

Kitchen Story

This document contains the following.

- Description
- Sprint planning and tasks achieved
- Core concepts used in the project
- Flowchart of the application
- Links to the GitHub repository
- Developer Detail

Description

Project objective:

Kitchen Story is an e-commerce portal that lets people shop basic food items on their website. The website needs to have the following features:

- A search form in the home page to allow entry of the food items to be purchased by the customer.
- Based on item details entered, it will show available food items with price.
- Once a person selects an item to purchase, they will be redirected to the list of available items. In the next page, they are shown the complete breakout of the order and details of the payment to be made in the payment gateway. When payment is done, they are shown a confirmation page with details of the order.

For the above features to work, there will be an admin backend with the following features:

- Admin login page where admin can change password after login if he wants to
- A master list of food items available for purchase
- A functionality to add or remove food items

Developer Detail

Mahendra Kumar Singh

mahendrakumarsingh9893@gmail.com

Links to the GitHub repository:

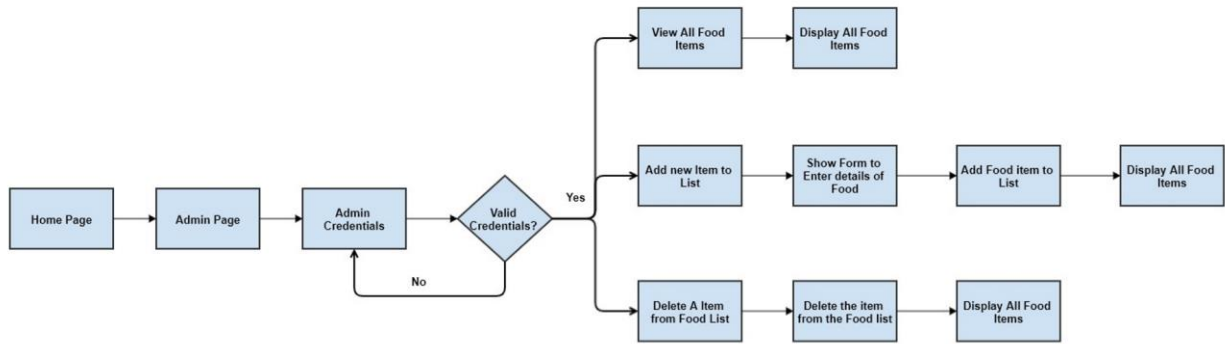
<https://github.com/Mahendra1272/Kitchen-Story.git>

Pushing the code to GitHub Repository:

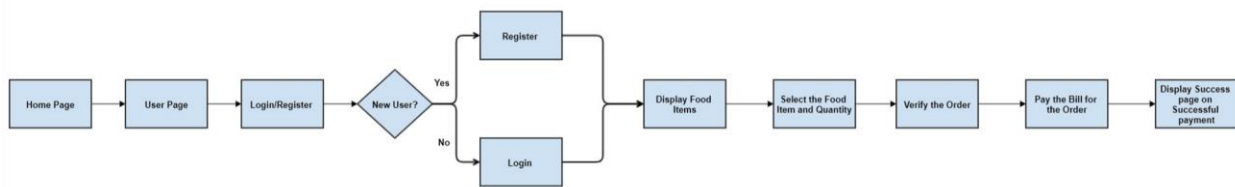
- Open Git Bash and navigate to the folder where you have created your files.
cd Kitchen-Story\Kitchen Story
- Initialize the repository using the below command **git init**
- Add all the files to your git repository using the below command **git add .**
- Commit the changes using the below command **git commit -m "Initial commit"**
- Add the URL for the remote repository where your local repository will be pushed **git remote add origin https://github.com/Mahendra1272/Kitchen Story.git**
- Push the files to the folder you initially created using below command **git push -u origin master**

Flow of the Application

Below is the High level flow diagrams which shows the flow of the SprotyShoes.com application **Admin Portal** :



User Portal :



Demonstrating the Product capabilities, Appearance, and User interactions

1. index.html :
Entry point for the Application which displays App-Route component

```
<!doctype html>

<html lang="en">

<head>

  <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js" integrity="sha384-
DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVY Uew+OrCXaRkfj"
crossorigin="anonymous"></script>

  <script src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.0/dist/js/bootstrap.bundle.min.js"
integrity="sha384-Piv4xVNRyMGpqrS2by6br4gNJ7DXjqk09RmUpJ8jgGtD7zP9yug
3goQfGII0yAns" crossorigin="anonymous"></script>

  <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js" integrity="sha384-
DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVY Uew+OrCXaRkfj"
crossorigin="anonymous"></script>

  <script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js"
integrity="sha384-9/reFTGAW83EW2RDu2S0VVKalzap3H66lZH81PoYlFhbGU+6BZ
p6G7niu735Sk7lN" crossorigin="anonymous"></script>

  <script src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.0/dist/js/bootstrap.min.js"
integrity="sha384-YQ4JLhgyBLPDQt//I+STsc9iw4uQqACwlvpslubQzn4u2UU2UF
M80nGisd026JF" crossorigin="anonymous"></script>

  <meta charset="utf-8">

  <title>Kithcen Story</title>
  <base href="/">

  <meta name="viewport" content="width=device-width, initial-scale=1">

  <link rel="icon" type="image/x-icon" href="favicon.ico">

  <!-- font awesome cdn link -->

  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/font-awesome@5.15.3/css/all.min.css">
```

>

```
<!-- font Baloo+Bhai link -->  
<link rel="stylesheet"  
href="https://fonts.googleapis.com/css?family=Baloo+Bhai|Bree+Serif&display=swap" >
```

```
<!-- custom css file link -->  
<link rel="stylesheet" href="style.css">  
</head>  
<body>  
  <app-root></app-root>  
</body>  
</html>
```

