

Software Architecture Document Otenro

Table of content

Content	page
1. Introduction-----	03
1.1. Purpose of the document-----	03
1.2. Scope-----	03
2. Architectural Representation-----	04
3. Use Case view-----	05
3.1. Otenro user use case-----	05
3.1.1.Architecturally Significant use cases-----	05
3.1.2.Brief Description of use cases-----	07
3.2. App user use case-----	09
3.2.1.Architecturally Significant use cases-----	09
3.2.2.Brief Description of use cases-----	11
4. Logical view-----	12
4.1. Application Layer – Otenro -----	13
4.2. Application Layer – App user-----	30
5. Database -----	39

1. Introduction

1.1. Purpose of the Document

This document provides the architectural outline of the Otenro system. Different architectural views are used to illustrate different aspects of the system. This document also presents the significant architectural decisions that are made on the system.

1.2. Scope

The scope of the document is to describe the architectural goals and constraints, the Use Case View and the Logical View.

2. Architectural Representation

Use Case view

Audience: all stakeholders

Area: describes the set of scenarios and/or use cases that are critical to the architecture

Related Artifacts: Use Case Document

Logical view

Audience: Otenro designers

Area: Functional Requirements, to assess functionality

Related Artifacts: Package Diagram

3. Use-Case View

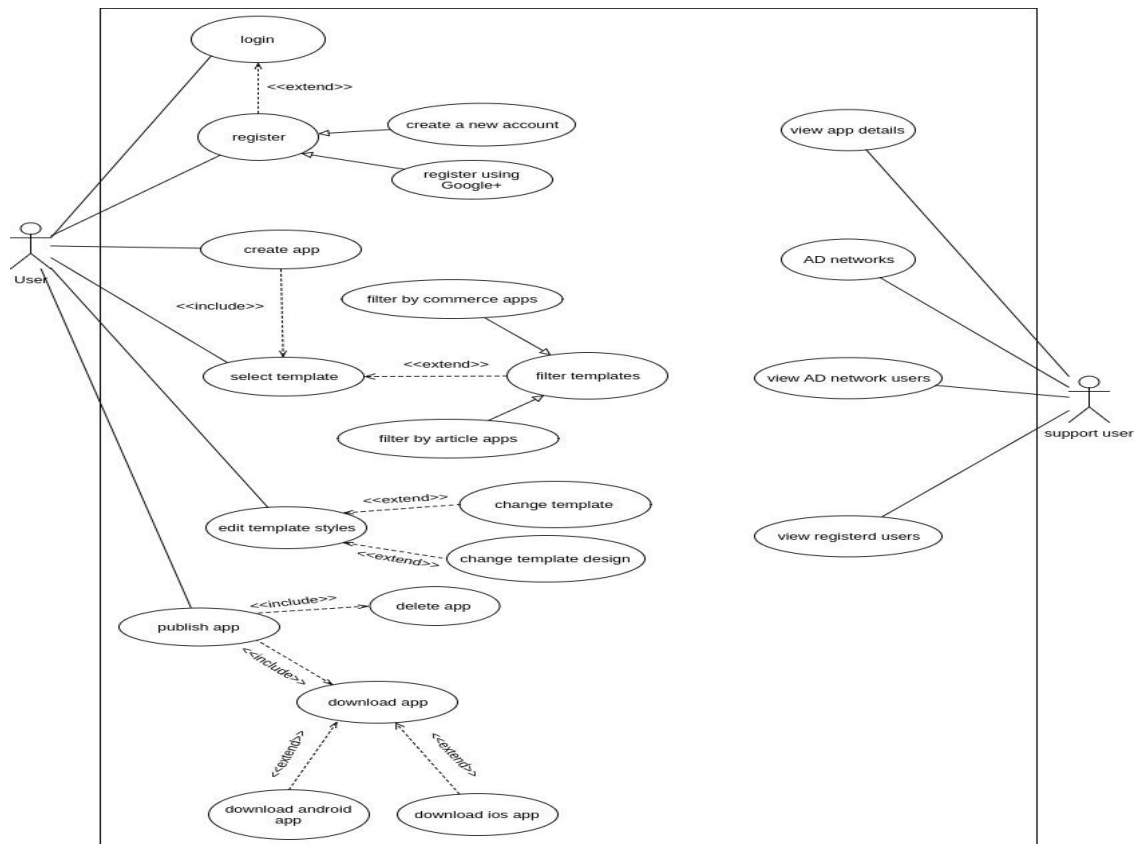
3.1 Otenro User Use case

“The use case view describes a view of the system’s architecture that covers the general system behavior from the point of view of the involved stakeholders.” This view exposes the set of use cases that defines the core functionality of Otenro.

3.1.1 Architecturally significant use cases

The following are the set of use cases that are architecturally significant to the Otenro:

- Register User
- Login user
- Create Application
- Select template
- Edit Style
- Publish Application
- Delete Applications
- Manage Payments
- Manage Store settings
- Manage Site settings
- Manage Orders
- Manage Inventory
- Manage articles
- Manage Email
- add Shipping
- add Taxes
- add Categories
- add Products
- add Articles



3.1.2 Brief Description of Use Cases

- **Register User:**
 - User must register in-order to use Otenro.
- **Login User:**
 - User must logged-in to create Applications.
- **Create Application:**
 - A privileged user can create applications.
- **Select template:**
 - Users will be provided with a template view to select their desired templates user two categories “commerce” and “article”.
- **Edit style:**
 - User have been provided with the functionalities to design his Application by changing styles and select templates.
- **Publish application:**
 - Ability to download the application from both “Google play” and “Apps store”.
- **Delete Application:**
 - User can delete his Applications.
- **Manage Payment:**
 - Users can configure their payment settings, “cash on pickup, cash on delivery, pay pal, Stripes or Authorize.net”
- **Manage store settings:**
 - User save his store settings “General details, Contact Us, About Us, Policies and Terms and Conditions”, under commerce Applications.
- **Manage Site settings:**
 - Users can save his site settings “Contact Us, About Us, Policies and Terms and Conditions” under Article Applications.
- **Manage orders:**
 - A privileged user can update and view the order details of products in his application, which comes under 'commerce' Applications.
- **Manage emails:**
 - Users can setup email details for orders on his application.

- **Manage Inventory:**
 - Users are provided with inventory view of available products. Can update delete products from the view.
- **Manage Articles:**
 - Users can view their articles and manage them on their Applications under Article category.
- **Add Shipping:**
 - User can add shipping details for his products, Under three shipping categories
 - “Flat Rates, Weight based and pickup”.
- **Add taxes:**
 - Taxes can be add to the products according to the user specifications.
- **Add categories:**
 - Categories can be added for both “commerce” and “article” type applications. Under each specific categories users can publish their products or articles.
- **Add Products:**
 - Users can add new products having their own specifications for a product, “name, variants, price, weight, quantity”. A product can only have maximum 4 variants.
- **Add Articles:**
 - Users can add new Articles and publish then under categories having “name” and “descriptions”.

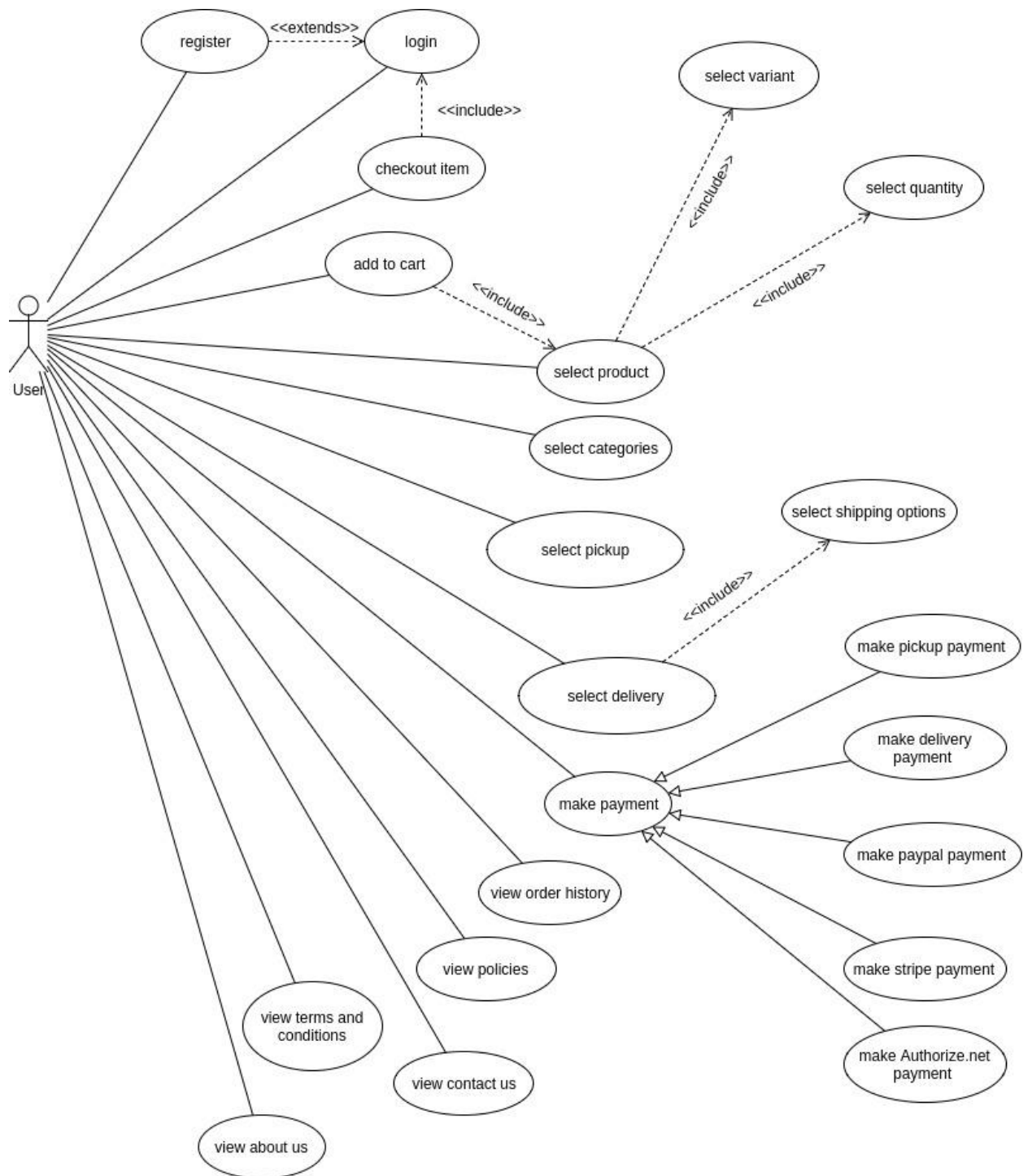
3.2 App user use cases

“The use case view describes a view of the application’s architecture that covers the general behavior from the point of view of the App user.” This view exposes the set of use cases that defines the functionality of the Application.

3.2.1 Architecturally significant use cases

The following are the set of use cases that are architecturally significant to the App user:

- User Login
- User register
- Select categories
- Select products
- Add to cart
- Select pickup
- Select delivery
- Make payment
- View orders history
- View about us
- View policies
- View contact us
- View terms and condition

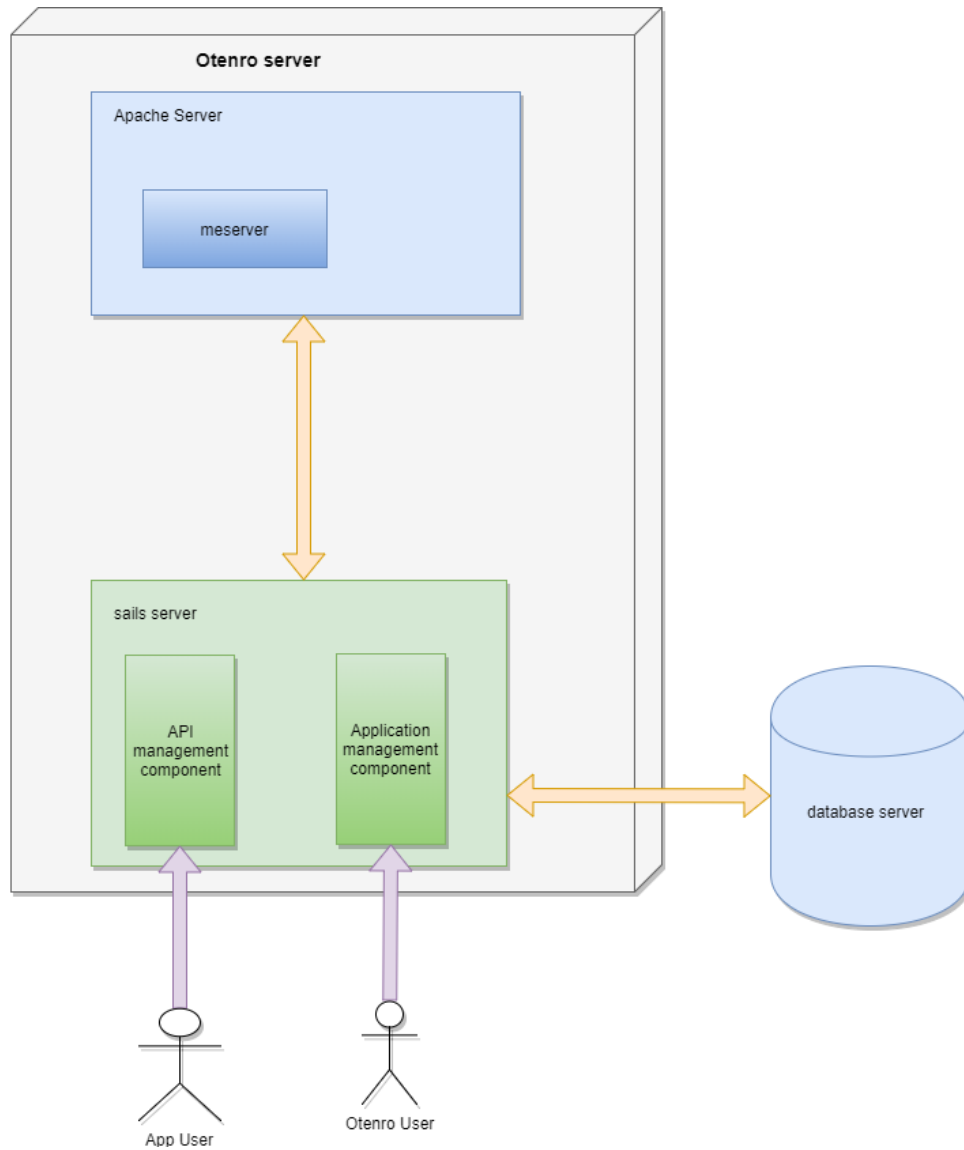


3.2.2 Brief Description of Use Cases

- User login :
 - User need to log in in order to complete an order process.
- User Register:
 - User must register to complete any order processes
- Select categories:
 - User can select categories in order to select a product.
- Select product:
 - User can select products.
- Add to cart:
 - User can add select products to the cart.
- Select pickup:
 - Users can select pickup options by providing pickup information's for their products.
- Select delivery:
 - Users can select delivery option by selecting shipping details for their products.
- Make payment:
 - User can make payment “on pickup or on delivery” also he can make payments using papal, stripes and authorize.net.
- View orders history:
 - Users can view the orders they have made.
- View about us:
 - Can view about us details of the application.
- View contact us:
 - User can view contact us information.
- View policies:
 - User can view policies data.
- View terms and conditions:
 - User can view terms and conditions on the applications.

