



IT 309 SOFTWARE ENGINEERING

PROJECT DOCUMENTATION

Second Submission (b)

“Smell of East and West Shop”

Created by:

Arman Ćulah

Edina Kurto

Proposed to:

Nermina Durmić, Assist. Prof. Dr.

Lejla Brešćić, Laboratory Assistant

Ajla Korman, Laboratory Assistant

TABLE OF CONTENTS

Submission A.....	4
1. Introduction.....	4
1.1. About the Project.....	4
GitHub Link: www.github.com/armanculah/SEProject.....	4
2. High-level Plan.....	5
2.1.Goal-Oriented Roadmap.....	5
2.2. Release Plan.....	5
3. Functional and Nonfunctional Requirements.....	6
3.1. Functional Requirements.....	6
3.2. Non-Functional Requirements.....	11
4. User Stories.....	12
4.1. Smell of East and West Customer - User Stories.....	12
4.2. Smell of East and West Admin - User Stories.....	20
5. UML Diagrams.....	23
5.1. Activity Diagram.....	23
5.2. Class Diagram.....	27
5.3. Sequence Diagram.....	28
Submission B.....	33
6.Project Structure.....	33
6.1. Project Structure.....	33
6.2. Frontend Technologies.....	34
6.2.1. Frontend Technologies.....	34
6.2.2. Architectural Pattern.....	35
6.3. Backend Technologies.....	36
6.3.1. Backend Technologies.....	36
6.3.2. Database Entities.....	37
6.3.3. Design Patterns.....	38
6.3.3.1. DAO.....	38
6.3.3.2. Front controller pattern.....	38
7. Project Functionalities.....	39
7.1. Sign In.....	39
7.2. Sign Up.....	39
7.3. Dashboard.....	40
7.4. Explore Page.....	40
7.5. User Profile.....	41
7.6. Cart.....	41
7.7. Checkout.....	42
7.8. Admin Dashboard.....	43
6. Individual Contributions.....	44

Submission A

1. Introduction

1.1. About the Project

Smell of East and West is an e-commerce platform designed to provide a seamless and secure shopping experience for users. Whether you're browsing for products, managing orders, or making secure payments, Smell of East and West ensures efficiency, convenience, and reliability.

The platform includes all core e-commerce functionalities, such as user authentication, category-based filtering, shopping cart management, checkout, and order tracking. It also features admin tools for inventory management, sales reports, and user analytics, making it a comprehensive solution for both customers and admins.

With secure payment processing, real-time notifications, and an intuitive UI, Smell of East and West delivers a modern and user-friendly shopping experience, ensuring that customers can effortlessly find and purchase their desired products.

GitHub Link: www.github.com/armanculah/SEProject

2. High-level Plan

2.1.Goal-Oriented Roadmap

Date	May 2025	June 2025
Name	"Smell of East and West" Basic (v1.0)	"Smell of East and West" Advanced (v2.0)
Goal	Deliver a prototype of the online perfume shop with frontend done to collect early feedback and monitor user interactions and behaviour.	Deploy a functional web shop with backend integration, essential features upgraded, role based access control, and security advancements implemented.
Features	<ul style="list-style-type: none"> - Static SPA frontend using HTML/CSS/JS with Bootstrap - Navigation bar, footer, search, product catalogue, and detail pages - Profile update forms, cart management, checkout (UI) only - Basic database schema design and DAO layer setup (CRUD for main entities) - Postman API integration. 	<ul style="list-style-type: none"> - Full backend integration with FlightPHP REST API - JWT authentication and role-based access (user/admin) - Admin dashboard: order management, statistics, edit products - Middleware: validation, error handling, security (JWT, roles) - Deployment of app with hosted DB and API docs (Swagger) - Frontend refactor with MVC + CI/CD
Metrics	<p>All main UI pages is completed and navigable without page reload.</p> <p>Registration form and login form validate user inputs</p> <p>At least 5 products are displayed in the catalogue with working filter</p> <p>DAO layer is implemented for 5 entities with working CRUD operations</p>	<p>User login generates a valid JWT token, and access must be restricted based on roles</p> <p>Users are able to place an order and edit it while in "pending" status.</p> <p>Admin dashboard displays and allow management of orders</p> <p>Application is deployed</p>

2.2. Release Plan

Name of the Release	GO Plan Features	User Stories
Smell of East and West - Basic (v1.0)	- Static SPA frontend using HTML/CSS/JS with Bootstrap	Register User Account
	- Navigation bar, footer, search, product catalogue, and detail pages	Navigation Bar
	- Profile update forms, cart management, checkout (UI only)	Footer
	- Basic database schema design and DAO layer setup (CRUD for main entities)	Search Bar
		Filter Explore Page
		Product Catalogue
		Product Detail Overview
		User Dashboard
		Profile Update Information
		Profile Update Picture
		Delete User Profile
		Cart Management - Add item
		Cart Management - Edit items
	Checkout Management - Information	
Responsive Layout		
Contact Support Form		
Checkout Management - Payment (UI only)		
Smell of East and West - Advanced (v2.0)	- Full backend integration with FlightPHP REST API	Login User
	- JWT authentication and role-based access (user/admin)	Logout
	- Admin dashboard: order management, statistics, edit products	Register Admin
	- Middleware: validation, error handling, security (JWT, roles)	View Order Details
	- Deployment of app with hosted DB and API docs (Swagger)	Cancel Order Shippment
	- Frontend refactor with MVC + CI/CD	Admin Dashboard
		Admin Order Management
		Admin Statistic Overview
	Edit Product Details	

3. Functional and Nonfunctional Requirements

3.1. Functional Requirements

Feature name: Registration

Role(s): Customers, Admin

Feature description:

1. A screen with the registration form, where the user will input their own credentials.
2. Users will enter their email address, full legal name and password.
3. The system validates the email format and password strength.
4. Email with the confirmation link will be sent to the entered email address for confirmation.
5. The user is redirected to the login page and can now access their unique account.
6. After the login process is finished the user will be redirected to the dashboard where they can start shopping for a range of perfumes.

Dependencies:

All input fields must be filled in correctly; otherwise, registration will not proceed. Email format validation must be applied on both frontend and backend (user@gmail.com). Passwords must be at least 8 characters long, with uppercase letters, lowercase letters, and numbers.

Feature name: Login

Role(s): All previously registered users

Feature description:

1. A login screen is presented where the user enters their email address and password.
2. The system checks if the email exists in the database.
3. The system verifies the password by comparing it with the stored hashed password.
4. If the email address or password is incorrect, a message will appear stating, "Invalid email or password."
5. If the credentials are correct, the user will be directed to the dashboard page.

Dependencies: Only verified users can log in. Email format validation is required. Failed login attempts are logged for security monitoring. The system should prevent brute-force attacks by limiting login attempts.

Feature name: Dashboard

Role(s): Customers

Feature Description:

1. Users access the dashboard automatically upon opening the website.
2. The system dynamically displays a visually engaging grid layout of all products, with buttons to view more and add to cart.
3. Users can scroll through an infinite scrolling layout or pagination for more products.
4. Each product card displays: product image, name and price.
5. Users can quickly add an item to their cart directly from the dashboard.
6. Clicking on a product redirects to the single product page for more information.
7. Users can only proceed to shop perfumes only after registration / login, to ensure security.

Dependencies: Dashboard is implemented in a way to ensure all products are visible to view and search, but no product can be added to cart or bought before the user completes the registration process at least once. Different layouts for customers and admins must be implemented, since they need different functionalities.

Feature name: Search Functionality

Role(s): Customers (primary), Admin

Feature Description

1. Users enter a keyword (name of the product) in the search bar.
2. The system retrieves matching products to the description.
3. Users can apply filters (category, price, availability).
4. Users can sort results by relevance, price, or rating.

Dependencies:

Search queries must be optimized for performance. If there are no products with the given description the page will show a message “No product found based on your search history. Try again.”.

Feature name: Admin Panel**Role(s):** Admin**Feature Description:**

1. Admins can log in using a specified email and access the admin panel.
2. Admins can view, edit, delete, and add new products.
3. Admins can manage user accounts (activate/deactivate users).
4. Admins can view order statistics and sales reports.
5. Admins can track and update order statuses.

Dependencies: Only verified admins can access the panel. All admin actions should be logged for security.

Feature name: Shopping Cart**Role(s):** Customers**Feature Description:**

1. Users can add items to the shopping cart from the product page.
2. When adding an item, the system checks stock availability before confirming.
3. Users can view their cart, which displays product details (name, quantity, and price).
4. Users can increase, decrease, or remove items from the cart.
5. If users increase quantity beyond available stock, an error message appears.
6. The system updates the total price dynamically as changes occur.
7. Users can proceed to checkout when ready.

Dependencies:

Stock must be checked before adding/updating items. Cart data should persist across sessions for logged-in users. For non-logged-in users, cart data should be stored temporarily using session storage.

Feature name: Checkout

Role(s): Customers

Feature Description:

1. Users click the Button "Checkout."
2. Users enter their personal information (full name, email, phone).
3. Users enter or select a saved shipping address.
4. Users choose a payment method (Credit Card, Cash on Delivery), then proceed to press the "Submit Payment" button.
5. The system verifies the entered payment details.
6. If using a credit card, the system processes the payment via a third-party gateway.
7. The system confirms the order and deducts the stock from the inventory.
8. The user receives an order confirmation email with tracking details.

Dependencies:

Payment must be successfully verified before completing the order. The order cannot be placed if the cart is empty. For credit card payments, a secure third-party gateway must be used.

Feature name: Explore Page

Role(s): Customers

Feature Description:

1. Users browse products without searching.
2. The system displays trending, best-selling, and recommended items.
3. Users can click on a given filter to categorize the perfume products.

Dependencies: Recommendations should be personalized based on user activity.

Feature name: Profile Management

Role(s): Customers, Admin

Feature Description:

1. Users navigate to their profile settings.
2. Users can update their: full legal name, email, shipping address, password...
3. Users have an option to update their profile picture.
4. The system saves changes and updates the user record.
5. If the email was changed, a new verification email is sent.

Dependencies: Full names must contain only letters. Email changes require re-verification. Password changes require the current password for security.

Feature name: Single Product Page

Role(s): Customers

Feature Description/Steps:

1. Users view product details, including images, price, stock, and reviews.
2. Users can select quantity and add to cart.
3. Users can leave a review after purchasing.
4. Related products are shown below.

Dependencies:

Users can only leave reviews if they have purchased the item. Stock availability must be updated dynamically.

3.2. Non-Functional Requirements

1. The system must comply with local shopping data security standards to safeguard and maintain the integrity of information.
2. The system should respond to requests within 5 seconds.
3. The system must accommodate at least 100 users at the same time without a decline in performance.
4. Availability should be ensured 24 hours a day, 7 days a week, with minimal planned downtime.
5. User data must be stored in a database and transmitted securely using industry standard encryption.
6. Regular automated backups of data should be conducted.
7. The system should provide real-time or near real-time data retrieval and update capabilities.
8. The system should utilize hardware resources (CPU, memory, storage) efficiently to optimize performance.
9. The system must be scalable to accommodate growth in data and user load as the perfume shop expands.
10. It should accommodate support for modular design for the easy addition of new modules and integrations.

