Software Requirements Specification Template

EMart

Software Requirements Specification

Version 2.1

12/02/2019

Jyoshna Boppidi
Sai Prakash Reddy Mamidi
Ramesh Nutulapathi
Havya Ravipati
Sai Sri Lakshmi Vancha
Aakash Valluru
Venkata Sai Krishna Dasari

Submitted in partial fulfilment
Of the requirements of
CSIS 44-691 Graduate Directed Project II

Revision History

Date	Description	Author	Comments
6/11/2019	Version 1.0	Havya Ravipati Jyoshna Boppidi Aakash Valluru Ramesh Nutulapathi Sai Prakash Reddy Mamidi SaiSriLakshmiVancha Venkata Sai Krishna Dasari	First Revision
6/30/2019	Version 1.1	Havya Ravipati	Second Revision
12/02/2019	Version 2.1	Jyoshna Boppidi Havya Ravipati SaiSriLakshmiVancha	Third revision

Document Approval

The following Software Requirements Specification has been accepted and approved by the following:

Signature	Printed Name	Title	Date

Table of Contents — Page Number

1. Introduction 5

1.1. Purpose 5

1.2. Scope 5

1.3. Definitions, Acronyms, and Abbreviations 6

1.4. References 6

1.5. Overview 6

2.General Description 7

2.1. Product Perspective 7

2.2. Product Functions 7

2.3. User Characteristics 7

2.4. General Constraints 7

2.5. Assumptions and Dependencies 7

3.Specific Requirements 8

3.1. External Interface Requirements 8

3.1.1. User Interfaces 8

3.1.2. Hardware Interfaces 8

3.1.3. Software Interfaces 8

3.1.4. Communications Interface 8

3.2. Functional Requirements 8

3.3. Use Cases 9

3.4. Class/Objects 9

3.5. Non-Functional Requirements 9

3.5.1. Performance

3.5.2. Reliability

3.5.3. Availability

3.5.4. Security

3.5.5. Portability

3.6. Prototypes (for complete project) 10

3.7. Use Case Diagrams 13

4. Design 14

4.1. ER diagram 14

4.2. Mockups 15

5.Analysis Models

5.1. Data Flow Diagram

5.2. Sequence Diagram

6. Technical Manual 18

6.1. Document Identification 18

6.2. System Overview 18

6.3. Product Description 18

6.4. Cloud Repository 18

6.5. Planned Objectives 19

6.6. Tools Used 19

6.7. Development Process 20

1. Introduction

1.1 Purpose:

EMart is an ecommerce website to shop for various products. We are using HTML5, JavaScript, CSS, and Oracle to develop the project. This website reduces the stress on the customers by providing a good user interface. The payment is done securely using a real time system using PayPal or google pay.

1.2 Scope:

	Scope Statement
Project Title	EMart
Date	6/11/2019
Project Description	This is an interactive ecommerce website which makes the effort of shopping simple to the users. By using this web application, users should be able to buy products from the online store. The customers should be able to track their products and they should be able to Add/Remove the products from the cart without any trouble.
Requirements and Functions	● The User Interface should be simple to use. ● The website is supposed to have a login/register page. ● There should be an option for guest checkout. ● There should be a search tab where the user can shortlist the desired product by typing in the key words. ● There should be a Homepage where all the products are supposed to be listed.
Deliverables	Primary Deliverable: A prototype i.e. static to show how the web application looks like. Final Deliverable: A website with Homepage and login/register page working as stated in the requirements.
Future Scope	At a later point, This website will be added with pages like My Cart, Orders etc. There is a plan to implement some more features where the user can filter the desired product by adjusting the price, color or gender etc.

1.3. Definitions, Acronyms, and Abbreviations:

ER – Entity-relationship diagram

1.4. References

- Codecourse (Director). (Feb 24, 2012). *PHP Tutorials: Email Address Validation* [Motion Picture]. From **https://www.youtube.com/watch?v=5qElPBrtHCg**
- Refsnes Data. (1998). *HTML, CSS, JavaScript*. From W3 Schools: **https://www.w3schools.com/**
- Turki, A. H. (2018, Feb 18). *phpmailer complete tutorial with debugging*. From myphpnotes.com: **https://www.myphpnotes.com/post/phpmailer-complete-tutorial-with-debugging**

1.5. Overview:

EMart is an interactive ecommerce website to shop for various products. For this website we are using HTML5, CSS, and JavaScript in front-end development, PHP as middleware and MySQL in back-end development.

2. General Description

2.1 Product Perspective:

EMart is a one stop website application for purchasing products online. The main features of this application is:

- Login & Signup: User can register to the site and create an account at ease.
- Home Page: User can view the different products based on different categories.
- Account Profile: User can check order history and track packages.
- Chat Page: Can easily reach customer service for any assistance related to the product.
- Ratings: User can compare ratings and reviews of each product before purchase.
- Payment Page: User can choose the payment methods.
- Forgot Password: User can retrieve the password.

2.2 Product Functions:

Using this webpage, a user will be navigated to different departments where user gets an experience to visit different products in each categorized department. They get an option to add to cart or buy now button.

2.3 User Characteristics

A user can review the products and have a glance at frequently asked questions and can purchase a product even without signing up.

2.4 General Constraints

Extraction of data from the database server might delay the response time depending on the internet service provider.

2.5 Assumptions and Dependencies

We assume that the applications will be suitable on all the devices with any type of browser installed in it.

3. Specific Requirements

3.1. External Interface Requirements

3.1.1. User Interfaces:

The website will be using the Menu driven Interfaces where the customer can choose a category from the menu bar, navigation bar and other options that will be provided on the web page.

3.1.2. Hardware Interfaces:

This website should be able to work on any device with an appropriate browser.

3.1.3. Software Interfaces:

- Visual Studio Code
- Oracle

3.1.4. Communications Interface:

We are using Skype, Emails for communication interface.

3.2 Functional Requirements:

Login Page: This page should display two text fields with Username and password. There should a feature Forgot Password which helps the users to recover the password.

Admin Login Page: The admins and suppliers can login here in order to add the products, review the products they had added and check the orders they had received. The suppliers should register with the admin directly, so there is no register option for suppliers.

Register Page: The users without a Username and password must register in this page. This page consists of username, email address, password and contact information.

Home Page: The products are supposed to be displayed in this page. There should be an image, price and product description for every item in this page.

Forgot Password Page: This page allows user to change the password and when user clicks on forgot password, the link is sent to the users registered Email.

Admin Home Page: This page is where the admin can see the added products, admin can add the new products from this page. They can review on the received orders and post the status of the orders.

My Cart Page: The users can see the products they want to order and add/remove the products from the cart.

My Profile Page: The user can view/edit his profile which consists of User Id, addresses, payment information and they can change the password and also the payment information.

Support/Customer service Page: The user can easily reach customer service for any assistance related to the product using this page.

Payment page: The user should enter the card number, expiration date, CVV etc in this page to complete the order (transaction).

Menu (Navigation) Bar: The list of categories are displayed in this bar.

3.3. Use Cases:

A use case is a hierarchal steps typically defined as the interaction between end user and the system to achieve a goal. Use case analysis are widely used in modern day software engineering.

3.5. Non-Functional Requirements

3.5.1. Performance:

Web performance is the speed that web pages are loaded onto a client's web browser. In EMart website customer can login, signup and view products in less than seconds. The website will be maintaining high performance all the time.

3.5.2. Reliability:

EMart is a reliable website where all the ACID rules are applied in order to provide a reliable website for the users.

3.5.3. Availability:

Ensure that the website is available all the time for the user to view the products and place the order.

3.5.4. Security:

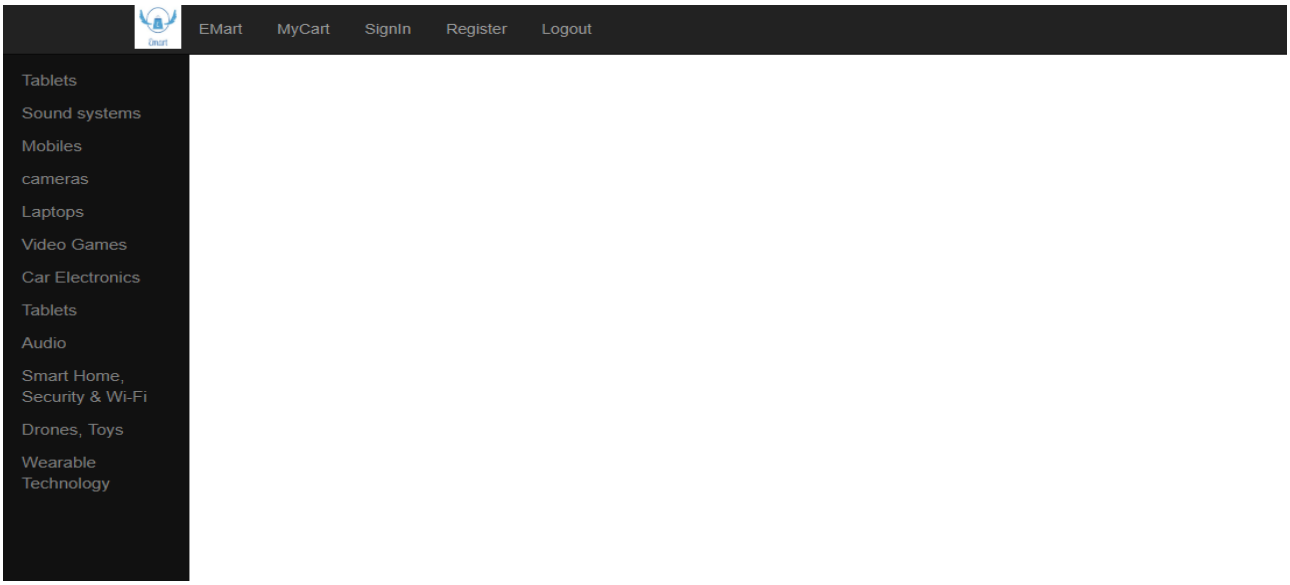
The website ensures that the payment information and the personal information of the customer is safe and secured.

3.5.5. Portability:

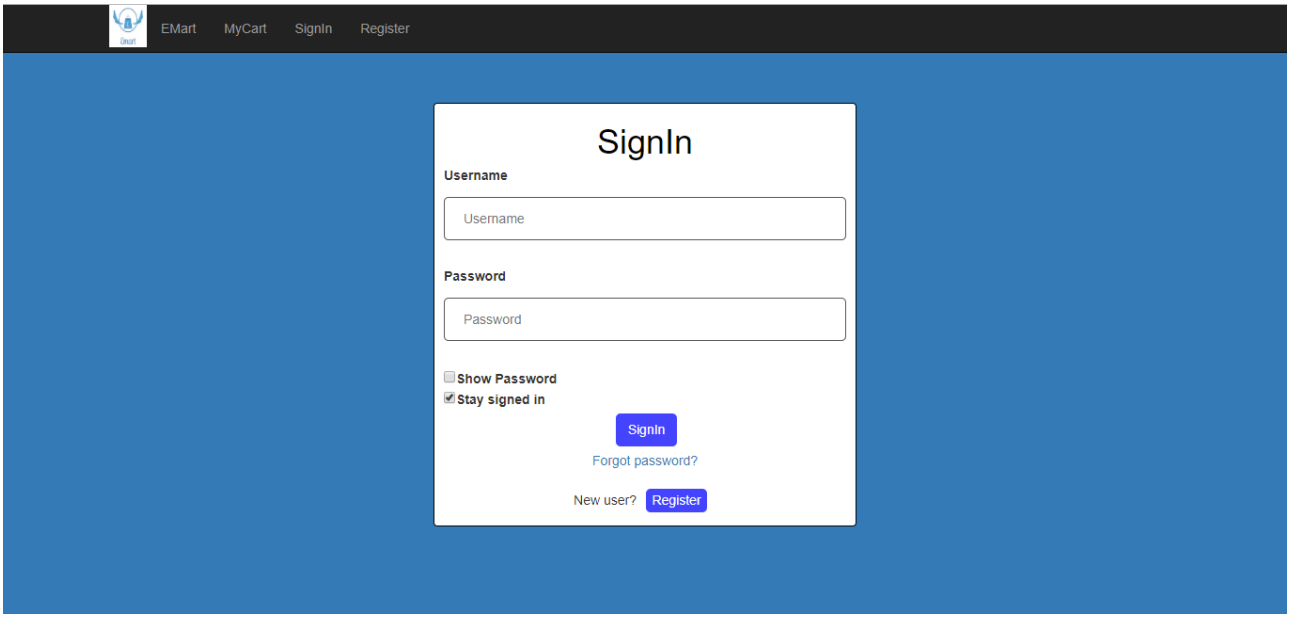
The website can be used on multiple platforms or instantly accessed from any device with proper internet and software.

3.6. Prototypes (for complete project)

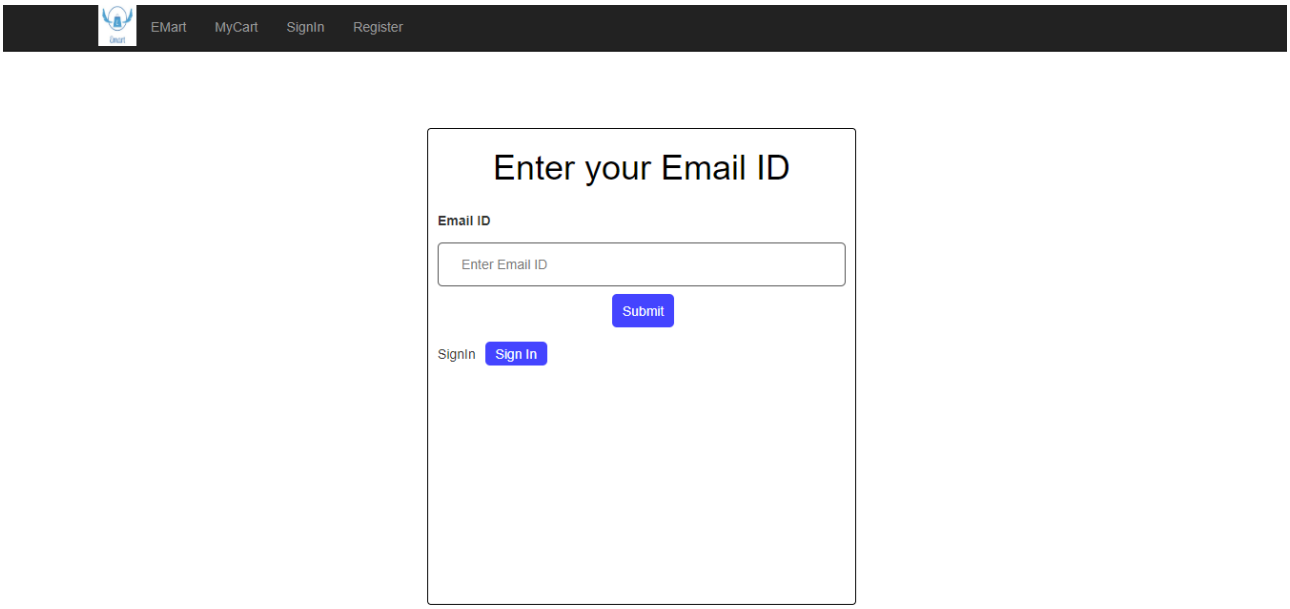
Home page: In this page different categories of the products are displayed and there should be an image, price and product description for every item.



