

**E-Furniture**

**Software Requirement Specification**

Project Code: SWR302G5

Document Code: EF302 – v1.0

**Issued date: *20.01.2024***

**Record of change**

\*A - Added M - Modified D - Deleted

| **Effective Date** | **Changed Items** | **A\* M, D** | **Change Description** | **New Version** |
| --- | --- | --- | --- | --- |
| 20/02/2024 | Initial RSR file | a | Add project overview | 1.0 |
| *20/01/2024* | *Initial Document* | *a* | *V1* | 1.0 |
| *24/01/2024* | *Use Case Diagram* | *a* | *V1* | 1.0 |
| *26/01/2024* | *ER Diagram* | *a* | *V1* | 1.0 |
| *26/01/2024* | *Figma* | *a* | *v1* | 1.0 |
|  |  |  |  |  |

**SIGNATURE PAGE**

**ORIGINATOR:** Nguyễn Dương Gia Bảo - 20/02/2024

Project Manager

**REVIEWERS:** NET1702 - Group 1, Lại Đức Hùng - Lecturer

**APPROVAL:** Lại Đức Hùng - Lecturer

**TABLE OF CONTENTS**

[**1. Introduction 6**](#_3znysh7)

[1.1. Purpose 6](#_2et92p0)

[1.2. Scope 6](#_tyjcwt)

[1.3. Definitions, Acronyms, and Abbreviations 7](#_3dy6vkm)

[1.4. References 8](#_1t3h5sf)

[1.5. Overview 8](#_4d34og8)

[**2. Overall Description 9**](#_2s8eyo1)

[**3. FUNCTIONAL REQUIREMENT 11**](#_17dp8vu)

[3.1. Use Case Diagram 11](#_3rdcrjn)

[3.2. Log In 11](#_26in1rg)

[3.3. Log out 13](#_wxf9nhrjgins)

[3.4. Sign Up 15](#_qvzvflgvahm)

[3.5. View Furniture List 18](#_n9c92chie702)

[3.6. View Cart 20](#_ueu4tom2p3rc)

[3.7. Update Item’s Quantity 22](#_fppqirbo32io)

[3.8. Remove Item From Cart 24](#_pds7tm465002)

[3.9. Buy Items In Cart 26](#_b72hae5f6sgi)

[3.10. Pay By E-coin 28](#_us2wvu5ytnew)

[3.11. Pay In Cash 30](#_9iise2l0b6iu)

[3.12. View Wishlist 32](#_n2e19h3muh7i)

[3.13. Add Item To Wishlist 33](#_v4dbwqku6tog)

[3.14. Add Item To Cart 34](#_b0whv7nnqxra)

[3.15. Remove Item From Wishlist 36](#_yk5sg0lqs535)

[3.16. View Customer’s Orders 38](#_h3qba814qnvc)

[3.17. View A Furniture Detail 40](#_i8ok78e7946v)

[3.18. View A Showroom 42](#_nrbyfnrmd3n)

[3.19. Add Furniture 44](#_lnxbz9)

[3.20. Update Furniture 46](#_bd2vr2icoh4r)

[3.21. Delete Furniture 48](#_4fleju67jlsv)

[3.22. View Transactions Log 49](#_pvfv2fc1h9bo)

[3.23. Assign Voucher 50](#_wmeccumjn4t8)

[3.24. View User Reports 52](#_vbomn5wklm7h)

[3.25. View Showroom List 53](#_b87jtuynvblu)

[3.26. View Revenue and Item Reports 55](#_iahldtg16nmm)

[3.27. View Orders History 56](#_fmfw3ysgnjiz)

[3.28. Booking Date & Time 58](#_fyqb7pl59wiv)

[3.29. Buy eWallet Credit 60](#_ppeozfo5e1fa)

[3.30. Refund Into eWallet 61](#_cjig97iw4flq)

[3.31. View A Furniture List 63](#_xctlc1h886cw)

[**4. NON-FUNCTIONAL Requirements 66**](#_35nkun2)

[4.1. Usability 66](#_1ksv4uv)

[4.2. Reliability 66](#_44sinio)

[4.3. Performance 66](#_2jxsxqh)

[**5. Supporting Information 67**](#_qsh70q)

# Introduction

* This SRS file contains information regarding the construction of this application. Its many use cases, use case diagrams, scope, etc… about the web application we are building to help clear any misinformation or miscommunication between our developers and the client.

## Purpose

* We've built a visually appealing and user-friendly website for a furniture showroom that serves as a one-stop digital hub for its incredible range of products. The web interface is designed for easy navigation enabling customers to browse through diverse furniture collections at their comfort. Detailed product descriptions accompany each item, helping customers make informed decisions. Enhanced with high-quality images, customers can view furniture specifications in comprehensive detail. Our state-of-the-art virtual and augmented reality features provide an immersive shopping experience by enabling customers to visualize how the furniture would look within their own spaces. Enhancing the overall customer journey on the website even further, we've established e-commerce capabilities. This allows for online ordering, ensuring a seamless process from browsing to buying. Alongside, we offer secure payment options, giving customers the peace of mind they need while shopping.

## Scope

1. Design and Development: Create a visually appealing and user-friendly website for the furniture showroom which includes a seamless interface for browsing furniture collections and detailed product descriptions.
2. Photo and Visual Effects Integration: Incorporate high-quality pictures of the furniture and add virtual or augmented reality features to enable customers to visualize furniture in real-world settings.
3. E-Commerce Integration: Implement on-site eCommerce capabilities for online ordering and secure payments to enable easy and safe transactions.
4. Interactivity Features: Integrate a review system for customer testimonials, a contact system for inquiries and customer support, as well as links to the showroom's social media platforms.
5. Testing and launch: Conduct comprehensive testing to ensure website functionality and user experience before going live.

## Definitions, Acronyms, and Abbreviations

| **Acronyms** | **Definition** |
| --- | --- |
| *SRS* | Software requirements specification |
| API | Application Programming Interfaces |
| *ReactJS* | *Front-end javascript library* |
| *.Net* | *C# library. Use to create Back-end RESTFUL API* |
| RDBMS | Relational Database Manager System |

## References

B+ Furniture:[*https://bplusfurniture.com.vn*](https://bplusfurniture.com.vn)

## Overview

Overall, the website serves not only as a digital showroom but also as a platform for interaction, engagement, and a seamless shopping experience. The rest of the SRS file will contain a use case diagram, every use case details, interfaces, legal, copyrights, and application standards.

# Overall Description

1. Product Perspective: E-furniture application as a standalone web application, which integrates its e-wallet to help with user purchases.
2. Product Functions: Detailed product descriptions accompany each item, helping customers make informed decisions. Enhanced with high-quality images, customers can view furniture specifications in a comprehensive detail. Our state-of-the-art virtual and augmented reality features provide an immersive shopping experience by enabling customers to visualize how the furniture would look within their own spaces.
3. User Characteristics: guest, customer, staff, admin.

* Guests: As initial users, guests have access to browse through the website and its available content. However, their interaction is limited and does not extend to key actions such as purchasing items, wishlist creation, booking a showroom visit, or e-wallet management.
* Customers: Active registered users who, in addition to viewing the site's content, have the prerogative to execute actions such as purchasing items, creating wishlists, making reservations for showroom visits and managing their e-wallet transactions.
* Staff: Crucial to the operational activities, the staff is categorized into:
  + Consultants: Their responsibility lies in assisting the customers in their selection process, ensuring that they choose the most suitable furniture.
  + Delivery Staff: They handle the delivery of the items and take care of any transactions that are made in cash at the time of delivery.
  + Inventory Staff: They commit to oversee stock management within the store.
* Admin: Admin role involves handling reports, transaction logs, and revenue tracking. They assign available vouchers to items and oversee stock management, ensuring smooth operational procedures.

