МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет кораблебудування імені адмірала Макарова

Навчально-науковий інститут комп'ютерних наук та управління проєктами
Кафедра програмного забезпечення автоматизованих систем
Спеціальність 121 «Інженерія програмного забезпечення»
Освітня програма «Інженерія програмного забезпечення»

«Допущений до захисту»
Завідувач кафедри
_____
«___» _____ 2022 р.

КВАЛІФІКАЦІЙНА РОБОТА
на здобуття ступеня вищої освіти «бакалавр»

на тему: Розробка програмного забезпечення вебсайту для клубу прокату настільних ігор _____

Виконав: студент 4151 групи
_____ Алікіна Д.І.
(підпис, ПІБ)

Керівник роботи:
Викладач, к.т.н.
(посада, науковий ступень вчене звання)
_____ Пухалевич А.В.
(підпис, ПІБ)

Миколаїв – 2022 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет кораблебудування імені адмірала Макарова

Навчально-науковий інститут комп'ютерних наук та управління проєктами
Кафедра програмного забезпечення автоматизованих систем
Спеціальність 121 «Інженерія програмного забезпечення»
Освітня програма «Інженерія програмного забезпечення»

«ЗАТВЕРДЖУЮ»
Гарант освітньої програми
_____ доц. Макарова Л.М.
(підпис)
«___» _____ 2022 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ
на здобуття ступеня вищої освіти «бакалавр»

Студенту Алікіній Дарині Ігорівні
(Прізвище, ім'я, по батькові)
1. Тема роботи: Розробка вебсайту для клубу прокату настільних ігор
Керівник роботи Пухалевич Андрій Володимирович
Затверджені наказом ректора №256-уч від «11» травня 2022 р.
2. Термін подання роботи: 10.06.2022 р.
3. Вихідні дані по роботі:

4. Перелік питань, що належать до розробки (найменування розділів)
Титульний аркуш; Завдання на кваліфікаційну роботу; Анотація (українською та англійською мовами); Зміст; Перелік умовних позначень, символів, одиниць та термінів (при необхідності); Вступ; Опис предметної галузі та постановка задачі; Проєкт програмного забезпечення (ескізний, технічний та робочий); Результати розробки програмного забезпечення; Розділ з охорони праці; Висновки; Список використаних джерел; Додатки (технічне завдання, текст програми, опис програми, інструкція користувача, програма та методика випробувань програмного забезпечення).

5. Перелік презентаційних матеріалів тема, виконавець, керівник; мета кваліфікаційної роботи; аналіз предметної галузі та постановка задачі; прийняті рішення з розробки ПЗ (моделі ПЗ та даних); приклади програмного коду; інтерфейс користувача (форми, вікна, екрани); висновки.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
4	Гурець Н.В., ст. викладач		

7. Дата видачі завдання 11.05.2022 р. _____

К А Л Е Н Д А Р Н И Й П Л А Н

Номер	Назва етапів роботи	Терміни виконання	Примітка
1.	Підготовка розділу Вступ	24.03.2022	ПП*
2.	Опис та аналіз предметної галузі	28.03.2022	ПП*
3.	Постановка задачі	31.03.2022	ПП*
4.	Ескізний проєкт програмного забезпечення	27.04.2022	
5.	Технічний проєкт програмного забезпечення	06.05.2022	
6.	Робочий проєкт програмного забезпечення	13.05.2022	
7.	Підготовка розділу Результати розробки програмного забезпечення	17.05.2022	
8.	Підготовка розділу з охорони праці	18.05.2022	ПП*
9.	Підготовка розділу Висновки	19.05.2022	
10.	Оформлення списку використаних джерел та додатків	20.05.2022	
11.	Подання на кафедру ПЗАС тексту остаточного варіанту роботи, підписаного її керівником, у роздрукованому та електронному форматі разом із заявами щодо самостійності виконання роботи та ідентичності друкованої та електронної версій роботи (Додатки 1 і 2 «Порядку здійснення заходів з перевірки робіт на наявність текстових збігів/ідентичності/схожості із використанням програмно-технічних засобів», який введений в дію наказом ректора НУК за №20 від 20.01.2020 р.)	10.06.2022	не пізніше, ніж за 14 діб до захисту (згідно з п. 4.1 зазначеного Порядку)
12.	Підготовка презентації та доповіді	12.06.2022	
13.	Попередній захист роботи на засіданні кафедри ПЗАС	14.06.2022	
14.	Подання на кафедру ПЗАС електронних версій наступних документів у форматі pdf: кваліфікаційної роботи; файлу-опису кваліфікаційної роботи (згідно з Додатком до наказу ректора НУК за №287-уч від 19.05.2020 р.); презентації доповіді	24.06.2022	

* - за результатами переддипломної практики (ПП), яка була з 21.03.2022 до 08.05.2022 р.

Студент _____ Алікіна Д.І.
(підпис) (ПІБ)

Керівник роботи _____ Пухалевич А.В.
(підпис) (ПІБ)

АНОТАЦІЯ

Кваліфікаційна робота «Розробка програмного забезпечення вебсайту для клубу прокату настільних ігор» присвячена розв’язанню практичної задачі інженерії програмного забезпечення, що характеризується комплексністю та невизначеністю умов, а саме розробці програмного забезпечення відповідного вебсайту.

Кваліфікаційна робота містить аналіз предметної області, постановку задачі на розробку програмного забезпечення вебсайту для клубу прокату настільних ігор, ескізний, технічний та робочий проєкти програмного забезпечення, розробку програмного забезпечення вебсайту для клубу прокату настільних ігор, його тестування та розділ з охорони праці.

Кваліфікаційну роботу виконано на 96 сторінках друкованого тексту, робота містить 28 рисунків, 5 таблиць 5 додатків та список використаних джерел з 9 найменувань.

ABSTRACT

The qualification work "Website software development for a board game rental club" is devoted to solving a practical problem of the software engineering characterized by complexity and uncertainty - software development of the corresponding website.

Qualification work includes subject area analysis, task development for website software for board game rental club, sketch, technical and working software projects, website software development for board game rental club, its testing and labor protection section.

Qualification work is performed on 96 pages of printed text, the work contains 28 figures, 5 tables, 5 appendices and a list of used sources of 9 items.

ЗМІСТ

ВСТУП 7

1 ОПИС РОБОТИ ТА АНАЛІЗ СПОСОБІВ СТВОРЕННЯ ВЕБСАЙТІВ КЛУБУ ПРОКАТУ НАСТІЛЬНИХ ІГОР. ПОСТАНОВКА ЗАДАЧІ НА РОЗРОБКУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ВЕБСАЙТУ ДЛЯ КЛУБУ ПРОКАТУ НАСТІЛЬНИХ ІГОР 8

1.1 Опис роботи клубу прокату настільних ігор 8

1.2 Аналіз способів створення вебсайтів 10

1.3 Аналіз вебсайтів прокату настільних ігор 13

1.4 Постановка задачі на розробку програмного забезпечення вебсайту для клубу прокату настільних ігор 15

2 ПРОЄКТ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ВЕБСАЙТУ ДЛЯ КЛУБУ ПРОКАТУ НАСТІЛЬНИХ ІГОР 18

2.1 Ескізний проєкт програмного забезпечення програмного забезпечення вебсайту для клубу прокату настільних ігор 18

2.1.1 Побудова контекстної діаграми програмного забезпечення вебсайту для клубу прокату настільних ігор 18

2.1.2 Побудова моделі варіантів використання 20

2.1.3 Специфікації варіантів використання 21

2.1.4 Сценарії варіантів використання 32

2.1.5 Розробка інтерфейсу програмного забезпечення 33

2.2 Технічний проєкт програмного забезпечення вебсайту для клубу прокату настільних ігор 36

2.2.1 Логічна модель даних 36

2.2.2 Структура класів 36

2.2.3 Специфікація класів 37

2.2.4 Динамічна модель програмного забезпечення 38

2.3 Робочий проєкт програмного забезпечення вебсайту для клубу прокату настільних ігор 40

2.3.1 Фізична модель даних програмного забезпечення вебсайту для клубу прокату настільних ігор 40
2.3.2 Обґрунтування вибору мови та системи програмування 42
2.3.3 Тестування варіантів використання 44
2.3.4 Випробування програми 52
3 РЕЗУЛЬТАТИ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ВЕБСАЙТУ ДЛЯ КЛУБУ ПРОКАТУ НАСТІЛЬНИХ ІГОР 55
4 ОХОРОНА ПРАЦІ 56
ВИСНОВКИ 59
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 60
ДОДАТОК А ТЕХНІЧНЕ ЗАВДАННЯ 61
ДОДАТОК Б ПРОГРАМА ТА МЕТОДИКА ВИПРОБУВАНЬ 73
ДОДАТОК В ІНСТРУКЦІЯ КОРИСТУВАЧА 76
ДОДАТОК Г ТЕКСТ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ 79
ДОДАТОК Д ОПИС ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ 93

ВСТУП

Для полегшення роботи, ведення обліку та інформування відвідувачів про свої послуги сучасні розважальні заклади часто використовують вебсайти. Для того, щоб у відвідувача склалось гарне враження та виникло бажання користуватись послугами закладу надалі – необхідно мати вебсайт з сучасним дизайном та зручним функціоналом, що задовольнить усі потреби в отриманні інформації, щодо послуг закладу. Також важливою підставою для створення вебсайту є можливість просування закладу та його послуг у мережу, що допоможе у знаходженні нових відвідувачів та просуванні послуг. Клуб прокату настільних ігор «Гральна кістка» не має власного вебсайту, тому розробка програмного забезпечення вебсайту для клубу прокату настільних ігор «Гральна кістка» має практичну цінність.

Мета кваліфікаційної роботи – розв’язання практичної задачі інженерії програмного забезпечення, що характеризується комплексністю та невизначеністю умов, а саме розробка програмного забезпечення вебсайту для клубу прокату настільних ігор «Гральна кістка».

Для досягнення поставленої мети потрібно виконати наступні завдання:

1) Виконати аналіз предметної області клубу прокату настільних ігор та аналіз чинних вебсайтів таких закладів.
2) Розробити проєкт програмного забезпечення вебсайту для клубу прокату настільних ігор «Гральна кістка».
3) Виконати програмну реалізацію вебсайту клубу прокату настільних ігор «Гральна кістка».
4) Провести випробування функціональності вебсайту клубу прокату настільних ігор «Гральна кістка».
5) Розглянути питання охорони праці.

1 ОПИС РОБОТИ ТА АНАЛІЗ СПОСОБІВ СТВОРЕННЯ ВЕБСАЙТІВ КЛУБУ ПРОКАТУ НАСТІЛЬНИХ ІГОР. ПОСТАНОВКА ЗАДАЧІ НА РОЗРОБКУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ВЕБСАЙТУ ДЛЯ КЛУБУ ПРОКАТУ НАСТІЛЬНИХ ІГОР

1.1 Опис роботи клубу прокату настільних ігор

У наш час використання комп’ютерів та передових технологій стало незамінною частиною життя більшості людей, адже їх застосовують у багатьох сферах життя для проведення розрахунків, зберігання документів, обробки даних, також, як засіб зв’язку та розваг. Зараз важко знайти когось, хто жодного разу не тримав у руках хоча б сучасний смартфон або не чув про високі досягнення у можливостях новітньої техніки, яка і надалі еволюціонує та ще більше вдосконалюється з кожним днем. Автоматизації та оптимізації вже підлягло багато речей та дій, що дозволяє економити час, зусилля і значно скорочувати трудомісткість процесів.

Все частіше відкриваються різні заклади, в яких виникає потреба у зберіганні інформації про клієнтів, обліку надання послуг, наочному зручному представленні їх асортименту. До таких закладів відноситься клуб прокату настільних ігор «Гральна кістка». Робота цього закладу організована таким чином, що коли приходить відвідувач, то співробітник пропонує йому журнал з вартістю послуг, де описано список ігор та вартість прокату гри за годину. Коли відвідувач зробить свій вибір, він повідомляє про це співробітнику та проходить за стіл у приміщенні клубу прокату настільних ігор. Співробітник записує у журнал прізвище відвідувача, номер столу, назву гри та час, на який гру взято в прокат. Після вичерпання часу прокату гри співробітник повідомляє відвідувача про це та стягує вартість за час користування грою, а гру ставить на вітрину для подальшої здачі в прокат. У разу пошкодження гри заставна вартість не повертається відвідувачеві.

Для полегшення роботи, ведення обліку та інформування відвідувачів про свої послуги клубу прокату настільних ігор «Гральна кістка» потрібно створити вебсайт, що містить інформацію про співробітників, яка містить ПІБ, фото, електронну адресу та посаду. Для кожного співробітника створюється свій обліковий запис користувача з логіном та паролем, що одразу записується до бази даних. Окрім цього розроблюване програмне забезпечення містить інформацію про ігри, яка містить назву, вартість прокату за годину, опис складу компонентів, її правила та статус, що показує чи може взяти у прокат гру відвідувач, прізвище відвідувача, якому видана гра та працівника, що видав гру, час початку та тривалість прокату обраної гри та суму заставної вартості, що повертається відвідувачеві після закінчення тривалості прокату гри. Розрахунок готівкою містить суму, яку необхідно сплатити відвідувачеві за час користування грою, дату та прізвище співробітника, що приймав розрахунок готівкою. У разі якщо відбулось пошкодження гри або певних її компонентів – заставна вартість не буде повернута відвідувачеві. Також у разі неповернення гри – заставна вартість лишається на придбання такої ж нової гри.

Розроблюване програмне забезпечення дозволить надати всю необхідну інформацію для користувача, яка може бути корисною і дає користувачу уявлення про роботу клубу прокату настільних ігор. Ця розробка полегшить роботу працівникам завдяки автоматизації процесу введення, додавання та видалення даних.

Робота програмного забезпечення вебсайту «Клуб прокату настільних ігор» буде організована таким чином, що коли відвідувач відкриває головну сторінку сайту, то бачить перед собою список ігор, ціну прокату за годину, опис компонентів гри, її правила, а також те, чи можна взяти в прокат бажану гру, записавшись у працівника, вказавши йому час прокату та залишивши вказану ним суму заставної вартості за користування грою. Після закінчення часу прокату працівником буде перевірено відповідність складу компонентів гри її опису на вебсайті, після чого при наявності усіх вказаних компонентів

відбувається розрахунок готівкою, де заставна вартість буде повернута відвідувачеві або просто вирахувана з вартості прокату гри за витрачений відвідувачем час. А у разу відсутності або пошкодженні одного або декількох компонентів – заставна вартість буде вирахувана як сплата за пошкодження гри.

1.2 Аналіз способів створення вебсайтів

Конструкторами вебсайтів називають вебсервіси, які дозволяють створити вебсайт, вказавши необхідні блоки та вибравши дизайн з певного переліку.

Поява конструкторів вебсайтів сприяла тому, що індустрія створення вебсайтів досягла небачених темпів розвитку та навіть простий користувач інтернету може зробити це самостійно, але новачків може заплутати різноманіття платформ для створення вебсайтів.

Наприклад, Wix, який є візуальним конструктором з принципом no-code, що дозволяє створити вебсайт просто редагуючи блоки. Його перевагами є:

- Простота реєстрації через обліковий запис Google або Facebook.

- Понад 500 шаблонів для сайту будь-якого типу.

- Повноцінна хмарна платформа, якою можна користуватись з будь-якого пристрою за наявністю інтернету.

- Створений сайт одразу розміщується в інтернеті, що дозволяє не платити за окремий хостинг.

Недоліками Wix є:

- Відсутність можливості додавання ділянок коду HTML і CSS.

- Більшість корисних та цікавих опцій – платні.

- Файли вебсайту не можливо забрати з Wix у разі зміни платформи розміщення.

Weebly схожий на Wix, але має трохи дорожчі плани для всіх пакетів. Візуальний конструктор Weebly простіший, ніж Wix, що спрощує

використання платформи. нього менше шаблонів — всього близько 70. Переваги Weebly:

- Хмарна платформа, що надає у користування хостинг та безплатний домен.

- Можливість додавання ділянок коду HTML.

Недоліки Weebly:

- Комісія за транзакцію розміром у 3% за кожну покупку у магазині електронної комерції, якщо не використовується план Business або Business Plus.

- Довгий та некрасивий домен та адреса або плата за підключення власної адреси.

WordPress.com – найрозповсюдженіша платформа для створення вебсайтів. Вартість розміщення сайту на WordPress.com складає від 5$ за місяць.

SquareSpace приділяє особливу увагу саме дизайну, пропонує понад 80 шаблонів, містить функцію редагування фотографій та має адаптивні шаблони для того, щоб сайт можна було переглядати на мобільних пристроях.

Переваги SquareSpace:

- Елегантний та інтуїтивно зрозумілий інтерфейс.

- Має достатню кількість добре продуманих шаблонів.

- Можна легко створити свій макет за допомогою методу перетягування.

Недоліки SquareSpace:

- Відсутність безплатного тарифу.

- Висока вартість планів.

- Менша гнучкість для налаштування сайту.

- Відсутність доступної мовної версії [1].

Система управління контентом (CMS) – це програма, що дозволяє публікувати вебсторінки та управляти контентом вебсайту. CMS має містити такі функції:

- Спосіб легко публікувати контент.

- Функції редагування сторінок.
- Варіанти таксономії тексту.
- Високий рівень налаштованості.

Наприклад WordPress була задумана як платформа для ведення блогу, але виросла досить далеко за ці межі можливостей, є простою у використанні та може буде налаштована будь-яким чином. Публікація є простою з доступом до редактора для форматування контенту. Крім цього підтримується реєстрація користувачів.

Переваги WordPress:

- Безкоштовна CMS для створення вебсайтів.
- Безліч доступних ресурсів, щоб розібратись у використанні платформи.
- Можливість робити будь-які типи вебсайтів.

Недоліки WordPress:

- Його необхідно вивчати, зробити сайт без розуміння платформи буде важко.
- Відсутнє обслуговування клієнтів, доведеться з усіма проблемами розбиратися самостійно.

Joomla - відмінний варіант, якщо є необхідність управляти безліччю настроюваних типів повідомлень. Тобто, це ідеальний варіант для запуску вебсайту на текстовому контенті.