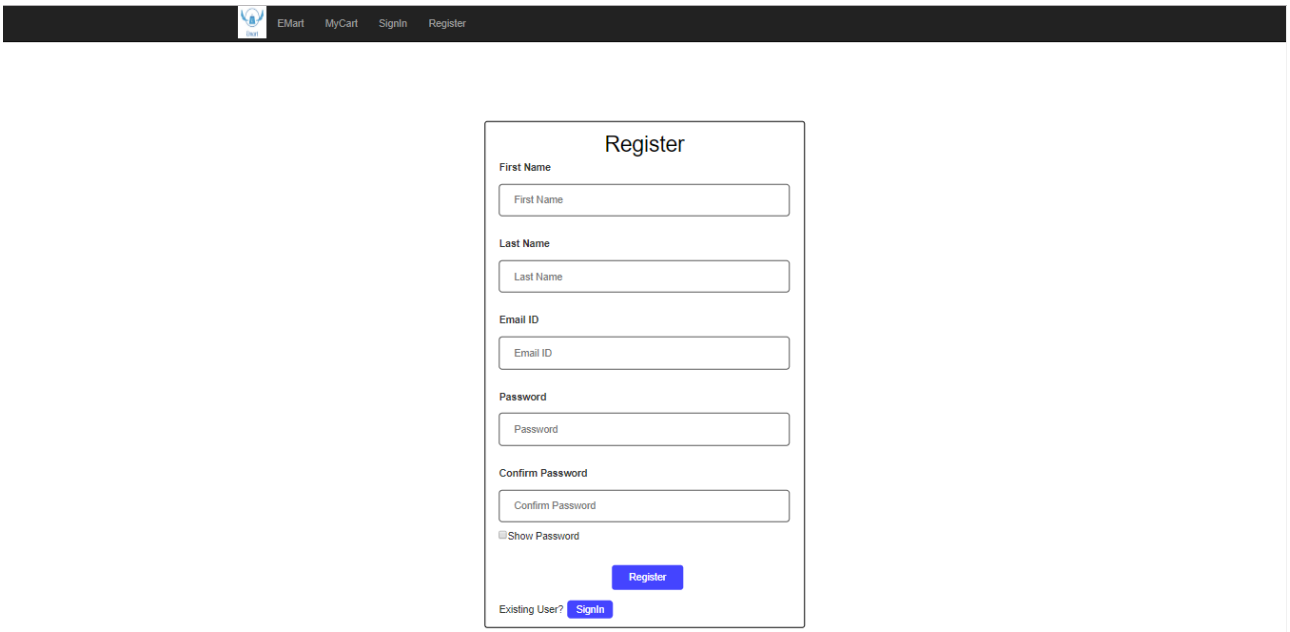
Login/Register page: In this page the user can see the two text fields with Username and password.



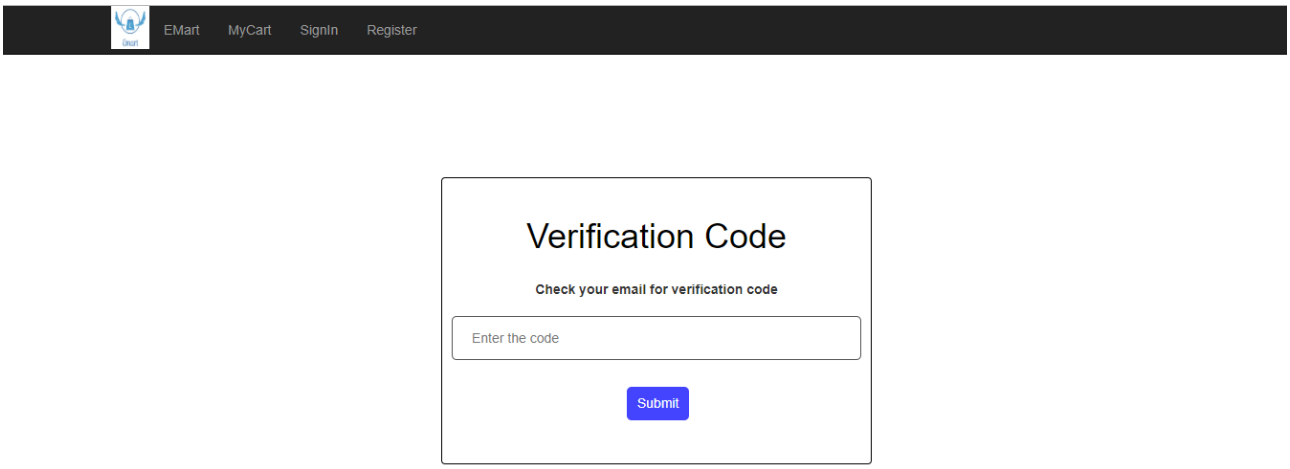
Forgot password: In this page the user can see the forgot password feature which helps the users to recover the password.



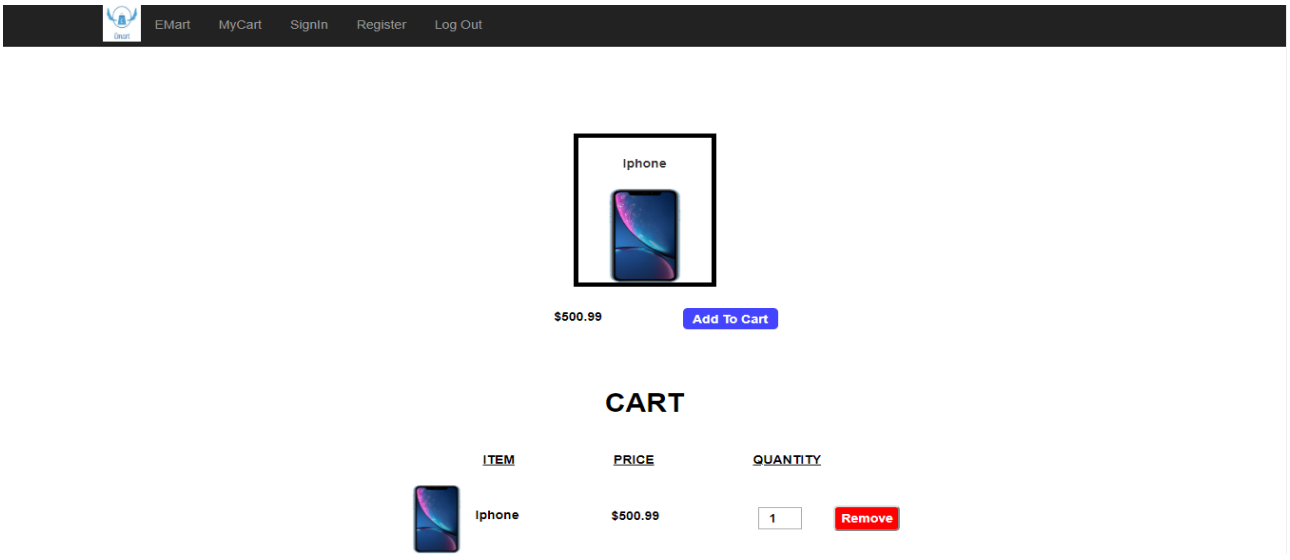
Registration page: In this page the users without a Username and password must register in this page. This page consists of username, email address, password and contact information.



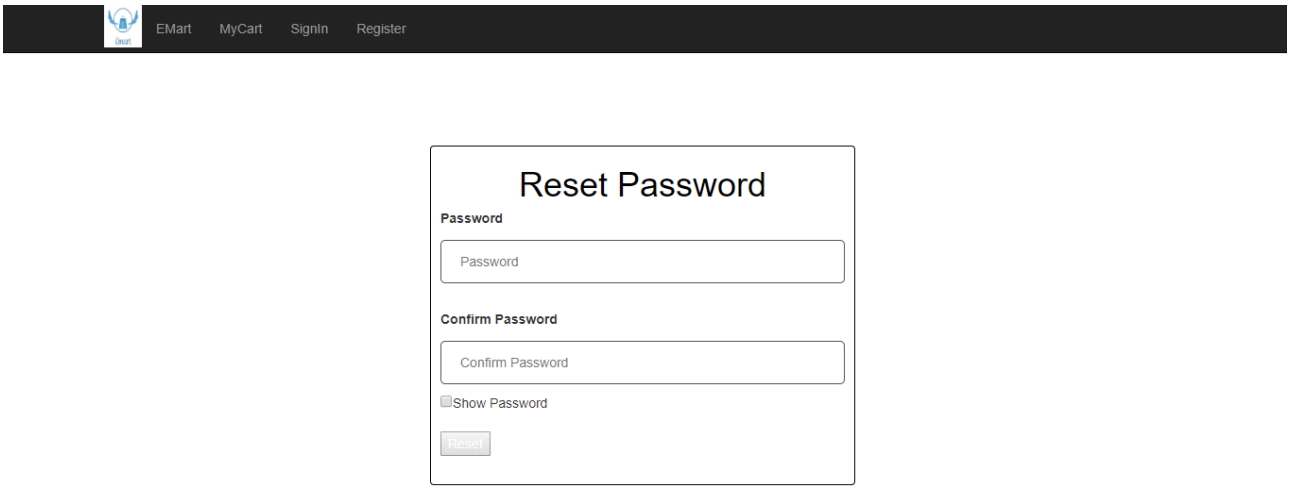
Verification code: In this page the users will get the verification code while doing the registration process to the website.



My Cart Page: In this page the users can see the products they want to order and add/remove the products from the cart.

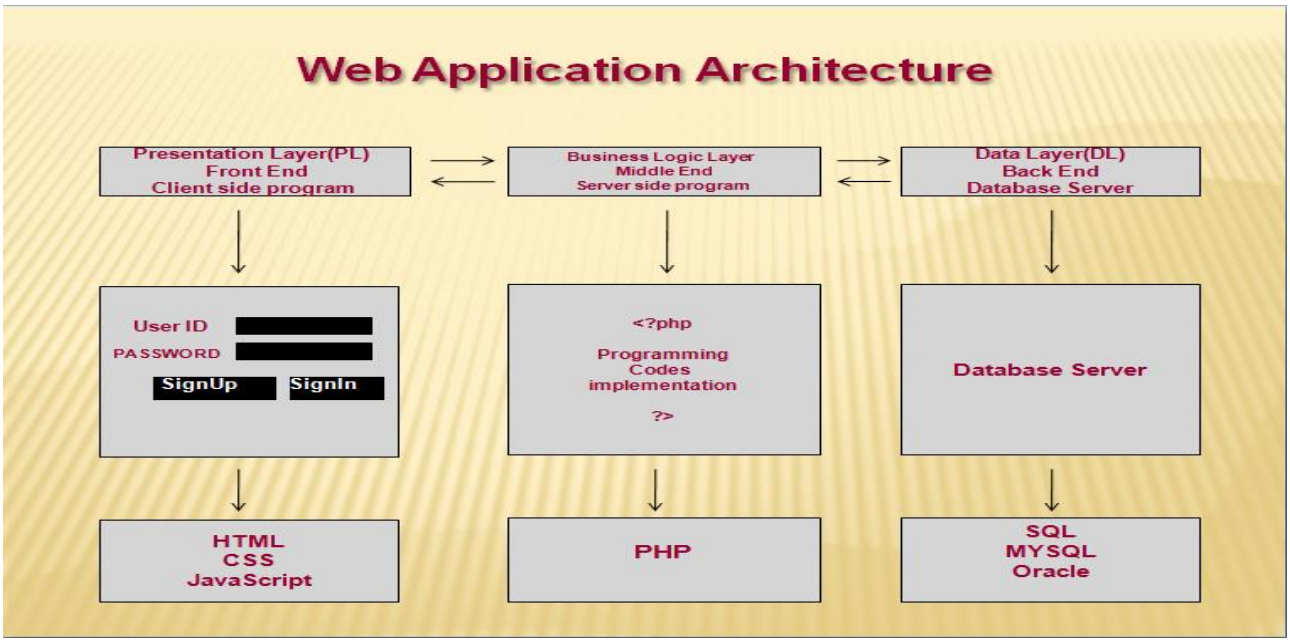


Reset Password Page: In this page user can reset the password by entering the text fields password and confirm password.



3.7. Use Case Diagrams

Web Architecture:



4. Initial Draft of Design

In this chapter we are going to discuss about the ER diagrams and Mockups we used while creating and developing this project idea.

4.1. ER Diagram:

Our ER Diagram has 9 entities with 2 associative entities.

In the Customer entity, the information about the customer such as name, address, phone number etc. are stored.

The Seller entity is a weak entity and used to store the seller ID, first and last name, company, email, phone number, address, city, state, zipcode and country.

In this ER diagram product seller is associative entity between the product and seller. It stores seller ID, product ID, manufacture date, quantity on hand. Here the seller ID, product ID acts as both primary key and foreign key.

The order entity is used to store the order ID, product ID, Customer ID, order date, delivery date and payment ID. Here the order ID plays as a primary key in order to make the orders unique.

The product entity stores the product ID as primary key, seller ID, category ID as foreign keys, name, price and description of the product.

Here, the customer payment details are a weak entity which depends on the customer entity because the customer payment details exist only when a customer exists.

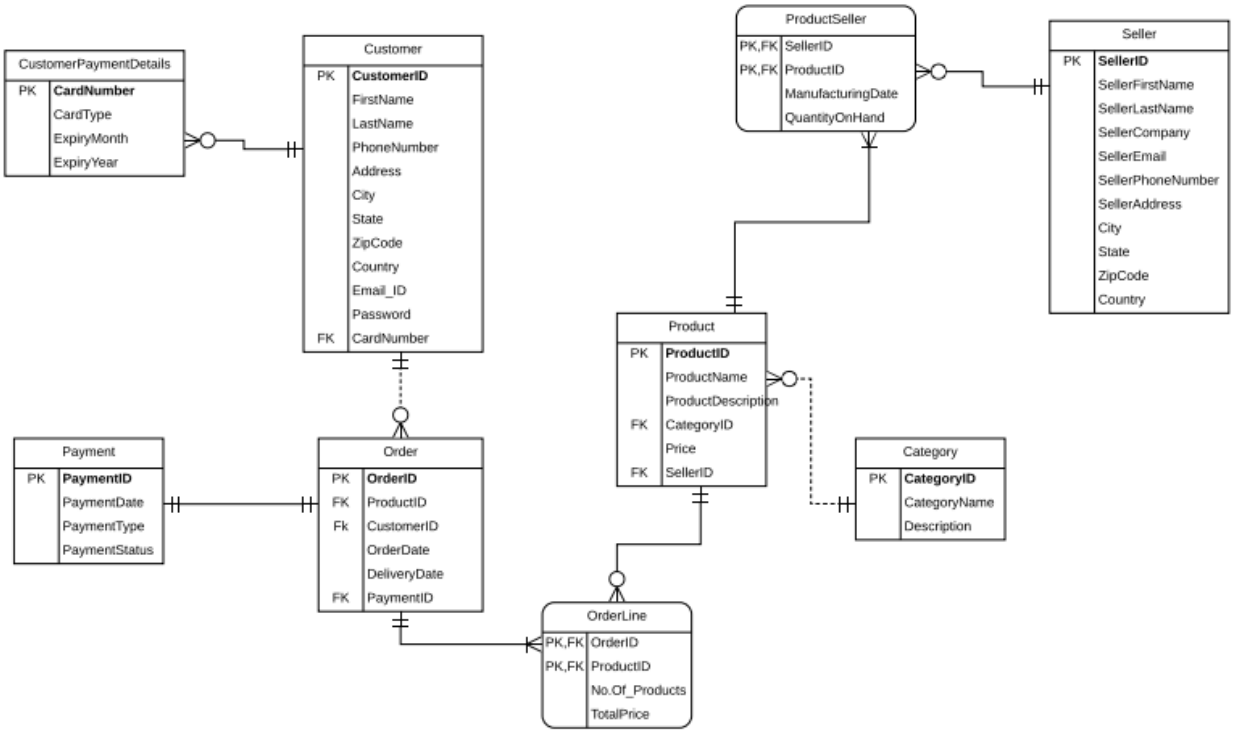
Similarly, payment is also a weak entity because the payment only exists when there is an order. The product seller and order line are considered as weak as well as associative entities.

