# ASSIGNMENT COVER SHEET

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Course: BSc Computing (GC)** | | | **Year: 2** | CSY2027 | | |
| **Group Project** | **Title:** | | | | | |
| Deadline for submission:  23:59 on May 3rd, 2020 | | Extension agreed by: | | | | Extension date: |
| **Student Names (List each member of the group)**   1. **Huynh Ngoc Duc** 2. **Din** 3. **Popinder Singh** 4. **Akram Ahmed** | | | | | **Tutor:** Dr. James Xue | |

**ASSESSMENT FEEDBACK:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **RATING SCALE** | **Excellent** | | **Good** | **Satisfactory** | **Needs some more work** | | **Needs much more work** |
| Problem Domain and Literature Research (10%) |  | |  |  |  | |  |
| Requirement Gathering and Specification (10%) |  | |  |  |  | |  |
| System Design (10%) |  | |  |  |  | |  |
| Implementation & Quality of code (20%) |  | |  |  |  | |  |
| Testing & Evaluation  (10%) |  | |  |  |  | |  |
| Overall Report Quality (20%) |  | |  |  |  | |  |
| Teamwork and Project Management (10%) |  | |  |  |  | |  |
| Presentation (10%) |  | |  |  |  | |  |
| Specific aspects of your assignment that the marker likes: | | | | Specific aspects of your assignment that need more work: | | | |
| Tutor’s Signature: | | Date: | | | | Grade: | |

|  |  |
| --- | --- |
| 1. **Introduction**     1. Aims and objectives    2. Project Background |  |
| 1. **Requirements Analysis and System Design** |  |
| 1. **Design of the Proposed System**   3.1) Database Design  3.2) User Interface Design  3.2.1) Main Webpage  3.2.2) Product detail page  3.3) System Interaction Design |  |
| 1. **Prototype Implementation**   4.1) Programming Language/tool and Framework  4.1.1) XAMPP tool  4.1.2) PHP programming language  4.2) Software Architecture  4.2.1) Brief  4.2.2) The backend  4.2.2.1) MVC model  4.2.2.2) Model  4.2.2.3) View  4.2.2.4) Controller    4.2.3) The frontend |  |

|  |  |
| --- | --- |
| **Contents** | **Page Numbers** |

1. **INTRODUCTION**
   1. **Aims and Objectives**

We were approached by a company to design and develop a web application to sell their products online. So, we are a group suggested them for an e-commerce website that sells a range of products from shoes to electronics. Different software’s were used to develop the website. Windows OS was used as a base Operating System. Vagrant VM was used as a local host. Sublime text will be used as a code editor. For sharing of code GitHub was used. It’s a fully functional e-commerce website which allows the customers to view the products, register to the website, login to the website, select their desired product and add to cart. After they have chosen their desired product, they can checkout which will lead them to the payment portal (PayPal).

* 1. **Project Background**

Before we started designing the website, we did an extensive research by looking at different website such as amazon.com, ebay.com etc. we carefully looked at the layout of the website, the user-friendliness of the website, the layout of the website. Different features of the website were also carefully researched. The most unique feature of amazon was product variation, the variety of products from a-z (as the logo itself suggests). So, the same feature was implemented on the website, the customer can search for the product and choose the product. Certain features were difficult to implement so we didn’t implement that such as filtering the product.

1. **Requirements Analysis and System Design**

First different websites were compared like amazon.com and ebay.com, they were carefully analyzed and then some features from it were implemented in the website which was required by the client. Idea of login and signup page was from different websites. Almost all the pages of the website are static webpages. The login and signup pages have buttons like the “login” and “not a member? Sign up” after the user sign in they can login into the website with a button. There is a home button on the navigation bar which shows the home page of the website if you are on another page of the website. There is also a cart button which takes the user to the cart. Some of the non-functional features include different colors of the login, signup page etc. the main aim of this project is to create an e commerce website which has good functionalities. The website is made according to the client’s specifications.

1. **Design of the Proposed System**

The database of the e-commerce website has 8 tables namely orders, cart, cart\_item, users, products, category\_link\_products, images and category.

A close up of text on a white background

Description automatically generated

By using MySQL Workbench, the above class diagram can be generated (by importing the SQL file)

**3.1) Database Design**

The table specifications of all the tables is given below

* **Products table:**

|  |  |  |  |
| --- | --- | --- | --- |
| ATTRIBUTE | DATATYPE | CONSTRAINT | DEFAULT |
| products | | | |
| id | INT(32) | Pk, NOT NULL | AUTO\_INCREMENT |
| title | VARCHAR(100) |  |  |
| description | LONGTEXT |  |  |
| price | DECIMAL(5,2) |  |  |
| image | VARCHAR(2083) |  |  |
| status | ENUM() | CHECK IN (Available, Out of stock, Coming soon, Unavailable) |  |
| rate | ENUM() | CHECK IN (1,2,3,4,5) |  |

* **Categories:**

|  |  |  |  |
| --- | --- | --- | --- |
| ATTRIBUTE | DATATYPE | CONSTRAINT | DEFAULT |
| category | | | |
| id | INT(32) | Pk, NOT NULL | AUTO\_INCREMENT |
| name | VARCHAR(30) |  |  |
| description | LONGTEXT |  |  |

* **Users table:**

|  |  |  |  |
| --- | --- | --- | --- |
| ATTRIBUTE | DATATYPE | CONSTRAINT | DEFAULT |
| users | | | |
| user\_id | INT(32) | Pk, NOT NULL | AUTO\_INCREMENT |
| username | VARCHAR(54) |  |  |
| password | CHAR(60) |  |  |
| name | VARCHAR(30) |  |  |
| status | ENUM() | CHECK IN (Waiting ,Active ,Disable ,Lock ) |  |
| date\_create | DATETIME |  | current\_timestamp() |
| type | ENUM() | CHECK IN (admin, user) |  |

* **Images:**

|  |  |  |  |
| --- | --- | --- | --- |
| ATTRIBUTE | DATATYPE | CONSTRAINT | DEFAULT |
| images | | | |
| images\_id | INT (11) | Pk, NOT NULL | AUTO\_INCREMENT |
| product\_id | INT(32) | Fk\_product |  |
| type | ENUM() | CHECK IN (header, thumbnail, slide) |  |
| name | CHAR(50) |  |  |
| date\_create | DATETIME |  | current\_timestamp() |

