Software Requirement and Design Specifications

Price Comparison Website

Course Code:	**CS-303**
Instructor:	**Miss Rubab Jaffer**
Project Team:	**Muhammad Taha Khan 17K-3656** **Syed Unaiz Haider 17K-3667** **Talha Ahmed 17K-3721**
Submission Date:	**23rd May 2020**

Table of Contents:

1. Introduction:

1.1. Purpose of Document

The purpose of this document is to explain briefly how the idea of a **Price Comparison Website** came to our thoughts, why we pursued this project and how we implemented it.

Price comparison sites and comparison-shopping engines give ecommerce traders a good opportunity to boost their sales, attract new customers and go ahead against their competitors. Even Shoppers often look for best deals and offers for a particular product which they are looking for. It is difficult to visit each and every website for comparing the deals or the price of a particular product.

Hence this comparison-shopping site is proposed which will help ecommerce shoppers for collecting product information, including price list, from participating retailers and then it displays collective information on a single result page in response to a shopper's search query.

In this way, shoppers can compare prices, and service from multiple websites on a single page. Users need to open the website and search for a particular product, it will compare the prices from different websites. This will make shopping easy with best deals and user experience.

1.2. Intended Audience

The intended audience in this project are:

- Ecommerce Companies
- Online Customers
    - Buyers
    - Sellers

1.3. Definition of Terms, Acronyms and Abbreviations

Term	Description
Web Scraping	A technique for collecting data from websites and combining them in file or store in databases.
API	Application programming interface. It refers to a set of rules that determine how pieces of software interact with each other.
UI	User interface. The visual aspect of a tool that a person uses for controlling it.
URL	Uniform resource locator. The web address used for identifying a website or page.

1.4. Document Convention

The document is written in Calibri Font. The main headings are of Font Size 18 while sub-headings are of Font Size 16 and the text written is in Font Size 14.

2. Overall System Description

2.1. Project Background

Pakistan's e-trading mainly involves buying and selling goods and services using the internet or telephone, through the use of electronic means such as computer, fax machine, cellular phone, automated teller machines (ATM) and other electronic appliances with or without using the internet.

Online banking, e-ticketing for air travelling, share trading in stock exchange are few examples of e-commerce of modern advancement. With its potential, e-trading can reduce the cost per transaction, increase efficiency, support contest, lower prices and boost international demand. It can open new areas for business in the service sector like on-line education, medical services, consultancy, and data exchange. It can also provide expansion in trade through domestic and international market research, advertising and marketing. The interesting feature of online trading is that an investor simply sitting in his office or home can buy or sell through the Internet via mobile/tablet or PC and before being an experienced trader he may learn a lot by watching market screens or web portals at his convenience.

Pakistanis are sensitive to price in purchasing goods. Therefore, there are few websites in Pakistan in the market that are promoting and comparing prices. One of the existing websites is Cartright.pk. This website displays products from big ecommerce companies in Pakistan such as Daraz, ishopping, Hummart, goto, telemart and so on with price. In conclusion, in this project we are going to develop a Price Comparison Website similar to Cartright and other price comparison websites in Pakistan with new and advanced features.

2.2. Project Scope

Our website is an online promoting platform to help consumers to spend money wisely and at the same time this is also a platform for advertisers to promote their products.

Our project is designed to extract all the information about the product from different e-commerce websites and after extraction it stores the information in a centralized database, finally ranking them and giving the top results to the users according to their query. The final products that are displayed on the interface would be the result of the ranking that is calculated on the basis of price. We also provide tracking options so the user can get price updates on a daily basis.

2.3. Not In Scope

Not applicable

2.4. Project Objectives

The objective of this project is to define a nice and easy pathway to a customer so that he/she can easily find the right product from various websites in one platform. It becomes a hassle for a customer to visit different websites and choose the best product among them. We are going to rank the products according to the price customers see the best product for the query they provided, they will type their product name and select category and top products will be shown to them from various e-commerce websites. If registered on our platform a feature called tracking is also available which is to track the price of the selected product. The tracking module sends daily updates of the product prices.

2.5. Stakeholders

Customers

- Buyers
- Sellers

E-commerce websites

2.6. Operating Environment

This project requires a computer running Microsoft Windows operating system.

The supported browsers are supported:

- Mozilla Firefox
- Internet Explorer
- Microsoft Edge
- Google Chrome.

2.7. System Constraints

- **Software Constraints**

  There is no software constraint as a person with an internet connection can access the site easily from any browser.

- **Hardware constraints**

  There is also no hardware constraint as the project is going to run through an internet website so it doesn't depend on any hardware to run.

- **Cultural constraints**

  The entire project including the signup page, login page, search result page, profile pages etc are in English.

- **Legal constraints**

  Not applicable

- **Environmental constraints**

  Not applicable

- **User constraints**

  Not applicable

- **Off the shelf components**

  Not applicable

2.8. Assumptions & Dependencies

There are no assumptions taken and there are no dependencies of the system on any external factor.

3. External Interface Requirements

3.1. Hardware Interfaces

There are mainly three types of interfaces in this project:

- **Registration Interface**
  In this interface the user will register to our platform using its email address. Registration is not necessary for using our website. Registration adds on only the tracking functionality of the user.
- **Login Interface**
  Login interface is used to sign in to our website; it directs the user to the dashboard where the tracked products by the user are displayed.
- **Search Result Interface**
  When a user searches for any product the results are displayed in the search result interface where the user can click on any product card which will redirect it to the product link. If the user is signed in then the tracking button is also displayed which is to track product price.

3.2. Software Interfaces

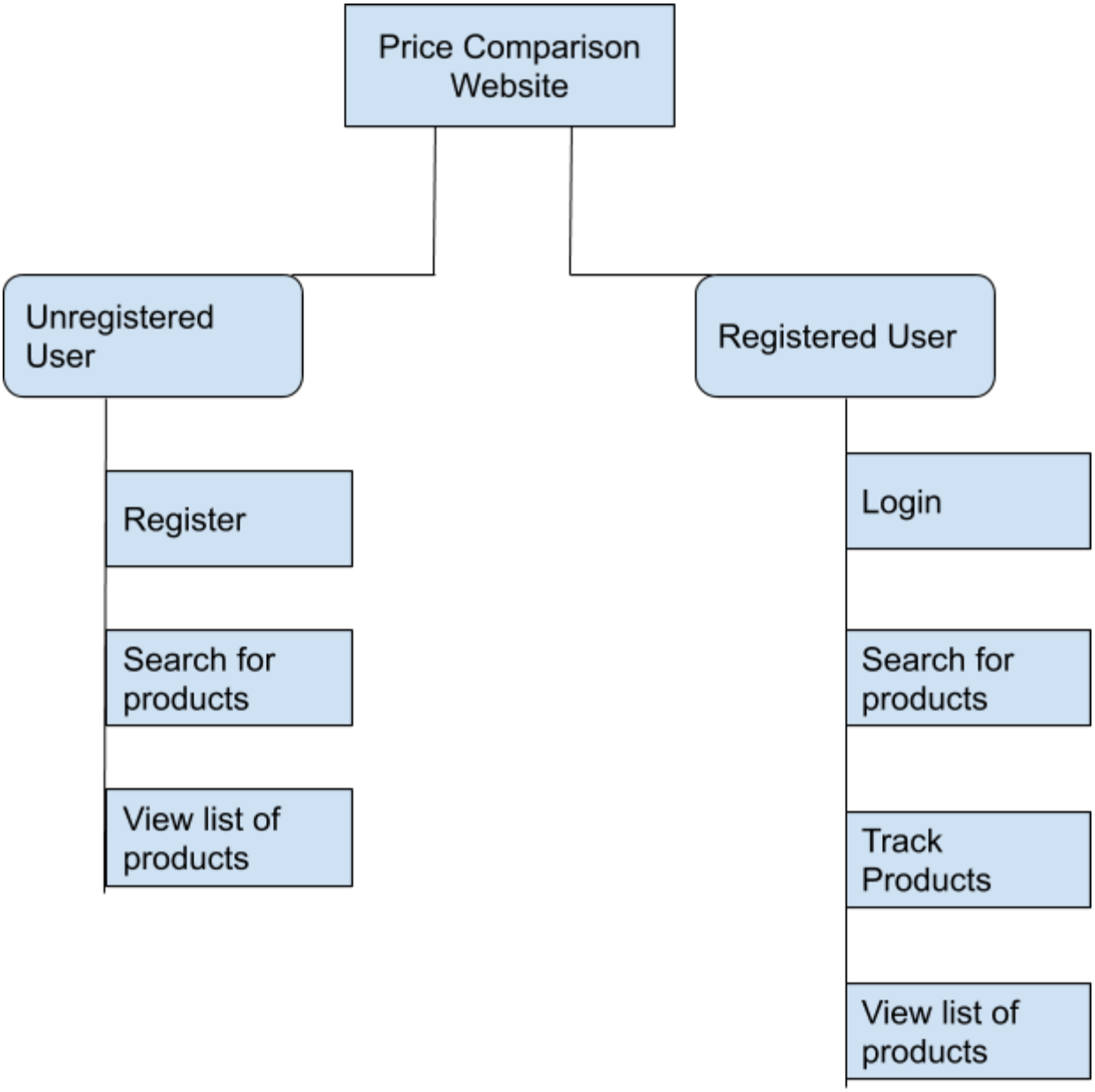
- Database used in this project is mongodb 4.2.1
- Operating system is MicroSoft Windows 10.
- Tools and Libraries Python and for frontend on HTML5
- Libraries:
  - Flask
  - Pymongo
  - Wtforms
  - Functools
  - Passlib
  - Selenium
  - BeautifulSoup

3.3. Communications Interfaces

The frontend which is made in HTML5 and backend where all the algorithms and connection to database are made is written in python 3 on SpyderIDE, are connected using Flask API.

