DESIGN DOCUMENT

for

E-COMMERCE WEBSITE

Version 1.0

Prepared by : 1.Nitin Krishna Makula (190001033)
2. Revanth Thota (190001063)
3. Sairam Kola(190001026)
4. Aakash Reddy Kondampally (1900010268)
5. Abhiram Khajjayam (190001025)

Submitted to : Dr.Puneet Gupta
May 7, 2021

Contents

1 Introduction **3**
- 1.1 Purpose . . . 3
- 1.2 Scope . . . 3

2 Design Overview **4**
- 2.1 Technologies Used . . . 4
- 2.2 System Architecture . . . 4
- 2.3 System Operation . . . 5

3 User Interface **8**

4 Seller Interface **12**

5 Admin Interface **14**

1 Introduction

1.1 Purpose

The purpose of this document is to describe the implementation of E-commerce website described in the E-commerce software requirements specification.The E-commerce website is an all in one platform for buyers and sellers to come together.

1.2 Scope

This document describes the implementation details of the E-commerce software.This software will consist of two major functions first to design sessions which are made up of a series of tasks and second to perform these sessions to perform the tasks as per the design.This document will not specify any actual sessions or the testing of this software.

2 Design Overview

2.1 Technologies Used

The E-commerce website can be accessed by any user with a browser in his/her operating system which supports java Script and an active internet connection There is no specific target platform for this software as the technologies used to build the software are compatible with most browsers.The technologies used to build the software are

1. **Front-end:** HTML,CSS,Bootstrap,Java Script,React JS
2. **Back-end:** Python,Django
3. **Database:** PostgreSQL
4. **Hosting:** Heroku

2.2 System Architecture

We have used MVC architecture in our website.MVC stands for Model-View-Controller The MVC pattern is a software architecture pattern that separates data presentation from the logic of handling user interactions. It has also been described as one of the best ways to create client-server applications, all of the best frameworks for web are all built around the MVC concept.As we are developing a website MVC seemed to be the best architecture to implement. To break it down, here's a general overview of the MVC Concept:

- **Model:** This handles your data representation, it serves as an interface to the data stored in the database itself, and also allows you to interact with your data without having to get perturbed with all the complexities of the underlying database. In this E-commerce website we use the models made with the ORM feature in **Django** which include the USER model, PRODUCTS model..etc these models server as the interface to the data in the database and reduces the complexity.
- **View:** As the name implies, it represents what you see while on your browser for a web application or In the UI for a desktop application. The views in the E-commerce website are made and handled using the **ReactJS** library which handles all the rendering and routing of the HTML pages being served to the client.

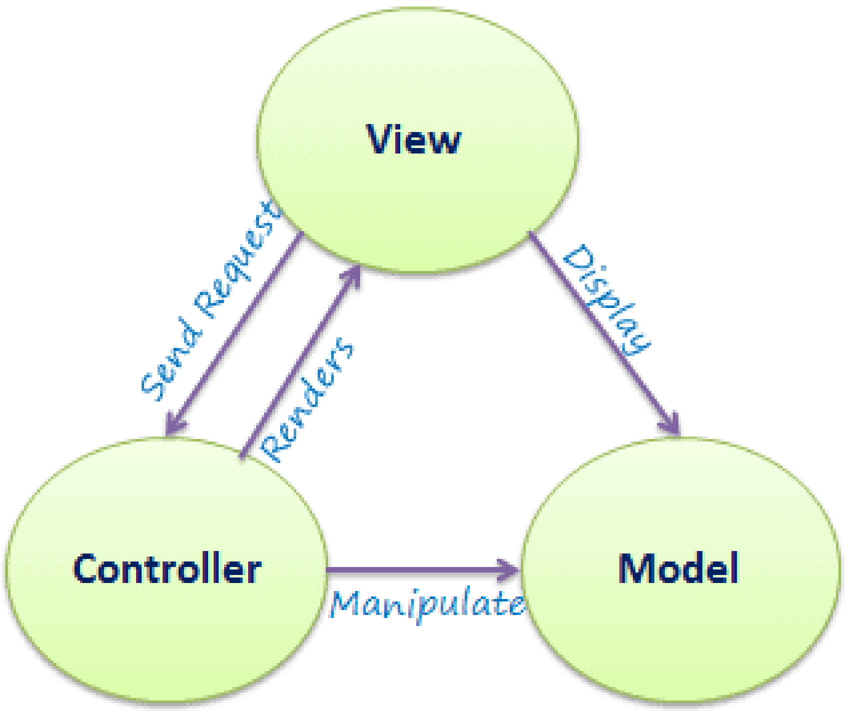


Figure 2.1: Functions Of MVC

- **Controller:** provides the logic to either handle presentation flow in the view or update the model's data i.e it uses programmed logic to figure out what is pulled from the database through the model and passed to the view,also gets information from the user through the view and implements the given logic by either changing the view or updating the data via the model. the Controller in Our website is implemented using the **Django REST framework** as it handles all the API calls to the database and it acts a medium between model and view and fetches data from postgres database and displays it in JSON format in UI which is implemented using ReactJS.