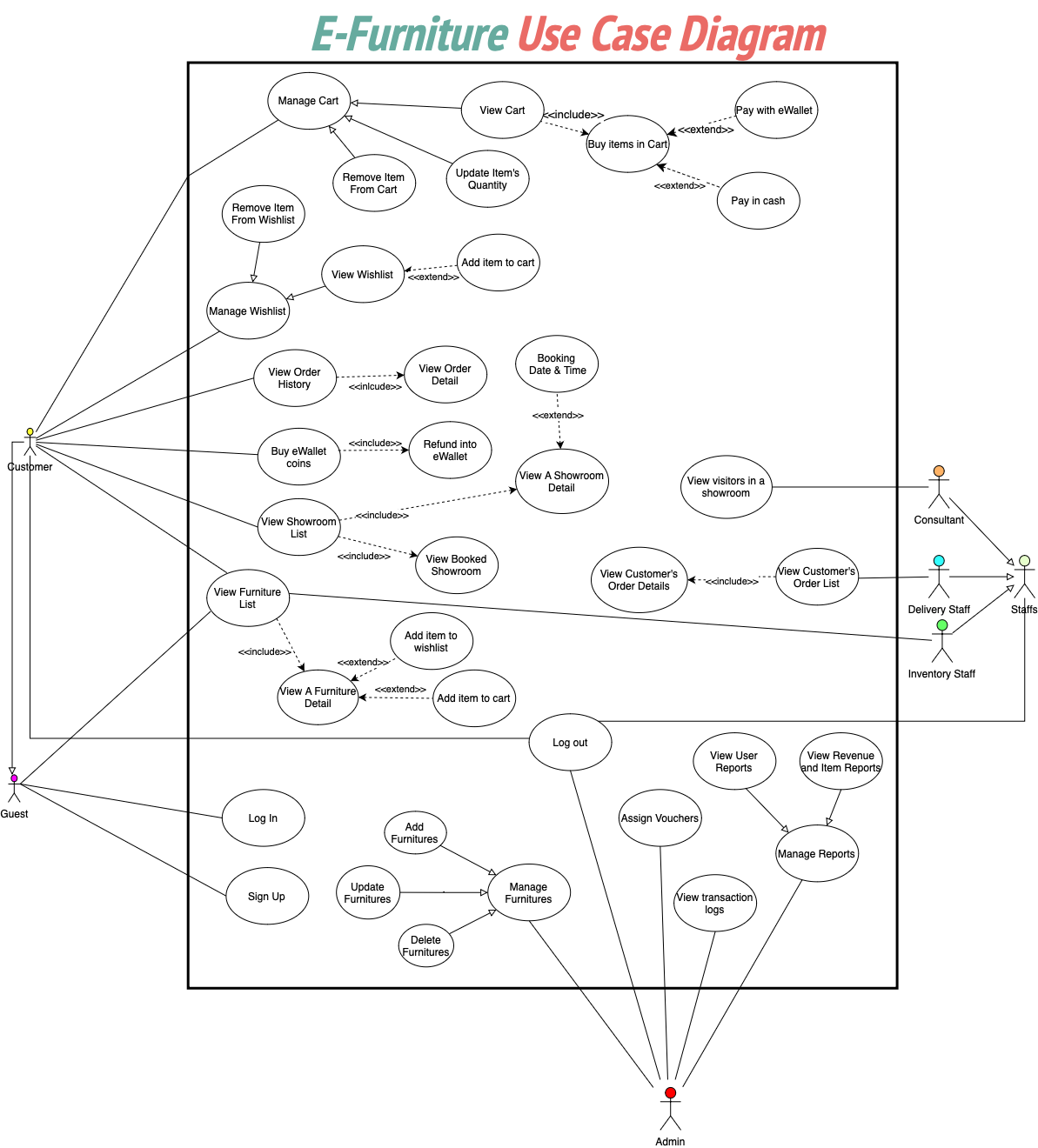
1. Constraints: technology constraints: Front-end will be developed by a single-page web application preferably by ReactJS. The back-end will be hosted using Azure - Microsoft and .Net framework to make restful API. The database will be hosted on Aiven.io using MySQL RDBMS.
2. Assumptions and Dependencies:

* Dependencies:
  + Technology: The deployment of virtual and augmented reality features requires availability and access to advanced programming capabilities and hardware requirements.
  + Product Range: The success of this project heavily relies on the range and quality of furniture items sourced for the showroom.
  + Supply Chain: The efficiency of furniture delivery is a critical factor. This further depends on collaboration with reliable logistic partners.
  + Website Maintenance: Regular updates and maintenance of the website to ensure seamless user experience.
  + Payment Partnerships: The e-commerce capabilities depend on partnerships with financial entities for secure transactions.
* Assumption:
  + Market Acceptance: The assumption that customers will find value in purchasing furniture online and will adopt the use of the digital platform.
  + Tech Compatibility: The majority of users have devices compatible with AR/VR technology.
  + Resource Availability: Adequate resources, including human skills and technological infrastructure, will be available throughout the project's timeline.
  + Vendor Reliability: The vendors supplying the furniture will be able to meet the standards of quality and punctuality.

1. Requirements Subsets: These could be specific requirements for subsets of your product, like different versions for distinct user segments, or varying requirements for different platforms.

# FUNCTIONAL REQUIREMENT

## Use Case Diagram



## Log In

| ***USE CASE 01 SPECIFICATION*** | | | | |
| --- | --- | --- | --- | --- |
| ***Use-case No.*** | UC001 | ***Use-case Version*** | | 1.0 |
| ***Use-case Name*** | Sign In | | | |
| ***Author*** | N.D.G.Bao | | | |
| ***Date*** | 13/01/2024 | ***Priority*** | High | |
| ***Actors:***  User, Admin, Staff.  ***Summary:***  This use case describes how users will enter a login form and input their email, and password to begin authorizing their accessibility into the system (customer, staff, admin).  ***Goal:***  Sign in to the website to perform their purposes.  ***Triggers:***  A log-in button on the website’s main page.  ***Preconditions:***  Must already register an account.  ***Post Conditions:***  Able to use key features like: manage e-wallet, booking showrooms, making purchases.  ***Main Success Scenario:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | * From the Main page navigation bar - user press “login” button. | System will redirect them to the login page.  Login Form will have:   * Textfield “Enter email” (Validation: must be a valid email address: @gmaill.com, …) * Textfield “Enter Password” (Validation: must be 8 characters or more such as: annie1141, carlsimpton, lolvipezgame) * “Submit” Button * “Login With Google” (Using Google Oauth2.0 method)   \*note: “password” field’s characters will be hidden (\*\*\*) with an icon to reveal it. | | 2 | * Enter Email and Password * Press “Submit” button | System will take the inputs and cross-reference them with what is in the database. then redirect you to the appropriate page for your role (customer, staff, admin). |   ***Alternative Scenario:***     | ***No.*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | 1 | 1. At the login form, users can press a button with “Login With Google”. | The Google account page will pop up for you to select the account you want. | | 2 | 1. The users will then choose which account to login. | * The system will process the request from Google Authentication. * Google responds with the token. * The Token is then being processed if the account is in the database. * Once complete, the system redirects you to the Home Page. |   ***Exceptions:***     | ***No.*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | 1. *Submit the wrong email or password.* | *A pop-up will appear warning that you have failed to login “(wrong password, account doesn’t exist)”.* | | *2* | 1. *The account is not in the database.* | *System popups a warning saying “ the account is invalid.”*  *A Line of text will appear below the text fields, written: “Don’t have an account? Register here”.* |   ***Relationships:***  None.  ***Business Rules:***  Staff and admin must use their company-provided email to log in to the system. | | | | |

## Log out

| ***USE CASE 02 LOG OUT*** | | | | |
| --- | --- | --- | --- | --- |
| ***Use-case No.*** | UC002 | ***Use-case Version*** | | 1.0 |
| ***Use-case Name*** | Log out | | | |
| ***Author*** | Chau Quoc Bao | | | |
| ***Date*** | 23/01/2024 | ***Priority*** | High | |
| ***Actor:***  User, staff, admin  ***Summary:***  This feature allows users to log out of their account.  ***Goal:***  The actor logs out of the account to back to the normal home page with a guest role.  ***Triggers:***  The actor wants to log out of their account.  ***Preconditions:***  Must be logged in to be able to log out.  ***Post Conditions:***  *The*Actor will lose their respective role’s privileges.  ***Main Success Scenario:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Press the ‘Log out’ button on the upper right website’s main page, on the taskbar. | System will pop up a confirmation board right below with 2 buttons:   * “Confirm” Button. * “Cancel” Button. | | 2 | Press ‘Confirm’. | System will log the user out of their account.  User will be redirected to the main page. | | 3 | Press ‘Cancel’ | System will close the pop up and remain logged in. |   ***Alternative Scenario:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Session Timeout. | System will automatic log out the user after a certain amount of inactivity time (30 minutes).  User will be redirected to the main page. And can proceed to log in again. |   ***Exceptions:***     | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | The user presses the ‘Cancel’ button | System will close the pop-up and the user remain logged in. |   ***Relationships:***  None.  ***Business Rules:***  None. | | | | |

