

# FOODIE APP

In this project we have designed a food ordering app called “Foodie” in Java. This project fulfilled the concepts of:

1. Abstract Factory Pattern
2. Decorator Pattern
3. Observer Pattern
4. MVC Pattern
5. Java Swing
6. File I/O (XML)

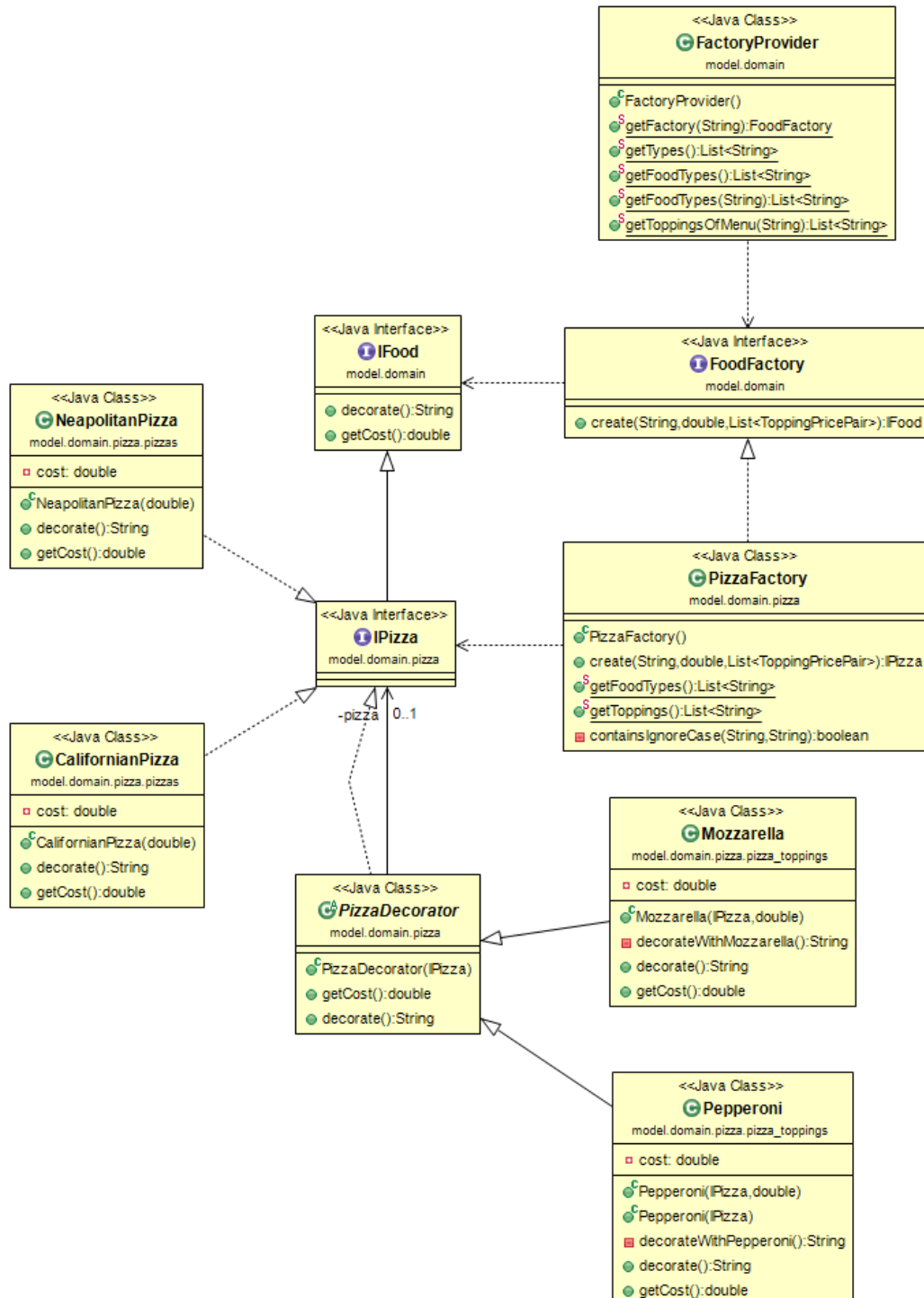
In this application, abstract factory pattern is used to create the food with its toppings by utilizing the decorator pattern. The factory provider takes the food type as a string to create either a PizzaFactory or a HamburgerFactory (can be expanded). It also takes the toppings as a list of strings which is then given to the PizzaFactory or the HamburgerFactory. These two factories create the given food which can be different types such as Neapolitan, New York Style etc. And then they go through the list of toppings to wrap the created IFood item in a decorator pattern. In the end, the created IFood item is returned all the way. The complete food string can be obtained by using the decorate method of IFood. Total cost can be obtained by using the getCost method. Both of these methods work by utilizing the decorator pattern.

There are two user types and both of these extend the User class with some added properties and methods. The customer has the currentOrder and the restaurant has the menu as the extra properties. The restaurant is able to manipulate the menu by creating new menu items, editing existing menu items or removing menu items. The customer is able to initialize the order, add an item to order, remove an item from order and place the order. Both of these classes utilize the observer pattern to let the views and controllers know about the changes happening to the user/restaurant information and menu items.

There is a Repository class that keeps track of all the objects in the current state of the program. This class uses the IO class to input and output the objects. The IO class uses the IOParser class to parse the objects from and to XML format. For any changes to the users etc. the FoodieService is used as an abstraction for the controllers as this is the only class with access to the Repository. After every change to the objects, the outputData method is called which calls the output method of the Repository.

