

PIZZA ORDER SYSTEM

Name: Amrit Harikrishnan(845330)

TABLE OF CONTENTS

SL NO	CONTENTS	PAGE NO
1.	Abstract	3
2.	Introduction	4
3.	Requirement Specification	6
4.	Architecture Design	8
5.	Conclusion and Future Work	16
6.	References	17

ABSTRACT

“Just Eat Pizzas” is a web application which allows customers to place an order for Pizza and display the order details. Once the customer logs in, he/she has the options to Place, Modify, Delete and Display the order. The order consists of a base price for the pizza and a single selection of toppings from the given options. Thus an order is created with the calculated total price and personal details including the physical address, phone number & name provided by the customer along with an order date. The scope of the project is not extended to the payment stage.

INTRODUCTION

“Just Eat Pizzas” is an online pizza ordering system where the customer can place an order once he/she logs in.

Project Objective:

The main objective of this project is to allow the customer to place a pizza order with the necessary topping. The customer can place the order only after successful logging in to the web application. Further features available for the customer are to delete the order, modify the topping and to view the placed order.

Project Scope:

The scope of the project is not extended to the payment and delivery stage. The project scope is limited to placing the order, displaying the order, deleting the order and modifying the topping selected. The customer can logout once the necessary operations are done.

Problem Definition:

In the earlier systems, the pizza can only be received by takeout counters in real time. To solve this problem, the online pizza ordering system has been made where the customer can place a pizza order online.

“Just Eat Pizzas” is a web application to order pizza online. For this the customer needs to login by providing the username and password credentials. The customer has the options to place an order, delete the already placed order, modify the placed order and see the placed order.

Placing the order requires the customer to enter the name, address, mobile number and select the appropriate topping from the toppings. The order id will be displayed once the order is placed. Deleting the order requires the customer to enter the order id, upon which the order is displayed. The order is deleted once the customer confirms the displayed order. Modifying the order requires the user to enter the order id, upon which the toppings section will appear. The customer

can only modify the toppings of that particular order. Finally, to view the order, the customer enters the order id, upon which the corresponding order is displayed. The customer can also logout once the desired transaction is completed.

The language used to code this web application is Java, HTML and CSS for front-end. The IDE used is eclipse and frameworks used are Spring MVC and Hibernate. Forms in the front-end are mostly spring forms except for the toppings selection. JSTL is also incorporated in the front-end. User-defined exception in the case of order id mismatch is also used. For back-end, MySQL Workbench is used to create the database design and Hibernate framework.

Requirement Specification

The project has four modules excluding login and logout. In login the user logs in by entering the username and password. Once the user logs in, he/she can do the following operations.

Place order: The user enters this module upon clicking the “place order” link. Here, the customer gives the name, address, phone number and the toppings for the pizza. The order id of the order is returned when the order is placed successfully. The customer can only select one topping and each topping has its own individual price which would be calculated by adding it to the base pizza price.

Delete order: The customer enters this module by clicking the “delete order” link. Here, the customer gives the order id to be deleted. The customer is then redirected to a page where the order is displayed. Upon confirmation, the order is deleted.

Modify order: The customer enters this module by clicking the “modify order” link. When the order id and topping is entered, the customer is redirected to success page if the order is modified.

Display order: The customer enters this module by clicking the “display order” link. When the order id is entered by the customer, the order details will be displayed which includes the order date, order id, customer id and price.

