68_Adnan Shaikh

Experiment No. 3

Aim: To prepare SRS document in IEEE format

Problem Statement: To prepare SRS document for E-commerce shopping website in IEEE format.

Theory:
A software requirements specification (SRS) is a document that describes what the software will do and how it will be expected to perform. It also describes the functionality the product needs to fulfill all stakeholders (business, users) needs.

A typical SRS includes:

- A purpose
- An overall description
- Specific requirements

The best SRS documents define how the software will interact when embedded in hardware — or when connected to other software. Good SRS documents also account for real-life users.

Why Use an SRS Document?

A software requirements specification is the basis for our entire project. It lays the framework that every team involved in development will follow.

It’s used to provide critical information to multiple teams — development, quality assurance, operations, and maintenance. This keeps everyone on the same page.

Using the SRS helps to ensure requirements are fulfilled. And it can also help us make decisions about our product’s lifecycle — for instance, when to retire a feature.

Writing an SRS can also minimize overall development time and costs. Embedded development teams especially benefit from using an SRS.

Software Requirements Specification

For

E-COMMERCE SHOPPING WEBSITE

Version 1.0 approved

Prepared by

ADNAN SHAIKH

SARASWATI COLLEGE OF ENGINEERING

29-08-2021

Table of Contents

1. Introduction

E-commerce is fast gaining ground as an accepted and used business paradigm. More and more business houses are implementing web sites providing functionality to purchase goods. It is reasonable to say that the process of shopping on the web is becoming commonplace. The objective of this project is to develop a general-purpose e-commerce store where product like clothes can be bought from the comfort of home through the Internet. However, for implementation purposes, this paper will deal with an online shopping for clothes. An online store is a virtual store on the Internet where customers can browse the catalog and select products of interest. The selected items may be collected in a shopping cart. At checkout time, the items in the shopping cart will be presented as an order. At that time, more information will be needed to complete the transaction. Usually, the customer will be asked to fill or select a shipping address, and payment information (COD). Customer Orders will be visible in my order page where customer can see the status of its orders

1.1 Purpose

- The software is for shopping products online.
- It maintains three levels of users
    1. Administrator level (Admin + User privileges).
    2. User level.
    3. Cookie User.
- The software includes home page for checking product.
- Product view to give feedback and for product query.
- Cart page to add desire product.
- Checkout page to purchase product (User level).

1.2 Document Conventions

- E-com.: E-commerce shopping website.
- HP: Home Page.
- PP: Product Preview.
- PF: Product Feedback.
- LP: Login Page.
- SP: Signup Page.
- ACP: Add To Cart Page.
- CP: Checkout Page.
- COD: Cash on Delivery.
- AP: Administrator Page.
- MOP: My Order Page

1.3 Intended Audience and Reading Suggestions

This document is to be read by the development team, the project managers, marketing staff, testers and documentation writers. Our stakeholders, company manufacturing associated hardware, company providing embedded operating system, shareholders, and distributors who markets the finished product, may review the document to learn about the project and to understand the requirements. The SRS has been organized approximately in order of increasing specificity. The developers and project managers need to become intimately familiar with the SRS.

1.4 Product Scope

E-com. is a web application that is designed to allow users to easily checkout products of their choice and store the desire product in ACP where total price and quantity of products can be view. For purchasing product user can move further to CP for online transaction and COD. Purchased product status and receipt will available in MOP. LP and SP can be used for Login, Sign up and recovery of account.

1.5 References

- Django: https://docs.djangoproject.com/en/3.2/
- Bootstrap: https://getbootstrap.com/docs/5.1/getting-started/introduction/
- Vue JS: https://vuejs.org/v2/guide/
- Flow Chart: https://app.diagrams.net/
- Modelling:https://miro.com/welcomeonboard/WGpZT2FEZDFRTW91QnpaUTJVTVViYjJVaFlJREdOcXVhYjltQjBKdWFjeFZvbVRNbThqYXZnRTZFRlBmdzR1SnwzMDc0NDU3MzU3ODAwNTI0NDYx

2. Overall Description

2.1 Product Perspective

The various system tools that have been used in developing both the front end, back end and other tools of the project are being discussed in this section.