View classes are used to render the data and buttons. The functionality to these buttons are added by that view's respective controller. Both the view and controller classes are observers of the user classes that are the subjects of the current window so that any changes will trigger an update immediately.

## DOMAIN CLASS DIAGRAM



## VIEWS



### Restaurants View (Customer's Perspective)

<div><b>Foodie</b> <b>Restaurants</b></div> <div><div>Restaurants</div><div>My Profile</div><div>Shopping Cart</div><div>Logout</div></div>	<div>Izzy's BBQ Smokehouse</div> <div>Tom's</div>
---	---


### Profile View (Customer's Perspective)

<div><b>Foodie</b> <b>My Profile</b></div> <div><div>Restaurants</div><div>My Profile</div><div>Shopping Cart</div><div>Logout</div></div>	<div><b>Name:</b> Jermaine Cole</div> <div>Change Name</div> <div><b>Username:</b> jcole</div> <div>Change Username</div> <div><b>Address:</b> New York Ave, Brooklyn, NY 11203, USA</div> <div>Change Address</div> <div><b>ORDER HISTORY</b></div> <div><div><div><b>Restaurant:</b> calipso</div><div><b>Order Date:</b> Fri Jan 15 09:45:41 TRT 2021</div><div><b>Order Address:</b> adress</div><div><b>Total Cost:</b> \$8.10</div><div><b>Items:</b></div><div><div><b>Item:</b> Big Mac Hamburger + Mayonnaise + Cheese + Pickles</div><div><b>Cost:</b> \$3.10</div></div><div><div><b>Item:</b> Neapolitan Pizza + Pepperoni + Mushroom + Olive</div><div><b>Cost:</b> \$5.00</div></div></div></div> <div><div><b>Restaurant:</b> calipso</div><div><b>Order Date:</b> Fri Jan 15 09:45:41 TRT 2021</div><div><b>Order Address:</b> adress</div></div>
--	---

## Restaurant View (Customer's Perspective)

<div><b>Foodie</b> Restaurant</div> <div><a href="#">Restaurants</a></div> <div><a href="#">My Profile</a></div> <div><a href="#">Shopping Cart</a></div> <div><a href="#">Logout</a></div>	<div><b>Restaurant Name:</b> Izzy's BBQ Smokehouse</div> <div><b>Restaurant Address:</b> Troy Ave, Brooklyn, NY 11213, USA</div> <div><b>MENU</b></div> <div><div><b>Pizza Menu</b></div><div></div><div><b>Californian Pizza \$4.2</b></div><div><b>Available Toppings:</b> Pepperoni, Mozzarella, Mushroom, Olive, Corn</div><div></div></div>
---	---


## Food View (Customer's Perspective)

<div><b>Foodie</b> Food</div> <div><a href="#">Restaurants</a></div> <div><a href="#">My Profile</a></div> <div><a href="#">Shopping Cart</a></div> <div><a href="#">Logout</a></div>	<div></div> <div><b>Californian Pizza</b></div> <div><b>Cost: \$4.2</b></div> <div><div><b>Pepperoni \$0.2</b> <input type="radio"/> Yes <input checked="" type="radio"/> No</div><div><b>Mozarella \$0.2</b> <input type="radio"/> Yes <input checked="" type="radio"/> No</div><div><b>Mushroom \$0.3</b> <input type="radio"/> Yes <input checked="" type="radio"/> No</div><div><b>Olive \$0.1</b> <input type="radio"/> Yes <input checked="" type="radio"/> No</div><div><b>Corn \$0.6</b> <input type="radio"/> Yes <input checked="" type="radio"/> No</div><div><a href="#">Add To Cart</a></div></div>
---	--

## Shopping Cart (Customer's Perspective)

Foodie Restaurant	
<a href="#">Restaurants</a>	<div><p><b>Restaurant:</b> Izzy's BBQ Smokehouse</p><p><b>Order Address:</b> New York Ave, Brooklyn, NY 11203, USA</p><p><b>Total Cost:</b> \$4.40</p><p><b>Items:</b></p><div><p><b>Item:</b> Californian Pizza + Pepperoni</p><p><b>Cost:</b> \$4.40</p></div><p><a href="#">Place Order</a></p></div>
<a href="#">My Profile</a>	
<a href="#">Shopping Cart</a>	
<a href="#">Logout</a>	

## Restaurant Profile View (Restaurant's Perspective)

Foodie Restaurant Profile	
<a href="#">Restaurant Profile</a>	<p><b>Restaurant Name:</b> Izzy's BBQ Smokehouse</p> <p><a href="#">Change Name</a></p> <p><b>Restaurant Username:</b> izzy</p> <p><a href="#">Change Username</a></p> <p><b>Restaurant Address:</b> Troy Ave, Brooklyn, NY 11213, USA</p> <p><a href="#">Change Address</a></p> <p><b>MENU</b> <a href="#">Create New Menu</a></p>
<a href="#">Order History</a>	<div><p><a href="#">Add Item To Menu</a> <b>Pizza Menu</b> <a href="#">Remove This Menu</a></p><div></div><p><a href="#">Remove This Item</a></p></div>
<a href="#">Logout</a>	