In the below diagram, there is a mandatory one to optional many relationship between the customer and the order because a customer can place any or no number of orders, but the order should be placed by exactly one customer.

The relationship between the order and the product is considered many too many because a single order must consist of at least one product or many products whereas the single product can be in multiple orders or may not be in any order.

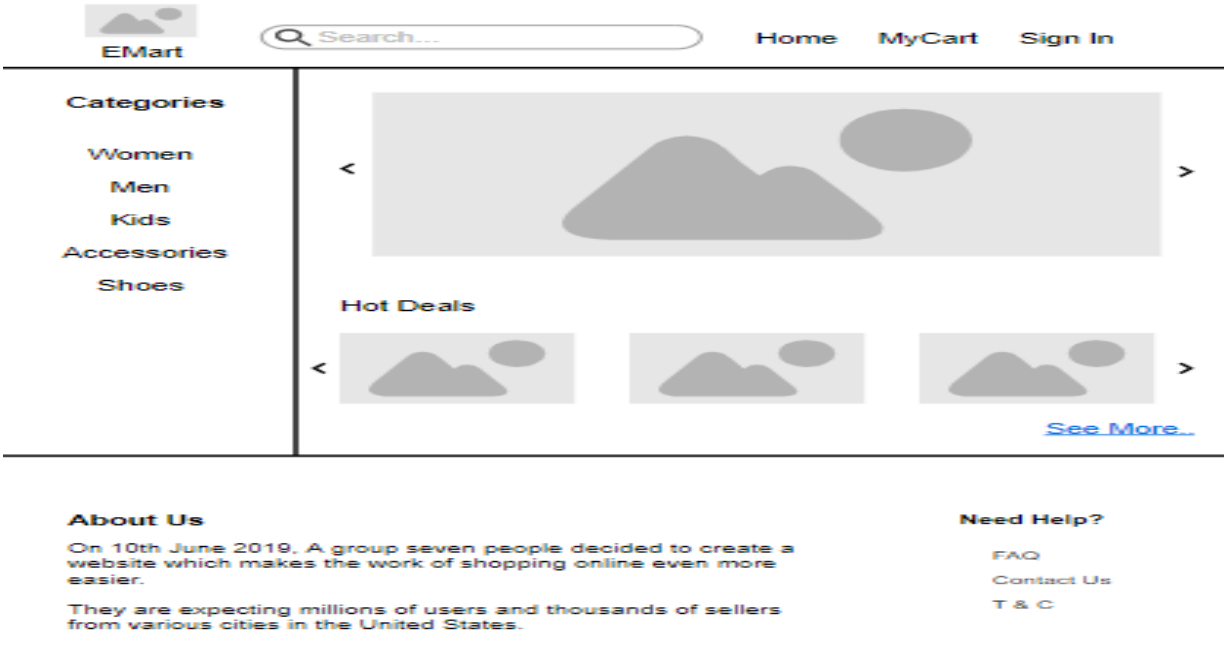
A seller can add multiple products or no products at all whereas the product can be sold by one or many sellers. Therefore, the relationship between the seller and product is many to many.

A product should belong to exactly one category whereas the category can no products or multiple products. So, the relationship was considered as mandatory one to optional many.

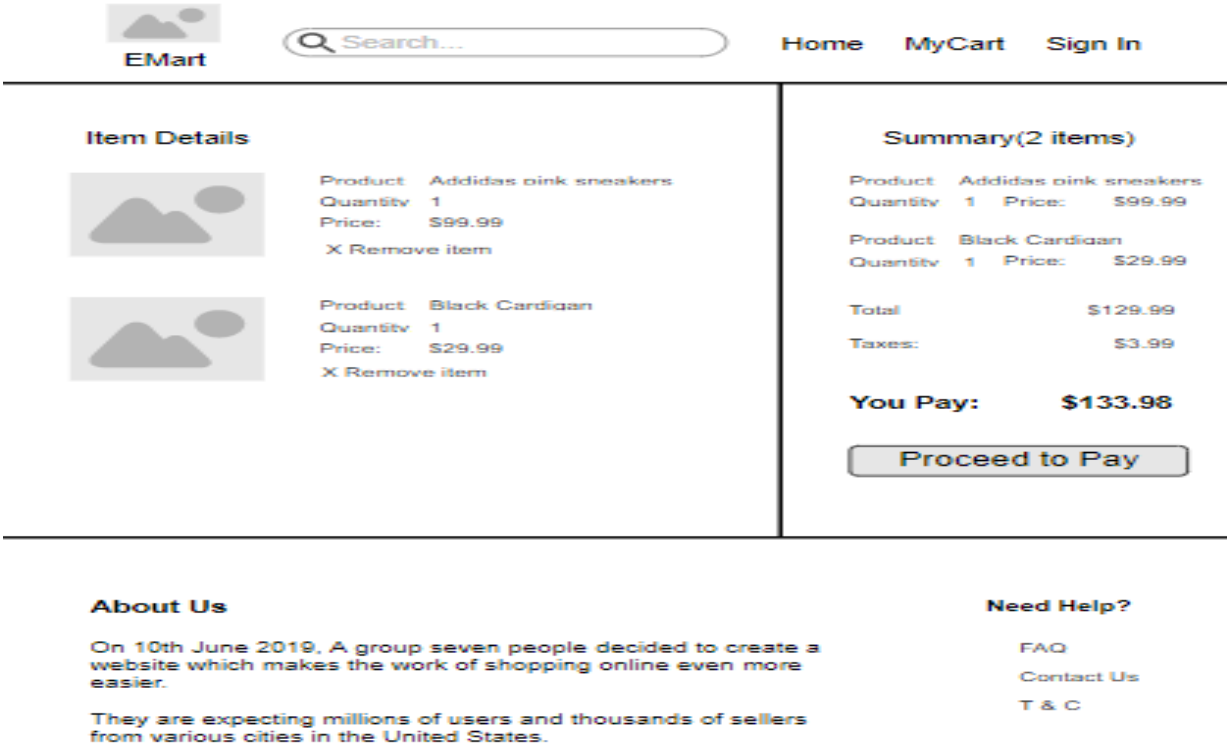


4.2 Mockups:

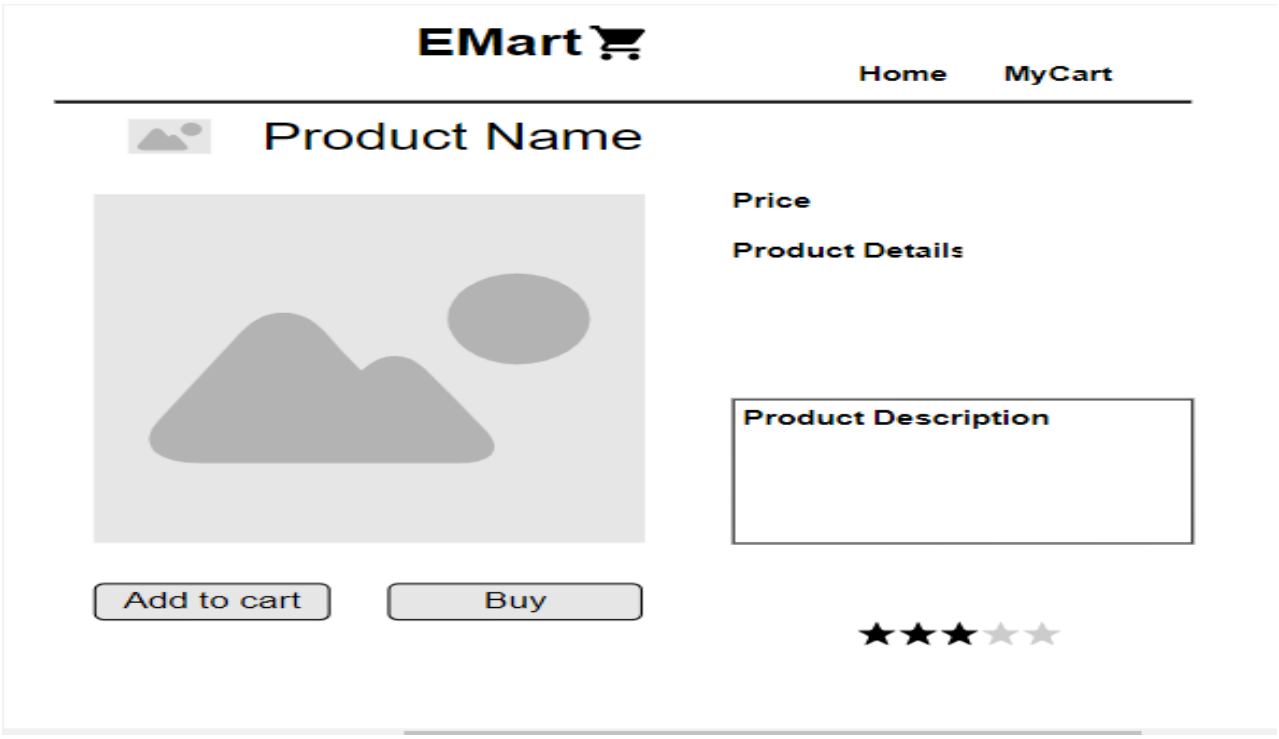
Home Page:



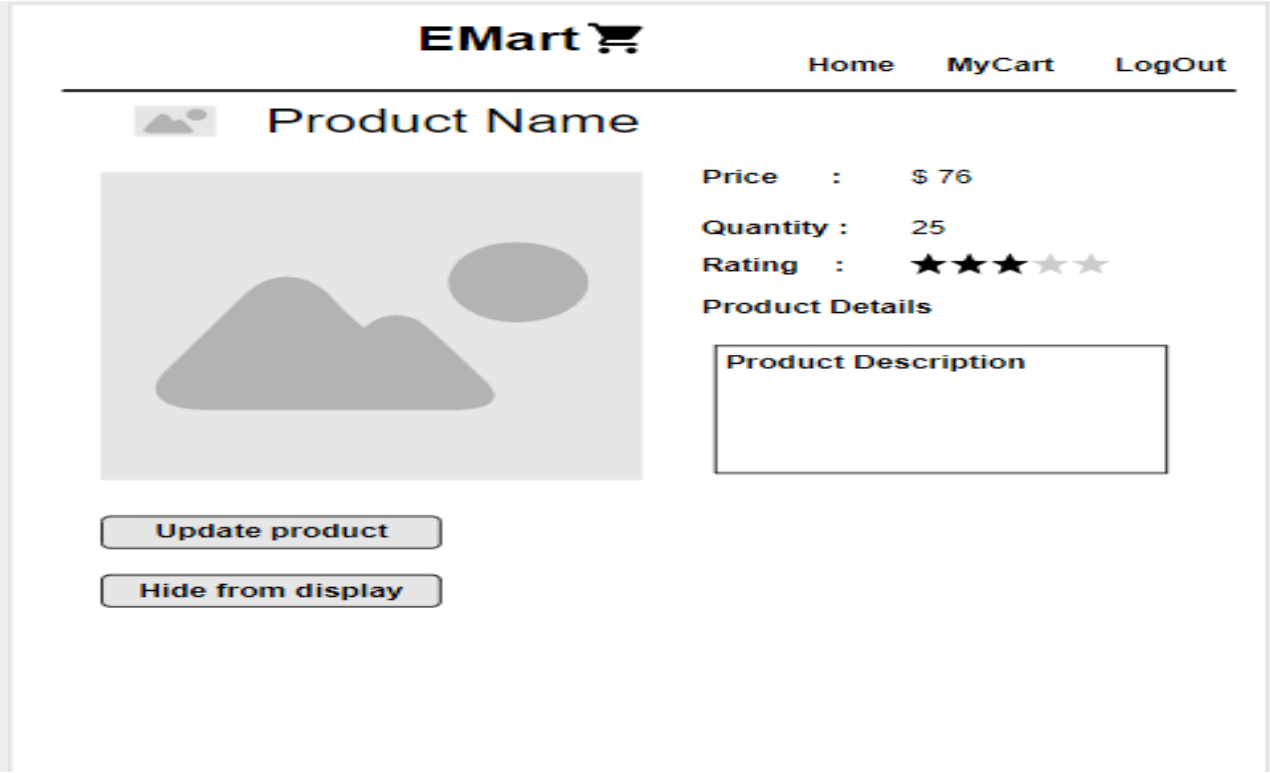
My Cart page:



User Product Page:

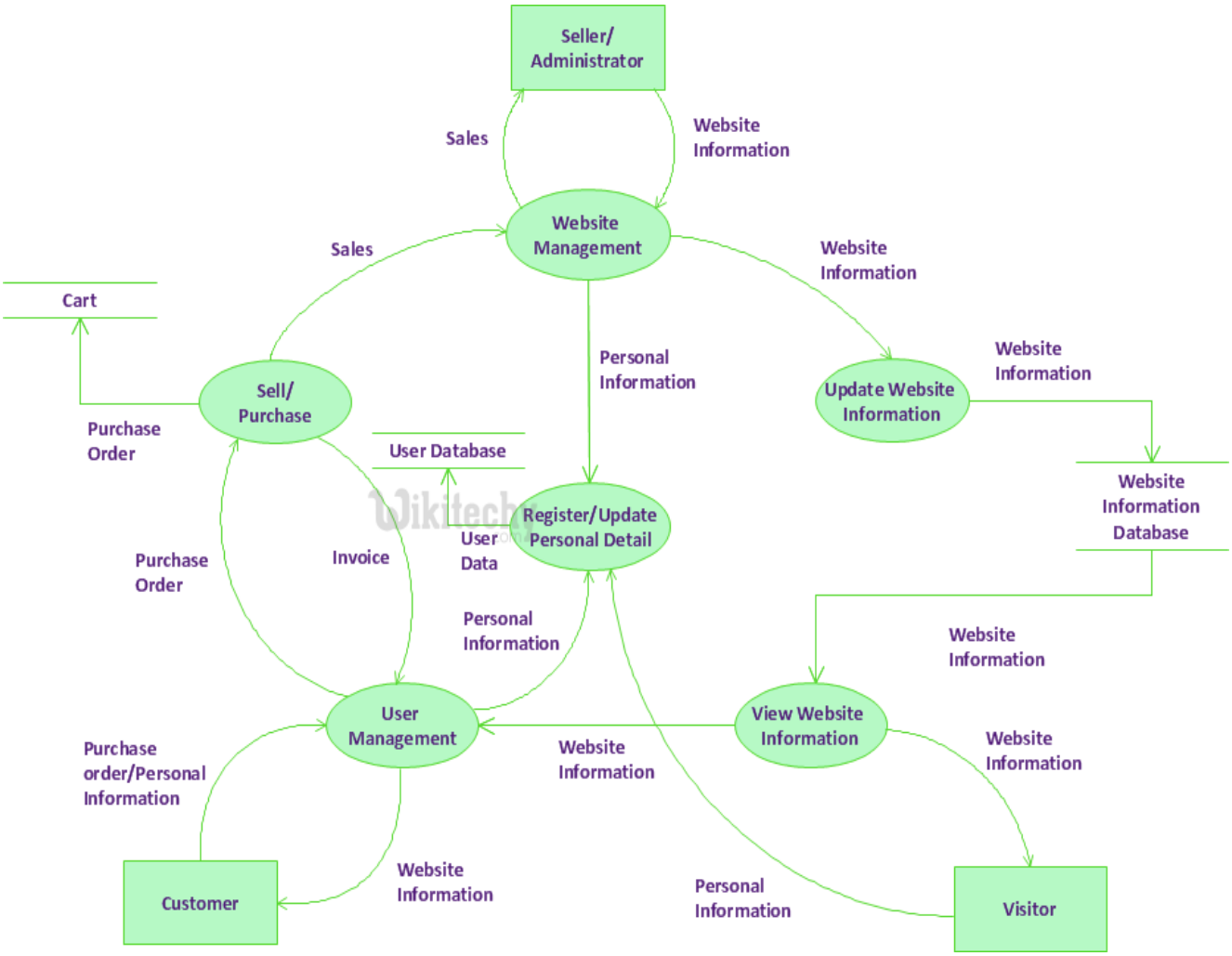


Admin Product Page:

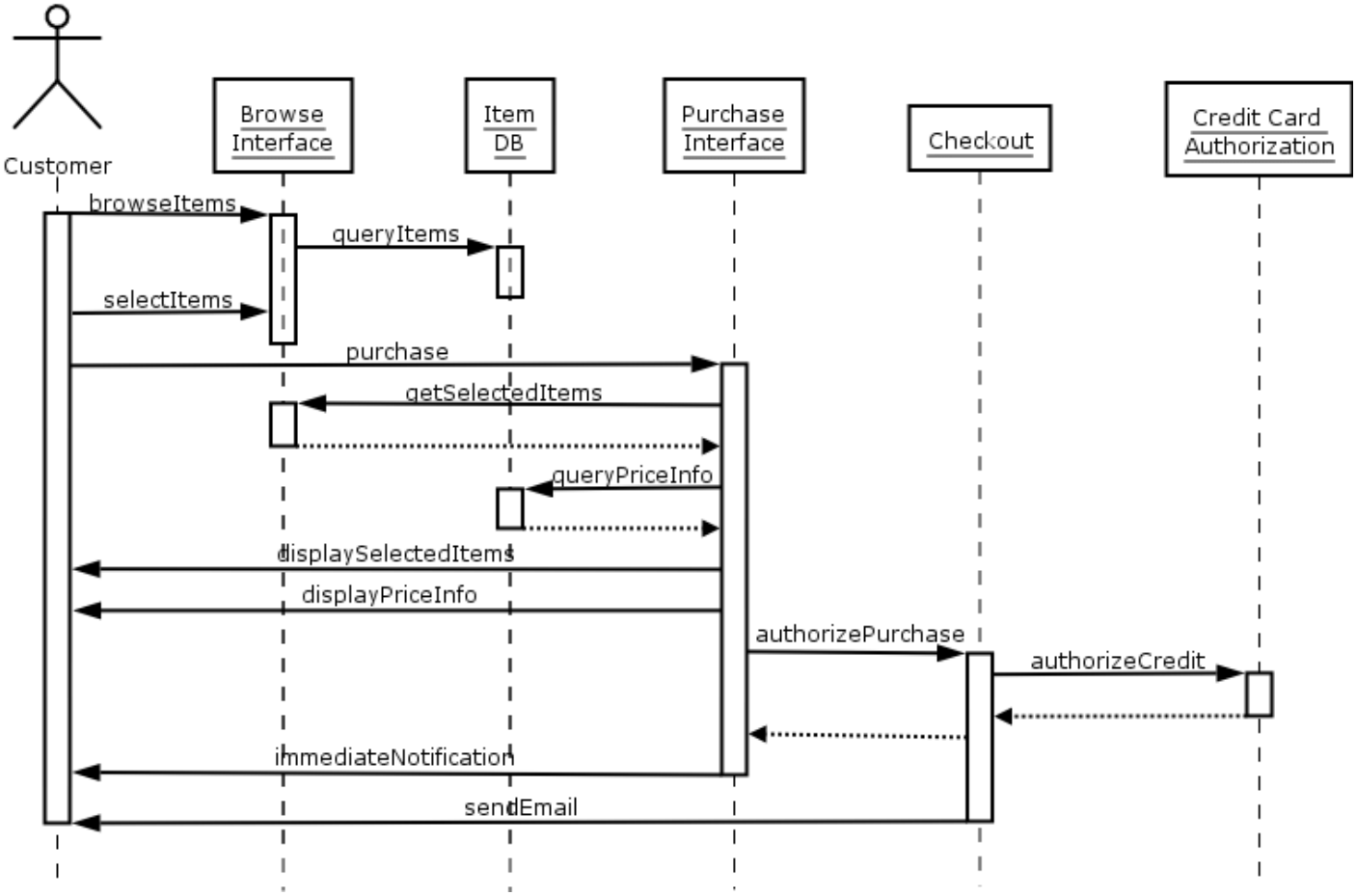


5. Analysis Models

5.1. Data Flow Diagram



5.2. Sequence Diagram



6. Technical Manual

In this manual we will discuss the overview of the technologies we used and how to do the installation of some tools.

6.1. Document Identification:

This document describes the technical aspects of design and implementation of the EMart web application. This document is prepared by the Web Application Team of GDP-02 (44691-04) Summer 2019

Product Name: EMart (Ecommerce website)

6.2. System Overview:

EMart is an online e-commerce application which allows the user to register and login. The user will be able to browse through the different categories of electronic appliances and place their order. Once the user places an order, he/she will be redirected to the payment page and once the payment is done, the order will be accepted by us and provide the user with shipping details.

6.3. Product Description:

This website has all the features that are generally found in an ecommerce website. They include home page, my cart page, product page, payment page, login page, register page etc.

6.4. Cloud Repository:

The website is being built using the GitHub repository. The repository consists of codes, documentation and readme.md. The codes folder consists of the actual codes that are required for the website. These codes have been developed using various to tools. The documentation folder consists of the various documents that help to check the progress of the project.

Link to the cloud repository: https://github.com/prakashreddy-97/EMart

6.5. Planned objectives:

Phase-1: This phase consists of implementation of the login and register pages.

Objectives	**Status**
Decide the name of the product	Done

Gathering the requirements for the entire product	Done
Build the ER diagram for the entire website	Done
Develop prototypes	Done
Implementation of the frontend part of the website with login and register pages	Done
Implementation of the backend part for the login and register pages	Done
Connect the frontend and backend for the login/register pages	Done
Implementation of Email verification for security purposes	In progress
Implementation of password recovery	In progress

6.6. Tools Used:

Tools	**Front-ware**	**Middle-ware**	**Back-ware**
HTML5	✓		
CSS4	✓		
JavaScript	✓		
PHP		✓	
MySQL			✓

HTML5: HTML5 is used to create web pages and to format text as titles and headings. It defines the structure and layout of a web document by using a variety of tags and attributes.

CSS Version 4: CSS is used for describing the presentation of web pages, including colors, layouts, and fonts. It allows one to adapt the presentation to different types of devices, such as large screens, small screens, or printers.

PHP 7.1: PHP is a server-side scripting language which is used to develop static or dynamic websites or web applications.

JavaScript: JavaScript is most commonly used as a client side scripting language. This means that JavaScript code is written into an HTML page. When a user requests an HTML page with JavaScript in it, the script is sent to the browser and it's up to the browser to do something with it.

MySQL: MySQL is a Relational Database Management System (RDBMS) that uses Structured Query Language (SQL). SQL is the most popular language for adding, accessing and managing content in a database. It is most noted for its quick processing, proven reliability, ease and flexibility of use.

6.7. Development Process

Front-end Implementation:

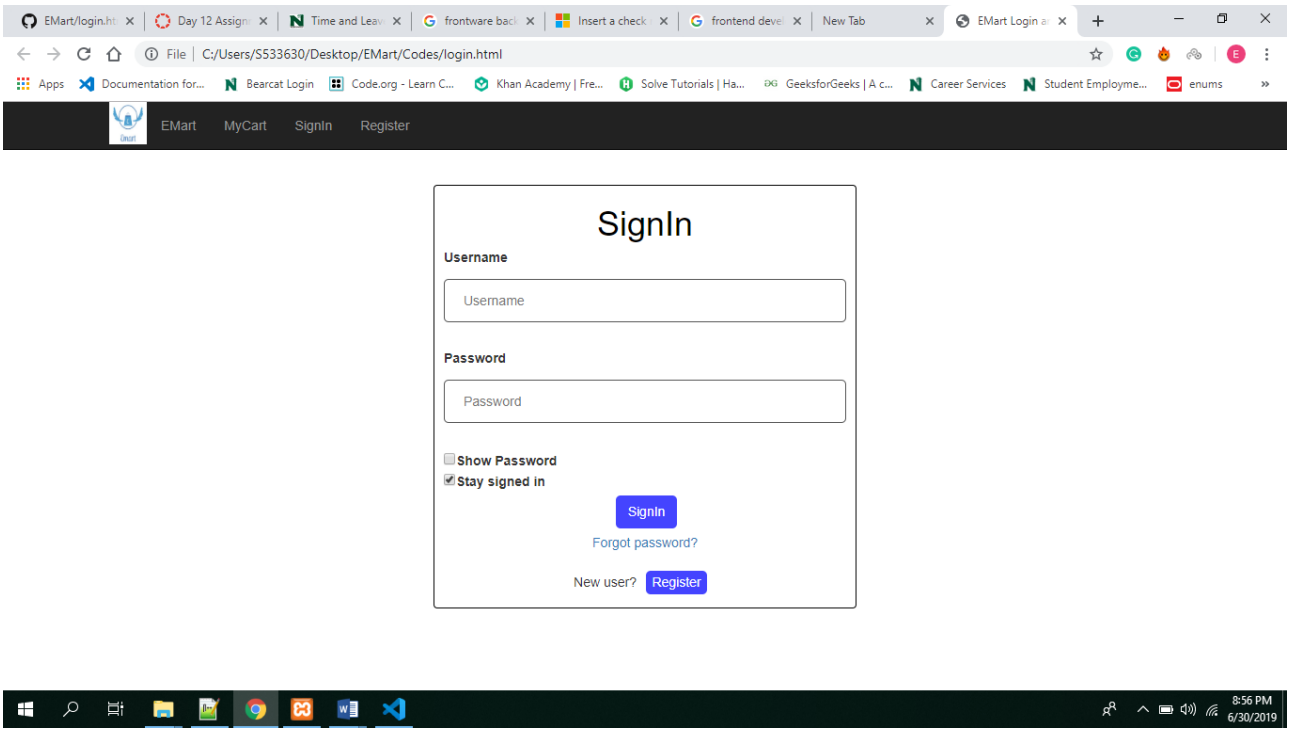
In the GitHub EMart Repository, there are several files including .html, .css & .js as their extensions. All these folders have been created and developed by the frontend developers. This website was made responsive using bootstrap.

Login page:

In the above image, there are mainly two fields where the user can enter the username and password. By default, the username is the email address of the user. The fields' username and passwords are given as inputs in the login.html. This page also consists of Sign-In and Register buttons where the sign-in button takes the user to the home page of the website (if the user name and password are correct). If there is a new user, the user can register by simply clicking on the register button. The user can't leave any of the above mentioned text fields empty. If the user leaves these text fields empty, there would be an alert stating that he has to fill all the fields.

login.html: https://github.com/prakashreddy-97/EMart/blob/master/Codes/login.html

login.css: https://github.com/prakashreddy-97/EMart/blob/master/Codes/login.css



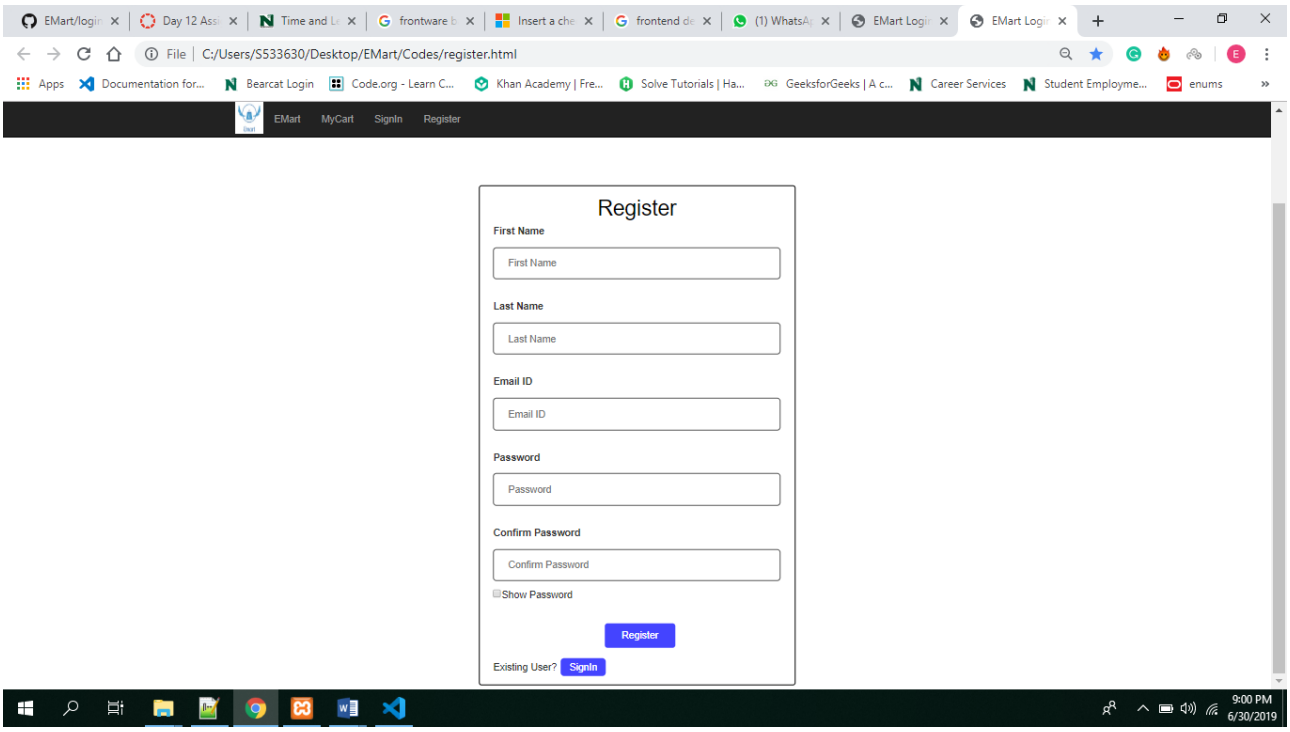
Register page:

In the below image, The user enter the first name, last name, Email Id, Password, Confirm Password. All these above fields are given as inputs in the register.html code. The fields first name, last name, Email id, password and confirm password are given as inputs in register.html. There are certain validations made in the register.html to make the user account secure.

