<Newbie Coder>

Coffee Base
Use-Case Specification

Version <1.4>

Revision History

Date	Version	Description	Author
05/12/25	1.4	Fix UC	Phát
22/11/25	1.3	Add website UI Designs	Cường
20/11/25	1.2	finish 1. And 2. Text information 2. Still need UX/UI images	Thanh
20/11/25	1.1	finish first version for sprint 2	Quang, Thanh

©<Newbie Coder>, 2025

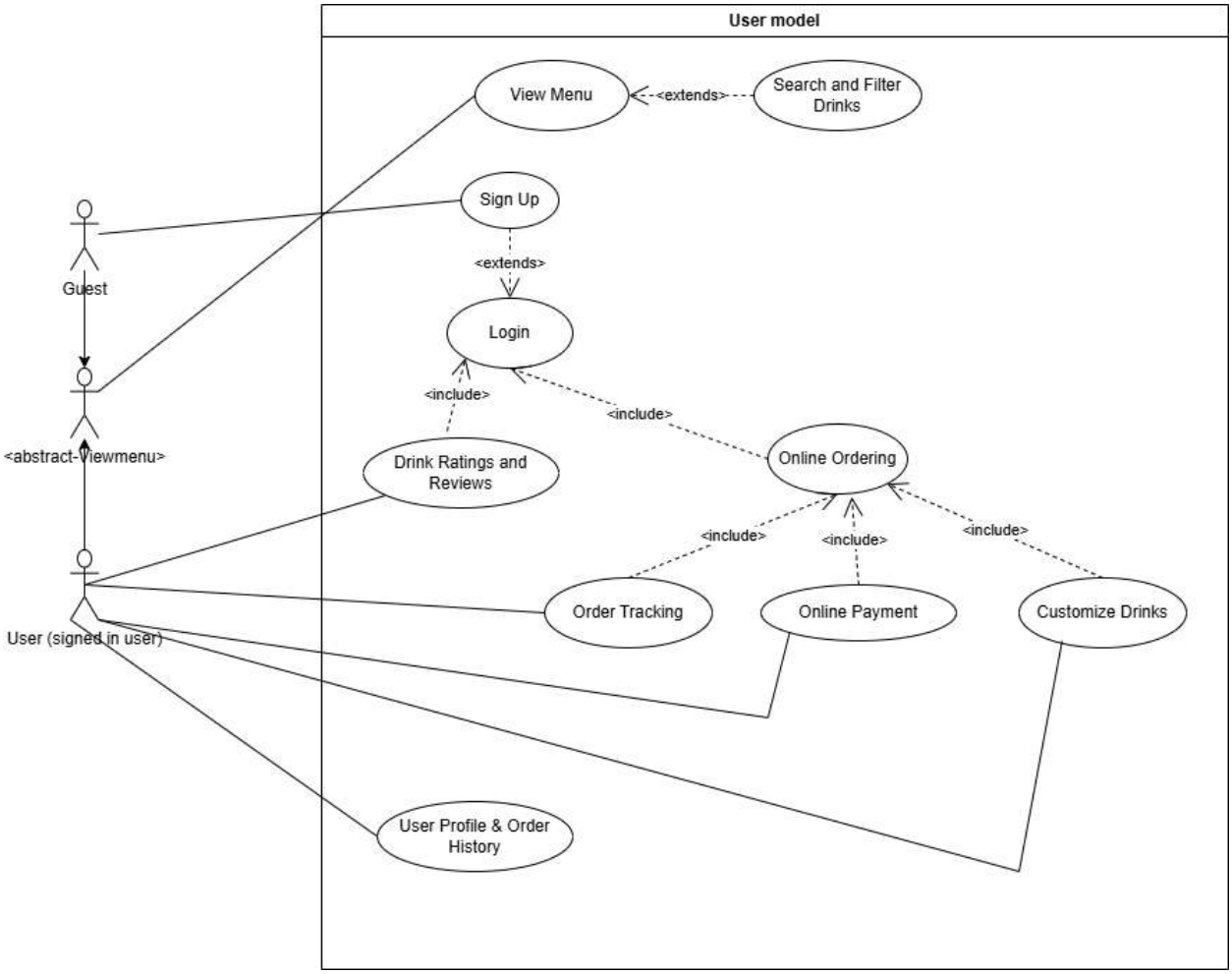
Table of Contents

1. Use-Case Model 4
    1.1 User Model 4
    1.2 Admin Model 5
2. Use-case Specifications 5
    2.1 Use-case Specifications for User Model 5
        2.1.1 Use-case specification: View menu 5
        2.1.2 Use-case specification: Sign Up 6
        2.1.3 Use-case specification: Login 6
        2.1.4 Use-case specification: Search and filter drinks 6
        2.1.5 Use-case specification: Drinks rating and review 6
        2.1.6 Use-case specification: Online Ordering 6
        2.1.7 Use-case specification: Order Tracking 6
        2.1.8 Use-case specification: Online Payment 6
        2.1.9 Use-case specification: Customize Drinks 6
        2.1.10 Use-case specification: Users profile and order history 6
    2.2 Use-case Specifications for Admin Model 6
        2.2.1 Use-case specification: Login and Authentication 6
        2.2.2 Use-case specification: Manage menu 7
        2.2.3 Use-case specification: Manage users 8
        2.2.4 Use-case specification: Manage orders 8
        2.2.5 Use-case specification: View sales statistics (time - based) 9
        2.2.6 Use-case specification: Manage feedback & reviews 9
        2.2.7 Use-case specification: Manage promotions / discounts 10