## Food View (Restaurant's Perspective)


### Foodie

#### Food

Restaurant Profile

Order History

Logout



**Californian Pizza**

**Cost: \$4.2**

Change Food Cost

Add Topping

Pepperoni \$0.2

Change Topping Cost

Remove Topping

Mozarella \$0.2

Change Topping Cost

Remove Topping

Mushroom \$0.3

Change Topping Cost

Remove Topping

Olive \$0.1

Change Topping Cost

Remove Topping

## Order History View (Restaurant's Perspective)

### Foodie

#### Order History

Restaurant Profile

Order History

Logout

**Customer:** customer1

**Order Date:** Fri Jan 15 09:45:41 TRT 2021

**Order Address:** adress

**Total Cost:** \$8.10

**Items:**

**Item:** Big Mac Hamburger + Mayonnaise + Cheese + Pickles

**Cost:** \$3.10

**Item:** Neapolitan Pizza + Pepperoni + Mushroom + Olive

**Cost:** \$5.00

**Customer:** customer2

**Order Date:** Fri Jan 15 09:45:41 TRT 2021

**Order Address:** adress

**Total Cost:** \$9.80

**Items:**

**Item:** Turkey Burger + Mayonnaise + Cheese + Pickles + Tomato

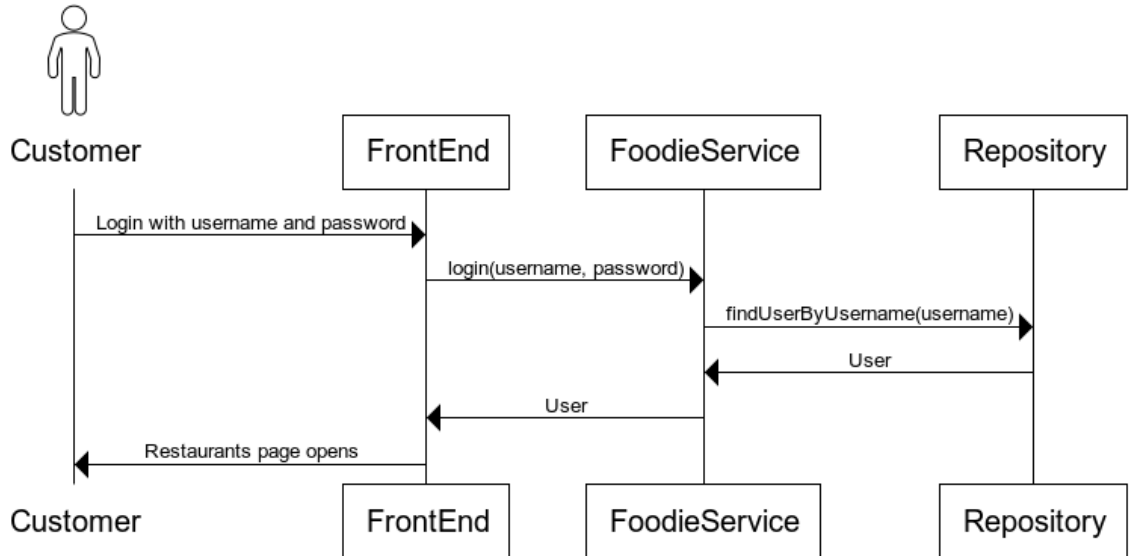
