

CPSC 476

Web Back-End Engineering for Enterprise



Graduate Project
Restaurant Rate Reviews

By: -

Member Name

Dipika Mahashabde

Email ID

dipika.2204@csu.fullerton.edu

CWID

803004274

Contents

Introduction 3

 Purpose 3

 Project Description 3

 Document Scope 3

Project’s Architecture 4

 Hibernate 4

 Model 4

 View 4

 Controller 4

Setup Instructions 5

Functions: 6

 login_check() 6

 signup_add() 6

 Add restaurant 6

 show_review() 6

 show_reviewform() 6

 add_ratereview() 6

Application in Action 7

Challenges 15

References 16

Introduction

Purpose

The purpose of the document is to present the implementation details of an Enterprise Java web application. The application implements rating and reviews of different restaurants. This is done using Spring Hibernate api.

Project Description

The project includes a signup page where users can create an account with username and password. After the user is logged in, a private page is displayed where users can see and rate, review the restaurants. The application also includes a public front page where users can see the restaurants ratings and reviews but cannot rate or review unless they are logged in. When the user rates the restaurant the average of all the rates is displayed on the card of every restaurant. New restaurant will be added by the Admin with credentials.

Document Scope

This document describes the functionalities of the Restaurant Rate and Review application. It also describes the configurations and required settings to run it on the system and also information of hibernate api.

Project's Architecture

The architecture we have used for this project is MVC (Model View Controller) and Hibernate how it talks to the database.

Hibernate

Hibernate is an object relational mapper(orm). It directly maps your Java Object to the database and back the database row into and Java Object. You need to mention some properties of hibernate and mention the database schema in an xml file. The extension of the xml file should be xyz.hbm.xml. it maps the java object to this file and converts into the format the database can read it. (Gavin , Bauer, Andersen, Bernard, & Ebersole)

Model

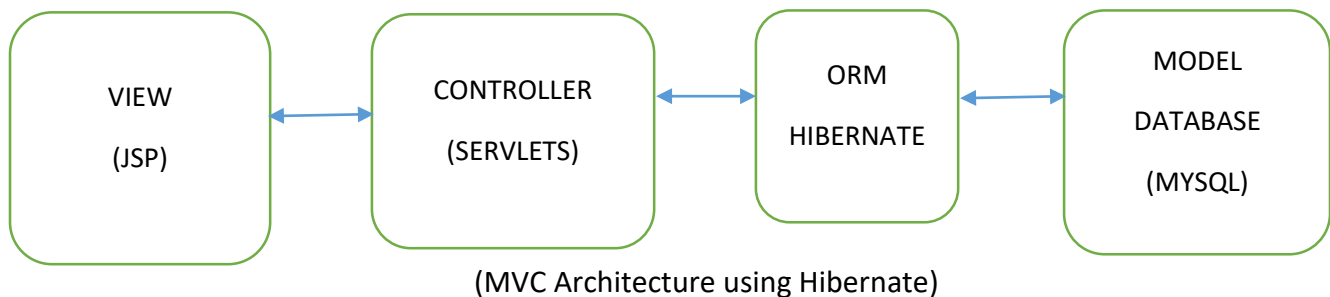
For storing data, we have used Mysql and POJO(Plain Old Java Object) for getter and setter. In our application, we have used 3 tables for storing data. First, "user" used for storing id, username and password. Second, " restaurant_details" for storing restaurant id, user id, restaurant name, its location average rating and how many ppl rated and reviewed it. Third,"rate and review" for storing the detailed information of the rates and reviews given to the restaurant.

View

For view, we have four JSP files. First is the home page jsp named "login.jsp" that asks the user for the login credentials such as username and password and also show all the restaurants and its location and rating and reviews. If the username and password is correct , it will redirect to another jsp which is priavtepage.jsp. Where you can also see the information also here the user can rate and review the restaurant which the user cannot when the user is not logged in.

Controller

For handling the requests and responses, we have used three controllers. First is the LoginServlet controller which handles all the login, logout and signup functions and show the details of the restaurant. Second is the RestaurantServlet which handles private functions such as show_review (), add_ratereview () and show_reviewform() etc. Third is the AdminsServlet controller which adds new restaurant.