* **Orders:**

|  |  |  |  |
| --- | --- | --- | --- |
| ATTRIBUTE | DATATYPE | CONSTRAINT | DEFAULT |
| orders | | | |
| order\_id | INT (32) | Pk, NOT NULL | AUTO\_INCREMENT |
| user\_id | INT(32) | Fk\_user, NOT NULL |  |
| address | VARCHAR(60) |  |  |
| phone | CHAR(13) |  |  |
| email | VARCHAR(320) |  |  |
| status | VARCHAR(32) |  |  |
| paymentID | CHAR(50) |  |  |

* **Categories\_link\_products** (for convenient during code):

|  |  |  |  |
| --- | --- | --- | --- |
| ATTRIBUTE | DATATYPE | CONSTRAINT | DEFAULT |
| categories\_link\_products | | | |
| id | INT(32) | Pk, NOT NULL | AUTO\_INCREMENT |
| product\_id | INT(32) | Fk\_product, NOT NULL |  |
| category\_id | INT(32) | Fk\_category, NOT NULL |  |
| category\_name | VARCHAR(100) |  |  |
| product\_title | VARCHAR(100) |  |  |

* **Cart table:**

|  |  |  |  |
| --- | --- | --- | --- |
| ATTRIBUTE | DATATYPE | CONSTRAINT | DEFAULT |
| cart | | | |
| cart\_id | INT(32) | Pk, NOT NULL | AUTO\_INCREMENT |
| user\_id | INT(32) | Fk\_user |  |
| order\_id | INT(32) | Fk\_orders |  |

* **Cart\_item table:**

|  |  |  |  |
| --- | --- | --- | --- |
| ATTRIBUTE | DATATYPE | CONSTRAINT | DEFAULT |
| cart\_item | | | |
| cart\_id | INT(32) | Fk,\_cart NOT NULL | AUTO\_INCREMENT |
| product\_id | INT(32) |  |  |
| quantity | INT(32) |  |  |

**3.2) User Interface Design**

The GUI of the website should be simple so that it is easy for the user to navigate through the website. The user interface for the website needs to be user-friendly and mobile friendly so that the user can access it using both mobile and PC.

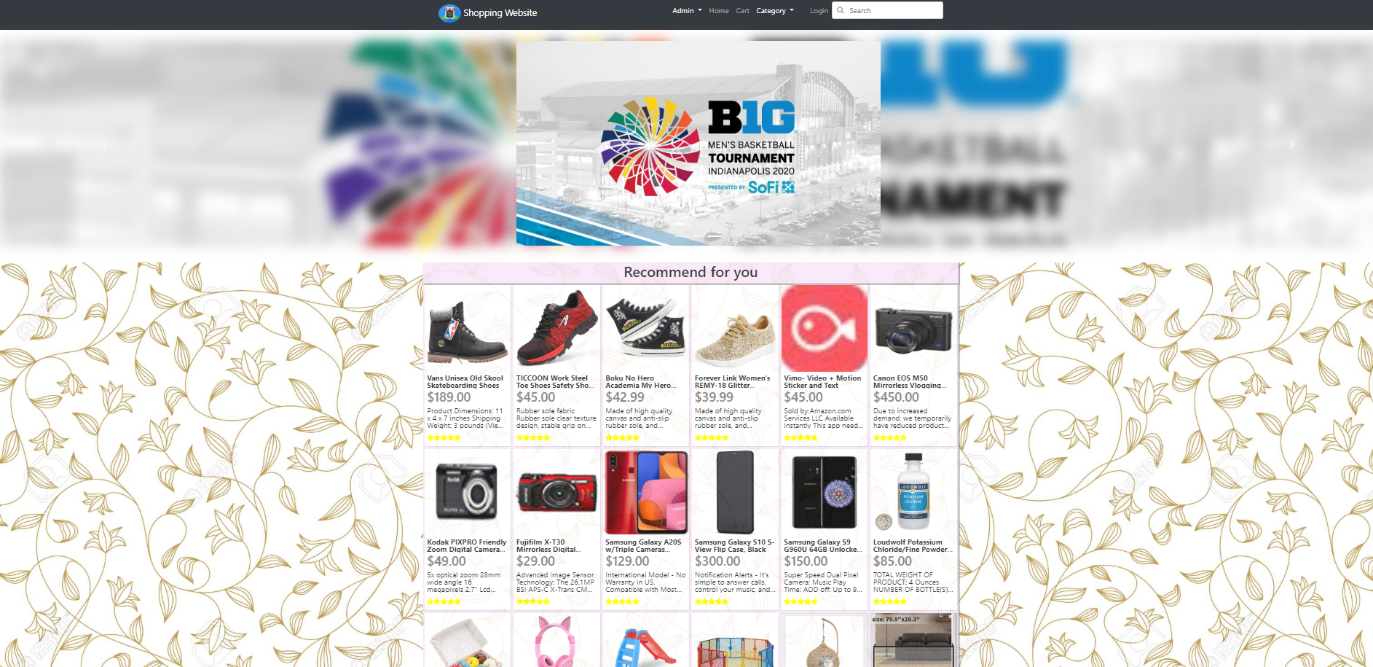
For this project the front end of the layout of the website is something similar to Amazon, eBay, Lazada etc.

**3.2.1) Main Webpage**

The main Webpage of the website is the homepage of the website. As soon as the user enters the URL the user sees the main webpage. This page contains list of recently added products. The user can search for the desired product. Each product can have more than one category and each category can be added by the Admin.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | Home | Your Cart | Okay | Login |   **Navigator Bar** | | | | |
| 🢠 | | **Banner** | 🢡 | |
|  | Latest Listings/ Search Results/ Category listing | | |  |
| |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | Product image | Product image | Product image | Product image | Product image | Product image | | Description | Description | Description | Description | Description | Description | | | |
| Footer | | | | |

The above diagram is the layout of the website, this is we page:



**3.2.2)** **Product Detail Information(Modal)**

When the user clicks on any of the products displayed on the homepage, the floating window called modal will appear. Basically, a modal is a dialogue box/popup window that is displayed on the current page (modal is one of the plugins of bootstrap).

