

Design Manual
Central Games Platform



Kyle Hennessy
C00227463

Table of Contents

1.Abstract	4
2. Project Description	5
2.1 Video Games	5
2.2 Casino Games	5
3.Project Plan	6
4.Technologies	7
4.1 Bootstrap 4	7
4.2 jQuery	7
4.3 Azure SQL Database	7
4.4 Stripe Payment Gateway API	7
4.5 PayPal Payouts API	8
4.6 Entity Framework Core	8
4.7 ASP.NET Core Identity API	8
4.8 Azure App Service	8
5.System Architecture	9
6.Domain Model	10
7.Class Diagram	11
8.System Sequence Diagrams	12
8.Entity-Relationship Diagram	20
9.User Interface flow diagram	21
10.Wireframes	22

Table of Figures

Figure 1 – Project plan	6
Figure 2 – System Architecture diagram	9
Figure 3 - Domain model of the system	10
Figure 4 – Class diagram of the system	11
Figure 5 – Login system sequence diagram	12
Figure 6 – Logout system sequence diagram	13
Figure 7 – Register system sequence diagram	14
Figure 8 – View Catalogue system sequence diagram	15
Figure 9 – CRUD Game system sequence diagram	16
Figure 10 – CRUD Favourite Games system sequence diagram	17
Figure 11 – Purchase Cart system sequence diagram	18
Figure 12 – View Transactions system sequence diagram	19
Figure 13 – Entity relationship diagram for the SQL database	20
Figure 14 – User interface flow diagram	21
Figure 15 – Login/Register page	22
Figure 16 – Home Page	22
Figure 17 – Catalogue page	23
Figure 18 – Store page	23
Figure 19 – Shopping cart page	24
Figure 20 – Purchase Page	24
Figure 21 – Review Purchase Page	25
Figure 22 – Order Confirmation Page	25
Figure 23 – Payout page	26
Figure 24 – Review Payout page	26
Figure 25 – Game page	27
Figure 26 – Transaction History page	27
Figure 27 – Create/Update Game Page	28

1. Abstract

The purpose of the Central Games Platform project is to develop a web based application intended for Windows 10 and Android that acts as a hybrid of both an e-commerce store to purchase and play video games, and an online casino, which is intended to be a one stop shop for gamers and casino players alike. Users will be able to purchase and play the latest and greatest blockbuster video game titles, or purchase casino passes to participate in casino games for a chance of winning real world currency. Video games can either be playable from the browser, or be installed to the supported device, depending on the game. Video games that are available on both platforms will be accessible to the user at no additional cost. The application will allow users to transfer winnings from casino games to a registered PayPal account. Impulsive gambling is a huge ethical issue, so a limit of 10 casino passes purchasable per day will be imposed on the user.

Central Games Platform is designed with a business to consumer/business to business approach. Users purchase games from the platform. Publishers of these games will receive the revenue the game has generated every month with a 15% cut. Casino games will be developed in-house. 100% of revenue from casino games goes straight to Central Games Platform.

2. Project Description

CENTRAL Games Platform is an application aimed at Games Players and online casino players alike, where players can manage and play owned games or frequently played casino games. For traditional games, an ecommerce storefront will host a catalogue of games. Casino games will have its own section that is entirely separate from the traditional games catalogue and will only be available to users who have verified that they are age 18+. This project has a desktop client and a mobile client which aims to seamlessly tie purchases of cross platform games as well as displaying details on owned/played games that are not supported on the current device platform. The dashboard will seamlessly tie both traditional and casino games together. Info and statistics on both types of games will be displayed to the player. The option will also be there to omit any mention of casino games to Games Players and vice versa. The unique selling point of this application is to provide an all in one games platform that combines traditional games with casino games which have a huge overlap in users which has never been done before possibly due to the stigma involved around gambling.

The project's two main aspects of development are video game licencing and delivery, and the online casino game delivery system.

2.1 Video Games

Traditional games are games that are not intended to be played for money but instead are to be played recreationally. These can range from very simple and lightweight games to high fidelity performance intensive games. Given the drastic differences between these games, some of them may not be suitable to be played directly in the application and will need to be installed on to the players device, although more simpler games can be played directly from the dashboard.

The games storefront will contain a comprehensive catalogue of all the games offered. Each game featured on the catalogue will have their own store page where details about the game will be available such as in-game screenshots, a gameplay description, supported platforms and pricing information. A game purchased by a customer will be added to their account and can then be played directly in the app or downloaded at any point from their library of games.

2.2 Casino Games

Casino games are games that reward monetary winnings to players. Due to the nature of offering monetary rewards, these must be played directly in the app to prevent any potential cheating from occurring. The logic in these games must be processed on the server to ensure that all results are fair and were not tampered with.

The casino section will contain a catalogue of games where each game will also have their own store page providing details about it. These games are only playable by game pass holders. A casino game pass can be used for any casino game. Game passes are one time use and are removed from a player's account once used. Casino games offer the chance of winning real-world currency. Winnings from these games are added to the player account which can then be used to purchase more game passes, traditional games or can be transferred to the players bank account for a transfer fee which will be a small percentage of the amount they are transferring. To prevent the occurrence of compulsive gambling, there will be a limit to the number of passes that can be purchased every day.

3. Project Plan

The project plan or timeline for the CENTRAL games platform details all of the proposed tasks and milestones of the project. Any additional time remaining will be spent on refactoring any outdated information in previous documents.

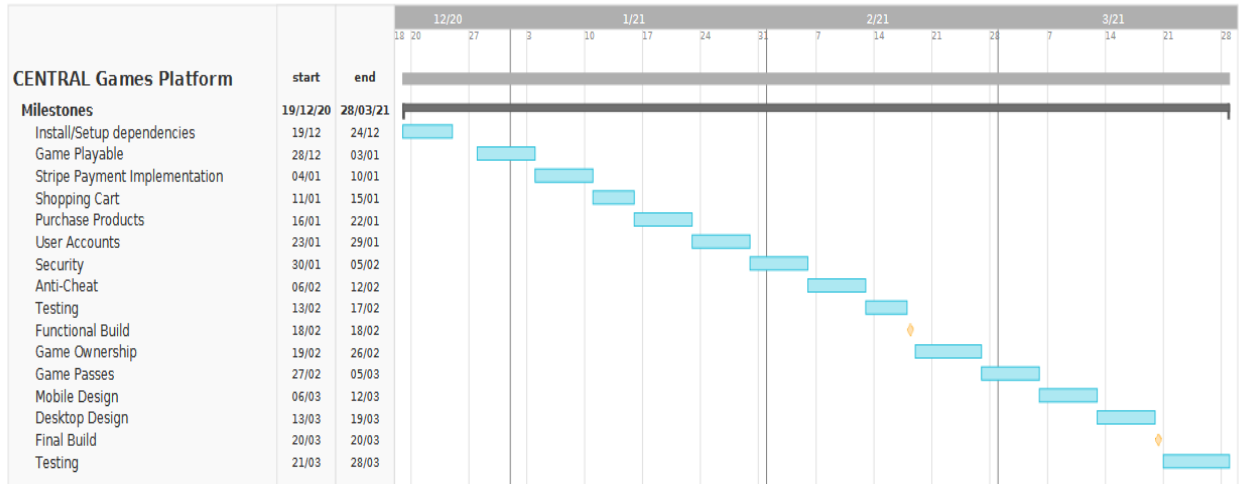


Figure 1 – Project plan

4. Technologies

4.1 Bootstrap 4

CENTRAL Games Platform is a web application first and foremost meaning that most if not all design choices and user interfaces will be created using a combination of HTML, CSS and JavaScript. Additionally, because both mobile and desktop platforms are going to be targeted, a mobile first approach to design will be taken which can then be scaled up to larger screens without the need for a major redesign of the interface. A mobile first approach will ensure that the application will function on all screen sizes and increase the level of user experience (UX) on all devices. This is due to how restrictive the screen sizes of mobile devices are so when it is designed for mobile platforms first, it is guaranteed to be usable on a larger screen without minimal refactoring of code. Bootstrap 4 is a HTML, CSS and JavaScript web development framework that provides the libraries necessary to design user interfaces that scale depending on the user's screen size. Additionally, it will make it faster for first time users to load the web page as Bootstrap 4 is used everywhere across the internet and will likely already be loaded into the users' browser cache.

4.2 jQuery

JavaScript will need to be used to design intuitive and functional user interfaces for the CENTRAL Games Platform. JavaScript code will significantly increase the development time of user interfaces, leaving less time to spend on more critical functionality. Instead, jQuery will be used as it is a JavaScript library that aims to give developers a "write less do more" design philosophy. jQuery makes it possible to complete commonly needed tasks that would usually need many lines of code and hours of work and instead makes them into methods that can be called with one line of code. jQuery allows for easy manipulation of the DOM, CSS manipulation, effects, animations and HTML event methods.

4.3 Azure SQL Database

An SQL database will be required to store user accounts, game store pages, order information and many more pieces of data associated with CENTRAL Games Platform. Additionally, the SQL database must be hosted online to ensure that all users will be able to send and retrieve data for the application to be able to function as intended. Azure SQL Database is a solution offered to developers that host their application using the Azure App Service, which is where CENTRAL Games Platform is being hosted on. The tier being used is the basic SQL database which dramatically lowers the price to just €5 per month as the system operates on limited resources which sacrifices speed and performance but unfortunately other pricing schemes would go over the budget of this project, although it will be possible to upgrade to a faster performing database later on if required.

4.4 Stripe Payment Gateway API

A payment gateway provider is needed for this application as a method to receive money from the customers is the main way that CENTRAL Games Platform will generate revenue. Stripe Payment Gateway will be implemented as it handles all of the transactions and communicates with the user's bank account without intervention from business owners. Additionally, there is no set up fee and it leaves full freedom up to how their API is used and presented on forms.

4.5 PayPal Payouts API

The primary selling point to the application will be the ability for the players to transfer winnings to their bank account too so this will be necessary to implement. This will be achieved through PayPal's payouts API. All that is required from the user to transfer winnings to their PayPal account is their email address.