Setup Instructions

1. Download and unzip the project on your local drive.
2. Database setup. Command.sql is given to execute please look below for instruction
3. In the command prompt of the project folder type: `mvn clean`. This command will clear all the class files.
4. After doing the clean check the target file you should not have anything in it.
5. Run the `mvn package` this will create a war in the target subdirectory of the project.
6. **Rename the war file to => restaurant.war**
7. Deploy the war file to the tomcat manager or copy the war file in the webapps folder of tomcat.
8. Start the tomcat server by Startup.bat command.
9. Go to this url => <http://localhost:8080/restaurant/login>

Database setup:

(While doing the execute command file do the whole file as it contains some insert queries otherwise the code wouldn't work)

(ref: challenges)

1. Install MySQL on the local Machine.
2. Start a new connection where
Connection Name: localhost
Connection Method: TCP/IP
Host Name: 127.0.0.1
Port: 3306
Username: root
Password: password
3. Execute the Command.sql file in mysql

Functions:

`login_check()`

checks username and password from the Database. If the username and password is not correct, it redirects again to the login.jsp. If the username and password is correct, it redirects to the privatepage.jsp page where users can see the rating, review and also can give one. Session is used to maintain state. To go to the admin check username="admin" and password="admin"

`signup_add()`

it adds a new user into the Database. If the user already exists, it prompts an error message saying that the user already exists.

`Add restaurant`

Add new restaurant into the database.

`show_review()`

Shows all the ratings and reviews of the restaurant also the location of the restaurant.

`show_reviewform()`

Opens up a form to give your rating and reviews about the restaurant.

