**THE KITCHEN STORY**

Updated- Final\_PROJECT REPORT

**Chandni Grocery Store**

(Online grocery store)

## Table of contents

[1. PROJECT DESCRIPTION](#_Toc10848)

[2. PROJECT FEATURES](#_Toc10849)

[3. DATABASE DESIGN](#_Toc10850)

[4. TECHNOLOGIES USED](#_Toc10851)

[5. SCREENSHOTS](#_Toc10852)

# PROJECT DESCRIPTION

This project has been divided in to two modules namely admin module and customer/user module.

* **Administrator**

When the web user signs in as an administrator, he/she acquires the access rights to add new brands, add new products, delete the existing product and can also update the existing product details. The administrator also can also view/edit the details regarding the quantity of grocery products in stock. The groceries added by the administrator will be visible both to the administrator and the regular website customers/visitors. Administrator can also filter the products based on categories and brands which enables admin to easily choose the product which he/she wants to update or delete.

* **User**

All users can view the homepage of the website, for a user to purchase any product he/she needs to be registered and have an online account. If the user had already registered on the website and have an account, he/she needs to provide login credentials and login before making any purchases or adding a product to cart. All the products available would be listed on the homepage. All these products can be filtered using brand names and categories. Also, a search engine is provided which facilitates the user to search for a specific product as per his requirements. User can add product to cart, update the products in cart and can also delete the products from the cart. After making the necessary changes in cart, user has to checkout to place an order. When an order is placed, details of the products purchased will be stored in the database. Once an order has been placed, cart gets cleared and user can place next order. All the orders placed by a customer can be viewed in order history.

# PROJECT FEATURES

This project facilitates many functionalities which user admin can perform. Detailed description of the functionalities is given below:

***a. User functionalities:***

* **Index page**: This is the first page we come up when website is opened (i.e Home page). It displays all the products available with product image, title, price details with an add to cart option below. On this page, users can not add an item to cart or make any purchases until they are logged in. For that purpose, user needs to sign in using his/her login credentials. This page will also have sign in and signup links.
* **Signup**: User needs to have an online account (registered user) to place the orders or add a product to cart. If the user doesn’t have an account, he needs to create an account online before making any purchases. For the registration purpose, a form is provided in which the user needs to enter username, first name, last name, email address and password. The data provided in each of the above-mentioned fields is validated using jQuery. User availability and email availability will be checked when user clicks on submit button on registration form. If the username is already present error message will be displayed indicating username already exists. Then user must provide a different username. Once all the validation of all the fields succeeds user will be registered successfully with the store.
* **Login**: User needs to enter username and password. If username and password provided are valid then user will be successfully logged into website. Otherwise, error message will be displayed.
* **Home Page**: This page displays all the available products along with its name, price, image and an add to cart button for each product. As this page can be viewed only after logging in, user can add products to cart, checkout and place orders.
* **Filters**: We can filter the grocery products based on their category (fruits, vegetables, etc.) or brands (Great Value, Dannon, Vaseline, Organic etc.)
* **Search**: Users can search through the products using search keywords. These keywords are provided along with the product details by admin when he adds the product to the store. Product search keywords include product name, brand and the product related keywords which can making the searching process much more efficient.
* **Cart**: Every product has an add to cart button below it. By clicking this button, the grocery product is added to the user’s cart. The user can find the product he added to cart by clicking on the cart link. We can also delete or update product quantity before checkout. After the user checks out, an order is placed and cart is cleared.
* **Order history:** A user can view the products he ordered/purchased before, along with order date in the order history.
* **Log out:** This option can be used log out the user from the portal.

***b.* Ad*ministrator functionalities:***

Admin sign's in using his/her login credentials. On successful login, he can view all the products available in the database. The functionalities being implemented by admin are adding a product, updating product details, deleting a product. Admin can also add new brands.

* **View products**: Admin can view all the products stored in the database.
* **Add products**: To add a new product, admin provides details like product title, product price, available product quantity, product image, product brand, product category and product search key words.
* **Update Product**: Administrator has the access to update all the product details. This functionality helps administrator to update the price of a product from time to time so store product price matches with the market price. Also, it enables to provided discounted price for special occasions. It also enables to update other details like product title, quantity, image, brand and category. Product quantity updating will be useful when the inventory is empty and more stock arrives. In these scenarios admin can update the inventory when new stock arrives.
* **Delete Product**: Administrator is given all the rights and privileges to delete any product from the database. The deleted product is not showed on the user interface anymore. This functionality will be useful when the production of product is stopped or banned due to certain reasons. In these scenarios admin can delete it so that will not be able to purchase it.
* **Logout**: When admin logs out, page will be redirected to index page where the home page/index page will be displayed.