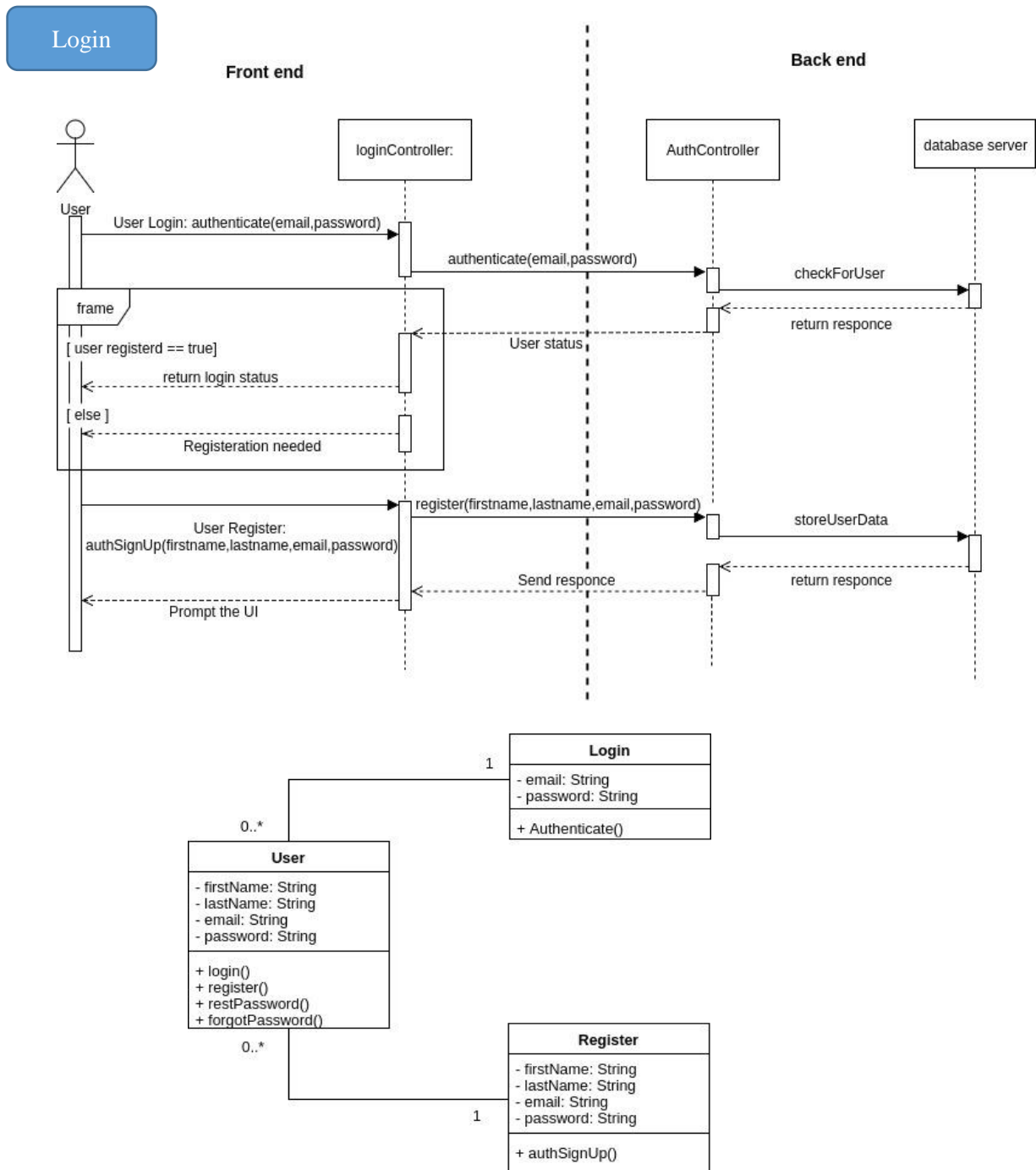
4. Logical View

“The conceptual organization of the system is presented in the Logical view in terms of layers, packages, subsystems, and classes.”

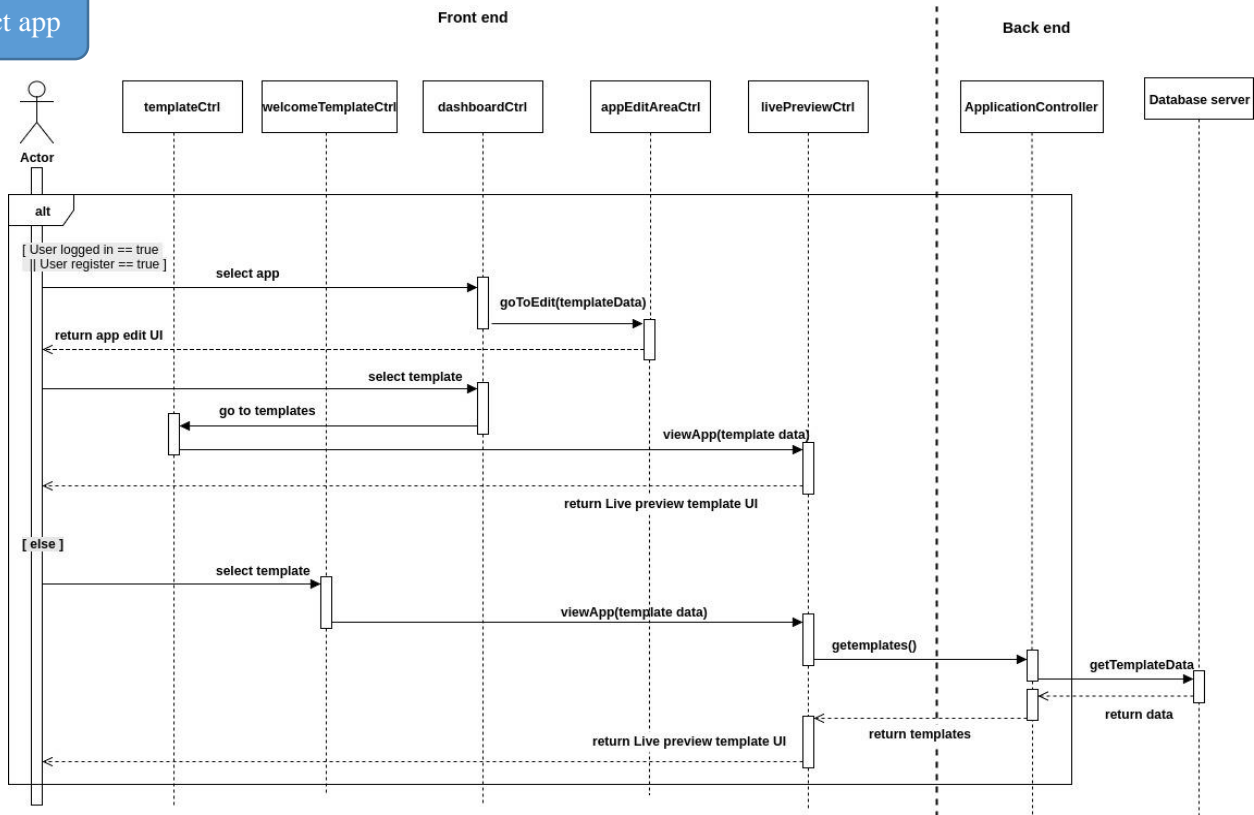


4.1. Application Layer - Otenro

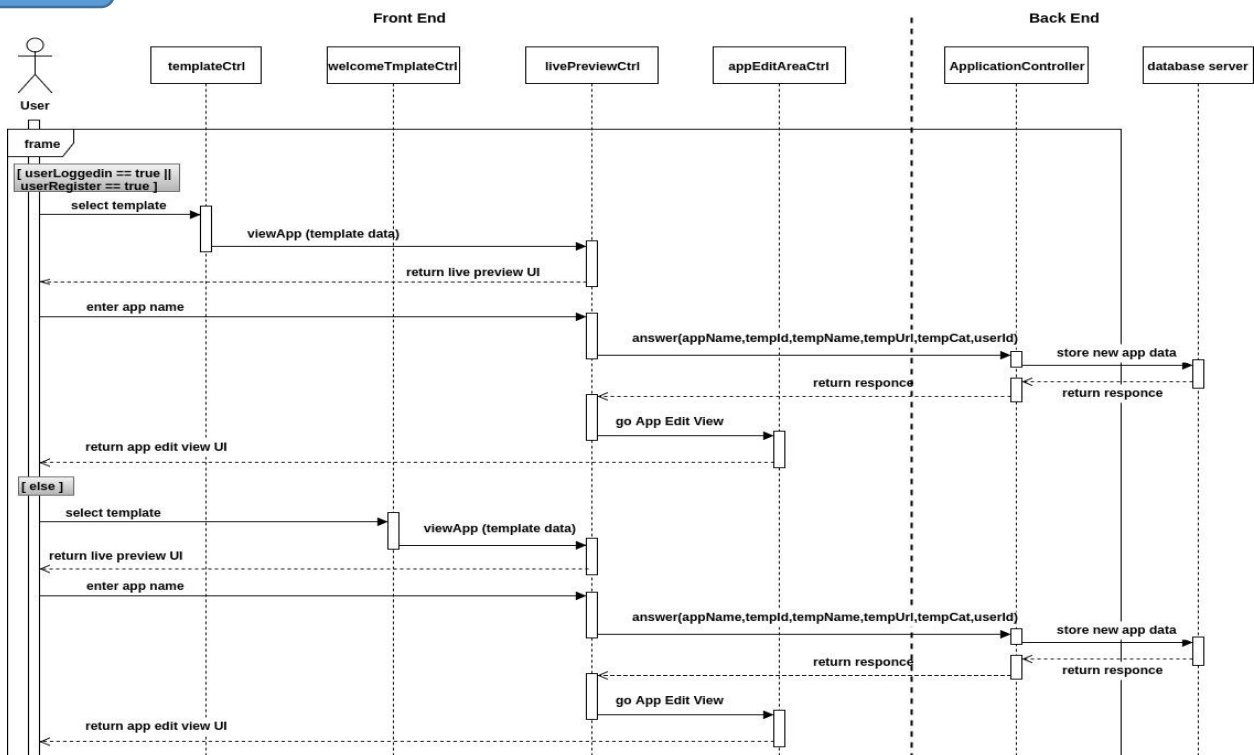
The application Layer in this implementation modifies the traditional Front Controller with the template view organization with Back-end components, developed with the Sails framework. The front Controller now handles request from the client to the server. The services are acting in between front Controllers and back Controllers as a bridge.