**Cost:** \$4.80

**Item:** Neapolitan Pizza + Pepperoni + Mushroom + Olive

**Cost:** \$5.00

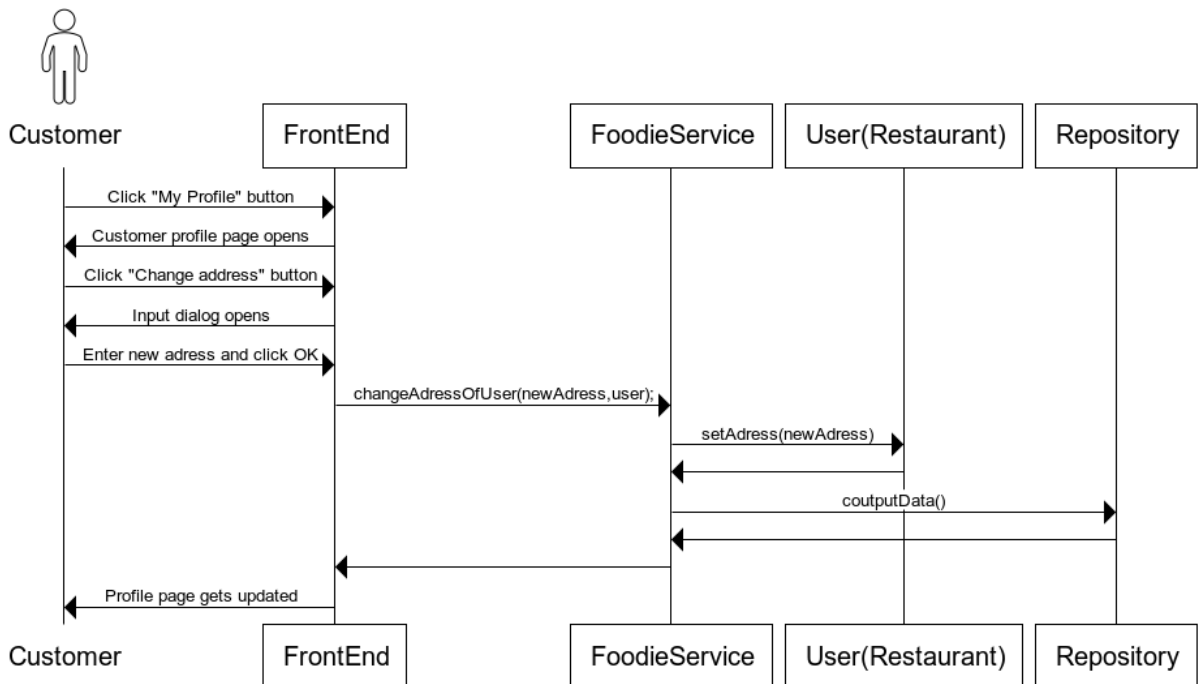
## SEQUENCE DIAGRAMS

### Login for customer



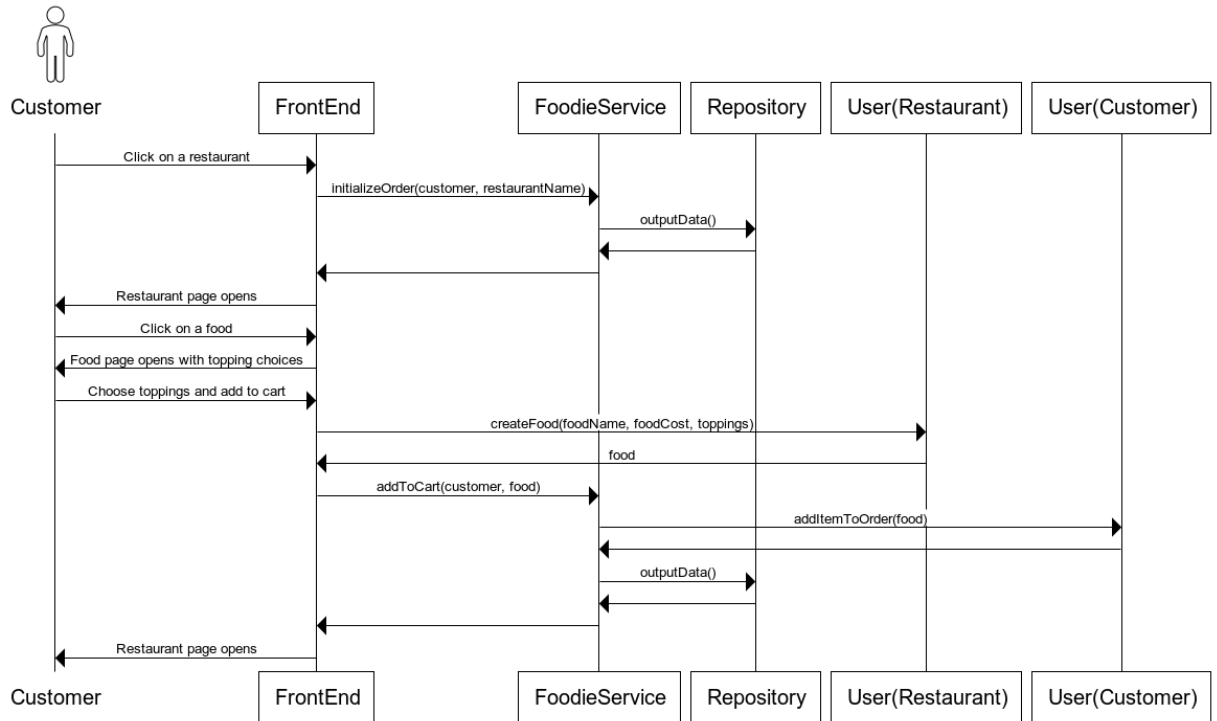
www.websequencediagrams.com

### Update Customer Info



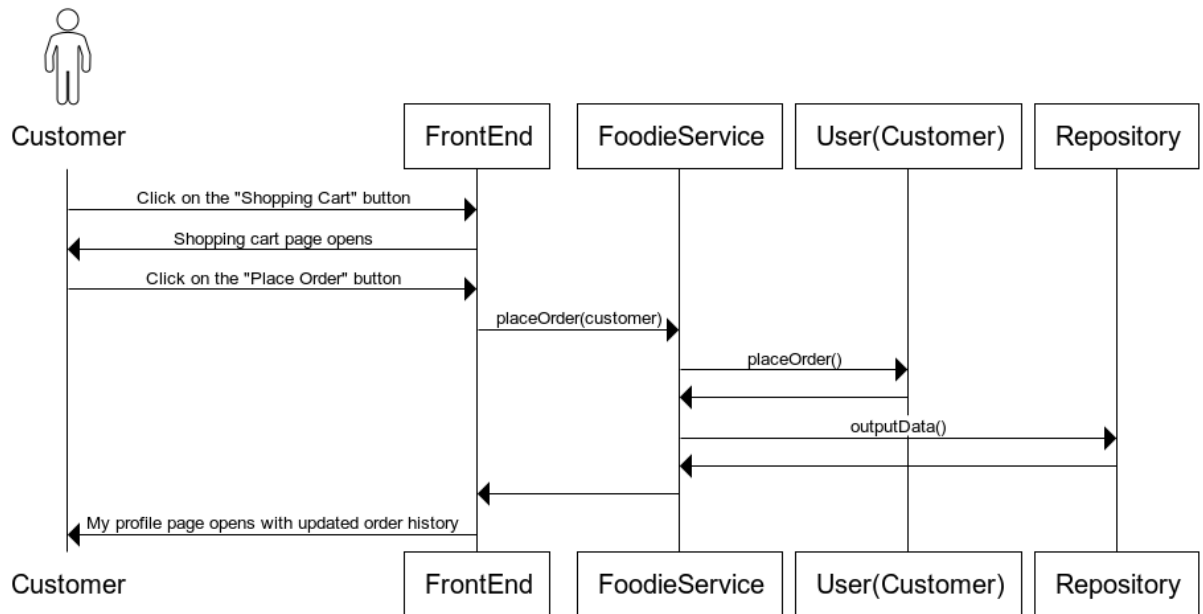
www.websequencediagrams.com

## Add food to cart



www.websequencediagrams.com