2. app-routing.module.ts :
Typescript file contains all the Routing information
import { NgModule
} from '@angular/core'; import { RouterModule, Routes } from
'@angular/router'; import { AboutusComponent } from
'./aboutus/aboutus.component';

p p

```
import { AddComponent } from './add/add.component'; import {
  AdditemComponent } from './additem/additem.component'; import {
  AdminComponent } from './admin/admin.component'; import { AdpwdComponent
} from './adpwd/adpwd.component'; import { BuyComponent } from
  './buy/buy.component'; import { ContactusComponent } from
  './contactus/contactus.component';

import { FinalComponent } from './final/final.component'; import {
  FoodItemsComponent } from './food-items/food-items.component'; import {
  HomeComponent } from './home/home.component'; import { LoginComponent } from
  './login/login.component'; import { PayComponent } from './pay/pay.component'; import
  { RegisterComponent } from './register/register.component'; import {
  SfooditemsComponent } from './sfooditems/sfooditems.component';

const routes: Routes = [
  {path:'home',component:HomeComponent},
  {path:'Fooditems',component:FoodItemsComponent},
  {path:'sFooditems',component:SfooditemsComponent},
  {path:'about',component>AboutusComponent},
  {path:'contact',component:ContactusComponent},
  {path:'login',component:LoginComponent},
  {path:'register',component:RegisterComponent},
  {path:'Fooditems/:id',component:BuyComponent},
```

p p

```
{path:'buy',component:BuyComponent},
{path:'additem',component:AdditemComponent},
{ath:'final',component:FinalComponent},
{ath:'pay',component:PayComponent},
{path:'addi',component:AdditemComponent},
{path:'adpwd',component:AdpwdComponent},
{path:'admin',component:AdminComponent},
{path:'add',component:AddComponent},
{path:'',component:HomeComponent},
{path:'**',component:ContactusComponent}

];
```

```
@NgModule({ imports:
  [RouterModule.forRoot(routes)], exports:
  [RouterModule]
})
export class AppRoutingModule { }
```

3. app.module.ts :

App Component's module file which contains all the Child Components

```
details import { NgModule } from '@angular/core'; import { FormsModule,
ReactiveFormsModule } from '@angular/forms'; import { BrowserModule } from
'@angular/platform-browser'; import { AppRoutingModule } from './app-
routing.module'; import { AppComponent } from './app.component'; import {
```

p p

```
HomeComponent } from './home/home.component'; import { FoodItemsComponent }  
from './food-items/food-items.component'; import { AboutusComponent } from  
 './aboutus/aboutus.component';  
  
import { ContactusComponent } from './contactus/contactus.component'; import {  
LoginComponent } from './login/login.component'; import { RegisterComponent } from  
 './register/register.component'; import { FilterPipe } from 'src/Shared/filter.pipe'; import  
 { HighlightDirective } from 'src/Shared/highlight.pipe'; import { BuyComponent } from  
 './buy/buy.component'; import { HeaderComponent } from  
 './header/header.component'; import { FinalComponent } from './final/final.component';  
import { AdminComponent } from './admin/admin.component'; import { AddComponent  
 } from './add/add.component'; import { AdpwdComponent } from  
 './adpwd/adpwd.component'; import { AdditemComponent } from  
 './additem/additem.component'; import { FooterComponent } from  
 './footer/footer.component'; import { PayComponent } from './pay/pay.component';  
import { SfooditemsComponent } from './sfooditems/sfooditems.component'; import {  
NheaderComponent } from './nheader/nheader.component';
```

```
@NgModule({  
  declarations: [  
    AppComponent,  
    HomeComponent,  
    FoodItemsComponent,  
    AboutusComponent,
```


p p

ContactusComponent,

LoginComponent,

RegisterComponent,

p p

```
FilterPipe,  
HighlightDirective,  
BuyComponent,  
HeaderComponent,  
FinalComponent,  
AdminComponent,  
AddComponent,  
AdpwdComponent,  
AdditemComponent,  
FooterComponent,  
PayComponent,  
SfooditemsComponent,  
NheaderComponent,  
    ],  
    imports: [  
BrowserModule,  
FormsModule,  
AppRoutingModule,  
ReactiveFormsModule  
    ],  
    providers: [], bootstrap:  
    [AppComponent]  
  })  
  export class AppModule { }
```

4. **app.component.css** : CSS file for App Component h5

```
{
```

p p

color:goldenrod; margin-left: auto; margin-right:

auto;

```
} p{ font-style:oblique; size: 15;
```

```
color:saddlebrown;
```

```
}
```

- 5. app.component.html:** HTML file for App Component
which is the entry point for the all Components

```
<router-outlet></router-outlet>
```

6. package.json :
Application JSON file which contains all the Necessary information regarding the
application

```
{  
  "name": "kitchenstory",  
  "version": "0.0.0",  
  "scripts": {  
    "ng": "ng",  
    "start": "ng serve",  
    "build": "ng build",  
    "test": "ng test",  
    "lint": "ng lint",  
    "e2e": "ng e2e"  
  },  
  "private": true,  
  "dependencies": {  
    "@angular/animations": "~10.2.4",  
    "@angular/common": "~10.2.4",  
    "@angular/compiler": "~10.2.4",
```