This is how a modal looks

|  |  |
| --- | --- |
| **Product title** | |
| Product Image | Description |
| Description Text area |
|  | |

This is a sample from the website, this is how a modal look

A screenshot of a cell phone

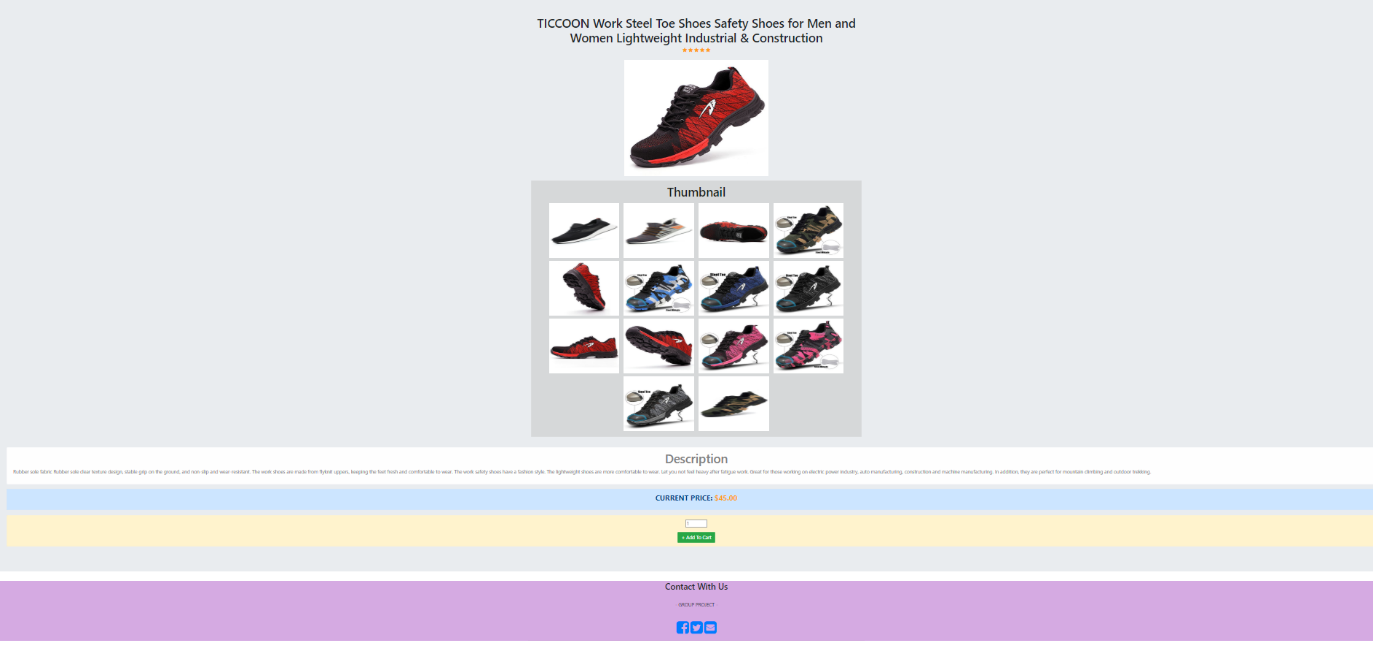
Description automatically generated

**3.2.2)** **Product Detail Page**

The product detail page can be accessed by clicking Show Detail button that is on the modal window

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | Home | Your Cart | Okay | Login |   **Navigator Bar** | | |
|  | **Banner** |  |
| Footer | | |

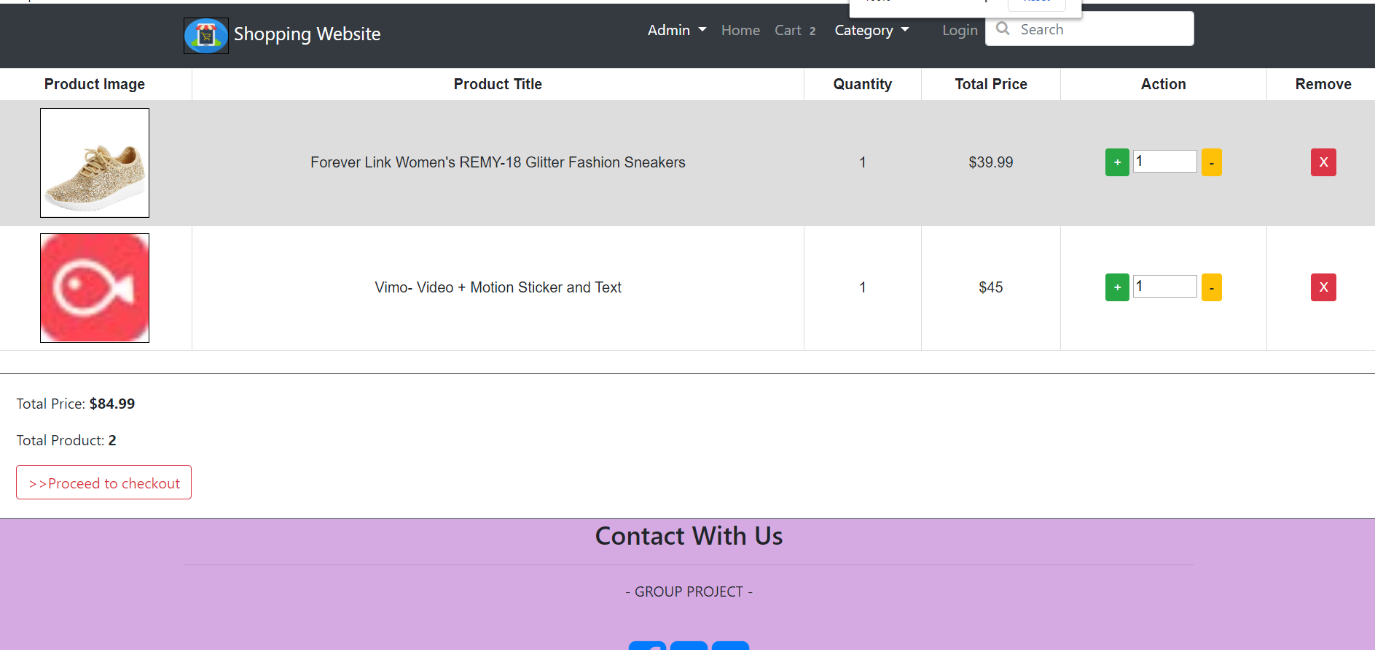
This is detail product page:



**3.2.3)** **Customer Cart**

This is how customer cart looks

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | Home | Your Cart | Okay | Login |   **Navigator Bar** | | |
|  | Latest Listings/ Search Results/ Category listing |  |
| |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | Product image | Product image | Product image | Product image | Product image | Product image | | Description | Description | Description | Description | Description | Description | |
| Footer | | |



**3.2.4) Cart Checkout**

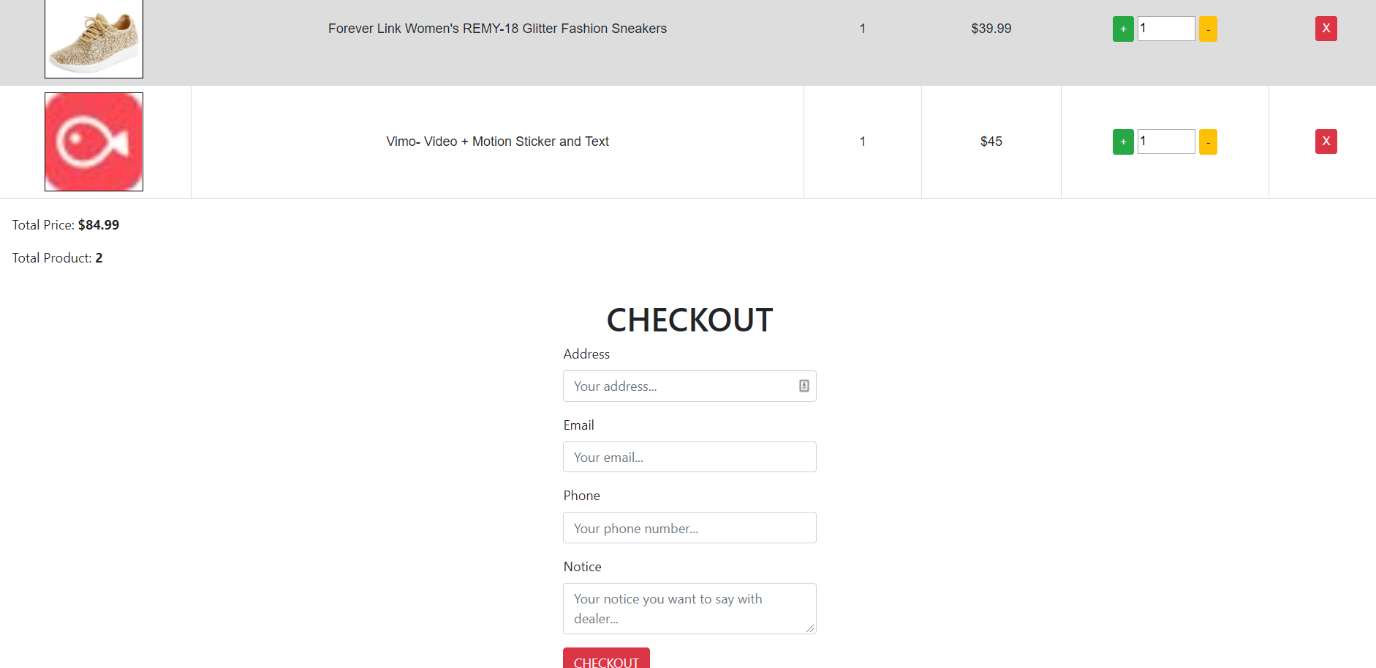
There are two layouts for the cart checkout

Layout 1: -

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | **Checkout** | | | | | | | First name: | |  | | --- | |  | |  |  | Last name: | |  | | --- | |  | | | Notice: | |  | | --- | |  | |  |  |  |  | | Address: | |  | | --- | |  | | Phone: | |  | | --- | |  | | Email: | |  | | --- | |  | | | |  | | --- | | Submit | | | | | | | | Please make sure you are logged in before making a payment!!! | | | | | | | Click here to log in | | | | | | |
| Footer |

Layout 2: -

|  |
| --- |
| **Check out** |
| Address |
| |  | | --- | | Your address… | |
| Email |
| |  | | --- | | Your email… | |
| Phone |
| |  | | --- | | Your phone number… | |
| Notice |
| |  | | --- | | Your notice you want to say with dealer… | |
| |  | | --- | | CHECKOUT | |

