15.05.2023 – 25.06.2023р.
- 3. Вихідні дані до роботи:** дані про веб–застосунки, документація, засоби та техніки тестування, база даних MySQL, пакет phpMyAdmin.
- 4. Зміст пояснювальної записки:** аналіз існуючих сайтів з продажу ігрових акаунтів, розробка технології побудови власного WEB–сервісу з продажу ігрових акаунтів, системний аналіз та розробка загальної схеми веб–сайту, пояснення розробки веб–системи, представлення роботи веб–сервісу.
- 5. Перелік обов'язкового графічного (ілюстративного) матеріалу:** слайди презентації Microsoft Power Point.

6. Календарний план–графік

№ пор.	Завдання	Термін виконання	Відмітка про виконання
1	Аналіз та дослідження предметної області. Огляд та аналіз доступних технологій та фреймворків.	15.05.2023– 19.05.2023	Виконано
2	Розробка деталізованого змісту розділів кваліфікаційної роботи. Написання вступу.	22.05.2023– 23.05.2023	Виконано
3	Розробка структури веб–сервісу. Написання 1 розділу кваліфікаційної роботи.	24.05.2023– 26.05.2023	Виконано
4	Створення веб–сервісу з використанням досліджених технологій. Написання 2 розділу кваліфікаційної роботи.	29.05.2023– 02.06.2023	Виконано
5	Написання 3 розділу кваліфікаційної роботи. Усунення недоліків та завершення створення пояснювальної записки кваліфікаційної роботи.	05.06.2023– 09.06.2023	Виконано
6	Оформлення та друк пояснювальної записки кваліфікаційної роботи.	12.06.2023– 13.06.2023	Виконано
7	Підготовка презентації та доповіді для виступу.	14.06.2023– 16.06.2023	Виконано
8	Захист кваліфікаційної роботи.	19.06.2023– 23.06.2023	Виконано

7. Дата видачі завдання: 15.05.2023 р.

Керівник кваліфікаційної роботи

(підпис керівника)

Ігор РАЙЧЕВ

(П.І.Б.)

Завдання прийняв до виконання

(підпис випускника)

Андрій КАЛЕНСЬКИЙ

(П.І.Б.)

РЕФЕРАТ

Кваліфікаційна робота «WEB–сервіс продажу ігрових акаунтів» містить 62 сторінок, 46 рисунків, 18 використаних джерел.

Об’єкт дослідження: WEB–сервіс продажу ігрових акаунтів.

Предмет дослідження: розробка веб–сайту та його функціоналу.

Мета кваліфікаційної роботи: розробка веб–сервісу з використанням різних технологій для продажу ігрових акаунтів, аналіз предметної області, визначення вимог, огляд доступних технологій.

Методи дослідження: аналіз літературних джерел, дослідження та аналіз уже існуючих сервісів.

Результат проекту: розроблений веб–сервіс, який надає функціонал для сфери продажу ігрових акаунтів.

WEB–SERVIC, CONSTRUCT 2, CSS, PHP, HTML.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ, ТЕРМІНІВ	7
ВСТУП.....	8
РОЗДІЛ 1. ДОСЛІДЖЕННЯ МОЖЛИВОСТЕЙ СТВОРЕННЯ ТА ПІДТРИМКИ ВЕБ–САЙТІВ	9
1.1. Створення та підтримка сайтів	9
1.1.1. Технології створення веб–сайтів	10
1.1.2. Дослідження процесу розробки веб–сайтів.....	12
1.1.3. Основні принципи дизайну веб–сайтів.....	14
1.2. Порівняння існуючих сайтів для продажу ігрових акаунтів.	16
1.2.1. Funpay	16
1.2.2. Anylvl.....	17
1.2.3. G2G	19
1.2.4. PlayerAuctions	20
1.3. Функції для створення WEB–сервісу.....	21
1.4. Висновки до Розділу 1	22
РОЗДІЛ 2. ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ РОЗРОБКИ ВЕБ–СЕРВІСУ	24
2.1. Засоби розробки сайту	24
2.2. Construct 2.....	25
2.3. Мова програмування PHP	28
2.4. Мова стилізації CSS	30
2.5. СУБД phpMyAdmin.....	33
2.6. Висновки до Розділу 2	34
РОЗДІЛ 3. РОЗРОБКА ВЕБ–СЕРВІСУ	35
3.1. Аналіз вимог	35
3.1.1. Функціональні вимоги	35
3.1.2. Вимоги до продуктивності	36
3.1.3. Вимоги до безпеки	37
3.2. Структура веб–сервісу	38
3.3. Розробка та реалізація.....	40

3.3.1. Клієнтська частина.....	40
3.3.2. База даних.....	56
3.4. Висновки до Розділу 3	57
ВИСНОВКИ.....	59
СПИСОК БІБЛІОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ.....	61

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ, ТЕРМІНІВ

AJAX – Asynchronous JavaScript and XML

ID – identity document

HTTP – HyperText Transfer Protocol

URL – Uniform Resource Locator

WWW – World Wide Web

WEB – World Wide Web

ВСТУП

В сучасному світі, де онлайн–торгівля продуктами та послугами стає все більш популярною, створення WEB–сервісу з продажу ігрових акаунтів може бути перспективною ідеєю для підприємницької діяльності.

Геймінг–індустрія є однією з найбільш прибуткових та швидко зростаючих галузей розваг, де гравці щодня придбають нові ігри та аксесуари. Продаж ігрових акаунтів може бути ще одним елементом для вибору, яким користуються гравці, щоб підвищити свій рівень гри та отримати більше задоволення від процесу.

У даному проекті будуть досліджені технічні та функціональні аспекти створення WEB–сервісу з продажу ігрових акаунтів. Будуть проаналізовані різні підходи до розробки та дизайну веб–сайтів для залучення користувачів та забезпечення їх задоволення від використання сервісу.

Крім того, будуть проаналізовані ключові аспекти економічної ефективності та рентабельності WEB–сервісу з продажу ігрових акаунтів. В рамках дослідження будуть розглянуті основні вимоги до забезпечення безпеки та захисту персональних даних клієнтів, а також різні підходи до організації оплати товарів та послуг через веб–сервіс.

Дослідження технічних та функціональних аспектів створення WEB–сервісу з продажу ігрових акаунтів має на меті вивчення оптимальних підходів до розробки та дизайну веб–сайтів, які привернуть користувачів та забезпечать їх задоволення від використання сервісу. Це може включати дослідження ефективного використання графічного дизайну, інтерактивності, зручності навігації та інших факторів, що впливають на користувацький досвід.

РОЗДІЛ 1

ДОСЛІДЖЕННЯ МОЖЛИВОСТЕЙ СТВОРЕННЯ ТА ПІДТРИМКИ ВЕБ-САЙТІВ

1.1. Створення та підтримка сайтів

Щоб створити веб-сайт, необхідно мати певні знання і вміння. Перш за все, потрібно вивчити HTML, CSS і JavaScript – основні мови програмування для веб-розробки. HTML відповідає за структуру веб-сторінки, CSS – за її зовнішній вигляд, а JavaScript – за динамічний зміст та взаємодію з користувачем.

Після вивчення основних мов програмування потрібно знайти відповідний хостинг та доменне ім'я для веб-сайту. Хостинг – це місце, де зберігається веб-сайт, тому його потрібно обрати з урахуванням потреб користувачів та можливостей власника сайту. Доменне ім'я – це унікальне ім'я сайту в Інтернеті, що дозволяє користувачам знаходити його через адресу в рядку браузера.

Після реєстрації доменного імені та хостингу можна починати розробку веб-сайту. Для цього можна використовувати різні редактори коду, наприклад, Sublime Text, Atom, Visual Studio Code тощо. Редактори коду дозволяють створювати код веб-сторінок та зберігати їх у потрібному форматі.

При розробці веб-сайту також варто враховувати такі аспекти, як дизайн, взаємодія з користувачем та оптимізація для пошукових систем.

Дизайн веб-сторінок включає в себе вибір кольорової гами, шрифтів, графічних елементів тощо. Взаємодія з користувачем може включати такі елементи, як форми зворотного зв'язку, кнопки соціальних мереж, навігаційне меню та інші.

Кафедра КІТ(47)				НАУ 23 10 92 000 ПЗ			
Виконав	Каленський А.В.			ДОСЛІДЖЕННЯ МОЖЛИВОСТЕЙ СТВОРЕННЯ ТА ПІДТРИМКИ ВЕБ- САЙТІВ	Літера	Аркуш	Аркушів
Керівник	Райчев І.Е.				Д	9	15
Консульт.					УС 411Б		122
Норм. контр.	Шевченко О.П.						

Оптимізація для пошукових систем включає в себе використання ключових слів, правильну структуру сайту та інші фактори, які допоможуть веб-сайту займати вищі позиції в результатах пошукових систем.

Одним з головних аспектів розробки веб-сайту є його підтримка. Після того, як веб-сайт створений, потрібно забезпечити його безперебійну роботу та підтримувати його в актуальному стані. Для цього потрібно регулярно оновлювати контент на сайті, підтримувати його безпеку, виконувати резервне копіювання, виправляти помилки та інші технічні проблеми.

Для підтримки веб-сайту можна використовувати різні інструменти, наприклад, системи керування контентом (CMS) – WordPress, Joomla, Drupal, Magento тощо. CMS дозволяють швидко та просто додавати новий контент на сайт, виконувати зміни в дизайні, додавати функціонал та інші операції без необхідності вмінь програмування.

Для підтримки веб-сайту також варто використовувати аналітичні інструменти, наприклад, Google Analytics, які дозволяють відстежувати поведінку користувачів на сайті та збирати статистику відвідувань. Ці дані можуть бути корисними для подальшої оптимізації сайту та покращення його роботи.

У підсумку, створення та підтримка веб-сайту – це складний та тривалий процес, який потребує знань та вмінь у сфері веб-розробки та маркетингу. Від успіху веб-сайту залежить його відвідуваність та популярність серед користувачів. Однак, при належній увазі до всіх аспектів розробки та підтримки, веб-сайт може стати ефективним.

1.1.1. Технології створення веб-сайтів

Веб-сайт є важливим елементом в онлайн-присутності будь-якого бізнесу або організації. За допомогою веб-сайту можна представити свій бренд, продукти та послуги в Інтернеті, а також забезпечити доступ до необхідної інформації користувачам.

Для створення веб-сайту використовуються різні технології, такі як HTML, CSS, JavaScript, PHP, Python тощо.



Рис.1.1. Технології для створення веб-сайту

HTML відповідає за структуру веб-сторінки, CSS – за її зовнішній вигляд, а JavaScript – за динамічний зміст та взаємодію з користувачем. PHP та Python – це мови програмування, які використовуються для розробки складніших веб-додатків, таких як системи управління контентом (CMS) або електронні магазини.

Ознайомлення з технологіями створення веб-сайтів також включає в себе вивчення фреймворків та бібліотек, які використовуються для швидкої та ефективної розробки веб-додатків. Фреймворки, такі як Laravel, Django, Ruby on Rails тощо, дозволяють розробникам швидко створювати складні веб-додатки з використанням готових компонентів та структурних рішень. Бібліотеки, такі як React, Vue, Angular

тощо, дозволяють розробникам ефективно створювати динамічні інтерфейси користувача та інтерактивні веб-додатки.

Ознайомлення з технологіями створення веб-сайтів також включає в себе вивчення баз даних та їх інтеграції з веб-додатками. Базы даних є важливою складовою будь-якого веб-додатку, оскільки вони забезпечують збереження, оновлення та видалення даних. Для роботи з базами даних використовуються мови запитів, такі як SQL (Structured Query Language), яка є стандартом для взаємодії з реляційними базами даних. Однак, з розвитком технологій з'явилися і альтернативні бази даних, такі як NoSQL, які дозволяють зберігати дані в іншому форматі, залежно від потреб веб-додатку.

При розробці веб-сайтів також важливо враховувати питання безпеки. Веб-додатки можуть бути підвергнуті різним атакам, таким як кросс-сайтовий скриптинг, SQL-ін'єкції, DDoS-атаки тощо. Тому, при створенні веб-сайту важливо враховувати заходи безпеки, такі як валідація даних, використання захисту від CSRF (Cross-Site Request Forgery), використання SSL-шифрування тощо.

Навчання технологіям створення веб-сайтів можна здійснити різними способами, такими як самостійне навчання з використанням онлайн-курсів та ресурсів, навчання у вищих навчальних закладах або навчання в рамках професійних курсів. Важливо враховувати, що технології створення веб-сайтів постійно змінюються та розвиваються, тому важливо бути в курсі останніх трендів та змін.

1.1.2. Дослідження процесу розробки веб-сайтів

Дослідження може включати в себе аналіз різних етапів процесу розробки, від визначення вимог до тестування та розгортання веб-сайту. Основна мета дослідження полягає в тому, щоб знайти ефективні способи покращення процесу розробки веб-сайтів, що призведе до скорочення часу розробки, зменшення витрат та покращення якості продукту.

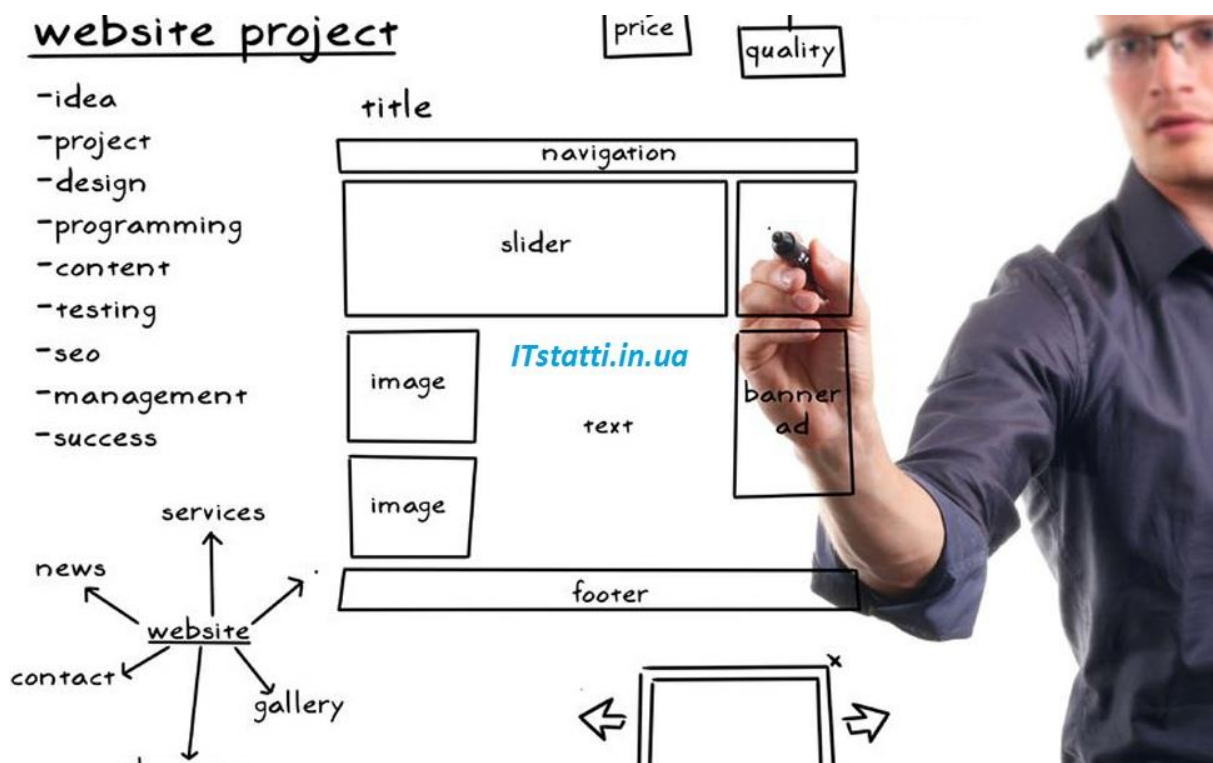


Рис. 1.2. Етапи створення сайту

Дослідження може бути проведене різними способами, включаючи опитування веб-розробників, аналіз веб-сайтів, що були розроблені різними командами, або спостереження за процесом розробки в реальному часі. Незалежно від методу, дослідження повинне включати в себе оцінку кожного етапу процесу розробки веб-сайту.

Перший етап дослідження повинен включати аналіз вимог, зокрема визначення цілей проекту та потреб користувачів. Основна мета – зрозуміти, що потрібно включити до веб-сайту, як його будуть використовувати користувачі та які функціональні вимоги повинні бути задоволені.

Другий етап – проектування веб-сайту. На цьому етапі розробляється концептуальна модель веб-сайту, включаючи структуру, розташування вмісту та інтерфейс користувача. Дослідження може включати аналіз різних методів проектування веб-сайту та визначення ефективних підходів.

Третій етап – розробка та програмування. На цьому етапі розробляються функціональні компоненти веб-сайту, які повинні бути задоволені вимогами проекту.

Дослідження може включати аналіз різних методів програмування та визначення найбільш ефективних інструментів для розробки веб-сайтів.

Четвертий етап – тестування та випробування. На цьому етапі перевіряється якість веб-сайту та його функціональність. Дослідження може включати аналіз різних методів тестування та визначення найбільш ефективних підходів.

Останній етап – розгортання та підтримка веб-сайту. На цьому етапі веб-сайт публікується та налагоджується. Дослідження може включати аналіз різних методів розгортання веб-сайту та визначення найбільш ефективних підходів до підтримки веб-сайту після розгортання.

Після проведення дослідження можна отримати важливі висновки, що допоможуть удосконалити процес розробки веб-сайтів. Наприклад, можна визначити найбільш часті помилки під час розробки та способи їх виправлення, ефективні методи програмування та тестування, а також знайти можливості для покращення комунікації та співпраці між різними членами команди розробників. Дослідження процесу розробки веб-сайтів є важливим етапом у забезпеченні якості та ефективності розробки веб-сайтів, що може мати значний вплив на успіх проекту.