4.6 Entity Framework Core

Entity Framework core enables developers to take a code first approach to database development. Data can be defined in classes made in C# where they can then be migrated to a database without the need for writing any SQL queries. It also allows for data access and modification using this same code first approach.

4.7 ASP.NET Core Identity API

ASP.NET Core Identity API is an API that allows for easy implementation of user authentication and authorization in any project. User information is also stored securely, which makes it so that the developer does not have to worry about hashing or salting passwords. A number of tools are also available to implement role based authorization, which will be useful to enable admin accounts on the platform.

4.8 Azure App Service

The azure cloud app service will be what the live version of the application will be run on. It will be hosted with its own domain name so that anyone with an internet connection and a browser will be able to visit the web page and benefit from any of the features offered on the site.

5. System Architecture

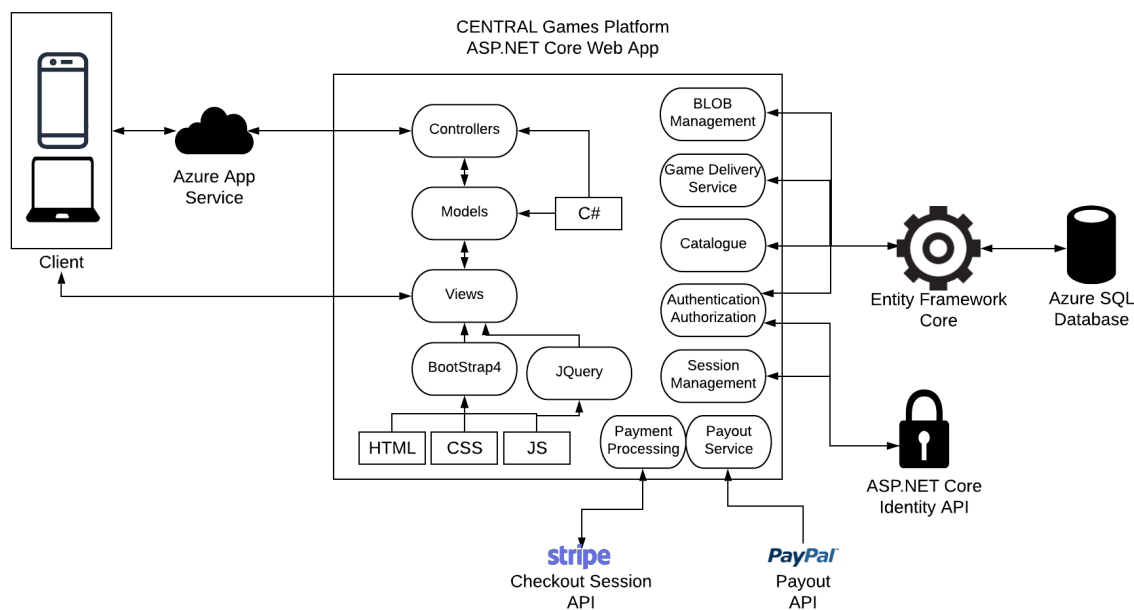


Figure 2 – System Architecture diagram

The system architecture diagram goes over the overall architecture of the project and the technologies incorporated into it. The main application is an ASP.NET Core Web App that is hosted on the Azure App Service. The client makes all interactions through the Azure App Service which in turn interacts with the various controllers to manipulate and return data to the client. The models and controllers are made using the C# programming language. The view is the only thing that the client can physically see and interact with directly. The views are built entirely using Bootstrap 4 and jQuery which are HTML, CSS and JavaScript libraries. BLOB (Binary Large Object) Management, Game Delivery System, Catalogue, and Authentication/Authorization are all processes that require the use of the Azure SQL Database. Authentication/Authorization require the use of the ASP.NET Core Identity API for user verification and security. Payment Processing and Payout Service will be made possible through the use of the Stripe Payment Gateway API.

6.Domain Model

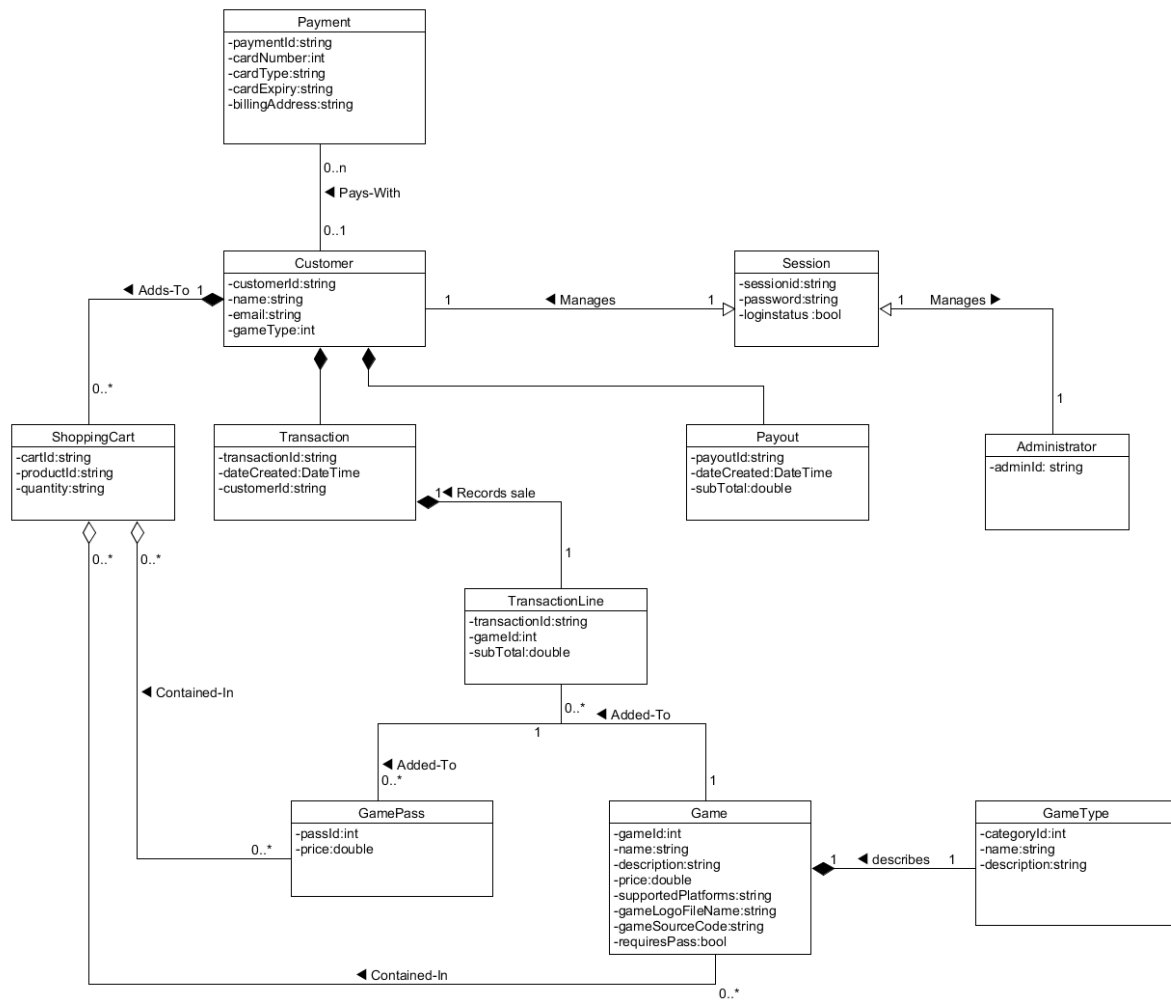


Figure 3 - Domain model of the system

The domain model is intended as a starting point for coding and is subject to change throughout the development of the project. Object relationships and members may vary greatly between now and the development of the project.

7. Class Diagram

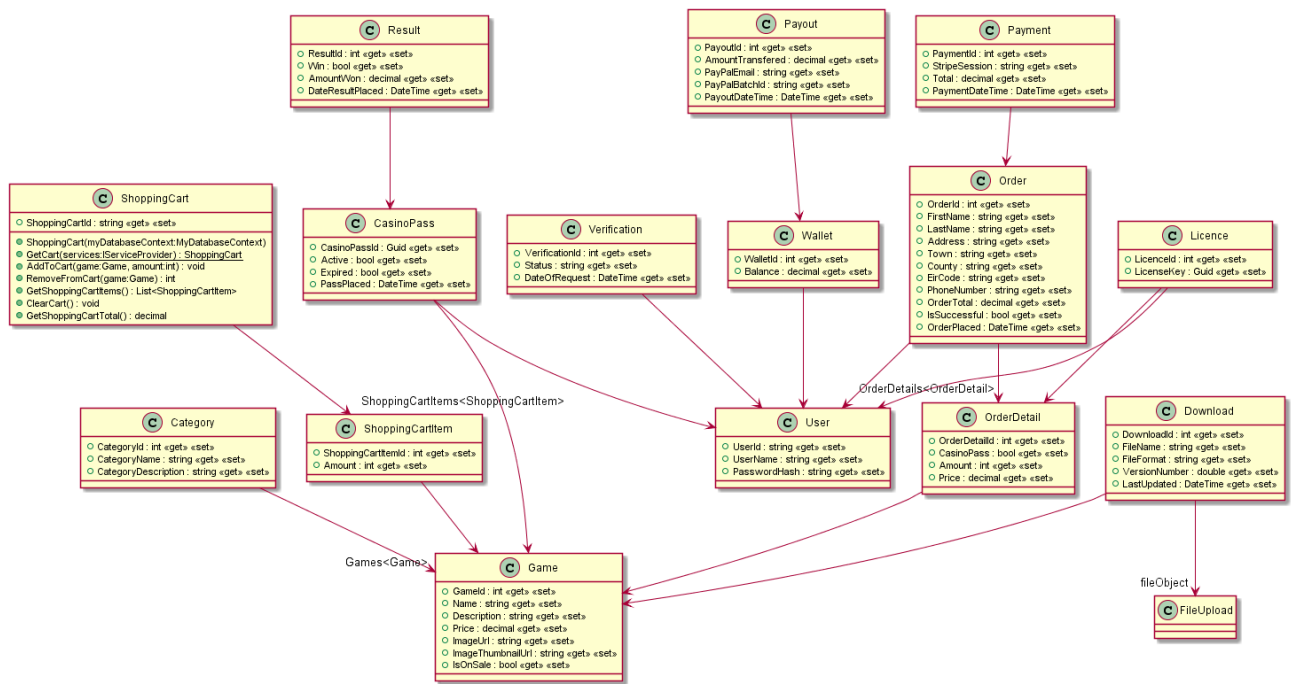


Figure 4 – Class diagram of the system

This class diagram was created after the final iteration of the project. This class diagram is created based off of the models for the project, as the entire project depends on this data structure and the relationships between the classes.

