University of British Columbia

**COSC** **304**: **Introduction to Databases**

Group 13

**S-MART**

**Project Design Document**

Om Mistry

Syed Aamir Ahmed

GitHub Repo Link: <https://github.com/OM200401/COSC304_project>

**Project Overview**

The project aims to develop a robust and user-friendly grocery shopping platform named "S-MART." This platform offers a diverse range of products across various categories, facilitating easy product search, purchase, and management for users. The platform provides a secure and seamless shopping experience while emphasizing user convenience and satisfaction.

**Executive Summary**

Our objective for this project was to create an intuitive and visually appealing interface for users to browse and purchase products effectively. We also made sure to implement a secure authentication system to ensure user privacy and data protection.

We used the backend MySQL server to save all our data and then use the data to retrieve information from the database to display it to the users. Our backend efficiently handles all the products in the system with their information, orders placed by the customers, the customer information, reviews by the customer, their cart, and their shipments.

The website enables users to add products to cart thus facilitating them with a smooth checkout process. It also allows customers to leave reviews on products which is a good way to encourage user interaction and feedback. Maintain scalability and flexibility to accommodate future enhancements and increase in traffic.

The website boasts a multitude of features including user-friendly product search and filtering products based on categories. User account creation and authentication for secure login and logout functionality. Interactive product pages with details for each product. Cart management allowing users to change the items in their cart accordingly.

**S-MART** aims to provide an exceptional online shopping experience by focussing on user convenience, security, and quality. With its diverse features and user-centric design, the platform aims to become a go-to destination for users seeking a seamless and enjoyable shopping journey.