Переваги Joomla:

- Підтримка користувацьких повідомлень.
- Багатомовна підтримка.
- Підтримка унікальних шаблонів для кожного типу сторінки.

Недоліки Joomla:

- Joomla не так зручна для початківців, як WordPress, тому знадобиться більше часу на вивчення та освоєння цієї CMS.

Drupal - CMS з відкритим вихідним кодом, що пропонує гнучку систему для роботи з налаштованими типами повідомлень

Переваги Drupal**:**

- Підтримка багатомовних сайтів.
- Детальна система таксономії.
- Розширені налаштування управління користувачами та дозволом [2].

Для розробки вебсайтів з нуля потрібно освоїти деякі з призначених для цього мов, наприклад:

- JavaScript, яка є однією з найпоширеніших мов, що використовуються для створення вебсайтів. Усі сценарії, що написані на JS, виконуються у браузері.
- PHP – це серверна мова, яку в основному використовують для програмування вебсайтів. Популярність PHP обумовлена простотою, гнучкістю та продуктивністю.
- CSS – відповідає за зовнішній вигляд вебсторінок, її можна тільки умовно назвати мовою програмування, з використанням якої можна створювати анімації та втілювати різні дизайнерські ідеї.
- Python – це серверна мова, яку можна використовувати для програмування вебсайтів. В цій мові є безліч бібліотек та динамічна типізація, що може призвести до помилок, через те, що код пишеться без оголошення типів змінних.
- SQL – це мова запитів, призначена для роботи з СУБД. Ця мова є основною для роботи з БД [3].

1.3 Аналіз вебсайтів прокату настільних ігор

D20.in.ua - це вебсайт, через який можливо придбати або взяти у прокат гру чи додати її у бажання, щоб потім повернутись до неї. Головна сторінка прокату ігор вебсайту представлена на рисунку 1.1. На сайті можна здійснити покупку або прокат гри, ігри можна сортувати та шукати за їх типами та кількістю гравців за допомогою критеріїв, які можна обрати у списку зліва. Опис гри та її компонентів на сайті вказано не для всіх ігор, а правила відсутні, сума прокату вказана за тиждень, що представлено на рисунку 1.2, кожна

сторінка з грою містить правила її прокату, які представлено на рисунку 1.3 [4].

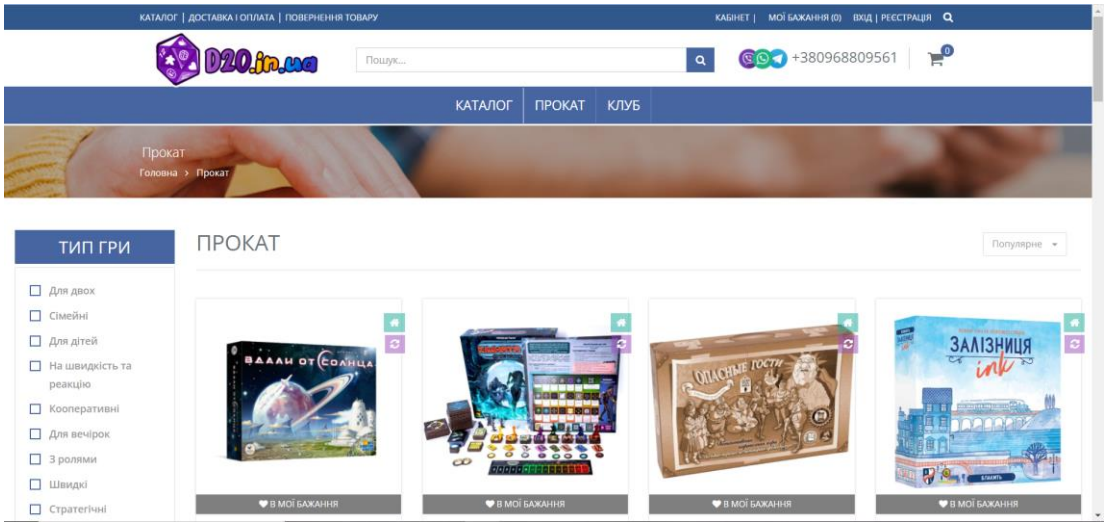

Рисунок 1.1 – Головна сторінка прокату ігор вебсайту D20.in.ua

Рисунок 1.2 – Сторінка прокату гри вебсайту D20.in.ua

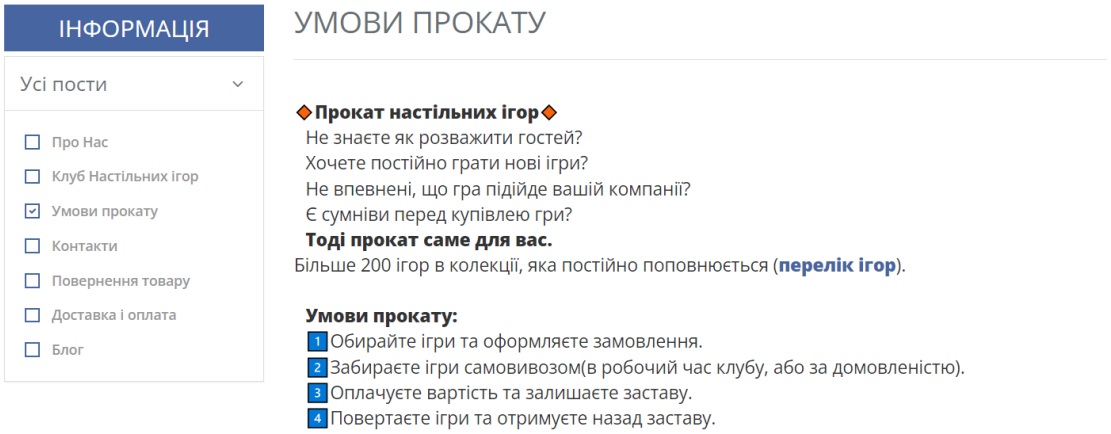

Рисунок 1.3 – Правила прокату ігор вебсайту D20.in.ua

На відміну від вебсайту D20.in.ua, розроблюваний вебсайт має надавати можливість додавання гри в прокат на будь-який термін, а оплата здійснюється по годинах користування. Опис компонентів гри та її правила можна буде вказати для всіх ігор.

1.4 Постановка задачі на розробку програмного забезпечення вебсайту для клубу прокату настільних ігор

Зробивши аналіз предметної області було поставлено питання про розробку програмного забезпечення вебсайту для клубу прокату настільних ігор для автоматизації процесу ведення обліку прокату настільних ігор. Розроблюваний вебсайт має сприяти відмові від паперових документів та журналів з записами про прокат ігор.

1) Вимоги до функціональних характеристик:

Створити вебсайт для «Клубу прокату настільних ігор» з такими можливостями:

- Додавання співробітника;
- Авторизація;
- Вихід з облікового запису користувача;
- Виведення даних про конкретну гру;
- Можливість додавання, видалення та редагування даних;
- Зміна статусу гри;
- Адміністратор зможе мати доступ до вебсайту, бачити всі зайняті та вільні ігри для прокату, данні про співробітників, редагувати всі дані, ставити позначки, заповнювати поля даними або видаляти їх, наймати та звільняти співробітників, створюючи чи видаляючи з бази даних їх облікові записи для користування вебсайтом;
- Співробітник зможе мати доступ до вебсайту, бачити всі зайняті та вільні ігри для прокату, редагувати, додавати або видаляти ігри з таблиці, брати заставну вартість з відвідувачів, здійснювати

розрахунок готівкою, видавати в прокат ігри та приймати їх назад, а також змінювати статус гри, що дасть відвідувачам можливість бачити чи вільна для прокату гра;

- Відвідувач зможе лише бачити всі зайняті та вільні ігри та обирати зі списку вільних ту, яку він хотів би взяти в прокат.

2) Вимоги до вхідних та вихідних даних:

Вхідними даними є ПІБ, фотографія, логін, електронна пошта, посада та пароль користувача розроблюваного вебсайту, який є співробітником клубу настільних ігор, також інформація про ігри у вигляді назви, вартості прокату за годину, опису складу та правил гри та її логічний статус, що дає зрозуміти чи вільна гра для прокату в цей момент, інформація про відвідувача у вигляді взятої у прокат гри, розміру заставної вартості, часу прокату гри та прізвища відвідувача та співробітника, який його обслуговував, інформація про розрахунок готівкою у вигляді дати, id гри та id співробітника, який проводив розрахунок. Введення текстової інформації буде здійснюватися за допомогою комп’ютерної клавіатури. Завантаження зображення буде здійснюватися через вибір вже наявного зображення у пам’яті комп’ютера.

Вихідними даними є запис у базі даних про створення облікового запису для користувача, що є співробітником клубу настільних ігор та показ у вебсайті інформації про ігри, що зберігається у базі даних. Вихідні дані будуть мати такий вигляд:

- Список ігор на головній сторінці вебсайту;
- Завантажена раніше фотографія співробітника, як складова його облікового запису з зображенням;
- Наявність облікового запису користувача у вебсайті та базі даних та можливість авторизації та виходу з облікового запису.

3) Вимоги до параметрів технічних засобів:

- Процесор Intel Pentium/Celeron 1800 МГц і вище;

- Оперативна пам'ять 1 Гбайт і вище;
- Жорсткий диск близько 1 Гбайт(система + дані);

4) Вимоги щодо інформаційної та програмної сумісності:

Вебсайт розроблюється для браузерів: Internet Explorer 10.0 та вище, або Firefox 3.5 та вище, або Opera 9.5 та вище, або Safari 3.2.1 та вище, або Chrome 2 та вище

2 ПРОЄКТ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ВЕБСАЙТУ ДЛЯ КЛУБУ ПРОКАТУ НАСТІЛЬНИХ ІГОР

2.1 Ескізний проєкт програмного забезпечення програмного забезпечення вебсайту для клубу прокату настільних ігор

Після проведення аналізу вимог до програмного забезпечення було розроблено ескізний проєкт за допомогою мови моделювання UML. Для визначення та аналізу вимог було розроблено діаграму варіантів використання та виконано специфікацію варіантів використання.

Для формалізації представлення варіантів використання було побудовано діаграми діяльності для кожного варіанту використання, розроблена інформаційна модель та інтерфейс користувача.

2.1.1 Побудова контекстної діаграми програмного забезпечення вебсайту для клубу прокату настільних ігор

На рисунку 2.1 представлено контекстну діаграму програмного забезпечення вебсайту для клубу прокату настільних ігор.

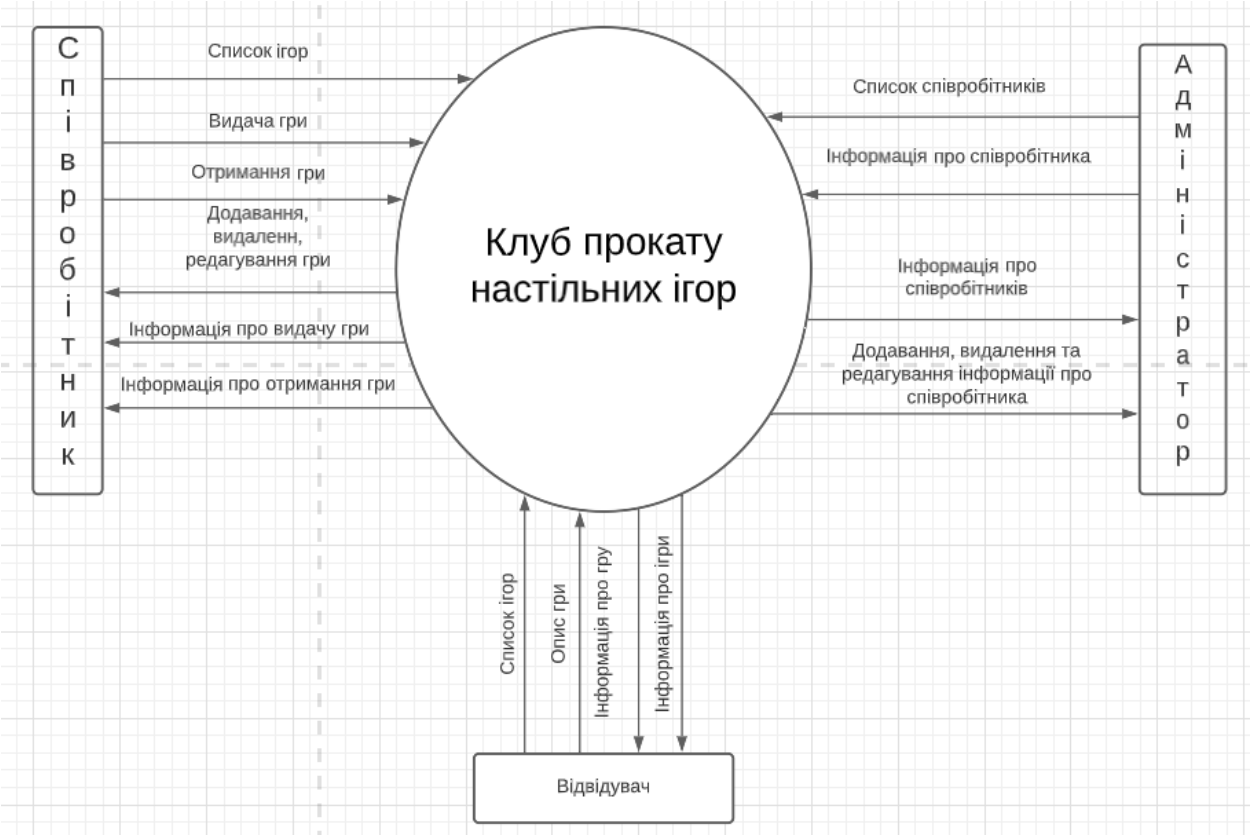

Рисунок 2.1 – Контекстна діаграма

Словник даних та потоки даних, що відповідають побудованій діаграмі представлено у таблиці 2.1

Таблиця 2.1 – Словник даних і потоки даних контекстної діаграми

Найменування потоку	Опис потоку
Список ігор	Перелік усіх існуючих у базі даних ігор
Видача гри	Id, дата, заставна вартість
Отримання гри	Id, дата, наявність пошкоджень гри
Додавання, видалення, редагування гри	Id, id співробітника, назва, опис, вартість прокату за годину, логічний статус
Інформація про видачу гри	Id співробітника, дата, заставна вартість
Інформація про отримання гри за прокату	Id співробітника, дата, стан гри

Продовження таблиці 2.1

Список співробітників	Таблиця зі співробітниками
Інформація про співробітника	Id, ПІБ, електронна пошта, фото, логін, пароль, посада
Інформація про співробітників	Таблиця зі співробітниками
Додавання, видалення, редагування інформації про співробітника	Id, ПІБ, електронна пошта, фото, логін, пароль, посада
Опис гри	Правила та склад компонентів гри
Інформація про гру	Id, id співробітника, назва, опис, вартість прокату за годину, логічний статус
Інформація про ігри	Список ігор

2.1.2 Побудова моделі варіантів використання

Проаналізувавши функціональні вимоги до програмного забезпечення було виділено основні варіанти використання:

- Авторизація;
- Вихід з облікового запису
- Додавання співробітника;
- Перегляд списку співробітників;
- Редагування списку співробітників;
- Додавання гри;
- Перегляд списку ігор;
- Редагування інформації про гру;
- Видалення гри;
- Перегляд інформації про гру;
- Видача гри;
- Розрахунок готівкою;
- Отримання гри з прокату;

На рисунку 2.2 наведено діаграму варіантів використання.

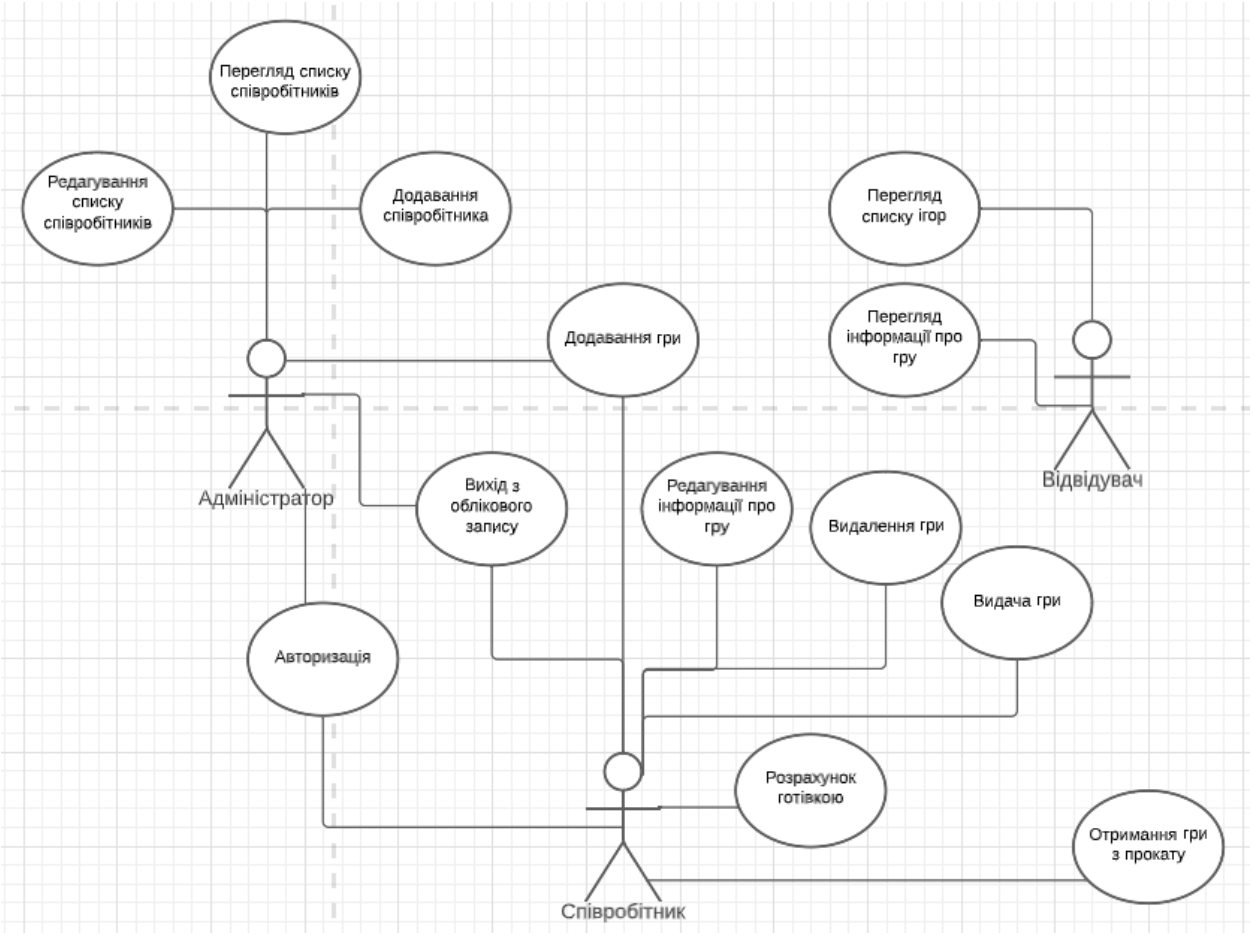

Рисунок 2.2 – Діаграма варіантів використання програмного забезпечення вебсайту для клубу прокату настільних ігор

2.1.3 Специфікації варіантів використання

Проаналізувавши діаграму варіантів використання, складено специфікації для кожного з варіантів використання, які представлено нижче.