1) **FRONT END**:
HTML, CSS, JAVA SCRIPT, BOOTSTRAP, VUE JS, JINJA are utilized to implement the frontend.
   - **HTML** (Hyper Text Mark-up Language)**:**
     HTML is a syntax used to format a text document on the web.
   - **CSS** (Cascading Style Sheet)**:**
     CSS is a style sheet language used for describing the look and formatting of a document written in a mark-up language.
   - **Java Script:**
     JS is a dynamic computer programming language. It is most commonly used as part of web browsers, whose implementations allow client-side scripts to interact with the user, control the browser, communicate asynchronously, and alter the document content that is displayed.
   - **Bootstrap:**
     Bootstrap is the most popular HTML, CSS, and JavaScript framework for developing responsive, mobile-first websites.
   - **Vue JS:**
     Vue.js is an open-source progressive JavaScript framework for building user interfaces (UIs) and single-page applications; it is commonly referred to as Vue. This framework uses "high decoupling", allowing developers to progressively create user interfaces (UIs).
   - **Jinja:**
     Jinja is a fast, expressive, extensible templating engine. Special placeholders in the template allow writing code similar to Python syntax. Then the template is passed data to render the final document.

2) **BACK END:**
The back end is implemented using MYSQL which is used to design the databases.
   - **MYSQL:**
     MySQL is the world's second most widely used open source relational database management system (RDMS). The SQL phrase stands for structured query.

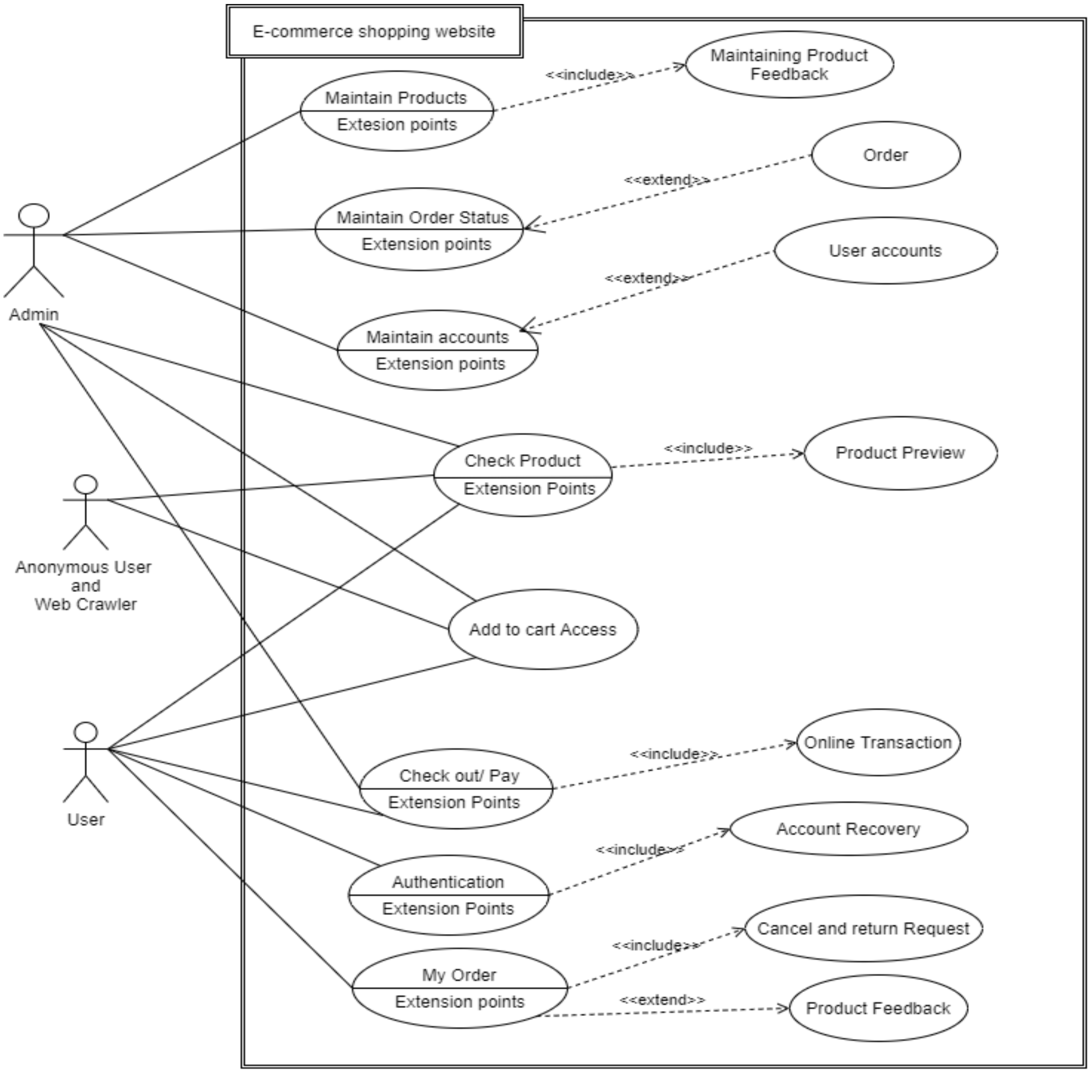
3) **API:**
To connect frontend and backend and writing logical operation Django framework is used.
   - **Django:**
     Django is a high-level Python web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of web development, so we can focus on writing our app without needing to reinvent the wheel.