8. System Sequence Diagrams

The system sequence diagrams represent a high level overview of how the various actors in the system interact with each other. These were designed before the project had any code implemented to give an idea of how to approach the coding phase. As a result, this is not a 1:1 reflection of how the system will behave.

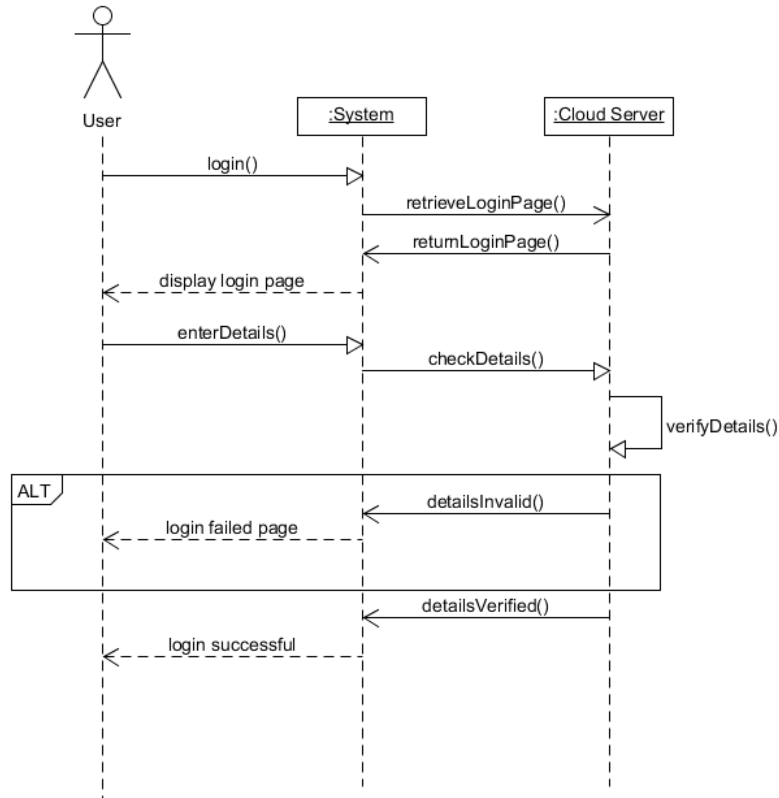


Figure 5 – Login system sequence diagram

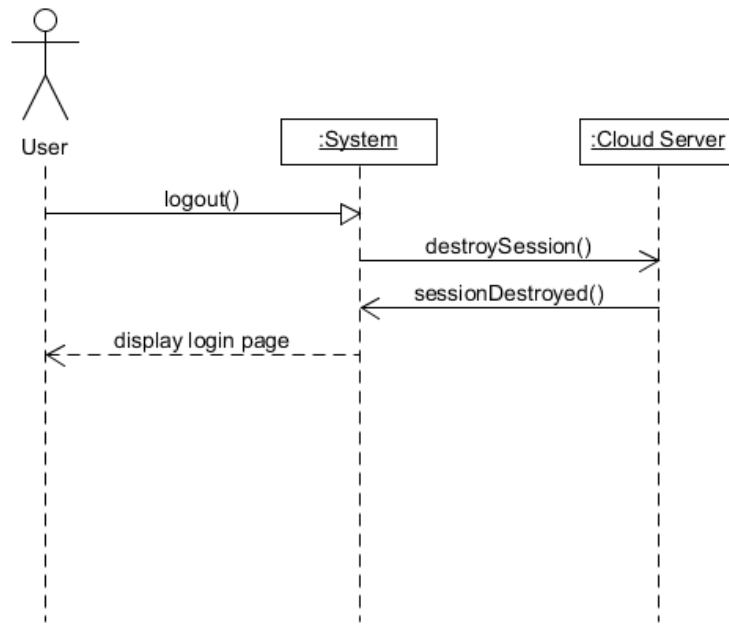


Figure 6 – Logout system sequence diagram