If the user leaves these text fields empty, there would be an alert stating that he has to fill all the fields. The password requires special characters, numerical value, uppercase alphabet and at least one lowercase alphabet. If the user enter different values in password and confirm password, the user may not be able to register.

register.html: https://github.com/prakashreddy-97/EMart/blob/master/Codes/register.html

register.css: https://github.com/prakashreddy-97/EMart/blob/master/Codes/register.html



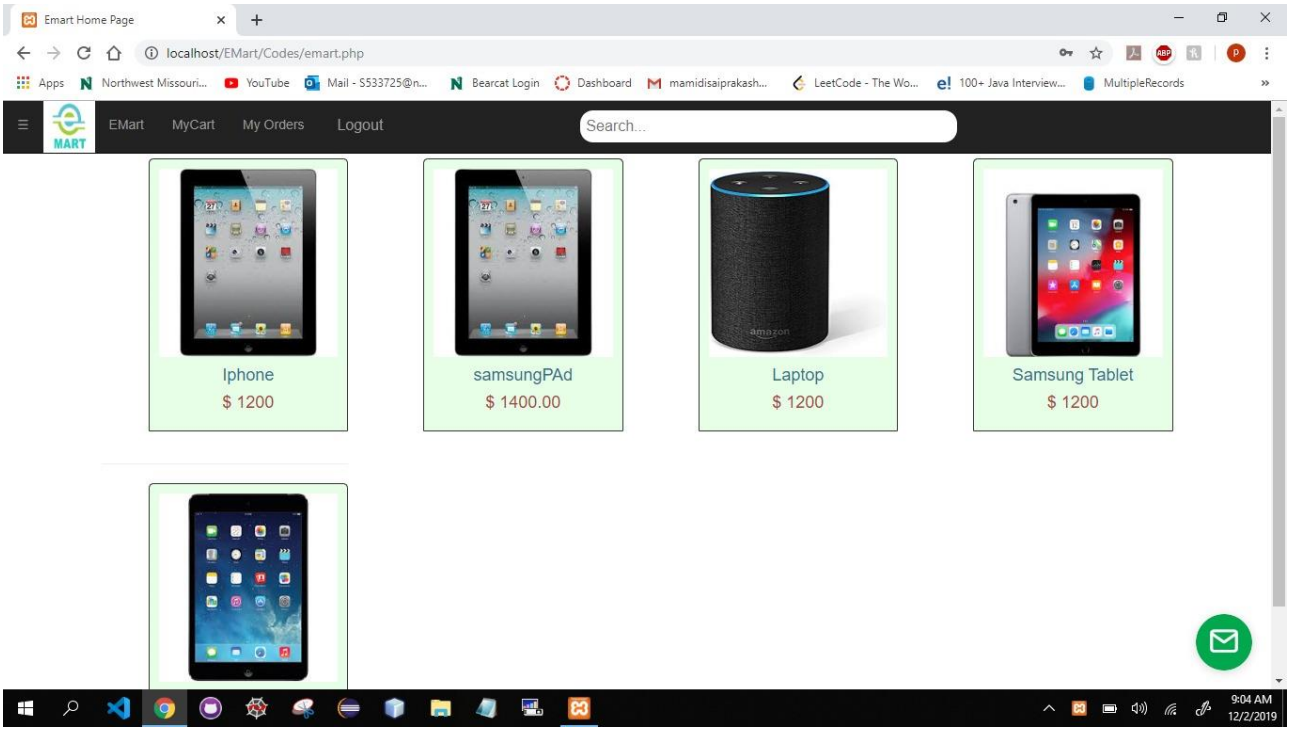
Home page:

The user is supposed to be navigated to the below page once he/she successfully logs in. The products will be displayed in this page. An array has been used to store the details of the product such as unique id (auto generated), product price, product name and quantity of the item. If the users selects a specific product to add to the cart. The details of the product are stored and will be carried to the next page i.e. cart page.

There will be several alert messages to help the user understand the action he/she has performed such as “product has been added to the cart” when a product is added and similarly “product already exist in the cart” when a product is deleted from the cart etc.

Emart.php: https://github.com/prakashreddy-97/EMart/blob/master/Codes/emart.php

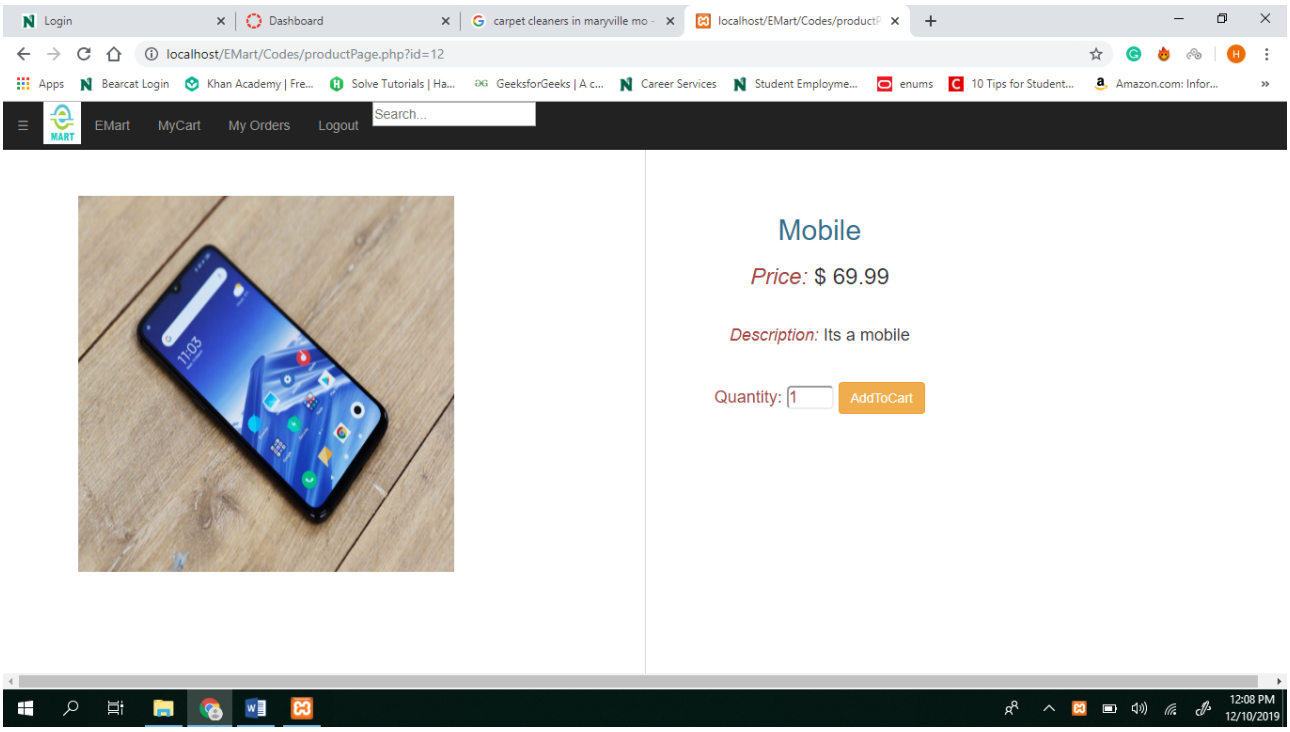
Emart.css: https://github.com/prakashreddy-97/EMart/blob/master/Codes/emart.css



Product Page:

There are several elements in this page such as the price of the item, description of the item, quantity and a button to add the object into the cart. The product from home page are dynamically carried to this page. Several variables such as product name, unique id, product description are given to store the details of the product.

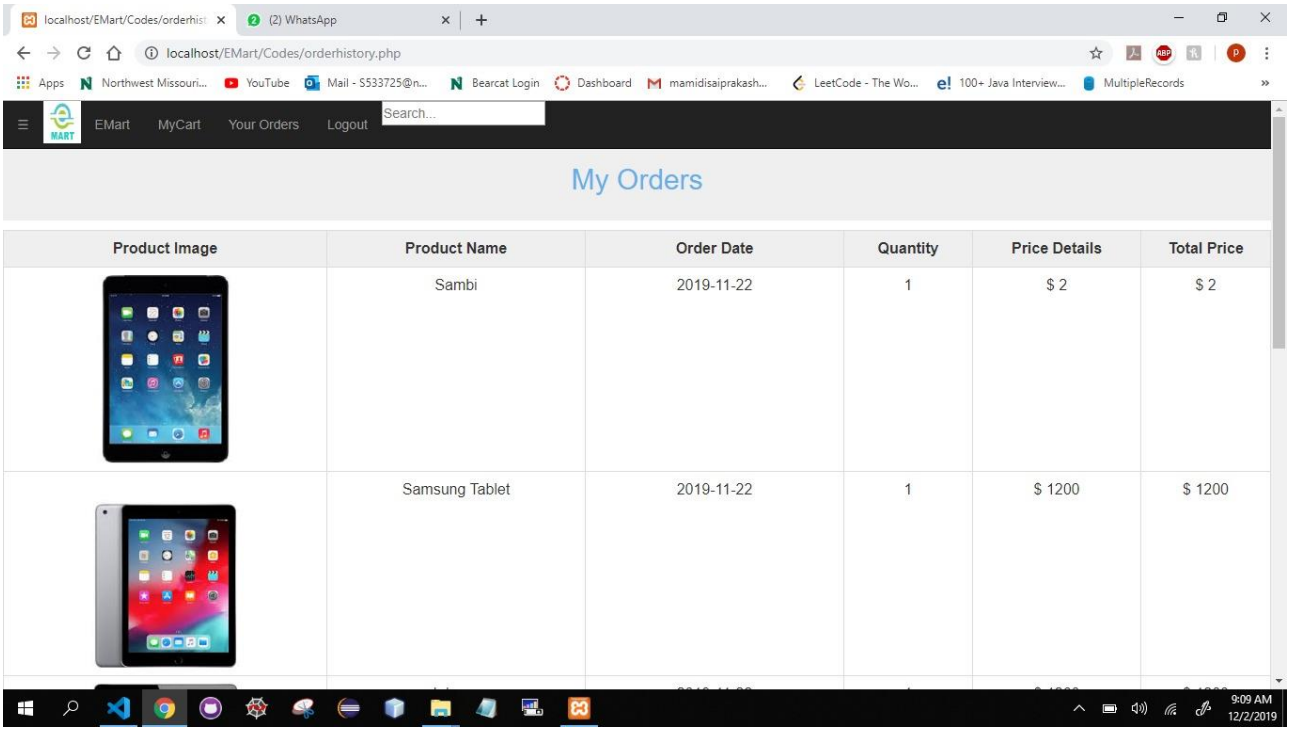
productPage.php: https://github.com/prakashreddy-97/EMart/blob/master/Codes/productPage.php



Cart Page:

In product page, there is an “add to cart” button. This page has been designed in such a way that by hitting that button the user can add products to the cart page. With the help of unique id the product that has been selected will be navigated to the cart page. The user will have the ability to have a look at the image, price and quantity of product. There is chance to delete the product from the cart page to remove the item from the cart. The total price of the products will be represented at the bottom of the page with checkout button.

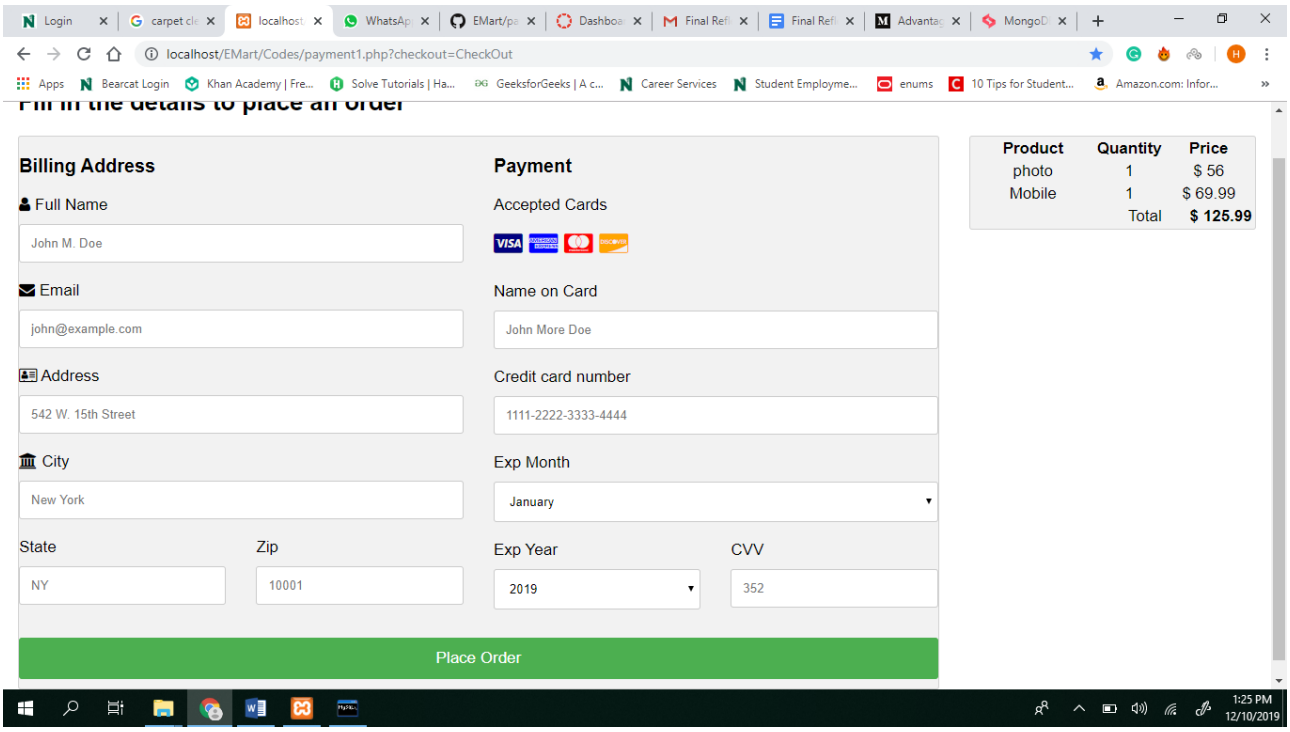
Mycart.php: https://github.com/prakashreddy-97/EMart/blob/master/Codes/mycart.php



Payment page:

The page has been designed in such a way that the user can navigate to the payment page by clicking on the checkout button in cart page. The payment page consists of several fields such as full name, email id, address of the customer, city, state and zip. All these will be stored in the customer table in the backend to maintain the records of the customer. Also, the card details shall also be obtained for the payment confirmation. Once, the order has been place the user shall be sent an email about the order confirmation and a page that states the order has been placed.

