1. Introduction

1.1 Existing System:

- ✓ In current scenario, all the information related to the customer and their orders are maintained manually.
- ✓ It is less user-friendly.
- ✓ Owner manage all the information of products manually.
- ✓ Not any type of security is providing so it may lose data or product.
- ✓ Customer must go to shop and select products.
- ✓ More human power is required.
- ✓ It is a time-consuming process.

1.2 Need for The New System:

- ✓ There are less chances of losing data because data will be secure in login.
- ✓ New customer can access site using register himself/herself and after login process.
- ✓ Data will be Secure.
- ✓ There is less paperwork; all details will be store on web site.
- ✓ It won't consume much time.
- ✓ Visitors can also view product details.

1.3 Objective Of the New System:

- ✓ In our "Electronics Shopping System" will manage all the information related to the customer and their orders are maintained dynamically.
- ✓ The objective of the project is to make a website to purchase items in an existing shop.
- ✓ So, this is very useful for owner to manage all the information of products and customer's orders.
- ✓ A complete and efficient website which can provide the online shopping experience is the basis objective of the project.
- ✓ The new system will provide user online facility that customers can purchase easily instead of going and purchasing from local shop.
- ✓ The purpose of online shopping is to save time, save money.

1.4 Problem Definition:

✓ The Project "Electronics Shopping System" will be a web-based application for "A to Z Electronics". By this, one can manage product details, orders details, order return, generate reports.

✓ Customers can easily buy a product like Smartphones, tablets, iPads.

✓ This project is an attempt to provide the advantages of online shopping to customers of a real shop.

1.5 Core Components:

1. **Login:**

   To make any products available in the viewer's cart or order viewer must be logged in the website. Viewers have to fill some details about them and create an account for the further process.

2. **Admin**:

   In the Admin Module Admin is the User which has overall management of site.

3. **User:**

   This module is for the users who have not registered. Here user will be able to create an account for login process. The account creation is done by filling the registration form with user details such as name, password etc.

4. **Product**:

   This module has information regarding the products such as its name, model, colour, price, information, its features etc. The Admin has the authority to Add, Delete, Update etc. The User can only view the Electronics Gadgets and Accessories, add to cart and purchase it.

5. **Category:**

   This module will have the electronics categories like smartphones, Tablets, IPads and etc. which user can easily find the estimate product.

6. **Sub Category:**

   This module will come under category module to make more facility to the users to save the time and find the estimate product easily and purchase it.

7. **Cart**:

   This Module User can select any of the Electronics Gadgets and Accessories and add to the cart. User also remove from the cart items, User Edit the items also.

8. **About Us:**

   Viewer can see the details about "Electronics Shopping System's Site".

9. **Contact Us:**

   Viewers can contact admin/owner directly through the number or email for Any Query, Complain related.

1.6 Project Profile:

- ✓ The "Electronics Shopping System" website will provide facility to purchase Smartphones, Tablets, IPads.
- ✓ This website will have Registration page, Login page, Home page, Category page, Cart page, product order page, View product, About Us page, Contact page.
- ✓ The user will be able to add product, edit product and remove product.
- ✓ The admin will have overall management of the website.
- ✓ The users which are not registered can only view the product.

Project Name	Online Electronic Shopping System
Framework	Laravel (PHP)
Operating System	Windows 10
Front End	HTML, CSS, JavaScript, Bootstrap, jQuery, Ajax
Back End	MySQL 5.5.24
Web Server	Xampp 7.4.20
Diagram Tool	MS Visio professional 2013, Visual Paradigm 8.0
Presentation Tool	Microsoft PowerPoint 2016

Table :1.1 Project Profile

1.7 Assumptions And Constraints

Assumptions:

- ✓ Every client wants a fast, responsive and best service from website, if time support so will also provide the product return and login process using user name and password facilities to the users.

Constraints:

- ✓ In website they will maintain different types of services. It will give notification when customer place order. So, admin can manage status, shipped, and delivery.
- ✓ Client should be happy with the service.
- ✓ Client should easily get appointment.

1.8 Advantages and Limitations of The Proposed System

Advantages:

- ✓ The Customers can register / login, view products details, purchase the products from anytime and anywhere.
- ✓ Required Invoice & Reports can be generated easily.
- ✓ Data will be Secure.
- ✓ Real-time information of availability of product.
- ✓ Save the customer's time.
- ✓ Quick and affordable marketing.
- ✓ 24*7 Service availability.

Limitations:

- ✓ Internet bandwidth.
- ✓ Takes time for delivery.

2. Requirement Determination And Analysis

2.1 Requirement Determination:

- Feasibility Study:

✓ A feasibility study is a preliminary study undertaken to determine and document a project's viability. The results of this study are used to make a decision whether to proceed with the project, or dismiss it. If it indeed leads to a project being approved, it will – before the real work of the proposed Project starts – be used to ascertain the likelihood of the project's success. It is an analysis of possible alternative solutions to a problem and a recommendation on the best alternative.

✓ Three types of project feasibility have been considered:

i) Technical Feasibility:

The following factors suffice for considering the given project as Technically Feasible. The system developed in Laravel framework which is well known and today we can easily get the technical help of Laravel technology from the internet. We have used this technology and similar types of tools that can be useful to develop this system. Which is readily available for the development environment?

ii) Economic Feasibility:

Economic feasibility is very important in development of the software for any company. It is necessary as it gives an idea, whether the project going to be developed can be completed at a cost affordable both by the client and developer. The availability of the required hardware and software used to develop our project makes it economically very feasible. Also, we have all the other required resources needed for the project hence the project is feasible with respect to economy.

iii) Operational Feasibility:

Proposed System is beneficial only if they are turned into Information Systems that will meet the organisation's operating requirements.

This test of feasibility asks if the system will work when it is developed and deployed. Are there any major barriers to implementation? The following factors suffice for considering the given project as operationally feasible. As the System is going to be developed at the place where it is going to be implemented, the track of the operations related to the software is constantly monitored by them and sufficient support is available.

IV) Functional Requirements:

- Admin can login and manage the customers.
- Customers can register and then login.
- Customers can manage and edit their profile.
- Admin can add, update and delete the products.
- Customers can buy product online/COD.

❖ Non-Functional Requirements:

- Should work efficiently on regular internet connection.
- Secure database.

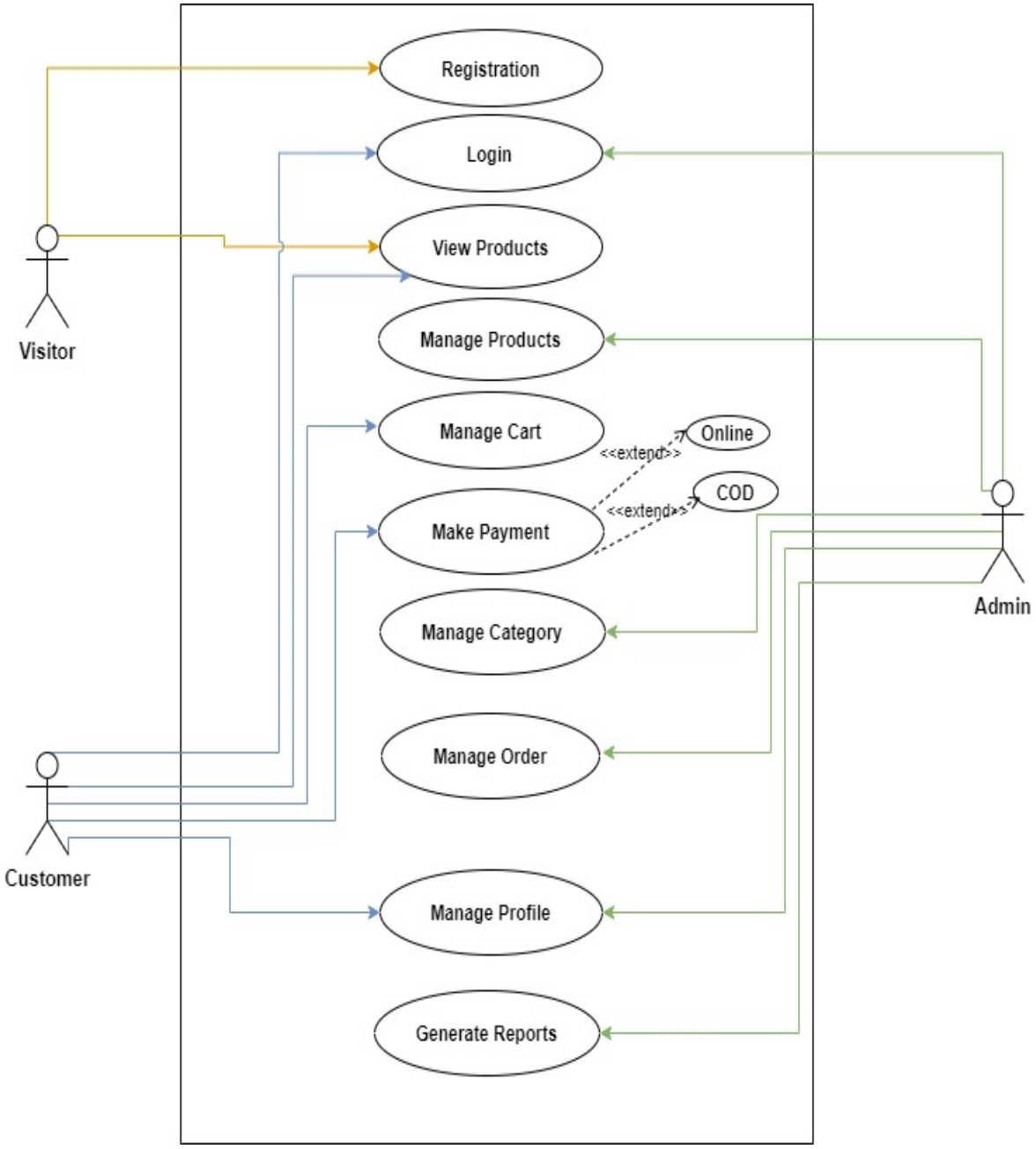
2.2 Targeted Users:

Name	Description	Task/Responsibility
Visitor	Visitor of the System	Visitors can view the products and register.
Admin	Admin has a control over the whole system	Admin can manage products, categories, orders.
Customer	Customer of the system	Customer can log in system Customer can view products with their detail and can add to cart and purchase it.

Table: 1.2 Targeted User

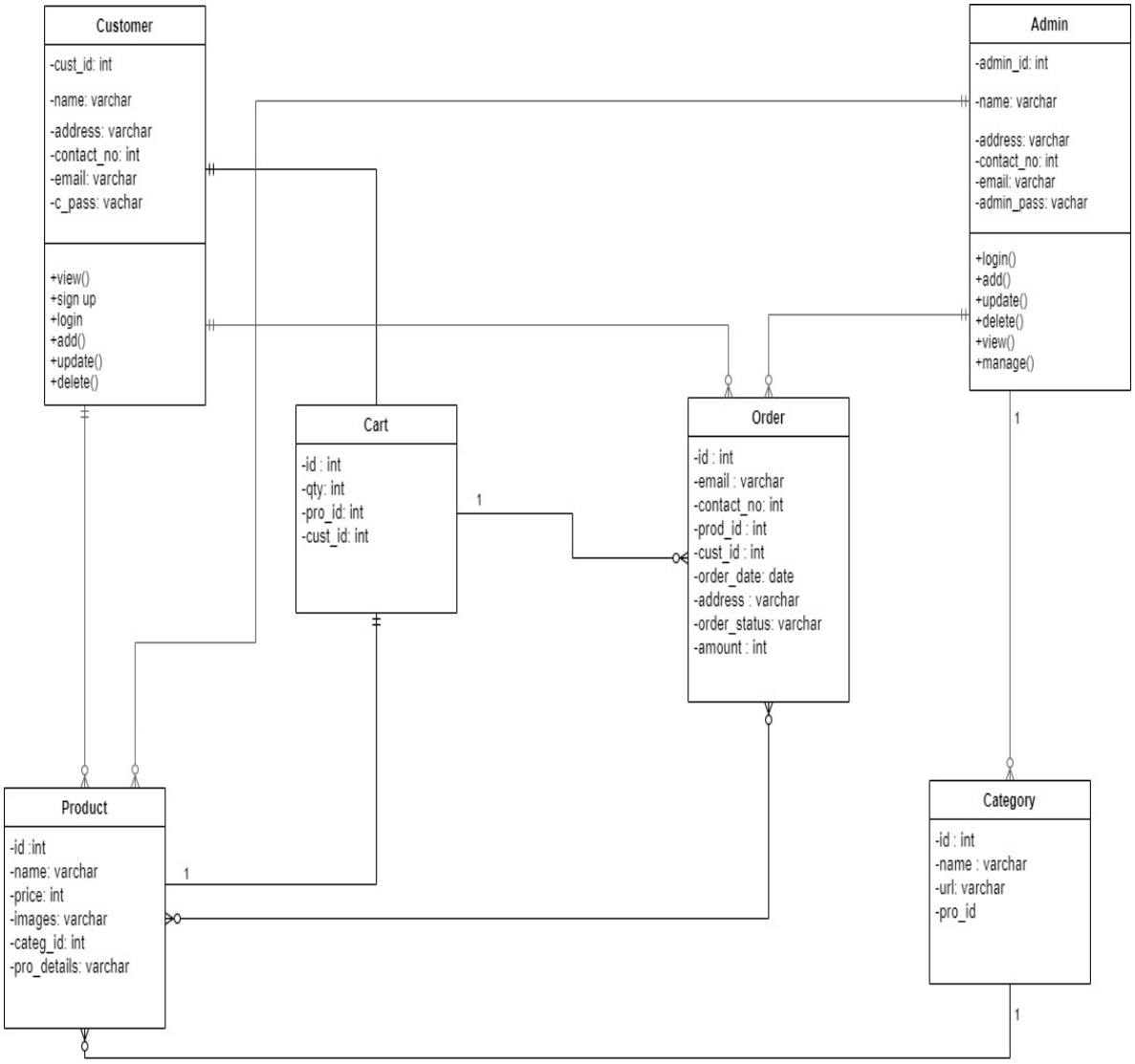
3. System Design

3.1 Use Case Diagrams:



Fiqure: 3.1 Use Case Diagram

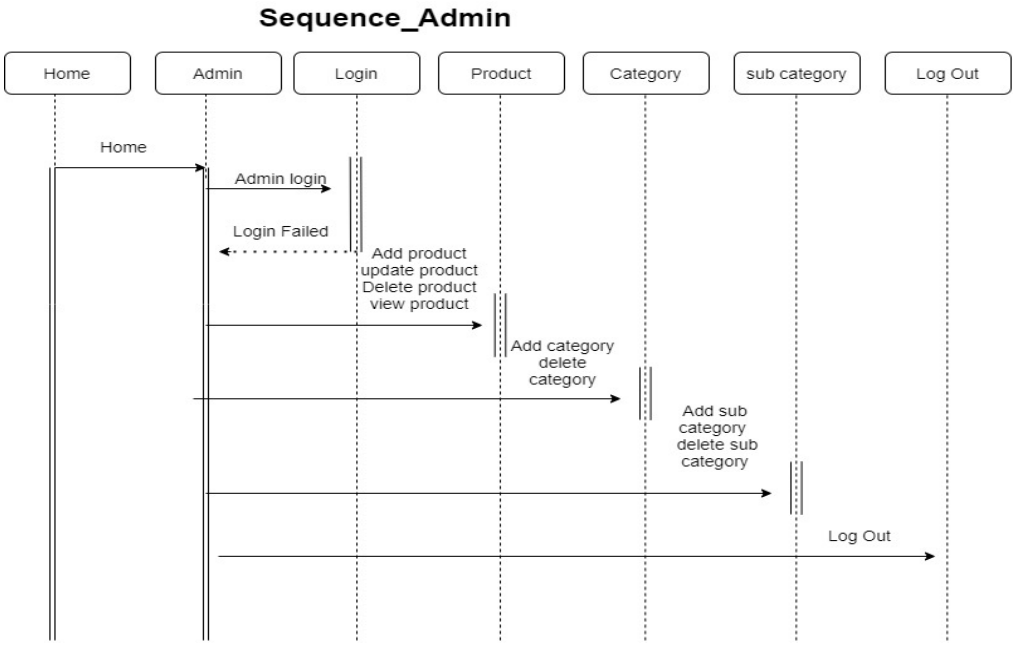
3.2 Class Diagram:



Fiqure: 3.2 Class Diagram

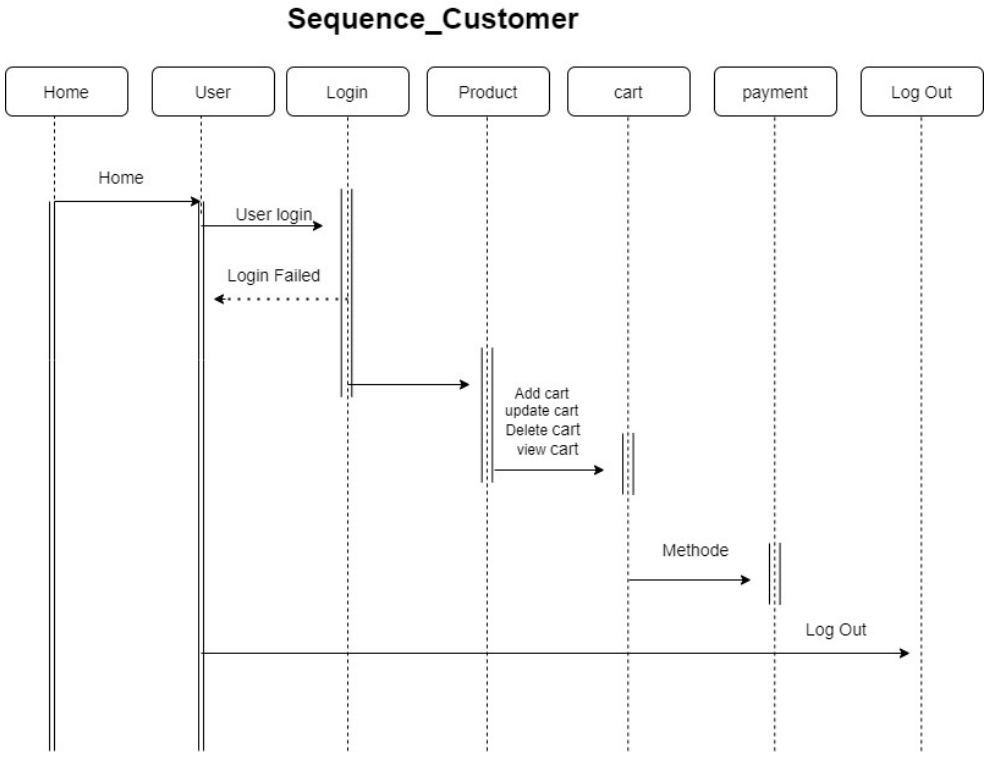
3.3 Sequence Diagrams:

Admin:



Fiqure: 3.3.1 Sequence_Admin Diagram

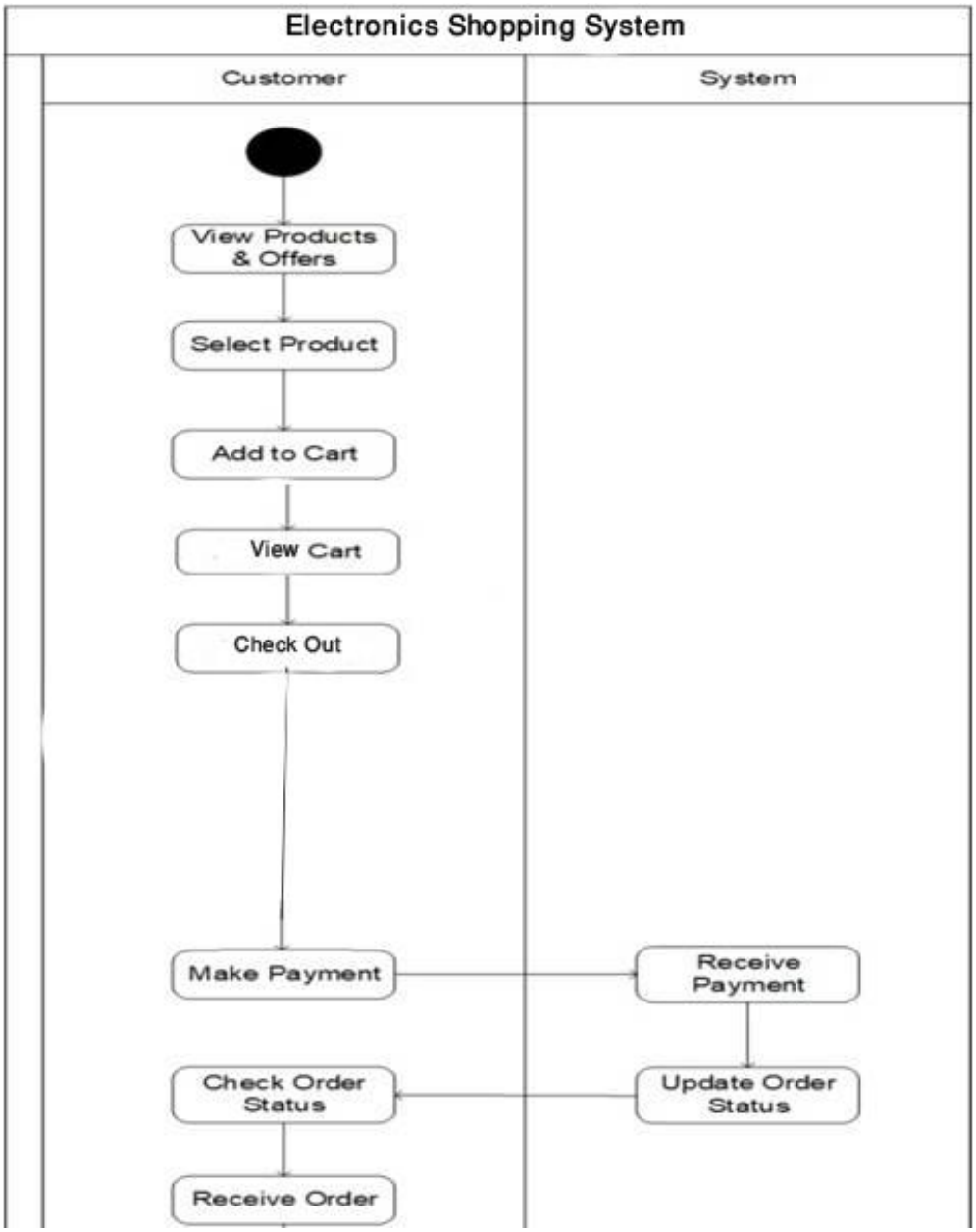
Customer:



Fiqure: 3.3.2 Sequence_Customer Diagram

3.4 Activity Diagram:

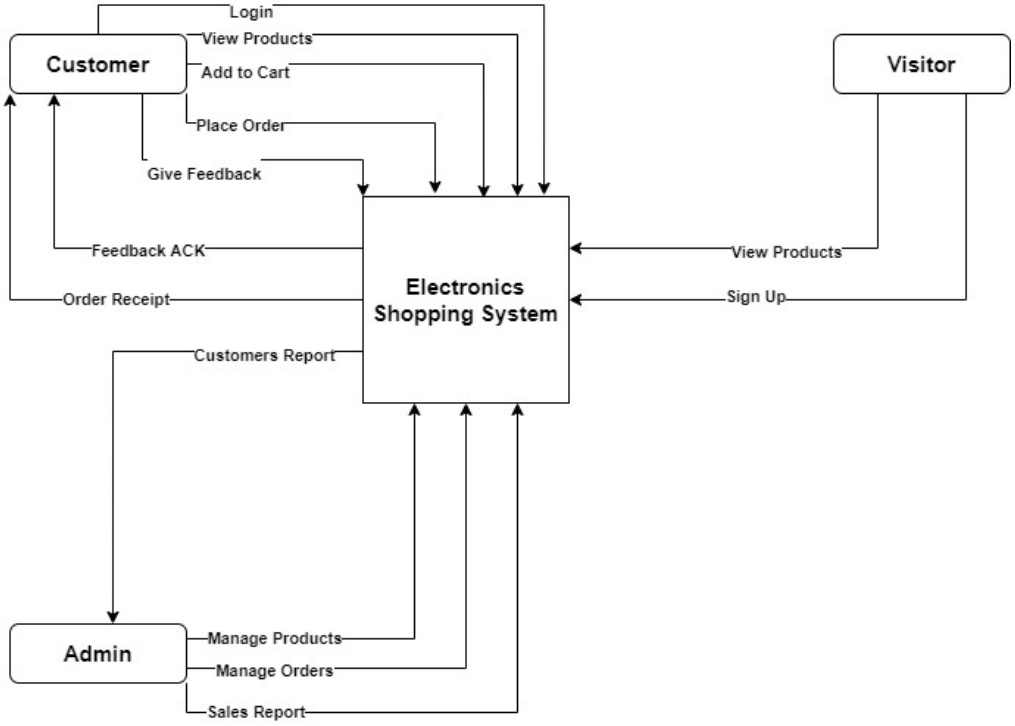
Place Order:



Fiqure: 3.4 Activity Diagram

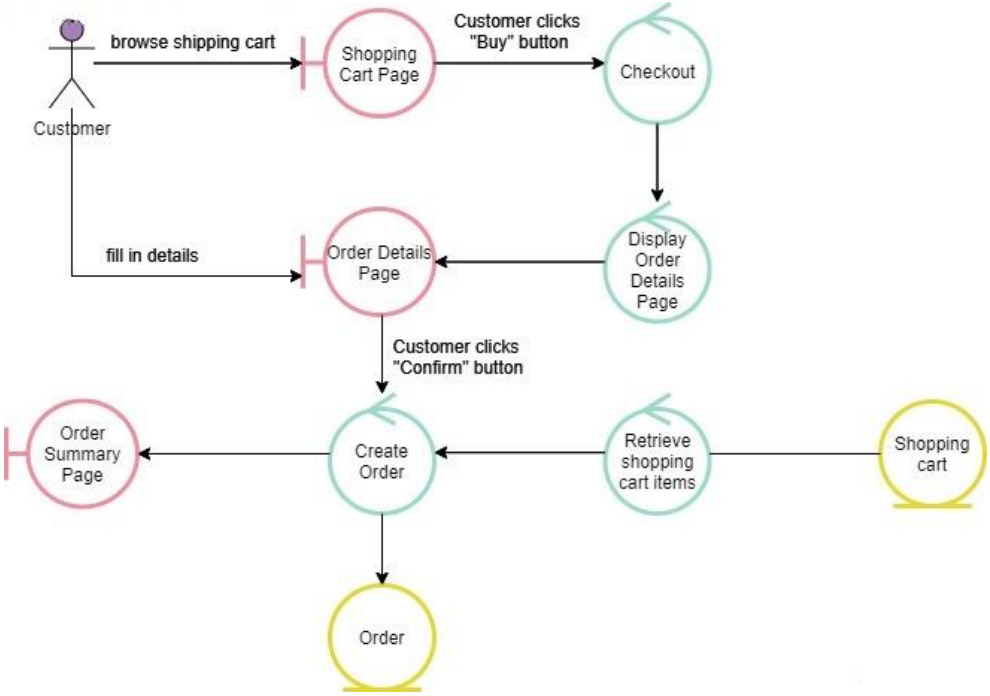
3.5 DFD Diagrams:

DFD Level 0:



Fiqure: 3.5.1 DFD Level 0

DFD Level 1:



Fiqure: 3.5.2 DFD Level 1

3.6 Data Dictionary:

1. Company:

Description: This table shows all the company details.

Field Name	Data Type	Constraint	Sample Data
Company_id	Tinyint(1)	Primary Key	1
company_name	Varchar(30)	Not Null	Saifu Tech
Address	Varchar(100)	Not Null	Morbi Road,Rajkot
conctact_no	Bigint(13)	Not Null	9714535459
Email	Varchar(40)	Not Null	codewithsaifu@gmail.com

Table: 1.3

2. Customer:

Description: This table shows all the customer details.

Field Name	Data Type	Constraint	Sample Data
cust_id	Mediumint(6)	Primary Key	1
name	Varchar(100)	Not Null	Saifullah Rahimi
address	Varchar(255)	Not Null	Marwadi Compus
contact_no	Bigint(13)	Not Null	9714535459
email	Varchar(50)	Not null	codewithsaifullah@gmail.com
password	Varchar(16)	Not null	Eshgdjded854dednd (MD5 Format)

Table: 1.4

3. Category:

Description: This table shows all the category details.

Field Name	Data Type	Constraint	Sample Data
category_id	Smallint(3)	Primary Key	1
p_id	Smallint(3)	Foreign Key	12
Name	Varchar(50)	Not Null	Mobile
url	Varchar(50)	Not null	Mobile/mi

Table: 1.5

4. Product:

Description: This table shows all the product details.

Field Name	Data Type	Constraint	Sample Data
product_id	Smallint(3)	Primary Key	1
category_id	Smallint(3)	Foreign Key	1
Price	Int(20)	Not Null	1000000
Description	Text	Not null	This isRedmi MobilePhone
Image	Varchar(100)	Not null	Redmi9pro.jpg

Table: 1.6

5. Cart:

Description: This table shows all the cart details.

Field Name	Data Type	Constraint	Sample Data
cart_id	Smallint(3)	Primary Key	1
prodcut_id	MediumInt(3)	Foreign Key	1
customer_id	Smallint(3)	Foreign key	1
Qty	Tinyint(2)	Not Null	2

Table: 1.7

6. Order:

Description: This table shows all the order details.

Field Name	Data Type	Constraint	Sample Data
order_id	Mediumint(6)	Primary Key	1
product_id	Mediumint(3)	Foreign key	1
customer_id	smallint(6)	Foreign key	1
customer_name	Varchar(50)	Not Null	Rahul
address	Varcahr(255)	Not null	Block no 4,Shahpur
email	Varchar(50)	Not Null	rahulzala999@gmail.Com
order_status	Varchar(30)	Not Null	Shipped/Delivered
amount	Smallint(6)	Not Null	28999
order_date	Date	Not Null	25-05-2020
qty	Tinyint(2)	Not Null	2

Table: 1.8

4. Development

4.1 Coding Standard

CRUD Operation:

```
public function addCompany(Request $req){
    if($req->isMethod('post')){
        $data = $req->all();
        $company = new Company;
        $company->name = $data['name'];
        $company->address = $data['address'];
        $company->mobile = $data['mobile'];
        $company->email = $data['email'];
        $company->save();
        return redirect('/admin/view-company')->with('flash_message_success','Company has been added Successfully.!!');
    }
    return view('admin.company.add_company');
}

public function viewCompany(){
    $companies = Company::get();
    return view('admin.company.view_company')->with(compact('companies'));
}

public function editCompany(Request $req, $id=''){
    if($req->isMethod('post')){
        $data = $req->all();

        Company::where(['id'=>$id])->update(['name'=>$data['name'],'address'=>$data['address'],'mobile'=>$data['mobile'],'email'=>$data['email']]);
        return redirect('/admin/view-company')->with('flash_message_success','Company Details has been updated successfully..!!');
    }
    $companyDetails = Company::where(['id'=>$id])->first();
    return view('admin.company.edit_company')->with(compact('companyDetails'));
}
public function deleteCompany($id = NULL){
    Company::where(['id'=>$id])->delete();
    return redirect()->back()->with('flash_message_success','Company Details has been deleted successfully..!!');
}
```

Figure: 4.1.1 CRUD Operation

Upload Multiple Images:

```php
public function addImages(Request $request, $id=null){
    $productDetails = Product::with('attributes')->where(['id'=>$id])->first();
    if($request->isMethod('post')){
        $data = $request->all();
        if ($request->hasFile('image')) {
            $files = $request->file('image');
            foreach($files as $file){
                // Upload Images after Resize
                $image = new ProductsImage;
                $extension = $file->getClientOriginalExtension();
                $fileName = rand(111,99999).'.'.$extension;
                $large_image_path = 'images/backend_images/product/large'.'/'.$fileName;
                $medium_image_path = 'images/backend_images/product/medium'.'/'.$fileName;
                $small_image_path = 'images/backend_images/product/small'.'/'.$fileName;
                Image::make($file)->save($large_image_path);
                Image::make($file)->resize(600, 600)->save($medium_image_path);
                Image::make($file)->resize(300, 300)->save($small_image_path);
                $image->image = $fileName;
                $image->product_id = $data['product_id'];
                $image->save();
            }
        }

        return redirect('admin/add-images/'.$id)->with('flash_message_success', 'Product Images has been added successfully');

    }
    $productImages = ProductsImage::where(['product_id'=>$id])->get();
    return view('admin.products.add_images')->with(compact('productDetails','productImages'));
}
```

Figure 4.1.2 Upload multiple Images

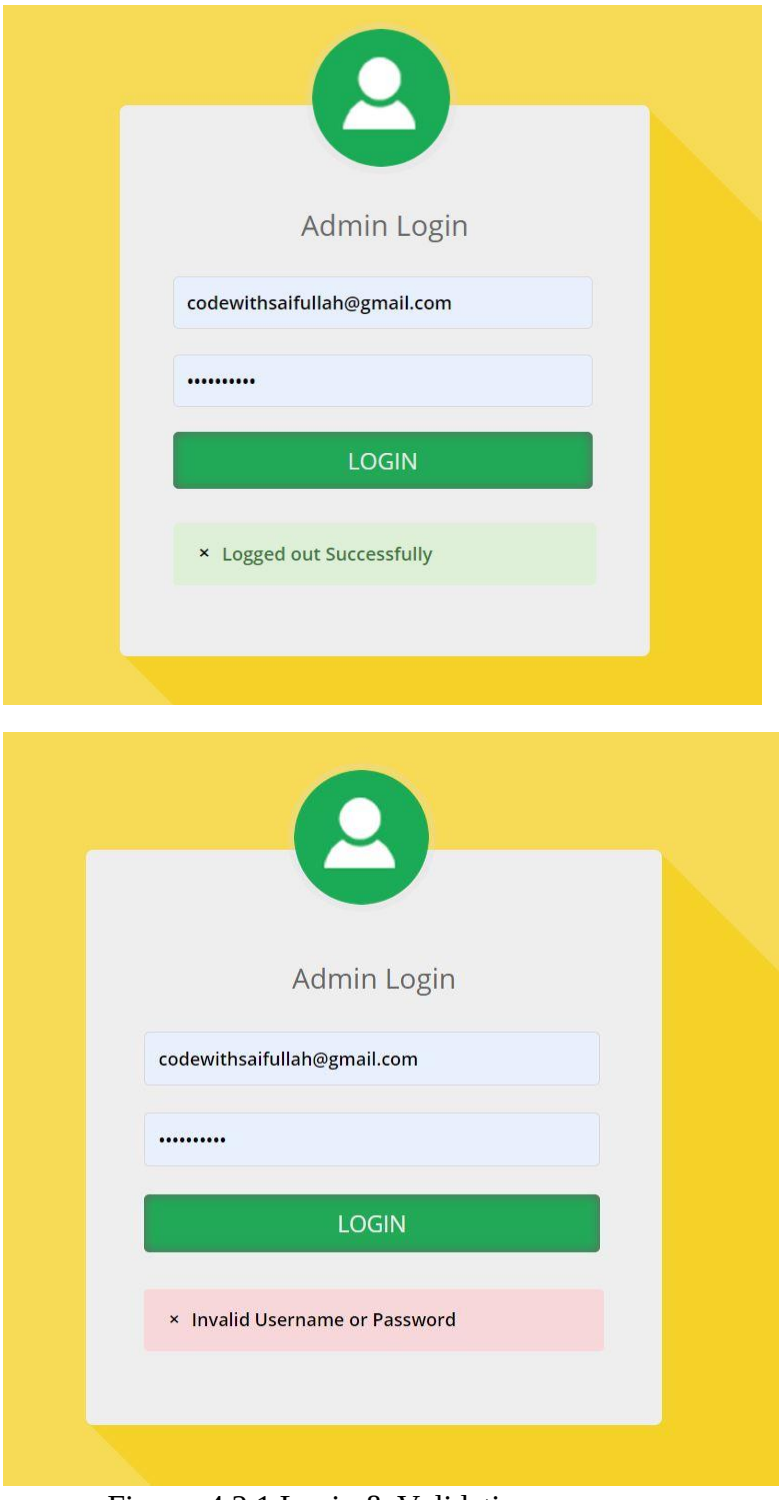
User Register:

```php
public function userRegister(Request $req){
    if($req->isMethod('post')){
        $data = $req->all();
        // echo "<pre>"; print_r($data); die;
        $emailCount = User::where('email',$data['email'])->count();
        $mobileCount = User::where('mobile',$data['mobile'])->count();
        if($emailCount>0){
            return redirect()->back()->with('flash_message_error','Email is already exist');
        }
        if($mobileCount>0){
            return redirect()->back()->with('flash_message_error','Mobile is already exist');
        }else{
            $user = new User;
            $user->name = $data['name'];
            $user->mobile = $data['mobile'];
            if(!empty($data['email'])){
                $user->email = $data['email'];
            }else{
                $user->email = '';
            }
            $user->admin = 0;
            $user->password = bcrypt($data['password']);
            $user->save();
            return redirect('/login-user');
        }
    }
}
```