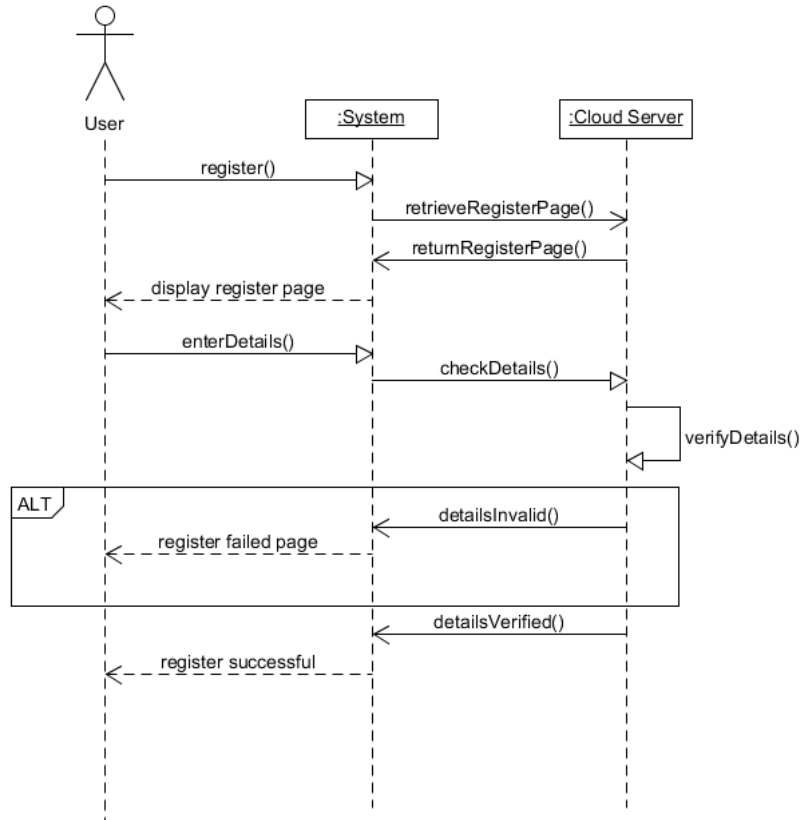


Figure 7 – Register system sequence diagram

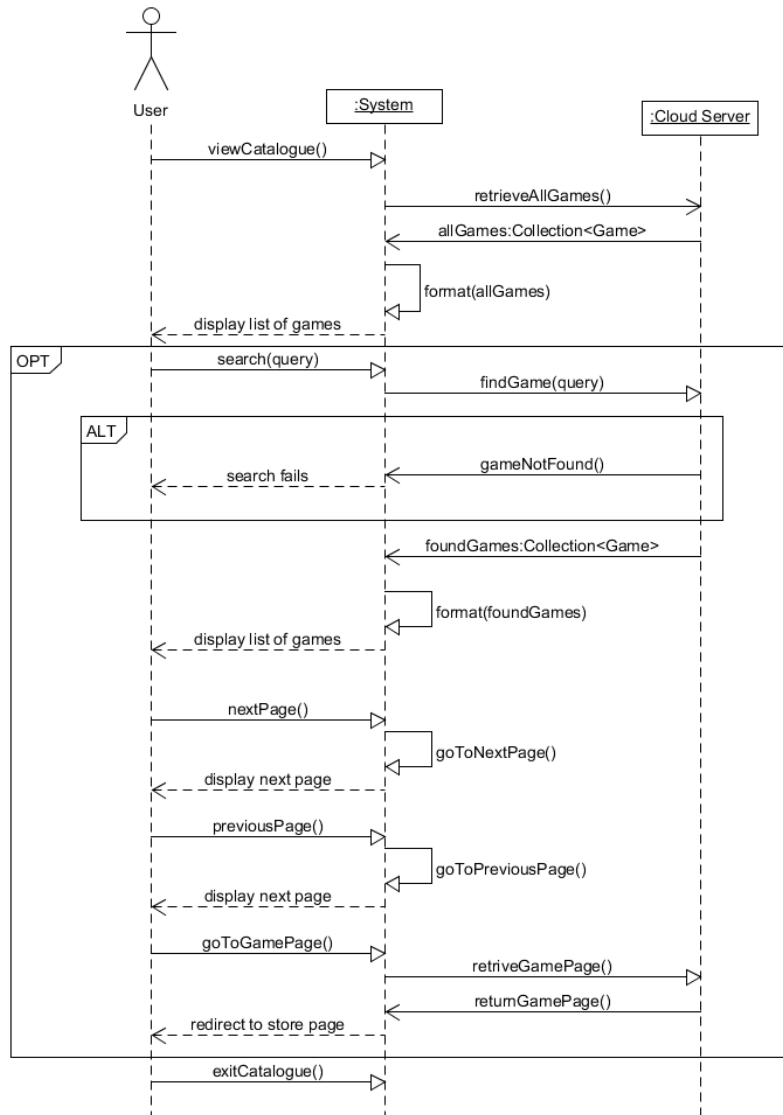


Figure 8 – View Catalogue system sequence diagram

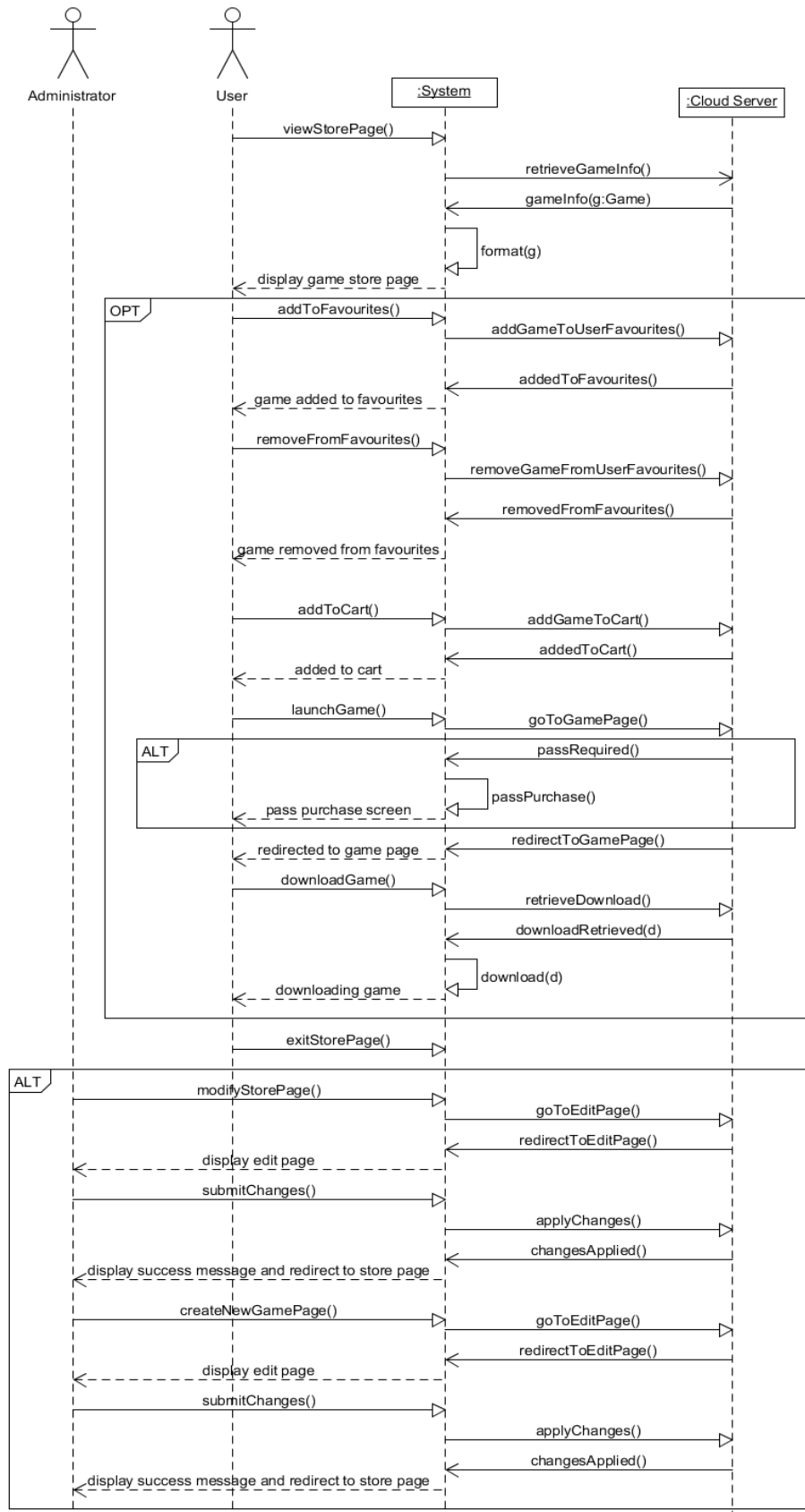


Figure 9 – CRUD Game system sequence diagram

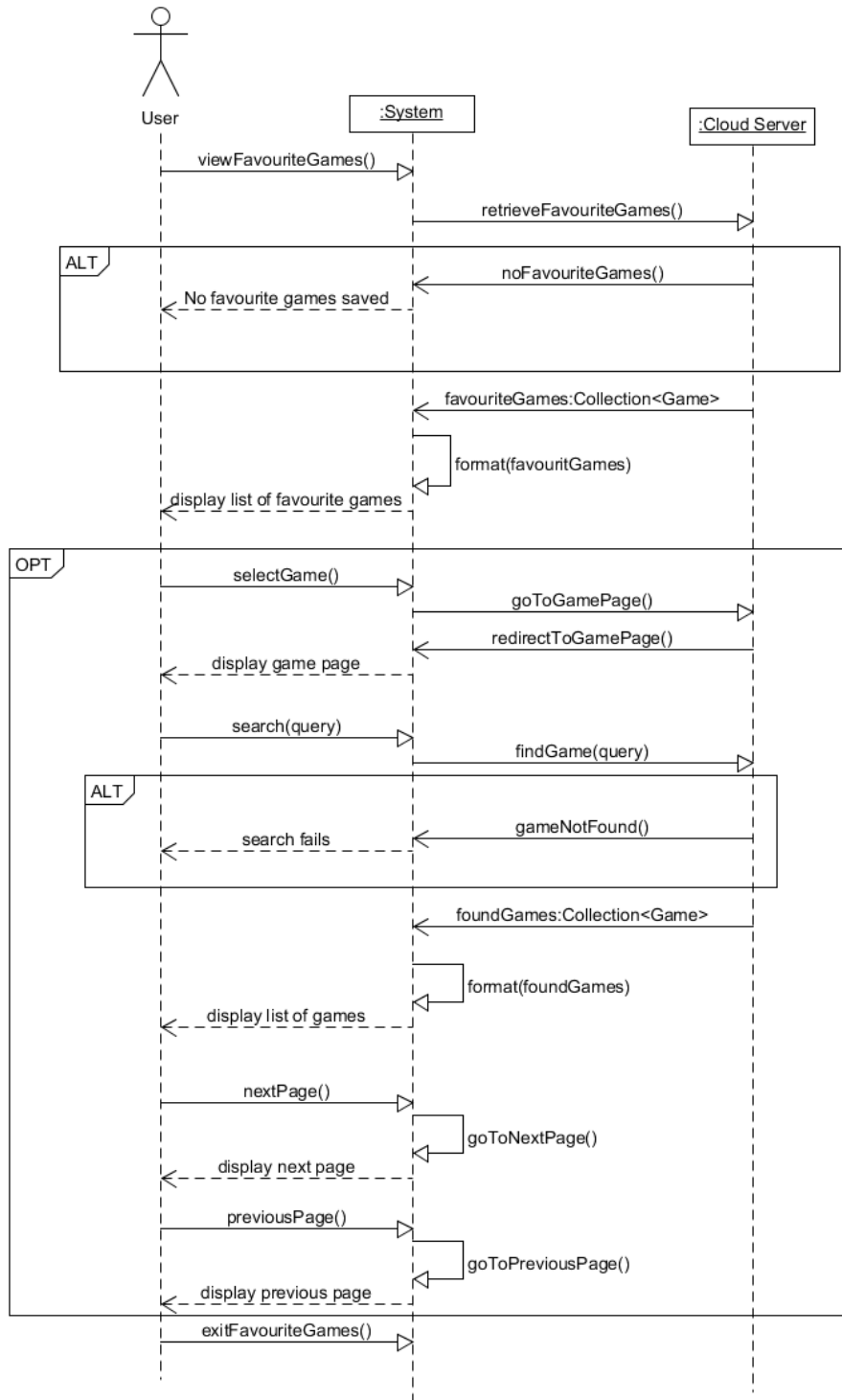


Figure 10 – CRUD Favourite Games system sequence diagram

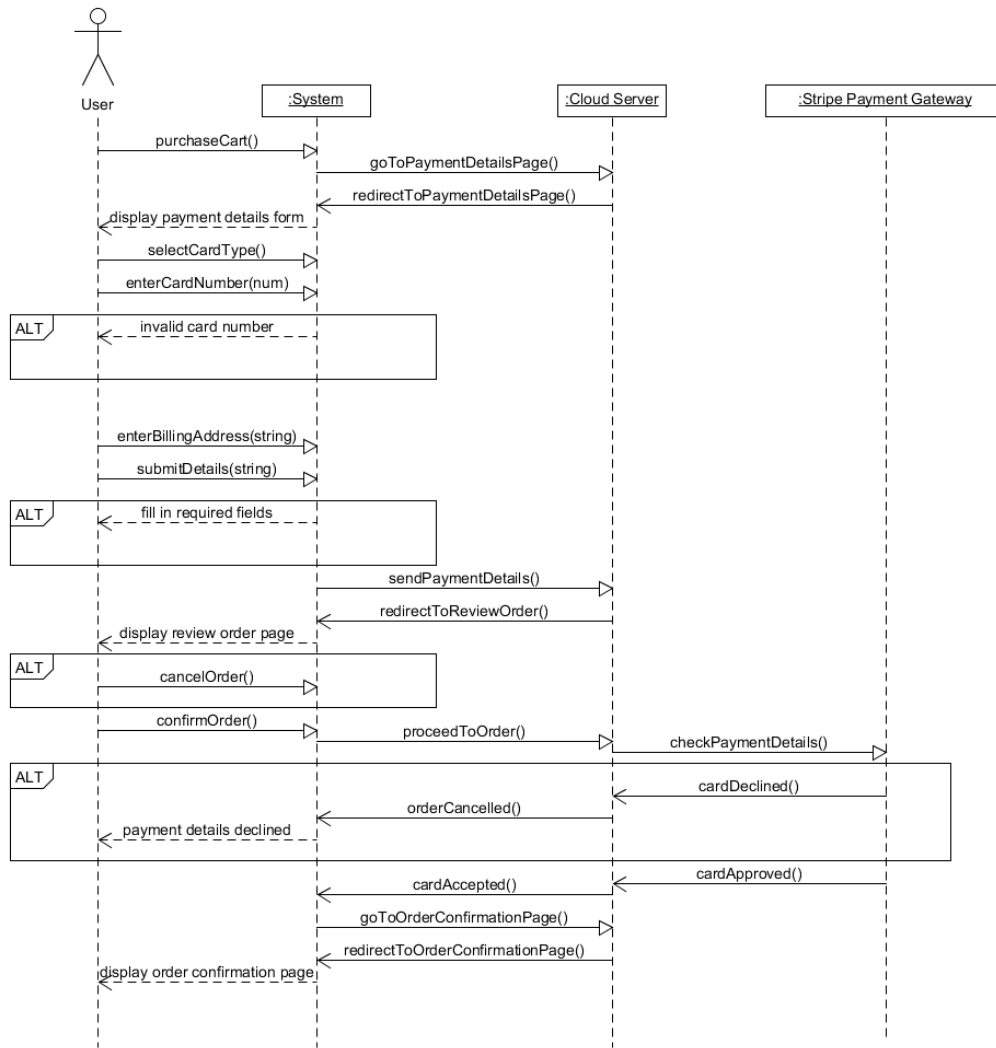


Figure 11 – Purchase Cart system sequence diagram

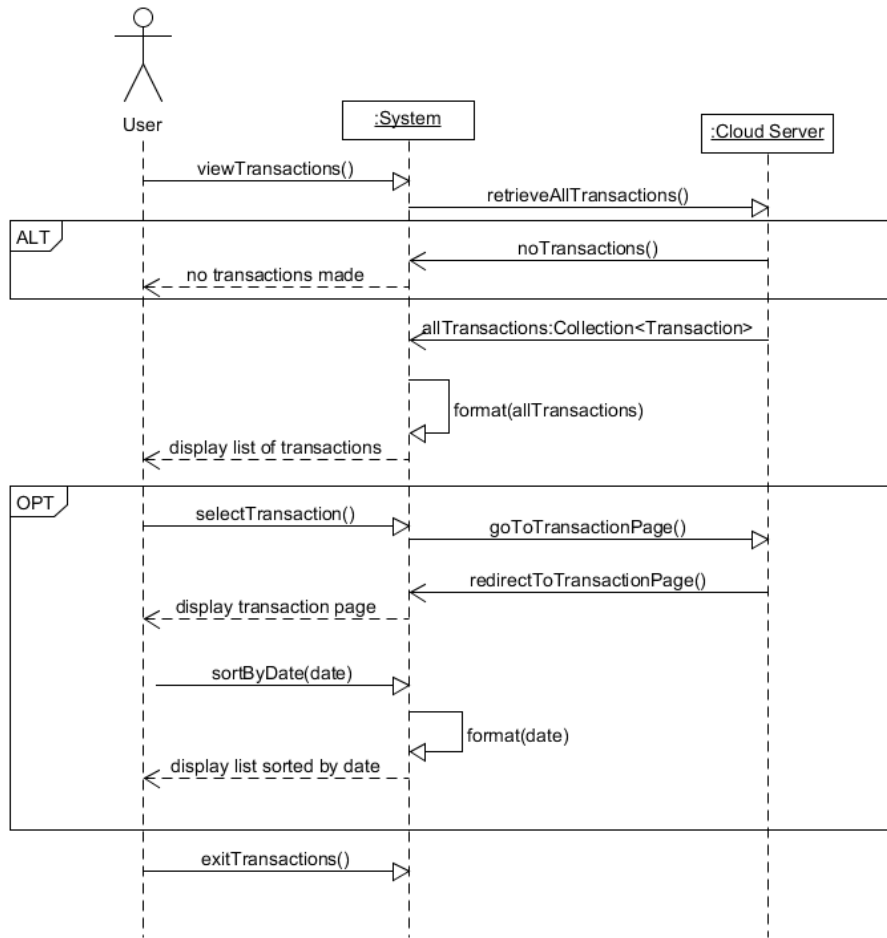