2.2 Product Functions

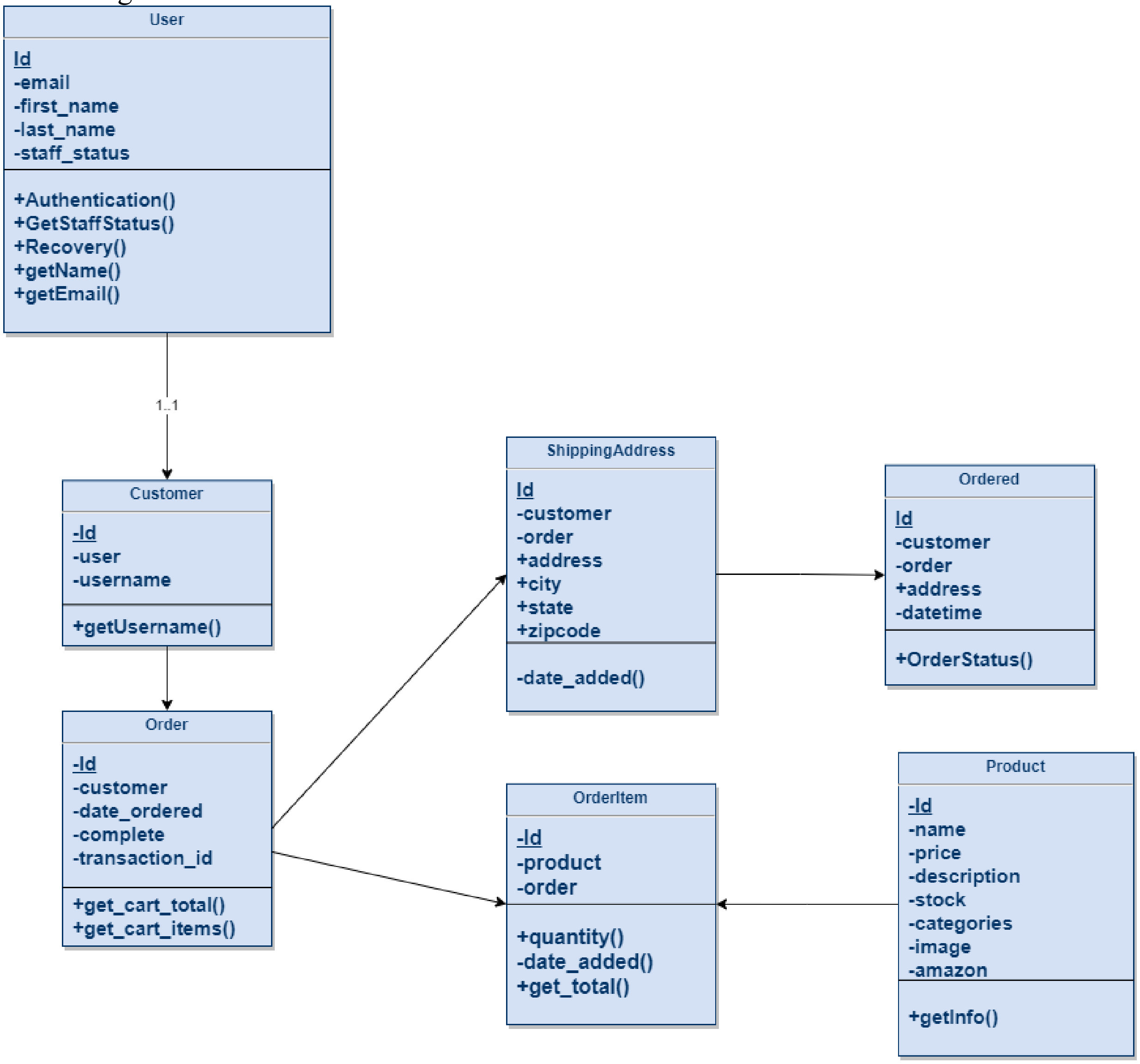
- The system will allow user to use the app without logging.
- Cookie data is used to remember activity of anonymous user.
- To purchase any product user need to login and provide address and some other necessary requirements.
- User can always check what products they purchased and what is the status of product in MOP section
- If user forget their account recovery functionality is also available.
- If user have any query regarding product they can ask in PF section.
- Admin have all privilege of user as well as additional administrator functionality which let them add, update and delete products, accounts, feedbacks and orders.