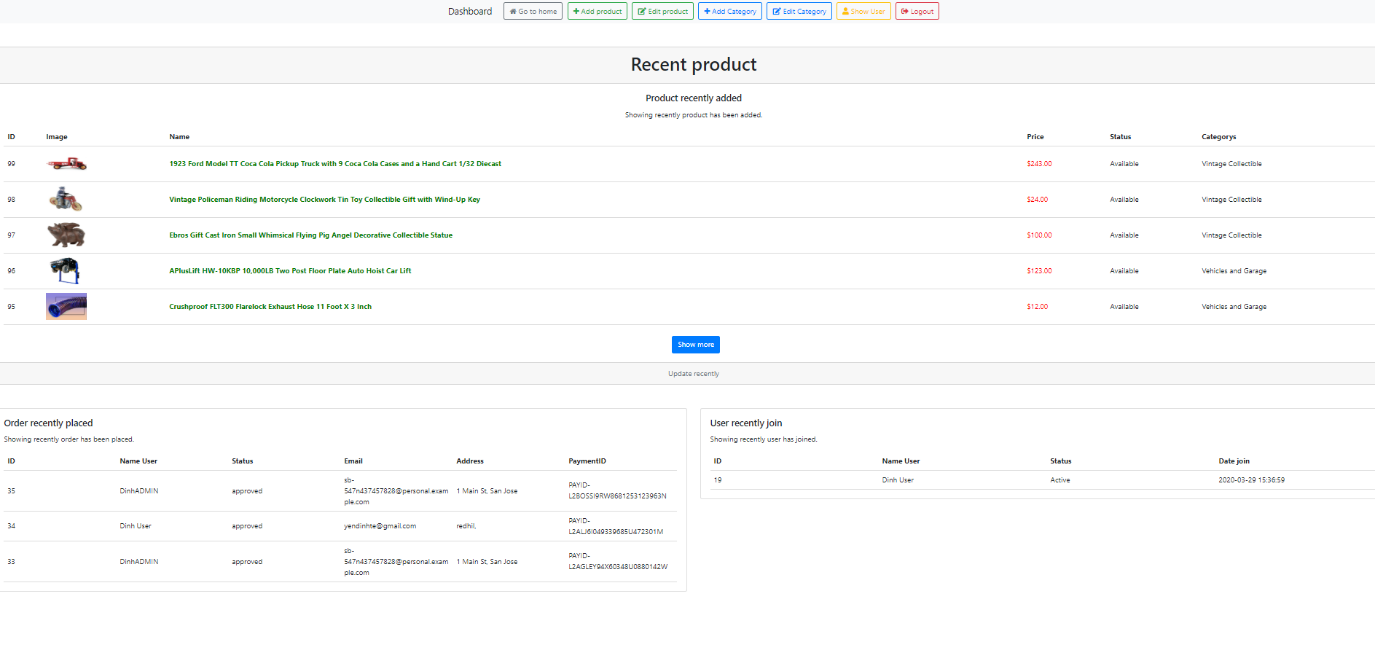


**3.2.5)** **Admin Page Layout**

The admin page can be accessed by the URL: **/WEBASSIGNMENT2/admin/dashboard**

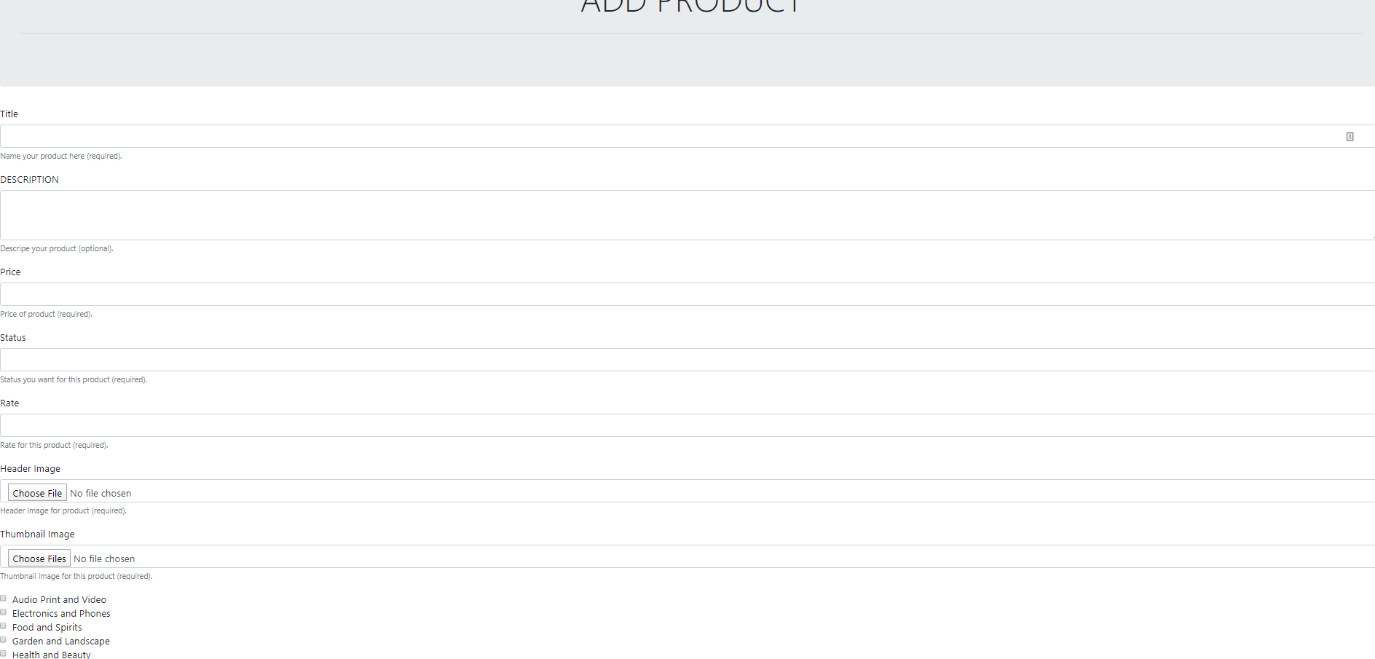
The admin has his own privileges. The admin page cannot be accessed by the customers. Only admins have the permission to modify the website.

|  |
| --- |
| **Navigator Bar** |
| **Page header (add product, edit category, etc)** |
| **Page body (Show content of the page)** |
| **Footer** |