Специфікація варіанту використання «Додавання співробітника»

Найменування варіанта використання: Додавання співробітника

Короткий опис: Цей варіант використання дозволяє адміністратору створити обліковий запис для співробітника у системі.

Актори: Головний актор – Адміністратор.

Основний потік

1) Адміністратор обирає функцію додати співробітника.
2) Система відкриває форму для внесення даних про співробітника.
3) Адміністратор вводить дані про співробітника в форму та обирає функцію.
4) Система записує дані про співробітника до бази даних.
5) Система виводить повідомлення про успішне додавання співробітника.

Альтернативний потік

1. Будь-яке з полів введення інформації про співробітника залишилось пустим: система повідомляє про обов'язковість заповнення всіх полів та пропонує заповнити або завершити варіант використання.

Спеціальні вимоги: спеціальні вимоги не виявлено.

Передумови: передумови не виявлено

Постумови: постумови не виявлено.

Додаткові зауваження: додаткових зауважень немає.

Специфікація варіанту використання «Авторизація»

Найменування варіанта використання: Авторизація

Короткий опис: Цей варіант використання дозволяє співробітнику увійти у свій обліковий запис у системі.

Актори: Головний актор – Співробітник.

Основний потік

1) Співробітник обирає функцію авторизації.
2) Система виводить на екран форму авторизації.
3) Співробітник вводить у поля логін та пароль та обирає функцію авторизуватись.
4) Система відкриває головну сторінку вебсайту.

Альтернативний потік:

1. Якесь з полів логіну та пароля залишилось пустим або введено неправильно: система виводить повідомлення про помилку та

повідомляє про обов’язковість заповнення всіх полів та пропонує заповнити або завершити варіант використання..

Спеціальні вимоги: спеціальні вимоги не виявлено.

Передумови:

Перед активізацією варіанта використання повинен бути виконаний потік варіанта «Додавання співробітника».

Постумови: постумови не виявлено.

Додаткові зауваження: додаткових зауважень немає.

Специфікація варіанту використання «Додавання гри»

Найменування варіанта використання: Додавання гри.

Короткий опис: Цей варіант використання дозволяє адміністратору додати гру.

Актори: Головний актор – Адміністратор, другорядний актор - Співробітник.

Основний потік

1) Адміністратор обирає функцію авторизації.
2) Система виводить на екран форму авторизації.
3) Адміністратор вводить у поля логін та пароль та обирає функцію авторизуватись.
4) Система відкриває головну сторінку вебсайту.
5) Адміністратор обирає функцію додати гру.
6) Система відкриває форму для внесення даних для опису гру.
7) Адміністратор вводить дані в форму для опису гри та обирає функцію збереження гри.
8) Система записує гру до бази даних та виводить її у списку з іграми на головній сторінці.

Альтернативний потік

1. Будь-яке з полів введення інформації про гру залишилось пустим: система повідомляє про обов’язковість заповнення всіх полів та пропонує заповнити або завершити варіант використання.

Спеціальні вимоги: спеціальні вимоги не виявлено.

Передумови: передумови не виявлено

Постумови: постумови не виявлено.

Додаткові зауваження: додаткових зауважень немає.

Специфікація варіанту використання «Редагування інформації про гру»

Найменування варіанта використання: Редагувати інформацію про гру.

Короткий опис: Цей варіант використання дозволяє співробітнику редагувати інформацію про гру.

Актори: Головний актор – Співробітник.

Основний потік

1) Співробітник обирає функцію авторизації.
2) Система виводить на екран форму авторизації.
3) Співробітник вводить у поля логін та пароль та обирає функцію авторизуватись.
4) Система відкриває головну сторінку вебсайту.
5) Співробітник обирає функцію редагувати інформацію про гру гру.
6) Система відкриває форму для редагування даних про гру.
7) Співробітник вводить дані в форму для редагування гри та обирає функцію збереження гри.
8) Система записує зміни до бази даних та виводить їх у списку з іграми на головній сторінці.

Альтернативний потік: альтернативний потік відсутній.

Спеціальні вимоги: спеціальні вимоги не виявлено.

Передумови: передумови не виявлено

Постумови: постумови не виявлено.

Додаткові зауваження: додаткових зауважень немає.

Специфікація варіанту використання «Вихід з облікового запису»

Найменування варіанта використання: Вихід з облікового запису.

Короткий опис: Цей варіант використання дозволяє співробітнику вийти з облікового запису.

Актори: Головний актор – Співробітник.

Основний потік

1) Співробітник обирає функцію авторизації.
2) Система виводить на екран форму авторизації.
3) Співробітник вводить у поля логін та пароль та обирає функцію авторизуватись.
4) Система відкриває головну сторінку вебсайту.
5) Співробітник обирає функцію перейти в обліковий запис.
6) Система відкриває обліковий запис співробітника.
7) Співробітник обирає функцію вийти з облікового запису.
8) Система відкриває сторінку авторизації.

Альтернативний потік: альтернативний потік відсутній.

Спеціальні вимоги: спеціальні вимоги не виявлено.

Передумови: передумови не виявлено

Постумови: постумови не виявлено.

Додаткові зауваження: додаткових зауважень немає.

Специфікація варіанту використання «Отримання гри з прокату»

Найменування варіанта використання: Отримання гри з прокату

Короткий опис: Цей варіант використання дозволяє співробітнику повернути гру в прокат після її повернення.

Актори: Головний актор – Співробітник.

Основний потік

1) Співробітник обирає функцію авторизації.
2) Система виводить на екран форму авторизації.

3) Співробітник вводить у поля логін та пароль та обирає функцію авторизуватись.
4) Система відкриває головну сторінку вебсайту.
5) Співробітник обирає функцію перегляду складу взятої у прокат гри.
6) Система відкриває вебсайт з повним описом складу гри.
7) Співробітник зрівнює склад повернутої гри з вказаним на вебсайті.
8) Співробітник обирає функцію повернутися.
9) Система робить перехід на головну сторінку вебсайту.
10) Співробітник змінює логічний статус гри.

Співробітник повертає заставну вартість гри та здійснює розрахунок готівкою за користування грою.

11) Система виконує збереження даних.

Альтернативний потік:

1. Пошкодження гри: при пошкодженні гри співробітник не повертає заставну вартість.

Спеціальні вимоги: спеціальні вимоги не виявлено.

Передумови:

Перед активізацією варіанта використання повинен бути виконаний потік варіанта «Видача гри».

Постумови: постумови не виявлено.

Додаткові зауваження: додаткових зауважень немає.

Специфікація варіанту використання «Видалення гри»

Найменування варіанта використання: Видалення гри.

Короткий опис: Цей варіант використання дозволяє співробітнику видалити зі списку ігор будь-яку позицію.

Актори: Головний актор – Співробітник.

Основний потік

1) Співробітник обирає функцію авторизації.
2) Система виводить на екран форму авторизації.

3) Співробітник вводить у поля логін та пароль та обирає функцію авторизуватись.
4) Система відкриває головну сторінку вебсайту.
5) Співробітник обирає функцію видалити гру на головній сторінці вебсайту.
6) Система видаляє обрану співробітником гру з таблиці з іграми.

Альтернативний потік: альтернативний потік відсутній.

Спеціальні вимоги: спеціальні вимоги не виявлено.

Передумови: передумови не виявлено

Постумови: постумови не виявлено.

Додаткові зауваження: додаткових зауважень немає.

Специфікація варіанту використання «Перегляд списку ігор»

Найменування варіанта використання: Перегляд списку ігор.

Короткий опис: Цей варіант використання дозволяє відвідувачу ознайомитись зі списком ігор на головній сторінці вебсайту.

Актори: Головний актор – Відвідувач.

Основний потік

1) Відвідувач переходить на головну сторінку вебсайту.
2) Система виводить на екран таблицю з іграми.
3) Відвідувач обирає функцію перегляду вільних для прокату ігор.
4) Система видає список вільних для прокату ігор.
5) Відвідувач обирає бажану для прокату гру та повідомляє про це співробітнику.
6) Система видаляє обрану співробітником гру з таблиці з іграми.

Альтернативний потік: альтернативний потік відсутній.

Спеціальні вимоги: спеціальні вимоги не виявлено.

Передумови: передумови не виявлено

Постумови: постумови не виявлено.

Додаткові зауваження: додаткових зауважень немає.

Специфікація варіанту використання «Перегляд інформації про гру»

Найменування варіанта використання: Перегляд інформації про гру.

Короткий опис: Цей варіант використання дозволяє відвідувачу отримати інформацію про конкретну гру.

Актори: Головний актор – Відвідувач.

Основний потік

1) Відвідувач переходить на головну сторінку вебсайту.
2) Система виводить на екран таблицю з іграми.
3) Відвідувач обирає функцію перегляду інформації про конкретну гру.
4) Система відкриває вебсторінку з описом конкретної гри.

Альтернативний потік: альтернативний потік відсутній.

Спеціальні вимоги: спеціальні вимоги не виявлено.

Передумови: передумови не виявлено

Постумови: постумови не виявлено.

Додаткові зауваження: додаткових зауважень немає.

Специфікація варіанту використання «Видача гри»

Найменування варіанта використання: Видача гри.

Короткий опис: Цей варіант використання дозволяє співробітнику видати у прокат обрану відвідувачем гру.

Актори: Головний актор – Співробітник.

Основний потік

1) Співробітник обирає функцію авторизації.
2) Система виводить на екран форму авторизації.
3) Співробітник вводить у поля логін та пароль та обирає функцію авторизуватись.
4) Система відкриває головну сторінку вебсайту.
5) Співробітник обирає названу відвідувачем гру, міняє її логічний статус з вільної на зайняту.

6) Система зберігає зміни.

Альтернативний потік:

1. Обрана відвідувачем гра зайнята для прокату: співробітник повідомляє про це відвідувача та пропонує почекати, дока звільниться гра або обрати нову зі списку вільних для прокату ігор.

Спеціальні вимоги: спеціальні вимоги не виявлено.

Передумови: передумови не виявлено

Постумови: постумови не виявлено.

Додаткові зауваження: додаткових зауважень немає.

Специфікація варіанту використання «Розрахунок готівкою»

Найменування варіанта використання: Розрахунок готівкою.

Короткий опис: Цей варіант використання дозволяє співробітнику видати у прокат обрану відвідувачем гру.

Актори: Головний актор – Співробітник.

Основний потік

1) Співробітник обирає функцію авторизації.
2) Система виводить на екран форму авторизації.
3) Співробітник вводить у поля логін та пароль та обирає функцію авторизуватись.
4) Система відкриває головну сторінку вебсайту.
5) Співробітник обирає функцію розрахунок готівкою.
6) Система видає форму для розрахунку готівкою.
7) Співробітник вводить всі необхідні дані, обирає функцію зберегти та повертає відвідувачу залишену ним заставну вартість.
8) Система зберігає зміни

Альтернативний потік:

1. Будь-яке з полів введення інформації форми розрахунку готівкою залишилось пустим: система повідомляє про обов’язковість

заповнення всіх полів та пропонує заповнити або завершити варіант використання.

Спеціальні вимоги: спеціальні вимоги не виявлено.

Передумови:

Перед активізацією варіанта використання повинен бути виконаний потік варіанта «Отримання гри з прокату».

Постумови: постумови не виявлено.

Додаткові зауваження: додаткових зауважень немає.

Специфікація варіанту використання «Перегляд списку співробітників»

Найменування варіанта використання: Перегляд списку співробітників.

Короткий опис: Цей варіант використання дозволяє адміністратору переглядати список співробітників.

Актори: Головний актор – Адміністратор.

Основний потік

1) Адміністратор обирає заходить до бази даних.
2) Система виводить на екран таблиці бази даних.
3) Адміністратор обирає функцію перейти до таблиці зі співробітниками.
4) Система відкриває таблицю зі співробітниками.

Альтернативний потік: альтернативний потік відсутній.

Спеціальні вимоги: спеціальні вимоги не виявлено.

Передумови: передумови не виявлено.

Постумови: постумови не виявлено.

Додаткові зауваження: додаткових зауважень немає.

Специфікація варіанту використання «Редагування списку співробітників»

Найменування варіанта використання: Редагування списку співробітників.

Короткий опис: Цей варіант використання дозволяє адміністратору редагувати список співробітників.

Актори: Головний актор – Адміністратор.

Основний потік

1) Адміністратор обирає заходить до бази даних.
2) Система виводить на екран таблиці бази даних.
3) Адміністратор обирає функцію перейти до таблиці зі співробітниками.
4) Система відкриває таблицю зі співробітниками.
5) Адміністратор обирає функцію редагування даних про співробітника.
6) Система відкриває форму з даними про співробітника.
7) Адміністратор змінює дані та натискає кнопку збереження.
8) Система зберігає дані.

Альтернативний потік:

1. У полі редагування інформації про співробітника є некоректний тип даних: некоректний тип даних та пропонує заповнити або завершити варіант використання.

Спеціальні вимоги: спеціальні вимоги не виявлено.

Передумови: передумови відсутні.

Постумови: постумови не виявлено.

Додаткові зауваження: додаткових зауважень немає.

2.1.4 Сценарії варіантів використання

На цій стадії процесу розробки програмного забезпечення вебсайту для клубу прокату настільних ігор створюються діаграми діяльності. Доцільно створити діаграми діяльності для основних варіантів використання.

На рисунку 2.3 наведено діаграму діяльності для варіанта використання «Авторизація»

Рисунок 2.3 – Сценарій варіанту використання «Авторизація»

На рисунку 2.4 наведено діаграму діяльності для варіанту використання «Додавання гри»

Рисунок 2.4 – Сценарій варіанту використання «Додавання гри»

На рисунку 2.5 наведено діаграму діяльності для варіанту використання для варіанту використання «Отримання гри з прокату»

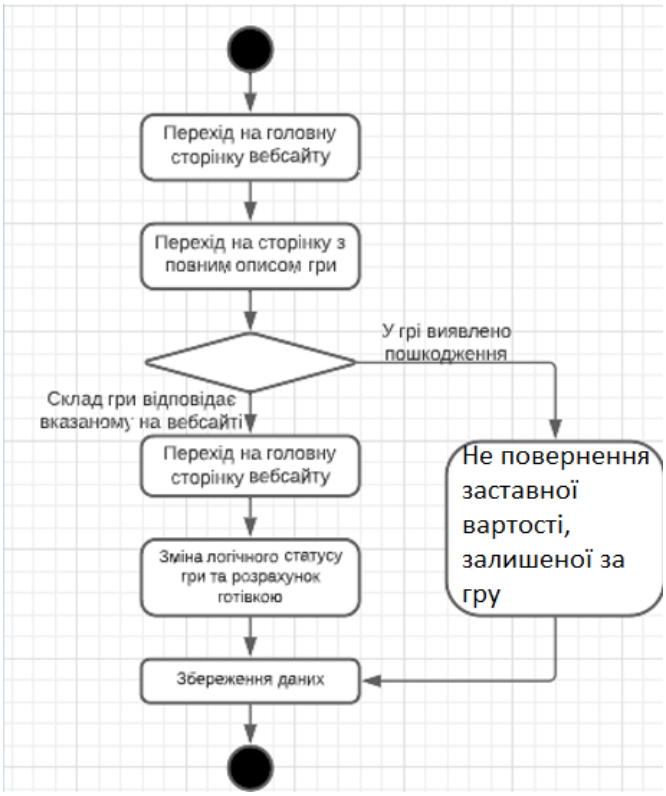

Рисунок 2.5 – Сценарій варіанту використання «Забрати гру»

2.1.5 Розробка інтерфейсу програмного забезпечення

У наступному розділі приведено діалоги та екранні форми для проєктування вебсайту. Інтерфейс – це зовнішня оболонка програми, яка являє собою систему засобів і правил та забезпечує взаємодію користувача з програмою.

При відкритті вебсайту користувачу пропонується авторизація з полями вводу логіну та паролю, яке наведено на рисунку 2.6.