2.3 User Classes and Characteristics

1. Use Case Diagram:



2. Class Diagram:

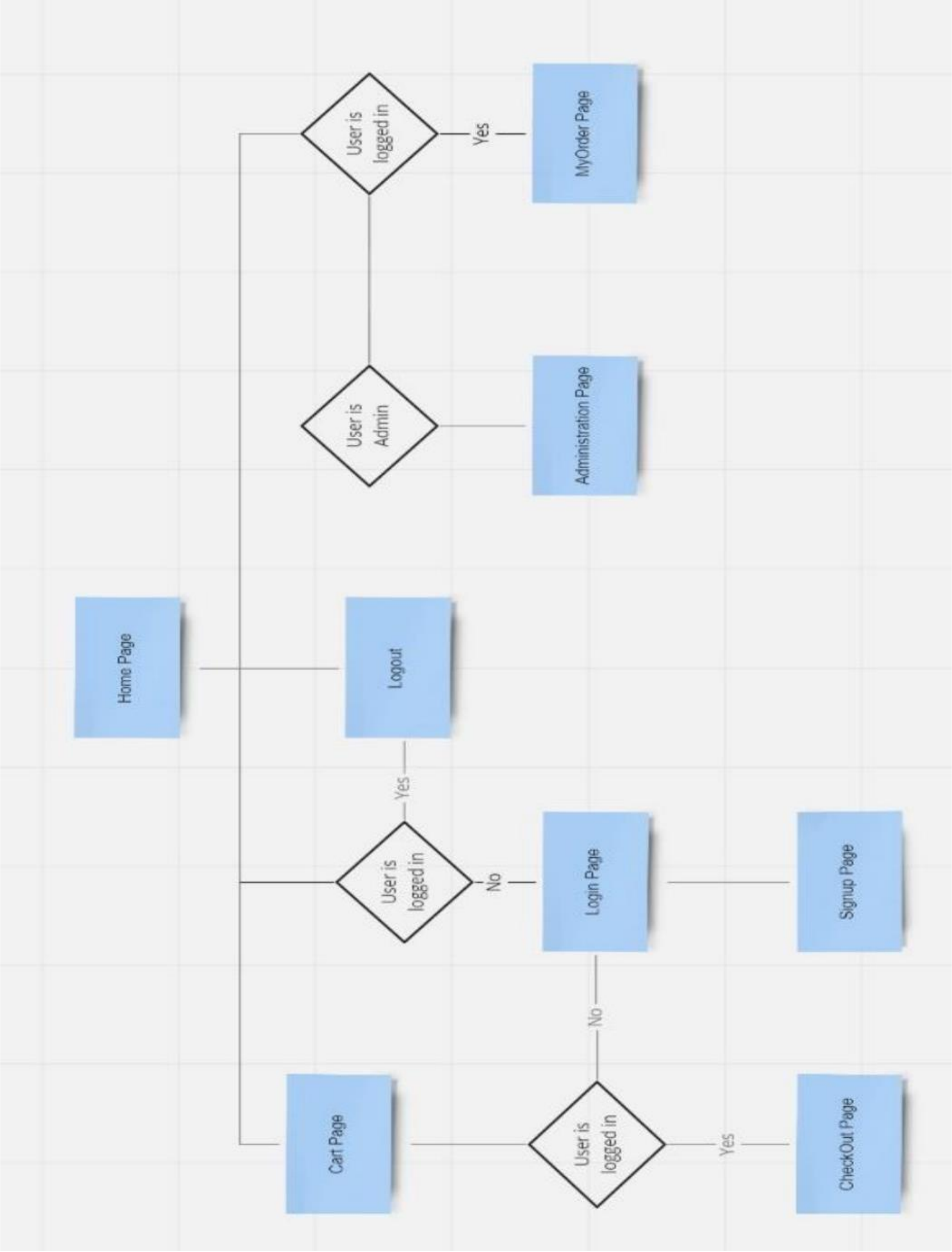


2.4 Operating Environment

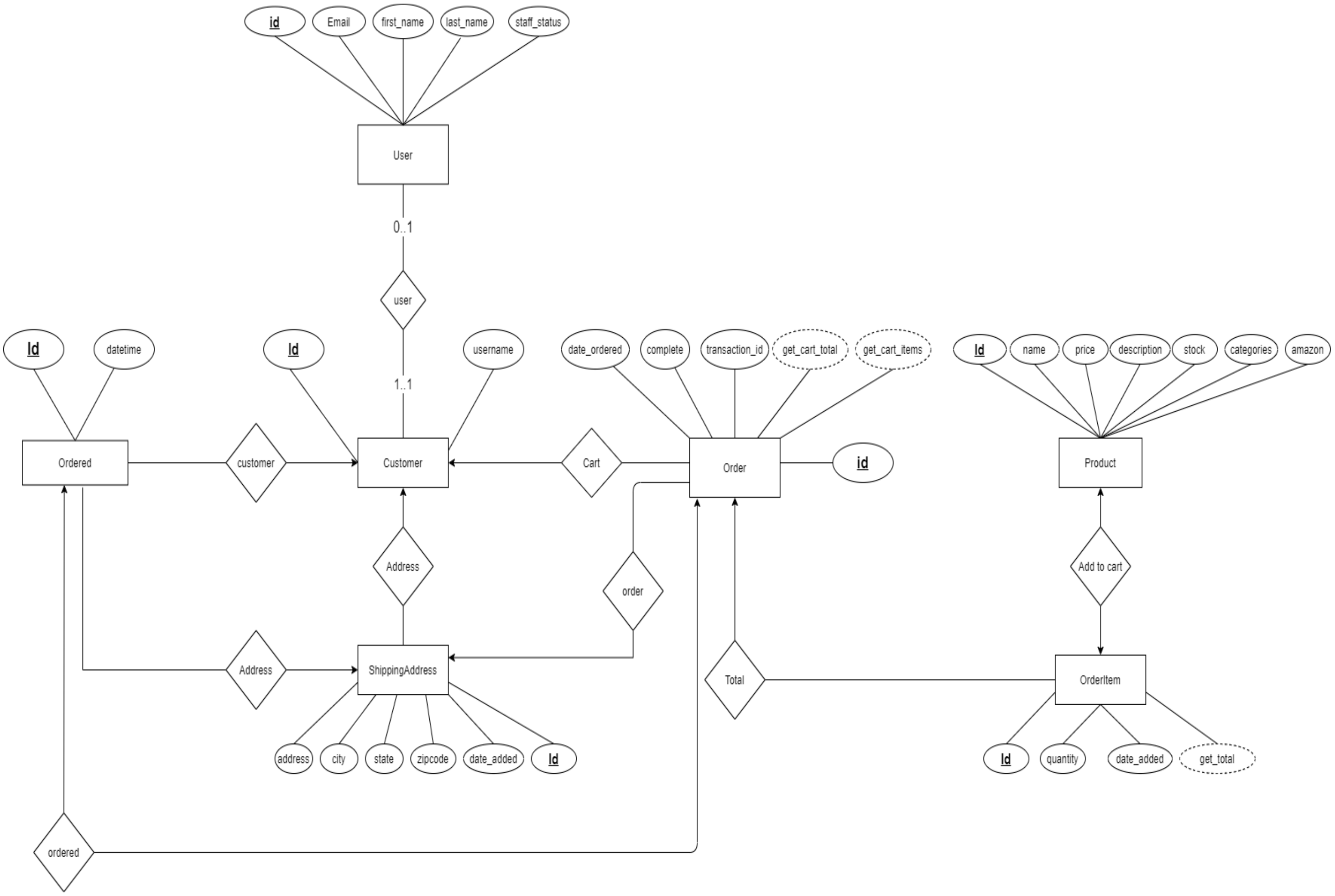
The system will be accessible through any web browser with JavaScript enabled, user should have good internet connectivity and proper version of web browser in Mobile/P.C, O.S is not a barrier as long as it can handle web easily. Some of the best browsers to access: Chrome, Firefox, Brave, Safari and Opera.

