

# CALTECH Full Stack Java Developer Capstone Project

## SANDEEP KUMAR JAKKARAJU

*A Food Delivery Website -- Foody !!*

Project Objective:

Create a dynamic and responsive online food delivery web application for ordering food items of different cuisines from a restaurant.

Contains an Admin and an User portal.

When you login with Admin user you will be redirected to Admin Portal.

When you login with Non-Admin user you will be redirected to the End User Portal.

Admin Portal Features:

1. Signup/Sign In
2. Add and Remove Cuisines
3. Add and Remove Food Items For a Cuisine
4. Edit Food Items
5. Change Password
6. Logout

User Portal Features:

1. Signup/Sign In
2. Search food items based on keywords
3. Apply Filters on Search Results.
4. Add selected items to cart and customise order.
5. Pay for the order and receive an order summary.
6. Change Password and Logout.

=====

GIT REPOSITORY: <https://github.com/sandycaltechpgp/capstone-project>

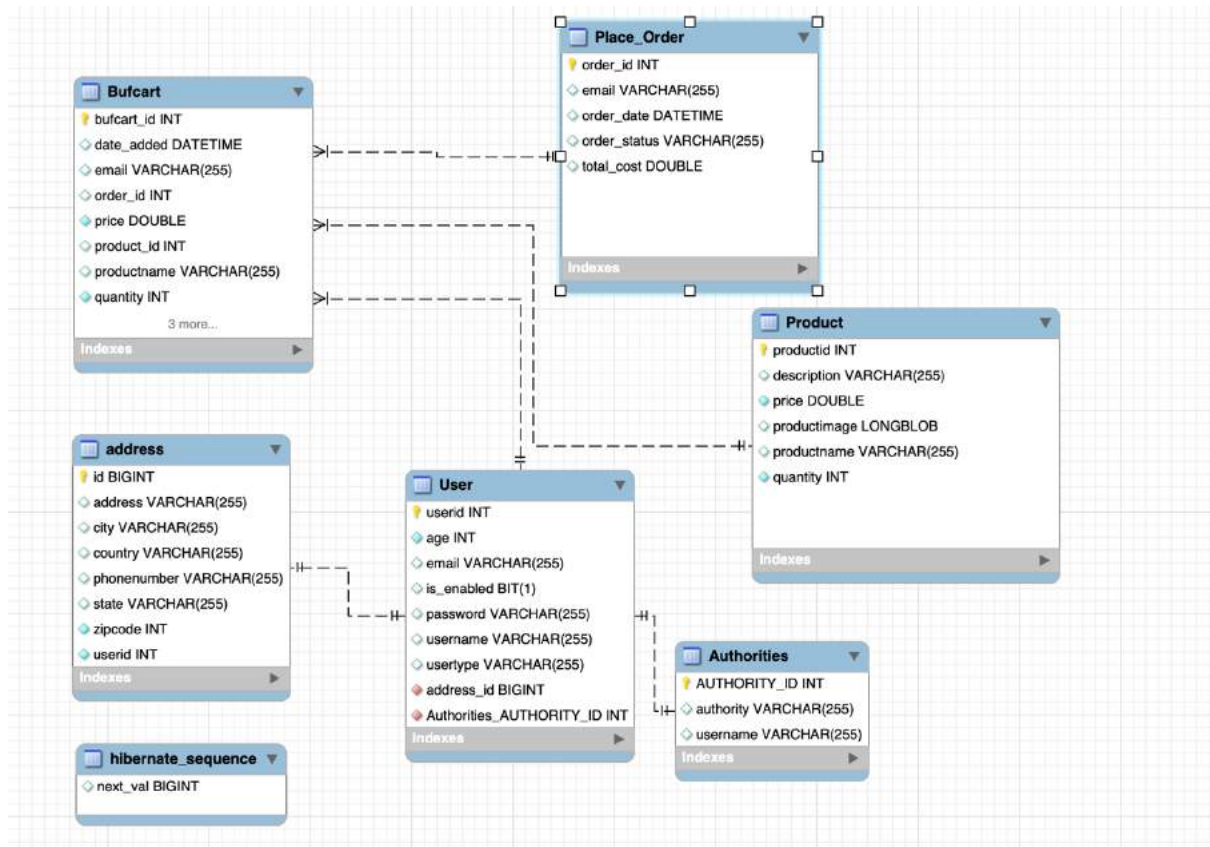
=====

TECH STACK:

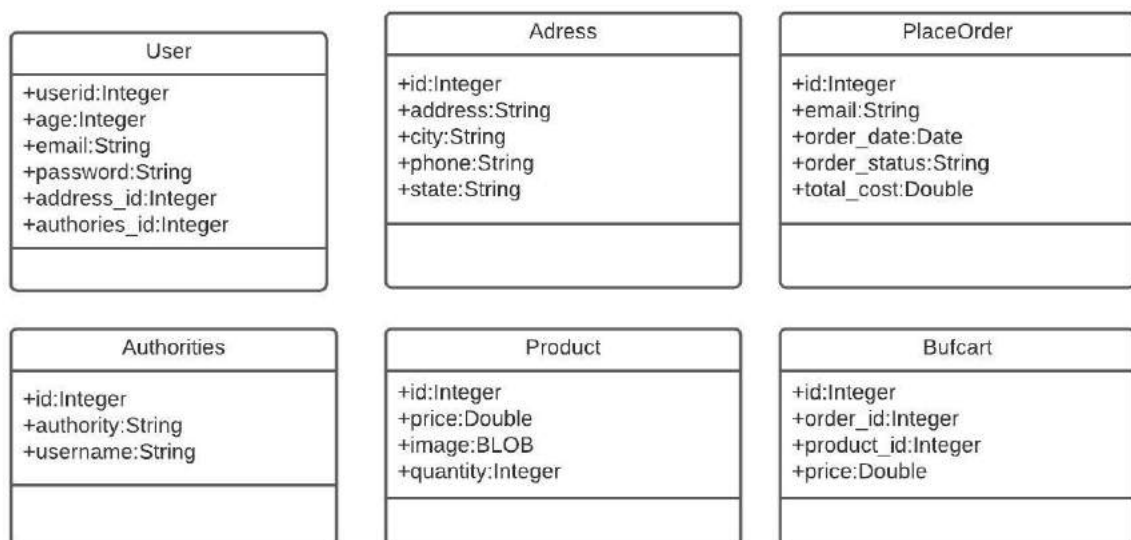
Java 8, Spring boot, JPA, Hibernate, MySQL, Docker, AWS, Angular 8, HTML, JS, Jenkins, Maven, GIT, GITHUB, H2 Database.