2.3 System Operation

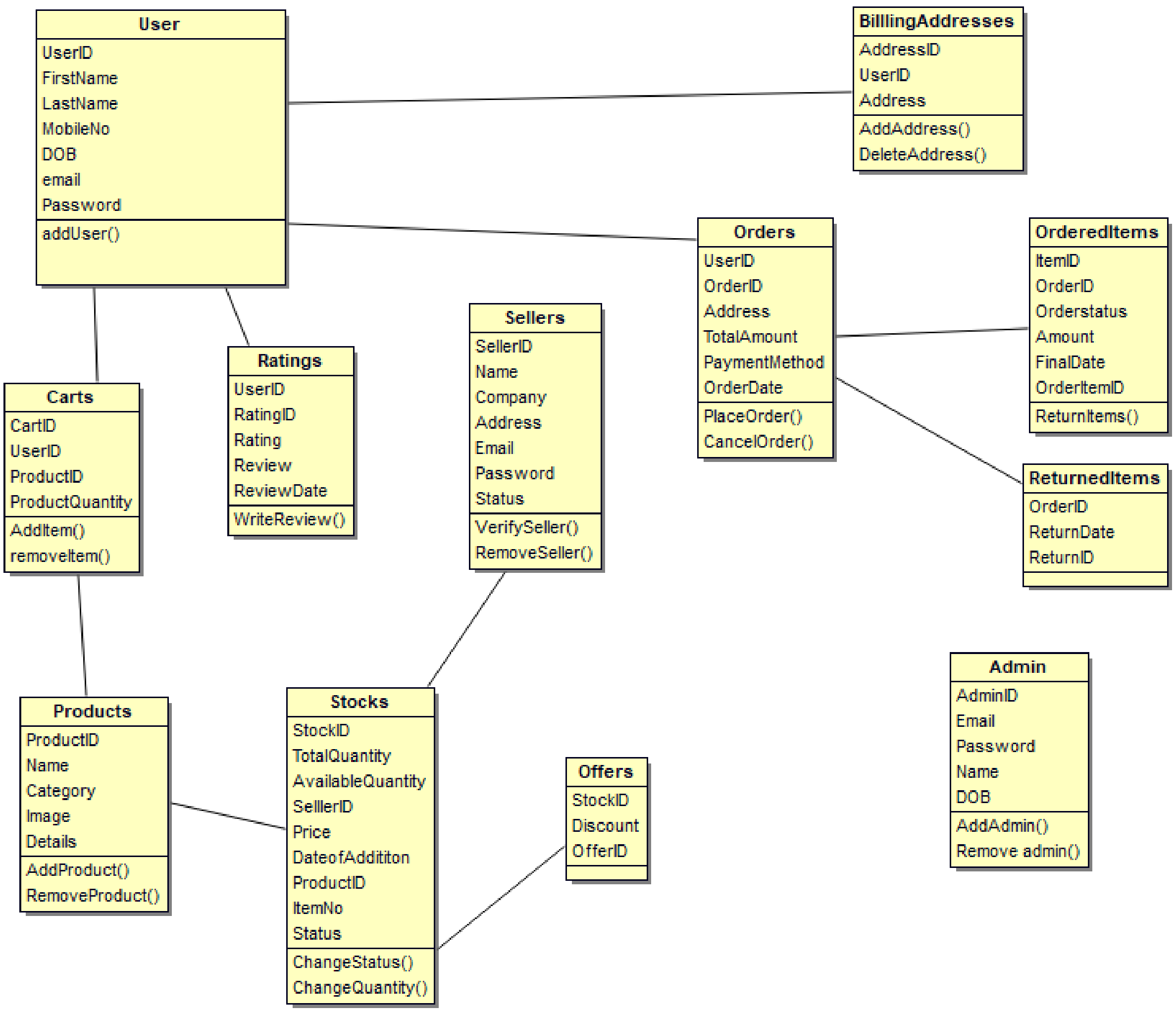


Figure 2.2: Different Tables and their functions

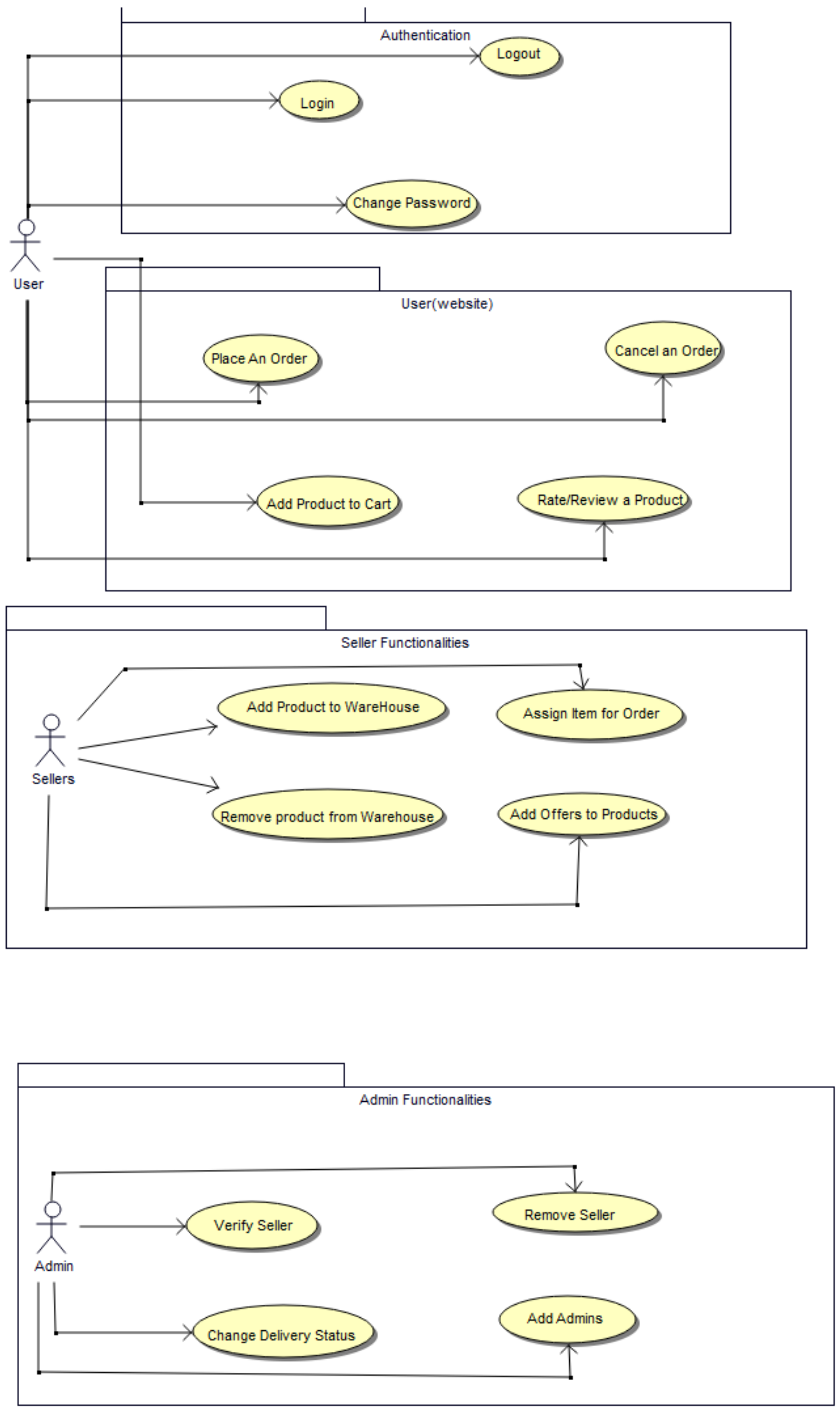


Figure 2.3: Different Types of User and Their Functionalities

3 User Interface

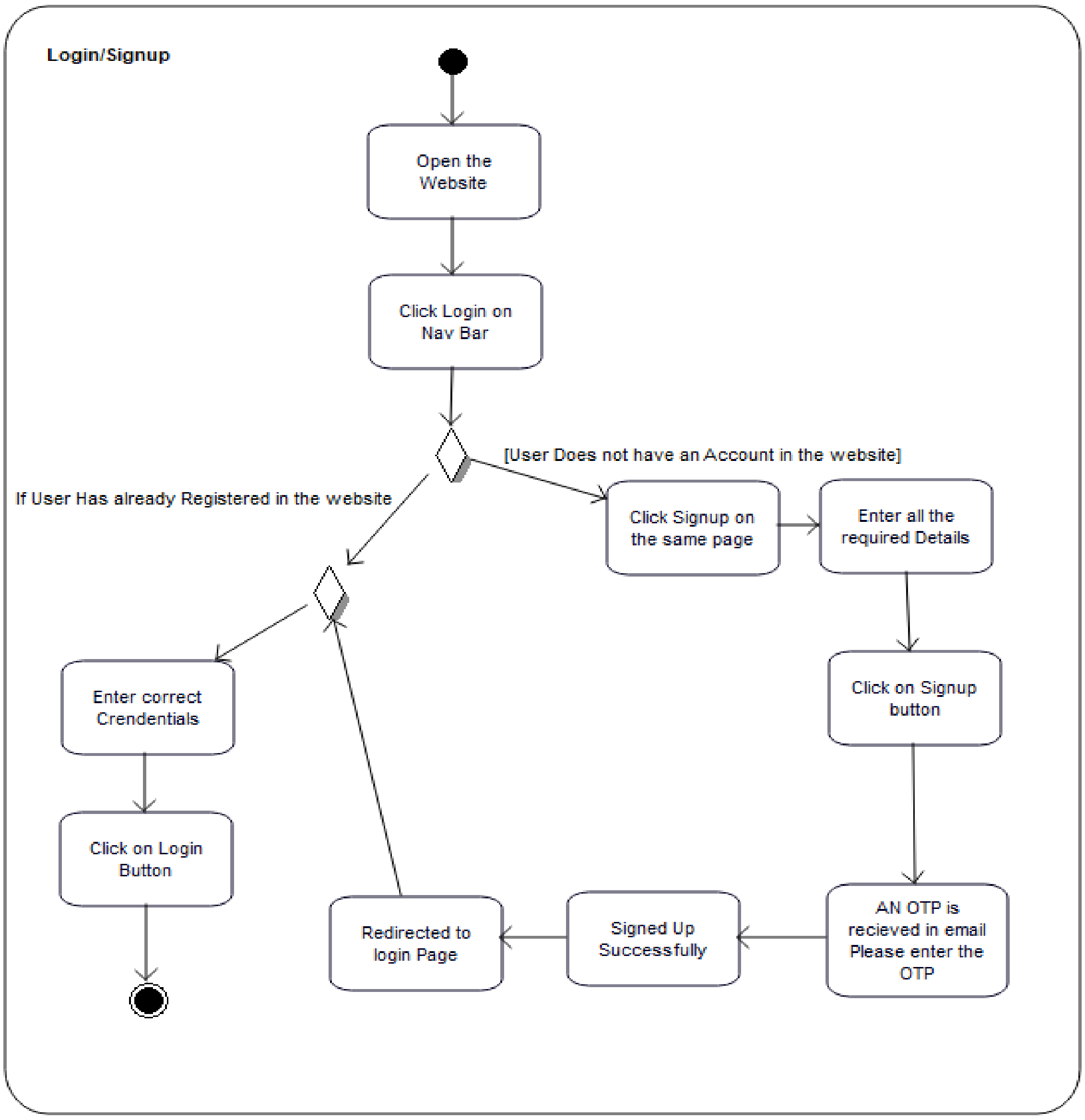
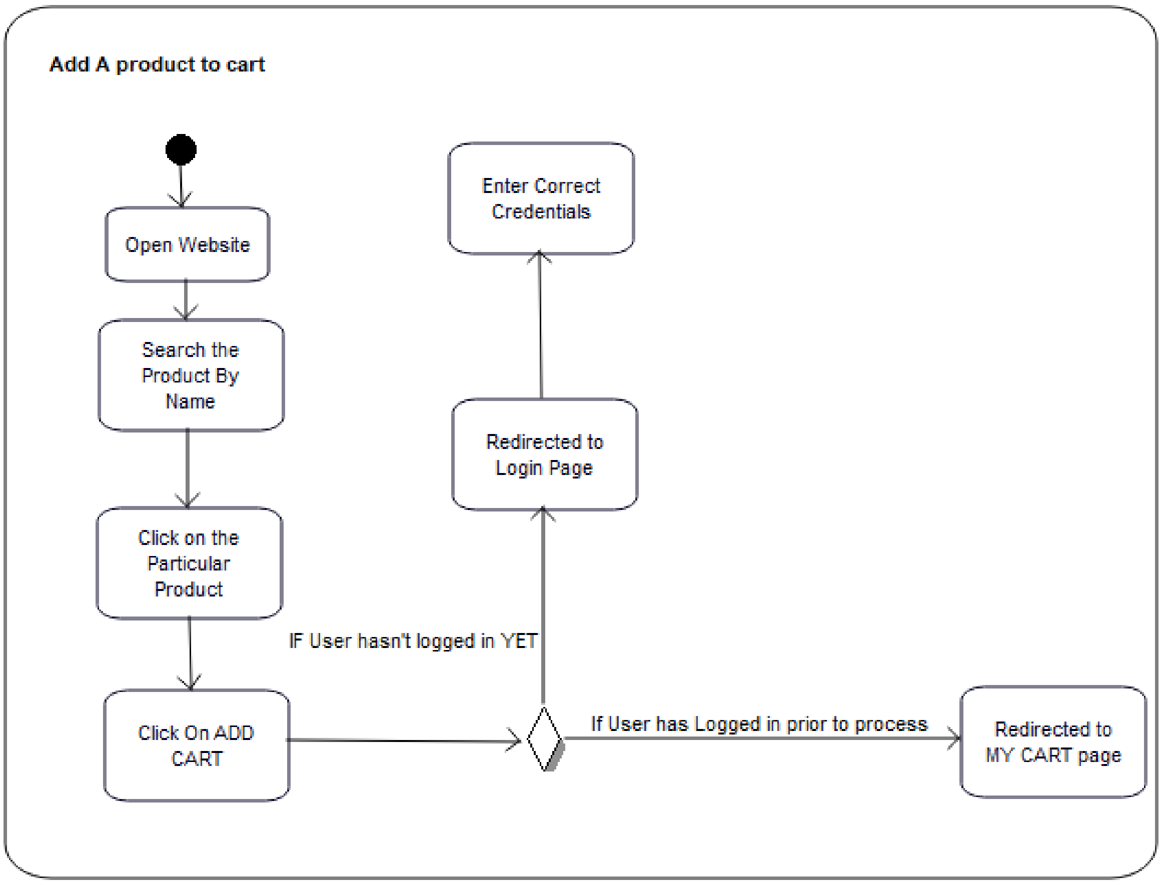

Login/Signup
Open the Website
Click Login on Nav Bar
[User Does not have an Account in the website]
If User Has already Registered in the website
Click Signup on the same page
Enter all the required Details
Click on Signup button
Enter correct Crendentials
Click on Login Button
Redirected to login Page
Signed Up Successfully
AN OTP is recieved in email Please enter the OTP

Figure 3.1: Login/Sign-up



Add A product to cart
Open Website
Search the Product By Name
Click on the Particular Product
Click On ADD CART
Enter Correct Credentials
Redirected to Login Page
IF User hasn't logged in YET
If User has Logged in prior to process
Redirected to MY CART page