## Sign Up

| ***USE CASE 03 SPECIFICATION*** | | | | |
| --- | --- | --- | --- | --- |
| ***Use-case No.*** | UC003 | ***Use-case Version*** | | 1.0 |
| ***Use-case Name*** | Sign Up | | | |
| ***Author*** | N.D.G.Bao | | | |
| ***Date*** | 13/01/2024 | ***Priority*** | High | |
| **Actor:**  Guest.  **Summary:**  This feature allows guest users to sign up for an account with their details to access some of the more advanced features (booking showroom, purchasing, seeking consultant, etc).  **Goal:**  Create an account as a customer to access more actions on the website.  **Triggers:**  When guest users click on a customer-specific action (booking, purchasing), a pop-up will appear asking them to sign in.  A button to sign up in the sign-in form.  **Preconditions:**  Must not already have an account.  **Post Conditions:**  User accounts will be stored in the database. The user can now log in to the website as customers.  **Main Success Scenario:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | From the sign-in form, the user will press the “sign-up” button to create a new account | System will redirect them to the sign-up page to input the necessary information.  The form includes:   * “Email” textfield. Validation (xxx.gmail.com, xxx.edu, etc) * “Password” textfield. Validation “password” field’s letters will be hidden (\*\*\*\*) and at least 8 characters * User “Full Name” textfield. Validation: only accept char and at least 8 chars * “Address” textfield. Validation: must not be empty. * “Phone number” textfield. Validation: number only - must be valid phone number (“+<country code>XXXXXXX”) * “Submit” button | | 2 | The user inputs personal information.  (Name, email, password, address, phone number).  Press “Submit”. | * An email will be sent to the user to confirm their registration. | | 3 | User open their e-mail and confirm the email. | * System will accept the confirmation and add the account to the database. * Creating a new user with the appropriate role. |   **Alternative Scenario:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | At the sign-in form, users can press a button with “Sign-up With Google” | The Google account page will pop up for you to select which account to choose. | | 2 | The users will then choose which account to sign up with. | 1. The system will process the request from Google Authentication 2. Save the user’s email with the password. 3. Users will still have to manually input the rest of the information:  * User “Full Name” textfield * “Address” textfield * “Phone number” textfield | | 3 | The user fills in the rest of the information and presses “submit” | * An email will be sent to the user to confirm their registration. | | 4 | User open their e-mail and confirm the email. | * System will take the inputs and add them to the database, creating a new user. * User then will be redirected back to the login page. |   **Exceptions:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Input fields left empty. | The system pops a warning that the field must not be empty: “Please fill in all the required information”. | | 2 | If the email is already in the database. | System pops a warning that the field must not be empty: “Account already exist”. | | 3 | Password doesn’t match the requirements. | The system pops a warning that the field must not be empty: “Password must fit the requirements: [requirements]”. |   **Relationships:**  Login.  **Business Rules:**  None. | | | | |

## View Furniture List

## 

| ***USE CASE 04 VIEW FURNITURE LIST*** | | | | |
| --- | --- | --- | --- | --- |
| ***Use-case No.*** | UC004 | ***Use-case Version*** | | 1.0 |
| ***Use-case Name*** | View Furniture List | | | |
| ***Author*** | Nguyễn Hùng Khương | | | |
| ***Date*** | 13/01/2024 | ***Priority*** | High | |
| ***Actors:***  Staff, User, Guest, Admin  ***Summary:***  On the “Home page”, there will be a “View furniture list” button for the customer to press. When pressed, users will be redirected to the “View furniture list” page. Where users see all the furniture products.  ***Goal:***  See the list of all furniture products.  ***Triggers:***  A “View furniture list” button on the “Home page”.  ***Preconditions:***  *None.*  ***Post Conditions:***  *None.*  ***Main Success Scenario:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Press the “View furniture list” button on the “Home page” | The System will redirect them to the “View furniture list page”.  View furniture list page will have:   * Search bar “Enter the name of furniture product” * “Search” button. * “Detail” button. | | 2 | Press the “Detail” button | System will be redirected to the “View furniture list detail” page. |   ***Alternative Scenario:***     | ***No.*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | 1 | Click on the image of the furniture product | System will be redirected to the “View furniture list detail” page. |   ***Exceptions:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | If the user entering the name of furniture and press the “Search” button | The system returns products that match the showroom customer searching. |   ***Relationships:***  View furniture detail page  ***Business Rules:***  None | | | | |

## View Cart

| ***USE CASE 05 VIEW CART*** | | | | |
| --- | --- | --- | --- | --- |
| ***Use-case No.*** | UC005 | ***Use-case Version*** | | 1.0 |
| ***Use-case Name*** | View Cart | | | |
| ***Author*** | N.D.G.Bao | | | |
| ***Date*** | 13/01/2024 | ***Priority*** | Normal | |
| ***Actor:***  User.  ***Summary:***  On this feature, user can view a list of items they want to buy in the cart, the total price, the quantity of each item.  ***Goal:***  View a list of items they want to purchase.  ***Triggers:***  A “add to cart” button on a specific furniture page.  ***Preconditions:***  The user must log in to perform this action.  ***Post Conditions:***  None.  ***Main Success Scenario:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Customer goes to “Cart” Page. A dashboard of a list of items user want to buy. | And show the appropriate page with the correct furniture.  The Cart page has a dashboard with each item display with the details:   * Item’s name * Price per item * Quantity * Total: (Quantity x price)   The page includes:   * Edit quantity button (Validation: require the quantity must not below 1 and must not larger than the maximum quantity instock - integer number) on each item grid. * Remove from cart (X icon or Trashcan) on each item grid. * Purchase button (which will appear with a “cart” icon) at the end of the page. * Delivery address (From user’s infos). |   ***Alternative Scenario:***  None.  ***Exceptions:***  None.  ***Relationships:***  View furniture details, add to cart (from wishlist), buy items in cart.  ***Business Rules:***  None. | | | | |

## 

## Update Item’s Quantity

| ***USE CASE 06 UPDATE ITEM’S QUANTITY*** | | | | |
| --- | --- | --- | --- | --- |
| ***Use-case No.*** | UC006 | ***Use-case Version*** | | 1.0 |
| ***Use-case Name*** | Update Item’s Quantity | | | |
| ***Author*** | N.D.G.Bao | | | |
| ***Date*** | 13/01/2024 | ***Priority*** | Normal | |
| ***Actor:***  User.  ***Summary:***  On this use case, user can adjust the quantity of an item they want to purchase.  ***Goal:***  Change an item’s quantity they want to purchase.  ***Triggers:***  A number textfield.  ***Preconditions:***  The user must log in to perform this action.  ***Post Conditions:***  None.  ***Main Success Scenario:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | On the dashboard of the Cart page.  A number textfield (with two arrow keys to increase or decrease) to update an item’s quantity | * Item’s quantity will be update sequecially (+1 or -1) accordingly. * The total price will also update. * Validation: Quantity will be fixed from 1 to max quantity of said item. |   ***Alternative Scenario:***  None.  ***Exceptions:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | A rare instance occurs when the items in the cart are out-of-stock (by another buyer’s purchases). | * The system will disabled the item in the wishlist. * Disable the number texfield. * Set the number to 0. |   ***Relationships:***  View Cart.  ***Business Rules:***  None. | | | | |

## 

## Remove Item From Cart

| ***USE CASE 07 REMOVE ITEM FROM CART*** | | | | |
| --- | --- | --- | --- | --- |
| ***Use-case No.*** | UC007 | ***Use-case Version*** | | 1.0 |
| ***Use-case Name*** | Remove Item From Cart | | | |
| ***Author*** | N.D.G.Bao | | | |
| ***Date*** | 13/02/2024 | ***Priority*** | Normal | |
| ***Actor:***  User.  ***Summary:***  On this use case, user can remove the item from the cart if they don’t want to buy it anymore.  ***Goal:***  Remove the item from the cart.  ***Triggers:***  User press a Button “Delete”.  ***Preconditions:***  The user must log in to perform this action.  ***Post Conditions:***  Item user selected will be remove from the cart.  ***Main Success Scenario:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | On the dashboard of the Cart page.  User can press the “Delete” button with “Trash Can” icon to remove the item they don’t want to buy. | * System will remove the item from the cart. * The total price update accordingly. |   ***Alternative Scenario:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | If the admin remove an item from the system. | If such item is in the customer cart, the item will be automatically remove from the cart. |   ***Exceptions:***  None.  ***Relationships:***  View Cart.  ***Business Rules:***  None. | | | | |

## 

## Buy Items In Cart

| ***USE CASE 08 BUY ITEMS IN CART*** | | | | |
| --- | --- | --- | --- | --- |
| ***Use-case No.*** | UC008 | ***Use-case Version*** | | 1.0 |
| ***Use-case Name*** | Buy Items In Cart | | | |
| ***Author*** | N.D.G.Bao | | | |
| ***Date*** | 13/01/2024 | ***Priority*** | Normal | |
| ***Actor:***  User.  ***Summary:***  This page is for the customer to finalize their purchase.  ***Goal:***  Input their delivery address, choosing payment method (by cash or e-wallet).  ***Triggers:***  This page pop-up when user press “purchase” button from the previous page.  ***Preconditions:***  The user must log in to perform this action.  ***Post Conditions:***  An order is created with the provided information.  ***Main Success Scenario:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Customer input additiona transanctional information.  Input their address  Choose a payment method | The page has a form which display:   * User’ address textfield (pre written with the address user provide when they sign up) * Total price * Payment methods options:   + “By E-coin”   + “By Cash” | | 4 | User choose either “E-coin” or “Cash” | The system will take the inputs. And create an order. |   ***Alternative Scenario:***  *None.*  ***Exceptions:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | A rare instance occurs when the items in the cart are out-of-stock (by another buyer’s purchases), but you can still can press the “purchase” button | The system will then send an error “Item is out of stock. But you can wishlist it for further purchase”. And remove the item from the cart. | | 2 | User press “confirm” but their e-coin is not enough to purchase the items. | The system will then send an error “Not enough e-coin” and will redirect the user back to reselect their payment option. |   ***Relationships:***  View, manage cart.  ***Business Rules:***  None. | | | | |