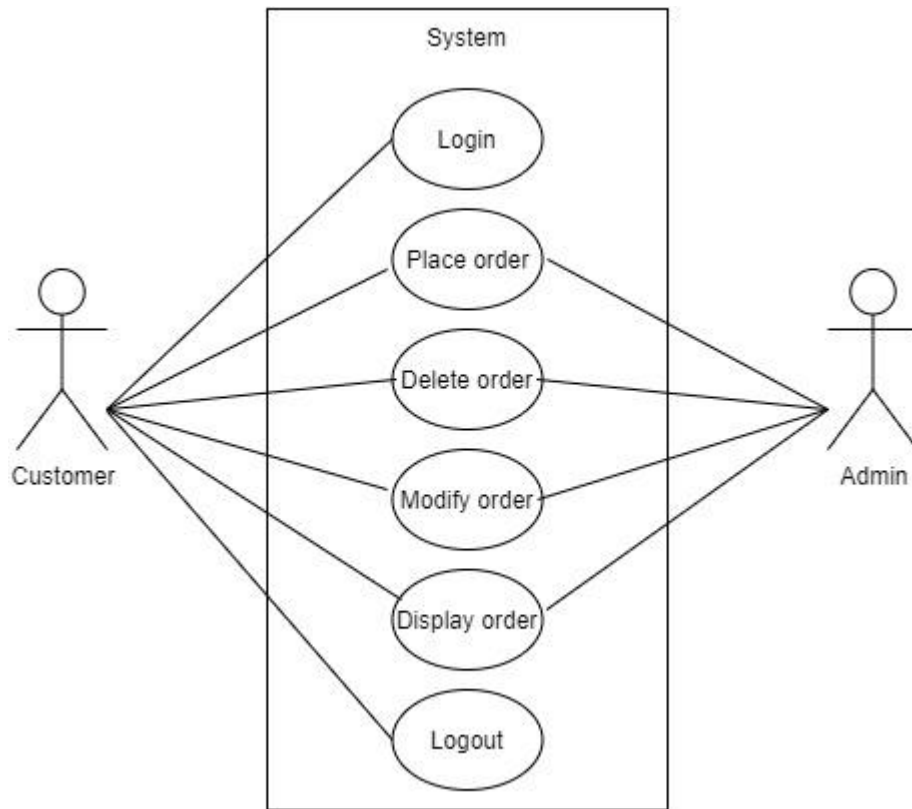
Hardware and Software Requirements

- Programming language: Java.
- Front-End: Html5, CSS3, JSP, JSTL.
- Databases: MySQL Workbench.
- OS: Windows 7 or higher.
- RAM: 4GB or higher.
- Browser: Internet Explorer or Chrome or Firefox.
- Web Server: Apache Tomcat 7 and higher.

- Tools: Eclipse IDE.
- Frameworks: Spring MVC and Hibernate.

Architecture Design

Use Case Diagram

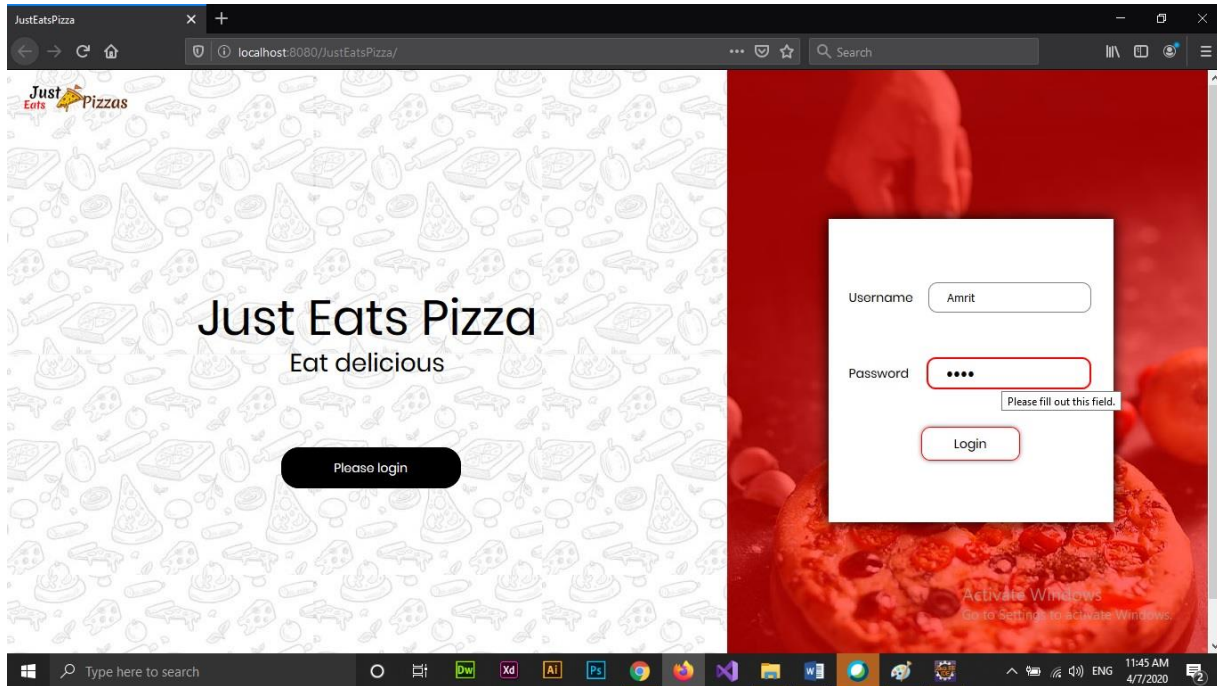


Customer: The customer can login, place order, delete order, modify the toppings, display order and logout.