Figure 12 – View Transactions system sequence diagram

8. Entity-Relationship Diagram

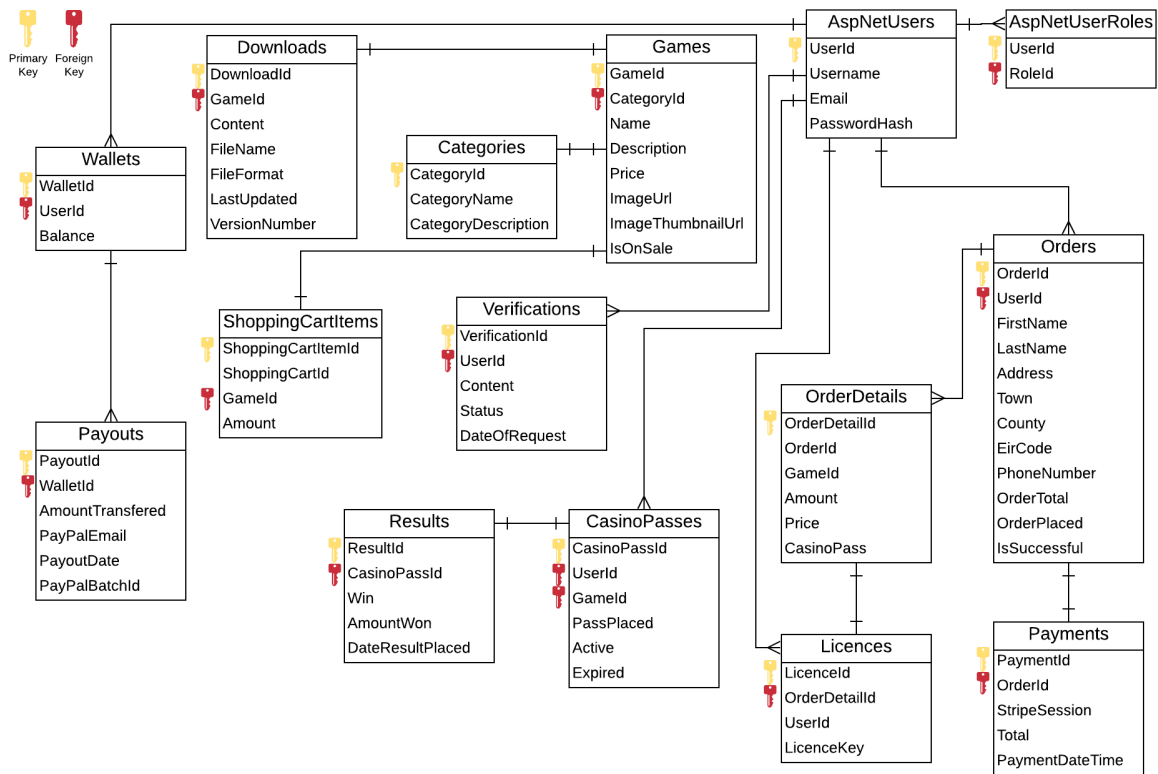


Figure 13 – Entity relationship diagram for the SQL database

9. User Interface flow diagram

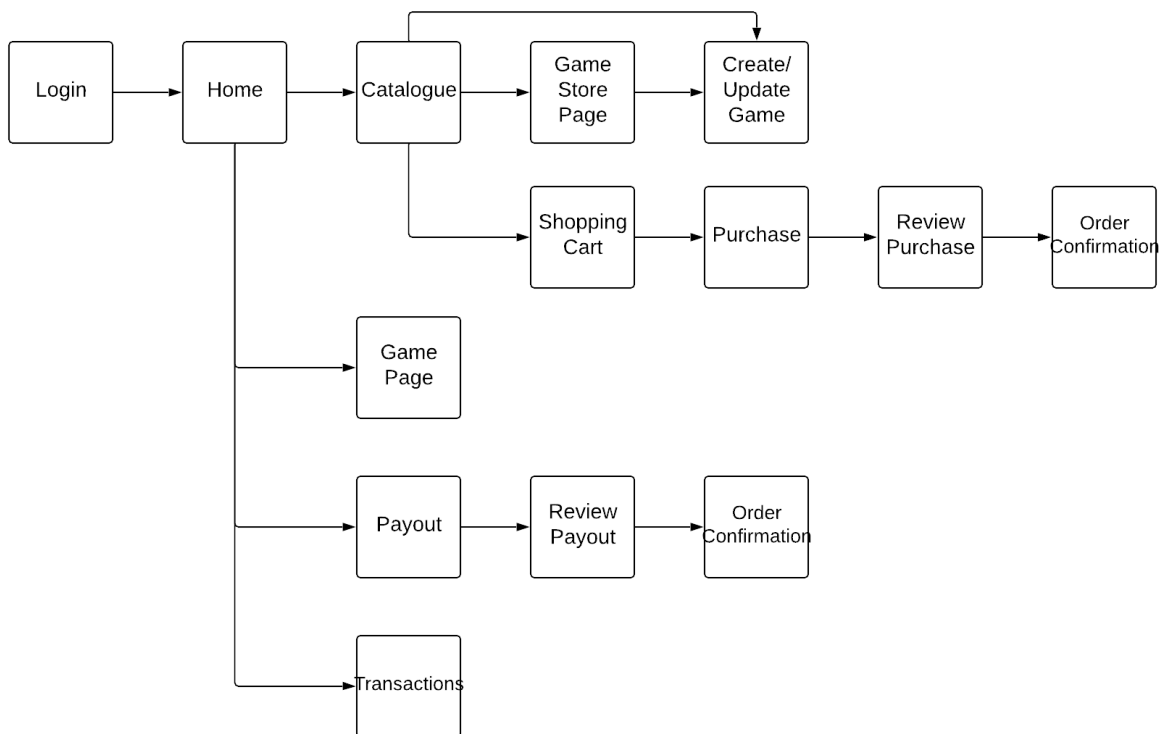


Figure 14 – User interface flow diagram

The user flow diagram was created prior to any coding that took place and acted as a rough guideline for the various screens to include in the project. The final project may vary slightly from this UI flow diagram

