

VIETNAM NATIONAL UNIVERSITY, HO CHI MINH CITY
HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING



WEB PROGRAMMING

ASSIGNMENT REPORT

UniteDevs - Online Game Buying Website

Instructor: Nguyễn Đức Phúc

Students: Lương Anh Khoa - 2011421
Đương Nguyễn Nguyên Nghĩa - 2011672
Nguyễn Sơn Tín - 2012214
Đỗ Minh Quân - 2011908
Contact: khoa.luong2011421@hcmut.edu.vn

Ho Chi Minh City, 4/2023



Task Assignment

Name	ID	Role	Completion Rate
Dương Nguyễn Nguyên Nghĩa	2011672	Main coder, designer, researcher	100% + Above and Beyond
Lương Anh Khoa	2011421	Team leader, supporting back-end coder, UI designer, supporting report creator	100%
Nguyễn Sơn Tín	2011421	Supporting backend-coder, UI designer, supporting report creator	100%
Đỗ Minh Quân	2011908	UI designer, main report creator, testing website	100%



Mục lục

1 Requirement analysis	4
1.1 Project description	4
1.2 Goals	4
1.3 Functional requirement	5
1.4 Nonfunctional requirement	5
1.4.1 Product requirements	5
1.4.2 Organizational requirements	6
1.4.3 External requirements	7
1.5 External links of our group	7
2 Use-case	8
2.1 Overall use case	8
2.2 Login and info function branch	8
2.2.1 Login	9
2.2.2 Log out	9
2.2.3 Forget password	9
2.2.4 View purchase history	10
2.2.5 Personal information	10
2.2.6 Edit personal information	10
2.3 Game management branch	11
2.3.1 Use-case Game list	11
2.3.2 Use-case Search games	12
2.3.3 Use-case Add games	12
2.3.4 Use-case Delete games	13
2.3.5 Use-case Detailed Information	13
2.3.6 Use case Edit information	13
2.3.7 Use case Delete game	14
2.4 Pick and wishlist branch	14
2.4.1 Use-case Search game	15
2.4.2 Use-case Detailed information	15
2.4.3 Use-case Add to cart	15
2.4.4 Use-case Remove from cart	16
2.4.5 Use-case Add to wishlist	16
2.4.6 Use-case Remove from wishlist	17
2.5 Buying branch	17
2.5.1 Use case Scenario for viewing product cart	18
2.5.2 Use case Scenario for buying games	19
2.6 Customer Management Branch	20
2.6.1 Use case Scenario for Customer Management	20
3 Implementation	21
3.1 Technology	21
3.2 Mock up	22
3.2.1 Customer	22
3.2.2 Admin	25
3.3 Implementation	28
3.3.1 Customer	29
3.3.2 Admin	33
4 Implementation 2	37
4.1 Database	37
4.2 Server	37
4.3 Installation for review purpose	38



5 Reference Material	39
5.1 Front-End	39
5.1.1 React JS	39
5.1.2 CSS	39
5.1.3 JavaScript	39
5.1.4 Bootstrap	39
5.2 Back-End	39
5.2.1 Server-Side Programming Languages	39
5.2.2 Relational Databases	39



1 Requirement analysis

1.1 Project description

This project's website application aims to help the customers who do not have access to international payment methods to be able to purchase video games from publishers and platforms with a small extra fee (about 5% more compare to the original price).

The website will serve two types of users:

- Customers: These users' purposes of using the application are:
 - Visit the game list (and search) to see if their desired games are presented in the store or having discounts.
 - Get detailed information about the games (description, status, ratings...) and being able to add them into shopping carts or wishlists for later purchases.
 - Visit their wishlists to see whether their games are discounted or not to make final decisions.
 - Visit the shopping carts to purchase their games via various non-international payment methods.
- Administrators: These type of users have other uses of the website:
 - Visit the game list (and search for a specific game).
 - Get detailed information about the games (total sales, prices, discounts, status, ratings...).
 - Edit games' information (prices, discounts, status...).
 - Add new games to the store.
 - Delete old games or games that are no longer popular from the store.
 - Visit the customer list (and search for a specific customer).
 - Get detailed information about the customers (name, email, phone number, membership discounts, history purchases...)
 - Edit customers' membership discounts (maximum of 5%).

1.2 Goals

- Ease of Use: The website should be user-friendly, with an intuitive interface that allows users to quickly find and purchase the games they want. Navigation should be simple and straightforward, and information about the purchasing process, including fees and transaction details, should be clear and concise.
- Customer Satisfaction: Providing excellent customer service and support should be a top priority. Response times to inquiries and complaints should be fast, and returns and refunds should be easy and hassle-free. Feedback from users should be collected regularly to identify areas for improvement and enhance the overall customer experience.
- Security and Transparency: The website must provide a secure and transparent platform for purchasing video games. Industry-standard security measures should be implemented, such as SSL encryption and two-factor authentication, to protect user data and transactions. Pricing and transaction details, including fees and exchange rates, should be transparent and easy to understand, to ensure users can make informed purchasing decisions. By prioritizing security and transparency, the website aims to establish a strong reputation in the gaming community and build trust with its users.



1.3 Functional requirement

Login and info function branch:

- Login
- Logout

Game management branch:

- View game list: Administrators get the store's game list.
- Search Item: Administrators look for specific products.
- View detailed information: Administrators get a game's detailed information.
- Edit game information: Administrators edit the game's information (price, discount, status...).
- Add game: Administrators add a game to the store.
- Delete games: Administrators delete game(s) from the store.

Pick and wishlist branch:

- Detailed information: User can check product information before they decide to buy it or not
- Add to cart: all the products user want to purchase immediately.
- Add to wishlist: for products that user want to buy in the future, user can store them in wishlist.

Buying branch

- The user can choose to view their shopping cart and then can decide whether to buy the product(s) or not
- If they decide to buy then they can choose payment method (Card/transaction/Momo)
- And then they can choose to receive the game copy by either a physical copy or a digital edition

1.4 Nonfunctional requirement

1.4.1 Product requirements

Usability requirements:

- The system must be easy to use and have a simple interface so that customers can use it without instructions. 95 percent of customers can use it on the first trial.
- The system interface must be friendly, i.e. large icons, no obscure buttons, and clear within 1-meter visibility.
- People with vision problems such as color blindness can also order, pay, and view order information. (Depends on the chosen theme color).
- The ordering interface uses words that are easy to understand for most customers.

Efficiency requirement:

- The system must satisfy(running fast with little to no interruption) at least 1000 customers using it at the same time.
- Time to verify payment method does not exceed 30s in case the network is stable.



- The time to upload/ update/ delete the product is 2 seconds at maximum.
- The maximum response time is 2 seconds (the time that the customer interacts with the website).

Dependability requirements:

- High availability: The website must be available for customer usage at all times, except for scheduled maintenance periods.
- Reliable performance: The website should function reliably (does not crash in most circumstances) and perform well. The website should be thoroughly tested to ensure that all features and functions work as intended.
- Regular backups: The website should have a reliable backup system in place to ensure that customer data is not lost in the event of a hardware failure or other disaster. Regular backups should be performed, and a disaster recovery plan should be in place to restore the website quickly in the event of an outage.

Security requirement:

- Secure Authentication: User authentication must be secure and require a strong password. In addition, multi-factor authentication(MFA), such as SMS or email verification, should be enabled to ensure that only authorized users can access the site.
- Secure Data Storage: All customer data must be stored securely using encryption, and access to it should be strictly controlled to prevent unauthorized access.
- Secure Transactions: The website should use encrypted connections (HTTPS) and implement robust security measures to protect against unauthorized access. This includes securely storing all sensitive data and ensuring that all transactions are conducted in a secure and protected environment, minimizing the risk of fraud or other security breaches.
- Access Control: Access to sensitive customer data and other important information must be limited to authorized personnel only. This includes implementing access control measures such as user authentication, and authorization to ensure that only authorized users can access sensitive information.

1.4.2 Organizational requirements

Environmental requirements:

- Can run normally on most web browser applications: Chrome, Firefox, and IE,...
- Compatible with most operating systems: Windows, Mac, Android, IOS,...
- All information (such as products, customers' information,...) is stored in the database.
- The system is built on [MVC model], a framework.

Operational requirements:

- The system has administrators who manage all system's activities (times of access, remaining, update, authorization,...), and customers do not have permission to access the system. Access the website through [link], or a mobile application
- Customers must log in for some activities (payment, comment, rate,...)
- Resource consumption
- Back up data once a month



Development requirements:

- Easy to edit, rebuild
- Flexible
- Find and fix the bug as soon as possible

1.4.3 External requirements

Regulatory requirements:

- Data protection and privacy laws: The system must comply with data protection and privacy laws in their country or region. This may include requirements for obtaining user consent for collecting personal data, providing privacy policies, and protecting user data from unauthorized access or use.
- Electronic commerce laws: The system must comply with electronic commerce laws,...This includes requirements for displaying pricing information, terms and conditions, and refund policies
- Consumer protection laws: The system must comply with consumer protection laws, which may include requirements for providing accurate product information, offering refunds or returns, and protecting consumers from false or misleading advertising.

Ethical requirements:

- Honesty: The company must be honest in its product descriptions and claims, avoiding any false or misleading information. The company should also be honest about its ability to deliver products, including shipping times and availability.
- Fairness: The company must be fair in its pricing and promotions, avoiding any price gouging or deceptive pricing practices. The company should also be fair in its treatment of customers, providing equal access and opportunities for all customers.
- Transparency: The company must be transparent about its business practices and policies, including pricing, returns, and data collection. The company should also be transparent about any potential conflicts of interest or biases that could impact its recommendations or sales.