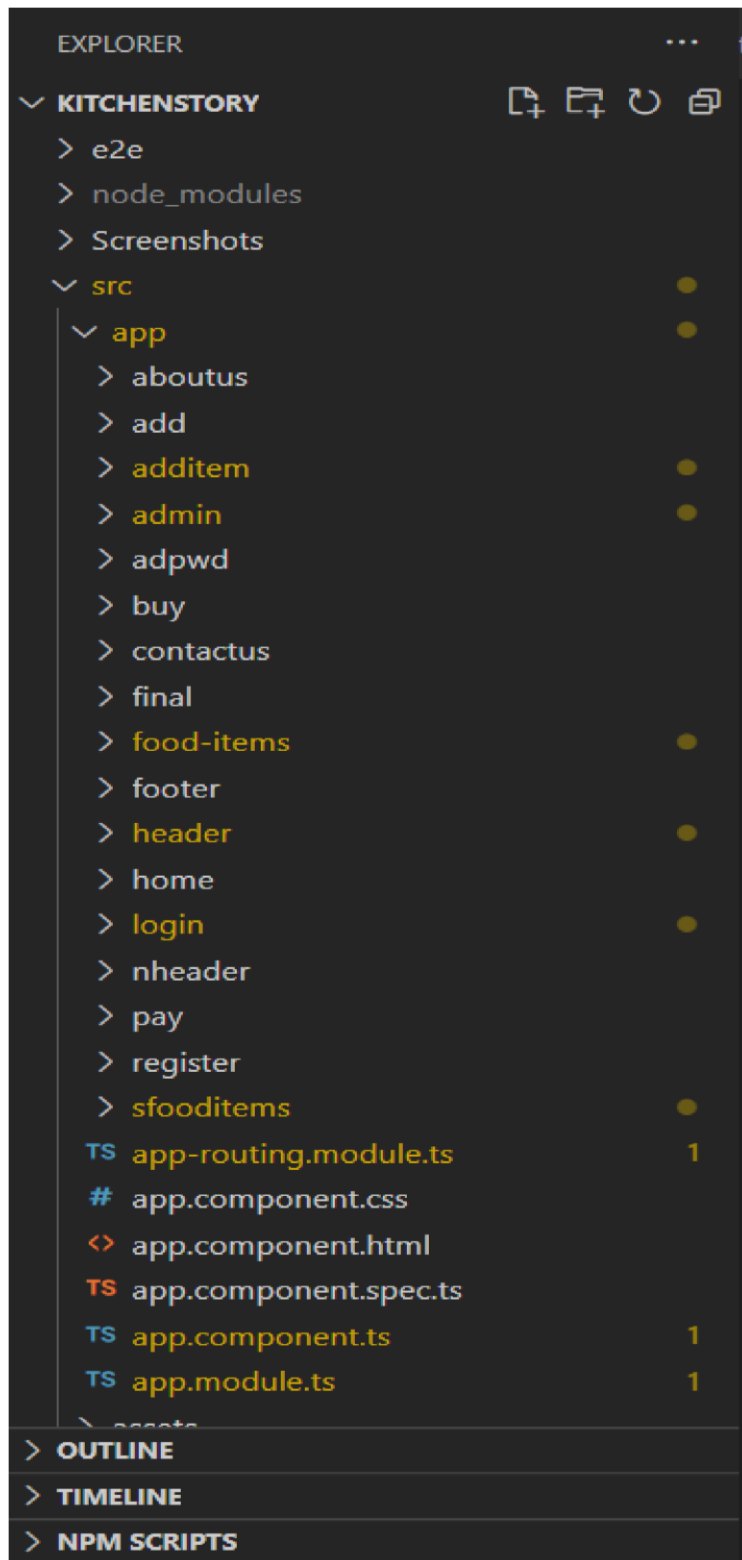
p p

```
"@angular/core": "~10.2.4",
"@angular/forms": "~10.2.4",
"@angular/platform-browser": "~10.2.4",
"@angular/platform-browser-dynamic": "~10.2.4",
"@angular/router": "~10.2.4",
"rxjs": "~6.6.0",
"tslib": "^2.0.0",
"zone.js": "~0.10.2"
  },
  "devDependencies": {
"@angular-devkit/build-angular": "~0.1002.1",
"@angular/cli": "~10.2.1",
"@angular/compiler-cli": "~10.2.4",
"@types/node": "^12.11.1",
"@types/jasmine": "~3.5.0",
"@types/jasminewd2": "~2.0.3",
"codifyer": "^6.0.0",
"jasmine-core": "~3.6.0",
"jasmine-spec-reporter": "~5.0.0",
"karma": "~5.0.0",
"karma-chrome-launcher": "~3.1.0",
"karma-coverage-istanbul-reporter": "~3.0.2",
"karma-jasmine": "~4.0.0",
"karma-jasmine-html-reporter": "^1.5.0",
"protractor": "~7.0.0",
"ts-node": "~8.3.0",
"tslint": "~6.1.0",
"typescript": "~4.0.2"
  }
```

p p

}

Project Structure



Unique Selling Points

1. Admin can create new add new food details
2. Only admin can access Admin portals
3. Everyone has access User portals
4. Admin can Delete the food items
5. Single user can order multiple food items at once
6. Easy maintenance of Dependencies

Sprint 1:

- Create Admin Module
- Admin can change its own password that will be reflected in the database.
- Admin can perform CRUD operations over the list of items available to purchase that will be reflected in the database.
- Take reviews from the product owner after adding these functionalities.

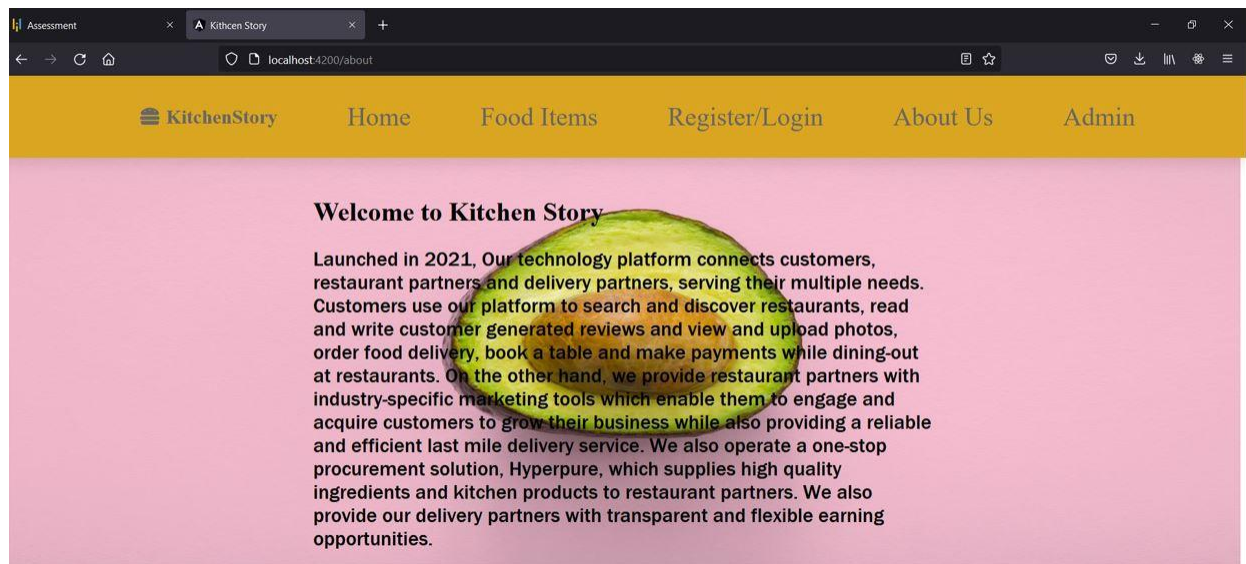
Sprint 2:

- Create User Module
- User can login through its login credentials.
- User can add available food items to cart, add go for payment.
- Take reviews from the product owner after adding these functionalities.