4. User Stories

4.1. Smell of East and West Customer - User Stories

1. Register User Account

As a visitor I want to be able to register my account so that I can access the Smell of East and West shop and all functionalities.

Acceptance criteria:

When a visitor user clicks on the “Register” button located on the login page, they are redirected to the user dashboard page after they enter their information.

Register user account page will have the following inputs one below another:

- Username - textfield
- Email – textfield
- Password – textfield
- Repeat Password – textfield

Input Validation:

All inputs are required, meaning that email must be a valid email address (user@example.com). Password and repeat password must be the same. Button “Register” is below all the inputs and clicking on it submits the entered data which creates a record in our database.

In case the record cannot be created in the database, an alert message will be displayed with appropriate description or text “Oops. Something went wrong...”, if it can be created the user is redirected to the dashboard upon successful registration.

2. Login User

As a registered user, I want to be able to login so that I can have access to the website.

Acceptance criteria:

Once the user clicks on the “Login” button on the dashboard, they are taken to the login page that is primarily composed of a form with the following inputs: email and password. If the credentials from the login page are confirmed with the data about the user in the database, the user is redirected to the user dashboard page. If the credentials entered do not exist in the database, the validation message is “Wrong email or password.” In case the database is down upon trying to submit data, an alert message will be displayed with text “Oops. Something went wrong.”

3. Logout

As a registered user, I want to be able to logout so that I can stop using the webpage.

Acceptance criteria:

Upon clicking on the primary profile page the button “Logout”, the user is redirected to the login page. At this point this user is no longer an active user in the database and cannot access the Smell of East and West platform without re-entering their user credentials. In case the database is down upon trying to logout, an alert message will be displayed with text “Oops. Something went wrong.”.

4. User Dashboard

As a registered or guest user, I want to access the dashboard so that I can browse, search for, and purchase fragrances easily.

Acceptance criteria:

The top navigation bar includes all pages that the users can view freely, if the user is already registered they have an option to buy products as well. Upon navigating to the dashboard, users can view available fragrance products in a grid layout. On the bottom of the page the user has footer navigation as well that can lead them to desired pages in our website.

Users should see products that contain images, names, pricing, and an "Add to Cart" button. Clicking on this button leads the user to their personal cart where they can view, add, delete products from their inventory.

If there are any issues loading products, the user will be shown an error message prompting them to refresh the page or try to load later.

5. Navigation Bar

As a guest, registered user or admin, I want to be able to navigate between different pages so that I can access different features Smell of East and West has to offer.

Acceptance criteria:

On the left side of the navigational toolbar there will be the Smell of East and West logo. The Navigation toolbar on the desktop app will be on the top : Home, Search, Explore, Account and Cart and login button. The admin panel will have the same navigation toolbar showcasing the same links for easier management.

6. Search Bar

As a user, I want to search for products easily so that I can find what I need without much effort.

Acceptance criteria:

The search bar is located in the header right beside the navigation bar. Users can enter keywords or names related to a product they are searching for.

The system retrieves matching products dynamically. Search results update in real time based on filters applied. Users can filter search results by: for men, women, on sale, and different perfume categories. If there is no product with the same name or description the page will show a message “No products with this description were found. Please try again.”.

7. Footer

As a user, I want to access links and information in the footer so that I can easily navigate the website.

Acceptance criteria:

The footer is displayed at the bottom of the webpage. It contains the following sections: Products, Useful links (Your Account, Our Ingredients, Shipping Rates, Help) and finally the Contact section showcasing the phone number, email address and location of our company.

When a user clicks a link, they are redirected to the appropriate page. The footer should be responsive and properly formatted on all devices (using bootstrap code). If there are any issues loading the pages, the user will be shown an error message prompting them to refresh the page or try to load later.

8. Profile Update Information

As a registered user, I want to update my profile so that I can manage my private/public information.

Acceptance criteria:

On the top screen the profile page includes user information and picture. Under that there is the “Settings” and “Logout” button.

In the Profile Settings form there is an option to change:

- Full name
- Email Address
- Shipping Address
- Password

Below the input fields there is a "Save" button to submit updates and a “X” on top of the popup to cancel any changes made. Users must be logged in to access and update their profile. After successful submission, updates are saved in the database and immediately reflected on the user’s profile page.

If the email address is changed, a confirmation email is sent to the new address. The user’s email is updated only after confirming the change via a link in the email.

9. Profile Update Picture

As a registered user, I want to be able to change my profile picture so that I can personalize my profile to my personal liking.

Acceptance criteria:

The profile page shows a current user picture. Right below the picture, there is a "Change Picture" button indicating that the user can click on it. Clicking the button allows the user to upload a new image. This can be achieved via a small popup box where the user can search or drop a new image. The new picture is shown immediately after submission of this information box.

10. Delete User Profile

As a user, I want to delete my profile so that I can permanently leave the platform if I so wish.

Acceptance criteria:

The user can search up the delete option after entering the profile page. A "Delete Account" button is found on the bottom of the personal information, next to the "Edit Profile" button. When clicked, the system shows a confirmation message asking for the user's current password, this being done this way to ensure user security and prevent the customer from accidentally deleting their profile (especially if their orders are still being shipped). If the password is entered and confirmed, the account is deleted. A goodbye message is shown and the user is redirected to the home page

11. Filter Explore Page

As a user, I want to use filters on the explore page so that I can browse products by category or price.

Acceptance criteria:

After the user enters the webpage, they can either browse recommended products on the dashboard or go to the explore page to automatically filter the perfume products based on their needs. Below the search bar there is a filter system that after clicking a certain filter category, updates the product list dynamically. Filter options include: for men, women, on sale, and different perfume categories and even prices from lowest to highest (as well as highest to lowest). These products refresh on screen without page reload.

12. Cart Management - Add item

As a user, I want to add items to my cart so that I can manage my shopping before the checkout process.

Acceptance criteria:

The users can pick and choose a variety of perfume products that can be added to their cart. Users can add products to their cart from the product page. In case the user decides to want to buy more than one of the products they can click on the number of products on the right side of the screen. By doing so, the total price will automatically change to the current (new) total price. If the user increases the quantity beyond stock availability, an error message appears.

13. Cart Management - Edit items

As a user, I want to add or remove an item from my cart so that I can update my shopping selection to the desired total price amount.

Acceptance criteria:

After choosing all products the user wants to buy they will open the cart in their navigation bar. If by chance, the user is unsatisfied with the total price or they choose an item they do not want anymore they will have an option to delete the item from their cart. Cart page shows an 'X' button next to each item. Clicking the button removes the product from the cart and the total price is updated immediately. Users can also clear the entire cart if needed and proceed to checkout when ready.

14. Checkout Management - Information

As a registered user, I want to enter my shipping and contact information so that my order can be delivered correctly and at the right address.

Acceptance criteria:

After completing the cart process the website will next show the checkout page contains form inputs: full legal name, email, shipping address. Checkout page is prefilled for registered users that provided their information beforehand, but they will still be able to edit their information. All fields in the checkout process are required and validated. A 'Continue to Payment' button appears after successful validation of all information.

15. Checkout Management - Payment

As a user, I want to pay for my order so that I can complete my purchase on this website.

Acceptance criteria:

Completing the checkout process, the website leads the customer to the payment page where the user is required to choose a payment method including credit card, paypal or even cash on delivery. After the user enters valid information including their legal name, credit card number and code, a loading message is shown while the system verifies payment and upon success, the user is shown an order confirmation page. If the customer chooses to pay by cash, the second party will receive and send the money after the transaction.

16. View Order Details

As a registered user, I want to view all my previous and shipped orders so that I can track my purchases efficiently.

Acceptance criteria:

Users can easily manage all their orders on their profile page, since the information is shown directly below their personal information. Each order shows: product name(s), total price, order date, and status (shipped, processed, pending..). Clicking an order expands additional shipment information, making sure that the amount of information doesn't overwhelm the customer by bombarding them with useless information. On the order details, the users personal information will be shown so they can make sure all their information was correctly written down and no further editing or action is needed. Below that, the products will be displayed (name, image. amount).

17. Cancel Order Shipment

As a user, I want to cancel orders that have not yet shipped so that I can manage my purchases.

Acceptance criteria:

On the profile page, all order information is displayed for the user to review and edit. In case that the user in view order sees that some information is incorrect they can either edit their information or delete the order entirely. In case the user decides to delete their order, on the side of order information there will be a "Cancel Order" button. Important note is that the orders with status "Pending" show this button since the order has not yet been shipped. Clicking the button opens a confirmation popup asking the user if they are sure they want to cancel the whole order. If this popup is confirmed, the order is cancelled and updated in the table on their profile (marked as "Canceled").

18. Product Catalog

As a registered user, I want to scroll through a product catalogue so that I can easily view all available perfumes.

Acceptance Criteria:

The product catalogue displays items in a grid layout on the explore page as well as on the dashboard page for practicality. Each product card shows: product name, image, price, and add to cart button under the information that redirects the user to the product detail page. Products load dynamically and respond to filtering or sorting. This catalog is used for better view of all available products, from mens to womens perfumes, as well as managing products that are not in stock to be removed from view. If a product is currently unavailable it will not be shown in the catalog.

19. Product Detail Overview

As a user, I want to view full product information so that I can decide if I want it before buying.

Acceptance criteria:

On the dashboard and explore page there are product cards containing information about the product like the name and price. Clicking on the button “View product” opens the product detail page. On this page more information is stored providing users with more details of what product they are buying. A detailed description will be shown under the product image and price, containing all needed information from what ingredients are in the perfume to the perfume bottle size. This page is highly important to contain only crucial information for the user, since this is where they can decide if they want to purchase this product or not, meaning that the page has to be innovative and reliable.

20. Contact Support Form

As a user, I want to be able to contact customer support so that I can report issues or ask questions to the organisation.

Acceptance Criteria:

By clicking the “Contact Us” button that is available in the footer of the webpage, a chat box like form pops up that includes the fields Name, Email, Subject, Message and all fields are required. Upon submission, a confirmation message appears, and the message is saved/sent.

4.2. Smell of East and West Admin - User Stories

21. Register Admin

As a future admin user, I want to create an admin account so that I can gain access to administrative features of the platform.

Acceptance criteria:

When a user clicks on the “Register” button located on the dashboard, they are redirected to the register page, which also supports admin account creation through additional input validation. All register inputs are the same except the admin access code that is hidden by default.

In case that a required field is left blank, an error message should appear, such as:

- "Full name is required."
- "Please provide a valid email address."
- "Password must meet the minimum security requirements."

Admin-Specific Functionality:

The "Admin Access Code" text field is hidden by default. The "Admin Access Code" field becomes visible only when the page is accessed with a specific query parameter in the URL (e.g., /register?admin=true). If the user provides the correct admin access code, the account will be created with admin privileges.

If the access code is incorrect or missing, an error message is displayed: “Invalid admin access code.” If an email is already associated with an existing account, there will be a display "This email address is already registered. Please log in or reset your password."