1.1.3. Основні принципи дизайну веб-сайтів

Основні принципи дизайну веб-сайтів включають в себе різні елементи, які взаємодіють між собою, щоб створити естетичний та функціональний дизайн. Ось декілька з основних принципів дизайну веб-сайтів:

- 1. Спрощення та мінімалізм:** дизайн веб-сайту повинен бути простим та легким для сприйняття користувачами. Найкраще використовувати мінімальну кількість елементів та функцій, щоб зробити сайт більш читабельним та легким у використанні.
- 2. Підбір кольорів:** вибір кольорової палітри повинен бути відповідним для вашого бренду, а також забезпечувати зручність користування сайтом. Кольори повинні забезпечувати достатню контрастність між текстом та фоном, щоб зробити сайт більш зручним для читання.

3. **Інтерактивність:** інтерактивні елементи дизайну, такі як анімація, відео та інші, можуть додати веб-сайту інтерактивність та естетику. Вони можуть бути використані для покращення користувацького досвіду та привернення уваги користувачів.
4. **Доступність:** дизайн веб-сайту повинен бути доступним для всіх користувачів, включаючи людей з обмеженими можливостями та тих, хто використовує асистивні технології. Наприклад, текст повинен бути легкочитаним, а кнопки та інші елементи повинні мати достатньої розміру для того, щоб їх можна було легко натискати.
5. **Відповідність:** дизайн веб-сайту повинен бути відповідним для вашої цільової аудиторії та виду бізнесу, який ви пропонуєте. Наприклад, веб-сайт, призначений для молоді, може мати більш яскравий та веселий дизайн, тоді як веб-сайт, призначений для професіоналів, може мати більш серйозний та професійний вигляд.
6. **Чіткість та логічність:** дизайн веб-сайту повинен бути логічним та легко зрозумілим для користувачів. Навігаційні елементи повинні бути розташовані в логічному порядку, щоб допомогти користувачам швидко знайти те, що вони шукають. Також важливо використовувати чіткий та зрозумілий текст, щоб допомогти користувачам зрозуміти, що вони можуть зробити на сайті та як це зробити.
7. **Адаптивність:** дизайн веб-сайту повинен бути адаптивним до різних розмірів екранів та пристроїв, щоб забезпечити користувачам зручний досвід використання сайту на будь-якому пристрої.
8. **Швидкість завантаження:** дизайн веб-сайту повинен бути оптимізований для швидкого завантаження, щоб користувачі не відходили через довгий час очікування. Це можна зробити, наприклад, застосовуючи стиснення зображень та інших файлів, використовуючи кешування та мінімізуючи кількість запитів до сервера.
9. **Консистентність:** дизайн веб-сайту повинен бути консистентним на всій сторінці, з використанням однакового стилю та макету. Це допомагає

користувачам зорієнтуватися на сайті та забезпечує єдиний стиль, який може зміцнити бренд.

10. Відсутність перешкод: дизайн веб-сайту повинен бути створений з урахуванням того, щоб не створювати перешкод для користувачів.

За допомогою цих основних принципів дизайну веб-сайтів можна створити ефективний та привабливий веб-сайт. Спрощення та мінімалізм, підбір кольорів, інтерактивність, доступність, відповідність, чіткість та логічність та адаптивність – це ключові складові, які можуть допомогти забезпечити, що веб-сайт буде привабливим для користувачів, легким у використанні та доступним для всіх. Необхідно також пам'ятати, що дизайн веб-сайту повинен відповідати вашій цільовій аудиторії та типу бізнесу. Крім того, створюючи веб-сайт, слід прагнути до постійного вдосконалення та аналізу результатів, щоб забезпечити його ефективність та корисність для користувачів.

1.2. Порівняння існуючих сайтів для продажу ігрових акаунтів.

1.2.1. Funpay

Funpay – це сервіс, який дозволяє продавати та купувати внутрішню валюту, предмети, акаунти та інші ресурси в онлайн-іграх. Деякі з популярних ігор, підтримуваних сервісом, включають World of Warcraft, Final Fantasy XIV, League of Legends, Fortnite та багато інших.

Переваги FunPay:

- Широкий асортимент товарів: на сайті присутні аккаунти для багатьох популярних онлайн-ігор, а також інші ресурси для геймерів;
- Безпека та надійність: FunPay гарантує безпеку покупцям та продавцям, забезпечує безпечну оплату та відстеження операцій;
- Зручний та простий інтерфейс сайту: клієнти можуть швидко та легко знайти потрібний товар та здійснити покупку;

- Цілодобова підтримка: клієнти можуть звернутися за допомогою в будь-який час.

Недоліки FunPay:

- Залежність від продавців: оскільки FunPay працює з багатьма продавцями, які продають свої товари на сайті, можуть виникати проблеми з якістю товарів та сервісом;
- Висока комісія: FunPay бере високу комісію з продажу товарів, що може зробити кінцеву ціну для покупців вище, ніж на інших сайтах.

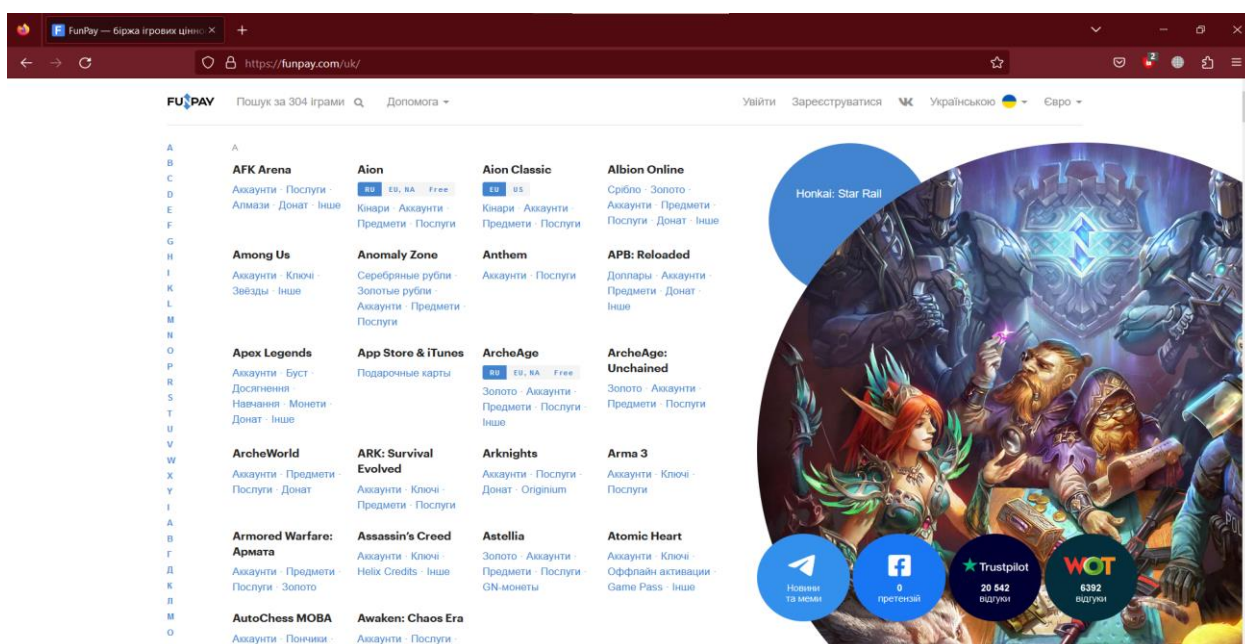


Рис.1.3. Сайт Funpay

1.2.2. Anylvl

AnyLvl – це такий же сайт для продажу ігрових акаунтів, як і FunPay. Основною метою цього сайту є надання можливості продавцям і покупцям зустрітися на одному майданчику і здійснити операції з обміну готівкою на віртуальні речі.

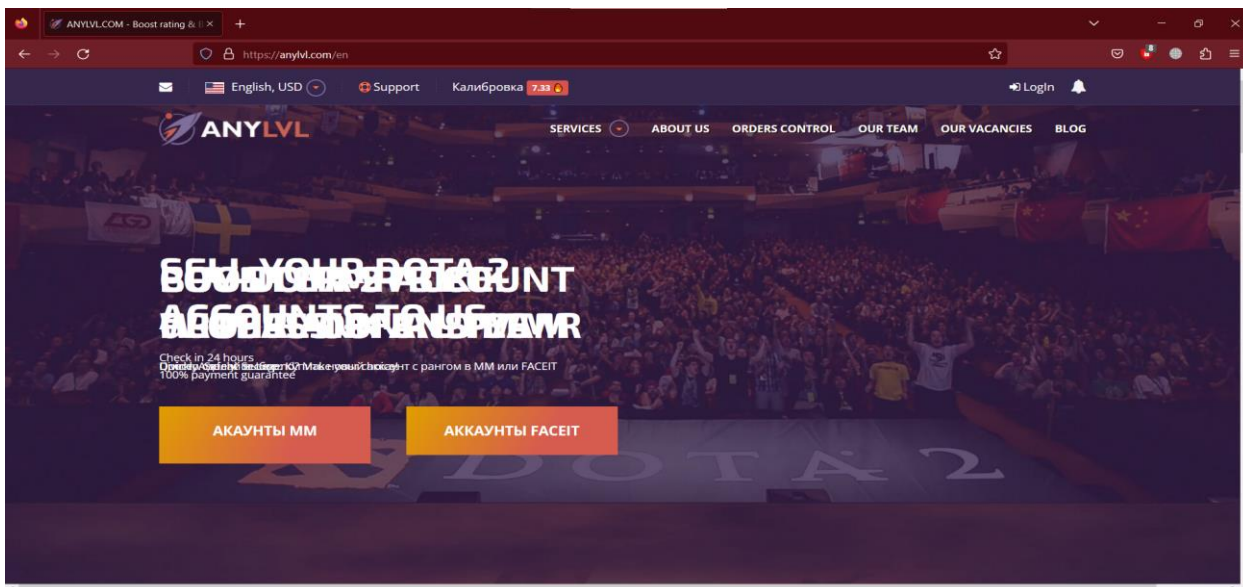


Рис.1.4. Сайт Anylvl

Одразу можна виділити поганий дизайн сайту. Текст налазить один на одний, підтримка мови також мала(не увесь текст має переклад, кнопки також не мають аналогу на інші мови).

Переваги:

- Anylvl має широкий асортимент ігор, в які можна грати з різних пристроїв. Сервіс пропонує продаж акаунтів для таких ігор, як League of Legends, World of Warcraft, Dota 2, Counter–Strike: Global Offensive та інші.
- Можливість купівлі акаунтів з різним рівнем. Крім того, користувачі можуть купити ігрові предмети, кредити та інші речі, що допоможуть поліпшити їхню гру.
- Низькі ціни на акаунти та ігрові ресурси порівняно з іншими сервісами.

Недоліки:

- Сайт не має різноманітних методів оплати, що може змусити деяких користувачів шукати інші сервіси, які підтримують їхні улюблені методи оплати.
- Можливість купівлі крадених акаунтів. Anylvl дозволяє користувачам продавати свої акаунти на сайті, але немає жодного способу перевірити, чи був акаунт проданий власником, чи він був вкрадений.

1.2.3. G2G

G2G (Got2Game) – це онлайн–платформа, яка спеціалізується на купівлі та продажу ігрових аккаунтів, грошей, предметів та послуг для гравців з усього світу. На платформі працюють продавці з більш ніж 150 країн, які продають предмети з понад 1 000 ігор.

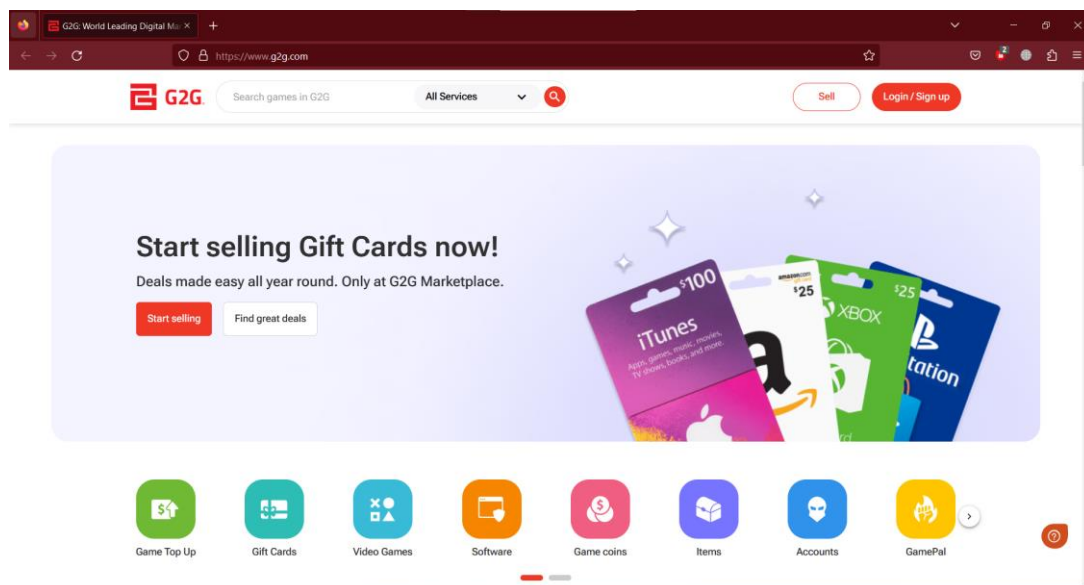


Рис. 1.5. Сайт G2G

Однією з переваг G2G є широкий вибір ігор та предметів, що продаються на платформі, а також широкий спектр послуг для гравців, які можуть бути куплені. Крім того, G2G забезпечує гарантію на покупки, яка захищає покупців від можливих шахрайств або неправомірних дій продавців.

Однак, серед недоліків G2G можна відзначити відсутність чіткого регулювання цін, що може призвести до надмірного збільшення вартості деяких ігор та предметів. Крім того, платформа може бути вразливою до шахрайства та шахрайських дій, особливо коли мова йде про дорогі предмети чи гроші.

1.2.4. PlayerAuctions

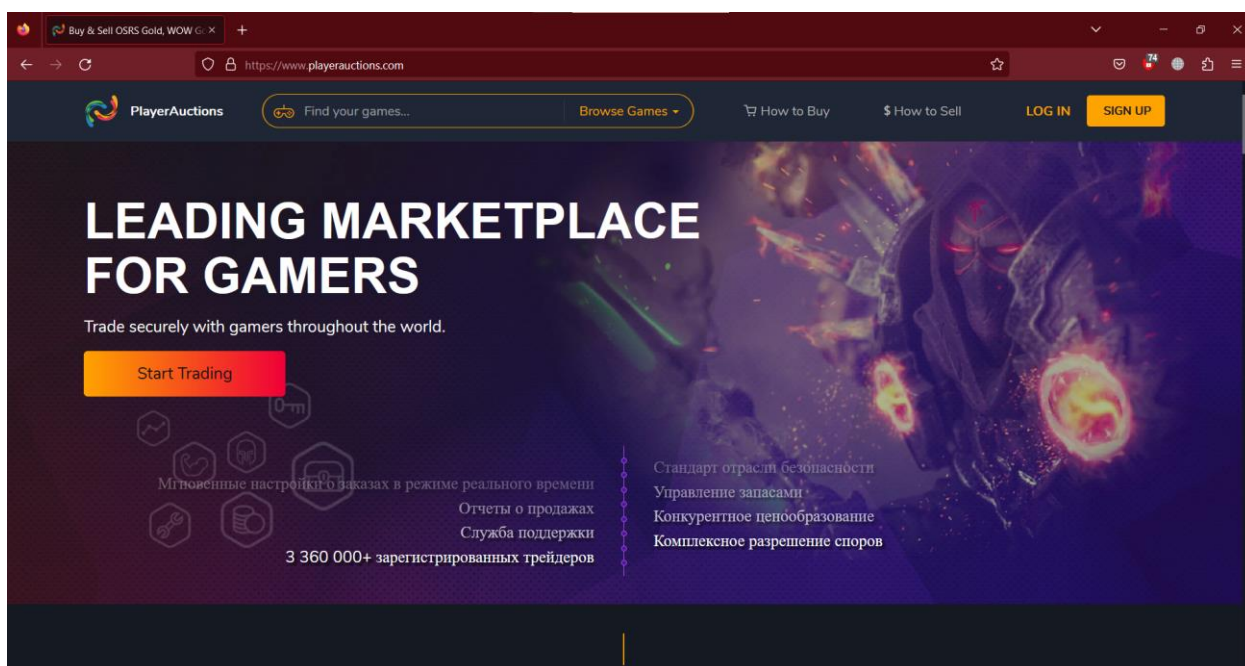


Рис.1.6. PlayerAuctions

PlayerAuctions – це ще один популярний онлайн–ресурс для купівлі та продажу ігрових аккаунтів, валют та інших ігрових ресурсів. Сайт був заснований у 1999 році і має досить довгу історію у галузі онлайн–ігор.

Переваги PlayerAuctions:

- PlayerAuctions має довгу історію і хорошу репутацію в галузі продажу віртуальних товарів.
- Сайт має велику базу покупців та продавців з усього світу.
- На PlayerAuctions є можливість проводити торгівлю в різних валютах.
- Сайт забезпечує захист від шахрайства та шахрайських дій.

Недоліки PlayerAuctions:

- Захист від шахрайства може бути обмеженим, іноді користувачам доводиться звертатися до служби підтримки для вирішення проблем.
- Деякі продавці можуть пропонувати неправомірно отримані товари, що може призвести до блокування аккаунту під час перевірки.

Загалом, PlayerAuctions є одним з провідних сайтів для продажу ігрових аккаунтів та інших ресурсів, і він може бути корисним для тих, хто шукає надійний ресурс для купівлі та продажу віртуальних товарів.

1.3. Функції для створення WEB-сервісу

Для створення WEB-сервісу з продажу ігрових акаунтів необхідно реалізувати декілька функцій. Ось деякі з них:

1. Реєстрація користувачів: Функція реєстрації користувачів дозволить клієнтам створювати акаунти на вашому веб-сайті та зберігати свої дані, такі як історія замовлень, список обраних товарів і т.д. Крім того, реєстрація може забезпечувати додаткові переваги, наприклад, знижки для постійних клієнтів або можливість зберігати свої налаштування профілю.
2. Обліковий запис продавця: Окрім облікових записів користувачів, може знадобитися також обліковий запис продавця, де ви можете керувати своїми продуктами, замовленнями та іншими настройками веб-сайту. Ця функція дозволяє вам вести облік продажів, змінювати ціни та додавати нові товари до магазину.
3. Відображення детальної інформації про товари: Клієнти мають бути забезпечені детальною інформацією про товари, які вони купують. Для цього ви можете додати фотографії, описи, відгуки попередніх покупців, рейтинги тощо. Чим більше інформації про товар, тим вірогідніше, що клієнти будуть задоволені покупкою та повернуться до вашого веб-сервісу знову.
4. Оформлення замовлення та доставка: Вам потрібно мати систему оформлення замовлень та доставки, щоб клієнти могли легко та швидко купувати товари. Функції, такі як розрахунок вартості доставки, вибір методу доставки та отримання інформації про статус замовлення, допоможуть забезпечити зручність для клієнтів та збільшити вірогідність повторних покупок.