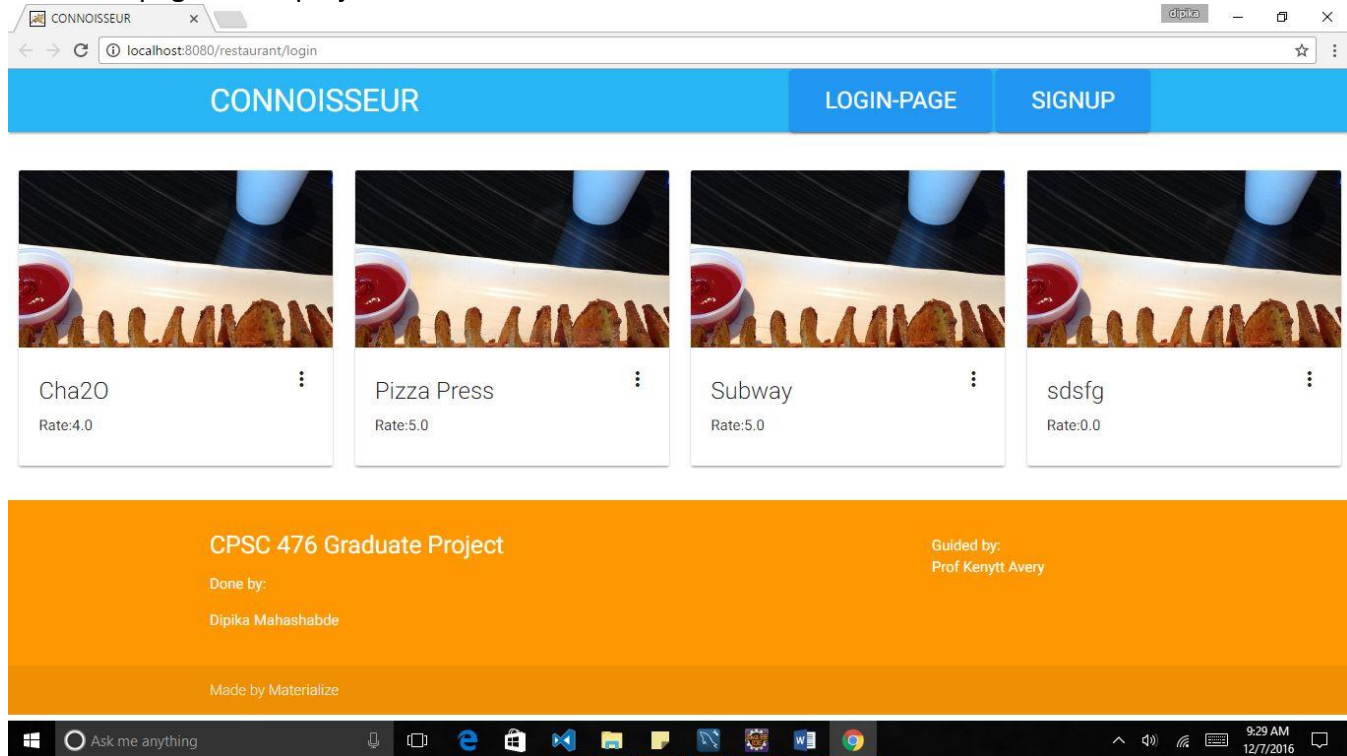
`add_ratereview()`

After the adding the ratings and review, it updates the database and calculates the new average rating.

Application in Action

Please follow the given steps:

The homepage of our project is shown below.



Admin log-in

CONNOISSEUR

signin Home-page

LOGIN

Enter Username:

Enter Password:

SUBMIT

CPSC 476 Graduate Project

Done by:

Guided by:
Prof Kenytt Avery

Windows taskbar: 9:37 AM 12/7/2016

Add new restaurant

CONNOISSEUR

cheesecake factory - Google

LOGOUT

ADD

Enter Restaurant Name:

Enter Location

SUBMIT

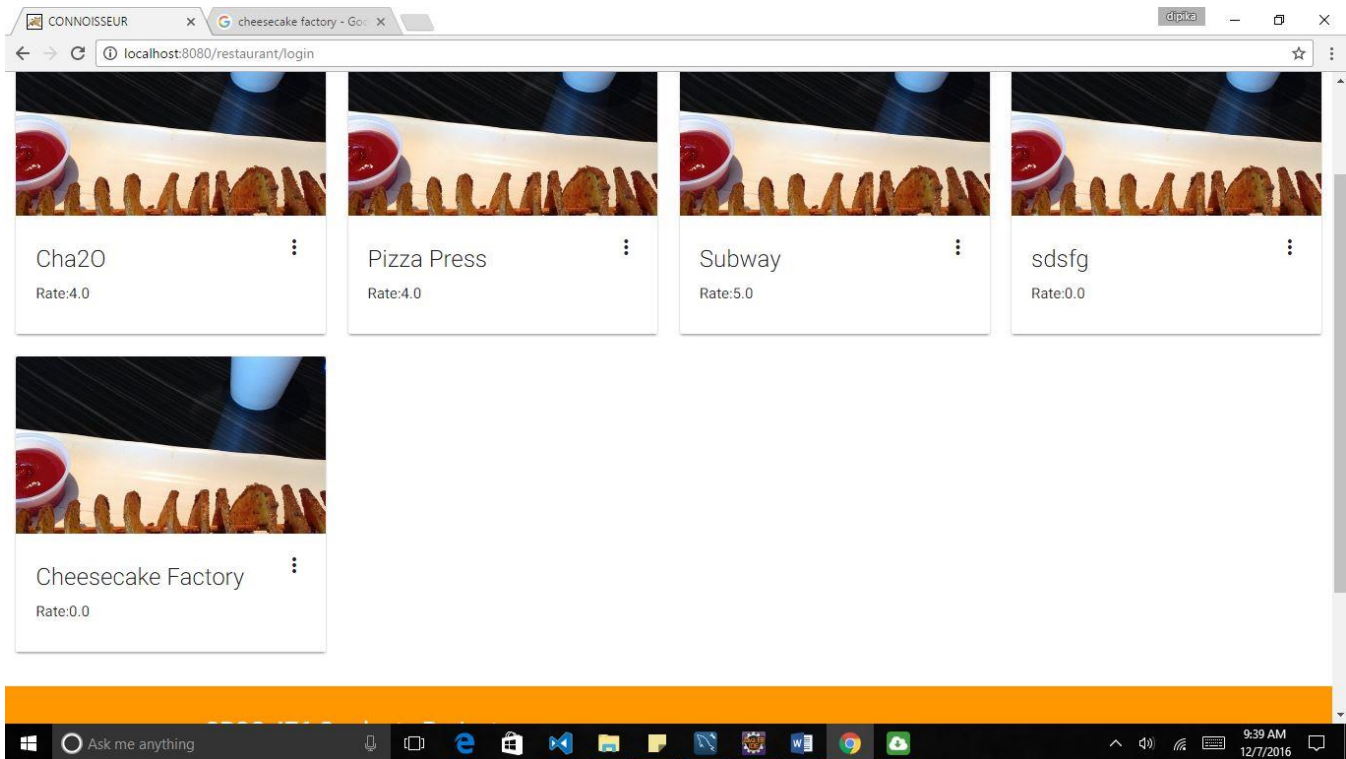
CPSC 476 Graduate Project

Done by:

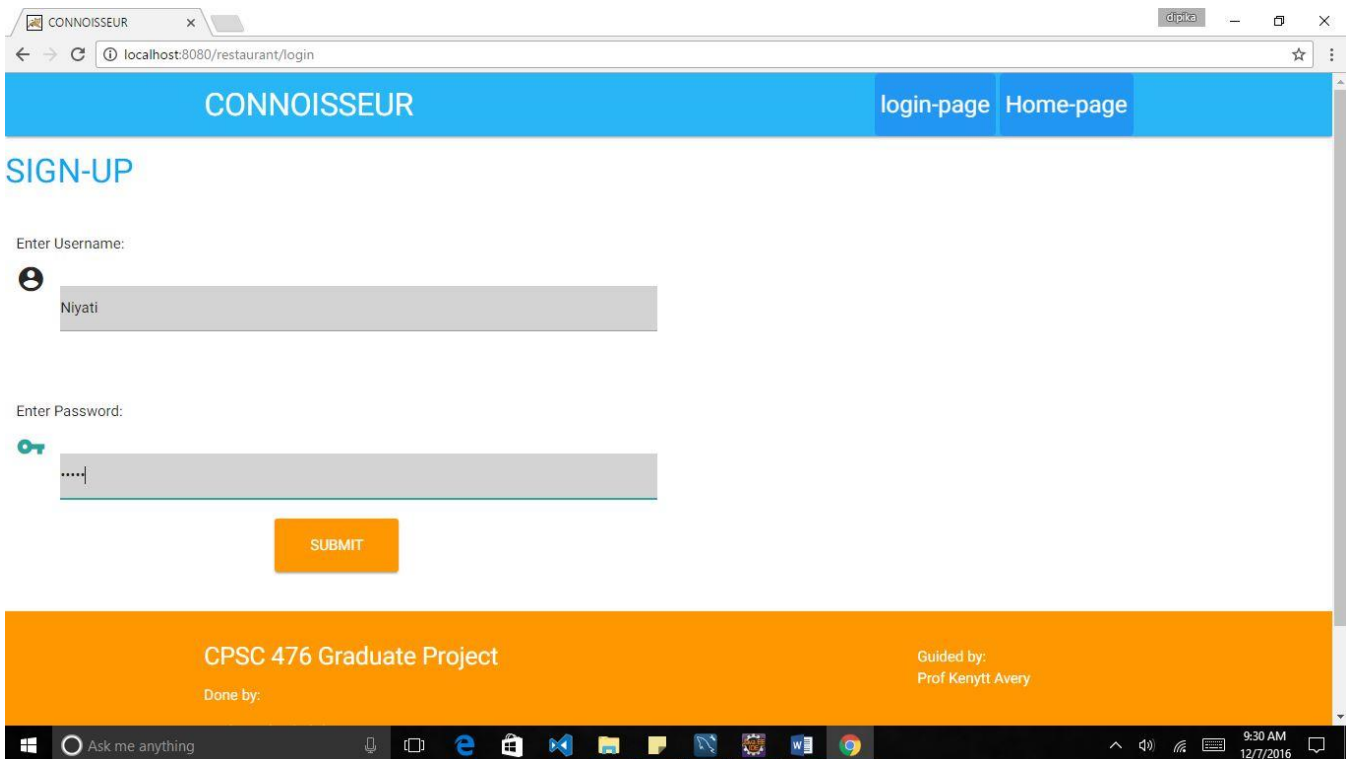
Guided by:
Prof Kenytt Avery

Windows taskbar: 9:39 AM 12/7/2016

Reflection of new restaurant



Users can sign up by providing a username and password.