## 

## Pay By E-coin

| ***USE CASE 09 PAY BY E-COIN*** | | | | |
| --- | --- | --- | --- | --- |
| ***Use-case No.*** | ***UC009*** | ***Use-case Version*** | | ***1.0*** |
| ***Use-case Name*** | ***Pay By E-coin*** | | | |
| ***Author*** | ***N.D.G.Bao*** | | | |
| ***Date*** | ***16/01/2024*** | ***Priority*** | ***Normal*** | |
| ***Actor:***  User.  ***Summary:***  From the” buy items” page, user will confirm that they want to finalize the purchase using their e-coin from the system.  ***Goal:***  Confirm the purchase using company e-coin.  ***Triggers:***  A “Confirm” button.  ***Preconditions:***  The user selects “buy with e-coin” button.  ***Post Conditions:***  None.  ***Main Success Scenario:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | User confirms they want to buy the item with company e-coin by pressing the “By E-coin” option. | * The system will redirect them accordingly to their payment method. * An order will be created and stored in the database. * A “highlight” will print the “Pay with e-coin” on the customer order. Delivery Staff won’t ask for payment when deliver. |   ***Alternative Scenario:***  ***None.***  ***Exceptions:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | User press “confirm” but their e-coin is not enough to purchase the items. | * The system will then send an error popup: “Not enough e-coin”. * Then will redirect the user back to the buy items page to reselect their payment option. |   ***Relationships:***  ***Buy items in the cart.***  ***Business Rules:*** | | | | |

## Pay In Cash

| ***USE CASE 10 PAY IN CASH*** | | | | |
| --- | --- | --- | --- | --- |
| ***Use-case No.*** | ***UC010*** | ***Use-case Version*** | | ***1.0*** |
| ***Use-case Name*** | Pay In Cash | | | |
| ***Author*** | N.D.G.Bao | | | |
| ***Date*** | 16/01/2024 | ***Priority*** | Normal | |
| ***Actor:***  User.  ***Summary:***  From the” buy items” page, user will confirm that they want to finalize the purchase in cash when the item is delivered to them.  ***Goal:***  Confirm the purchase using their physical currency with the delivery staff.  ***Triggers:***  A “Confirm” button.  ***Preconditions:***  The user selects the “paid with cash” button.  ***Post Conditions:***  A highlight “Pay with cash” will appear on the order for delivery staff to see.  ***Main Success Scenario:***   | ***No.*** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | * The user confirms they want to buy the item with cash when the delivery comes. * They select the “Confirm” button and finalize the purchase. | * The system will redirect them accordingly to their payment method * An order will be created and stored in the company database. * A “Highlight” will be printed on the order for the staffs to know and transact. |   ***Alternative Scenario:***  None.  ***Exceptions:***  None.  ***Relationships:***  Buy items in the cart, and view user order.  ***Business Rules:*** | | | | |

## View Wishlist

| ***USE CASE 11 VIEW WISHLIST*** | | | | |
| --- | --- | --- | --- | --- |
| ***Use-case No.*** | UC0011 | ***Use-case Version*** | | 1.0 |
| ***Use-case Name*** | View Wishlist | | | |
| ***Author*** | Vu Duc Anh | | | |
| ***Date*** | 13/01/2024 | ***Priority*** | Normal | |
| ***Actor:***  Customer.  ***Summary:***  This use case allows actor to view their wishlist.  ***Goal:***  The wishlist appears with all of the wish furniture on it.  ***Triggers:***  The actor wants to see their wishlist and clicks on the “Wishlist” button on the navigator.  ***Preconditions:***  The user must log in to perform this action.  ***Post Conditions:***  None.  ***Main Success Scenario:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | From the navigator bar, the user presses the button “Wishlist”. | The system will then redirect to the Wishlist page.  In that page contain the table of furniture which show up as a “card” with:   * Furniture name * Price * A “In-stock”/”Out-of-stock” tag * A “Remove” button to remove it from wishlist * A “Add to cart” button. |   ***Alternative Scenario:*** None  ***Exceptions:*** None  ***Relationships:***  Manage wishlist***.***  ***Business Rules:***  None. | | | | |

## Add Item To Wishlist

| ***USE CASE 12 ADD ITEM TO WISHLIST*** | | | | |
| --- | --- | --- | --- | --- |
| ***Use-case No.*** | UC0012 | ***Use-case Version*** | | 1.0 |
| ***Use-case Name*** | Add Item To Wishlist | | | |
| ***Author*** | Chau Quoc Bao | | | |
| ***Date*** | 12/03/2024 | ***Priority*** | Normal | |
| ***Actor:***  Customers.  ***Summary:***  This feature allows actors to add an item to their wishlist.  ***Goal:***  The selected item will move to the wishlist.  ***Triggers:***  The actor wants to add the item to the wishlist.  ***Preconditions:***  The user must log in to perform this action.  ***Post Conditions:***  None.  ***Main Success Scenario:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | From furniture detail, the user presses the “Add to wishlist” button. | The system will notice a pop up right where the user clicked “Add success”. |   ***Alternative Scenario:*** None  ***Exceptions:*** None  ***Relationships:***  View A Furniture Detail  View Wishlist  ***Business Rules:***  None. | | | | |

## 

## Add Item To Cart

| ***USE CASE 13 ADD ITEM TO CART*** | | | | |
| --- | --- | --- | --- | --- |
| ***Use Case No.*** | UC0013 | ***Use-case Version*** | | 1.0 |
| ***Use-case Name*** | Add Item To Cart | | | |
| ***Author*** | Vu Duc Anh | | | |
| ***Date*** | 13/01/2024 | ***Priority*** | Normal | |
| ***Actor:***  Customer.  ***Summary:***  This use case allows actor to add their furniture from wishlist to cart.  ***Goal:***  The selected furniture from the wishlist will be moved to the cart for further action.  ***Triggers:***  The actor wants to add their furniture from the wishlist to the cart.  ***Preconditions:***  The user must log in to perform this action.  ***Post Conditions:***  None.  ***Main Success Scenario:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | From the navigator, the user presses the button “Wishlist”. | The system will then redirect to the Wishlist page.  In that page contain the table of furniture which show up as a “card” with:   * Furniture name * Price * A “In-stock”/”Out-of-stock” tag * A “Remove” button to remove it from wishlist * A “Add to cart” button. | | 2 | User select their furniture they want to add to the cart, then click the “Add to cart” button. | The system will add that furniture data into that customer cart data.  The system will redirect to the Cart page with that furniture appearing on it.  Also, remove that furniture from the wishlist. |   ***Alternative Scenario:*** None  ***Exceptions:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Actor try to press the “Add to cart” button in the furniture that has an “Out-of-stock” tag. | The pop-up “Unable to add this to the cart since it's out of stock” appears. |   ***Relationships:***  View Wishlist***.***  ***Business Rules:***  None. | | | | |

## Remove Item From Wishlist

| ***USE CASE 14 REMOVE ITEM FROM WISHLIST*** | | | | |
| --- | --- | --- | --- | --- |
| ***Use-case No.*** | UC0014 | ***Use-case Version*** | | 1.0 |
| ***Use-case Name*** | Remove Item From Wishlist | | | |
| ***Author*** | Vu Duc Anh | | | |
| ***Date*** | 13/01/2024 | ***Priority*** | Normal | |
| ***Actor:***  Customer.  ***Summary:***  This use case allows actor to remove their furniture from the wishlist.  ***Goal:***  The selected furniture will be removed from the wishlist.  ***Triggers:***  The actor wants to remove their furniture from the wishlist.  ***Preconditions:***  The user must log in to perform this action.  ***Post Conditions:***  None.  ***Main Success Scenario:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | From the navigator, the user presses the button “Wishlist”. | The system will then redirect to the Wishlist page.  In that page contain the table of furniture which show up as a “card” with:   * Furniture name * Price * A “In-stock”/”Out-of-stock” tag * A “Remove” button to remove it from wishlist * A “Add to cart” button. | | 2 | User select their furniture they want to add to the cart, then click the “Remove” button. | The system will remove that furniture from the wishlist data.  The system will redirect to the Wishlist page with that furniture removed on it. |   ***Alternative Scenario:*** None  ***Exceptions:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | User clicks the “Removed” button at the non-exist furniture. | The pop-up “This product does not exist” appears. |   ***Relationships:***  Manage Wishlist***.***  ***Business Rules:***  None. | | | | |

## View Customer’s Orders

| ***USE CASE 15 VIEW CUSTOMER’S ORDERS*** | | | | |
| --- | --- | --- | --- | --- |
| ***Use-case No.*** | UC015 | ***Use-case Version*** | | 1.0 |
| ***Use-case Name*** | View Customer’s Orders | | | |
| ***Author*** | N.D.G.Bao | | | |
| ***Date*** | 21/01/2024 | ***Priority*** | High | |
| ***Actor:***  Delivery Staff.  ***Summary:***  When the delivery staff is ready to begin their trip. They can view the orders made by the user. This includes their address, phone number, and the payment method (in case they use cash as the payment option).  ***Goal:***  Enable the delivery staff to do their job by providing the relevant information  ***Triggers:***  None.  ***Preconditions:***  Only Delivery Staff can see this page.  ***Post Conditions:***  None.  ***Main Success Scenario:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | The delivery Staff log in to their company account and see a list of orders. | Each order in the list will contain these information:   * Phone number. * Address * Payment method (E-wallet or cash). * “Delivery Success” button. * “Delivery Failed” button   \* Note: the case of cash, there will be a “Highlight” to indicate the user will pay for the furniture. | | 2 | After reaching the customer (and after the customer paid for the furniture).  Delivery Staff will press the button “Delivery Success”. | The system will process and return the “Success” status to the customer order history.  System will save the order and send it to the admin. |   ***Alternative Scenario:***  None.  ***Exceptions:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | In case the user didn’t receive the order (either they canceled it or couldn’t receive it in any case). The delivery staff will press the button “Delivery Failed”. | * The system will process and return the Failure to the customer order history. * A form will pop up for the staff to note in why the failure.The form contain  1. Textfield “Reason Of Failure” 2. “Submit Button” | | 2 | Staff Write in the reason. Press ‘’Submit” | System will save the report and send it to the admin. |   ***Relationships:***  Log in  ***Business Rules:*** | | | | |