5. Система підтримки клієнтів: Важливо мати систему підтримки клієнтів, щоб допомагати їм з вирішенням питань та проблем. Функції, такі як чат або електронна пошта для зв'язку з представниками підтримки клієнтів, допоможуть забезпечити високу якість обслуговування та задоволення клієнтів.

Усі ці функції легко буде виконати за допомогою технологій програми Construct

2.

1. Створення інтерфейсу користувача: Construct 2 має декілька інструментів для створення інтерфейсу користувача, таких як кнопки, тексти та форми. Буде використано ці інструменти для створення елементів інтерфейсу, таких як реєстраційна форма, кошик покупок та інші.
2. Логіка та управління: Construct 2 має систему подій та дій, яка дозволяє вам програмувати логіку та управління веб-сайтом. Їх використання необхідно для того, щоб реагувати на дії користувачів, такі як натискання кнопок або заповнення форм. Також система дій, щоб змінювати стани елементів інтерфейсу, зберігати дані користувачів та виконувати інші дії.
3. База даних: Для збереження даних користувачів та продуктів потрібно працювати з базою даних. Construct 2 має можливість роботи з базами даних, такі як MySQL та SQLite. Буде створено таблиці для зберігання даних та виконання запитів до бази даних для отримання та зміни даних.
4. Мультиплатформеність: Construct 2 чудова програма для створення мультиплатформних веб-сайтів. В ній можливо створити веб-сайт, який буде працювати на комп'ютерах, смартфонах та планшетах.

1.4. Висновки до Розділу 1

В Розділі 1 було розглянуто створення сайтів, розглянуто базові деталі мов HTML, CSS, JavaScript, проведено аналіз існуючих сайтів з продажі ігрових акаунтів, проведено вибір програмного забезпечення та поставлено цілі які необхідно досягнути при створення власного WEB-сервісу.

В результаті було визначено мету кваліфікаційної роботи створення WEB-сервісу з продажу ігрових акаунтів з урахуванням недоліків інших виробників. Процес розробки передбачає розробку дизайну та функціоналу веб-сайту, розробку системи автоматичної оцінки вартості ігрових акаунтів, щоб користувачі могли швидко дізнатися, скільки коштує їхній акаунт, розробку системи підтримки користувачів, щоб забезпечити якісне обслуговування покупців та продавців.

РОЗДІЛ 2

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ РОЗРОБКИ ВЕБ–СЕРВІСУ

2.1. Засоби розробки сайту

Розробка веб–сайтів – це складний і трудомісткий процес, який вимагає від програмістів і веб–розробників високої кваліфікації та глибоких знань у галузі програмного забезпечення. Для створення веб–сайту, який буде привабливим, функціональним та зручним у використанні, важливо володіти потужними інструментами та програмними продуктами.

У світі програмного забезпечення існує безліч інструментів та технологій, які дозволяють веб–розробникам швидко та ефективно створювати високоякісні веб–сайти. Один з найбільш популярних способів створення веб–сайтів є використання спеціальних програмних продуктів, які дозволяють зосередитися на розробці функціоналу та дизайну веб–сайту, а не витратити час на написання коду з нуля.

Програмне забезпечення для створення сайту має величезний потенціал та можливості, які дозволяють веб–розробникам створювати не тільки класичні веб–сайти, а й веб–додатки, інтернет–магазини, соціальні мережі та багато інших цифрових продуктів. Безліч інструментів для створення сайтів дозволяють розробникам вибирати оптимальний інструментарій для своїх потреб та вимог, що дозволяє зменшити час та затрати на розробку веб–сайту.

Крім того, програмне забезпечення для створення сайту дозволяє веб–розробникам швидко та ефективно внести зміни та доповнення у веб–сайт, що дуже важливо в умовах постійно змінюючогося ринку та вимог клієнтів.

Кафедра КІТ(47)				НАУ 23 10 92 000 ПЗ			
Виконав	Каленський А.В.			ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ РОЗРОБКИ ВЕБ–СЕРВІСУ	Літера	Аркуш	Аркушів
Керівник	Райчев І.Е.				Д	24	11
Консульт.					УС 411Б 122		
Норм. контр.	Шевченко О.П.						

Завдяки інструментам для автоматизованої розробки веб-сайтів, розробники можуть легко і швидко додавати новий функціонал, змінювати дизайн та адаптувати веб-сайт до різних пристроїв та браузерів.

Програмне забезпечення для створення сайту також дозволяє веб-розробникам зосередитися на бізнес-логіці та взаємодії з клієнтами, замість того, щоб витратити час на написання великої кількості коду. Це дозволяє розробникам більш ефективно вирішувати проблеми та задачі, що стосуються функціоналу веб-сайту та допомагає забезпечити більш високу якість продукту.

У загальному, програмне забезпечення для створення сайту є важливим інструментом для веб-розробників та підприємств, які хочуть мати високоякісний та функціональний веб-сайт. З його допомогою можна швидко та ефективно розробити веб-сайт будь-якої складності та функціональності, забезпечити його гнучкість та адаптивність до різних умов та зменшити час та затрати на розробку.

2.2. Construct 2

Для створення свого WEB-сервісу було обрано програму Construct 2.



Рис. 2.1. Логотип програми Construct 2

Construct 2 – це потужний графічний редактор, призначений для розробки ігор та інтерактивних веб-додатків без написання коду. Він дозволяє користувачам

створювати групи, рівні, анімацію, графіку та інші складові для своїх ігор за допомогою перетягування та опускання елементів.

Construct 2 використовує технологію HTML5, що дозволяє розробляти ігри, які працюють на будь-якому пристрої, який підтримує цю технологію, включаючи комп'ютери, смартфони та планшети. Редактор дозволяє користувачам додавати спеціальні ефекти, фізику, звуки та інші функції до своїх ігор без написання коду.

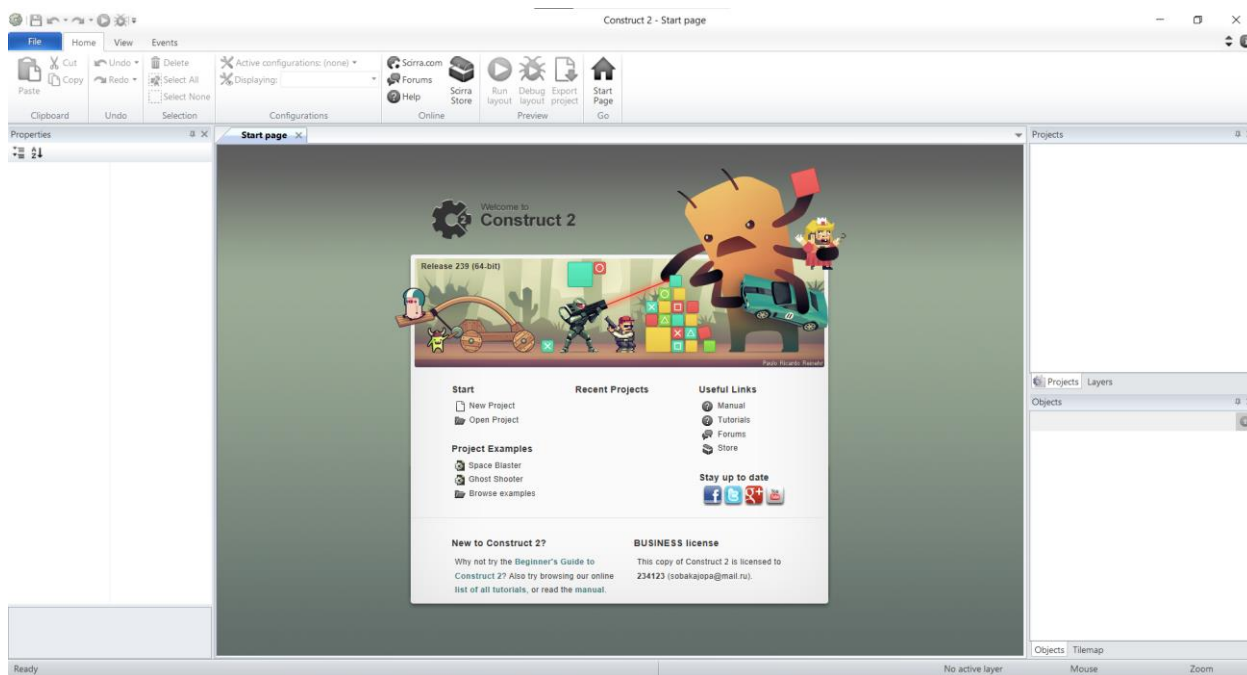


Рис. 2.2. Інтерфейс програми Construct 2

Один з найбільш важливих переваг Construct 2 – це простота використання та інтуїтивно зрозумілий інтерфейс. Розробникам не потрібно мати великих знань в програмуванні, щоб створювати ігри. Редактор включає шаблони для різних жанрів ігор, таких як платформери, шутери та головоломки, що спрощує створення власних ігор.

Construct 2 дозволяє експортувати готові ігри в HTML5-форматі, що дає можливість розмістити їх на веб-сайтах, розсилати друзям або продавати в магазинах додатків. Крім того, Construct 2 підтримує платформи мобільних додатків, такі як Android та iOS, що дозволяє розробникам створювати мобільні ігри для цих платформ.

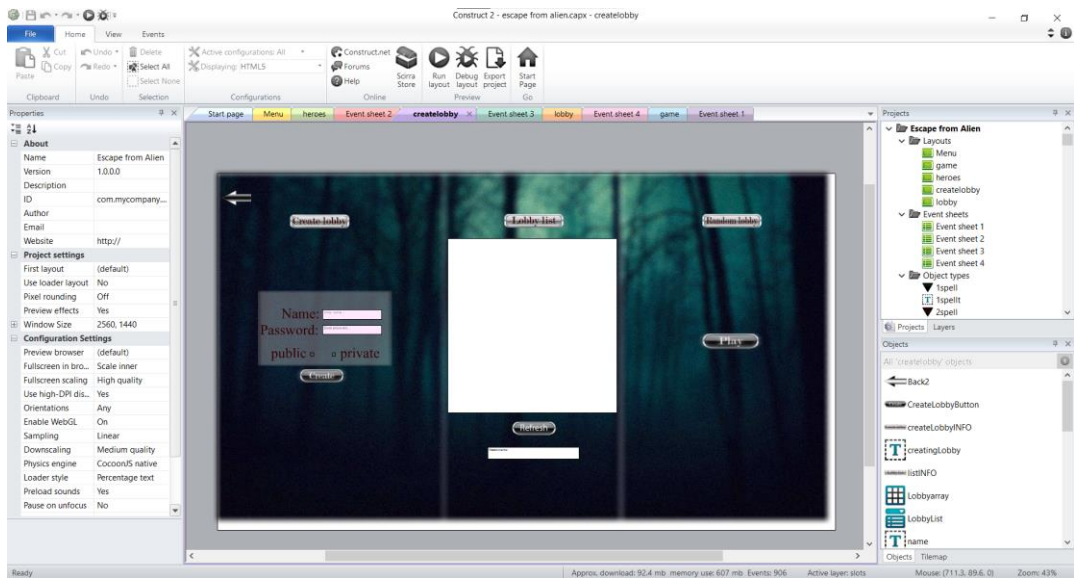


Рис. 2.3. Приклад створення WEB-сторінки в Construct 2

Узагальнюючи, Construct 2 – це потужний редактор, який дозволяє створювати високоякісні ігри та інтерактивні додатки без необхідності програмування. Він має простий інтерфейс та шаблони для різних жанрів ігор, що робить його доступним для користувачів з будь-яким рівнем досвіду. Крім того, він дозволяє експортувати готові ігри в HTML5-форматі для веб-сайтів та мобільних платформ, що зробить їх доступними для гравців з усього світу. Загалом, Construct 2 є потужним інструментом для розробки ігор та інтерактивних додатків, який може допомогти користувачам реалізувати свої творчі ідеї без обмежень.

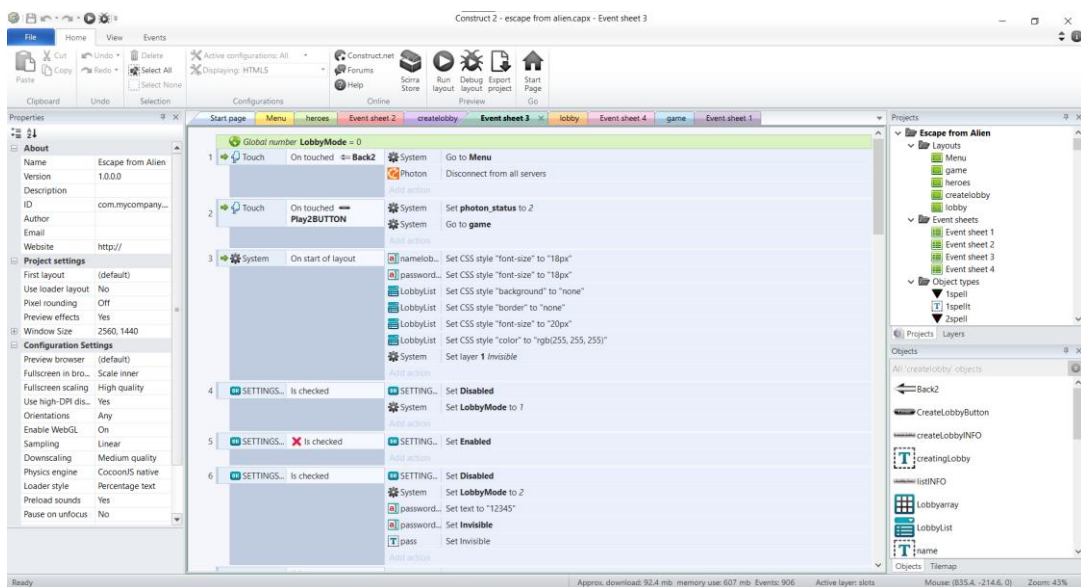


Рис. 2.4. Видяг коду в Construct 2

Є кілька причин, чому варто обрати Construct 2 для створення своїх ігор чи сайтів:

1. Простота використання: Construct 2 має інтуїтивно зрозумілий інтерфейс, який дозволяє створювати ігри без необхідності програмування. Це особливо корисно для тих, хто не має досвіду в програмуванні, але хоче створити власну гру.
2. Висока продуктивність: ігри, створені за допомогою Construct 2, використовують технологію HTML5, що дозволяє їм працювати на будь-якому пристрої, який підтримує цю технологію. Крім того, Construct 2 має вбудовану систему оптимізації, яка дозволяє підтримувати високу продуктивність ігор, незалежно від рівня складності.
3. Широкий функціонал: Construct 2 має великий набір інструментів та функцій для створення ігор, включаючи вбудовану фізику, різноманітні ефекти, звуки та інші складові. Крім того, він підтримує різні жанри ігор, що дозволяє користувачам створювати ігри за своїми вподобаннями.
4. Можливість експортування ігор: ігри, створені за допомогою Construct 2, можна експортувати в HTML5-форматі, що дозволяє розміщувати їх на веб-сайтах або продавати в магазинах додатків. Крім того, Construct 2 підтримує платформи мобільних додатків, що дозволяє розробникам створювати мобільні ігри для різних платформ.

2.3. Мова програмування PHP

PHP – це скриптова мова програмування, що використовується для розробки веб-додатків та взаємодії з базами даних. Вона має синтаксис, схожий на мови C та Perl, і підтримується багатьма веб-серверами та операційними системами.

Основна перевага PHP полягає в його веб-орієнтованих можливостях. Він дозволяє вбудовувати PHP-код безпосередньо в HTML-сторінки, що спрощує розробку динамічного контенту. PHP також підтримує роботу з формами, кукісами,

сеансами, аутентифікацією користувачів та іншими важливими функціями для веб-розробки.



Рис. 2.5. Логотип PHP

Одна з найбільших переваг PHP – велика спільнота розробників та велика кількість доступних бібліотек і фреймворків. Це дозволяє швидко розробляти складні веб-додатки і забезпечує багато готових рішень для різних задач.

PHP є мовою з відкритим кодом, що сприяє її поширенню та постійному оновленню. Завдяки своїй популярності PHP є однією з найпоширеніших мов програмування для веб-розробки.

Загалом, PHP – потужний і гнучкий інструмент для розробки веб-додатків, який дозволяє створювати динамічні та інтерактивні веб-сайти зі зручним синтаксисом та багатим набором функцій.

Переваги PHP:

1. Підтримка баз даних: PHP має вбудовану підтримку для багатьох систем управління базами даних, таких як MySQL, PostgreSQL, SQLite і багатьох інших. Це дає можливість взаємодіяти з базами даних, зберігати, отримувати і модифікувати дані з веб-додатків.
2. Розширюваність: PHP підтримує можливість розширення за допомогою зовнішніх модулів і бібліотек. Розробники можуть створювати свої власні модулі або використовувати сторонні бібліотеки для розширення функціональності PHP і виконання специфічних завдань.

3. Велика спільнота розробників: PHP має широку спільноту розробників, яка активно співпрацює і спільно вирішує проблеми. Існує безліч ресурсів, форумів, блогів та документації, де можна знайти відповіді на питання та допомогу.
4. Фреймворки: У PHP існує велика кількість фреймворків, таких як Laravel, Symfony, CodeIgniter, Yii і Zend Framework, які допомагають розробникам створювати веб-додатки швидше і ефективніше, надаючи готові рішення для багатьох завдань, включаючи маршрутизацію, роботу з базами даних, автентифікацію та інше.
5. Безпека: PHP має вбудовані функції безпеки, такі як фільтрація введених даних, захист від хакерських атак, шифрування та інші механізми, що допомагають забезпечити безпеку веб-додатків.
6. Переносимість: PHP може працювати на різних платформах, таких як Windows, Linux, macOS, і багатьох інших. Це робить його універсальним і зручним для розробки веб-додатків на різних серверах.