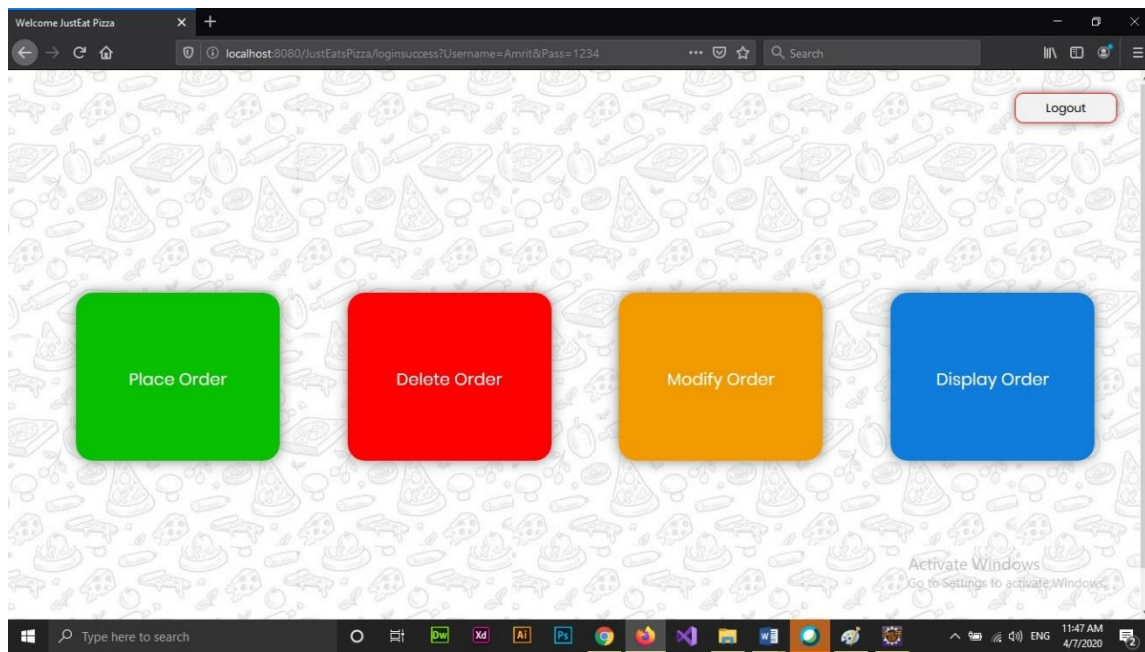
Admin: The admin has the control over all the database related operation that the user performs.

Project Modules

Login: The customer logs in by entering his/her username and password (screenshot 1). After logging in, the customer can access the next four modules (screenshot 1.1).



Screenshot 1



Screenshot 1.1

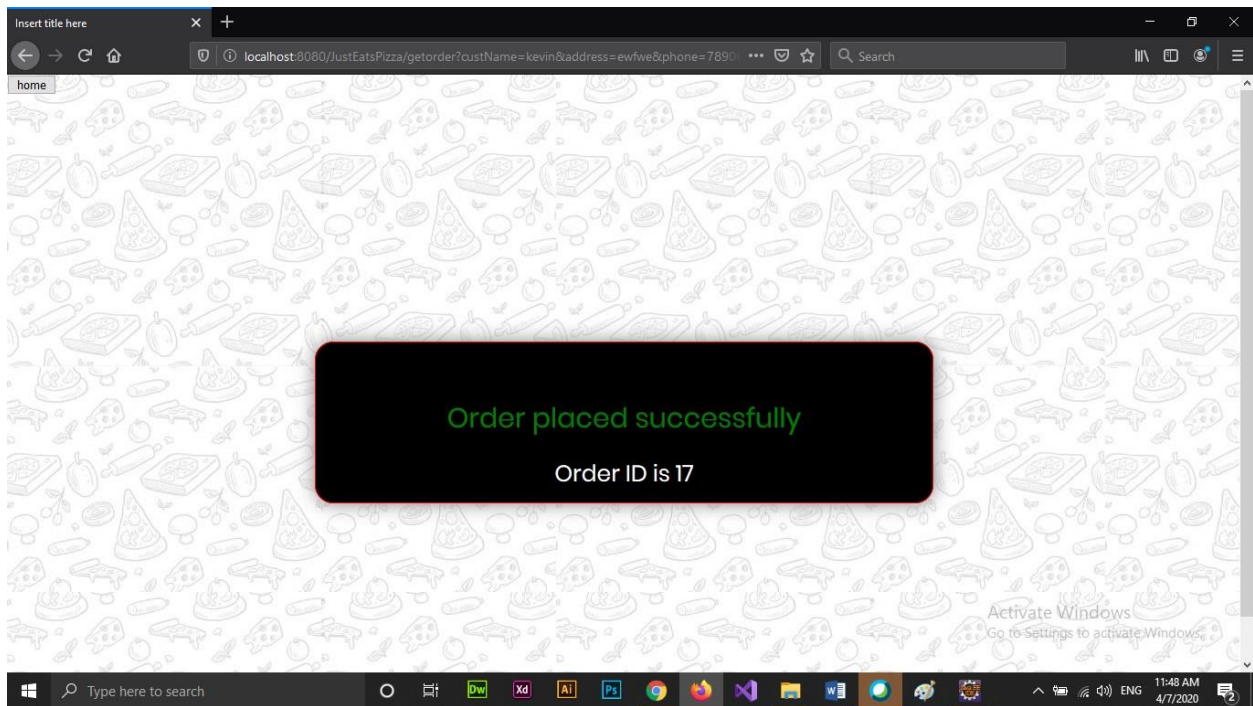
Place order: The customer can then place order by entering the credentials and selecting the topping (screenshot 2). Order id will be displayed after successfully placing the order (screenshot 2.1).

