Vietnamese - German University

Group 10

Software Requirements Specification (SRS)

for

Online Game Key Shopping System

Version 1.0
March 1, 2023

Prepared By

Vo Nguyen Duy Bach	16770
Nguyen Gia Bao	16885
Hoang Duc Hung	16207
Hoang Ngoc Anh Khoi	15900
Pham Phuc Khoa	16765
Le Duc Minh	16669
Nguyen Tuan Ngoc	15809
Tran Duc Cao Nguyen	14117

Table of Contents

I. Introduction

I.I. Purpose:

The purpose of this document is to show the development process of the Online Game Key Shopping System. The document consists of Use Cases, Sequence Diagrams, etc. Several methods of Software Engineering are imple mented to help developers and clients understand the application.

I.II. Intended audience and Reading suggestions:

The project is a prototype for an online store system and it is restricted within college premises. This has been implemented under guidance of Dr. Huynh Trung Hieu. This project is useful for customers to purchase games remotely. In addition, some businesses which have the business model of retailing game codes, can apply the application for e-commerce.

I.III. Project scope:

The application scope is to create a place to instantly provide activation code for customers without moving an inch from their chair. The system is based on a relational database with product purchase and management functions. Moreover, we hope to provide a comfortable user experience alongside with the best pricing available.

I.III. References:

- Elmasri, R. and Navathe, S. (2020) Fundamentals of Database Systems. Hoboken, NJ: Pearson.on) - Elmasri Navathe
- Peterson, L.L. and Davie, B.S. (2022) Computer Networks: A systems approach. Cambridge, MA: Morgan Kaufmann Publishers, an imprint of Elsevier.
- React JS Documentation: https://react.dev/
- Node JS Documentation : https://nodejs.org/en/docs

II. Overall Descriptions

II.I. Product perspective:

A distributed online store database system stores the following information:

- Customer description:

+Includes customer id, email, password, e-mail, first name, middle name, last name, age.

- Product description:

+ Includes product ID, title, genre, developer, publisher, release date, description, price, quantity.

- **Product file description:**

+ Includes product ID, cover images, banner image and description images.

- Genre description:

+ Includes genre ID and game genres.

- Admin description:

+ Includes admin ID, password, name, e-mail, age.

- Cart description:

+ Includes cart ID, user ID, quantity and total price.

- Cart inventory description:

+ Includes cart inventory ID, cart ID, product ID and quantity

- Order description:

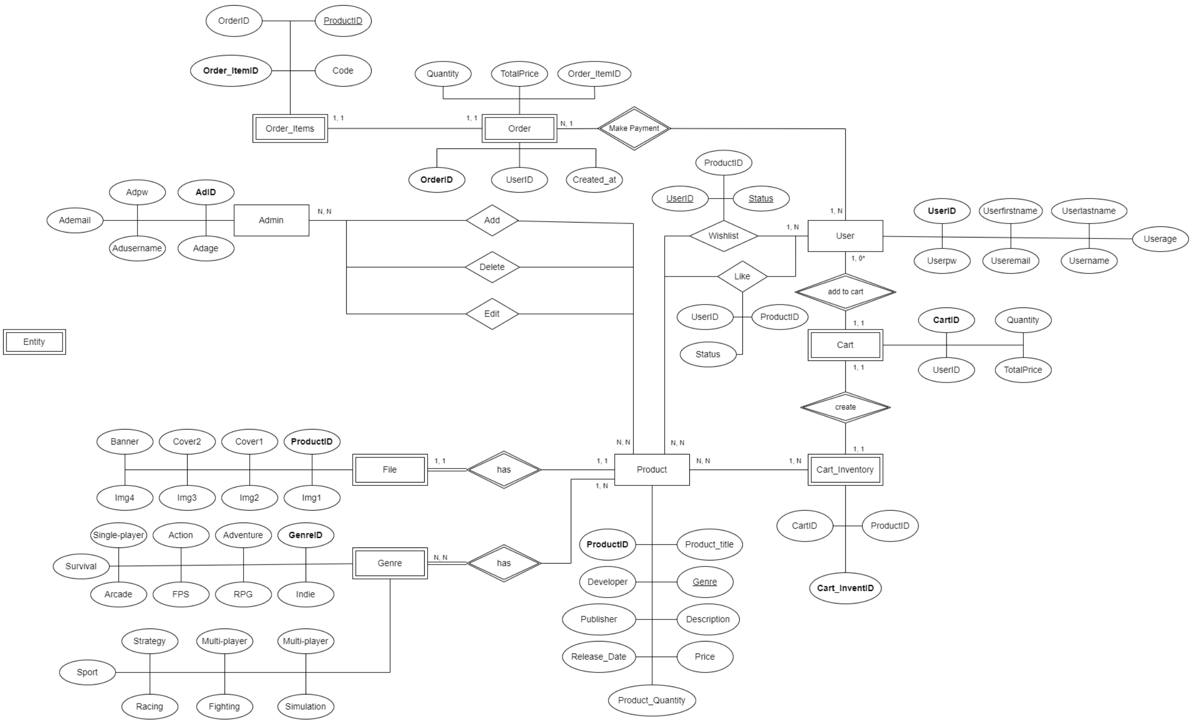
+ Includes order ID, user ID, quantity, total price, order item ID and creation date.

- Order items description:

+ Includes order items ID, product ID, order ID and code

II.II. Product features:

The major features of online store database system as shown in below entity-relationship model:



Entity-Relationship Model

Strong Entities	Weak Entities
- User - Admin - Product	- Cart - Cart Inventory - Order - Order Items - Product File - Genre - Like - Wishlist

II.III. User class and Characteristics:

Users of the system should be able to view product, add to cart, purchase product, and receive activation code of purchased products. The system will support two types of user's privileges, Customer and Admin. Customer will have access to customer functions, and admin will have access to both customer and product management functions. The customer should be able to do the following functions.

II.III.a. User:

- User signs up account.
- User signs in.
- User adds product to Cart.
- User wishlists product.
- User likes product.
- User edits user's profile information.
- User purchases product.

II.III.b. Admin:

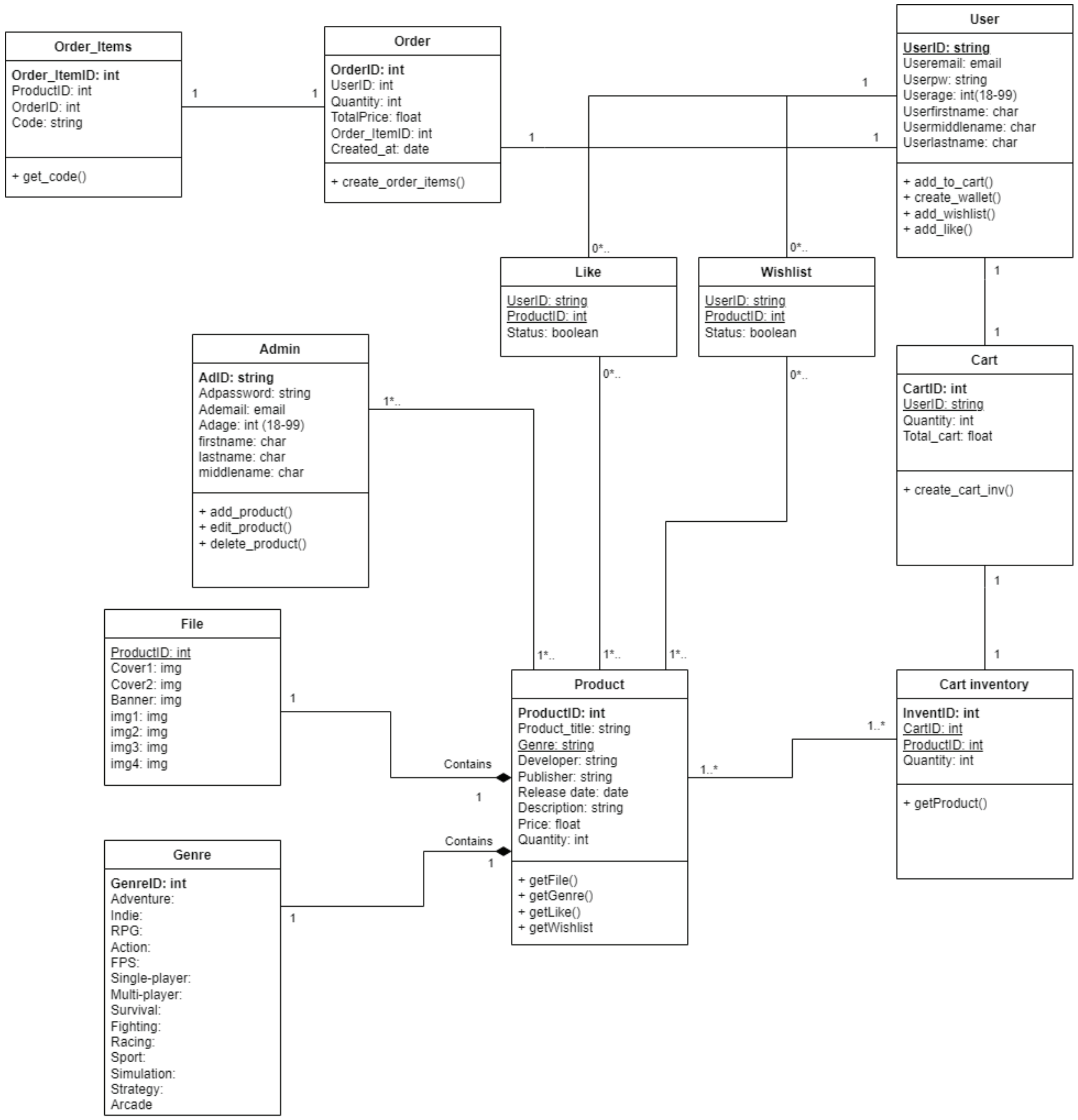
- Admin signs up account
- Admin signs in.
- Admin adds product to Database.
- Admin edits product.
- Admin deletes product.
- Admin views website's dashboard.
- Admin edits admin's profile information.
- Admin adds discount promotion.

II.IV. Operating environment:

- Operating system: Windows / MacOS / Linux / Ubuntu
- Web browser: Google Chrome, Safari, Microsoft Edge
- Language: Javascript
- Framework/ Library: ReactJS, NodeJS, ExpressJS.
- Database: PostgreSQL.

II.V. Design and Implementation constraints:

As the system is combined of several entities, the design and implementation constraints need to be built explicitly. The system design is illustrated in the below class diagram.



Class Diagram

III. System Features

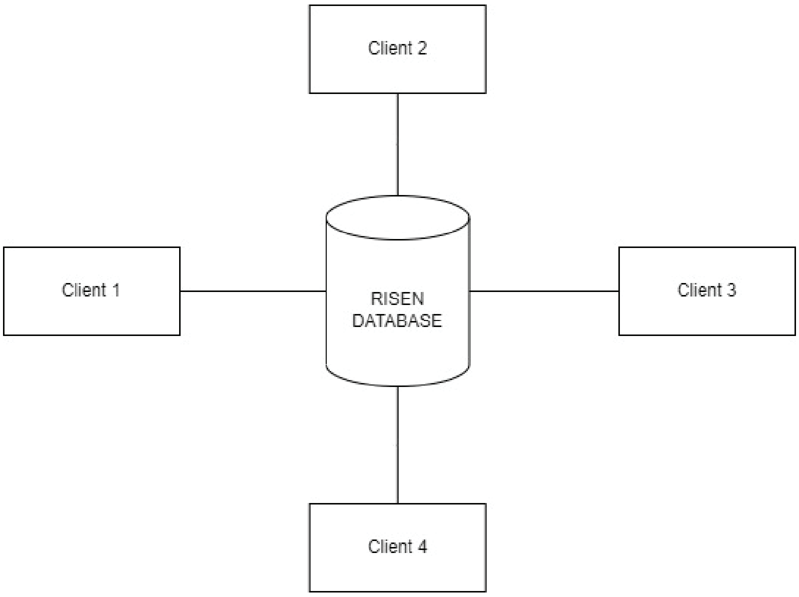
III.I. Description and Priority:

The online store system holds the information of games, customer's references (wishlist and like), customer inventory and cart inventory. Overall, this project has a medium priority as E-commerce platform brings the convenience to customers.

III.II. Functional Requirements:

a. Centralized Database:

Centralized database implies that a database is located, stored and maintained in a single location. A centralized data platform optimizes better for its scalability, better tracking for business processes and minimizes risks for businesses while preventing negative impact of inaccuracies and redundancies.



Centralized Database Illustration

b. Client/Server System:

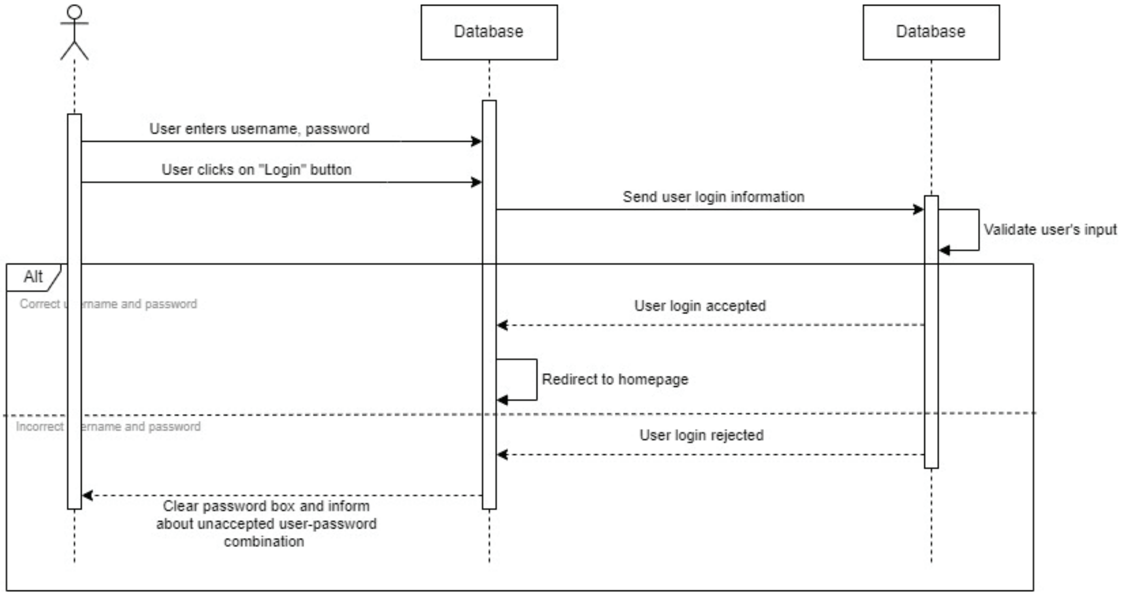
The phrase "client/server" primarily refers to an architecture or logical separation of duties, where the client is the application (also known as the front-end) and the server is the database management system (also known as the back-end).