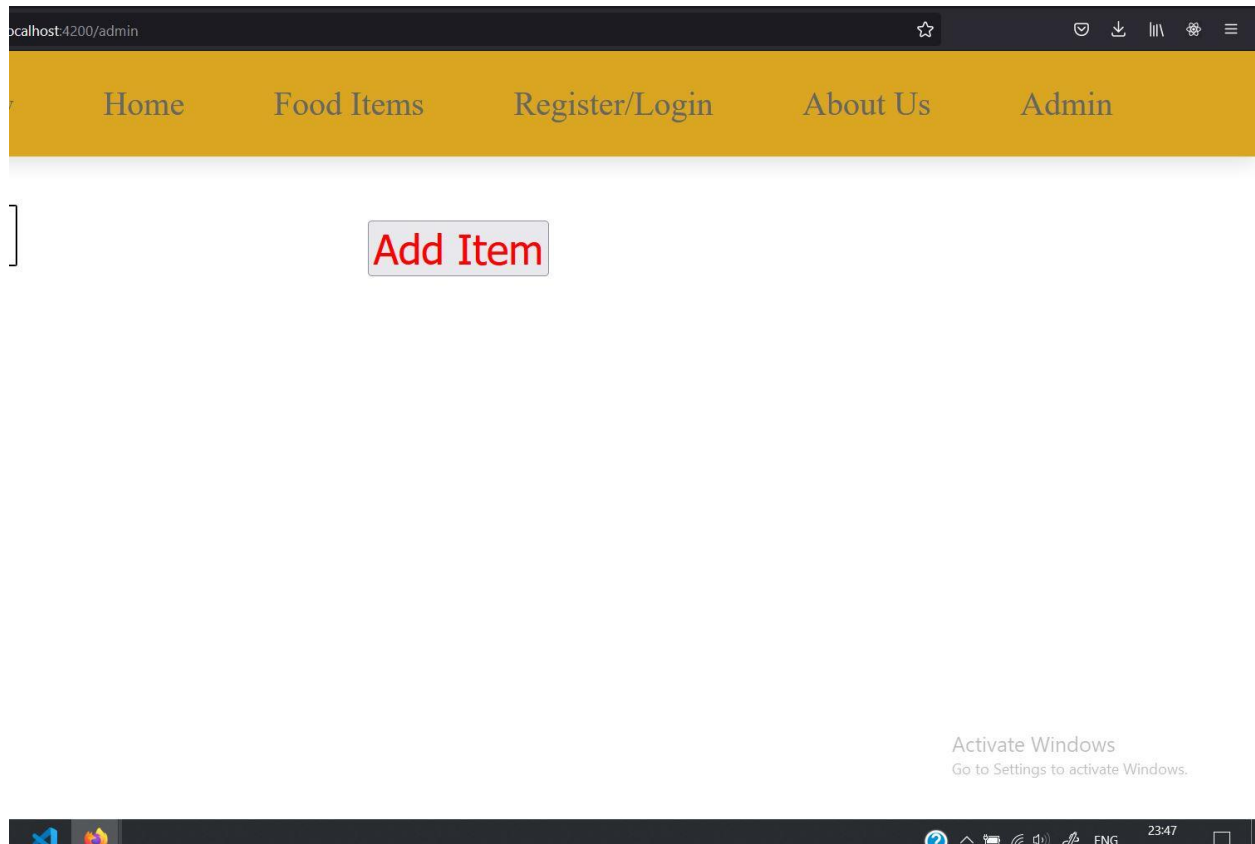
Sprint 3:

- Create backend for User and Admin.
- Create CRUD operations as required for both modules.
- Interlink the modules created in sprint1 and spint2 with this backend.
- Take reviews from the product owner after adding these functionalities.

p p



p p



p p

	Name	Cost	Delete
--	------	------	--------



Add Item

p p

Assessment x Kitchen Story x +

localhost:4200/admin

KitchenStory Home Food Items Register/Login About Us Admin

FoodItem	Name	Cost	Delete
	Burger	199	Delete
	Pizza	250	Delete

Add Item


Activate Windows
Go to Settings to activate Windows.

00:00

p p

Assessment x Kitchen Story x New Tab x +

← → ↻ 🏠 🛡️ 📄 localhost:4200/admin ☆ 🖨️ ⬇️ 📱 ⚙️ ☰

 KitchenStory Home Food Items Register/Login About Us Admin

Item Number :

Item Name :

Cost :

Rating :

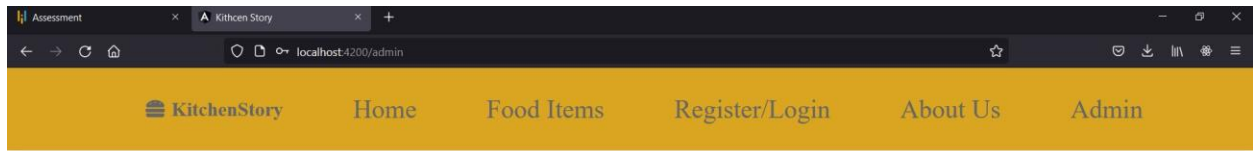
Qnt :

Image URL :

Activate Windows
Go to Settings to activate Windows.

23:59

p p



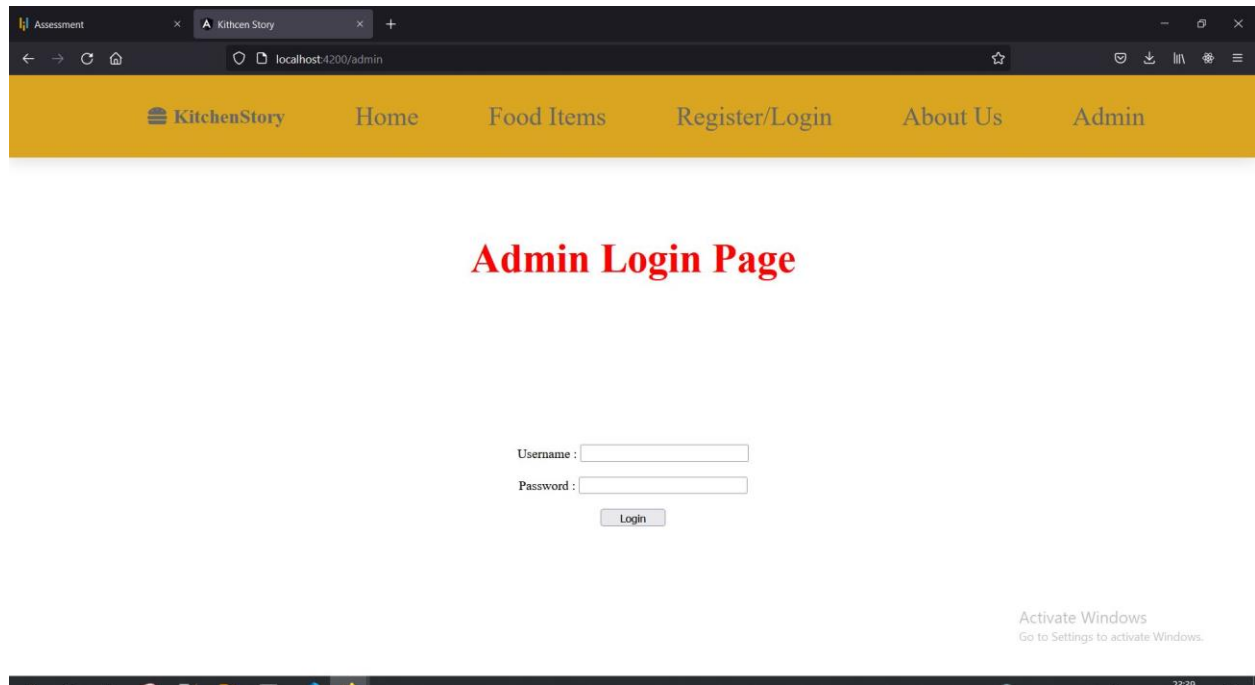
Admin Login Page

Username :