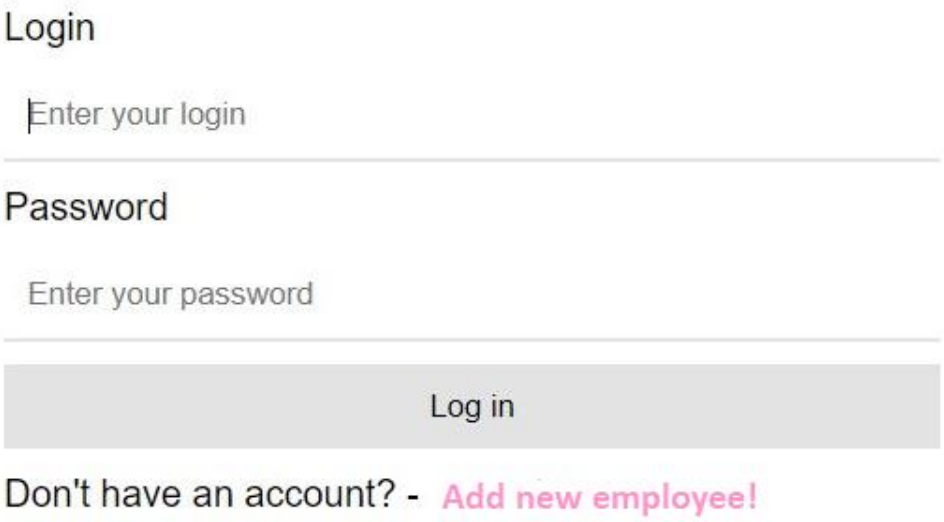

Рисунок 2.6 – Вікно авторизації

При відсутності облікового запису користувача у вебсайті – пропонується здійснити додавання співробітника, форму якої приведено на рисунку 2.7.

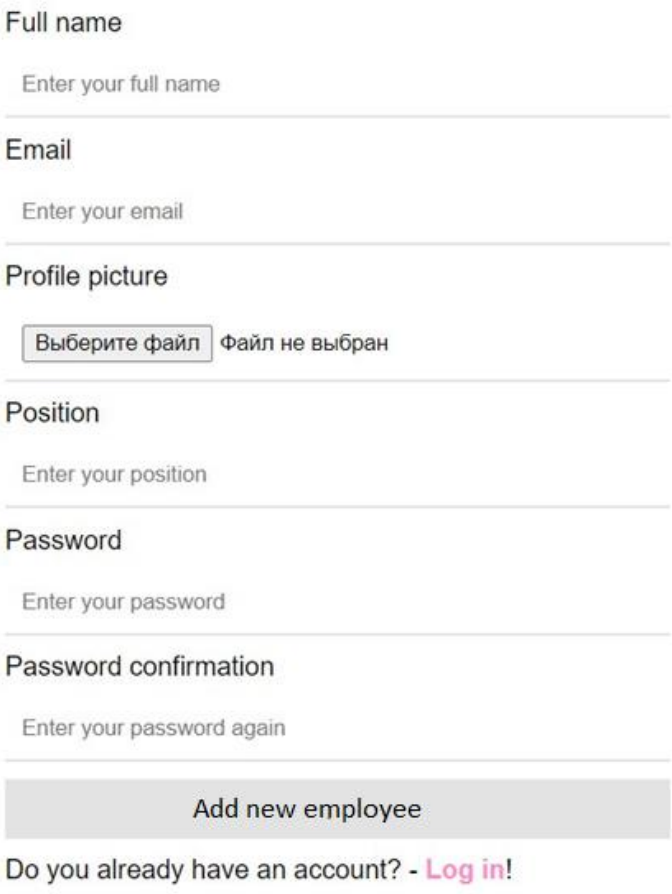

Рисунок 2.7 – Форма додавання співробітника

Після проходження додавання співробітника у вебсайті створюється його обліковий запис, який приведено на рисунку 2.8.

Рисунок 2.8 – Обліковий запис співробітника

Після проходження додавання співробітника та авторизації відкривається головна сторінка вебсайту, що містить таблицю з іграми, яка представлена на рисунку 2.9.

1	Шашки	75$	Игровая доска, 12 белых и 12 чёрных шашек.	1	View	Update	Delete
2	Твистер	50$	Игровое поле, рулетка	0	View	Update	Delete
3	Пазл	100$	1000 пазлов	1	View	Update	Delete
4	Шахматы	150$	В шахматах существует шесть разных (видов или наименований) фигур — король, ферзь, ладья, слон, конь и пешка. В шахматы играют два соперника: один играет белыми фигурами, другой чёрными. У каждого игрока 16 фигур — один король, один ферзь, две ладьи, два слона, два коня и восемь пешек.	0	View	Update	Delete
5	Лото	120$	Игровые картонные карточки 24 штуки, на каждой карточке имеется три ряда клеток, в каждом ряду в произвольном порядке расположены по пять цифр от 1 до 90; - пластиковые или картонные фишки (маркеры) для закрывания клеток с цифрами; - непрозрачный мешочек для хранения деревянных бочонков.	1	View	Update	Delete
6	Дженга	135$	54 Деревянных блока. Рукав для строительства башни.	0	View	Update	Delete
7	Уно	75$	108 карт	1	View	Update	Delete
8	dnjferjgnrt	15$	ertvbyrnun	0	View	Update	Delete
19	Мафия	78$	18 карт	1	View	Update	Delete

Рисунок 2.9 – Головна сторінка вебсайту

2.2 Технічний проєкт програмного забезпечення вебсайту для клубу прокату настільних ігор

Проаналізувавши ескізний проєкт, за допомогою побудови динамічної та статичної моделей було розроблено технічний проєкт програмного забезпечення.

2.2.1 Логічна модель даних

На рисунку 2.10 представлено логічну модель даних програмного забезпечення для клубу прокату настільних ігор.

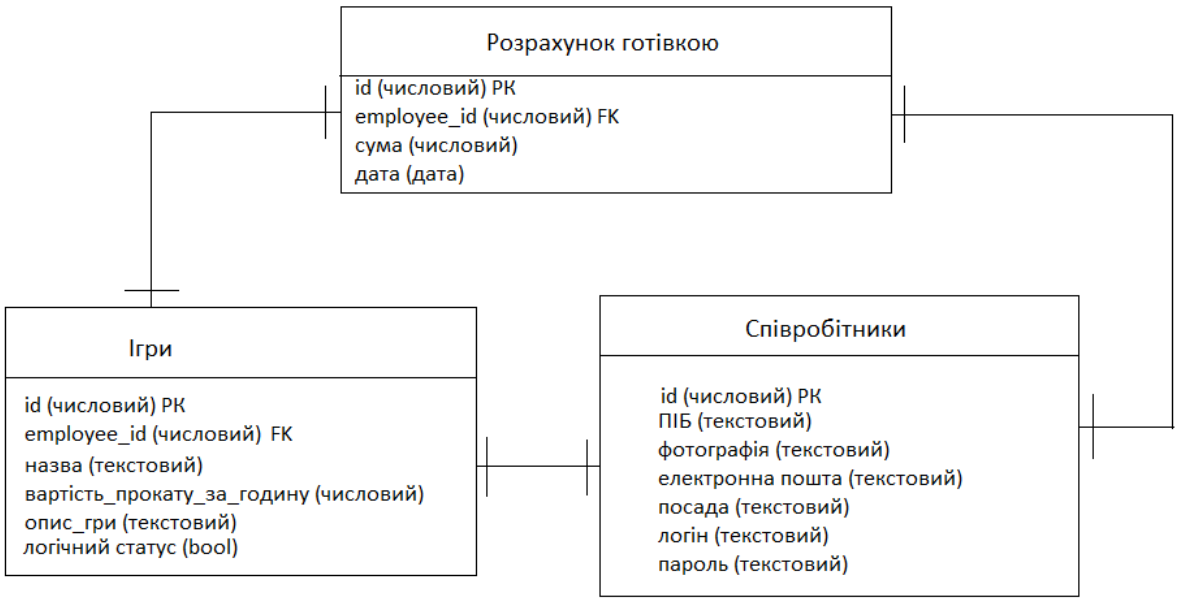

Рисунок 2.10 – Логічна модель даних

2.2.2 Структура класів

На основі моделі варіантів використання, інформаційної моделі та моделі програмного інтерфейсу було розроблено статичну модель, яка представлена діаграмою класів та приведена на рисунку 2.11:

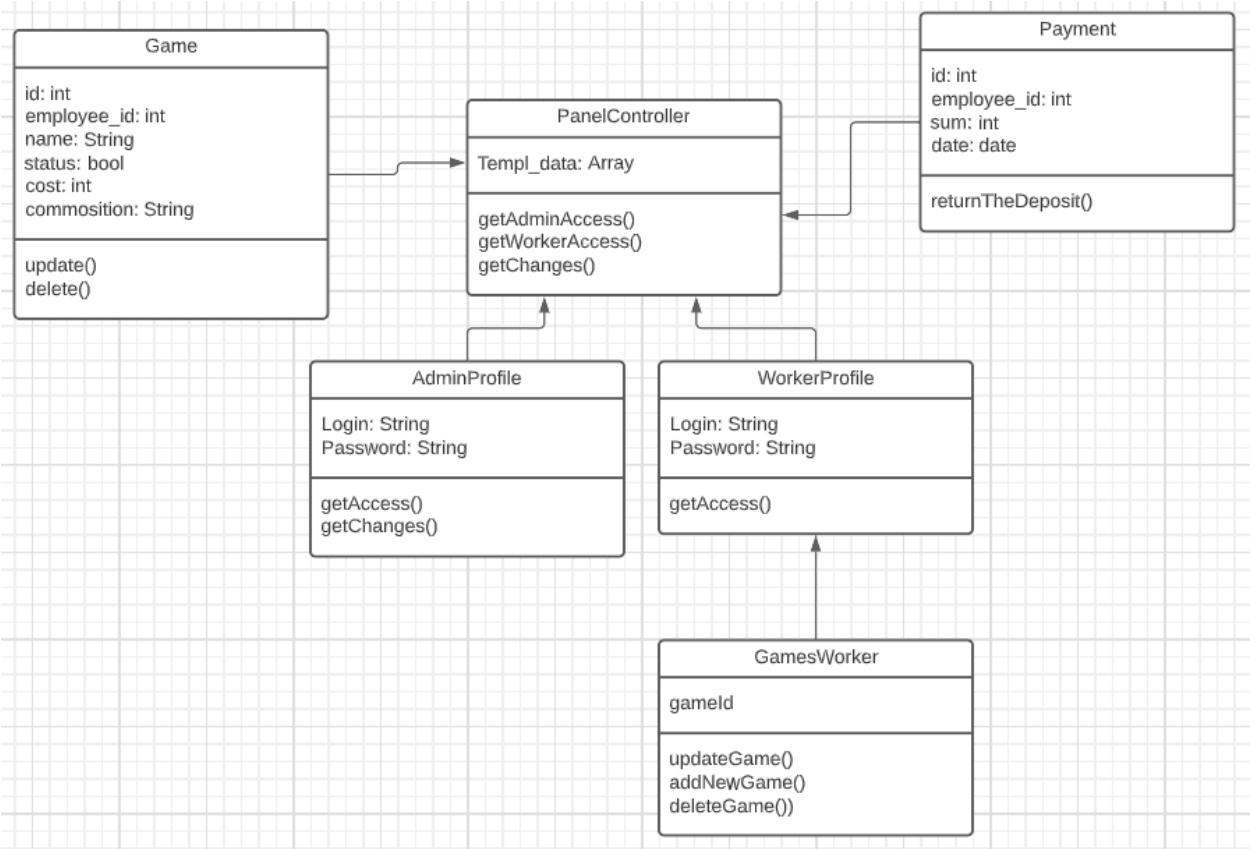

Рисунок 2.11 – Діаграма класів

2.2.3 Специфікація класів

Клас: «PanelController»

Відповідальність: Даний клас відповідає за авторизацію в системі.

Атрибути: Templ_Data

Операції зі специфікаціями нетривіальних операцій:

Нижче представлені специфікації класів програми

getWorkerAccess() – функція отримання доступу до вебсайту с правами авторизованого співробітника;

getAdminAccess() – функція отримання доступу до вебсайту з правами адміністратора;

getChanges() – функція збереження змін у БД облікових записів при додаванні співробітника;

Клас: «GameWorker»

Відповідальність: Даний клас відповідає за взаємодію співробітника зі списком ігор.

Атрибути: gameId.

Операції зі специфікаціями нетривіальних операцій:

updateGame() – функція оновлення опису гри;

addNewGame() – функція додання нової гри в таблицю;

deleteGame() – видалення будь-якої гри з таблиці;

Клас: «AdminProfile»

Відповідальність: Даний клас відповідає за взаємодію адміністратора з обліковим записом.

Атрибути: Login, Password.

Операції зі специфікаціями нетривіальних операцій:

getAccess() – функція отримання доступу;

getChanges() – функція здійснення будь-яких змін;

Клас: «WorkerProfile»

Відповідальність: Даний клас відповідає за взаємодію співробітника зі списком ігор.

Атрибути: Login, Password.

Операції зі специфікаціями нетривіальних операцій:

getAccess() – функція отримання доступу;

2.2.4 Динамічна модель програмного забезпечення

На основі діаграми варіантів використання та побудованої статичної моделі розробимо динамічну модель програмного забезпечення.

Спочатку користувач повинен пройти додавання облікового запису співробітника та авторизацію, після чого він потрапляє на головну сторінку

вебсайту, де може додавати, редагувати або видаляти ігри, здавати їх в прокат, оформляти розрахунок готівкою.

На рисунку 2.12 наведена діаграма послідовності, яке демонструє виконання варіанту використання «Отримати гру з прокату».

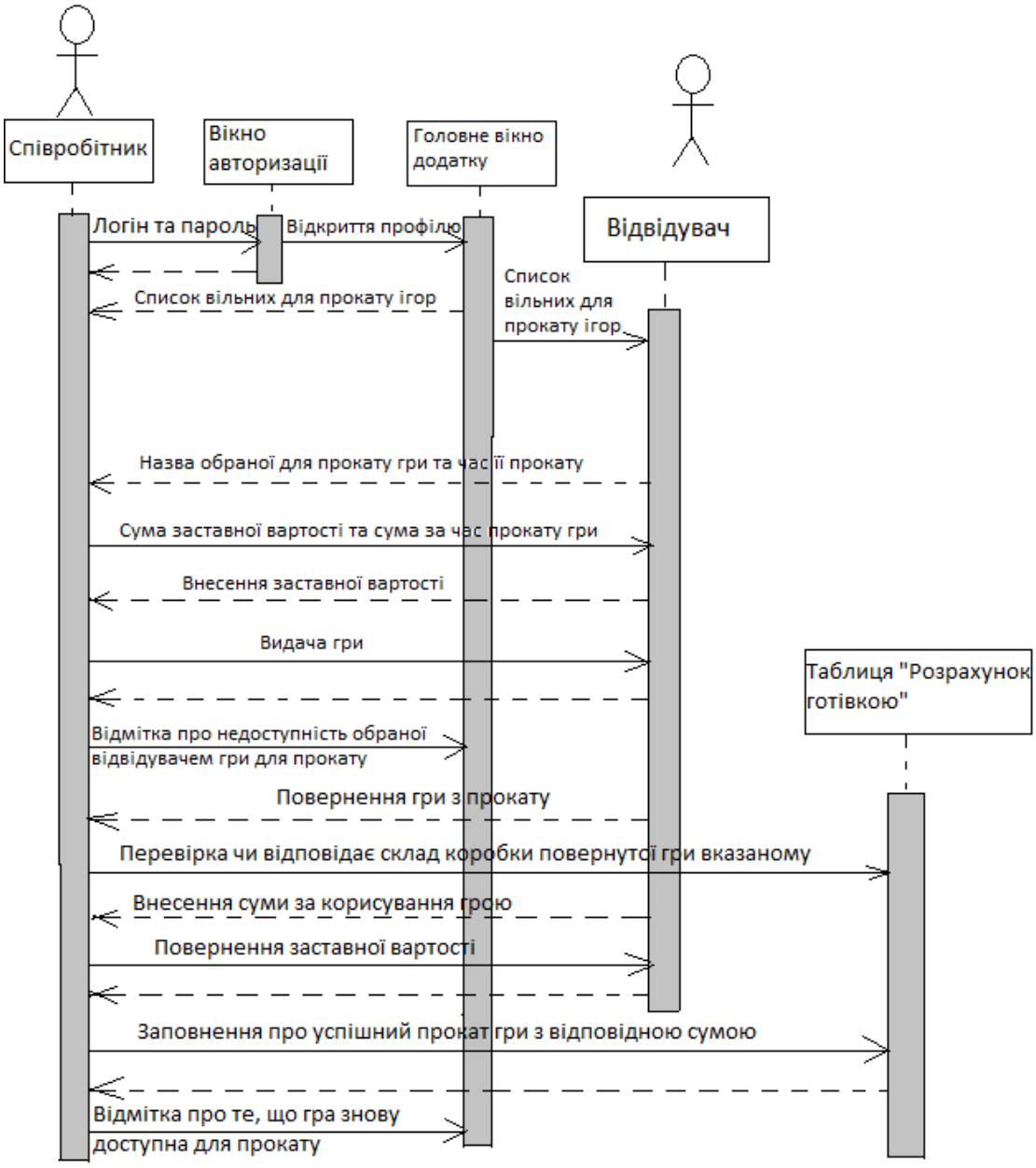

Рисунок 2.12 - Виконання варіанту використання «Отримати гру з прокату»

2.3 Робочий проєкт програмного забезпечення вебсайту для клубу прокату настільних ігор

2.3.1 Фізична модель даних програмного забезпечення вебсайту для клубу прокату настільних ігор

Фізичну модель даних представлено на рисунку 2.13.