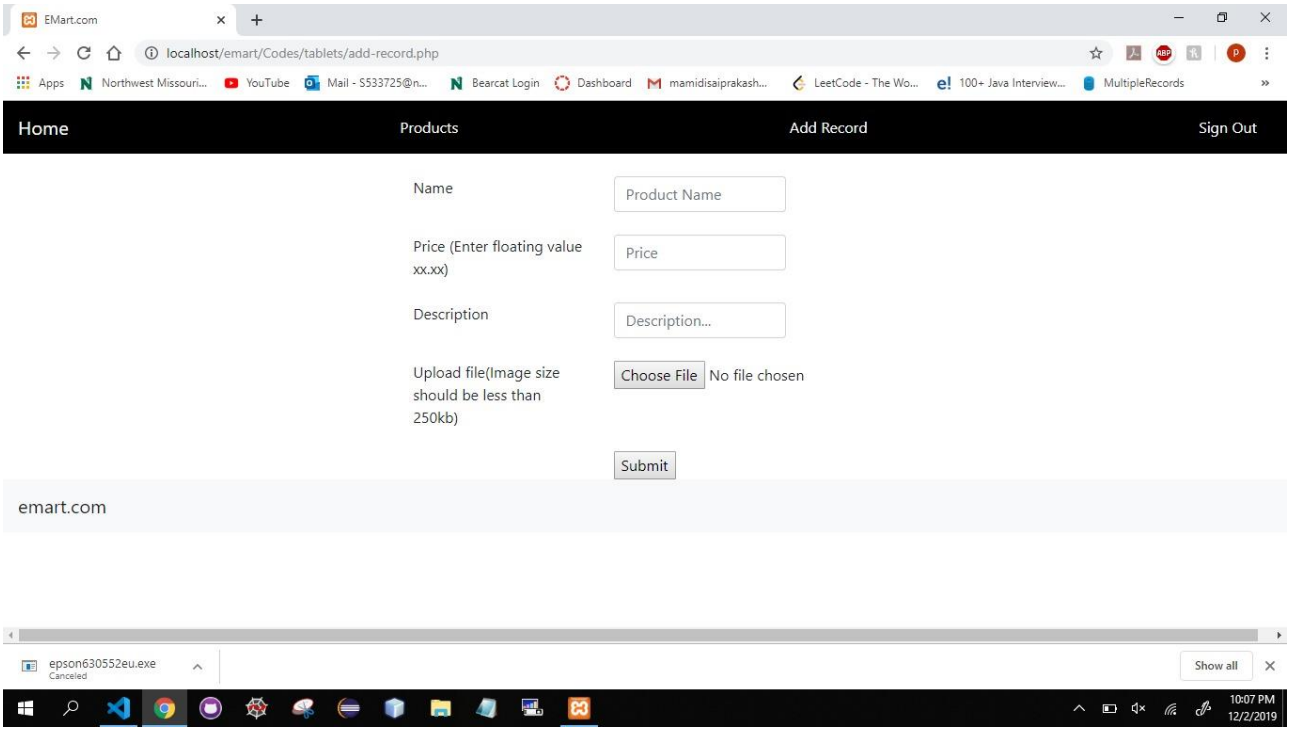
Payment1.php: https://github.com/prakashreddy-97/EMart/blob/master/Codes/payment1.php



Admin add product page:

This page consists of several text fields to enter the details of the product such as name of the product, price, and description for the product and an option to upload the image of the product in jpg, jpeg or png format. By adding the products in admin page, the product should also be inserted into the mysql table. There are several queries to perform this action. In fact, the products added here will also be reflected in the user's home page.

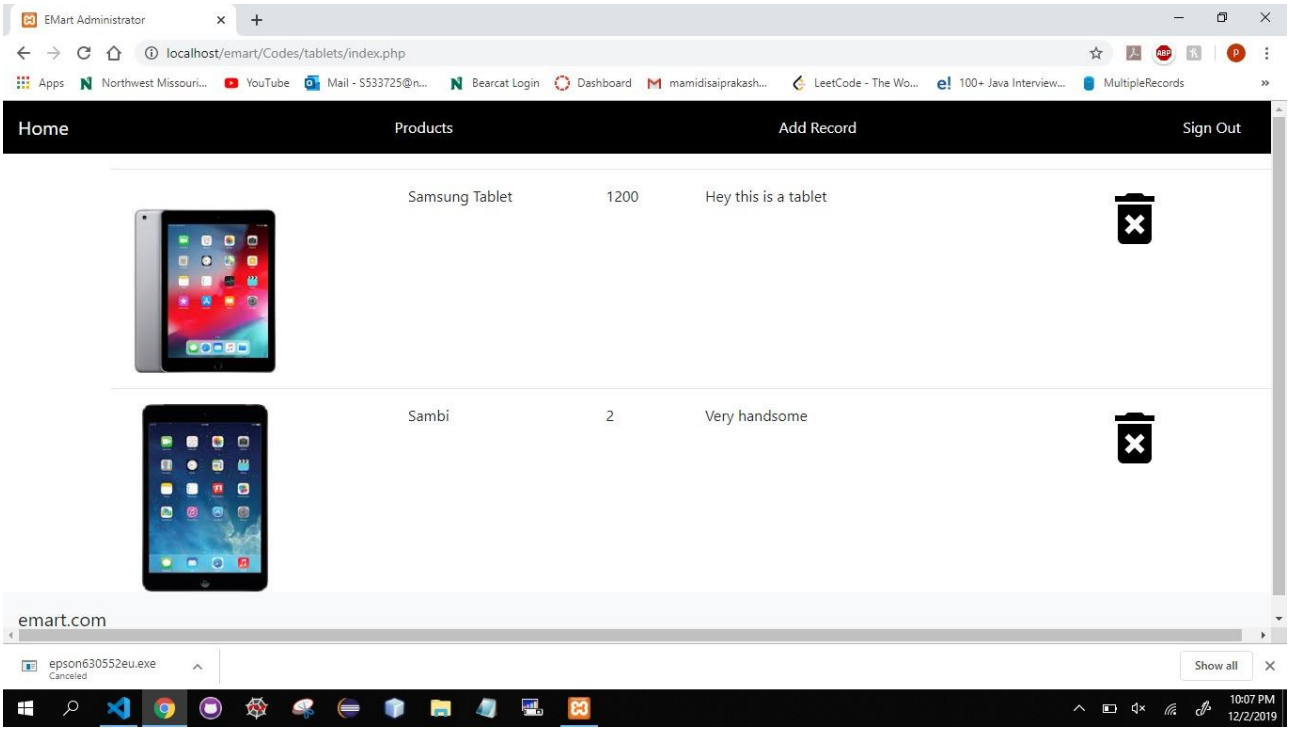
Add-record.php: https://github.com/prakashreddy-97/EMart/blob/master/Codes/laptops/add-record.php



View/Delete Product page:

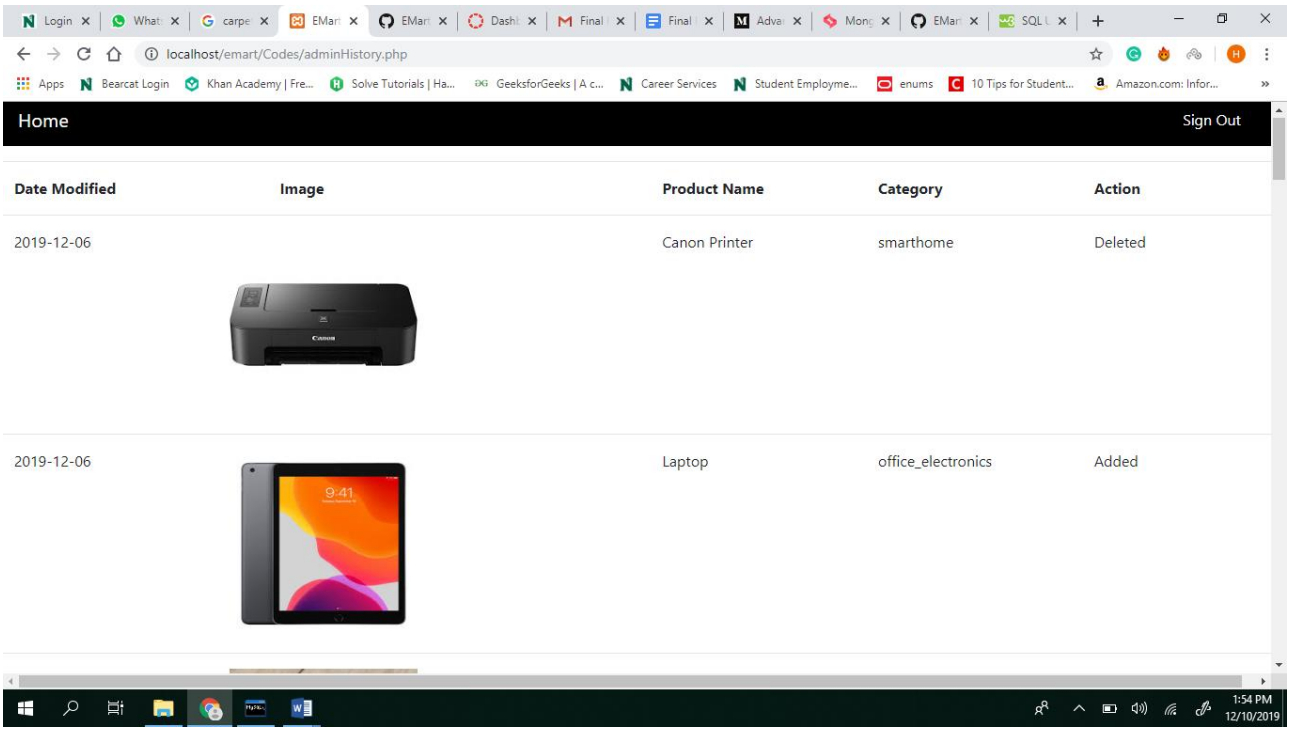
The product that are inserted in the add product pages are pulled and displayed in this page with an option to delete. So by clicking on the delete button will result in permanent deletion of the product from the page as well as from the database too. The image along with price and description of the product can be viewed here.

Delete-Record:https://github.com/prakashreddy-97/EMart/blob/master/Codes/laptops/delete-record.php



Admin history page:

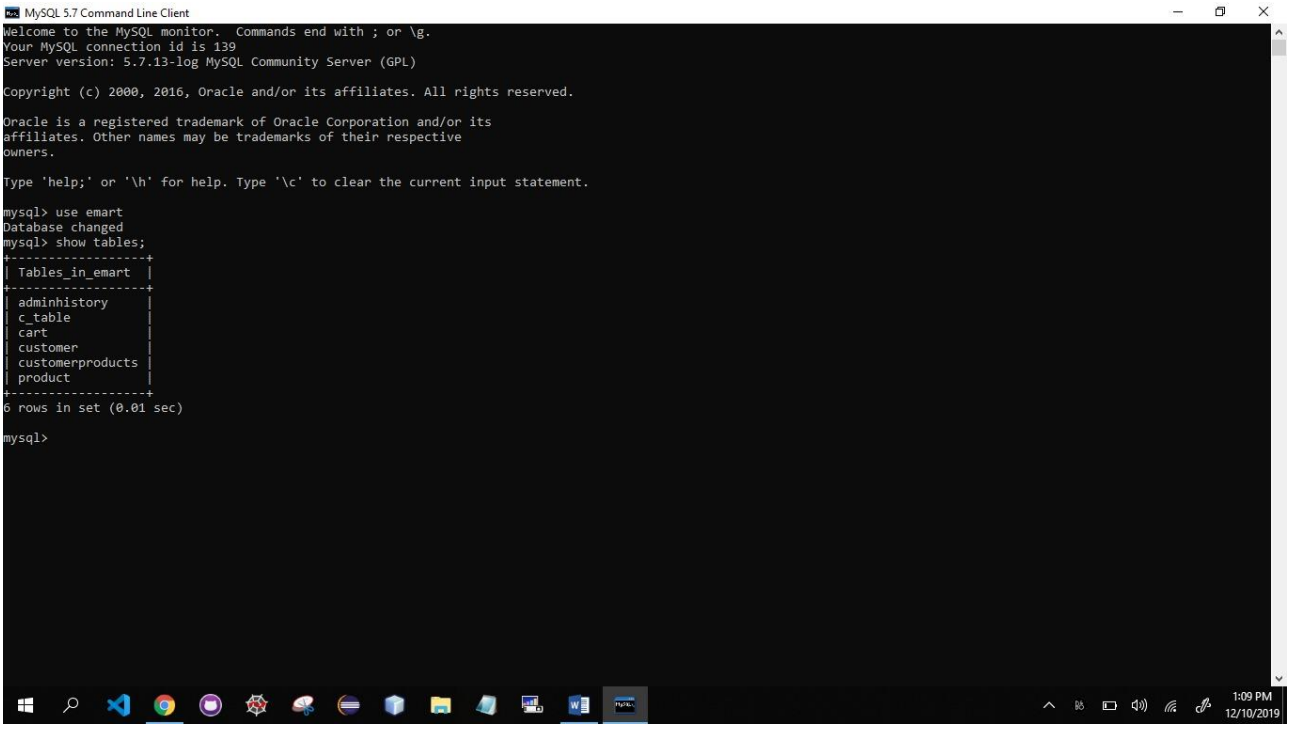
The actions of the admin are stored and displayed in this page. Every action such adding, deleting or updating will be saved and displayed in this page. There is an extra column with the name action which help to represent which type of action was performed by the admin.



Admin History: https://github.com/prakashreddy-97/EMart/blob/master/Codes/adminHistory.php

Back-end Implementation:

As per the ER Diagram, A database has been created with the name ‘Emart’. For now, this database consists of five table with the name customer, c_table, customer products, products, cart, and admin history. The details of the user are stored in this table. This table consists of the users first name, last name, phone number, address, city, state, zip, payment details, password (in encrypted format), a verification key (in encrypted format) and admin access as its attributes. So, whenever a person clicks on the register button in the front end. The details provided are supposed to be stored in this table. Similarly, there are several other table to store the details of the products and the admin details and the product he added and deleted to maintain the admin history.



Middleware Implementation:

There are several folders in the repository with .php as extension. The website has PHP as a connection between the front end and the backend.