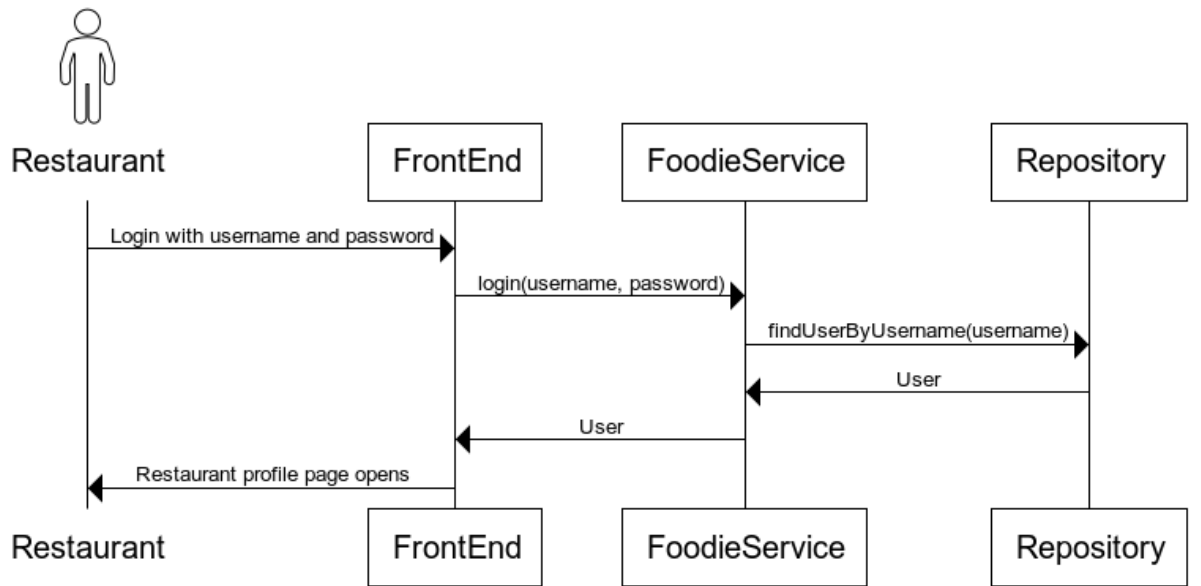
## Place order



www.websequencediagrams.com

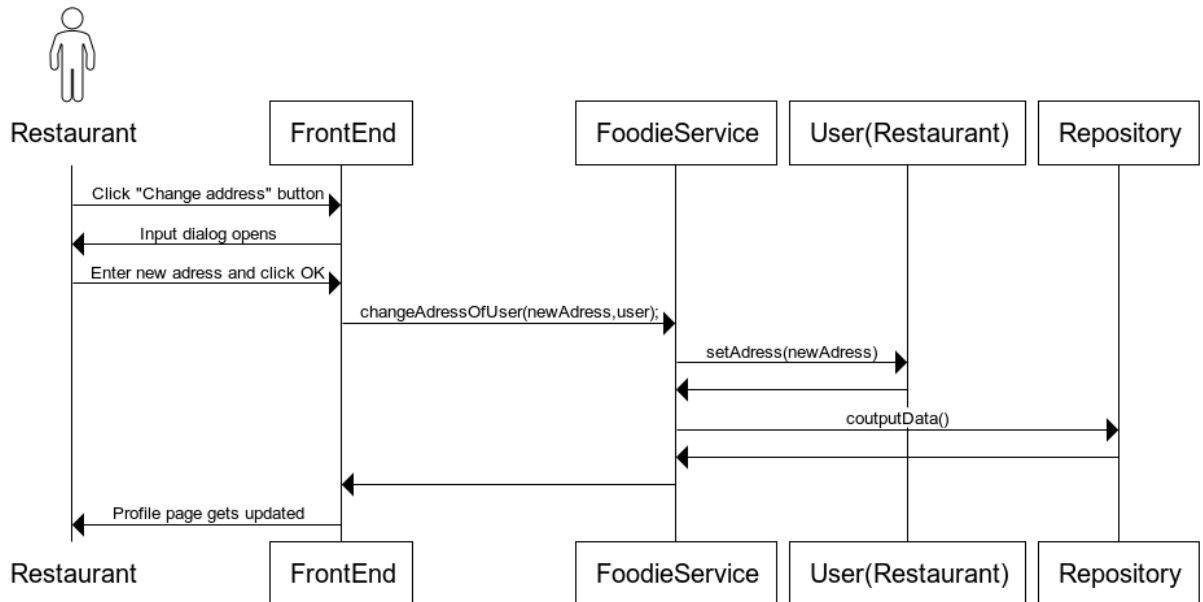


## Login for restaurant



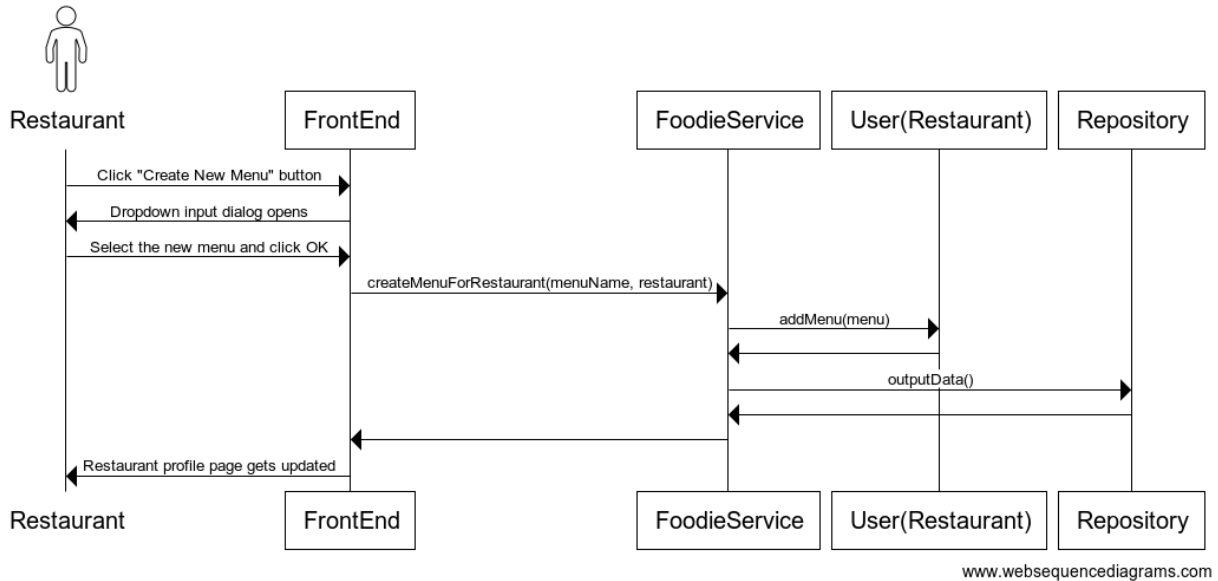
www.websequencediagrams.com

## Update Restaurant Info

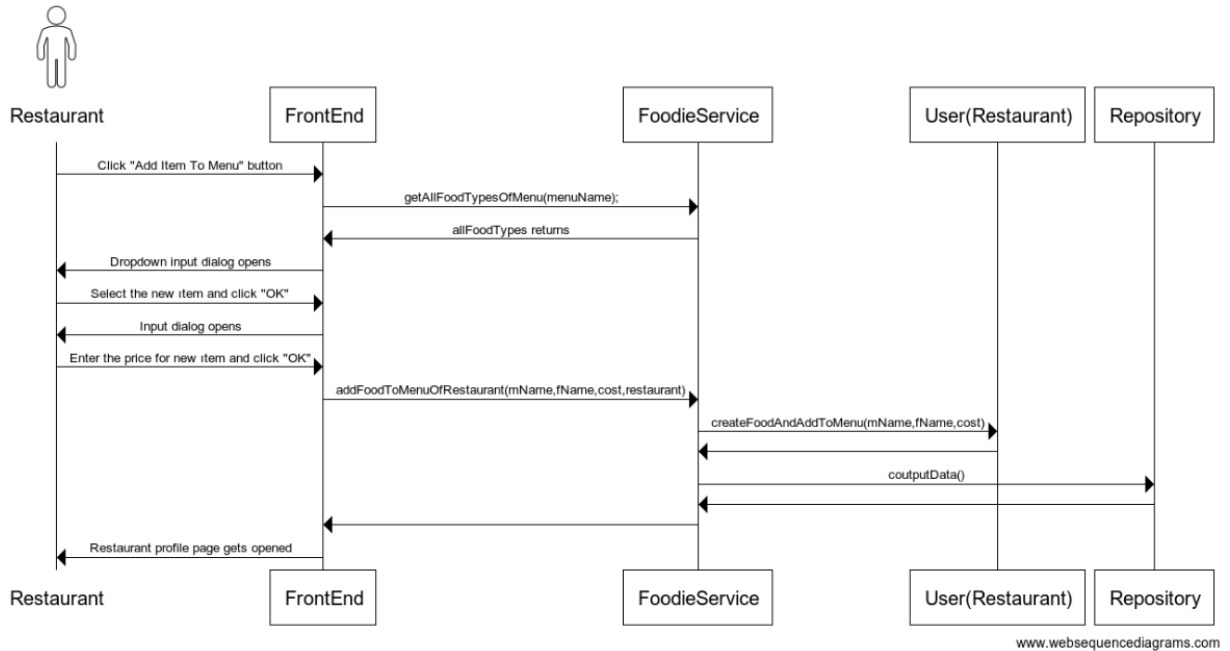


www.websequencediagrams.com

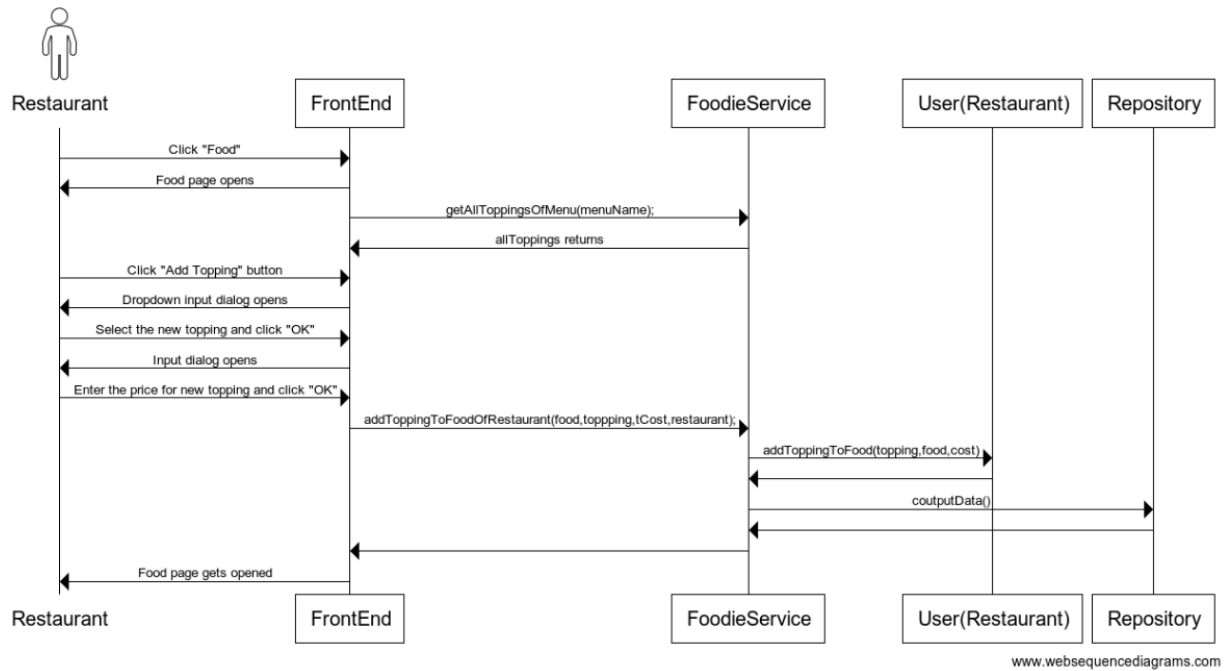