=====

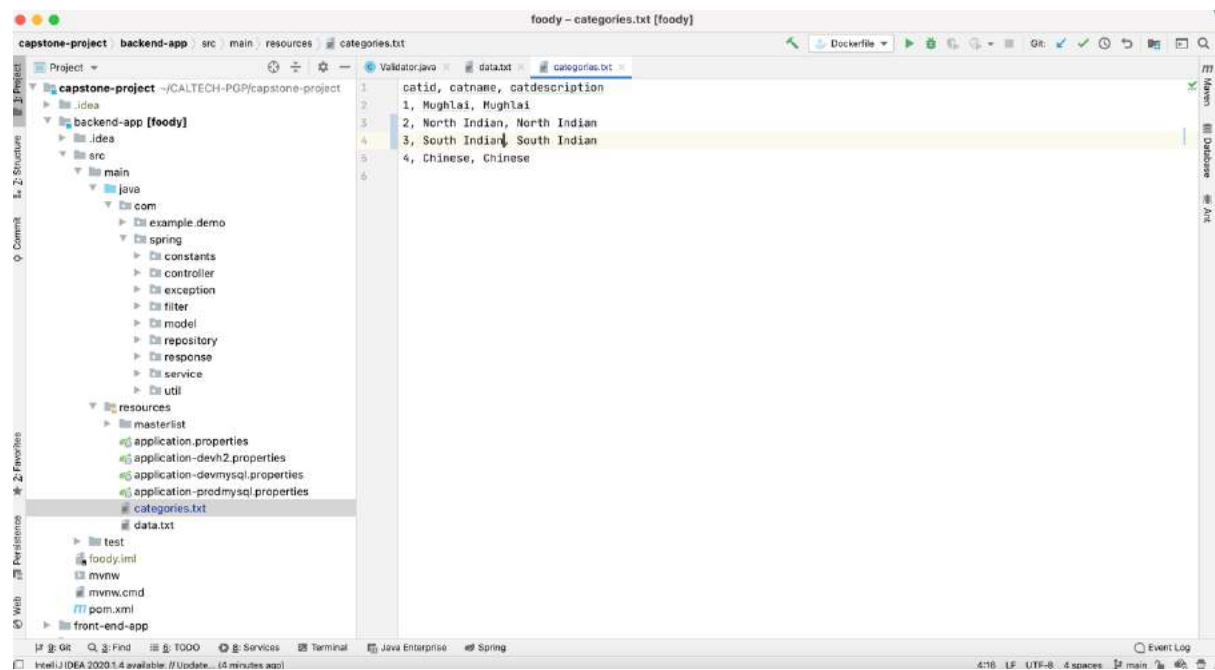
=====



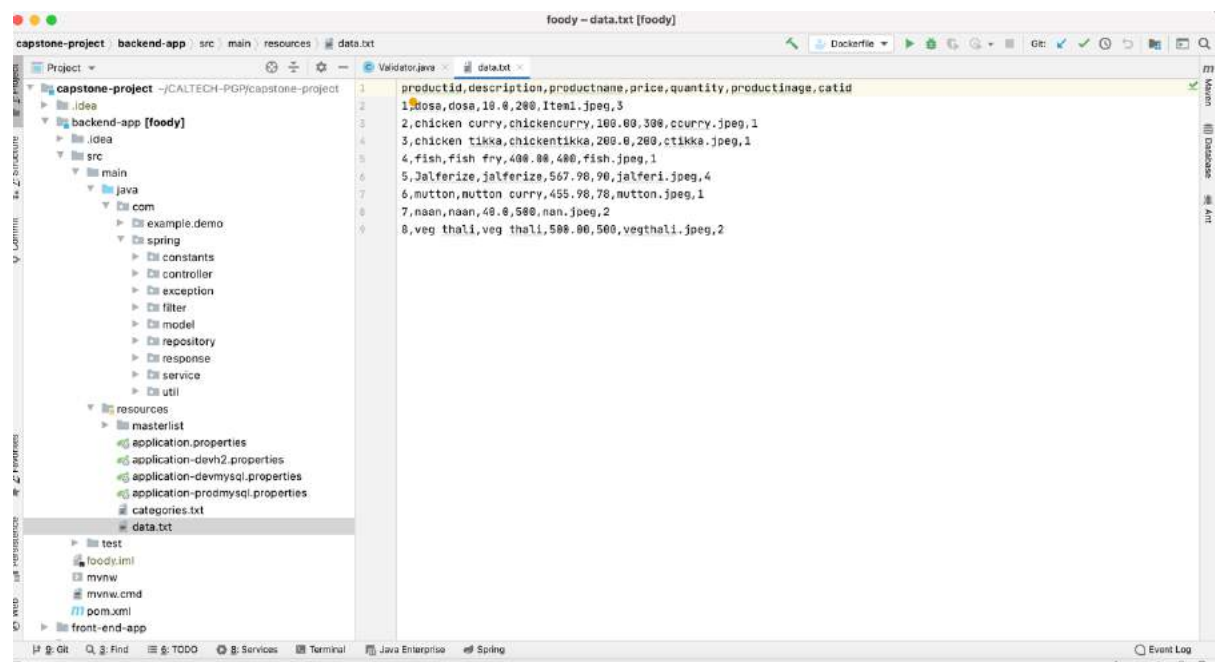
=====



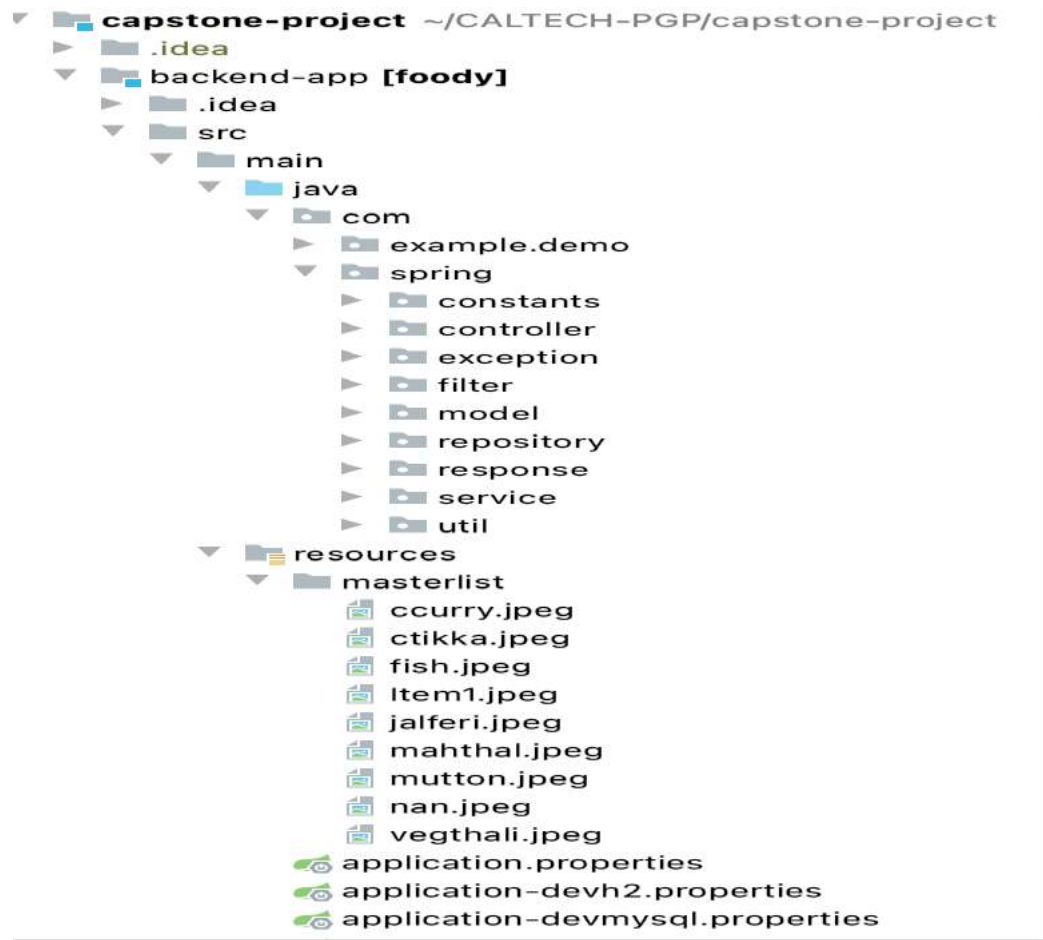
## MASTER LIST OF FOOD CATEGORIES/CUISINES



## MASTER LIST OF FOOD ITEMS



## MASTER LIST OF FOOD ITEMS -IMAGES



Spring Profiles:

1. devh2
2. devmysql
3. prodmysql

REGISTRATION PAGE: ( BOTH USER AND ADMIN CAN REGISTER HERE )