4. Functional Requirements

4.1. Functional Hierarchy



4.2. Use Cases

Use Case: Register
Primary Actor: Customer(User)
Stakeholders and Interests: System
Preconditions:
User has launched the website and clicked the register button.
Success Guarantee (Postconditions):

- Account is registered successfully.

Main Success Scenario (or Basic Flow):

- User presses the register button on the main screen.
- User provides the required information.
- System prompts the user to provide account details which will be used as system credential.
- System saves the user information.

Extensions (or Alternative Flows):

- If the user doesn't provide the necessary information prompt to fill the fields.

Use Case: Login
Primary Actor: Registered Customer(User)
Preconditions:

- User has visited the web portal.
- User has registered/signed up for the account.

Success Guarantee (Postconditions):

- User has successfully logged in.

Main Success Scenario (or Basic Flow):

- Users visit the website.
- User clicks on the login button.
- User provides the email and password.
- System checks the email and password is valid.
- If the username and password is valid then login to the account.

Extensions (or Alternative Flows):

- If the user hasn't registered, prompt to register/signup.

Use Case: Search products
Primary Actor: Customer(User)
Stakeholders and Interests: System
Preconditions:

- User has launched the website
- User has typed the query for the product
- User has selected the category of the product

Success Guarantee (Postconditions):

- Product list is displayed successfully.

Main Success Scenario (or Basic Flow):

- User enters the search query for the product in the search bar.
- User selects the category of the product.
- The product list fetched from different website is displayed to the user

Extensions (or Alternative Flows):

- If the user query is not found in any of the websites an empty page is displayed.

Use Case: Shop product
Primary Actor: Customer(User)
Preconditions:

- User has visited the web portal.
- User has queried for the specific product.

Success Guarantee (Postconditions):

- User is redirected to the main product page.

Main Success Scenario (or Basic Flow):

- Users visit the website.
- Users enter the search query and category.
- Product list is displayed.
- User selects the specific product to see and clicks on the shop button.
- User is redirected to the main product page.

Use Case: Track Product
Primary Actor: Registered Customer(User)
Stakeholders and Interests: System
Preconditions:

- User has visited the web portal.
- User has queried for the specific product.

Success Guarantee (Postconditions):

- Product information is stored in a database to be tracked.

Main Success Scenario (or Basic Flow):

- Users visit the website.
- Users enter the search query and category.
- Product list is displayed.
- User selects a specific product to be tracked and clicks on the track button.
- System prompts product is added to track.

Extensions (or Alternative Flows):

Use Case: Delete product from tracking
Primary Actor: Registered Customer(User)
Preconditions:

- User has visited the web portal.
- User has registered/signed up for the account.
- User has logged in to his/her account.

Success Guarantee (Postconditions):

- Product is deleted from the track product database.

Main Success Scenario (or Basic Flow):

- Users visit the website.
- User logs in to its account.
- Tracked products are displayed to the user.
- User clicks on the delete button.
- Product is removed from the database and stopped displaying.

Extensions (or Alternative Flows):

Use Case: Manage Product
Primary Actor: Admin
Stakeholders and Interests: System
Preconditions:

- User has submitted the query for the product.

Success Guarantee (Postconditions):

- Relevant Product information is shown to the user.

Main Success Scenario (or Basic Flow):

- Users visit the website.
- Users enter the search query and category.
- Admin automates the query and fetches the products from various websites and sort them according to the price.

Extensions (or Alternative Flows):

Use Case: Send emails
Primary Actor: Admin
Preconditions:

- Registered user has added products for tracking.

Success Guarantee (Postconditions):

- Daily price reminder email is generated and sent to the user

Main Success Scenario (or Basic Flow):

- Users tracks products
- Automated admin sent email according to the users

Extensions (or Alternative Flows):

- Email is not sent

5. Non-functional Requirements

5.1. Performance Requirements

- NF1: Updated information on the frontend should be reflected in the database and vise-versa. This should not take more than a few seconds.
- NF2: A query will respond in a few seconds.
- NF2: Users should get daily email updates on time of the tracked products.