©<Newbie Coder>, 2025

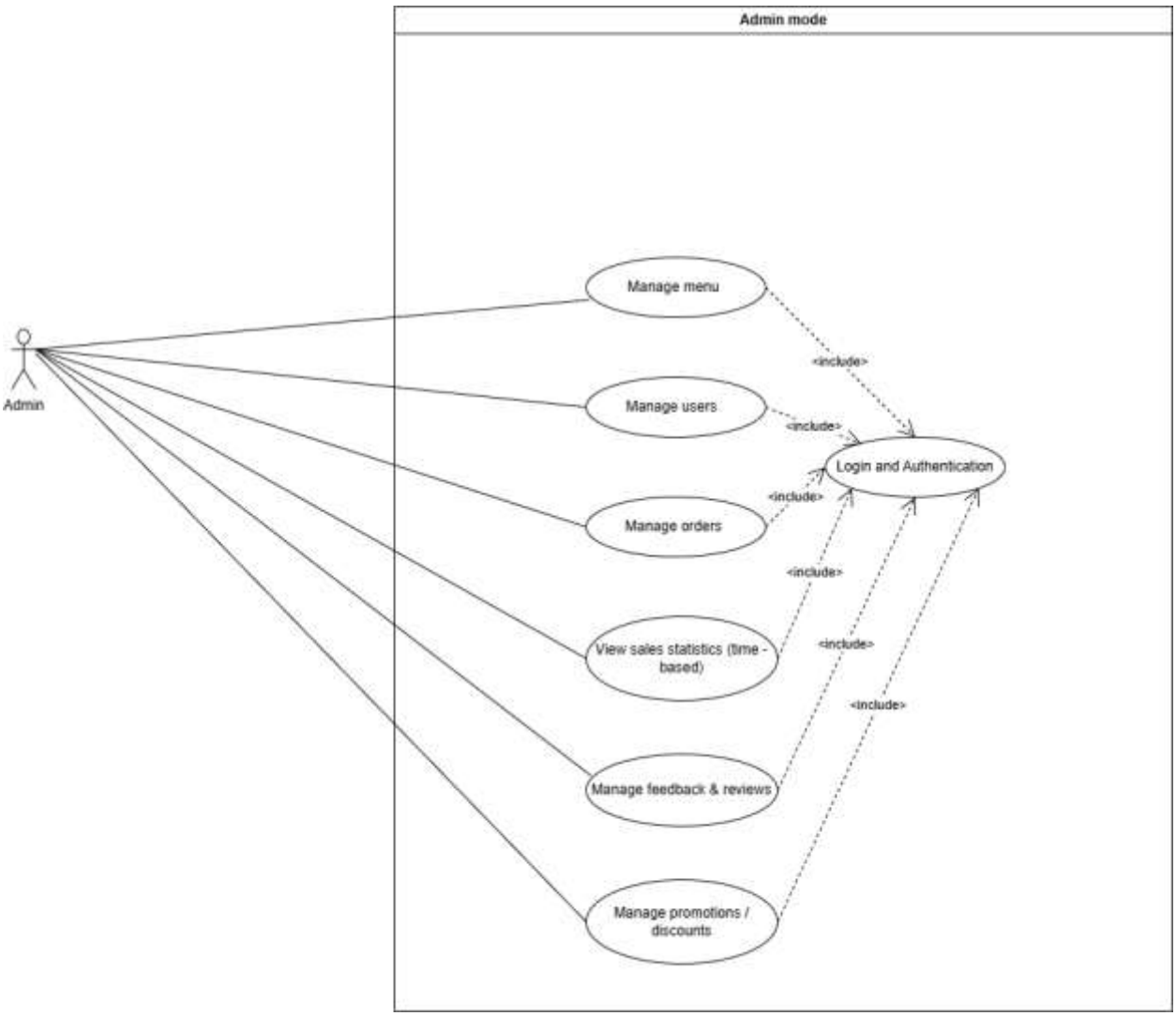
1. Use-Case Model

1.1 User Model



©<Newbie Coder>, 2025

1.2 Admin Model

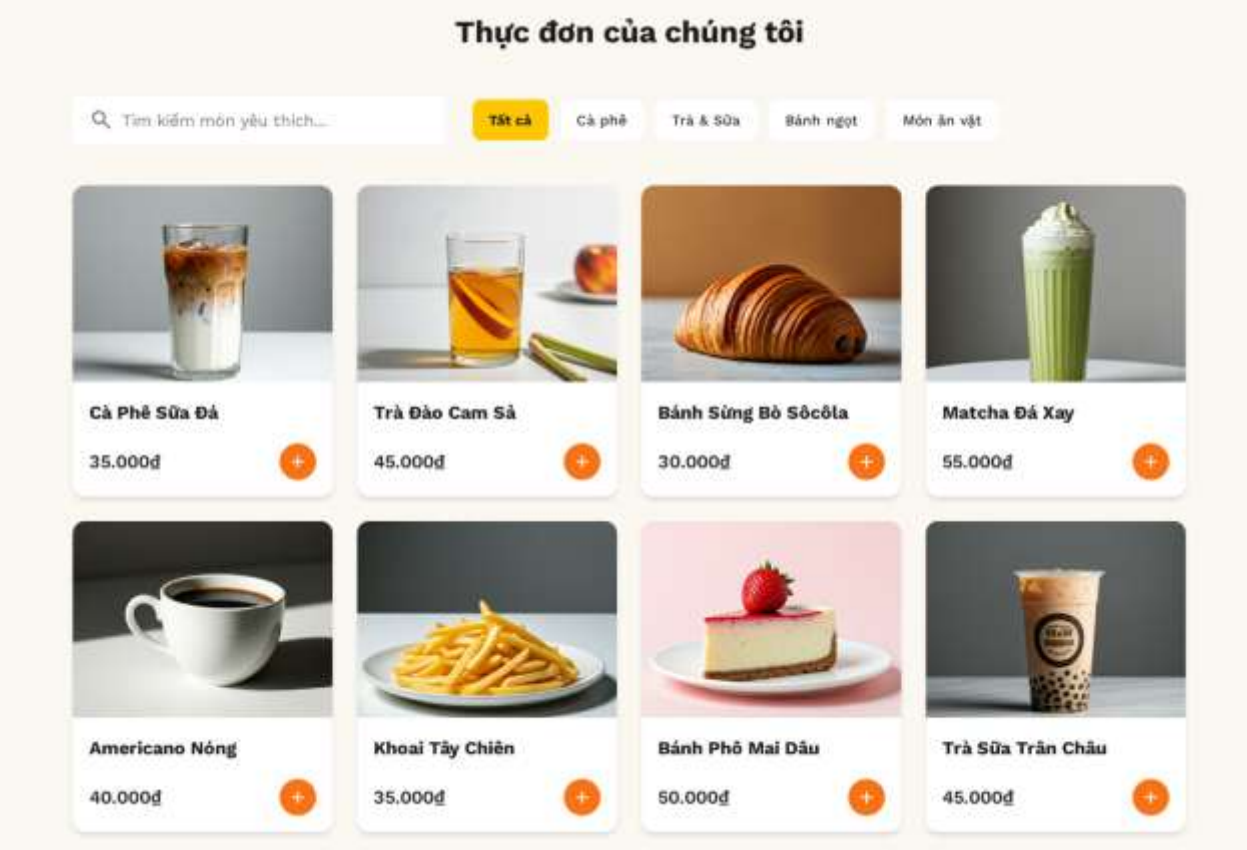


©<Newbie Coder>, 2025

2. Use-case Specifications

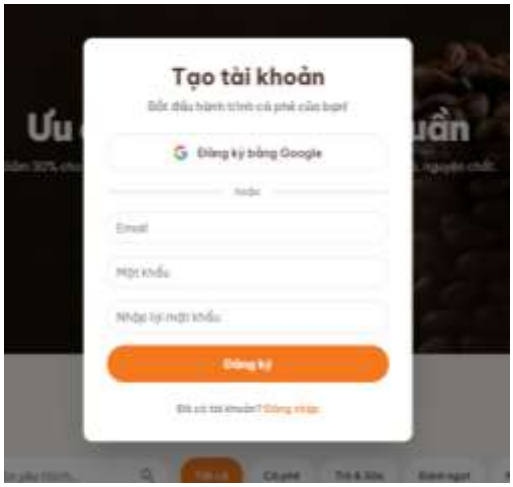
2.1 Use-case Specifications for User Model

2.1.1 Use-case specification: View menu



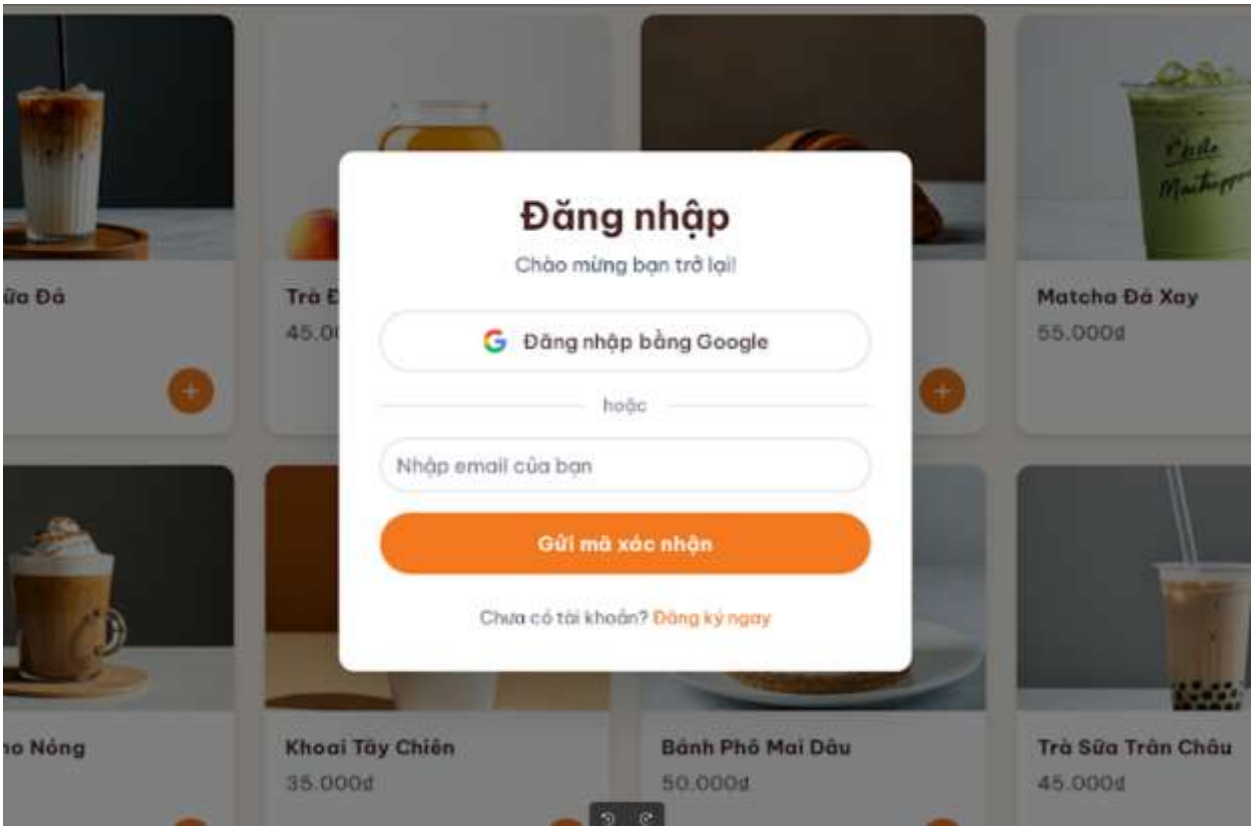
Use case ID	UCU01
Use case Name	View drink menu
Brief description	Describes how a user (Guest or Signed-in User) can view the list of available drinks and side items.
Actors	Guest, User (signed-in user)
Pre-conditions	User/Guest is on the website.
Basic Flow	1. The user navigates to the "Menu" page. 2. The system displays a list of drinks and side items with images, descriptions, prices, and categories. 3. The user can browse the menu.
Alternative Flows	2a. If an item is out of stock, it is displayed but marked as unavailable.
Post-conditions	The user has seen the available menu items.
Non-functional Requirements	**Performance**: Main pages (Menu) should load within 2–4 seconds. **Usability**: Interface must be simple and clear.

2.1.2 Use-case specification: Sign Up



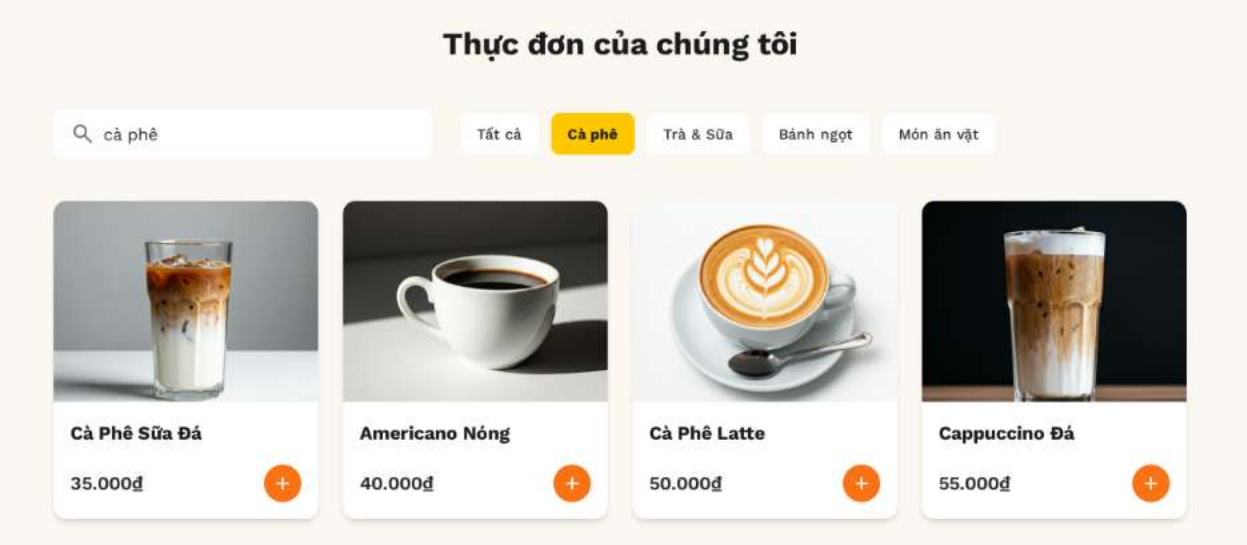
Use case ID	UCU02
Use case Name	Sign Up
Brief description	Describe how a new customer creates an account to access online ordering and payment services.
Actors	Guest
Pre-conditions	User is on the website and is not logged in.
Basic Flow	1. The user clicks the "Sign Up" button. 2. The system displays the registration form (Name, Email, Password, etc.). 3. The user enters valid information and submits the form. 4. The system validates the input (e.g., unique email, strong password). 5. The system creates a new user account and automatically logs the user in.
Alternative Flows	4a. **Invalid Input:** If the input is invalid (e.g., email already exists, password too weak), the system displays an error message and keeps the form open.
Post-conditions	A new User account is created, and the User is logged in.
Non-functional Requirements	**Security**: All user passwords must be encrypted before being stored.

