FreshFlora e-Commerce Website

with AI Text based Chatbot

Nikhilesh Kovvuri

3058058

Dr. Mario Kolberg

April 2023

Dissertation submitted in partial fulfilment for the degree of Bachelor of Science with Honours Software Engineering

Computing Science and Mathematics

University of Stirling

ABSTRACT

This dissertation discusses a project involving an AI-powered chatbot for use on an online store's website around the clock. As a result of the COVID-19 pandemic, there is a greater need than ever before for nutritious and wholesome food. This presents a golden opportunity for farmers, consumers, and the planet as a whole. To facilitate local producers' ability to communicate directly with consumers (B2C agricultural marketplace), it may be worthwhile to create an online buying and selling platform for organic food.
Many farmers opted to go digital during the outbreak, utilizing social media and other internet tools like WhatsApp, Facebook, and Twitter to help them cope with difficulties and boost their operations. However, not in terms of the accessibility of data sources and the value-driven information provided to clients. Recognizing the unfilled market space, social entrepreneurs are exploring the potential of the internet and the widespread use of mobile phones to facilitate direct trade between farmers and consumers. People will be motivated to make healthier dietary choices thanks to this project because of the abundance of resources available to them. Encourages long-term viability as well Third Primary Aim: Optimal Health and Happiness Focus on climate change mitigation and adaptation in Goal 13. In addition, No. 9: Industries, Innovation, and Infrastructure, receives assistance.

Technology being used to develop end to end application are latest in current trend and which was given each and every stage.
Chatbot AI used text based chatbot is highlight in this project for 24/7 communication to consumers to providing information.
Consumers able to locate seller & their location more information about the product & nutrient values of the product through search & through 24/7 available chatbot.
Sellers (Farmer) can add their products to their dashboard them self with complete information of their location & price /quantity upon approval of the product by admin able to choose by consumer directly.
Admin Who will be sole responsible for managing the application including user accounts and products, sales directing to seller.

ATTESTATION

I certify that this dissertation reports original work by me during my University Project except for the following:

The Wireframe design diagrams from figure 1 to figure 4 was created by the use of the Adobe XD Software.

The external Node JS Library used were package imports which are essentials for the project.[2]

The use of React and mongo DB were completely based on Tourtial and taken from the online sources. [6]-[9]

The Creation of the chatbot Dataset were completely from online sources[19]-[29].

I understand the nature of plagiarism, and I am aware of the academic misconduct policy of the University of Stirling. I certify that this dissertation report is my own original work.

Signature: Nikhilesh Kovvuri **Date:** 28/04/2023

ACKNOWLEDGEMENTS

I would like to thank my project supervisors Ms. Preetha & Dr. Mario for their guidance, support throughout the course of this project . A special Thanks to Dr. Mario for his encouragement through his jovial criticism (new supervisor after I came as SEM Visit to Stirling campus) his guidance and straight point to point information helped me to achieve my goal on time.

My special thanks to my friends and Family for continue support of testing feedback of Test cases & encouraging me in hard times for pushing me to work on latest technologies which used in this project .

Contents

1. INTRODUCTION

This document illustrates the scope of a B2C solution for an Agri crop venture which aims to connect business with fresh crop producers (Farmers). The document also provides a clear understanding of what all technologies and problems are focused on and especially what all steps and methods are taken into consideration for the problem to be solved. The main focus of the project is showing the user all available products and details about the product, nutrition values and Price. Consumer to choose the required product and can contact the Seller(Farmer). 24/7 AI Text based chatbot which supports providing Information about the product. Making Consumer getting in contact with farmer directly.

1.1 BACKGROUND AND CONTEXT

The e-Commerce Market is a leading producer of fresh products aiming for the health-centric food needs of the future. In the current state of business, the consumer is unable to determine whether the produce is organic or chemicalized. There are several setbacks in the business due to adulterer produce and this e-Commerce website should essentially help the Project to authenticate the produce and synthesize fresh and pure organic produce at a competitive price and predictive market demand for business continuity and expansion.

The main aim of the project is to serve where consumers can directly have contact with farmers sellers who sell their products .The main of the Project is to provide more information & nutrition facts about Fruits & vegetables in line with a healthy perspective and also provide the right information about the product of what they are looking for should be accurate & genuine, which can't be manipulated by wrong sources, Farmers to get value to their efforts through their Product. AI Text based Chatbot will be used to support consumers with instant information about Fruits & vegetables. In summary, a website is made which will reach out to customers directly from Farmers for buying Fruits & Vegetables. Get information on which location is available the most through insights. The value that Projects gets out of this e-commerce website is that in one place information & direct access to buying from the firm farmer.

1.2 SCOPE AND OBJECTIVES

Purpose of this project is to create user friendly website with AI Text based chatbot with standalone functionality. The ideal result is an eye-catching user interface of well-functioning systems that showcase the features of a hypothetical finished product, complete elements like

Responsive website: The key concentration is on building an intuitive interface & Eye-catching responsive design website. While placing a wide range of products in a manner that visitors find shopping a hassle-free experience.

AI Text based chatbot: Consumers can get information 24/7 on the availability of products & more information on different types of products available places & nutrition facts about product.

1.3 ACHIEVEMENTS

The e-commerce website with AI Text based chatbot is a demonstration of the systems and activities present that make up the main goal to be expected in a full release. The website is designed with Eye-catching and responsive, and various elements of the design and implementation have been created entirely with that in mind.

For the Project, 20 Test cases prepared with 4 users' roles for testing and the prototype evaluated using each Test case. The prototype adjusted based on the results of these tests and the tests performed again with corrected components fail to pass them. All test cases passed and the prototype considered fully functional hence released to the final Project development. Functionalities covered in this project are

AI Text based chatbot – 24/7 virtual assistance to provide information about the products.

Unregistered user to search about products available & detailed information about the product.

Registered consumer login (verified user status only) with userid & password and select product to buy it. Selected product will be available to seller (Farmer) dashboard direct.

Registered Seller (Farmer) adds their products information & price, details about the product and send for approval. Approved product will be available for consumer.

Admin who will be handling entire application like Product approval & maintaince user (consumer /seller(Farmer)) and allocating product to dispatch etc..

The final product met the expectations of the Project by satisfying the functional needs. The Project can be asked to provide feedback on the final prototype to be able to assess satisfaction. Initial planning to use basic technology HTML/CSS & SQL as database and estimated limited time line.

Upon exploring the latest technologies along with academics, it's been decided to use latest available technologies and optimize the solution hence its used CSS/JavaScript with React library for eye catching front end and upon evaluating the data volume & growth of the usage Database changed from SQL to NoSQL (MongoDB) in considering below aspects .

- Reduce efforts for development from developer hence cost optimization.
- Storage Cost optimization
- Unstructured data collected through chatbot and other channels to be queried.
- Data input by sellers being more in numbers can't predict in sizing of servers, amount of data to store for this application and query increased.
- To adopt the changes required quickly & rapidly to application till database level NoSQL gives flexibility in creating new queries.
- NoSQL have capability to cloud computing to distribute data to multiple servers & regions.

1.4 Overview of Dissertation

This dissertation is comprised of State-of-the-art that is defined with Functional & Technical Elements.

- **Functional** - AI Text based chatbot / e-Commerce website
- **Technical** - Client side /Server Side

  - **AI Text based chatbot -** which is built on AI, is an emerging idea that places an emphasis on the dialogue between humans and machines. Users are increasingly turning to speech-enabled interfaces, intelligent virtual assistants, and chatbots in search of a more humanlike interaction. This e-Commerce platform's chatbot was built to provide business partners with instantaneous responses via chat Interaction. In order to attract new customers and make sales, a quick reaction time is essential. With a possible 99% uptime, AI-based chatbots will replace human engagement. This has the potential to enhance both business dependability and the creation of new opportunities.[1]

  - **e-Commerce website -** e-Commerce website with good user experienced navigations for user types consumer /seller and admin. Also, provision for non-registered users to search about products & information. AI Text Based Chatbot available through website for 24/7 text chat with users.

- **Technical Client side-** The computer languages and techniques used to create a website or web application's user interface are referred to as "client-side web development." This refers to the HTML, CSS, and JavaScript code that the user's web browser executes. The content of a web page is organized using HTML, or Hypertext Markup Language. This comprises the headings, paragraphs, pictures, links, and other page components.

  CSS: A web page's HTML content is styled using Cascading Style Sheets (CSS). Fonts, colors, layout, and responsiveness are a few examples of them. JavaScript is a computer language that enhances a web page's functionality and interactivity. Animations, form validation, and dynamic content that alters based on user input or other circumstances are examples of this type of content. [11]

- **Technical Server side -** Cloud databased No SQL (Mongo DB).

  The term "server-side web development" pertains to the set of programming languages and tools that are utilized in constructing the backend or server-side of a website or web application. The section of the website in question is accountable for the handling of user requests, as well as the storage and retrieval of data. Additionally, it is responsible for the creation of dynamic content, which is subsequently transmitted to the client-side.