The screenshot shows a web browser window with the address bar displaying 'localhost:8080/JustEatsPizza/placeorder'. The page features a repeating pattern of various pizza-related icons (slices, whole pizzas, toppings) in a light gray color. In the center, there is a white rectangular form with a green border and the title 'Place Order' in green text. The form contains the following fields and controls:

- Name:** A text input field.
- Address:** A text input field.
- Phone:** A text input field.
- Toppings:** A dropdown menu currently showing 'None'.
- Submit Query:** A button located at the bottom of the form.

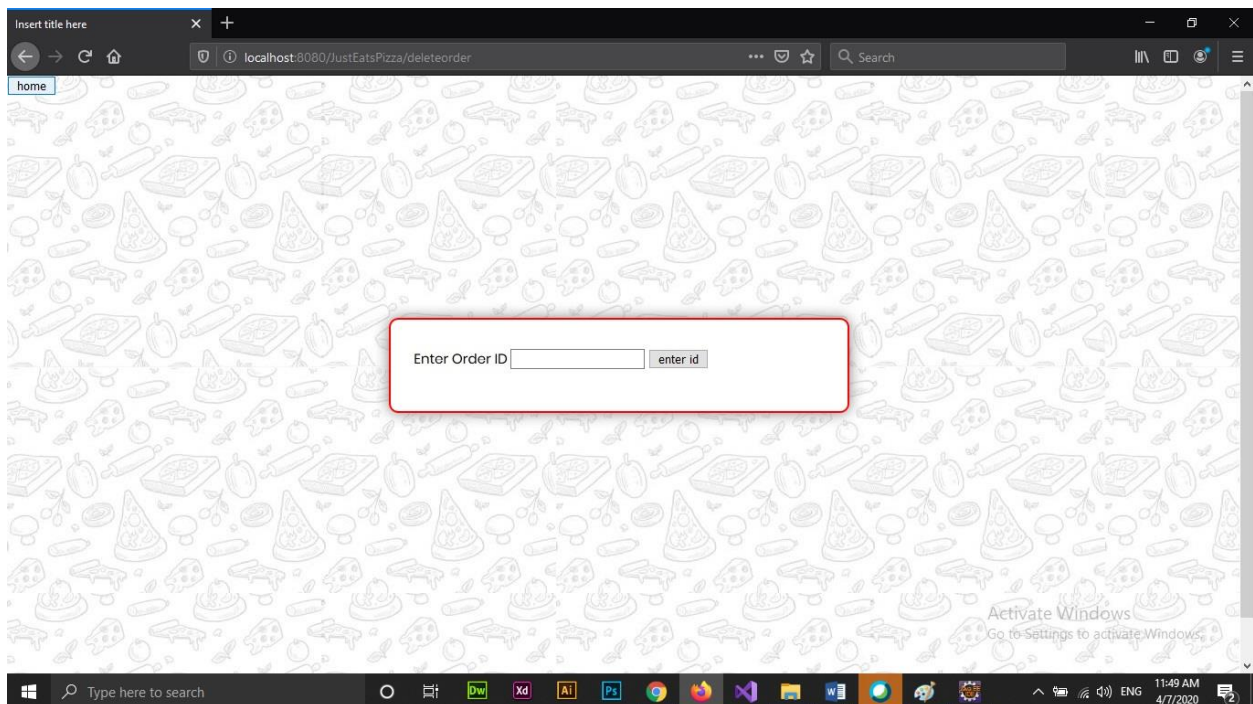
The Windows taskbar is visible at the bottom, showing the search bar, task view button, and several application icons. The system clock indicates 11:47 AM on 4/7/2020.