2.1.3 Use-case specification: Login



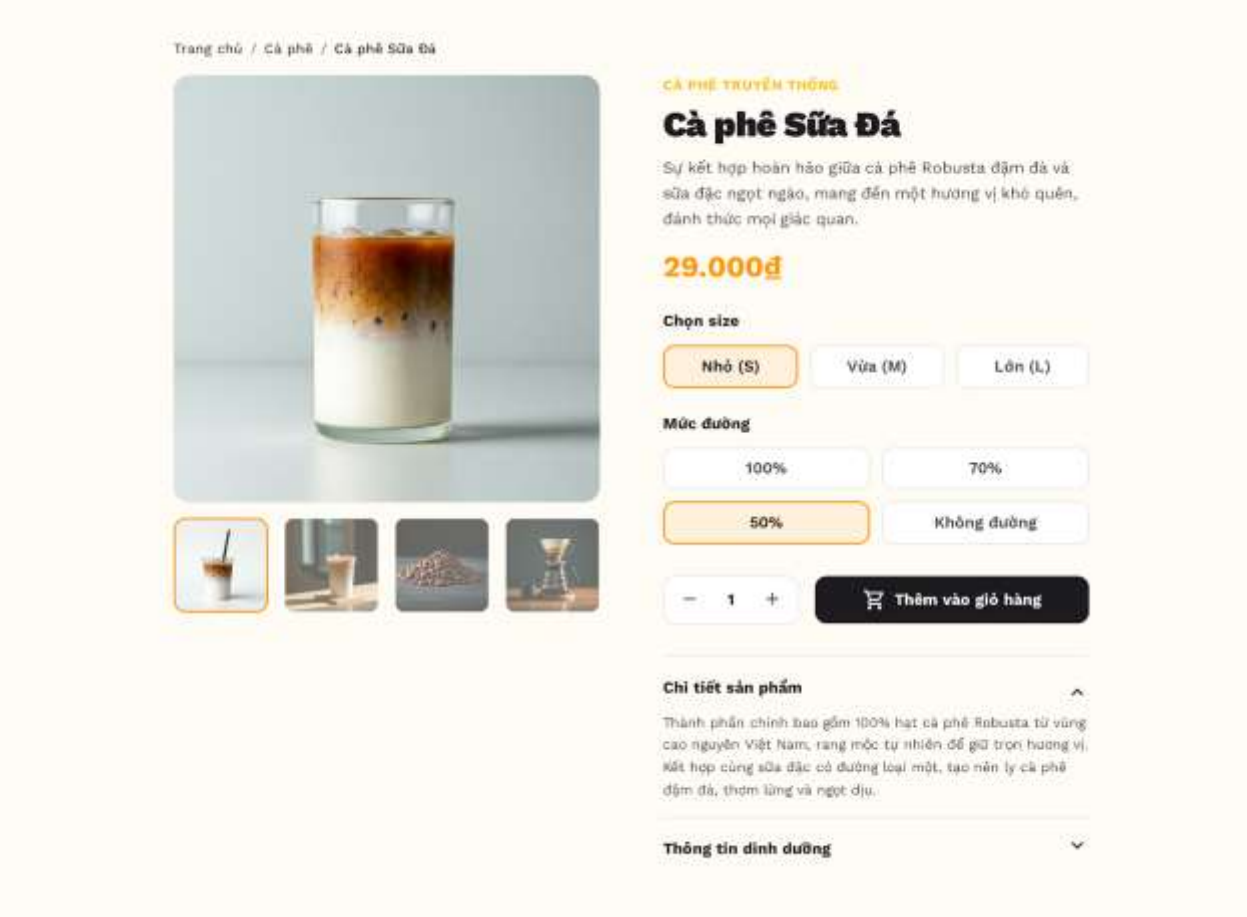
Use case ID	UCU03
Use case Name	Login
Brief description	Describes how a registered customer can log in to access personalized services and history information.
Actors	User (who is currently logged out)
Pre-conditions	User is on the website and is not logged in
Basic Flow	1. The user clicks the "Login" button. 2. The system displays the login form (Email/Username and Password). 3. The user enters credentials and submits. 4. The system verifies the credentials. 5. The system grants access and redirects the user to the home page or a personalized dashboard.
Alternative Flows	4a. **Invalid Credentials:** If credentials do not match, the system displays an error message. 4b. **Forgot Password:** User clicks "Forgot Password," and the system initiates a password reset process (e.g., sending an email).
Post-conditions	The User is successfully logged in and goes to the homepage.
Non-functional Requirements	**Security**: Login sessions should expire after a certain period of inactivity.

2.1.4 Use-case specification: Search and filter drinks



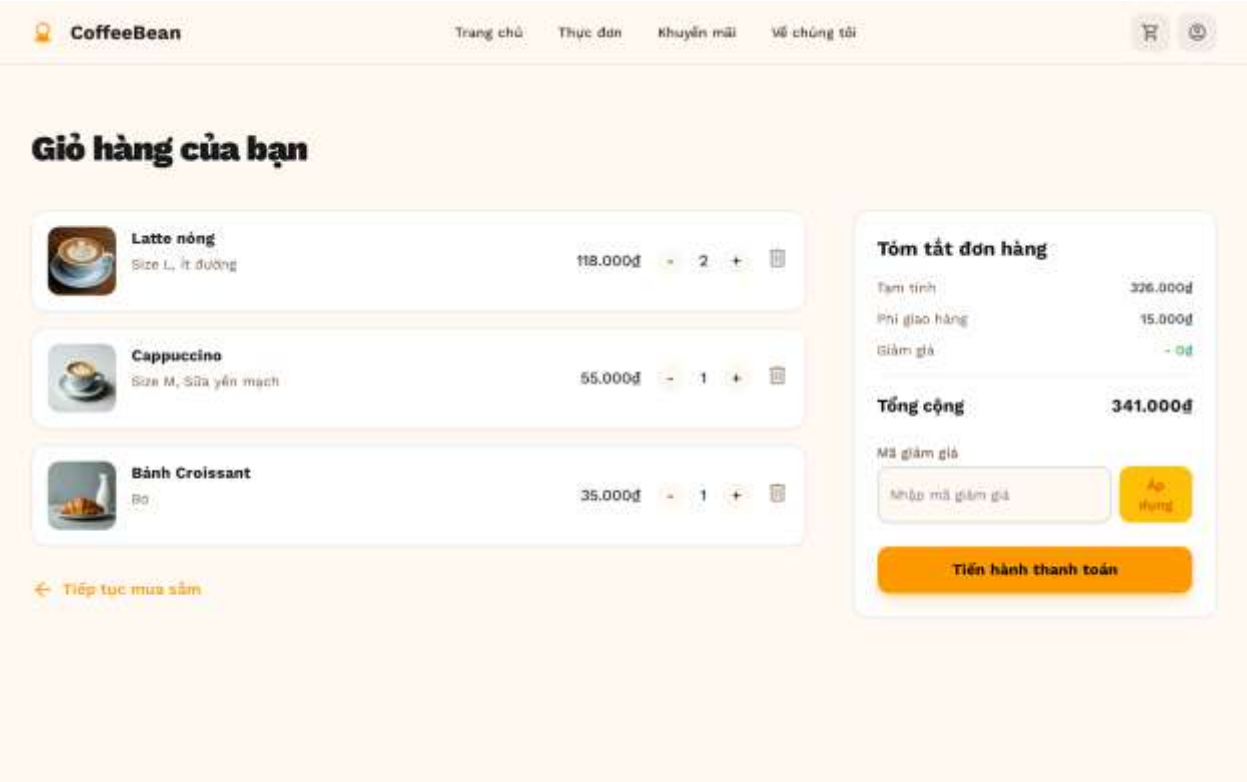
Use case ID	UCU04
Use case Name	Search and filter drinks
Brief description	Allows users to find specific drinks by searching by name, category, price, or promotions, and to sort the results.
Actors	Guest, User (signed-in user)
Pre-conditions	Guest is on the website and is not logged in
Basic Flow	1. The user enters text into the search bar or selects filtering/sorting options (e.g., Category, Price: range number). 2. The system processes the query. 3. The system displays a new list of menu items that match the search and filter criteria.
Alternative Flows	3a. **No Results:** If no items match the criteria, the system displays a "No Results Found" message.
Post-conditions	The user is viewing a filtered/sorted list of menu items.
Non-functional Requirements	**Performance**: Search results should be returned quickly (<2.5 seconds).