10. Wireframes

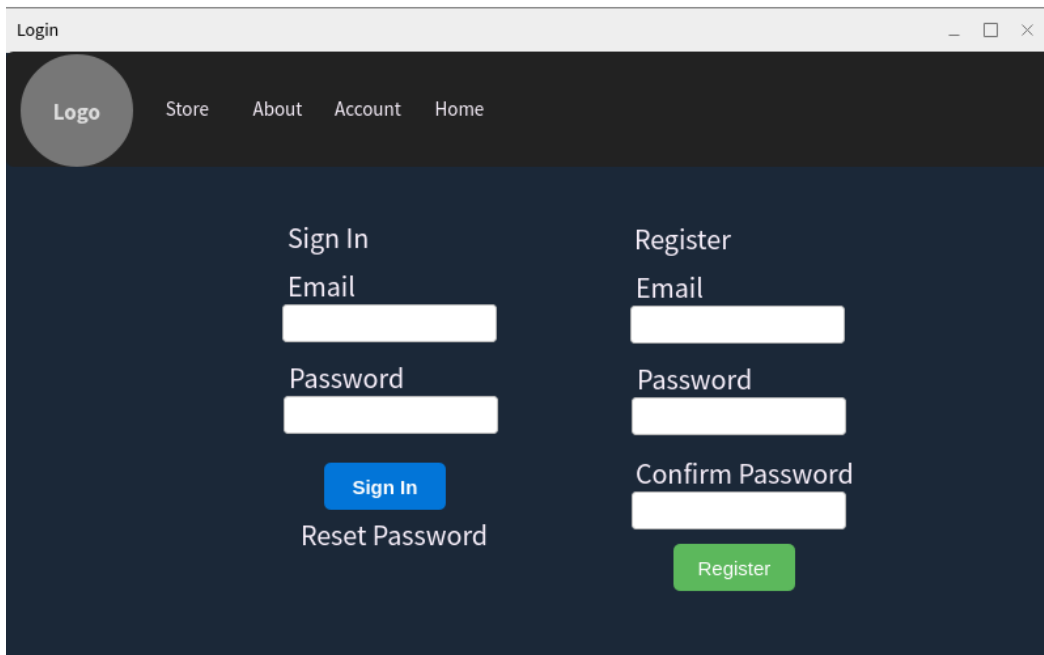


Figure 15 – Login/Register page

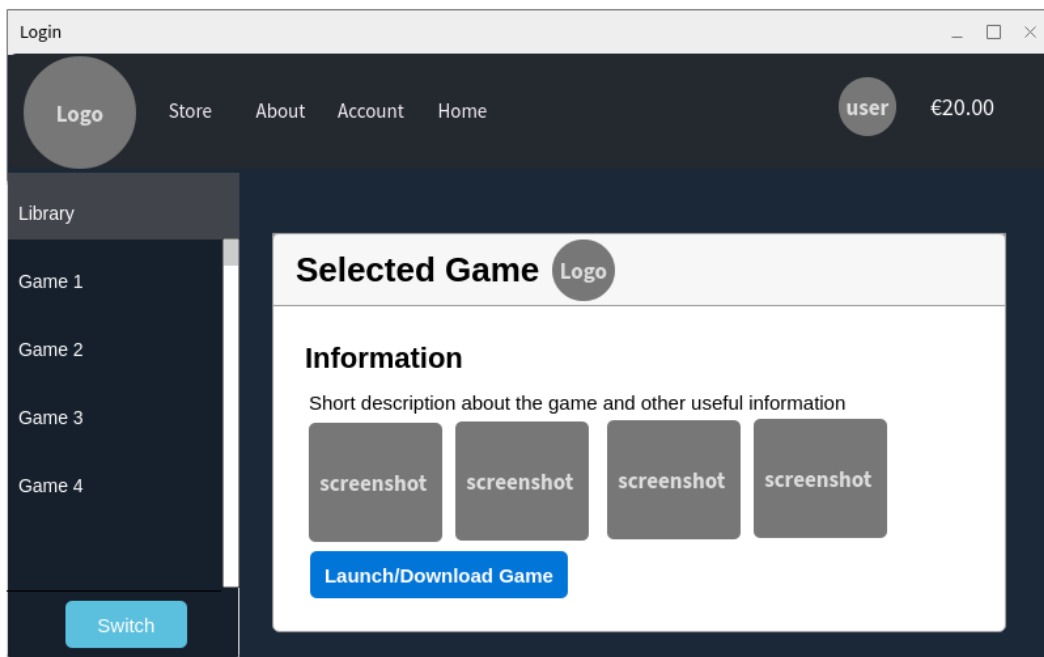


Figure 16 – Home Page

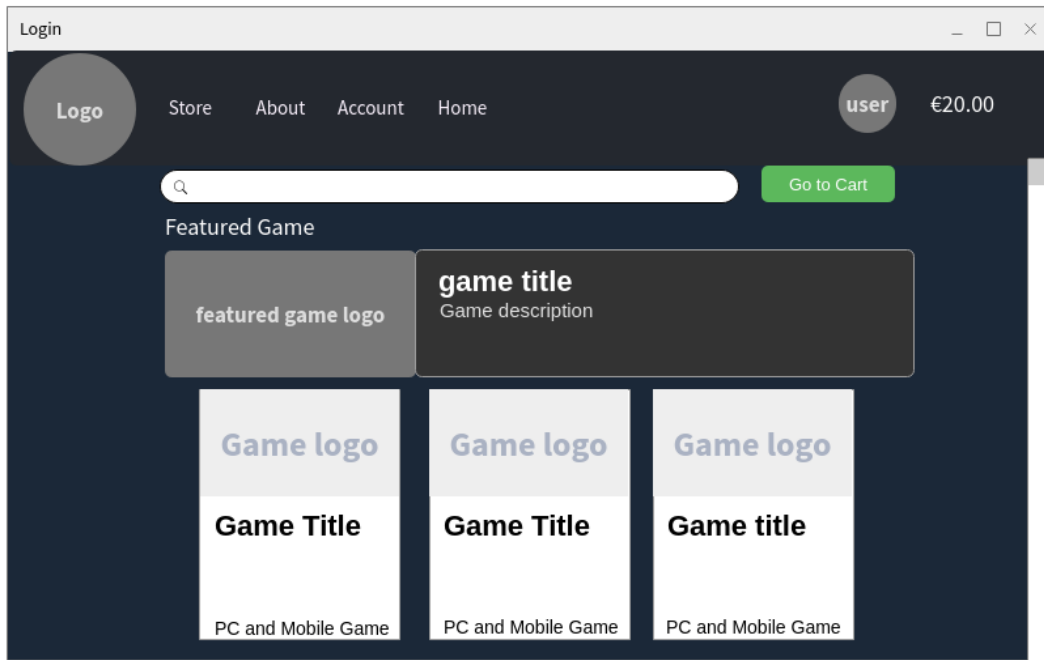


Figure 17 – Catalogue page

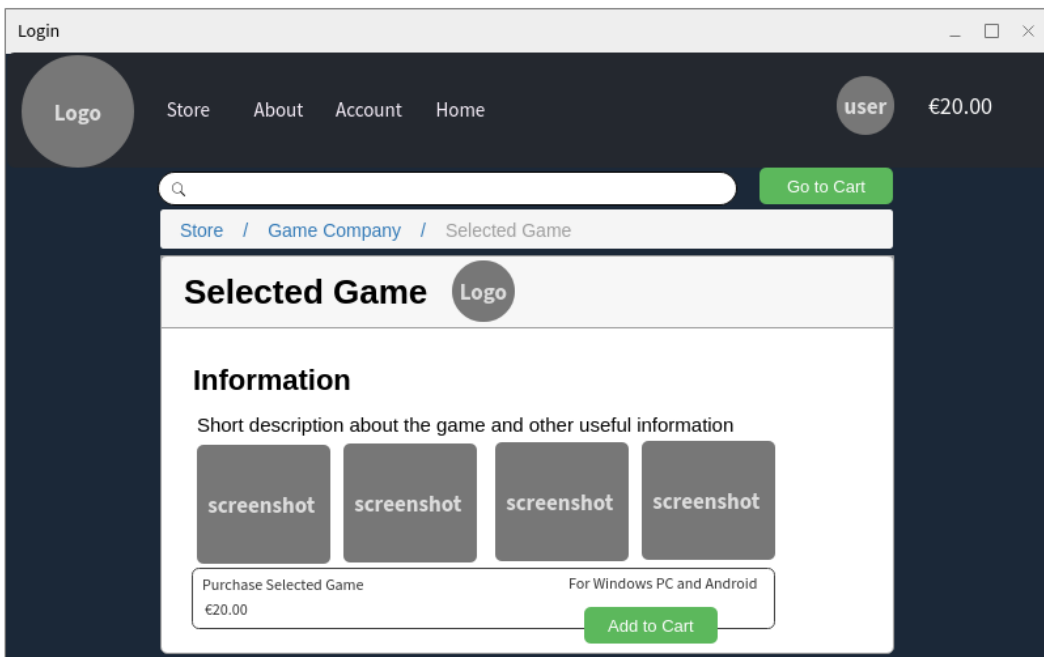


Figure 18 – Store page

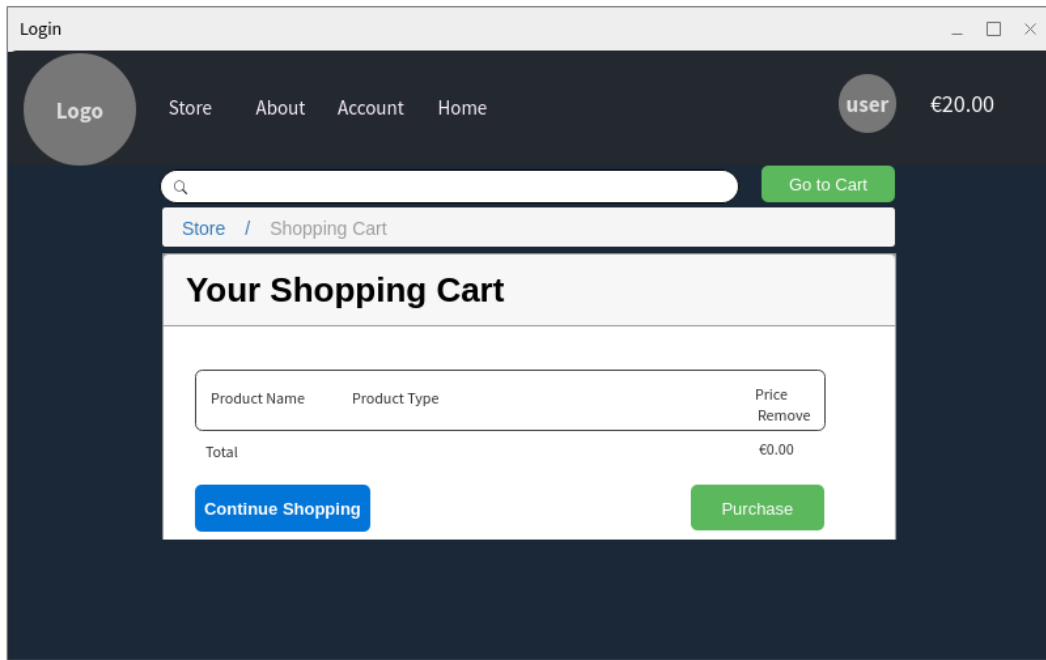


Figure 19 – Shopping cart page

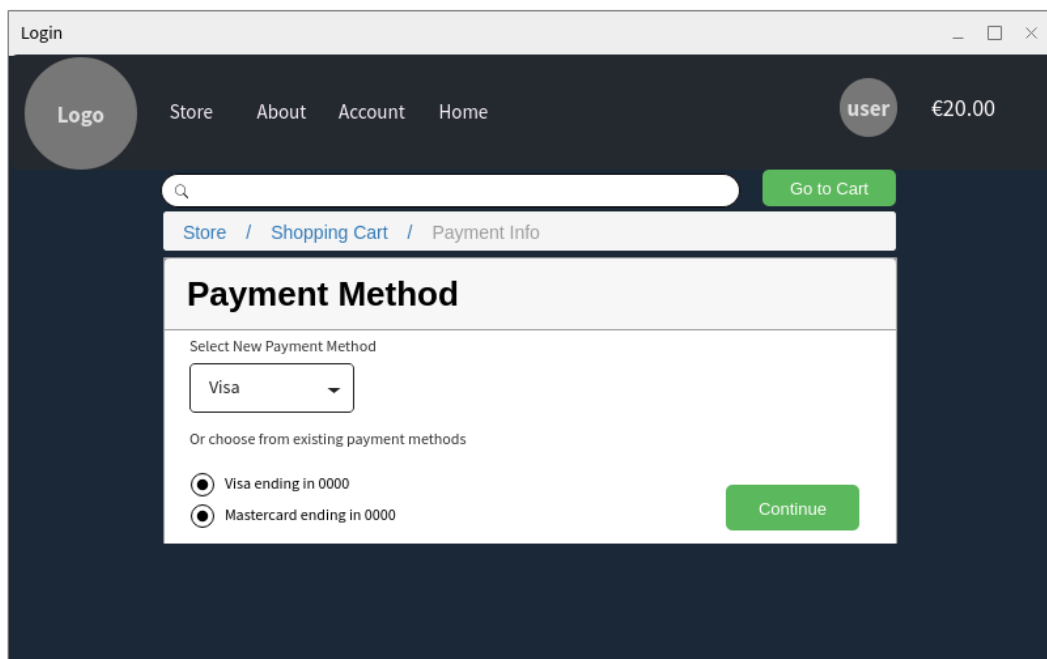


Figure 20 – Purchase Page

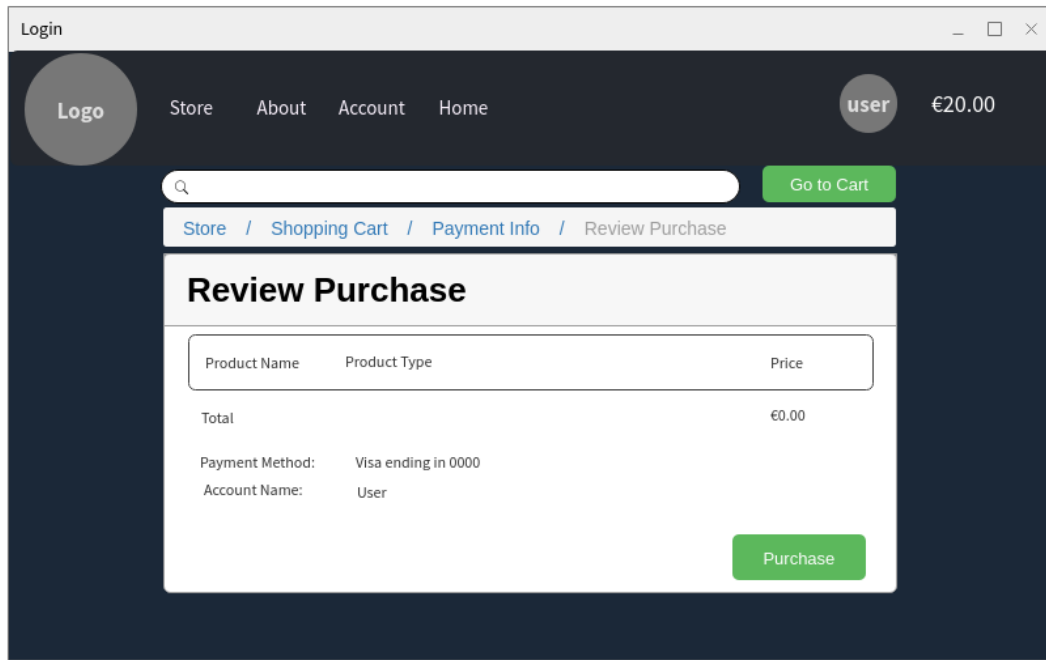


Figure 21 – Review Purchase Page

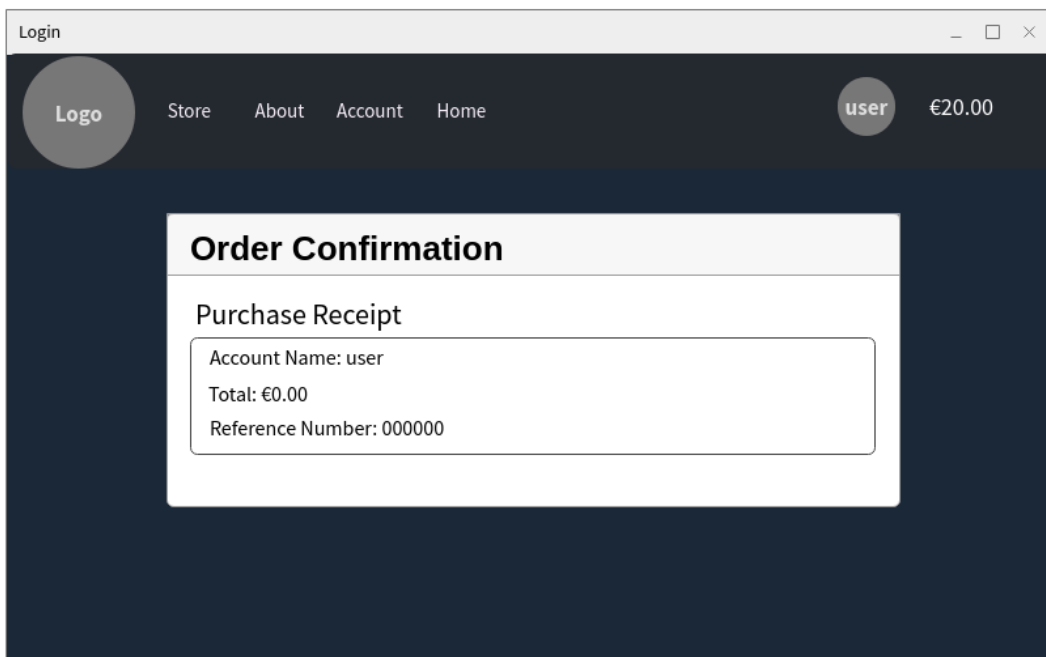


Figure 22 – Order Confirmation Page

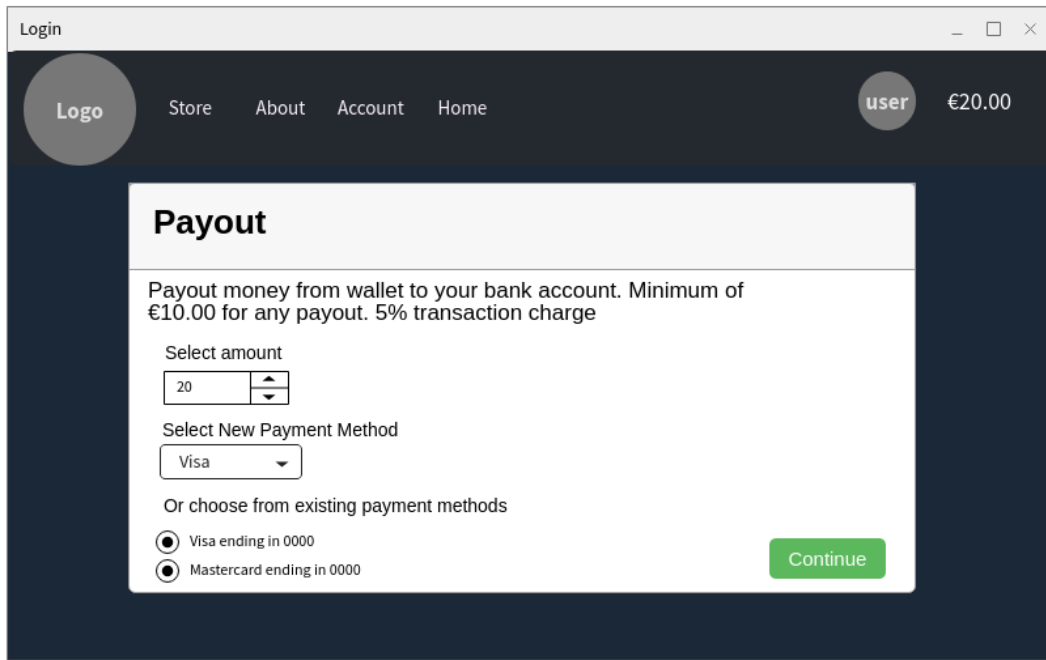