Screenshot 2

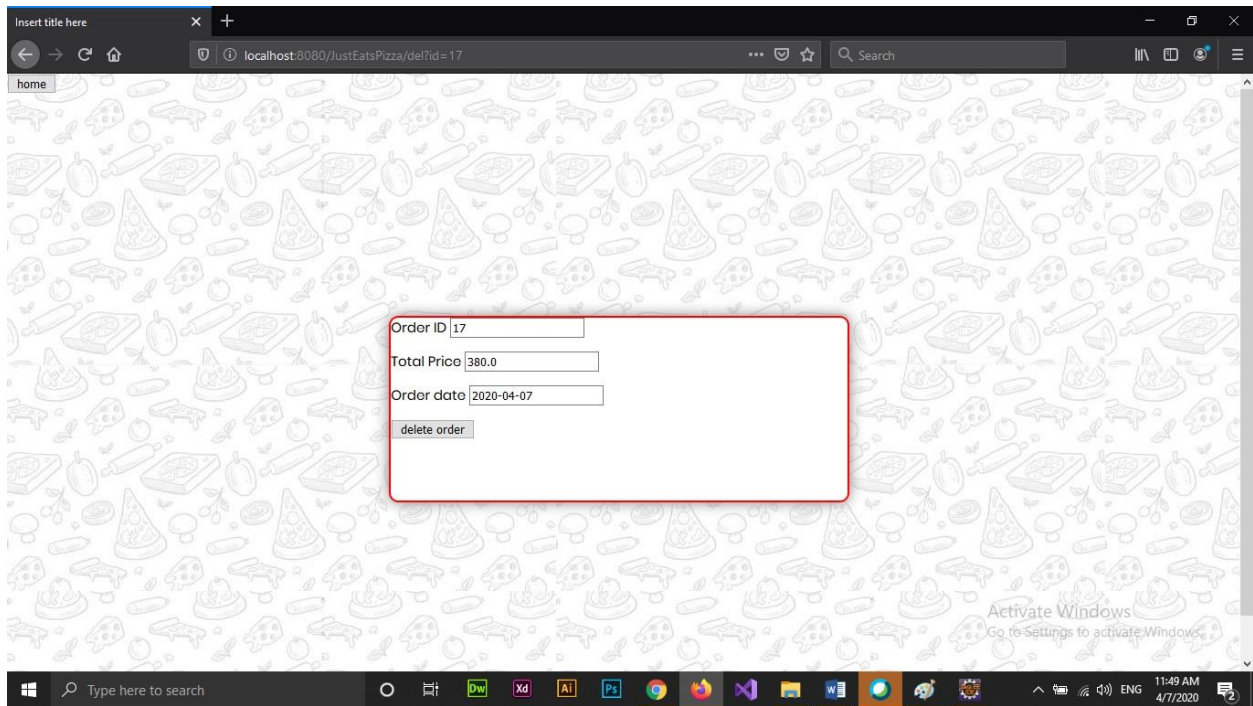


Screenshot 2.1

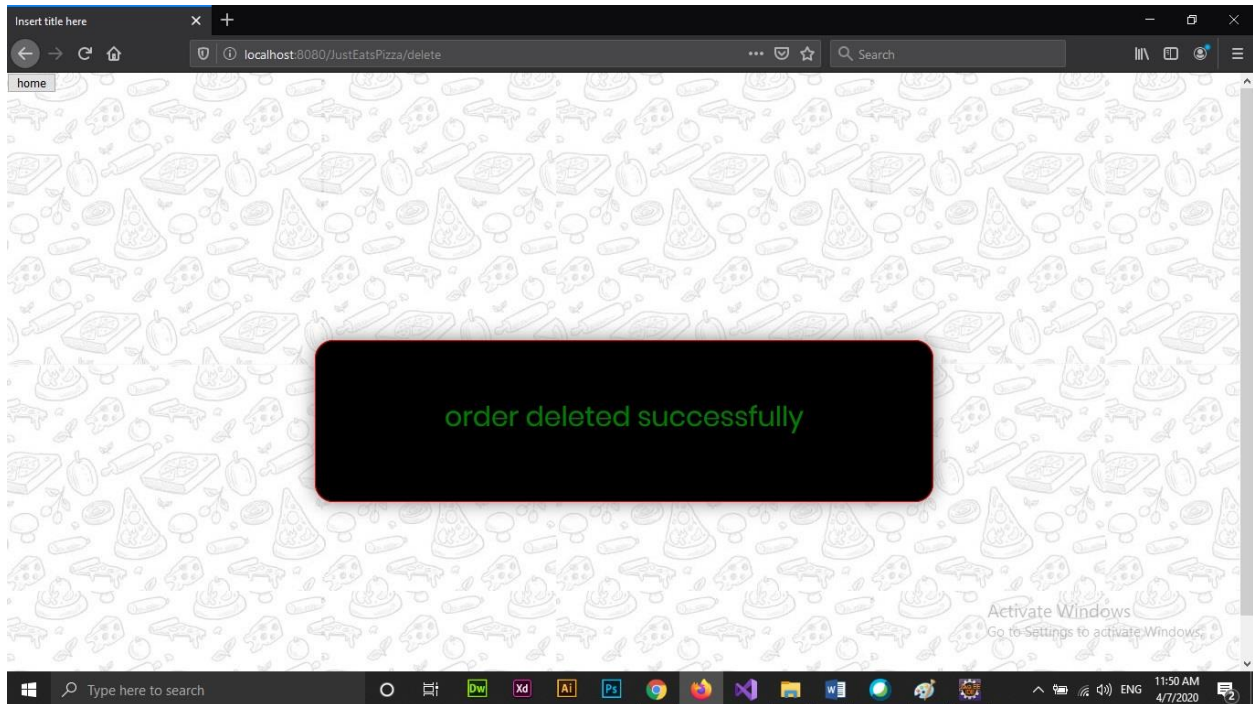
Delete order: The order is displayed by entering the order id (screenshot 3). The order is deleted if confirmed (screenshot 3.1). And success message is shown if the order is successfully deleted (screenshot 3.2).