2.1.5 Use-case specification: Customize Drinks



Use case ID	UCU05
Use case Name	Customize Drinks
Brief description	Allows users to select options like cup size, toppings, sweetness/ice level, and milk type for a specific drink.
Actors	User (signed-in user)
Pre-conditions	User is logged in and is viewing a specific drink's detail page.
Basic Flow	1. The user selects a drink from the menu. 2. The system displays the customization options (size, toppings, milk, sugar, ice, etc.). 3. The user selects their preferences. 4. The user clicks "Add to Cart." 5. The system calculates the final price and adds the customized drink to the shopping cart.
Alternative Flows	3a. **Invalid Combination:** If a user selects an incompatible option (e.g., out-of-stock topping), the system prevents the selection and displays a message.
Post-conditions	The customized drink is added to the user's shopping cart.
Non-functional Requirements	**Data Integrity**: The custom order details and final price must be stored correctly.

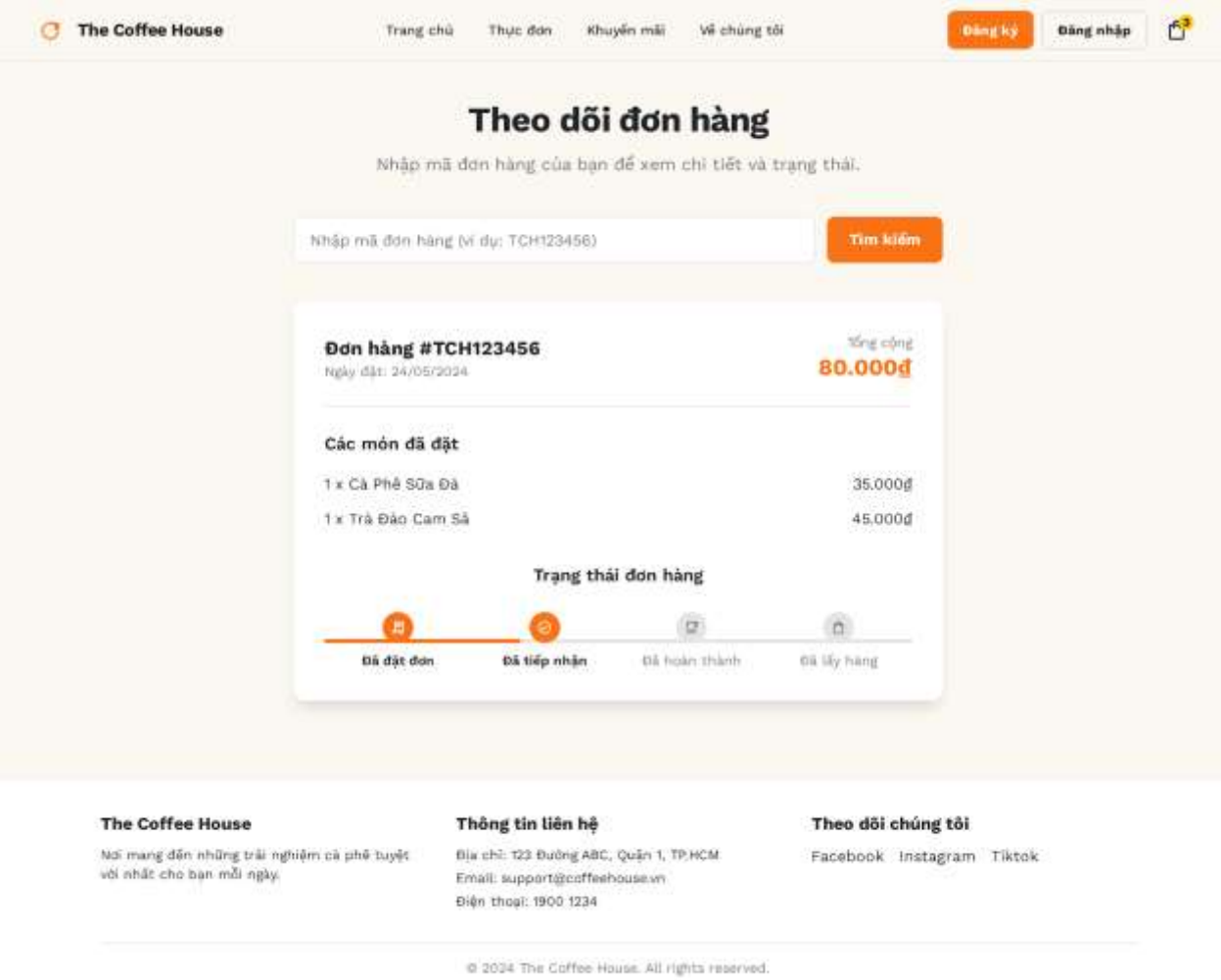
2.1.6 Use-case specification: Online Ordering



Use case ID	UCU06
Use case Name	Online Ordering
Brief description	Enables customers to select products, add them to the cart, confirm the order, and choose a payment method.
Actors	User (signed-in user)
Pre-conditions	User is logged in and has items in their shopping cart.
Basic Flow	1. The user reviews the items in the Cart and clicks "Proceed to Checkout." 2. The system prompts for delivery/pickup details (e.g., pickup time, location). 3. The system prompts for payment method selection (online or in-store). 4. The user confirms all details and clicks "Place Order." 5. The system creates the order record and sends a confirmation to the user.
Alternative Flows	2a. **Empty Cart:** If the cart is empty, the system displays a message and prevents checkout. 4a. **Order Fails:** If the order cannot be placed (e.g., a critical item is suddenly out of stock), the system notifies the user and reverts the cart.
Post-conditions	A new order record is created with 'Ordered' status; The cart is cleared.
Non-functional Requirements	**Data Integrity**: All order information must be stored correctly without data loss or duplication.

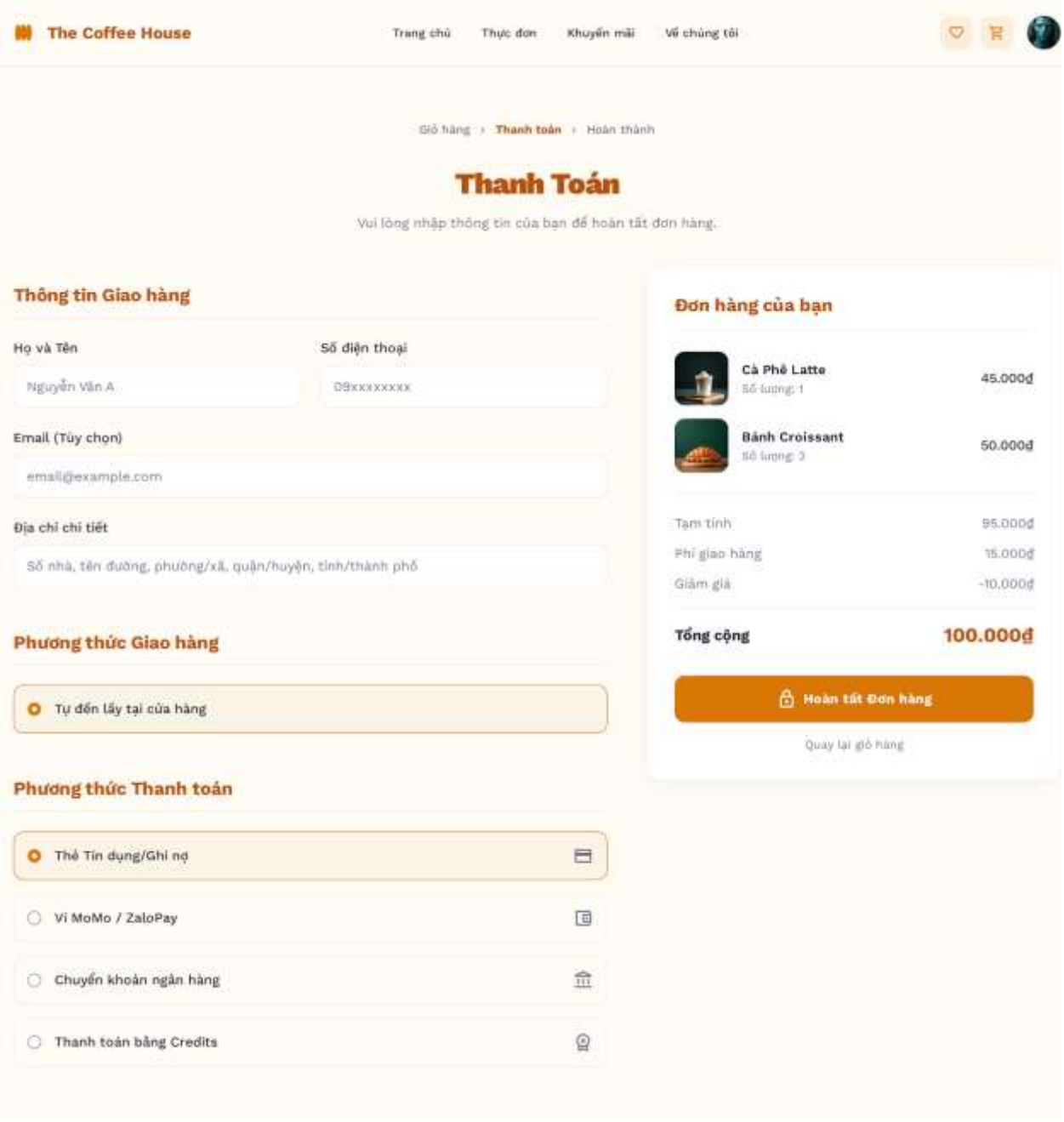
©<Newbie Coder>, 2025

2.1.7 Use-case specification: Order Tracking



Use case ID	UCU07
Use case Name	Order Tracking
Brief description	Shows the customer the real-time status of their placed order (preparing, delivering, completed).
Actors	User (signed-in user)
Pre-conditions	User is logged in and has an active (not completed/canceled) order.
Basic Flow	1. The user navigates to the "Order Tracking" page (or the specific order's detail page). 2. The system retrieves the latest status of the order from the database. 3. The system displays the real-time order status (e.g., 'Processing', 'Ready for Pickup').
Alternative Flows	None.
Post-conditions	The user is informed of the order's current status.
Non-functional Requirements	**Responsiveness**: Order status updates should be near real-time.