Figure 3.2: Adding A Product To CART

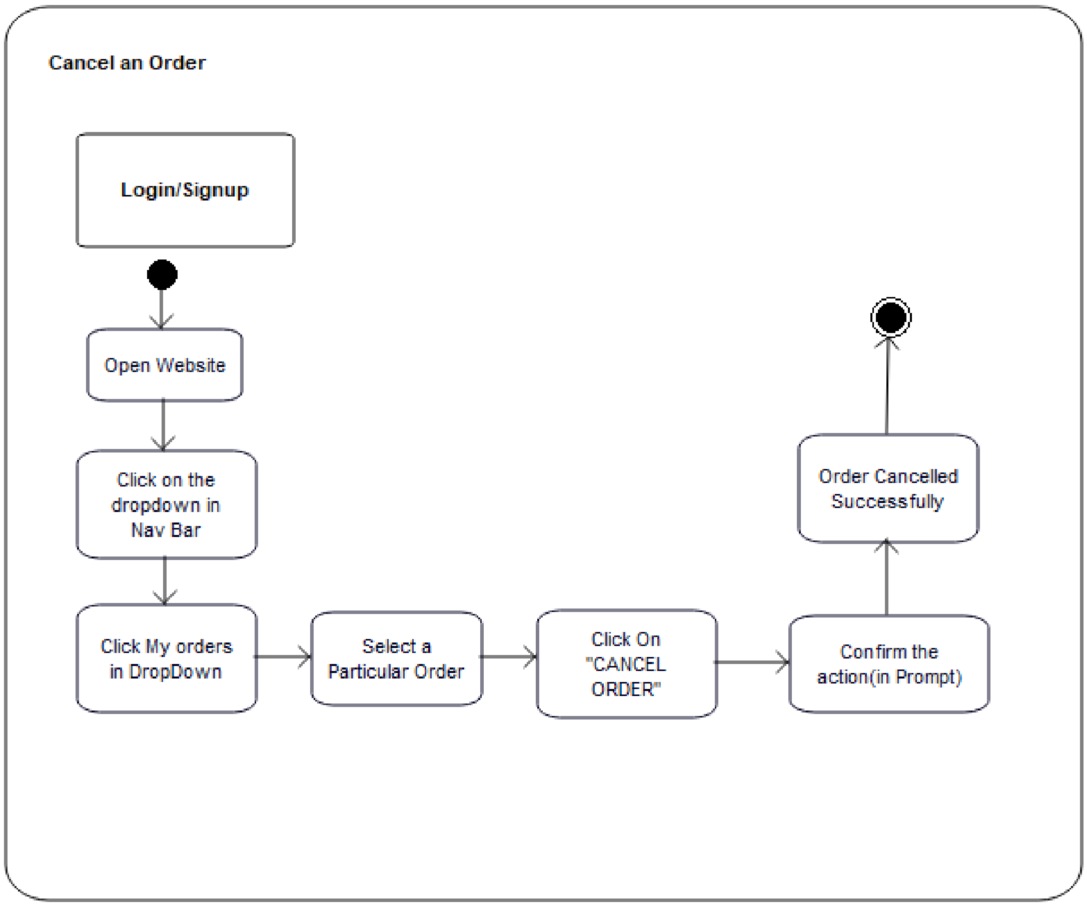

Cancel an Order
Login/Signup
Open Website
Click on the dropdown in Nav Bar
Click My orders in DropDown
Select a Particular Order
Click On "CANCEL ORDER"
Confirm the action(in Prompt)
Order Cancelled Successfully