**Development Project Features List Template**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Final Overall System Features** | **Category** | **Possible Points** | **Points** | **Comments / Code Files** |
| **Documentation (15 total points)** | | | | |
| Executive Summary (up to 1 page) | Core | 2 | 2 |  |
| System Features (2 pages) | Core | 2 | 2 |  |
| List of Web and AI sources used | Core | 3 | 3 |  |
| Walkthrough with screenshots (up to 15 pages) | Core | 8 | 8 |  |
|  | | | | |
| **Hosting (10 total points)** | | | | |
| Running on cloud server | Advanced | 10 |  |  |
|  | | | | |
| **Main/Home Page (up to 14 total points)** | | | | |
| Search for a product by name | Core | 1 | 1 | listprod.js |
| Browse products by category | Core | 1 | 1 | listprod.js |
| List products (by search/browse) | Core | 1 | 1 | listprod.js |
| List products with image | Core | 1 | 1 | listprod.js |
| Page header with menu | Basic | 1 | 1 | index.js (header included on every page) |
| Page header shows current logged in user | Basic | 1 | 1 | index.js |
| Dynamic products on page based on sales | Basic | 2 |  |  |
| Improved UI (user-friendly, images, etc.) | Basic | 2 | 2 | listprod.js |
| Product recommendation based on user | Advanced | 4 |  |  |
|  | | | | |
| **Shopping Cart (up to 8 total points)** | | | | |
| Add to shopping cart | Core | 1 | 1 | addcart.js |
| View shopping cart | Core | 1 | 1 | showcart.js |
| Update quantity (with data validation) | Basic | 1 | 1 | showcart.js |
| Remove item from shopping cart | Basic | 1 | 1 | showcart.js |
| Improved formatting/UI (e.g. in header bar) | Basic | 1 | 1 | showcart.js |
| Cart stored in database between sessions | Basic | 3 | 3 | showcart.js |
|  | | | | |
| **Checkout (up to 9 total points)** | | | | |
| Checkout with customer id | Core | 1 | 1 | checkout.js |
| Checkout with payment/shipment info | Basic | 2 | 2 | checkout.js |
| Checkout with data validation | Basic | 2 | 2 | checkout.js |
| Multiple shipments per order supported | Advanced | 2 |  | Note: Requires DB change. |
| Calculate taxes/shipping (by items/state) | Advanced | 2 |  | Note: Requires DB change. |
|  | | | | |
| **Product Detail Page (up to 2 total points)** | | | | |
| Product detail page and item description | Core | 1 | 1 | product.js |
| Product detail has an image from database | Core | 1 | 1 | product.js |
|  | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **User Accounts/Login (up to 12 total points)** | | | | |
| Create user account page | Basic | 2 | 2 | signup.js |
| Create account with data validation | Basic | 2 | 2 | signup.js |
| Edit user account info (address, password) | Basic | 2 | 2 | updateCustomer.js |
| Login/logout | Core | 1 | 1 | login.js/logout.js |
| Register user/Forgot password email | Advanced | 4 |  |  |
| Page listing all orders for user | Core | 1 | 1 | admin.js |
|  | | | | |
| **Product Reviews (up to 5 total points)** | | | | |
| Ability to enter a review on a product | Basic | 2 | 2 | addreview.js |
| Display product review on product detail page | Basic | 1 | 1 | addreview.js/product.js |
| Restrict to one review per user on item purchased | Advanced | 2 |  |  |
|  | | | | |
| **Warehouses/Inventory (up to 5 total points)** | | | | |
| Display item inventory by store/warehouse | Basic | 2 | 2 | product.js |
| Edit item inventory by store/warehouse | Advanced | 3 |  |  |
|  | | | | |
| **Administrator Portal (up to 19 total points)** |  |  |  |  |
| Secured by login | Core | 1 | 1 | login.js |
| List all customers | Core | 1 | 1 | admin.js |
| List report showing total sales/orders | Core | 1 | 1 | admin.js |
| Report with a graph | Advanced | 3 |  |  |
| Add new product | Basic | 2 | 2 | addproduct.js |
| Update/delete product | Basic | 2 | 2 | updateproduct.js |
| Change order status/ship order | Basic | 1 |  |  |
| Upload a photo to file system for product | Advanced | 2 |  |  |
| OR: Upload a photo to database for product | Advanced | 4 |  |  |
| Database restore with SQL script | Basic | 2 |  |  |
| Add/update warehouse, customer | Basic | 2 |  |  |
|  |  |  |  |  |
| **Database System/General** |  |  |  |  |
| Implement some validation using triggers | Advanced | 2 |  |  |
| Use AJAX for some pages | Advanced | 4 |  |  |
| User interface and navigation/usability |  | Up to 5 | 5 | Interface is appealing for users ; styling has been done along with a Navbar on every page for easy access |
| *Others (your suggestions)* |  | Up to 5 |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
| **Total: (out of 50 with maximum of 10 bonus marks. i.e. 60 points out of 50 is max.)** |  |  |  |  |

**Walkthrough**

**A screenshot of a computer

Description automatically generated**

**S-MART** is your daily shopping grocery store but online which means you can shop for your favourite things within a few clicks while sitting on your couch and enjoying a movie! The above picture displays the landing page for the website.

A screenshot of a website

Description automatically generated

The above picture displays the shopping page of the website that allows you to browse through the variety of products we sell!

A green square with white text

Description automatically generated

We can login to our account by clicking on the login button in the navigation bar. It also has an option to sign up that allows the user to create their own account.

A screenshot of a computer

Description automatically generated

Once we are signed in, we can click on the customers option and look at our account details which includes our information and the orders that we have placed in the past.

A screenshot of a computer

Description automatically generated

The Edit Profile button allows you to edit your own profile. It allows you to change your address and log in details such as password or username.

A screenshot of a computer

Description automatically generated

We can also access the admin dashboard that shows the sales report by day and all the customers in the database on here.

A screenshot of a shopping cart

Description automatically generated

This is how our cart looks like after adding certain products and we have an option to remove the item or update the quantity if we want to.

A screenshot of a computer

Description automatically generated

Then we can checkout by entering our customerID and password.

A white background with black text

Description automatically generated

The above page shows our order summary page with the items we purchased and the name and order reference number.

A screenshot of a grocery order list

Description automatically generated

We can then go to our main page and check all the orders that were placed by clicking on the ORDERS button in the navbar.