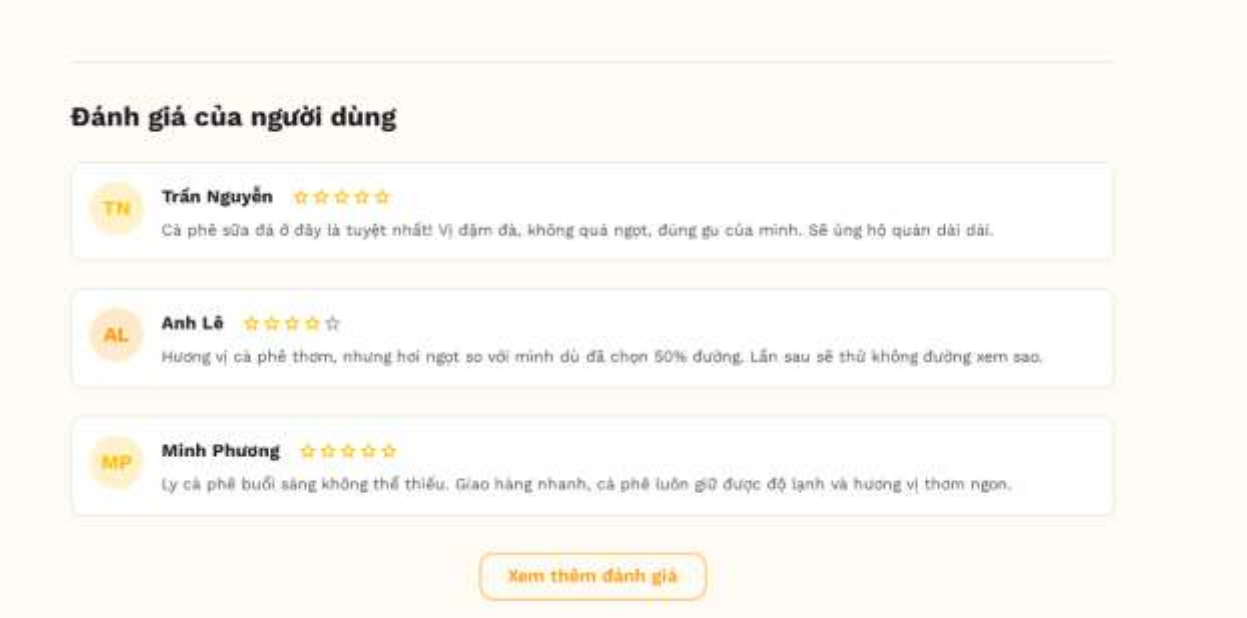
2.1.8 Use-case specification: Online Payment



Use case ID	UCU08
Use case Name	Online Payment
Brief description	Supports the user in completing payment using various methods (bank cards, e-wallets, or credits).
Actors	User (signed-in user)

Pre-conditions	User is placing an order (UCU06) and has selected an online payment method.
Basic Flow	1. The user selects an online payment option (e.g., Momo, Credit Card, our credit score). 2. The system redirects the user to the secure payment gateway/page. 3. The user completes the payment authorization. 4. The payment gateway confirms the successful payment to the system. 5. The system updates the order status to 'Paid' and notifies the user.
Alternative Flows	4a. **Payment Failure:** If payment fails, the system notifies the user, reverts the order status, and prompts the user to try again or select a different method.
Post-conditions	The payment is successfully processed, and the order is marked as 'Paid'.
Non-functional Requirements	**Security**: The payment process must be secure. **Data Integrity**: Payment information must be stored correctly.

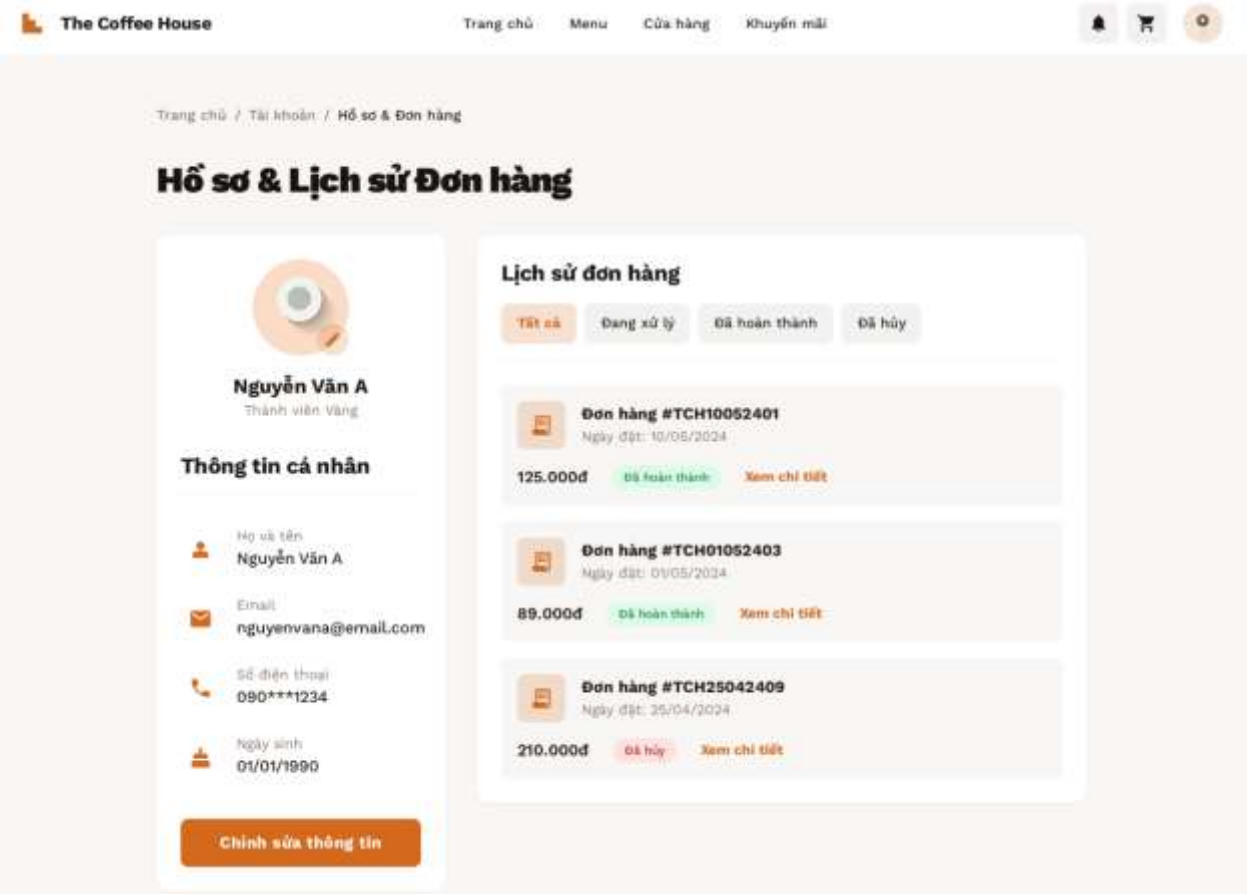
2.1.9 Use-case specification: Drink ratings and reviews



Use case ID	**UCU09**
Use case Name	Drink ratings and reviews
Brief description	Allows customers to rate products with stars and leave comments.
Actors	User (signed-in user)
Pre-conditions	User is logged in.
Basic Flow	1. The user navigates to their order history and selects a completed order. 2. The user selects a drink to review. 3. The system displays the rating/review form (stars and comment box). 4. The user submits a star rating and/or a comment. 5. The system saves the review and updates the drink's average rating.
Alternative Flows	4a. **Edit Review:** The user can edit a previously submitted review, and the system updates the saved content and rating.
Post-conditions	The drink item has a new or updated rating and/or review.

Non-functional Requirements	**Data Integrity** applies to review data.

2.1.10 Use-case specification: User profile and order history



Use case ID	UCU010
Use case Name	User Profile & order history
Brief description	Displays the "My Account" page with personal details, past orders, saved addresses, and payment methods
Actors	User (signed-in user)
Pre-conditions	User is logged in.
Basic Flow	1. The user clicks on their profile icon/name to access the "My Account" page. 2. The system displays personal details, past orders, saved addresses, and payment methods. 3. The user can select a section (e.g., 'Edit Profile', 'Order History'). 4. The system loads the selected page, allowing the user to view or edit the information.
Alternative Flows	4a. **Edit Personal Details:** User edits details, submits, and the system validates and saves the new information.
Post-conditions	The user's personal information or order history is viewed/updated.