PHP є популярною мовою програмування для веб-розробки з багатьма функціональними можливостями і широкою підтримкою. Вона дозволяє розробникам створювати потужні та динамічні веб-додатки з легкістю і ефективністю.

2.4. Мова стилізації CSS

CSS (Cascading Style Sheets) є мовою стилізації веб-документів і використовується для визначення зовнішнього вигляду веб-сторінок. Ось кілька ключових аспектів CSS:

1. Селектори: CSS використовує селектори для вибору елементів HTML, яким будуть застосовуватись стилі. Селектори можуть використовувати ідентифікатори, класи, теги та інші атрибути для точного визначення елементів.

2. Властивості та значення: CSS надає широкий набір властивостей для налаштування зовнішнього вигляду елементів. Наприклад, властивість "color" визначає колір тексту, "font-size" визначає розмір шрифту, "background-color" визначає колір фону тощо. Кожна властивість має значення, яке вказується після двокрапки.
3. Каскадність та спадковість: CSS працює за принципом каскаду, що означає, що стилі можуть бути визначені на різних рівнях і мають пріоритети. Каскадність дозволяє визначати загальні стилі на рівні документа, а потім уточнювати їх для конкретних елементів.
4. Класи та ідентифікатори: CSS дозволяє використовувати класи та ідентифікатори для зручного групування та стилізації елементів. Класи використовуються для групування елементів зі спільними стилями, тоді як ідентифікатори використовуються для вибору конкретного елемента.
5. Розташування елементів: CSS надає можливості для управління розташуванням та розміщенням елементів на сторінці. За допомогою властивостей, таких як "display", "position", "float" та інші, можна контролювати потік елементів і їх розміщення.
6. Респонсивний дизайн: CSS також використовується для створення респонсивного дизайну, який адаптується до різних розмірів екранів. За допомогою медіа-запитів та розмітки, такої як ґрид (grid) та флексбокс (flexbox), можна створювати адаптивні та гнучкі макети.
7. Анімації та переходи: CSS дозволяє створювати анімації та переходи для елементів, що надає додаткову живість та взаємодію на веб-сторінках. За допомогою ключових кадрів (keyframes) та властивостей, таких як "transition" і "animation", можна задавати рухи, зміни кольорів, затемнення та багато іншого.
8. Боксова модель: CSS використовує боксову модель, що означає, що кожен елемент на веб-сторінці представляється як прямокутник з внутрішнім контентом, відступами, рамками та полів. Властивості, такі як "padding"

- (відступи), "border" (рамки) та "margin" (зовнішні відступи), дозволяють контролювати розміри та відстані між елементами.
9. Інкапсуляція: CSS працює на принципі інкапсуляції стилів, що дозволяє розділити структуру веб-сторінки (HTML) від її зовнішнього вигляду (CSS). Це дозволяє розробникам легко змінювати стиль веб-сторінки без впливу на її структуру або зміни HTML-коду.
 10. Селектори та спадковість: CSS має багатий набір селекторів, які дозволяють точно вибрати елементи для стилізації. Крім того, CSS використовує спадковість, що означає, що стилі, визначені на батьківському елементі, можуть успадковуватися й примінятися на дочірні елементи.
 11. Префікси та кросс-браузерна сумісність: Деякі властивості CSS можуть вимагати використання вендорних префіксів для забезпечення сумісності з різними браузерами. Префікси, такі як `-webkit-` (для Safari та Chrome), `-moz-` (для Firefox) та `-ms-` (для Internet Explorer), допомагають розробникам забезпечити правильне відображення стилів у різних браузерах.
 12. Медіа-запити: CSS також підтримує медіа-запити, які дозволяють адаптувати стилі до різних пристроїв та екранів. Медіа-запити дозволяють визначати розміри екрана, орієнтацію пристрою, піксельну щільність та інші параметри для зміни стилів на різних пристроях.



Рис. 2.6. Логотип PHP

CSS є потужним інструментом для стилізації веб-сторінок, надаючи безліч можливостей для красивого та функціонального вигляду. Вивчення CSS дозволить

вам створювати привабливі та професійні веб–дизайни, підлаштовані під різні потреби та пристрої.

2.5. СУБД phpMyAdmin

PhpMyAdmin є веб–інтерфейсом для управління базами даних MySQL. Він надає зручний спосіб взаємодії з базою даних через веб–браузер, не потребуючи ручного написання SQL–запитів.

PhpMyAdmin має широкий спектр функцій, які полегшують адміністрування баз даних. Цей інструмент дозволяє користувачам створювати, видаляти та редагувати бази даних, таблиці, стовпці, індекси, а також виконувати запити на вибірку, вставку, оновлення і видалення даних.



Рис. 2.7. Логотип phpMyAdmin

Однією з особливостей phpMyAdmin є можливість імпорту та експорту даних. Користувачі можуть завантажувати дані зовнішніх файлів у базу даних або експортувати дані з бази даних у різних форматах, таких як SQL, CSV або XML. Це дозволяє зручно переміщувати дані між різними системами або зберігати їх для резервного копіювання.

Крім того, phpMyAdmin має можливості для керування користувачами та їхніми привілегіями. Адміністратори можуть створювати нових користувачів,

надавати їм доступ до конкретних баз даних, таблиць або запитів, а також налаштовувати їхні права доступу.

phpMyAdmin також надає інструменти для оптимізації та аналізу баз даних. Користувачі можуть переглядати статистику використання простору, виконувати аналіз покриття запитів, перевіряти та оптимізувати таблиці для поліпшення продуктивності.

Будучи відкритим програмним забезпеченням, phpMyAdmin доступний безкоштовно і має активну спільноту розробників, що забезпечує постійну підтримку та оновлення. Його легкість встановлення і налаштування робить його популярним вибором для веб-розробників та адміністраторів баз даних, які шукають зручний спосіб управління своїми даними.

2.6. Висновки до Розділу 2

У сучасному світі розробка веб-сервісів є надзвичайно важливою галуззю, яка стверджується високим попитом та зростаючою конкуренцією. Програмне забезпечення розробки веб-сервісу відіграє ключову роль у створенні та підтримці цих сервісів, забезпечуючи розробникам потрібні інструменти для ефективної роботи.

Цей розділ надав загальний огляд програмного забезпечення, яке буде використано в розробці веб-сервісу. Було розглянуто різноманітні інструменти та технології, які допоможуть створити високоякісний веб-сервіс такі як: Construct 2, PHP, CSS і phpMyAdmin.

Ці інструменти разом забезпечать потужний набір інструментів для створення, розгортання та управління веб-сервісом. Кожен з них має свої особливості і призначення, що дозволяє ефективно працювати над веб-проектами різного рівня складності і функціональності.

РОЗДІЛ 3

РОЗРОБКА ВЕБ–СЕРВІСУ

3.1. Аналіз вимог

Аналіз вимог присвячений визначенню основних вимог для нашого веб–сервісу, який спеціалізується на продажу ігрових акаунтів. Цей етап є ключовим у процесі розробки, оскільки він визначає функціональні, продуктивнісні та безпекові характеристики сервісу.

Важливо збирати та розуміти вимоги користувачів, які прагнуть купувати або продавати ігрові акаунти. Також потрібно зосередитися на вимогах щодо продуктивності, щоб забезпечити швидку та ефективну роботу сервісу.

Розробка веб–сервісу продажу ігрових акаунтів має свої особливості та унікальні вимоги. Мета полягає в створенні зручного та безпечного середовища для покупців та продавців.

У цьому аналізі детально розглянуть функціональні, продуктивнісні та безпекові вимоги, що дозволить зрозуміти потреби користувачів та забезпечити найкращий досвід використання сервісу. Крім того, будуть розглянуті аспекти, пов'язані з аутентифікацією, авторизацією, безпекою платежів та зберіганням особистих даних.

3.1.1. Функціональні вимоги

До функціональних вимог свого веб сервісу віднесено такі:

Кафедра КІТ(47)				НАУ 23 10 92 000 ПЗ			
Виконав	Каленський А.В.			РОЗРОБКА ВЕБ–СЕРВІСУ	Літера	Аркуш	Аркушів
Керівник	Райчев І.Е.				Д	35	24
Консульт.					УС 411Б 122		
Норм. контр.	Шевченко О.П.						

- Реєстрація користувачів: можливість створення облікових записів для покупців та продавців.
- Пошук ігрових акаунтів: можливість шукати акаунти за конкретними параметрами, наприклад, гра, рейтинг, рівень тощо.
- Перегляд деталей акаунтів: відображення інформації про акаунт, такої як гра, рейтинг, рівень, предмети, досягнення тощо.
- Взаємодія між користувачами: можливість спілкування покупців і продавців, обмін повідомленнями або коментарями.

3.1.2. Вимоги до продуктивності

Вимоги до продуктивності відіграють важливу роль у розробці веб-сервісу і мають на меті забезпечити його ефективну та швидку роботу. Ось кілька причин, чому вимоги до продуктивності є важливими:

1. Користувацький досвід: Користувачі очікують, що веб-сервіс буде працювати швидко та без затримок. Швидка відповідь і швидке завантаження сторінок забезпечують задоволення користувачів, підвищують їхню впевненість у сервісі і зменшують ймовірність відхилень.
2. Збільшення конверсії: Якщо веб-сервіс повільно працює або має довгий час завантаження, користувачі можуть втратити терпіння та покинути сторінку або перейти до конкурентів. Швидка продуктивність допомагає збільшити конверсію, тобто кількість користувачів, які здійснюють покупки або виконують бажані дії на вашому сервісі.
3. Масштабованість: Вимоги до продуктивності також враховують можливість масштабування сервісу. При зростанні кількості користувачів і обсягу даних потрібно забезпечити, щоб сервіс залишався продуктивним і ефективним. Правильне планування та впровадження продуктивних архітектурних рішень дозволить забезпечити масштабованість та витримку навантаження.
4. Оптимізація ресурсів: Ефективне використання ресурсів, таких як серверне обладнання, мережа та бази даних, дозволяє забезпечити оптимальну

продуктивність при економному використанні ресурсів. Це може допомогти уникнути перевитрат на обладнання та забезпечити оптимальну роботу сервісу.

5. Стійкість та завантаження: Вимоги до продуктивності враховують також навантаження, з яким сервіс повинен впоратися. Вони визначають, як система повинна працювати при пікових навантаженнях та забезпечувати стійку роботу без перебоїв.

3.1.3. Вимоги до безпеки

Вимоги до безпеки є критично важливими для будь-якого веб-сервісу, включаючи сервіс продажу ігрових акаунтів. Ось декілька причин, чому вимоги до безпеки необхідні:

1. Захист конфіденційності: Вимоги до безпеки спрямовані на забезпечення конфіденційності інформації користувачів. У випадку продажу ігрових акаунтів можуть зберігатися особисті дані користувачів, такі як адреса електронної пошти, платіжні дані або інша конфіденційна інформація. Вимоги до безпеки спрямовані на запобігання несанкціонованому доступу до цих даних і забезпечення їхньої конфіденційності.
2. Захист від шахрайства: У сфері продажу ігрових акаунтів можуть виникати ризики пов'язані зі шахрайством, зловживанням та шахрайськими діями. Вимоги до безпеки спрямовані на забезпечення захисту від таких ситуацій, включаючи захист від підробки акаунтів, фішингових атак, шахрайських платежів тощо.
3. Запобігання втраті даних: Вимоги до безпеки також спрямовані на запобігання втраті даних. Збитки від втрати даних можуть бути великими і мають серйозні наслідки для користувачів та вашого бізнесу. Вимоги до безпеки включають резервне копіювання даних, використання захисту від вірусів і зловмисного програмного забезпечення, а також заходи безпеки на рівні мережі та інфраструктури.

4. **Забезпечення довіри користувачів:** Вимоги до безпеки також спрямовані на підтримку довіри користувачів до вашого сервісу. Коли користувачі мають впевненість у безпеці своїх даних і транзакцій, вони більш схильні використовувати ваш сервіс і здійснювати покупки.
5. **Відповідність нормативним вимогам:** Вимоги до безпеки також пов'язані з вимогами регуляторних органів та нормативних актів. Багато секторів, включаючи продаж ігрових акаунтів, підпадають під різні правила та обмеження, що стосуються захисту даних та безпеки. Вимоги до безпеки допомагають вам дотримуватися цих вимог та уникати можливих штрафів та проблем з владними органами.

Врахування вимог до безпеки у розробці веб-сервісу допоможе забезпечити захист користувачів, довіру до сервісу та відповідність правовим та регуляторним вимогам.

3.2. Структура веб-сервісу

У кожного веб-сервісу є своя структура, яка визначає, з чого він складається і як його елементи взаємодіють між собою. Для мого веб-сервісу, присвяченого продажу ігрових акаунтів, визначено наступну структуру, яка забезпечує зручність і ефективність користування:

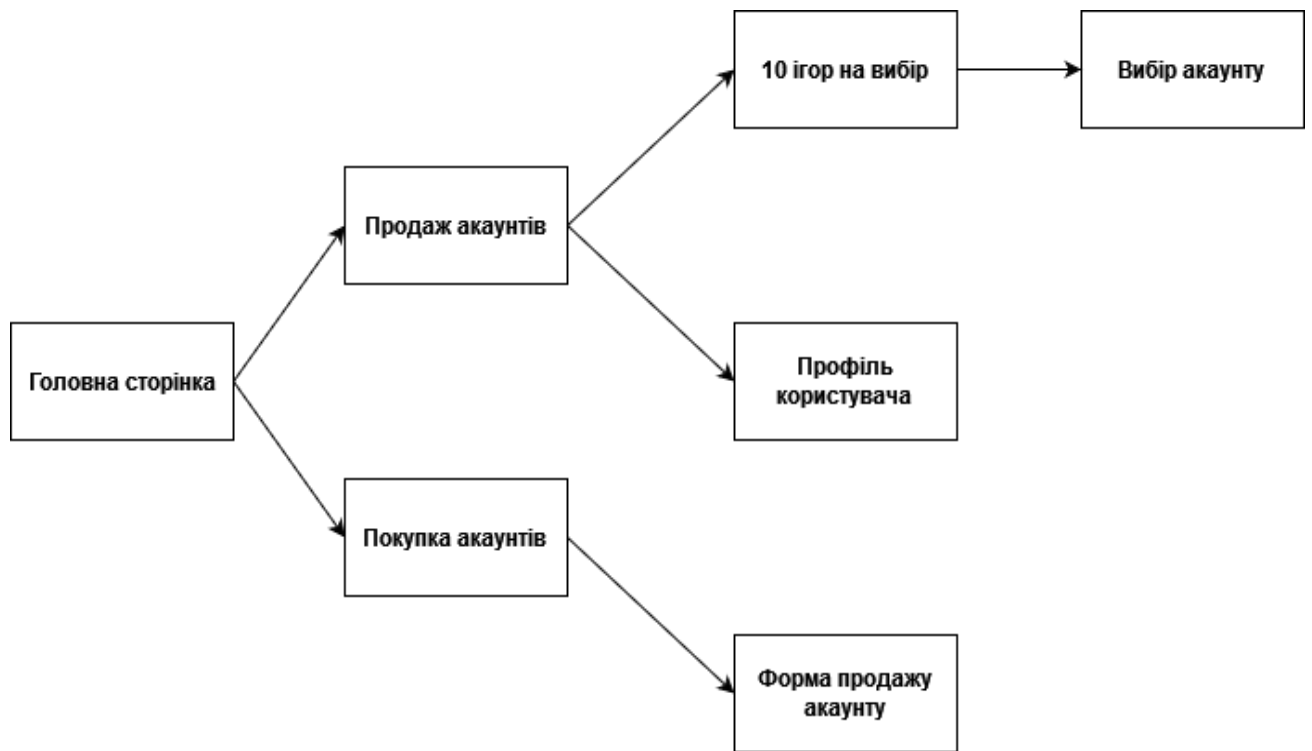


Рис.3.1. Структура веб-сервісу

1. Головна сторінка: Це стартова точка веб-сервісу, де користувачі можуть ознайомитися з загальною інформацією про сервіс та його перевагами.
2. Розділ "Покупка акаунтів": У цьому розділі користувачі можуть переглядати список доступних ігор для придбання акаунту. Ми надаємо докладну інформацію про кожен акаунт, ранг, наявність додаткових ресурсів тощо. Користувачі можуть вибрати акаунт, який їх цікавить, і перейти до процесу покупки.
3. Розділ "Продаж акаунтів": У цьому розділі користувачі мають можливість розмістити свої власні ігрові акаунти для продажу. Вони можуть заповнити форму, вказати необхідну інформацію про акаунт (гра, ранг, ціна тощо) та розмістити оголошення.
4. Розділ "Профіль користувача": Цей розділ надає можливість користувачам створити свій особистий кабінет на нашому веб-сервісі. Вони можуть редагувати профільну інформацію, відстежувати статус продажів та виконувати інші дії, пов'язані зі своїм акаунтом.

Це загальна структура мого веб–сервісу, яка дозволяє ефективно взаємодіяти з користувачами та забезпечує їх потреби в придбанні та продажу ігрових акаунтів. Детальніше про кожен розділ буде описано в подальшому.

3.3. Розробка та реалізація

У цьому підрозділі ми розглянемо процес створення та реалізації веб–сервісу для продажу ігрових акаунтів. Цей етап включає розробку клієнтської частини одразу з тестуванням, проектування бази даних.

3.3.1. Клієнтська частина

Клієнтська частина веб–сервісу включає інтерфейс користувача, з яким користувачі будуть взаємодіяти для покупки акаунтів, продажу акаунтів та керування своїм профілем. Основна мета цієї частини полягає у наданні зручного та інтуїтивно зрозумілого інтерфейсу для користувачів. Важливі аспекти, які варто врахувати, включають наступне:

1. **Дизайн інтерфейсу:** розробка привабливого та сучасного дизайну інтерфейсу є ключовим елементом клієнтської частини. Використання принципів дизайну, які забезпечують легку навігацію, привабливість та зручність взаємодії з елементами інтерфейсу.
2. **Функціональність:** забезпечення всіх необхідних функцій для покупки акаунтів, продажу акаунтів та керування профілем користувача. Розробка зручних форм та елементів керування для заповнення інформації, вибору товарів та виконання необхідних операцій.
3. **Авторизація та безпека:** реалізація механізмів авторизації та аутентифікації користувачів, щоб забезпечити захист від несанкціонованого доступу до акаунтів. Застосування надійної практики безпеки, такі як шифрування даних та захист від атак, щоб забезпечити конфіденційність та цілісність інформації.