2.5 Design and Implementation Constraints

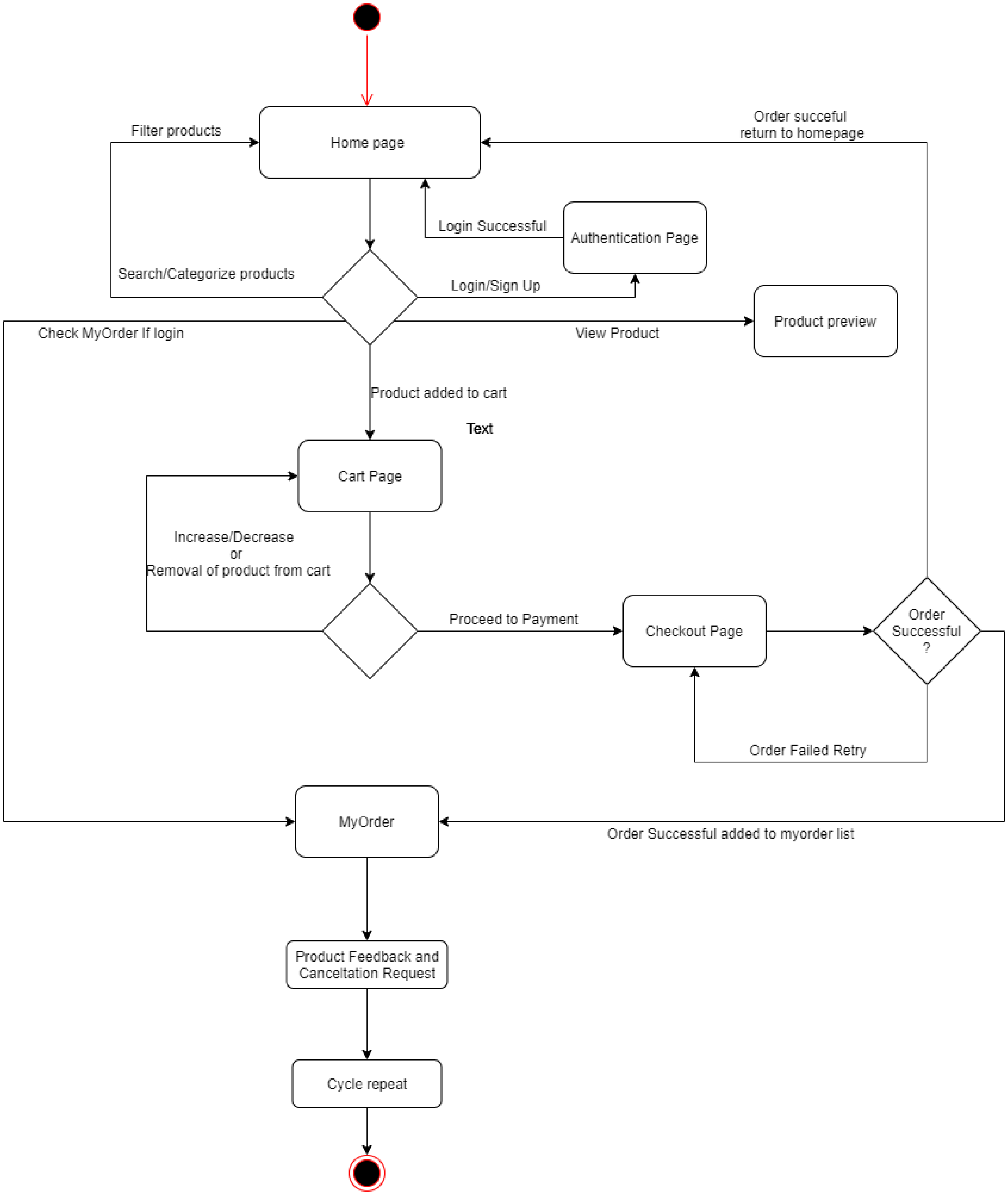
1. Data Flow Diagram:



2. Entity Relationship Diagram:



3. Activity Diagram:



2.6 Assumptions and Dependencies

- Server can take load of 10,000 users at any instance.
- User is familiar with browser interface.
- Minimum display aspect ratio is 5:3.