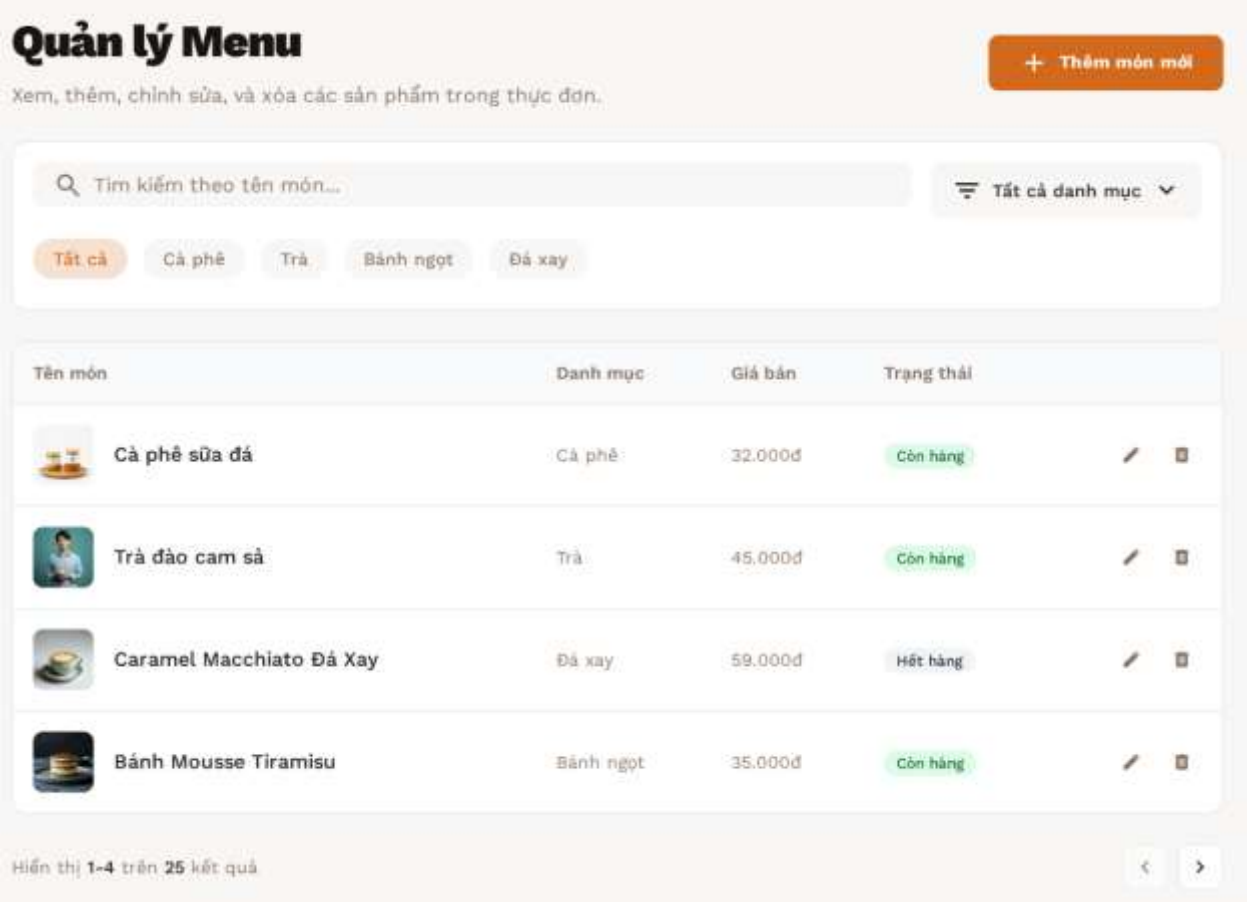
Non-functional Requirements	**Privacy**: Customer personal data must be kept confidential.

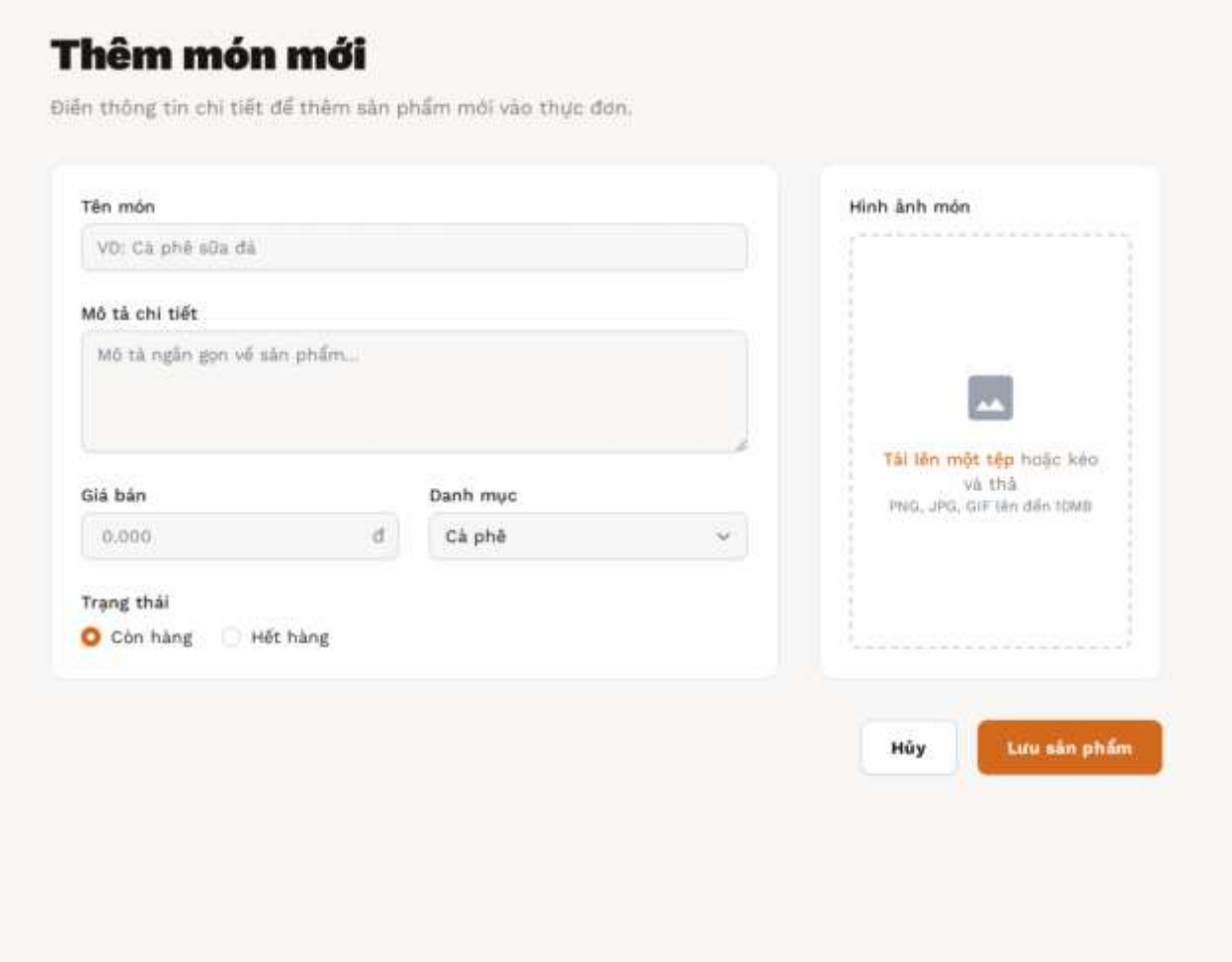
2.2 Use-case Specifications for Admin Model

2.2.1 Use-case specification: Login and Authentication

Use case ID	UCA1
Use case Name	Admin login & authentication
Brief description	Allows only authorized admins to access the management dashboard, supporting role-based access.
Actors	Admin
Pre-conditions	Admin is on the Admin login page.
Basic Flow	1. The Admin enters their credentials (Username/Email and Password). 2. The Admin submits the form. 3. The system verifies the credentials and checks the user's role/authorization. 4. The system grants access to the management dashboard appropriate for their role.
Alternative Flows	3a. Invalid Credentials: If credentials are incorrect, the system displays an error message. 3b. Unauthorized User: If the user logs in successfully but does not have an Admin role, the system denies access to the dashboard.
Post-conditions	The Admin is successfully logged in to the management dashboard.
Non-functional Requirements	**Security**: 1. Supports role-based access. 2. Login sessions should expire after a period of inactivity.