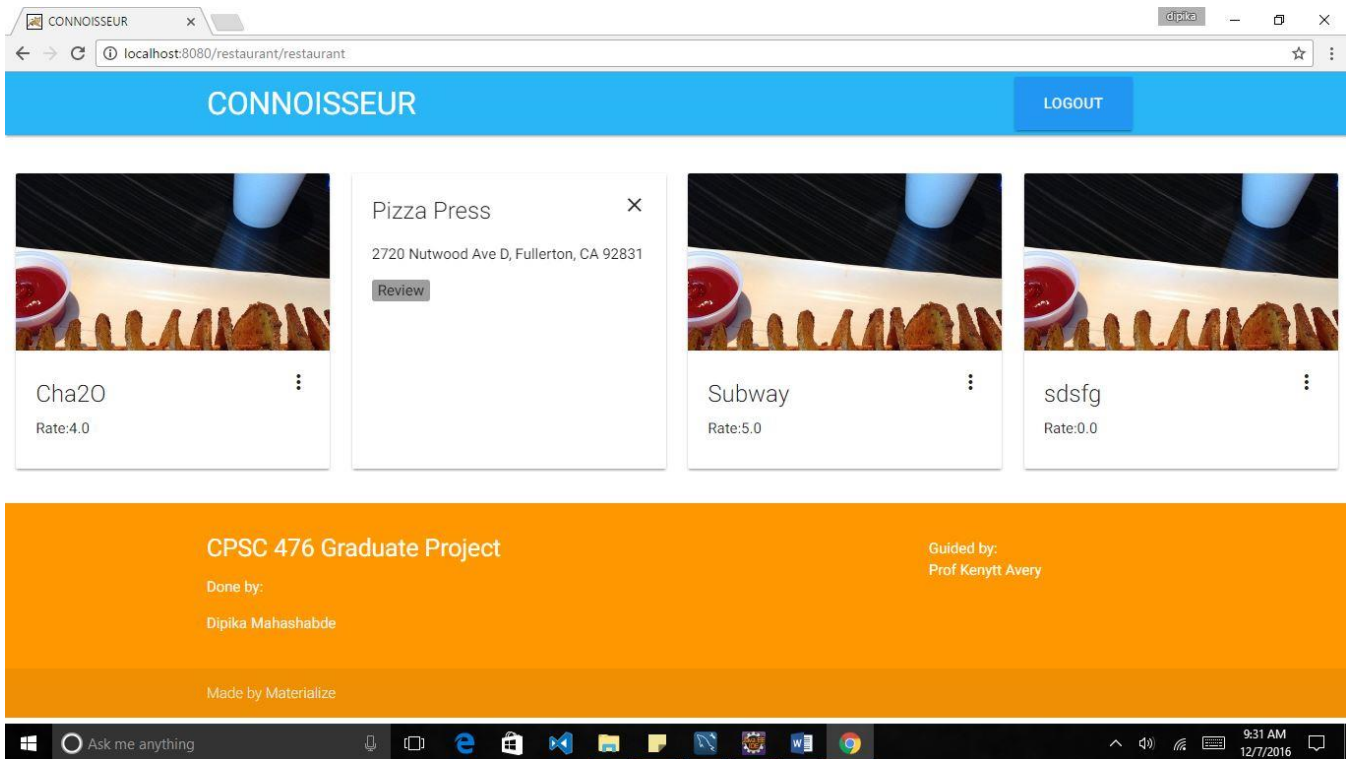


Users can login by entering their username and password.