  The following are essential elements of web development on the server-side:

  Server-side scripting languages are utilized to produce dynamic web pages that can generate content in real-time in response to user requests. Several commonly used server-side scripting languages are PHP, Python, Ruby, and Java. [18]

  Databases are frequently utilized in websites and web applications to facilitate the storage and retrieval of data. Their primary function is to manage information. MySQL, MongoDB, and PostgreSQL are among the frequently utilized database systems in web development.

  Web servers are software applications that operate on a server and manage incoming requests from clients. Some of the commonly used web servers are Apache, Nginx, and Microsoft IIS.

  APIs, short for Application Programming Interfaces, are a collection of regulations and procedures that facilitate the exchange of information between various software applications. Application Programming Interfaces (APIs) are frequently employed in the realm of web development to facilitate the integration of disparate services and systems. [6]

2. STATE OF THE ART

2.1 CHATBOT

The AI Text based chatbot situated on the electronic commerce enterprise's webpage for Fresh Flora the AI Text based chatbot , is capable of addressing consumer inquiries and accommodating requests at any given time. The implementation of a round-the-clock digital brand representative for the business is expected to enhance customer engagement, foster customer loyalty, and drive website traffic. The implementation of a proactive chatbot can potentially increase customer engagement and retention on a website. By encouraging customers to remain on the site, explore available resources, and interact with the business, the chatbot may effectively promote desired customer behaviors [1][19][21].

The hyperlink furnished by the automated conversational agent navigates the user to the specific and distinct webpage on the platform that the customer is obligated to visit. A website featuring a product or service is furnished with an automated conversational agent that initiates interaction with users to facilitate the completion of an application form. Furthermore, the webpage includes a downloadable button that provides access to product or service[1][19][21]..

2.1.1 Purpose of the AI chatbot

AI chatbot used in this project for customer service and support, for marketing and sales, for customer engagement, questionnaires, FAQ tools, making orders and general information, for collecting customer feedback[1][19][21]..

2.1.2 Who are using chatbot

The target groups are Customers, business wants to give 24/7 customer service and support, selling products and services and advertising them in a more interesting and engaging way to customers hence decided an AI chatbot is the right solution. [1][19][21].

2.1.3 Type of the chatbot

In contemporary times, it is noteworthy that even the elderly demographic has embraced messaging as a means of communication, commencing with conversing with their grandchildren via chat and progressing to online shopping and service inquiries, which were necessitated by the pandemic. [1][19][21].Consequently, a chatbot that operates on a keyword-based system was deemed appropriate. Chatbots that operate based on keywords are designed to analyze text for specific combinations of keywords and produce a response that is pertinent to the input. This feature facilitates greater conversational flexibility and authenticity, enabling users to pose inquiries of a more intricate nature. [1][19][21].

2.2 WEBSITE

2.2.1 STATIC WEBSITE

A static website is one that has Web pages stored on the server in the format that is sent to a client Web browser. Primarily coded with REACT which comprises of HTML (Hypertext Markup Language) and CSS (Cascading Style Sheet) some pages in current project website used static pages these pages have information about the products and the information to be updated by admin as on when new addition [6][7][8][9].

2.2.2 INTERACTIVE WEBSITE

Interactive sites allow for interactivity between the site owner and site visitors or users. In this project (website) all the pages are interactive when consumer register data will go back to server and fulfill required conditions & verify for user type & grant permission based on role defined in backend.

Also, when seller update product in profile data will be added to dashboard and available, upon approvals of the admin source data automatically available to consumer for selecting and buying the same. When consumer selected product same data will be available to seller who is the product owner.

2.2.3 E-COMMERCE (ELECTRONIC COMMERCE)

e-commerce (electronic commerce) is the activity electronically buying or selling of products on online services or over the Internet. Products or services are purchased directly through the website. Consumer searching & buying the product through website (Interactive website) .

2.2.4 WEBSITE FUNCTIONALITY USER TYPES

Unregistered User: user can use website for searching products, general information & Chabot text in case if customer wishes to buy any product its mandatory to register & provide some information and create account.

Registered user: upon navigating website for products & general information if want to buy any product user should be registered as Consumer. If user want to sell any product through website register as seller .

2.2.5 WEBSITE FUNCTIONALITY USER ROLES

- Consumer – Buyer who looking for product to buy from seller directly .
- Seller (Farmer or Wholesaler) seller who provide product in sale with availability and content of the Product details (logged in user).
- Administrator – Who manages the whole process (sellers /buyers records /approvals) manage all details and support to sellers & buyers.
- AI Text Based Chatbot – 24/7 text-based assistant to provide information to users.

2.2.6 USER REGISTRATION

User registration happened by selecting User Type (consumer /Seller)

By entering details: Name /email/phone number /address/City/Password

Success registration message will receive by User & user need to be verified by administrator & OTP sent to email.

2.2.7 User logon

User to login by entering registered user ID & password based on User type upon login user will be routed to dashboard based on type of user Seller or Consumer.

2.2.8 – User forgot password

Its normal user tendency to forgot hence provision provided to retrieve their password by entering registered emails.

2.2.9– Consumer Profile

As registered user consumer login with userid & Password landing page to Dashboard where consumer can view & update details add address & track the order.

Consumer can search products through search functionality for required product & if interested view details about product & interested to buy add to cart .For delivery of the product can choose existing address or add new address. consumer can delete product from cart if changed decision.

The Product chosen by the consumer will be added to Seller (Farmer) dashboard whose product is selected. In this way consumer will directly be interacting and Reaching the Seller(Farmer).

2.2.10 – Seller(Farmer) Profile

As registered user Seller (Farmer) login with user ID & Password landing page to Dashboard where supplier can view & update details add products & view orders.

Add product: Seller (Farmer) wish to add his product to market choose add product option and enter all the details about product & product picture & URL , select the category of product & submit. Product will be endorsed by Administrator (approve /reject) approved product will be available to consumer for purchase. While consumer selects the product add to cart same will be updated to seller dashboard this way consumer reaching direct to Seller (Farmer) for buying product. Here website facilitating to connect both of them consumer & seller.

2.2.11 – Administrator

Administrator can perform below activity through admin login

- Category & subcategories
- Add Products & manage products
- Manage all users (sellers /consumers)
- Manage orders

Add category: By entering Category name, Category URL ,Category Image ,Category Description click on add category will create new category to all category's gallery .

Maintain Category: From all category's gallery admin can change category details or delete category if it's not in use condition apply (Image Size).

Add Subcategory : select category from dropdown list and entering Sub Category name, Sub Category URL , Sub Category Image , Sub Category Description click on add Sub category will create new Sub category to all Sub categories gallery.

Maintain Sub Category : From all category's gallery admin can change Sub category details or delete category. Category & subcategory are the primary inputs for supplier to choose while adding Product to the store .

Add Product : other than supplier Admin also can enter product by choosing category & subcategory Add Product Tile /description /Product URL /Price /Location /Image – then Submit product will add to Product Gallery.

Manage Product : From all products gallery admin can change Sub category details or delete category.

Manage Sellers : Registered sellers' data to be verified by admin & take action approve or reject only approved seller can use website to add their products .

Manage seller products : Registered sellers can add their products to gallery which will be received by admin, upon verification of product admin take action approve /reject /keep pending . Approved product will be added to availability list to consumer for buying through cart.

Manage Consumers: All registered users list will be available to admin for further action if required

Manage Orders: orders placed by consumer will be visible through dashboard Order ID /Consumer email/phone number/price/date/status/Approve /Reject/ full details view of the order of the product. upon approval order will be available to seller for delivery of the product.

2.2.12 – Responsive Webpages

In this project, focused on making the website responsive is more prioritized. The functionality will be applied in this as nowadays responsive web design is a widely accepted and familiar concept in website design. This concept not only enriches the user experience but also eliminates the turnaround time to build native applications in various devices and environments. Responsive web design is all about creating web pages that share similar user experiences on all devices!

Responsive Web Design shall be developed using HTML and CSS to automatically resize, hide, shrink, or enlarge the website and enforce similarity on all devices (desktops, tablets, and phones).

2.2.13 – Home page

Home page is a Static page with changeable slider data in this page remain same till further changes. this doesn't have any backend content . React framework. Its primary function is to display a carousel slider and a category slider on a webpage. We can add as many as sliders to UI depending on requirement. Top products & latest products updated latest will be render on home page .

2.2.14 – Contact us

Contact us is a Static page with contact details in this page remain same till further changes. this doesn't have any backend content . React framework. Its primary function is to display contact information email phone number address etc. on a webpage

2.2.15 – search

About also is a Static page with visions/mission/values in this page remain same till further changes. this doesn't have any backend content .

2.2.16 – shop

Shop page have actual functionality of interactive website from shop page user will be directed to Search product /Buy product /Register /Login etc.