2.2.2 Use-case specification: Manage menu

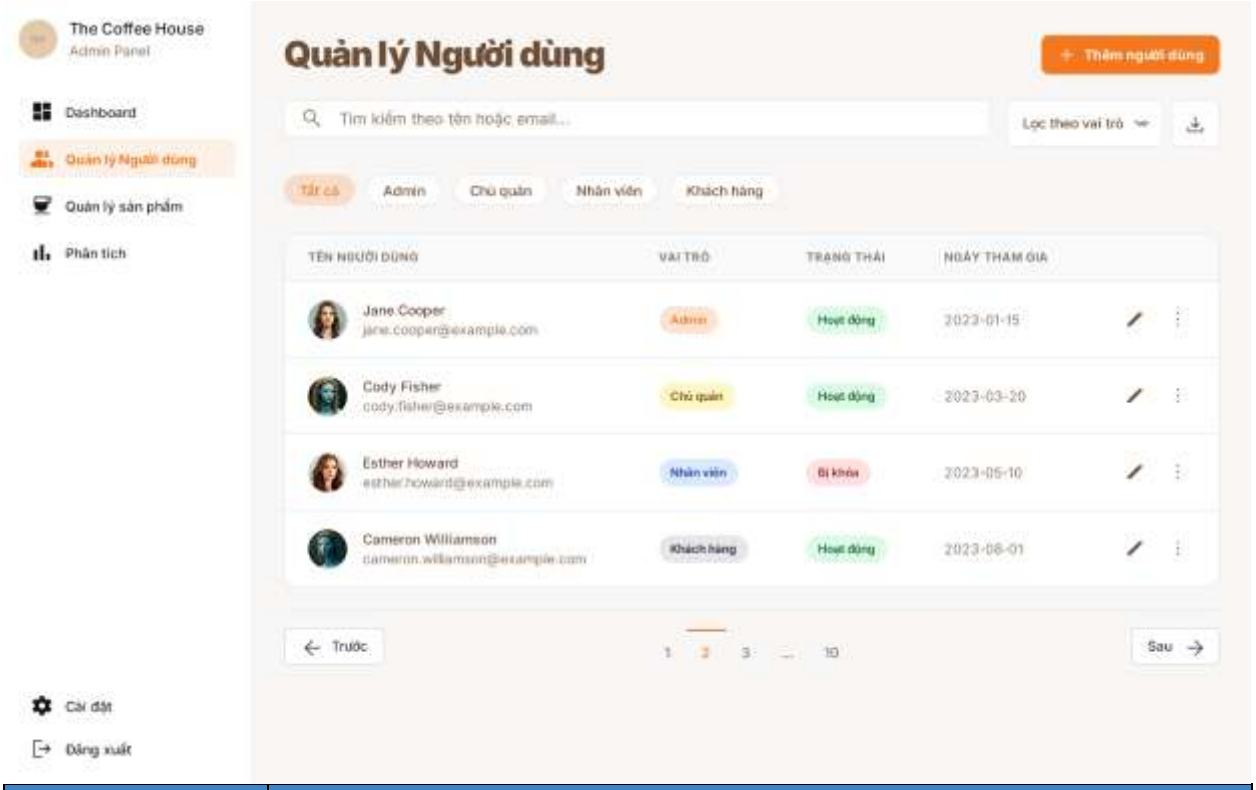




Use case ID	UCA2
Use case Name	Manage menu
Brief description	Admin can create, edit, or remove products in the menu, including setting product details (Name, Price, Tags, Availability Status, etc.).
Actors	Admin
Pre-conditions	Admin is logged in and on the Admin Dashboard.
Basic Flow	1. The Admin navigates to the "Manage Menu" section. 2. The system displays the list of current menu items. 3. (Add) The Admin clicks "Add New Product," fills in required details, and clicks "Save." 4. (Edit) The Admin selects an item, modifies its details, and clicks "Save." 5. The system validates the input and updates the menu list.
Alternative Flows	3a/4a. Invalid Input: If a required field is missing (e.g., Price), the system displays an error message and prevents saving. 4b. Remove Product: Admin selects an item and clicks "Delete." The system prompts for

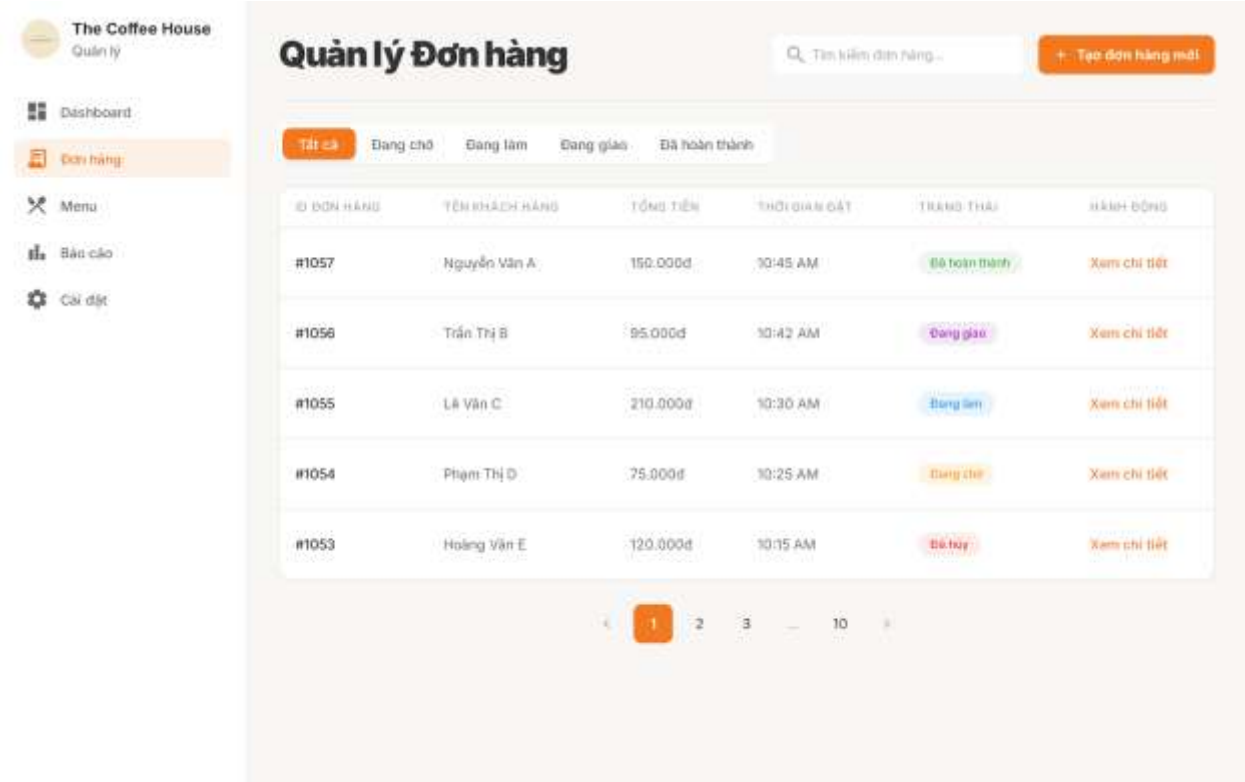
	confirmation and removes the item.
Post-conditions	The menu reflects the changes (new item added, existing item modified, or item removed).
Non-functional Requirements	**Data Integrity**: All product information must be stored correctly.

2.2.3 Use-case specification: Manage users



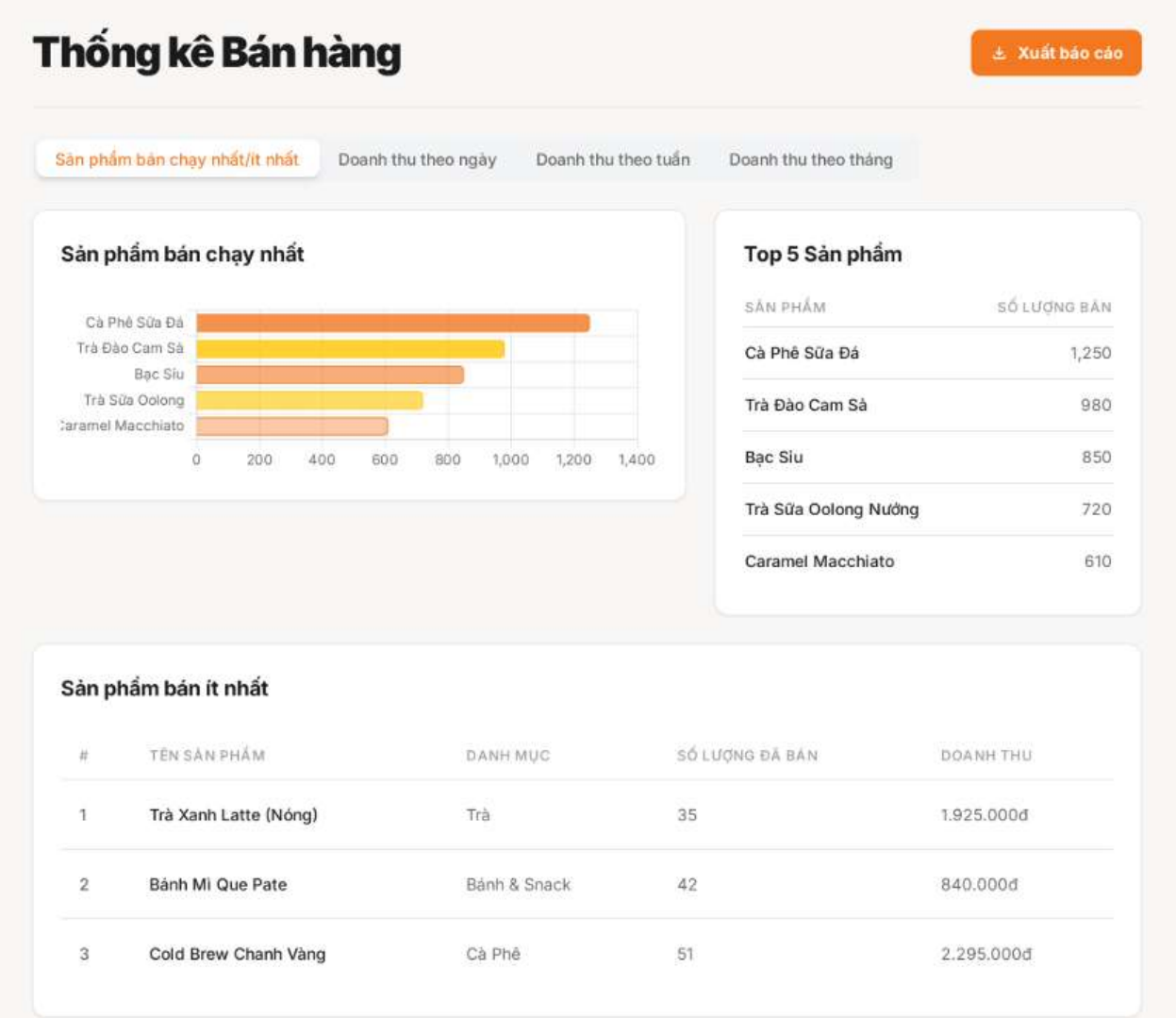
Use case ID	UCA3
Use case Name	Manage users
Brief description	Admin can view and manage user accounts, including personal details, credit points, and purchase history.
Actors	Admin
Pre-conditions	Admin is logged in and on the Admin Dashboard.
Basic Flow	1. The Admin navigates to the "Manage Users" section. 2. The system displays a list of all user accounts with key details. 3. The Admin can search or filter the list. 4. The Admin selects a user account to view/edit. 5. The Admin modifies the user's details (e.g., Credit Points) and clicks "Save." 6. The system updates the user record.
Alternative Flows	5a. Ban/Deactivate User: Admin selects the option to deactivate the user's account. The system prompts for confirmation and disables the account.
Post-conditions	The user account information is viewed or updated.
Non-functional Requirements	**Privacy**: Customer personal data must be kept confidential and managed securely.