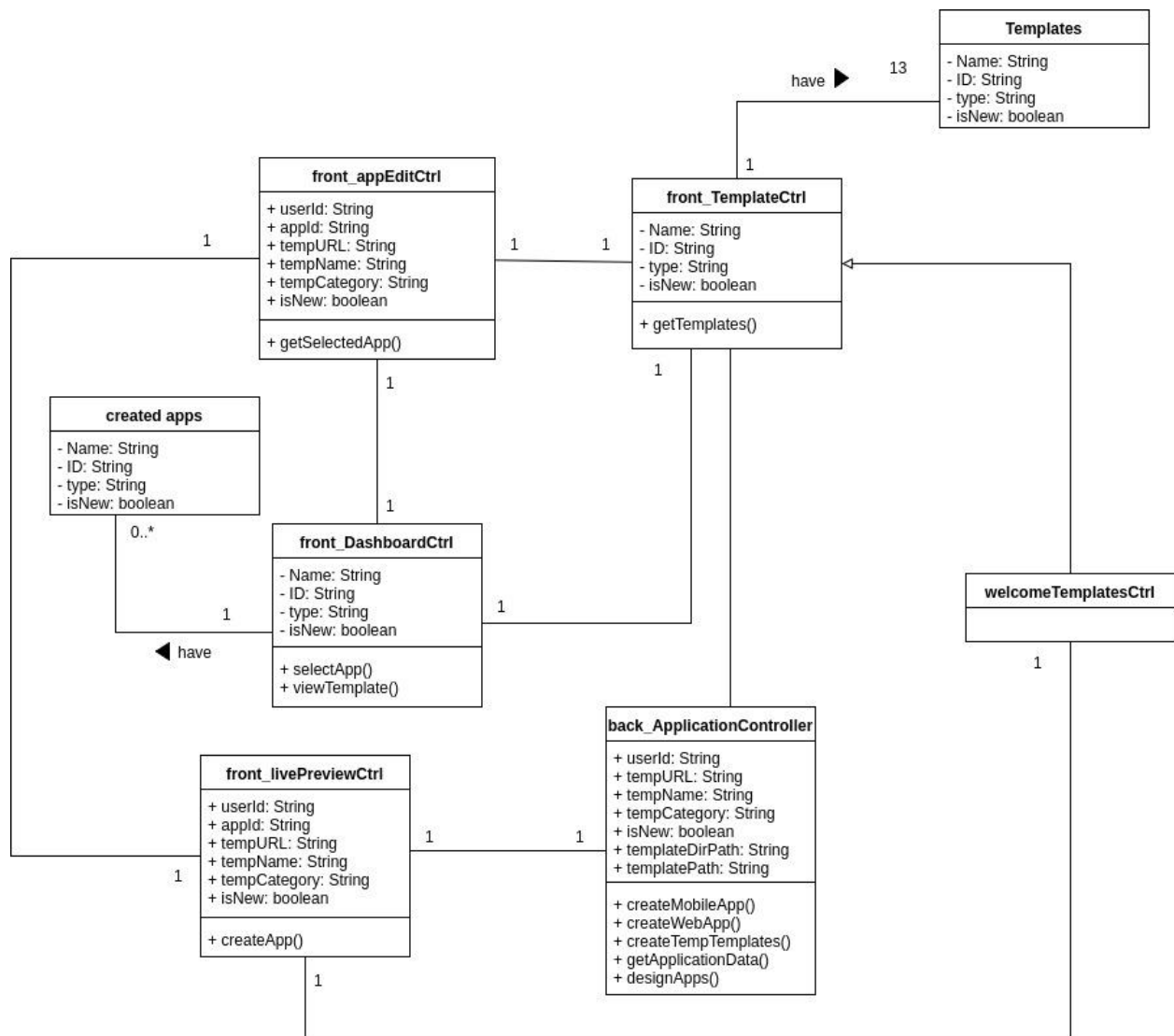


Select app

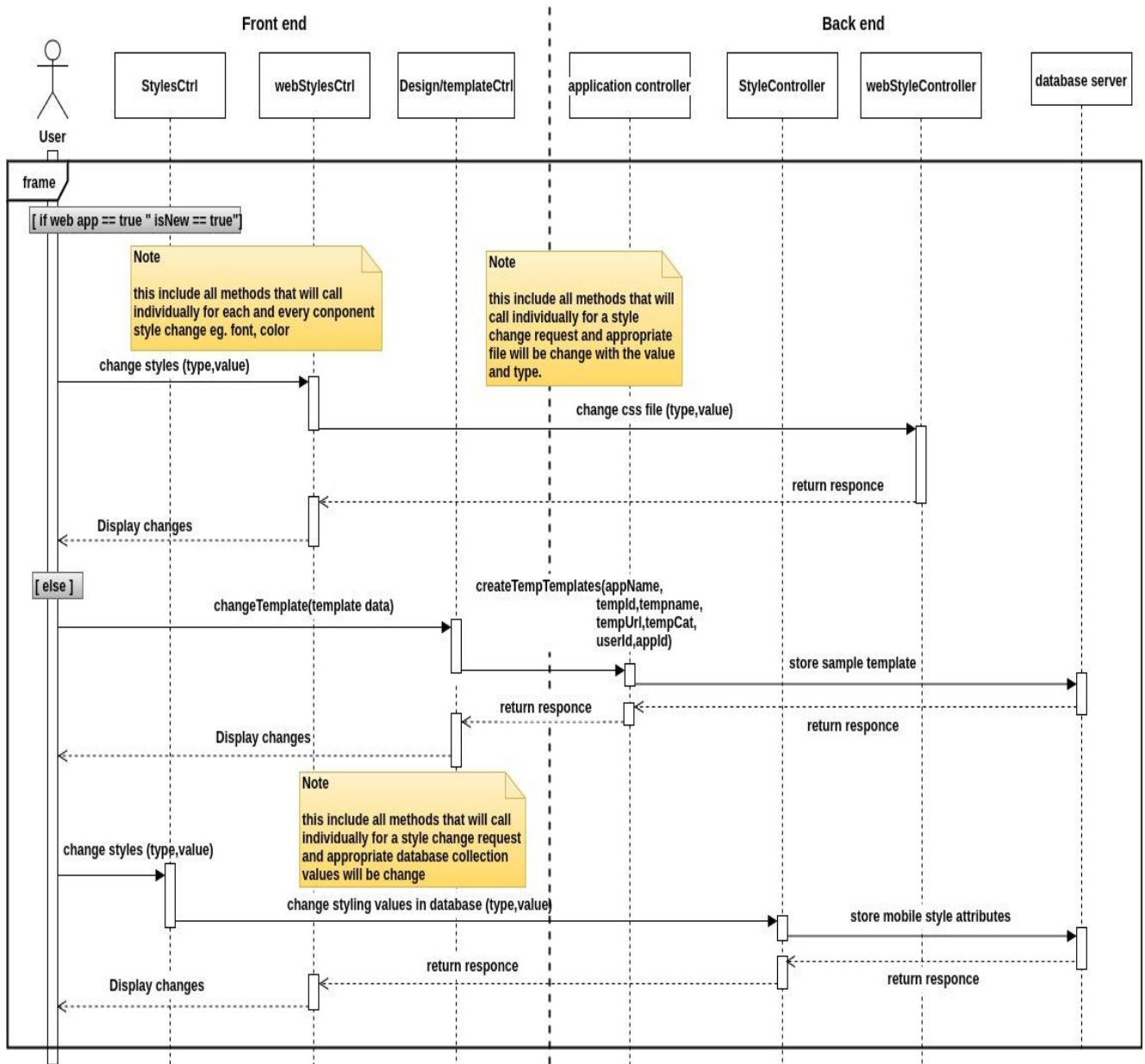


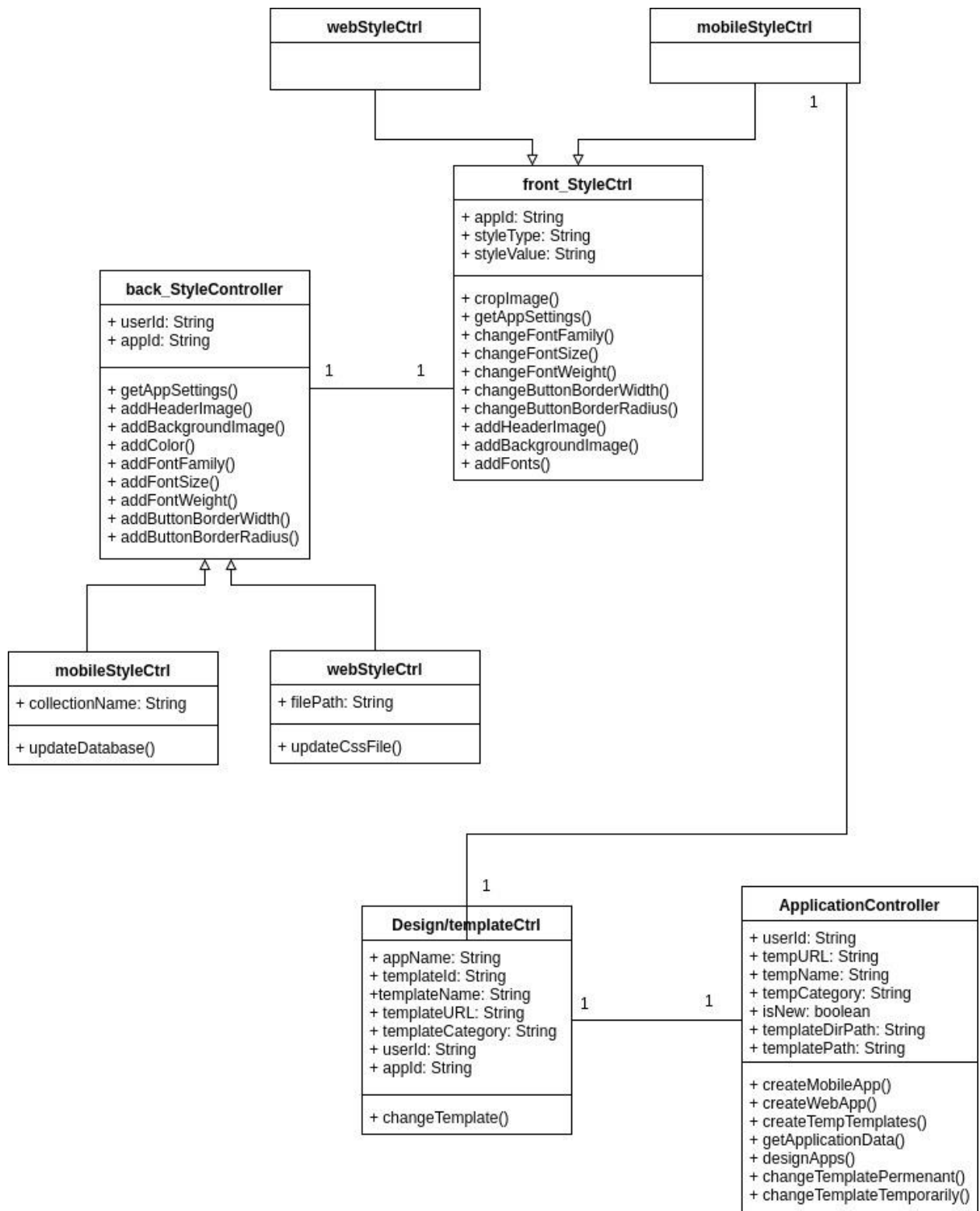
Create app



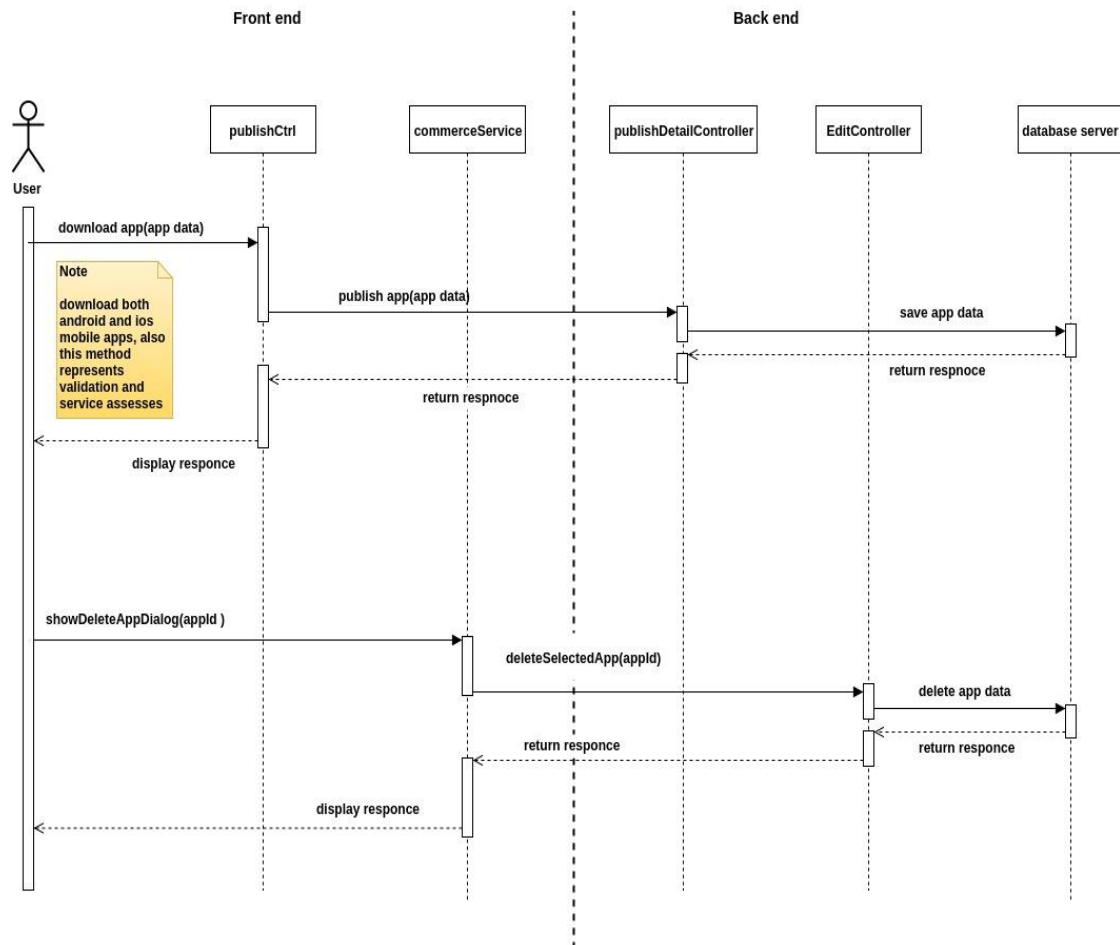


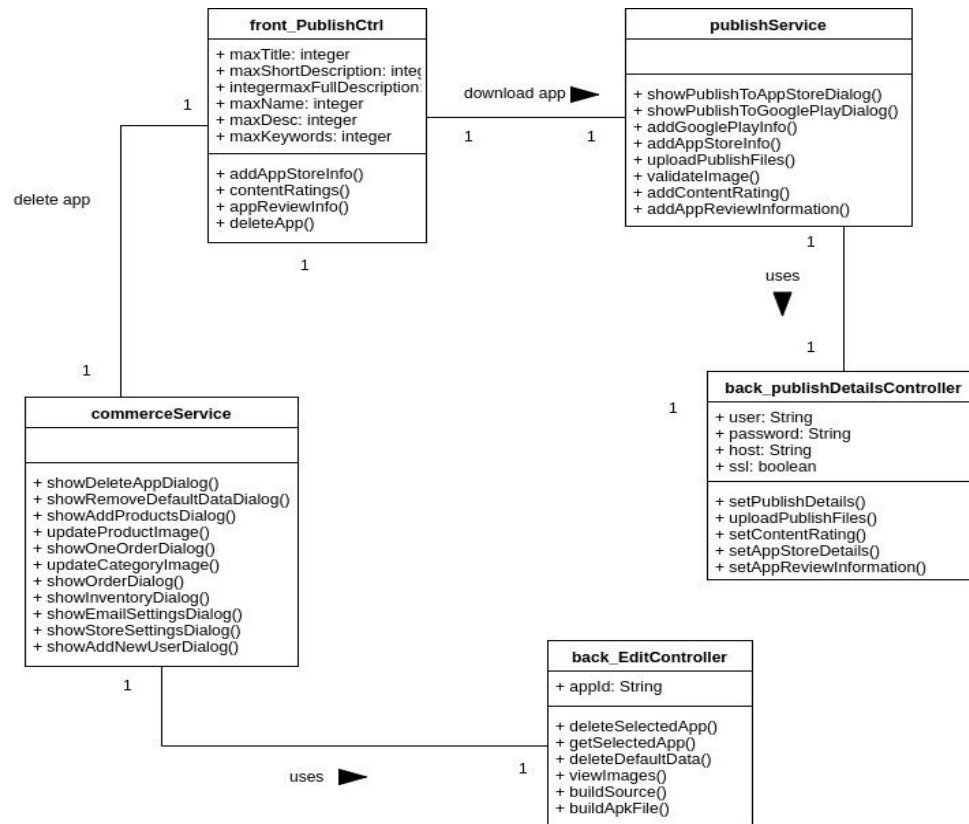
App styling



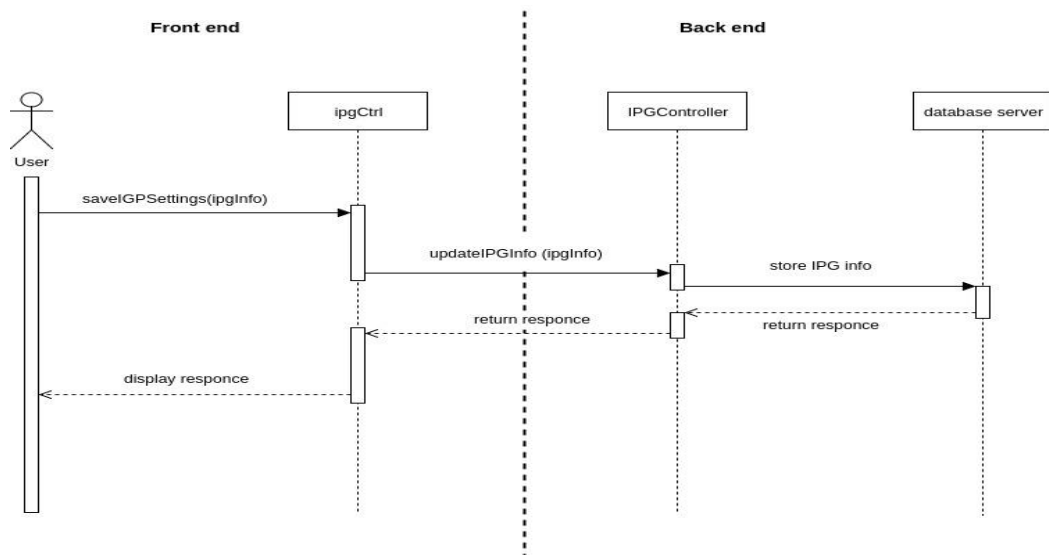


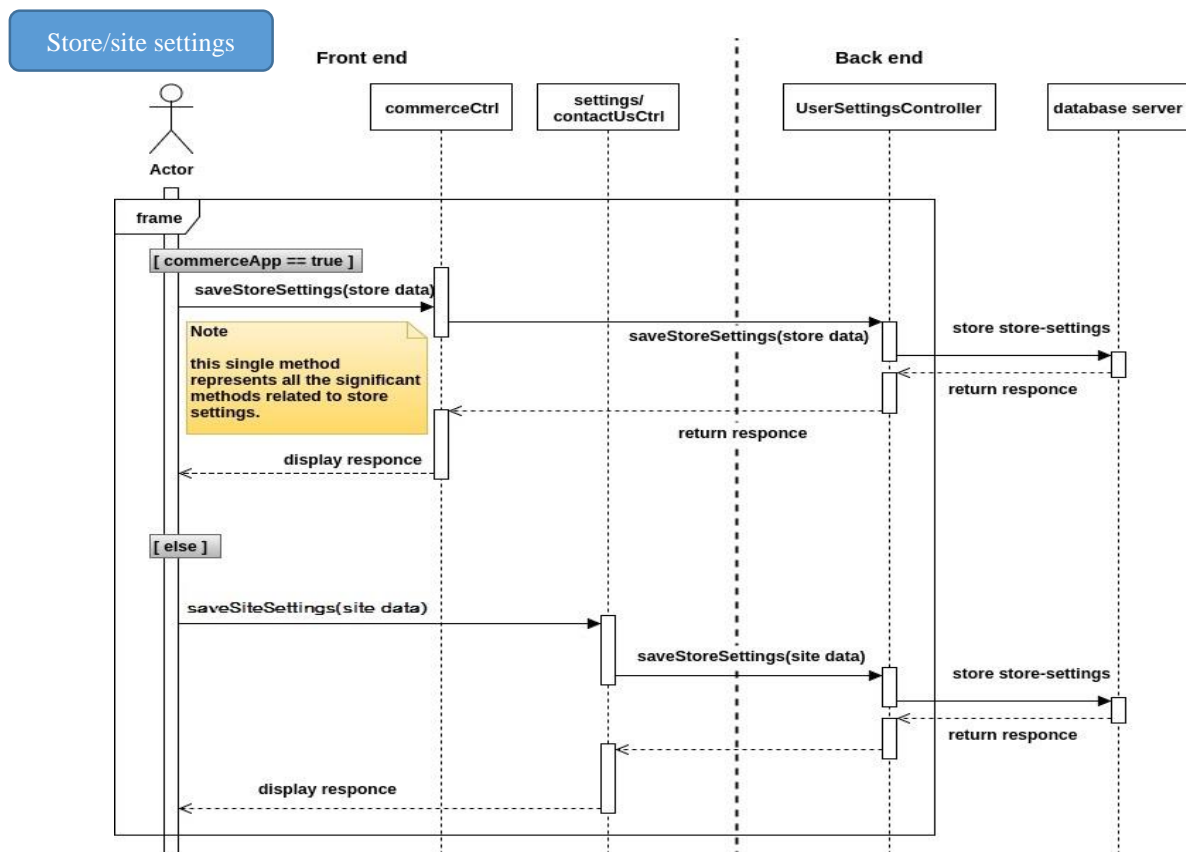
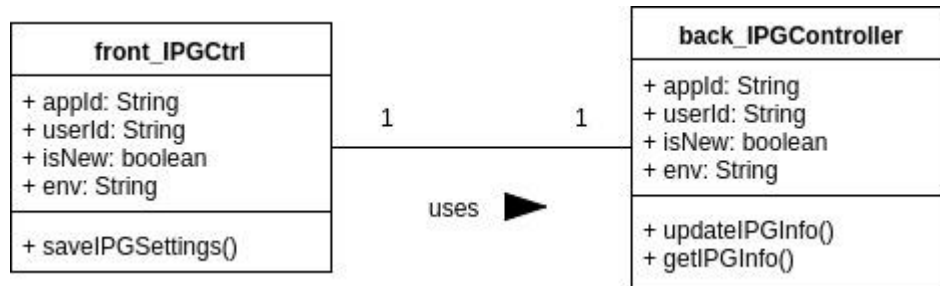
Publish app