localhost:4200/register

# Foody!!

Register Login

xyz@gmail.com

Username

Password

Age

Register

CHOOSE – ADMIN/USER

localhost:4200/register

# Foody!!

Register Login

xyz@gmail.com

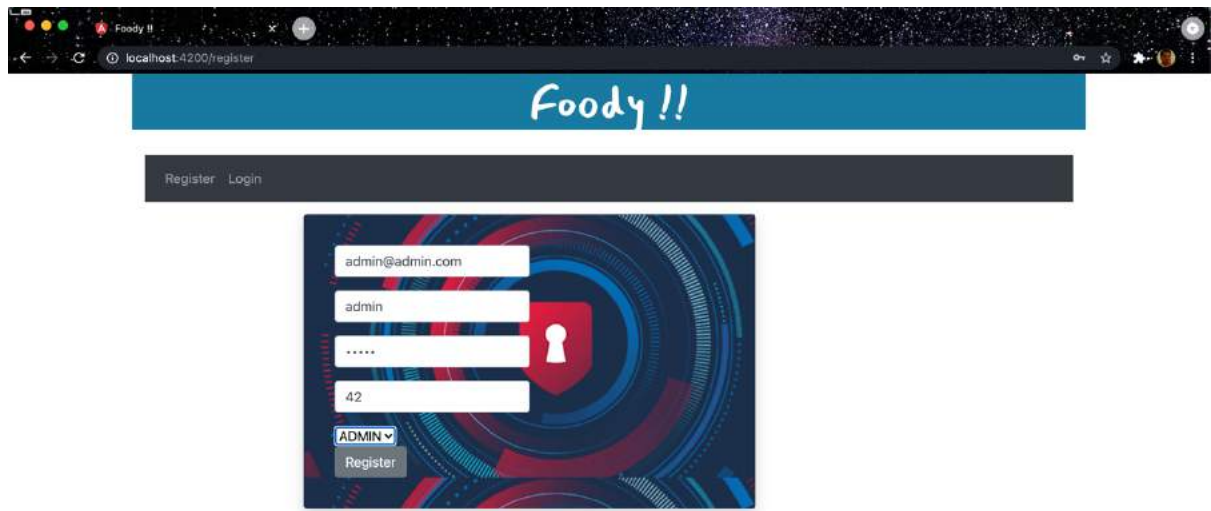
Username

Password

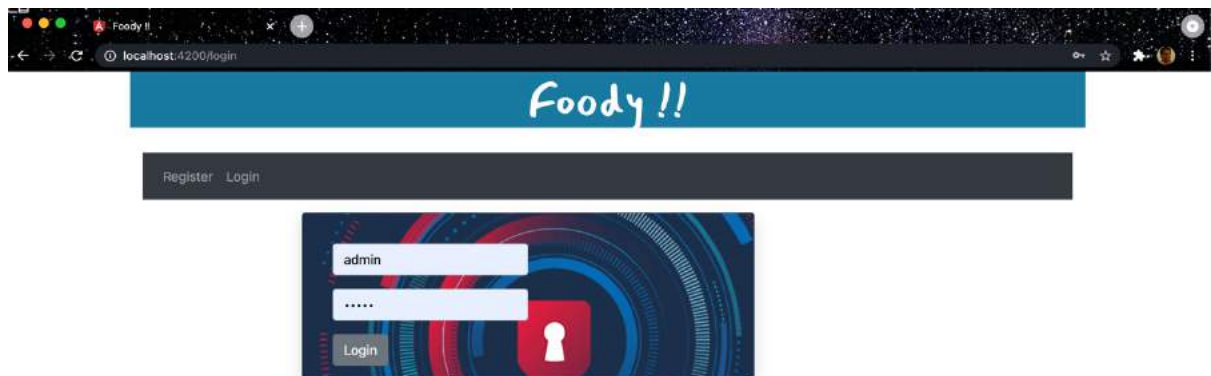
Age

✓ ADMIN  
USER

Register

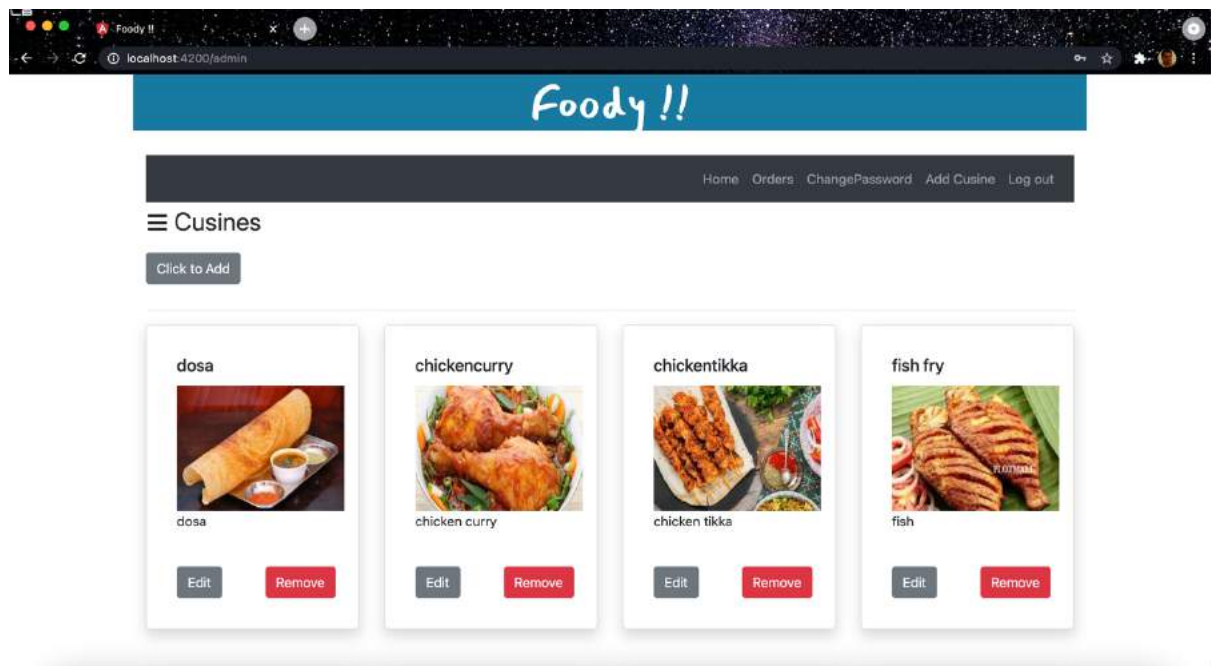


LOGIN : ADMIN

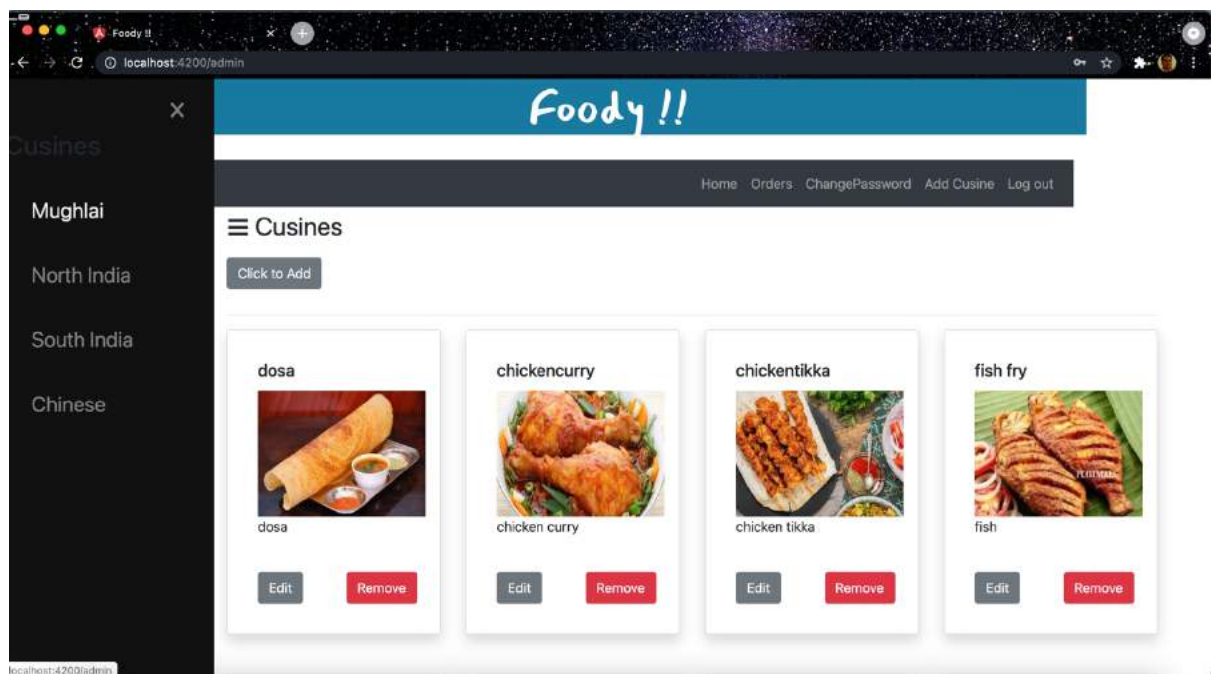




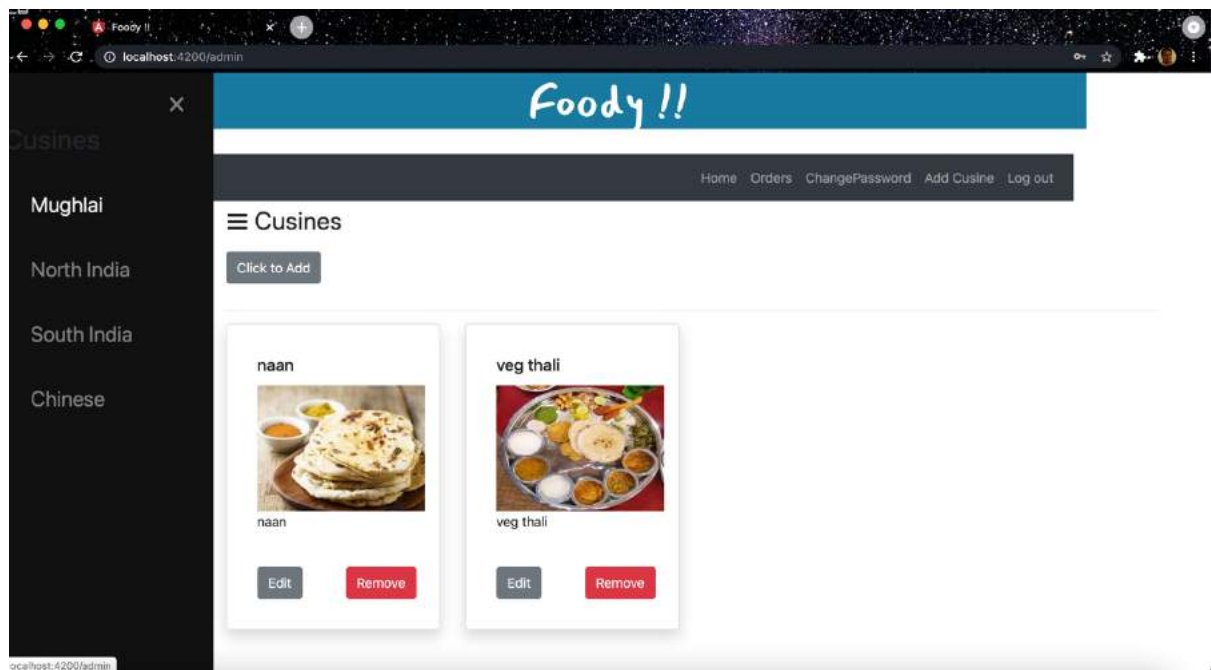
## LANDING PAGE FOR ADMIN USER:



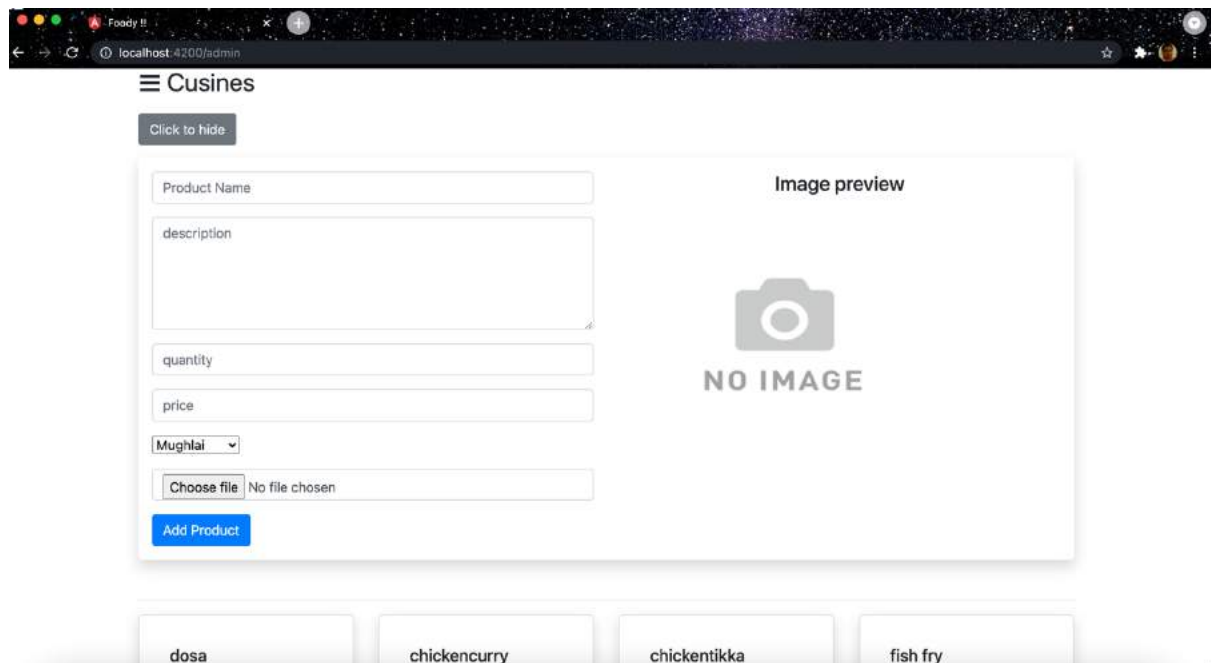
## SIDE MENU TO FILTER BY CUISINE:



FILTER BY CUISINE:



ADD FOOD ITEMS - ADMIN

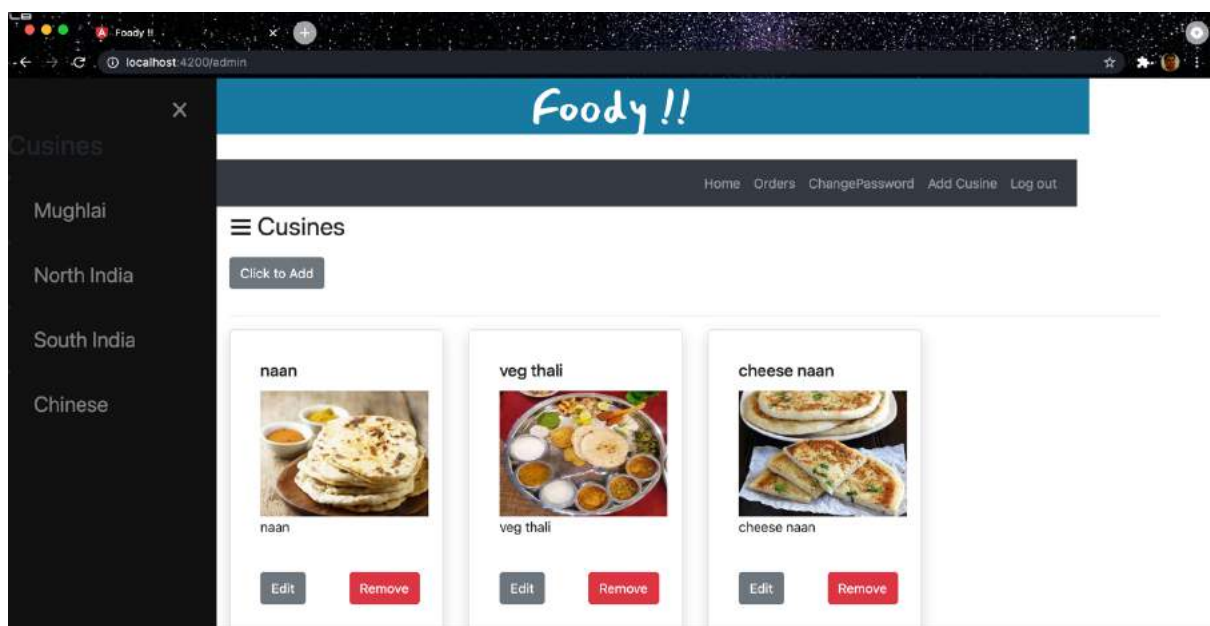




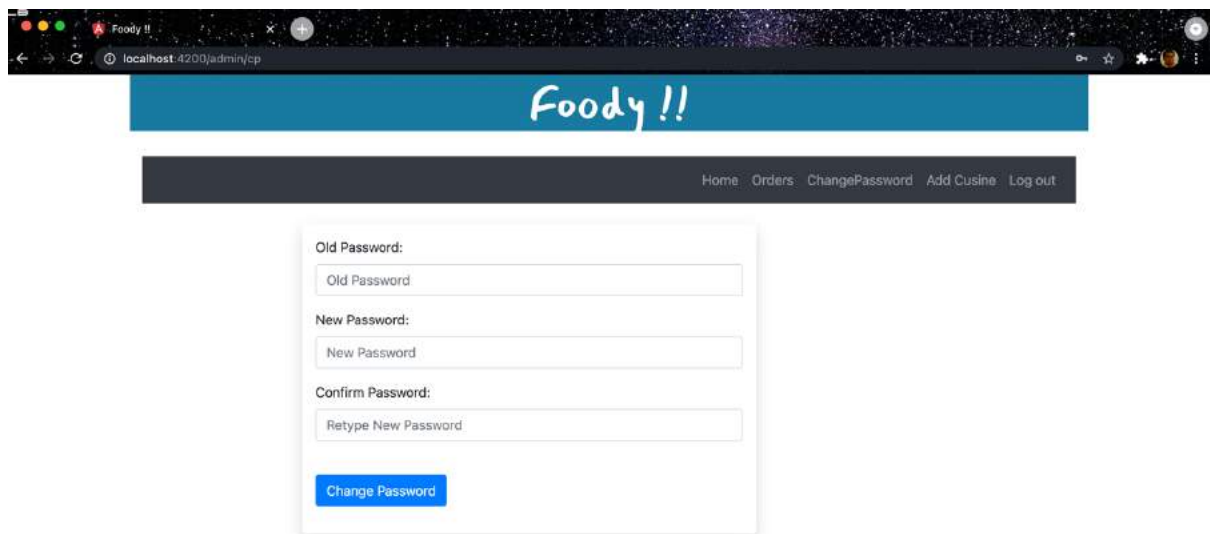
DEMO OF ADD FOOD ITEM WITH CUISINE:

The screenshot shows a web browser window with the URL `localhost:4200/admin`. The page title is "Foody !!". Below the title, there is a "Cusines" section with a "Click to hide" button. The main form for adding a food item is displayed. It includes two text input fields for the name, both containing "cheese naan". There are two numeric input fields for price, with values "1000" and "100". A dropdown menu for "Cuisine" is set to "North India". A "Choose file" button is next to a text field containing "cheese-naan.jpeg". A blue "Add Product" button is at the bottom. To the right of the form is an "Image preview" section showing a stack of cheese naans.

COME BACK TO LANDING PAGE OF ADMIN:



## CHANGE PASSWORD - ADMIN



The screenshot shows a web browser window with the URL `localhost:4200/admin/cp`. The page features a blue header with the "Foody!!" logo and a dark navigation bar with links: Home, Orders, ChangePassword, Add Cuisine, and Log out. The main content area displays a form for changing the password. The form includes three input fields: "Old Password" (containing "Old Password"), "New Password" (containing "New Password"), and "Confirm Password" (containing "Retype New Password"). A blue "Change Password" button is positioned at the bottom of the form.

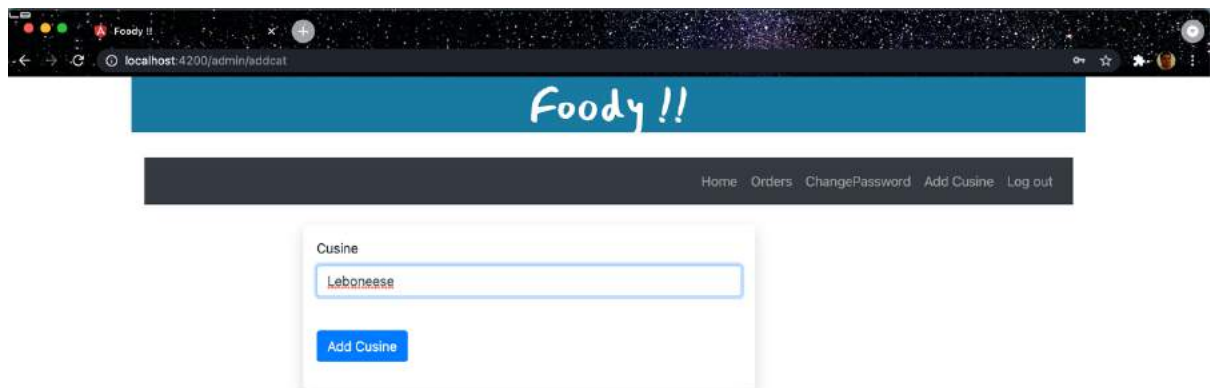
Old Password:

New Password:

Confirm Password:

Change Password

## ADD CUISINE:

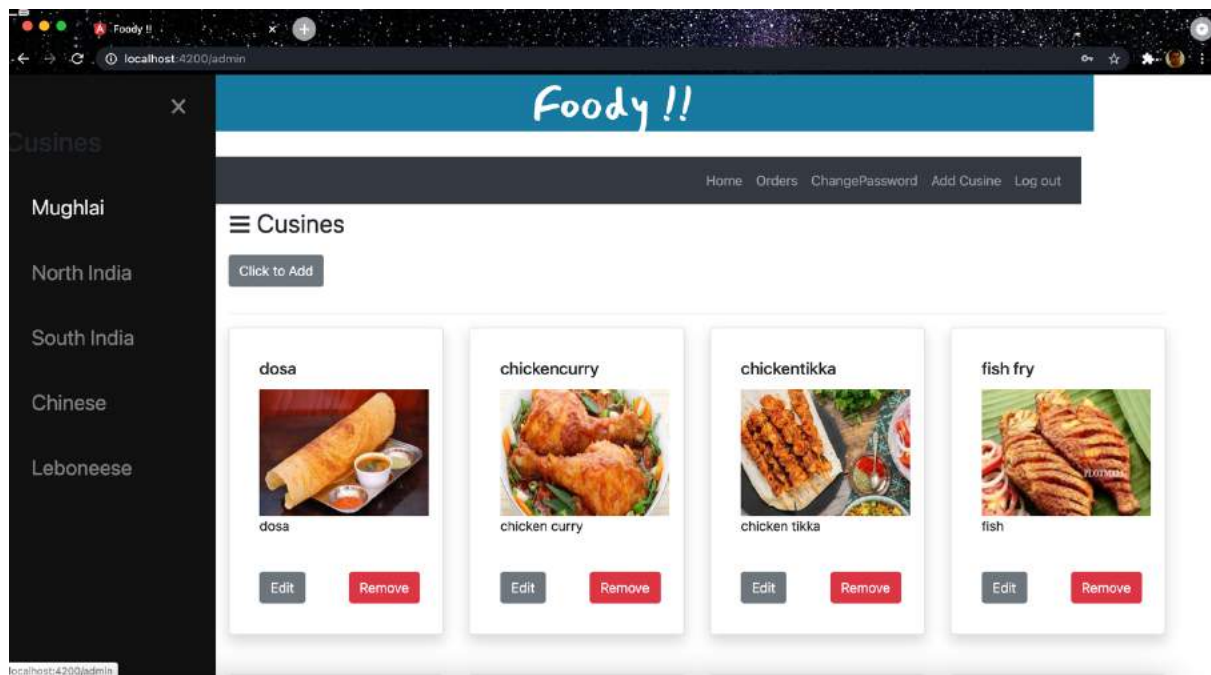


The screenshot shows a web browser window with the URL `localhost:4200/admin/addcat`. The page features a blue header with the "Foody!!" logo and a dark navigation bar with links: Home, Orders, ChangePassword, Add Cuisine, and Log out. The main content area displays a form for adding a new cuisine. The form includes a single input field labeled "Cuisine" containing the text "Lebanese". A blue "Add Cuisine" button is positioned below the input field.