Figure 4.1.3 User Register

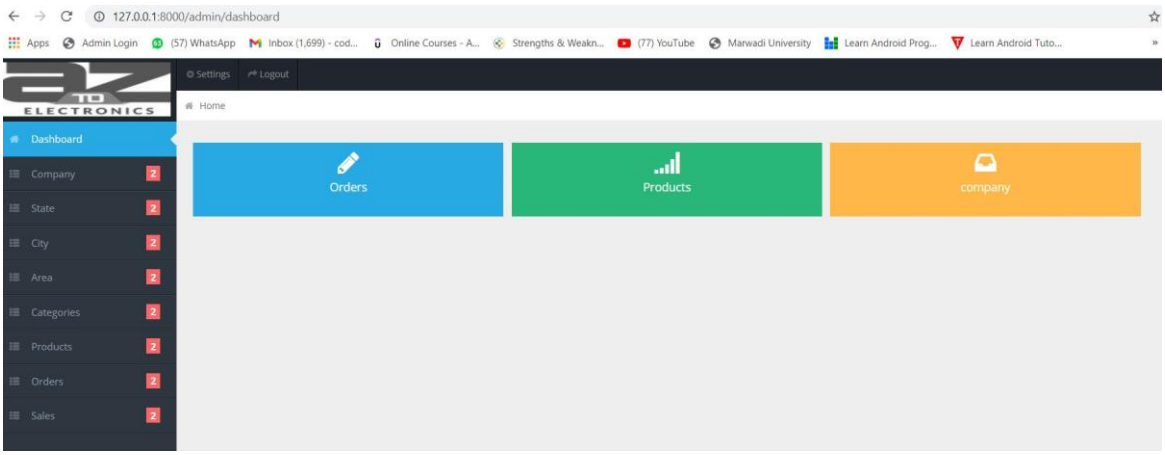
4.2 Admin Screen Layouts:

Login & Validation:



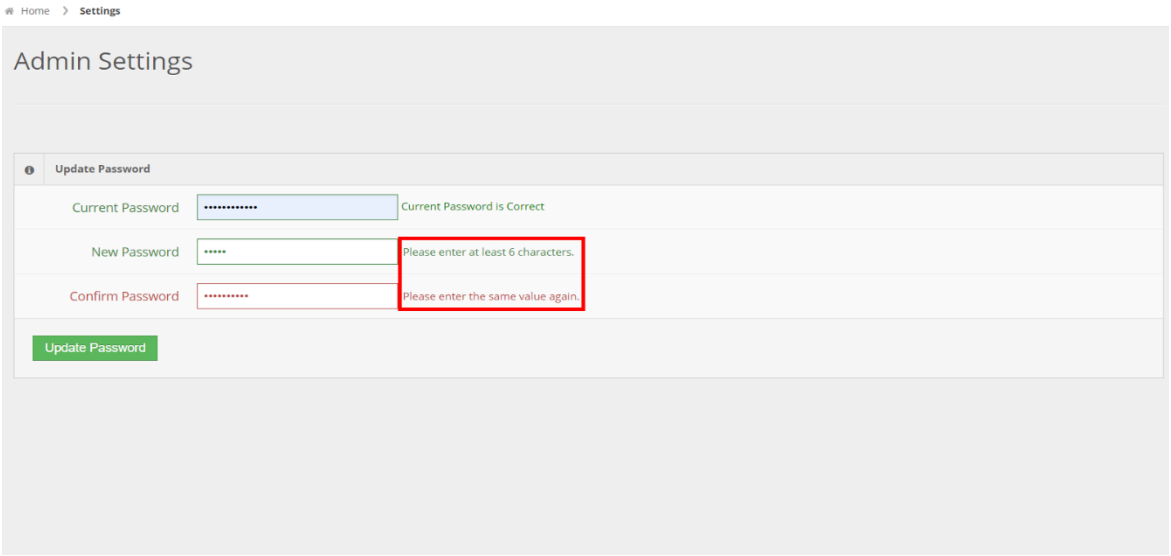
Fiqure: 4.2.1 Login & Validation

Dashboard:



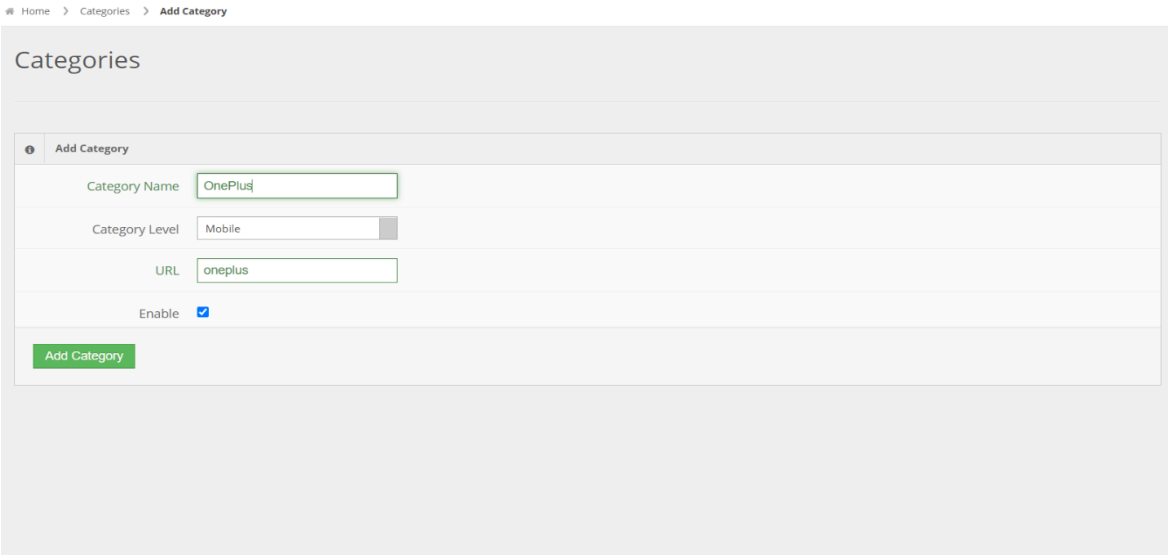
Fiqure: 4.2.2 Dashboard

Update Password & Validation:



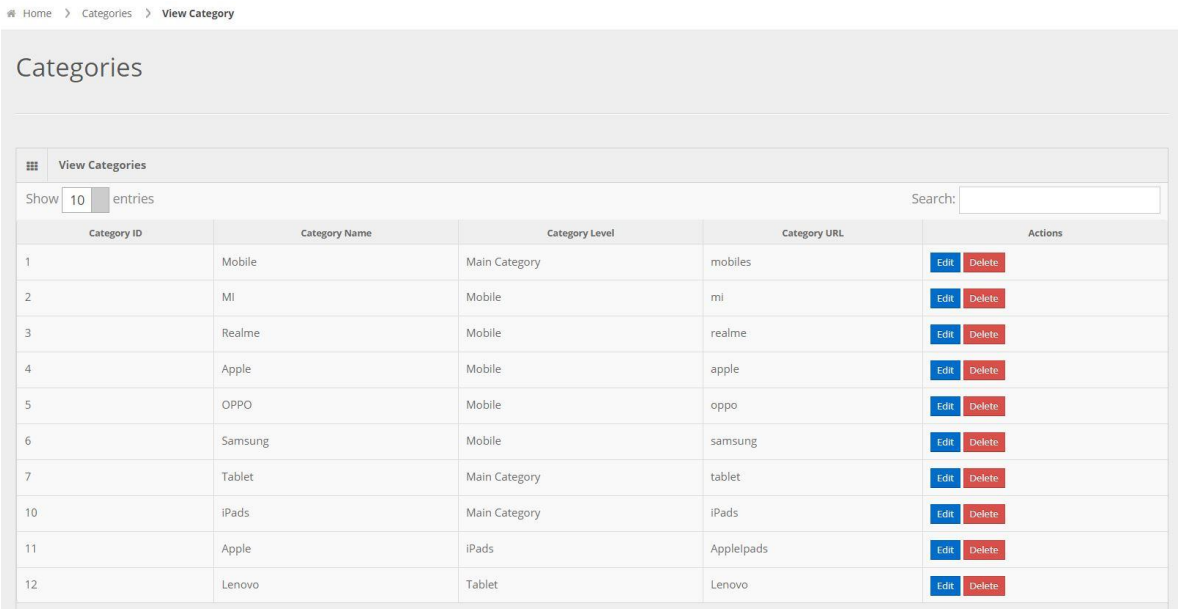
Fiqure: 4.2.3 Update Password

Add Category:



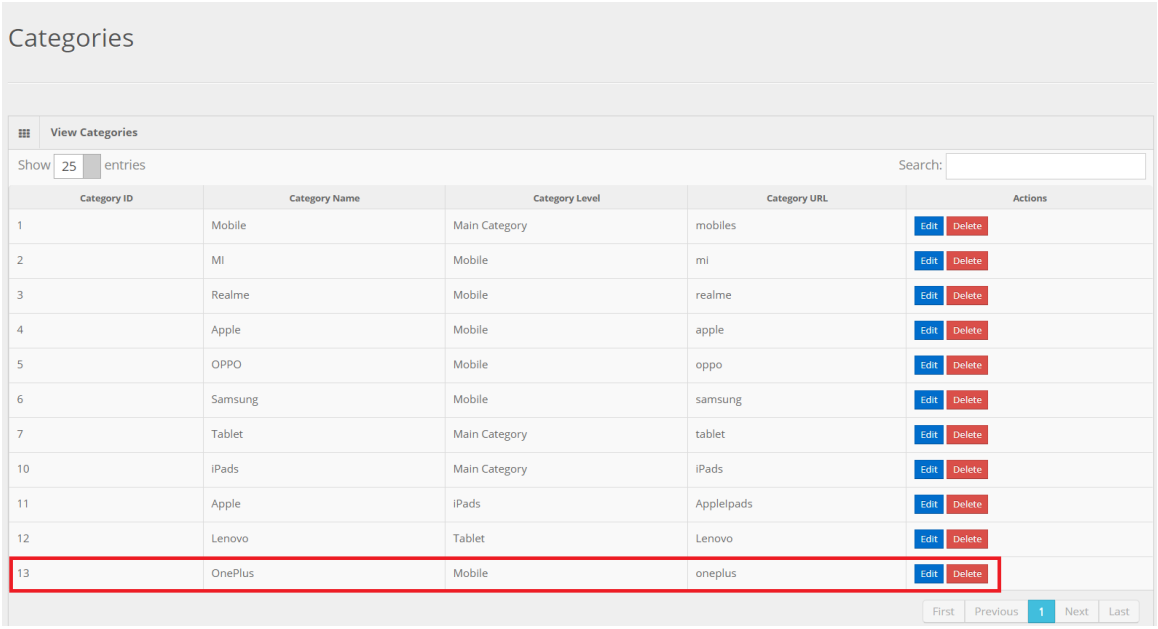
Fiqure: 4.2.4 Insert Category

View Category:



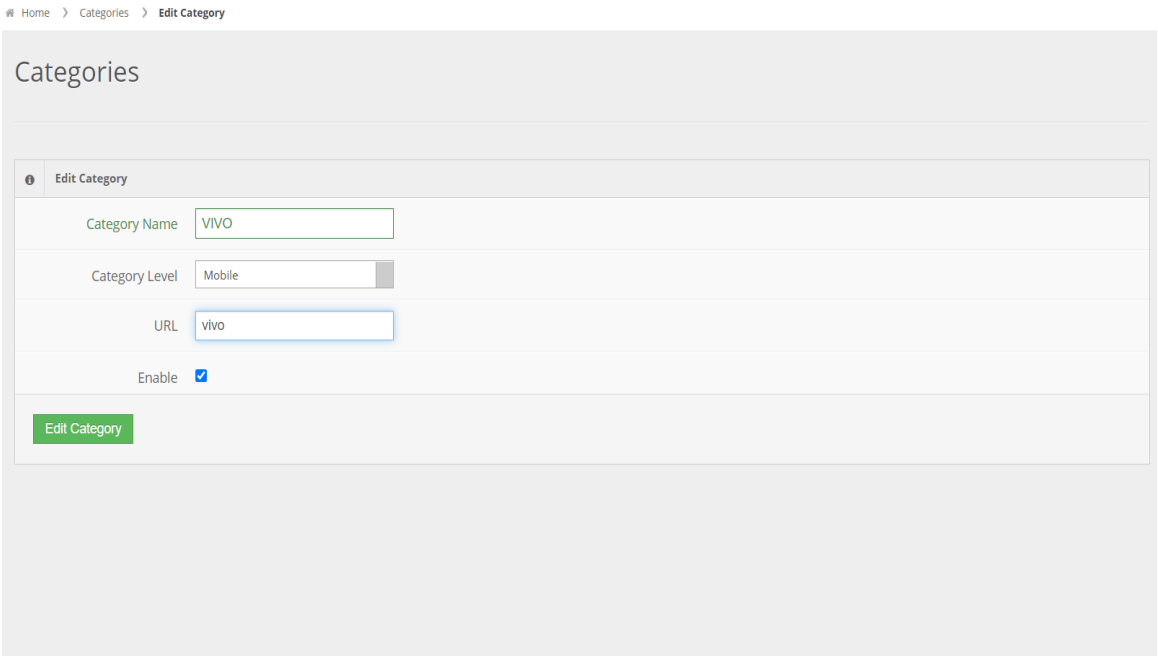
Fiqure: 4.2.5 view Category

Inserted Successfully:



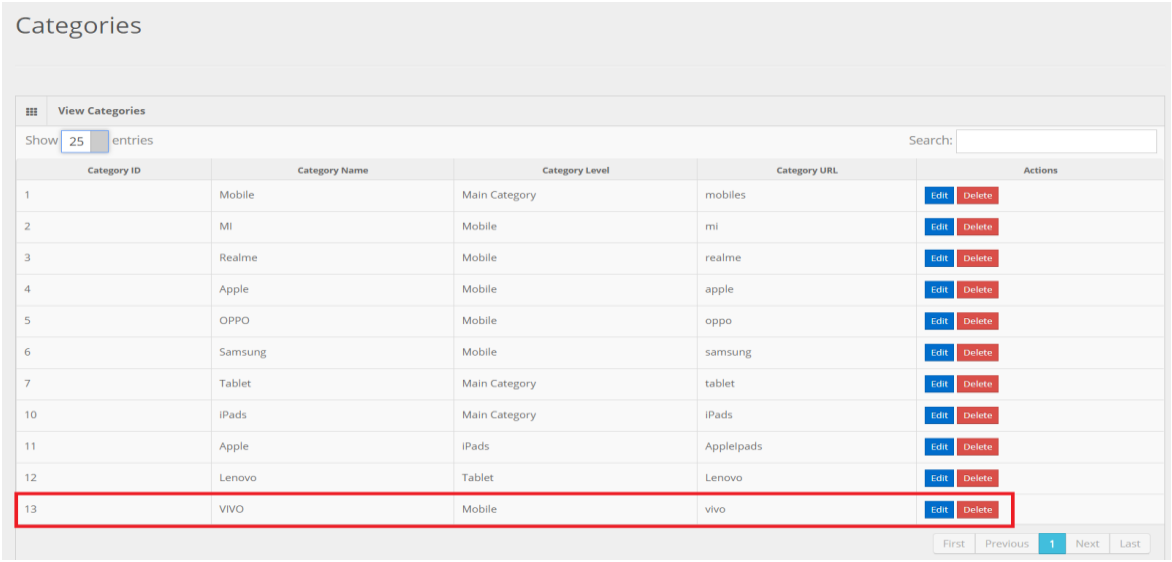
Fiqure: 4.2.6 After Insertion

Edit Category:



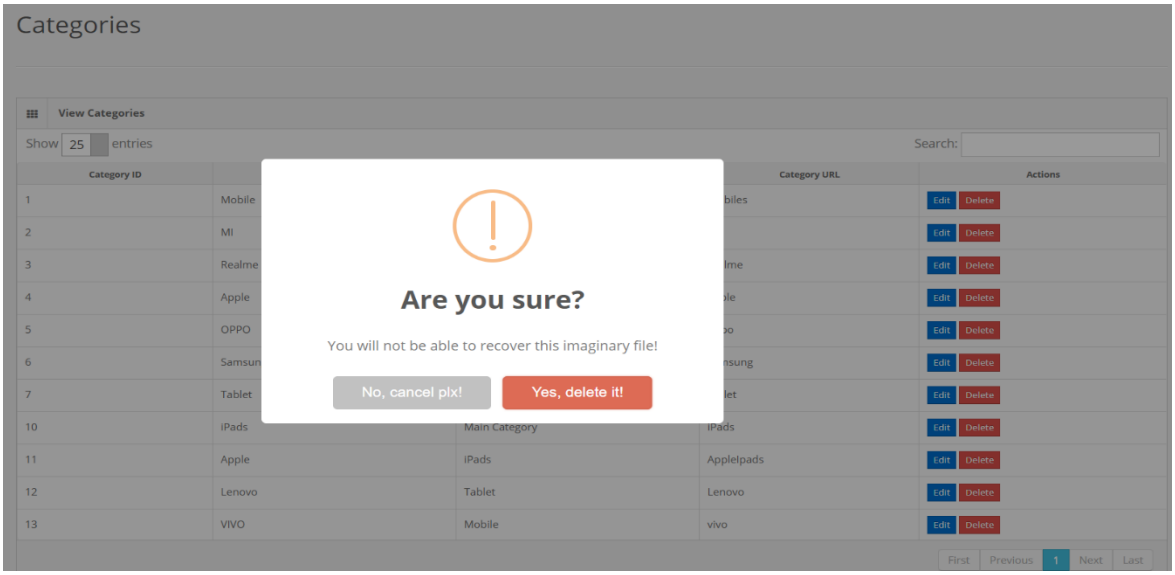
Fiqure: 4.2.7 Edit Category

Category Updated Successfully:



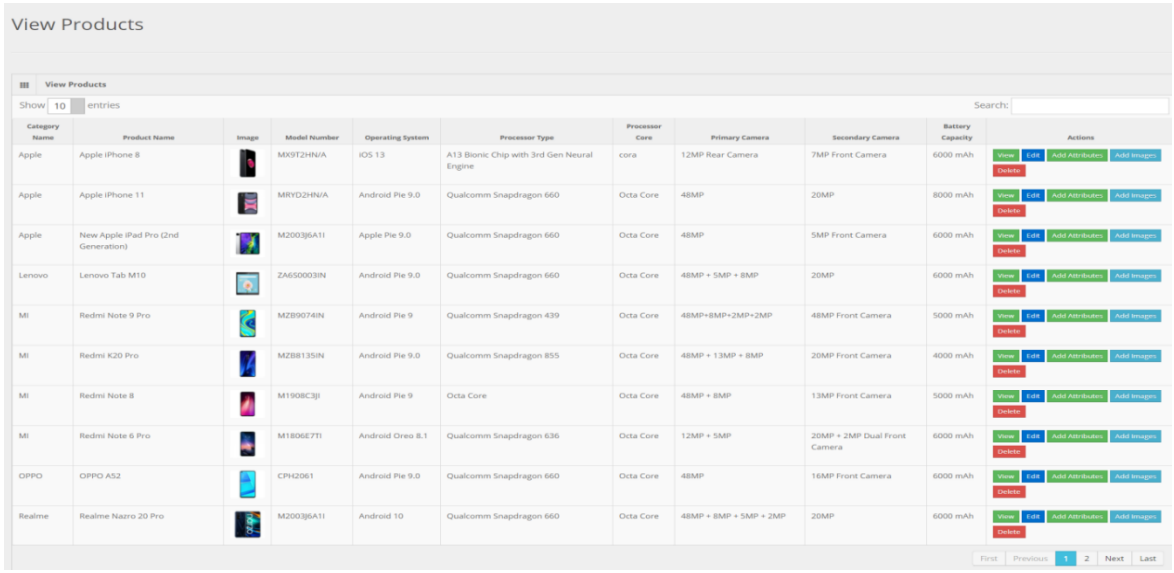
Fiqure: 4.2.8 After Update

Delete Category Confirmation Alert:



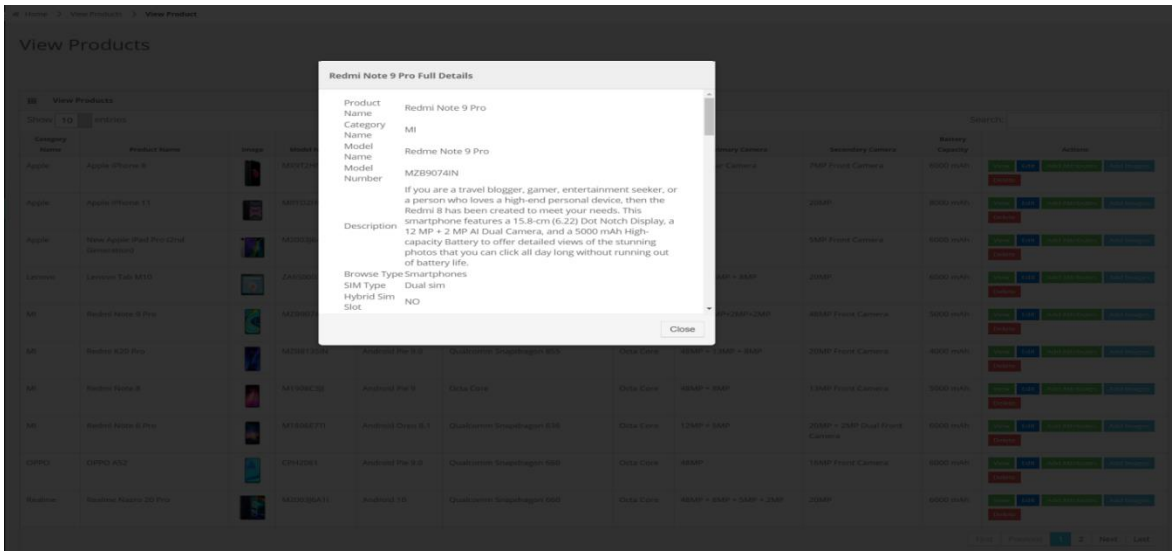
Fiqure: 4.2.9 Delete Alert Message

View Product:



Fiqure: 4.2.10 View Product

View Product Details In Modal:

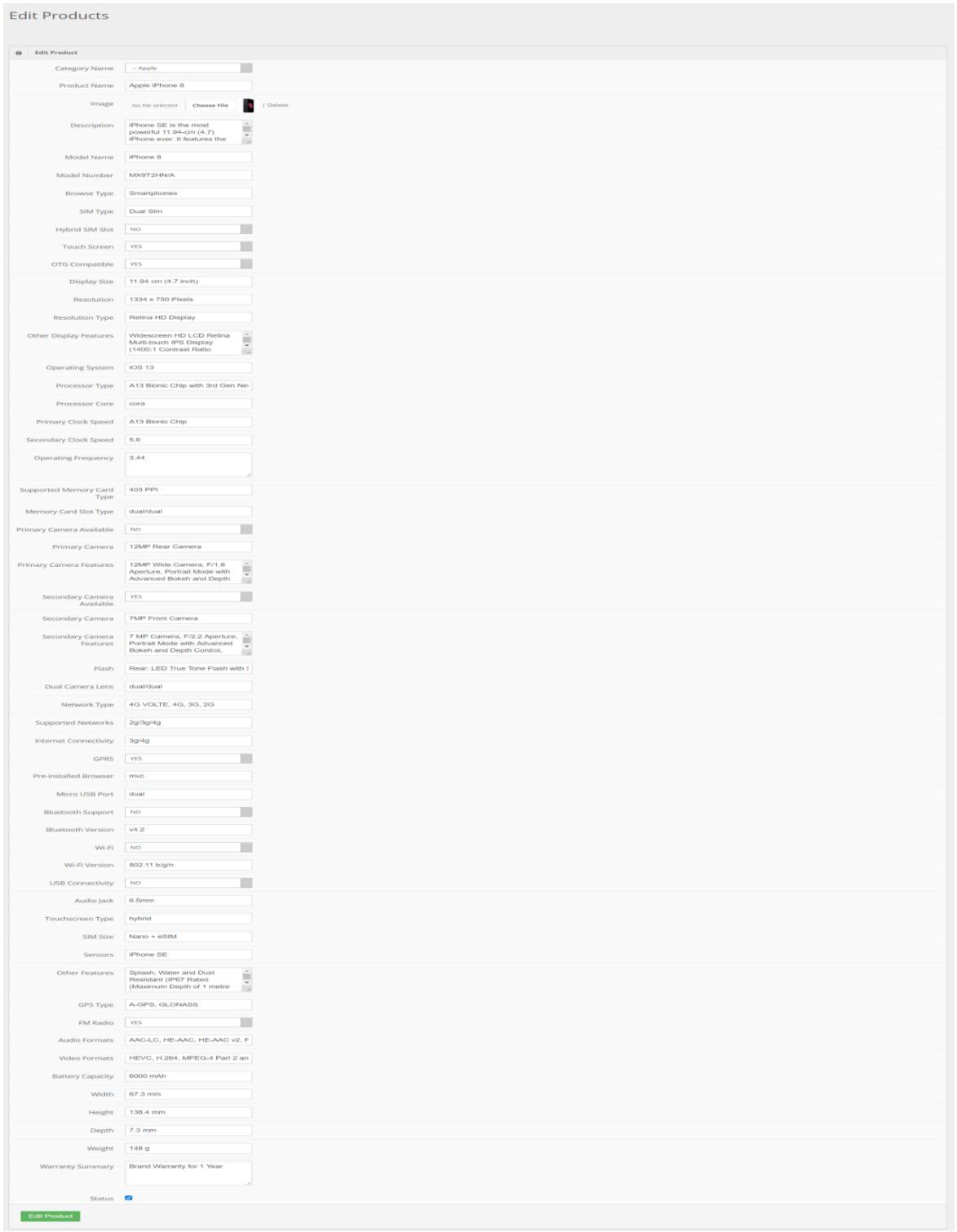


Fiqure: 4.2.11 View Product Details in Modal

Add Product Form:

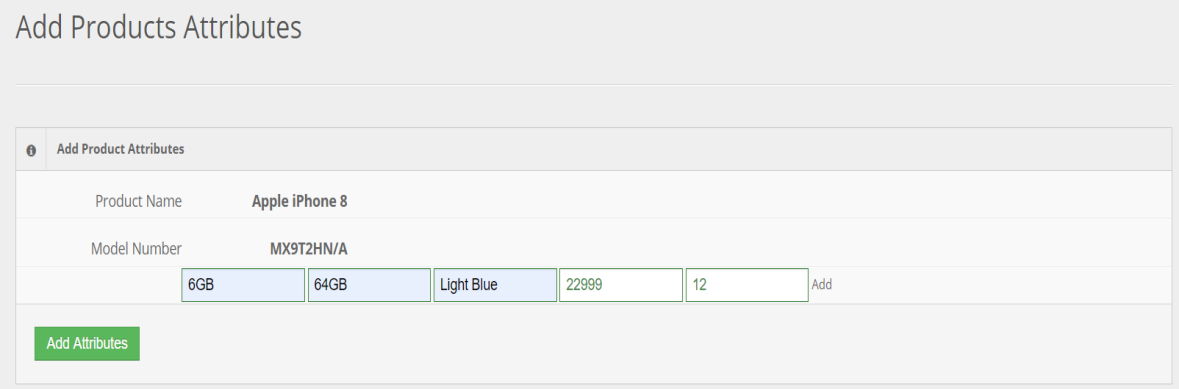
Fiqure: 4.2.12 Add Product Form

Edit Product:



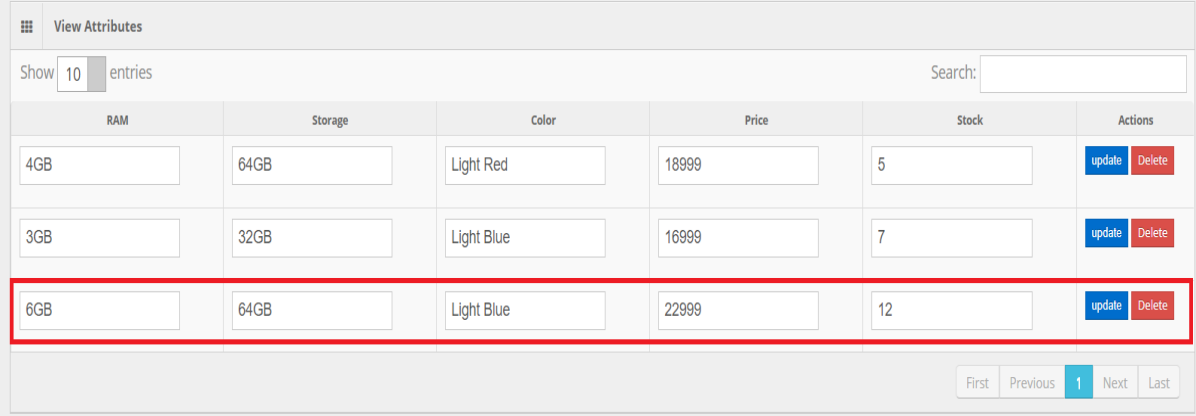
Fiqure: 4.2.13 Edit Product

Insert Product Attributes:



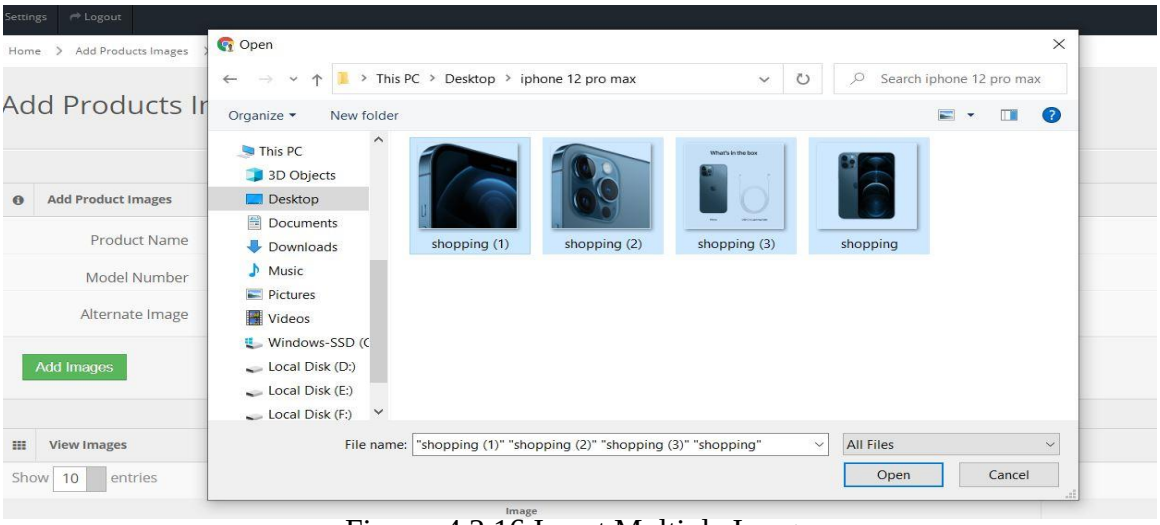
Fiqure: 4.2.14 insert Product Attributes

Product Attributes Inserted Successfully:



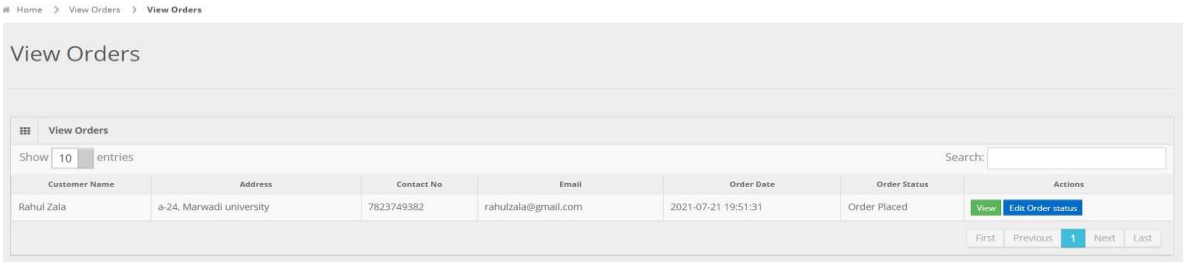
Fiqure: 4.2.15 After Insertion

Insert Multiple Images:



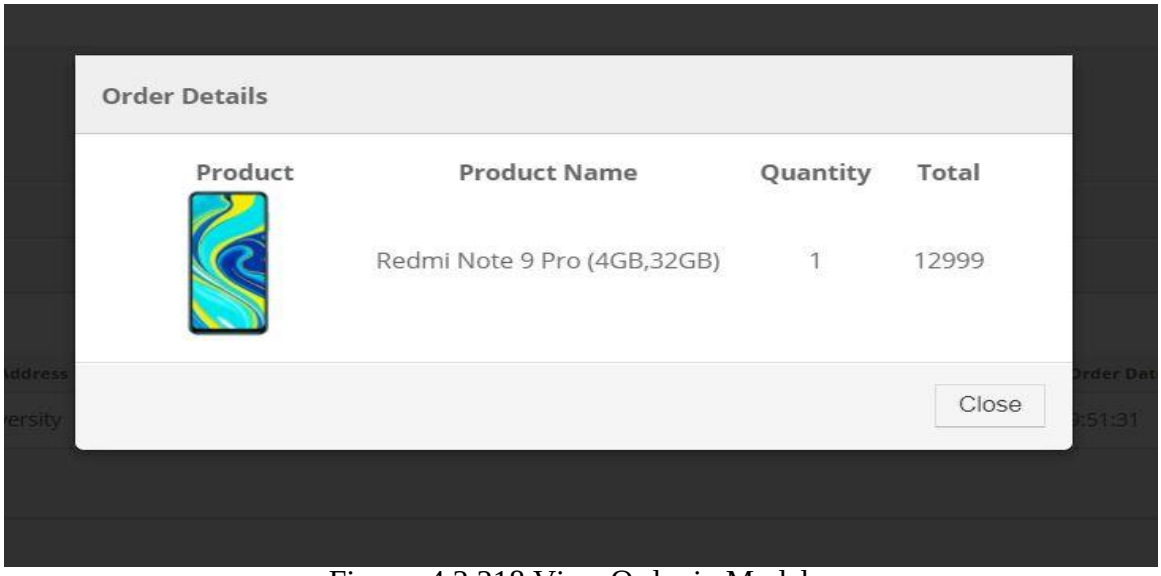
Fiqure: 4.2.16 Insert Multiple Images

View Orders:



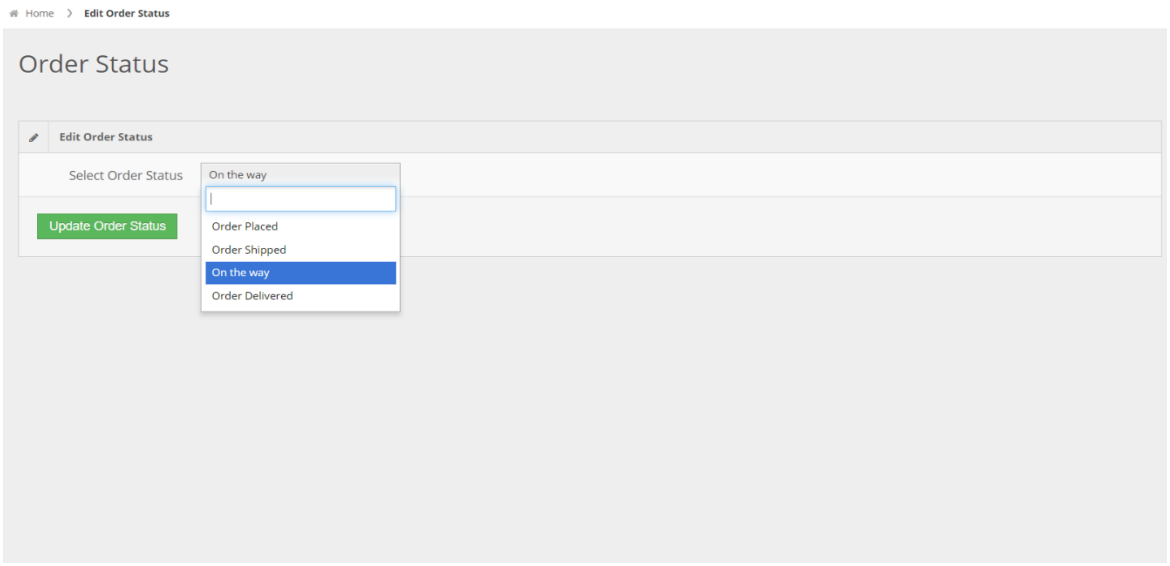
Fiqure: 4.2.17 View Orders

View Orders Detail In Modal:



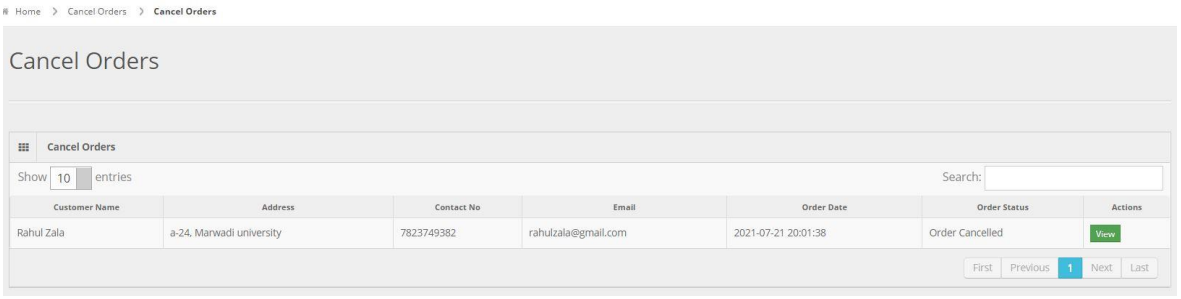
Fiqure: 4.2.218 View Order in Modal

Update Order Status:



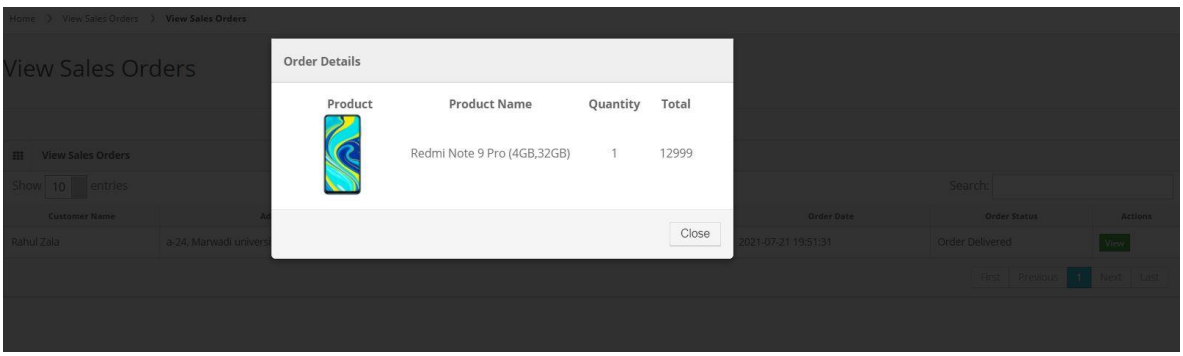
Fiqure: 4.2.19 Update Order Status

View Cancel Orders:



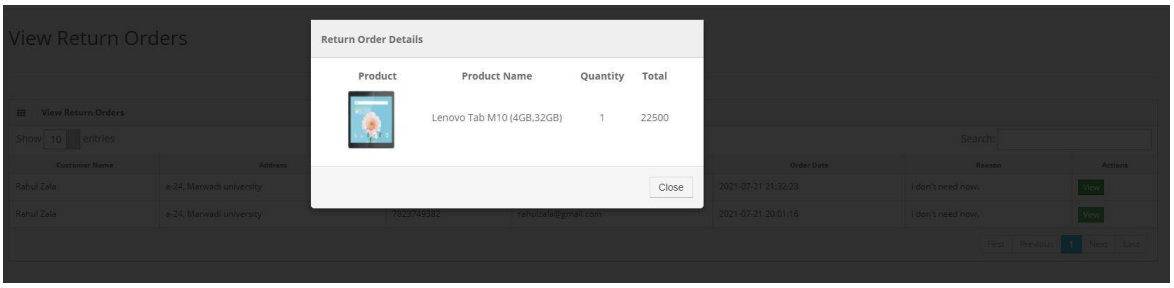
Fiqure: 4.2.20 View Cancel Order

View Sales Orders – Delivered Orders:



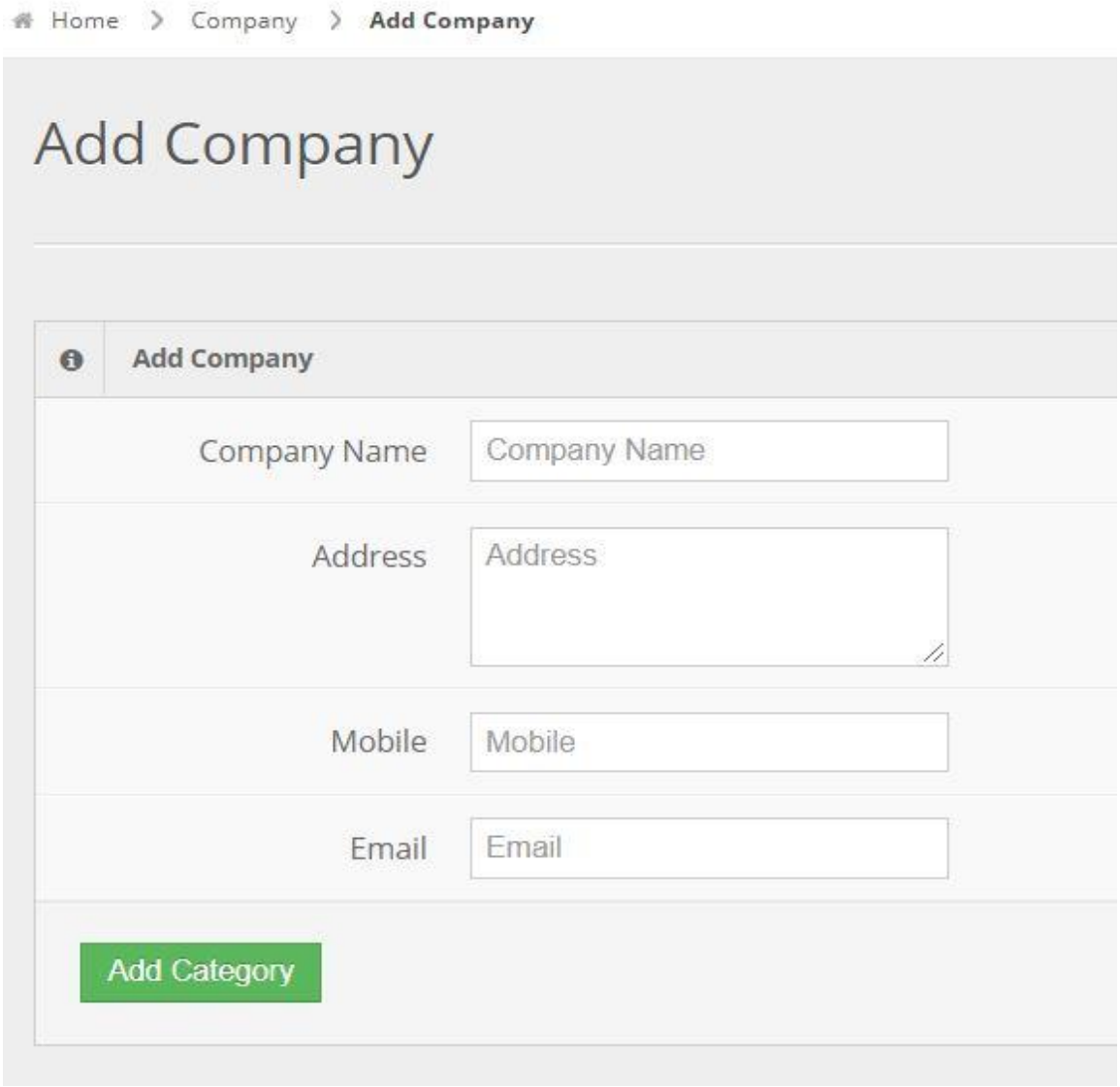
Fiqure: 4.2.21 View Sales Order

View Sales Return Orders :



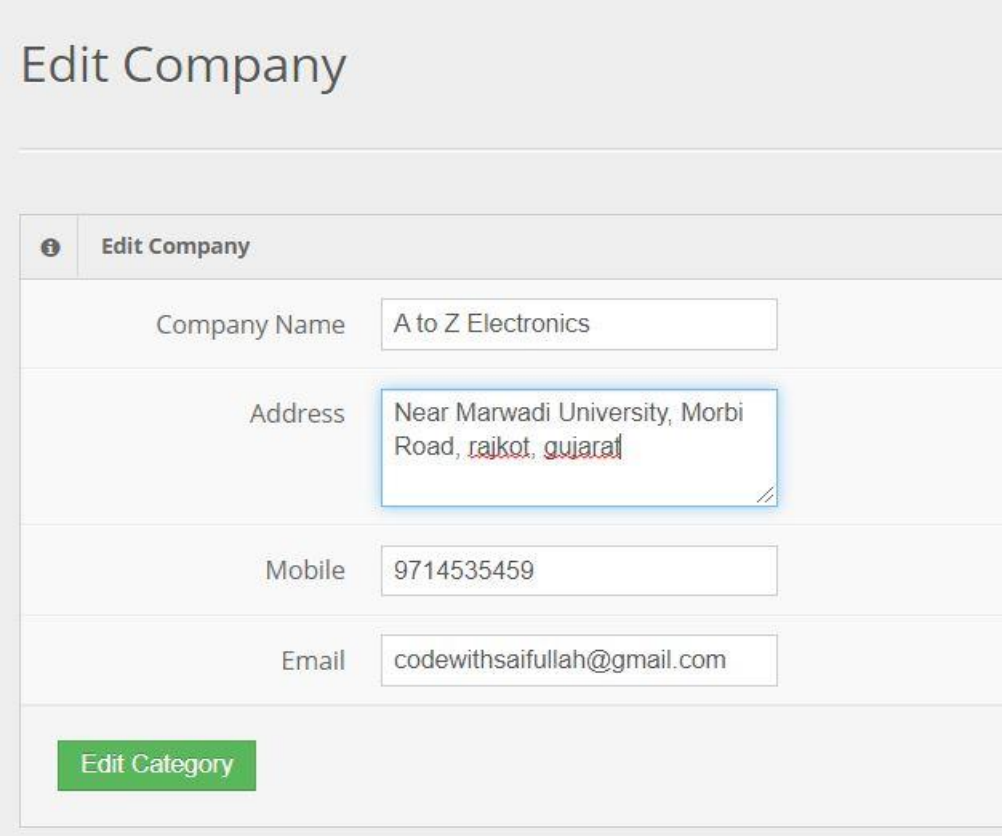
Fiqure: 4.2.22 Vew Sales Return Order

Insert Company:



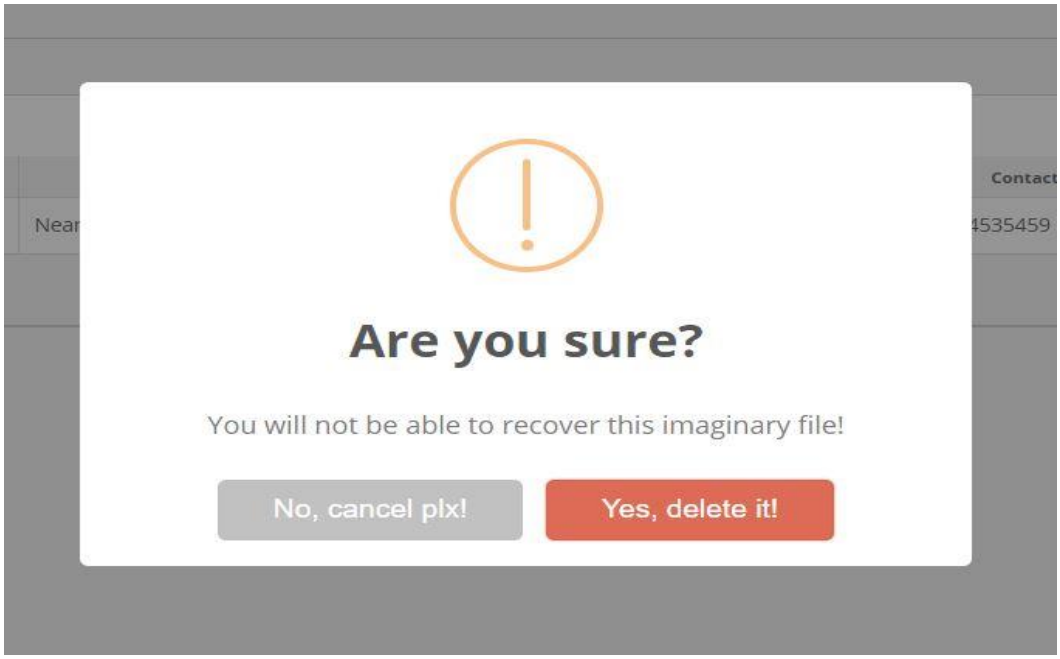
Fiqure: 4.2.23 Add Company

Update company details:



Fiqure: 4.2.24 Edit Company

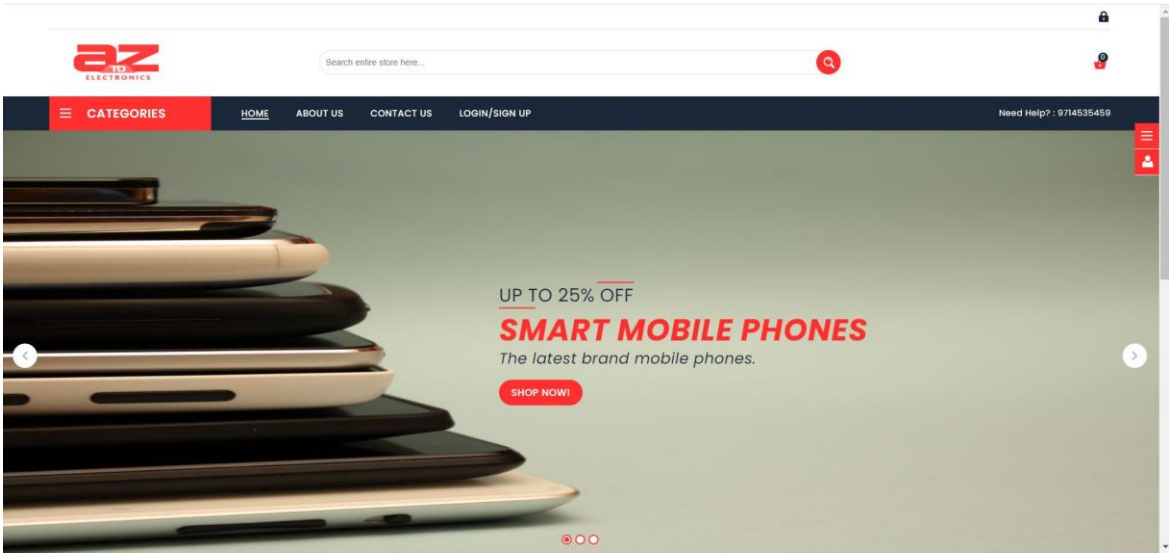
Delete Company:



Fiqure: 4.2.25 Delete Company

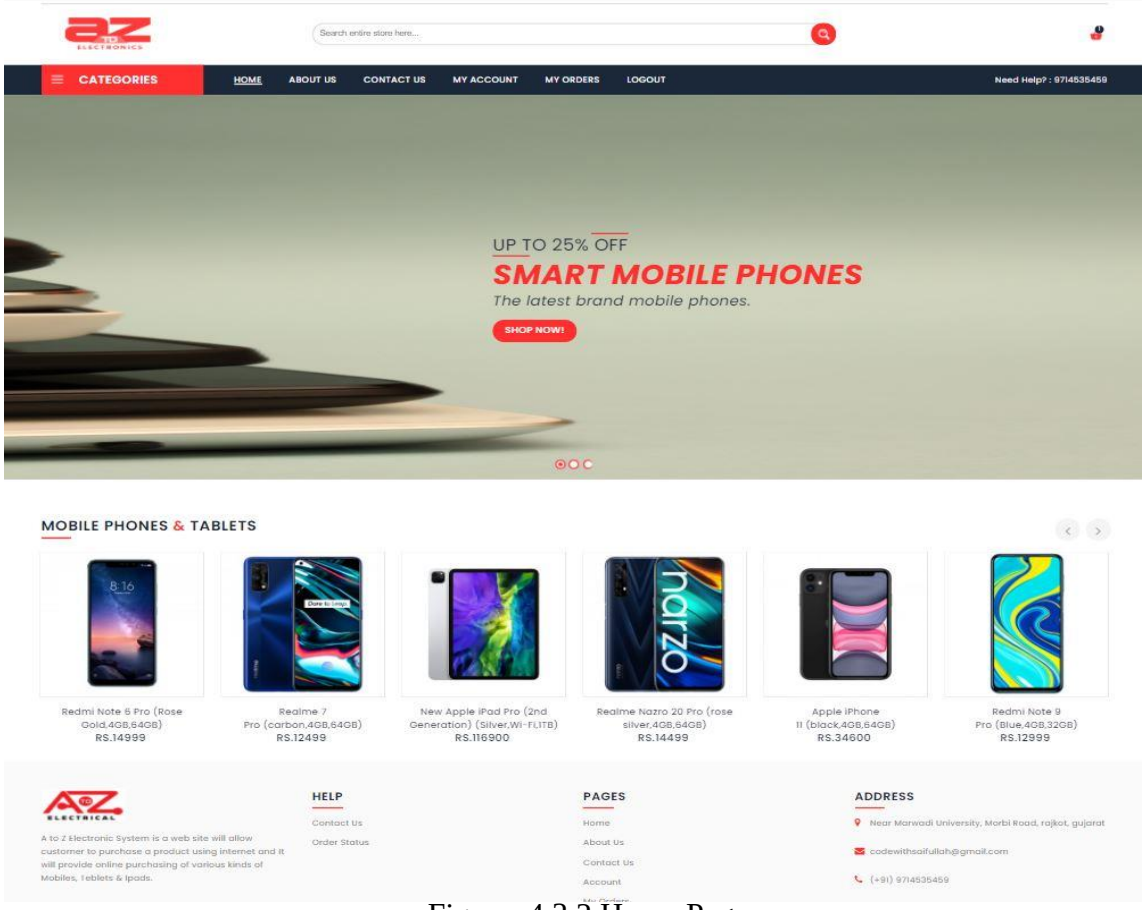
4.3 Visitor Screen Layouts

Header:



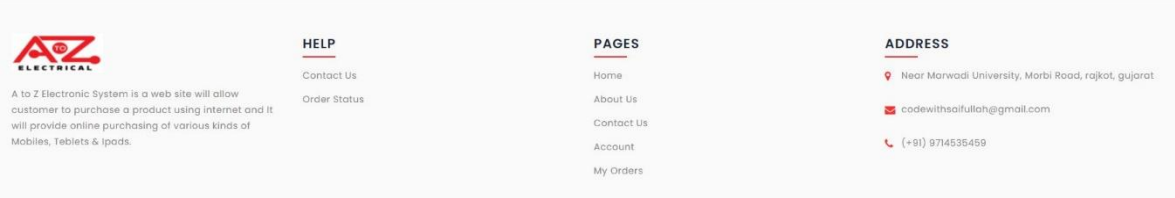
Fiqure: 4.3.1 Header

Home Page:



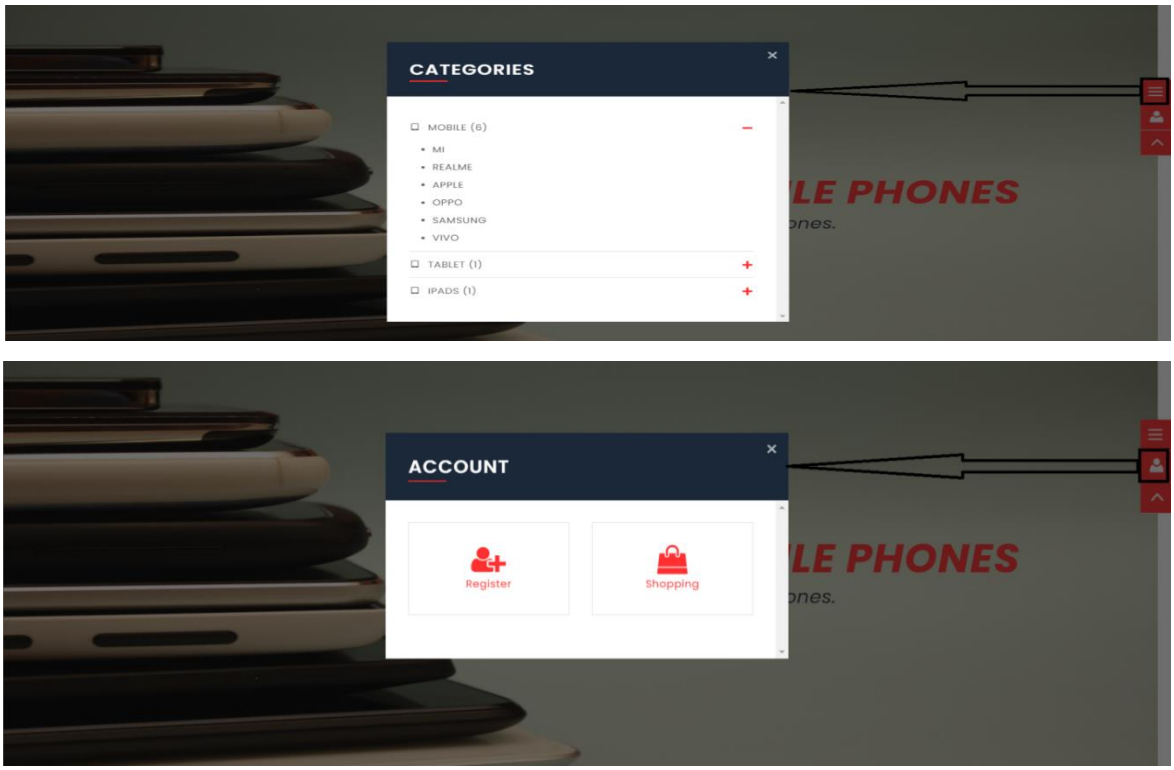
Fiqure: 4.3.2 Home Page

Footer:



Fiqure: 4.3.3 Footer

Popup form:

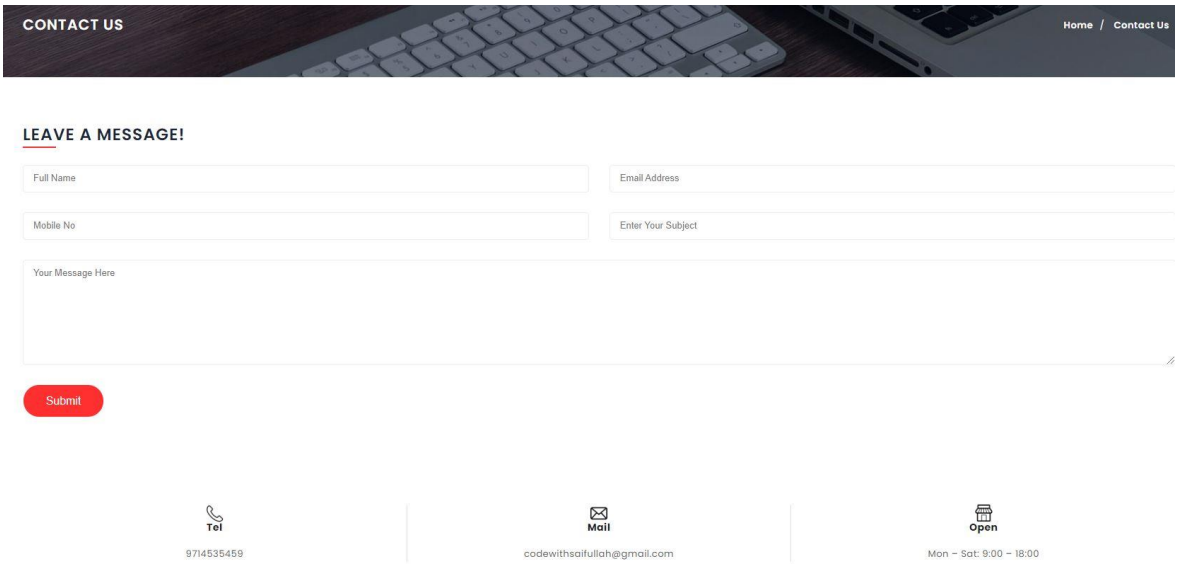


Fiqure: 4.3.4 Popup Form

About Us:

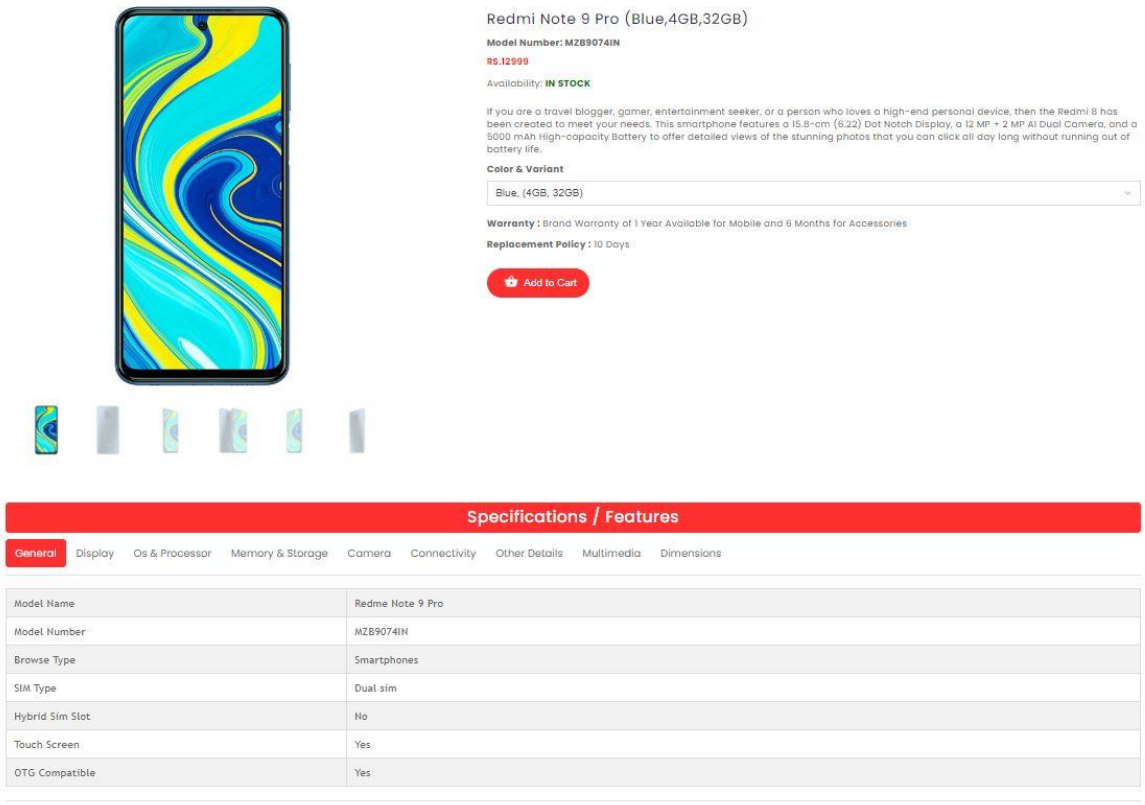
Fiqure: 4.3.5 About Us

Contact Us:



Fiqure: 4.3.6 Contact Us

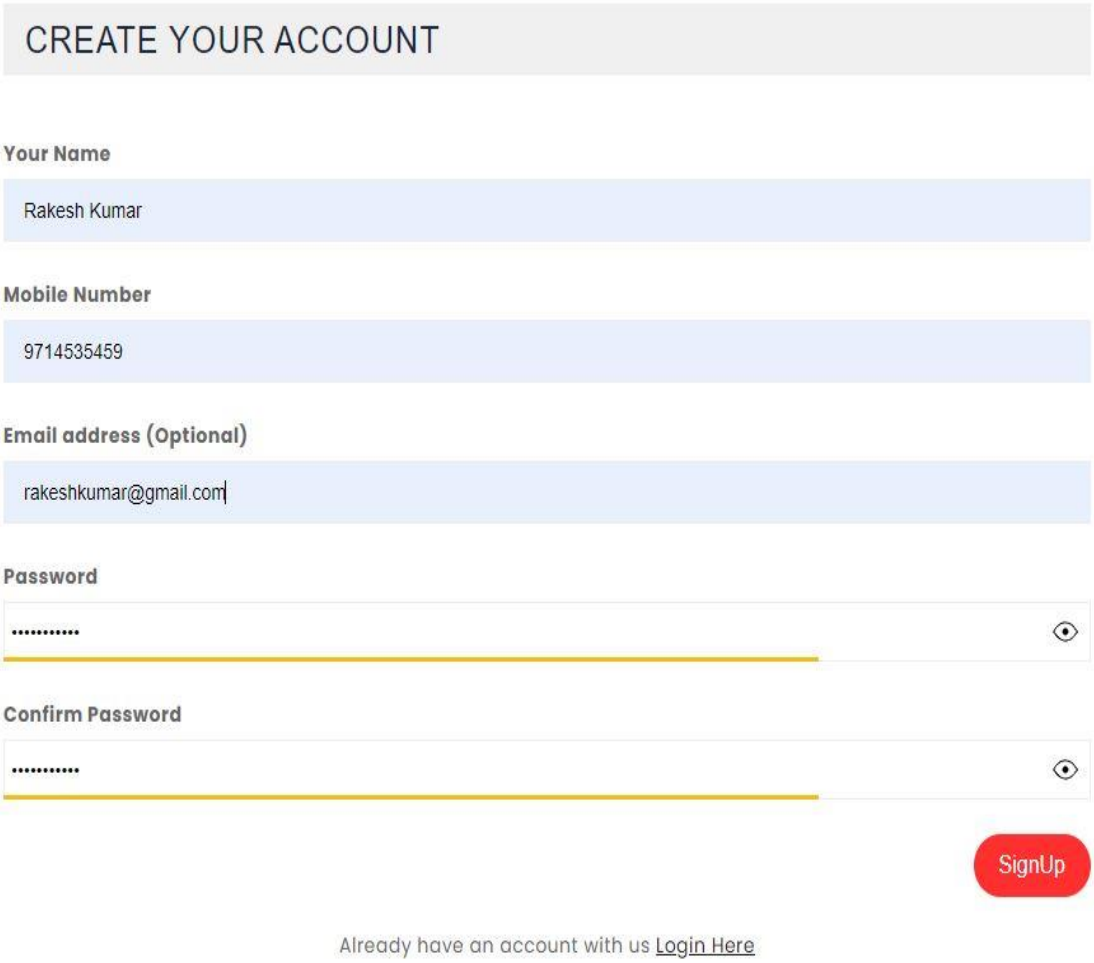
Product Detail:



Fiqure: 4.3.7 Product Details

4.4 CUSTOMER SCREEN LAYOUTS

Customer Signup:



Fiqure: 4.4.1 Customer Signup

Signup Form Validation:

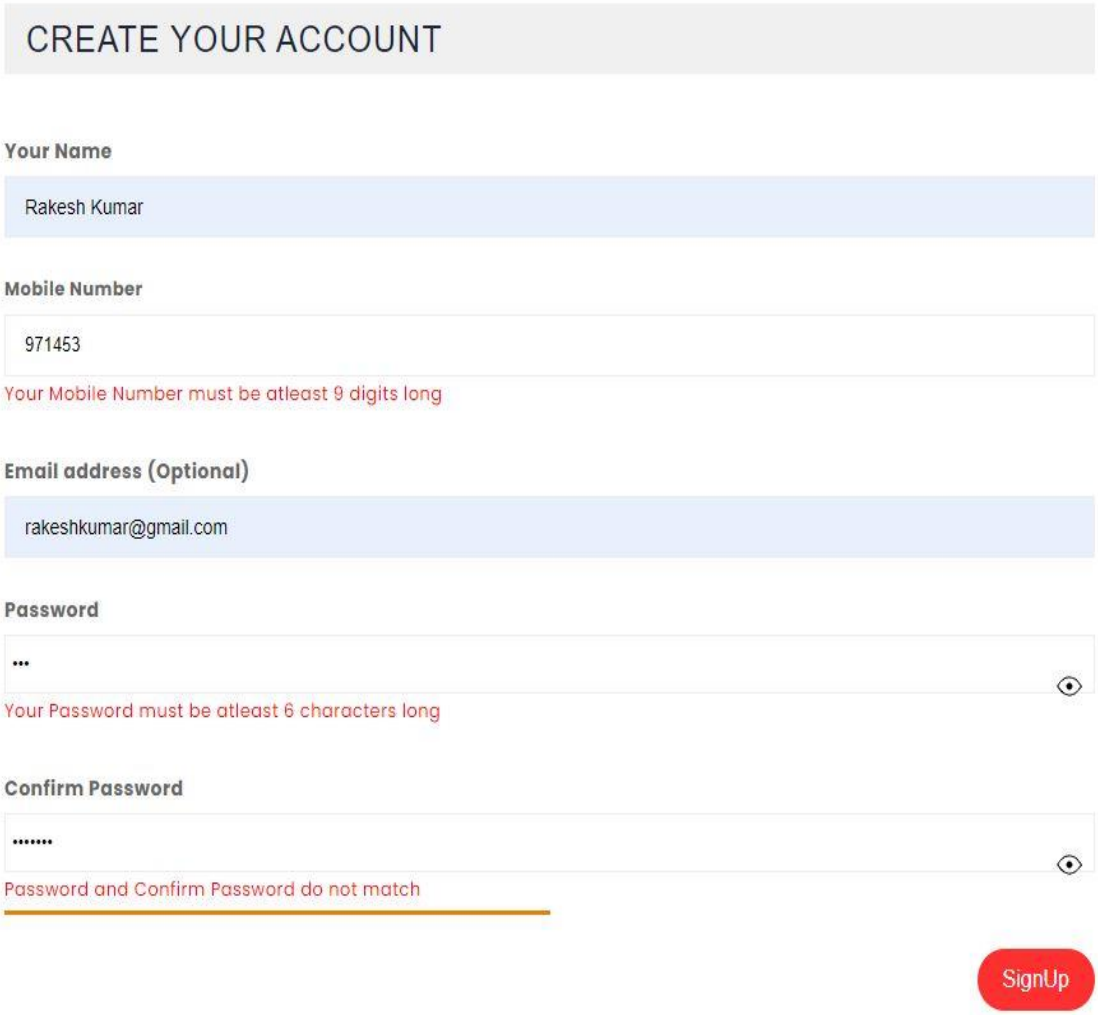


Figure: 4.4.2 Form Validation

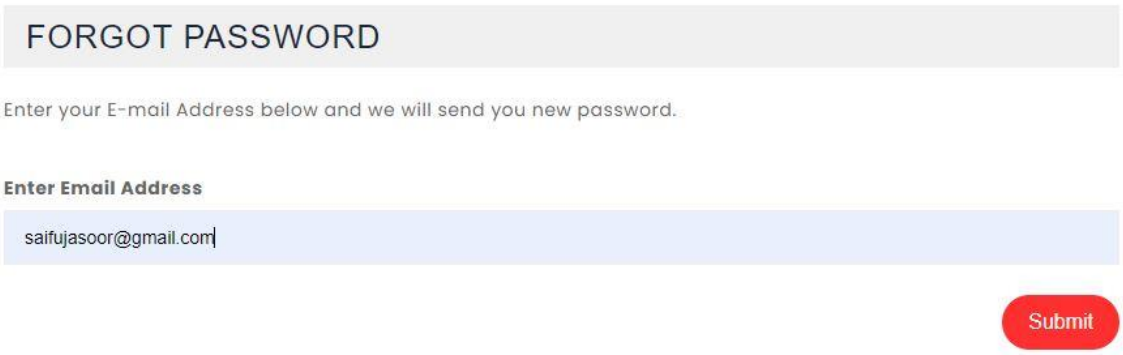
Customer Login:

Fiqure: 4.4.3 Cusomter Login

Login Validation:

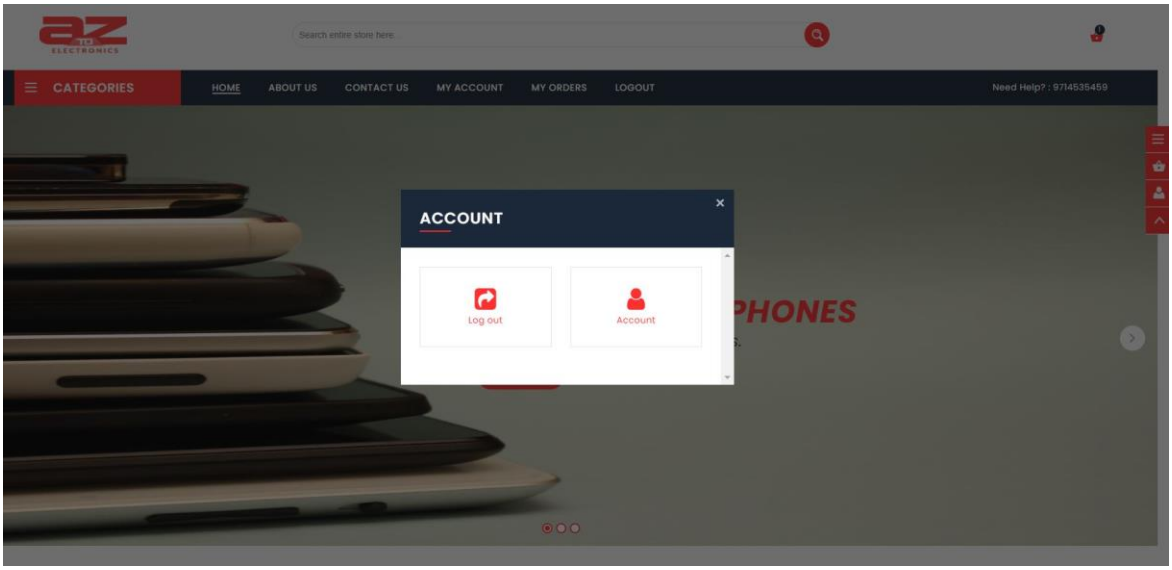
Fiqure: 4.4.4 Login Validation

Forgot Password:



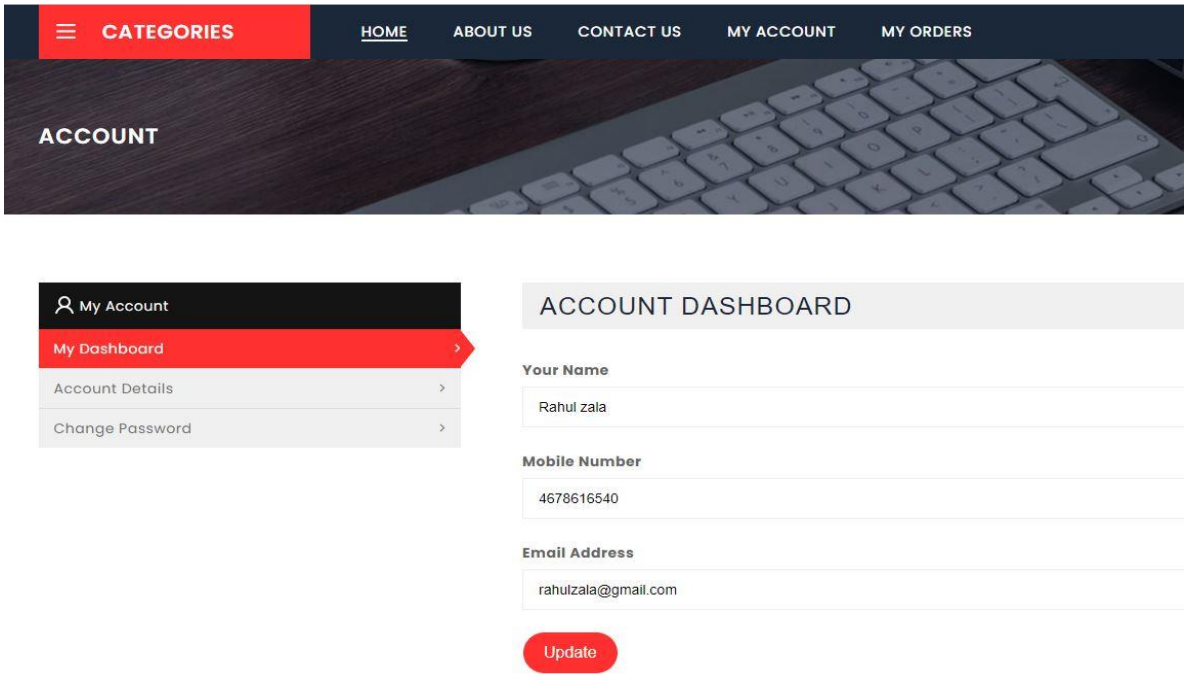
Fiqure: 4.4.5 Forgot Password

Home Page after Login:



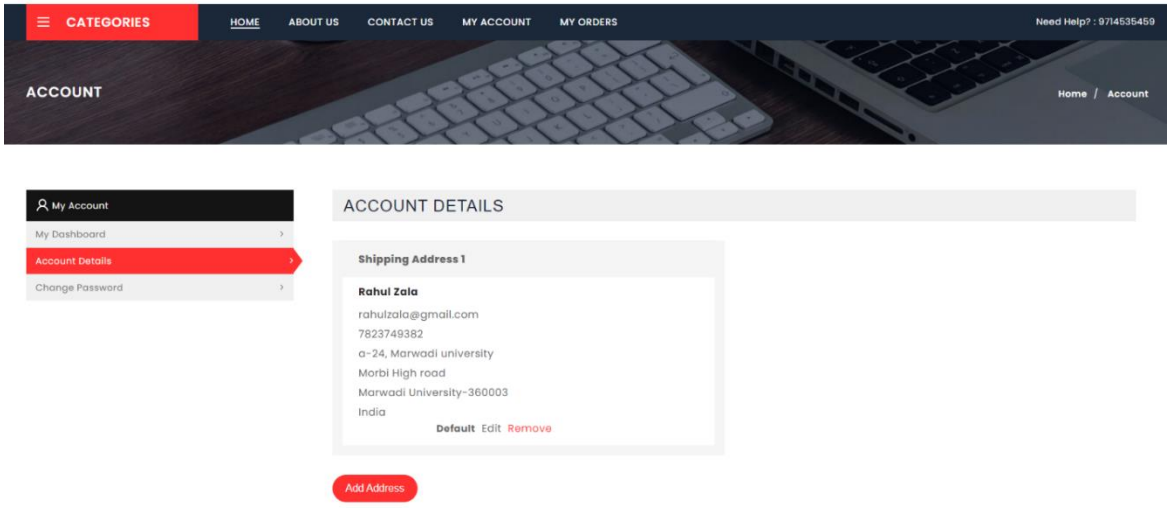
Fiqure: 4.4.6 Home Page After Login

User Dashboard:



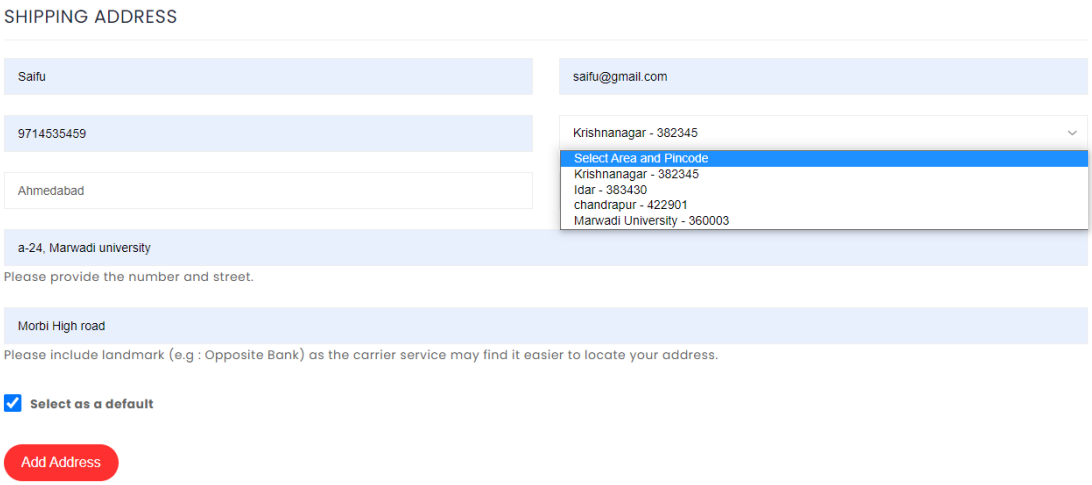
Fiqure: 4.4.7 User Dashboard

Account Details:



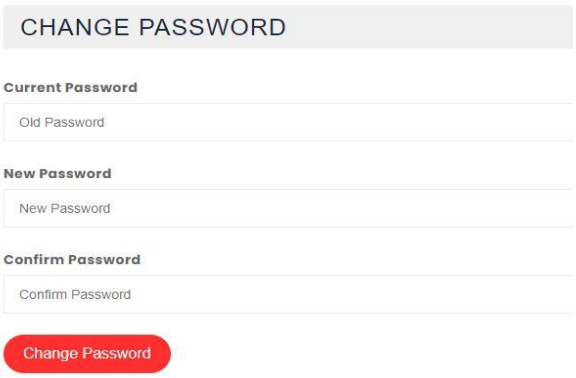
Fiqure: 4.4.8 Account Details

Add Shipping Address:



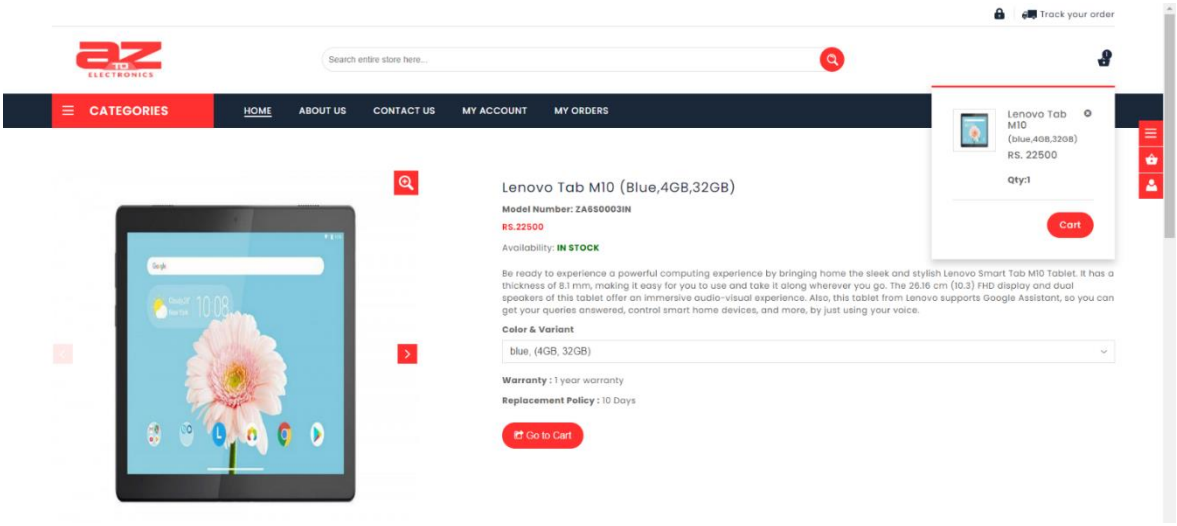
Fiqure: 4.4.9 Shipping Address

Change Password:



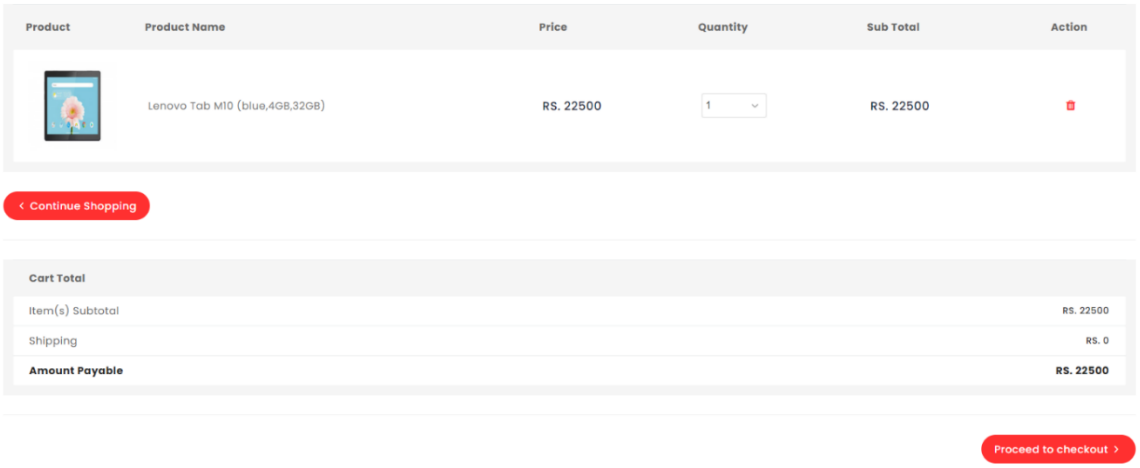
Fiqure: 4.4.10 Change Password

Product – Add to Cart:



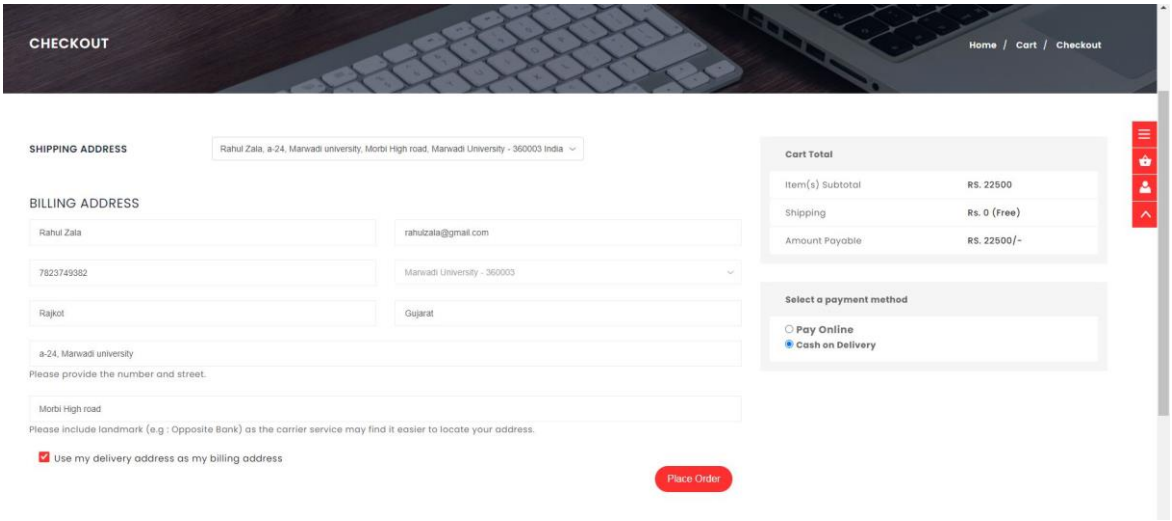
Fiqure: 4.4.11 Add to Cart

Shopping Cart:



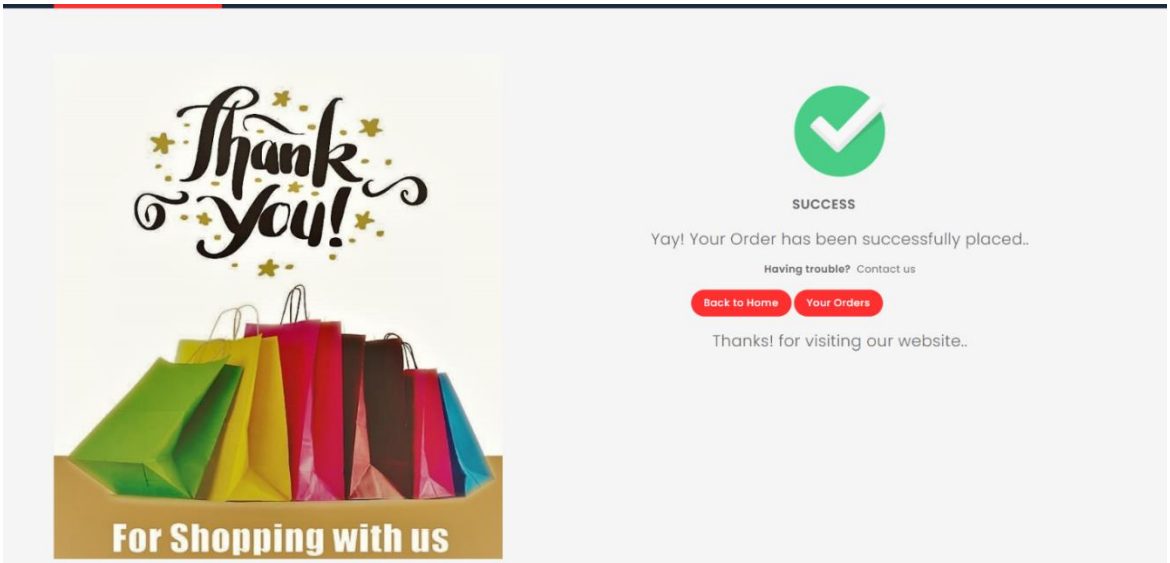
Fiqure: 4.4.12 Shopping Cart

Checkout Page:



Fiqure: 4.4.13 Checkout Page

Order Placed Successfully –Thank You Page:

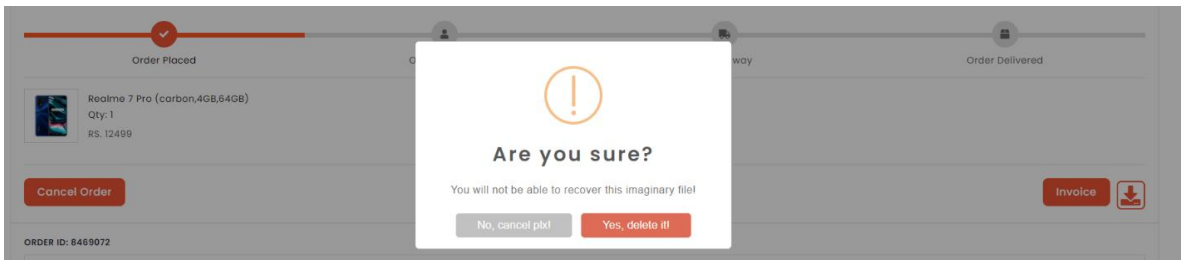


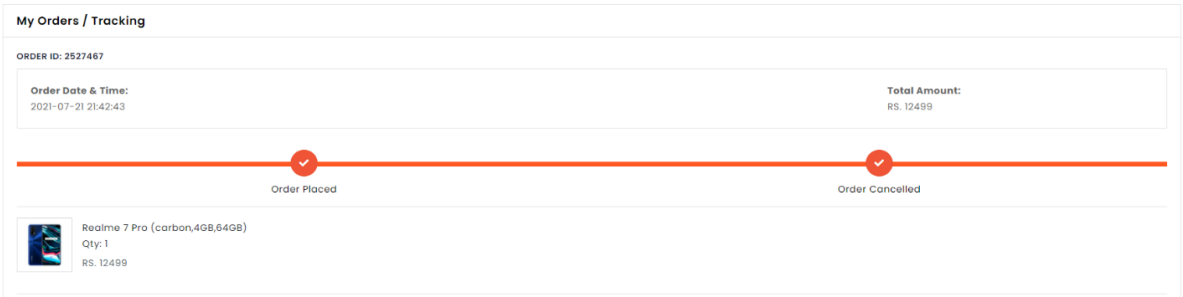
Fiqure: 4.4.14 Oder Placed

Place Order – Tracking Details:

Fiqure: 4.4.15 Order Tracking

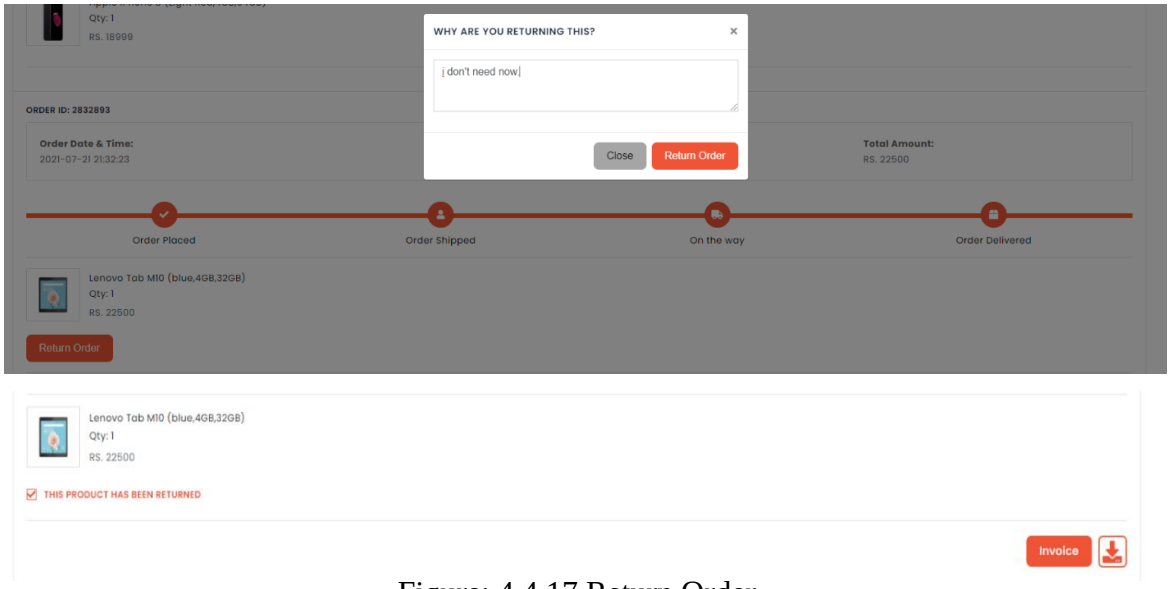
Cancel Order:





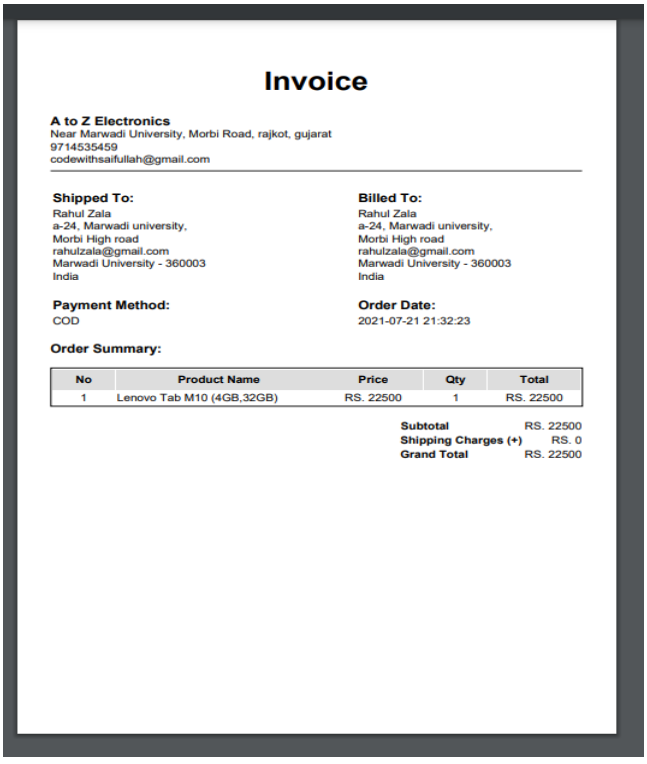
Fiqure: 4.4.16 Cancel Order

Return Order:



Fiqure: 4.4.17 Return Order

Invoice:



Invoice

A to Z Electronics
Near Marwadi University, Morbi Road, rajkot, gujarat
9714535459
codewithsaifullah@gmail.com

Shipped To:
Rahul Zala
a-24, Marwadi university,
Morbi High road
rahulzala@gmail.com
Marwadi University - 360003
India

Billed To:
Rahul Zala
a-24, Marwadi university,
Morbi High road
rahulzala@gmail.com
Marwadi University - 360003
India

Payment Method:
COD

Order Date:
2021-07-21 21:32:23

Order Summary:

No	Product Name	Price	Qty	Total
1	Lenovo Tab M10 (4GB,32GB)	RS. 22500	1	RS. 22500

Subtotal RS. 22500
Shipping Charges (+) RS. 0
Grand Total RS. 22500

Fiqure: 4.4.18 Invoice

4.5 Testing And Implementation:

4.5.1 Testing Approaches Used: -

We all have to agree that in today's ever-changing and competitive world, the internet has become an integral part of our lives.

Most of us make our decisions by searching the information on the internet these days. Hence hosting a website is no longer optional but mandatory for all kind of businesses. It is the first step in becoming and staying relevant in the market.

Just having a website is not enough. An organization is needed to develop a website that is informative, accessible and user-friendly. To maintain all these qualities, the website should be well tested, and this process of testing a website is known as web testing.

What is Web Testing?

Web testing is a software testing practice to test the websites or web applications for potential bugs. It's a complete testing of web-based applications before making live.

A web-based system needs to be checked completely from end-to-end before it goes live for end users.

By performing website testing, an organization can make sure that the web-based system is functioning properly and can be accepted by real-time users.

Models of Testing:

There are different models of testing. On the basis of testing methods there are two type of testing:

- **Black-box testing**
- **White-box testing**

Black-box tests are used to demonstrate the software function are operational, that input is properly accepted and output is correctly produced, and that integrity of external information is maintained.

White-box tests are used to examine the procedural details. It checks the logical paths by test case. It can also check the conditions, loops used in the software coding. It checks that loops are working correctly on defined boundary values.

1) Black-box testing

Black-box testing focuses on the functional requirement of the software. The black-box testing enables the software engineer to drive sets of input condition that will fully exercise all functional.

Requirements for the program. Black-box testing is not an alternative to white-box testing techniques. Rather; it is complementary approach that is likely to uncover a different class of error than white-box methods.

We use in our coding to find error in the following categories

- Incorrect or missing functions.
- Interface errors.
- Performance errors.
- Initialization and termination errors.

Unlike White-box testing, which is performed earlier in the testing process black-box testing tends to be applied during later stages of testing. Because black-

box testing purposely disregards control structure, attention is focused on the information domain.

By applying black-box techniques, we derive a set of test cases that satisfy following criteria.

Typical Black box test design technique includes: -

- Functional Testing
- Non-Functional Testing
- Regression Testing

2) White-box testing

White-box testing sometime called glass-box testing, is a test case design method that users the control structure of the procedural design to drive the test case.

Always we are thinking that is no execute or checks the loops and Conditions. And so large number of error is uncovered.

- All independent paths within a function have been executed at least once.
- All logical decisions on their true and false side.
- All loop working correctly at their boundary values and within their specified conditions.

In SDMS coding we test that all loop works truly in each module. The one technique of white-box testing is basis path testing

Typical White box test design technique includes: -

- Control testing
- Data flow testing

4.5.2. Test cases: -

User Test Case: -

Test Id	Purpose	Required Input	Expected Result
1.	Verify the user is able to login	User Email and password	Successful login
2.	Registration user In required field complete	Truth Information	Successful user create
3.	The better user to interface for dropdown, radio button and checkbox etc	right option to select and click	Successful css is work
4.	The submit page and reset page to redirect to page	Click to reset and submit button or refresh page	Successful link is work
5.	the database with connectivity and operation to perform	Internet is require And right information	Successful database response
6.	The user is change the password	Old password is require and login	Successful change to password
7.	Verify to the client side validation of basic input	Like email , phone, payment etc	Successful validation is response
8.	Registration form in field is blank put given error	Blank required field	Successful given to error
9.	Database with data mismatch and database given error	Wrong information	Successful given to database response
10.	forgotten password to link click after sent email to user	Given to email right for email	Successful sent the email To user
11.	Verify to the numeric and special character in field	Email,phon,address	Successful field is complete
12.	Product is Add cart to proper way and see the cart option	Select product and login	Successful see to the cart
13.	Address in complete all field with detail verified	Right address	Successful verify address and store to the database
14.	Successful payment to online method	Debit card ,credit card ,net banking, Upi ,paythm	Successful payment by server
15.	User place order and bill generate	Successful payment	Successful generate bill
16.	Tack to order and give information	Login	Successful server is response

Table: 1.9

Admin Test case: -

Test Id	Purpose	Required Input	Expected Result
1.	Verify the admin is able to login	User Email and password	Successful login
2.	Registration company By admin	Login with emails	Successful admin create Company
3.	Admin add product and details	Information and image	Successful add product database and client side
4.	Add pin code, state, city.	Information	Successful store database and client side
5.	View the user contact information	Admin login	Successful database response
6.	The Admin is change the password	Old password is require and login	Successful change to password
7.	Verify to the Admin side validation of basic input	Product details ,image.	Successful validation is response
8.	Add product form in field is blank put given error	Blank required field	Successful given to error
9.	Database with data mismatch and database given error	Wrong information	Successful given to database response
10.	Place order information	Login	Successful sent the information To user
11.	Verify to the numeric and special character in field	Email,phon,address	Successful field is complete
12.	Add category wise product	Product information	Successful add to category
13.	Product details to edit	Product information	Successful edit product details
14.	Successful payment to online method	Debit card ,credit card ,net banking, Upi ,paythm	Successful payment by server
15.	Delivery information	Delivery information	Successful response to database
16.	Tack to order and give information	Login	Successful server is response

Table: 1.10

4.5.3. Implementation Approaches: -

Initiation Phase:

Throughout the initiation phase, project management consultants use pilot projects to build process momentum, overcome natural resistance to change, and gain first-hand knowledge of your . The goal of this phase is to successfully mobilize your, remediate any current at-risk projects, and set the stage for the next two installation phases. During this phase, the project management methodology is introduced and software training is conducted; but only for those individuals who will be specifically associated with pilot project teams. Also, a plan for the project-level installation phase is developed and key tools are created that will be utilized during the remaining installation phases.

Project-Level initiation Phase:

The second phase utilizes information gathered from pilot projects in the initiation phase to roll-out structured project planning and control processes for all remaining projects, as well as to formally establish the Project Management Office.

This phase can include the creation of PMO job descriptions, formal guidelines for project planning/control, a project web site, and a web-based activity update system - basically the necessary infrastructure to support the consistent, successful application of project management techniques by the PMO.

Project Management Training is also rolled-out to the entire during the project-level installation phase. By the conclusion of this phase, the nucleus of a Project Management Office is in-place, all project team members have been trained, and the project management consultants are ready to begin transitioning from their role of supporting project team requirements to supporting the PMO staff.

Enterprise-Level Installation Phase:

During the enterprise-level installation phase, tools are implemented that are focused on managing as entire portfolio of projects. Examples of these tools include; enterprise performance metrics, a management "dashboard" to gain summary-level visibility to project status, and project scheduling based on limited resources and project priority (enterprise resource levelling).

The intent of these types of tools is to

(1) Provide management with timely and accurate information about the status of the all the projects being undertaken by the and (2) Support business decision-making that impacts the successful completion of projects such as: changes to staffing, funding, project prioritization, and workload.

During the enterprise-level installation phase, the Project Management Office staff has already begun to assume some of the day-to-day responsibilities for developing and maintaining ongoing project plans.

In doing so, the PMO staff is able to free-up the project management consulting firm to focus on the design and implementation of the enterprise-level tools. By the end of this phase, all responsibility for developing and updating individual project plans have been transitioned from the Project Management Consultants to the PMO staff.

Maintenance Phase:

The final phase marks the important transition of the Project Management Office from the project management consultants back to the. In addition to supporting the day-to-day responsibilities for planning and controlling individual projects, the PMO staff will now become the focal point for providing the enterprise-level information and analysis required by management.

At this point in the project management implementation process, the has been well trained, numerous success stories have been created and communicated, virtually all projects have well-developed project plans, and there is widespread support for investing in a formal project planning and control process.

Also, the Project Management Office infrastructure is in place, the PMO staff has been trained, and management has necessary visibility to the key project portfolio level information.

Successful completion of this phase creates long-term continuity by implementing the necessary policies and incentives to permanently inculcate project management into the culture of the. Ideally, formal project planning and control processes will become recognized as a required core competency and an essential function within the.

Phase 1: Initiation Phase:

- Initial communication(s) to management and assistance in the identification of pilot projects.
- Project management methodology and software training for identified pilot team members.
- Project plans and formal control processes in place for all identified pilot projects.
- Standardized project coding structures and project-level report formats.
- Finalized requirements and a plan for the project-level installation phase.

Phase 2: Project-Level Installation Phase:

- Network-based, structured project plans and formal control process for all targeted projects.
- Rollout of PM/software training to all project leaders and team members.
- Training and mentoring of PMO personnel.
- Implementation of the initial PMO infrastructure.
- Finalized requirements and a plan for the enterprise-Level implementation phase.

Phase 3: Enterprise-Level Installation Phase:

- Implementation of the enterprise-level PMO infrastructure.
- Turnover to PMO staff of the day-to-day responsibility for developing and maintaining individual project plans.
- Finalized requirements and a plan for the maintenance phase.

Phase 4: Maintenance Phase:

- Turnover to Project Management Office staff the responsibility for supporting all of the project management requirements.
- Recommendations to management for policies and incentives required to permanently establish project management as a core competency and essential function.

5. Proposed Enhancements:

We think that not a single project is ever considered as complete forever because our mind is always thinking new and our necessities also are growing. Our system Also, if you see at the first glance that you find it to be complete but we want to make it still mature and fully automatic. The system is modified in future as per the owner requirement.

6. Conclusion:

There may be some error or some defect in the work. I have taken enough care to make the project user friendly and more interactive. I have tried my level best to make the service useful and to provide maximum facility but I never claim that this system may fulfil all requirement in every condition. I hereby conclude these reports which give information regarding the development and the user of application with an overview of different aspect of it.

7. Bibliography:

Bibliography means that we have referred while doing project like, which sites we gone through and which books we have referred or which magazines we have seen to implements this project. There are many books and websites that can help us proper guild line to implements my system in the right direction.

References Books:

Laravel

The Ultimate Beginner's Guide to Learn Laravel Step by Step, 2nd Edition

- By: Rufus_Stewart

Websites:

- ✓ https://laravel.com/
- ✓ https://www.tutorialspoint.com/
- ✓ https://www.w3schools.in/laravel-tutorial/
- ✓ https://www.youtube.com/