4. Відповідність до різних пристроїв: забезпечення, веб-сервісу, щоб був адаптований для відображення та працездатності на різних пристроях, таких як комп'ютери, планшети та мобільні телефони.
5. Тестування та відладка: проведення інтенсивного тестування клієнтської частини, щоб переконатися в її працездатності, стабільності та відповідності функціональним вимогам.

Для початку представимо що зможе робити відвідувач сайту:

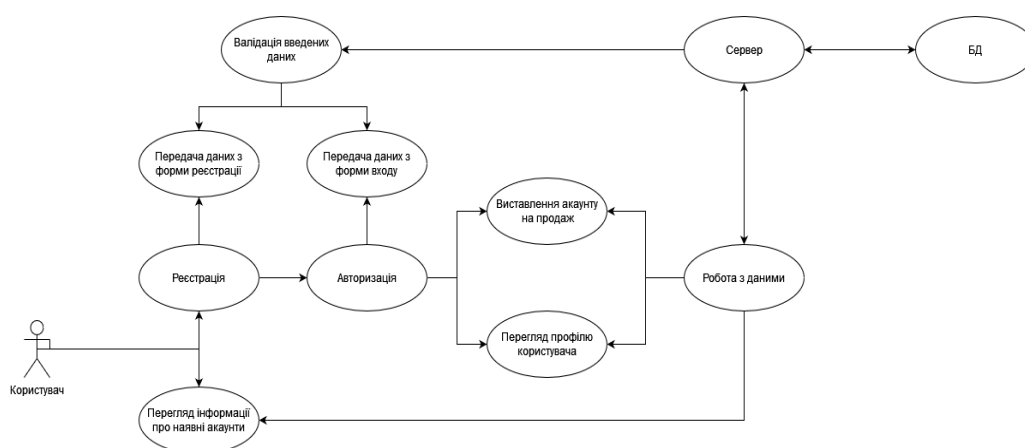


Рис.3.2. Принцип роботи системи для нового користувача

Розпочнемо зі створення головної сторінки. На ній буде відображено назву веб-сервісу, та дві кнопки для переходу на інші сторінки: «Buy account» та «Sell account».

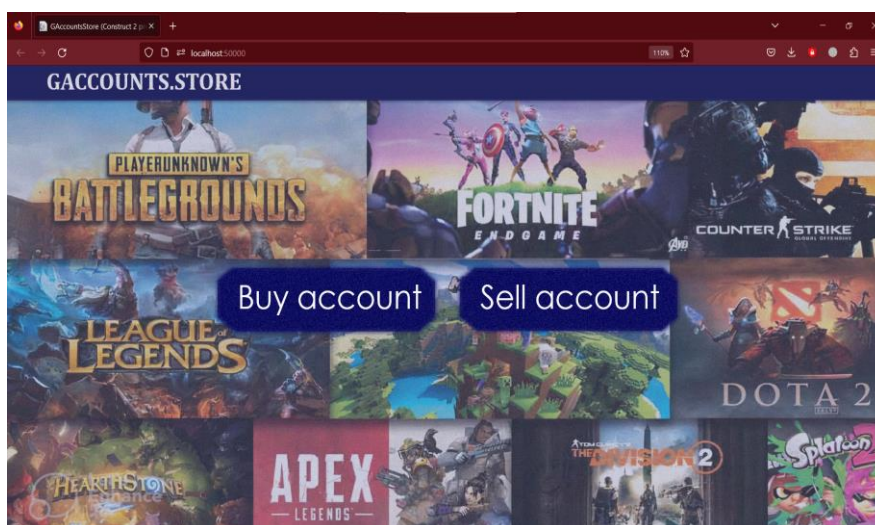


Рис.3.3. Головна сторінка

Щоб відбувся перехід на наступні сторінки сервісу використано функцію «Mouse: On left button Clicked on (назва кнопки)». Ця функція робить перехід на відповідну сторінку після натискання на відповідну кнопку.

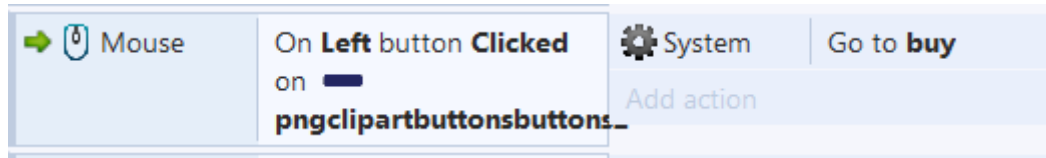


Рис.3.4. Код переходу на сторінку покупки акаунту

Але щоб відбувся перехід на сторінку покупки акаунту система перевіряє чи користувач авторизований. Перевірка авторизації на сайті відбувається за допомогою функції «System: log=1». На початку ця функція дорівнює 0, але при вході в акаунт вона стає рівна 1.

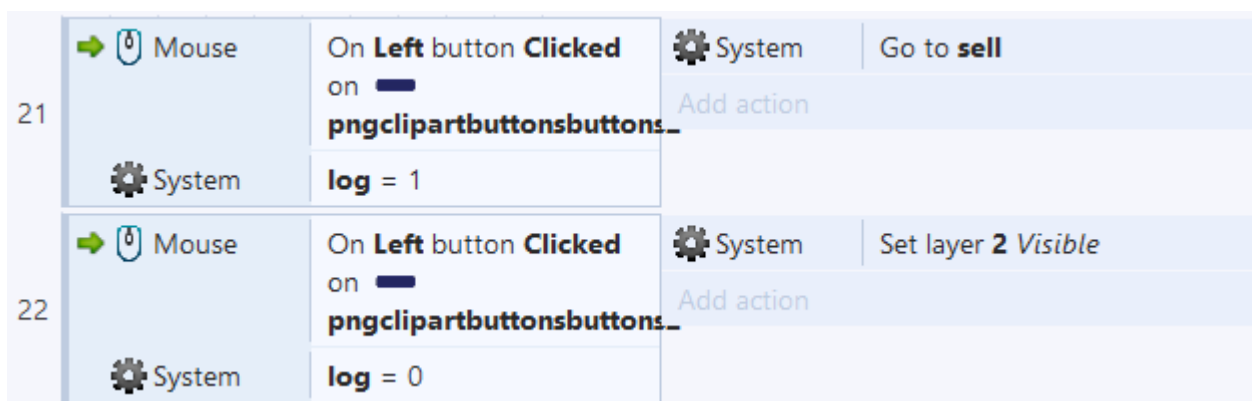


Рис.3.5. Код перевірки на авторизацію користувача

Якщо користувач не авторизований в системі з'являється вікно Входу/реєстрації акаунту. Після чого він проходить процедуру авторизації і перенаправляється до сторінки Продажу акаунту.



Рис.3.6. Вікно входу до акаунту

При вході в акаунт вводить свій Логін та пароль, система перевіряє їх в базі даних та надає доступ для входу, якщо дані вірні, якщо ні з'являється повідомлення з причиною відмовлення у вході до акаунту.

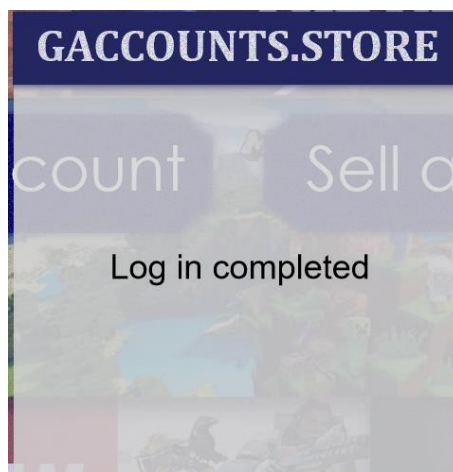
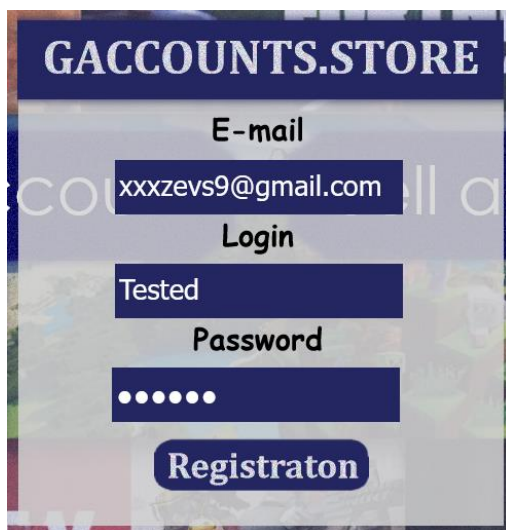


Рис.3.7. Успішний вхід до акаунту

Якщо користувач не має акаунту він може зареєструвати новий унікальний акаунт, якщо акаунту з даним логіном та поштою немає в базі, то він буде успішно зареєстрований



The image shows a registration form for GACCOUNTS.STORE. At the top, the site name 'GACCOUNTS.STORE' is displayed in white text on a dark blue background. Below this, the form consists of several fields and buttons: an 'E-mail' field containing 'xxxzevs9@gmail.com', a 'Login' field containing the text 'Tested', a 'Password' field with five dots for masking, and a blue button labeled 'Registraton' (sic) at the bottom.

Рис.3.8. Вікно реєстрації акаунту

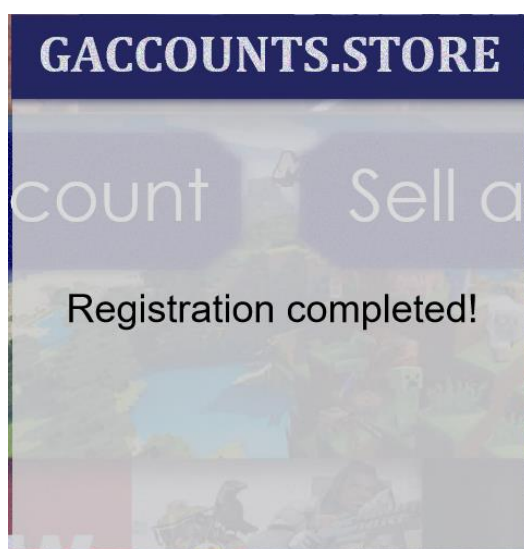


Рис.3.9. Реєстрація успішна

Вхід та авторизація виконуються за допомогою плагіну Ajax. При натисканні клавіши «Log in» або «Regestration» система робить Ajax запит до відповідного файлу написаного на мові PHP в директорії сайту. Цей файл виконує пошук в базі даних та повертає повідомлення про успішне виконання запиту. Після того система перевіряє що повернув запит Ajax та виконує відповідні функції далі.

24	Mouse	On Left button Clicked on reg	System	Set layer 3 Visible
			System	Set layer 2 Invisible
			Add action	
25	Mouse	On Left button Clicked	Add action	
	Mouse	Cursor is over logBox		
28	System	Layer 2 is visible	Add action	
30	Mouse	On Left button Clicked on reg2	AJAX	Send "login=&login_box.Text&&password=&password_box.Text&&email=&email_box.Text to URL "https://gaccountsstore.000webhostapp.com/reg.php" (method "POST", tag "reg")
			Add action	
31	Mouse	On Left button Clicked on login	AJAX	Send "login=&log_box.Text&&password=&password.Text to URL "https://gaccountsstore.000webhostapp.com/auth.php" (method "POST", tag "auth")
			Add action	
32	AJAX	On "reg" completed	System	Set inforeg to AJAX.LastData
			Add action	
33	AJAX	On "auth" completed	System	Set logininfo to AJAX.LastData
			Add action	
34	System	log = 1	sitename	Set text to login
	sitename	id = 5	Add action	

Рис.3.10. Код Construct 2 для перевірки введених даних авторизації чи реєстрації

```
header('Access-Control-Allow-Origin: *');
mysql_host = "localhost"; // Хост бази даних
mysql_database = "id20788486_andriy"; // ІМЯ бази даних
mysql_user = "id20788486_kalensky"; // ІМЯ користувача бази даних
mysql_password = "cfsw4561200272_"; // Пароль бази даних

$link = mysql_connect($mysql_host, $mysql_user, $mysql_password) or die("Ошибка при подключении MySQL" );
mysql_select_db($mysql_database, $link) or die ('Ошибка при подключении к БД');

if (isset($_POST['login'])) { $login = $_POST['login']; if ($login == '') { unset($login); } }
if (isset($_POST['password'])) { $password=$_POST['password']; if ($password == '') { unset($password); } }
if (isset($_POST['email'])) { $email=$_POST['email']; if ($email == '') { unset($email); } }

if (empty($login) or empty($password) or empty($email))
{
    echo ("123");
}
$login = stripslashes($login);
$login = htmlspecialchars($login);
$data = stripslashes($data);
$data = htmlspecialchars($data);
$password = stripslashes($password);
$password = htmlspecialchars($password);
$email = stripslashes($email);
$email = htmlspecialchars($email);
$country = stripslashes($country);
$country = htmlspecialchars($country);
$login = trim($login);
$password = trim($password);
$email = trim($email);
$q1 = mysql_query("SELECT login FROM `accounts` WHERE `login`='".$login."'");
$loginb = mysql_fetch_array($q1);
$loginbd=$loginb['login'];
if($loginbd == $login){
    echo("5");
}
else
$result2 = mysql_query ("INSERT INTO accounts (login,data,password,email,country) VALUES ('$login','$data','$password','$email','$country')");
if ($result2=='TRUE')
{
    echo "6";
}
}
```

Рис.3.11. Код файлу reg.php


```

<?php
//Разрешение браузеру на осуществление кроссдоменных запросов
header('Access-Control-Allow-Origin: *');

//Записываем в переменные данные для доступа к БД
header('Access-Control-Allow-Origin: *');
mysql_host = "localhost"; // Хост базы данных
mysql_database = "id20788486_andriy"; // Имя базы данных
mysql_user = "id20788486_kalensky"; // Имя пользователя базы данных
mysql_password = "cfsw4561200272_"; // Пароль базы данных

//Подключаемся к MySQL и выбираем БД
$link = mysql_connect($mysql_host, $mysql_user, $mysql_password) or die("Ошибка при подключении MySQL" );
mysql_select_db($mysql_database, $link) or die ('Ошибка при подключении к БД');

//Проверяем наличие передаваемых данных
if(isset($_POST['login'])) $login = $_POST['login'];
if(isset($_POST['password'])) $password = $_POST['password'];

//Проверяем наличие полученных значений
if(isset($login) && isset($password)){

//Запрос к БД на получение нужной строки
$q1 = mysql_query("SELECT login FROM `account` WHERE `login`='".$login.'" AND `password`='".$password.'");
$logindb = mysql_fetch_array($q1);
$logindb=$logindb['login'];
$q2 = mysql_query("SELECT password FROM `account` WHERE `login`='".$login.'" AND `password`='".$password.'");
$passworddb = mysql_fetch_array($q2);
$passworddb=$passworddb['password'];

//Проверка введенных данных
if($passworddb == $password){
$r1 = mysql_query("SELECT data FROM `account` WHERE `login`='".$login.'" AND `password`='".$password.'");
$result = mysql_fetch_array($r1);
$r1=$result['data'];
echo $r1;
}
else{
echo "1";}
}
}

```

Рис.3.12. Код файлу auth.php

Перейдем до сторінки продажу акаунту.

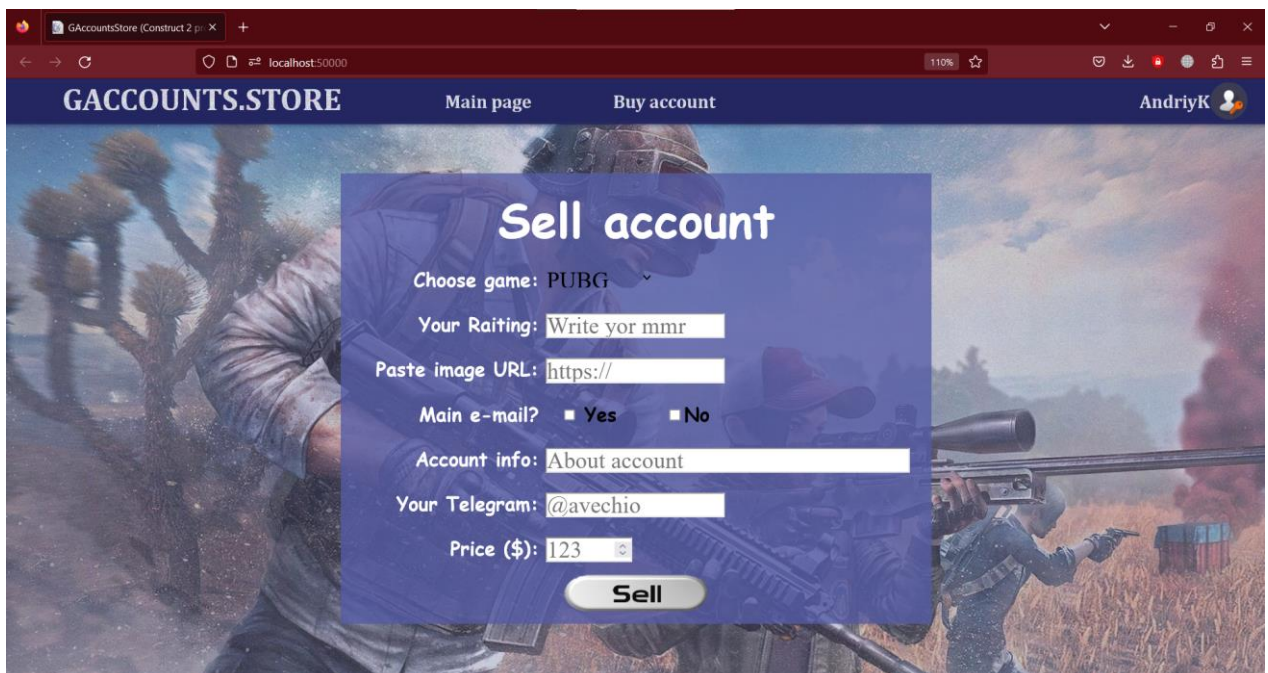


Рис.3.13. Сторінка продажу акаунту

Перед нами є форма з полями «Choose game», «Your rating», «Paste image URL», «Main e-mail», «Account info», «Your telegram», «Price». Користувач має заповнити кожен й після цього натиснути на кнопку «Sell».