In a distributed system called a client/server, all data are stored in the server and and all applications execute on the client sites.

c. Sequence Diagrams:

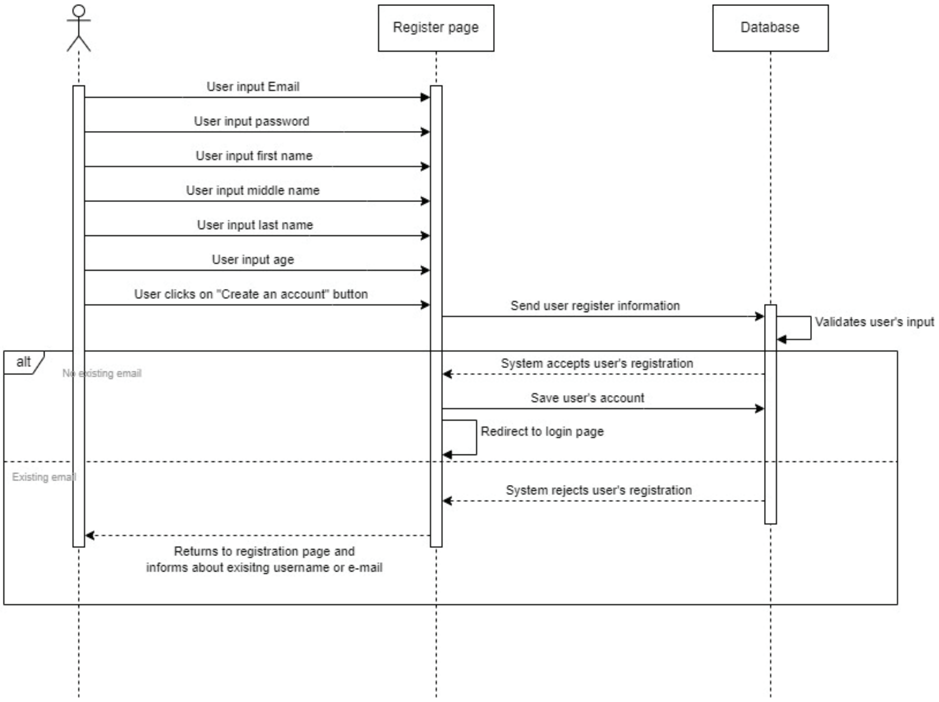
The following sequence diagram demonstrates how the interaction between process is done in a time sequence. These are for developers and business professionals to understand requirements for a new system.

c.1 Login Function:


Database
Database
User enters username, password
User clicks on "Login" button
Send user login information
Validate user's input
Alt
Correct u ername and password
User login accepted
Redirect to homepage
Incorrect ername and password
User login rejected
Clear password box and inform
about unaccepted user-password
combination

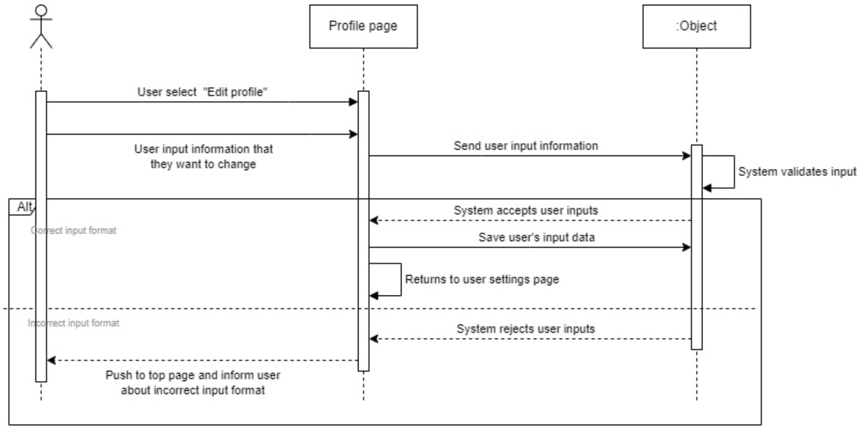
Login Function Sequence Diagram

c.2. Register Function:



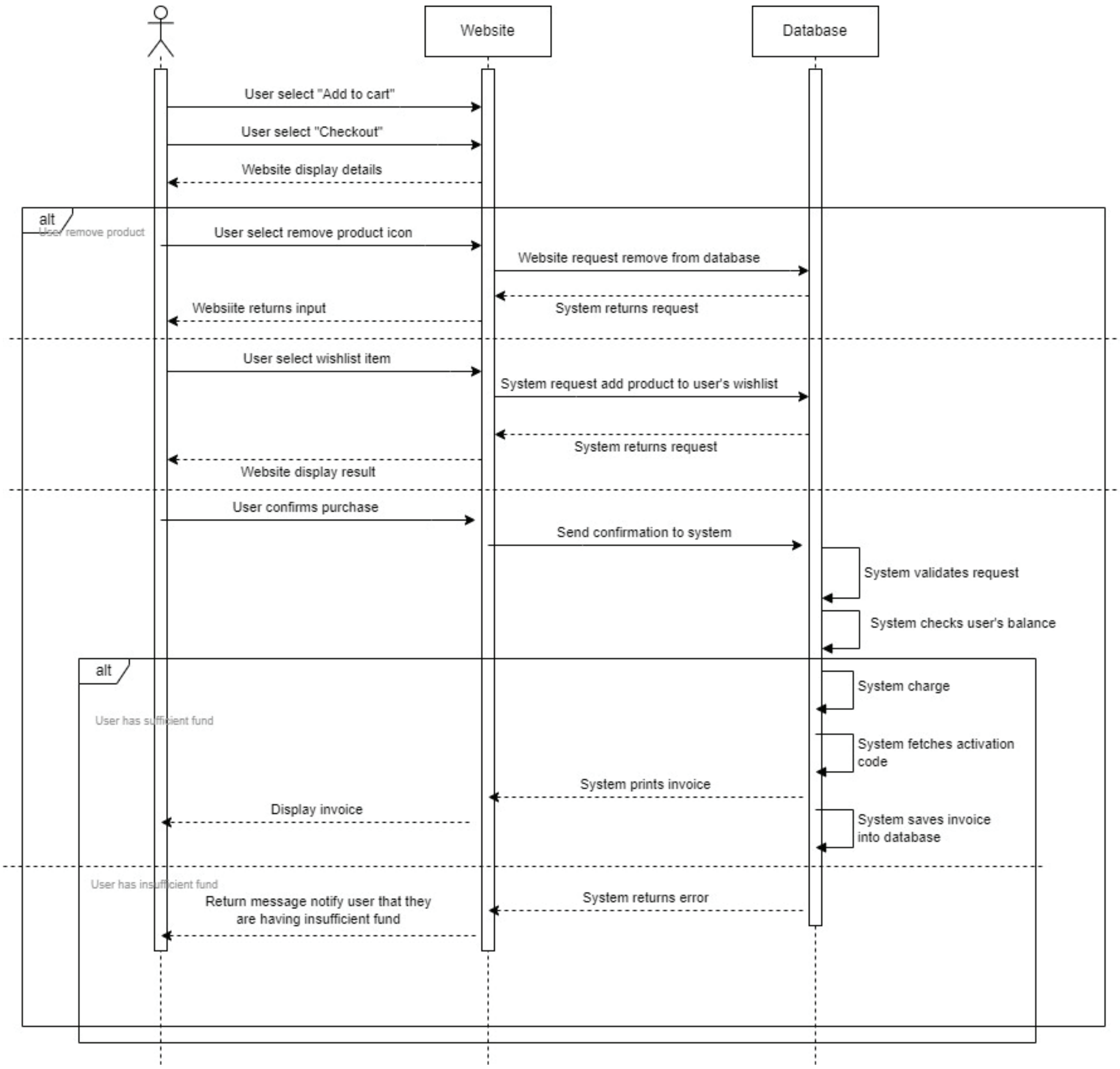
Register Function Sequence Diagram

c.3. Edit profile function:


Profile page
:Object
User select "Edit profile"
User input information that
they want to change
Send user input information
System validates input
Alt
Correct input format
System accepts user inputs
Save user's input data
Returns to user settings page
Incorrect input format
System rejects user inputs
Push to top page and inform user
about incorrect input format

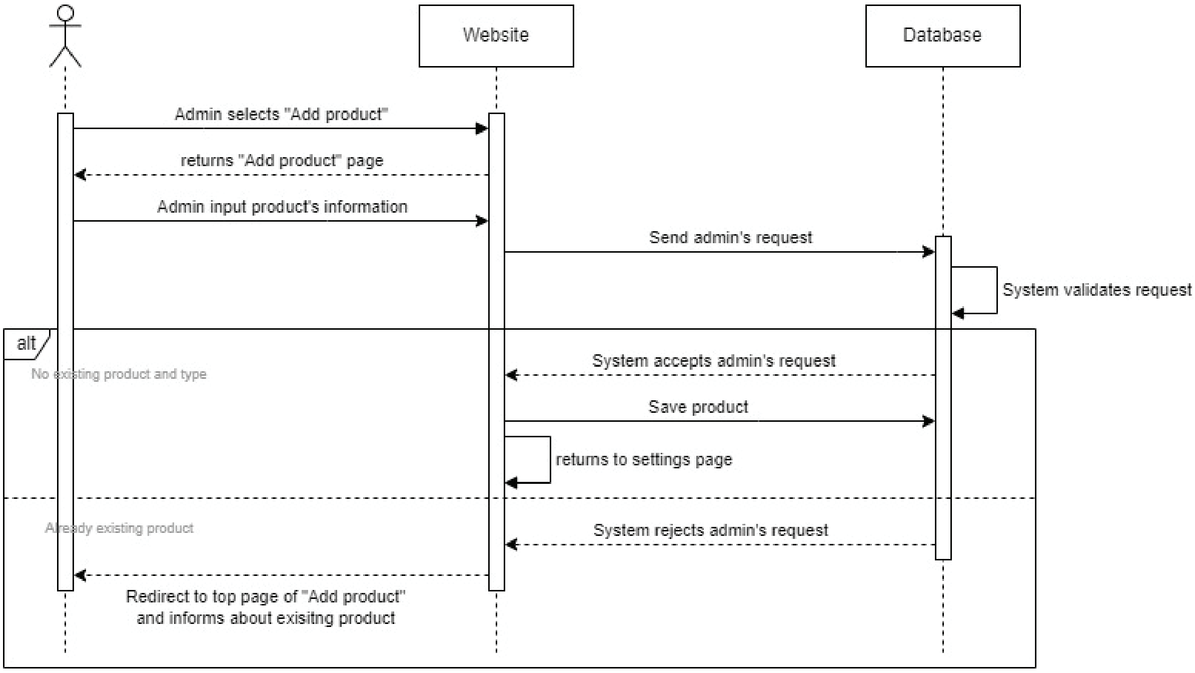
Edit Profile Function Sequence Diagram

c.4. Product Payment Function:


Website
Database
User select "Add to cart"
User select "Checkout"
Website display details
alt
User remove product
User select remove product icon
Website request remove from database
System returns request
Websiite returns input
User select wishlist item
System request add product to user's wishlist
System returns request
Website display result
User confirms purchase
Send confirmation to system
System validates request
System checks user's balance
alt
User has sufficient fund
System charge
System fetches activation code
System prints invoice
Display invoice
System saves invoice into database
User has insufficient fund
Return message notify user that they are having insufficient fund
System returns error