3. External Interface Requirements

3.1 User Interfaces (GUI Design)

User will be able to interact with the system using Mobile (Smartphone), Laptop or P.C. User can use touch or Keyboard mouse to interact with system functions.

3.2 Hardware Interfaces

All components able to be executed on any device which handle web browser and have minimum display aspect ratio 5:3.

- O.S: Ubuntu, Debian, Windows, Android, MAC OS, IOS, etc.
- Hard Disk: 1Gb able to handle web browser.
- RAM: 100 MB for E-com. + 250 MB for browser.
- Processor: Dual Core up to 1.5-2GHZ.

3.3 Software Interfaces

The System will be hosted using AWS (Amazon Web Server) and system uses MYSQL to store data in database and code store in GitHub Repository.

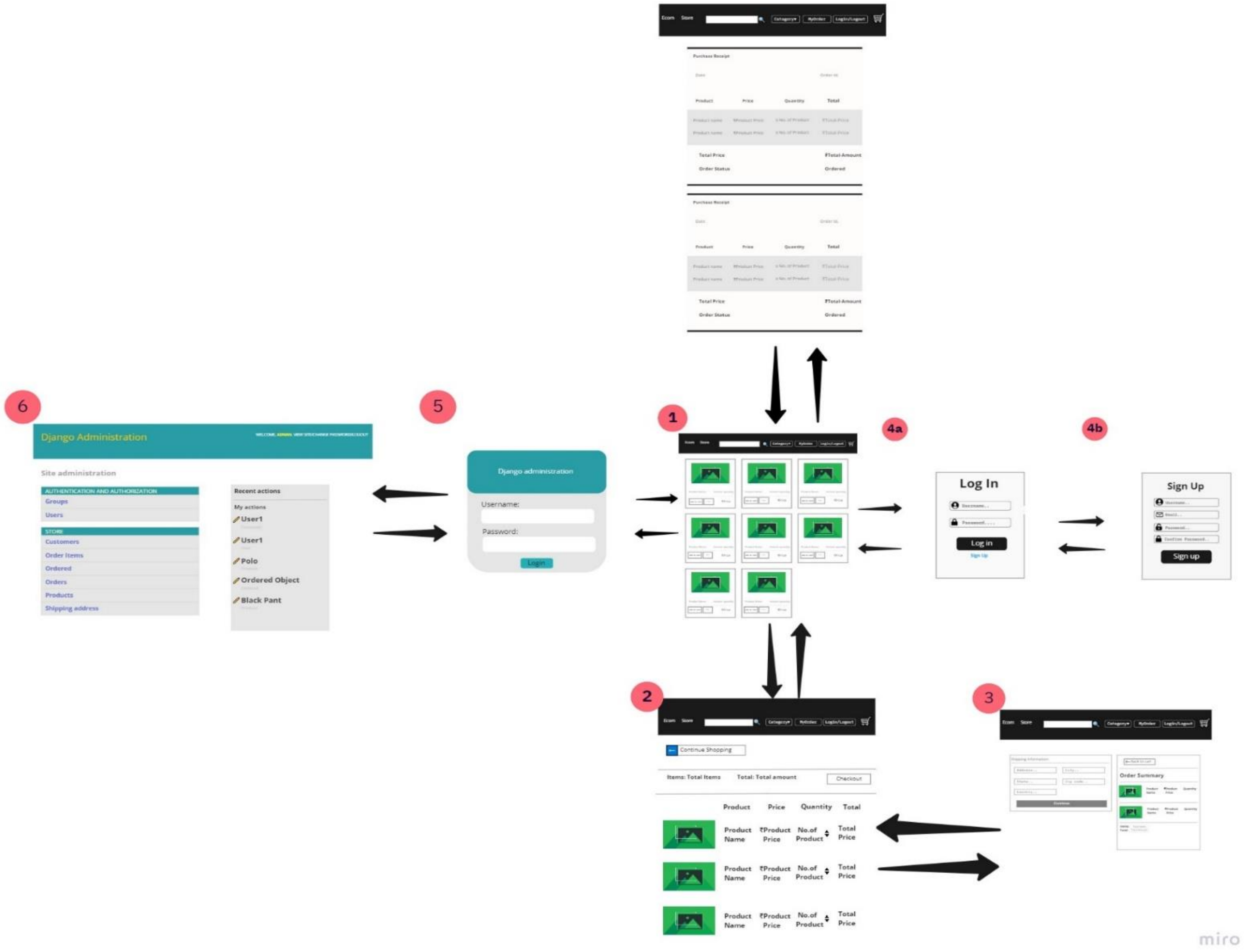
- Python and Cython Language
- MySQL DB
- AWS for cloud hosting.
- GitHub for cloud repository.