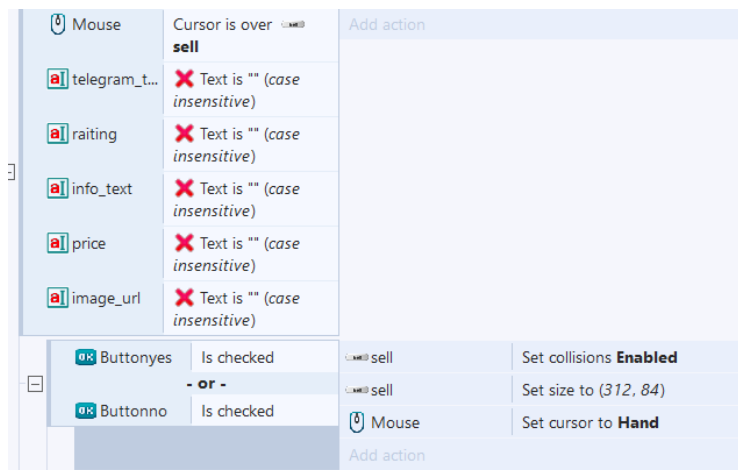


Рис.3.14. Код на перевірку чи всі поля заповнені

При зміні поля «Choose game» буде зміна фону сторінки.

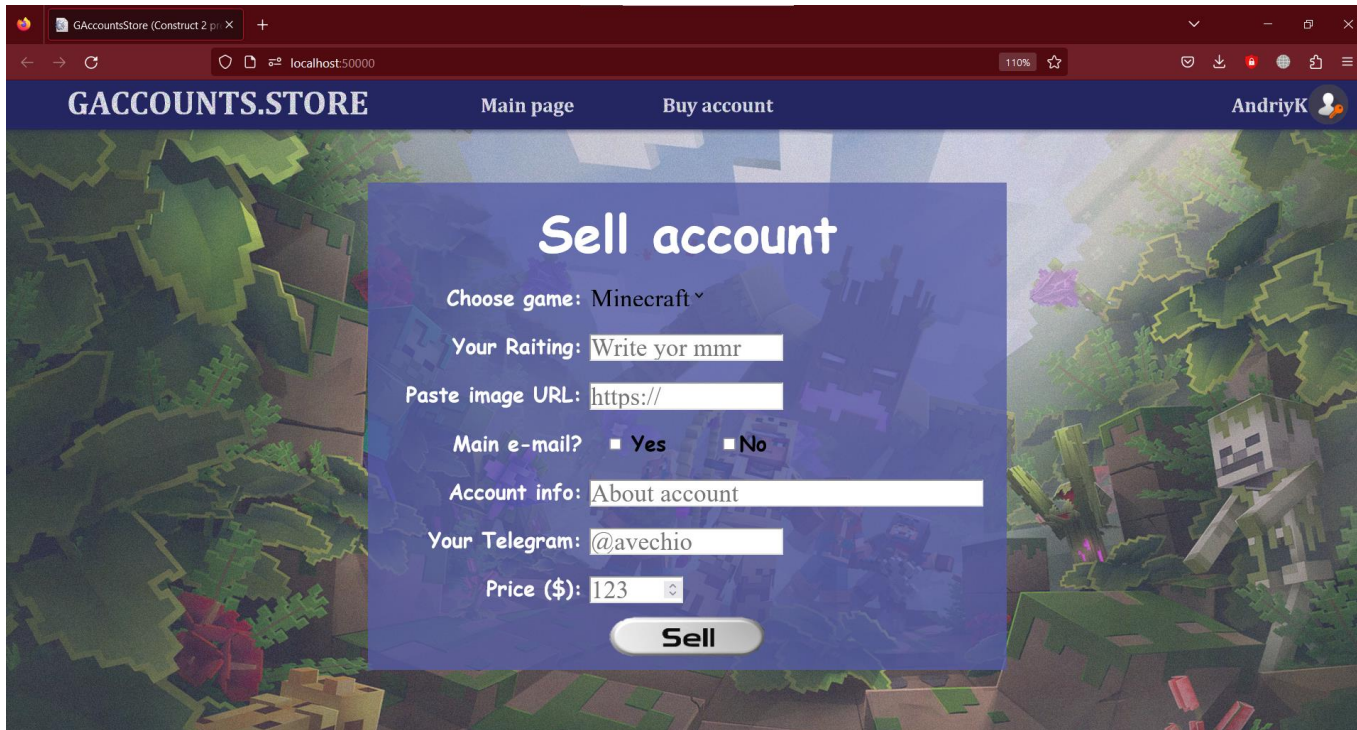


Рис.3.15. Зміна фону сторінки

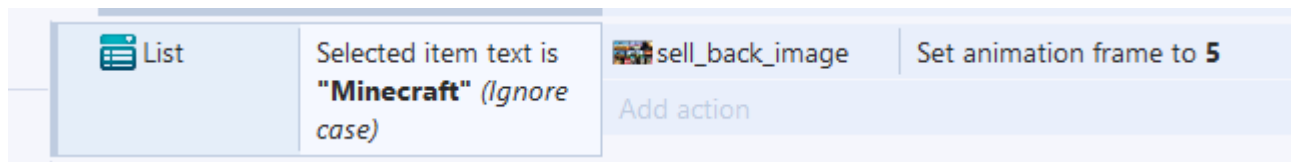


Рис.3.16. Реалізація зміни фону сторінки

Після натискання на кнопку «Sell», також за допомогою Аїах буде відправлено запит до бази даних про створення лоту для продажу акаунту. Також у базі даних для акаунту буде додано унікальний іd для швидкого пошуку та виведення на сторінці продажу та login продавця для відображення інформації про його акаунт у профілі.

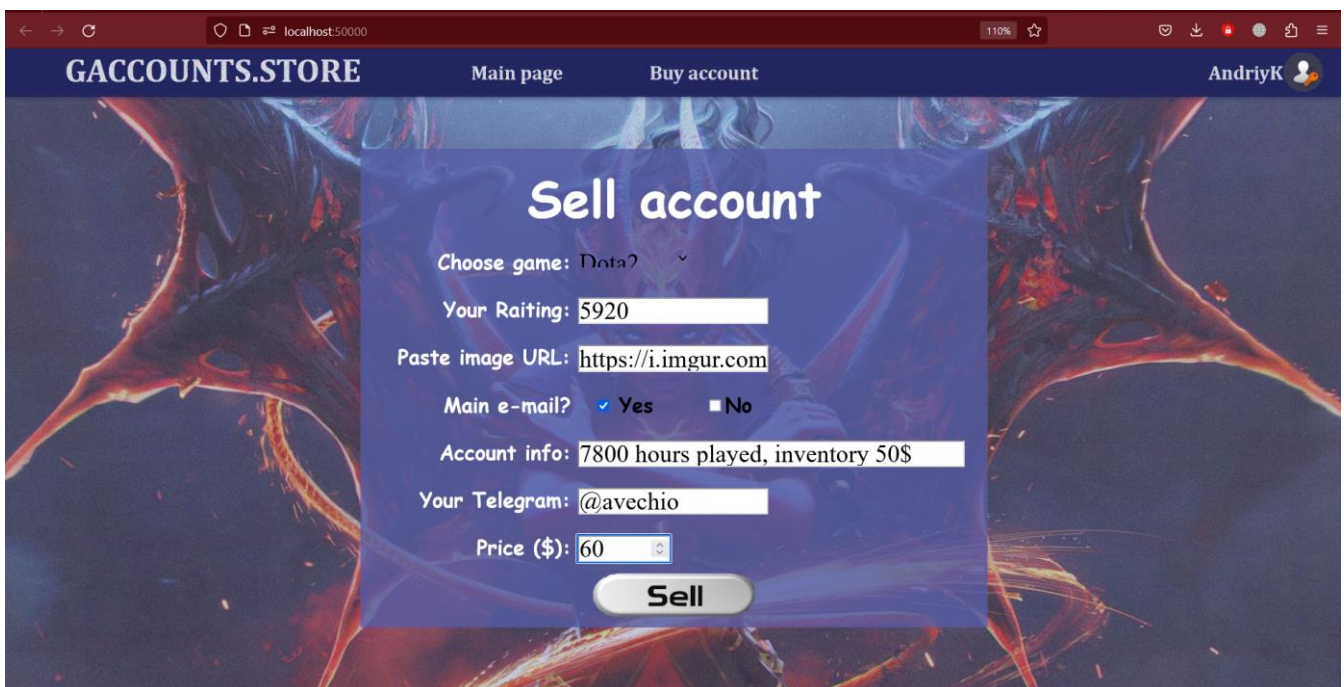


Рис.3.17. Створення лоту на продаж акаунту

	id	rating	image	email	info	telegram	price
elete	1	5877	https://i.imgur.com/1bGNmbx.png	main	inventory 15\$	@bebcki47	17.00
elete	2	4542	https://i.imgur.com/uOQ5atB.png		4600 hours played, csgo,gta5,rdr2	@dinside1794	25.00
elete	3	4205	https://i.imgur.com/xC54RTe.jpg	main	main account,main post, 10 games	@hotgirl59	10.00
elete	4	4400	https://i.imgur.com/ABxKilc.png	NULL	need calibration	@manchik	7.00
elete	5	5920	https://i.imgur.com/KIDFFJS.jpg	main	7800 hours played, inventory 50\$	@avechio	60.00

Рис.3.18. Відображення інформації про акаунт в базі даних


```

<?php
header('Access-Control-Allow-Origin: *');
mysql_host = "localhost"; // Хост базы данных
mysql_database = "id20788486_andriy"; // Имя базы данных
mysql_user = "id20788486_kalensky"; // Имя пользователя базы данных
mysql_password = "Cfsw4561200272_"; // Пароль базы данных

$link = mysqli_connect($mysql_host, $mysql_user, $mysql_password, $mysql_database);
if (!$link) {
    die("Ошибка подключения к MySQL: " . mysqli_connect_error());
}

// Получение данных из запроса
$rating = $_POST['rating'];
$image = $_POST['image'];
$email = $_POST['email'];
$info = $_POST['info'];
$telegram = $_POST['telegram'];
$price = $_POST['price'];
$login = $_POST['login'];

// Подготовка и выполнение SQL-запроса для вставки данных
$sql = "INSERT INTO dota (rating, image, email, info, telegram, price, login) VALUES (?, ?, ?, ?, ?, ?, ?)";
$stmt = mysqli_prepare($link, $sql);
mysqli_stmt_bind_param($stmt, "sssssss", $rating, $image, $email, $info, $telegram, $price, $login);

if (mysqli_stmt_execute($stmt)) {
    echo "Данные успешно добавлены в базу данных.";
} else {
    echo "Ошибка при добавлении данных: " . mysqli_error($link);
}

// Закрытие соединения
mysqli_stmt_close($stmt);
mysqli_close($link);
?>

```

Рис.3.19. Код створення лоту в базі даних

Далі створимо сторінку з вибором гри для придбання акаунту.

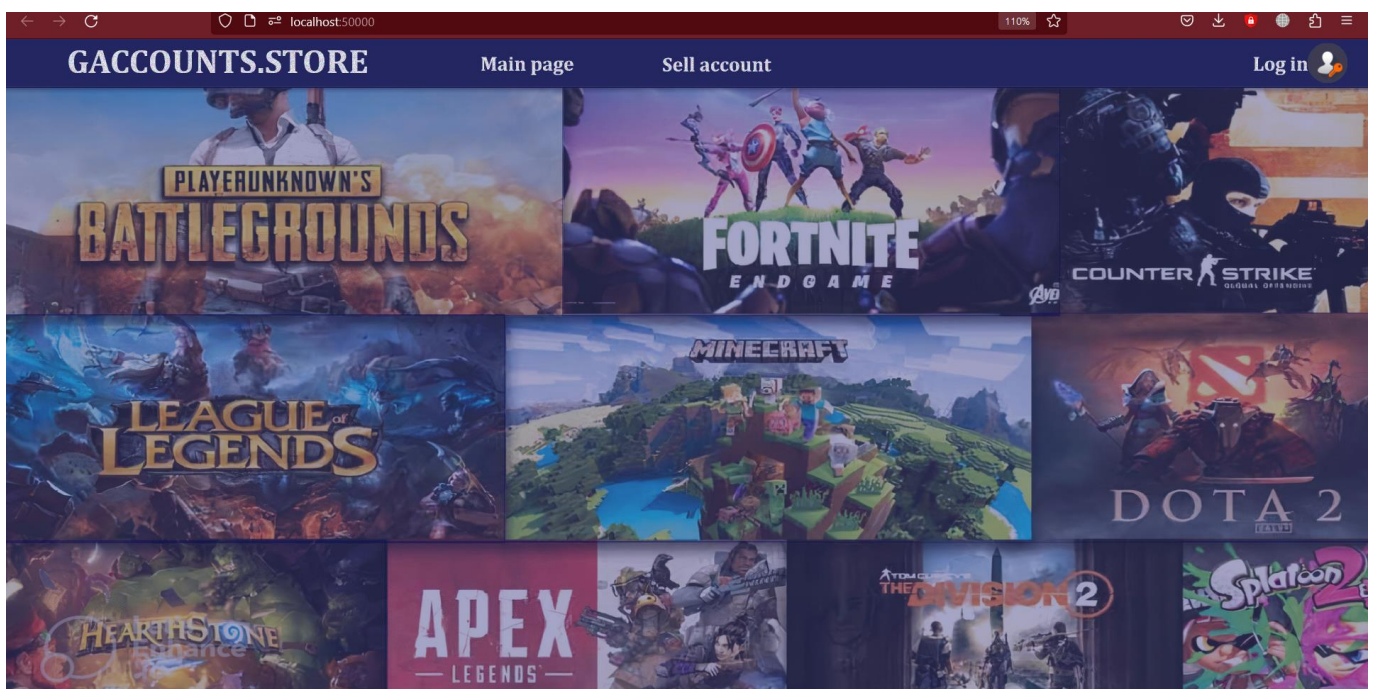


Рис.3.20. Сторінка «Buy account»

На даній сторінці користувач обирає гру в якій він хотів би придбати акаунт. При наведенні миші на обрану гру буде підсвечення її зображення, після натискання перехід до відповідної сторінки з акаунтами цієї гри.

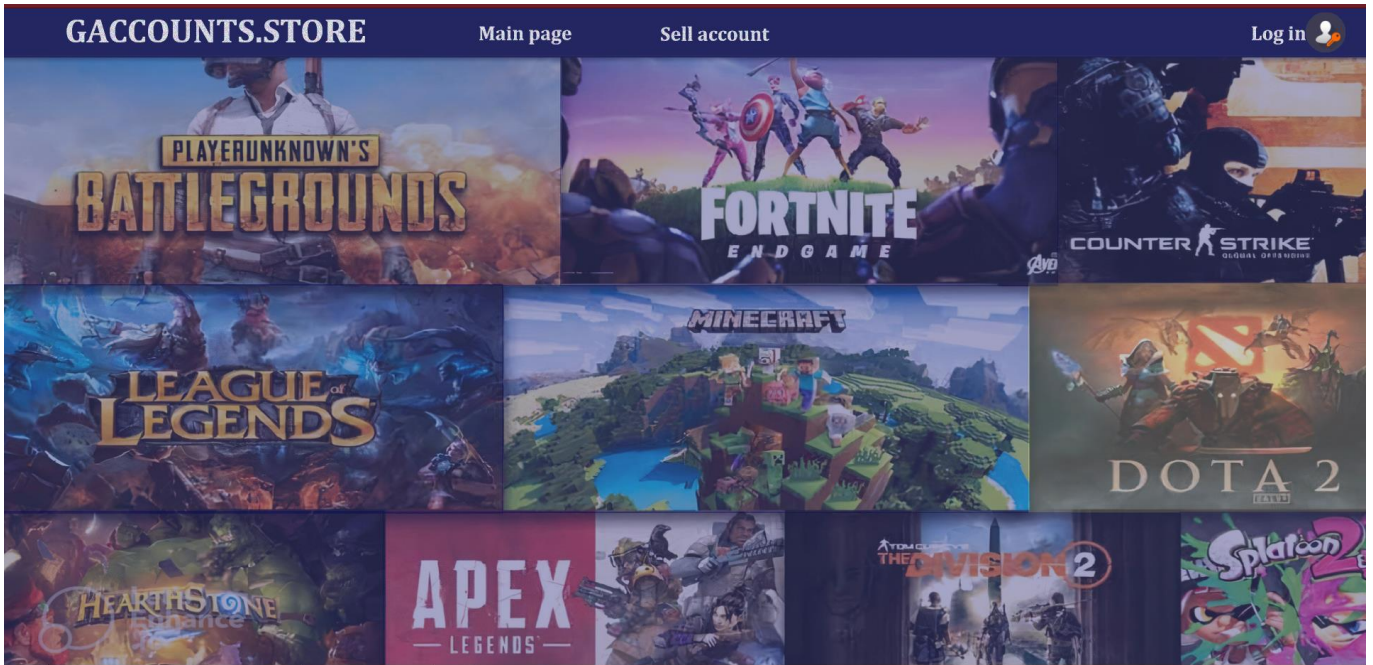


Рис.3.21. Вибір гри «Dota 2»

На сторінці придбання акаунту ми бачимо інформацію про доступні акаунти, також в деяких іграх є свої послуги наприклад «Бустинг чи калібрування акаунту».