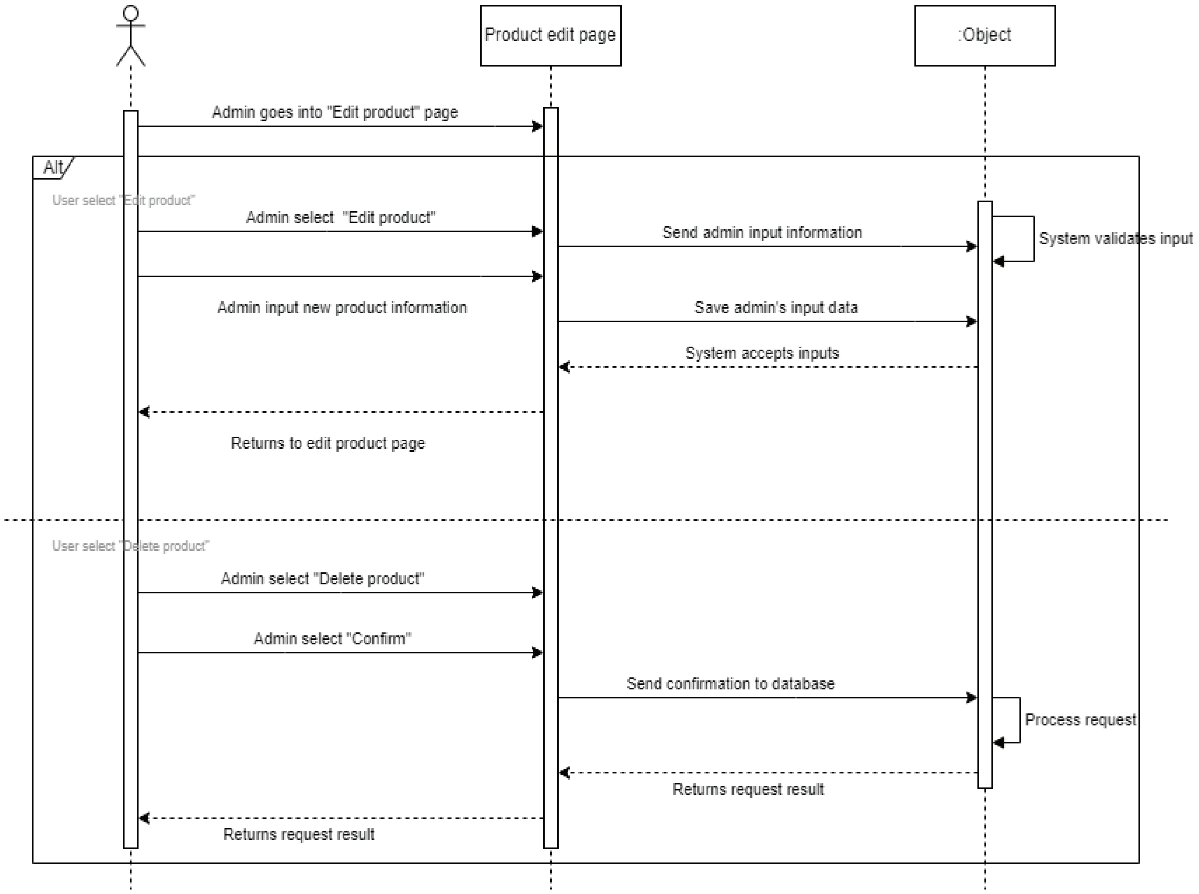
Register Function Sequence Diagram

c.5. Add Product Function:


Website
Database
Admin selects "Add product"
returns "Add product" page
Admin input product's information
Send admin's request
System validates request
alt
No existing product and type
System accepts admin's request
Save product
returns to settings page
Already existing product
System rejects admin's request
Redirect to top page of "Add product"
and informs about exisitng product

Add Product Function Sequence Diagram

c.6. Edit Product Function:


Product edit page
:Object
Admin goes into "Edit product" page
Alt
User select "Edit product"
Admin select "Edit product"
Send admin input information
System validates input
Admin input new product information
Save admin's input data
System accepts inputs
Returns to edit product page
User select "Delete product"
Admin select "Delete product"
Admin select "Confirm"
Send confirmation to database
Process request
Returns request result
Returns request result

IV. External interface requirements

IV.I User interfaces:

Front-end software: ReactJS.
Back-end software: NodeJS, ExpressJS.

IV.II Hardware interfaces:

Any working OS.
A browser that supports HTML/CSS and Javascript (ideally Chrome based website).

IV.III. Software interfaces:

Software used	Description
Operating system	This website can functional normally on any operating system
Database	To save records of users and products information, we have chosen noSQL database

V. Non-functional requirements

V.I. Performance requirements

The use case description is a written description describing how users will perform tasks on the website. Moreover, we use these descriptions for function checklist.

The following use cases are listed as follow.

a. Use case description:

a.1. User login:

Use case name	User login
Primary actors	Customer, Admin
Secondary actors	Database
Description	User logs in into the website
Trigger	User press login
Piority	1
Pre-condition	User has registered account.
Post-condition	None
Basic workflow	1. User select "Login" 2. User entered username 3. User entered password 4. System validates user's username and password 5. System redirect to homepage
Alternative workflow	At step 4: 4.1. If incorrect username or password -> return message and clear out password box 4.2. If password is left blank -> return error message 4.3. If username is left blank -> return error message
Exceptions	

a.2. Account registration:

Use case name	Account registration
Primary actors	Customer, Admin
Secondary actors	Database
Description	User registers an account to purchase product
Trigger	User presses "Create an account"
Piority	1
Pre-condition	None
Post-condition	None
Basic workflow	1. User select "Create an account" 2. User enters e-mail 3. User enters password 4. User enters first name 5. User enters middle name (optional) 6. User enters last name 7. User enters age 8. User select "Create an account" 9. System validates user's information 10. System updates account's information 11. Returns to login page
Alternative workflow	None
Exceptions	At step 2: 3.1. If incorrect email format (xx@xxxxx.xxx) -> return error message 3.2. If e-mail is left blank -> return error message At step 3: 3.1 If incorrect password format (At least 8 letters, at least 1 captial letter included) -> return error message 3.2 If password is left blank -> return error message At step 4: 4.1 If DoB is left blank -> return error message At step 5: 5.1 If first name is left blank -> return error message At step 6: 5.1 If last name is left blank -> return error message At step 7: 7.1 If age is left blank -> return error message At step 8: 8.1 If existing e-mail address found in the database -> return error message

a.3. Edit profile:

Use case name	Edit profile
Primary actors	Customer, Admin
Secondary actors	Database
Description	User change their information on the website
Trigger	User press “Edit profile”
Piority	2
Pre-condition	User already logged in
Post-condition	None
Basic workflow	1. User select “Edit profile” 2. User entered information that they want to change 3. User select “Confirm” 4. System validates user’s information 5. System changes information 6. System display successful message 7. Returns to user dashboard
Alternative workflow	At step 2: 2.1 If any information box if left blank -> returns error message At step 3: 3.1 If incorrect information format -> return message
Exceptions	At step 3: 3.1 If user press “Cancel”, return to account settings

a.4.Add product:

Use case name	Add product
Primary actors	Admin
Secondary actors	Database
Description	Admin adds a product to the website
Trigger	Admin presses "Add product" button
Piority	1
Pre-condition	Admin account already logged in
Post-condition	
Basic workflow	1. Admin selects "Add product" 2. Admin enters product title 3. Admin selects product genre (minimum 1) 4. Admin enters price 5. Admin enters developer 6. Admin enters publisher 7. Admin enters release date 8. Admin enters product description 9. Admin enters product images 10. Admin selects Submit button 11. System validates admin's input 12. System updates admin's input 13. Return to Admin dashboard
Alternative workflow	None
Exceptions	At step 2: 2.1 If product title is left blank -> returns error message At step 3: 3.1 If product genre is left blank -> returns error message At step 4: 4.1 If price is left blank -> returns error message At step 5: 4.1 If developer is left blank -> returns error message At step 6: 6.1 If publisher is left blank -> returns error message At step 7: 7.1 If release date is left blank -> returns error message At step 8: 8.1 If product description is left blank -> returns error message At step 9: 1.1 If image is left blank -> returns error message At step 11: 11.1 If if existing product is found -> returns error message

a.5. Edit product:

Use case name	Edit product
Primary actors	Admin
Secondary actors	Database
Description	Admin deletes a product on the website
Trigger	User press “Edit product”
Piority	2
Pre-condition	Admin account already logged in
Post-condition	None
Basic workflow	1. Admin select “Edit product” 2. Admin select product to edit 3. Admin input products information 4. System validates admin’s input 5. System updates product’s information into database 6. System updates product on the website
Alternative workflow	At step 2: 2.1 If user wishes to remove product -> select the “Remove product” button 2.1.1 Admin confirms that they want to delete their product 2.1.2 System validates admin’s input 2.1.3 System updates product’s information 2.1.4 System updates product on the website
Exceptions	None