Wireframe OF the Design

a. Homepage

Figure 1 – Wire Frame of Home Page

b. Login Page Seller (farmer)/Consumer

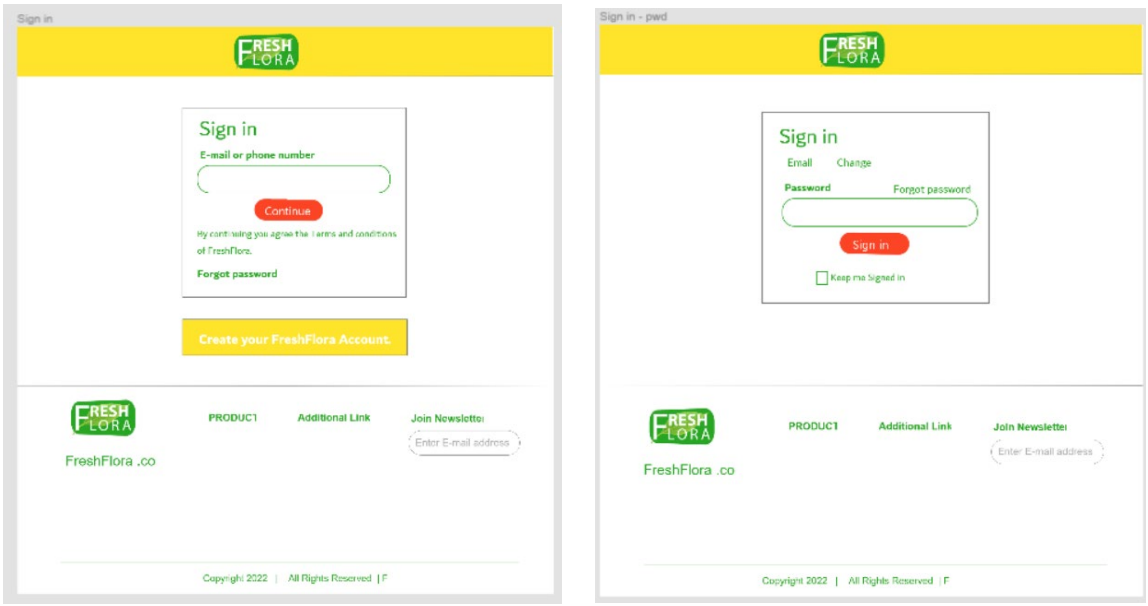


Figure 2 – Wire Frame of Login Page

c. Sign up Page

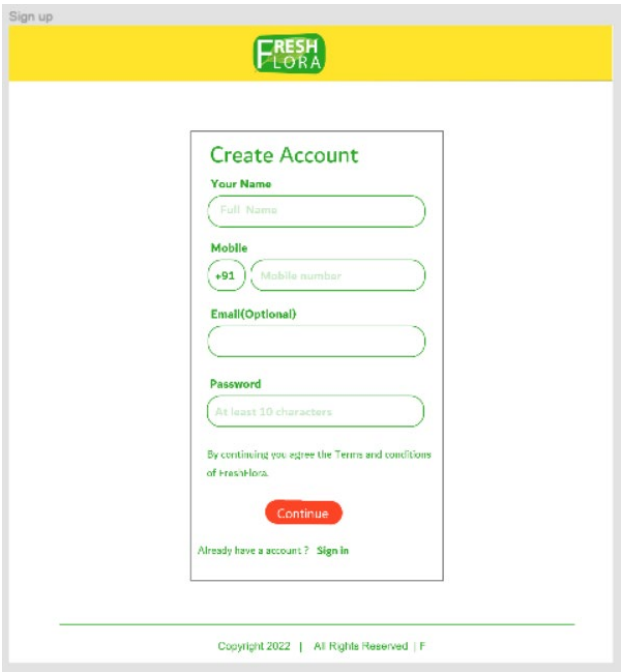


Figure 3 – Wire frame of Signup Page

d. Dashboard Page

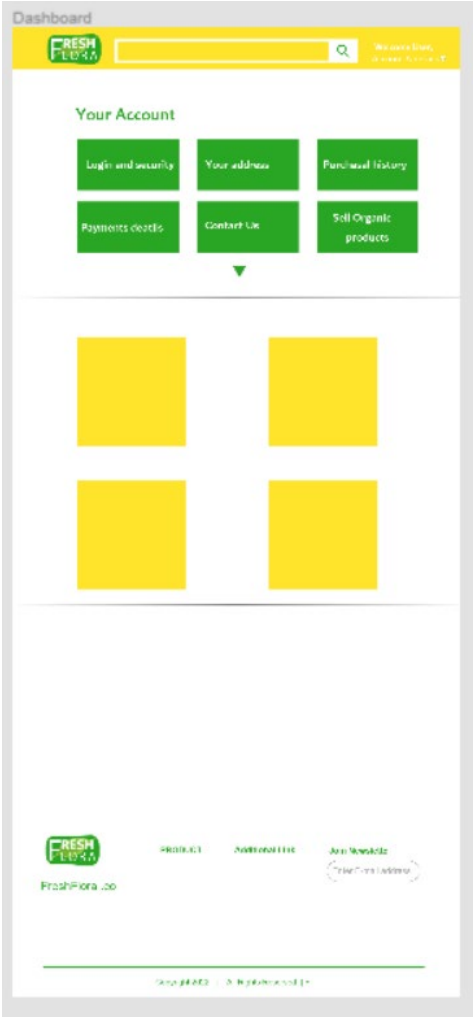


Figure 4 – Wire Frame of Dashboard

3. TECHNICAL DETAILS

3.1 Chatbot

The chatbot design for this e-Commerce website which will allow instant response in chats as well. This will act as standalone virtual assistant for any business response time is very crucial to generate leads and Business opportunities. The AI Based chatbots will substitute human interaction with possible 99% uptime. This can essentially optimize business reliability and opportunity generation. [19][21]

The whole dataset of the chatbot is created from a website (Welcome to APEDA)which provide the statistics and the details related to the fruits and vegetables. Also, to be informed that the chatbot dataset is trained only based on the India country as the main focus of the website for India.[27][26][28][17].

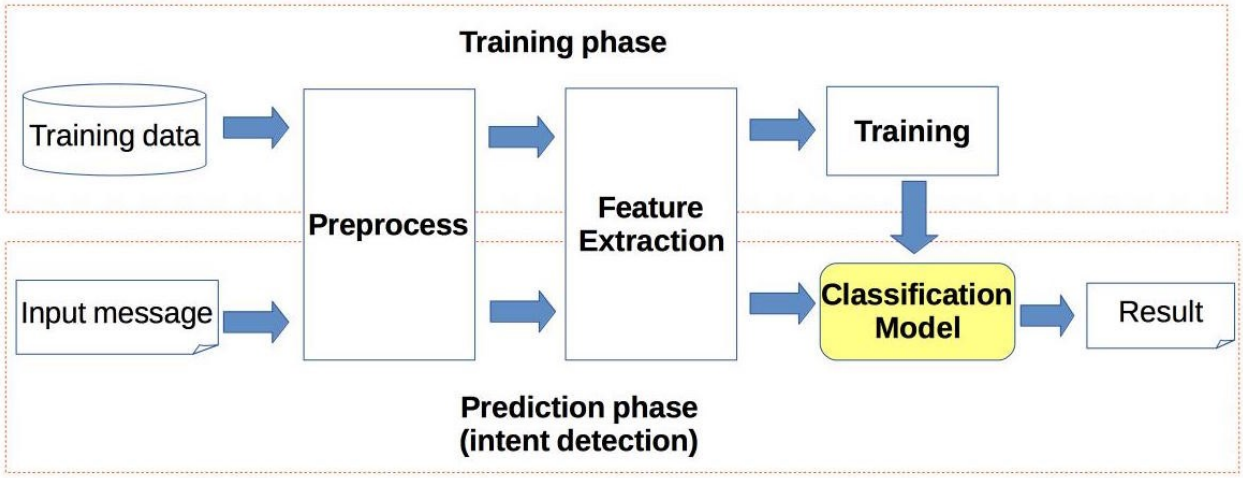


Figure 5 – Process of creation And Training Chatbot

3.1.1 PYTHON FOR CHATBOT

Python is a widely-used high-level programming language that has many different uses. Some of these include web development, scientific computing, and AI.[19]

Python is widely used for chatbot creation since it provides numerous tools and frameworks that simplify the process of creating conversational user interfaces. Natural Language Toolkit (NLTK), spaCy, and TensorFlow are just a few of the widely-used libraries for creating chatbots. [19]

These libraries allow programmers to create chatbots with features including natural language understanding, user input processing, and question answering. Tokenization, part-of-speech tagging, and entity recognition are just a few of the many NLP tasks that can be accomplished by programmers with the help of libraries like NLTK and spaCy. [19]

Further, deep learning models may be developed and trained in TensorFlow for chatbot creation, allowing the chatbot to adapt to individual users and their needs. [19]