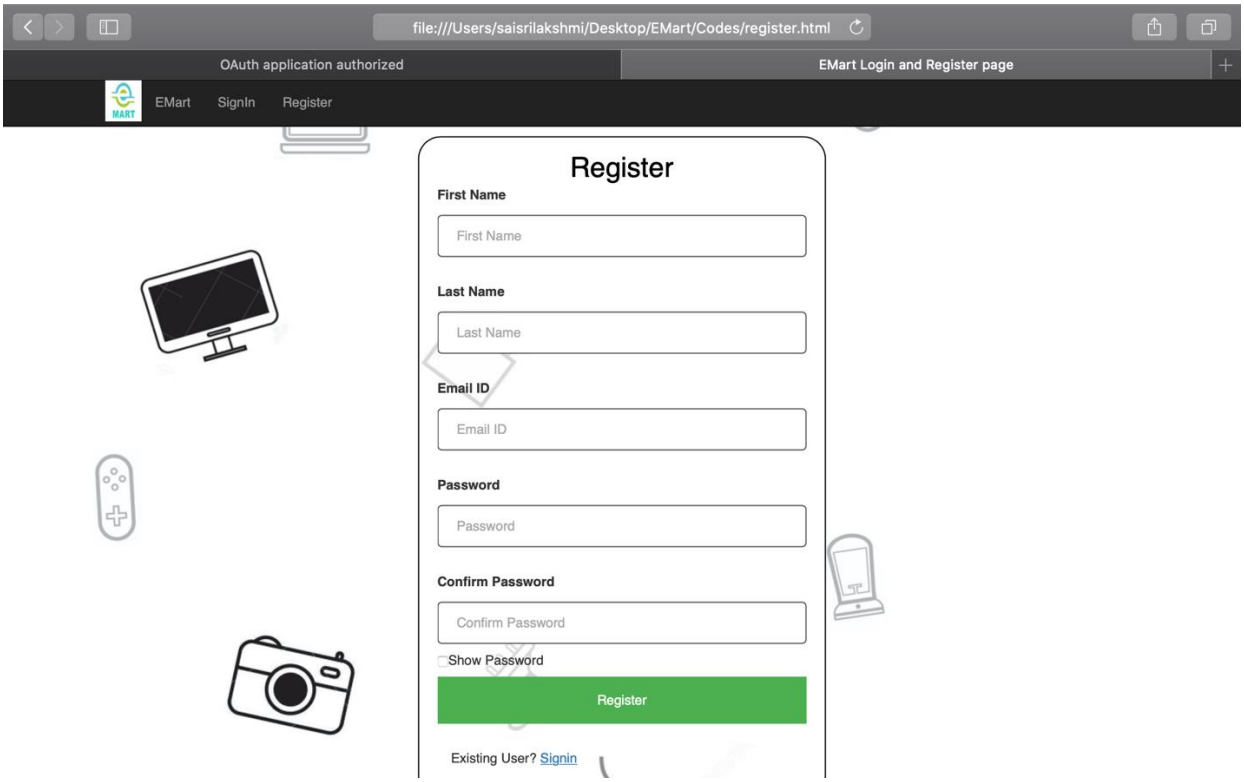
register.php: https://github.com/prakashreddy-97/EMart/blob/master/Codes/registration.php

connect.php: https://github.com/prakashreddy-97/EMart/blob/master/Codes/connect.php

Register.php helps to establish a connection between the register.html page and the EMart database. Similarly, connect.php helps to retrieve the stored data during the process of login.

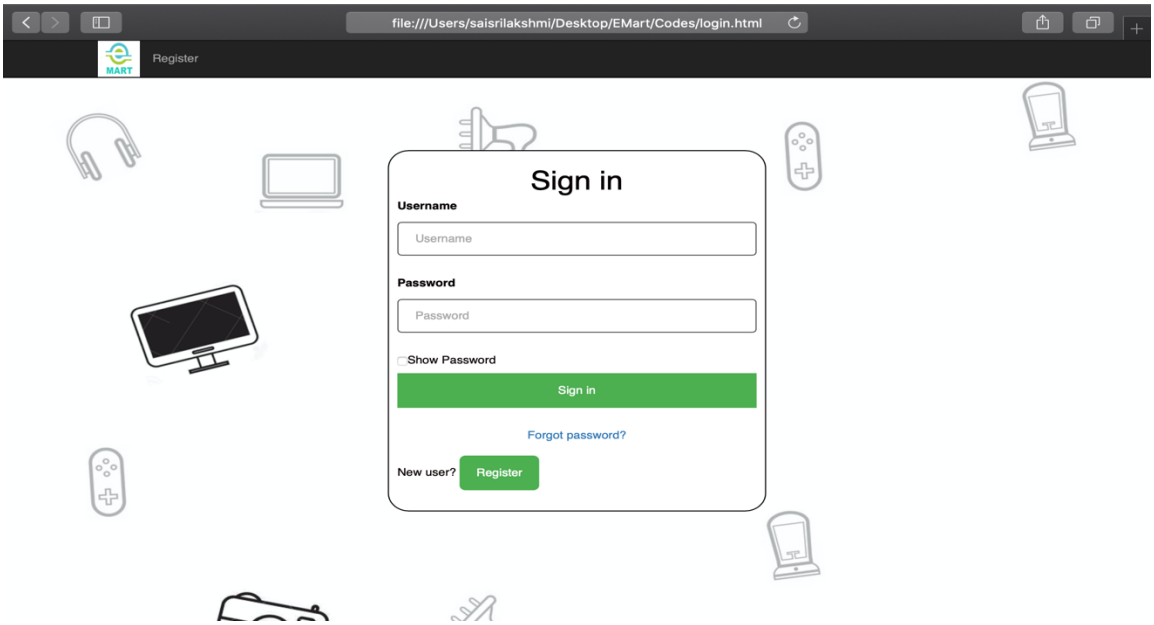
7. User/Admin Manual

7.1 Register Page:

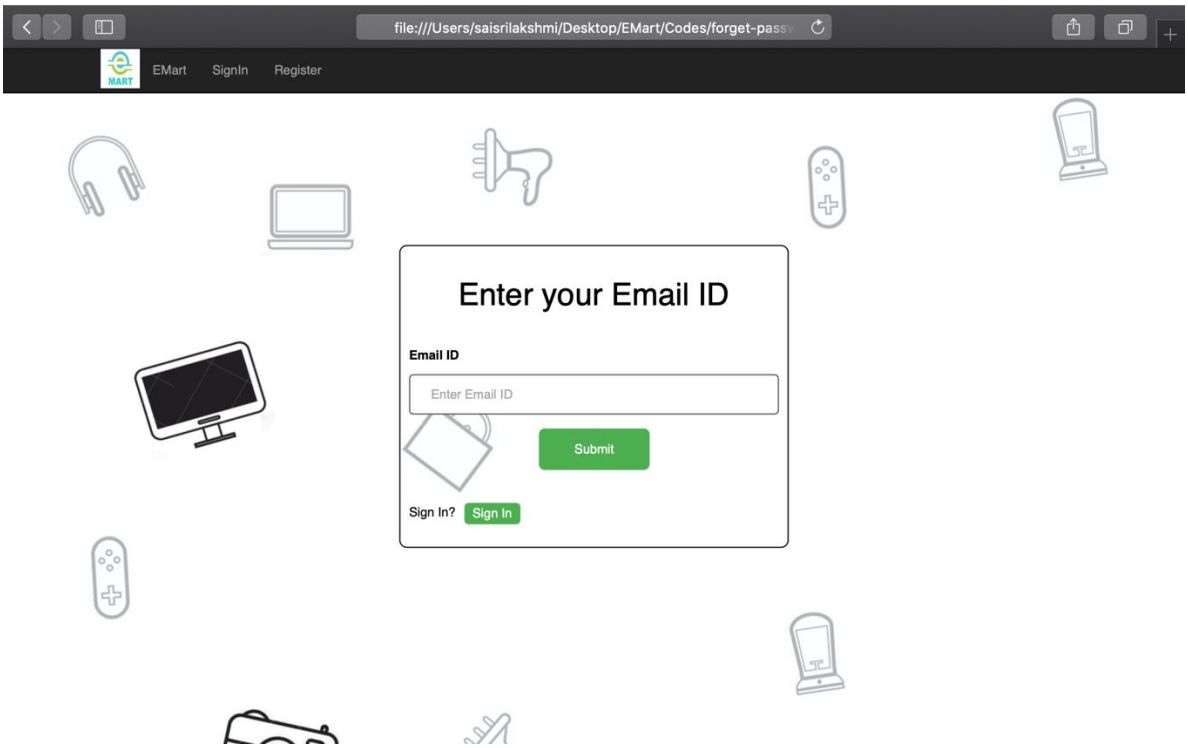


Firstly we have to register in order to login to the sight, to register enter the given fields with a valid email ID and password, after registering a confirmation pin will be sent to the given email ID, confirm the pin if it is correct a confirmation message will be given then you are ready to login.

7.2 Login Page:

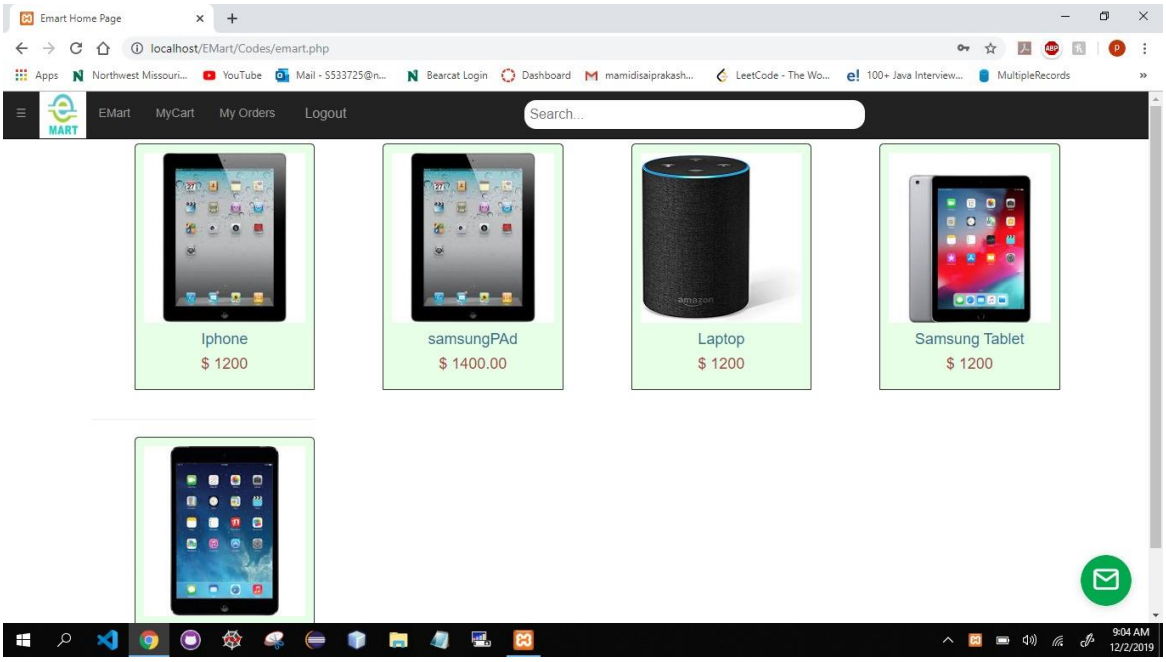


To Sign In enter the registered email and password, if you forgot the password click on the forgot password you will be redirected to a page where you have to enter the registered email to reset the new password. If you are a new user go and register.



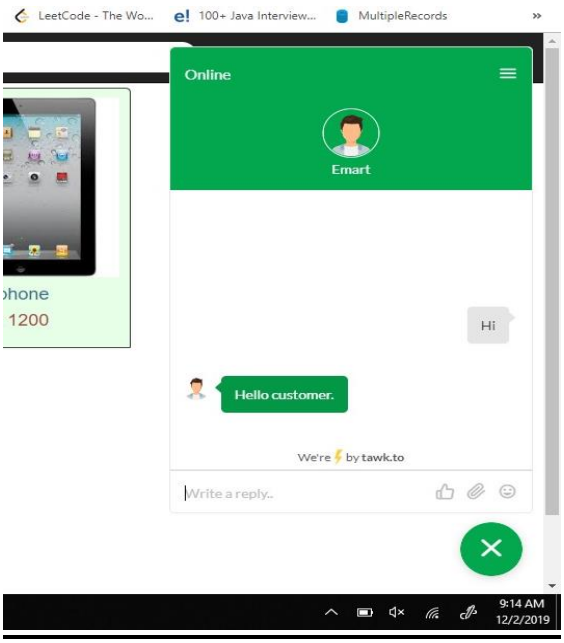
7.3 Home Page:

In home page the user can click on the desired object or device and checkit out, if he cannot find any one particular device he can search for it. The burger option shows the categosries of the electronic devices, Emart takes user to the home page, My cart will have the devices user selected to buy, My orders is the page where user can check his previous orders, Logout is the option where the user can logout.



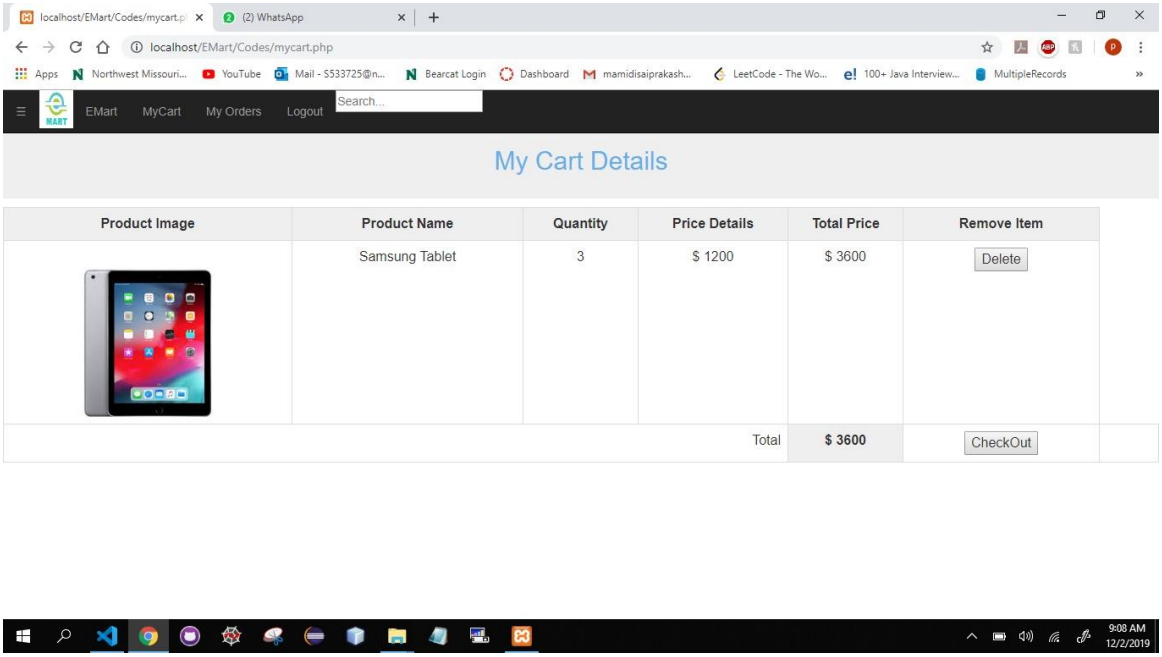
7.4 Chat Box:

In the home page we are having a chat option where the user can chat with the customer support if he has any queries to address.