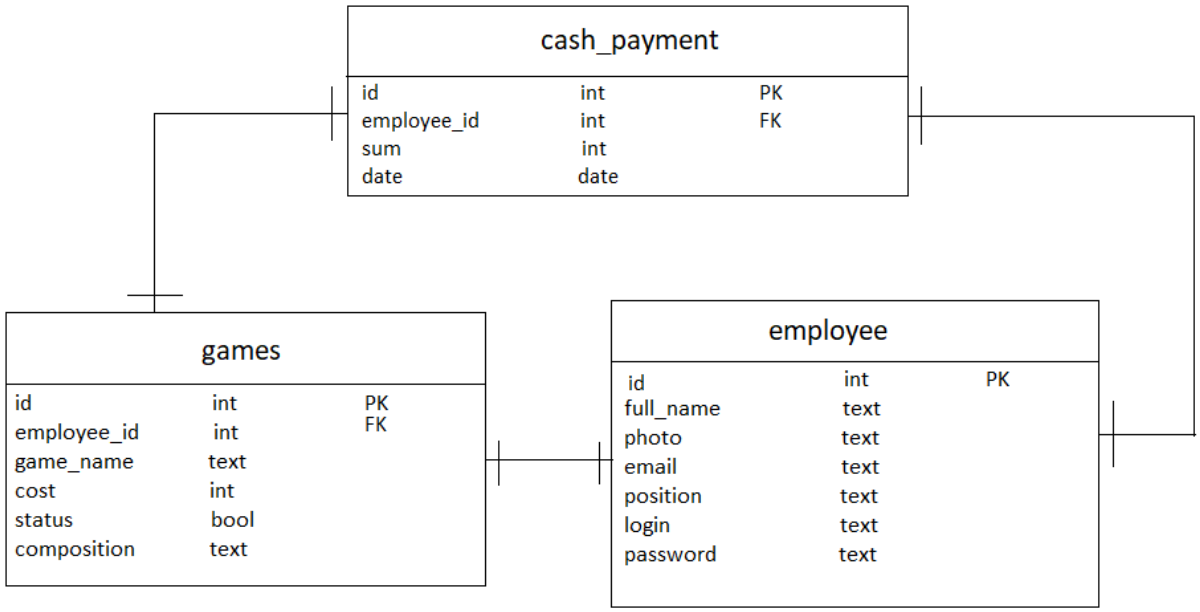

Рисунок 2.13 – Фізична модель даних

Фізична модель даних представлена у наведених у цьому розділі таблицях 2.2 – 2.4. Кожна таблиця містить назву полів, типи даних, довжину, формат і ключі. Поля та їх типи зображені згідно обраного середовища розробки програми.

Таблиця 2.2 – Структура таблиці cash_payment (Розрахунок готівкою)

№ п/п	Назва поля	Тип даних	Опис	Ключ
1	id	INT	Унікальний номер гри	PK
2	employee_id	INT	Унікальний номер співробітника	FK
3	date	DATE	Дата здійснення розрахунку	
	sun	INT	Сума за час користування грою	

Таблиця 2.3 – Структура таблиці games (Ігри)

№ з/п	Назва поля	Тип даних	Опис	Ключ
1	id	INT	Унікальний номер гри	PK
2	employee_id	INT	Унікальний номер співробітника	AK
2	game_name	TEXT	Назва гри	
3	cost	INT	Вартість прокату гри за годину	
4	status	BOOL	Логічний статус гри	
5	composition	TEXT	Опис компонентів гри	

Таблиця 2.4 – Структура таблиці employee (Співробітники)

№ з/п	Назва поля	Тип даних	Опис	Ключ
1	employee_id	INT	Унікальний номер співробітника	PK
2	full_name	TEXT	Повне ім’я співробітника	
3	photo	TEXT	Посилання на фото співробітника	
4	email	TEXT	Електронна пошта співробітника	
5	position	TEXT	Посада співробітника	
6	login	TEXT	Логін від облікового запису співробітника	
7	password	TEXT	Пароль від облікового запису співробітника	

2.3.2 Обґрунтування вибору мови та системи програмування

В ході розробки програмного забезпечення вебсайту для клубу прокату настільних ігор використовувались мови програмування PHP, HTML [7], CSS [8], JavaScript з використанням середовища розробки PhpStorm. Підставами для вибору цього середовища стали:

- Автодоповнення та рефакторинг коду;
- Вбудована інтеграція з різними системами контролю версій;
- Можливість управління базами даних SQL;
- Ефективні функції навігації;
- Візуальний відладчик, який дозволяє показувати зміни в браузері.[5]

Підстави для вибору PHP:

- Орієнтація на веброзробку – більшість конструкцій та рішень у ньому створені для зручності в у вебсередовищі;

- Використання цієї мови не перекриває для розробника використання інших технологій;
- Знання, прийоми роботи PHP знадобляться і в іншій системі, відмінній від PHP [6].

Переваги для вибору JavaScript:

- Ні один браузер не обходиться без підтримки JavaScript;
- Постійне удосконалення мови;
- JavaScript відмінно працює з іншими мовами та може використовуватись у різних додатках [9].

У якості СУБД було обрано MySQL, яка є оптимальним варіантом, що застосовується для створення вебсайтів різної складності зі зберіганням та роботою з даними.

Важливим фактором є її безкоштовність. MySQL поширюється за умов загальної ліцензії GNU.

Переваги MySQL:

– одна з найбільш швидких баз;

– високопродуктивна та відносно проста;

– безкоштовна для домашнього використання;

– розуміє команди SQL, підтримує інтерфейс ODBC, протокол інтерфейсу з базами даних, розроблений фірмою Microsoft.

– може підключитися велика кількість користувачів одночасно;

– призначена для роботи в мережі, захищена від сторонніх втручань;

– працює і під UNIX і під Windows;

– доступний вихідний код.

Є сенс вказати і деякі недоліки СУБД MySQL. Це відсутність можливості використання цілісності посилань, тригерів, процедур і курсорів, що зберігаються.

2.3.3 Тестування варіантів використання

Доцільно провести тестування лише основних варіантів використання. Тестування проводиться з варіантами використання «Авторизація», «Додати гру», «Додати співробітника».

Тестування варіанта використання «Авторизація»

Вхідні дані: слова.

Слово – набір символів латиницею.

Довжина пароля обмежена діапазоном від 1 до 20 символів.

При недопустимих даних на екран виводиться повідомлення в залежності від типу помилку:

- Поле введення логіну або пароля залишилось пустим – повідомлення про обов’язковість заповнення кожного поля з текстом «Заповніть поле»;
- Поле введення логіну заповнене символами кирилиці – повідомлення про обов’язковість заповнення кожного поля з текстом «Заповніть поле».
- У полі введення пароля більше, ніж дозволено символів – повідомлення з текстом «Допустима довжина пароля до 20 символів».

Тестові вимоги:

- Перевірити, чи буде відбуватися перевірка, якщо введене слово написане латиницею;
- Перевірити, чи пройде авторизація, якщо поле введення логіну чи пароля залишилось пустим;
- Перевірити, чи пройде авторизація, якщо перевищено діапазон введення символів.

Розробка випадків для тестових вимог

Перевірити, чи буде відбуватися перевірка діапазону введеного паролю

- Клас еквівалентності: довжина введеного пароля більше ніж 1 і менше ніж 20 символів

Вхідні дані: «jhytdre»

Результат: «true», повідомлення про помилку відсутнє.

- Клас еквівалентності: речення пароля 1 символ.

Вхідні дані: «a»

Результат: «true», повідомлення про помилку відсутнє.

- Клас еквівалентності: довжина пароля більше ніж 20 символів

Вхідні дані: «dsgshjyfkyuoioioioipyewrdfsdghtyjyyyjgfgjytiuiuoyfkykfyser».

Результат: повідомлення «Допустима довжина пароля до 20 символів».

- Клас еквівалентності: поле введення пароля залишилось пустим

Вхідні дані: «».

Результат: повідомлення «Заповніть поле».

Перевірити, чи відбувається перевірка того, що введене слово містить лише латиницю.

- Клас еквівалентності: введений логін містить лише латиницю

Вхідні дані: «dkgtrekyttiy»

Результат: «true», повідомлення про помилку відсутнє.

- Клас еквівалентності: введений логін не містить латиниці.

Вхідні дані: «логін»

Результат: повідомлення «Заповніть поле».

- Клас еквівалентності: введений логін містить латиницю та кирилицю

Вхідні дані: «skajfiertjлогін».

Результат: повідомлення «Заповніть поле»

Перевірити, чи відбувається перевірка того, що в поле логіну введено лише одне слово.

- Клас еквівалентності: введено лише одне слово

Вхідні дані: «elfertpohkoyu»

Результат: «true», повідомлення про помилку відсутнє.

- Клас еквівалентності: введено 2 і більше слів.

Вхідні дані: «sadkfjidgj saofkwepotio»

Результат: повідомлення про помилку з текстом «Заповніть поле».

- Клас еквівалентності: не введено жодного слова

Вхідні дані: «».

Результат: повідомлення про помилку з текстом «Заповніть поле».

Тестування варіанту використання «Додати гру»

Вхідні дані: слова та числа.

Слово – набір символів латиницею або кирилицею.

Довжина назви гри обмежена діапазоном від 1 до 30 символів.

Довжина ціни при цілих числах обмежена діапазоном від 1 до 3 символів.

При дробових числах у введенні ціни ціла частина повинна відділятись комою від дробової.

При недопустимих даних на екран виводиться повідомлення в залежності від типу помилки:

- Будь-яке поле введення залишене пустим – повідомлення з текстом «Заповніть поле»;
- У полі введення ціни на гру ціла частина відокремлена крапкою від дробової – повідомлення з текстом «Заповніть поле»;
- У полі введення ціни на гру є текст - повідомлення з текстом «Заповніть поле»;
- У полі введення назви гри перевищено допустиму кількість символів.

Тестові вимоги:

- Перевірити, чи буде відбуватися перевірка діапазону введеного ім’я;

- Перевірити, чи буде відбуватися перевірка, що будь-яке з полів залишилось пустим;
- Перевірити, чи буде відбуватися перевірка, якщо в поле ціни ввести текст;
- Перевірити, чи буде збережено дані про гру, якщо ціла частина, від дробової частини її ціни відокремлена крапкою.

Розробка випадків для тестових вимог

Перевірити, чи буде відбуватися перевірка діапазону введеної назви гри

- Клас еквівалентності: довжина введеної назви більше ніж 1 і менше ніж 30 символів.

Вхідні дані: «Шашки»

Результат: «true», повідомлення про помилку відсутнє.

- Клас еквівалентності: назва містить речення.

Вхідні дані: «Настільний хокей»

Результат: «true», повідомлення про помилку відсутнє.

- Клас еквівалентності: довжина ім’я більше ніж 30 символів

Вхідні дані: «kjgjrejgithjitrhjhlregflregjrejgorejgojferjgrgjhjthjwjkyjfeufhef».

Результат: повідомлення «Назва має містити до 30 символів».

Перевірити, чи буде відбуватися перевірка, якщо в ціні ціла частина відокремлена крапкою від дробової.

- Клас еквівалентності: введене число відокремлене крапкою

Вхідні дані: «130.50»

Результат: повідомлення «Заповніть поле».

- Клас еквівалентності: введене число відокремлене комою.

Вхідні дані: «130,50»

Результат: «true», повідомлення про помилку відсутнє.

Перевірити, чи відбувається перевірка, що будь-яке з полів залишилось пустим:

- Клас еквівалентності: будь-яке поле залишилось пустим.

Вхідні дані: «».

Результат: повідомлення «Заповніть поле».

Перевірити, чи відбувається перевірка, якщо у поле ціни ввести текст:

- Клас еквівалентності: у поле ціни введено текст.

Вхідні дані: «dksfkgjf».

Результат: повідомлення «Заповніть поле».

- Клас еквівалентності: у поле ціни введено текст та число.

Вхідні дані: «123хаеуаос».

Результат: повідомлення «Заповніть поле».

- Клас еквівалентності: у поле ціни введено число.

Вхідні дані: «45».

Результат: «true», повідомлення про помилку відсутнє.

Тестування варіанта використання «Додати співробітника»

Вхідні дані: слова та числа.

Слово – набір символів латиницею або кирилицею.

Довжина імені обмежена діапазоном від 1 до 50 символів.

Ім’я може містити лише буквені символи.

Електронна пошта має містити символ @.

Логічний статус може приймати лише значення 0 та 1.

При недопустимих даних на екран виводиться повідомлення в залежності від типу помилки:

- Будь-яке поле введення залишене пустим – повідомлення з текстом «Заповніть поле»;
- У полі введення електронної пошти повинен бути символ @ – повідомлення з текстом «Заповніть поле»;
- У полі введення імені є цифри - повідомлення з текстом «Заповніть поле»;
- У полі введення логічного статусу є цифра більша за 1 або буквені символи – повідомлення з текстом «Заповніть поле»;;
- У полі введення імені перевищено допустиму кількість символів – повідомлення з текстом «Ім’я має містити до 50 символів»;.

Тестові вимоги:

- Перевірити, чи буде відбуватися перевірка діапазону введеного ім’я;
- Перевірити чи буде відбуватися перевірка, якщо в полі введення імені присутні цифри;
- Перевірити, чи буде відбуватися перевірка, що будь-яке з полів залишилось пустим;
- Перевірити, чи буде відбуватися перевірка, якщо в поле логічного статусу ввести текст або недопустимі цифри;
- Перевірити, чи буде відбуватися перевірка, якщо у полі введення електронної пошти пропущено символ @.

Розробка випадків для тестових вимог

Перевірити, чи буде відбуватися перевірка діапазону введеного ім’я

- Клас еквівалентності: довжина введеного ім’я більше ніж 1 і менше ніж 50 символів.

Вхідні дані: «Петренко Петро Петрович»

Результат: «true», повідомлення про помилку відсутнє.

- Клас еквівалентності: довжина ім’я більше ніж 50 символів

Вхідні дані:
«kjgjrejgithjitrhjhlregflregjrejgorejgojferjgrgjhjthjwjkyjfeufhefdskflefijeijfiefjiefjie
fjiefjiefjiefjiejiejeiejiejeijfei».

Результат: повідомлення «Ім’я має містити до 50 символів».

Перевірити, чи буде відбуватися перевірка, якщо полі введення логічного статусу недопустима цифра або текст.

- Клас еквівалентності: у полі введення логічного статусу недопустиме число.

Вхідні дані: «2»

Результат: повідомлення «Заповніть поле».

- Клас еквівалентності: у полі введення логічного статусу текст.

Вхідні дані: «dsfdsfdsf»

Результат: Результат: повідомлення «Заповніть поле».

- Клас еквівалентності: у полі введення логічного 0 або 1.

Вхідні дані: «1»

Результат: «true», повідомлення про помилку відсутнє.

Перевірити, чи відбувається перевірка, що будь-яке з полів залишилось пустим:

- Клас еквівалентності: будь-яке поле залишилось пустим.

Вхідні дані: «».

Результат: повідомлення «Заповніть поле».

Перевірити, чи відбувається перевірка, якщо у полі введення електронної пошти пропущено символ @:

- Клас еквівалентності: якщо у полі введення електронної пошти пропущено символ @.

Вхідні дані: «wlkejwqiorj.com».

Результат: повідомлення «Заповніть поле».

- Клас еквівалентності: якщо у полі введення електронної пошти присутній символ @.

Вхідні дані: «csjfjnjffnjg@gmail.com».

Результат: «true», повідомлення про помилку відсутнє.

Перевірити, чи відбувається перевірка, якщо у полі введення імені присутні цифри:

- Клас еквівалентності: якщо у полі введення імені присутні цифри.

Вхідні дані: «Іванов Іван 438458».

Результат: повідомлення «Заповніть поле».

- Клас еквівалентності: якщо у полі введення імені немає цифр.

Вхідні дані: «Іванов Іван Іванович».

Результат: «true», повідомлення про помилку відсутнє.

2.3.4 Випробування програми

Програму і методику випробувань розробленого програмного забезпечення наведено в Додатку Д. На основі інструкцій, що наведені в цьому додатку, проведемо випробування розробленого програмного забезпечення. Після переходу користувачем у розділ авторизації йому буде доступна спеціальна форма. Форма авторизації наведена на рисунку 2.14:

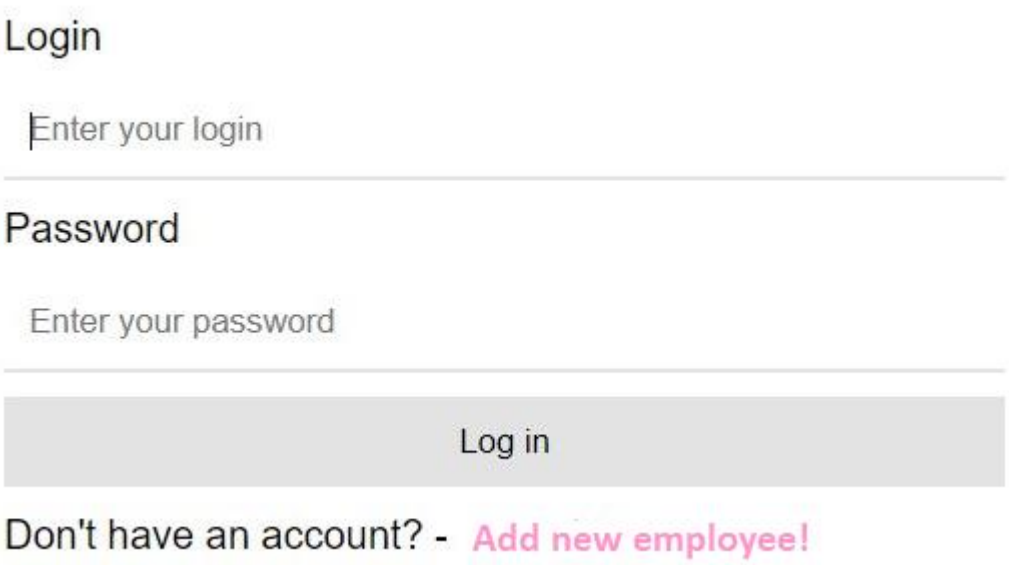

Рисунок 2.14 – Форма авторизації

Якщо у співробітника немає облікового запису, то є можливість його створити. Адміністратор натискає кнопку «Add new employee» та заповнює форму додавання співробітника. На рисунку 2.15 наведено форму додавання співробітника:

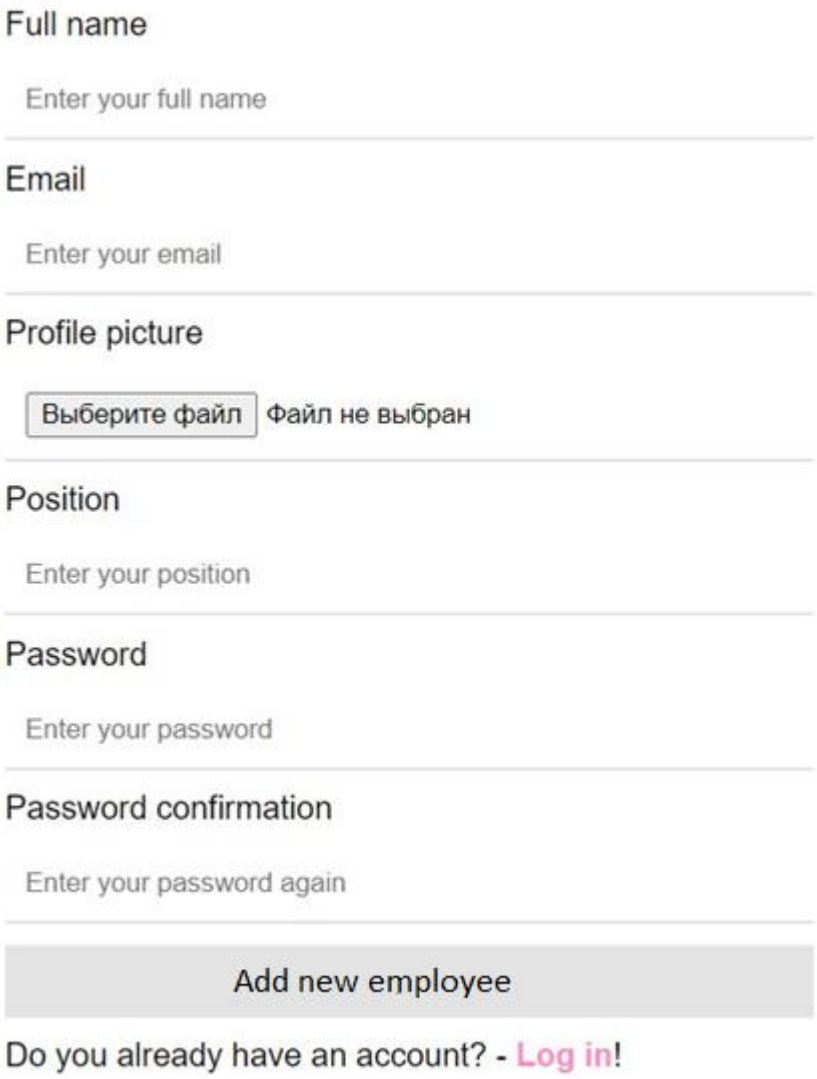

Рисунок 2.15 – Форма додавання співробітника

Коли при авторизації користувач вводить неправильні дані, то система виводить повідомлення про помилку, що приведене на рисунку 2.16.