## View A Furniture Detail

| ***USE CASE 16 VIEW FURNITURE DETAIL*** | | | | |
| --- | --- | --- | --- | --- |
| ***Use-case No.*** | UC16 | ***Use-case Version*** | | 1.0 |
| ***Use-case Name*** | View A Furniture Detail | | | |
| ***Author*** | Châu Quốc Bảo | | | |
| ***Date*** | 05/03/2024 | ***Priority*** | High | |
| ***Actor:***  Staff, User, Guest, Admin  ***Summary:***  This feature allows to see all details of a furniture and. You also can add to cart or wishlist by the “Add to cart” button or “Add wishlist” button.  ***Goal:***  The actor can view a furniture detail such as material, dimensions,etc.. and can add them to ‘Cart’ or ‘Wishlist’  ***Triggers:***  The actor wants to view a furniture detail and can add them to ‘Cart’ or ‘Wishlist’.  ***Preconditions:***  None.  ***Post Conditions:***  None.  ***Main Success Scenario:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | On the furniture page | System will show all details along the two halves of the screen:  The left side:   * Picture   The right side:   * Name * Category * Material * Dimensions * Package Dimensions * Package Method * “Add to cart” button * “Add wishlist” button |   ***Alternative Scenario:***  None  ***Exceptions:***     | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | If furniture out of stock, you can not “Add to cart” | System will notice that “Out of stock” when  you pressed “Add to cart” |   ***Relationships:***  *View Furniture List*  *Add To Cart*  *Add To Wishlist*  ***Business Rules:***  *None.* | | | | |

## View A Showroom Detail

| ***USE CASE 17 VIEW A SHOWROOM DETAIL*** | | | | |
| --- | --- | --- | --- | --- |
| ***Use-case No.*** | UC017 | ***Use-case Version*** | | 1.0 |
| ***Use-case Name*** | View A Showroom Detail | | | |
| ***Author*** | Nguyễn Hùng Khương | | | |
| ***Date*** | 16/01/2024 | ***Priority*** | High | |
| **Actor:**  Consultant, User  **Summary:**  This feature allows the actor to view all the information of a showroom.  **Goal:**  The actor can view all the information about a showroom like time open and address.  **Triggers:**  On the “View showroom list” page, there will be a “View Detail” button on the navigation bar for the customer to press. When pressed, we can go to the View A Showroom Detail page.  **Preconditions:**  **None.**  **Post Conditions:**  **None.**  **Main Success Scenario:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | The User can see the showroom address and its schedule. | The system will show all details such as   * Address of the showroom * Name of the showroom * Scheduled time open and close of the showroom * And a Button “Book” to redirect the user to the booking form. |   **Alternative Scenario:**  None.  **Exceptions:**  None.  **Relationships:**  View Showroom List, Booking Date & Time  **Business Rules:**  None. | | | | |

## Add Furniture

| **USE CASE 18 ADD FURNITURE** | | | | |
| --- | --- | --- | --- | --- |
| **Use-case No.** | UC018 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Add Furniture | | | |
| **Author** | Vu Duc Anh | | | |
| **Date** | 16/01/2024 | **Priority** | High | |
| **Actor:**  Admin  **Summary:**  This use case allows the admin to manage the furniture available on the website by adding new furniture to the furniture list.  **Goal:**  New furniture is added to the furniture list.  **Triggers**  Actor wants to add new furniture to their furniture list.  Actor clicks on the “Add Furniture” at Manage Furniture page.  **Preconditions:**  Actor must log in to the system as “Admin” role.  Actor must have their Internet connected.  **Post Conditions:**  The system will add data of new furniture into the furniture database.  The furniture list will appear with new furniture added to it.  **Main Success Scenario:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | The actor clicks the “Manage Furniture” button. | The system will redirect to the Manage Furniture page which contains:   * A table of all furniture, each row contain some information: * Name of furniture (string type, no limit) * Type (string type, no limit) * Material (string type, no limit) * Dimensions (array(int), no limit) * Package Method (string type, no limit) * An “Update” button * A “Delete” button * At the right corner of table there is a “New” button | | 2 | Actor clicks on the “New” button | The system will redirect to the Add New Furniture page. The form is appeared which contain this information with the required form:   * Name of furniture (string type, no limit) * Type (string type, no limit) * Material (string type, no limit) * Dimensions (array(int), no limit) * Package Method (string type, no limit) * A “Create” button. * A “Cancel” button.   All of those must be in the empty and required textbox. | | 3 | After the actor fills in the information, click the “Create” button. | The system will validate these data as the type given above. Once it is valid, add these data to the database and return to the Manage Furniture page with new furniture appearing in the list. |   **Alternative Scenario:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | After logging into the “Admin” role, the actor clicks the “Manage Furniture” button. | The system will redirect to the Manage Furniture page which contains:   * A table of all furniture, each row contain some information: * Name of furniture (string type, no limit) * Type (string type, no limit) * Material (string type, no limit) * Dimensions (array(int), no limit) * Package Method (string type, no limit) * An “Update” button * A “Delete” button * At the right corner of table there is a “New” button | | 2 | Actor clicks on the “New” button | The system will redirect to the Add New Furniture page. The form is appeared which contain this information with the required form:   * Name of furniture (string type, no limit) * Type (string type, no limit) * Material (string type, no limit) * Dimensions (array(int), no limit) * Package Method (string type, no limit) * A “Create” button. * A “Cancel” button.   All of those must be in the empty and required textbox. | | 3 | Actor clicks on the “Import File (.xlsx or .json)” button | The system will open the OS window to choose the file. After the actor chooses the file, the system will process the file and fill up the data into the form above. | | 4 | After filling the information, the actor clicks the “Create” button. | The system will validate these data as the type given above. Once it is valid, add these data to the database and return to the Manage Furniture page with new furniture appearing in the list. |   **Exceptions:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Actor clicks the “Create” button without filling in any information. | The pop-up message “Please fill out all missing information” appears. | | 2 | Actor choosing file that the system cannot process (ex: docx, pdf, etc.) | The pop-up message “Wrong type of file, please try again” appears. | | 3 | Actor clicks “Cancel” button | The system will return to the Manage Furniture page where no new furniture appears. | | 4 | Actor fill the “Dimensions” field with other characters other than numbers, then click the “Create” button. | The pop-up message “Numbers are required in this field” appears next to the “Dimensions” field. |   **Relationships:** Manage Furniture  **Business Rules:** None | | | | |

## Update Furniture

| **USE CASE 19 UPDATE FURNITURE** | | | | |
| --- | --- | --- | --- | --- |
| **Use-case No.** | UC019 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Update Furniture | | | |
| **Author** | Vu Duc Anh | | | |
| **Date** | 17/01/2024 | **Priority** | High | |
| **Actor:**  Admin  **Summary:**  This use case allows the admin to manage the furniture available on the website by updating the furniture present on the list.  **Goal:**  The furniture’s details are kept up to date with reality  **Triggers**  Actor clicks on the “Update” at Manage Furniture page.  **Preconditions:**  Actor must be logged in to the system as “Admin” role.  **Post Conditions:**  The furniture’s details will have changed to reflect the new information provided by the admin.  The furniture list will appear with updated furniture.  **Main Success Scenario:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | After logging into the “Admin” role, the actor clicks the “Manage Furniture” button. | The system will redirect to the Manage Furniture page which contains:   * A table of all furniture, each row contain some information: * Name of furniture (string type, no limit) * Type (string type, no limit) * Material (string type, no limit) * Dimensions (array(int), no limit) * Package Method (string type, no limit) * An “Update” button * A “Delete” button * At the right corner of table there is a “New” button | | 2 | Actor clicks on the furniture they want to update, then click on the “Update” button | The system will redirect to the Update Furniture page. The form is appeared which contain this information:   * Name of furniture (string type, no limit) * Type (string type, no limit) * Material (string type, no limit) * Dimensions (array(int), no limit) * Package Method (string type, no limit) * A “Update” button. * A “Cancel” button.   All of those must be in the textbox which contains all the current information of the furniture. | | 3 | After the actor fills in the information, click the “Update” button. | The system will validate these data as the type given above. Once it is valid, update these data to the database and return to the Manage Furniture page with updated furniture appearing in the list. |   **Alternative Scenario:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | The actor clicks the “Manage Furniture” button. | The system will redirect to the Manage Furniture page which contains:   * A table of all furniture, each row contain some information: * Name of furniture (string type, no limit) * Type (string type, no limit) * Material (string type, no limit) * Dimensions (array(int), no limit) * Package Method (string type, no limit) * An “Update” button * A “Delete” button * At the right corner of table there is a “New” button | | 2 | Actor clicks on the furniture they want to update, then click on the “Update” button | The system will redirect to the Update Furniture page. The form is appeared which contain this information:   * Name of furniture (string type, no limit) * Type (string type, no limit) * Material (string type, no limit) * Dimensions (array(int), no limit) * Package Method (string type, no limit) * A “Update” button. * A “Cancel” button.   All of those must be in the textbox which contains all the current information of the furniture. | | 3 | Actor clicks on the “Import File (.xlsx or .json)” button | The system will open the OS window to choose the file. After the actor chooses the file, the system will process the file and fill up the changed data into the form above (if any). | | 4 | After filling the information, the actor clicks the “Update” button. | The system will validate these data as the type given above. Once it is valid, update these data to the database and return to the Manage Furniture page with new furniture appearing in the list. |   **Exceptions:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Actor clicks the “Create” button without filling in any information. | The pop-up message “Please fill out all missing information” appears. | | 2 | Actor choosing file that the system cannot process (ex: docx, pdf, etc.) | The pop-up message “Wrong type of file, please try again” appears. | | 3 | Actor clicks “Cancel” button | The system will return to the Manage Furniture page where no new furniture appears. | | 4 | Actor fill the “Dimensions” field with other characters other than numbers, then click the “Update” button. | The pop-up message “Numbers are required in this field” appears next to the “Dimensions” field. |   **Relationships:** Manage Furniture  **Business Rules:** None | | | | |