5.2. Safety Requirements

Not applicable

5.3. Security Requirements

Not applicable

5.4. User Documentation

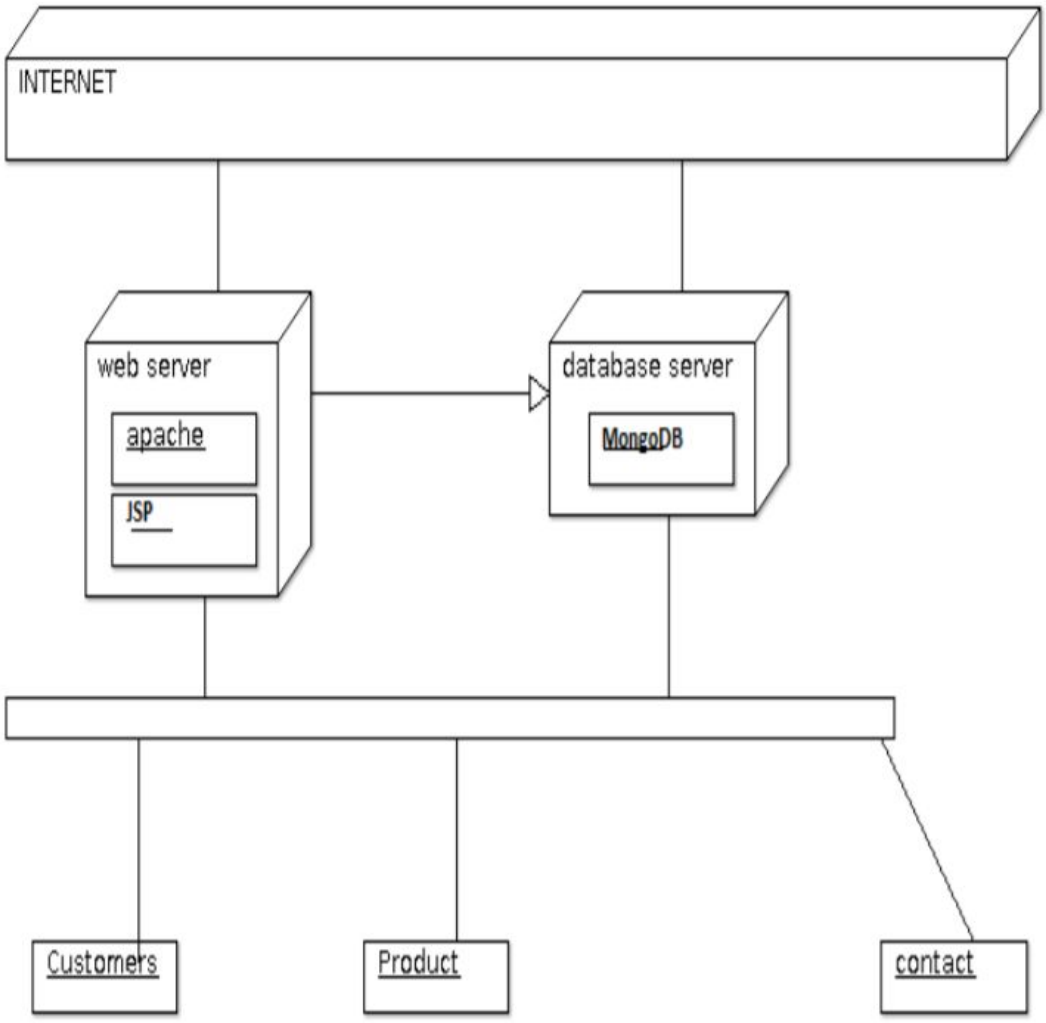
Not applicable

SDS

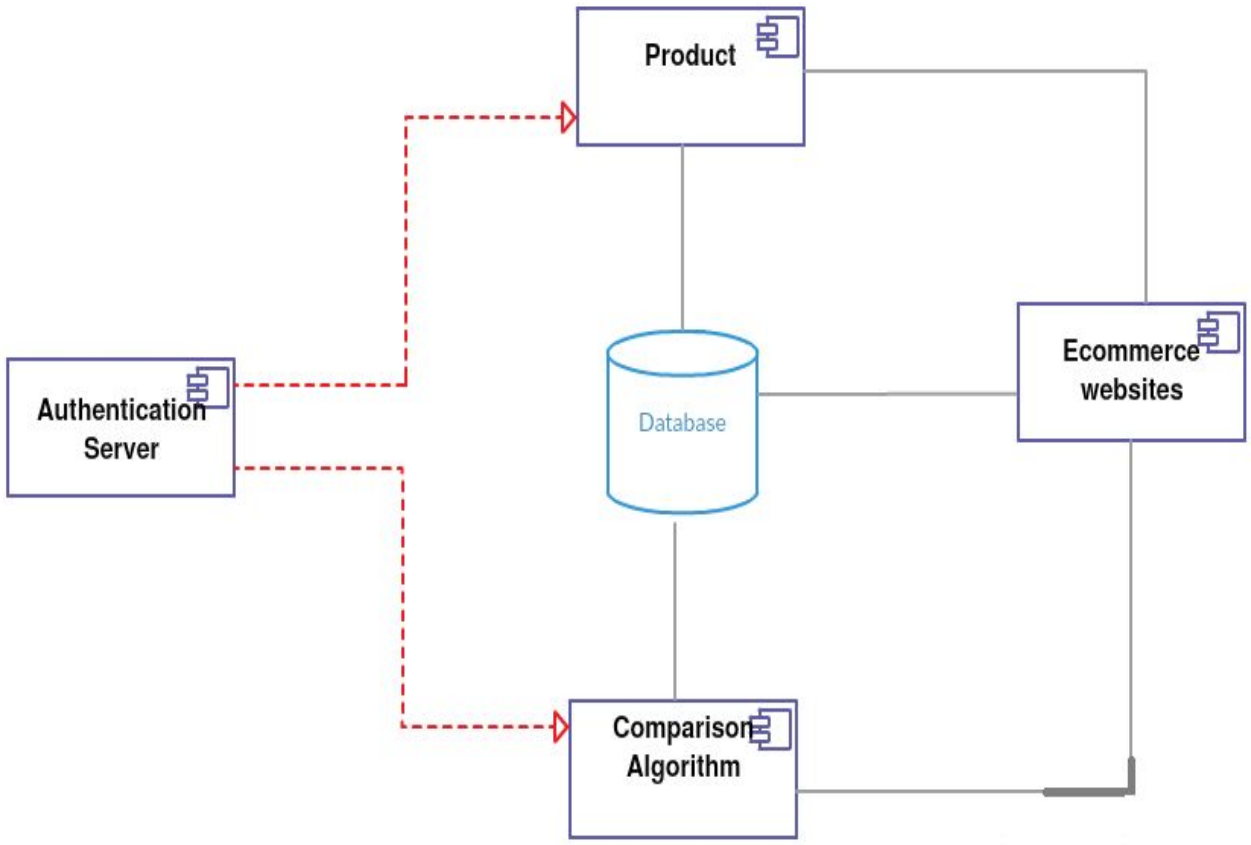
6. System Architecture

6.1. System Level Architecture

Deployment Diagram:



Component Diagram:



7. Design Strategy

Finding a right product to buy online is a very hectic and time consuming process, due to so many websites and mostly visiting each and every website explicitly and checking the price of the same product and comparing them. Our work makes this time consuming process summed up in one platform where users can get best products recommendation from various ecommerce websites in one place and, with ample potential for improvement in the future.

First we implement a scraping technique in which the crawler on each query scrapes each website for the products and then sorts them according to the price and displays the top result from the website.

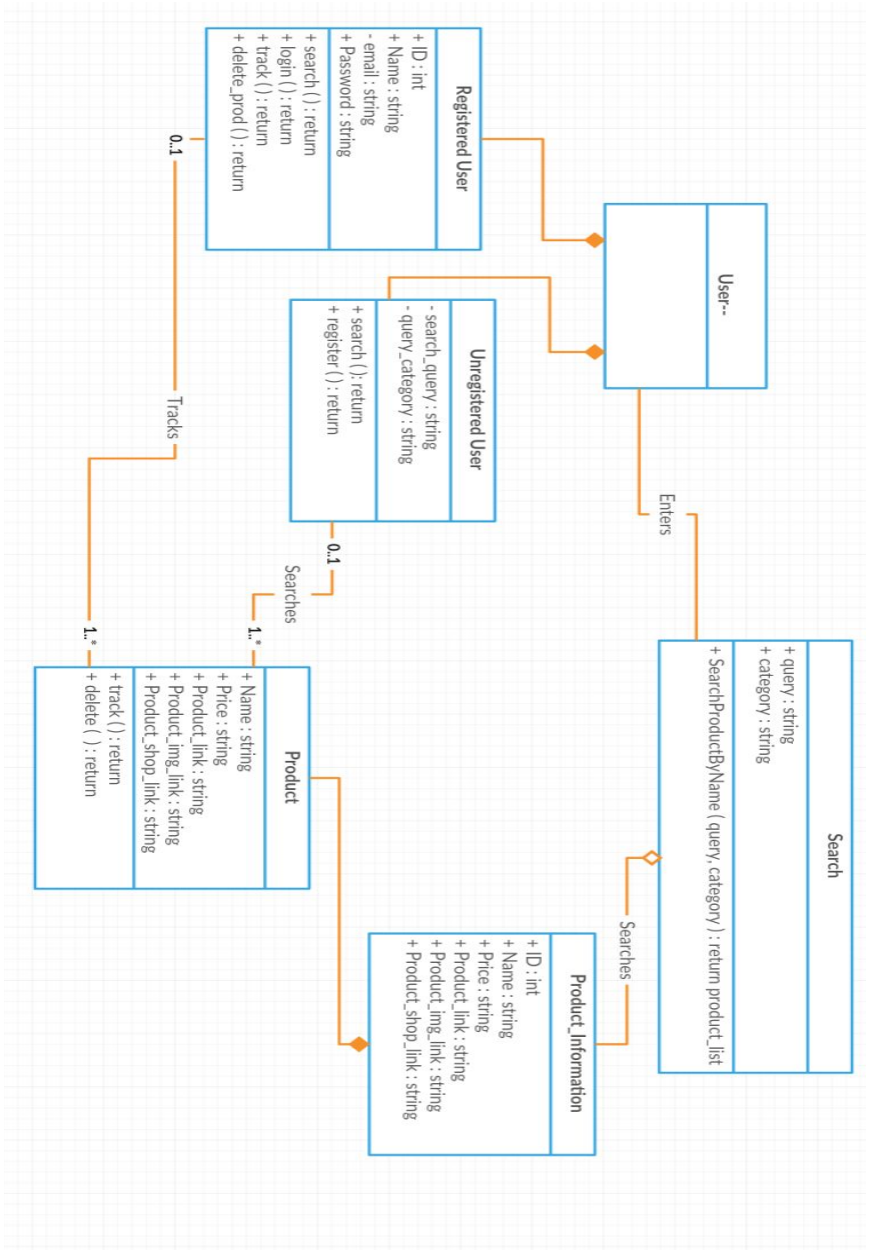
Second, to improve the search speed of our system, we implemented our system as for the first time the query runs the product information is stored in our database the for next time the product list will be fetched from the database therefore it reduces the scraping time of the system as it will not scrape the product information every time the user runs the query. This approach is implemented because mostly the product information is the same as many times we search any ecommerce site.

