JUST EAT PIZZA – A Pizza Ordering System

PROJECT REPORT

submitted by

AISWARYA S PRAKASH

(845331)

LIST OF CONTENTS

CHAPTER NO	TITLE	PAGE NO
1	ABSTRACT	3
2	INTRODUCTION	4
	2.1 Objective	4
3	REQUIREMENT SPECIFICATION	5
	3.1 Technology used in the project	5
	3.2 Hardware Components	5
	3.3 Functional Requirements	5
4	ARCHITECTURE DESIGN	6
	4.1 Project Modules	6
	4.2 Use case diagram	6
	4.3 Screenshots	7
5	CONCLUSION AND FUTURE WORK	8
6	REFERENCES	9

ABSTRACT

The purpose of Just Eats Pizza is to automate the existing manual system by the help of computerized equipment and computer software, fulfilling their requirements, so that their valuable data information can be stored for a longer period with easy accessing and manipulation of the same. The proposed system is a pizza ordering system that fulfills all the requirements of user and it provides an easy interface to navigate. The system uses Java, JSP and MySQL to developed the online pizza ordering system, which provides interface for searching and ordering Pizza. The system simplified the flow of Pizza ordering in this project, so if a person wants to buy Pizza then he can easily order on some clicks only.

This system has implemented filters so customer can choose different toppings, can modify an order and delete an order. Here the customer need to login with a valid username and password for secure login. After successful log in, customer will see the homepage where various options are available. Customer can place the order by simply filling the given form with customer details and order details. Customers can change their selected topping with new one whenever required. Customer can view/edit/delete the added pizza using the given order id. Once the order is placed and order id will be provided to the customer using which he/she can track the ordered pizza. Pizza can be order in more convenient way. It makes system very effective for ordering a pizza. Easy add/update/delete process of pizza. This system provides interface and the customers can easily place the order of Pizza on few clicks.

INTRODUCTION

Pizzas are one among the favourite and most popular food items that the modern day people love a lot. There are many pizza corners in and around the city. The pizza ordering system is an application that allows a person to online the pizzas through the online mode. As the internet users are increasing day by day, the pizzerias are including a service in ordering the pizzas through the online mode. It helps in the saviour of time, effort of an individual.

You must have ordered your pizzas on phone for home delivery. The process seems easy to use but at times there is miscommunication. As there is no visual menu shown during a phone call, the employees have to repeat a lot of things again and again to the customers. It's a time consuming process which at times irritates customers. It would be much more comfortable for the customers to have an online pizza ordering system which will be done through this application. The users can select the pizzas they want and this application will help in reducing the purchase time for customers.

2.1 Objective

The objective is to automate the existing manual system by the help of computerized equipment and computer software, fulfilling their requirements, so that their valuable data information can be stored for a longer period with easy accessing and manipulation of the same. Basically the project describes how to manage for good performance and better services for the clients.

REQUIREMENT SPECIFICATIONS

3.1 Technology used in the project

- HTML : Page layout has been designed in HTML
- CSS : CSS has been used for all the designing part
- JavaScript : All the validation task and animations has been developed by JavaScript
- JSP : All the front end logic has been written in JSP
- Java : All the business logic has been written in Java
- MySQL : MySQL database has been used as database for the project
- Tomcat : Project will be run over the Tomcat server
- Framework : Hibernate with spring MVC

3.2 Hardware Components

- Processor – i3
- Hard Disk – 5 GB
- Memory – 2GB RAM

These are the supported Operating System

- **Windows :** This project can easily be configured on windows operating system. For running this project on Windows system, you will have to install Tomcat 7, JDK 7, MySQL 5.
- **Linux :** We can run this project also on all versions of Linux operating system
- **Mac :** We can also easily configured this project on Mac operating system.

3.3 Functional Requirements

- User login page
- Home page with services to place order with fill up forms
- Service to delete existing order and update order
- Service to view ordered details
- Logout

ARCHITECTURE DESIGN

4.1 Project Modules

Login : Here the customer need to login with a valid username and password for secure login.

Home Page : After successful log in, customer will see the homepage where various options are available.

Place Order : Customer can place the order by simply filling the given form with customer details and order details.

Delete Order : Customer can delete the existing order using the given order id.

Modify Order : Customers can change their selected topping with new one whenever required.

Display Order : Customer can view the existing order using the given order id.

Logout : Logout from the page.