1.5 External links of our group

Github: https://github.com/KhoaLiang/LTW_222

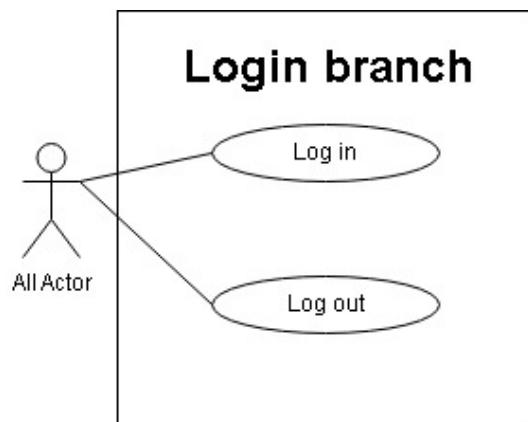
Task Assigning Form: <https://drive.google.com/drive/folders/1onep86pYqMID6eCB6Kg958lG4PDZht9K>

2 Use-case

2.1 Overall use case



2.2 Login and info function branch





2.2.1 Login

Use case	Log In
Goal	Log in successfully into the system
Primary Actor	All human actors in the system
Secondary Actor	Server
Preconditions	1. Have an authorized account 2. The system is currently in the login screen
Postconditions	Opened features depending on the administrative right of the account login in
Basic Flow	1. The user enters username and password 2. The system receives the account information and verifies the content. 3. The system verifies the login success, closes the login interface, and unlocks the appropriate interface depending on the administrative right of the account
Exception Flow	Exception 1: Step 2: 2.1. The system doesn't find the account with the login information the user provided 2.2. The system informs the user of the nature of the error encountered and refuses the login attempt

2.2.2 Log out

Use case	Log Out
Goal	User logs out of the system
Primary Actor	All actor
Secondary Actor	Server
Preconditions	The user is still inside the system
Postconditions	The user log out of the system
Basic Flow	1. The user presses the "Log out" button 2. The system inform the user of their success in logging out.

2.2.3 Forget password

Use case	Forget password
Goal	Reset pwd and get the user to login to the system
Primary Actor	User
Secondary Actor	None
Preconditions	the user is in the login interface
Post conditions	Reset pwd and get the user to login to the system
Basic Flow	1. The user press the forget password link 2. The user enter their email 3. The system send an email to the user, when clicked it log the user into the system and prompt them to set their new pwd right away. 4. The user change their pwd and then move to the homepage
Exception Flow	None



2.2.4 View purchase history

Use case	View purchase history
Goal	View purchase history
Primary Actor	User
Secondary Actor	None
Preconditions	The user is viewing their personal information
Postconditions	The user see their personal information
Basic Flow	1. The user press “purchase history” button 2. The user see their “purchase history”
Exception Flow	None

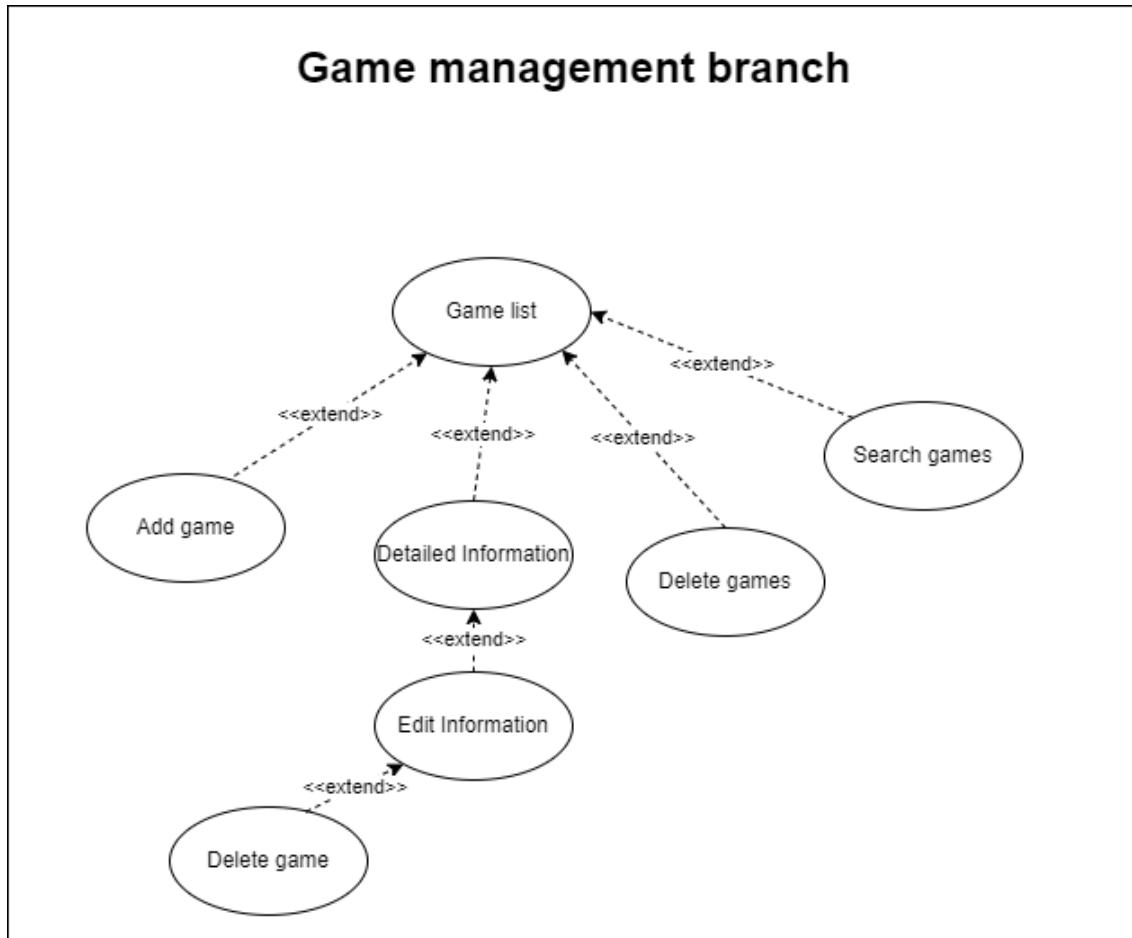
2.2.5 Personal information

Use case	Personal information
Goal	view personal information
Primary Actor	User
Secondary Actor	None
Preconditions	The user is logged in
Postconditions	The user can see their information
Basic Flow	1. The user press “Personal information” button 2. The user see their personal information
Exception Flow	None

2.2.6 Edit personal information

Use case	Edit personal information
Goal	Edit personal information
Primary Actor	User
Secondary Actor	None
Preconditions	The user is viewing their personal information
Postconditions	Edited personal information
Basic Flow	1. The user press “Edit” button 2. The user changes their personal information 3. The user press “save and exit”
Alternative Flow	Alternative 1: 3.1 The user press cancel 3.2 THe user is moved back to their viewing of personal information

2.3 Game management branch



2.3.1 Use-case Game list

Use case name	Search Item.
Actor	Administrators.
Description	The users want to retrieve the list of games that are presented in the store.
Precondition	The users have accessed the website with the administration accounts.
Trigger	The users click "Games" on the navigation bar.
Basic Flow	<ol style="list-style-type: none">1. The users click "Games" on the navigation bar.2. The system displays a list of products.
Postcondition	The users have successfully get the list of presented games of the website.



2.3.2 Use-case Search games

Use case name	Search games
Actor	Administrators
Description	The users want to search for specific games.
Precondition	The users have already been in the "Games" page.
Trigger	The users click the search bar and type in the name of the game they want to see.
Basic Flow	<ol style="list-style-type: none">1. The users click on the search bar and type in the name of the game they want to see.2. The system displays the list of products that matches the finding.
Postcondition	All games that match the keyword will be displayed
Notes	If the games that the users want to find are not presented in the store, the system will display nothing as result.

2.3.3 Use-case Add games

Use case name	Add games
Actor	Administrators
Description	The users want to add a game to the store.
Precondition	The users have already been in the "Games" page.
Trigger	The users click the "Add a game" button.
Basic Flow	<ol style="list-style-type: none">1. The users click on the "Add a game" button.2. The system displays a new form page for users to fill in the information about the new game.3. The user fill in the form and click "Confirm".4. The system updates the new game to the store and display a success pop-up.
Postcondition	The new game is added to the store.
Exceptions	<ol style="list-style-type: none">1. If the game have already been in the store, the system will display a pop-up to notice the users about it.2. If the uploaded files are not .txt type, the system will display a pop-up error.



2.3.4 Use-case Delete games

Use case name	Delete games
Actor	Administrators
Description	The users want to delete game(s) from the store.
Precondition	The users have already been in the "Games" page.
Trigger	The users click the "Delete games" button.
Basic Flow	<ol style="list-style-type: none">1. The users click the "Delete games" button.2. The system display a box at the end of every product.3. The users check the boxes to mark as delete.4. The users click "Confirm".5. The system will display a pop-up to ask again.6. The users click "Yes".7. The system remove the selected game(s) from the store.
Postcondition	The system remove the selected game(s) from the store.

2.3.5 Use-case Detailed Information

Use case name	Detailed Information
Actor	Administrators
Description	The users want to see the details of a specific game.
Precondition	The users have already been in the "Games" page.
Trigger	The users click on a specific game in the list.
Basic Flow	<ol style="list-style-type: none">1. The users click on a specific game in the list.2. The system will display the detail information of a game.
Postcondition	The system displays detailed information about a specific game.