Figure 3.3: Cancel an Placed Order

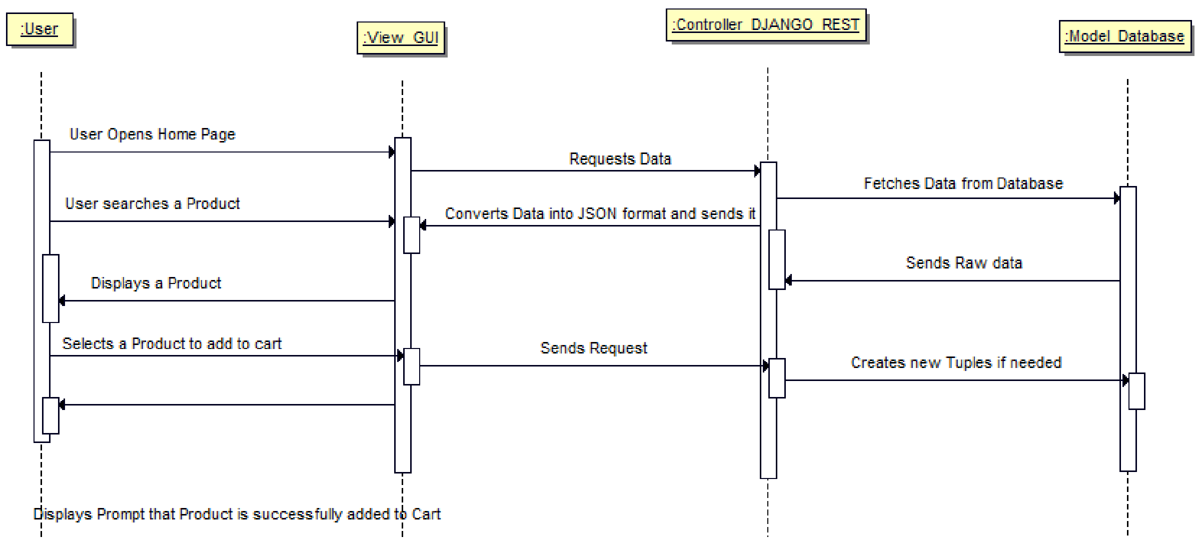
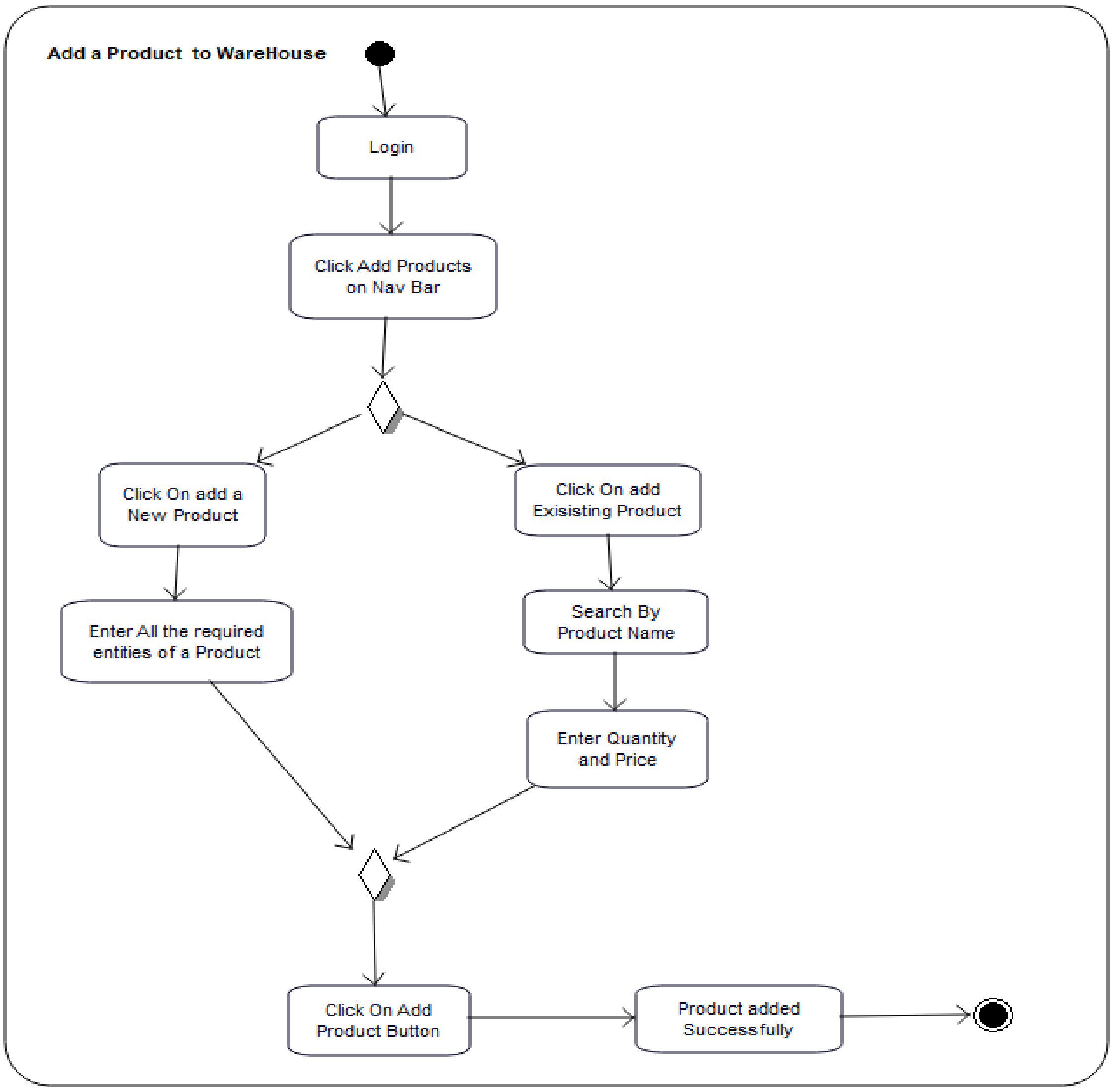

:User
:View_GUI
:Controller_DJANGO_REST
:Model_Database
User Opens Home Page
Requests Data
Fetches Data from Database
User searches a Product
Converts Data into JSON format and sends it
Sends Raw data
Displays a Product
Selects a Product to add to cart
Sends Request
Creates new Tuples if needed
Displays Prompt that Product is successfully added to Cart

Figure 3.4: Sequence diagram for Adding a Product To Cart

4 Seller Interface



Add a Product to WareHouse
Login
Click Add Products on Nav Bar
Click On add a New Product
Click On add Exisisting Product
Enter All the required entities of a Product
Search By Product Name
Enter Quantity and Price
Click On Add Product Button
Product added Successfully