3.1.2 NLP

Natural language processing (NLP) is a subfield of linguistics that makes it possible for machines to comprehend written and spoken language in much the same way that humans do. This is the backbone of the industry, allowing chatbots to understand and answer questions posed in written or spoken form. [19][21]
The initial stage is to train a chatbot through a series of chats and provide it with relevant data. Then, the NLP engine analyses the client's keywords and intent to determine what the consumer needs when they ask a query. The chatbot learns from the customer's comments after the discussion is done. [19][21]
An entity is a device used to derive the values of parameters from free-text user inputs. Take the question, "What are the different varieties of onions?" as an illustration. The goal of the user is to learn about the many species of onions. Different types of onions are the entity value. This leads to the request for "onion varieties." Both the system and the developer have the option of defining entities. Such as the entity in the system. The process of extracting domain items from a phrase, sometimes known as the "slot-filling problem," is stated as a sequential tagging problem. [19][21]
Finally, contexts are strings that keep track of the object's context when the user makes a reference to it. A user may, for instance, utilize an object that was just declared in the next statement. Someone may search for "organic onions." In this case, the onion is the context to be stored so that the intent "organic" may be invoked on the context "onion" upon the next input from the user. [19][21]

3.1.3 **Nltk_utils.py**

a. Tokenize

The tokenize function is the first step in text preprocessing, as it breaks down the input text into smaller units that can be further processed.

b. Stem

The stem function is used to reduce the number of unique words in the input text, which helps to reduce the dimensionality of the input data and improve the efficiency of the model.

c. Bag of words

The bag_of_words function is an important technique for converting text data into a numerical format that can be used as input to machine learning models.

```python
import nltk
nltk.download('punkt')

from nltk.tokenize import word_tokenize

# define some sample responses
responses = {
  "Hi": "Hello, how can I help you?",
  "What is your name?": "My name is Chatbot.",
  "How are you?": "I'm doing well, thank you. How about you?",
  "Bye": "Goodbye, have a nice day!",
}

# define a function to process user input
def process_input(input_text):
  tokens = word_tokenize(input_text.lower())
  for token in tokens:
    if token in responses:
      return responses[token]
  return "I'm sorry, I don't understand. Please try again."

# main loop
while True:
  user_input = input("User: ")
  chatbot_response = process_input(user_input)
  print("Chatbot: " + chatbot_response)
```

Figure 6 – Example code of NLTK elements used in Python

3.1.4 **Train Data.py**

The Main Purpose of the of the train data set is preprocessing of text data, the creation of a custom Dataset class for batch processing, and the definition and training of a neural network for the chatbot task. These are key components of any machine learning pipeline and demonstrate how PyTorch can be used to train complex models on text data. [19][28]

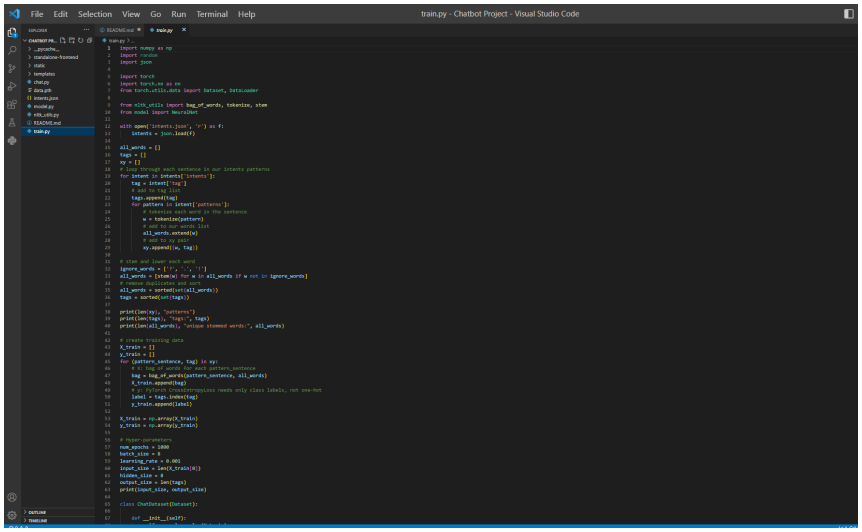


Figure 7 – Code of the Train Data

3.1.5 **App.js**

a. Flask

Flask is a web framework for building web applications using the Python programming language. In this code, Flask is used to create a simple web application that allows users to interact with a chatbot.

b. Flask Route

```python
@app.post("/predict")
def predict():
    text = request.get_json().get("message")
```

Figure 8 – Code of Flask route

When a POST request is received, the predict function is called. The function first retrieves the message data from the JSON payload using the request.get_json() method, and then extracts the message text using the .get() method on the resulting dictionary, using the key "message".[26][27]

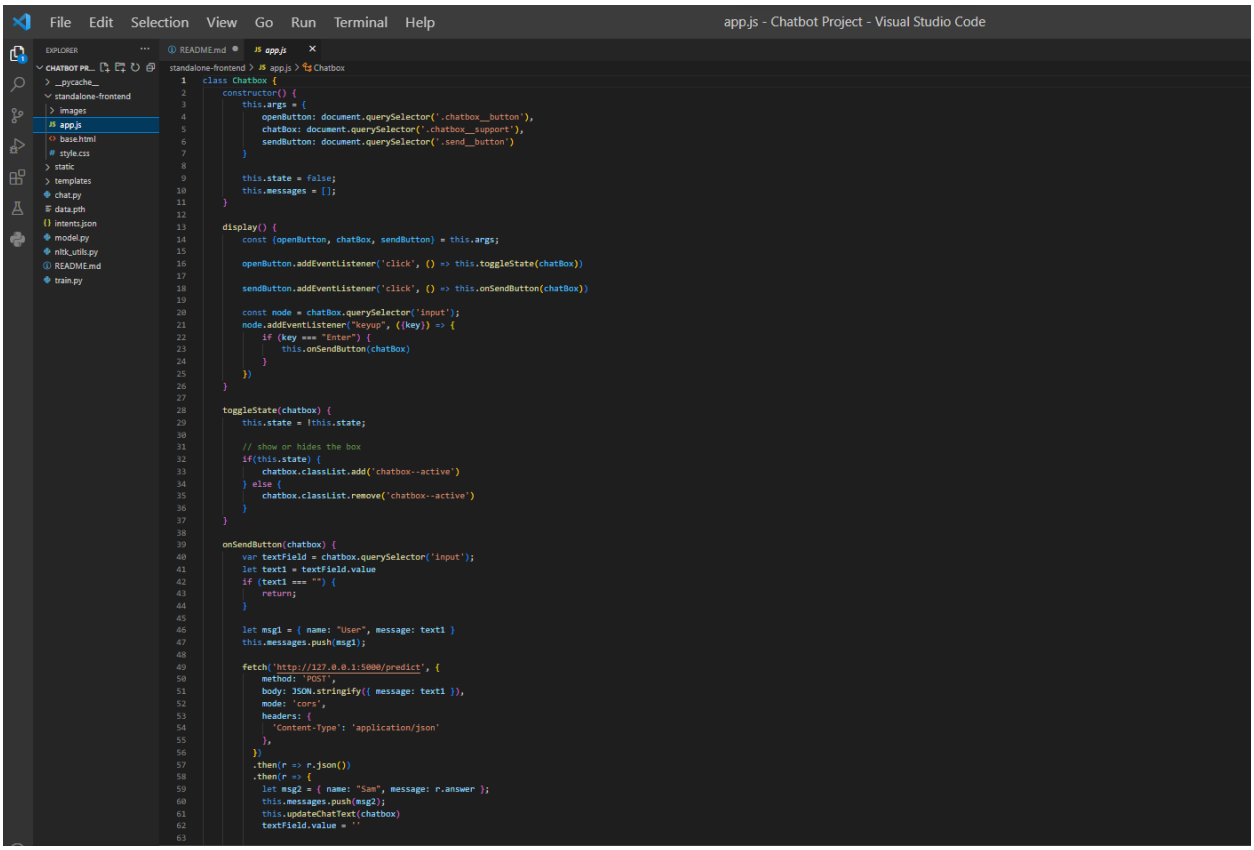


Figure 9 – JavaScript Code of the App which executes the code

3.1.5 **Intents . json (dataset) [17]**

a) Patterns

The patterns in the dataset (Intents) contains the training questions for the chatbot . These questions are basically the bot question which matches with the user inputs to provide answers.

b) Responses

These are the predefined output which is given to the User based on query which he posted to chatbot the syntax which is matched to pattern (Questions saved on the patterns.)

Figure 10 - Sample Code Screenshot of Intents File

3.2 WEBSITE FRONT END –CLIENT SIDE

The Web has been built on CSS (Design), HTML (content), and JavaScript (Logic) with the use of API.

3.2.1 JavaScript with (React)

The scripting language JavaScript may be used to create ever-changing content, manage multimedia, animate graphics, and do just about anything else [6][7][8][9].. It's a well-known and popular tool for creating websites. Using Java script, you may create a website that is both visually appealing and functional. JavaScript, sometimes known as JS, is a programming language that may be used in conjunction with Cascading Style Sheets (CSS) and Hyper Text Markup Language (HTML)[6][7][8][9]. Project-side JavaScript for webpage functionality is used by 98% of websites as of 2022, and third-party libraries are regularly incorporated.

JavaScript allows us to create numerous functions for usage in the project's front-end development. The Document Object Model (DOM) is used by Java Scripts, which are embedded in or linked to HTML documents. All mainstream browsers include native support for executing JavaScript scripts on the user's computer or mobile device[6][7][8][9].