Password :

Activate Windows
Go to Settings to activate Windows.

p p



Cart

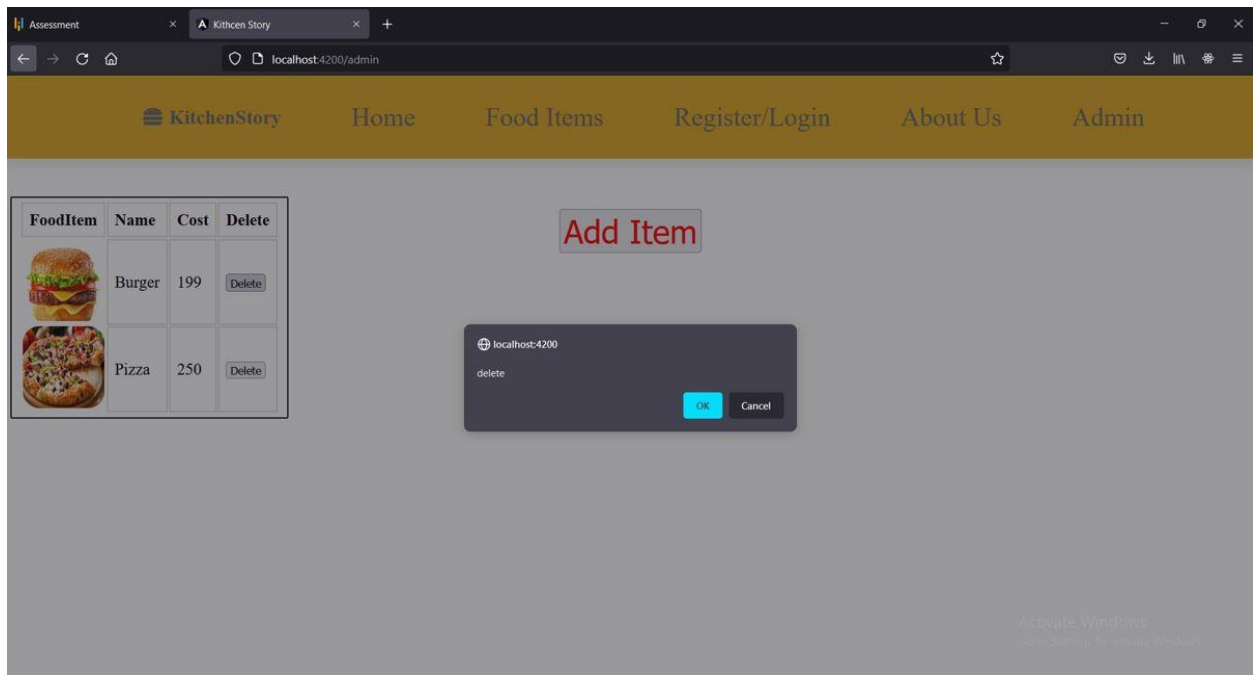
Name	Price	Qunatity	Sub Total	Remove Item
Veg Thali	₹120.00	<div><div>-</div><div>1</div><div>+</div></div>	₹120.00	<div><div></div></div>
Idly	₹50.00	<div><div>-</div><div>3</div><div>+</div></div>	₹150.00	<div><div></div></div>