2.2.4 Use-case specification: Manage orders



Use case ID	UCA4
Use case Name	Manage orders
Brief description	Displays all customer orders and allows the Admin to manually update the Order Status.
Actors	Admin
Pre-conditions	Admin is logged in and on the Admin Dashboard.
Basic Flow	1. The Admin navigates to the "Manage Orders" section. 2. The system displays a list of all orders with details (Timestamp, Products, Customer Info, Order Status). 3. The Admin filters or sorts the list (e.g., by status 'Processing'). 4. The Admin selects an order. 5. The Admin manually updates the Order Status (e.g., from 'Processing' to 'Done') and clicks "Update Status." 6. The system updates the order status in the database.
Alternative Flows	5a. Cancel Order: Admin selects the option to cancel the order. The system prompts for confirmation and marks the order as 'Canceled'.
Post-conditions	The selected order's status is updated.
Non-functional Requirements	**Data Integrity**: All order information must be maintained correctly.

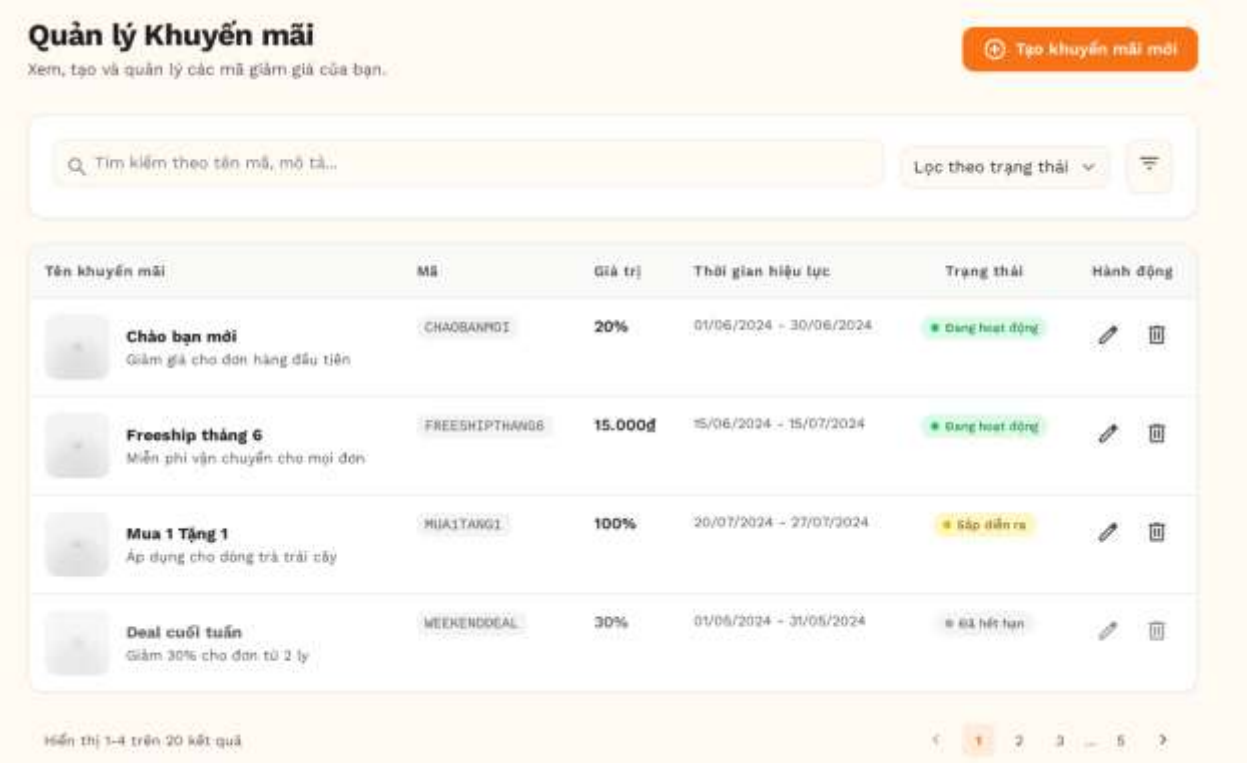
2.2.5 Use-case specification: View sales statistics (time - based)

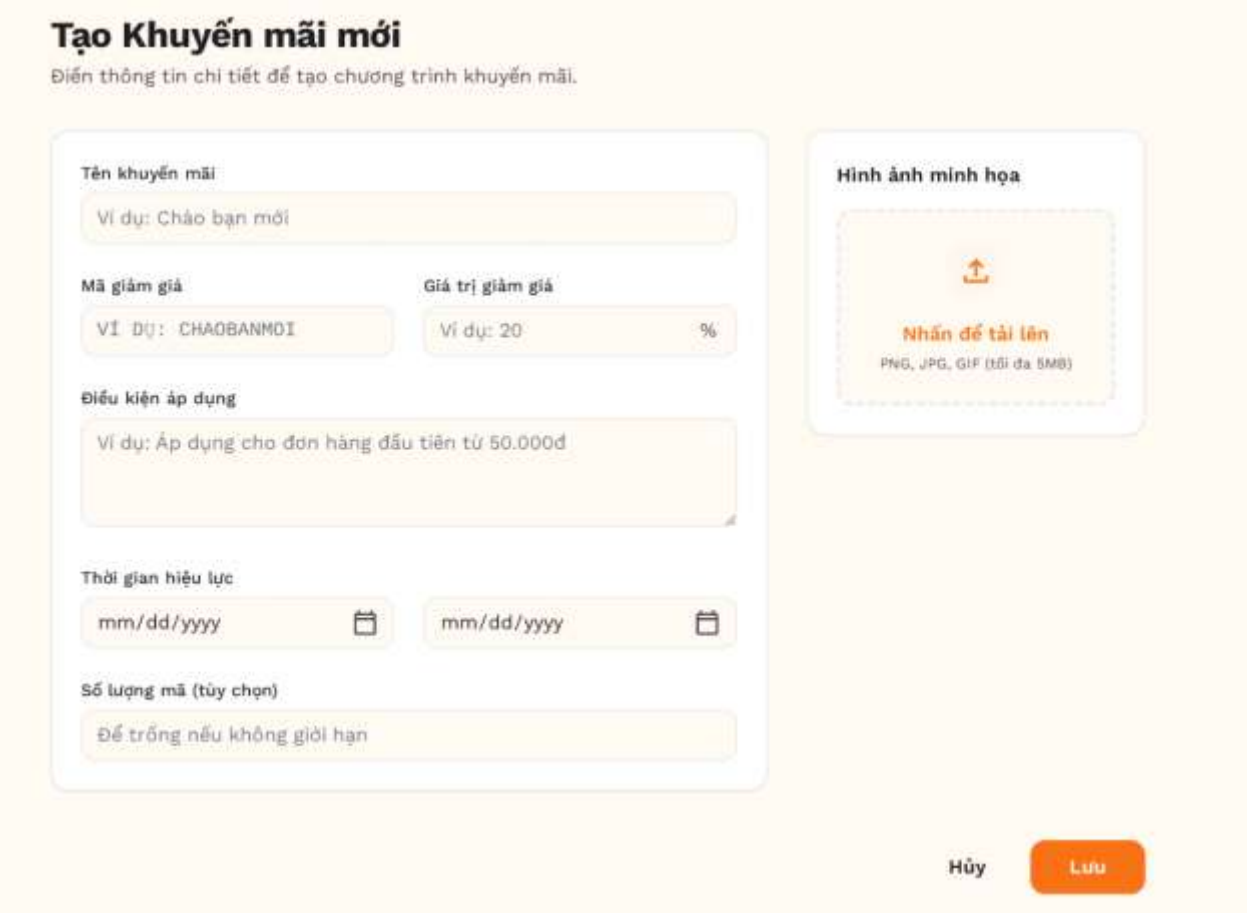


Use case ID	UCA5
Use case Name	View sales statistics (time - based)
Brief description	Generates analytics on income and sales over specific time ranges (daily, weekly, monthly), including visual charts.
Actors	Admin
Pre-conditions	Admin is logged in and on the Admin Dashboard.
Basic Flow	1. The Admin navigates to the "Sales Statistics" section. 2. The Admin selects a time range (e.g., 'Weekly') and a metric (e.g., 'Income'). 3. The system queries the database for historical order and income data for the selected range.

	4. The system generates and displays the analytics, including visual charts.
Alternative Flows	None.
Post-conditions	The Admin is viewing the sales and income analytics for the selected time range.
Non-functional Requirements	**Performance**: Statistical reports should be generated within 5 seconds.

2.2.6 Use-case specification: Manage promotions / discounts





Use case ID	UCA7
Use case Name	Manage promotions / discounts
Brief description	Admin can create or edit promotional campaigns, such as discount codes, percentage-based sales, or loyalty bonuses.
Actors	Admin
Pre-conditions	1. Admin is logged in and on the Admin Dashboard.
Basic Flow	1. The Admin navigates to the "Manage Promotions" section. 2. The system displays a list of active and past promotions. 3. (Create) The Admin clicks "Create New Promotion," fills in details (code, type, percentage/amount, expiry date, usage limit). 4. The Admin clicks "Save." 5. The system validates and saves the new promotion.
Alternative Flows	3a. Edit Promotion: Admin selects an existing promotion, modifies its details, and saves the changes. 4a. Deactivate Promotion: Admin selects an active promotion and clicks "Deactivate."
Post-conditions	The system has a new or updated promotional campaign.

Non-functional Requirements	**Data Integrity** is essential for all configuration data.

©<Newbie Coder>, 2025