Screenshot 3

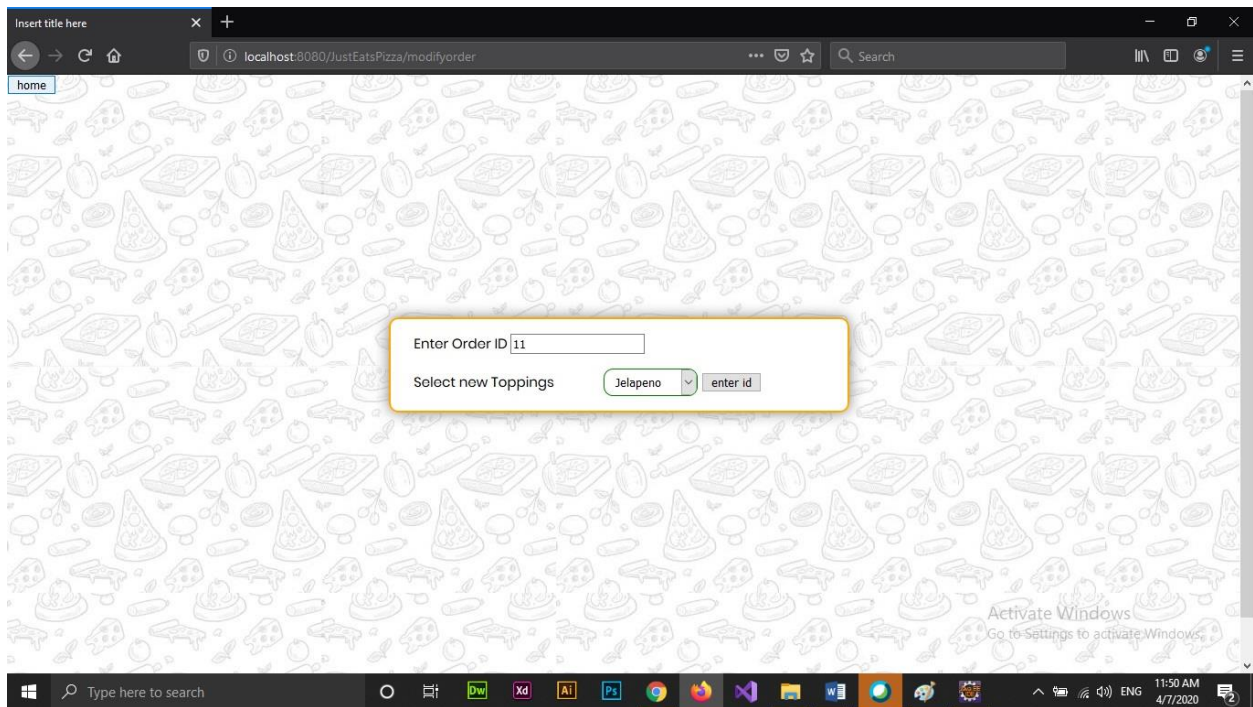


Screenshot 3.1

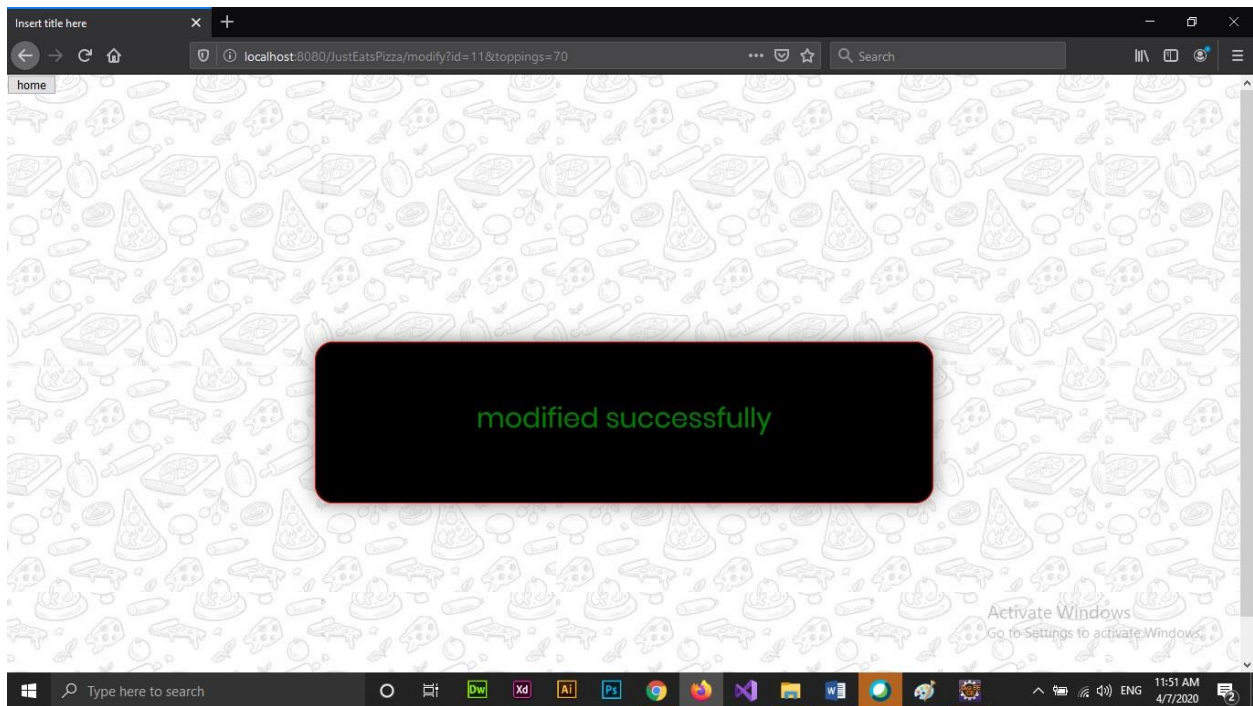


Screenshot 3.2

Modify order: The toppings is modified by the customer by entering the order id and selecting the topping from the toppings dropdown (screenshot 4). Success message is displayed if the modification is successful (screenshot 4.1).