## Delete Furniture

| **USE CASE 20 DELETE FURNITURE** | | | | |
| --- | --- | --- | --- | --- |
| **Use-case No.** | UC020 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Delete Furniture | | | |
| **Author** | Vu Duc Anh | | | |
| **Date** | 17/01/2024 | **Priority** | High | |
| **Actor:** Admin  **Summary:** This use case allows the adminto manage their furniture by removing furniture from the list.  **Goal:** Remove no longer available furniture from the list.  **Triggers:** The actor wants to delete their existing furniture from the manage furniture list and clicks on the “Delete” button at Manage Furniture page.  **Preconditions:**  Actor must be logged in to the system as “Admin” role.  Actor must have their Internet connected.  **Post Conditions:**  The furniture will be deleted from the database.  The manage furniture list will appear with updated furniture.  **Main Success Scenario:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | The actor clicks the “Manage Furniture” button at navigation. | The system will redirect to the Manage Furniture page which contains:   * A table of all furniture, each row contain some information: * Name of furniture (string type, no limit) * Type (string type, no limit) * Material (string type, no limit) * Dimensions (array(int), no limit) * Package Method (string type, no limit) * An “Update” button * A “Delete” button * At the right corner of table there is a “New” button | | 2 | Actor clicks on the furniture they want to update, then clicks on the “Delete” button. | The system will delete those data which are in the selected one from the database and return to the Manage Furniture page with an updated list. |   **Alternative Scenario:** None  **Exceptions:** None  **Relationships:** Manage Furniture  **Business Rules:** None | | | | |

## View Transactions Log

| **USE CASE 21 VIEW TRANSACTIONS LOG** | | | | |
| --- | --- | --- | --- | --- |
| **Use-case No.** | UC021 | **Use-case Version** | | 1.0 |
| **Use-case Name** | View Transactions Log | | | |
| **Author** | Vu Duc Anh | | | |
| **Date** | 17/01/2024 | **Priority** | High | |
| **Actor:**  Admin  **Summary:**  This use case allows the actor to view the transactions log in one or all orders.  **Goal:**  Actor can see transaction logs in one or all orders.  **Triggers**  The actor clicks the “Manage Transactions” or “View Transactions” button.  **Preconditions:**  Actor must be logged in to the system as “Admin” role.  Actor is on the “View Transactions” page or “Manage Transactions” page.  **Post Conditions:**  None  **Main Success Scenario:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | The actor clicks the “Manage Order” button. | The system will redirect to the Manage Order page which contains:   * A table of all orders, each row contain these information: * Order ID (string, no limit) * Order Name (string, no limit) * Quantity (int) * Total Price (real) * A “View Transactions” button | | 2 | Actor clicks on the order they want to see, then click on the “View Transactions” button | The system will load the “Transactions logs” page and point to that order base to its order ID. |   **Alternative Scenario:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | The actor clicks the “Manage Transactions” button at navigation. | The system will load the “Transactions logs” page to view all transaction logs of all orders. |   **Exceptions:** None  **Relationships:** None  **Business Rules:**   * The logs must be secured and private. | | | | |

## Assign Voucher

| **USE CASE 22 ASSIGN VOUCHER** | | | | |
| --- | --- | --- | --- | --- |
| **Use-case No.** | UC022 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Assign Vouchers | | | |
| **Author** | Vu Duc Anh | | | |
| **Date** | 30/01/2024 | **Priority** | Medium | |
| **Actor:**  Admin  **Summary:**  This use case allows the actor to manage their vouchers by adding a new voucher to the voucher list.  **Goal:**  A new voucher is added to the voucher list.  **Triggers**  The actor clicks on the “New” button in the Manage Voucher list.  **Preconditions:**  The actor must be logged in to the system as an “Admin” role.  **Post Conditions:**  The system will add data of new vouchers into the voucher database.  **Main Success Scenario:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | The actor clicks the “Manage Voucher” button. | The system will redirect to the Manage Voucher page which contains:   * A table of all vouchers, which contain these information: * Name of voucher * Date applied * Date expired * Furniture ID and the “All Products” checkbox * Percentage | | 2 | Actor clicks on the “New” button | The system will redirect to the Assign New Voucher page. The form is appeared which contain this information:   * Name of voucher * Date applied. * Date expired. * Furniture ID and the “All Products” checkbox * Percentage (which already have “%” symbol) * A “Create” button. * A “Cancel” button.   All of those must be in the empty textbox. | | 3 | After the actor fills in the information, click the “Create” button. | The system will add these data to the database and return to the Manage Voucher page with a new voucher appearing in the list. |   **Alternative Scenario:**   | **No**. | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | The actor clicks the “Manage Voucher” button. | The system will redirect to the Manage Voucher page which contains:   * A table of all vouchers, which contain these information: * Name of voucher * Date applied * Date expired * Furniture ID and the “All Products” checkbox * Percentage | | 2 | Actor clicks on the “New” button | The system will redirect to the Assign New Voucher page. The form is appeared which contain this information:   * Name of voucher * Date applied. * Date expired. * Furniture ID and the “All Products” checkbox * Percentage (which already have the “%” symbol) * A “Create” button. * A “Cancel” button.   All of those must be in the empty and required textbox. | | 3 | Actor clicks on the “Auto Assign” button | The system will load the “Auto Assign” dead pop-up containing the calendar, which lets the user choose the date that assigns this voucher. | | 4 | Actor after choosing date, click “OK” | The system returns to the main form of the Assign New Voucher page. | | 5 | After filling the information, the actor clicks the “Create” button. | The system will add these data to the database and return to the Manage Voucher page with a new voucher appearing in the list. |   **Exceptions:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Actor clicks the “Create” button without filling in any information. | The pop-up message “Please fill out all missing information” appears. | | 2 | Actor clicks the “Cancel” button at the Assign New Voucher page. | The system will return to the Manage Voucher page and no new voucher appears at that page. | | 3 | Actor clicks the “Cancel” button at the “Auto Assign” dead pop-up. | The system will return to the Assign New Voucher page which is not changing. |   **Relationships:** Manage Voucher  **Business Rules:**   * When auto-assign the voucher, the system lets the user choose multiple days. * The sale percentage must not exceed 99%. | | | | |

## View User Reports

| **USE CASE 23 VIEW USER REPORTS** | | | | |
| --- | --- | --- | --- | --- |
| **Use-case No.** | UC023 | **Use-case Version** | | 1.0 |
| **Use-case Name** | View User Reports | | | |
| **Author** | Vu Duc Anh | | | |
| **Date** | 02/02/2024 | **Priority** | Medium | |
| **Actor:**  Admin  **Summary:**  This use case allows the actor to manage their reports by viewing user reports.  **Goal:**  The actor can see user reports in detail such as blacklist, user revenue list, etc.  **Triggers**  The actor wants to see user reports to manage users on their system.  **Preconditions:**  The actor must log in to the system an “Admin” role.  The actor must have their Internet connected.  **Post Conditions:**  The system will load the User Reports dashboard and reveal all the components of that report.  **Main Success Scenario:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | The actor clicks the “Manage Reports” button. | The system will redirect to the Manage Reports page which contains two dashboards: User Report and Revenue And Item Report. | | 2 | Actor clicks on the “User Report” button | The system will redirect to the User Report dashboard. The dashboard appears which contains this information:   * User’s orders * Blacklist * User Revenue * Graphs (eWallet, User Revenue) * User eWallet information * A “Export to PDF” button |   **Alternative Scenario:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | If an actor is browsing at any dashboard (including the User Report dashboard), click on the “User Report” button | The system will redirect to the User Report dashboard. The dashboard appears which contains this information:   * User’s orders * Blacklist * User Revenue * Graphs (eWallet, User Revenue) * User eWallet information * A “Export to PDF” button |   **Exceptions:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | If the actor is selecting an empty dashboard | The system will redirect to that dashboard and show a “No information” message. |   **Relationships:** Manage Reports  **Business Rules:**   * The profile must be verified every 6 months. * The profile must be secured and encoded. | | | | |