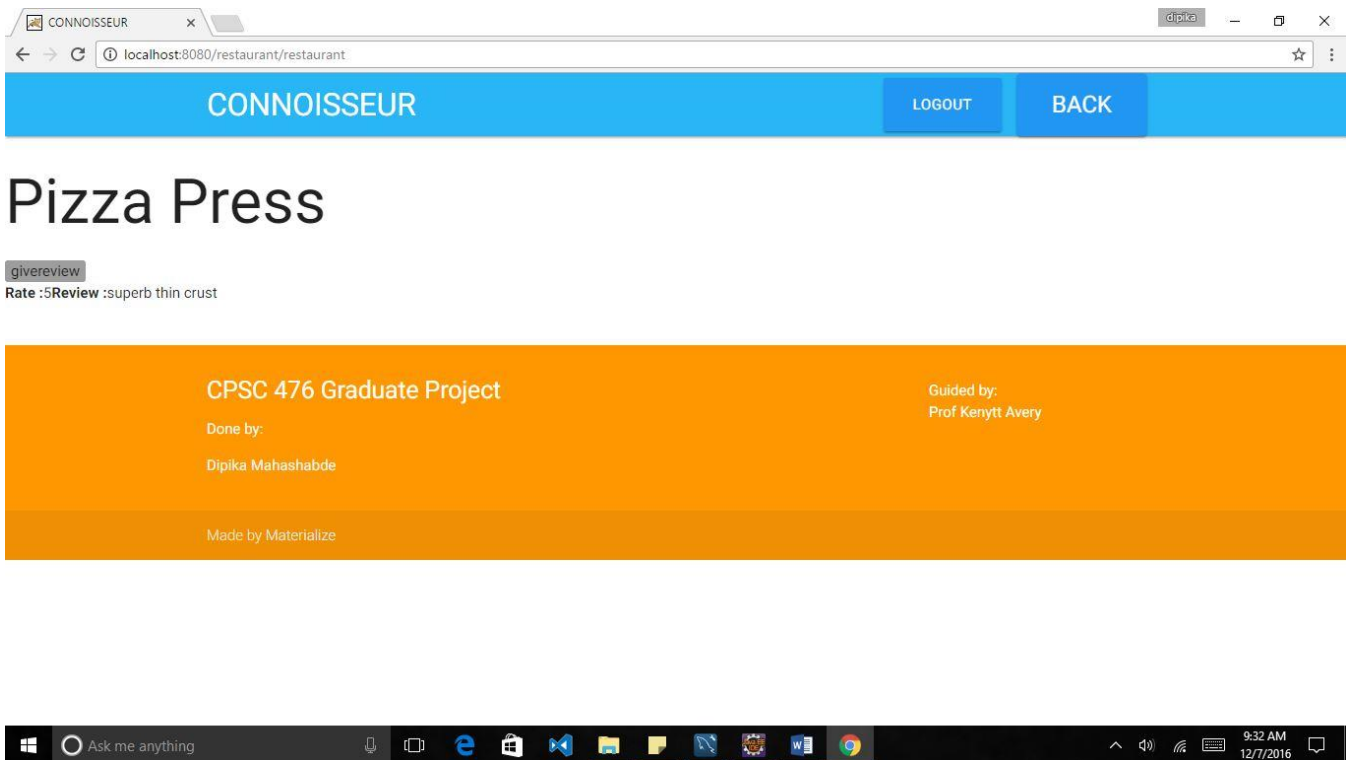
The screenshot shows a web browser window with two tabs: 'CONNOISSEUR' and 'cheesecake factory - Google'. The address bar shows 'localhost:8080/restaurant/login'. The page has a blue header with 'CONNOISSEUR' on the left and 'signup' and 'Home-page' on the right. Below the header, the word 'LOGIN' is displayed in blue. The login form consists of two input fields: 'Enter Username:' with a user icon and a text box containing 'Niyati', and 'Enter Password:' with a key icon and a text box containing '.....'. A 'Full-screen Snip' button is located below the username field. A yellow 'SUBMIT' button is positioned below the password field. At the bottom of the page, there is an orange footer bar with the text 'CPSC 476 Graduate Project' and 'Done by:' on the left, and 'Guided by: Prof Kenytt Avery' on the right. The Windows taskbar is visible at the bottom of the screen, showing the time as 9:44 AM on 12/7/2016.