Figure 4.1: Add A product To Warehouse

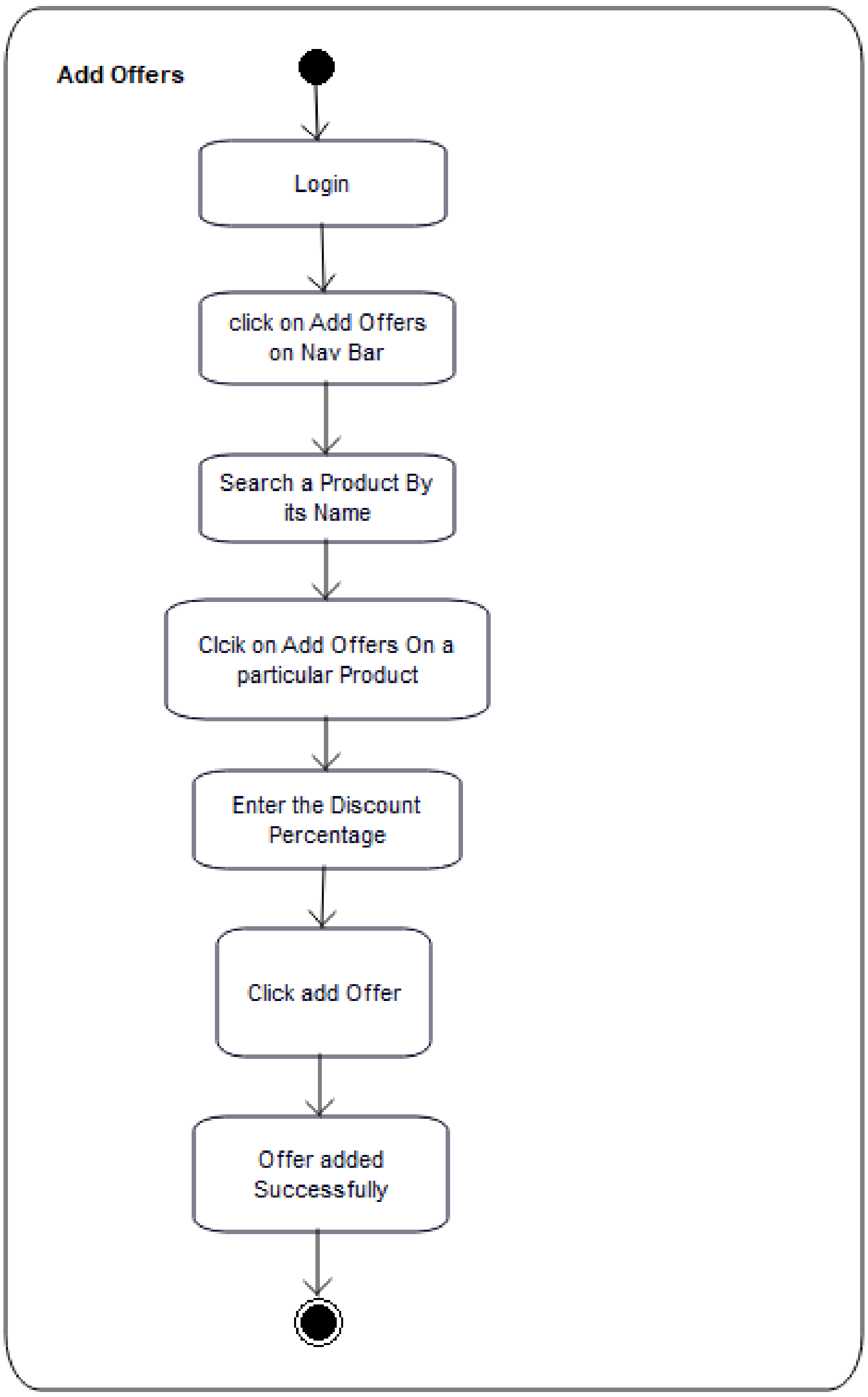

Add Offers
Login
click on Add Offers on Nav Bar
Search a Product By its Name
Clcik on Add Offers On a particular Product
Enter the Discount Percentage
Click add Offer
Offer added Successfully