**C.Other Functionalities:**

* **Inventory Management**: As soon as the user places an order for a product, we update the total quantity of the ordered product in the database.
* **Soft delete**: When a product ordered by the user is deleted by the admin in the later stages, user can still view the product in his order history. But a message that the ***item in not available*** is displayed beside the product details.
* **Pagination**: If the number of products displayed is greater than 10 then they will be displayed in multiple pages.

# DATABASE DESIGN

The below tables have been created to store the grocery store data.

**Relational schema:**

***USERS****( USERNAME, FIRST\_NAME, LAST\_NAME, EMAIL, PASSWORD)*

***BRANDS*** *(BRAND\_ID, BRAND\_TITLE, BRAND\_DESCRIPTION)*

***CATEGORIES*** *(CAT\_ID, CAT\_TITLE, CATEGORY\_DESCRIPTION)*

***PRODUCTS****( PRODUCT\_ID,PRODUCT\_CAT, PRODUCT\_PRICE, PRODUCT\_TITLE ,*

*PRODUCT\_IMG, PRODUCT\_BRAND, PRODUCT\_KEYWORDS, PRODUCT\_QUANTITY ,*

*PRODUCT\_FLAG )*

***CART****( ID, USERNAME, PRODUCT\_ID ,QUANTITY\_INCART ,TOTAL\_AMOUNT )*

***ORDERS****( ORDER\_ID, USERNAME, ORDER\_DATE , TOTAL\_AMOUNT )*

***ORDER\_ITEMS****(ORDER\_ID, PRODUCT\_ID , QUANTITY\_ORDERED ,PRICE,ORDER\_FLAG)*

**TABLES:**

**USERS TABLE:**

*CREATE TABLE USERS(*

*USERNAME VARCHAR(50) NOT NULL,*

*FIRST\_NAME VARCHAR(200),*

*LAST\_NAME VARCHAR(200),*

*EMAIL VARCHAR(200),*

*PASSWORD VARCHAR(200),*

*PRIMARY KEY(USERNAME));*

**BRANDS TABLE:**

*CREATE TABLE BRANDS(*

*BRAND\_ID INT NOT NULL,*

*BRAND\_TITLE VARCHAR(200) NOT NULL,*

*BRAND\_DESCRIPTION VARCHAR(200),*

*PRIMARY KEY(BRAND\_ID));*

**CATEGORIES TABLE:**

*CREATE TABLE CATEGORIES(*

*CAT\_ID INT NOT NULL,*

*CAT\_TITLE VARCHAR(200) NOT NULL,*

*CATEGORY\_DESCRIPTION VARCHAR(200),*

*PRIMARY KEY(CAT\_ID));*

**PRODUCT TABLE:**

*CREATE TABLE PRODUCTS(*

*PRODUCT\_ID INT NOT NULL,*