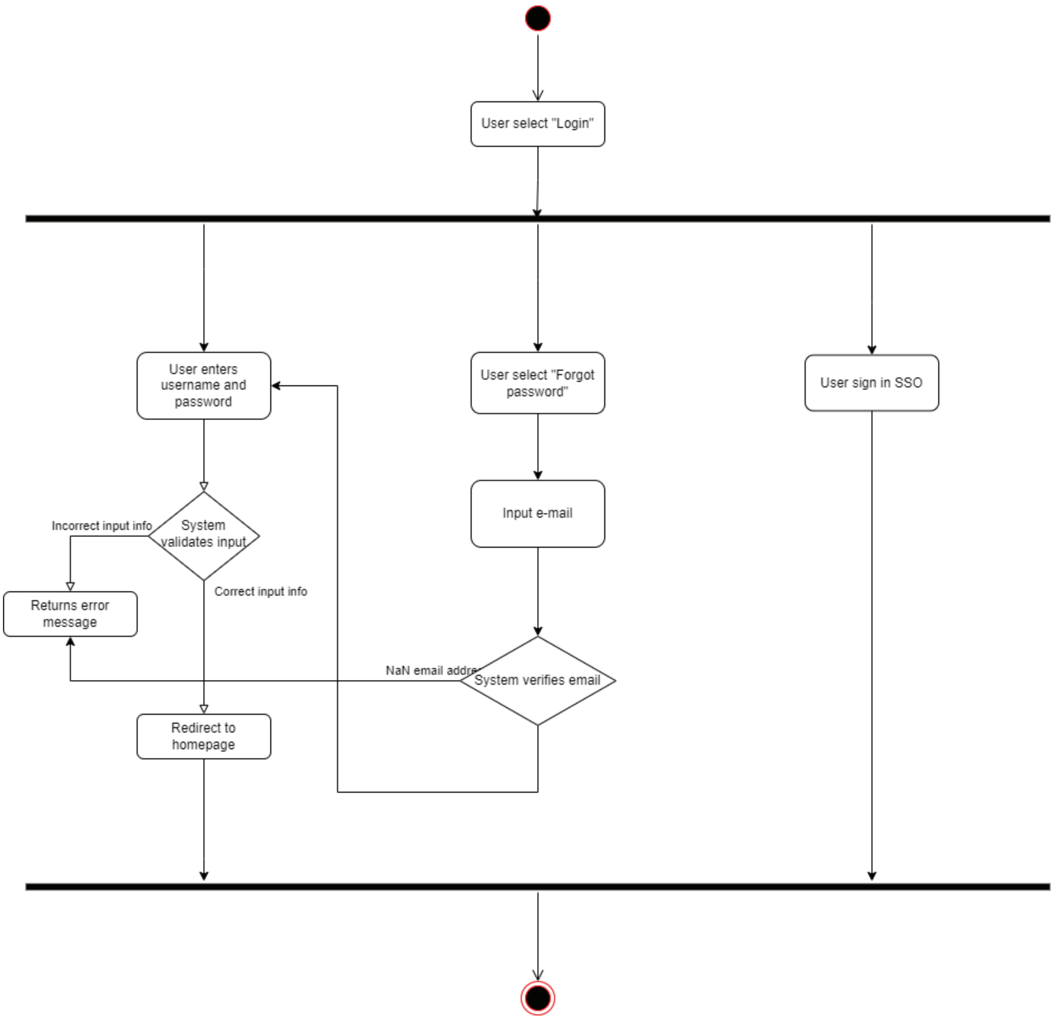
a.6. Purchase product:

Use case name	Purchase product
Primary actors	User
Secondary actors	Database
Description	Customer purchases product from the website
Trigger	User presses “Purchase product”
Piority	
Pre-condition	There is an existing product in the cart
Post-condition	None
Basic workflow	1. User press “Product” 2. User press “Add to cart” 3. User press “Checkout” 4. System display checkout UI 5. Customer confirms purchase 6. System checks user’s balance 7. System charge fee 8. System saves activation code into user’s database 9. System display invoice 10. Add invoice into the database Lorem ipsum
Alternative workflow	At step 4: 1.1. If customer select wishlist item, item will be removed from cart and added into user’s wishlist list 1.2. If customer select remove product from cart button, item will be removed from cart At step 6: 6.1. If insufficient fund, display message “Insufficient fund” then returns to “Checkout”
Exceptions	None

b. Flowchart

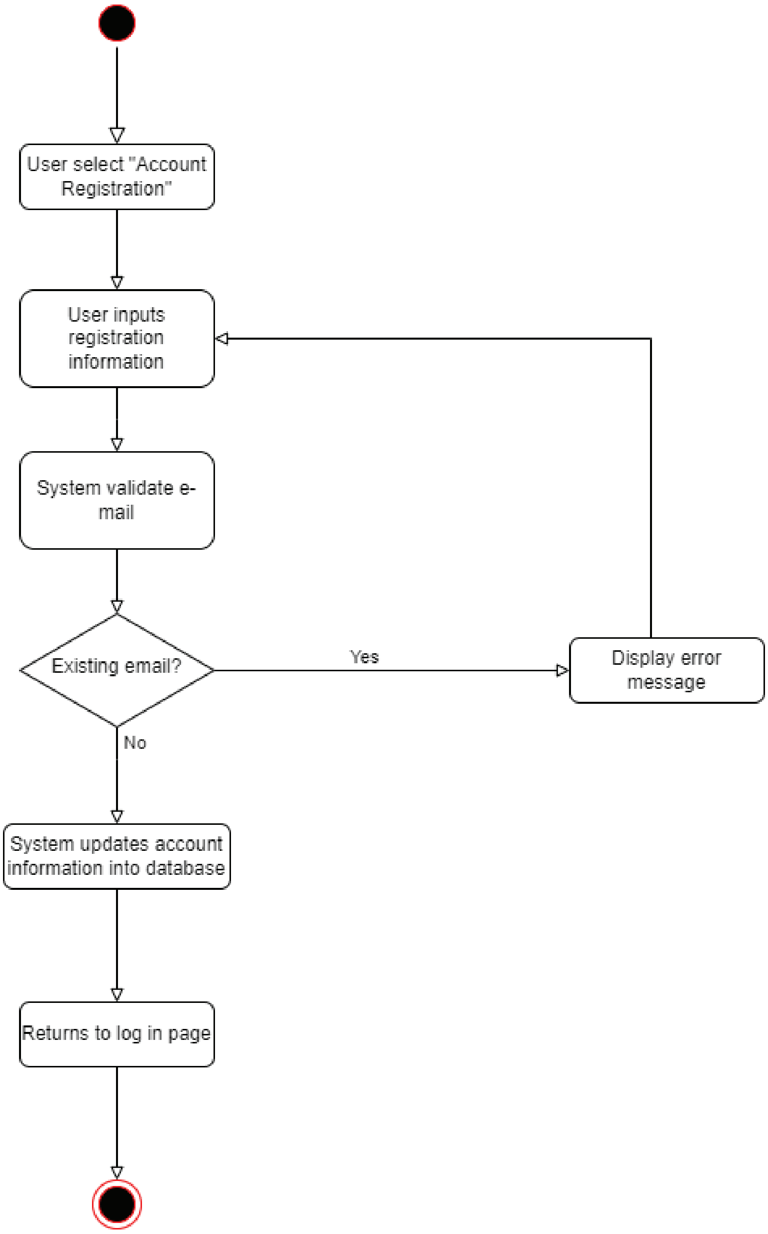
The flowchart is a picture that seperate different steps in an organized fashion. The following flowcharts are listed as follow.