Рисунок 2.16 – Повідомлення про неправильно введені дані

Коли при додаванні співробітника у полі введення електронної пошти пропущено символ @, то виводиться повідомлення про помилку, що приведене на рисунку 2.17.

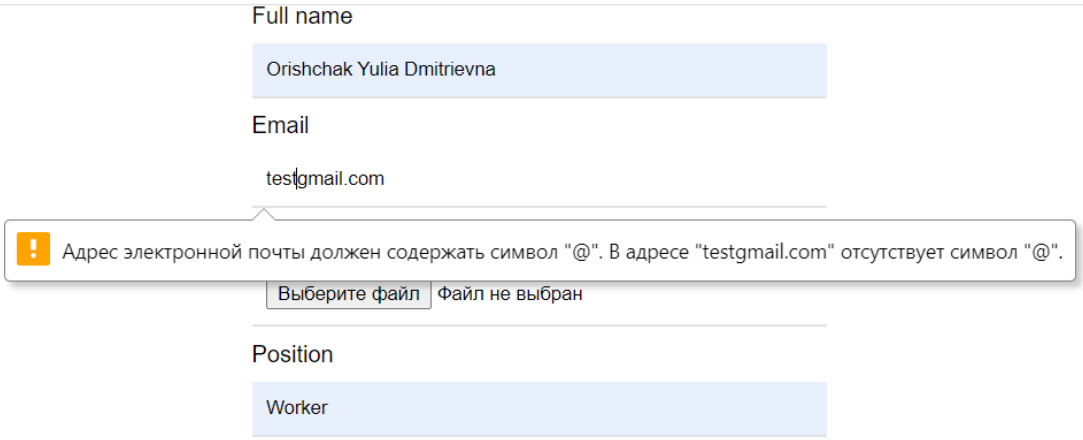

Рисунок 2.17 – Повідомлення про відсутність символу @ у полі введення електронної пошти

Коли у полях введення пароля та перевірки введення пароля введено різні паролі, то виводиться повідомлення про помилку, що приведено на рисунку 2.18.

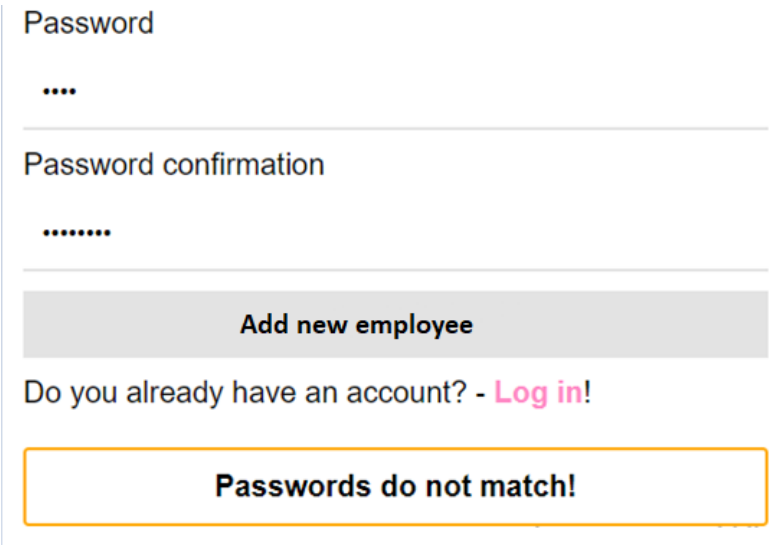

Рисунок 2.18 – Повідомлення про незбіжність паролів

Якщо додавання співробітника пройшло успішно, то про це виводиться відповідне повідомлення, що представлено на рисунку 2.19.

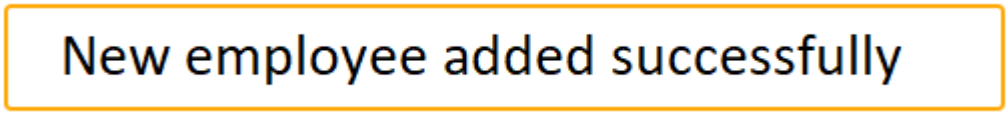

Рисунок 2.19 – Повідомлення про успішне додавання співробітника

3 РЕЗУЛЬТАТИ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ВЕБСАЙТУ ДЛЯ КЛУБУ ПРОКАТУ НАСТІЛЬНИХ ІГОР

В процесі виконання кваліфікаційної роботи було розв’язано практичну задачу інженерії програмного забезпечення, що характеризується комплексністю та невизначеністю умов, а саме розроблено програмне забезпечення вебсайту для клубу прокату настільних ігор «Гральна кістка». Розроблене програмне забезпечення задовольняє вимогам, які поставлені у технічному завданні.

Програмне забезпечення розроблено з використанням мов PHP, HTML, CSS.

Проведені випробування над програмним забезпеченням показали, що програма працездатна, відповідає характеристикам, готова до впровадження. Випробування були виконані згідно з програмою та методикою випробування які наведені у Додатку Б.

Програмно-апаратне забезпечення для клієнта:

– будь-яка операційна система, що забезпечує мережеву взаємодію по протоколу TCP/IP;

– будь-який веббраузер, бажано із підтримкою скриптових мов програмування JavaScript, наприклад, Mozilla Firefox, Internet Explorer або Google Chrome 30.0 і вище;

– підключення до Internet.

Інструкція користувача для роботи з розробленим програмним забезпеченням наведена в Додатку В.

4 ОХОРОНА ПРАЦІ

Інструкція з охорони праці ІТ фахівця містить обов’язкові для дотримання фахівцем з інформаційних технологій правил з охорони праці при виконанні робіт, визначених його функціональними обов’язками.

Фахівцем з інформаційних технологій може працювати особа, яка досягла 18-річного віку, має вищу освіту відповідного напряму, володіє екранним пристроєм на рівні стандартного прикладного програмного забезпечення, основами мережевих технологій, пройшла вступний інструктаж з питань охорони праці, первинний інструктаж на робочому місці, інструктаж з пожежної безпеки, а також пройшла медичний огляд та не має протипоказань до виконання обов’язків за станом здоров'я.

Робочий стіл для роботи фахівця підбирається достатнього розміру, з поверхнею, що має низьку відбивну здатність та допускає можливість зручного розміщення екрана, клавіатури, документів і відповідного устаткування. Крісло має бути стійким, легко рухатися та займати зручне положення. Сидіння повинно регулюватися по висоті, а спинка сидіння — як по висоті, так і по нахилу. Екран повинен вільно повертатися і нахилятися відповідно до потреби працівника, зображення на ньому має бути стабільним, яскравість та контрастність символів повинні легко регулюватися.

Вимоги безпеки перед початком роботи

Перед початком роботи фахівець повинен отримати інструктаж від безпосереднього керівника.

Переконатися візуально у відсутності механічних ушкоджень корпусів екранного пристрою та засобів оргтехніки, у наявності інженерно-технічних засобів безпеки, аптечки та засобів пожежогасіння.

Впевнитись у якості освітлення, наявності рекомендованого контрасту між екраном і навколишнім середовищем. Робоче місце має бути

розташованим таким чином, щоб природне світло падало збоку, переважно зліва.

Перевірити справність шнурів живлення, вимикачів, розеток, штепсельних з'єднань, надійність роботи вентиляційної системи.

Очистити екранний пристрій від пилу та інших забруднень.

Повідомити безпосереднього керівника про несправність екранного та периферійних пристроїв, засобів оргтехніки, меблів, приладів, електропроводки, іншого обладнання та не приступати до роботи до повного усунення недоліків.

Вимоги безпеки під час виконання роботи

Утримувати в порядку і чистоті робоче місце. Не захаращувати проходи до робочого місця сторонніми предметами та речами, які зменшують тепловіддачу засобів оргтехніки та іншого обладнання.

Стежити за справністю екранного пристрою, засобів оргтехніки та іншого устаткування, дотримуватися правил їх експлуатації.

При тривалій відсутності на робочому місці відключати від електромережі засоби оргтехніки та інше устаткування, крім обладнання, яке використовується цілодобово.

Забезпечувати щоденне вологе прибирання та щогодинне провітрювання робочого приміщення.

Контролювати виконання рекомендацій щодо: проведення регламентованих законодавством і правилами внутрішнього розпорядку короткочасних перерв в роботі; проведення комплексу вправ для очей, для рук, для хребта (вироблення правильної постави), для поліпшення мозкового кровообігу, а також сеансів психофізіологічного розвантаження.

Вимоги безпеки після закінчення роботи

Після закінчення роботи фахівець зобов'язаний: відключити від електромережі екранний пристрій, засоби оргтехніки та інше устаткування, крім обладнання, яке використовується цілодобово (апарати факсимільного зв'язку, мережеві сервери тощо); упорядкувати робоче місце, провітрити

приміщення; привести себе у порядок, вимити руки та обличчя та переодягнутися.

Про особливості в роботі та виявлені недоліки повідомити безпосереднє керівництво.

ВИСНОВКИ

В ході виконання кваліфікаційної роботи було розв’язано практичну задачу інженерії програмного забезпечення, що характеризується комплексністю та невизначеністю умов, а саме розроблено програмне забезпечення вебсайту для клубу прокату настільних ігор «Гральна кістка».

В роботі були вирішені наступні задачі:

- проведено аналіз предметної області
- розроблено ескізний проєкт програмного забезпечення;
- розроблено технічний проєкт програмного забезпечення;
- розроблено робочий проєкт;

Результати розробки програмного забезпечення повністю задовольняють поставленій меті

У кваліфікаційній роботі також було наведено положення та норми охорони праці в офісах, до яких також належить і клуб прокату настільних ігор.

Для подальшого розвитку програмного забезпечення слід приділити увагу покращенню наявних можливостей та реалізувати функцію реєстрації відвідувачів, що дозволить замовляти ігри в прокат онлайн.

Розроблене програмне забезпечення рекомендується для впровадження в клубах прокату настільних ігор.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Способи створення вебсайтів. URL: https://cityhost.ua/uk/blog/top-6-luchshih-konstruktorov-saytov-v-2022-godu.html#Wix (дата звернення 31.05.2022).
2. Види CMS. URL: https://katerinka1200.github.io/index02.html (дата звернення 31.05.2022).
3. Розробка вебсайтів з нуля за допомогою мов програмування. URL: https://kiev.itstep.org/blog/web-development-where-to-start-learning (дата звернення 05.06.2022).
4. D20.in.ua. URL: https://d20.in.ua (дата звернення: 08.04.2022).
5. PhpStorm переваги. URL: http://chvv.com.ua/top-populyarnyh-ide-dlya-programuvannya-na-s/ (дата звернення 07.06.2022).
6. Переваги мови Php. URL: https://oracle-patches.com/web/%D1%87%D1%82%D0%BE-%D0%BF%D1%80%D0%B5%D0%B4%D1%81%D1%82%D0%B0%D0%B2%D0%BB%D1%8F%D0%B5%D1%82-%D1%81%D0%BE%D0%B1%D0%BE%D0%B9-%D1%80%D0%BD%D1%80 (дата звернення 07.06.2022)
7. Переваги мови HTML. URL: https://kztarif.ru/kompjutery/html-pljusy-i-minusy (дата звернення 07.06.2021)
8. Переваги мови CSS. URL: https://www.prostobiz.ua/biznes/biznes_start/stati/css_dlya_nachinayuschih_istoriya_poyavleniya_plyusy_i_minusy#3 (дата звернення 07.06.2022)
9. Переваги мови JavaScript. URL: https://ipipe.ru/info/javascript (дата звернення 07.06.2022)

ДОДАТОК А ТЕХНІЧНЕ ЗАВДАННЯ

1 Вступ

1.1 Найменування програми

Найменування програми – вебсайт для клубу прокату настільних ігор «Гральна кістка».

1.2 Коротка характеристика використання

Розроблюваний вебсайт призначений для автоматизації роботи клубу прокату настільних ігор.

2 Підстави для розробки

2.1 Підстави для проведення розробки

Розробка вебсайту здійснюється на підставі завдання на кваліфікаційну роботу бакалавра.

2.2 Найменування та умовне позначення теми розробки

Найменування розробки – «Розробка програмного забезпечення вебсайту для клубу прокату настільних ігор».

Умовне позначення – «Гральна кістка».

3 Призначення розробки

3.1 Функціональне призначення

Функціональним призначенням програми є надання співробітникам клубу настільних ігор можливості вести облік прокату ігор через вебсайт замість паперових журналів.

3.2 Експлуатаційне призначення

Програма призначена для експлуатації на комп’ютері клубу прокату настільних ігор.

4 Вимоги до програми або програмного продукту

4.1 Вимоги для функціональних характеристик

4.1.1 Вимоги до переліку виконуваних функцій

Розроблений вебсайт має надавати співробітнику клубу настільних ігор можливість додавання співробітника, авторизуватися та виходити з облікового запису, переглядати список ігор, змінювати їх логічний статус, видавати ігри в прокат, здійснювати розрахунок готівкою. Обліковий запис співробітника буде містити наступну інформацію:

1) Id;
2) ПІБ;
3) Посада;
4) Фотографія;
5) Електронна пошта;
6) Логін;
7) Пароль.

Крім цього розроблюваний вебсайт має надавати можливість співробітнику здійснювати будь-які зміни в інформації про ігри, додавати нові ігри та видаляти наявні, а також синхронізувати внесені зміни з базою даних.

База даних буде містити всю інформацію про ігри, співробітників, розрахунки готівкою, а також синхронізуватися з вебсайтом.

Для збереження конфіденційності профілів користувача – база даних буде автоматично хешувати пароль, аби він був відомим лише одній людині.

Для забезпечення надійності роботи вебсайту було передбачено та виключено можливість входу та виходу з облікового запису через введення логіну у рядок пошуку браузера.

Для забезпечення коректного запису пароля співробітника – при додаванні співробітника необхідно ввести пароль двічі. Якщо в обох випадках пароль введено однаково – створюється обліковий запис співробітника, а якщо ні – виводиться повідомлення про помилку в паролі.

4.1.2 Вимоги до вхідних та вихідних даних

Вхідними даними є:

1) Додавання співробітника – id, ПІБ, фотографія, логін, електронна пошта, посада та пароль користувача розроблюваного вебсайту. Текстова інформація буде введена за допомогою комп’ютерної клавіатури. Фотографія – як зображення формату .png або .jpeg (jpg) буде додаватись з пам’яті комп’ютера.
2) Інформація про ігри у вигляді назви, вартості прокату за годину, опису складу та правил гри та її логічний статус, що дає зрозуміти чи вільна гра для прокату в цей момент.
3) Інформація про відвідувача у вигляді id взятої у прокат гри, розміру заставної вартості, часу прокату гри та прізвища відвідувача та співробітника, який його обслуговував.
4) Інформація про розрахунок готівкою у вигляді id гри, суми, дати проведення розрахунку та прізвища співробітника, який проводив розрахунок.

Вихідними даними є:

1) Запис у базі даних про створення облікового запису для користувача, що є співробітником клубу настільних ігор

та можливість авторизуватись на сайті та перейти до облікового запису у вебсайті та вийти з нього, як наслідок.

2) Показ у вебсайті інформації про ігри у вигляді таблиці, що зберігається у базі даних.
3) Завантажена раніше фотографія співробітника, як складова його облікового запису з зображенням.

4.1.3 Вимоги до часових характеристик

Вимогами до часових характеристик є оплата за кількість годин користування грою та наявність дати проведення розрахунку готівкою.

4.2 Вимоги до надійності

4.2.1 Вимоги до забезпечення надійного функціонування програми

Надійне (стійке) функціонування розроблюваного програмного забезпечення має бути забезпечене виконанням сукупності організаційно-технічних заходів ГОСТ 51188-98. Захист інформації.

4.2.2 Час відновлення після відмови

Час на відновлення після відмови, яка була викликана несправністю технічних засобів або фатальним збоєм операційної системи – не має перевищувати час, що необхідний для усунення причини несправності технічних засобів або перевстановлення операційної системи чи програмних засобів.