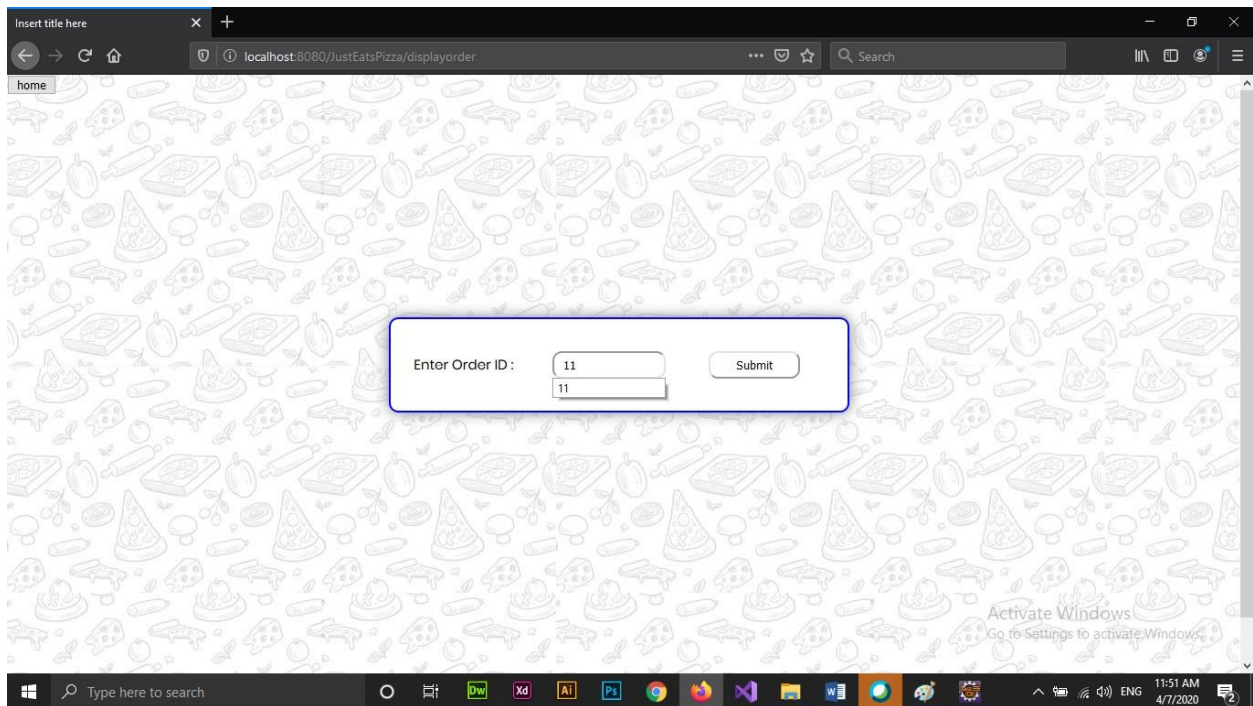


Screenshot 4

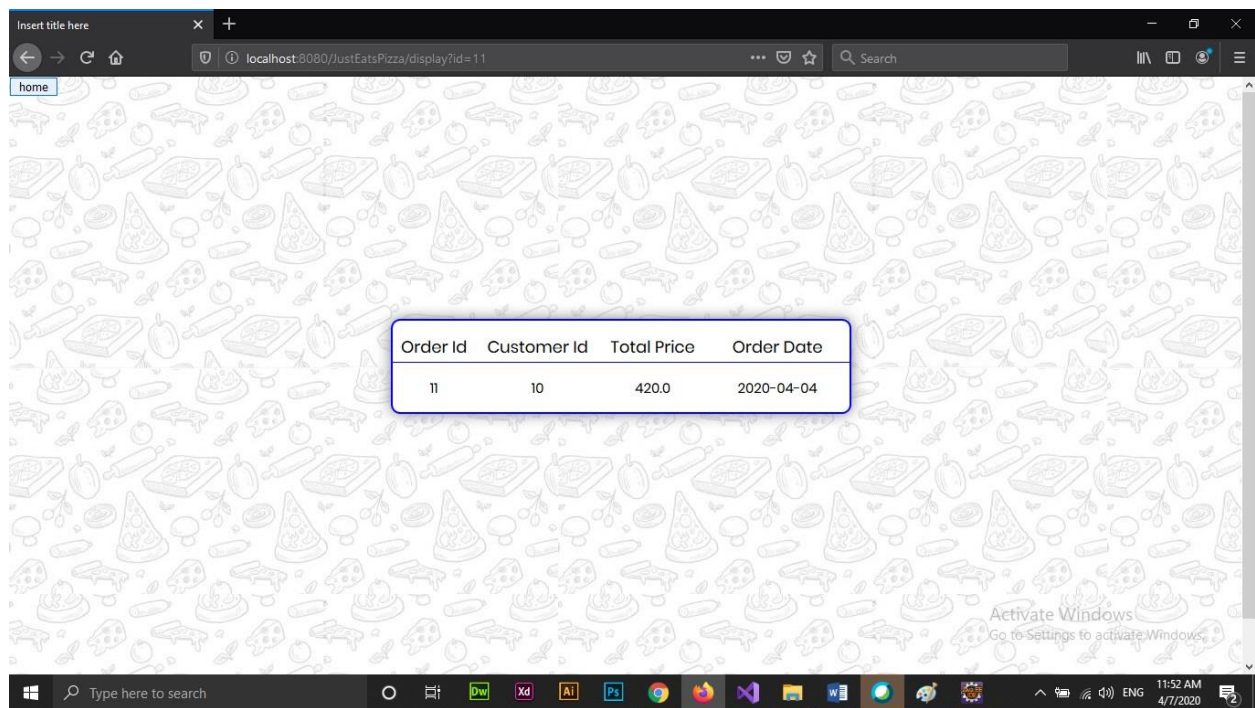


Screenshot 4.1

Display order: The placed order is displayed on entering the order id (screenshot 5 and 5.1).



Screenshot 5



Screenshot 5.1

Conclusion & Future Work

The project achieves a concept of how an online pizza delivery system works. The current system can be enhanced drastically by including a responsive web design by the inclusion of Bootstrap framework or by any other methods.

The order system developed is limited and as future work the scope of the project must be increased to accommodate delivery and payment functionalities. Stricter validations and more exception handling mechanisms are to be implemented.

References

- Spring Forms :
https://www.tutorialspoint.com/spring/spring_mvc_form_handling_example.htm
- Hibernate and Spring integration : <https://www.journaldev.com/3524/spring-hibernate-integration-example-tutorial>
- Basic of Spring :<https://spring.io/>
- W3 schools forms :
https://www.w3schools.com/html/html_form_attributes.asp