## Create new menu



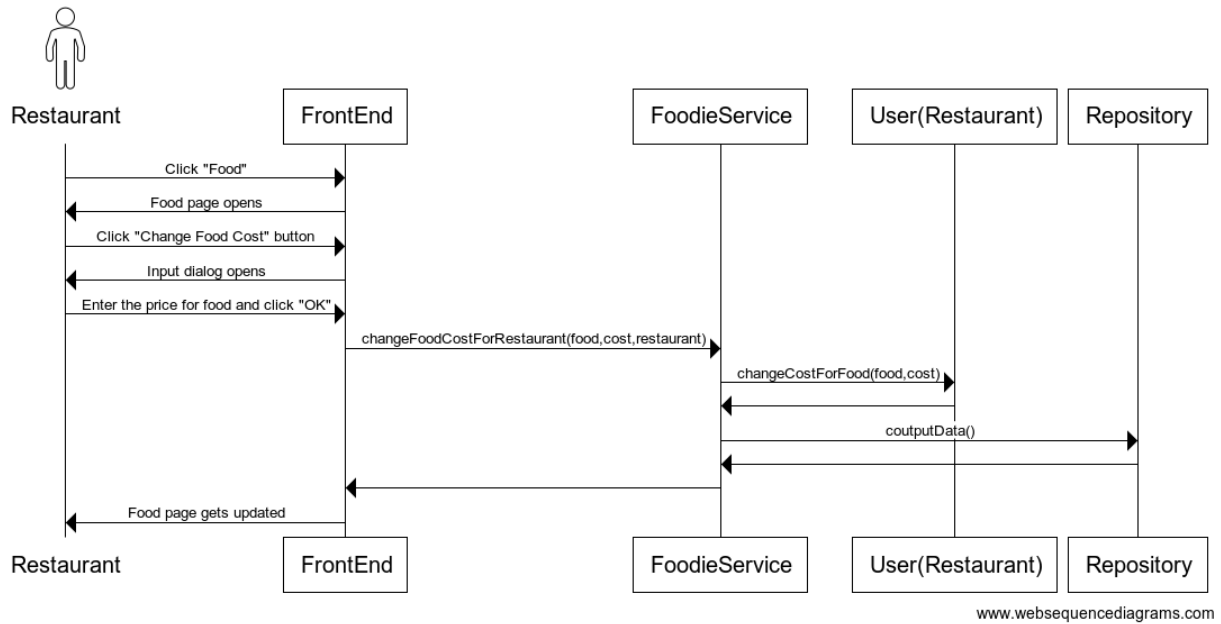
## Add Item To Menu



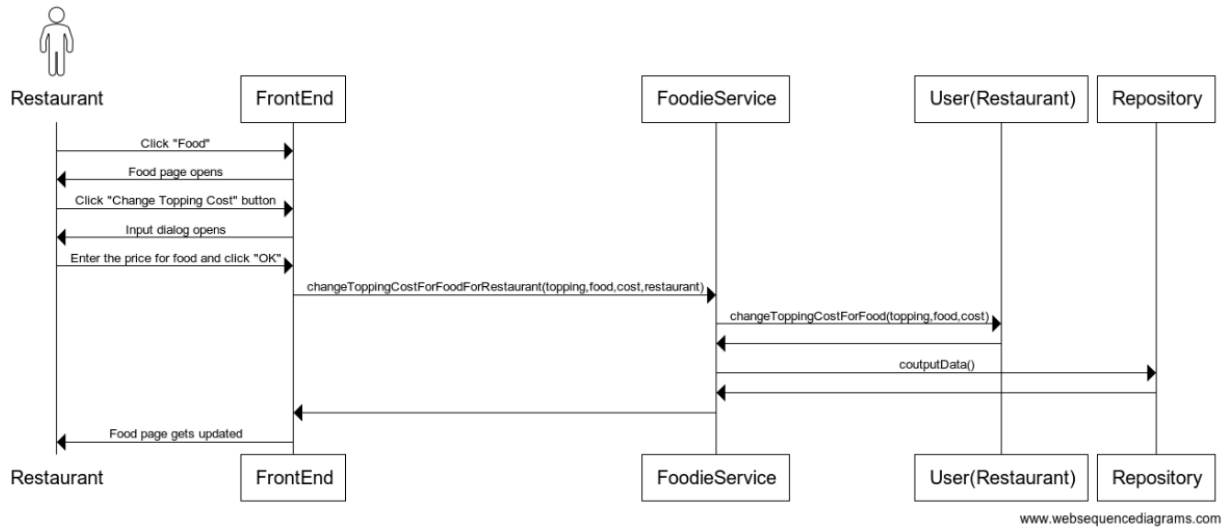
## Add Topping To Item



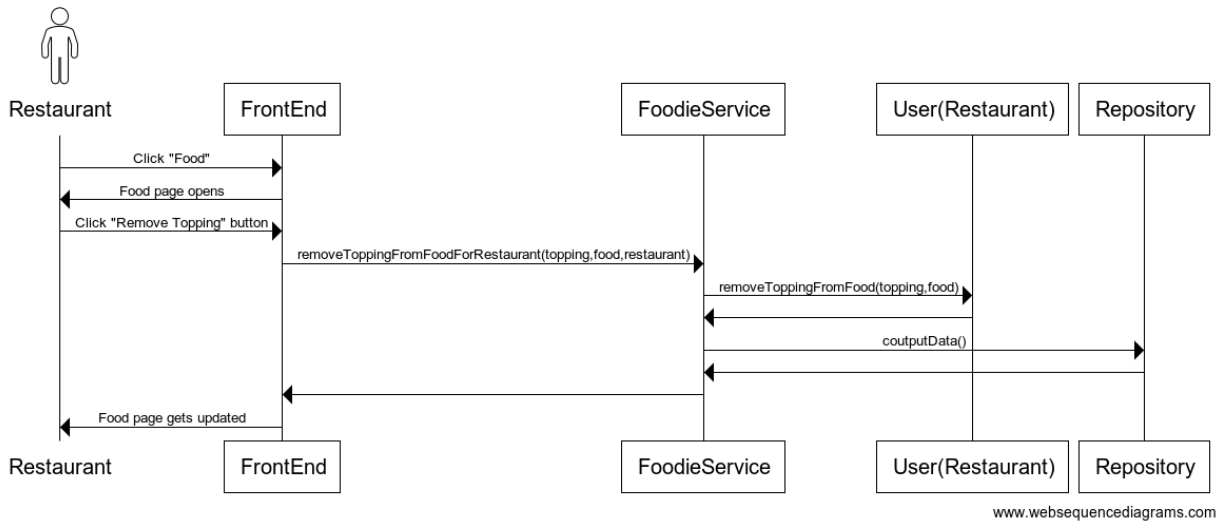
## Change Food Cost



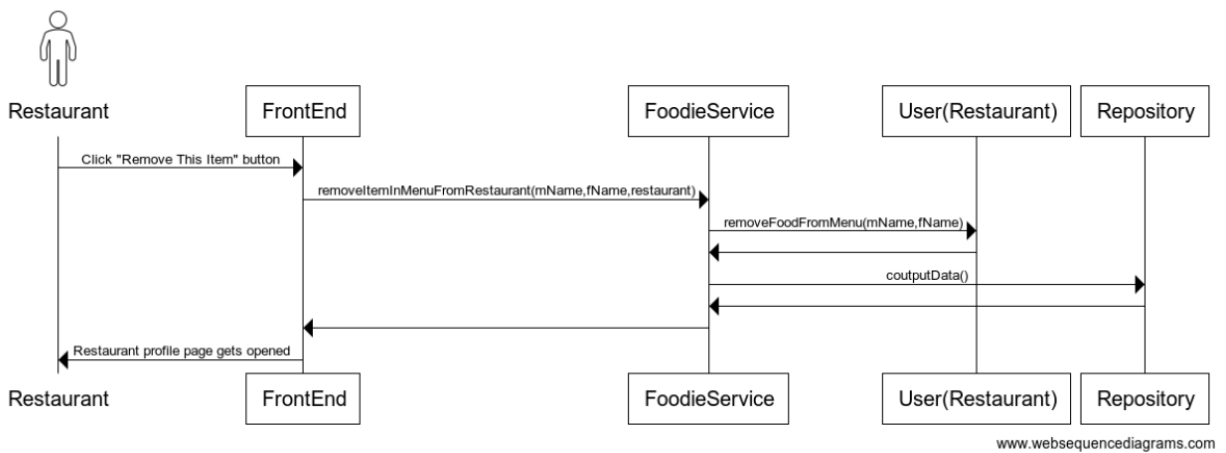
### Change Topping Cost



### Remove Topping



### Remove Item From Menu



## CLASS DIAGRAM