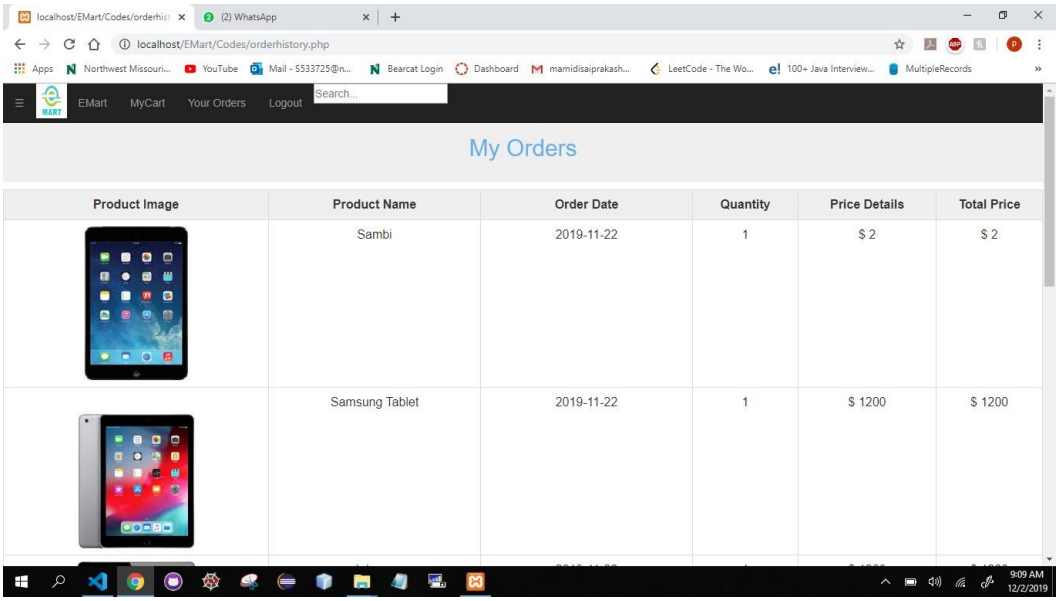
7.5 My Cart Page:

My cart is where the user can find the devices that he wished to buy, the user can delete the order if the user does not want to buy it anymore or the user can go to checkout to pay for the order.



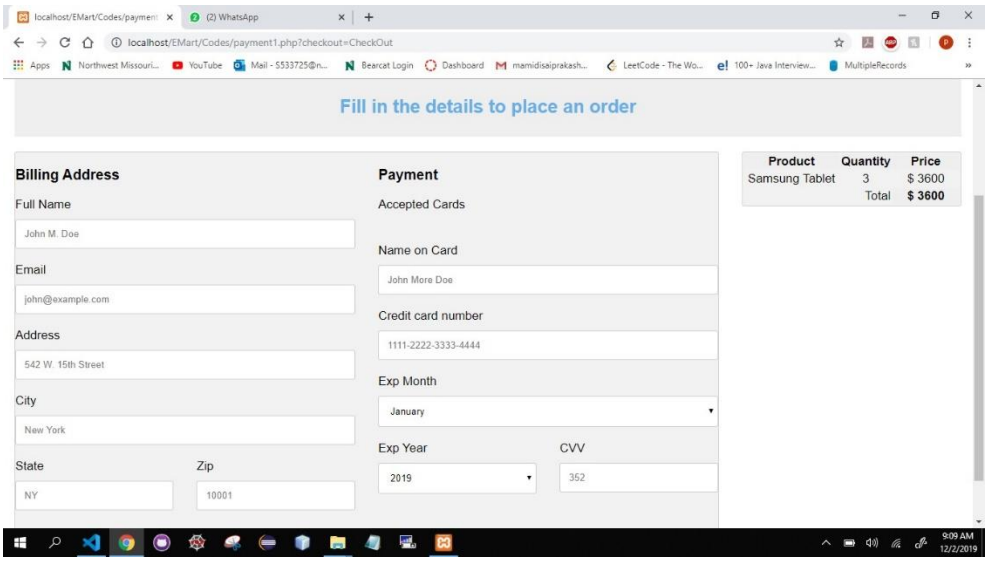
7.6 My Orders:

My orders is the page where the user can check his privious orders, it has the product name, image, oreder date, quantity, price details and total price. Here the user cannot change or edit because these orders are already placed.



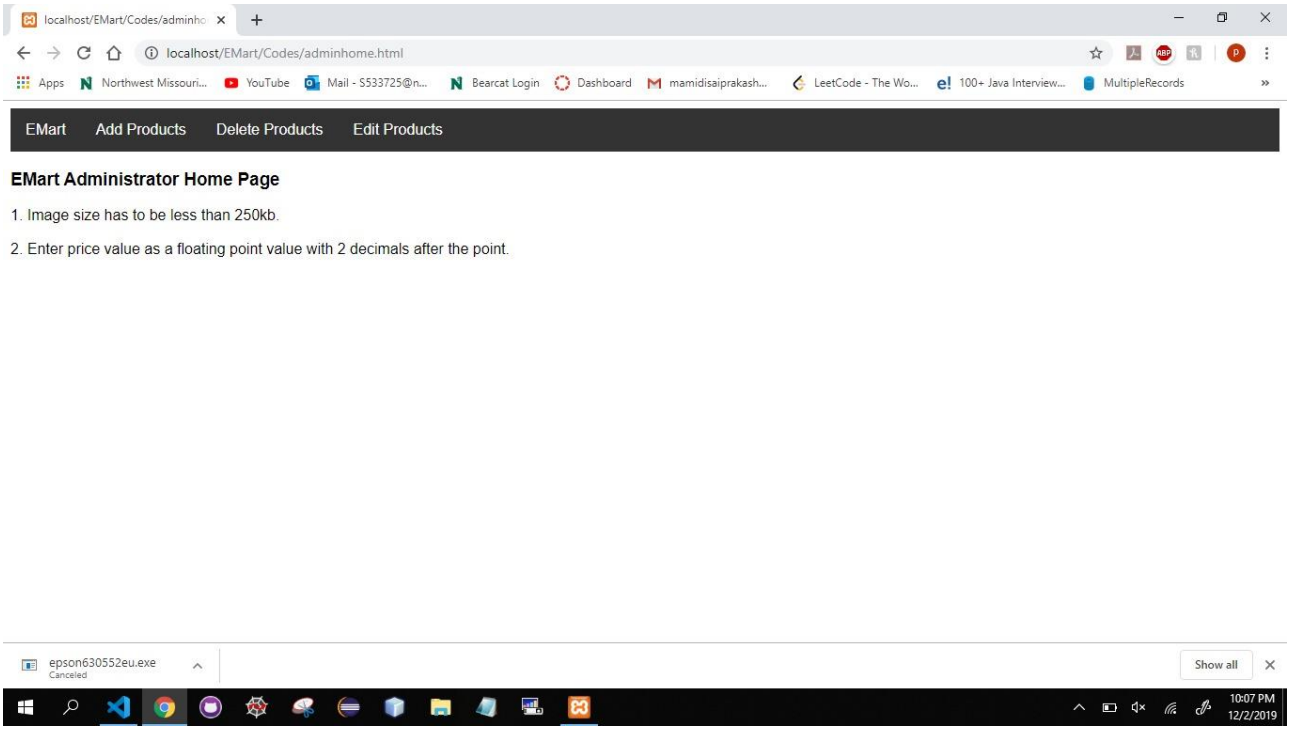
7.7 Payment page:

The payment page is redirected from the My cart page. Here the user will give his details for the address with full name, email, address, city, state, Zip. Payment details should be given with Name on Card, credit card number, expeiration month, expiration year and CVV. Then click the pay button to pay for the orders, if the payment is successful a messgae will be displayed.



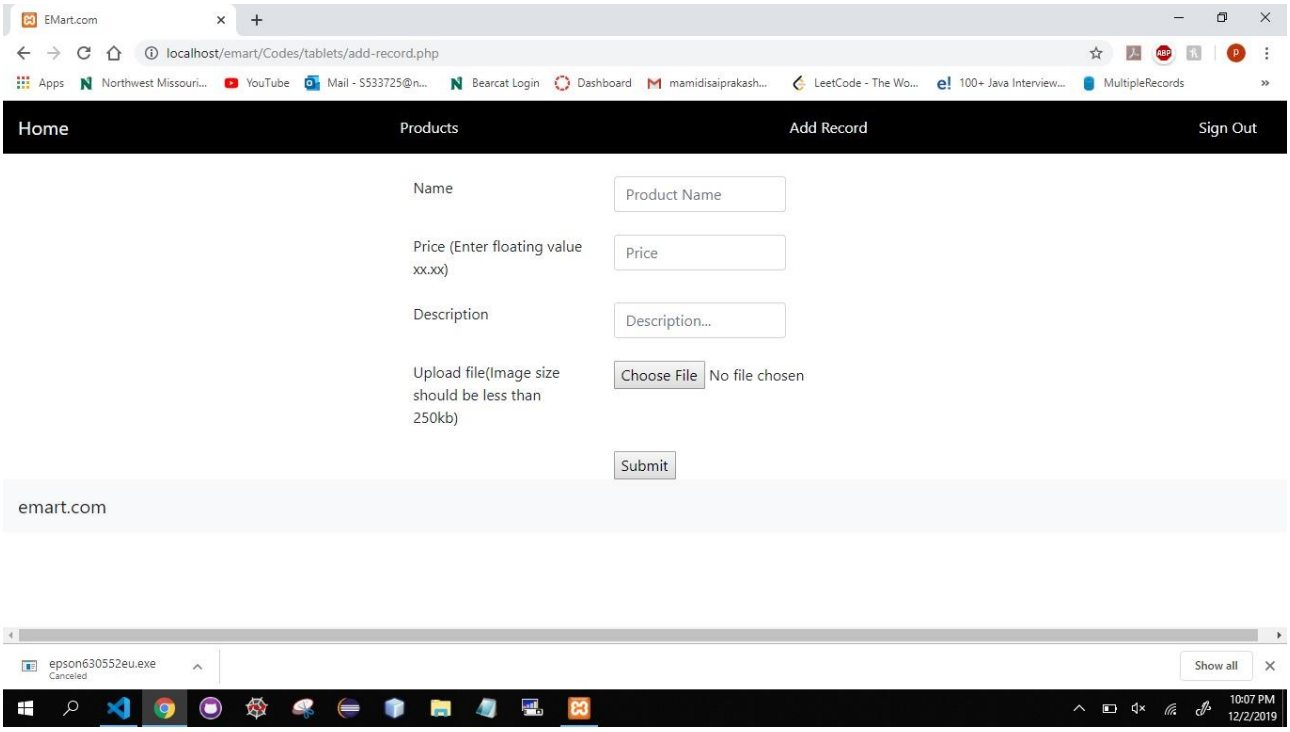
7.8 Admin Home page:

The admin will be provided with some instructions on the home screen such as the requirements for the image and the details about price. The admins can follow these instructions while adding a product.



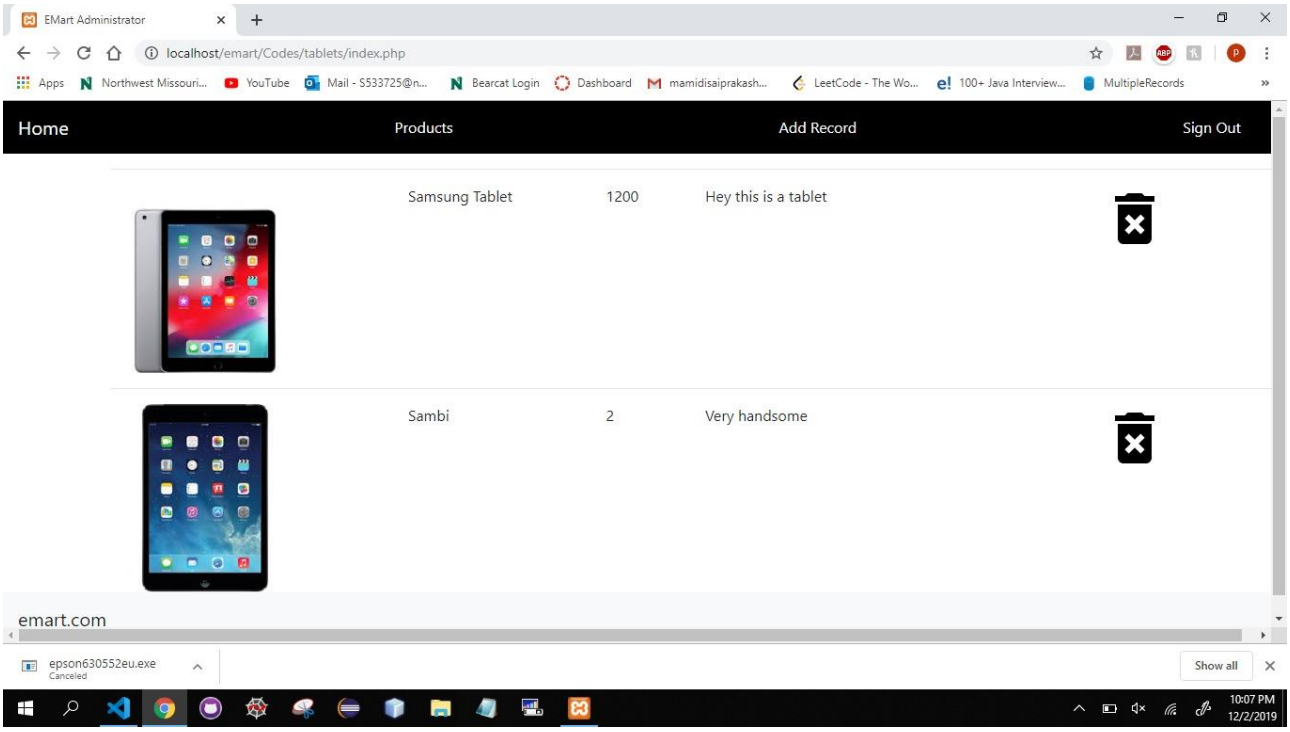
7.9 Add product page:

The add product page consists of several fields such as name, price, description about the product and an option to upload. The admin can upload the image from his personal device on to the website by clicking on the button upload file. The file format is supposed to be a .jpg or .jpeg or .png.



7.10 View/Delete product:

The admin will have the flexibility to delete the product or view the product that they have added from the add product tab. The admin can delete the product by hitting the delete button that is present on the right corner of each product.


EMart Administrator
localhost/emart/Codes/tablets/index.php
Apps
Northwest Missouri...
YouTube
Mail - S533725@n...
Bearcat Login
Dashboard
mamidisaiprakash...
LeetCode - The Wo...
100+ Java Interview...
MultipleRecords
Home
Products
Add Record
Sign Out
Samsung Tablet
1200
Hey this is a tablet
Sambi
2
Very handsome
emart.com
epson630552eu.exe
Canceled
Show all
10:07 PM
12/2/2019