****

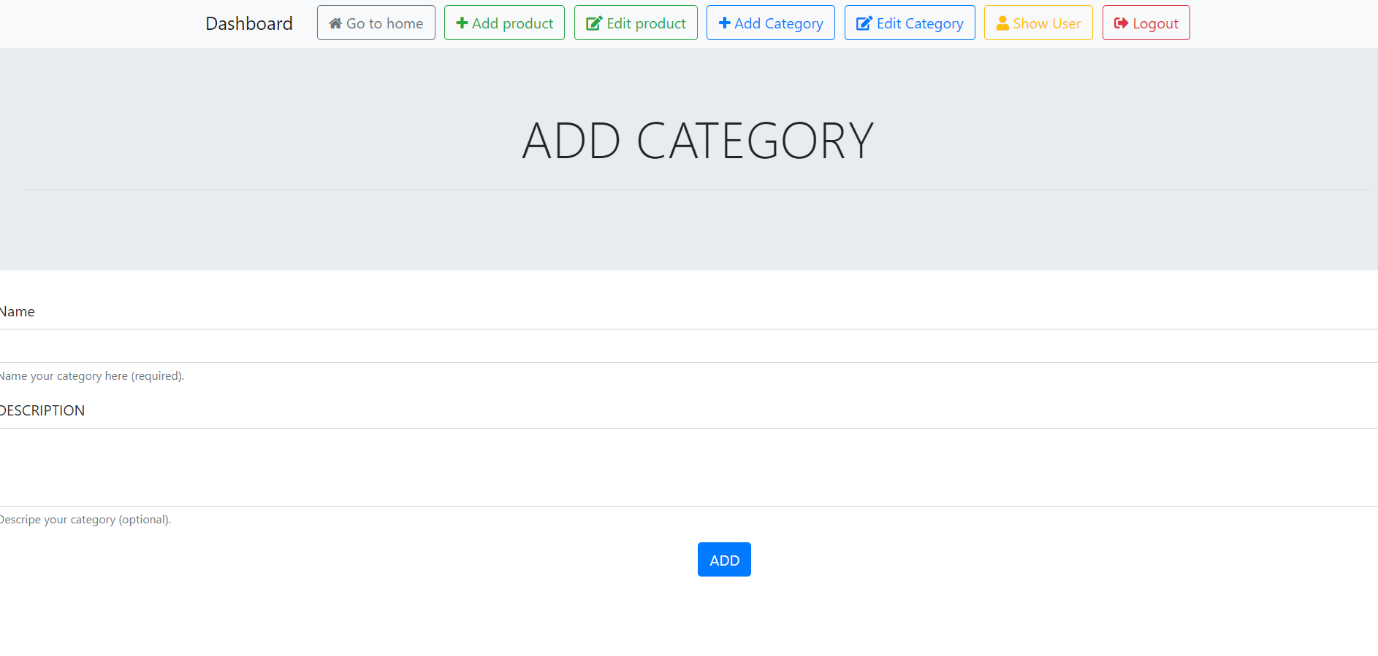
**3.2.6)** **Admin Add Product**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | Dashboard | |  | | --- | | Add product | | |  | | --- | | Edit product | | |  | | --- | | Add category | | |  | | --- | | Edit category | | |
| **ADD PRODUCT** |
| Product title:   |  | | --- | |  |   Name your product here (required). |
| Description:   |  | | --- | |  |   Describe your product (optional). |
| Price:   |  | | --- | |  |   Price of product (required). |
| Status:   |  | | --- | |  |   Status you want for this product (required). |
| Rate:   |  | | --- | |  |   Rate for this product (required). |
| Header image:   |  |  | | --- | --- | | |  | | --- | | Chose File |   No file chosen |   Header image for product here (required). |
| Thumbnail image:   |  |  | | --- | --- | | |  | | --- | | Chose File |   No file chosen |   Thumbnail image for this product (required). |
| Product category:  □ Sport  ☑Furniture  ☑Electronic  □ Electronics  □…  Please select at least 1 category (required). |
| |  | | --- | | ADD BUTTON | |
| Footer |



**3.2.7**) **Admin add category/edit category**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | Dashboard | |  | | --- | | Add product | | |  | | --- | | Edit product | | |  | | --- | | Add category | | |  | | --- | | Edit category | | |
| **ADD CATEGORY** |
| Category title:   |  | | --- | |  |   Name your category here (required). |
| Description:   |  | | --- | |  |   Describe your category (optional). |
| |  | | --- | | ADD BUTTON | |
| Footer |



**3.3)** **System Interaction Design**

The System Interaction design can be briefly described by a use case diagram. Use case diagrams summaries relationships between use cases, actors and systems.