b.1. User login:



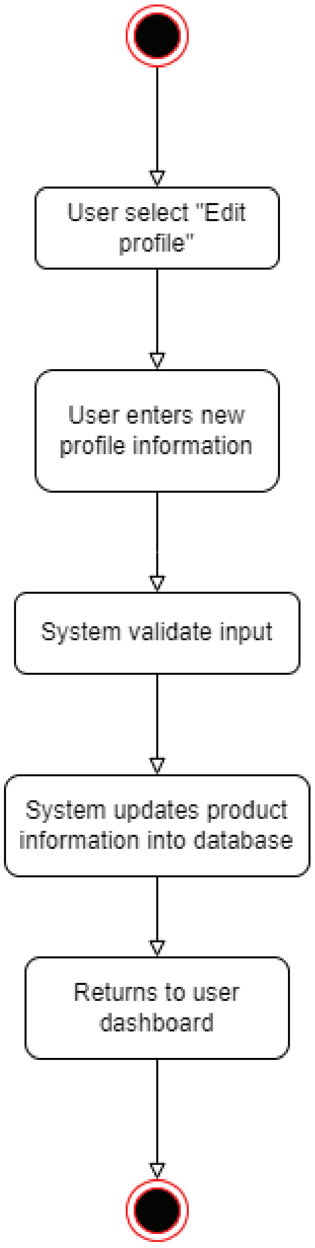
User Login Flowchart

b.2. Account Registration:


User select "Account Registration"
User inputs registration information
System validate e-mail
Existing email?
Yes
Display error message
No
System updates account information into database
Returns to log in page

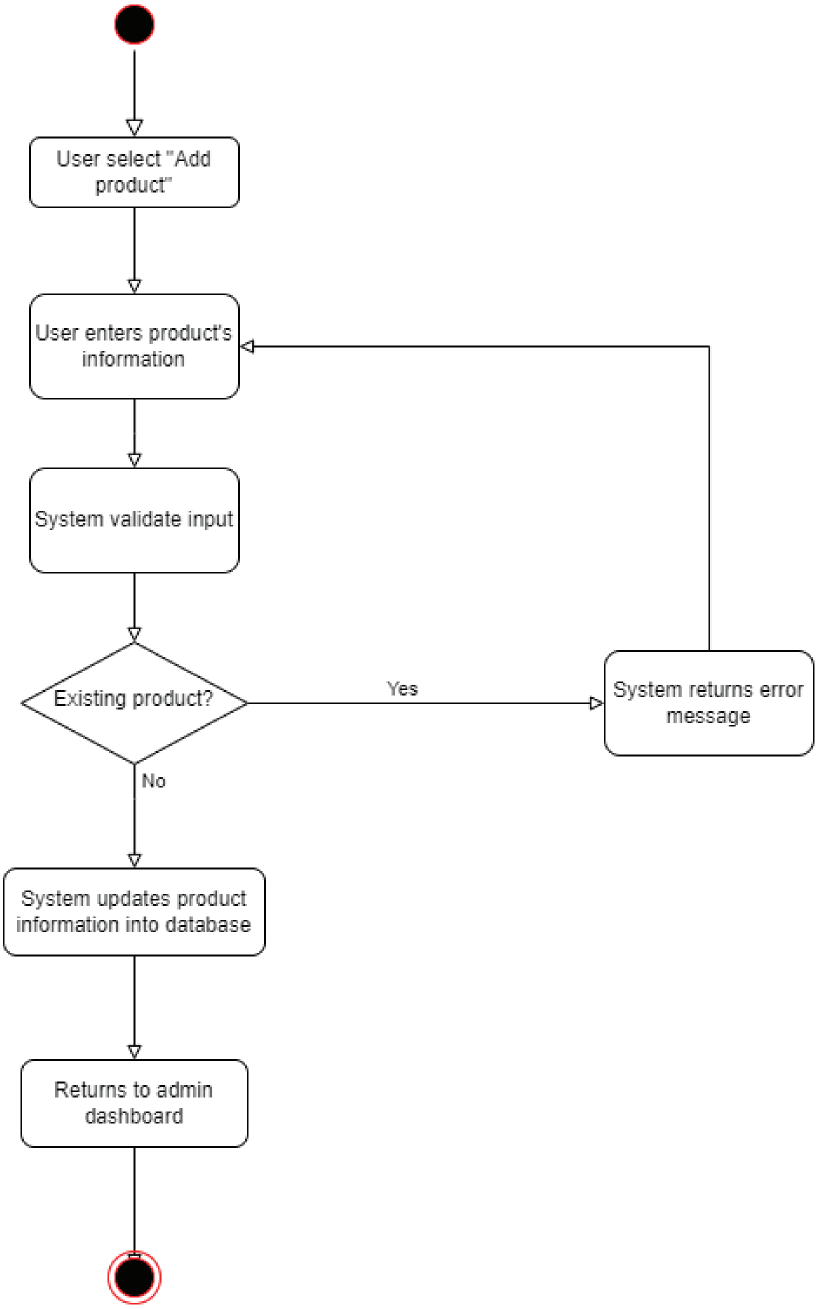
Account Registration Flowchart

b.3. Edit Profile:


User select "Edit profile"
User enters new profile information
System validate input
System updates product information into database
Returns to user dashboard

