**Backend Source code**

package com.ecommerce.model;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

@Entity

public class Product {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private int pid;

private String pbrand;

private String pname;

private String prate;

public Product() {

super();

}

public Product(int pid, String pbrand, String pname, String prate) {

super();

this.pid = pid;

this.pbrand = pbrand;

this.pname = pname;

this.prate = prate;

}

public int getPid() {

return pid;

}

public void setPid(int pid) {

this.pid = pid;

}

public String getPbrand() {

return pbrand;

}

public void setPbrand(String pbrand) {

this.pbrand = pbrand;

}

public String getPname() {

return pname;

}

public void setPname(String pname) {

this.pname = pname;

}

public String getPrate() {

return prate;

}

public void setPrate(String prate) {

this.prate = prate;

}

}

**package** com.ecommerce.repository;

**import** org.springframework.data.jpa.repository.JpaRepository;

**import** org.springframework.stereotype.Repository;

**import** com.ecommerce.model.Product;

@Repository

**public** **interface** ProductRepository **extends** JpaRepository<Product, Integer>{

}

package com.ecommerce.controller;

import java.util.HashMap;

import java.util.List;

import java.util.Map;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.http.ResponseEntity;

import org.springframework.web.bind.annotation.CrossOrigin;

import org.springframework.web.bind.annotation.DeleteMapping;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.PathVariable;

import org.springframework.web.bind.annotation.PostMapping;

import org.springframework.web.bind.annotation.PutMapping;

import org.springframework.web.bind.annotation.RequestBody;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RestController;

import com.ecommerce.exception.ResourceNotFoundException;

import com.ecommerce.model.Product;

import com.ecommerce.repository.ProductRepository;

@CrossOrigin(origins = "http://localhost:4200")

@RestController

@RequestMapping("/api/v1/")

public class ProductController {

@Autowired

private ProductRepository productRepository;

// get all products

@GetMapping("/Products")

public List<Product> getAllProducts(){

return productRepository.findAll();

}

// create product rest api

@PostMapping("/Products")

public Product createProduct(@RequestBody Product product) {

return productRepository.save(product);

}

// get product by id rest api

@GetMapping("/Products/{pid}")

public ResponseEntity<Product> getProductById(@PathVariable Integer pid) {

Product product = productRepository.findById(pid)

.orElseThrow(() -> new ResourceNotFoundException("Product not exist with id :" + pid));

return ResponseEntity.ok(product);

}

// update product rest api

@PutMapping("/Products/{pid}")

public ResponseEntity<Product> updateProduct(@PathVariable Integer pid, @RequestBody Product productDetails){

Product product = productRepository.findById(pid)

.orElseThrow(() -> new ResourceNotFoundException("Product not exist with id :" + pid));

product.setPbrand(productDetails.getPbrand());

product.setPname(productDetails.getPname());

product.setPrate(productDetails.getPrate());

Product updatedProduct = productRepository.save(product);

return ResponseEntity.ok(updatedProduct);

}

// delete product rest api

@DeleteMapping("/Products/{pid}")

public ResponseEntity<Map<String, Boolean>> deleteProduct(@PathVariable Integer pid){

Product product = productRepository.findById(pid)

.orElseThrow(() -> new ResourceNotFoundException("Product not exist with id :" + pid));

productRepository.delete(product);

Map<String, Boolean> response = new HashMap<>();

response.put("deleted", Boolean.TRUE);

return ResponseEntity.ok(response);

}

}

spring.datasource.url=jdbc:mysql://localhost:3306/phase4?useSSL=false

spring.datasource.username=root

spring.datasource.password=1234

# spring boot provided driver class name by default. If mentioned, once we run, it console mentions the same.

# Hibernate Properties

spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQL5InnoDBDialect

spring.jpa.hibernate.ddl-auto=update

package com.ecommerce.exception;

import org.springframework.http.HttpStatus;

import org.springframework.web.bind.annotation.ResponseStatus;

@ResponseStatus(value = HttpStatus.NOT\_FOUND)

public class ResourceNotFoundException extends RuntimeException{

private static final long serialVersionUID = 1L;

public ResourceNotFoundException(String message) {

super(message);

}

}

**Front End Source code**

<nav class="navbar navbar-expand-sm  navbar-dark" style="background-color: #213045;">

    <a class="navbar-brand" routerLink="" style="padding-left: 15px;font-size: 25px;font-weight: 500;">Kts</a>