Figure 23 – Payout page

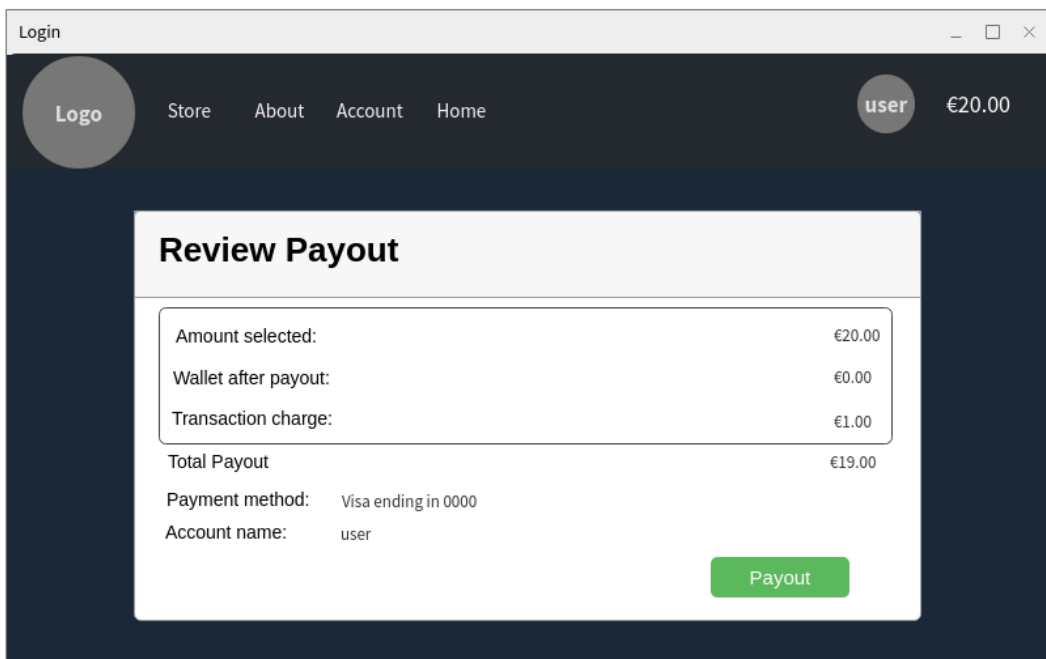


Figure 24 – Review Payout page

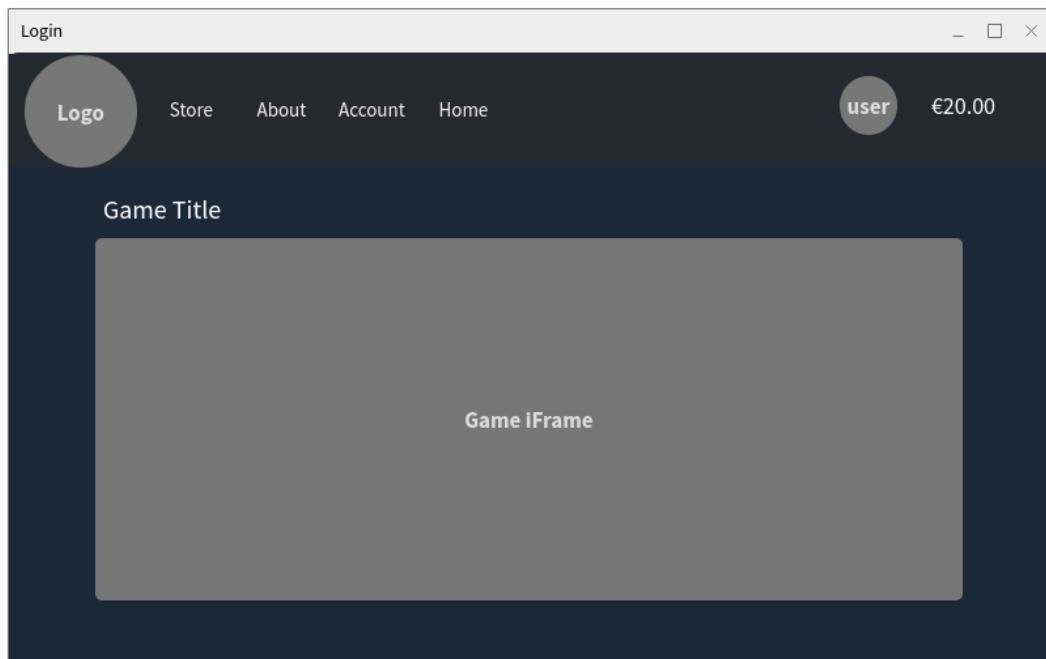


Figure 25 – Game page

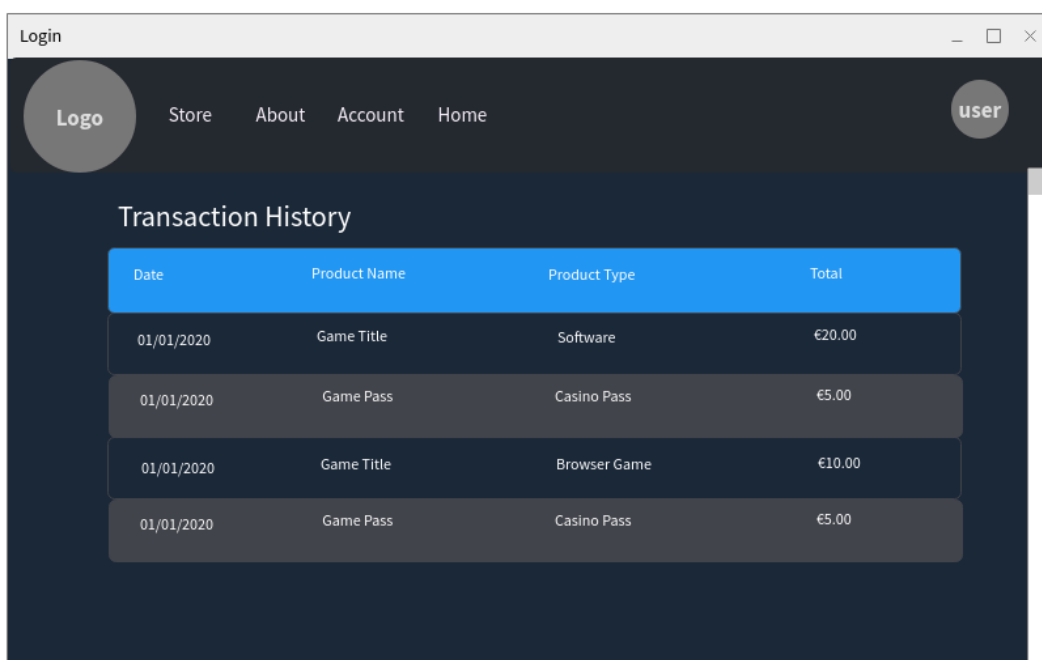


Figure 26 – Transaction History page

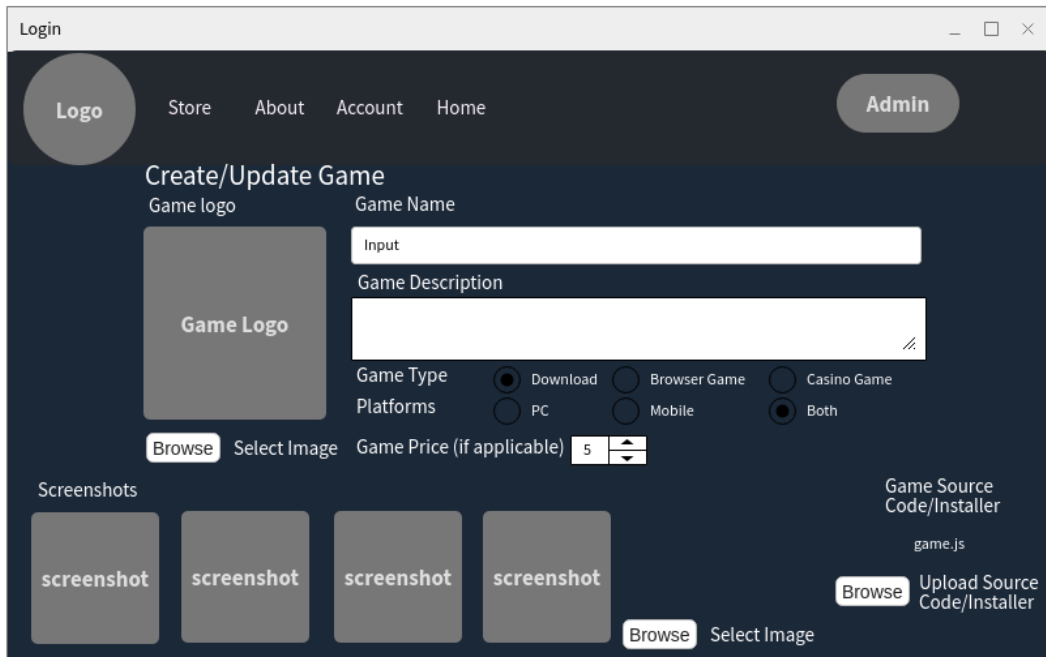


Figure 27 – Create/Update Game Page

DECLARATION

*I declare that all material in this submission e.g. thesis/essay/project/assignment is entirely my/our own work except where duly acknowledged.

*I have cited the sources of all quotations, paraphrases, summaries of information, tables, diagrams or other material; including software and other electronic media in which intellectual property rights may reside.

*I have provided a complete bibliography of all works and sources used in the preparation of this submission.

*I understand that failure to comply with the Institute's regulations governing plagiarism constitutes a serious offence.

Student Name: (Printed)KYLE HENNESSY_____

Student Number(s): C00227463_____

Signature(s): Kyle Hennessy_____

Date: 30/04/2021_____

Please note:

The Institute regulations on plagiarism are set out in Section 10 of Examination and Assessment Regulations published each year in the Student Handbook.