4.2.3 Відмова з причин некоректних дій користувача

Відмова програми можлива якщо користувач взаємодіє з операційною системою. Задля запобігання відмови такого роду – необхідно надати тільки необхідний мінімум прав користувачеві.

4.3 Умови експлуатації

4.3.1 Кліматичні умови експлуатації

Кліматичні умови експлуатації, за яких повинні забезпечуватись задані характеристики, повинні задовольняти вимогам до технічних засобів в частині умов їх експлуатації.

4.3.2 Вимоги до видів обслуговування

Програма не вимагає проведення будь-яких видів обслуговування.

4.3.3 Вимоги до чисельності та кваліфікації персоналу

Вимоги до чисельності та кваліфікації персоналу не висуваються.

4.4 Вимоги до складу та параметрів технічних засобів

Системні вимоги: Веббраузер: Internet Explorer 10.0 та вище, або Firefox 3.5 та вище, або Opera 9.5 та вище, або Safari 3.2.1 та вище, або Chrome 2 та вище. Включена підтримка javascript.

4.5 Вимоги до інформаційної та програмної сумісності

4.5.1 Вимоги до інформаційних структур та методів розв’язку

Для зберігання інформації по категоріях – необхідно створити базу даних. У якості СУБД використати mysql. У файлі бази даних створити п’ять таблиць: «Гра», «Співробітники», та «Розрахунок готівкою».

У таблиці «Ігри» зберігати наступну інформацію про гру:

1) Id;
2) Id співробітника;

3) Назва;
4) Вартість прокату за годину;
5) Опис гри;
6) Логічний статус.

У таблиці «Співробітники» зберігати наступну інформацію про співробітників:

1) Id;
2) ПІБ;
3) Фотографія;
4) Електронна пошта;
5) Посада;
6) Логін;
7) Пароль.

У таблиці «Розрахунок готівкою» зберігати наступну інформацію про розрахунок готівкою:

1) Id гри;
2) Id співробітника;
3) Дата;
4) Сума.

4.5.2 Вимоги до вихідного коду та мов програмування

Для розробки програмного коду будуть використані мови Php, HTML, Javascript, Css.

4.5.3 Вимоги до програмних засобів, що використовуються програмою

Системні програмні засоби, що використовуються програмним забезпеченням, повинні бути представлені операційною системою Windows.

4.5.4 Вимоги до захисту інформації та програм

Вимоги до захисту інформації та програм не висуваються.

4.6 Вимоги до маркування та пакування

Програмне забезпечення розповсюджуватиметься через мережу інтернет без використання фізичних носіїв, тому вимоги до маркування, пакування, транспортування та зберігання – не висуваються.

4.6.1 Вимоги до маркування

Вимоги до маркування не висуваються.

4.6.2 Вимоги до пакування

Вимоги до пакування не висуваються.

4.6.3 Умови пакування

Вимоги до умов пакування не висуваються.

4.6.4 Порядок пакування

Вимоги до порядку пакування не висуваються.

4.7 Вимоги до транспортування та зберігання

Вимоги до транспортування та зберігання не висуваються.

4.7.1 Умови транспортування та зберігання

Вимоги щодо умов транспортування та зберігання не висуваються.

4.7.2 Спеціальні вимоги

Програмне забезпечення має забезпечити взаємодію з користувачем через графічний інтерфейс користувача, розроблений за

побажанням замовника та інтуїтивно зрозумілий для інших користувачів.

5 Вимоги до програмної документації

5.1 Попередній склад програмної документації

Склад програмної документації має містити:

1) Технічне завдання;
2) Програму та методику випробувань;
3) Інструкцію користувача;
4) Опис програмного забезпечення.

6 Техніко-економічні показники

Орієнтовна економічна ефективність не розраховується.

7 Стадії та етапи розробки

7.1 Стадії розробки та етапи розробки

Розробка має бути проведена в п’ять етапів:

1) Розробка технічного завдання;
2) Ескізний проєкт;
3) Технічний проєкт;
4) Робочий проєкт;
5) Впровадження.

7.2 Склад робіт по етапах

На етапі розробки технічного завдання необхідно виконати наступні задачі:

1) Аналіз предметної області;
2) Виявлення та уточнення вимог до технічних засобів;
3) Виявлення вимог до програмного забезпечення;

4) Виявлення вимог до складу програмної документації. Щодо програмної документації користуватись ДСТУ 19.101-77 «Попередній склад програмної документації».
5) Вибір мови програмування та інших програмних засобів;
6) Узгодження та затвердження технічного завдання;
7) Постановка задачі.

На етапі ескізного проєктування необхідно виконати наступні кроки:

1) Змоделювати контекстну діаграму програмного забезпечення.
2) Змоделювати сценарії роботи програмного забезпечення за допомогою діаграми варіантів використання з наступною розробкою специфікації варіантів використання.
3) Змоделювати сценарії роботи основних варіантів використання за допомогою діаграми діяльності.
4) Розробити діаграму класів програмного забезпечення.
5) Розробити проєкт інтерфейсу користувача.

На етапі технічного проєктування необхідно виконати наступні види робіт:

1) Розробити статичну модель (діаграму класів).
2) Створити специфікації змодельованих класів.
3) Виконати динамічне моделювання програми за допомогою діаграми послідовностей або діаграми кооперацій для основних варіантів використання.
4) Виконати моделювання логічної моделі даних на основі інформаційної моделі розроблюваного програмного забезпечення.

На етапі робочого проєктування необхідно виконати наступні види робіт:

1) Обґрунтувати доцільність використання обраної мови програмування та інших програмних засобів.
2) Розробити фізичну модель програмного забезпечення.
3) Виконати моделювання компонентів програмного забезпечення за допомогою діаграми компонентів.
4) Виконати розробку та тестування програмних модулів.
5) Виконати розробки інструкції користувача.
6) Провести випробування розробленого програмного забезпечення.

Відповідно випробування розроблюваного програмного забезпечення включає наступні етапи:

1) Розробка, узгодження та затвердження програми та методики випробувань;
2) Виправлення програми та програмної документації за результатами випробувань;
3) Проведення приймально-здавальних робіт.

На етапі впровадження необхідно виконати роботи з підготовки та передачі розробленої програмного забезпечення та програмної документації Замовнику.

Строки виконання етапів розробки програми представлено в таблиці 1

Таблиця 1 – Строки виконання роботи етапів розробки програмного забезпечення «Гральна кістка»

Назва етапу	Початок роботи	Кінець роботи
Аналіз предметної області	01.02.2022	04.02.2022
Виявлення та уточнення вимог до технічних засобів	05.02.2022	08.02. 2022
Виявлення вимог до програмного забезпечення	09.02.2022	12.02. 2022
Виявлення вимог до складу програмної документації	15.02.2022	17.12. 2022
Вибір мови програмування та інших програмних засобів	18.02. 2022	18.02. 2022
Узгодження та затвердження технічного завдання	19.02.2022	23.02.2022
Постановка задачі	23.02.2022	23.02.2022
Проєктування поведінки за допомогою варіантів використання	24.02.2022	26.02.2022
Специфікація варіантів використання	01.03.2022	03.03.2022
Моделювання діяльності варіантів використання	04.03.2022	05.03.2022
Розробка концептуальної моделі	08.03.2022	10.03.2022
Розробка проєкту інтерфейсу користувача	11.03.2022	15.03.2022
Розробка статичної моделі	16.03.2022	17.03.2022
Розробка специфікації статичної моделі	18.03.2022	19.03.2022
Динамічне моделювання поведінки системи	22.03.2022	23.03.2022
Логічне моделювання системи на основі інформаційної моделі	23.03.2022	24.03.2022
Обґрунтування обраних програмних засобів	24.03.2022	31.03.2022
Моделювання компонентів системи	31.03.2022	15.04.2022
Моделювання розгортання системи	15.04.2022	26.04.2022
Розробка програмних модулів	26.04.2022	23.05.2022
Тестування програмних модулів	23.05.2022	30.05.2022
Розробка інструкції користувача	31.05.2022	09.06.2022
Розробка програми та методики випробувань	10.06.2022	14.06.2022
Виправлення програми та програмної документації за результатами випробувань	14.06.2022	26.06.2022
Проведення приймально-здавальних робіт	27.06.2022	03.07.2022

8 Порядок контролю та приймання

Контроль за кожним етапом проєктування програмного забезпечення відбувається з урахуванням усіх вимог, визначених в цьому технічному завданні. Кожна стадія розробки повинна бути представлена в зазначені строки та узгоджена із замовником.

Приймально-здавальні роботи мають проводитись під наглядом Замовника або уповноваженої ним особи з 27.06.22 по 03.07.22.

Приймально-здавальні випробування повинні проводитись згідно розробленої виконавцем не пізніше 03.07.22 та узгодженою з Замовником «Програми та методики випробувань».

Хід проведення приймально-здавальних робіт Замовник та Виконавець документують в «Протокол проведення випробувань».

На основі Протоколу проведення випробувань Виконавець та Замовник підписують Акт приймання та здачі програми в експлуатацію.

ДОДАТОК Б ПРОГРАМА ТА МЕТОДИКА ВИПРОБУВАНЬ

1. Об’єкт випробування

Об’єктом випробування є програмне забезпечення вебсайту для клубу прокату настільних ігор.

2. Мета випробування

Метою проведення випробування є перевірка працездатності даного про програмного забезпечення, перевірка відповідності характеристик та вимог розробленої програми, викладених у технічному завдані, приведення прикладу роботи та отримання відповідних навичок.

3. Вимоги до додатка

При проведенні випробувань функціональні характеристики (можливості) програми підлягають перевірці на відповідність вимогам, викладеним у пункті «Вимоги до функціональних характеристик» технічного завдання.

4. Вимоги до програмної документації

4.1 Склад програмної документації, пропонованої на випробування

Програмна документація повинна містити наступні документи:

1) Текст програми;
2) Опис програми;
3) Програма та методика випробувань.
4) Інструкція користувача;

4.2 Спеціальні вимоги

Спеціальні вимоги до програмної документації не висуваються.

5. Засоби та порядок випробувань

5.1 Технічні засоби, що використовуються під час випробувань

Склад технічних засобів, що використовувалися під час випробувань:

- Процесор Intel Core i3 8130U 2.2 ГГц;

- Оперативна пам’ять 4 Гб і вище;
- Жорсткий диск близько 500 Гб(система + дані);

5.2 Програмні засоби, що використовуються під час випробувань

Програмне забезпечення не пред'являє ніяких вимог до апаратної платформи.

Склад програмних засобів, що використовувалися під час випробувань:

Браузери:

- Mozzilla FireFox 89.0 ;
- Google Chrome 91.0 ;
- Opera 75.0.

5.3 Порядок проведення випробувань

Порядок проведення випробувань складається з перевірки вимог до функціональних характеристик програмного продукту та перевірки вимог до програмної документації.

6. Методи випробувань

6.1Методика проведення перевірки вимог до програмної документації

Перевірка дотримання вимог програмної документації на програмний продукт виконується візуально представником служби, відповідальної за експлуатацію. У ході перевірки зіставляється склад і комплектність програмної документації, представленої розробником, з переліком програмної документації, наведеним у пункті «Склад програмної документації, пропонованої на випробування» цього документа.

Перевірка вважається завершеною у випадку відповідності складу та комплектності програмної документації, представленої розробником, переліком програмної документації, наведеному у зазначеному вище пункті.

6.2Методика проведення перевірки вимог до функціональних характеристик програмного продукту

Перевірка працездатності програми виконується згідно з пунктом «Вимоги до функціональних характеристик» підрозділу 1.5 Постановка задачі.

Перевірка вважається завершеною у разі відповідності складу і послідовності виконаних дій пункту «Вимоги до функціональних характеристик». В процесі тестування була перевірена функціональність програмного забезпечення для клубу прокату настільних ігор. У табл. 1, що наведена нижче, вказано перелік випробувань основних функціональних можливостей.

ДОДАТОК В ІНСТРУКЦІЯ КОРИСТУВАЧА

Після завантаження вебсайту користувачу буде запропоновано авторизуватись. На рисунку В.1 представлено сторінку авторизації.

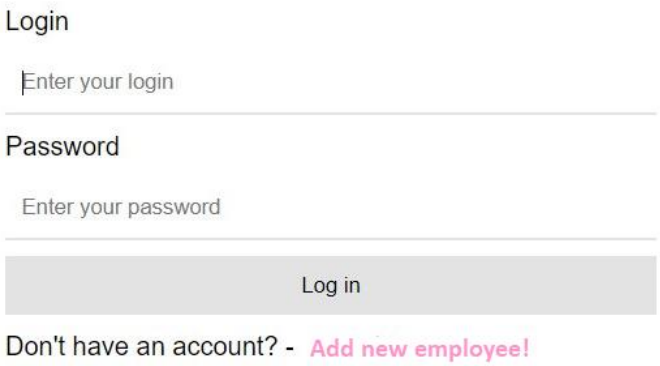

Рисунок В. 1 – Сторінка авторизації

Якщо співробітник ще не має облікового запису, то йому буде запропонована форма додавання співробітника, яку наведено на рисунку В.2.

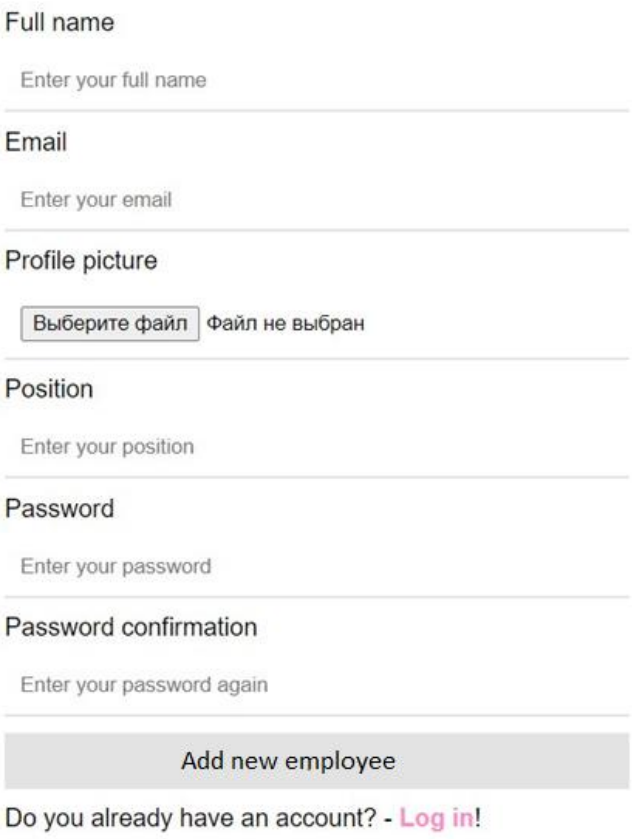

Рисунок В. 2 – Форма додавання співробітника

Користувач може додати гру, заповнивши відповідну форму, що приведена на рисунку В.3

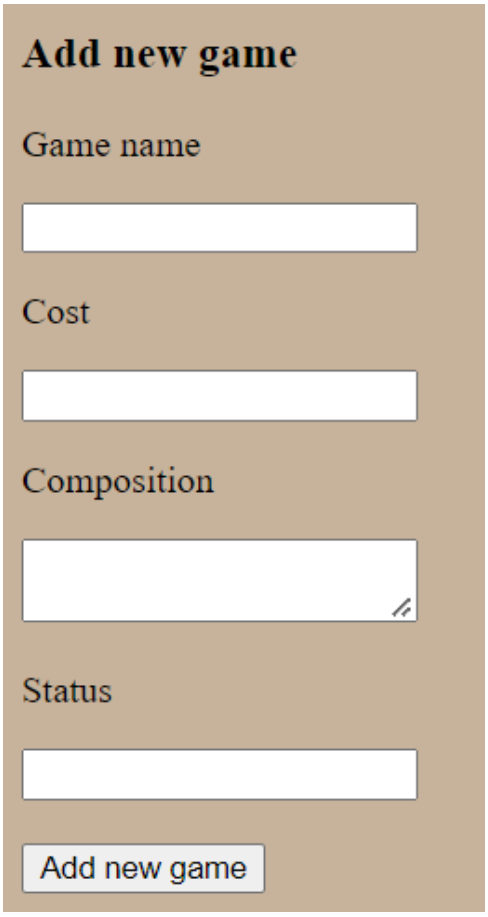

Рисунок В. 3 – Форма для додання нової гри

При переході до облікового запису користувача – йому надається можливість вийти з вебсайту обравши відповідну функцію, що показано на рисунку В.4

Рисунок В. 4 – Обліковий запис користувача

Для перегляду інформації про конкретну гру потрібно обрати відповідну функцію у таблиці з іграми, після чого відкриється вебсторінка з повним описом гри, яка показана на рисунку В.5

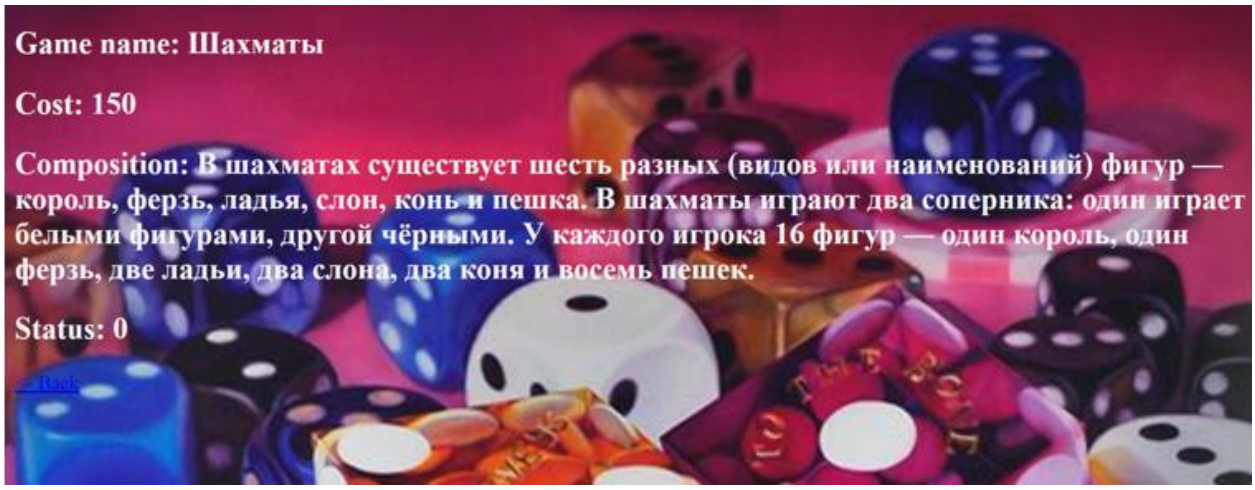

Рисунок В. 5 – Перегляд повного опису гри

При необхідності змінити інформацію про гру потрібно обрати необхідну функцію у таблиці з іграми та заповнити спеціальну форму, яку приведено на рисунку В.6