Websites that use Ajax or a WebSocket to load new material without reloading the page benefit from the use of JAVA script. Social media users, for instance, can communicate with one another in real time without leaving the current page they're on. To create an interface, you can utilize the JavaScript package called React. The front-end UI must be able to process individual elements, such as buttons, TEXT, pictures, etc. [6][7][8][9]. Using React, you can turn these modular building blocks into reusable, nestable building blocks. The screen from a website or mobileapps will be segmented into components according on the type of device being used to access the content. With React, you can create your own "components" out of HTML, CSS, and JavaScript.

3.2.1.1 React Jsx mark up

JavaScript now has an extension called JSX that lets you create HTML-like markup within a JavaScript file. Previously, HTML was generated by JavaScript. Because of this, in React, components house both rendering functionality and markup. In the browser, react renders each component, which is a JavaScript function that may also include markup. To express this markup, react components employ a syntactic extension known as JSX. JSX is similar to HTML in appearance but is more stringent and can show dynamic content. One of the easiest ways to grasp this is to make the transition from HTML to JSX syntax. [6][3]

3.2.1.2 React Components

React UI (User interface) is predicated on the idea of components. A React component is a markup-sprinkler JavaScript function.

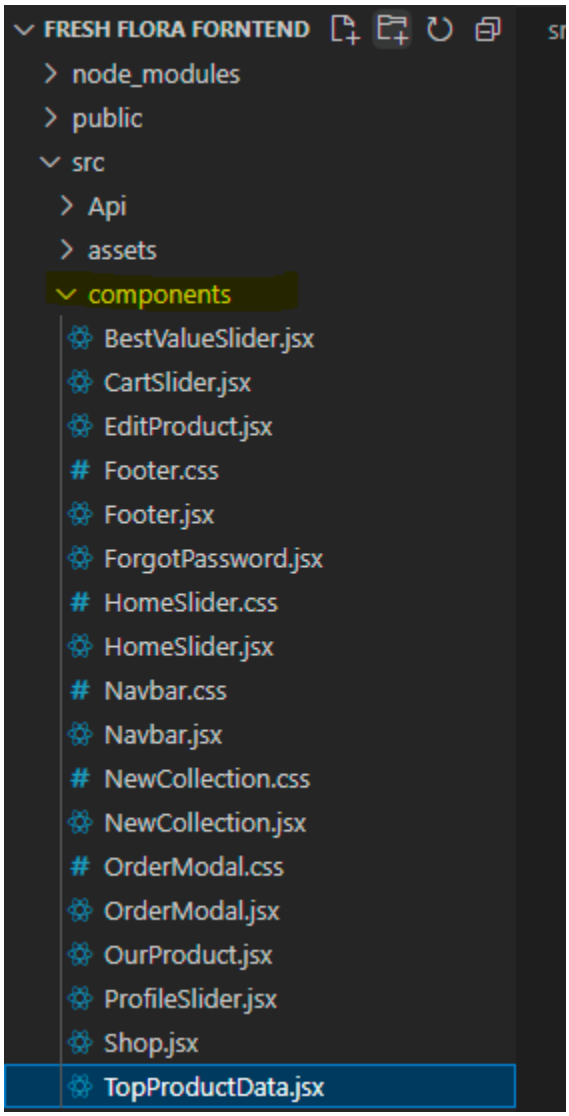


Figure 11 – Component JavaScript and CSS

3.2.1.3 Component forgot password

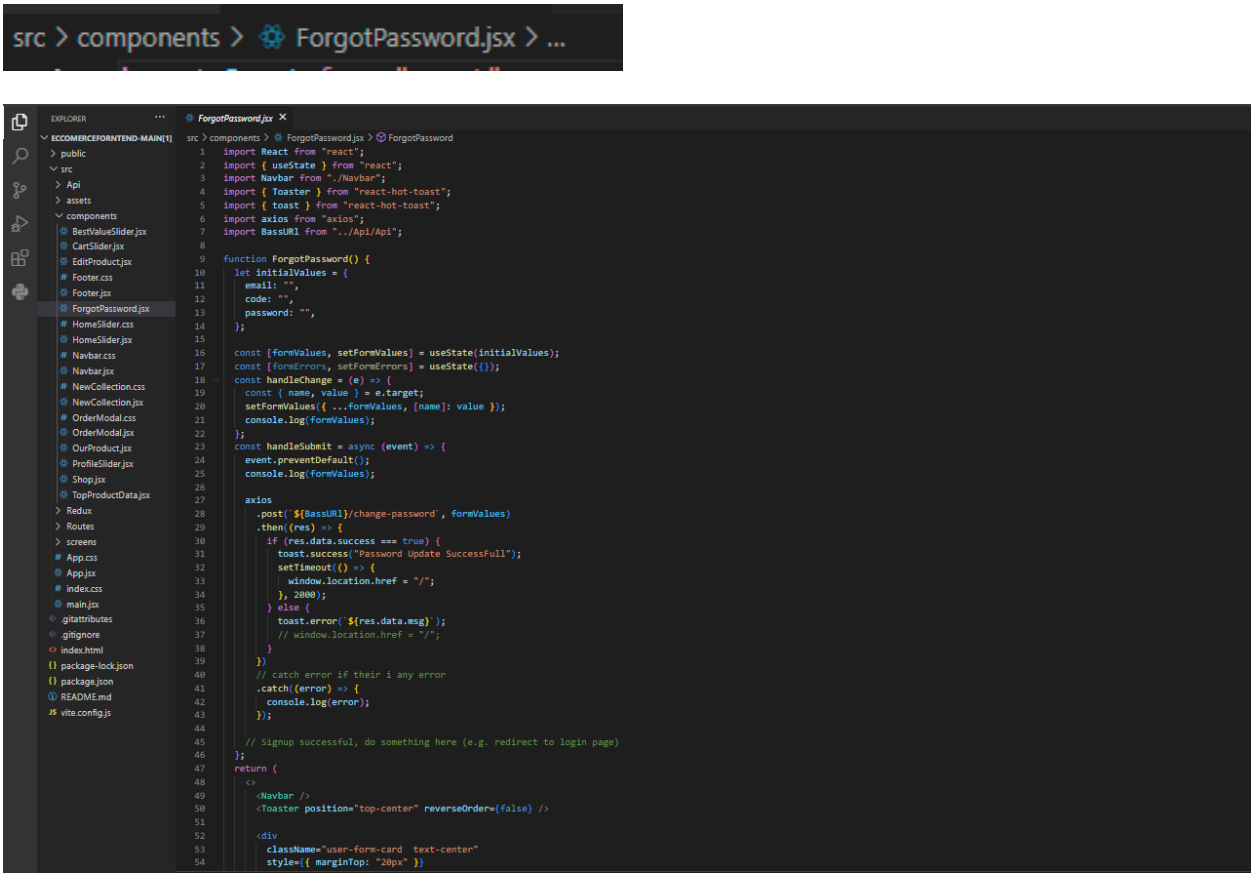


Figure 12 - Sample Code Screenshot of Component Forgot Password

The Forgot Password component is a React component that renders a form for users to reset their passwords if they have forgotten them. It imports necessary dependencies, such as React, useState, Navbar, Toaster, toast, axios, and BassURl from “. /Api/Api". It initializes a state object formValues with keys for email, code, and password, and an empty object formErrors for storing form validation errors. The handleChange function updates the form values when the user types in the input fields, while the handleSubmit function sends a POST request to the $BassURl/change-password endpoint with the form values to update the user's password. The component also renders a Navbar and Toaster for displaying toast messages, and a form with fields for the user to enter their email, OTP code, and new password. When the form is submitted, the handleSubmit function is called.

3.2.1.4 Component ProfileSlider

src > components > ProfileSlider.jsx > ...