Total:₹270.00

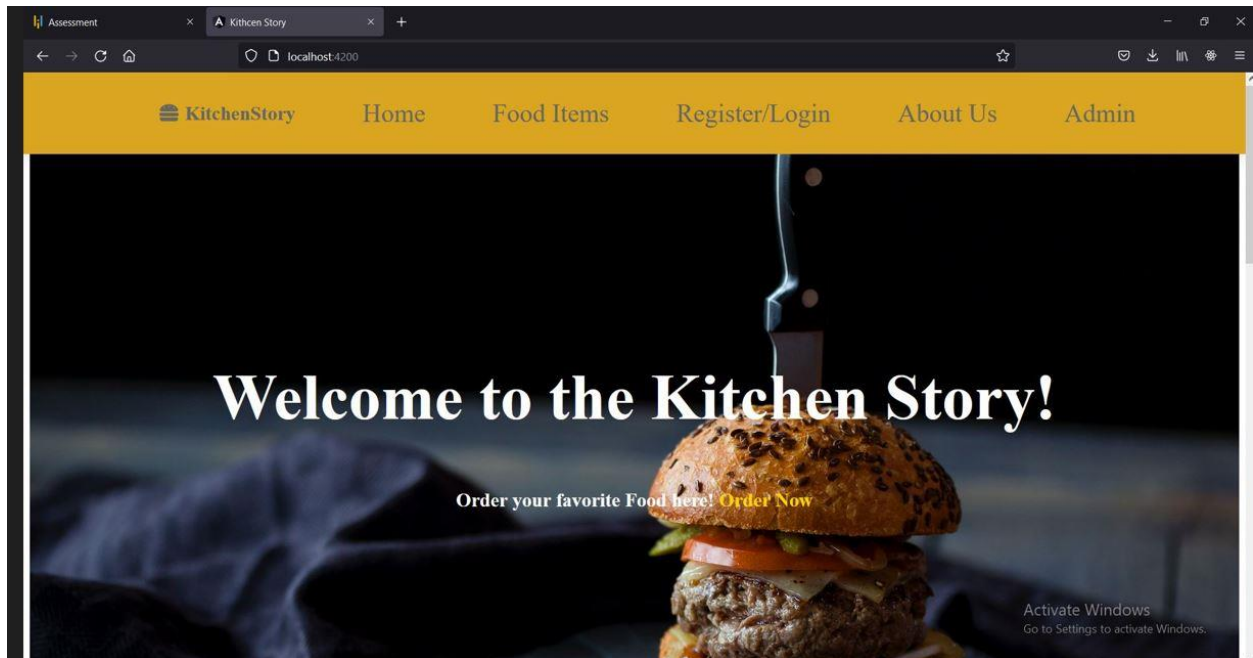
CheckOut

Removeall

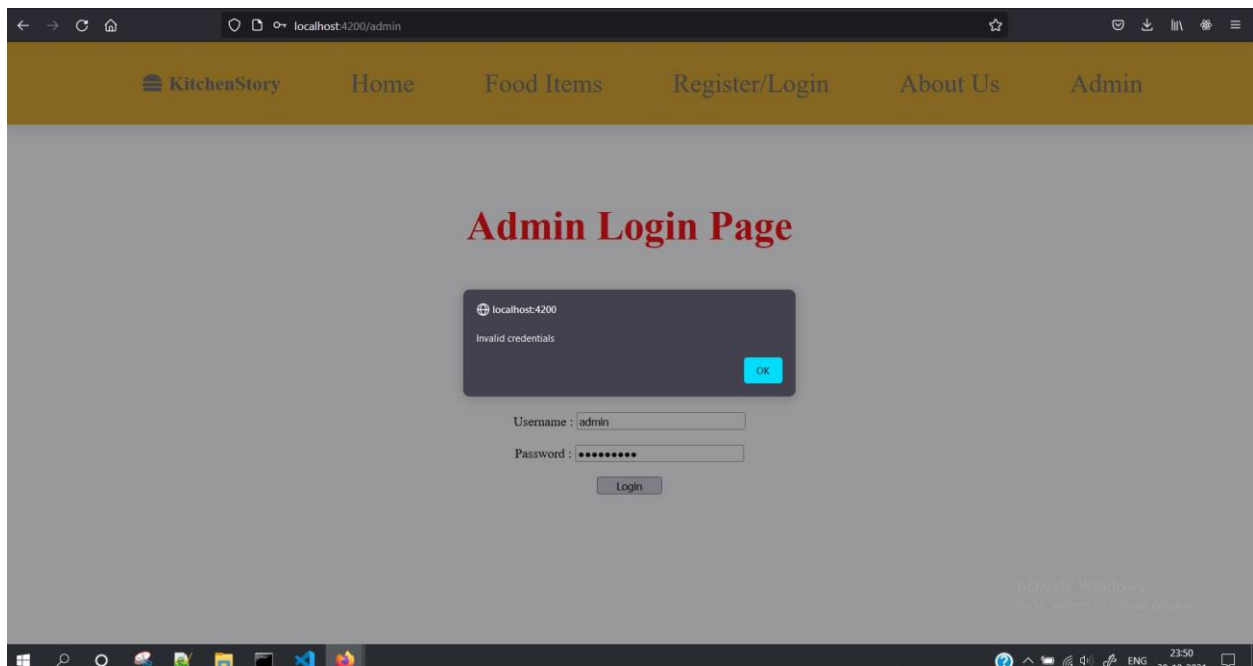
p p



p p



p p




p p

Assessment

Kithcen Story

localhost:4200/admin

 KitchenStory


Home

Food Items

Register/Login

About Us

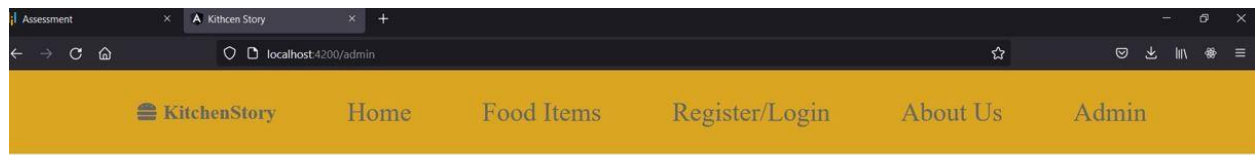
Admin

FoodItem	Name	Cost	Delete
	Pizza	250	<div>Delete</div>

Add Item

Activate Windows
Go to Settings to activate Windows.