*PRODUCT\_CAT INT NOT NULL,*

*PRODUCT\_PRICE NUMBER(9,2),*

*PRODUCT\_TITLE VARCHAR(200),*

*PRODUCT\_IMG VARCHAR(200),*

*PRODUCT\_BRAND INT NOT NULL,*

*PRODUCT\_KEYWORDS VARCHAR(200),*

*PRODUCT\_QUANTITY VARCHAR(200),*

*PRODUCT\_FLAG VARCHAR(10),*

*PRIMARY KEY(PRODUCT\_ID),*

*CONSTRAINT BRAND\_FK FOREIGN KEY(PRODUCT\_BRAND) REFERENCES BRANDS(BRAND\_ID)*

*ON DELETE CASCADE,*

*CONSTRAINT CATEGORY\_FK FOREIGN KEY(PRODUCT\_CAT) REFERENCES CATEGORIES(CAT\_ID)*

*ON DELETE CASCADE);*

**CART TABLE:**

*CREATE TABLE CART(*

*ID INT NOT NULL,*

*USERNAME VARCHAR(200) NOT NULL,*

*PRODUCT\_ID INT NOT NULL, QUANTITY\_PURCHASED INT,*

*TOTAL\_AMOUNT NUMBER(9,2),*

*PRIMARY KEY(ID),*

*CONSTRAINT USER\_FK FOREIGN KEY(USERNAME) REFERENCES USERS(USERNAME)*

*ON DELETE CASCADE,*

*CONSTRAINT PRODUCT\_FK FOREIGN KEY(PRODUCT\_ID) REFERENCES PRODUCTS(PRODUCT\_ID)*

*ON DELETE CASCADE);*

**ORDERS TABLE:**

*CREATE TABLE ORDERS(*

*ORDER\_ID INT NOT NULL,*

*USERNAME VARCHAR(200) NOT NULL,*

*ORDER\_DATE DATE,*

*TOTAL\_AMOUNT NUMBER(9,2),*

*PRIMARY KEY(ORDER\_ID),*

*CONSTRAINT USERS\_FK FOREIGN KEY(USERNAME) REFERENCES USERS(USERNAME)*

*ON DELETE CASCADE);*

**ORDER ITEMS TABLE:**

*CREATE TABLE ORDER\_ITEMS( ORDER\_ID INT NOT NULL,*

*PRODUCT\_ID INT NOT NULL,*

*QUANTITY\_ORDERED INT,*

*PRICE NUMBER(9,2),*

*FLAG VARCHAR(10),*

*PRIMARY KEY(ORDER\_ID,PRODUCT\_ID),*

*CONSTRAINT PRODUCT\_FK FOREIGN KEY(PRODUCT\_ID)*

*REFERENCES PRODUCTS(PRODUCT\_ID)*

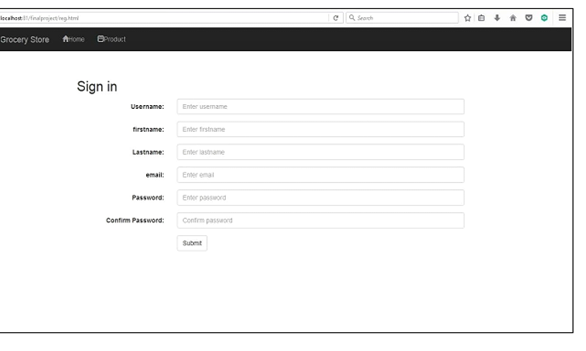
*ON DELETE CASCADE);*

# TECHNOLOGIES USED

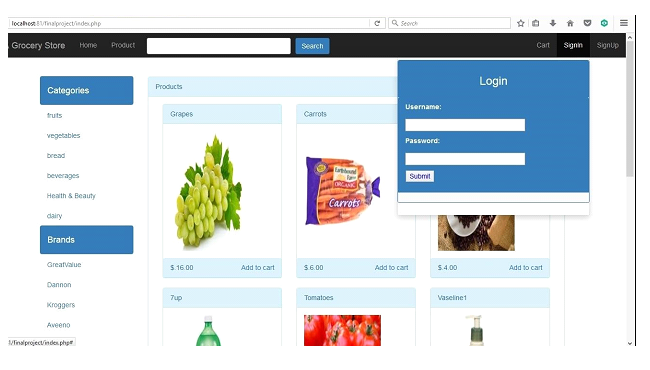
**HTML, CSS and Bootstrap** have been used to develop html pages on front end. Client side form validation is done using **jQuery**. **AJAX** has been used to make server side calls from client side. **MYSQL** has been used for database. **MAMP** server has been used for PHP and MySQL.