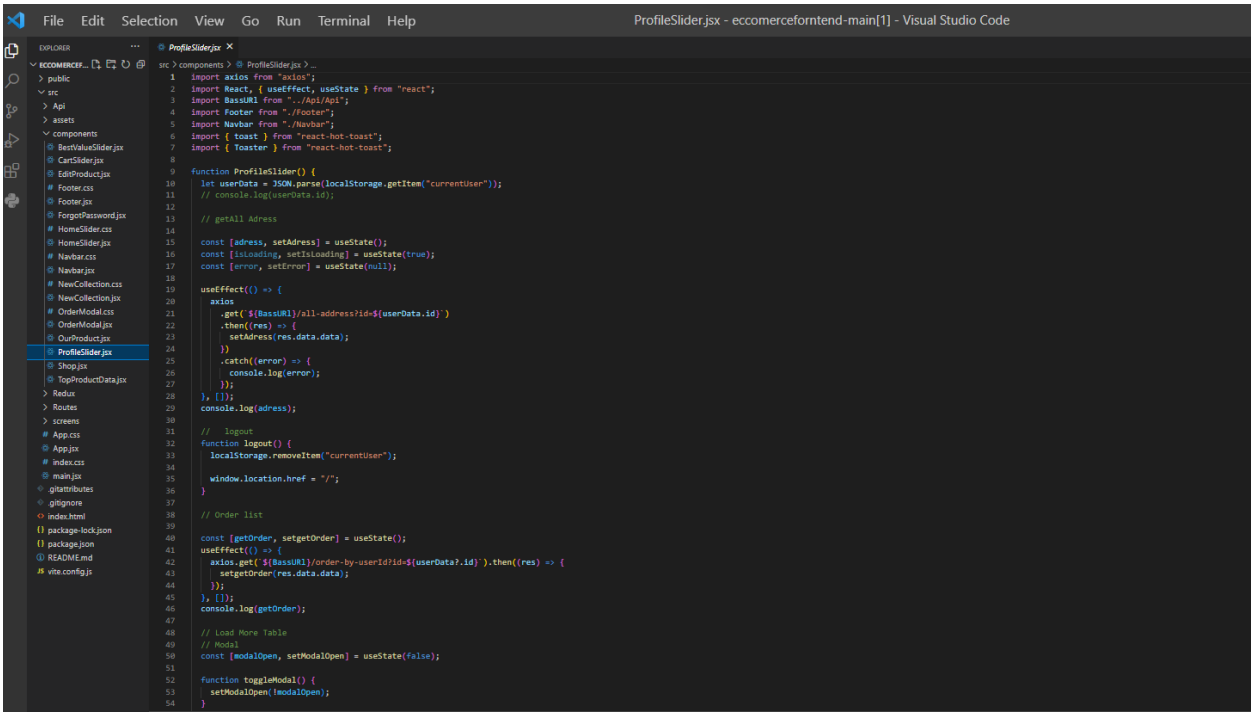


Figure 13 - Sample Code Screenshot of Component Profile Slider

The ProfileSlider is a React component that displays a user's profile page, including their personal details, order history, and options to modify their login credentials or log out of the system. The code imports various dependencies, including React, useState, useEffect, axios, BassURl, Navbar, Footer, toast, and Toaster, which are deemed necessary for the program's functionality. The aforementioned process involves the retrieval of user data from local storage and the initialization of various state variables, including but not limited to address, isLoading, error, getOrder, modalOpen, formValues, and formErrors. The addresses of the user are retrieved through an API call utilising the axios library within a useEffect hook, subsequently updating the state variable for the address. The code snippet delineates a logout function that eliminates the "currentUser" element from the local storage and navigates the user to the homepage. The user retrieves their orders by making an API request using the axios library within a given context.

3.2.1.5 Component HomeSlider

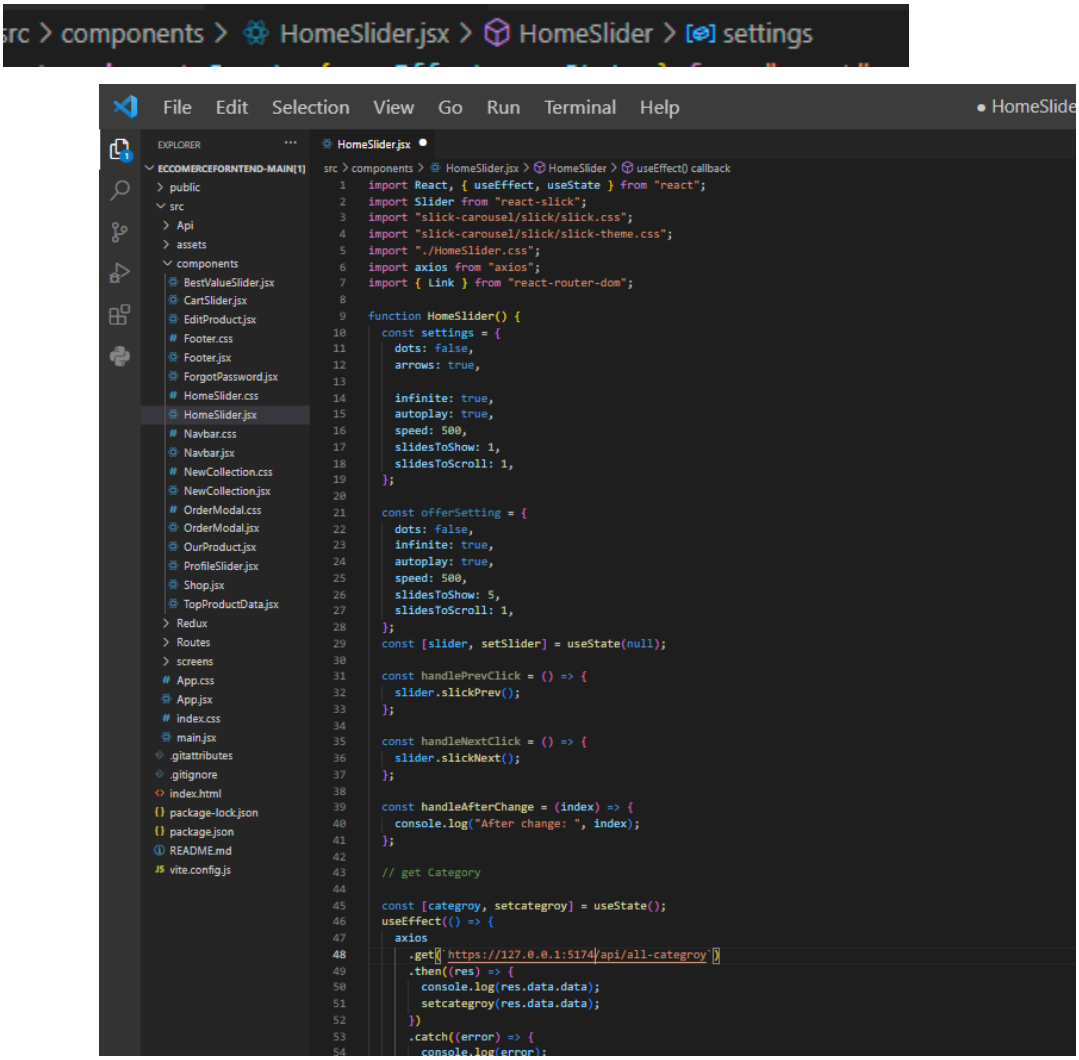


Figure 14 - Sample Code Screenshot of Component Home Slider

The HomeSlider is a software component developed using the React framework. Its primary function is to display a carousel slider and a category slider on a webpage. The implementation utilises the useState and useEffect hooks for state management and retrieval of data from the API. The code generates settings objects for the carousel slider and the category slider, defines handler functions to manage the sliders' navigation and behavior, retrieves category data from the API, produces the JSX structure that displays the carousel slider and the category slider, specifies two custom arrow components, and exports the component for utilization in other sections of the programmed.