    <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#collapsibleNavbar">

      <span class="navbar-toggler-icon"></span>

    </button>

    <ul class = "navbar-nav">

        <li class = "nav-item">

            <a routerLink="list" routerLinkActive="active" class="nav-link" >Product List</a>

        </li>

        <li class = "nav-item">

            <a routerLink="addProduct" routerLinkActive="active" class="nav-link" >Add Product</a>

        </li>

        <li class = "nav-item">

            <a routerLink="admin" routerLinkActive="active" class="nav-link right">Admin</a>

        </li>

    </ul>

</nav>

<br>

<h1 class = "text-center">{{title}}</h1>

<br>

<div class = "container">

    <router-outlet></router-outlet>

</div>

import { NgModule } from '@angular/core';

import { RouterModule, Routes } from '@angular/router';

import { BuyProductComponent } from './buy-product/buy-product.component';

import { CreateProductComponent } from './create-product/create-product.component';

import { HomeComponent } from './home/home.component';

import { PaymentComponent } from './payment/payment.component';

import { ProductListComponent } from './product-list/product-list.component';

import { UpdateProductComponent } from './update-product/update-product.component';

const routes: Routes = [

  {

    path: 'addProduct', component:CreateProductComponent

  },

  {

    path: 'list',component:ProductListComponent

  },

  {

    path:'updateProduct/:pid', component:UpdateProductComponent

  },

  {

    path: '',component:HomeComponent

  },

  {

    path: 'buyProduct/:pid',component:BuyProductComponent

  },

  {

    path: 'payment/:pid',component:PaymentComponent

  }

];

@NgModule({

  imports: [RouterModule.forRoot(routes)],

  exports: [RouterModule]

})

export class AppRoutingModule { }

<div class="container-fluid">

    <img src = "../../assets/5.jpg" width="100%" height="50%">

  </div>

  <br>

  <br>

<h2 class = "text-center">Browse Items</h2>

<hr>

<table class = "table table-striped" >

    <thead>

        <tr>

            <th> </th>

            <th> Product Name</th>

            <th> Product Description</th>

            <th> Product Price</th>

            <th> Actions </th>

        </tr>

    </thead>

    <tbody class = "text-center" style="vertical-align: middle;">

        <tr \*ngFor = "let product of products">

            <td><img [src]="myimg" width = "40%" height = "40%" alt=""></td>

            <td>{{product.pname}}</td>

            <td>{{product.pbrand}}</td>

            <td>{{product.prate}}</td>

            <td>

                <button  (click) = "buyProduct(product.pid)" class = "btn btn-info" style="background-color: #213045; color: aliceblue; border: #4a3c21;">Buy</button>

            </td>

        </tr>

    </tbody>

</table>

<h2 class = "text-center">Checkout</h2>

<hr>

<section>

    <div class="container py-5">

      <div class="row justify-content-center">

        <div class="col-md-8 col-lg-6 col-xl-4">

          <div class="card" style="border-radius: 15px;">

            <div class="bg-image hover-overlay ripple ripple-surface ripple-surface-light"

              data-mdb-ripple-color="light">

              <img src="../../assets/1.jpg"

                style="border-top-left-radius: 15px; border-top-right-radius: 15px;" class="img-fluid"

                alt="Laptop" />

              <a href="#!">

                <div class="mask"></div>

              </a>

            </div>

            <div class="card-body pb-0">

              <div class="d-flex justify-content-between">

                <div>

                  <p><a href="#!" class="text-dark">{{product.pname}}</a></p>

                  <p class="small text-muted">Item Name</p>

                </div>

                <div>

                  <div class="d-flex flex-row justify-content-end mt-1 mb-4 text-danger">

                    <i class="fas fa-star"></i>

                    <i class="fas fa-star"></i>

                    <i class="fas fa-star"></i>

                    <i class="fas fa-star"></i>

                  </div>

                  <p class="small text-muted">Rated 4.0/5</p>

                </div>

              </div>

            </div>

            <hr class="my-0" />

            <div class="card-body pb-0">

              <div class="d-flex justify-content-between">

                <p><a href="#!" class="text-dark">{{product.prate}}</a></p>

                <p class="text-dark">#### 8787</p>

              </div>

              <p class="small text-muted">VISA Platinum</p>

            </div>

            <hr class="my-0" />

            <div class="card-body">

              <div class="d-flex justify-content-between align-items-center pb-2 mb-1">

                <a href="#!" class="text-dark fw-bold">Cancel</a>

                <button type="button" (click) = "payment(product.pid)" class="btn btn-primary">Buy now</button>