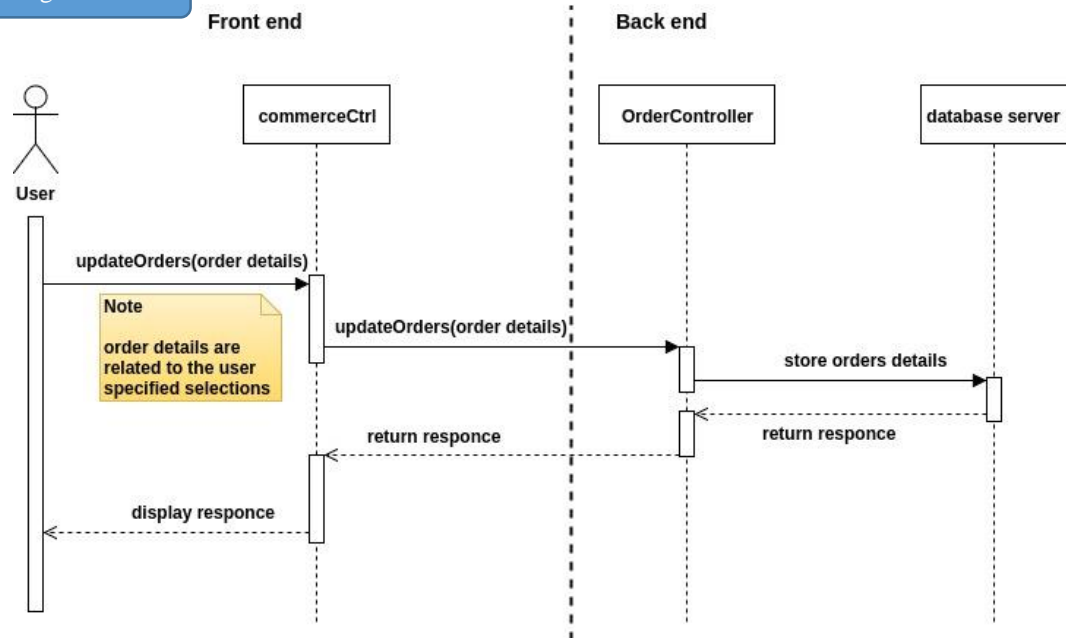


Manage payment

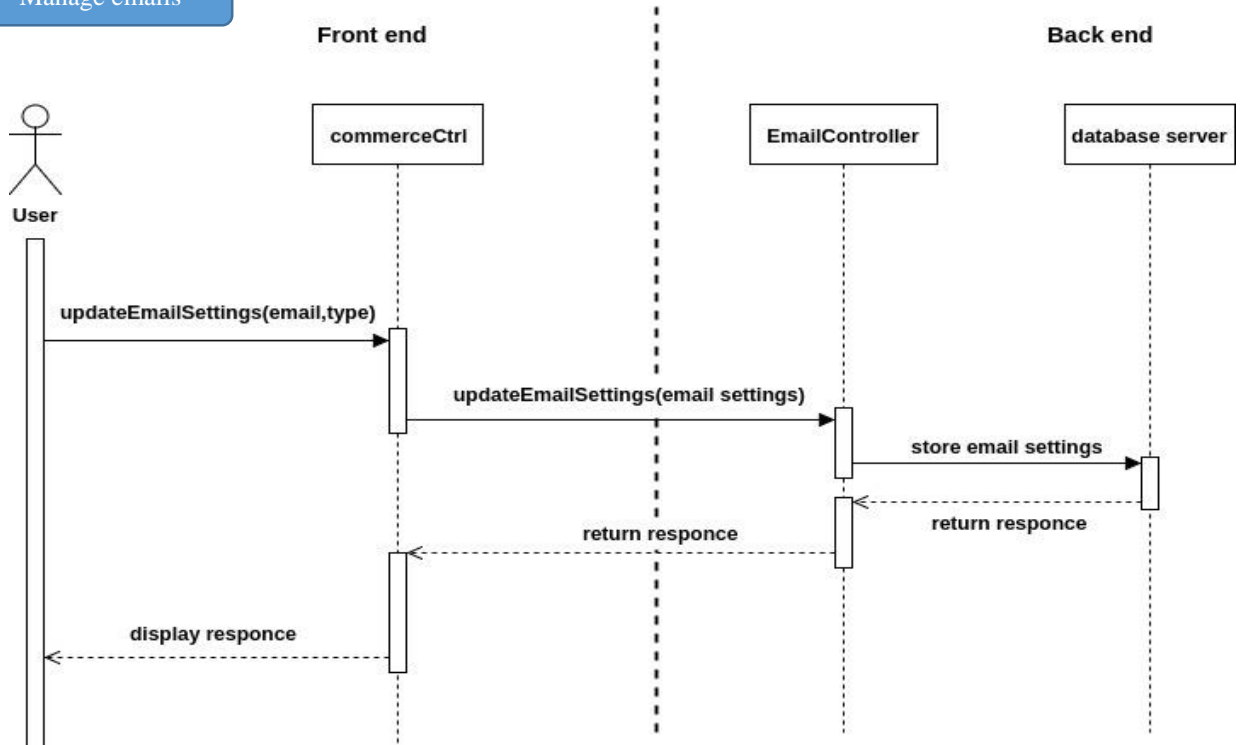


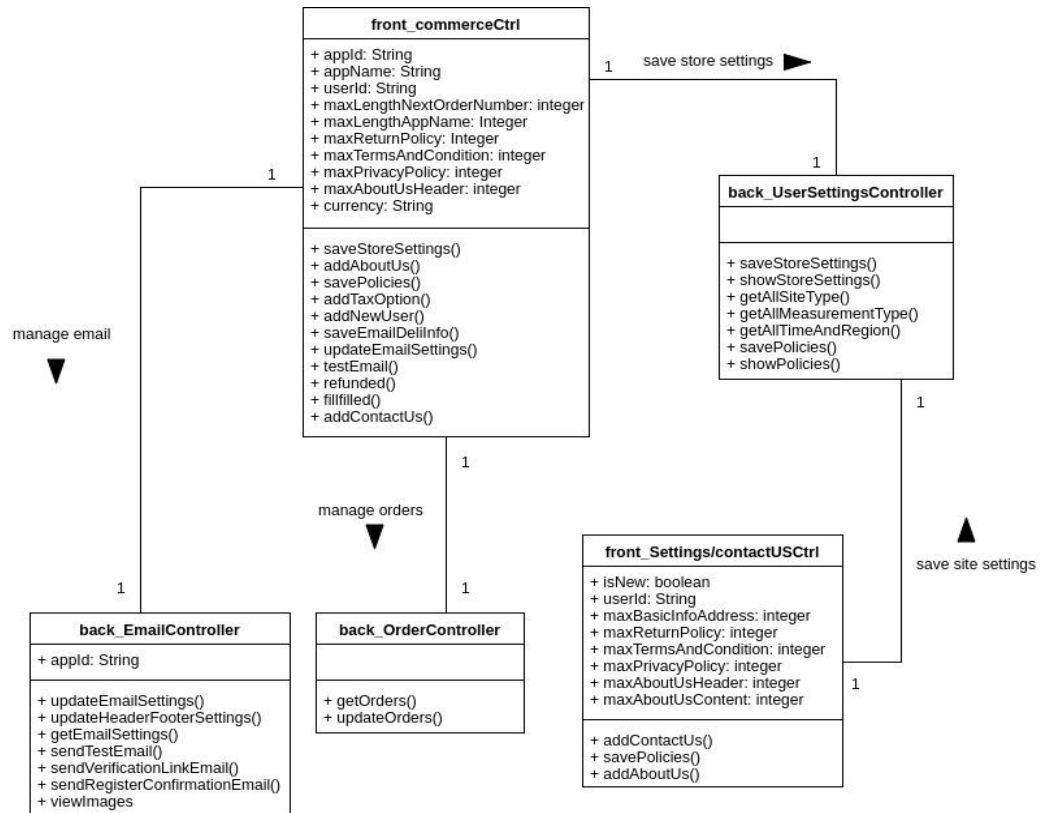


Manage orders

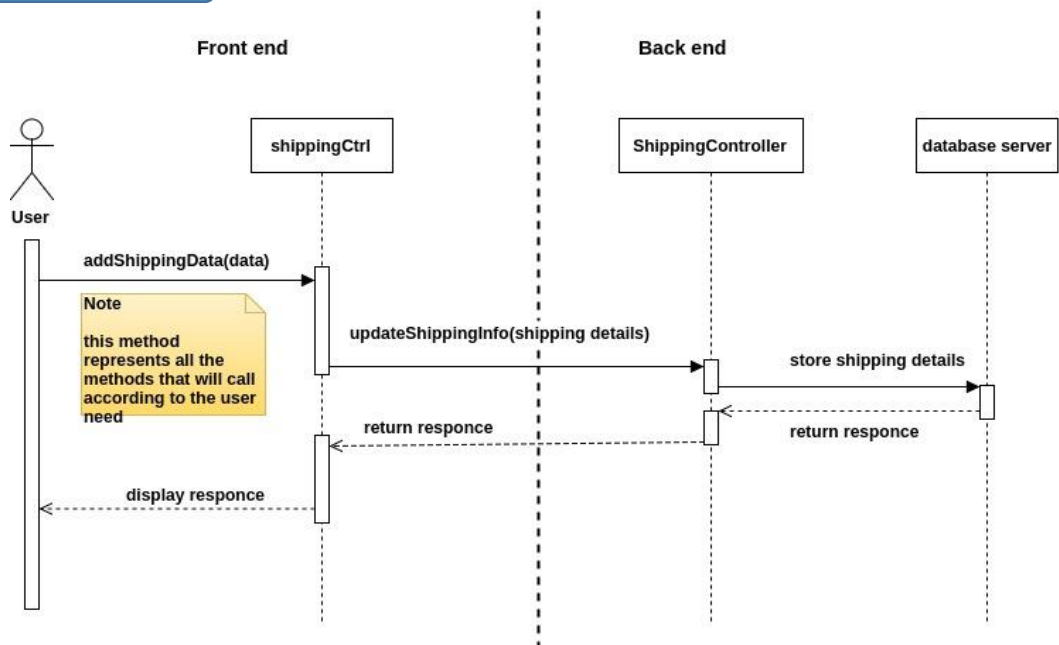


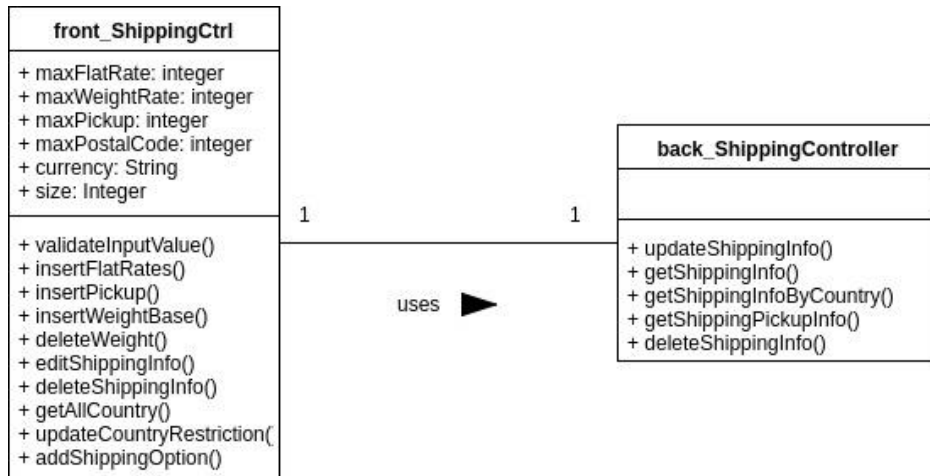
Manage emails



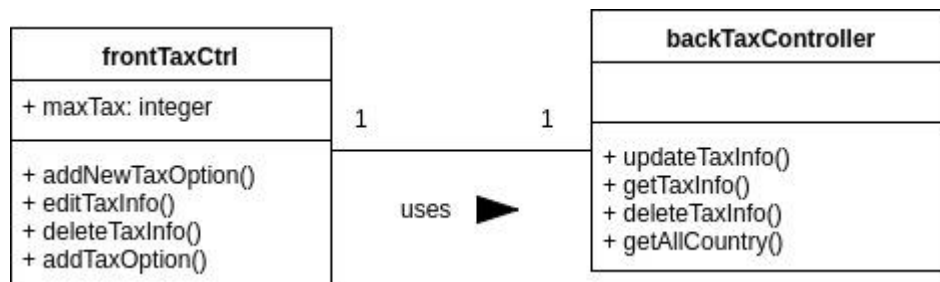