Admin is redirected to the home screen only after a successful registration.

22. Admin Dashboard

As an admin, I want to have a dedicated dashboard so that I can manage the shop efficiently.

Acceptance criteria:

Upon opening the website and after a successful registration, the admin will be redirected to the admin panel. The navigation bar is similar to the customers, allowing the admin to have access to the single product pages, all cart information and in stock products. The admin can directly from here view and manage information regarding the products, meaning they can edit, delete or even add new products.

The admin panel is designed in a way to provide the admin with all the necessary information, from statistics of all users, to order notifications, to how many products have been bought or even what products are currently out of stock. This panel helps the admin manage the website easily, showing everything they need to know on one single page (besides the more organized navigation bar with other links).

From here the admin can view users, control their orders and manage them directly (changing their order state to shipped, delivered...), as well as deleting or blocking users if necessary.

23. Admin Order Management

As an admin I want to be able to view all orders made, from completed orders to orders that are shipped and the ones I have not viewed yet.

Acceptance criteria:

In the admin dashboard all information regarding orders are stored. The admin has full access to view all orders that are made, shipped and completed. They manage the order status and customer information easily using the table system. The order section can be managed also by using filtered provided to categorize the orders by date, status or even user id. If some information is incorrect, using the customers provided information (name, phone, address, email), the admin can contact them to make sure there is no issue with the order. This can be helpful especially if the user has made one or more orders on the website before and the previous information does not match the current given one.

24. Admin Statistic Overview

As an admin, I want to view statistics on users and their orders so that I can track performance and engagement.

Acceptance criteria:

On the admin panel below the navigation bar the statistics segment shows charts and totals for sales, top products, and user activity. This information is crucial to the admins since they can easily track user engagement and activity on the webpage. Total revenue generated within a selected timeframe (daily, weekly, monthly, yearly) is provided on the left side of the webpage while the number of completed, pending, and canceled orders are shown on the right side. Under this information there is a ranked list of best-selling products based on total sales to help admins view what product is trending and need more in stock. Statistics segment is easy and productive to read as all information is displayed dynamically to fit the website's initial style. Data updates dynamically without requiring a full page reload.

25. Edit Product Details

As an admin, I want to be able to update product information so that listings stay up to date.

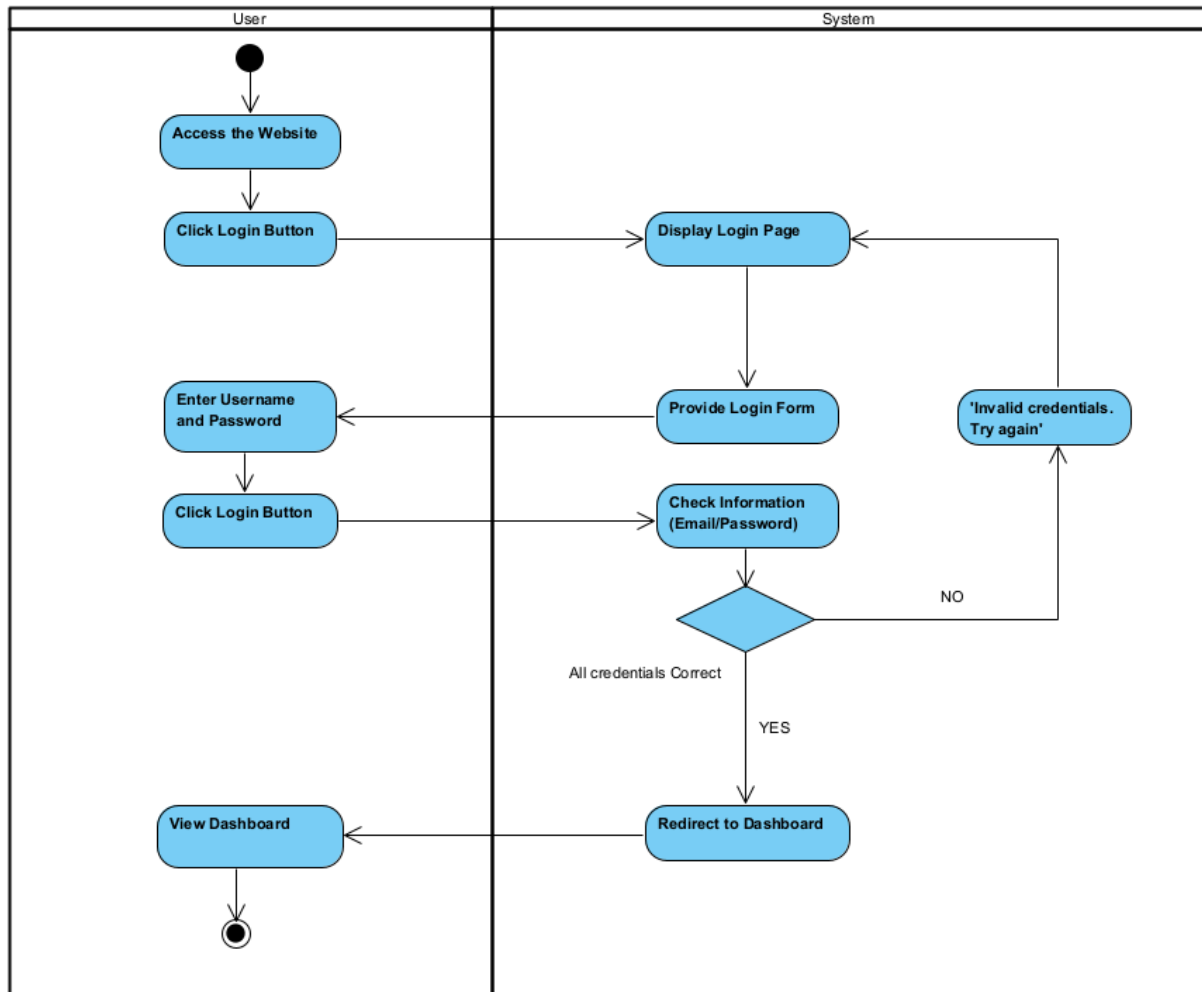
Acceptance criteria:

Admin can select a product from the product list in the admin panel as well as the explore page. After clicking the “Edit product” button the admin will be redirected to a special product page shown only for product editing. On this page the admin can edit product name, description, price, and category. Changes are saved instantly upon submission and updated product details reflect immediately in the product list and storefront. A “Save Changes” button confirms modifications before applying updates on the bottom of the page. A “Cancel” button allows admins to discard changes and return to the product list easily. After a successful update, the admin receives a confirmation message, and if an error occurs (e.g., invalid input, server issue), an error message is displayed.