Use case diagram of the website is given below

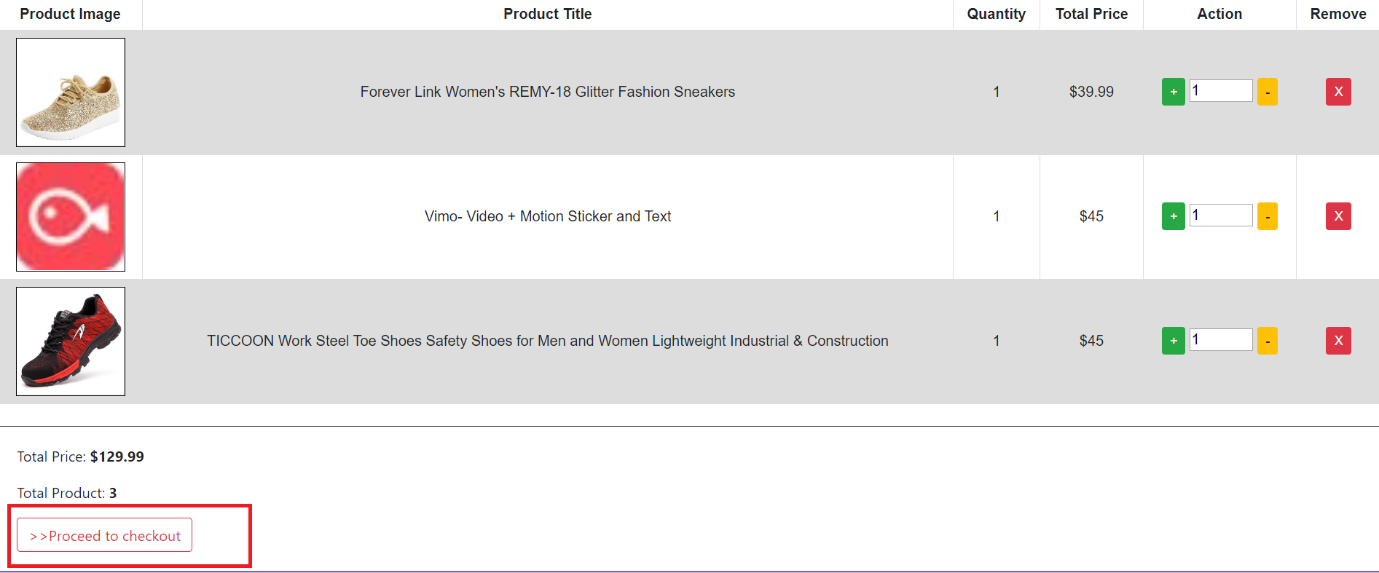
A close up of a logo

Description automatically generated

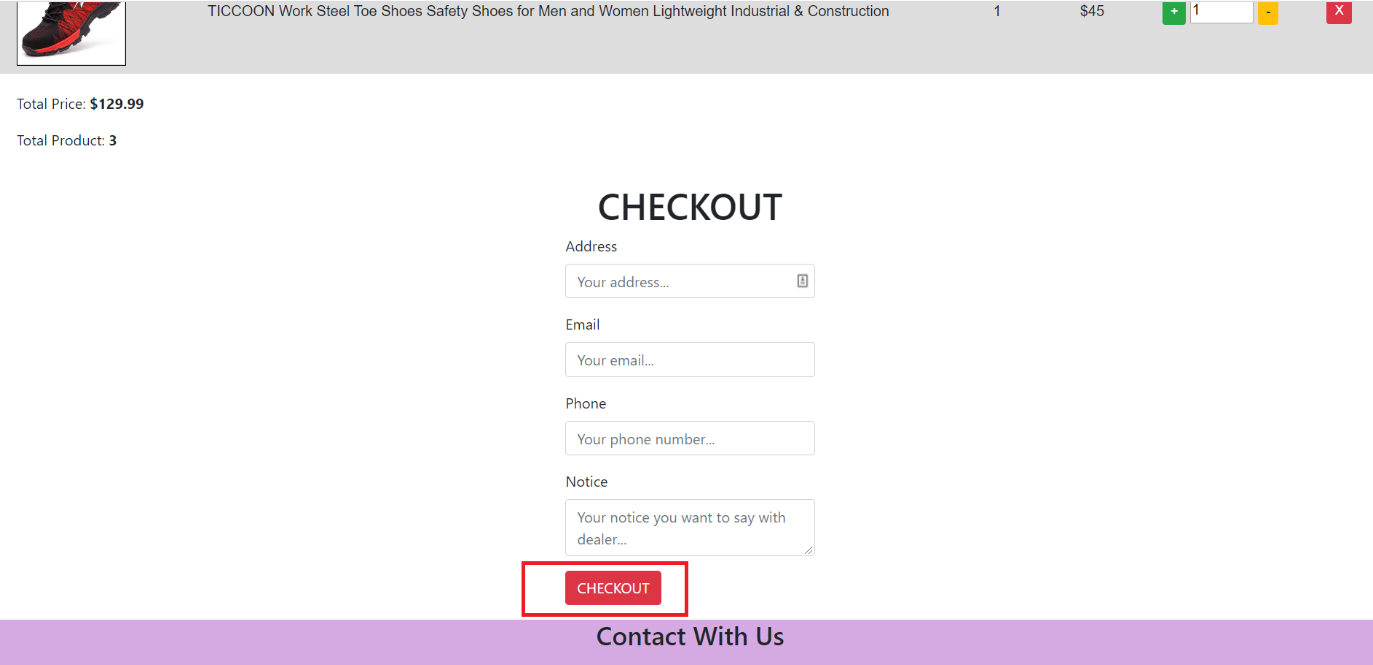
In this use case diagram, there are four actors: Customers, PayPal, Shipping department and Admin.

**USER:**

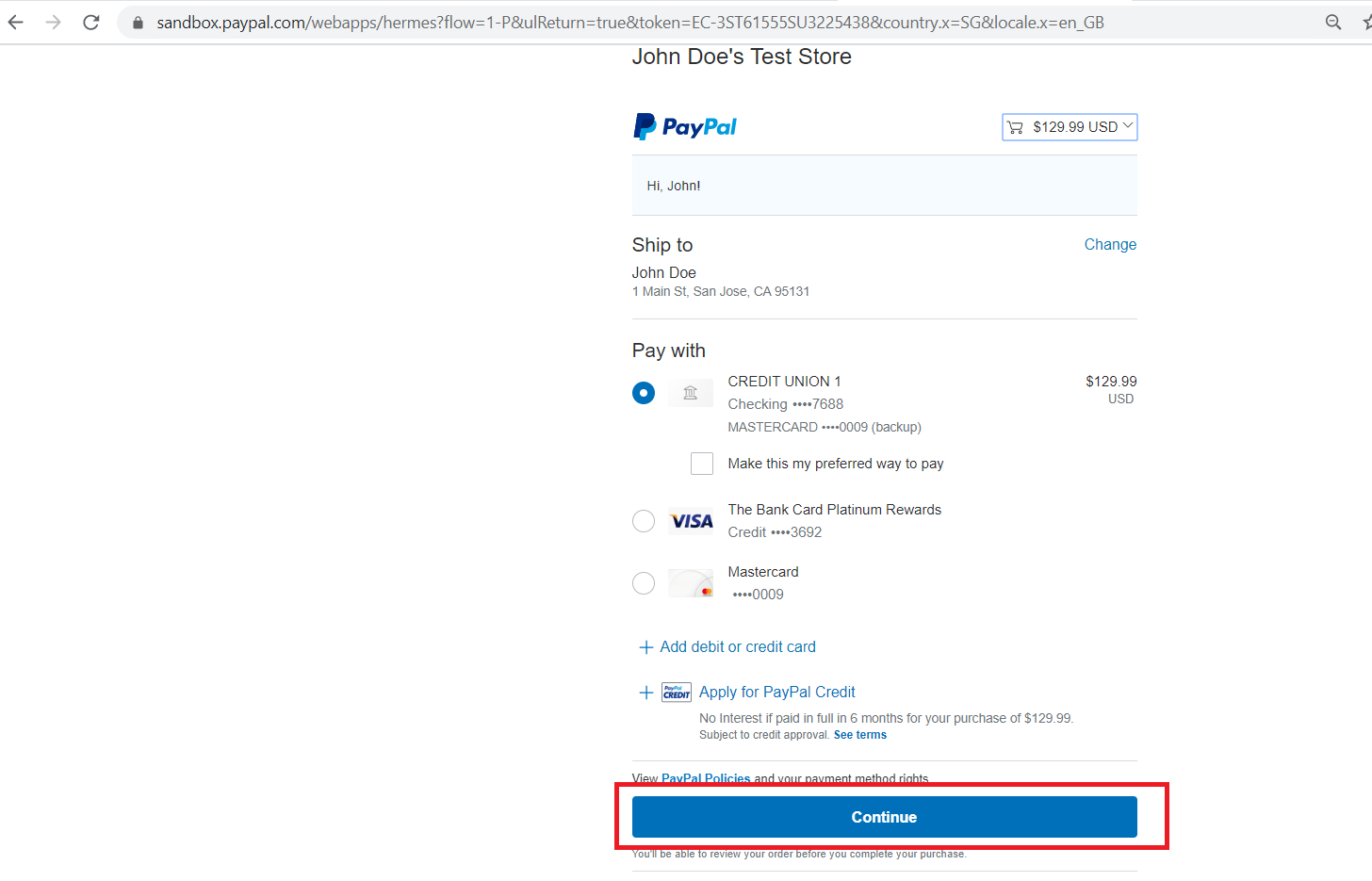
* When user browse our product, he/she want to add item to cart, then when finish, he/she come to view their cart, if done come to checkout part.