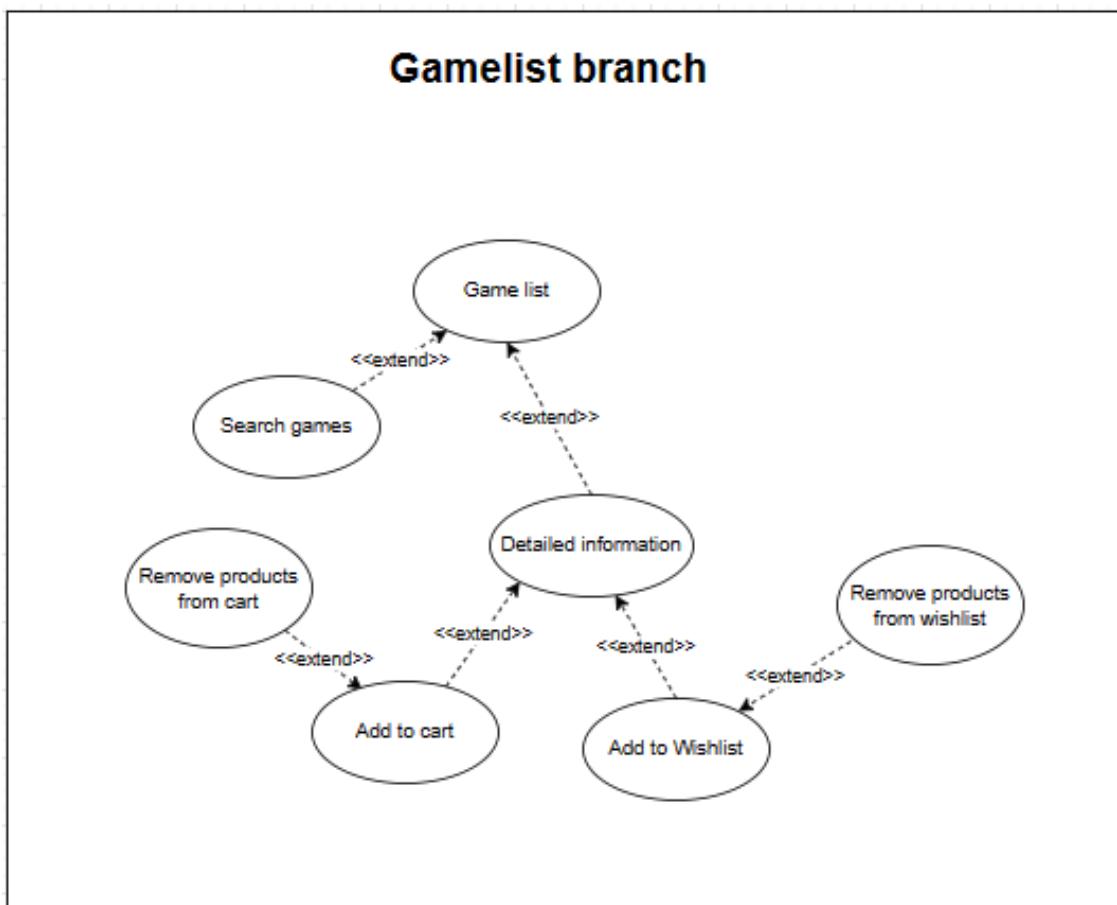
2.3.6 Use case Edit information

Use case name	Use case Edit information
Actor	Administrators
Description	The users want to edit the information of a game.
Precondition	The users have already been in the game detailed information page.
Trigger	The users click the "Edit" button.
Basic Flow	<ol style="list-style-type: none">1. The users click the "Edit" button.2. The system will display a form page for users to update the information.3. The users click "Confirm".4. The system will update the information of the game.
Postcondition	The system is updated with the newest information.
Exception	<ol style="list-style-type: none">1. If the users rename the game with a name that are already used for another product, the system will display an error pop-up.2. If the uploaded files are not .txt type, the system will display a pop-up error.

2.3.7 Use case Delete game

Use case name	Use case Delete game
Actor	Administrators
Description	The users want to delete a specific game.
Precondition	The users have already been in the game information edit page.
Trigger	The users click the delete icon.
Basic Flow	<ol style="list-style-type: none"> 1. The users click the delete icon. 2. The system will display a pop-up to ask again. 3. The users click "Yes". 4. The system will remove the game from the store.
Postcondition	The system removes the game from the store.

2.4 Pick and wishlist branch





2.4.1 Use-case Search game

Use case name	Search game
Actor	User
Description	User find product related to the game they want
Precondition	User in the home page
Trigger	User select search box
Basic Flow	<ol style="list-style-type: none">1. System search product base on the User's input2. System display result found.
Postcondition	The found product is displayed on the screen
Notes	None

2.4.2 Use-case Detailed information

Use case name	Detailed information
Actor	User
Description	User check detailed of a product
Precondition	User in the home page
Trigger	User select a product
Basic Flow	<ol style="list-style-type: none">1. System display information: game, code, code description and price
Postcondition	The information is displayed on the screen
Notes	None

2.4.3 Use-case Add to cart

Use case name	Add to cart
Actor	User
Description	User add a product to cart
Precondition	User has logged in and in detailed information page of product
Trigger	The User selects the "Add to cart" option on the website
Basic Flow	<ol style="list-style-type: none">1. The system displays a confirmation message.2. The User selects "Confirm".3. The system add product to User's cart.4. The system calculate total cost.5. The system come back to Detail information page
Alternative Flows	<ol style="list-style-type: none">2a. User select "Cancel". The system dont add product and jump back to Detail information page
Postcondition	<ol style="list-style-type: none">1. The selected product is added to User's cart2. "Remove from cart" option become available
Exceptions	The User is not logged into the system. The system displays an error message and prompts the User to log in before adding a product.
Notes	None



2.4.4 Use-case Remove from cart

Use case name	Remove from cart
Actor	User
Description	User remove a product from cart
Precondition	User has logged in and in detailed information page of product; product has been added to cart
Trigger	The User selects the "Remove from cart" option on the website
Basic Flow	<ol style="list-style-type: none">1. The system displays a confirmation message.2. The User selects "Confirm".3. The system remove product from User's cart.4. The system recalculate total cost.5. The system come back to Detail information page
Alternative Flows	<ol style="list-style-type: none">2a. User select "Cancel". The system dont remove product and jump back to Detail information page
Postcondition	<ol style="list-style-type: none">1. The selected product is remove from User's cart2. "Add to cart" option become available
Exceptions	The User is not logged into the system. The system displays an error message and prompts the User to log in before removing a product.
Notes	None

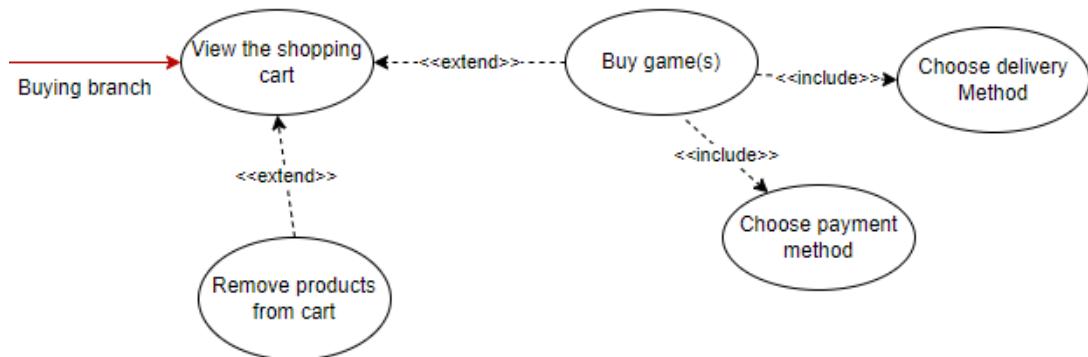
2.4.5 Use-case Add to wishlist

Use case name	Add to wishlist
Actor	User
Description	User add a product to wishlist
Precondition	User has logged in and in detailed information page of product
Trigger	The User selects the "Add to wishlist" option on the website
Basic Flow	<ol style="list-style-type: none">1. The system displays a confirmation message.2. The User selects "Confirm".3. The system add product to User's wishlist.
Alternative Flows	<ol style="list-style-type: none">2a. User select "Cancel". The system dont add product and jump back to Detail information page
Postcondition	<ol style="list-style-type: none">1. The selected product is added to User's wishlist2. "Remove from wishlist" option become available
Exceptions	The User is not logged into the system. The system displays an error message and prompts the User to log in before adding a product.
Notes	None

2.4.6 Use-case Remove from wishlist

Use case name	Remove from wishlist
Actor	User
Description	User remove a product from wishlist
Precondition	User has logged in and in detailed information page of product, product has been added to wishlist
Trigger	The User selects the "Remove from wishlist" option on the website
Basic Flow	<ol style="list-style-type: none"> 1. The system displays a confirmation message. 2. The User selects "Confirm". 3. The system remove product from User's wishlist.
Alternative Flows	<ol style="list-style-type: none"> 2a. User select "Cancel". The system dont remove product and jump back to Detail information page
Postcondition	<ol style="list-style-type: none"> 1. The selected product is remove from User's wishlist 2. "Add to wishlist" option become available
Exceptions	The User is not logged into the system. The system displays an error message and prompts the User to log in before removing a product.
Notes	None

2.5 Buying branch





2.5.1 Use case Scenario for viewing product cart

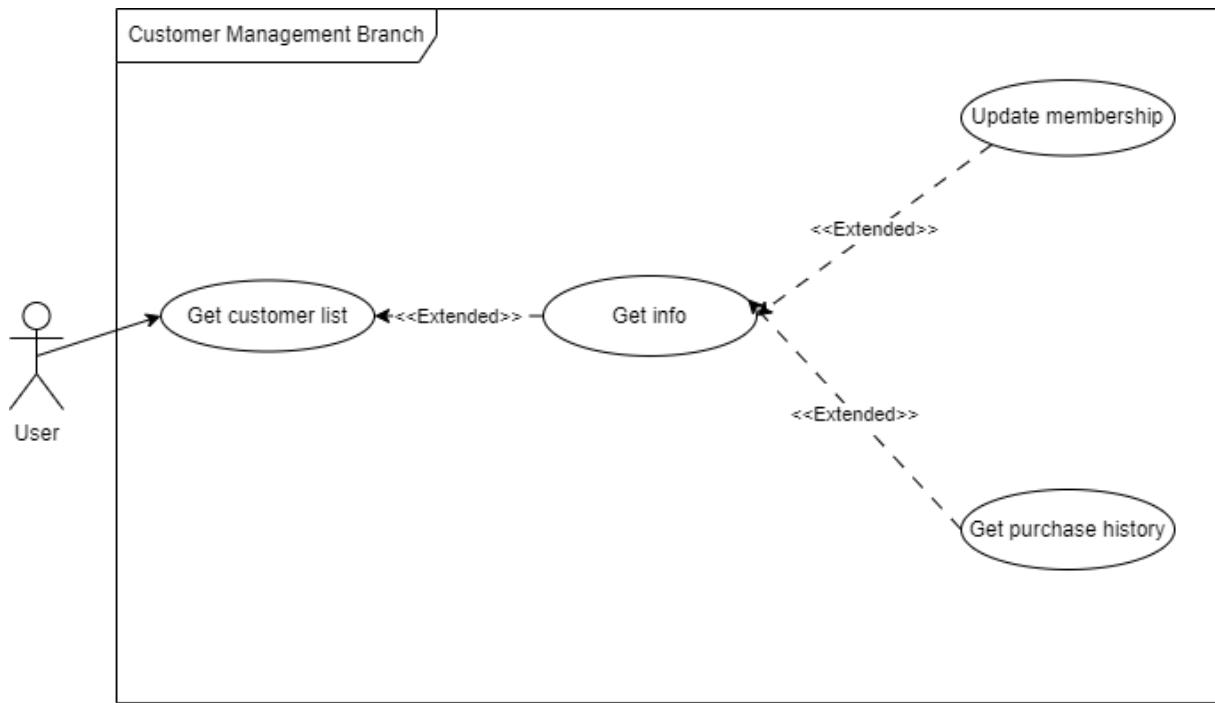
Use case	View Game Cart
Goal	Allows users to view and check their product cart
Primary Actor	Customer
Secondary Actor	Server
Preconditions	<ol style="list-style-type: none">1. User has a valid account on the website2. User has added one or more games to their product cart3. User has a valid payment method registered on the website
Postconditions	<ol style="list-style-type: none">1. The user can view the games in their game cart, their prices, and quantities.2. The user can update the quantities of the games or remove them from the cart.3. The user can proceed with the checkout process to complete the purchase or continue shopping.
Basic Flow	<ol style="list-style-type: none">1. User clicks on the "Cart" icon or link on the game website2. The game website displays the games that the user has added to their game cart, along with their prices and quantities3. User verifies that the games in the cart are correct and adjusts quantities or removes any unwanted games if necessary4. User clicks on the "Checkout" button to proceed with the purchase
Alternative Flow	Alternative 1: 4.1 If the user decides to continue shopping, they click on the "Continue Shopping" button 4.2 They return to the website's game catalog
Exception Flow	None