3.2.1.6 Component Navbar

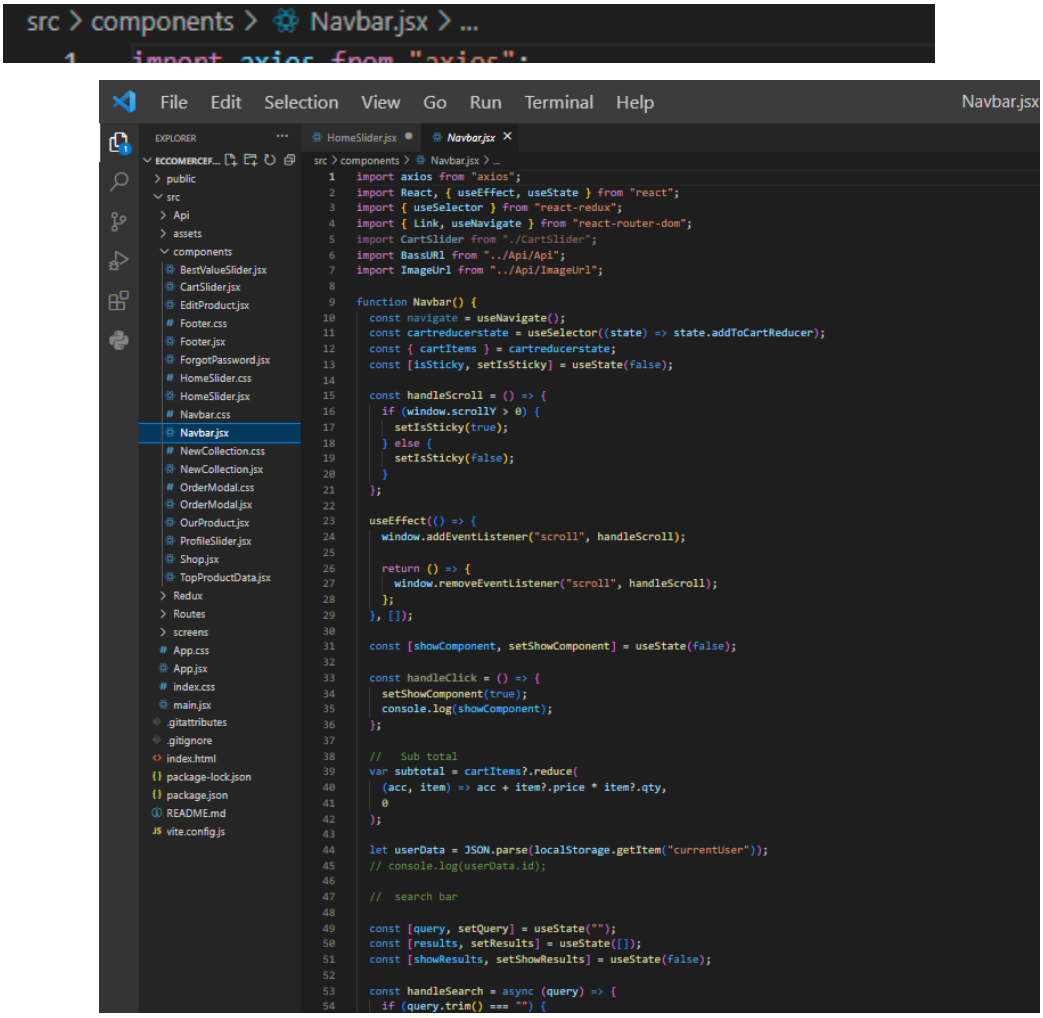


Figure 15 - Sample Code Screenshot of Component Navbar

The Navbar component is a functional component in the React framework that generates a navigation bar with responsive design capabilities for a web-based application. The code imports requisite dependencies, initialises local state variables, creates a scroll event listener, includes a search bar, dynamically displays navigation elements based on the user's authentication status, presents the aggregate count of items in the shopping cart, and generates the navigation bar structure with diverse links, images, and icons. The web application's primary navigation element is also equipped with functionalities such as user authentication, cart management, and search.

3.2.1.7 Component Footer

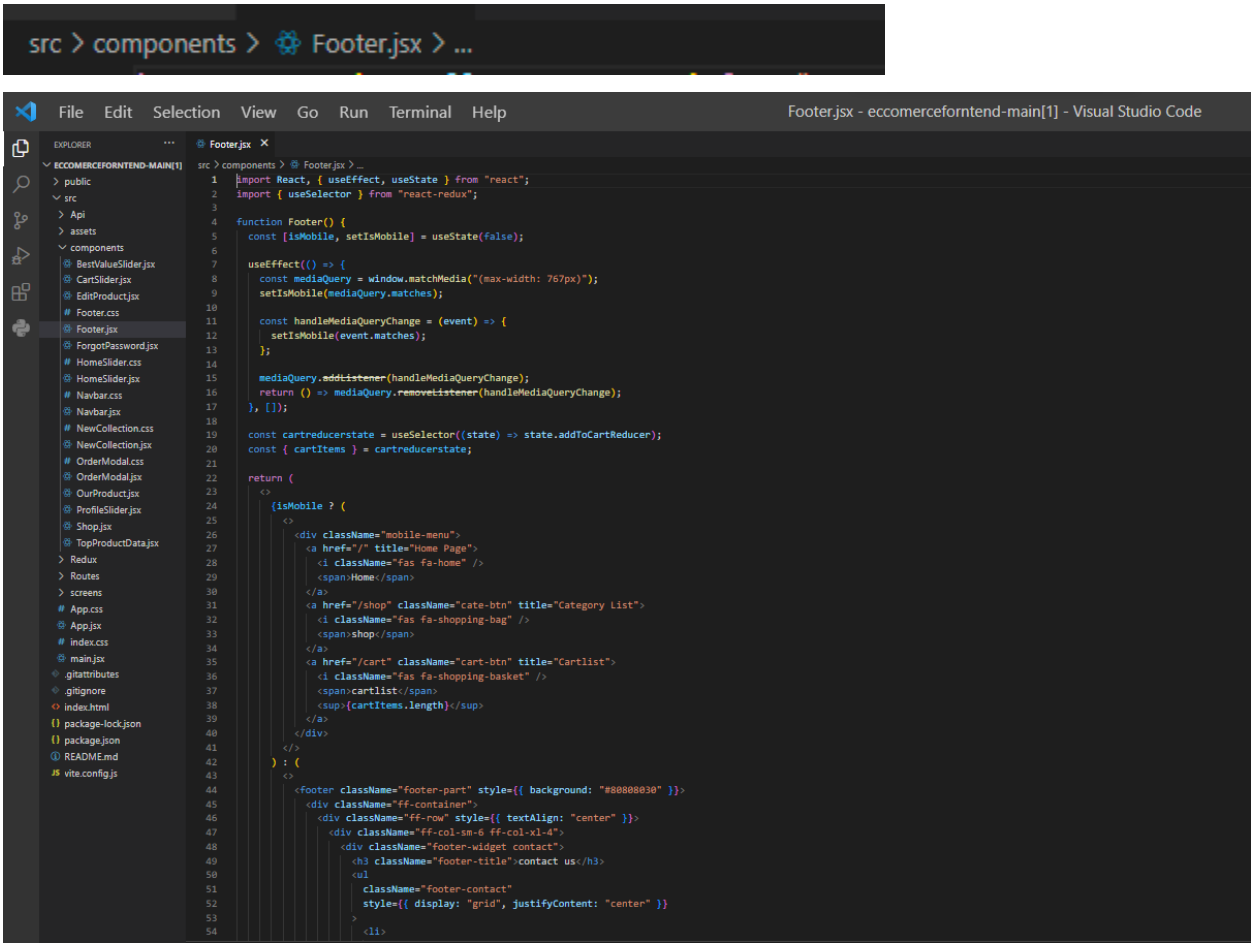


Figure 16 - Sample Code Screenshot of Component Footer

A responsive footer for a website that adjusts to various screen sizes, notably mobile and desktop devices, is rendered by the footer component, a React component. It defines a state named isMobile to check whether the device is a mobile device with a screen width of 767 pixels or less and imports the relevant dependencies from "react-redux." In order to access the addToCartReducer state from the Redux store and extract the cartItems array, it uses useEffect to set up a media query listener and useSelector. When the value of isMobile is true, a mobile menu with links to the home page, category list, and cart list is displayed. When the value of isMobile is false, a more detailed desktop footer with contact details, a logo, and quick connections to various website areas is displayed.

3.2.1.8 Component shop

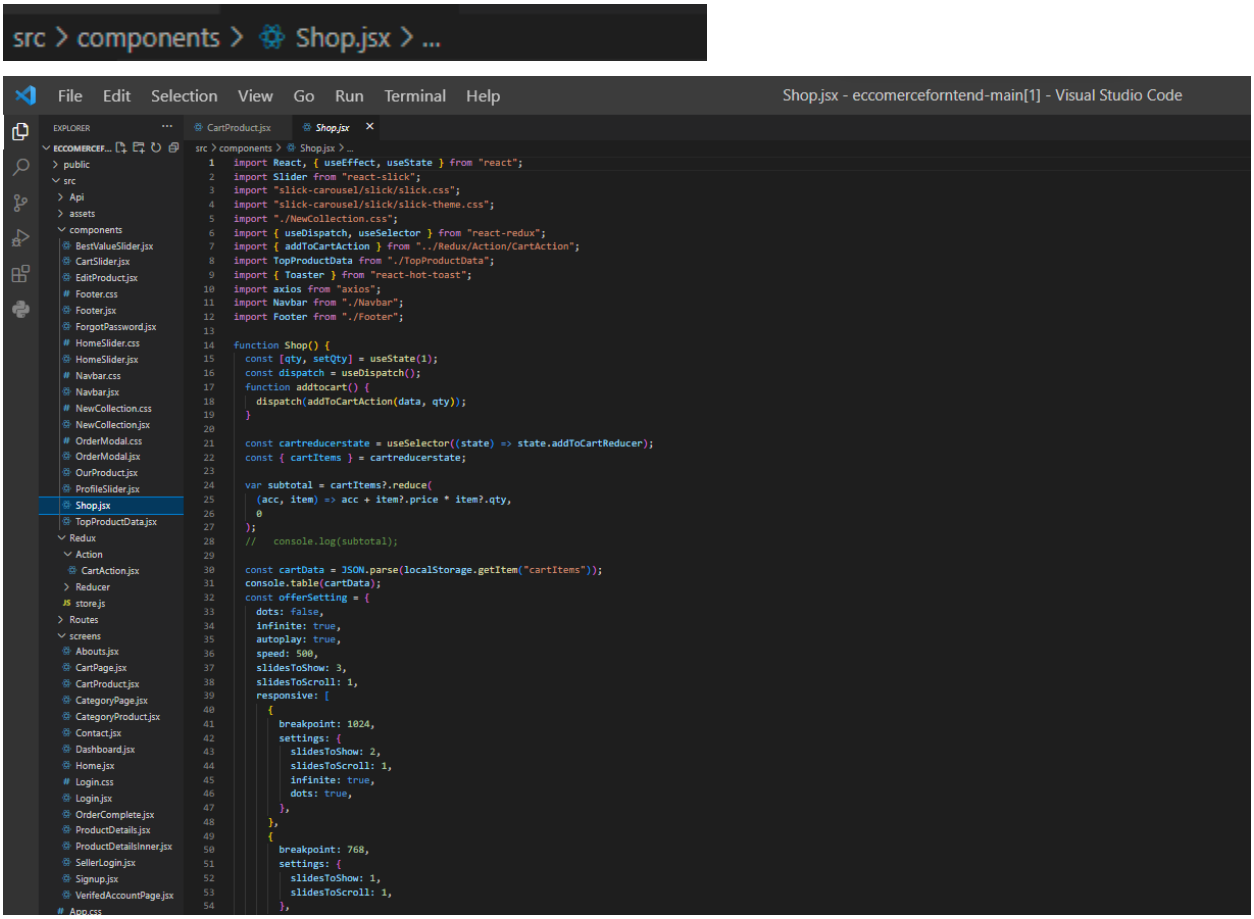


Figure 17 - Sample Code Screenshot of Component Shop

The Shop component is a software module implemented in React programming language, which is responsible for rendering a webpage that displays a collection of products. These products are obtained from an external data source through an API. The code imports essential dependencies, initialises state variables and hooks, computes the subtotal of the items in the cart, retrieves cart data from local storage, defines an offerSetting for configuring the Slider component, creates a static array named topProduct that contains product information, fetches all products using an API call, renders the component, and exports it. The functionality of the system allows for the presentation of a storefront, wherein individuals can peruse and examine a catalogue of merchandise obtained through an Application Programming Interface (API).