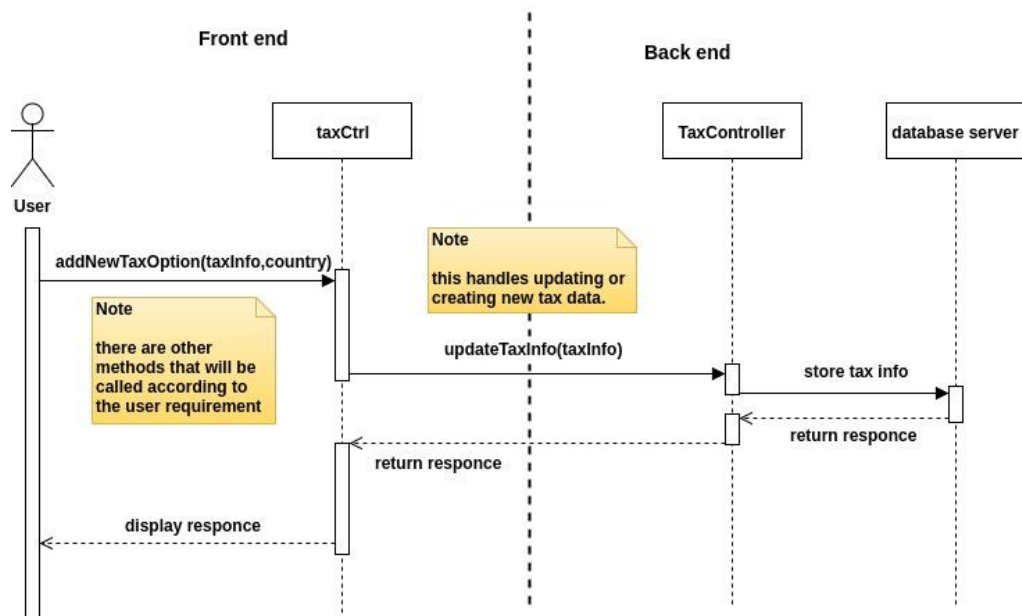


Add shipping

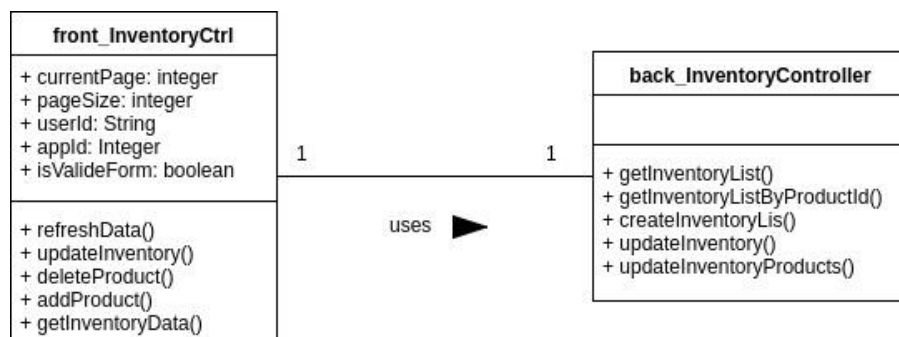
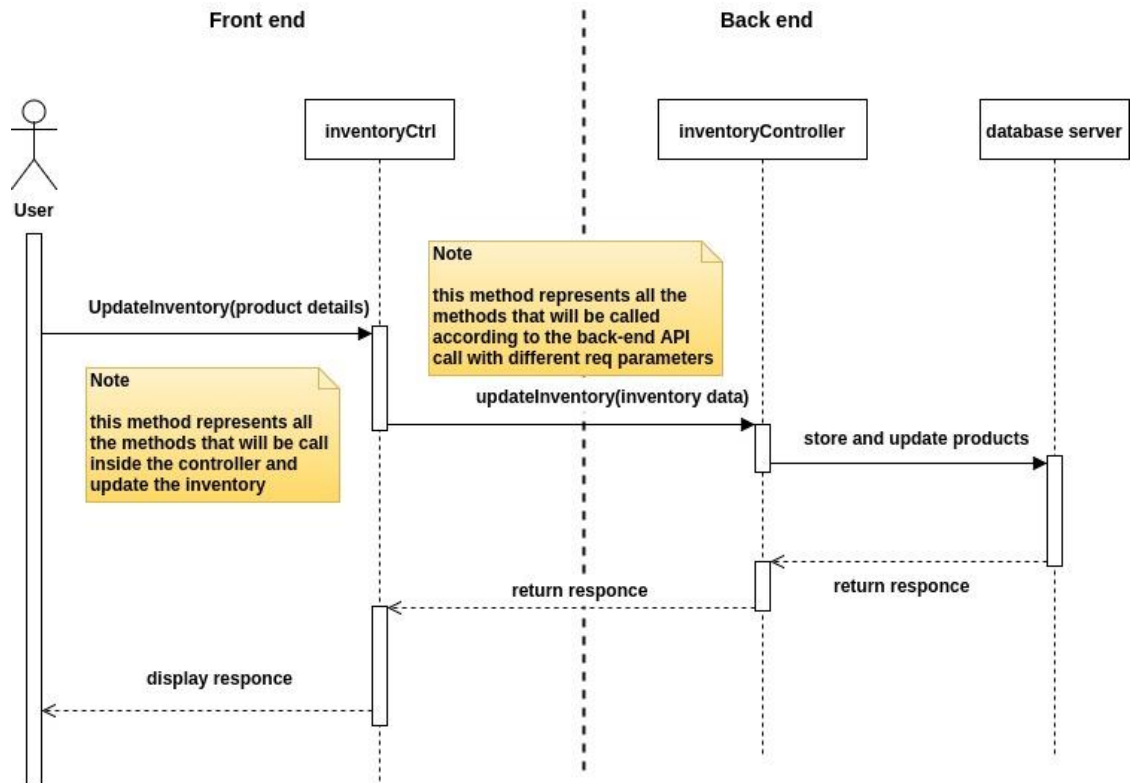




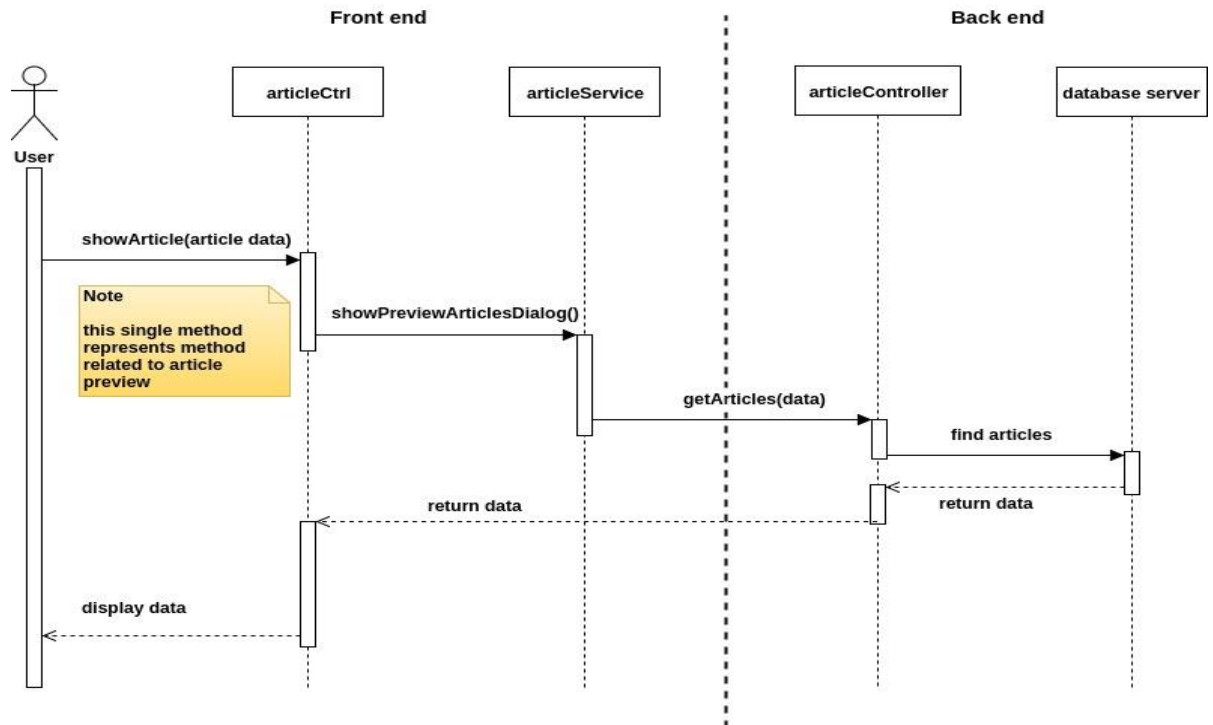
Add Tax



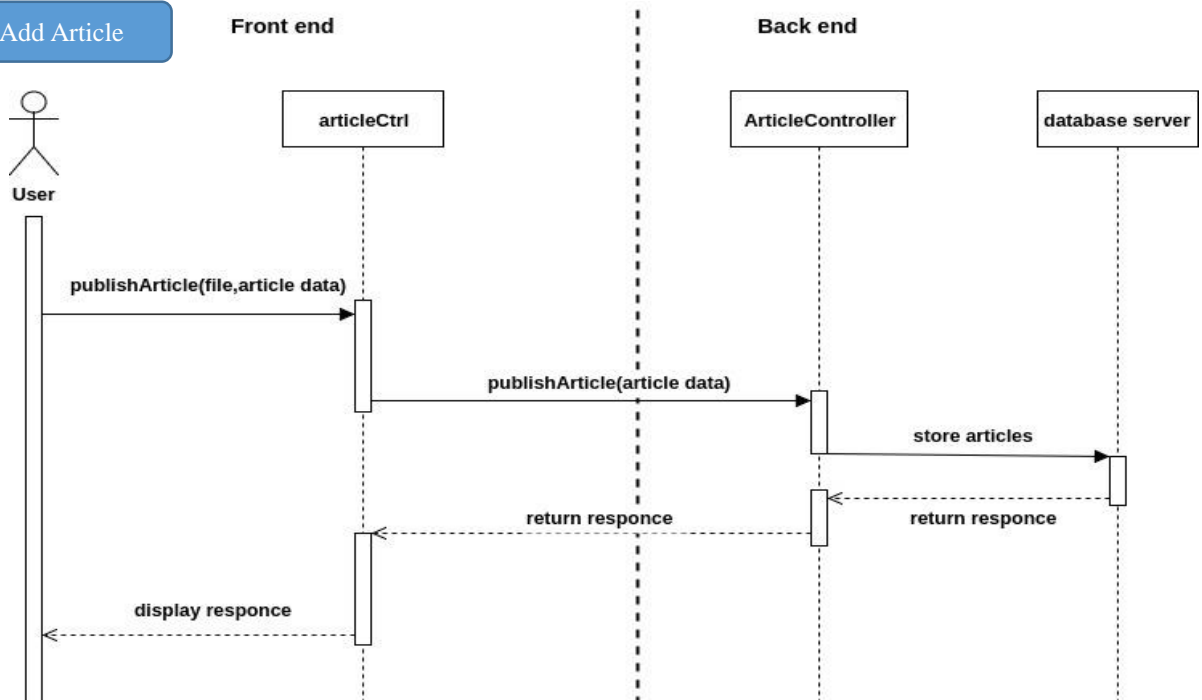
Manage inventory

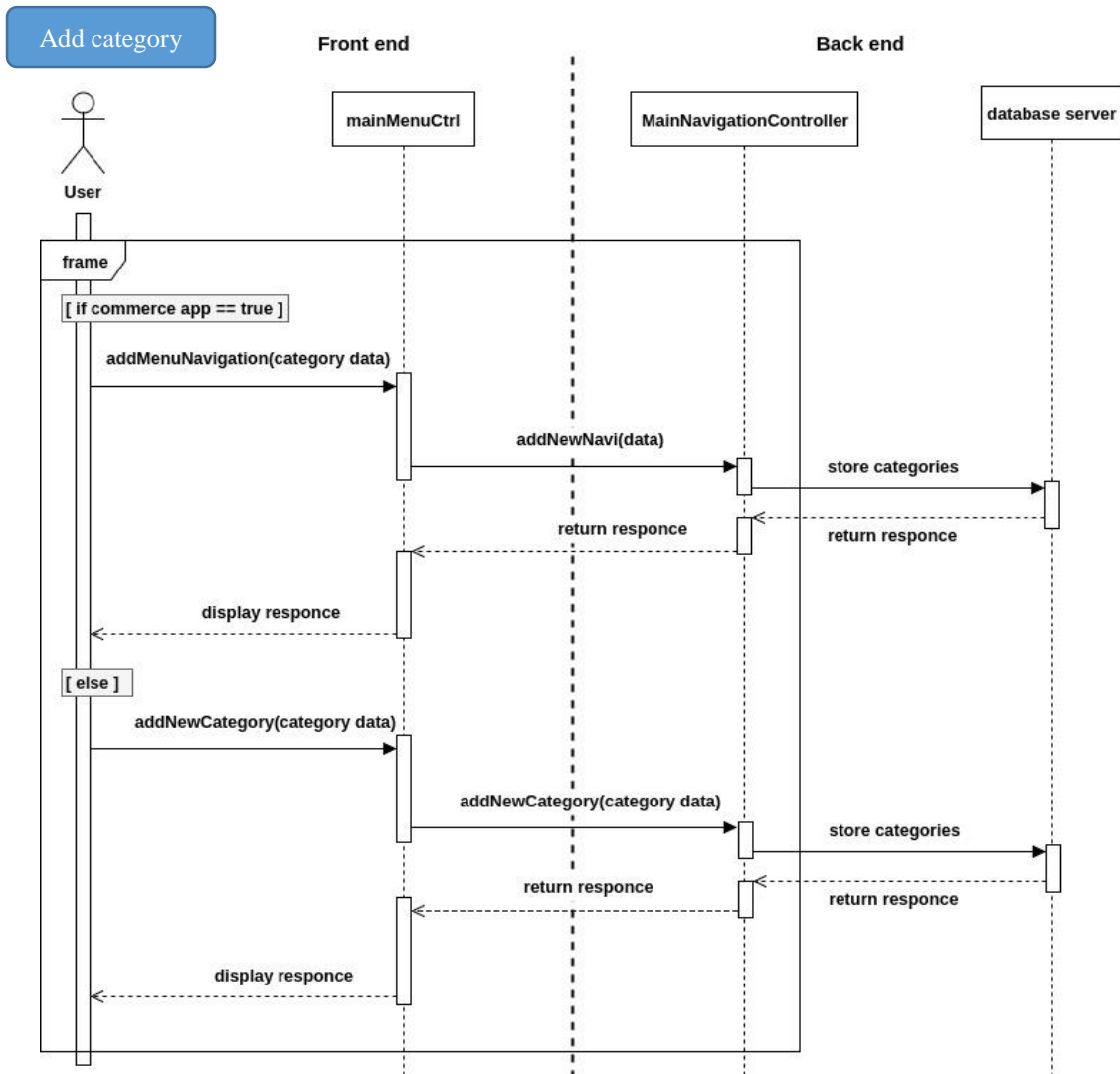
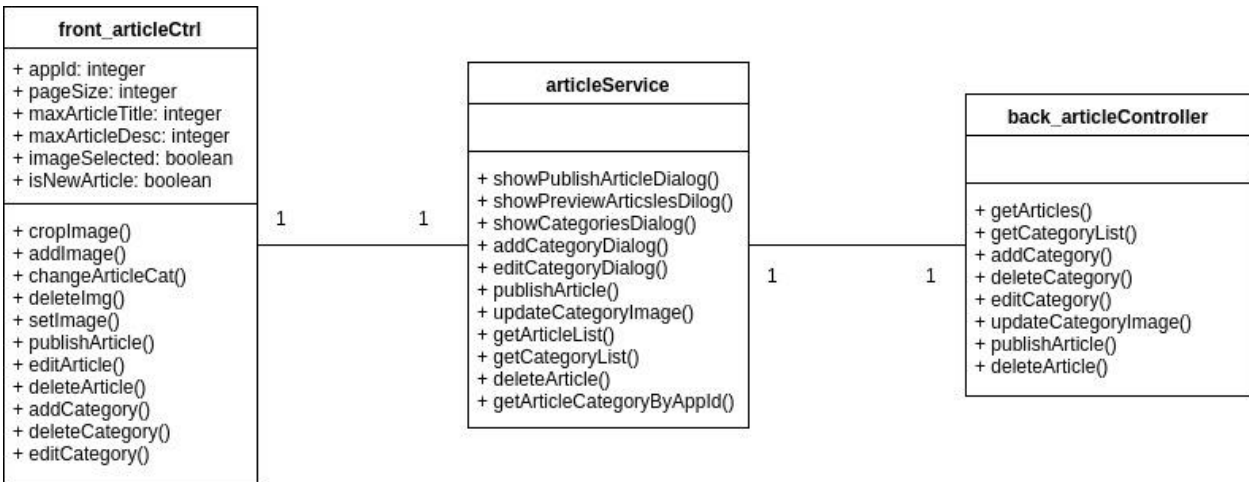