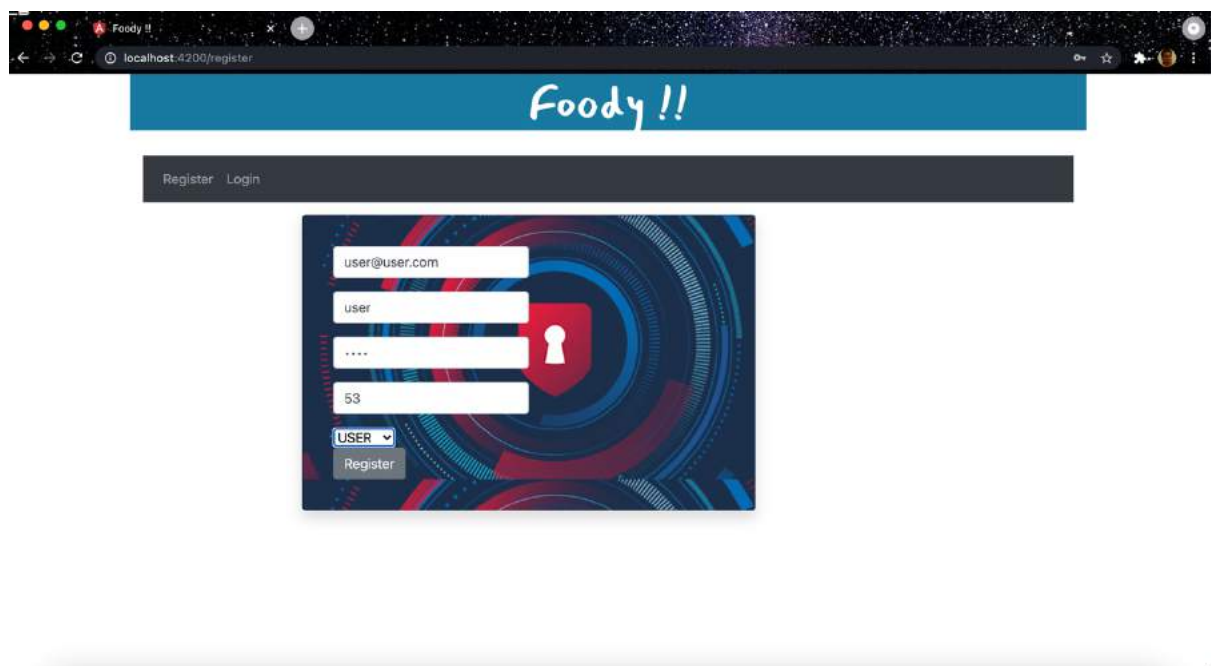
Cuisine

Add Cuisine

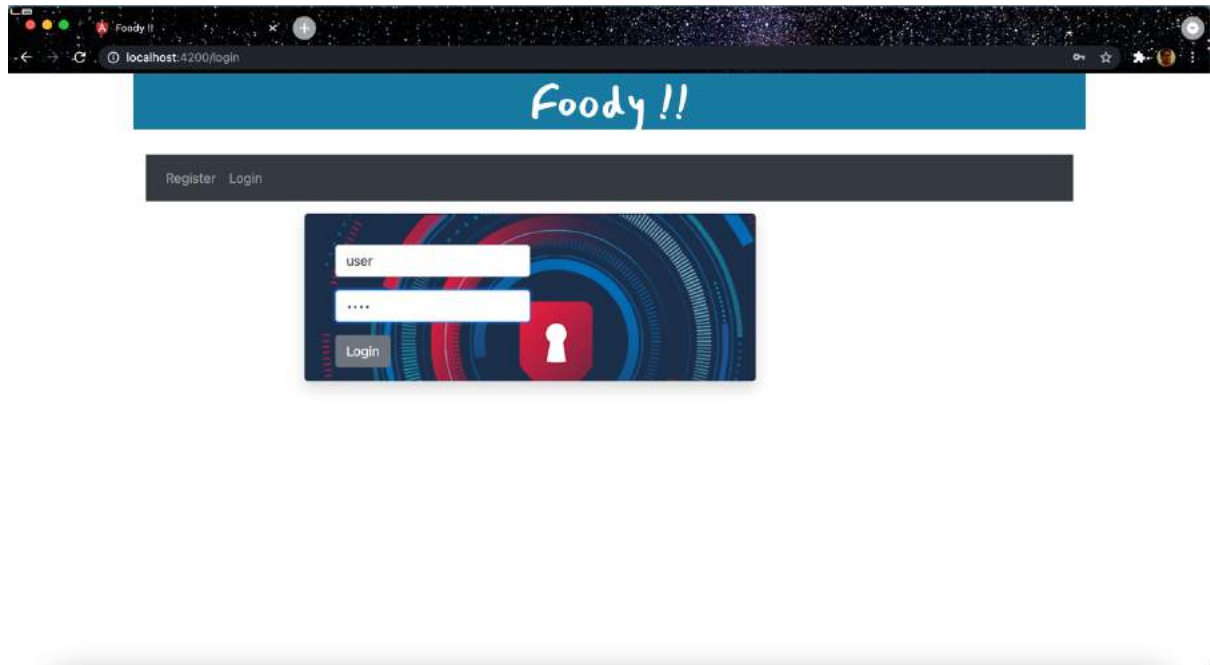
SEE THE NEW CUISINE IN LEFT MENU:



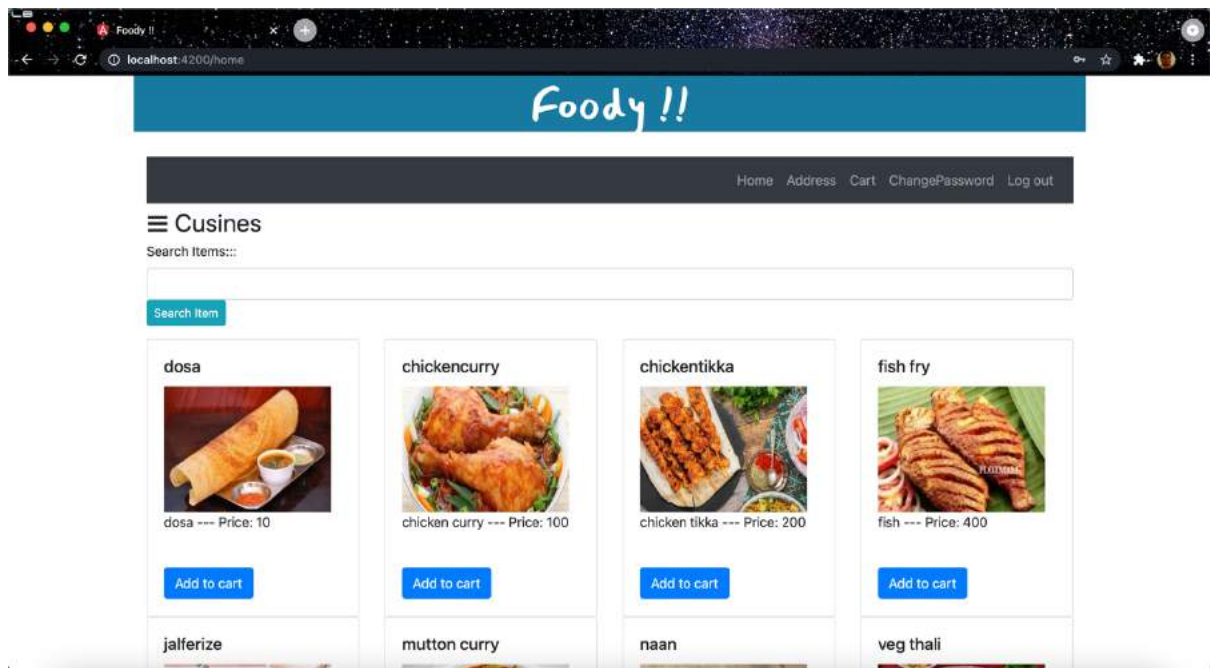
REGISTER END USER:



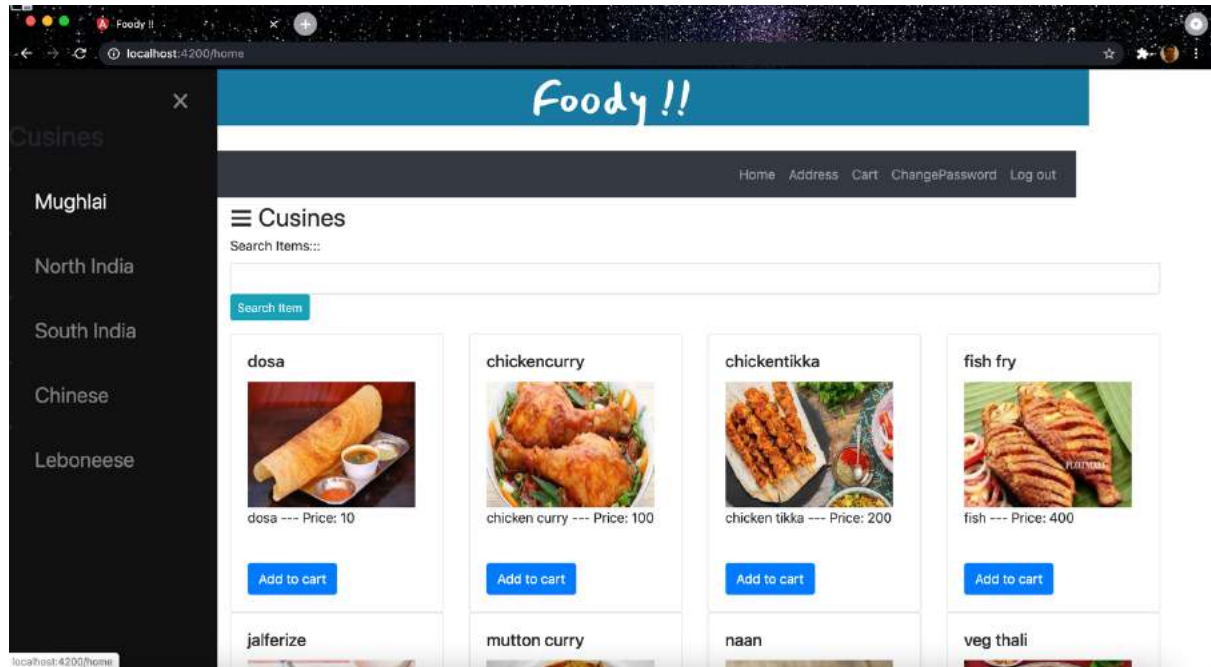
LOGIN END USER:



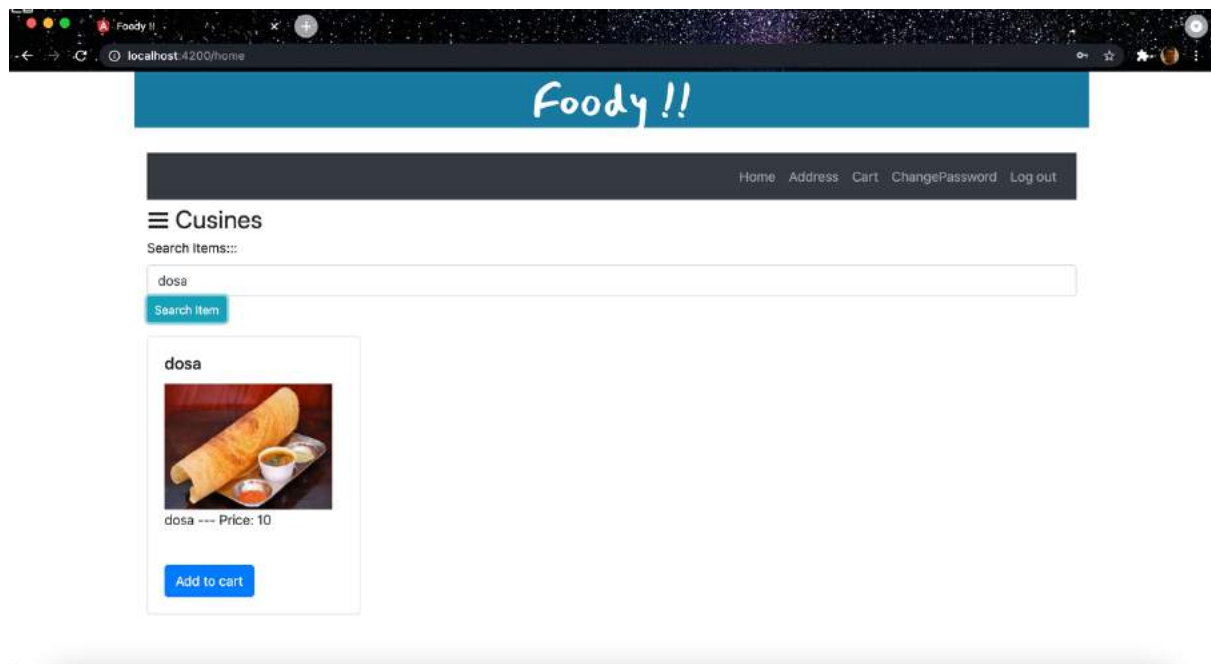
LANDING PAGE FOR END USER:



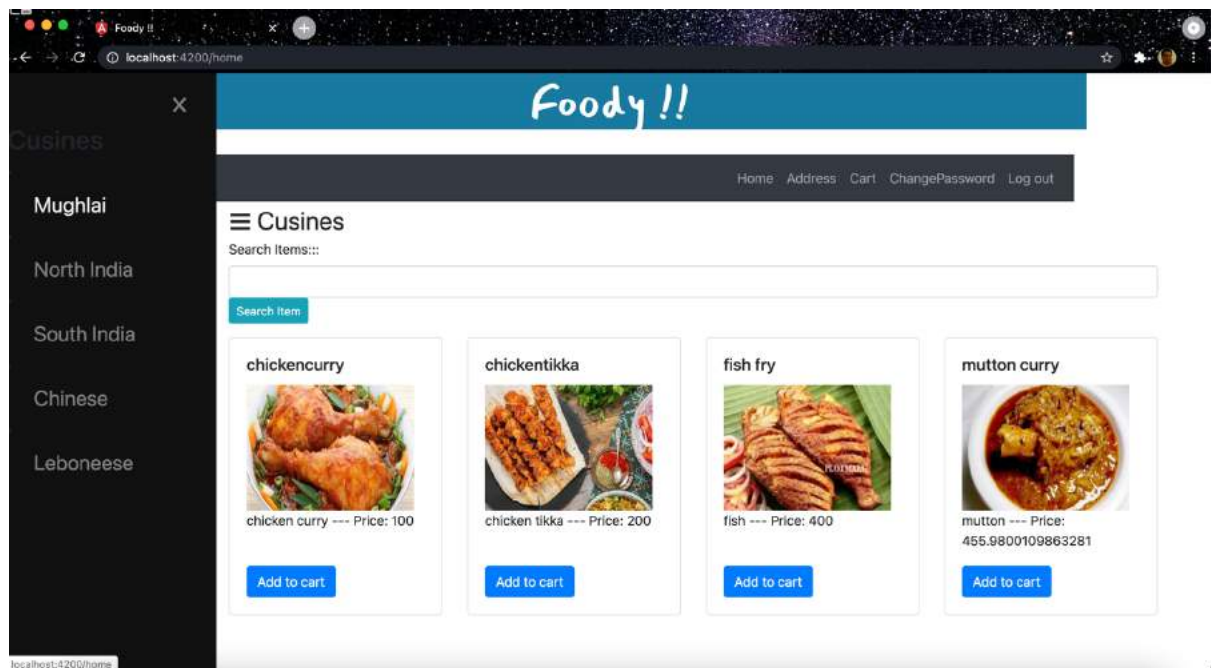
LEFT MENU FOR END USER:



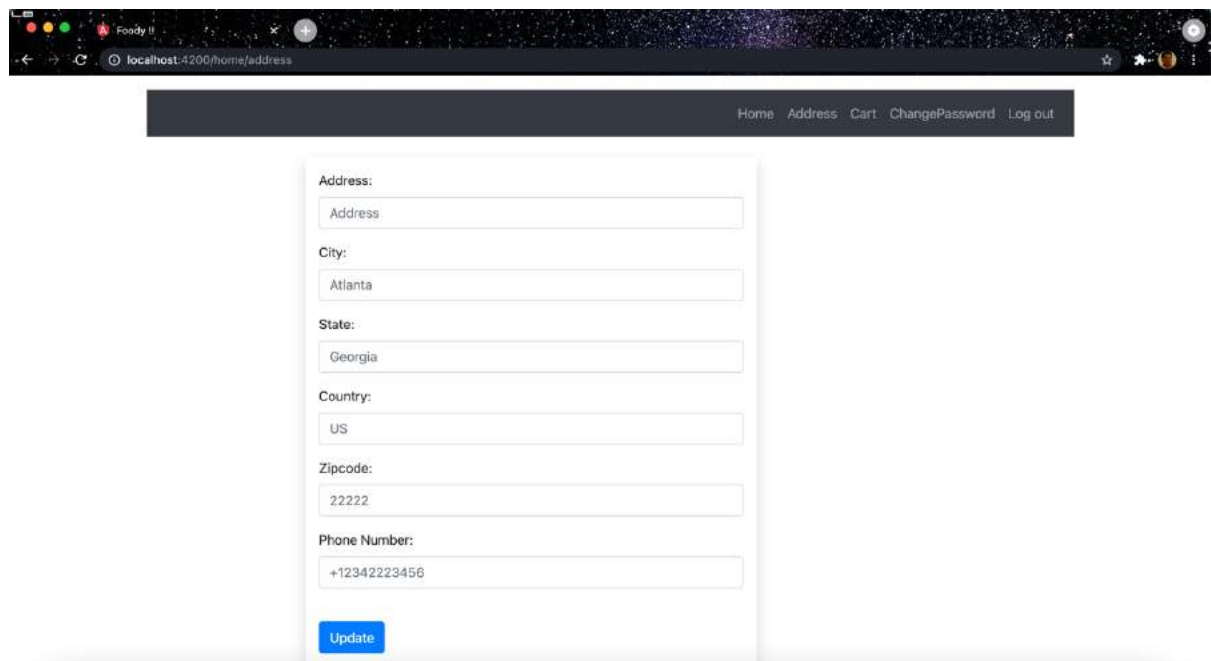
SEARCH WITH KEYWORDS END USER:



## SEARCH BY CUISINE LEFT MENU END USER

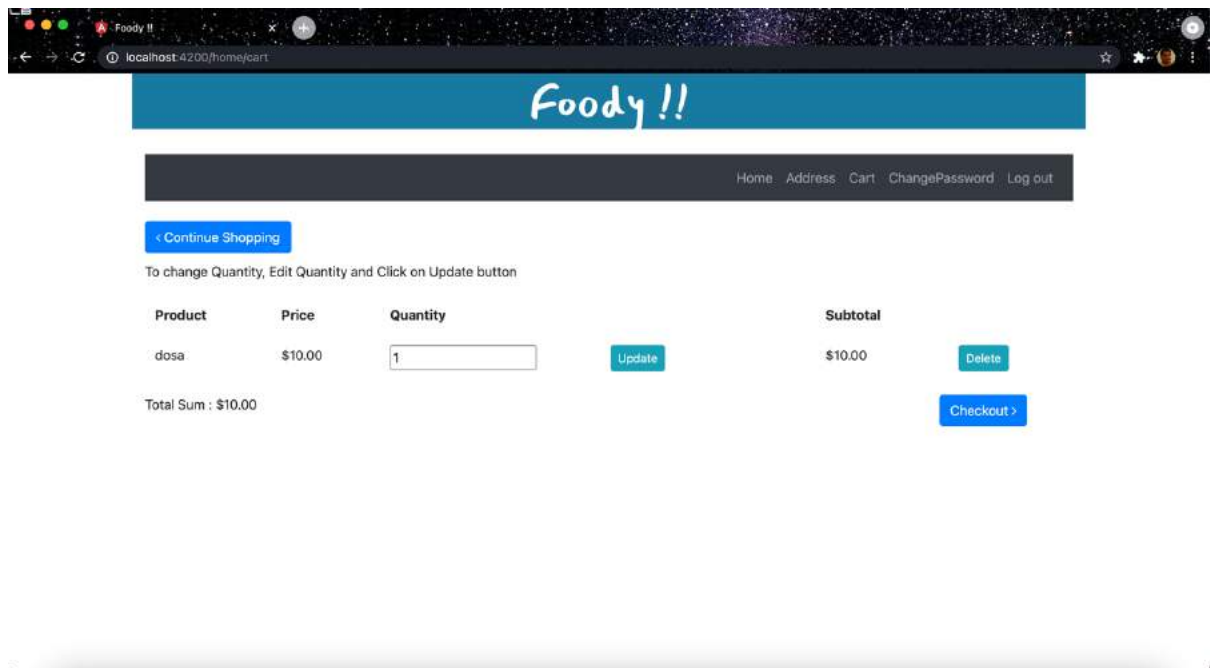


## ADD ADDRESS END USER

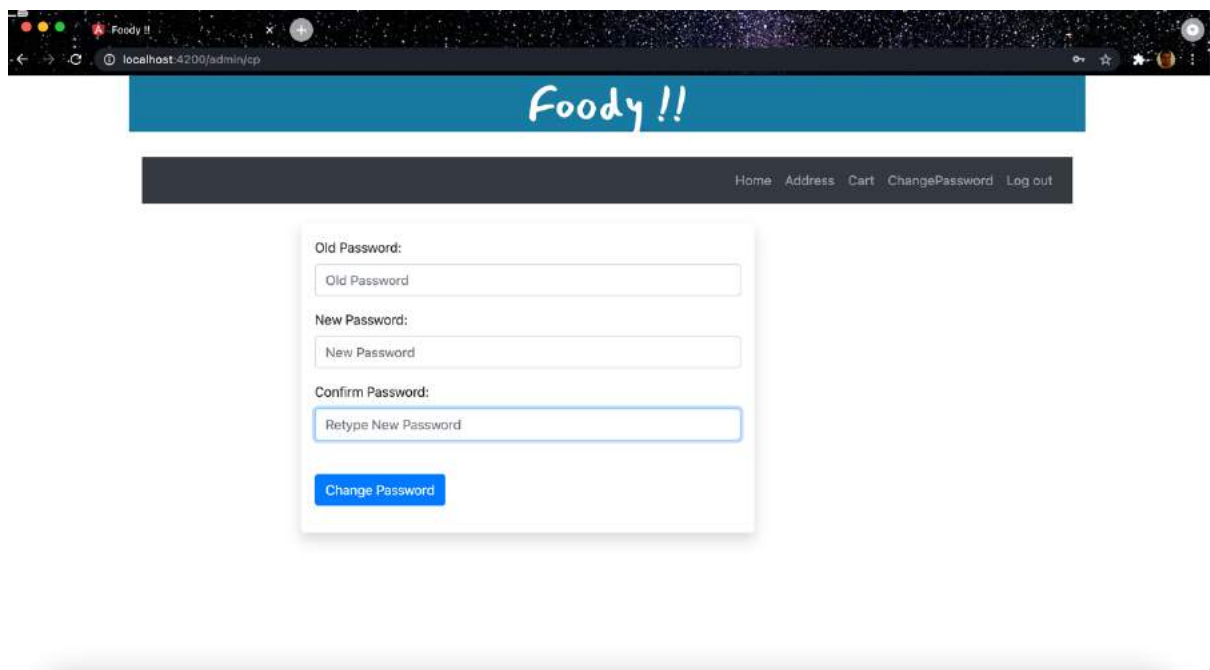




## ADD ITEM TO CART:



## CHANGE PASSWORD END USER:



=====

USING DOCKER TO RUN THE APP ON LOCALHOST:

*##### NOTE: NEEDS DOCKER SERVER RUNNING ON THE HOST #####*

GET INTO THE PROJECT DIRECTORY.

```
$ cd capstone-project
```

BUILD THE DOCKER IMAGE WITH BELOW COMMAND:

```
$ docker build . -t foody:latest
```

RUN THE DOCKER IMAGE ON LOCALHOST

```
$ docker run -p80:80 -p8087:8087 foody:latest
```

ACCESS THE APP AT BELOW URL IN BROWSER:

URL: <http://localhost/>

=====

RUN THE APP ON LOCALHOST WITH BASH COMMANDS: (MAC/LINUX)

```
~/capstone-project$ cd backend-app
```

```
~/capstone-project/backend-app$ mvn install -DskipTests
```

THIS WILL RUN THE BACKEND SPRING BOOT SERVICES

```
~/capstone-project/backend-app $ mvn spring-boot run
```

ACCESS THE APP ON :

<http://localhost:8087>

=====

```
~/capstone-project$ cd front-end-app
```

```
~/capstone-project/front-end-app$ npm install
```

THIS WILL RUN THE FRONTEND WEBSITE

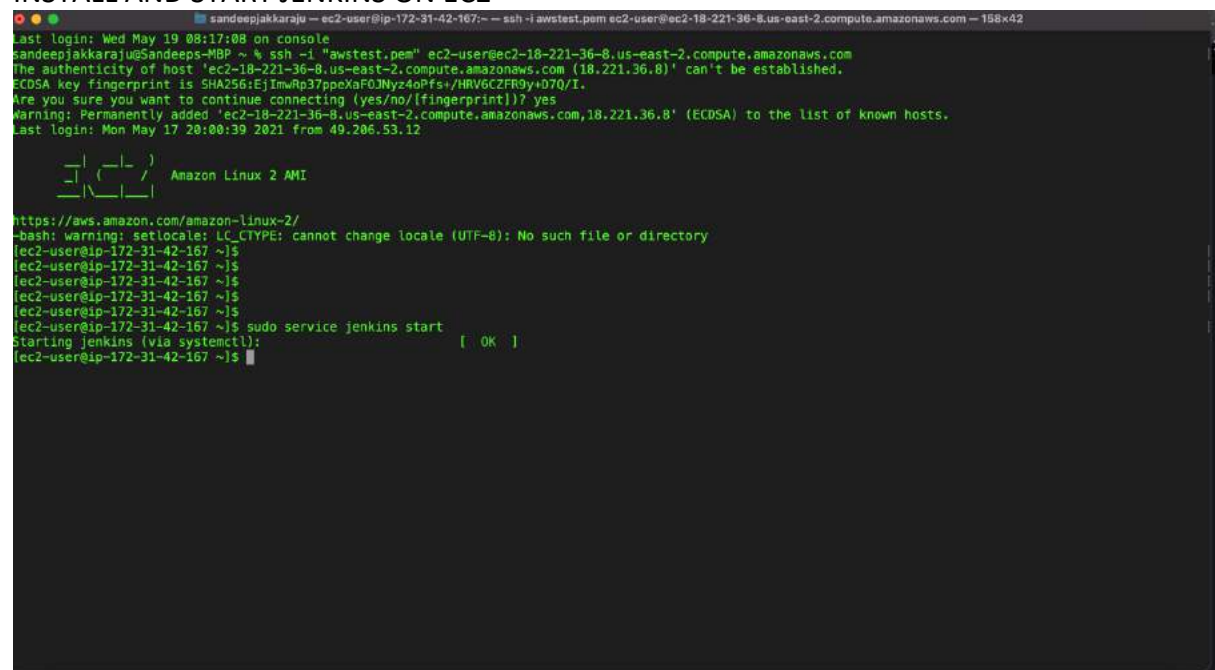
```
~/capstone-project/backend-app $ ng serve
```

ACCESS THE APP ON:

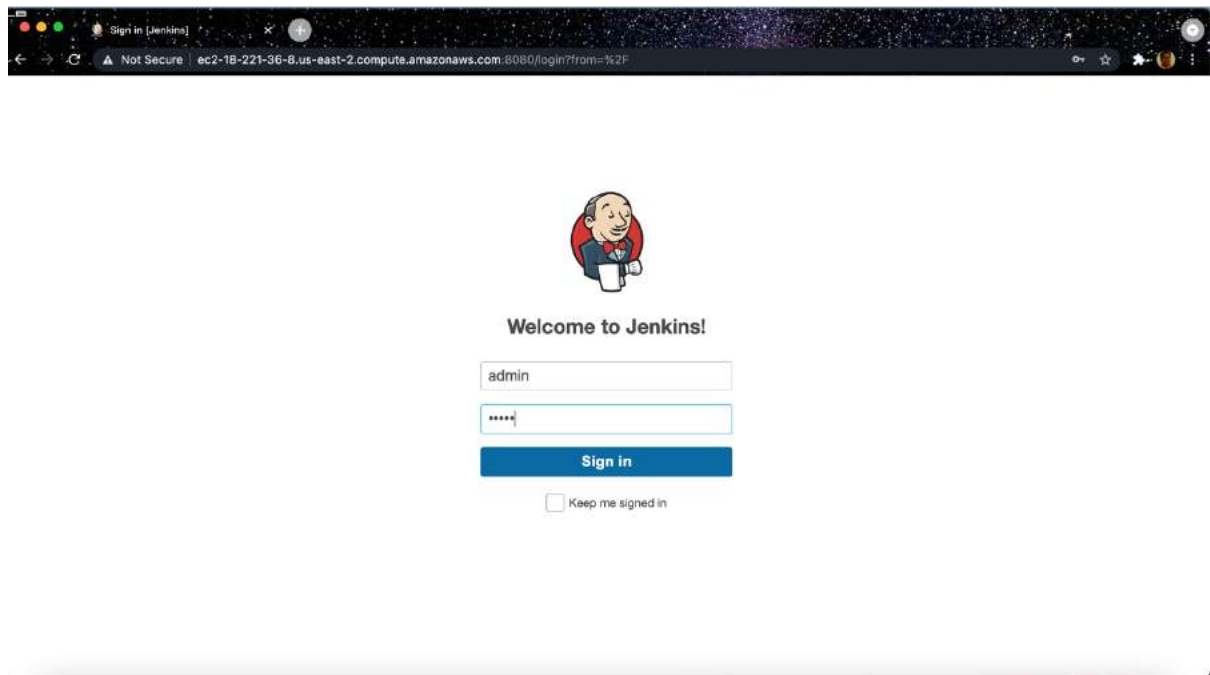
<http://localhost:4200>

=====

## AWS EC2 CONSOLE



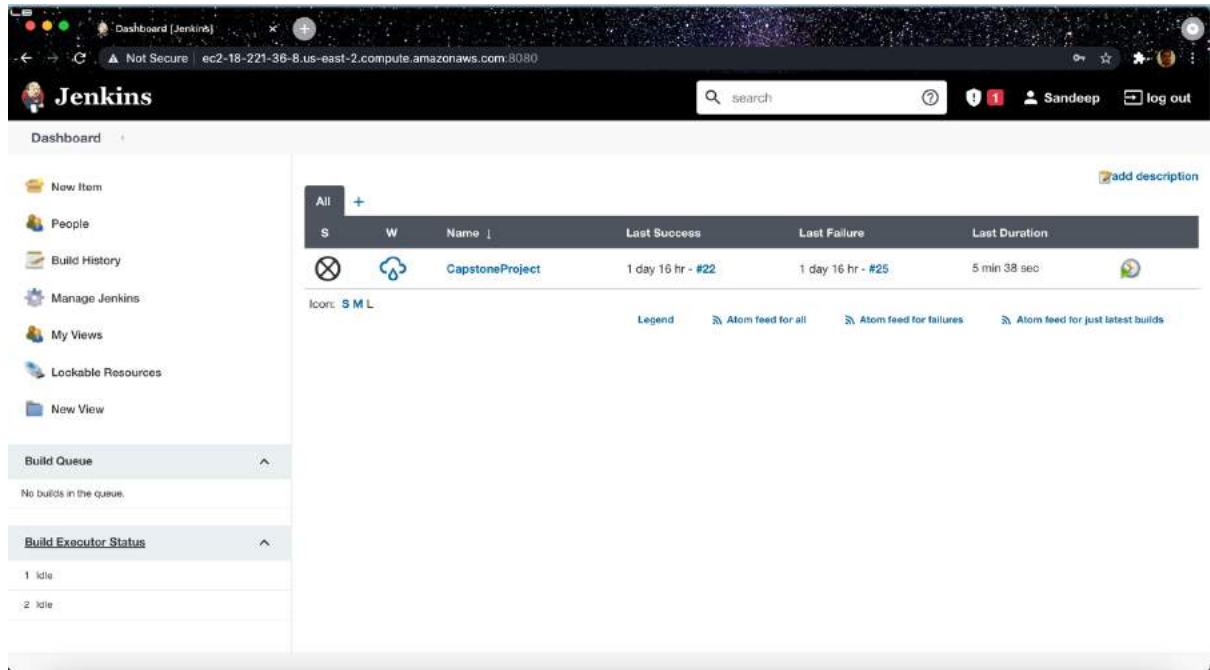
LOGIN TO JENKINS:



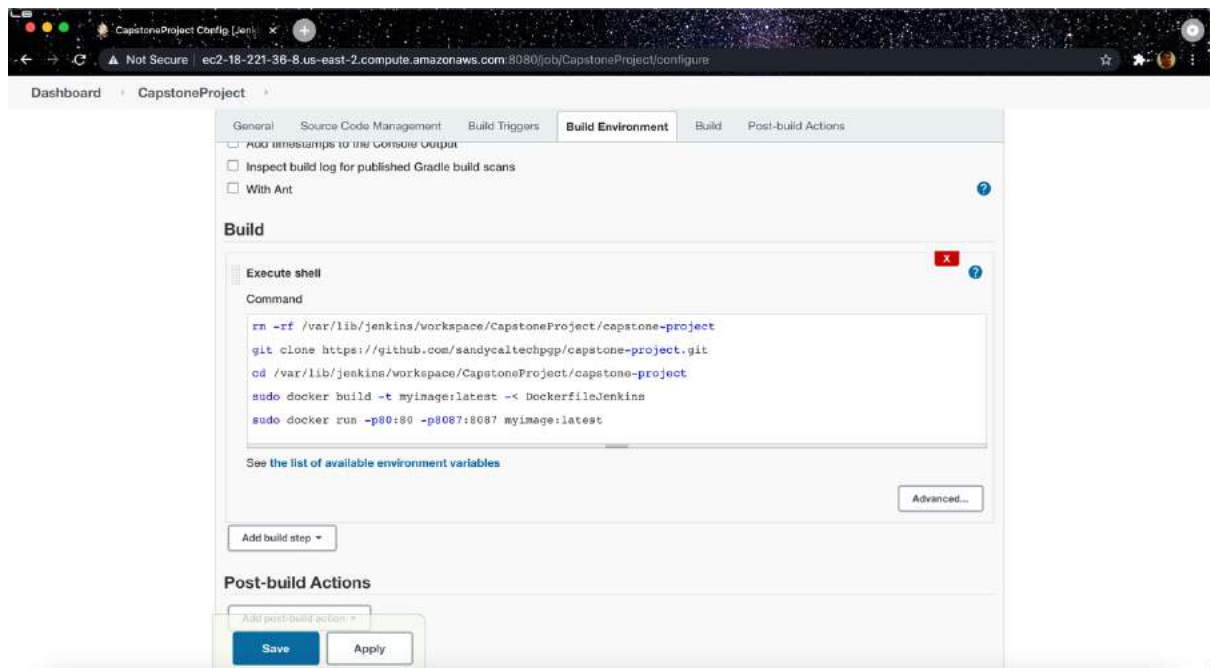
The screenshot shows a web browser window with the Jenkins login page. The address bar displays the URL: `ec2-18-221-36-8.us-east-2.compute.amazonaws.com:8080/login?from=%2F`. The page features the Jenkins logo (a cartoon man in a red suit) and the text "Welcome to Jenkins!". Below this, there are two input fields: the first contains the text "admin", and the second contains five asterisks "\*\*\*\*\*". A blue "Sign in" button is positioned below the password field. At the bottom, there is a checkbox labeled "Keep me signed in" which is currently unchecked.

---

Configure the JENKINS FREE STYLE PROJECT



## JENKINS PROJECT CONFIGURATION: using docker



START DOCKER SERVER ON EC2 INSTANCE

NOTE:: PLEASE HAVE MORE DISK SPACE IN THE INSTANCE FOR DOCKER TO RUN.

```
sandeepjakkaraju — ec2-user@ip-172-31-42-167:~ — ssh -i aws-test.pem ec2-user@ec2-18-221-36-8.us-east-2.compute.amazonaws.com — 158x42
[ec2-user@ip-172-31-42-167 ~]$ sudo dockerd &
```

## START RDS – MYSQL

The screenshot shows the Amazon RDS console interface. On the left is a navigation menu with options like Dashboard, Databases, Query Editor, Performance Insights, Snapshots, Automated backups, Reserved instances, Proxies, Subnet groups, Parameter groups, Option groups, Events, Event subscriptions, Recommendations, and Certificate update. The main content area is titled 'Databases' and includes a search bar, a 'Group resources' toggle, and buttons for 'Modify', 'Actions', 'Restore from S3', and 'Create database'. Below these is a table listing database instances.

DB Identifier	Role	Engine	Region & AZ	Size	Status	CPU	Current activity
foodydatabase	Instance	MySQL Community	us-east-2b	db.t2.micro	Starting	-	-

At the bottom of the console, there is a footer with 'Feedback', 'English (US)', and copyright information for Amazon Internet Services Private Ltd. or its affiliates, along with links to Privacy Policy, Terms of Use, and Cookie preferences.

----THE END----