TheCouchPotato.com: An E-commerce Platform for Comfort Items

Christopher Markel

CST-452 Milestone 5: Testing Phase (Testing)

Grand Canyon University

Instructor: Professor Donna Jackson

Revision: 1

Date: 9/24/23

**Implementation Plan**

The implementation plan is submitted as an external document. It can also be found [**here**](https://docs.google.com/spreadsheets/d/167k-tLOPiDIA_9y5-VwnUTuXOLP9x_N8YAA1f42ybDU/edit?usp=sharingEYjQU_wtrKOFzvK480N1Zp_e4T5unBuJdBZWDqTz9E/edit?usp=sharing).

Github Repository: <https://github.com/cylon-raider/TheCouchPotato/>

**Screencast Video URL**: <https://www.loom.com/share/d6bbdad144254f018f08e60b58f25492?sid=eafe67d4-e615-4a2a-be68-5e6afa3c0a1f>

**Source Code Listing**

*Main Application:*

Cst452Application.java

Description: Main Spring Boot application class. Contains the main method which is the entry point for the Spring Boot application.

*Test Layer:*

Cst452ApplicationTests.java

Description: General test class for the application.

*Configuration:*

SecurityConfig.java

Description: Configures Spring Security. Defines security rules for the application, such as authentication and authorization mechanisms.

SwaggerConfig.java

Description: Configures Swagger for documenting and testing RESTful APIs. Provides settings for the Swagger UI and API documentation.

*Root Files:*

.gitignore

Description: This file specifies intentionally untracked files that Git should ignore.

README.md

Description: The markdown file that provides an overview and documentation for the project.

mvnw & mvnw.cmd

Description: Maven wrapper scripts for Unix and Windows respectively.

pom.xml

Description: The Project Object Model (POM) file for Maven. It contains project and configuration details used by Maven to build the project.

*Business Services:*

CartBusinessService.java

Description: Contains business logic related to the cart. Interacts with the data layer for cart operations.

CategoryBusinessService.java

Description: Contains business logic related to product categories. Interacts with the data layer for category operations.

ProductBusinessService.java

Description: Contains business logic related to products. Interacts with the data layer for product operations.

ProductRestService.java

Description: Provides RESTful services related to products. Exposes endpoints for CRUD operations on products.

UserBusinessService.java

Description: Contains business logic related to users. Interacts with the data layer for user operations.

*Controllers:*

CartController.java

Description: Handles web requests related to the cart. Defines endpoints for cart operations.

HomeController.java

Description: Handles web requests related to the home page.

LoginController.java

Description: Handles web requests related to user login.

ProductController.java

Description: Handles web requests related to products. Defines endpoints for product operations.

RegistrationController.java

Description: Handles web requests related to user registration.

*Data Services:*

CartDataService.java

Description: Interacts with the database for CRUD operations related to the cart.

CategoryDataService.java

Description: Interacts with the database for CRUD operations related to product categories.

DataAccessInterface.java

Description: Interface defining common CRUD operations.

ProductDataService.java

Description: Interacts with the database for CRUD operations related to products.

UserDataService.java

Description: Interacts with the database for CRUD operations related to users.

*Models:*

CartItem.java

Description: Represents an item in the cart.

CartModel.java

Description: Represents the cart with a list of cart items.

CategoryModel.java

Description: Represents a product category.

LoginModel.java

Description: Represents user login data.

ProductList.java

Description: Represents a list of products.

ProductModel.java

Description: Represents a product with details like ID, name, and price.

RoleModel.java

Description: Represents a user role.

UserModel.java

Description: Represents a user with details like ID, username, and role.

*Layouts:*

common.html

Description: Common layout elements for the templates.

defaultTemplate.html

Description: Default layout template.

loginRegistrationTemplate.html

Description: Layout template for login and registration.

*Templates:*

addProduct.html

Description: Template for adding a product.

cart.html

Description: Template for the cart page.

deleteProduct.html

Description: Template for deleting a product.

index.html

Description: Template for the main index page.

login.html

Description: Template for the login page.

products.html

Description: Template for displaying products.

regSuccess.html

Description: Template for successful registration.

registration.html

Description: Template for user registration.

search.html

Description: Template for product search.

updateProduct.html

Description: Template for updating a product.

*Static Resources:*

couchpotato.png

Description: Image asset for the project.

under-construction\_geek\_man\_01.png

Description: Image asset indicating under construction.

*Resources:*

Application.properties

Description: Configuration properties for the Spring application.