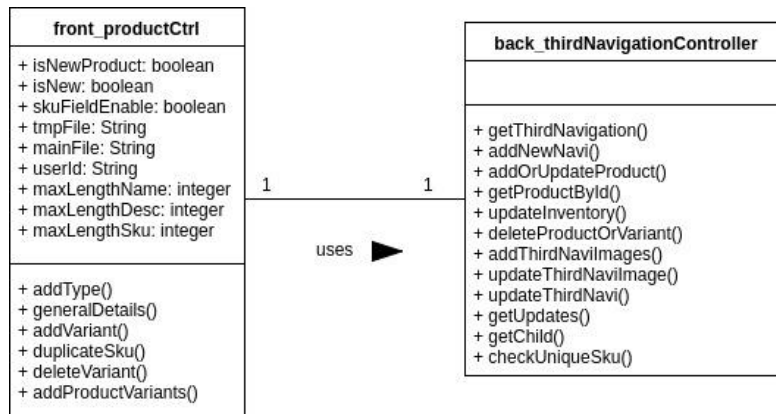
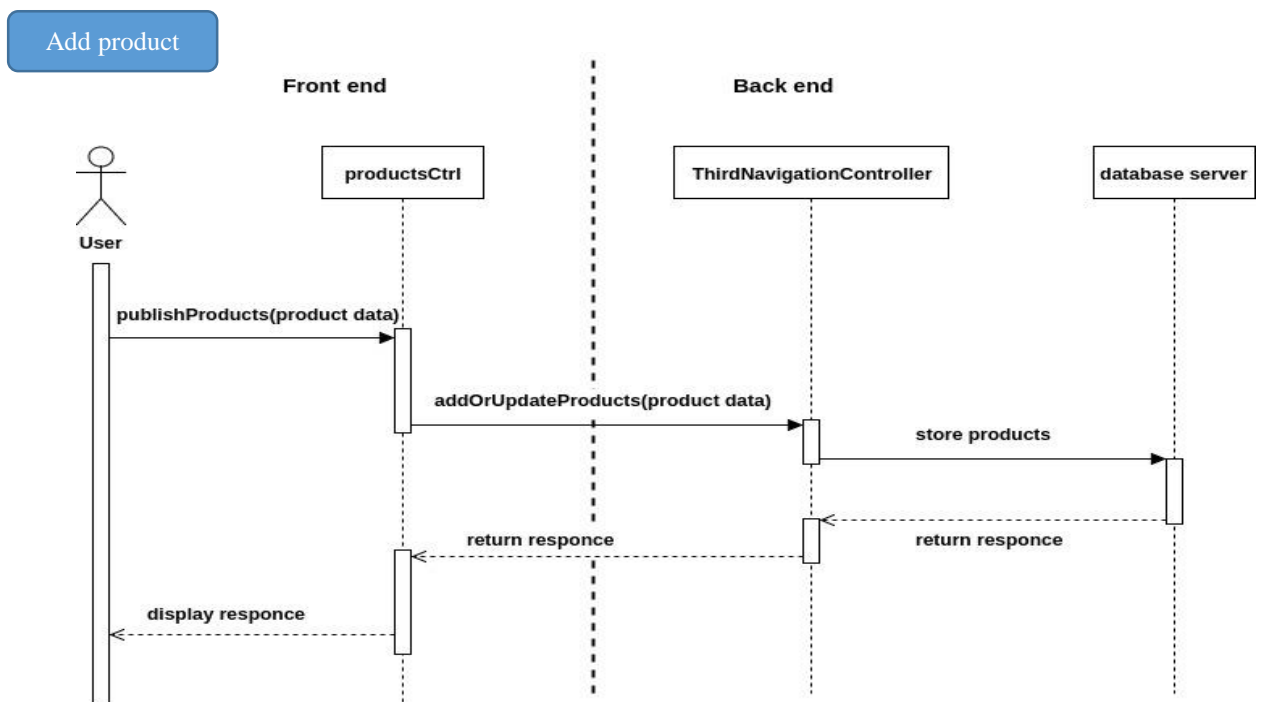
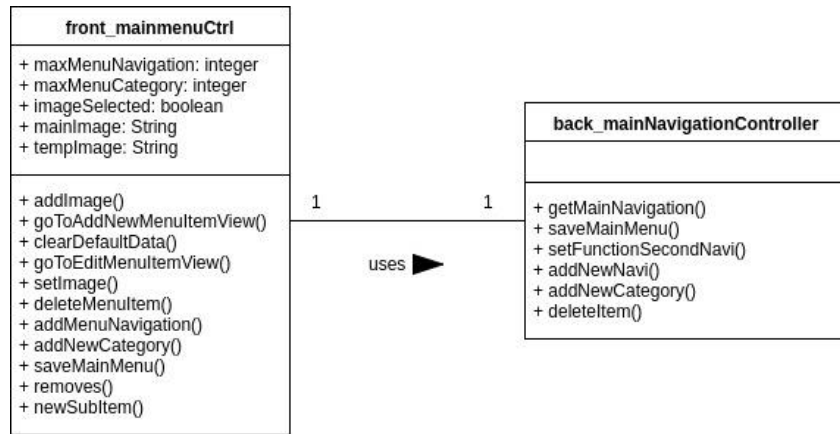
Manage Article



Add Article



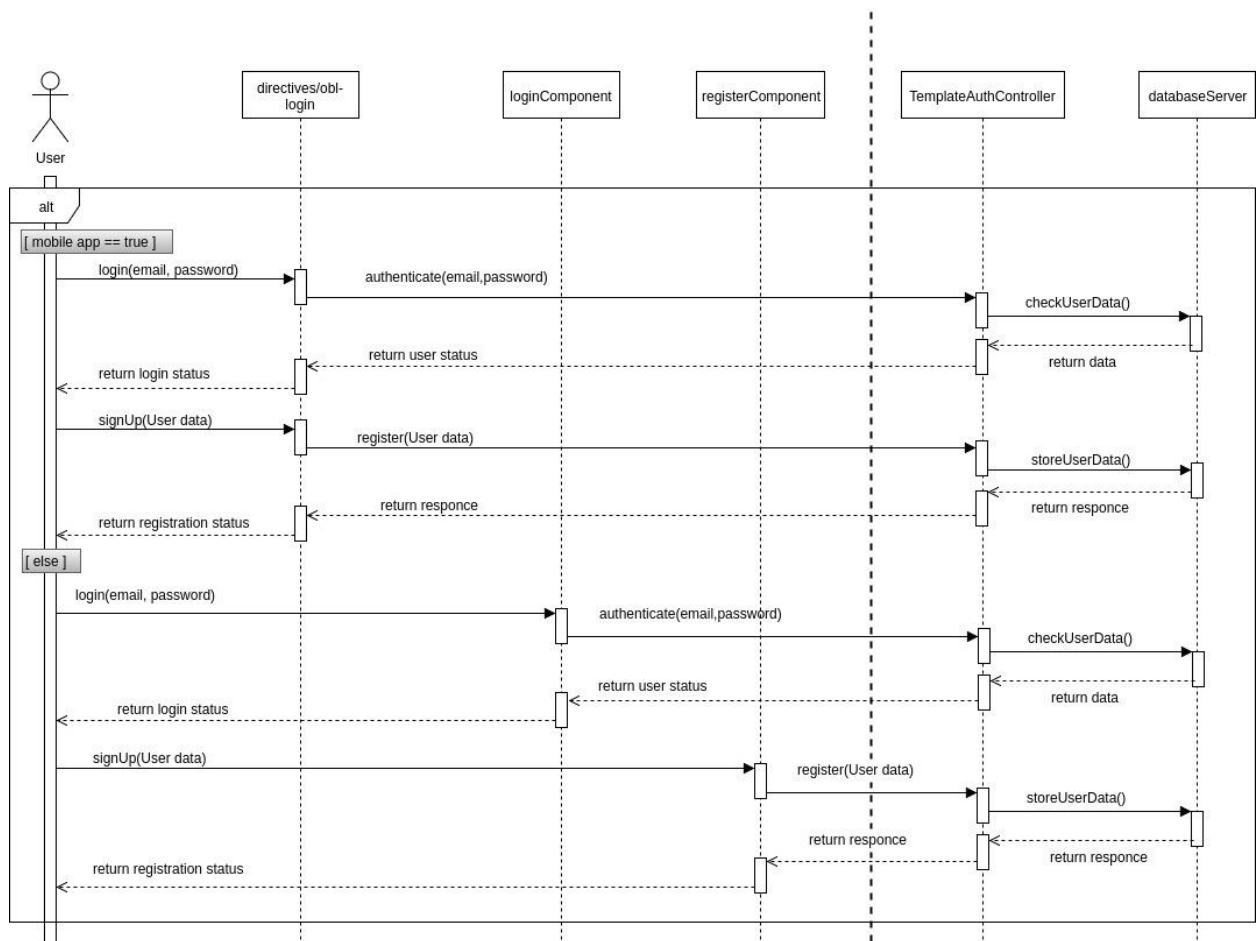




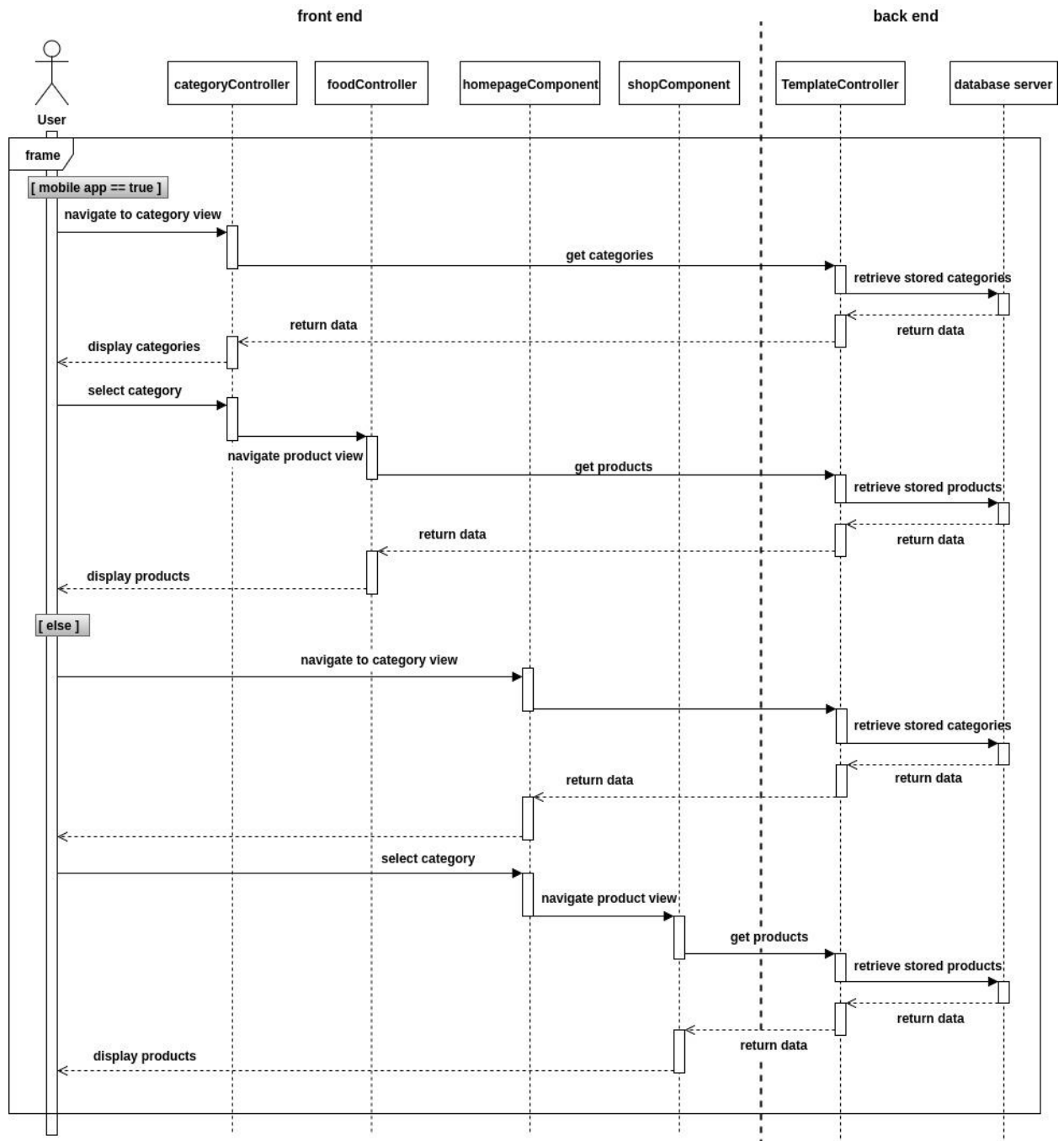
4.2. Application Layer – App user

This implementation derives both mobile and web applications. Mobile applications are build using AngularJS and Web Applications are implemented using Angular4 framework. Also the API requests are manages by the “Otenro” system which runs using a sails server.