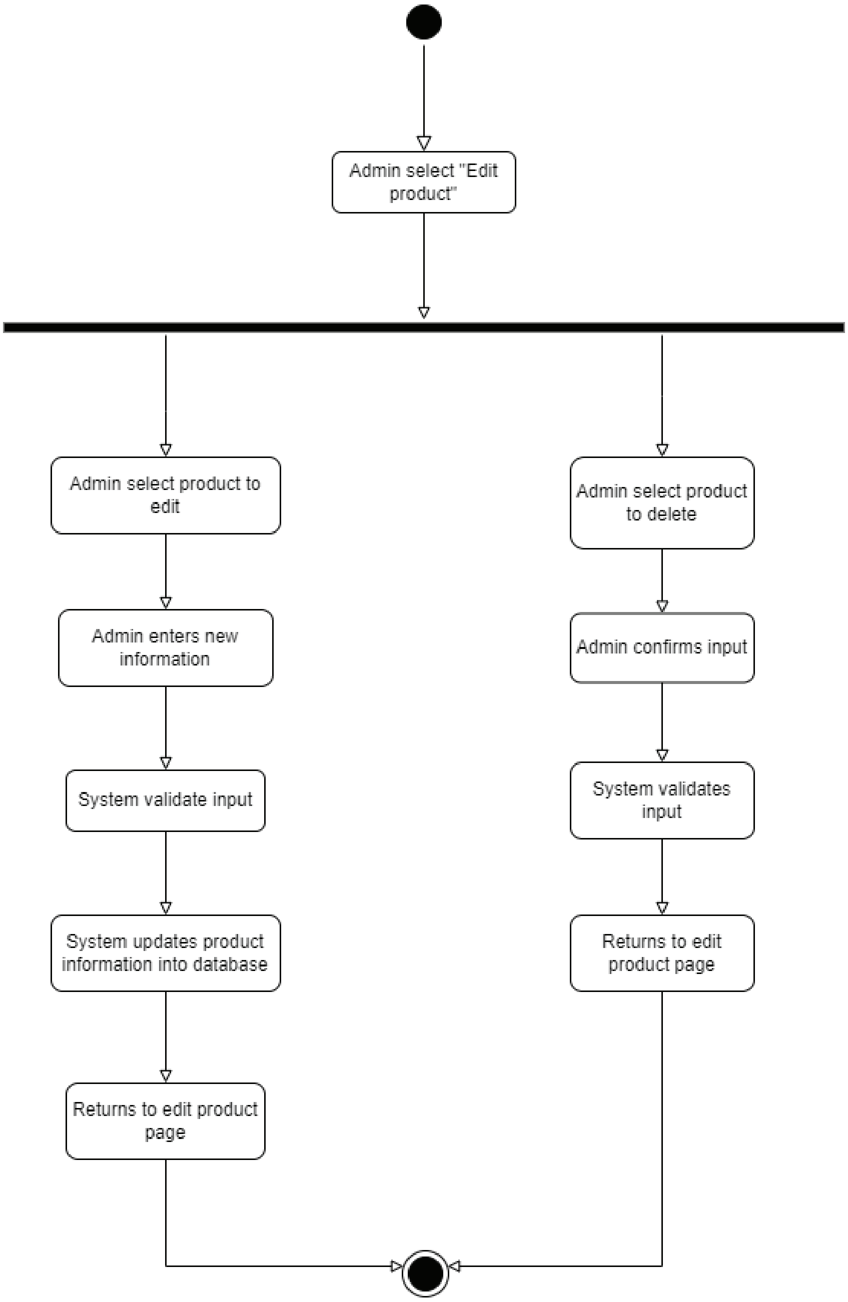
Edit Profile Flowchart

b.4. Add Product:


User select "Add product"
User enters product's information
System validate input
Existing product?
Yes
System returns error message
No
System updates product information into database
Returns to admin dashboard

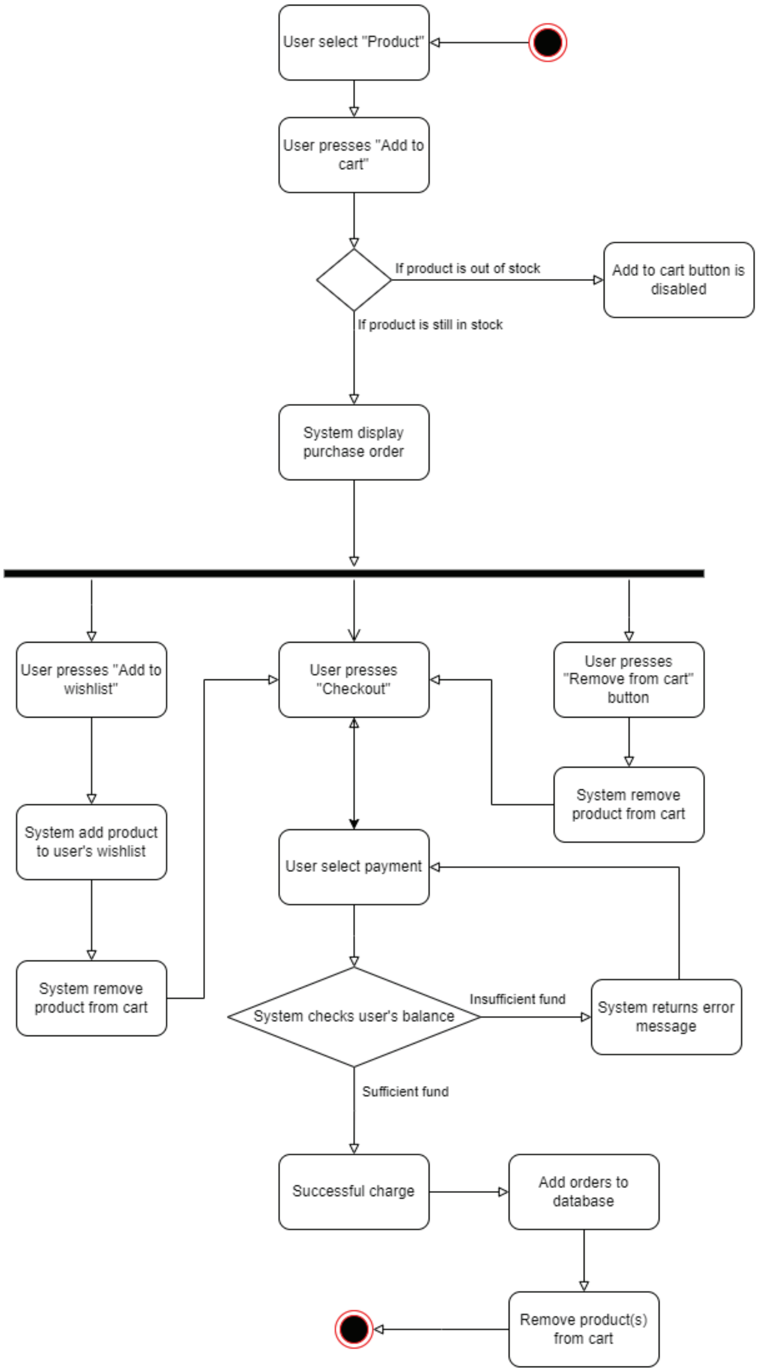
Add Product Flowchart

b.5. Edit Product:


Admin select "Edit product"
Admin select product to edit
Admin enters new information
System validate input
System updates product information into database
Returns to edit product page
Admin select product to delete
Admin confirms input
System validates input
Returns to edit product page

Edit Product Flowchart

b.6. Purchase Product:


User select "Product"
User presses "Add to cart"
If product is out of stock
Add to cart button is disabled
If product is still in stock
System display purchase order
User presses "Add to wishlist"
User presses "Checkout"
User presses "Remove from cart" button
System remove product from cart
System add product to user's wishlist
User select payment
System remove product from cart
System checks user's balance
Insufficient fund
System returns error message
Sufficient fund
Successful charge
Add orders to database
Remove product(s) from cart

Purchase Product Flowchart

V.II Security Requirements

We use the SSL/TLS security protocol to protect data sent over the Internet. This encryption prevents attackers and Internet Service Provider from viewing and alternating data exchanged by two nodes. Most website owners and operators requests SSL/TLS to keep sensitive information such as password, bank account information and other private information confidental.

V.III. Software Quality Attributes

- Availability: New product should be available on the specific date and time for customers to purchase.
- Correctness: The purchased product should be stored in the correct user's library
- Maintainability: Admins should keep up to date to majority of published games
- Usability: The quantity of games should satisfies a maximum number of customers needs.