3.4 Communications Interfaces

System uses DRF (Django Rest Framework) which provide JSON (JavaScript Object Notation) response through HTTP which comes under TCP/IP protocol, Naïve user of the E-com. doesn't need to know about these things, they just need to know Django uses CSRF token which provides high security and reliability so their data is safe while interacting with website using internet through any web browser.

4. System Features

4.1 System Architecture



The project is divided into 6 major sections:

1. Home Page
2. Cart Page
3. Check out Page (No access to Anonymous user and web crawlers).
4. Login and Sign Up Page.
5. My Order Page (No access to Anonymous user and web crawlers).
6. Admin Page (Only Accessible by Staff level authority account).

4.1.1 Description and Priority

Admin: Admin account have highest priority it can access and modify every single function provided by project, it can also increase authority level of user account to staff account.

Staff user: Can maintain order status and feedback of user, priority below admin level. Access permitted to administrator page

User: User can surf through website provide feedback for the product they purchased and request return or cancelation of product they purchased. No access to administrator page.

Cookie user and Web Crawler: Can only surf through home page and cart page can't purchase any product, lowest level priority, No access to Payment and My order page.

4.1.2 Action

1) Admin Page:

- Admin can add, delete and modify products, accounts, ordered products and address.
- Staff user can add, delete and modify products and ordered products.
- Each schema object is available to handle.

2) Home Page:

- Available product can be view here and product preview provide detail description of product.
- User and anonymous user can add product to cart from her.
- Search and categorized options are provided to quickly check whether or not the product is available.
- Most of the page are accessible from here.

3) Authentication:

- Login and Signup page is provided for authentication.
- If user forget the account recovery option is also there, user can recover their account by providing their corresponding email or phone number which they provided when signup.

4) Add to Cart Page:

- Product added from homepage to cart will be visible here.
- Added product can be increase or decrease in quantity or removed from cart.
- Summary of added products will be shown here i.e. total price of each product, total price of overall products and the total quantity.
- Login user can move to check out page from here.

5) Checkout Page:

- Login user is only allow on this page.
- Product payment can be done here (Online or COD).
- Address is needed for deliverable product.
- Once the order payment is successful user is redirect to homepage and receipt is generated in MyOrder section.
- Receipt is send on mail provided by user.

6) MyOrder Page:

- User order product receipt will be visible here.
- Delivery status of product will be shown in receipt.
- User can give product feedback once it is delivered.

- User can request for order cancelation and return from this page.
- Customer service number is also given here.

4.1.3 Functional Requirements

The software provides good graphical interface for the user any administrator can operate on the system, performing the required task such as create, update, and modify the details of products, orders and accounts.

Allows user and anonymous user to view products and add desire product to cart and purchase product option to user.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The performance of system is at its best because of Django we don't need to think about speed and performance, we just need to look for any bugs in the code and update the software time to time

5.2 Security Requirements

Django uses CSRF token and hashed user credential data, password of user is not even visible to Admin so user account is well secured and time to time updated software is mostly free of attack, for maintaining security we need to keep software update.

5.3 Software Quality Attributes

The Quality of the system is maintained in such a way so that it can be very user-friendly. The software quality attributes are assumed as under:

- Accurate and hence reliable.
- Secured.
- Fast Speed.
- Well maintained.

Conclusion: We have successfully prepared SRS document for E-commerce Shopping website.