## View Showroom List

| ***USE CASE 24 VIEW SHOWROOM LIST*** | | | | |
| --- | --- | --- | --- | --- |
| ***Use-case No.*** | UC024 | ***Use-case Version*** | | 1.0 |
| ***Use-case Name*** | View Showroom List | | | |
| ***Author*** | Nguyễn Hùng Khương | | | |
| ***Date*** | 13/01/2024 | ***Priority*** | High | |
| ***Actors:***  User  ***Summary:***  This feature allows the actor to see a View Showroom list.  ***Goal:***  View the list of all Showrooms available  ***Triggers:***  On each “Home page”, there will be a “Booking” button on the navigation bar for the customer to press. When pressed, we can go to the Booking Showroom page.  ***Preconditions:***  *None.*  ***Post Conditions:***  *None.*  ***Main Success Scenario:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Press the ”View Showroom” button on the navigation bar. | The system will redirect them to the booking showroom page.  The page will have:   * Each showroom address. * “Book Showroom” button on each showroom grid. * “View Booked Showroom” button |   ***Alternative Scenario:***     | ***No.*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | 1 | At the booking showroom page, there is a list of recommended showrooms available based on popularity | The system will retrieve information from the database, synthesize it, and return it in the recommended form. |   ***Exceptions:***    None.  ***Relationships:***  *Booking Date & Time*  ***Business Rules:***  *None* | | | | |

## 

## View Booked Showroom

| ***USE CASE 25 VIEW BOOKED SHOWROOM*** | | | | |
| --- | --- | --- | --- | --- |
| ***Use-case No.*** | UC025 | ***Use-case Version*** | | 1.0 |
| ***Use-case Name*** | View Booked Showroom | | | |
| ***Author*** | Vũ Đức Anh | | | |
| ***Date*** | 13/01/2024 | ***Priority*** | High | |
| ***Actors:***  Consultant, User  ***Summary:***  This feature allows the actor to see the Booked Showroom.  ***Goal:***  View the Booked Showroom available  ***Triggers:***  The user wants to see their booked showroom.  ***Preconditions:***  *None.*  ***Post Conditions:***  *None.*  ***Main Success Scenario:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Press the ”Booking” button on the navigation bar. | The system will redirect them to the booking showroom page.  The page will have:   * Search bar “Enter address”   and the “Book Showroom” button on each showroom grid.   * A “Booked Showroom” button * “Submit” button | | 2 | The user clicks on the “Booked Showroom” button | The system will redirect to Booked Showroom page which contain information of the user booked showroom:   * Address * Date and Time |   ***Alternative Scenario:*** None  ***Exceptions:***     | ***No.*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | If user do not book a showroom, and try to press at the “Booked Showroom” button | The pop-up “You do not have any booking” appears and that button is hidden. |   ***Relationships:***  *View Booking List*  ***Business Rules:***  *None* | | | | |

## View Visitors In A Showroom

| ***USE CASE 26 VIEW VISITORS IN A SHOWROOM*** | | | | |
| --- | --- | --- | --- | --- |
| ***Use-case No.*** | UC026 | ***Use-case Version*** | | 1.0 |
| ***Use-case Name*** | View Visitors In A Showroom | | | |
| ***Author*** | Nguyễn Hùng Khương | | | |
| ***Date*** | 13/01/2024 | ***Priority*** | High | |
| ***Actors:***  Consultant  ***Summary:***  This feature allows the Consultant to see the Booked Showroom.  ***Goal:***  View the List of visitor to that showroom.  ***Triggers:***  Must login as the consultant of that showroom.  ***Preconditions:***  *None.*  ***Post Conditions:***  *None.*  ***Main Success Scenario:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | The consultant will see a list of customer who visits the showroom. | The list will include the customer’s personal infos:   * The showroom schedule hours. * Customer name * Customer phone number   The page will include a select control textfield for the consultant to choose a specific schedule hour.   * Paging method for ease of view. |   ***Alternative Scenario:***     | ***No.*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | 1 | Consultant will select the specific hour of the showroom | The system will show only the list of customers who visit the showroom at that hour/ |   ***Exceptions:***  None.  ***Relationships:***  None.  ***Business Rules:***  *None* | | | | |

## View Revenue and Item Reports

| **USE CASE 27 VIEW REVENUE AND ITEM REPORTS** | | | | |
| --- | --- | --- | --- | --- |
| **Use-case No.** | UC027 | **Use-case Version** | | 1.0 |
| **Use-case Name** | View Revenue and Item Reports | | | |
| **Author** | Vu Duc Anh | | | |
| **Date** | 02/02/2024 | **Priority** | Medium | |
| **Actor:**  Admin  **Summary:**  This use case allows the actor to manage their reports by viewing revenue and item reports.  **Goal:**  The actor can see revenue and item reports in detail such as in-stock products, out-of-stock products, and product revenue.  **Triggers**  The actor wants to see revenue and item reports to manage.  **Preconditions:**  The actor must log in to the system an “Admin” role.  The actor must have their Internet connected.  **Post Conditions:**  The system will load the Revenue and Item Reports dashboard and reveal all the components of those reports.  **Main Success Scenario:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | The actor clicks the “Manage Reports” button. | The system will redirect to the Manage Reports page which contains two dashboards: User Report and Revenue And Item Report. | | 2 | Actor clicks on the “Revenue and Item Report” button | The system will redirect to the Revenue and Item Report dashboard. The dashboard appears which contains this information:   * In-stock products * Out-of-stock products * Product Revenue * A “Export to PDF” button |   **Alternative Scenario:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | If the actor is browsing at any dashboard (including the User Report dashboard), click on the “Revenue and Item Report” button | The system will redirect to the Revenue and Item Report dashboard. The dashboard appears which contains this information:   * In-stock products * Out-of-stock products * Product Revenue * A “Export to PDF” button |   **Exceptions:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | If the actor is selecting an empty dashboard | The system will redirect to that dashboard and show a “No information” message. |   **Relationships:** Manage Reports  **Business Rules:**   * The revenue and stocks must be updated every 2 mins. | | | | |

## View Orders History

## 

| ***USE CASE 28 VIEW ORDERS HISTORY*** | | | | |
| --- | --- | --- | --- | --- |
| ***Use-case No.*** | UC028 | ***Use-case Version*** | | 1.0 |
| ***Use-case Name*** | View Orders History | | | |
| ***Author*** | Nguyễn Hùng Khương | | | |
| ***Date*** | 13/01/2024 | ***Priority*** | High | |
| ***Actors:***  *Userf.*  ***Summary:***  This feature allows the actor to see all of the order history, divided by each time.  ***Goal:***  View all the orders History  ***Triggers:***  On each customer’s profile page, there will be a button for the customer to view the order history*.*  ***Preconditions:***  *None.*  ***Post Conditions:***  *None.*  ***Main Success Scenario:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Press the”View order history” button | System will redirect them to the order history page.  Order history page will have:   * Search bar “Enter ID order” * “Search” Button | | 2 | Enter ID order | System will check if the “ID order” is exist or not | | 3 | Press “Search” button | The system will take the input information and cross-reference them with what is in the database. then return the appropriate ordering data |   ***Alternative Scenario:***     | ***No.*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | 1 | 1. At the search bar, users can enter user name for searching if user don’t remember the ID order | The system will take the input information and cross-reference them with what is in the database. then return the appropriate ordering data |   ***Exceptions:***     | ***No.*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | 1. *Submit the wrong ID order or user name .* | *A pop-up will appear warning that the order does not exist “(Order doesn’t exist, …)”.* | | *2* | 1. *The order is not in the database.* | *System popups a warning saying “ the order is invalid.”* |   ***Relationships:***  *None.*  ***Business Rules:***  *None* | | | | |