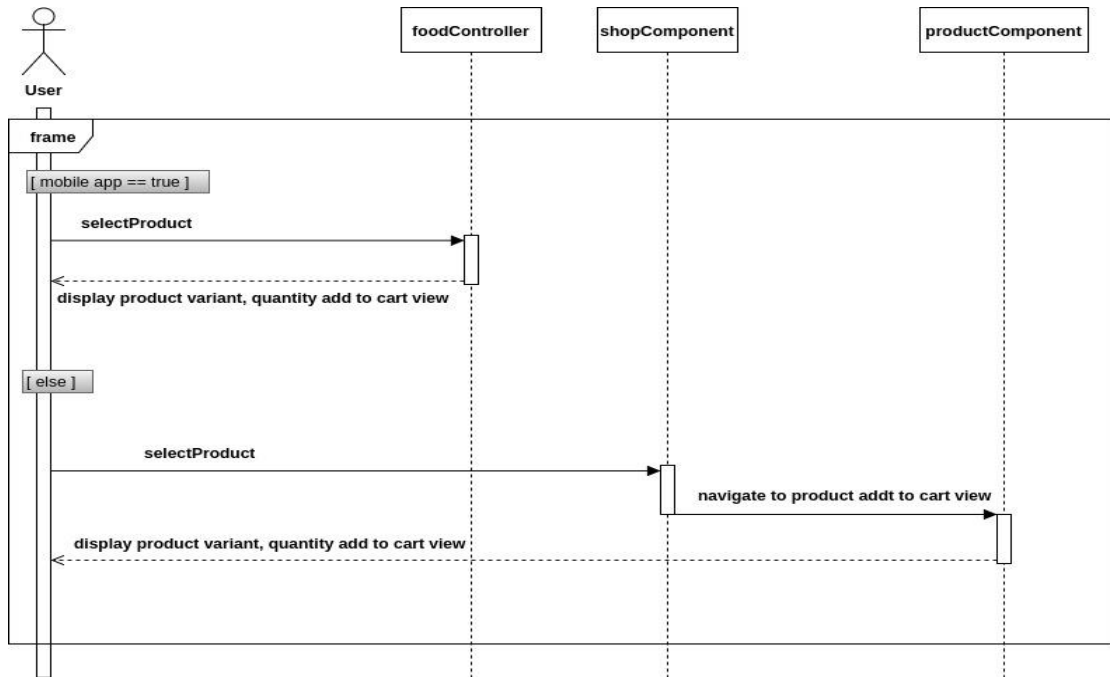
App user Login



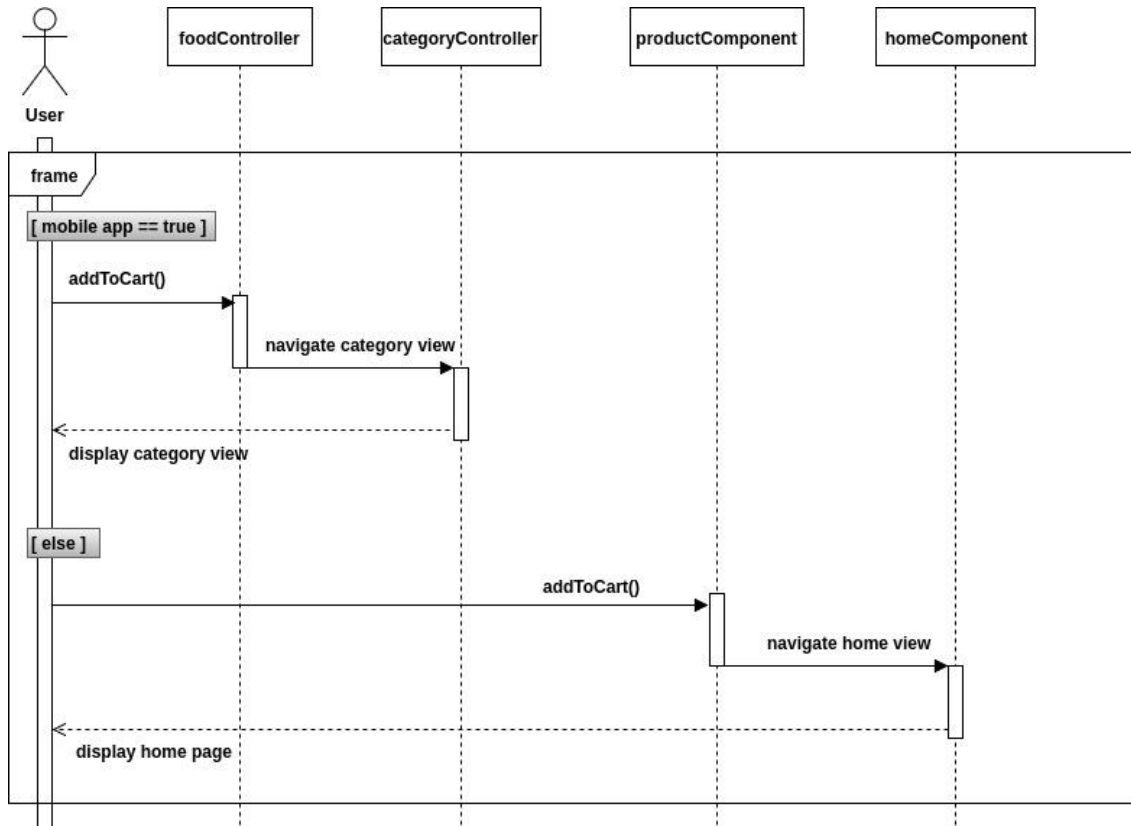
Select categories



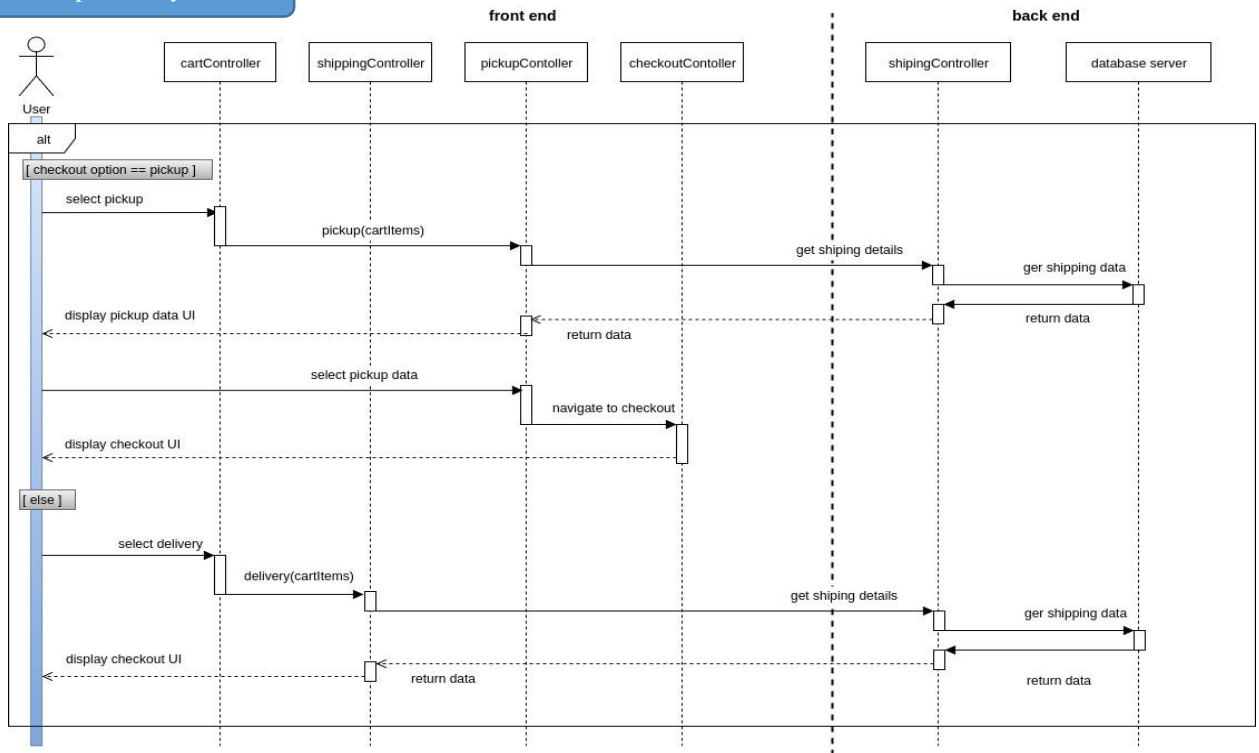
Select product



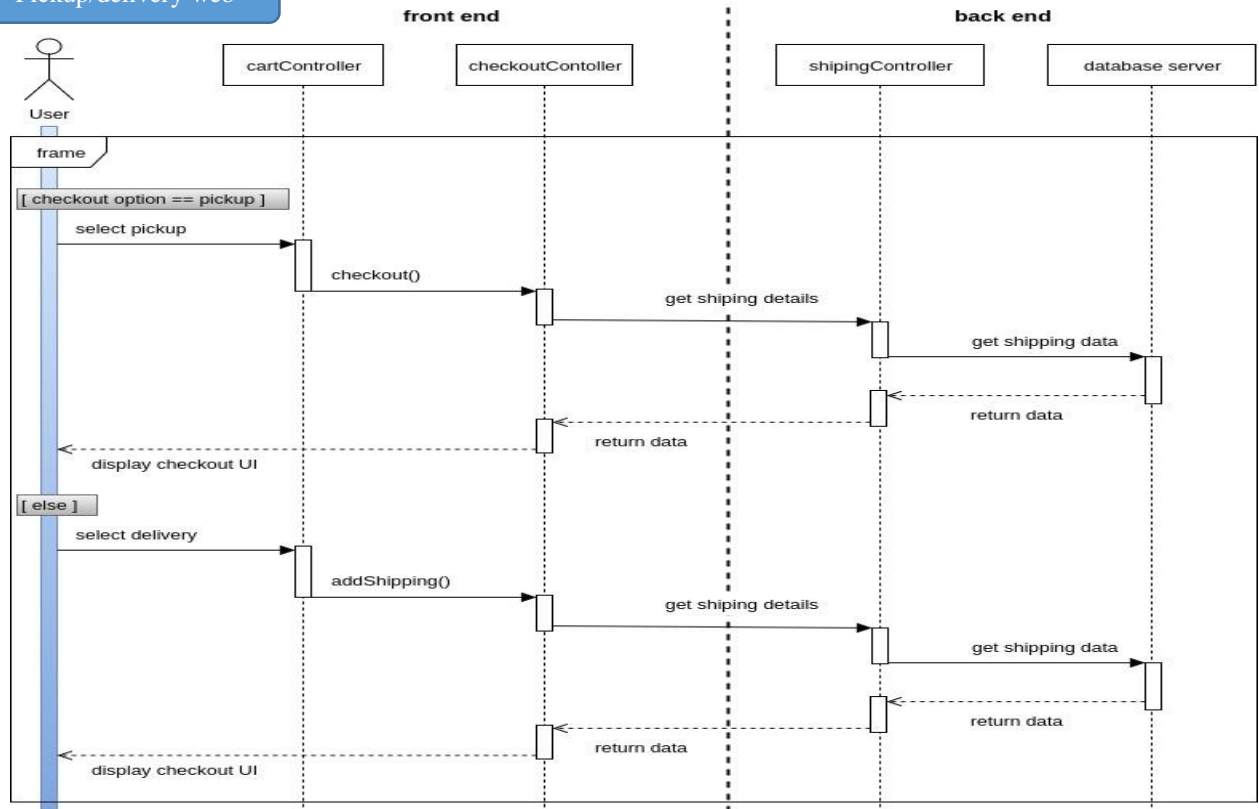
Add to cart

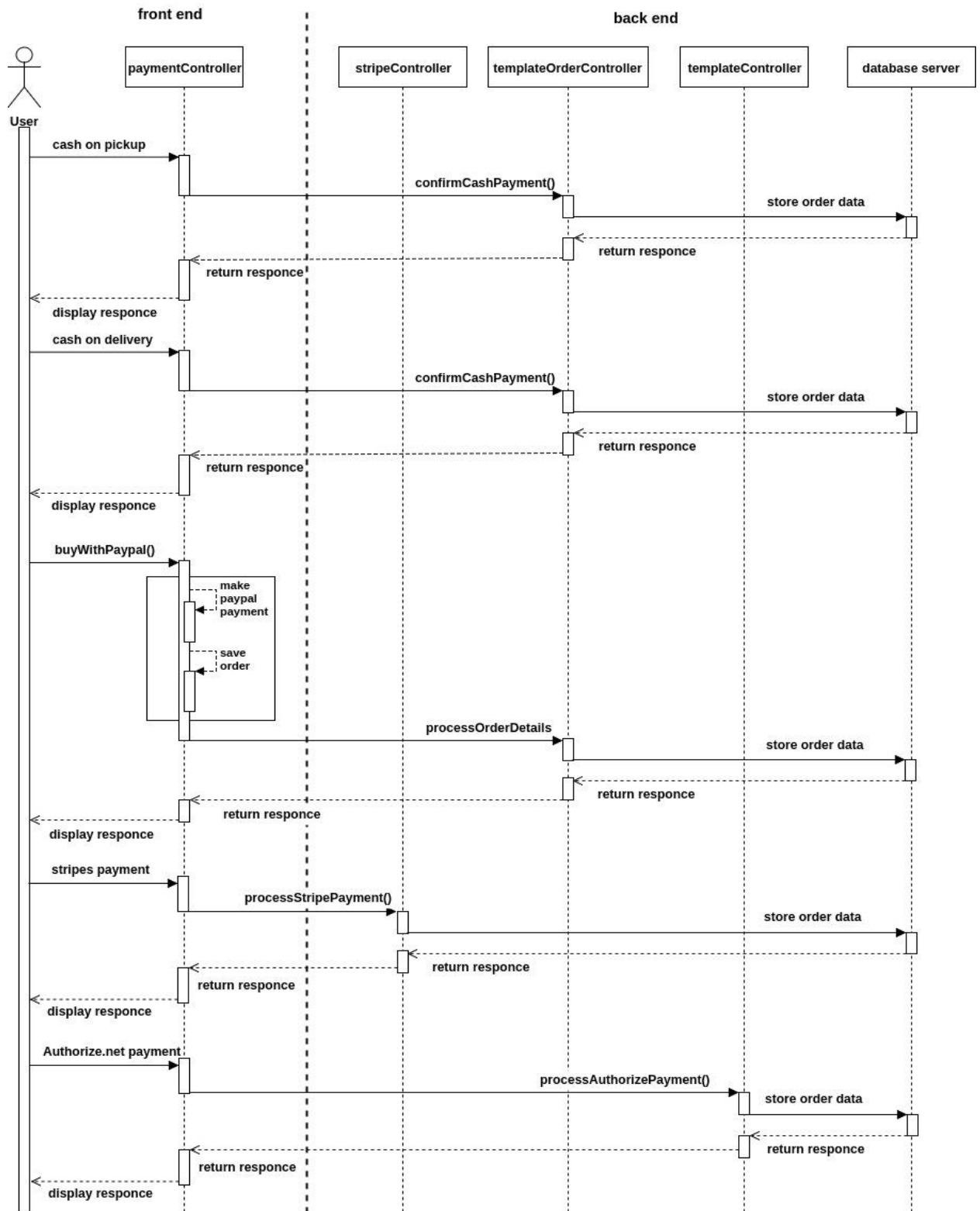


Pickup/delivery mobile

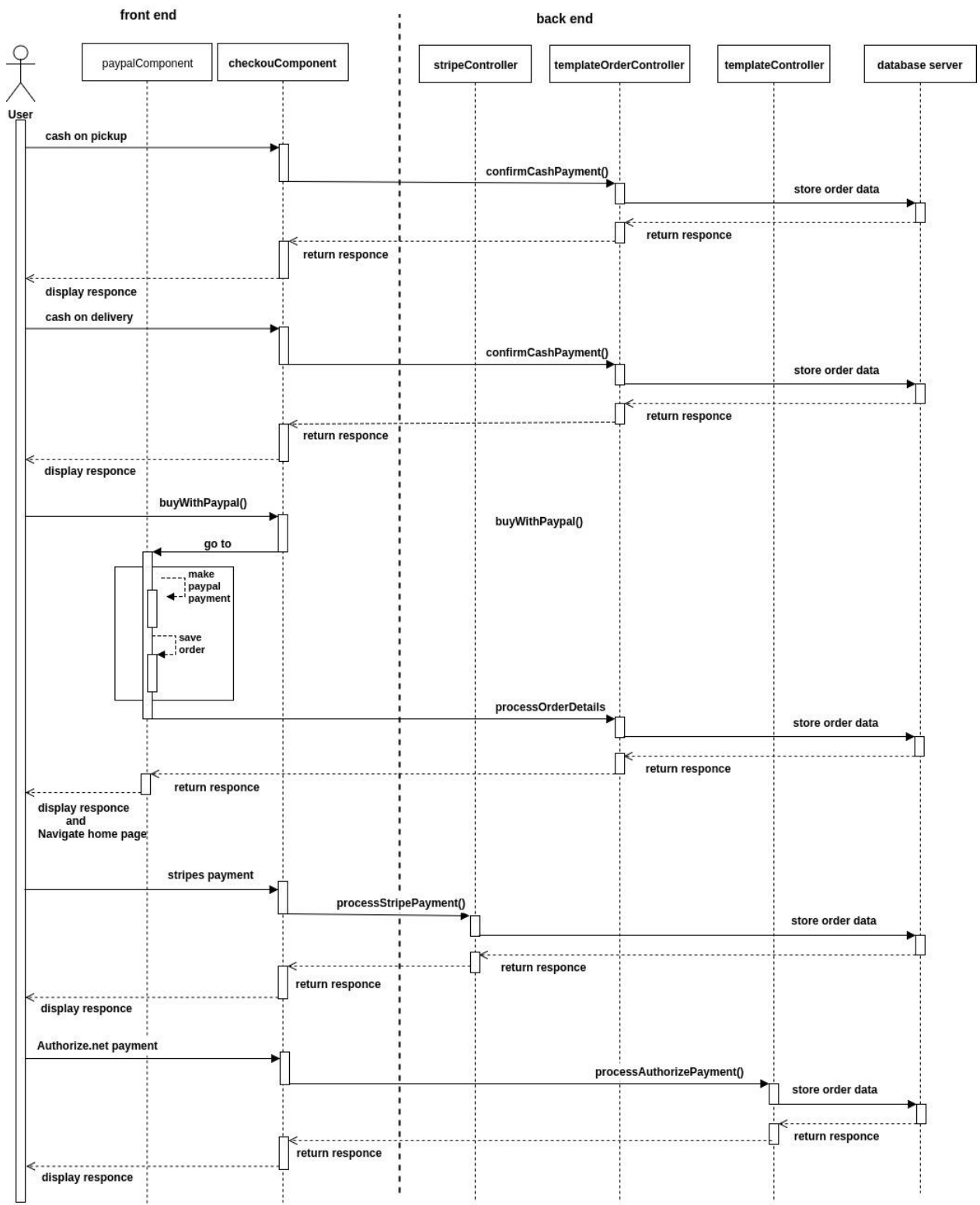


Pickup/delivery web

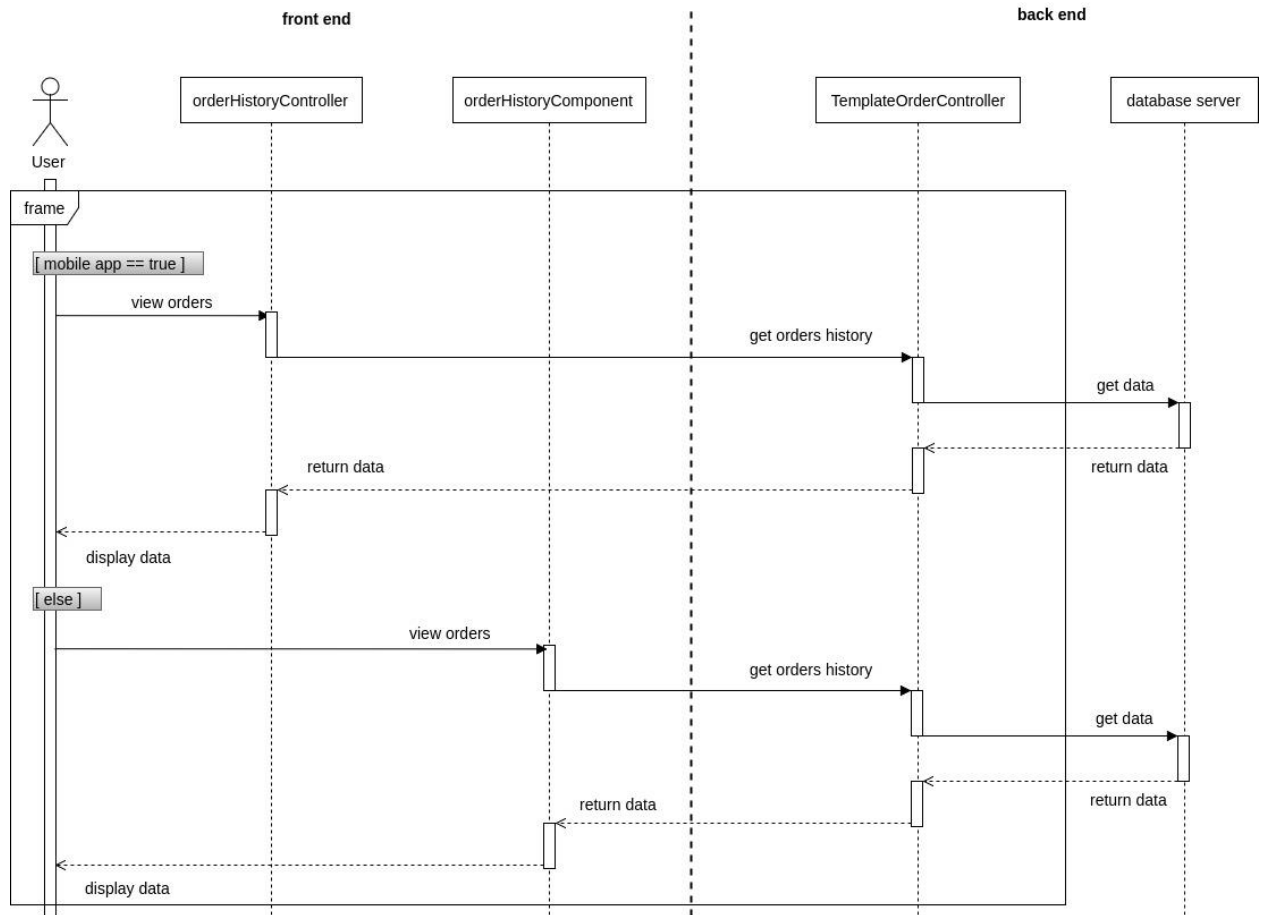




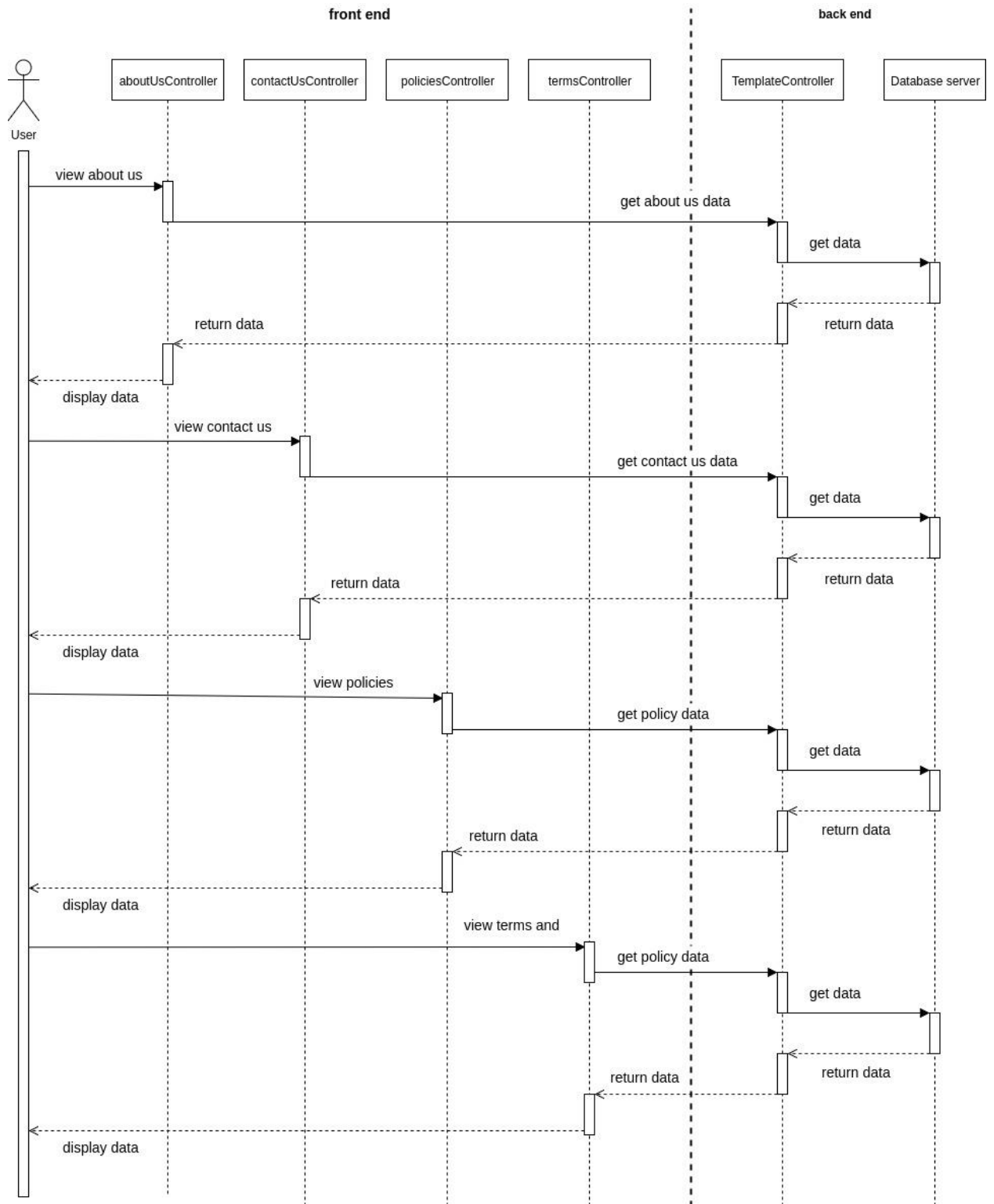