p p

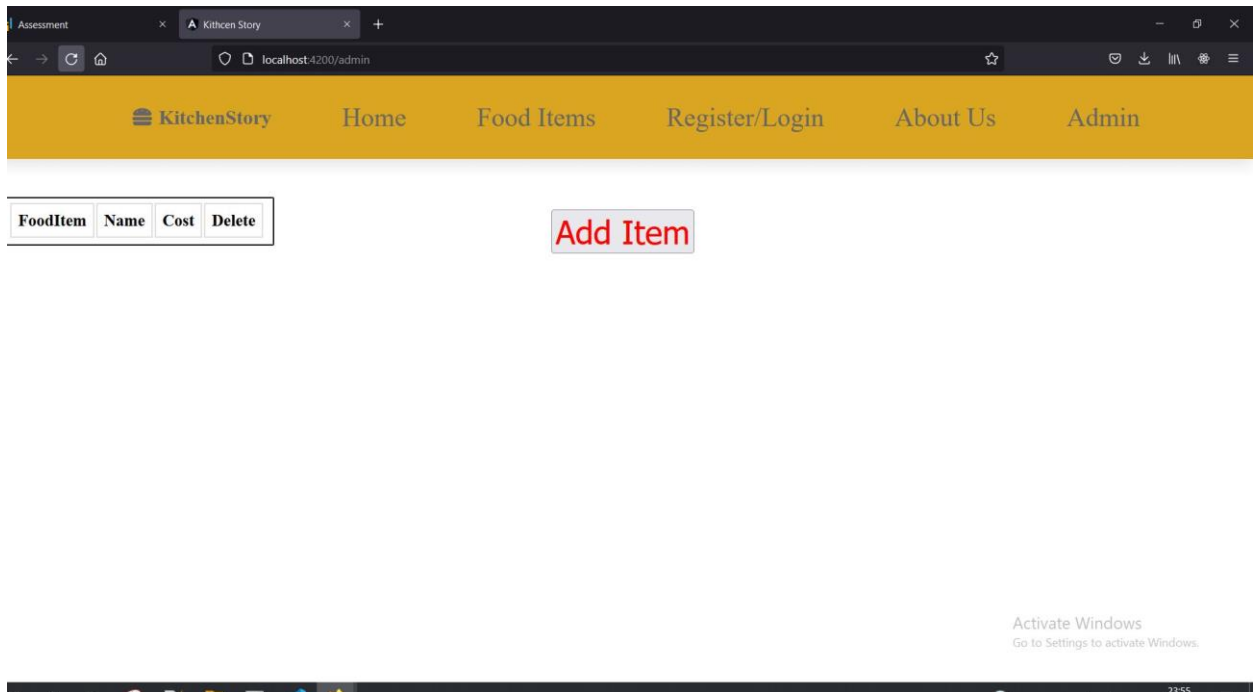


FoodItem	Name	Cost	Delete
	Burger	199	<input type="button" value="Delete"/>

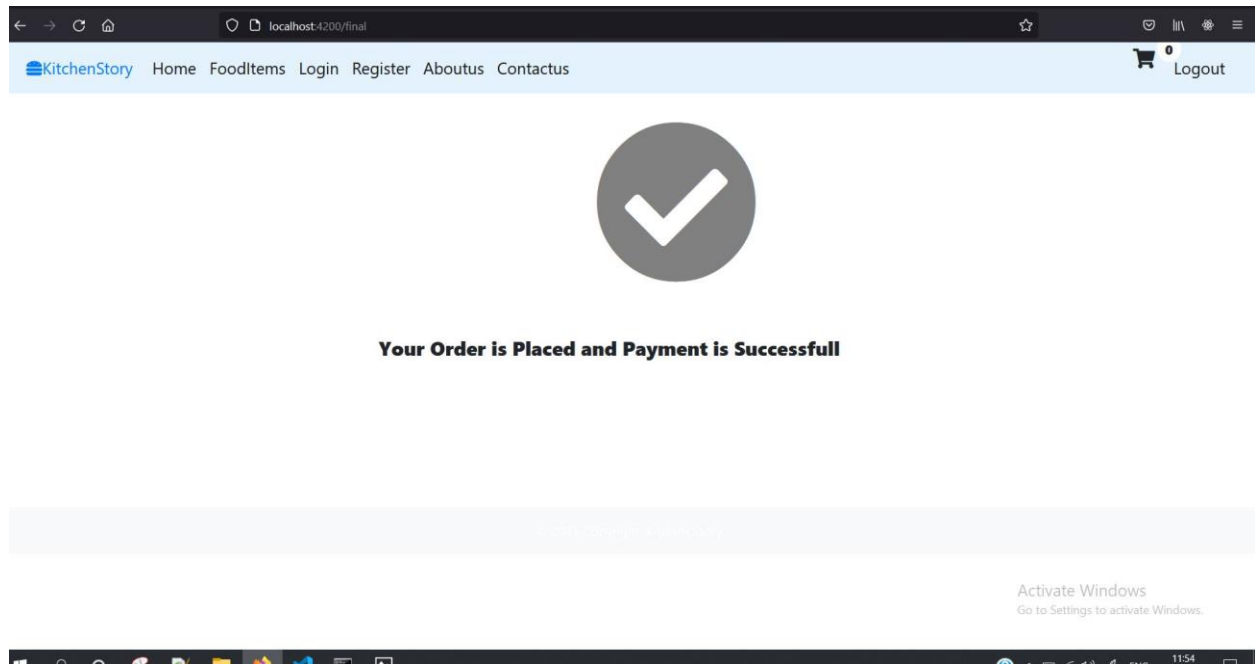
Add Item

Activate Windows
Go to Settings to activate Windows.

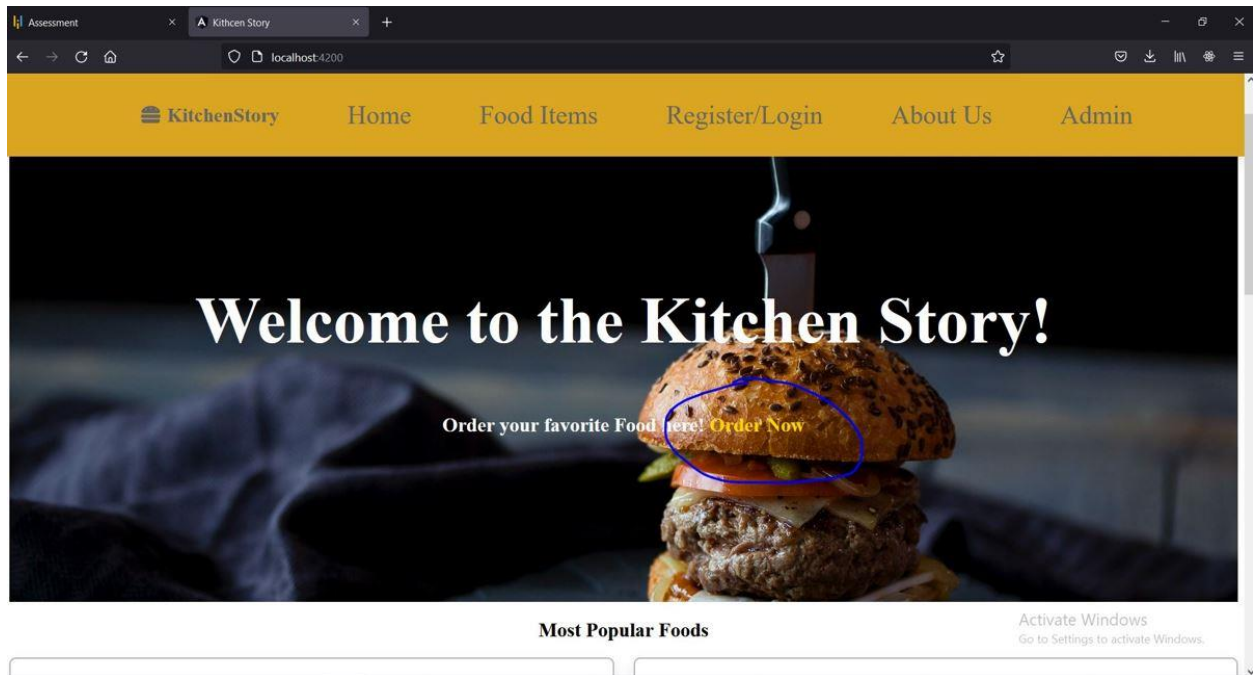
p p



p p



p p



p p

localhost:4200/pay

KitchenStory

Home

FoodItems

Login

Register

Aboutus

Contactus

Please Enter your card Details for purchase

CardNumber

1234567890

Card holder Name

B Sangappa

CVV

...