              </div>

            </div>

          </div>

        </div>

      </div>

    </div>

  </section>

import { Component, OnInit } from '@angular/core';

import { Router } from '@angular/router';

import {Product} from '../product'

import { ProductService } from '../product.service';

@Component({

  selector: 'app-home',

  templateUrl: './home.component.html',

  styleUrls: ['./home.component.css']

})

export class HomeComponent implements OnInit{

  myimg = "../../assets/food.jpg";

  products!: Product[];

  constructor(private productService: ProductService,

              private router: Router) {}

  ngOnInit(): void {

    this.getProducts();

  }

  private getProducts(){

    this.productService.getProductsList().subscribe(data => {

      this.products = data;

    });

  }

  buyProduct(pid: number){

    this.router.navigate(['buyProduct',pid]);

  }

  }

export class Product {

    pid!: number;

    pbrand!: string;

    pname!: string;

    prate!: string;

}

import { HttpClient } from '@angular/common/http';

import { Injectable } from '@angular/core';

import { Observable } from 'rxjs';

import { Product } from './product';

@Injectable({

  providedIn: 'root'

})

export class ProductService {

  private baseURL = "http://localhost:8080/api/v1/Products";

  constructor(private httpClient: HttpClient) { }

  getProductsList(): Observable<Product[]>{

      return this.httpClient.get<Product[]>(`${this.baseURL}`);

  }

  createProduct(product: Product): Observable<Object>{

      return this.httpClient.post(`${this.baseURL}`,product);

  }

  getProductById(pid: number): Observable<Product>{

    return this.httpClient.get<Product>(`${this.baseURL}/${pid}`);

  }

  updateProduct(pid:number, product:Product): Observable<Object>{

    return this.httpClient.put(`${this.baseURL}/${pid}`,product);

  }

  deleteProduct(pid: number): Observable<Object>{

    return this.httpClient.delete(`${this.baseURL}/${pid}`);

  }

  payment(pid: number): Observable<Product>{

    return this.httpClient.get<Product>(`${this.baseURL}/${pid}`);

  }

}

<h2>Product List</h2>

<table class = "table table-striped">

    <thead>

        <tr>

            <th> Product Name</th>

            <th> Product Description</th>

            <th> Product Price</th>

            <th> Actions </th>

        </tr>

    </thead>

    <tbody>

        <tr \*ngFor = "let product of products">

            <td>{{product.pname}}</td>

            <td>{{product.pbrand}}</td>

            <td>{{product.prate}}</td>

            <td>

                <button (click) = "updateProduct(product.pid)" class = "btn btn-info">Update</button>

                <button (click) = "deleteProduct(product.pid)" class = "btn btn-danger" style="margin-left: 10px"> Delete</button>

            </td>

        </tr>

    </tbody>

</table>

import { Component, OnInit } from '@angular/core';

import { Router } from '@angular/router';

import {Product} from '../product'

import { ProductService } from '../product.service';

@Component({

  selector: 'app-product-list',

  templateUrl: './product-list.component.html',

  styleUrls: ['./product-list.component.css']

})

export class ProductListComponent implements OnInit{

products!: Product[];

constructor(private productService: ProductService,

            private router: Router) {}

ngOnInit(): void {

  this.getProducts();

}

private getProducts(){

  this.productService.getProductsList().subscribe(data => {

    this.products = data;

  });

}

updateProduct(pid: number){

  this.router.navigate(['updateProduct',pid]);

}

deleteProduct(pid: number){

  this.productService.deleteProduct(pid).subscribe( data => {

    console.log(data);

    this.getProducts();

  })

}

}

<div class="col-md-6 offset-md-3">

    <h3> Add Item </h3>

    <form (ngSubmit) = "onSubmit()">

        <div class="form-group">

            <label> Product Name</label>

            <input type="text" class ="form-control" id = "pname"

                [(ngModel)] = "product.pname" name = "pname">

        </div>

        <div class="form-group">

            <label> Product Brand</label>

            <input type="text" class ="form-control" id = "pbrand"

                [(ngModel)] = "product.pbrand" name = "pbrand">

        </div>

        <div class="form-group">

            <label> Product Rate</label>

            <input type="text" class ="form-control" id = "prate"

                [(ngModel)] = "product.prate" name = "prate">

        </div>

        <button class = "btn btn-success" type ="submit">Submit</button>

    </form>

    </div>

import { Component, OnInit } from '@angular/core';

import { Router } from '@angular/router';

import { Product } from '../product';

import { ProductService } from '../product.service';

@Component({

  selector: 'app-create-product',

  templateUrl: './create-product.component.html',

  styleUrls: ['./create-product.component.css']