Payment Web

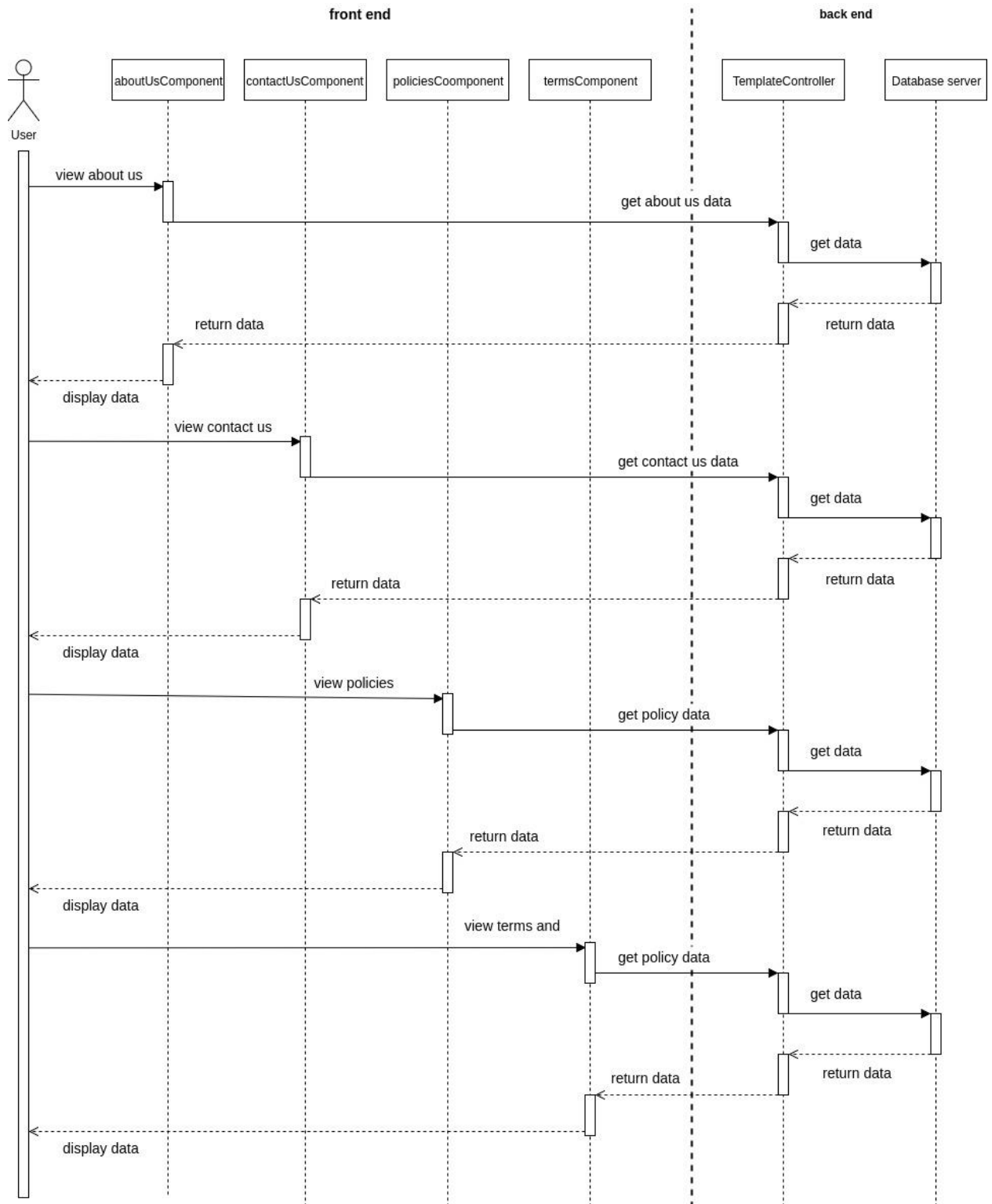


Orders details



App details Mobile





5 Database

The database utilized by the Otenro web application is MySQL 5. No user can create new tables through the Otenro system. However, anyone with access rights to the DB server may edit the current MongoDB script files to add new tables, create new MongoDB script files or simply access the MongoDB-Otenro database through a command prompt.