4.2 Use case diagram

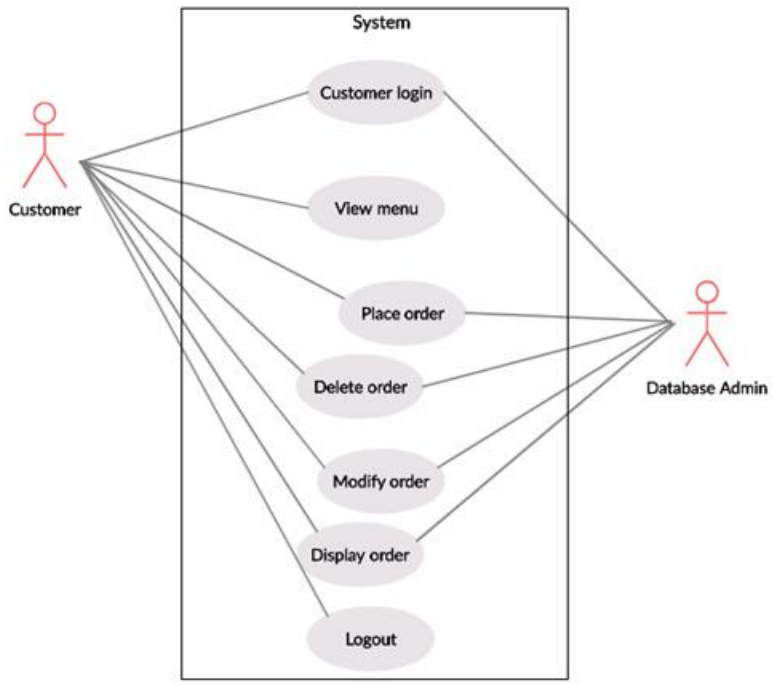
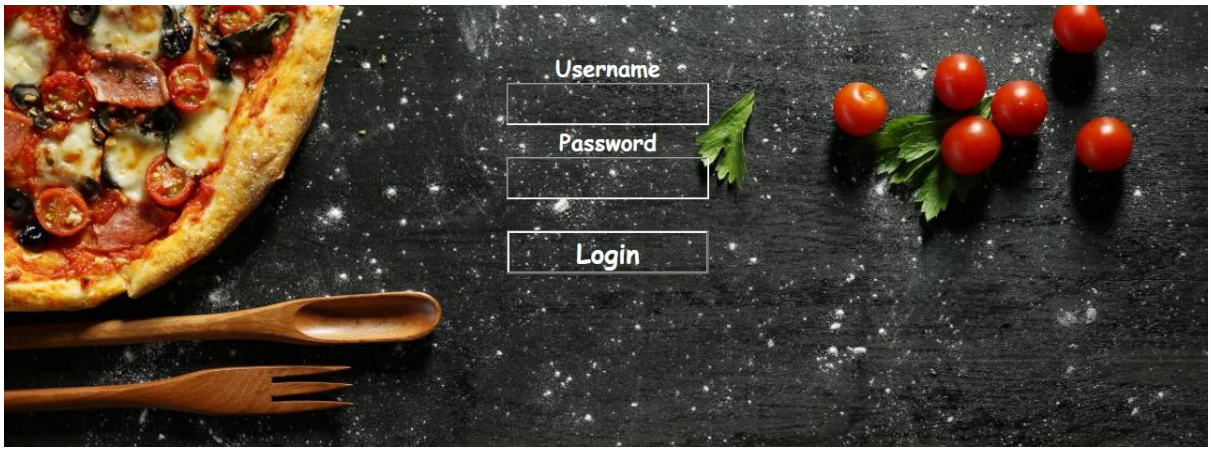


Figure shows the use case diagram of Just eat pizza System. A use case diagram is a dynamic or behavior diagram in UML Use cases are a set of actions, services, and functions that the system needs to perform. In this context, a "system" is something being developed or operated,

such as a web site. The "actors" are people or entities operating under defined roles within the system. Use case diagrams are valuable for visualizing the functional requirements of a system that will translate into design choices and development priorities. Use case diagrams specify how the system interacts with actors without worrying about the details of how that functionality is implemented.

Customer and Database admin are the actors in this use case diagram. An actor specifies a role played by a user or any system that interacts with the subject. Use cases used in this diagram are represented by circles and they are customer login, place order, delete order, modify order, display order and logout. Use cases of customer are customer login, view menu, place order, delete order, modify order, display order and logout. Use cases of database admin are customer login, place order, delete order, modify order and display order.

4.3 Screenshots



CONCLUSION AND FUTURE WORKS

The system can lead to error free, reliable and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on record keeping. Thus it will help system in better utilization of resources. The system can maintain computerized records without redundant entries. This system has implemented filters so customer can choose different toppings, can modify an order and delete an order. The system simplified the flow of pizza ordering in this project, so if a person wants to buy pizza then he can easily order on some clicks only.

In future work, the system can provide more advanced software including more facilities and will host the platform on online servers to make it accessible worldwide. Implement the backup mechanism for taking backup of codebase and database on regular basis on different servers.

REFERENCES

[1] http://javawebtutor.com/articles/spring/spring-mvc-hibernate-crud-example.php

[2] https://www.tutorialspoint.com/spring/spring_web_mvc_framework.htm

[3] https://www.javatpoint.com/

[4] https://www.tutorialspoint.com/hibernate/hibernate_query_language.htm

[5] https://www.tutorialspoint.com/hibernate/index.htm