**DAY – 9**

**Controller.java:**

package com.example.demo.Controller;

import java.util.List;

import java.util.Map;

import java.util.Optional;

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

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.example.demo.Model.ProductModel;

import com.example.demo.Model.UserModel;

import com.example.demo.Service.service;

import io.swagger.v3.oas.annotations.tags.Tag;

@RestController

@RequestMapping("/api")

@CrossOrigin("http://localhost:3000")

public class Controller {

@Autowired

private service service;

// Auth

@Tag(name = "Signin", description = "Login Endpoint")

@PostMapping("/Signin")

private String Login(@RequestBody Map<String, String> xLogin) {

String username = xLogin.get("username");

String password = xLogin.get("password");

String result = service.Login(username, password);

return result;

}

@Tag(name = "Signup", description = "Signup Endpoint")

@PostMapping("/Signup")

public String Signup(@RequestBody UserModel user) {

return service.Signup(user);

}

// Products

@Tag(name = "List Products", description = "List All Products")

@GetMapping("/list")

private List<ProductModel> Games(){

return service.getData();

}

@Tag(name = "Sort Product by ID", description = "View indudual Product Data")

@GetMapping("/data/{id}")

private Optional<ProductModel> viewGame(@PathVariable Long id) {

return service.findbyID(id);

}

@Tag(name = "Add Product", description = "Add New Product")

@PostMapping("/add")

private ProductModel addProduct(@RequestBody ProductModel data) {

return service.addData(data);

}

@Tag(name = "Edit Product", description = "Edit Existing Product")

@PutMapping("/edit/{id}")

private ProductModel editProduct(@PathVariable Long id, @RequestBody ProductModel data) {

return service.editData(data, id);

}

@Tag(name = "Delete Data", description = "Delete The Existing Product")

@DeleteMapping("/delete/{id}")

private String deleteProduct(@PathVariable Long id) {

return service.deleteData(id);

}

}

**Service.java:**

package com.example.demo.Service;

import java.util.List;

import java.util.Optional;

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

import org.springframework.stereotype.Service;

import com.example.demo.Model.ProductModel;

import com.example.demo.Model.UserModel;

import com.example.demo.Repo.Productrepo;

import com.example.demo.Repo.Userrepo;

@Service

public class service {

@Autowired

private Productrepo productrepo;

@Autowired

private Userrepo userrepo;

//Login Logic

public String Login(String username, String password) {

UserModel xuser = userrepo.findByusername(username);

if (xuser == null) {

return "invalidusername";

} else {

if (xuser.getPassword().equals(password)) {

return "success";

} else {

return "invalidpassword";

}

}

}

public String Signup(UserModel xuser) {

String username = xuser.getUsername();

UserModel useruser = userrepo.findByusername(username);

if (useruser == null) {

userrepo.save(xuser);

return "useradded";

} else {

return "existingusername";

}

}

// Get

public List<ProductModel> getData() {

return productrepo.findAll();

}

// Post

public ProductModel addData(ProductModel data) {

return productrepo.save(data);

}

// Edit (modify if data exists and save, else don't save)

public ProductModel editData(ProductModel data, Long id) {

ProductModel edx = productrepo.findById(id).orElse(data);

if (edx != null) {

edx.setProductName(data.getProductName());

edx.setProductStock(data.getProductStock());

edx.setQuantityUsed(data.getQuantityUsed());

edx.setProductPrice(data.getProductPrice());

edx.setAvailability(data.getAvailability());

edx.setProductimg(data.getProductImg());

return productrepo.saveAndFlush(edx);

} else {

return null;

}

}

//Delete

public String deleteData(Long id) {

productrepo.deleteById(id);

return "Deleted Successfully";

}

//Find by ID

public Optional<ProductModel> findbyID(Long id) {

return productrepo.findById(id);

}

}

**StockApplicationTests.java:**

package com.example.demo;

import org.junit.jupiter.api.Test;

import org.springframework.boot.test.context.SpringBootTest;

@SpringBootTest

class SpringBackendApplicationTests {

@Test

void contextLoads() {

}

}