Expiry Date

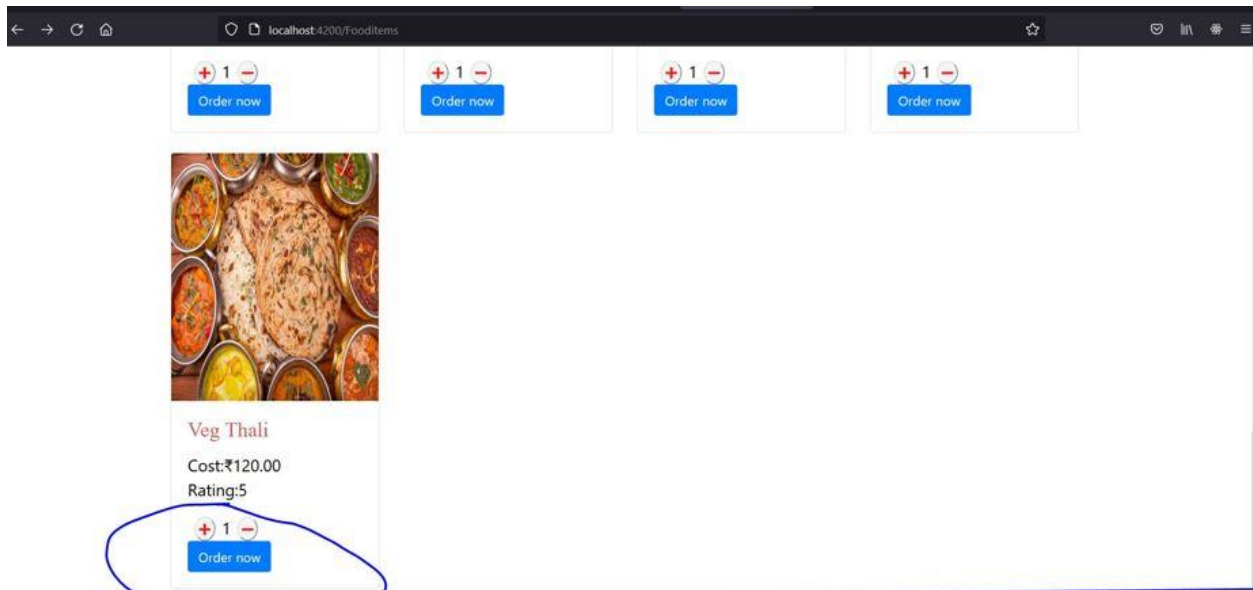
12/23

Submit

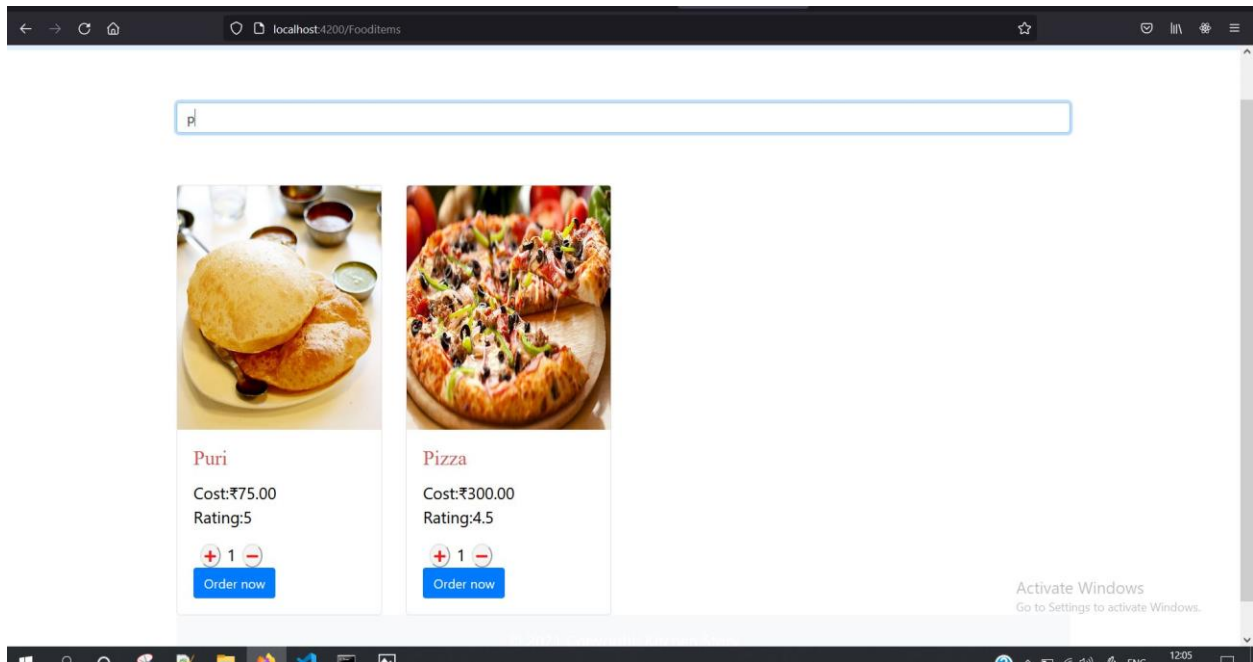
© 2021 KitchenStory All Rights Reserved

Activate Windows
Go to Settings to activate Windows

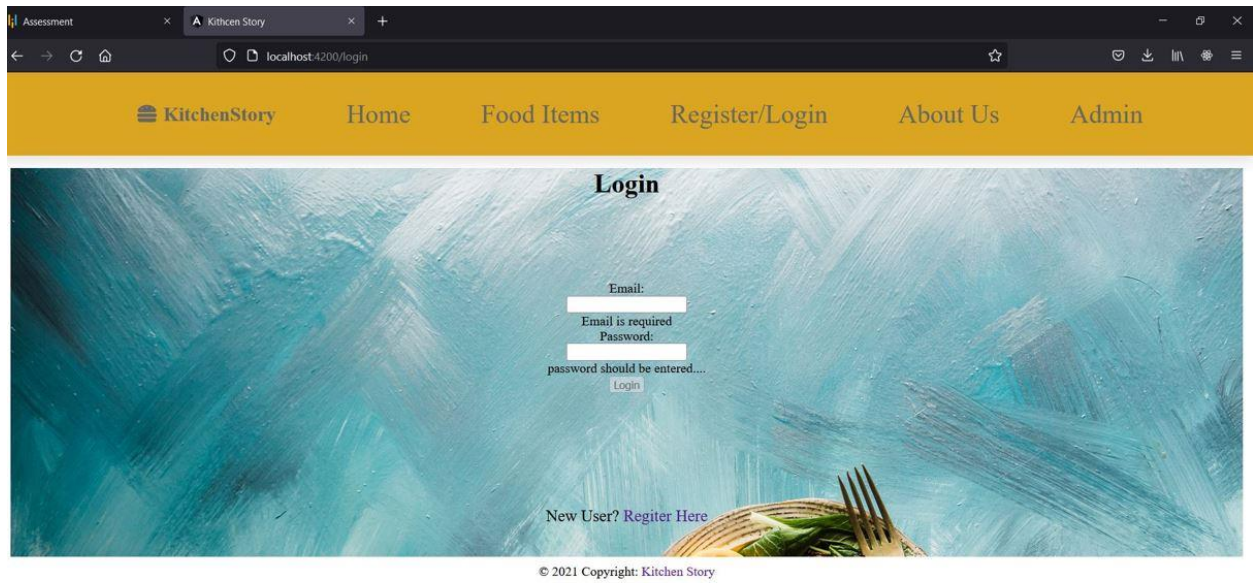
p p



p p



p p



p p

Assessment x Kitchen Story x +

localhost:4200/register

KitchenStory Home Food Items Register/Login About Us Admin

Register to Kitchen Story

FirstName:

The firstName should be entered

LastName:

The LastName should be entered

Email:

Valid Email required

SetPassword:

password should be entered

Existing User? [Login](#)

© 2021 Copyright: Kitchen Story

Activate Windows
Go to Settings to activate Windows.

p p