2.5.2 Use case Scenario for buying games

Use case	Buy Games in Product Cart
Goal	Allows users to finalize their transaction
Primary Actor	Customer
Secondary Actor	Server
Preconditions	<ol style="list-style-type: none">1. User has a valid account on the website2. User has added one or more games to their product cart3. User has a valid payment method registered on the website
Postconditions	<ol style="list-style-type: none">1. The purchased games are sent for download or physical delivery, depending on the delivery method selected by the user.2. The user's account is updated to reflect the purchased games.3. The user receives an email confirmation of the purchase with a summary of the purchased games, the total cost, and estimated delivery date.4. The website's inventory of the purchased games is updated to reflect the sales.
Basic Flow	<ol style="list-style-type: none">1. User navigates to the website's product cart page2. The website displays the games in the user's cart along with their prices, quantities, and any available discounts or promotions3. User verifies that the games in the cart are correct and adjusts quantities or removes any unwanted games if necessary4. The website prompts the user to select a delivery method (digital or physical) for each game in the cart5. User selects the delivery method for each game and proceeds to enter their shipping address, billing address, and payment information if applicable6. The system pops up and asks the customer one last time to confirm7. The website updates the user's account to reflect the purchased games and sends the games for download or physical delivery, depending on the delivery method selected by the user
Alternative Flow	Alternative 1: 5.1 If the user selects a digital delivery method for a game, the website prompts the user to download the game file(s) after completing the payment and sends an email confirmation to the user's registered email address with a link to download the game file(s). The website does not require shipping or billing information for digital delivery. Alternative 2: 5.1. If the user selects a physical delivery method for a game, the website prompts the user to enter their shipping address for each physical game and displays the additional shipping cost for each game. The user enters their shipping address and proceeds with the payment as usual. The website updates the user's account to reflect the purchased games and sends the games for physical delivery to the specified address.
Exception Flow	<p>Exception 1:</p> <ol style="list-style-type: none">6.1. The user does not have enough money in their chosen method to pay for the purchase6.2. The system informs the customer of the error and goes back to step 5

2.6 Customer Management Branch



2.6.1 Use case Scenario for Customer Management

Use case	Customer Management
Goal	To manage customer information and history on the shopping site
Primary Actor	Administrator
Secondary Actor	Server
Preconditions	1. Administrator has a valid account on the website 2. Administrator is authenticated and authorized to access customer management features.
Postconditions	1. Customer information is updated and/or retrieved as needed
Basic Flow	1. Administrator selects "Get User List" from the customer management page. 2. System retrieves a list of all registered users on the shopping site. 3. Administrator selects a specific user from the list. 4. System displays the user's information, including name, address, email, and membership status. 5. Administrator selects "Get Purchase History" to view the user's past purchases. 6. System displays a list of the user's past purchases, including date, item description, and price. 7. Site Administrator selects "Update Membership Status" to change the user's membership level. 8. System displays a list of available membership levels. 9. Site Administrator selects a new membership level for the user. 10. System updates the user's membership status and displays a confirmation message.
Alternative Flow	None
Exception Flow	None



3 Implementation

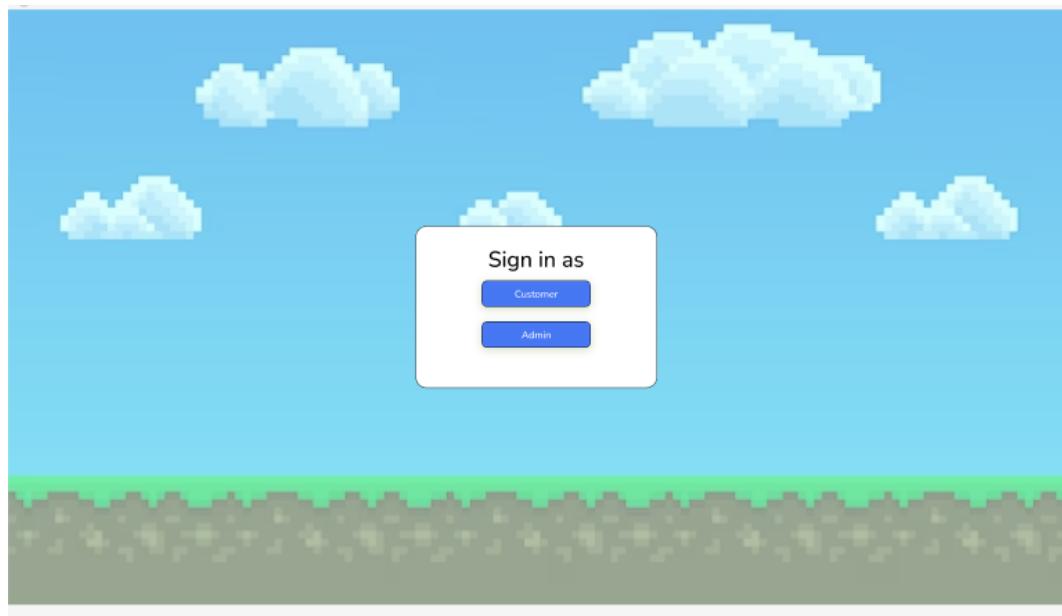
3.1 Technology

- React JS is a popular front-end JavaScript library used for building user interfaces. It provides a component-based architecture that enables developers to create reusable UI components, making it faster and easier to build complex web applications. React also offers a virtual DOM (Document Object Model) that allows for efficient updating and rendering of changes to the user interface. It was used to develop our development server
- JavaScript is a high-level programming language used to create dynamic and interactive web pages. It can be used both on the front-end and back-end of web development, and is widely used for adding interactivity to websites and web applications. It was used to design components and pages for the website
- CSS Bootstrap is a front-end framework used for building responsive and mobile-first web applications. It provides a set of pre-designed CSS classes and JavaScript components that make it easier to style and layout web pages. Bootstrap also includes a responsive grid system that allows developers to create responsive designs that adapt to different screen sizes. It helped accelerate the design of the project. So the team could focus more on the feature of the project
- PHP is a server-side scripting language that is commonly used for web development. It can be used to create dynamic web pages, process forms, and interact with databases. In the case of a game selling website, PHP could be used to handle user authentication, process payments, and interact with the database to retrieve and display game information. The server was designed by the usage of PHP
- XAMPP is a software package that includes Apache web server, MySQL database, and PHP. It allows developers to set up a local web development environment on their own machine. This can be useful for testing and developing web applications before deploying them to a live server. In the case of a game selling website, XAMPP could be used to develop and test the website on a local machine before deploying it to a live server. By putting the PHP code inside the Htdoc folder, the development server was implemented
- MySQL is a relational database management system that is commonly used for web applications. It can be used to store and retrieve data for the game selling website, such as information about games, user accounts, and orders. PHP could interact with the MySQL database using SQL queries to perform actions like inserting new data, retrieving data, and updating existing data.
- MySQL Workbench is a visual database design tool that can be used to create and manage MySQL databases. It provides a graphical user interface for creating, editing, and testing database schemas, as well as executing SQL queries and managing database connections. MySQL Workbench was used to design the database schema for the website, create tables to store information about games, users, orders, and payments, and establish relationships between the tables.

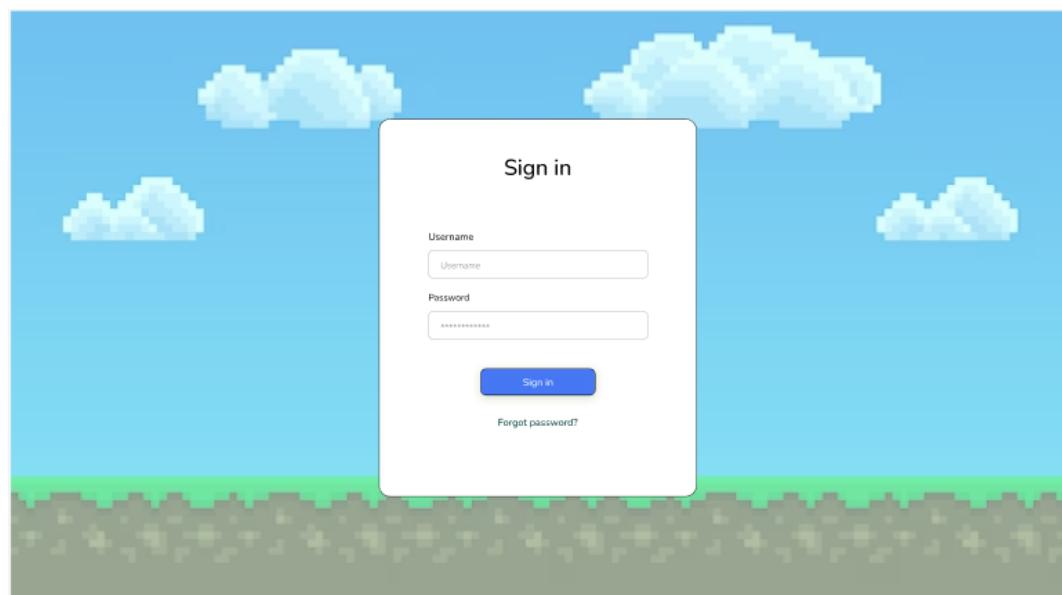
By using these technologies, We could build a robust and modern game buying website that provides a great user experience, is fast and responsive, and meets the security and usability requirements of our customers.