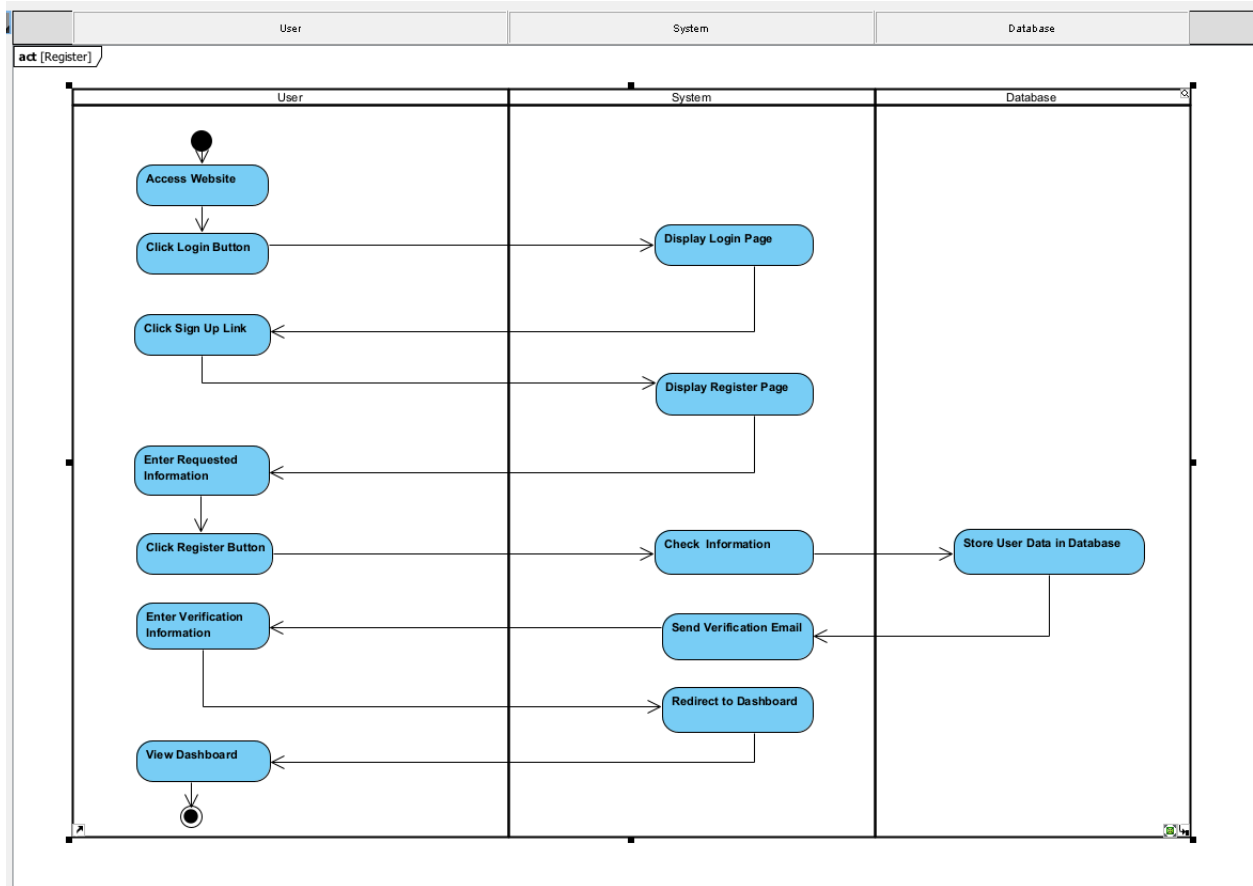
5. UML Diagrams

5.1. Activity Diagram

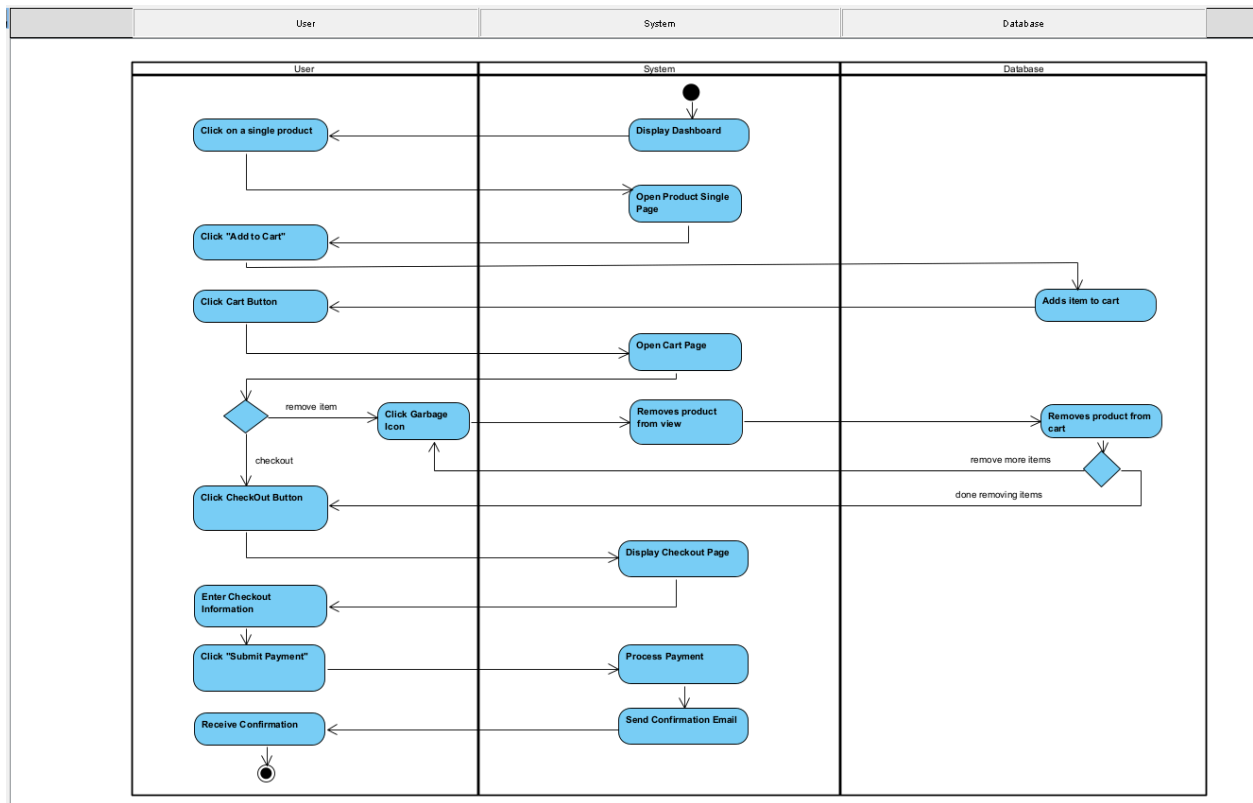
5.1.1 Login



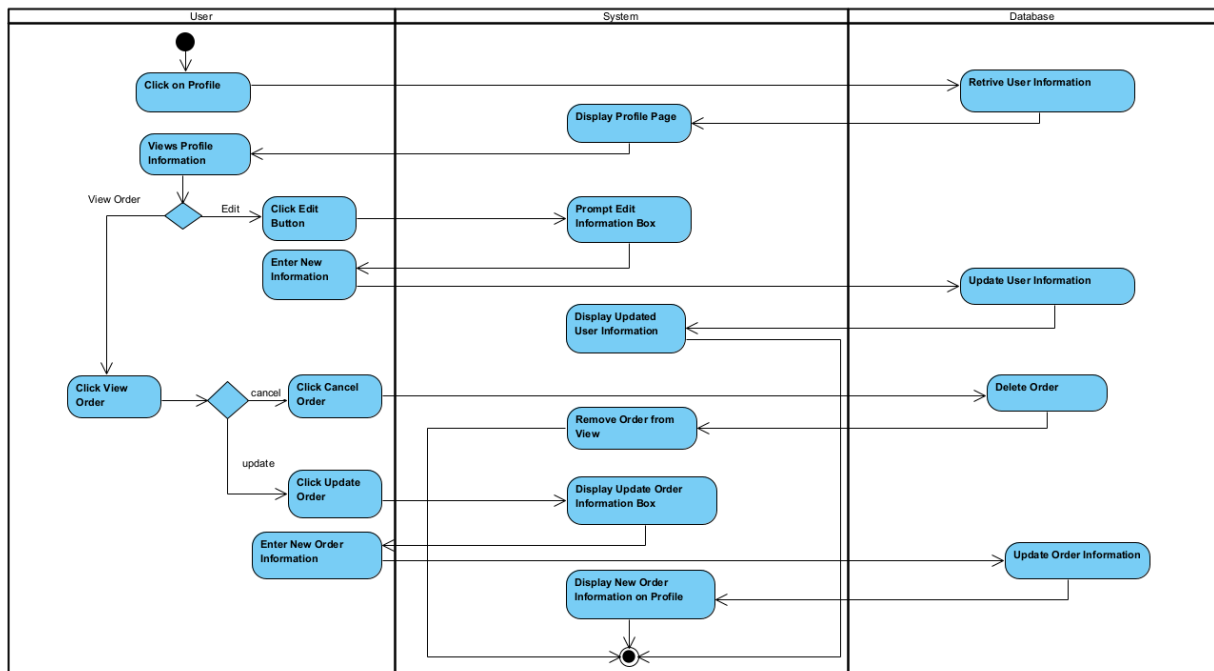
5.1.2 Register



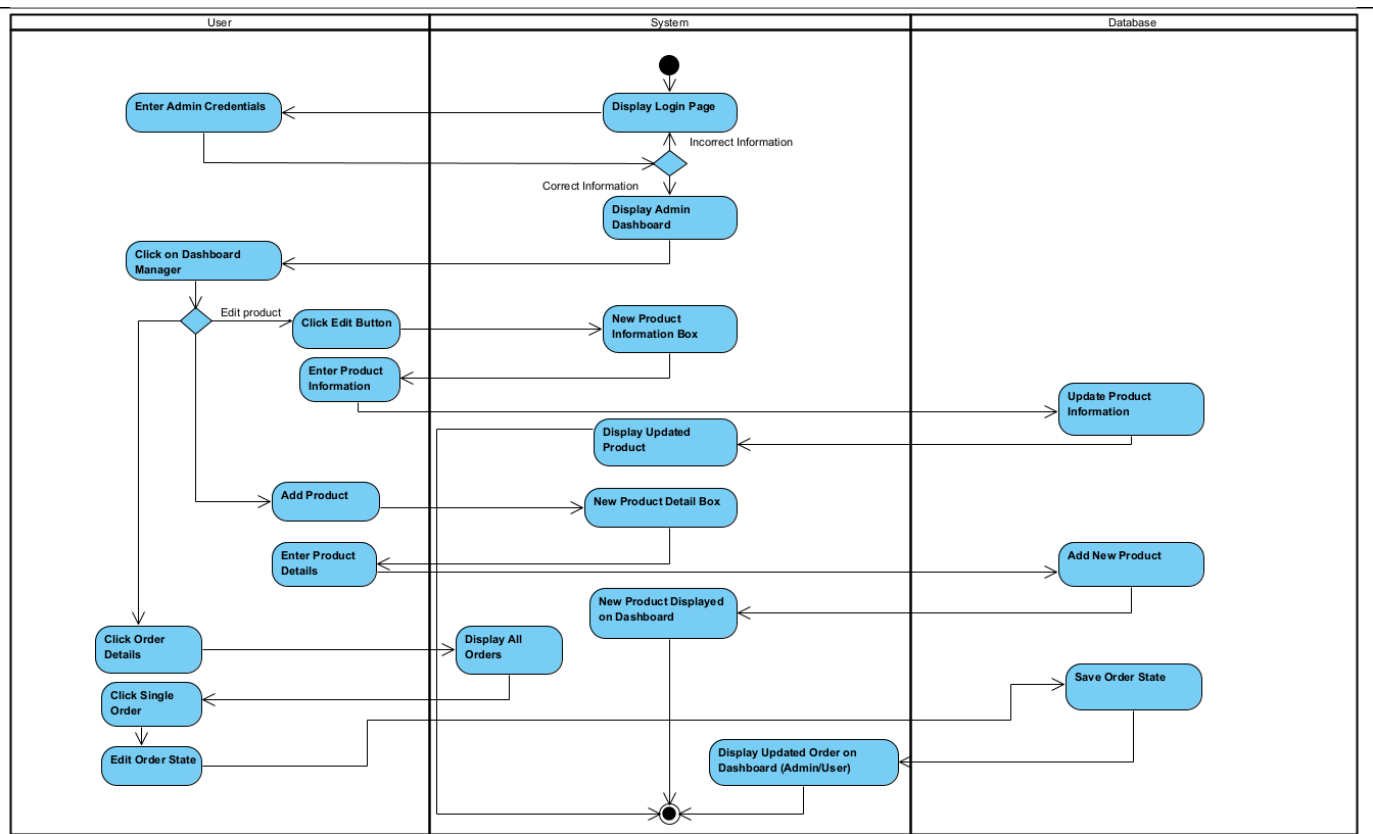
5.1.3 Cart



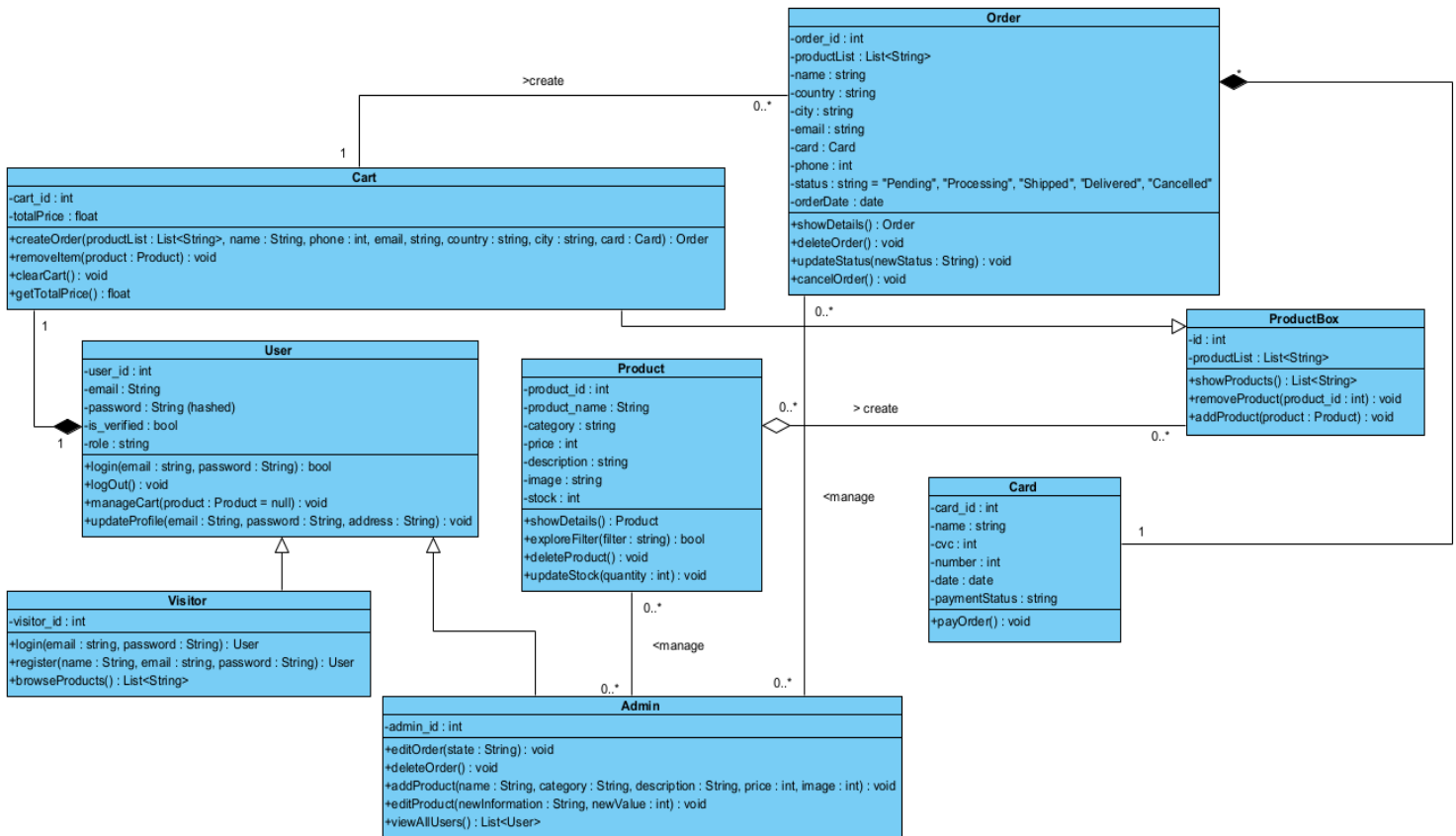
5.1.4 Profile



5.1.5 Admin Dashboard