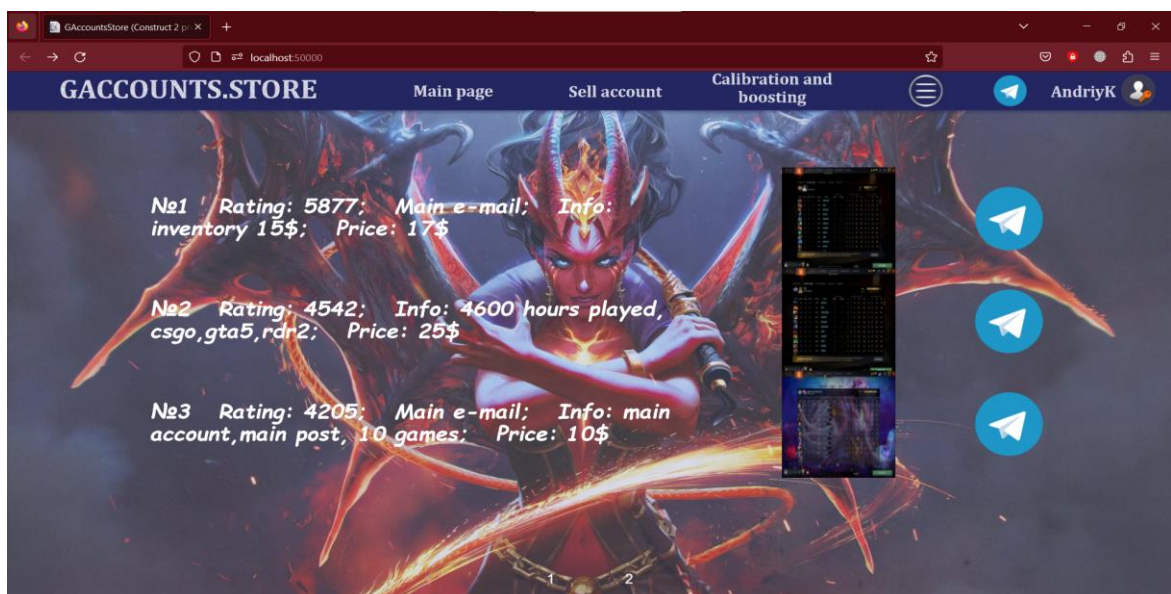


Рис.3.22. Вікно сторінки «Dota 2»

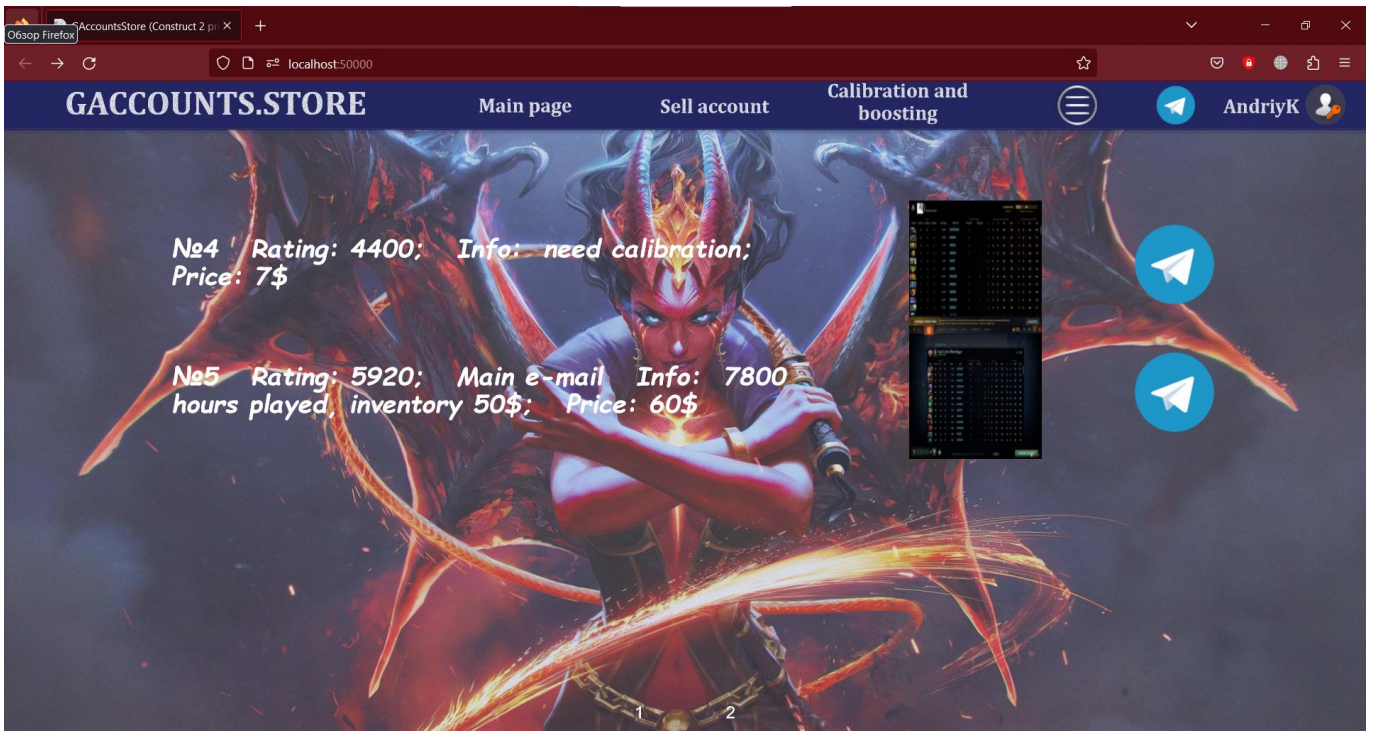


Рис.3.22. Вікно сторінки «Dota 2» при натисканні на клавішу «2»

Як ми бачимо інформація про доданий нами акаунт вище успішно додана на сторінку продажу акаунтів по грі Dota 2. Процес завантаження даних про акаунти з бази даних видно нижче.

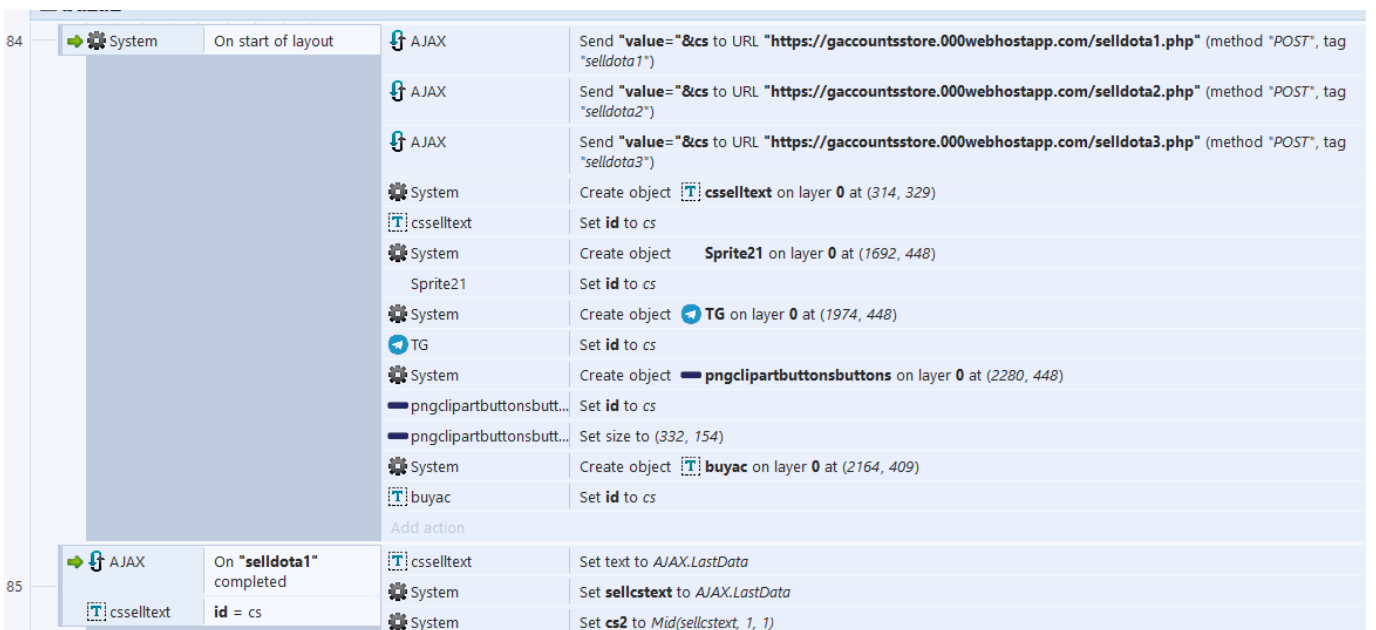


Рис.3.23. Код відображення даних про акаунти

Коли користувач обрав акаунт який він хоче придбати, йому потрібно натиснути на зображення телеграму, після цього відбудеться перехід до написання листа продавцю акаунту, де вони зможуть обговорити ціну, передати кошти та дані від акаунту.

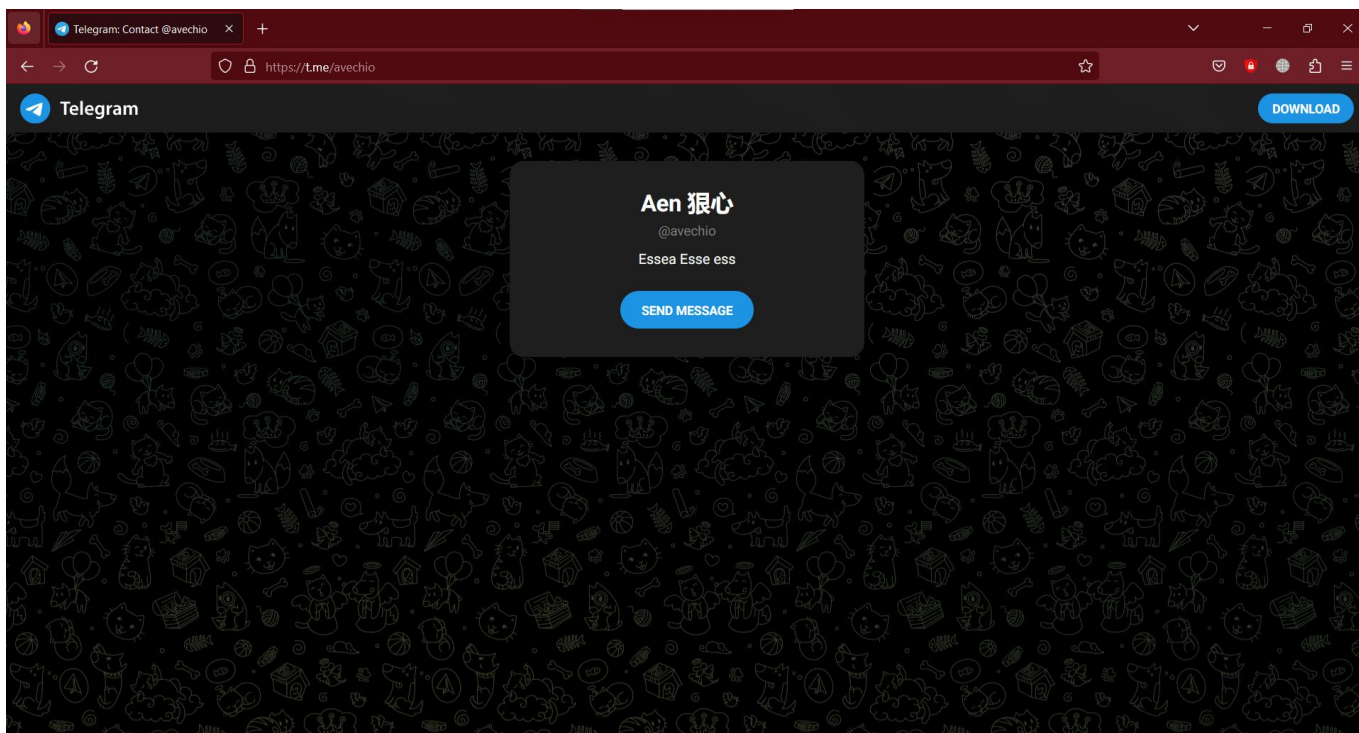


Рис.3.24. Перехід до телеграму продавця

Також на сторінці продажу акаунтів зверху видно назву сайту та навігаційне меню. У ньому є такі клавiшi як «Головна сторiнка», «Продаж акаунту», «Калiбрування та бустинг», меню з вибором iншої гри для покупки, телеграм адмiнiстрацiї сайту, логiн та фото профiлю користувача. При наведеннi на будь яку з цих клавiш курсор мишi стає у виглядi «руки», а текст змiнює колiр на жовтий.

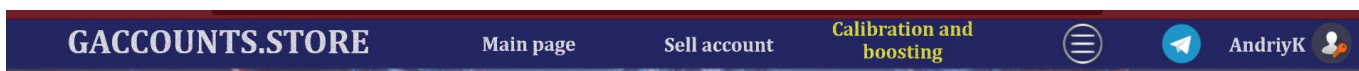


Рис.3.24. Навігаційне меню

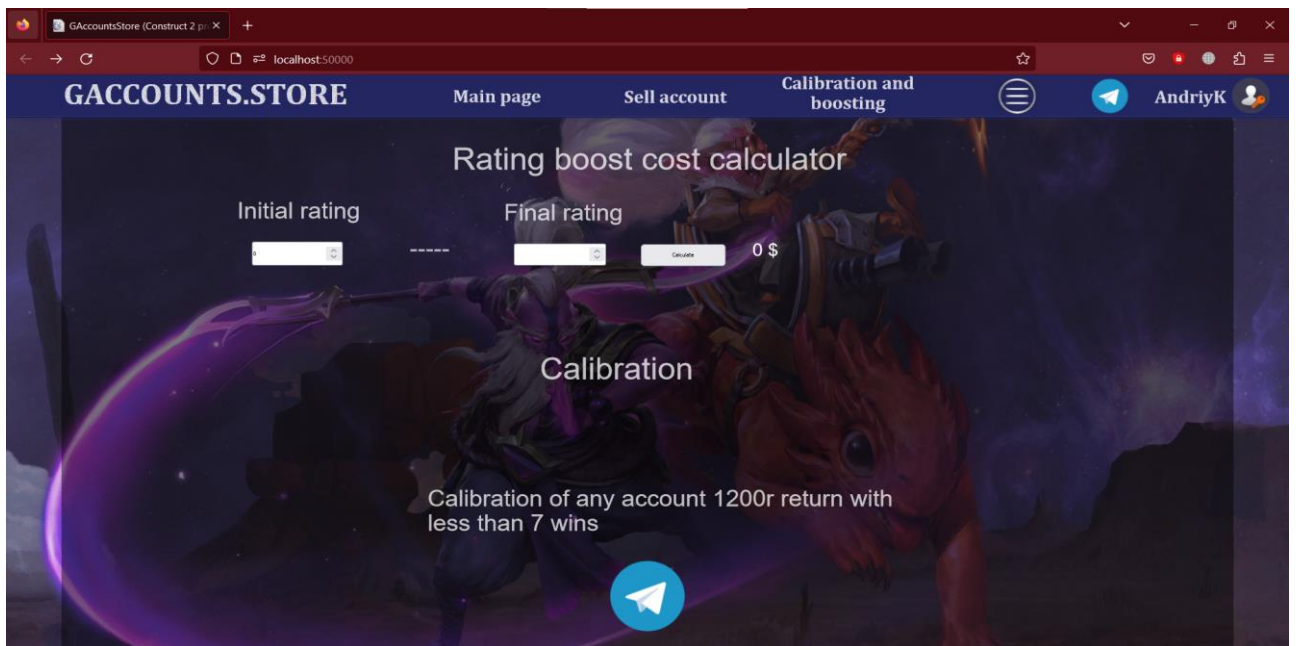


Рис.3.24. Перенос камери при натисканні на «Calibration and boosting»

На сайті є калькулятор вартості бустингу, який порахує покупцю вартість бустингу його акаунту, для цього потрібно лише ввести початковий рейтинг та бажаний і натиснути на клавішу «Calculate».