3.2 Mock up

3.2.1 Customer



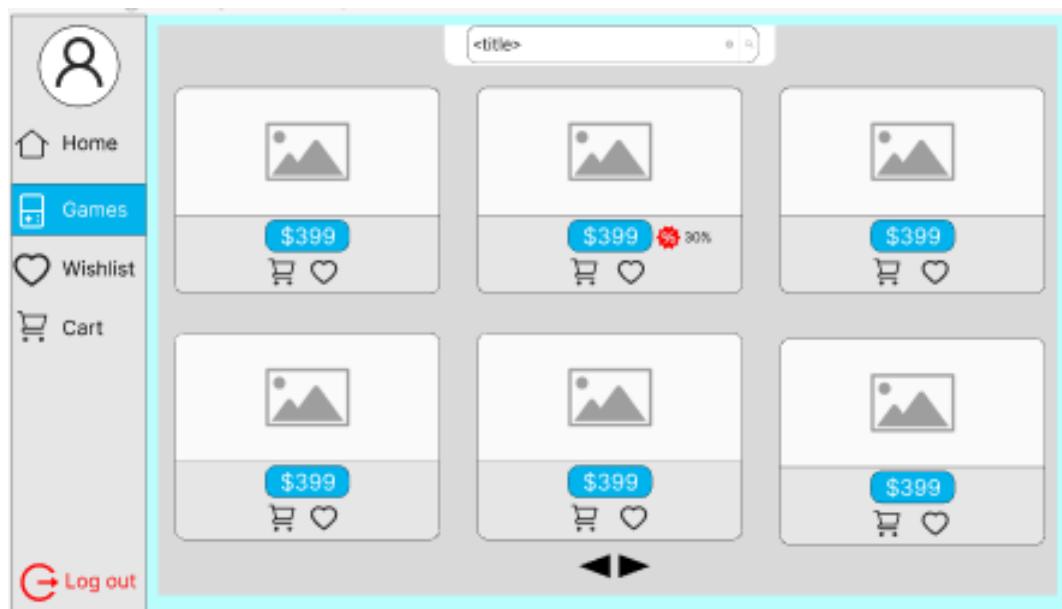
Hình 1: Choose Login Role Screen



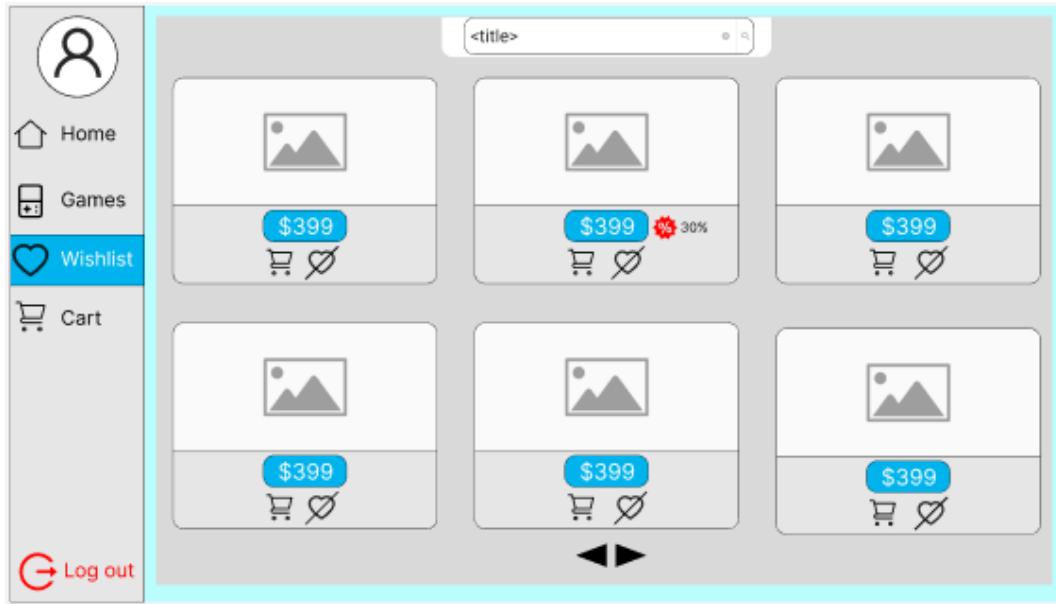
Hình 2: Login Screen



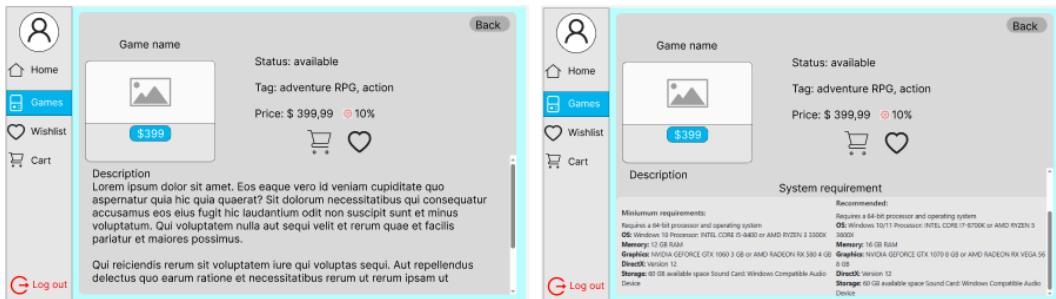
Hình 3: Customer DashBoard



Hình 4: Customer Gamelist



Hình 5: Customer Wishlist



Hình 6: Game Detail for customers



The screenshot shows a customer's shopping cart. On the left is a sidebar with icons for Home, Cart (highlighted in blue), Info, Statistics, and Log out. The main area is titled "Your Shopping cart". It contains two items, both labeled "Doki Doki Literature Club - \$10" and "Category: Visual Novel, Dating, Horror". Each item has a "Remove from cart" button. At the bottom, it says "Total: 20\$" and has a "Buy Now" button.

Hình 7: Cart for the customer

3.2.2 Admin

The screenshot shows the admin dashboard. On the left is a sidebar with icons for Home (highlighted in blue), Games, Customers, Statistics, and Log out. The main area features a "Best sellers" section with five game cards. The first three cards show a price of "\$399" and a "30%" discount icon. The last two cards also show a price of "\$399". Each card has a "Cart" and a "Heart" icon below the price.

Hình 8: Dashboard for admins



The screenshot shows a sidebar with icons for Home, Games (selected), Customers, Statistics, and Log out. The main area displays a list of games with columns for ID, Name, Price, Category 1, Category 2, Sold quantity, Ratings, and Detail link. Below the list are two buttons: 'Delete games' (red) and 'Add a game' (blue).

<id>	<Name>	<Price>	<Category 1>	<Category 2>	SOLD: <number>	Ratings: x stars	Detail
<id>	<Name>	<Price>	<Category 1>	<Category 2>	<number>	x stars	Detail
<id>	<Name>	<Price>	<Category 1>	<Category 2>	<number>	x stars	Detail
<id>	<Name>	<Price>	<Category 1>	<Category 2>	<number>	x stars	Detail
<id>	<Name>	<Price>	<Category 1>	<Category 2>	<number>	x stars	Detail
<id>	<Name>	<Price>	<Category 1>	<Category 2>	<number>	x stars	Detail
<id>	<Name>	<Price>	<Category 1>	<Category 2>	<number>	x stars	Detail
<id>	<Name>	<Price>	<Category 1>	<Category 2>	<number>	x stars	Detail
<id>	<Name>	<Discount>	<Category 1>	<Category 2>	<number>	- SOLD: <number> - Ratings: x stars	Detail

Hình 9: Gamelist for admins

The screenshot shows a sidebar with icons for Home, Games (selected), Customers, Statistics, and Log out. The main area displays a game detail page with fields for Name, Status (available), Tag, Price (\$399, 30% off), Sold (xxx), and Ratings (5 stars). Below these are sections for Description (Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed in orci ac ligula bibendum placerat vitae ut quam. Nullam scelerisque ornare ante, ut iaculis massa ultrices in.) and System requirements (Minimum and Maximum). The Minimum requirements include OS: Windows 10 64-bit, Processor: Intel i5 4th generation or AMD FX 6300 or equivalent, Memory: 8 GB RAM, Graphics: NVIDIA GeForce GTX 1050Ti or equivalent, DirectX: Version 11, Network: Broadband Internet connection, Storage: 20 GB available space, and Additional Notes: We advise you install the game on an SSD for a more streamlined experience. The Maximum requirements include OS: Windows 10 64-bit, Processor: Intel i7 7th generation or equivalent, Memory: 16 GB RAM, Graphics: NVIDIA GeForce GTX 1060 6G or equivalent, DirectX: Version 11, Network: Broadband Internet connection, Storage: 20 GB available space, and Additional Notes: We advise you install the game on an SSD for a more streamlined experience.

Hình 10: Game detail for admins



The screenshot shows a form for editing a game. On the left is a sidebar with icons for Home, Games (selected), Customers, Statistics, and Log out. The main area has fields for Name (<Name>), four picture slots (Browse picture 1-4), and a delete icon. Below these are fields for Price (\$), Discount (%), Tags (Category 1-2), and file attachments for Codes (.txt) and Description. There are also fields for System requirements (minimum and recommended) with browse file buttons. At the bottom are 'Cancel' and 'Confirm' buttons.