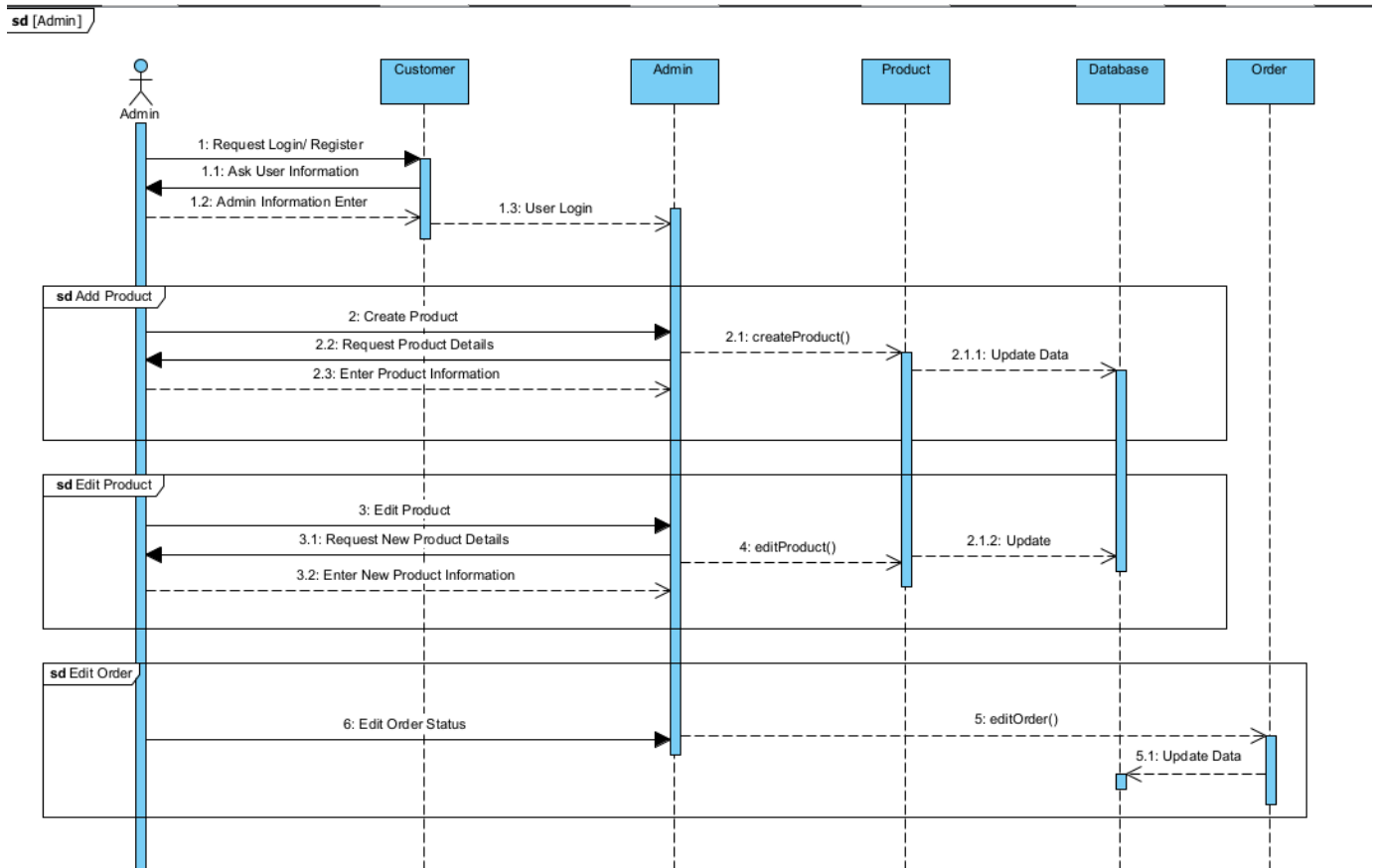


5.2. Class Diagram

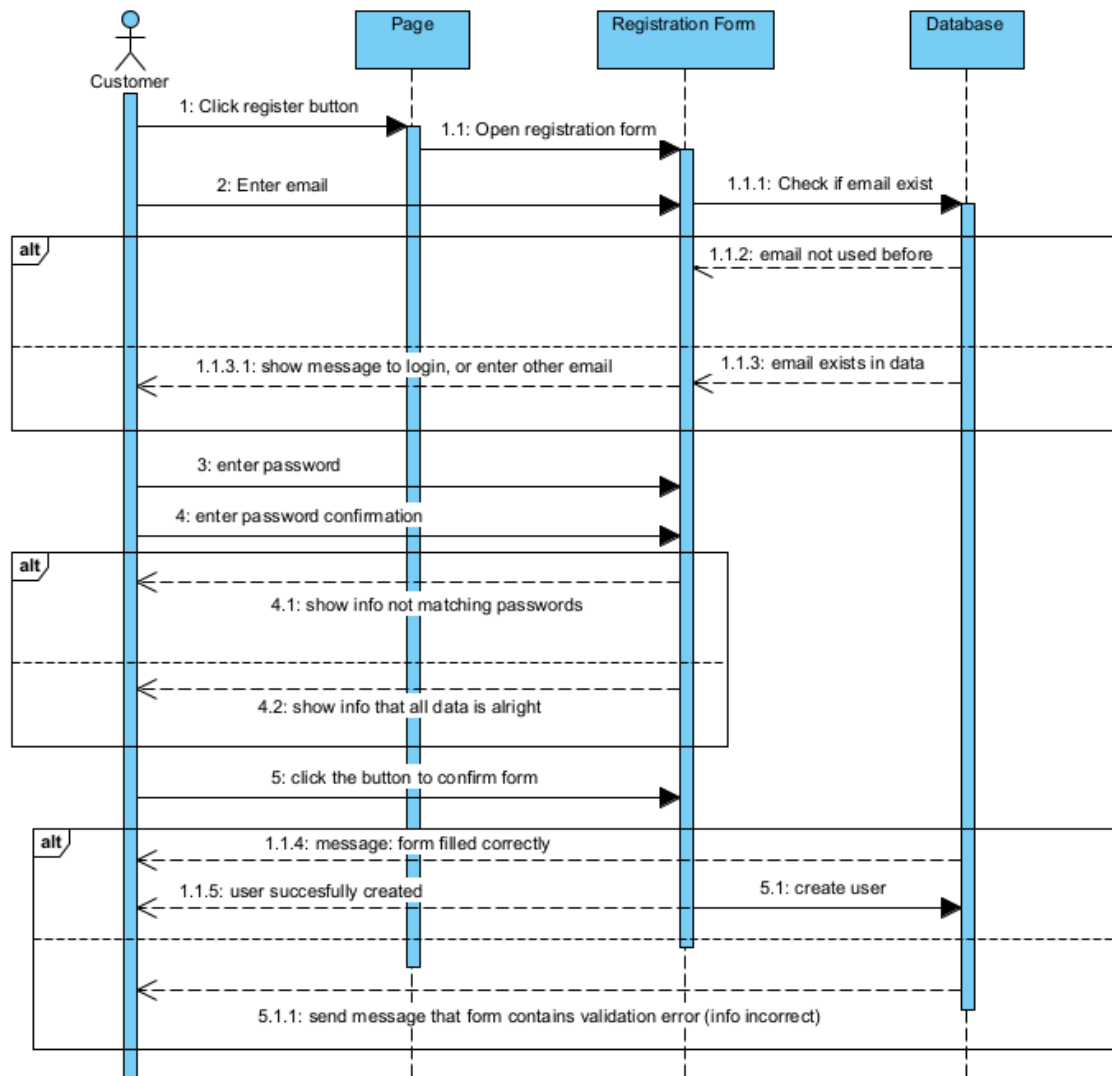


5.3. Sequence Diagram

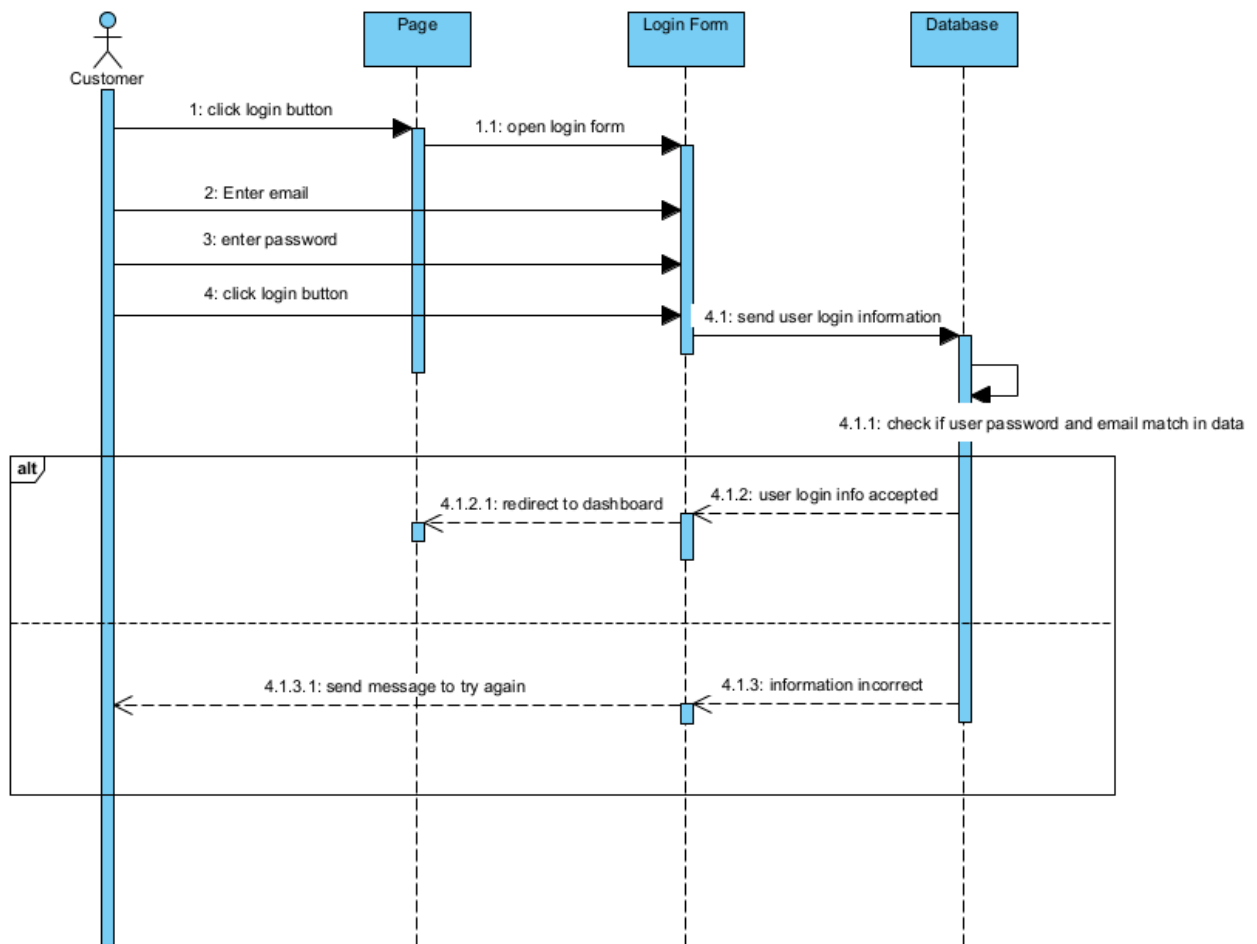
5.3.1 Admin Dashboard



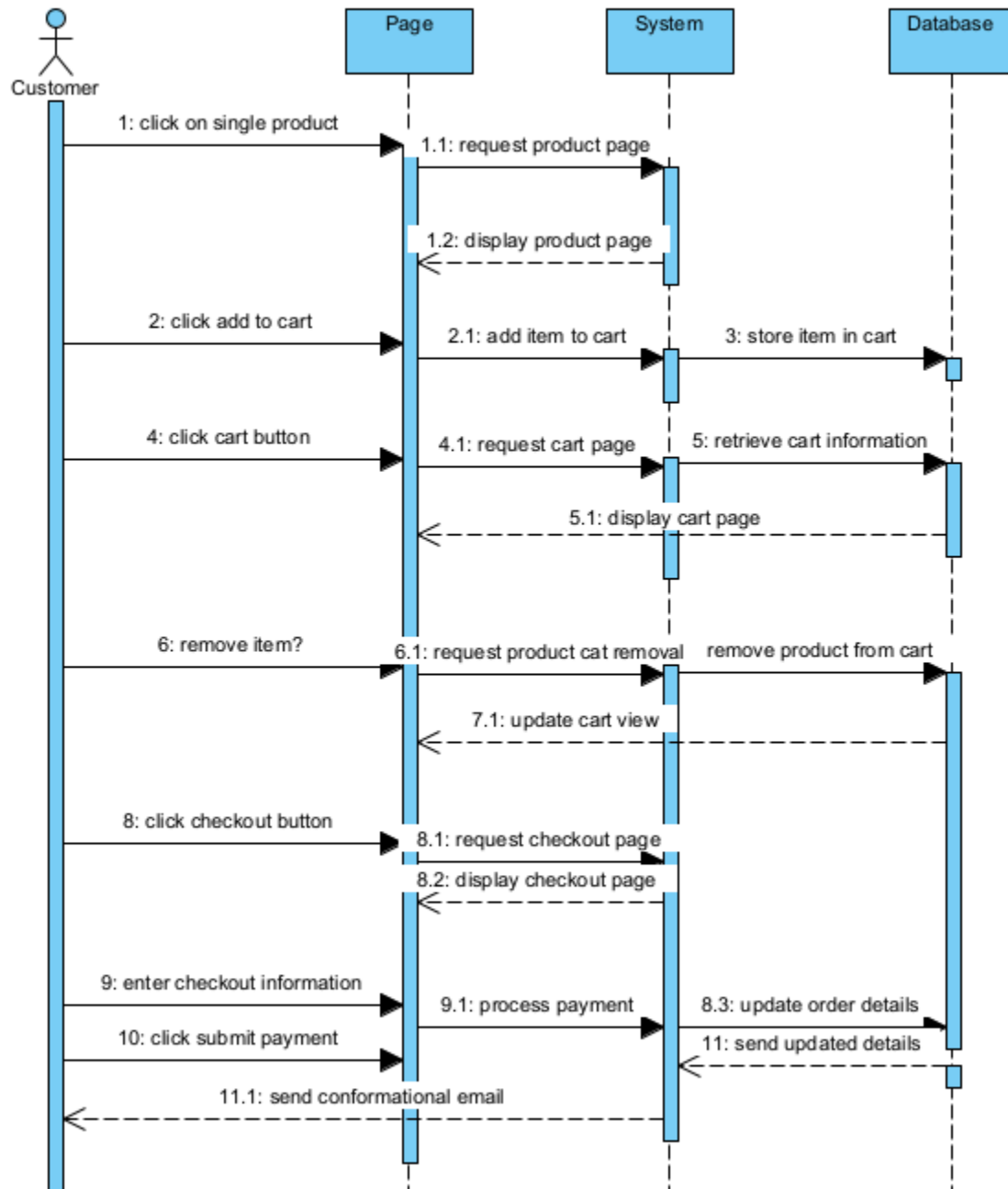
5.3.2 User Registration



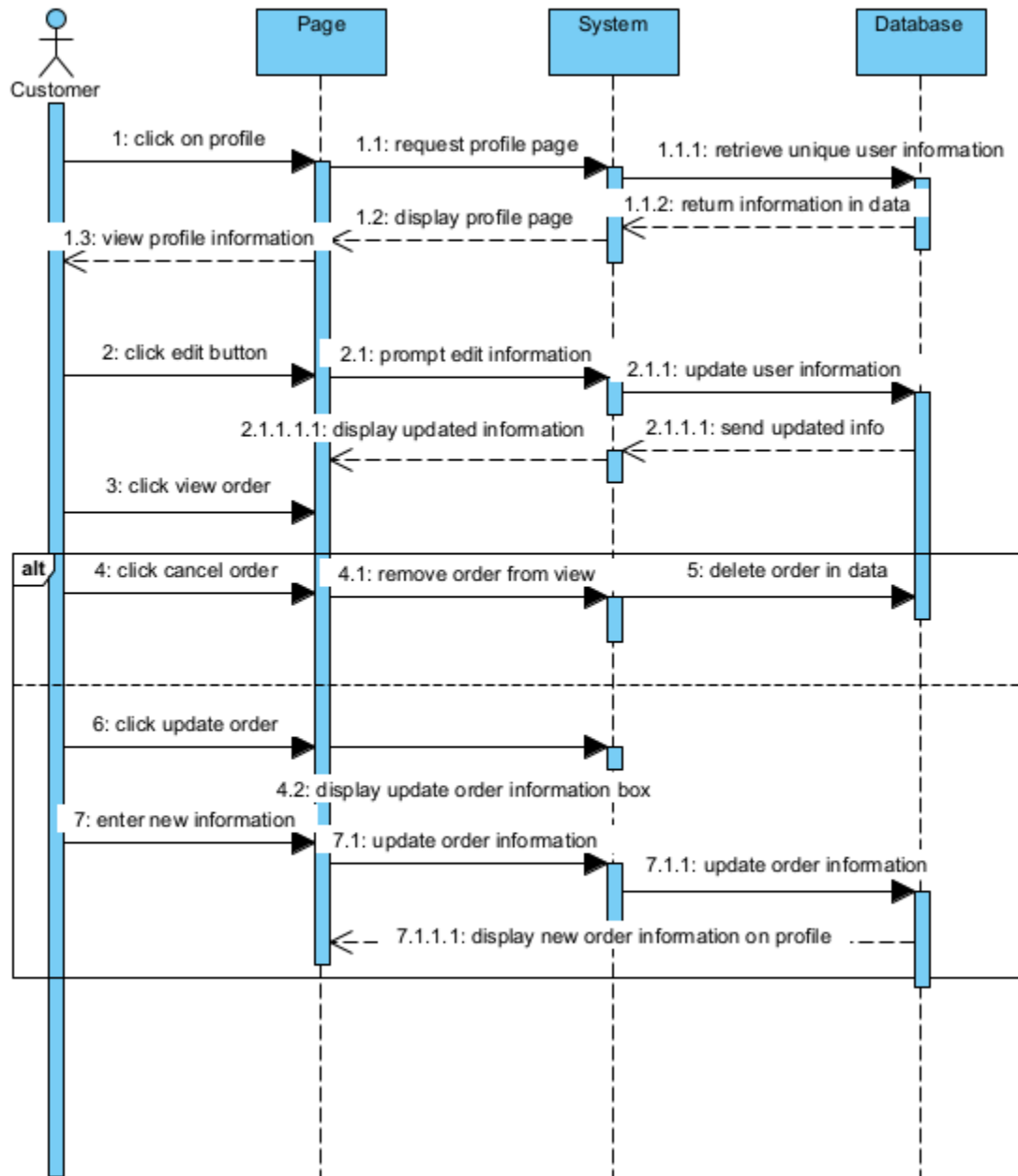
5.3.3 User Login



5.3.4 Cart



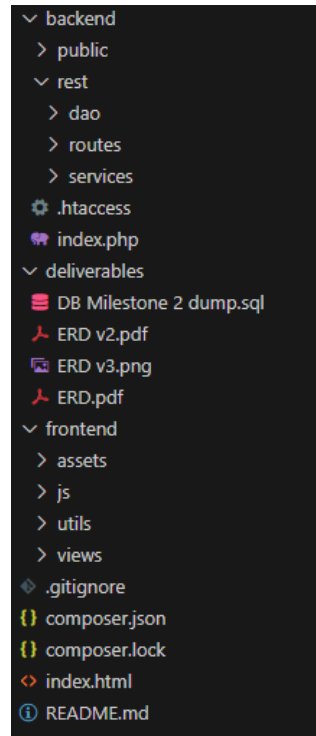
5.3.5 Profile



Submission B

6. Project Structure

6.1. Project Structure



Github structure

6.2. Frontend Technologies

6.2.1. Frontend Technologies

The frontend of the perfume shop is developed using HTML and CSS for structure and styling, ensuring a clean and user-friendly interface.