Figure 4.2: Include Offer For a Product

5 Admin Interface

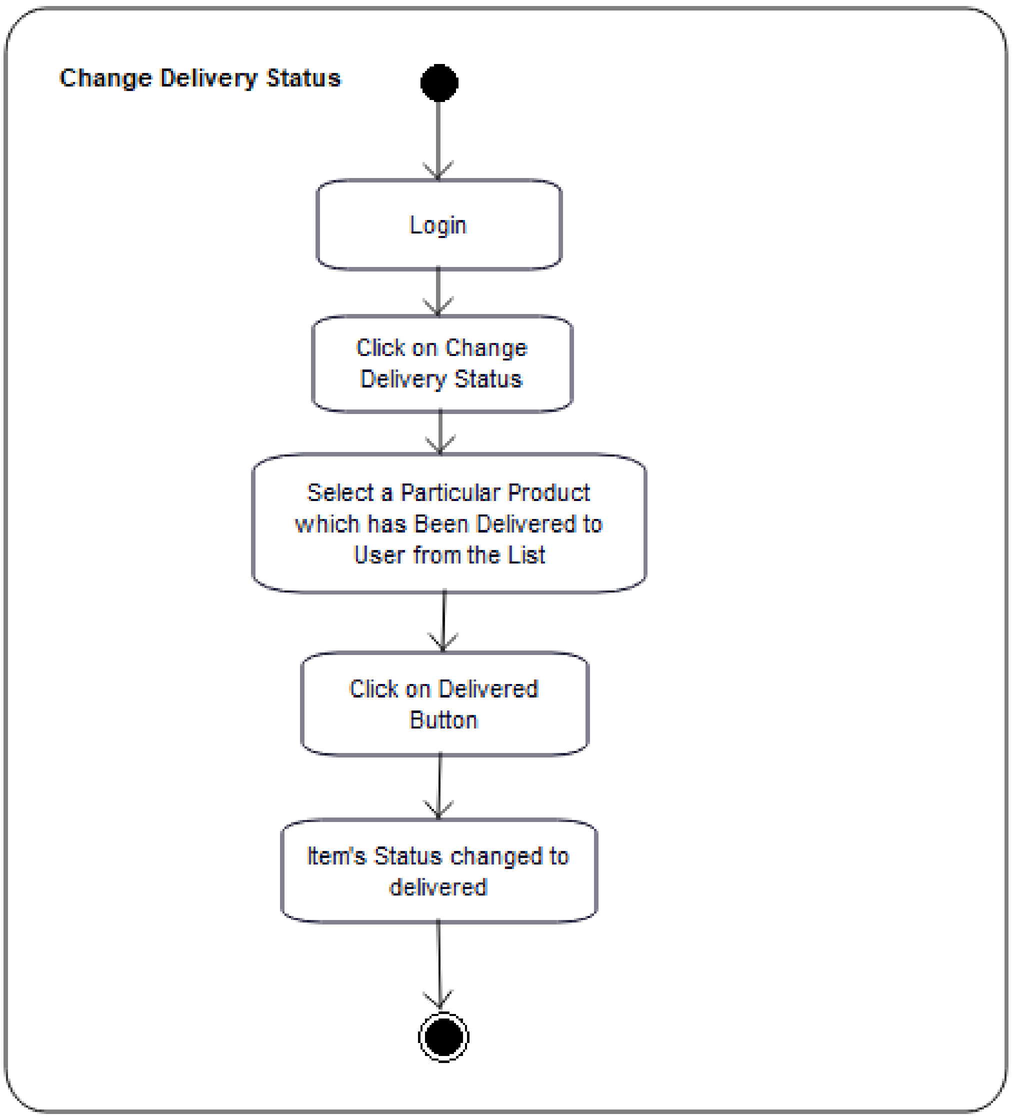

Change Delivery Status
Login
Click on Change Delivery Status
Select a Particular Product which has Been Delivered to User from the List
Click on Delivered Button
Item's Status changed to delivered

Figure 5.1: Caption

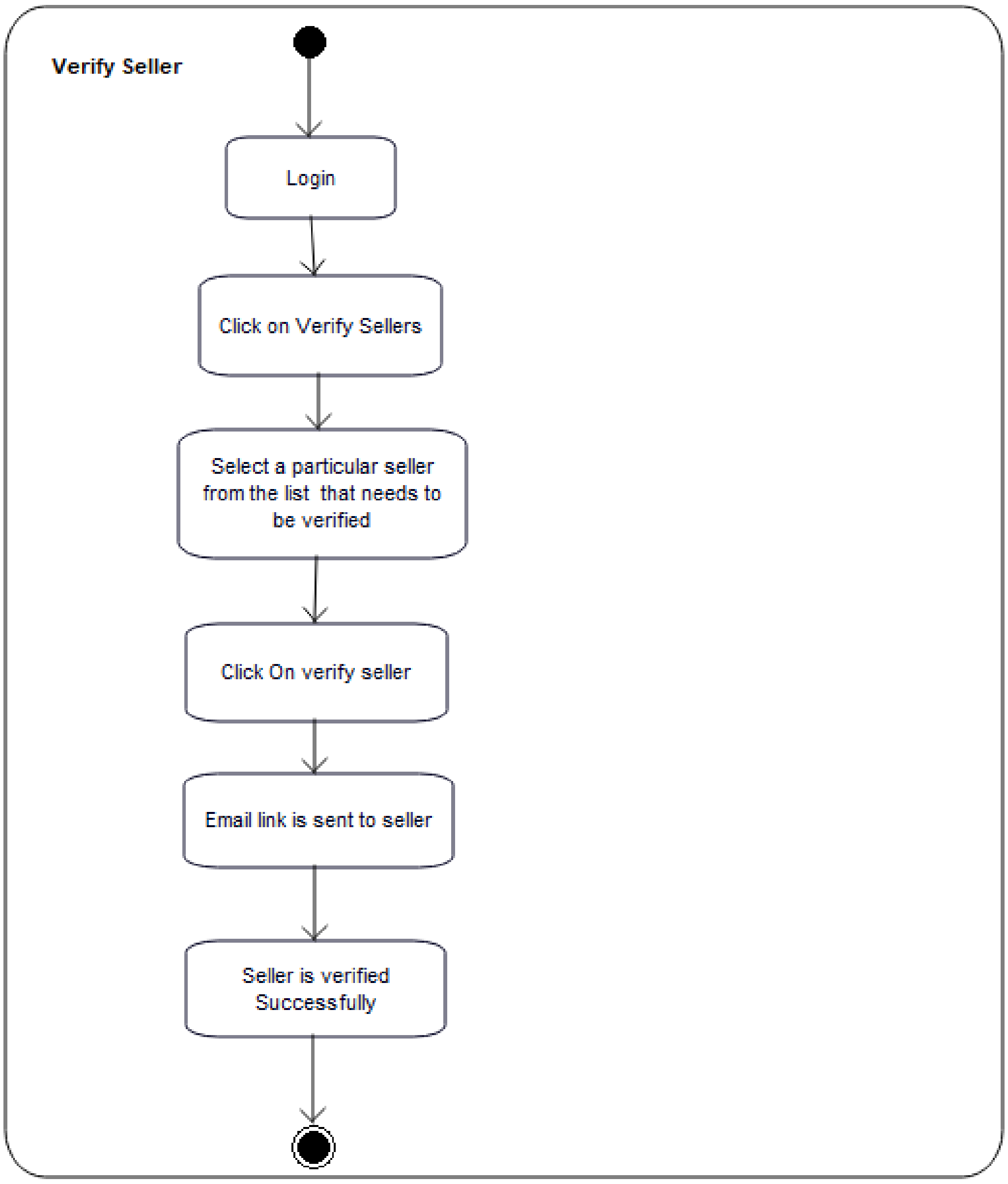

Verify Seller
Login
Click on Verify Sellers
Select a particular seller from the list that needs to be verified
Click On verify seller
Email link is sent to seller
Seller is verified Successfully

Figure 5.2: verify Sellers