Hình 11: Edit game screen

The screenshot shows a list of customer data. The sidebar includes Home, Games, Customers (selected), Statistics, and Log out. The main area displays a table with columns for User name, Email, Phone number, and Total spending. Each row has a 'Detail' button. At the bottom is a red 'Delete users' button.

<User name>	<Email>	<Phone number>	- <Total spending>	<input type="button" value="Detail"/>
<User name>	<Email>	<Phone number>	- <Total spending>	<input type="button" value="Detail"/>
<User name>	<Email>	<Phone number>	- <Total spending>	<input type="button" value="Detail"/>
<User name>	<Email>	<Phone number>	- <Total spending>	<input type="button" value="Detail"/>
<User name>	<Email>	<Phone number>	- <Total spending>	<input type="button" value="Detail"/>
<User name>	<Email>	<Phone number>	- <Total spending>	<input type="button" value="Detail"/>
<User name>	<Email>	<Phone number>	- <Total spending>	<input type="button" value="Detail"/>
<User name>	<Email>	<Phone number>	- <Total spending>	<input type="button" value="Detail"/>
<User name>	<Email>	<Phone number>	- <Total spending>	<input type="button" value="Detail"/>

Hình 12: Customer list

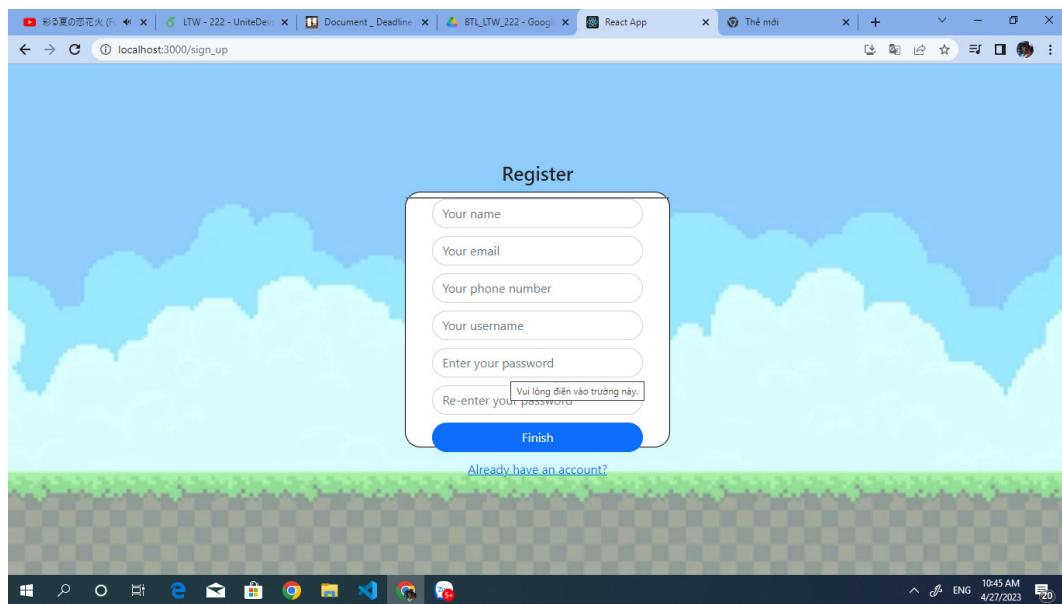


The screenshot shows a web application interface for managing customer details. On the left, there is a sidebar with icons for Home, Games, Customers (selected), and Statistics. A 'Log out' button is also present. The main content area displays customer information: Name (<name>), Email (<email>), Phone number (<phone number>), Total spending (\$<number>), Membership rank (Silver/Gold/Diamond/Special), and Membership discount (1/2/3/custom <=5%). Below this, a list of purchases is shown, each with a 'Detail' button:

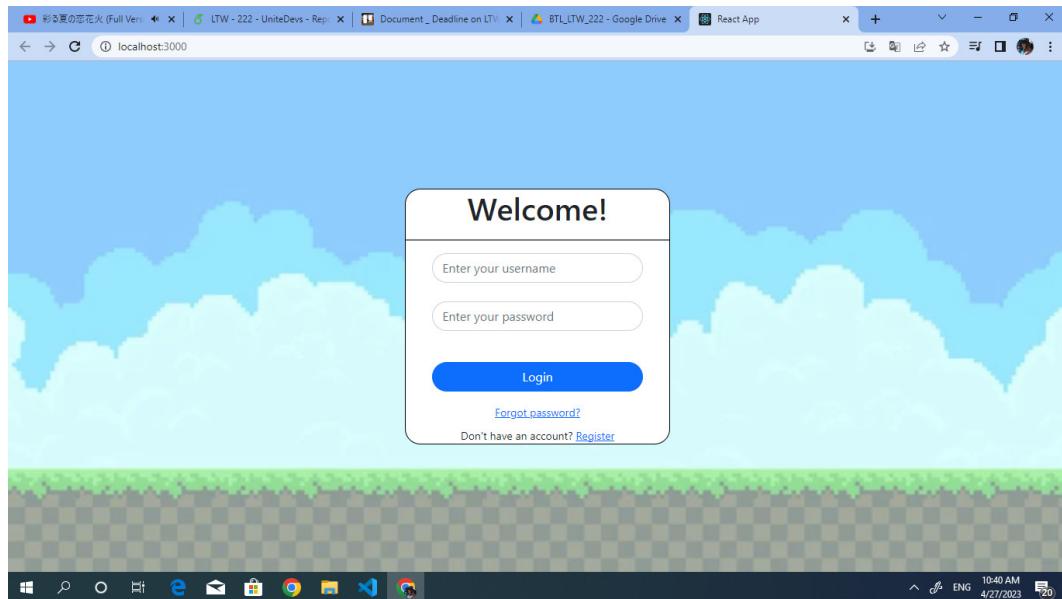
<Product name> - <Price> - <Date of purchase>	Detail
<Product name> - <Price> - <Date of purchase>	Detail
<Product name> - <Price> - <Date of purchase>	Detail
<Product name> - <Price> - <Date of purchase>	Detail
<Product name> - <Price> - <Date of purchase>	Detail

Hình 13: Customer detail for admins

3.3 Implementation

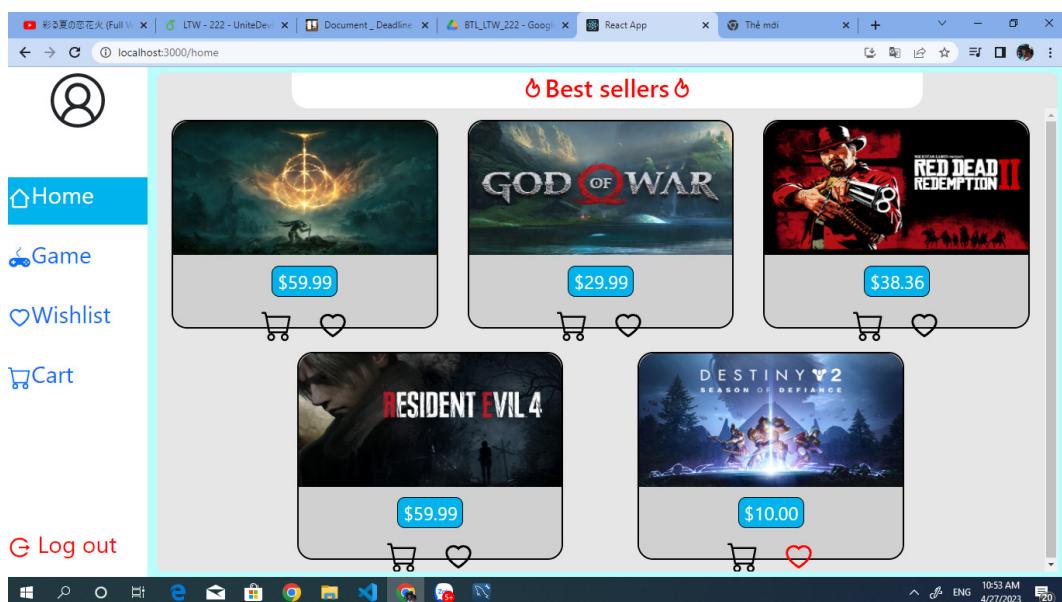


Hình 14: In this page users can create accounts to gain access to the server services



Hình 15: User can login to the website using the created account

3.3.1 Customer



Hình 16: The first page to appear when user log in the website as a customer



The screenshot shows a web-based game store interface. On the left, there's a sidebar with buttons for Home, Game (which is currently selected and highlighted in blue), Wishlist, Cart, and Log out. The main area displays a grid of six game covers. Each cover includes a price tag with a discount icon (e.g., '\$10.00', '\$59.99', '\$28.79', '\$29.99', '\$17.54', '\$44.99') and a shopping cart and heart icon below it. At the top right of the main area is a search bar with the placeholder 'Find' and a magnifying glass icon.

Hình 17: The whole game list will be shown to the customers when they click on this page, here they can view or add game to their cart

This screenshot shows a detailed view of a game from the store. The sidebar on the left remains the same. The main content is centered on 'Hogwarts Legacy'. At the top, it shows the game's status as 'Available', its tag as 'Adventure RPG, Story-Rich', its price as '\$44.99' with a 10% discount, and its rating as '4.1'. Below this is a large image of the game's cover art, which depicts a character in a dark robe facing a large, muscular creature. To the right of the cover are two small icons: a shopping cart and a heart. Underneath the cover, there's a 'Description' section that provides a brief summary of the game's plot and setting. At the bottom of the page is a 'System requirements' section, which lists the minimum and recommended system specifications for running the game.

Hình 18: Users can view the detail of the game they clicked on in the gamelist page here



The screenshot shows a user interface for a wishlist. On the left, there's a sidebar with navigation links: Home, Game, Wishlist (which is highlighted in blue), and Cart. Below these are Log out and a placeholder for a profile picture. The main area displays three game cards with their prices and a heart icon below each. A search bar is at the top right.