Future advancements in this system can be as follows:

- Better user experience
- Extension of the worldwide ecommerce websites
- Recommending products based on reviews and prices
- A feature that website is working or not

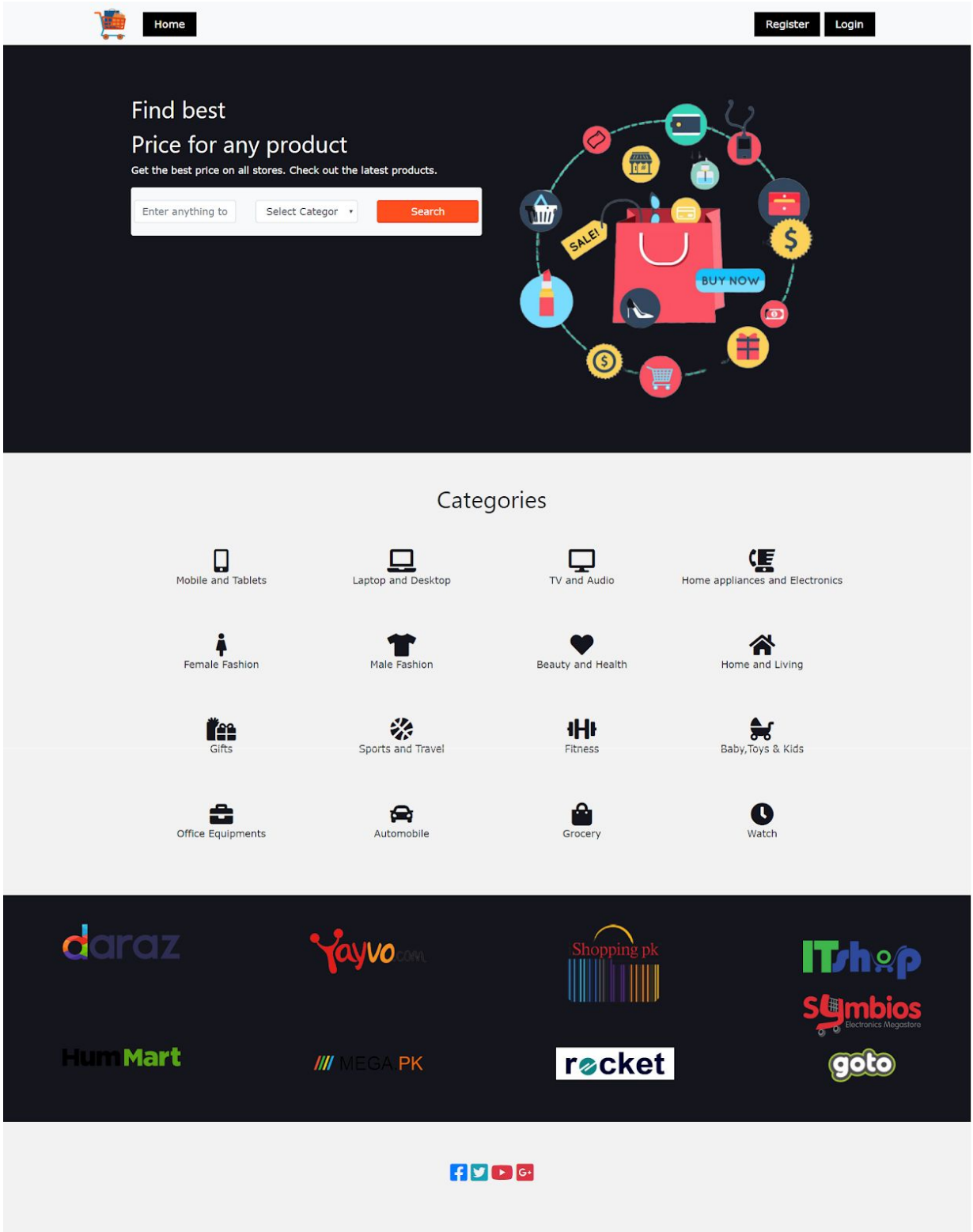
8. Detailed System Design

Class Diagram:

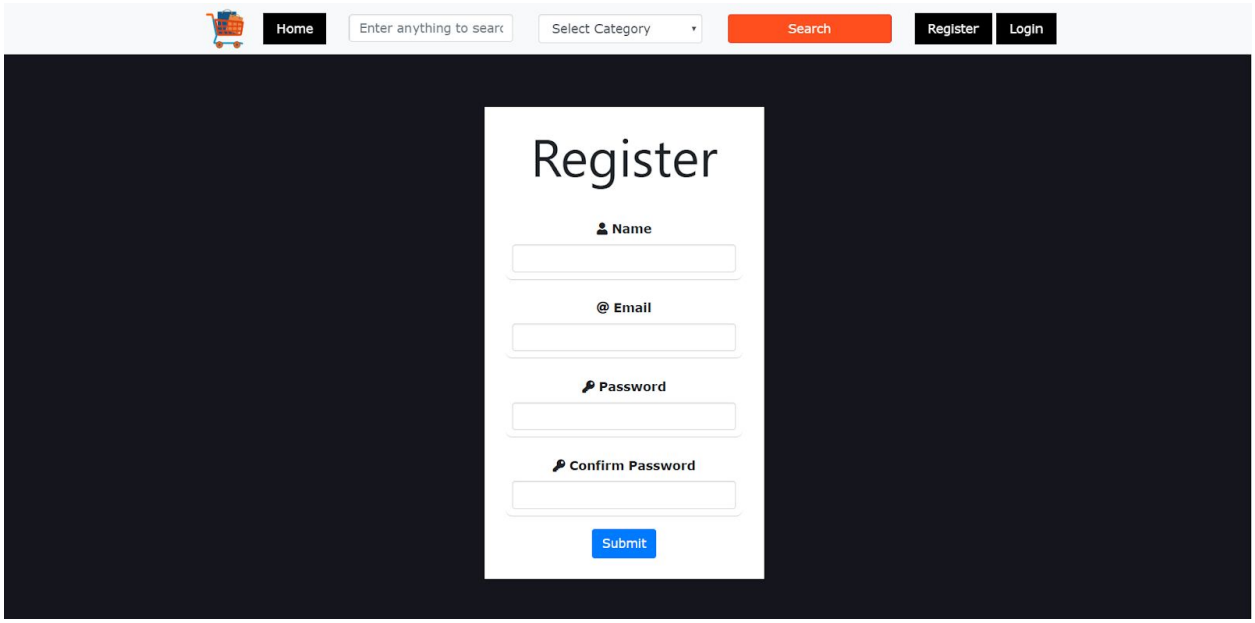


GUI:

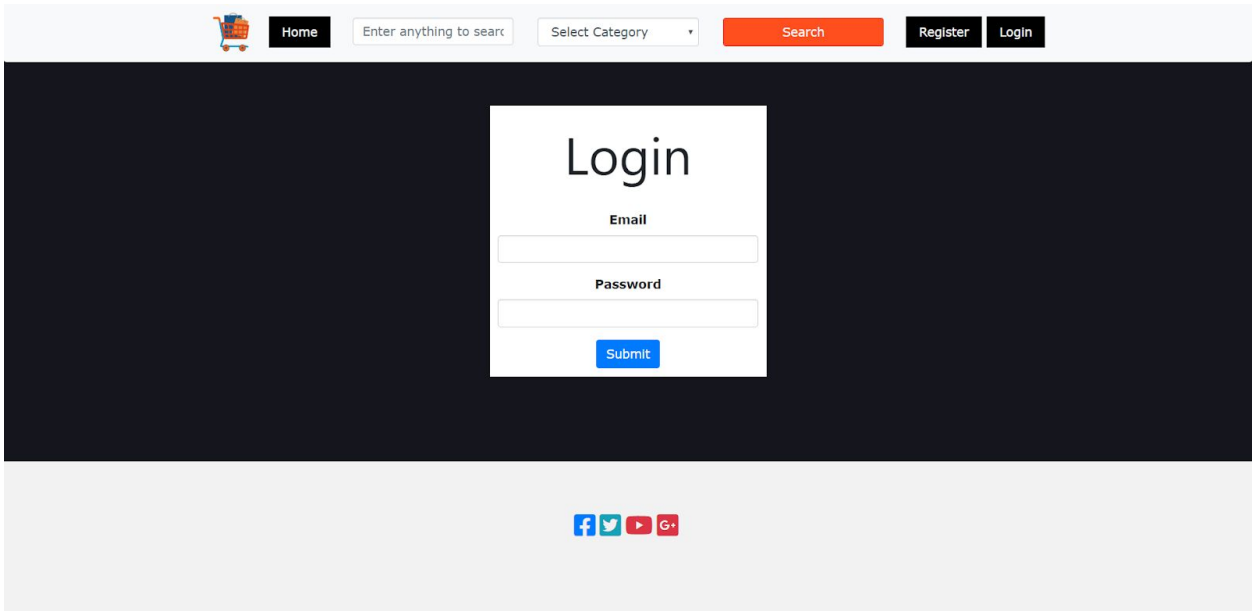
Homepage



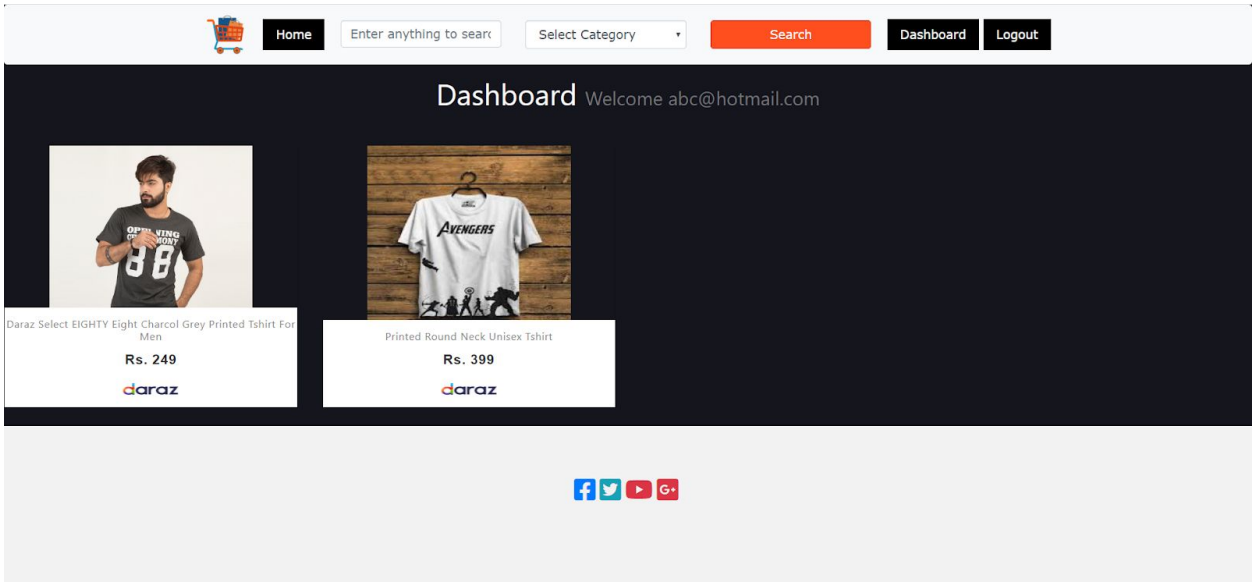
Register



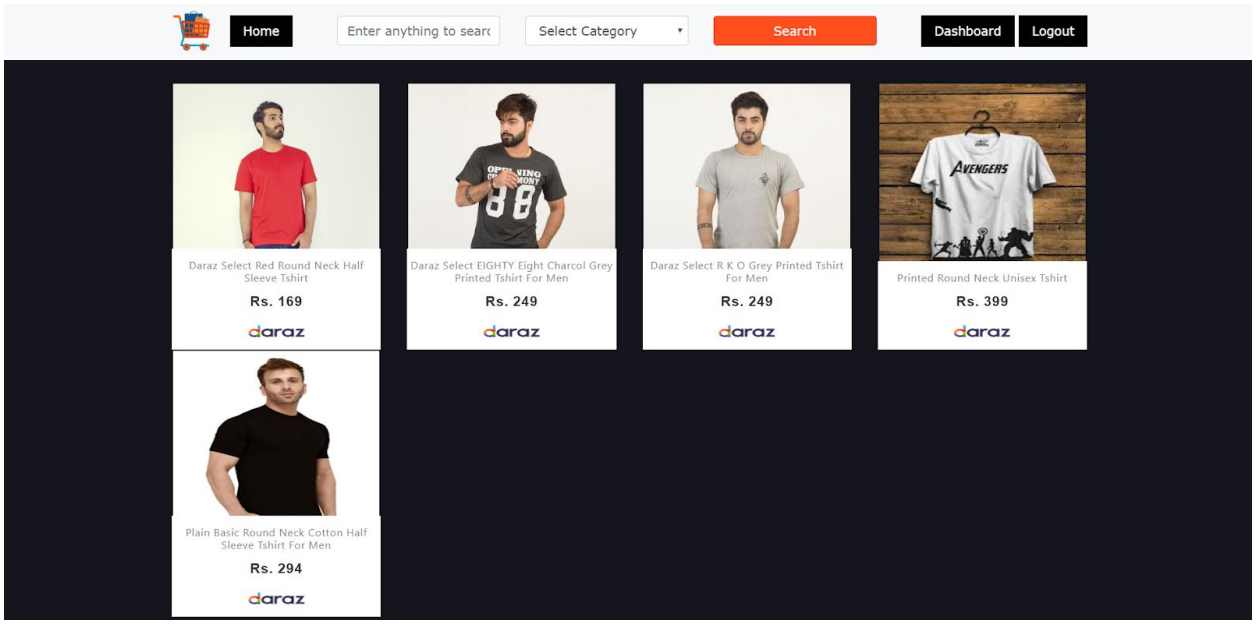
Login



Dashboard

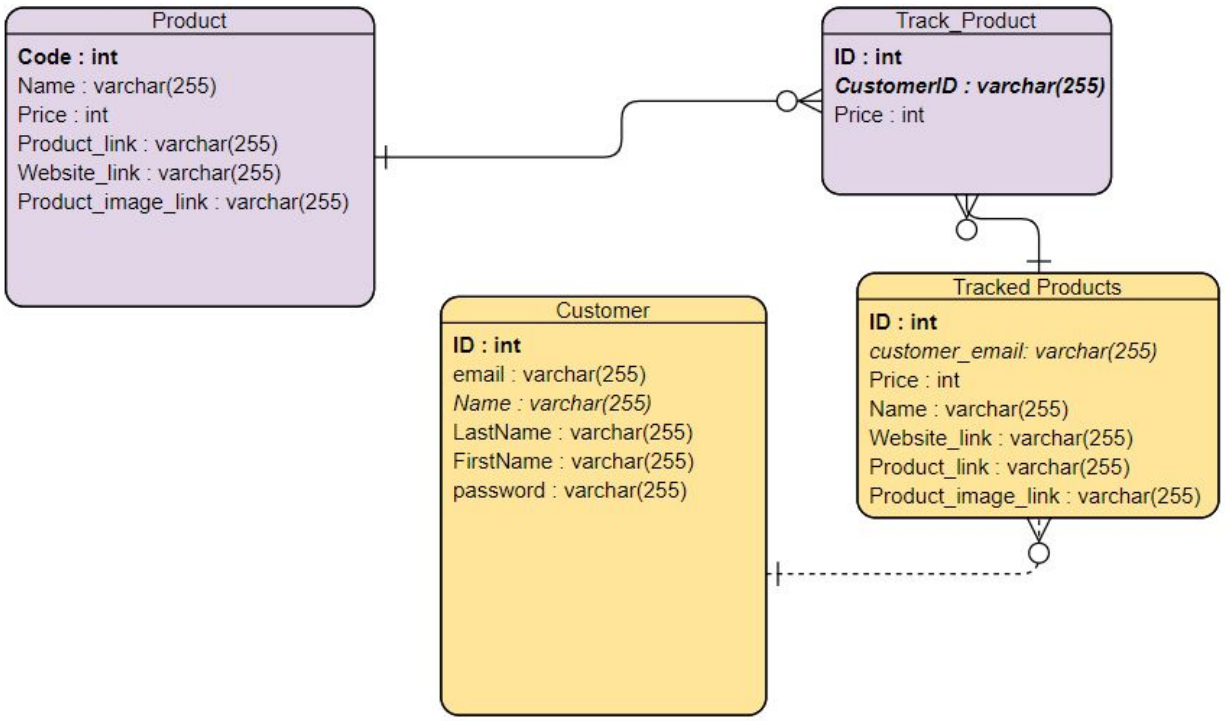


Search Result



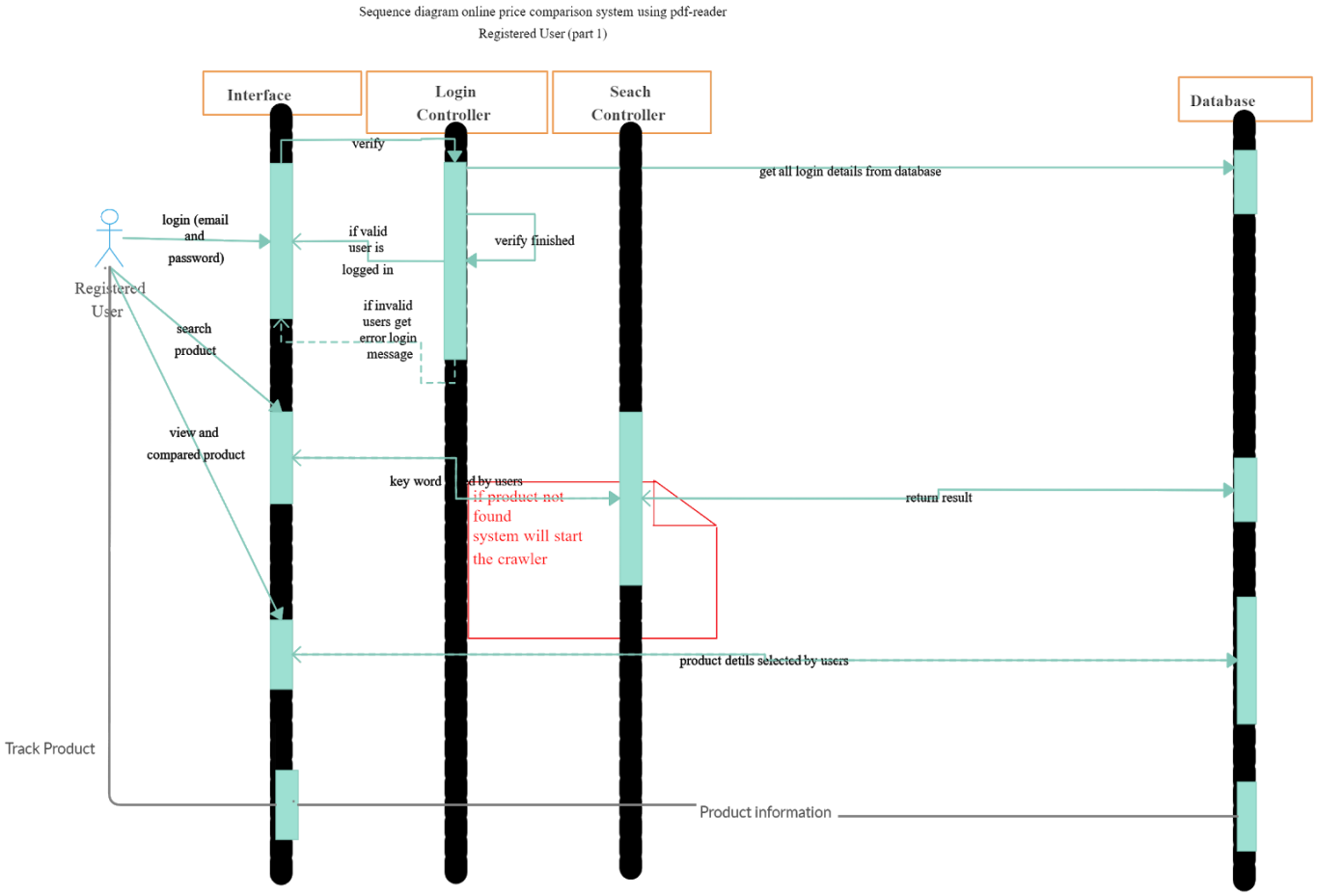
8.1. Database Design

ER Diagram:

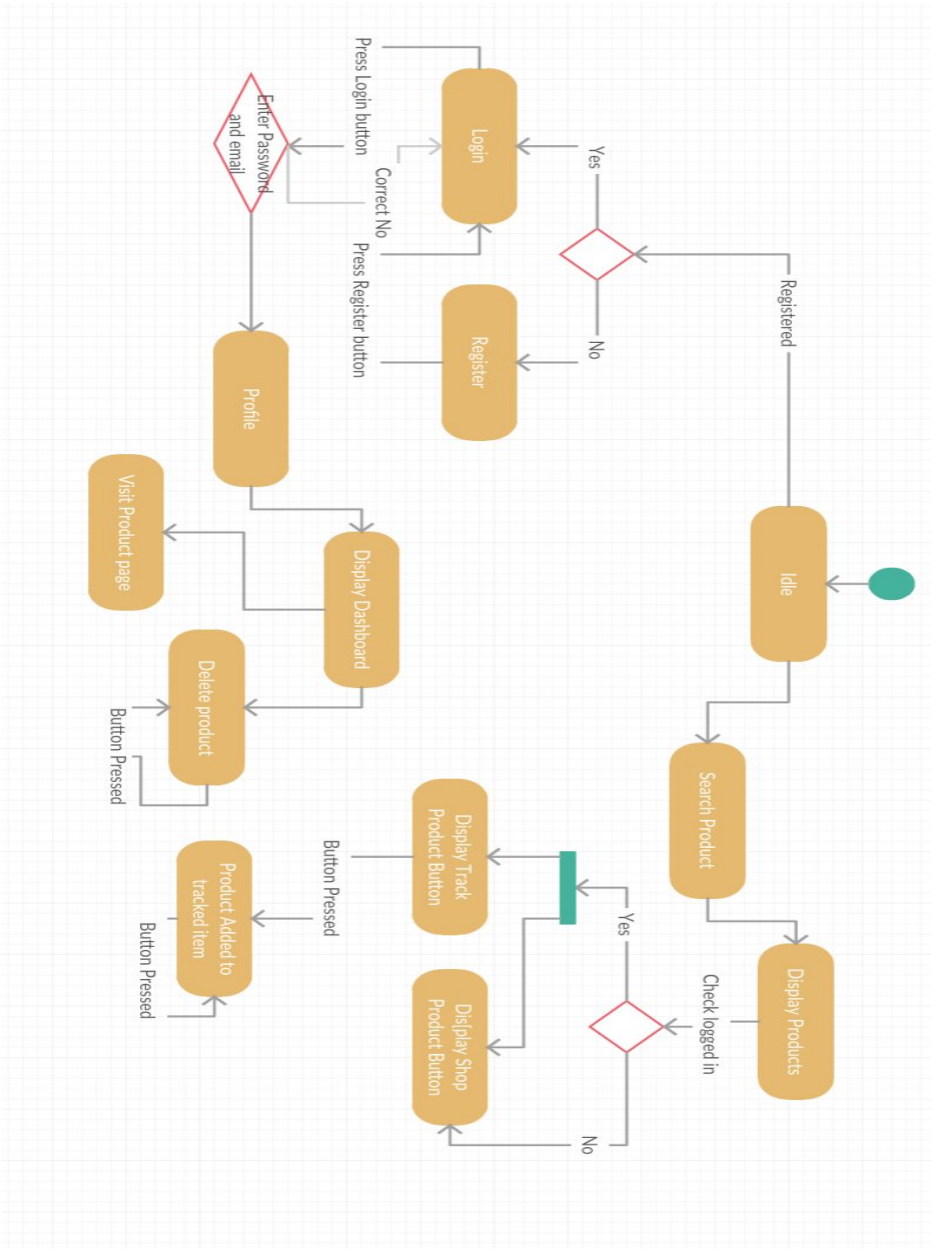


8.2. Application Design

8.2.1. Sequence Diagram:



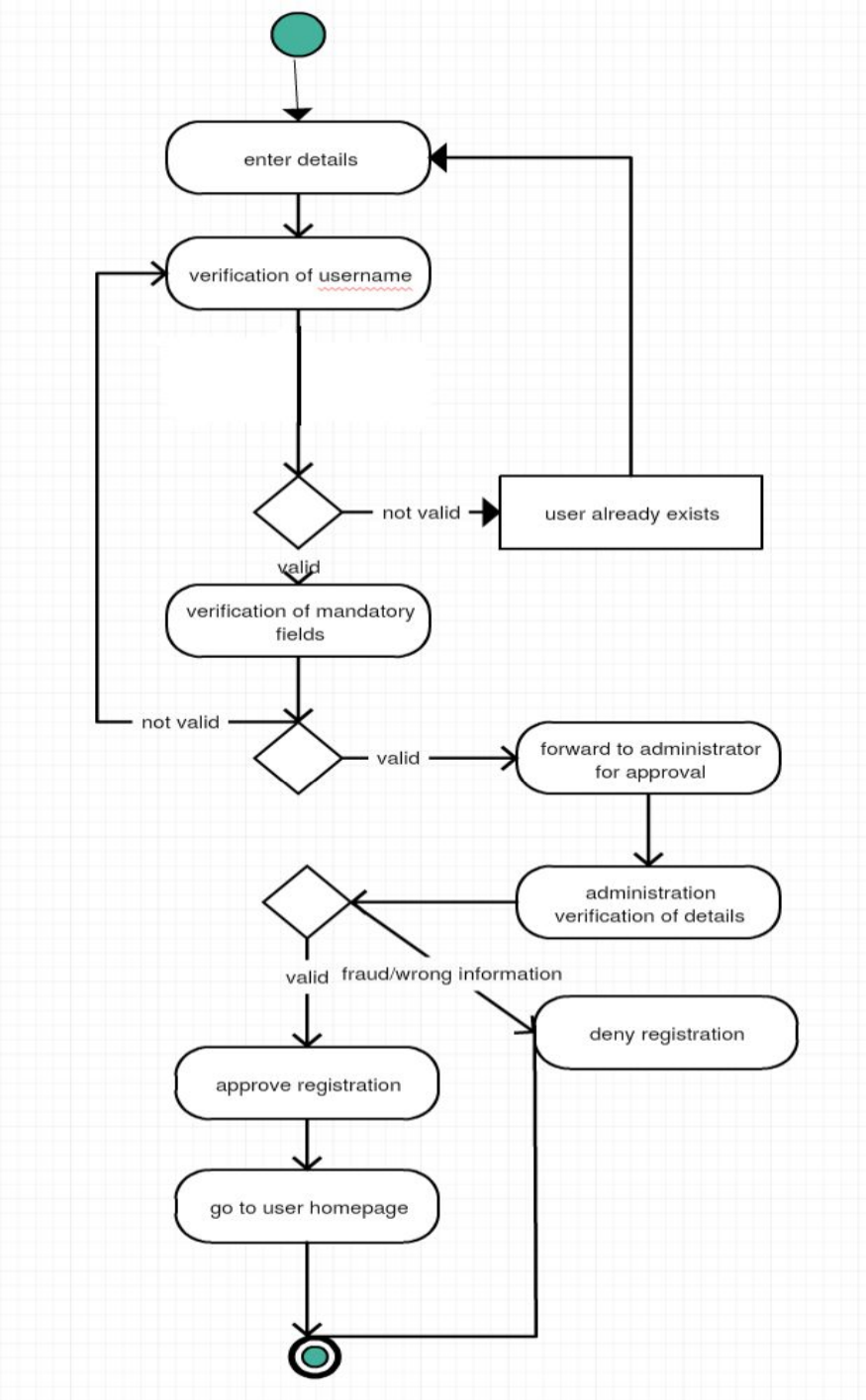
8.2.2. State Diagram:



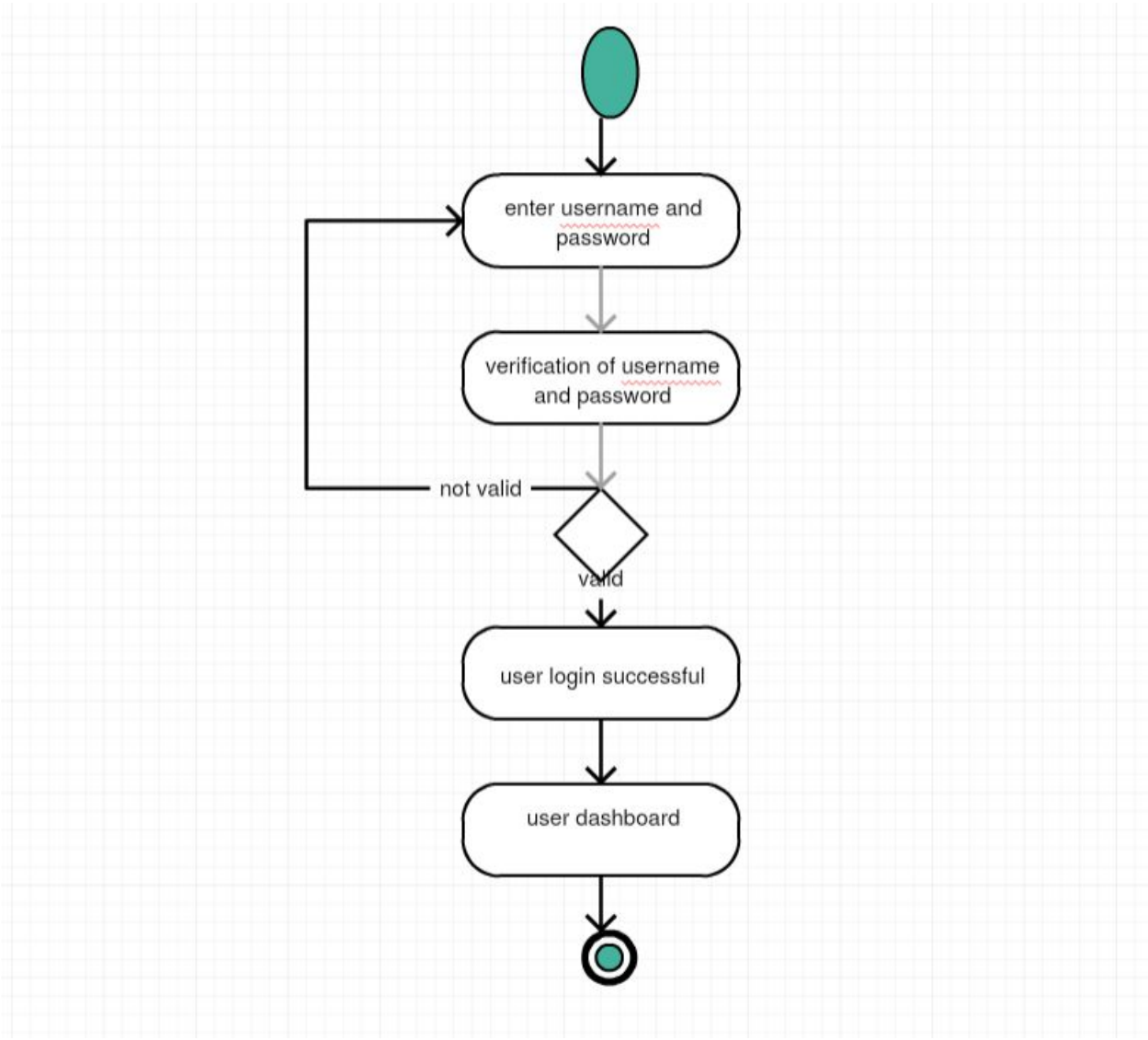
8.2.3. Use Case Diagram:

8.2.4. Activity Diagram:

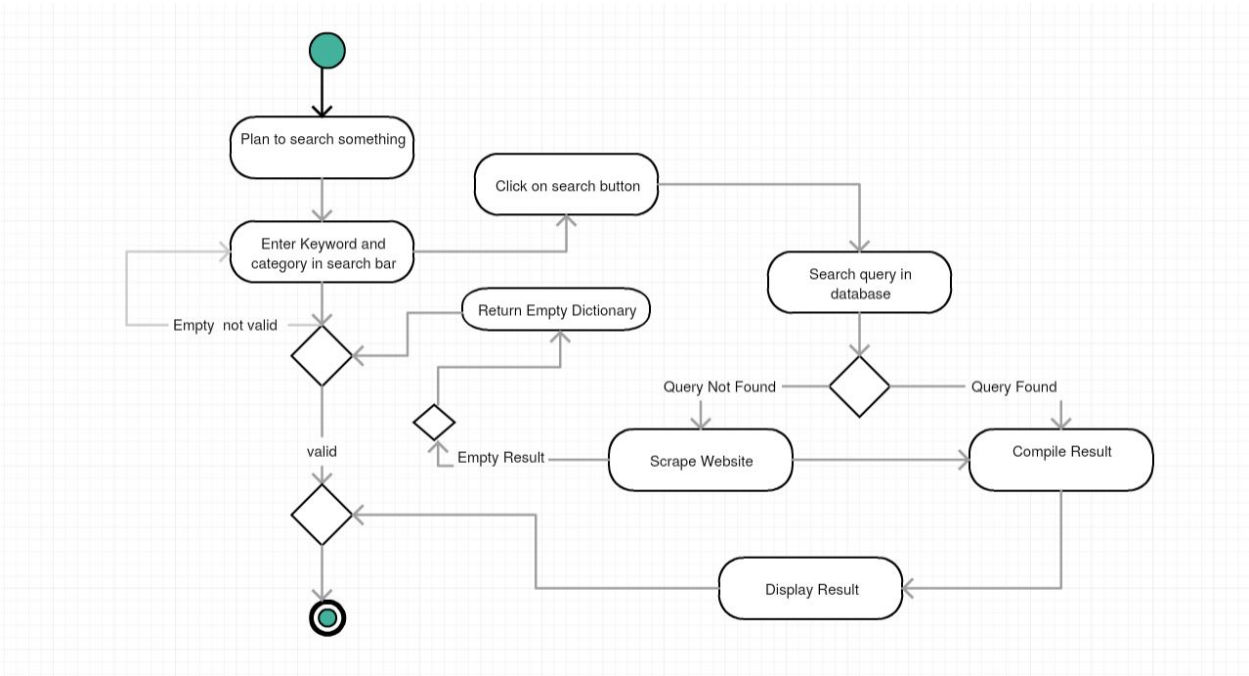
Register User:



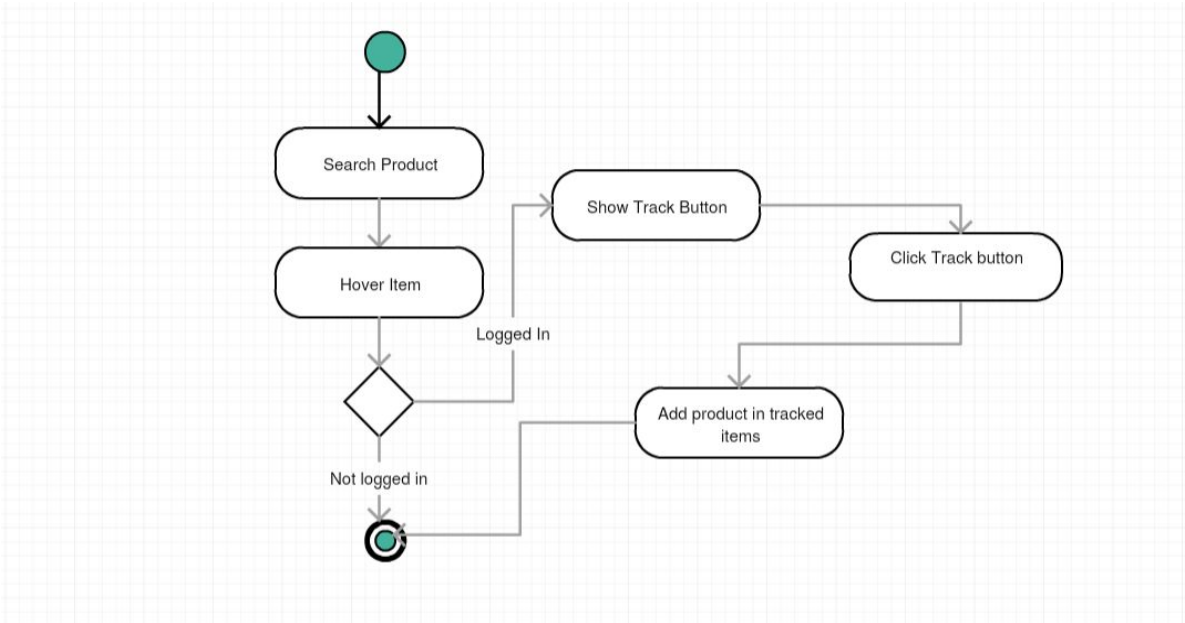
Login User:


enter username and password
verification of username and password
not valid
valid
user login successful
user dashboard

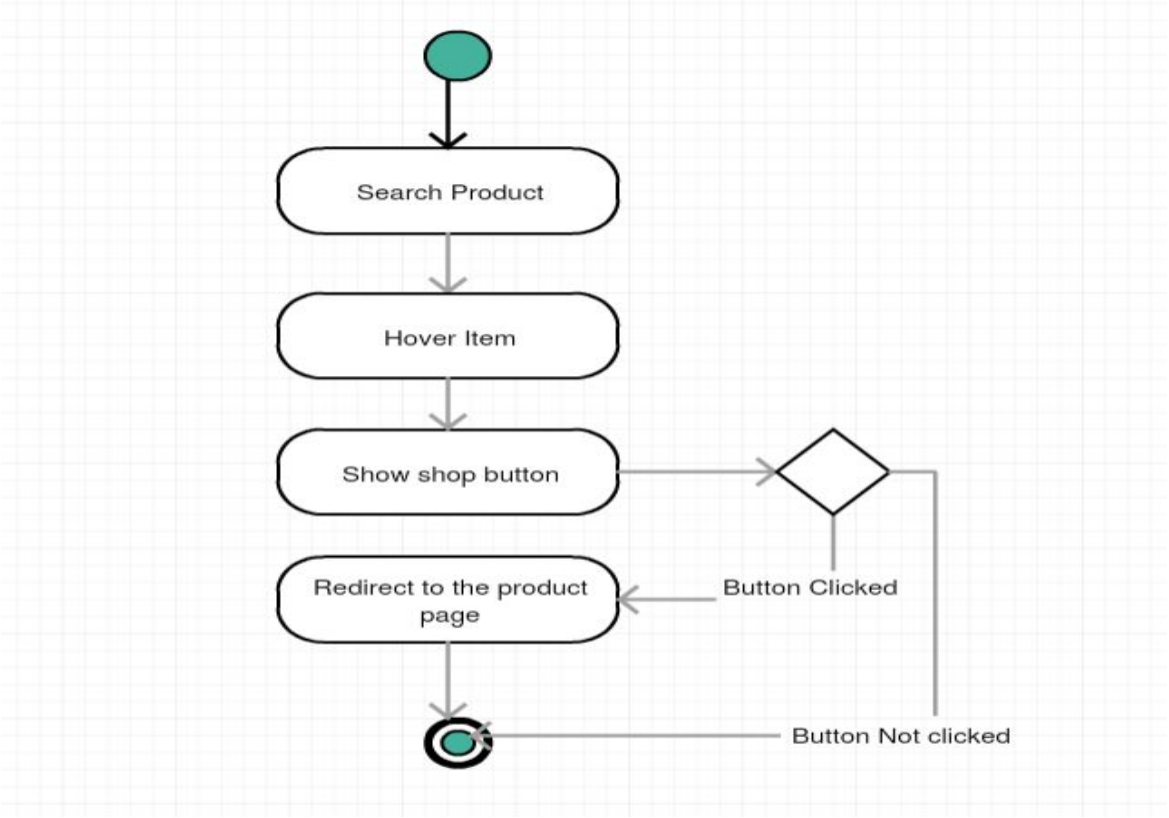
Search Product:



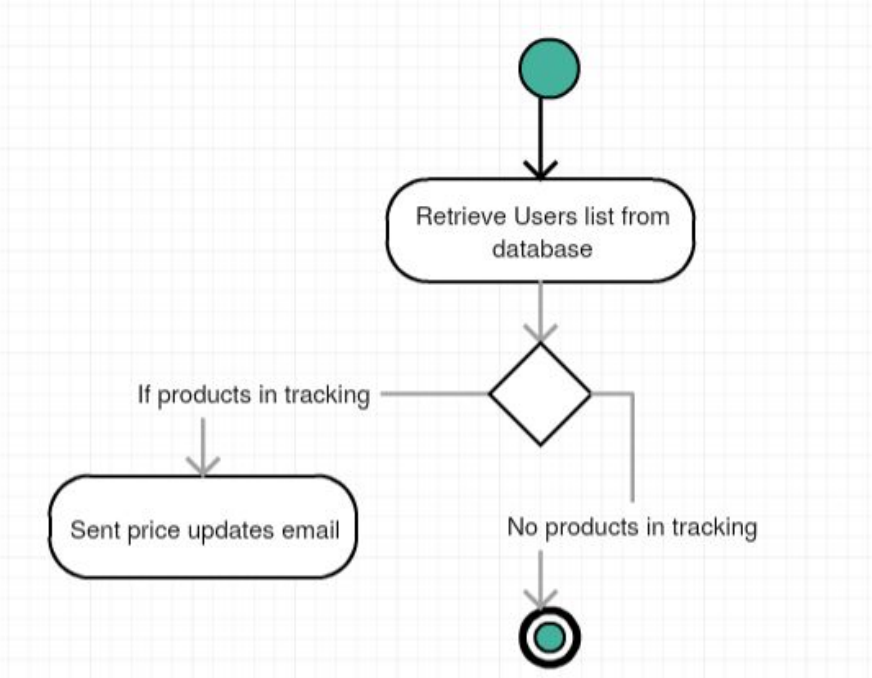
Track Product:



Shop Product:



Send Mail:



9. References

- MongoDB Documentation: https://docs.mongodb.com/
- Python Documentation: https://docs.python.org/
- Cartright.pk(Pakistani Price comparison website): https://www.cartright.pk/
- Selenium Webdriver: https://www.selenium.dev/
- Stackoverflow: https://stackoverflow.com/
- Flask Documentation: https://flask.palletsprojects.com/en/1.1.x/
- https://www.daraz.pk/
- http://yayvo.com/
- https://www.goto.com.pk/
- https://www.ishopping.pk/
- http://www.mega.pk/
- https://www.symbios.pk/
- https://hummart.com/
- https://www.shophive.com/

10. Appendices

Not Applicable