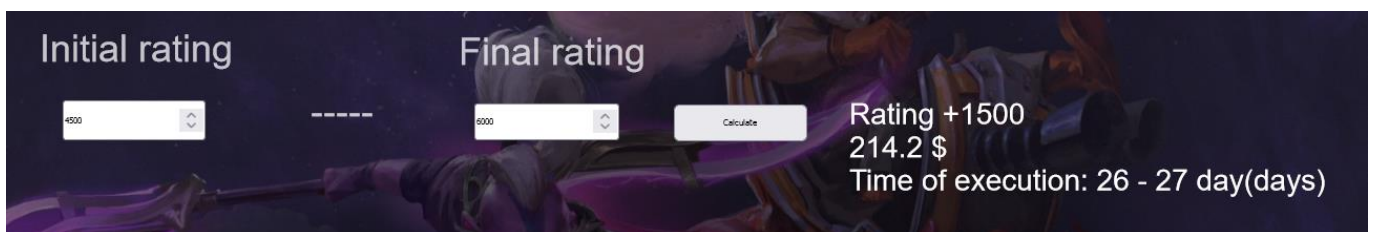


Рис.3.25. Прорахування вартості бустингу акаунту

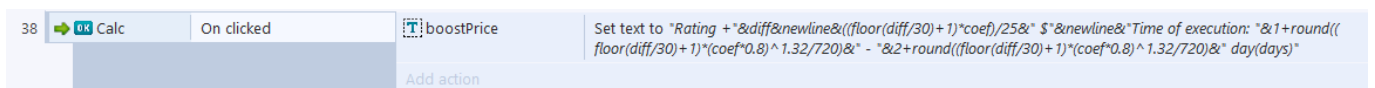


Рис.3.26. Код вирахування вартості бустингу

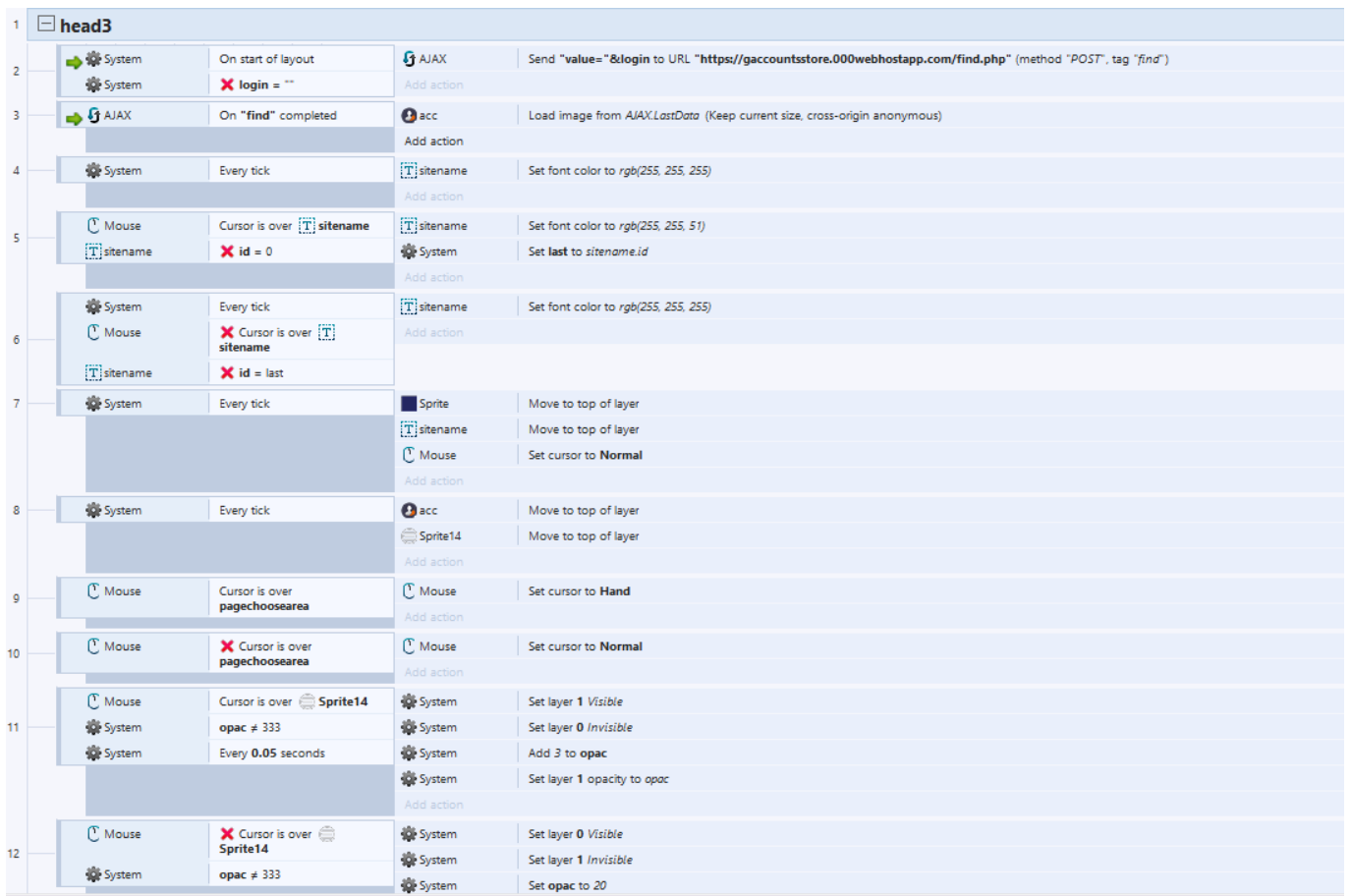


Рис.3.27. Код навігаційного меню

При натисканні на Логін користувача відбудеться перехід в його профіль. В профілі користувач може завантажити свій аватар, змінити пароль та телеграм для продажу акаунту та переглянути інформацію про свої виставлені акаунти або скасувати їх продаж. Також при зміні аватару він зміниться не тільки на сторінці профілю, а й на усіх сторінках в навігаційному меню.

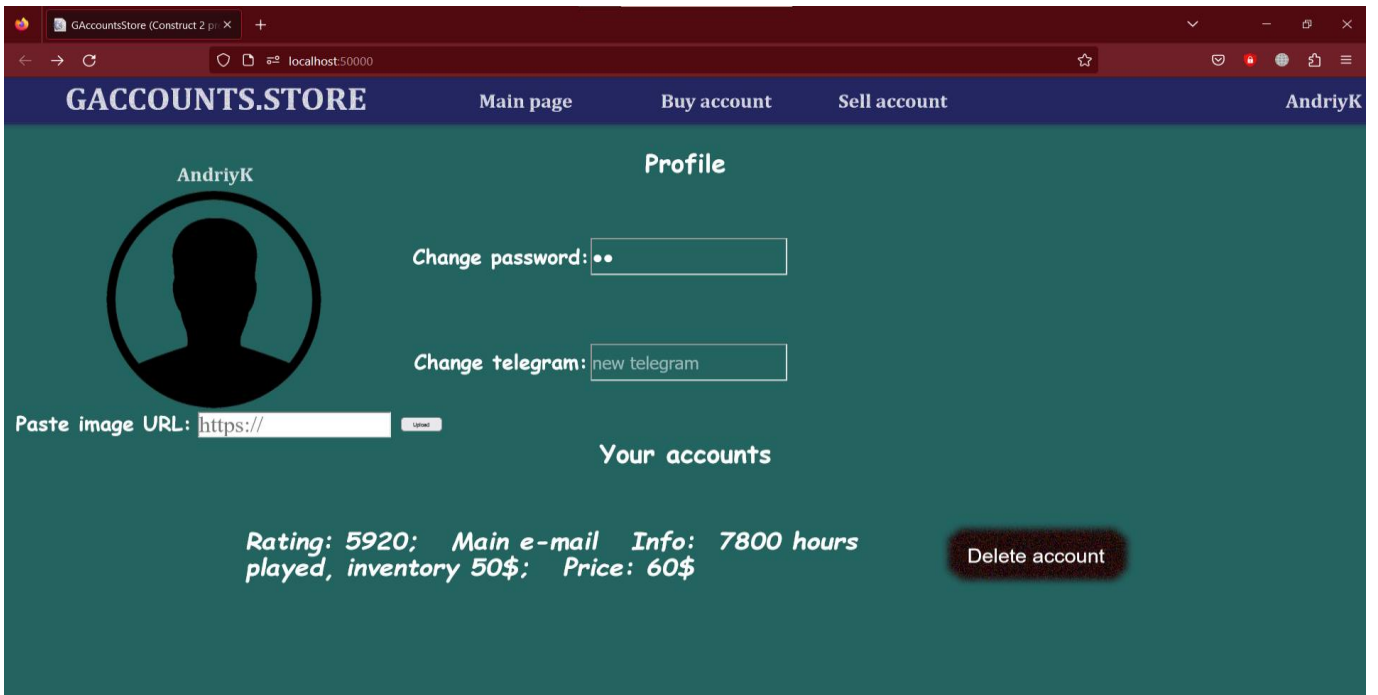


Рис.3.28. Профіль користувача

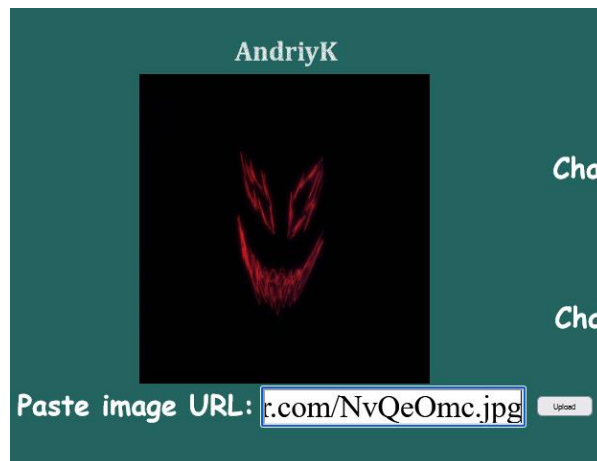


Рис.3.29. Зміна аватару

id	trigger	action
13	test	On clicked
	image_url2	Text is "" (case insensitive)
		Add action
	AJAX	Send "login"&login&"&image="&image_url2.Text to URL "https://gaccountsstore.000webhostapp.com/image.php" (method "POST", tag "image")
14	AJAX	On "image" completed
		Add action
	Sprite28	Load image from image_url2.Text (Keep current size, cross-origin anonymous)
	test_text	Set text to AJAX.LastData
		Add action
15	System	On start of layout
	System	login = ""
		Add action
	AJAX	Send "value"&login to URL "https://gaccountsstore.000webhostapp.com/find.php" (method "POST", tag "find")
16	AJAX	On "find" completed
		Add action
	Sprite28	Load image from AJAX.LastData (Keep current size, cross-origin anonymous)

Рис.3.30. Код зміни аватару

3.3.2. База даних

База даних є ключовим компонентом веб-сервісу для продажу ігрових акаунтів. Вона забезпечує зберігання та організацію інформації, яка стосується акаунтів, користувачів та інших важливих даних, необхідних для функціонування сервісу.

Моя база даних складається з таблиці «accounts», де зберігаються дані акаунтів користувачів сайту та 10 таблиць з відповідними назвами гри, де знаходиться інформація про ігри які продають користувачі.

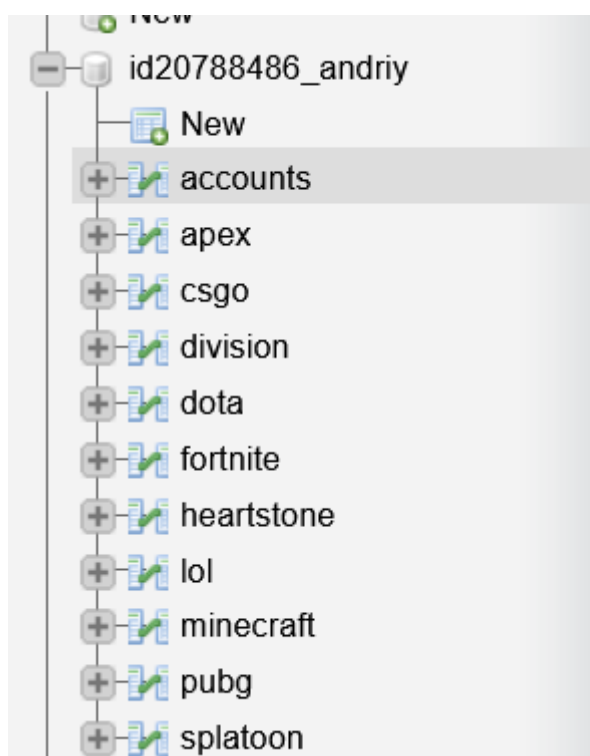


Рис.3.31. Таблиці бази даних

login	email	password	image
AndriyK	xxxuk@gmail.com	dfs2fkD	https://i.imgur.com/NvQeOmc.jpg
Tested	xxxzevs9@gmail.com	gd94Ld	NULL

Рис.3.32. Дані таблиці «accounts»

			id	rating	image	email	info	telegram	price	login	
<input type="checkbox"/>	Edit	Copy	Delete	1	5877	https://i.imgur.com/1bGNmbx.png	main	inventory 15\$	@bebcki47	17.00	Tested
<input type="checkbox"/>	Edit	Copy	Delete	2	4542	https://i.imgur.com/uOQ5atB.png		4600 hours played, csgo, gta5, rdr2	@dinside1794	25.00	Tested
<input type="checkbox"/>	Edit	Copy	Delete	3	4205	https://i.imgur.com/xC54Rte.jpg	main	main account, main post, 10 games	@hotgir159	10.00	Tested
<input type="checkbox"/>	Edit	Copy	Delete	4	4400	https://i.imgur.com/ABxKilc.png	NULL	need calibration	@manchik	7.00	Tested
<input type="checkbox"/>	Edit	Copy	Delete	5	5920	https://i.imgur.com/KIDFFJS.jpg	main	7800 hours played, inventory 50\$	@avechio	60.00	AndriyK

Рис.3.33. Дані таблиці «dota»

+ Options											
				id	rating	image	email	info	telegram	price	login
<input type="checkbox"/>	Edit	Copy	Delete	1	silver6	https://i.imgur.com/2Ru5P6u.jpg	main	cs inventory 12\$, prime csgo	@csplayer898	15.00	Tested
<input type="checkbox"/>	Edit	Copy	Delete	2	Global Elite	https://i.imgur.com/9EwwkEJ.png	main	dragon lore, legendary eagle - duo rank, gta5, inv...	@codollman	77.00	Tested

Рис.3.33. Дані таблиці «csgo»

3.4. Висновки до Розділу 3

В цьому розділі ми провели аналіз вимог і розглянули ключові аспекти розробки веб-сервісу для продажу ігрових акаунтів.

Почали ми з огляду вимог, де визначили функціональні та нефункціональні вимоги до сервісу. Ретельний аналіз вимог допоміг зрозуміти, які функції повинен мати веб-сервіс і які вимоги повинні бути задоволені, щоб забезпечити його ефективну роботу та задоволення потреб користувачів.

Далі ми перейшли до розгляду структури веб-сервісу. Виокремили головні розділи та функціональні блоки, зокрема головну сторінку, функції покупки та продажу акаунтів, а також профіль користувача. Ця структура допомагає уявити загальний макет та навігацію веб-сервісу, що сприяє зручності взаємодії користувача з сервісом.

Для реалізації клієнтської частини веб-сервісу, ми розглянули використання програми Construct 2. Це інструмент для розробки ігор та веб-додатків, який надає можливість створювати інтерактивні ігрові елементи та анімацію для покращення користувацького досвіду. Використання Construct 2 дозволяє ефективно розробляти та впроваджувати функціональність клієнтської частини веб-сервісу.

Нарешті, ми розглянули базу даних для веб-сервісу. База даних відіграє важливу роль у зберіганні та організації інформації, пов'язаної з акаунтами, користувачами та іншими даними, необхідними для функціонування.

ВИСНОВКИ

У рамках даної кваліфікаційної роботи був розроблений веб–сервіс продажу ігрових акаунтів, який спрямований на полегшення доступу до покупки та продажу ігрових акаунтів для користувачів.

Під час розробки було проведено аналіз вимог до сервісу, включаючи функціональні та нефункціональні вимоги. Було визначено, що користувачі повинні мати можливість переглядати доступні ігрові акаунти, встановлювати ціни, здійснювати покупки та продажі, а також комунікувати з іншими користувачами. Вимоги до безпеки, захисту персональних даних також були враховані під час розробки сервісу.

Веб–сервіс був реалізований з використанням сучасних технологій веб–розробки, включаючи Construct 2, HTML, CSS, PHP. Користувацький інтерфейс був розроблений з урахуванням зручності взаємодії для користувачів, а серверна частина містила функціонал для обробки запитів, валідації даних та зберігання інформації про ігрові акаунти.

Процес реалізації включав створення бази даних для зберігання ігрових акаунтів, користувачів, розробку логіки бізнес–процесів, таких як додавання ігрових акаунтів до бази даних.

Веб–сервіс успішно пройшов тестування та демонструє задовільні результати в роботі. Він може бути використаний як ефективний інструмент для купівлі та продажу ігрових акаунтів, дозволяючи користувачам з легкістю знаходити та здійснювати угоди з іншими гравцями.

Загалом, розроблений веб–сервіс продажу ігрових акаунтів має потенціал для подальшого розвитку та вдосконалення. Додаткові функціональні можливості, такі як оцінка акаунтів, система рейтингів або можливість обміну акаунтами, можуть бути додані для покращення досвіду користувачів. Крім того, важливо забезпечити

постійну підтримку безпеки та захисту від шахрайства, а також розширити базу користувачів та акаунтів для збільшення масштабу та активності на платформі.

У результаті даної кваліфікаційної роботи було успішно розроблено та впроваджено веб-сервіс продажу ігрових акаунтів, який надає зручний та безпечний спосіб здійснення покупки/продажу акаунтів для користувачів. Цей проект може слугувати основою для подальшого розвитку та вдосконалення подібних веб-сервісів у майбутньому.

СПИСОК БІБЛІОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Український вебдовідник. [Електронний ресурс] – Режим доступу: <https://css.in.ua/> (дата звернення: 04.06.2023). – Назва з екрану.
2. Ед Т. HTML, XHTML та CSS для чайників / Тігтел Ед – Київ: Видавництво "Діалектика", 2011. – 400 с.
3. Стівен Ш. HTML, XHTML та CSS. Біблія користувача – 5-е видання / Шафер Стівен – Київ: Видавництво "Діалектика", 2011. – 656 с.
4. Фаулер К. Рефакторинг коду на JavaScript: покращення проекту існуючого коду – 2-ге видання. /К. Фаулер – Київ: Видавництво "Діалектика", 2019. – 464 с.
5. Девід М. JavaScript та jQuery: вичерпне керівництво. / Макфарланд Девід – Київ: Видавництво "Діалектика", 2015. – 880 с.
6. Морган Н. Javascript для дітей. / Нік Морган – Київ: Видавництво "Діалектика", 2016. – 288 с.
7. Construct 2 [Електронний ресурс] – Режим доступу: <https://www.construct.net/> (дата звернення: 04.06.2023). – Назва з екрану.
8. Construct 2 Tutorials [Електронний ресурс] – Режим доступу: <https://www.construct.net/en/tutorials/construct-2?flang=36> (дата звернення: 04.06.2023). – Назва з екрану.
9. Уроки по Construct 2 [Електронний ресурс] – Режим доступу: <https://4pda.to/forum/index.php?showtopic=376729> (дата звернення: 04.06.2023). – Назва з екрану.
10. Фрімен Э. Вивчаємо HTML, XHTML и CSS / Э. Фрімен – Київ: Видавництво "Laurus", 2008. – 412 с.
11. Сухов К. HTML 5. Путівник з технології / К. Сухов – Київ: Видавництво "ДМК Прес", 2013. – 352 с.

12. Макаренко, Є. Каскадні таблиці стилів CSS: Посібник для початківців / Є. Макаренко. – Київ: Видавництво "Правильна книга", 2017. – 128 с.
13. Риккарди Г. Системи баз даних. Теорія та практика використання в Internet і середовищі Java / Г. Ріккарді – Вашингтон: Видавництво "Вільямс", 2001. – 579 с.
14. Новиков Б. Налаштування пристроїв баз даних / Б.А. Новіков, Г.Р. Домбровська – Київ: Видавництво "Laurus", 2006. – 328 с.
15. Наумов А. Системи управління базами даних і знань / А. Наумов – Київ: Видавництво "Фінанси та статистика", 1991. – 354 с.
16. Глушаков С.В. Бази Даних / С.В. Глушаков, Д.В. Ломотько – Київ: Видавництво "АСТ", 2002. – 474 с.
17. Суэринг С. PHP та MySQL. Біблія програміста / С. Суэринг, Т. Конверс, Д. Парк – Київ: Видавництво "Діалектика", 2010. – 452 с.
18. Харрінгтон Д. PHP. Трюки / Д. Харрінгтон – Київ: Видавництво "Laurus", 2008. – 397 с.