Game	Price
DESTINY 2 SEASON OF DEFIANCE	\$10.00
Fallout 76 WASTELANDERS	\$28.79
Grand Theft Auto V	\$17.54

Hình 19: All the games that users have clicked on the heart button will appear here

The screenshot shows a user profile page. The sidebar includes Home, Game, Wishlist, Cart, and Log out. The main content area shows a placeholder profile picture, user details (Name: Lê Văn B, Email: b_le@gmail.com, Phone number: N/A, Total spending: \$0, Membership rank: None, Membership discount: 0%), and an Edit button. Below this is a table of purchase history. At the bottom, there's a "Get history purchases" button.

Game name	Code	Date	Price	Method
Elden Ring	SQ78CXG8VWKBP9Q1	2023-04-17	\$59.99	Online banking
God of War	DP44FBXHCGBQSNI	2023-04-17	\$49.99	Online banking
Red Dead Redemption 2	HVYD219XR0CELSH3	2023-04-17	\$42.62	Online banking
Elden Ring	LV47LGEFPY2PKZ6F	2023-04-16	\$59.99	Online banking
Fallout 76	3KNE1VVD7XWDDK46N	2023-04-16	\$31.99	Online banking

Hình 20: users can view/edit their information here and view their purchase history



The screenshot shows a shopping cart interface. On the left sidebar, there are links for Home, Game, Wishlist, and Cart. The Cart link is highlighted in blue. The main content area displays two game items:

- Elden Ring**: Price \$59.99, Category Action RPG. Includes a thumbnail image and a "Remove from cart" button.
- Grand Theft Auto V**: Price \$17.54, Category Action RPG, First-Person Shooter. Includes a thumbnail image and a "Remove from cart" button.

At the bottom, it shows a total of **\$75.98** and a **Buy now** button. The browser's developer tools are open in the top right corner, showing network requests and responses related to the cart.

Hình 21: users can view/edit their chosen games here

The screenshot shows a shopping cart interface. The sidebar links are identical to the previous screenshot. The main content area displays two game items:

- Destiny 2**: Price \$10.00. Includes a thumbnail image and a "Remove from cart" button.
- God of War**: Partially visible below the Destiny 2 item.

A modal dialog box titled "Choose your payment method" is overlaid on the Destiny 2 item. It contains two options: "momo mobile money" and "Choose". The "momo mobile money" option is highlighted with a red border. Below the modal, the text "Category: Action RPG, Hack & Slash, Story-Rich" is visible. At the bottom, it shows a total of **\$39.59** and a **Buy now** button.

Hình 22: user will choose a payment method here to purchase their games



3.3.2 Admin

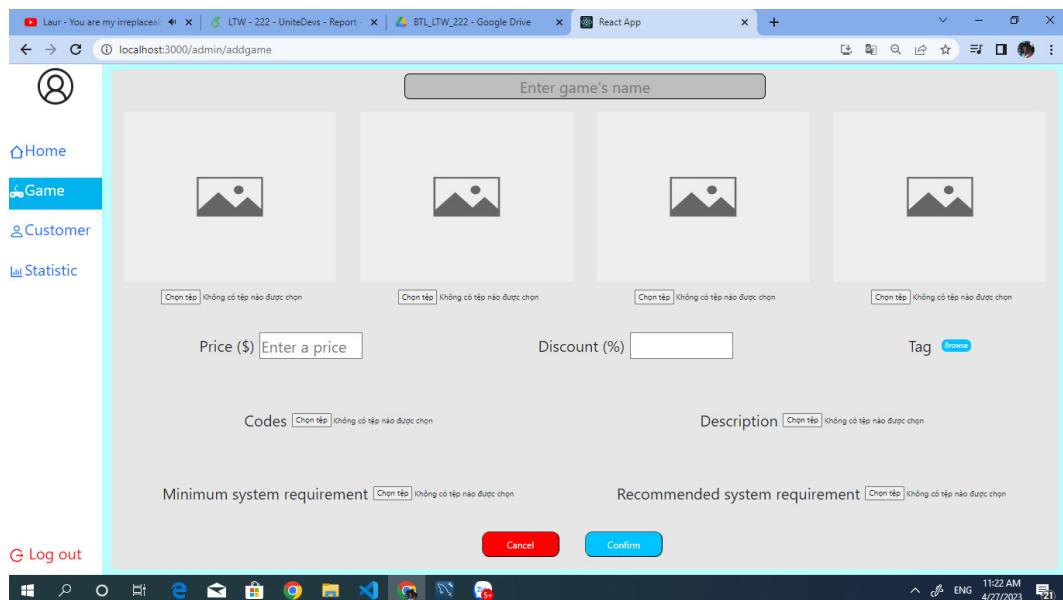
The screenshot shows a web browser window with a light blue header bar containing several tabs. The active tab is 'localhost:3000/admin/home'. The main content area has a sidebar on the left with icons for Home, Game, Customer, and Statistic, and a 'Log out' button. The main area displays five game cards under a 'Best sellers' heading. Each card includes a thumbnail, the game title, its price in a blue button, and the number of sales below it. The games shown are 'Destiny 2' (\$55.19, SOLD: 3), 'God of War' (\$29.99, SOLD: 3), 'Red Dead Redemption II' (\$38.36, SOLD: 3), 'Resident Evil 4' (\$59.99, SOLD: 3), and 'Destiny 2: Season of Defiance' (\$10.00, SOLD: 3).

Hình 23: This page is the home page when an admin logs in the website

The screenshot shows a web browser window with a light blue header bar containing several tabs. The active tab is 'localhost:3000/admin/gamelist'. The main content area has a sidebar on the left with icons for Home, Game, Customer, and Statistic, and a 'Log out' button. The main area displays a table of game data with columns for #, Name, Price, Number of solds, Ratings, and Action. Each row contains a game name, its price, the number of sales, its rating (represented by yellow stars), and a 'Detail' button. At the bottom of the table are two buttons: 'Delete game' (red) and 'Add a game' (blue). The table data is as follows:

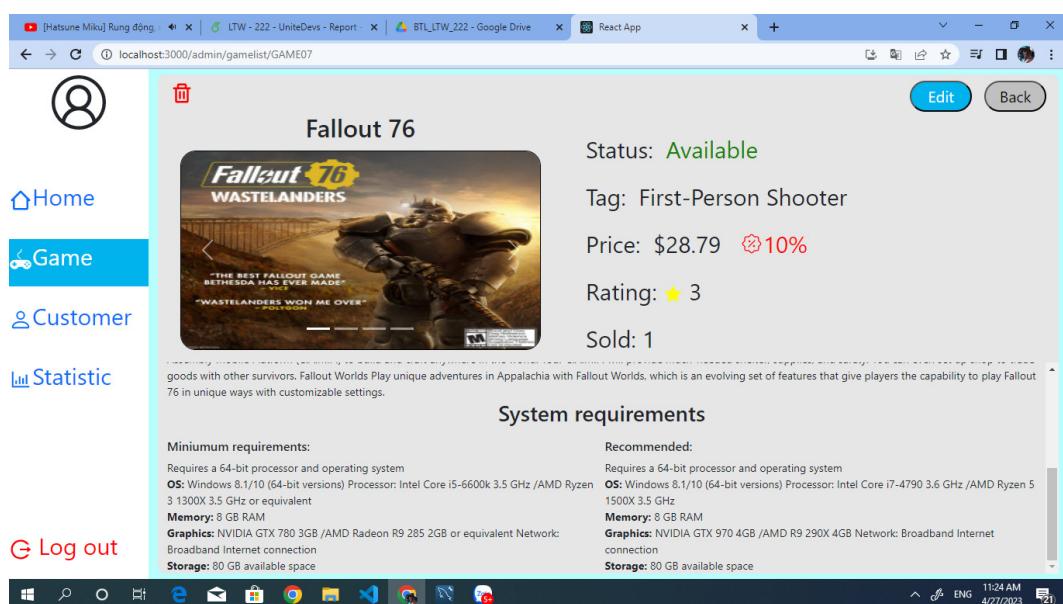
#	Name	Price	Number of solds	Ratings	Action
1	Destiny 2	\$10	2	★ 4.5	Detail
2	Elden Ring	\$59.99	3	★ 0	Detail
3	Fallout 76	\$31.99	1	★ 3	Detail
4	God of War	\$49.99	3	★ 0	Detail
5	Grand Theft Auto V	\$19.49	1	★ 4	Detail
6	Hogwarts Legacy	\$49.99	1	★ 4.1	Detail
7	MONSTER HUNTER RISE	\$39.99	1	★ 4.7	Detail
8	Parquet	\$15.99	1	★ 3.8	Detail
9	Red Dead Redemption 2	\$42.62	3	★ 0	Detail
10	Resident Evil 4	\$59.99	3	★ 0	Detail
11	Riddle Joker	\$19.99	1	★ 4.8	Detail
12	The Great War: Western Front™	\$19.99	1	★ 4	Detail
13	The Last of Us™ Part I	\$60.00	2	★ 0	Detail

Hình 24: Admins can view the game data that are currently available in the server



The screenshot shows a web-based application for managing games. On the left, there's a sidebar with navigation links: Home, Game (which is selected and highlighted in blue), Customer, and Statistic. Below these are Log out and a user profile icon. The main content area has a title 'Enter game's name' with a placeholder icon. Below it are four input fields for images, each with a placeholder 'Chọn tệp' (Select file). There are also fields for 'Price (\$)' with placeholder 'Enter a price', 'Discount (%)' with a placeholder input field, and a 'Tag' dropdown menu. Below these are sections for 'Codes' and 'Description', both with placeholder text. At the bottom are buttons for 'Cancel' and 'Confirm'.

Hình 25: Admins can add more games into the website by using this page



The screenshot shows a detailed view of a game entry. The sidebar on the left is identical to the previous screenshot. The main content shows a thumbnail image of 'Fallout 76: Wastelanders'. To the right of the image are details: Status: Available, Tag: First-Person Shooter, Price: \$28.79, Discount: 10%, Rating: ★ 3, and Sold: 1. Below this is a section titled 'System requirements' with two tables: 'Minimum requirements' and 'Recommended'. Both tables list processor, RAM, graphics, and storage requirements. At the bottom of the page are 'Edit' and 'Back' buttons.