For interactivity, JavaScript is used along with several supporting tools:

- Bootstrap provides a responsive layout and modern UI components, allowing the site to look good across devices.
- SPAPP (Single Page Application by Amiletti) powers routing and view management in a SPA structure, allowing seamless navigation without full page reloads.
- jQuery simplifies DOM manipulation and AJAX requests, while the jQuery Validation plugin handles client-side form validation for actions like login, registration, and checkout.
- Toastr is used to display non-intrusive toast notifications (e.g., “Item added to cart”).
- Block UI visually blocks parts of the UI during asynchronous actions, preventing duplicate actions or user confusion.
- DataTables enables dynamic management of tabular data for users, orders, and products with features like pagination and search.

6.2.2. Architectural Pattern

Smell of East and West project implements a Layered Architecture pattern, a design approach that separates responsibilities into distinct layers like DAO, Services, and Routes which enhances modularity, maintainability, and scalability of the codebase.

- The DAO layer handles direct interaction with the database. For instance, it performs SQL queries to retrieve perfumes, insert customer orders, or update payment details. Each DAO class is responsible for one entity, such as ProductDAO for perfumes or OrderDAO for orders, keeping database logic centralized and reusable.
- The Service layer contains business logic. It acts as a bridge between raw database operations and the application's rules. For example, the OrderService might validate product availability before placing an order, calculate totals, or handle status changes. In a perfume shop context, services ensure that only in-stock perfumes are processed for purchase and that gender filters or promotions are applied logically.
- The Routes layer (using FlightPHP) defines the API endpoints exposed to the frontend. These routes translate HTTP requests (like viewing perfume details, adding to cart, or checking out) into service calls. Each route corresponds to RESTful conventions (e.g., GET /products, POST /orders), ensuring clear and stateless communication with the frontend.

This architecture, combined with the RESTful API, enables smooth interaction between the client-side and server-side. It allows users to browse perfumes, filter by gender, manage their cart, and complete purchases—all through structured and decoupled components. This organization not only simplifies development but also makes future updates, like adding review systems or loyalty programs, more manageable.

6.3. Backend Technologies

6.3.1. Backend Technologies

The backend of the fragrance shop “Smell of East and West” is built using PHP with the lightweight FlightPHP micro-framework, which provides a simple and efficient foundation for developing RESTful web applications. It supports clean routing, dependency injection, and modular structure, making it ideal for a project with a layered architecture.

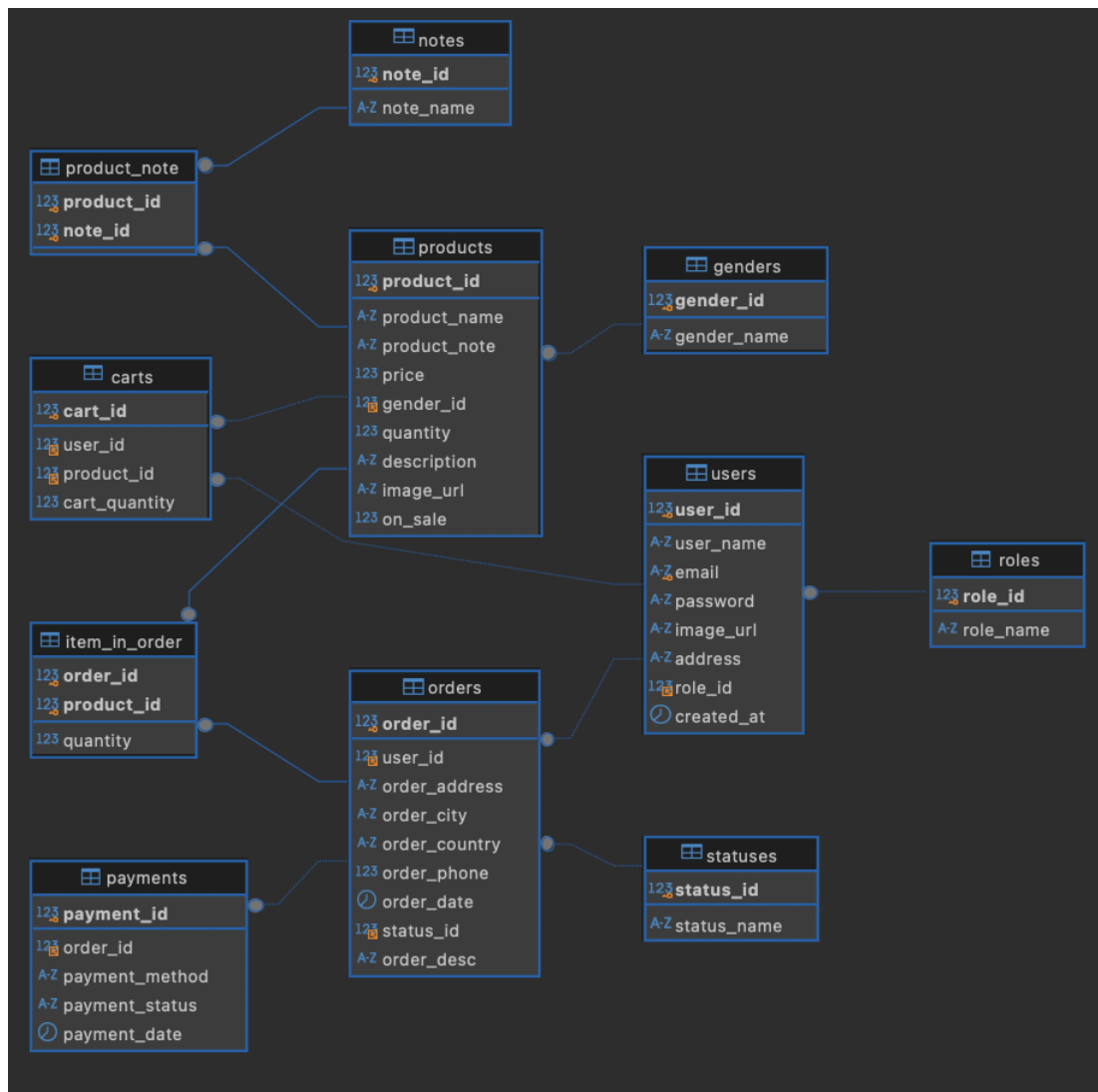
A RESTful API is implemented to ensure standardized, stateless communication between the frontend and backend. This design allows the frontend to interact with resources like products, carts, and orders through clear, consistent HTTP methods (GET, POST, PUT, DELETE).

To support development and ease integration, the project uses Swagger for API documentation. This provides a user-friendly interface and detailed information about available endpoints, expected inputs, and responses, helping both developers and testers understand and interact with the API effectively.

6.3.2. Database Entities

This database schema supports multiple tables. The “users” table stores customer account information, while the “roles” table defines user permissions such as admin or regular user. genders categorizes perfumes by target gender (e.g., male, female, unisex). “products” holds data about available perfumes, including name, brand, and price. Customers can leave” notes” (reviews or comments), and “product_note” links those notes to specific products. “carts” temporarily store items users intend to purchase, and “orders” record completed purchases. Each order is associated with one or more “item_in_order” entries, detailing the specific products bought. The “payments” table logs transaction information, and “statuses” track the progress of each order (e.g., pending, shipped, delivered).

Database Schema ER Diagram:



6.3.3. Design Patterns

6.3.3.1. DAO

This is achieved using the BaseDao class, which centralizes and simplifies all database interactions. As part of the Data Access Object (DAO) pattern, it serves as an intermediary between the application and the database, promoting separation of concerns and easier maintenance. It offers general-purpose methods for operations like insert, update, delete, and query.

For example, the insert method generates an SQL statement dynamically, based on the keys and values of the given entity, ensuring safe and accurate insertion into the designated table.

backend > rest > dao > BaseDao.php > ...

```
1  <?php
2  require_once __DIR__ . '/config.php';
3
4  class BaseDao {
5      7 references
6      protected $table;
7      4 references
8      protected $primaryKey;
9      46 references
10     protected $connection;
11
12     11 references | 11 overrides
13     public function __construct($table, $primaryKey = 'id') {
14         $this->table = $table;
15         $this->primaryKey = $primaryKey;
16         $this->connection = Database::connect();
17     }
18
19     6 references | 0 overrides
20     public function getAll(): mixed {
21         $stmt = $this->connection->prepare("SELECT * FROM {$this->table}");
22         $stmt->execute();
23         return $stmt->fetchAll();
24     }
25
26     7 references | 0 overrides
27     public function getById($id): mixed {
28         $stmt = $this->connection->prepare("SELECT * FROM {$this->table} WHERE {$this->primaryKey} = :id");
29         $stmt->bindParam(':id', $id);
30         $stmt->execute();
31         return $stmt->fetch();
32     }
33
34     9 references | 0 overrides
35     public function insert($data): mixed {
36         $columns = implode(separator: " , ", array: array_keys(array: $data));
37         $placeholders = ":" . implode(separator: " , :", array: array_keys(array: $data));
38         $sql = "INSERT INTO {$this->table} ($columns) VALUES ($placeholders)";
39         $stmt = $this->connection->prepare($sql);
40         return $stmt->execute($data);
41     }
42
43     5 references | 0 overrides
44     public function update($id, $data): mixed {
45         $fields = "";
46         foreach ($data as $key => $value) {
47             $fields .= "$key = :$key, ";
48         }
49         $fields = rtrim(string: $fields, characters: ", ");
50         $sql = "UPDATE {$this->table} SET $fields WHERE {$this->primaryKey} = :id";
51         $stmt = $this->connection->prepare($sql);
52         $data['id'] = $id;
53         return $stmt->execute($data);
54     }
55
56     7 references | 1 override
57     public function delete($id): mixed {
58         $stmt = $this->connection->prepare("DELETE FROM {$this->table} WHERE {$this->primaryKey} = :id");
59         $stmt->bindParam(':id', $id);
60         return $stmt->execute();
61     }
62 }
```


6.3.3.2. Front controller pattern

The `index.php` file acts as the front controller in an MVC setup, handling all incoming requests in one place. With `Flight::route()`, it connects each route to the right controller function, helping keep the code organized. This setup makes it easier to follow the MVC structure—controllers handle logic, models deal with data, and views take care of the interface. It also helps manage things like authentication, logging, and errors more cleanly and consistently across the app.

```
backend > php index.php
1  <?php
2
3  session_start(); // Start the PHP session
4  $_SESSION['user_id'] = 5; // Set user_id to 5 for testing purposes
5
6  ini_set(option: 'display_errors', value: 1);
7  ini_set(option: 'display_startup_errors', value: 1);
8  error_reporting(error_level: E_ALL);
9
10 require '../vendor/autoload.php'; //run autoloader
11 require 'rest/routes/routes.php'; // Include the routes file
12 // require 'rest/dao/BaseDao.php'; // Include the BaseDao file
13
14 Flight::start(); //start FlightPHP
15 ?>
```

7. Project Functionalities

7.1. Sign In



[Search](#) [Explore](#) [Account](#) [Cart](#) [Log in](#)

Sign in

Email

Password

☐ Remember password

[Log in](#)

Don't have an account? [Sign Up Here!](#)

ARAGON FRAGRANCES

Explore the East and West winds, embrace them and find a unique match for yourself.