# SCREENSHOTS

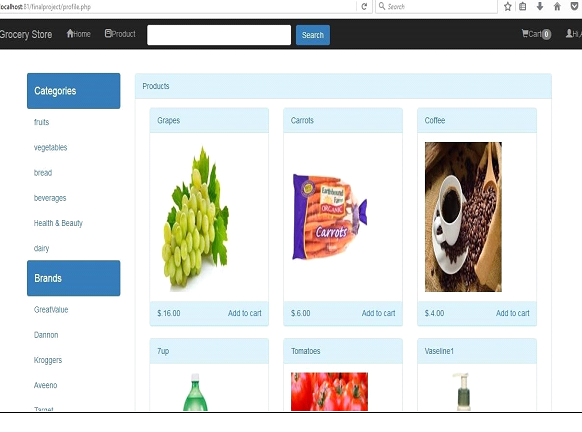
Registration page



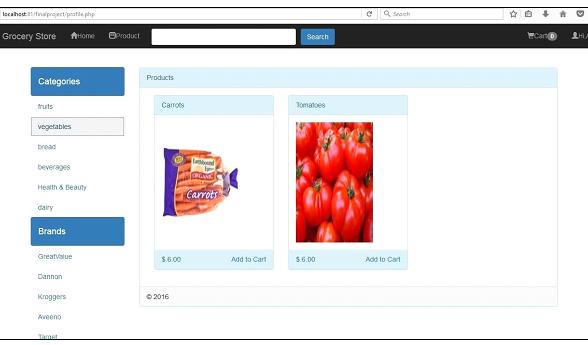
Login Page



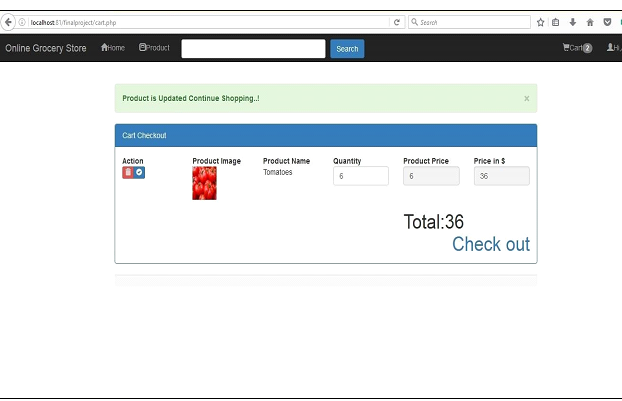
***Home Page:***

****

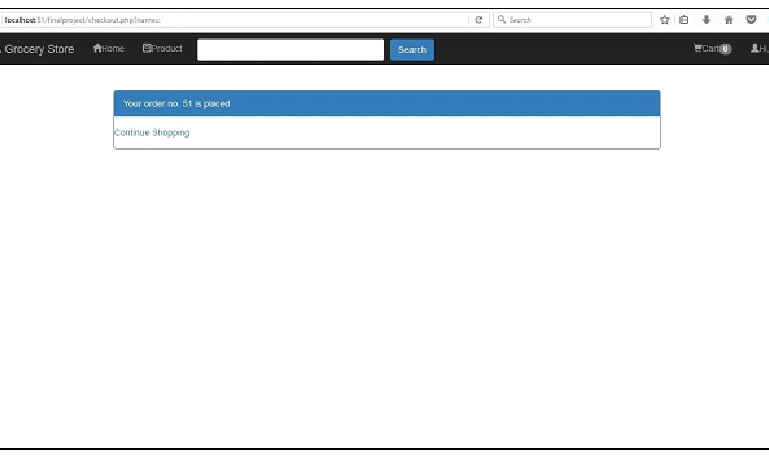
Filtering Products



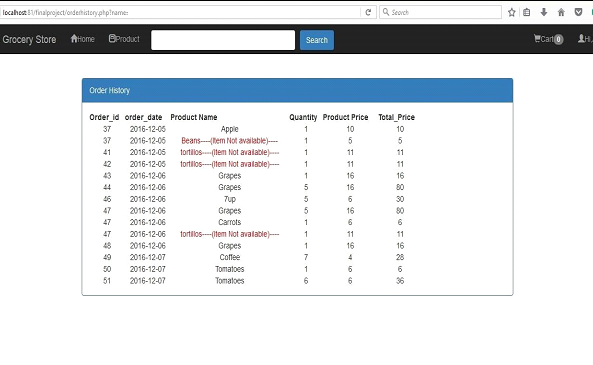
Cart Page:



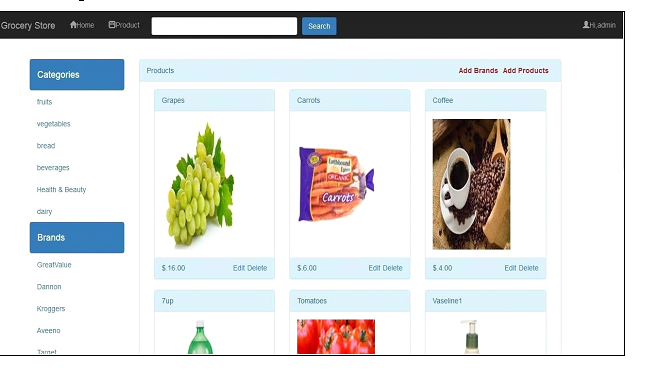
Order Page:



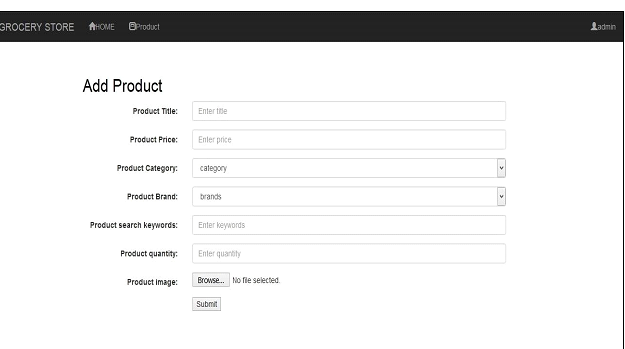
***Order History:***



Admin home page



Admin Product page: -



Admin Update Product Page :-