Hình 26: Admin can view the game detail here and further more they can click on edit to change the game's properties



The screenshot shows a web-based application for managing games. On the left, there's a sidebar with navigation links: Home, Game, Customer, and Statistic. The 'Game' link is highlighted. The main content area is titled 'Edit game's information'. It displays four screenshots of the game 'Fallout 76 Wastelanders'. Below each screenshot is a 'Chọn tệp' (Select file) button. There are input fields for 'Price (\$)' (31,99), 'Discount (%)' (10), and a 'Tag' dropdown menu. A 'Codes' section and a 'Description' section are also present. At the bottom, there are 'Back' and 'Confirm' buttons. The status bar at the bottom right shows the date and time: 11:24 AM 4/27/2023.

Hình 27: Admins can edit the game's information such as price, discount, status, etc.... here

The screenshot shows a web-based application for managing customers. The sidebar has links for Home, Game, Customer (which is selected and highlighted in blue), and Statistic. The main area displays a table of customer data with columns: #, Customer, Email, Phone number, Total spending, and Action. The table contains six rows of data. Each row has a 'Detail' button in the Action column. A 'Delete user' button is located at the bottom right of the table area. The status bar at the bottom right shows the date and time: 11:26 AM 4/27/2023.

#	Customer	Email	Phone number	Total spending	Action
1	Lê Văn B	b_le@gmail.com		0	<button>Detail</button>
2	John Wick	wicky@gmail.com		0	<button>Detail</button>
3	John Wick	wicky1@gmail.com		0	<button>Detail</button>
4	John Wick	wicky2@gmail.com		0	<button>Detail</button>
5	John Wick	wicky3@gmail.com		0	<button>Detail</button>
6	admin	ad@gmail.com	0988776655	0	<button>Detail</button>

Hình 28: A list of user can be access via this page,



The screenshot shows a web application interface for managing customer profiles. On the left, there is a sidebar with icons for Home, Game, Customer (which is highlighted in blue), and Statistic. Below the sidebar is a red 'Log out' button. The main content area displays a customer profile for 'John Wick'. It includes a placeholder profile picture, a 'Get history purchases' button, and a summary of the customer's details: Name (John Wick), Email (wicky@gmail.com), Phone number (N/A), Total spending (\$0), Membership rank (None), and Membership discount (0%). There is also an 'Edit' button. At the bottom, there is a table header for a purchase history table with columns: Game name, Code, Date, Price, and Method.

Hình 29: admins can view customer's non-sensitive information here and they can adjust the customer's role when needed

The screenshot shows a web application interface for editing an admin's profile. On the left, there is a sidebar with icons for Home, Game, Customer, and Statistic. Below the sidebar is a red 'Log out' button. The main content area displays a placeholder profile picture. To the right, there are input fields for personal information: Name (Nguyễn Văn A), Email (a_nguyen@gmail.com), Phone number (0123456789), Address (empty), Username (a_nguyen123), Password (empty), and Re-enter password (empty). At the bottom, there are 'Cancel' and 'Confirm' buttons. The status bar at the bottom indicates the time as 11:27 AM and the date as 4/27/2023.

Hình 30: This page is dedicated for the admin to view/edit their own information

4 Implementation 2

4.1 Database

The screenshot shows the MySQL Workbench interface with the 'game_store' schema selected. The 'Tables' section lists several tables: activation_code, admin, belongs_to, category, customer, game, purchase_history, purchase_history_descri, shopping_cart, and wishlist. The 'create_database' tab is active, displaying the SQL code for creating these tables. The code includes constraints like unique non-null for 'username' and foreign key references for 'game_id' and 'customer_id'. The 'SQLAdditions' panel on the right provides help for context.

```
55     username varchar(20) unique not null,
56     userpassword varchar(20) not null
57   );
58
59   • create table wishlist(
60     game_id varchar(10) references game(id) on delete cascade on update cascade,
61     customer_id varchar(10) references customer(id) on delete cascade on update cascade,
62     primary key(game_id,customer_id)
63   );
64
65   • create table shopping_cart(
66     game_id varchar(100) references game(id) on delete cascade on update cascade,
67     customer_id varchar(10) references customer(id) on delete cascade on update cascade,
68     amount int default 1 check(amount>=1),
69     primary key(game_id,customer_id)
70   );
71
72   • create table purchase_history_description(
73     id varchar(10) primary key,
74     method varchar(15) check(method='MoMo payment' or method='Online banking'),
75     date date not null
76   );
```

Hình 31: Database for the Website

The database for our game shopping website can store and manage data related to games, customers, admins, and wishlist information. It was designed using a relational database management system (RDBMS) MySQL. The database schema includes tables for example :

- Games table: This table includes game details such as title, description, genre, release date, price, rating, publisher, and platform.
- Customers table: This table includes customer details such as name, email, address, payment information, and order history.
- Admins table: This table includes admin details such as name, email, and password.
- Wishlist table: This table includes wishlist details such as customer ID, game ID, and date added.
- Cart Table: This contains information about the customer's shopping cart, including the customer ID, game ID, quantity, and price.

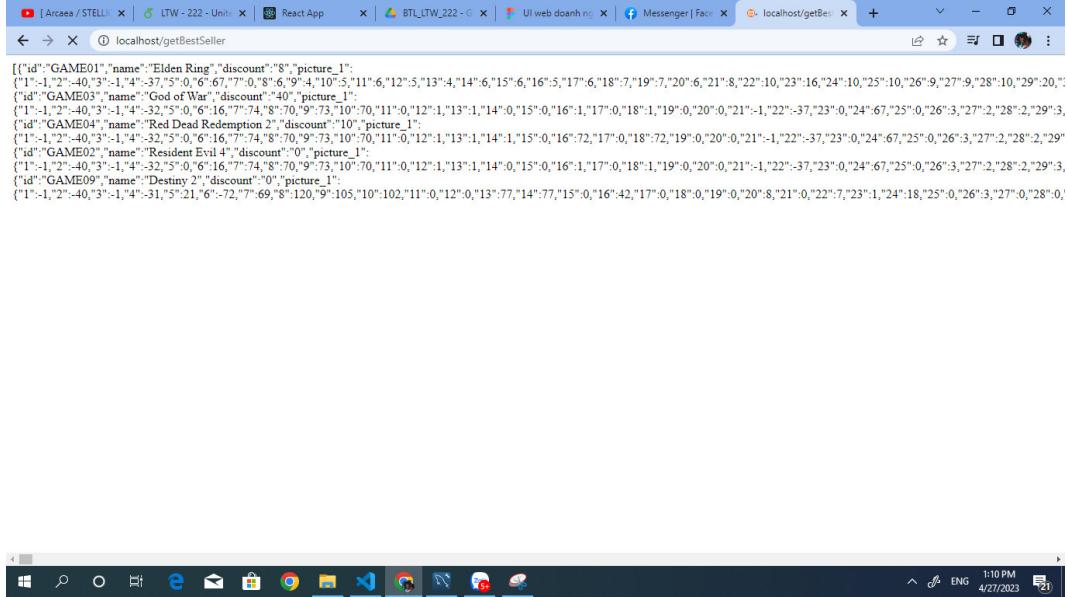
It is important that the database ensures data consistency and integrity, and enforces constraints and rules for data validation and referential integrity. This can be done using techniques such as indexing, partitioning, and caching. By implementing these techniques, the database will be optimized for high performance and scalability, which will improve user experience and increase customer satisfaction.

4.2 Server

The app was created based on the MVC model, with server include model-part and controller-part:

- Model: The model would represent the data of the website, including information about games, users, orders, and payments. This could be stored in a database and managed by an ORM (Object Relational Mapper). The model would also include any business logic needed to handle operations like adding or removing games, creating new user accounts, and processing payments.
- View: The view would be responsible for displaying the data to the user. In the case of a game selling website, this would include pages for browsing games, viewing individual game pages, creating user accounts, and completing purchases. The views would be implemented using React JS, CSS, and JavaScript

- Controller: The controller would act as the intermediary between the model and the view. It would handle user input and translate it into actions to be performed on the model. For example, when a user adds a game to their wishlist, the controller would update the model to reflect the new order and then redirect the user to the checkout page.



Hình 32: Example for using api to get data

- When a user navigates to the "getBestSeller" URL in their web browser, the request is routed to a specific controller action in the website.
 - The controller action then interacts with the model layer to retrieve the best-selling items. This involves querying a database or calling an external API.
 - Once the data has been retrieved from the model layer, the controller action prepares it for rendering in the view layer. This involves sorting or filtering the data, or transforming it into a format that is more suitable for display.
 - Finally, the controller action returns the data to the view layer, which is responsible for rendering it in the user's browser. This might involve generating HTML, CSS, and JavaScript code that displays the best-selling items in a user-friendly format.

4.3 Installation for review purpose

Go into the team's GitHub to find the repository: https://github.com/KhoaLiang/LTW_222
You will need to have a functional version of node.js, mysql 8.0 above, mysql workbench installed along with xampp. Extract the value of the file, you will find 3 folders called client, server and database. Now follow the steps below to be able to run the site:

First, enter the 'client' folder and open it in the terminal. Then type "npm i" to install all the necessary node modules, your folder should now look like this: Client node modules, public, src, and some more files

Next, go into the database folder and follow the readme.md file to set up the db

After that, openxampp "htdocs" folder and copy the content of the "server" folder inside xampp. Please make sure to NOT turn on the xampp mysql server so that it do not conflict with MySQL workbench

Finally, open the terminal for the client folder again, enter "npm start" and the site should be running now.



5 Reference Material

5.1 Front-End

5.1.1 React JS

- React Crash Course - <https://www.w3schools.com/REACT/DEFAULT.ASP>
- React Tutorial - <https://react.dev/learn>

5.1.2 CSS

- CSS Basics - <https://www.w3schools.com/css/>
- CSS3 Tutorial - https://www.tutorialspoint.com/css/css3_tutorial.htm

5.1.3 JavaScript

- JavaScript Tutorial - <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide>
- jQuery Documentation - <https://jquery.com/>

5.1.4 Bootstrap

- Bootstrap Documentation - <https://getbootstrap.com/docs/>
- Bootstrap Templates - <https://startbootstrap.com/templates/>

5.2 Back-End

5.2.1 Server-Side Programming Languages

- PHP - <https://www.php.net/>

5.2.2 Relational Databases

- MySQL - <https://www.mysql.com/>