PRODUCTS

[MDBootstrap](#)

[MDWordPress](#)

[BrandFlow](#)

[Bootstrap Angular](#)

USEFUL LINKS

[Your Account](#)

[Our Ingredients](#)

[Shipping Rates](#)

[Help](#)

CONTACT


[Francuske Revolucije BB, Iliđa, BiH](#)

[info@aragon.com](#)

[+225 883 000](#)

[+222-Aragon](#)

7.2. Sign Up



[Search](#) [Explore](#) [Account](#) [Cart](#) [Log in](#)

CREATE AN ACCOUNT

Your Name

Your Email

Password

Repeat your password

☐ I agree all statements in [Terms of service](#)

[Sign Up](#)

Already have an account? [Login here](#)

ARAGON FRAGRANCES

Explore the East and West winds, embrace them and find a unique match

PRODUCTS

[MDBootstrap](#)

[MDWordPress](#)

[BrandFlow](#)

[Bootstrap Angular](#)

USEFUL LINKS

[Your Account](#)

[Our Ingredients](#)

[Shipping Rates](#)

[Help](#)

CONTACT

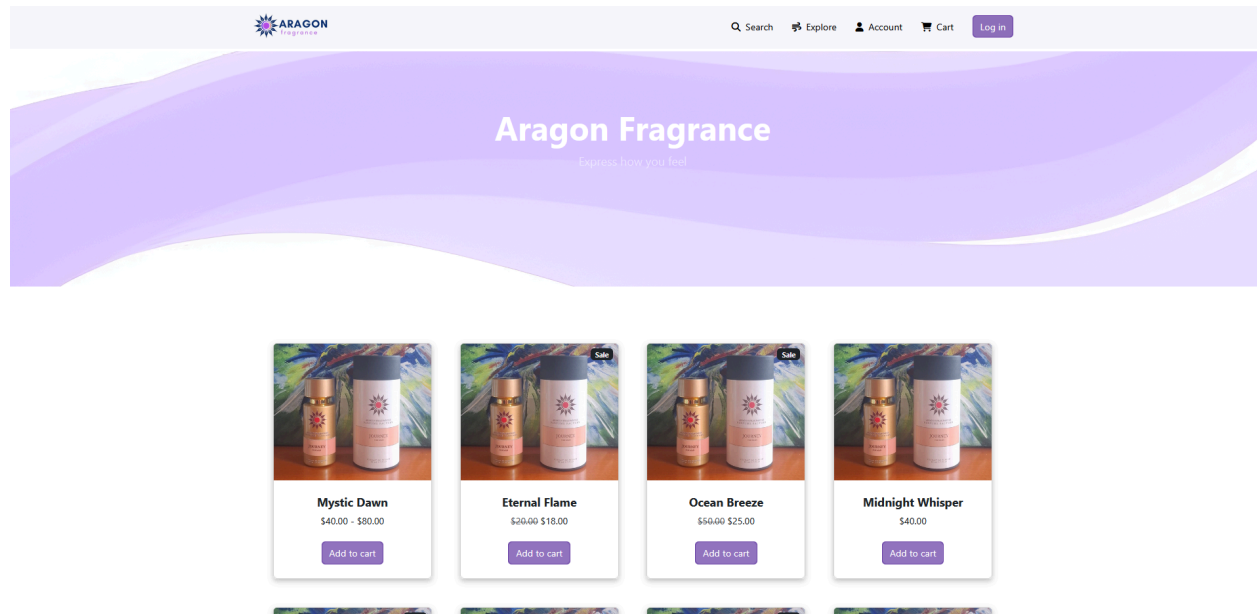
[Francuske Revolucije BB, Iliđa, BiH](#)

[info@aragon.com](#)

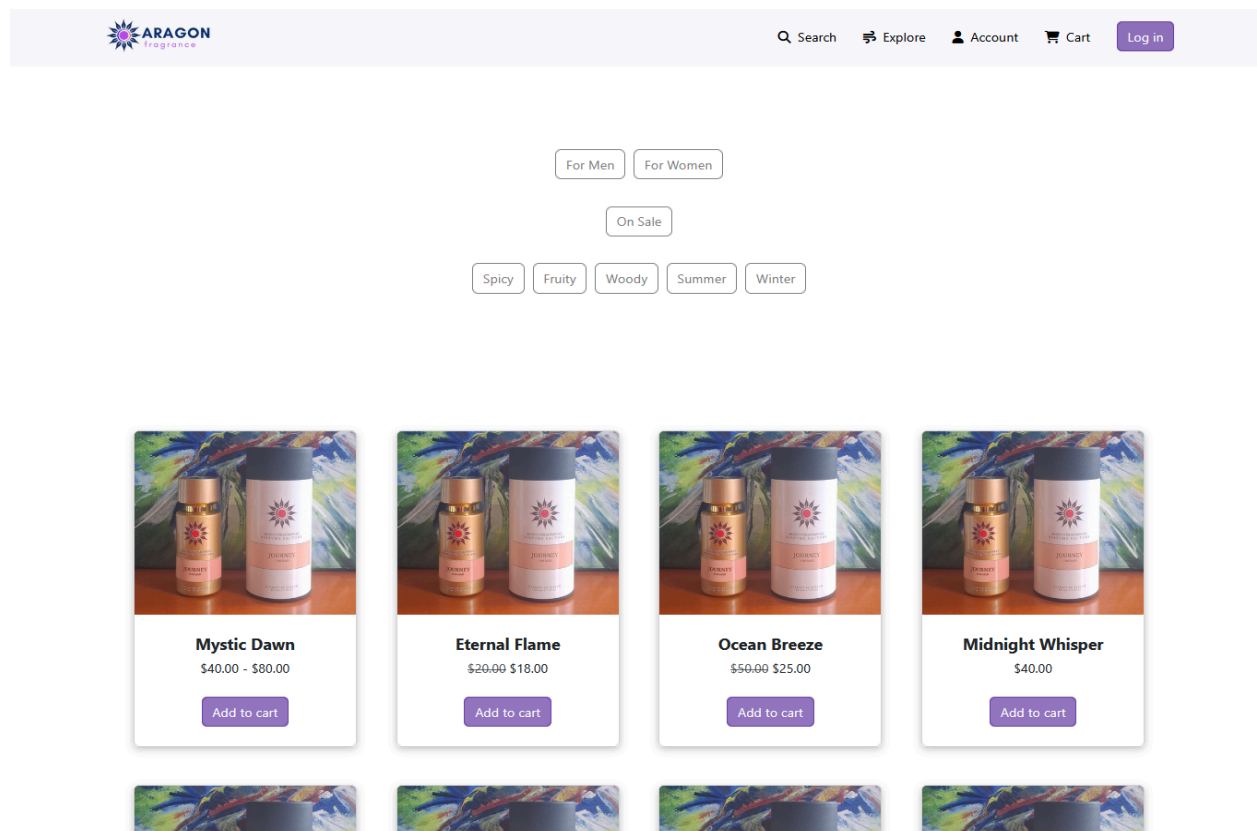
[+225 883 000](#)

[+222-Aragon](#)


7.3. Dashboard



7.4. Explore Page




7.5. User Profile



[Search](#) [Explore](#) [Account](#) [Cart](#) [Log in](#)

User Profile

Profile Picture



Profile Information

Username:

JohnDoe

Email:

johndoe@example.com

Delivery Address:


123 Main St, City, Country

Update Profile

Order History

Order ID	Description	Price	Date	Status	Actions
12345	Perfume A	\$50.00	2023-01-01	Delivered	View
12346	Perfume B	\$60.00	2023-02-01	Pending	View Cancel
12347	Perfume C	\$70.00	2023-03-01	Delivered	View
12348	Perfume D	\$80.00	2023-04-01	Pending	View Cancel
12349	Perfume E	\$90.00	2023-05-01	Delivered	View
12350	Perfume F	\$100.00	2023-06-01	Pending	View Cancel


7.6. Cart



[Search](#) [Explore](#) [Account](#) [Cart](#) [Log in](#)


Items in Cart

☐




Mystic Dawn - \$50.00

☐




Eternal Flame - \$60.00

☐



Ocean Breeze - \$70.00

☐



Midnight Whisper - \$80.00

Total Amount

\$260.00

Checkout

ARAGON FRAGRANCES

Explore the East and West winds, embrace them and find a unique match for yourself.

PRODUCTS

[MDBootstrap](#)
[MDWordPress](#)
[BrandFlow](#)
[Bootstrap Angular](#)


USEFUL LINKS

[Your Account](#)
[Our Ingredients](#)
[Shipping Rates](#)
[Help](#)

CONTACT

[Francuske Revolucije BB, Ilidža, BiH](#)
[info@aragon.com](#)
[+225 883 000](#)
[+222-Aragon](#)

7.7. Checkout



Search

Explore

Account

Cart

Log in

Checkout

Profile Information

Name:

John Doe

Email:

john.doe@example.com

Address:

123 Main St

Payment Information

Card Number:


Expiry Date:

MM/YY

CVV:

Submit Payment


Order Summary



Mystic Dawn

Experience the serene and enchanting scent of Mystic Dawn.


Price: \$50.00



Eternal Flame

Ignite your senses with the passionate scent of Eternal Flame.


Price: \$60.00



Ocean Breeze

Feel the refreshing and invigorating scent of Ocean Breeze.

Price: \$70.00




Midnight Whisper

Embrace the mysterious and alluring scent of Midnight Whisper.

Price: \$80.00

Total: \$260.00

7.8. Admin Dashboard



Search

Explore


Account

Cart

Log in

Admin Profile

Profile Picture



Profile Information

Username:

JohnDoe

Email:

johndoe@example.com

Delivery Address:

123 Main St. City, Country

Update Profile

Admin Panel

Total Users

150

Total Orders

320

Pending Orders

45

Engagement

User engagement metrics and analytics.

Username	Email	Number of Orders
JohnDoe	johndoe@example.com	5
JaneDoe	janedoe@example.com	3

All Orders

Order ID	Description	Price	Date	Status	View	Actions
12345	Perfume A	\$50.00	2023-01-01	Delivered	View	
12346	Perfume B	\$60.00	2023-02-01	Pending	View	Cancel Accept
12347	Perfume C	\$70.00	2023-03-01	Delivered	View	
12348	Perfume D	\$80.00	2023-04-01	Pending	View	Cancel Accept
12349	Perfume E	\$90.00	2023-05-01	Delivered	View	
12350	Perfume F	\$100.00	2023-06-01	Pending	View	Cancel Accept
12351	Perfume G	\$110.00	2023-07-01	Accepted	View	Cancel

6. Individual Contributions

For this submission both students equally contributed and collaborated on the development of the project structure, functionalities and ideas.

- (50%) Edina Kurto:

Main tasks:

- Writing of nonfunctional requirements, and writing user stories
- Release plan creation
- Diagrams
- Database
- Backend of the project, DAO Layer
- Documentation of the project

- (50%) Arman Çulah:

Main tasks:

- Creating Goal oriented plan
- Writing of functional requirements, and writing user stories
- Diagrams
- Frontend of the project, Services
- Plan manager