## 

## Booking Date & Time

## 

| ***USE CASE 29 BOOKING DATE & TIME*** | | | | |
| --- | --- | --- | --- | --- |
| ***Use-case No.*** | UC029 | ***Use-case Version*** | | 1.0 |
| ***Use-case Name*** | Booking Date & Time | | | |
| ***Author*** | Nguyễn Hùng Khương | | | |
| ***Date*** | 13/01/2024 | ***Priority*** | High | |
| ***Actors:***  User  ***Summary:***  This feature allows the actor to select available appointments .  ***Goal:***  Set available appointments time to view showroom  ***Triggers:***  On the “Booking showroom page”, there will be a “Choose Date & Time” button for the customer to press. When pressed, users will see a Date & Time dashboard.  ***Preconditions:***  *None.*  ***Post Conditions:***  *None.*  ***Main Success Scenario:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Press the “Booking” button on the “Booking showroom page”. | The system will redirect them to the Time dashboard with a schedule.  The time dashboard will have:   * A “Confirm” button on each time schedule. | | 2 | Press the “Confirm” button. | The system will take the input.  Then record the schedule information with the user information. |   ***Alternative Scenario:***     | ***No.*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | 1 | At the booking showroom page, there is a list of recommended setup time available based on popularity | The system will retrieve information from the database, synthesize it and return it in the recommended form |   ***Exceptions:***     | ***No.*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | Submit the wrong time form(time has passed) | *A pop-up will appear warning that invalid time“(Time does not invalid, …)”.* |   ***Relationships:***  *Booking Showroom*  ***Business Rules:***  *None* | | | | |

## Buy eWallet Credit

| **USE CASE 30 BUY EWALLET CREDIT** | | | | |
| --- | --- | --- | --- | --- |
| **Use-case No.** | UC030 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Buy eWallet Credit | | | |
| **Author** | Euan Cockburn | | | |
| **Date** | 02/02/2024 | **Priority** | Medium | |
| **Actor:**  Customer  **Summary:**  This use case will enable customers to purchase eFurniture credit..  **Goal:**  Credit is added to the eWallet  **Triggers**  A ‘Buy eFurniture credit’ button which will be on the customers eWallet page (not yet described by a use case).  **Preconditions:**  The customer must be logged in to his account and his email must be verified.  **Post Conditions:**  The customer’s eWallet balance will have increased by the amount of credits purchased.  **Main Success Scenario:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | The actor clicks on buy eFurniture Credit | The system displays a list of options for the amount of credit to buy. The list will be a range starting at a minimum of 100k vnd up to 10m vnd at intervals of 100k up to 1m and then 1m up to 10m. There will also be a custom amount option with a minimum of 100k and a maximum of 1000m vnd.  After the list of options will be a confirm button which will be enabled once the customer has picked an amount. | | 2 | The actor picks an option and clicks on confirm | The system will display the available payment options which should be: direct bank transfer, payment app, paypal and credit card. | | 3 | The actor picks the payment option he wishes to use. | The system redirects to the appropriate payment handling interface. Internally handled for the credit card and third party middleware for the other three options. | | 4 | The actor transfers the funds. | Once the middleware informs the system that the payment has been executed, the system adds the amount of credits to the customer's account. The system then displays payment confirmation to the customer for 5 seconds before redirecting them to the eWallet page. |   **Alternative Scenario:**  None  **Exceptions:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | There is an error during payment for the credits. | The payment middleware signals to the system that there has been an error in payment. The system displays a message informing the customer of the error and asks them to try again. At the end of the message is a link to redirect the customer back to the eWallet page.  No funds are added to the customers eWallet. | | 2 | The actor cancels payment | The payment middleware signals to the system that the payment has been canceled. The system displays a message saying the payment was canceled and after 5 seconds redirects the user to the eWallet page.  No funds are added to the customers eWallet |   **Relationships:** None  **Business Rules:** None | | | | |

## Refund Into eWallet

| **USE CASE 25 REFUND INTO EWALLET.** | | | | |
| --- | --- | --- | --- | --- |
| **Use-case No.** | UC31 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Refund Into eWallet | | | |
| **Author** | Euan Cockburn | | | |
| **Date** | 02/02/2024 | **Priority** | Medium | |
| **Actor:**  Customer, staff  **Summary:**  This use case will enable the customer to get a refund of the money they spent on a product in the form of eFurniture credit. This will be used when the customer returns a product or cancels an order before it is shipped.  **Goal:**  The customer will get their money back while still spending it on the eFurniture website.  **Triggers**  A customer returns their order or cancels it before it is shipped to them. This sends a refund ticket to a member of staff who will have a refund page on their interface.  **Preconditions:**  The customer must have ordered a product and paid for it.  **Post Conditions:**  The customer’s eWallet balance will have increased by the amount of credits refunded  **Main Success Scenario:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | The actor returns an item of furniture. Upon receiving the item the member of staff connects to their interface and clicks on the refund ticket. | The system displays the refund ticket, this includes details of what happened (when the refund request was issued, what the reason is for the refund, how much the furniture was purchased for) as well as options to handle the ticket. These options are as follows : refund full amount to customers eWallet, refuse refund. | | 2 | The staff picks the refund full amount option. | The system transfers the amount of credit equivalent to the price of the furniture into the relevant customer’s eWallet.  The system then notifies the customer of the refund by sending them an email. The email will contain the amount refunded, the date refunded and the information relevant to the order that was refunded. |   **Alternative Scenario:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | The actor returns an item of furniture. Upon receiving the item the member of staff connects to their interface and clicks on the refund ticket. | The system displays the refund ticket, this includes details of what happened (when the refund request was issued, what the reason is for the refund, how much the furniture was purchased for) as well as options to handle the ticket. These options are as follows : refund full amount to customers eWallet, refuse refund. | | 2 | The staff picks the refuse refund option.. | The system gives the staff a text field with a maximum of 240 characters to explain why they refused the refund. After the text field will be a submit button. | | 3 | The staff fills in the text field and presses the submit button | The system notifies the customer of the refusal in an email including the details relevant to the order as well as the message from the member of staff justifying the refusal. |   **Exceptions:** None  **Relationships:** None  **Business Rules:**  For refunds above 1m vnd, the refund must be validated by a manager. | | | | |

## View A Furniture List

| ***USE CASE 26 VIEW A FURNITURE LIST*** | | | | |
| --- | --- | --- | --- | --- |
| ***Use-case No.*** | UC032 | ***Use-case Version*** | | 1.0 |
| ***Use-case Name*** | View A Furniture List | | | |
| ***Author*** | Nguyễn Hùng Khương | | | |
| ***Date*** | 13/01/2024 | ***Priority*** | High | |
| ***Actors:***  Staff, User, Guest, Admin  ***Summary:***  This feture allows the user to see all the lists of furniture by clicking a button on the home page at the menu bar.  ***Goal:***  This feature allows the actor who wants to see the furniture detail to go through the furniture list page by clicking the “Furniture list” button.  ***Triggers:***  Actor want to view the furniture list can click the ”Furniture list” button on the menu bar on the home page  ***Preconditions:***  *None.*  ***Post Conditions:***  *None.*  ***Main Success Scenario:***   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | On the furniture list page | System will show all the lists of furniture.  View page will have:   * Search bar “Enter name or ID of furniture” on the top of page. * “Search” button on the right side of the search bar. * “Detail” button at the tail of each furniture. | | 2 | Enter ID or name furniture | System will check if the “Furniture” is available or not | | 3 | Press “Search” button | The system returns furniture that matches the search result. | | 4 | Press “Detail” button | System will be redirected to the “View a furniture detail” of that product |   ***Alternative Scenario:***     | ***No.*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | 1 | Actor clicks on the picture of the furniture | System will be redirected to the “View a furniture detail” of that product | | 2 | Enter the ID or name furniture.  Press the “Search” button. | The system will check if the “Furniture” is available or not.  The system returns furniture that matches the search result. |   ***Exceptions:***  None  ***Relationships:***  View A Furniture Detail  ***Business Rules:***  *None* | | | | |

# NON-FUNCTIONAL Requirements

## Usability

* User-Friendly Interface: The website will have an intuitive and easy-to-navigate interface. It shall be structured in a way that allows users to find desired information or carry out the required actions with ease.
* Responsive Design: The website shall be accessible on various devices with different screen sizes. It shall automatically adjust and remain aesthetically pleasing and functional without requiring a user to resize, pan, or scroll excessively.
* Accessibility: The website shall be built according to web accessibility standards, ensuring it is usable by people with a wide range of abilities and disabilities.

## Reliability

* Availability: The website should be available to users around the clock with a minimum of 99.5% uptime, except during maintenance periods.
* Error Handling: The website should handle errors gracefully, displaying friendly error messages to the user and logging these errors for further debugging and fixing.
* Data Integrity: All transactions on the site, such as online orders or payments, must be processed reliably, ensuring data isn't lost or corrupted.
* Backups: Regular backups must be maintained to ensure site recovery in the event of unforeseen circumstances.

## Performance

* Load Time: Website pages shall load in 2-3 seconds or less under standard network conditions.
* Scalability: The website should be capable of handling a large number of users simultaneously, without performance issues.
* Optimization: Images and other media files on the website should be optimized for the web to ensure they don't slow down the site's load time.
* Security: The website will utilize up-to-date security practices to secure user data and transactions conducted on the site.

# Supporting Information

* Customer Support: We will provide a dedicated customer support team reachable via email, phone, and live chat on the website. The support will be available during business hours and response times to all queries should not exceed 24 hours.
* Technical Support: A skilled technical support team will be available to handle system errors, malfunctions, or any user-facing issues with the website's functionality. This team will operate 24/7 to minimize downtime and improve the website's reliability.
* Documentation: There will be detailed user manuals and guides incorporated into the website which will assist users in navigating different website features. Additionally, a comprehensive FAQ section will be included for common inquiries.