After logging in This is the home page

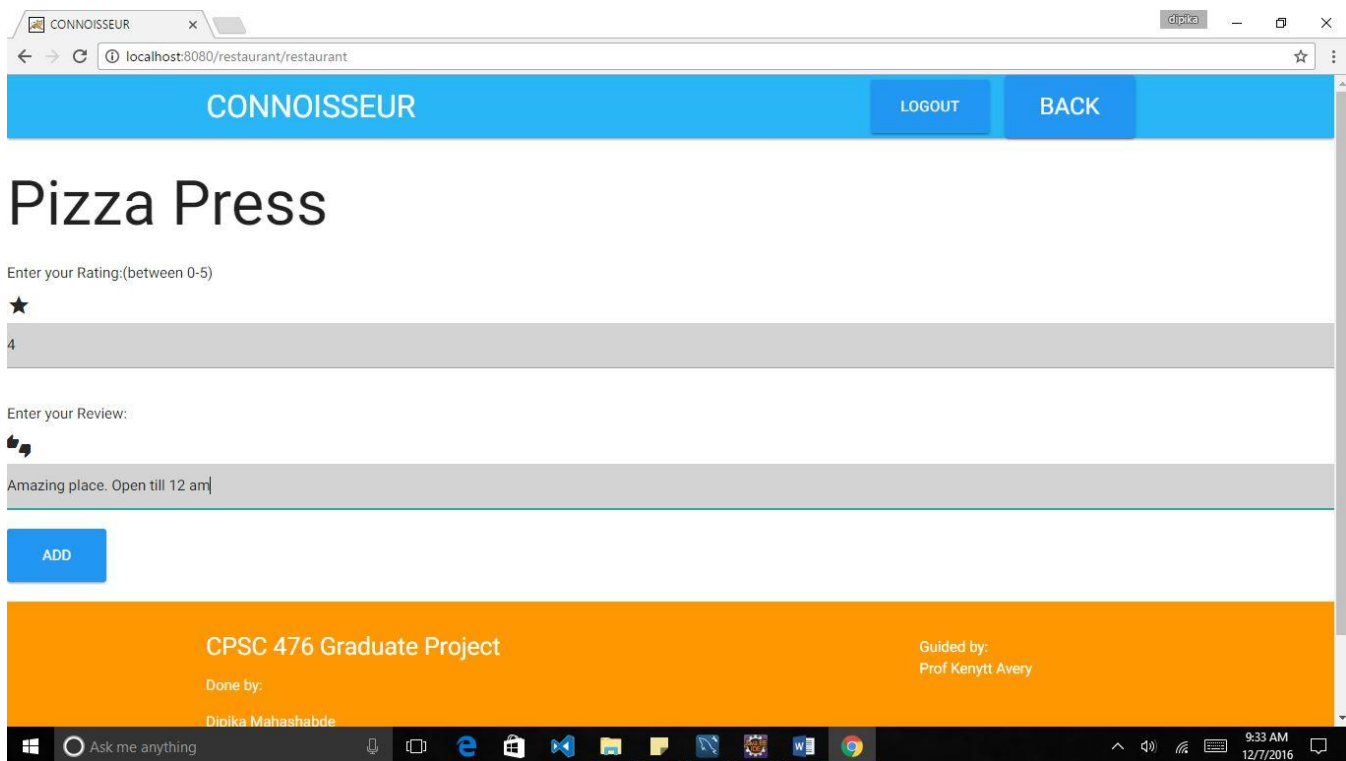
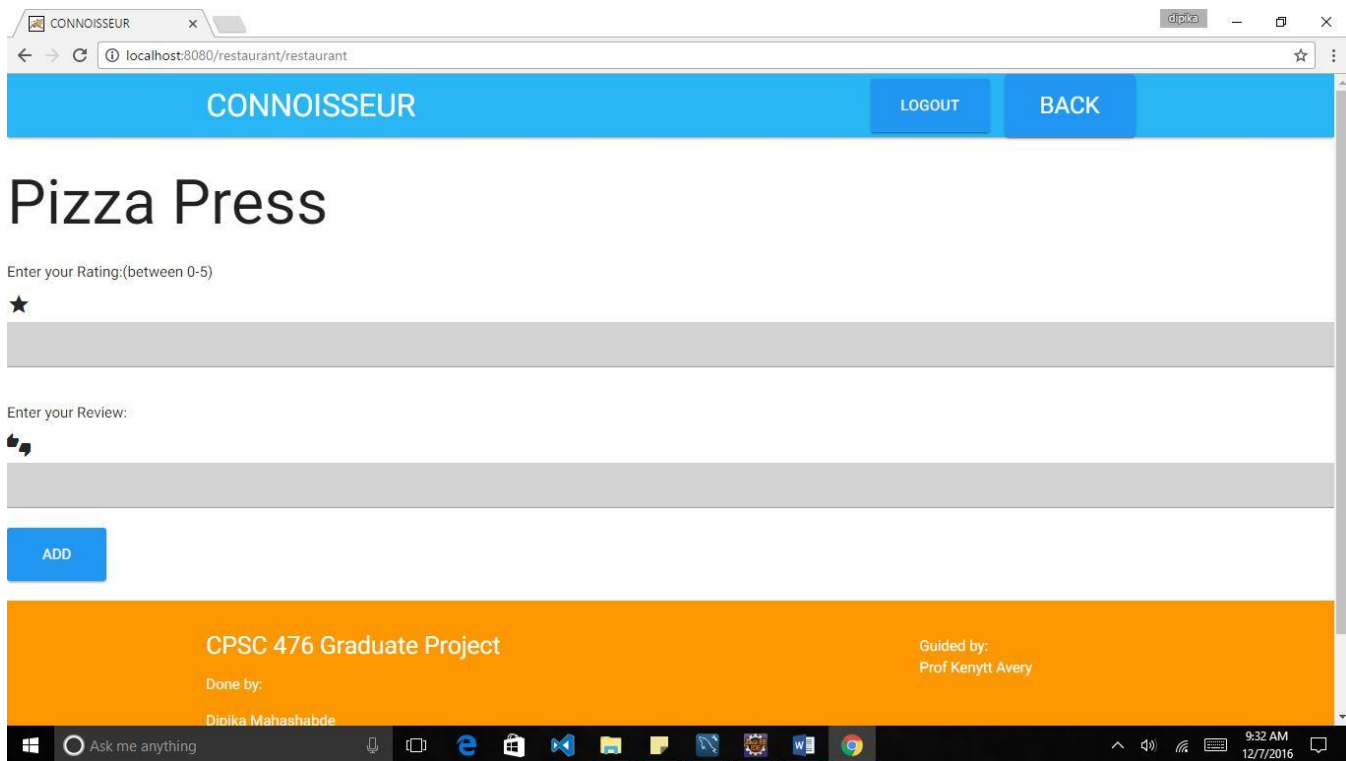
This is where the user has clicked on the restaurant details by clicking the name of the restaurant.