* At checkout part, server will check if user is login or not, if not redirect user to login page, ask them to login before continue checkout, if login already, ask them some information like address, email, phone, … then user send the checkout to server, server now will process with third party(PayPal).



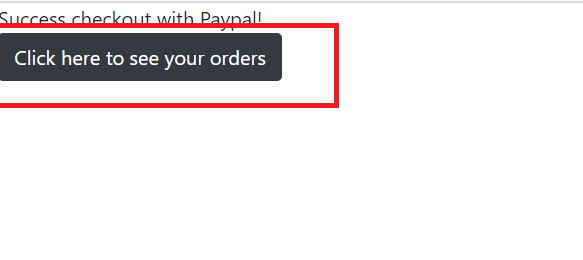
* Server will create checkout payment depend on PayPal API, then send to PayPal, from PayPal their will give back the URL that redirect user to come there to checkout (using debit card or using PayPal account):



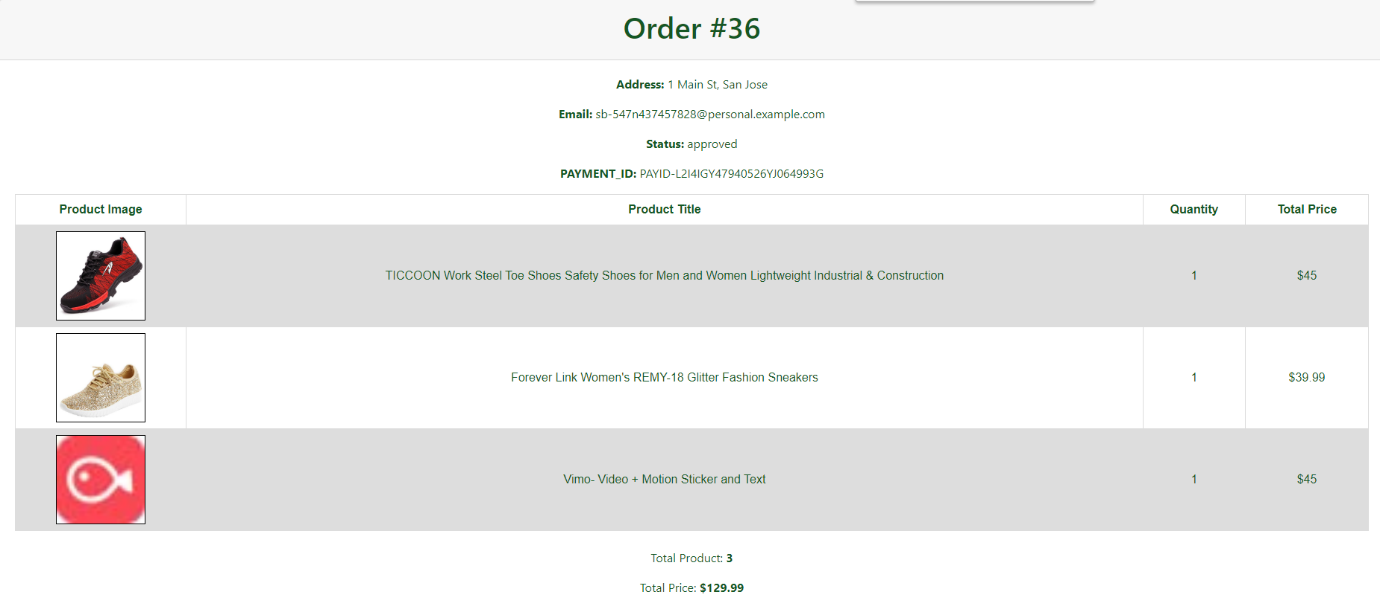
* After checkout success, the user comeback to our website to complete purchase, if success, the server will respond like this:



* This mean user successful buy the product with PayPal, if their want to see the order, just press the black button:



* And the order their just order will show:



1. **Prototype Implementation**

**4.1) Programming Language/tool and Framework**

The Programming language, tool and Framework used for this project are:

* Bootstrap
* Blade Template Engine (can be used outside Laravel)
* Vagrant
* Composer (For PayPal payment)
* PHP
* JQuery
* Font awesome

**4.2) Software Architecture**

Whenever a website is built a software architecture is divided into three parts: The front-end, the Backend and the Database Server. The front-end is made using HTML, CSS, blade template and bootstrap, jQuery. The backend is made using PHP and composer framework. The database server is made using MySQL, it can be connected using vagrant and box contain MySQL.

**4.2.1) The backend**

The backend acts as the backbone of any website. The code should be properly written and structured. The importance of this is that the code works properly, and it is reliable. The source code can be divided into many parts or flies which interconnect with each other and perform a specific task, like responsible for handling the session to process the information, interact with Database Server and also access the information for rendering the front-end.

The backend of the website is programmed using PHP and the host of the web server is Apache in box of vagrant, PHP is the server-side programming language and vagrant is the virtual machine have box that like cross platform program used to create web server with Apache, MySQL tools.

**4.2.1.1)** **MVC Model**

The design pattern of the Model View Controller (MVC) states that an application consists of a data model, information about the display and information about controls. The design involves splitting each of these into distinct objects. MVC is more of an architectural pattern but is not intended for complete use. MVC often relates to an application's UI / interaction layer

**Diagrammatic representation of MVC**



Model: The Model only compromises of the pure application data; it does not contain a logic explaining on how to present the data to a user.

View: The View provides the user with data from the model. The view knows how to access the data of the model, but it does not know what that data means or what the user may do to modify that data.

Controller: There is a Controller between the view and the model. It listens to the events caused by the view (or other external source) and executes the corresponding reaction to them. The reaction is often to call a method on the model. Because view and model are linked via a notification process, the effect of this action is then reflected in view automatically.

The indexfileis an entry point of every website: [**http://v.je/product**](http://v.je/product). Since the site is just searching for index.php file, writing the whole program into one index file is a tedious process which would make the code difficult to manage. The program can be split into several source code files; each file has a chunk of code. These files can be included in the index.php file using the include or require statement.