Рисунок В. 6 – Форма для редагування інформації про гру

ДОДАТОК Г ТЕКСТ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

conn.php

```
<?php
    $conn = mysqli_connect('127.0.0.1:3307', 'root', '', 
'boardgame');
if (!$conn) {
    die('Error connect to database!');
}
```

logout.php

```
<?php
session_start();
unset($_SESSION['user']);
header('Location: ../index.php');
```

signin.php

```
<?php
    session_start();
    require_once 'conn.php';
    $login = mysqli_real_escape_string($conn, $_POST['login']);
    $password = md5($_POST['password']);
    $check_user = mysqli_query($conn, "SELECT * FROM `employee` 
WHERE `email` = '$login' AND `password` = '$password'");
    if (mysqli_num_rows($check_user) > 0) {
        header('Location: ../datab.php');
        $user = mysqli_fetch_assoc($check_user);
        $_SESSION['user'] = [
            "full_name" => $user['full_name'],
            "avatar" => $user['avatar'],
            "email" => $user['email']
        ];
    } else {
```

```
        $_SESSION['message'] = 'Wrong login or password';
        header('Location: ../index.php');
    }
```

addEmployee.php

```
<?php
    session_start();
    require_once 'conn.php';
    $full_name =
mysqli_real_escape_string($conn,$_POST['full_name']);
    $email = mysqli_real_escape_string($conn,$_POST['email']);
    $position =
mysqli_real_escape_string($conn,$_POST['position']);
    $password =
mysqli_real_escape_string($conn,$_POST['password']);
    $password_confirm =
mysqli_real_escape_string($conn,$_POST['password_confirm']);
    if ($password === $password_confirm){
        $path = 'uploads/' . time() . $_FILES['avatar']['name'];
        if (!move_uploaded_file($_FILES['avatar']['tmp_name'],
'../' . $path)) {
            $_SESSION['message'] = 'Error loading message';
            header('Location: ../add.php');
        }
        $password = md5($password);
        mysqli_query($conn, "INSERT INTO `employee`
(`full_name`, `email`, `password`, `avatar`, `id`, `position`)
VALUES ('$full_name', '$email', '$password', '$path', NULL,
'$position')");
        $_SESSION['message'] = New employee added successfully';
        header('Location: ../index.php');
    } else {
        $_SESSION['message'] = 'Passwords do not match!';
        header('Location: ../add.php');
    }
    ?>
```

```
<pre>
</pre>
```

main.css

```
*{
    margin: 0;
    padding: 0;
    box-sizing: border-box;
}
body{
    height: 100vh;
    display: flex;
    align-items: center;
    justify-content: center;
    font-family: Montserrat, sans-serif;
}
a {
    color: #ff8ac6;
    font-weight: bold;
    text-decoration: none;
}
p{
    margin: 10px 0;
}
form{
    display: flex;
    flex-direction: column;
    width: 400px;
}
input{
    margin: 10px 0;
    padding: 10px;
    border: unset;
```

```
    border-bottom: 2px solid #e3e3e3;
    outline: none;
}
button{
    padding: 10px;
    background: #e3e3e3;
    border: unset;
    cursor: pointer;
}
.message {
    border: 2px solid #ffa908;
    border-radius: 3px;
    padding: 10px;
    text-align: center;
    font-weight: bold;
}
.logout {
    color: #ffa908;
}
```

delete.php

```
<?php
require_once '../config/conn.php';
$id = $_GET['id'];
mysqli_query($conn, "DELETE FROM `games` WHERE `games`.`id` =
'$id'");
header('Location: /');
```

vendor/update.php

```
<?php
require_once '../config/conn.php';
$id = mysqli_real_escape_string($conn,$_POST['id']);
//$gamename =
mysqli_real_escape_string($conn,$_POST['gamename']);
```

```
$composition =
mysqli_real_escape_string($conn,$_POST['composition']);

$cost = mysqli_real_escape_string($conn,$_POST['cost']);

$status = mysqli_real_escape_string($conn,$_POST['status']);

mysqli_query($conn, "UPDATE `games` SET `cost` = '$cost',
`composition` = '$composition', `status` = '$status' WHERE
`games`.`id` = '$id'");

header('Location: /');
```

create.php

```
<?php

require_once '../config/conn.php';

$gamename = mysqli_real_escape_string($conn,$_POST['gamename']);

$cost = mysqli_real_escape_string($conn,$_POST['cost']);

$composition =
mysqli_real_escape_string($conn,$_POST['composition']);

$status = mysqli_real_escape_string($conn,$_POST['status']);

mysqli_query($conn, "INSERT INTO `games` (`id`, `gamename`,
`cost`, `composition`, `status`) VALUES (NULL, '$gamename',
'$cost', '$composition', '$status')");

header('Location: /');
```

datab.php

```
<?php

require_once 'config/conn.php';

?>

<!doctype html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <title>Games</title>

</head>

<td><a href="profile.php?id=">Profile</a></td>

<style>

    body {

        background: #c7b39b; /* Колір фону та шлях до файлу */
```

```
        color: #000000; /* Колір тексту */
        background-size: 100%;
    }
    th, td {
        padding: 10px;
    }
    th {
        background: #ee00ff;
        color: #fff;
    }
    td {
        background: #f9abff;
    }
</style>
<body>
<table>
    <tr>
        <th>ID</th>
        <th>GAMENAME</th>
        <th>COST</th>
        <th>COMPOSITION</th>
        <th>STATUS</th>
    </tr>
    <?php
        $games = mysqli_query($conn, "SELECT * FROM `game`");
        $games = mysqli_fetch_all($games);
        foreach ($games as $game) {
            ?>
            <tr>
                <td><?= $game[0] ?></td>
                <td><?= $game[1] ?></td>
                <td><?= $game[2] ?>$</td>
```

```
                <td><?= $game[3] ?></td>
                <td><?= $game[4] ?></td>
                <td><a href="game.php?id=<?= $game[0]
?>">View</a></td>
                <td><a href="update.php?id=<?= $game[0]
?>">Update</a></td>
                <td><a style="color: #ff0000;"
href="vendor/delete.php?id=<?= $game[0] ?>">Delete</a></td>
            </tr>
            <?php
        }
    ?>
    </table>
<head>
    <meta charset="utf-8">
    <style>
        .leftimg {
            float:left; /* Вирівнювання по лівому краю */
            margin: 7px 7px 7px 0; /* Відступи навколо
зображення */
        }
        .rightimg  {
            float: right; /* Вирівнювання по правому краю  */
            margin: 7px 0 7px 7px; /* Відступи навколо
зображення */
        }
    </style>
</head>
<body>
<h3>Add new game</h3>
<form action="vendor/create.php" method="post">
    <p>Game name</p>
    <input type="text" name="gamename">
    <p>Cost</p>
```

```
    <input type="number" name="cost">
    <p>Composition</p>
    <textarea name="composition"></textarea>
    <p>Status</p>
    <input type="bool" name="status"><br> <br>
    <button type="submit">Add new game</button>
</form>
</body>
</html>
```

game.php

```
<?php
require_once 'config/conn.php';
$game_id = $_GET['id'];
$game = mysqli_query($conn, "SELECT * FROM `games` WHERE `id` =
'$game_id'");
$game = mysqli_fetch_assoc($game);
?>
<!doctype html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Games</title>
</head>
<style>
    body {
        background: #c7b39b url(image.jpg); /* Цвет фона и путь
к файлу */
        color: whitesmoke; /* Цвет текста */
        background-size: 100%;
    }
</style>
<body>
<body>
```

```
<h2>Game name: <?= $game['gamename'] ?></h2>
<h2>Cost: <?= $game['cost'] ?></h2>
<h2>Composition: <?= $game['composition'] ?></h2>
<h2>Status: <?= $game['status'] ?></h2>
<a href="javascript:history.back()"> <- Back </a>
</body>
```

index.php

```
<?php
    session_start();
    if ($_SESSION['user']) {
        header('Location: profile.php');
    }
?>
<!doctype html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Authorization and add new employee</title>
    <link rel="stylesheet" href="css/main.css">
</head>
<body>
    <!--Форма авторизации-->
    <form action="config/signin.php" method="post">
        <label>Login</label>
        <input type="text" name="login" placeholder="Enter your
login">
        <label>Password</label>
        <input type="password" name="password"
placeholder="Enter your password">
        <button type="submit">Log in</button>
        <p>
            Don't have an account? - <a href="/add.php">Add new
employee</a>!
```

```
        </p>
        <?php
            if ($_SESSION['message']) {
                echo '<p class="message"> ' .
$_SESSION['message'] . ' </p>';
            }
            unset($_SESSION['message']);
        ?>
    </form>
</body>
</html>
```

profile.php

```
<?php
session_start();
if (!$_SESSION['user']) {
    header('Location: /');
}
?>
<!doctype html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Profile</title>
    <link rel="stylesheet" href="css/main.css">
</head>
<body>
<form>
    <img src="<?= $_SESSION['user']['avatar'] ?>" width="100"
alt="">
    <h2 style="margin: 10px 0;"><?=
$_SESSION['user']['full_name']?></h2>
    <a href="#"><?= $_SESSION['user']['email'] ?></a>
    <a href="config/logout.php" class="logout">Log out</a>
```

```
</form>
```

add.php

```
<?php
    session_start();
    if ($_SESSION['user']) {
        header('Location: profile.php');
    }
?>
<!doctype html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Authorization and add new employee</title>
    <link rel="stylesheet" href="css/main.css">
</head>
<body>
<form action="config/signup.php" method="post"
enctype="multipart/form-data">
    <label>Full name</label>
    <input type="text" name="full_name" placeholder="Enter your
full name">
    <label>Email</label>
    <input type="email" name="email" placeholder="Enter your
email">
    <label>Profile picture</label>
    <input type="file" name="avatar">
    <label>Position</label>
    <input type="text" name="position" placeholder="Enter your
position">
    <label>Password</label>
    <input type="password" name="password" placeholder="Enter
your password">
    <label>Password confirmation</label>
```

```
    <input type="password" name="password_confirm"
placeholder="Enter your password again">
    <button type="submit">Add new employee!</button>
    <p>
        Do you already have an account? - <a href="/">Log
in</a>!
    </p>
    <?php
        if ($_SESSION['message']) {
            echo '<p class="message"> ' . $_SESSION['message'] .
'</p>';
        }
    unset($_SESSION['message']);
    ?>
</form>
</body>
</html>
```

update.php

```
<?php
require_once 'config/conn.php';
$game_id = $_GET['id'];
$game = mysqli_query($conn, "SELECT * FROM `games` WHERE `id` =
'$game_id'");
$game = mysqli_fetch_assoc($game);
?>
<!doctype html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Update Game</title>
    <p>
        <font size="18" color="#f5f5f5" face="Geneva">Board game
rental club</font>
    </p>
```

```
</head>
<style>
    body {
        background: #c7b39b url(image.jpg); /* Цвет фона и путь к файлу */
        color: whitesmoke; /* Цвет текста */
        background-size: 100%;
    }
    th, td {
        padding: 10px;
    }
    th {
        background: #ee00ff;
        color: #fff;
    }
    td {
        background: #f9abff;
    }
</style>
<body>
<h3>Update Game</h3>
<form action="vendor/update.php" method="post">
    <input type="hidden" name="id" value="<?= $game['id'] ?>">
    <p>Game name</p>
    <input type="text" name="title" value="<?= $game['gamename'] ?>">
    <p>Composition</p>
    <textarea name="composition"><?= $game['composition'] ?></textarea>
    <p>Cost</p>
    <input type="number" name="cost" value="<?= $game['cost'] ?>">
    <p>Status</p>
```

```
    <input type="boolean" name="status" value="<?=
$game['status'] ?>"> <br> <br>

    <button type="submit">Update</button><br> <br>

    <a href="javascript:history.back()"> <- Back </a>

</form>

</body>

</html>
```

ДОДАТОК Д ОПИС ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

1. Загальні відомості

1.1 Позначення і найменування програми

Найменування: Програмне забезпечення вебсайту для клубу прокату настільних ігор.

Позначення: «Вебсайт для клубу прокату настільних ігор.»

1.2 Програмне забезпечення, необхідне для функціонування програми

Вебсайт для клубу прокату настільних ігор як і більшість вебсайтів є кросплатформним, тому не пред'являє жодних вимог до апаратної платформи користувача. Основною необхідною вимогою для функціонування продукту є наявність сучасного браузера (Mozzilla FireFox, Opera, Google Chrome, Microsoft Edge) та доступ до мережі Інтернет.

1.3 Мови програмування

Програмне забезпечення вебсайту для клубу прокату настільних ігор з використанням мови програмування PHP, мови програмування JavaScript, мови тегів HTML, мови стилю сторінок CSS, а також проєктування та створення баз даних за допомогою вільної системи керування реляційними базами даних MySQL.

2. Функціональне призначення

2.1 Класи розв'язуваних завдань та призначення програми

Функціональним призначенням програми є надання співробітникам клубу настільних ігор можливості вести облік прокату ігор через вебсайт замість паперових журналів

Програма повинна мати функцію авторизації та розподілу прав доступу на адміністратора, співробітника та неавторизованого користувача – відвідувач.

З правами доступу користувача як адміністратора, програма повинна надати можливість виконувати такі функції:

- перегляд списку співробітників;
- додавання співробітника;
- редагування інформації про співробітника;
- авторизовуватись;
- виходити з облікового запису.

З правами доступу користувача як співробітника, програма повинна надати можливість виконувати такі функції:

- редагувати, додавати та видаляти ігри;
- брати заставну вартість;
- видавати в прокат ігри та забирати їх з прокату;
- змінювати логічний статус гри.

При неавторизованому вході в систему програма повинна надати можливість виконувати такі функції:

- перегляд списку ігор;
- перегляд інформації про конкретну гру.

2.2 Функціональні обмеження

Для роботи з програмним забезпеченням вебсайту необхідно мати підключення до мережі Інтернет. Користування вебсайтом за допомогою застарілих версії браузерів не можливо. Під час спроби завантаження вебсайту з’явиться повідомлення «Ваш браузер застарів».

3. Опис логічної структури

3.1 Структура програми

Програмне забезпечення вебсайту для клубу прокату настільних ігор має три складових частини:

- Клієнтська частина - це графічний інтерфейс. Графічний інтерфейс відображається в браузері.

- Серверна частина - це скрипт на сервері, що обробляє запити користувача.

- База даних - програмне забезпечення на сервері, що забезпечує зберігання даних та їх видачу в момент запиту.

3.2 Алгоритм програми

Програмне забезпечення вебсайту для клубу прокату настільних ігор має технологію «клієнт-сервер». Клієнтом виступає браузер, а сервером - вебсервер. Логіка вебсайту розподілена між сервером і клієнтом, зберігання даних здійснюється на сервері, обмін інформацією відбувається по мережі.

Робота вебсайту представлена взаємодією між клієнтською та серверною частинами, а також базою даних. Алгоритм роботи програми є наступним:

1) Користувач взаємодіє з вебсайтом через браузер, переходить за посиланням;

2) Браузер через Інтернет відсилає HTTP-запити вебсерверу;

3) Вебсервер обробляє запит, викликаючи PHP-скрипт;

4) PHP-скрипт звертається до бази даних (якщо це потрібно) та відсилає результат;

5) Браузер показує отриманий результат у вигляді чергової вебсторінки.

3.3 Використовувані методи

Програмне забезпечення вебсайту для клубу прокату настільних ігор розроблено з використанням мови програмування PHP, мови програмування JavaScript, мови тегів HTML, мови стилю сторінок CSS, а також проєктування та створення баз даних за допомогою вільної системи керування реляційними базами даних MySQL.

4. Технічні засоби, що використовуються

Програмне забезпечення вебсайту для клубу прокату настільних ігор передбачає використання на персональних комп'ютерах, але не пред'являє ніяких вимог до апаратної платформи, тому працює на будь-якій операційній системі.

Даний програмний продукт потребує від комп’ютера, на якому він буде встановлений, наступних характеристик, які треба розглядати як мінімальні:

- Операційна система Microsoft Windows Vista і вище;

- Процесор Intel Pentium/Celeron 1800 МГц і вище;

- Оперативна пам'ять 1 Гбайт і вище;

- Жорсткий диск близько 1 Гбайт(система + дані);

Для роботи з програмним забезпеченням необхідна наявність сучасного браузер (Mozzilla FireFox, Opera, Google Chrome, Microsoft Edge) та підключення до мережі Інтернет.

Швидкість та якість роботи програмного забезпечення вебсайту для клубу прокату настільних ігор на конкретному комп'ютері залежить також від характеристик окремих його компонентів.

5. Виклик та завантаження

Виклик та завантаження програмного забезпечення вебсайту відкривається у вікні браузера при зверненні в адресному рядку до URL-адреси розміщення ресурсу.

Запуск програмного забезпечення вебсайту для клубу прокату настільних ігор іншими способами неможливий.

6. Вхідні дані

Введення інформації здійснюється за допомогою клавіатури. Всі данні вводяться вручну у відповідні форми. Дані зберігаються в базі даних на сервері, що знаходиться на локальному диску комп’ютера. Вхідні дані:

- інформація з текстових полів;
- зображення.

7. Вихідні дані

- сторінки мовою HTML;