After clicking on the review button



After clicking on the give review button



Reflection of new rating and review

CONNOISSEUR

LOGOUT BACK

Pizza Press

givereview

Rate :5Review :superb thin crust

Rate :4Review :Amazing place. Open till 12 am

CPSC 476 Graduate Project

Done by:
Dipika Mahashabde

Guided by:
Prof Kenytt Avery

Made by Materialize

Quick Heal Internet Security
Quick Update
Updated successfully.


Ask me anything

9:34 AM
12/7/2016


rates are rounded of using Math function named floor

CONNOISSEUR


LOGOUT




Cha20
Rate:4.0



Pizza Press
Rate:4.0



Subway
Rate:5.0



sdsfg
Rate:0.0

CPSC 476 Graduate Project

Done by:
Dipika Mahashabde

Guided by:
Prof Kenytt Avery

Made by Materialize

Ask me anything

9:35 AM
12/7/2016

Challenges

The challenges faced during the project were:

- The problem faced was with redirection. While redirecting variables from jsp to java pages.
- Some problems were faced for updating the rates and calculating the rates. For that I tried with HQL query sum and count to calculate the average but it was difficult to access the data from a non defined object. So took a long procedure to do it.
- If there is no data in the database the project won't work as I have not done the try catch for null values from the database. So please execute the whole command.sql file.

References

Gavin , K., Bauer, C., Andersen, M. R., Bernard, E., & Ebersole, S. (n.d.). *Chapter 14. HQL: The Hibernate Query Language*. Retrieved from <https://docs.jboss.org/hibernate/orm/3.3/reference/en-US/html/queryhql.html>.