})

export class CreateProductComponent implements OnInit {

  product: Product = new Product();

  constructor(private productService: ProductService,

              private router: Router) {}

  ngOnInit(): void {

  }

  saveProduct(){

    this.productService.createProduct(this.product).subscribe(data =>{

      console.log(data);

      this.goToProductList();

    },

    error => console.log(error));

  }

  goToProductList(){

    this.router.navigate(['/']);

  }

  onSubmit(){

    console.log(this.product);

    this.saveProduct();

  }

}

<section>

    <div class="container py-5">

      <div class="row d-flex justify-content-center">

        <div class="col-md-12 col-lg-10 col-xl-8">

          <div class="card">

            <div class="card-body p-md-5">

              <div>

                <h4>Payment Page</h4>

                <p class="text-muted pb-2">

                  Please make the payment to start enjoying all the features of our premium

                  plan as soon as possible

                </p>

              </div>

              <div class="px-3 py-4 border border-primary border-2 rounded mt-4 d-flex justify-content-between">

                <div class="d-flex flex-row align-items-center">

                  <img src="https://i.imgur.com/S17BrTx.webp" class="rounded" width="60" />

                  <div class="d-flex flex-column ms-4">

                    <span class="h5 mb-1">Price</span>

                    <span class="small text-muted">CHANGE ORDER</span>

                  </div>

                </div>

                <div class="d-flex flex-row align-items-center">

                  <sup class="dollar font-weight-bold text-muted">₹ </sup>

                  <span class="h2 mx-1 mb-0">399</span>

                  <span class="text-muted font-weight-bold mt-2">/-</span>

                </div>

              </div>

              <h4 class="mt-5">Payment details</h4>

              <div class="mt-4 d-flex justify-content-between align-items-center">

                <div class="d-flex flex-row align-items-center">

                  <img src="https://i.imgur.com/qHX7vY1.webp" class="rounded" width="70" />

                  <div class="d-flex flex-column ms-3">

                    <span class="h5 mb-1">Credit Card</span>

                    <span class="small text-muted">1234 XXXX XXXX 2570</span>

                  </div>

                </div>

                <div>

                  <input type="text" class="form-control" placeholder="CVC" style="width: 70px;" />

                </div>

              </div>

              <div class="mt-2 d-flex justify-content-between align-items-center">

                <div class="d-flex flex-row align-items-center">

                  <img src="https://i.imgur.com/qHX7vY1.webp" class="rounded" width="70" />

                  <div class="d-flex flex-column ms-3">

                    <span class="h5 mb-1">Credit Card</span>

                    <span class="small text-muted">2344 XXXX XXXX 8880</span>

                  </div>

                </div>

                <div>

                  <input type="text" class="form-control" placeholder="CVC" style="width: 70px;" />

                </div>

              </div>

              <h6 class="mt-4 mb-3 text-primary text-uppercase">ADD PAYMENT METHOD</h6>

              <div class="form-outline">

                <label class="form-label" for="formControlLg">Full Name</label>

                <input type="text" id="formControlLg" class="form-control form-control-lg" />

              </div>

              <div class="form-outline">

                <label class="form-label" for="formControlLg">Email address</label>

                <input type="text" id="formControlLg" class="form-control form-control-lg" />

              </div>

              <div class="form-outline">

                <label class="form-label" for="formControlLg">Mobile Number</label>

                <input type="text" id="formControlLg" class="form-control form-control-lg" />

              </div>

              <div class="form-outline">

                <label class="form-label" for="formControlLg">Address</label>

                <input type="text" id="formControlLg" class="form-control form-control-lg" />

              </div>

              <div class="mt-3">

                <button class="btn btn-primary btn-block btn-lg">

                  Proceed to payment <i class="fas fa-long-arrow-alt-right"></i>

                </button>

              </div>

            </div>

          </div>

        </div>

      </div>

    </div>

  </section>

import { Component, OnInit } from '@angular/core';

import { ActivatedRoute } from '@angular/router';

import { Product } from '../product';

import { ProductService } from '../product.service';

@Component({

  selector: 'app-payment',

  templateUrl: './payment.component.html',

  styleUrls: ['./payment.component.css']

})

export class PaymentComponent implements OnInit {

  pid!: number

  product!: Product

  constructor(private route: ActivatedRoute, private productService: ProductService) { }

  ngOnInit(): void {

    this.pid = this.route.snapshot.params['pid'];

    this.product = new Product();

    this.productService.payment(this.pid).subscribe( data => {

      this.product = data;

    });

  }

}