3.2.1.9 Component CartSlider

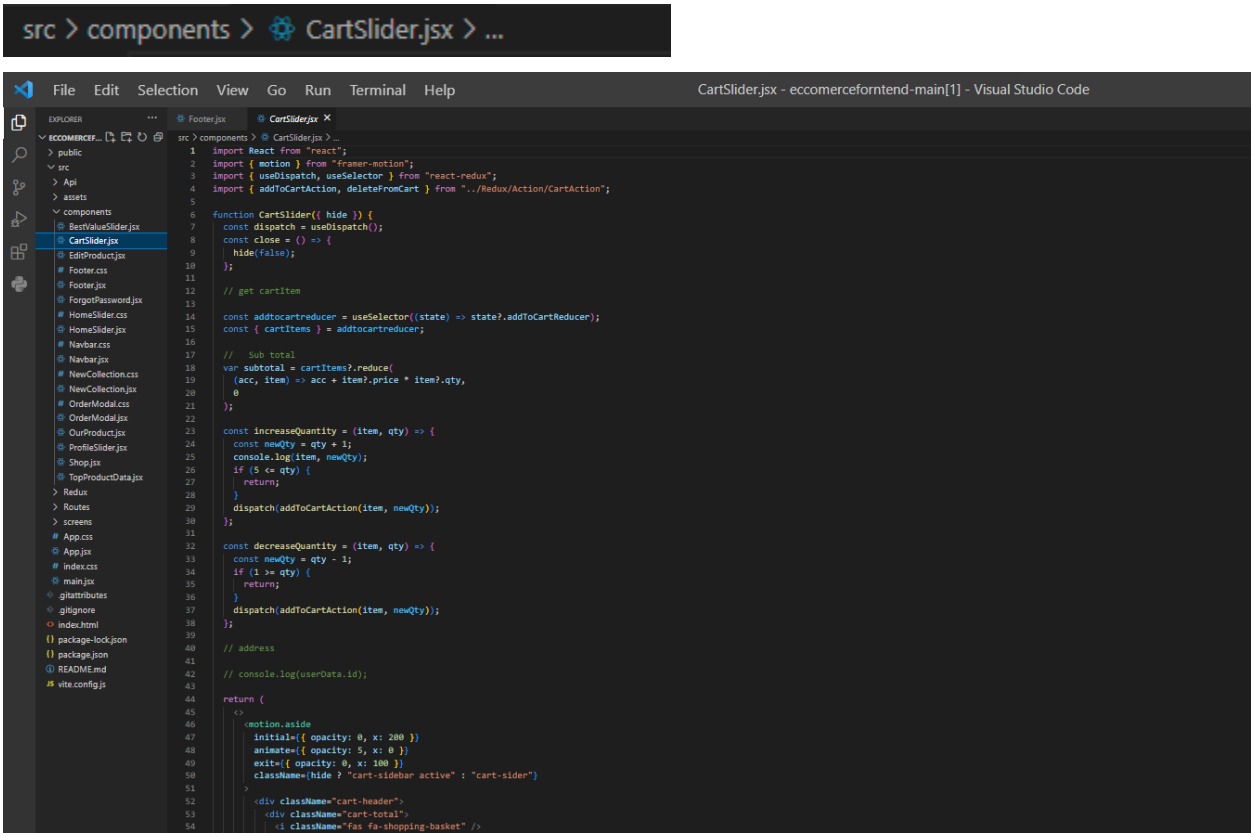


Figure 18 - Sample Code Screenshot of Component Cart Slider

CartSlider is a React component that interacts with a Redux store to manage the state of the cart items. It imports dependencies, uses useDispatch and useSelector, calculates the subtotal of the cart, defines increaseQuantity and decreaseQuantity functions, and renders a styled sidebar with cart items, quantity controls, individual prices, and a checkout button.

3.2.1.10 Component BestValueSlider

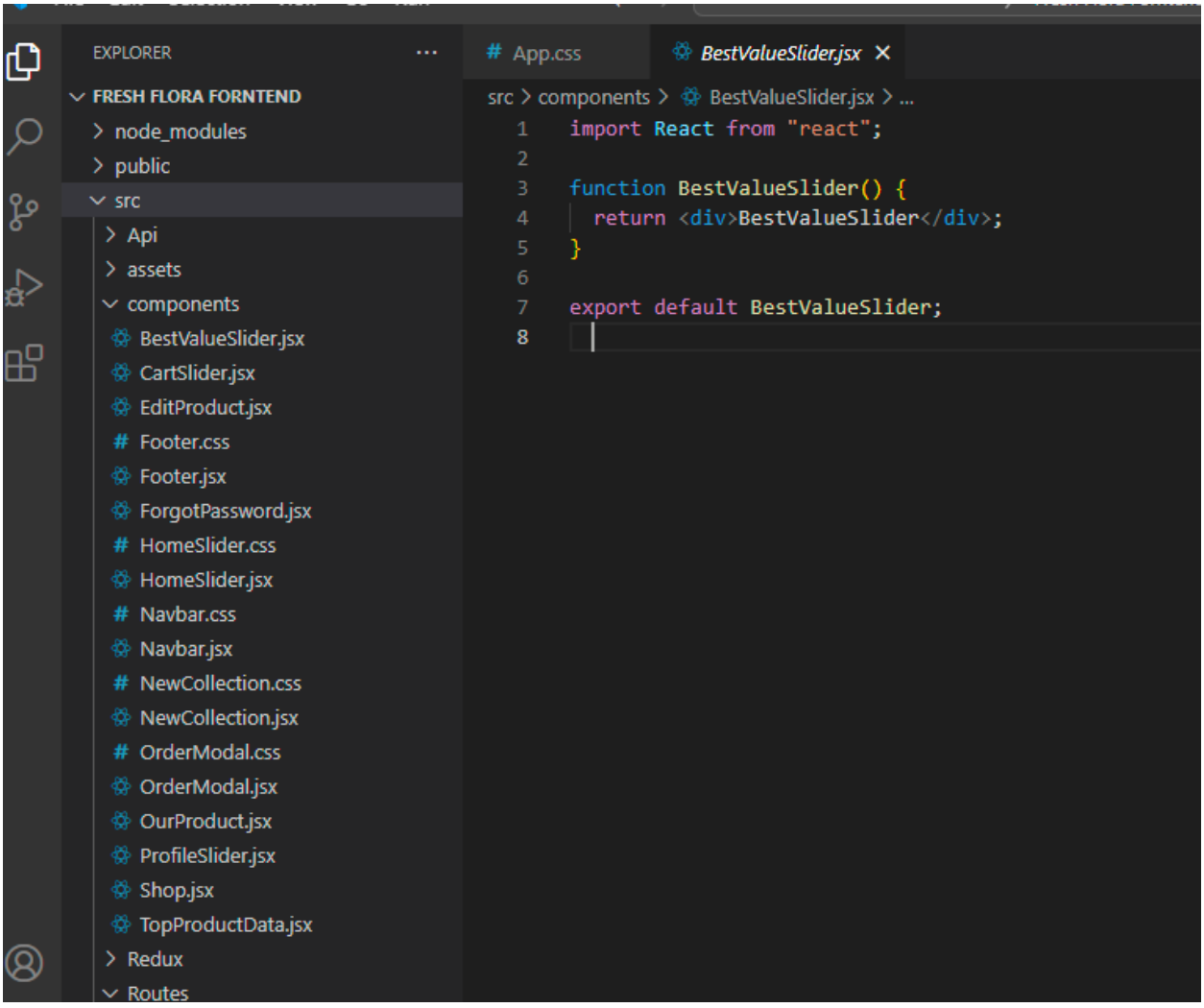


Figure 19 - Sample Code Screenshot of Component Best Slider

Using the "function" keyword, the creation of a React functional component known as "BestValueSlider" is made in the code. The component returns a simple JSX element BestValueSlider . In the final step, the component is exported by utilising the statement known as export default. Importing the component into other sections of the programme with the command import BestValueSlider from '. /BestValueSlider' makes it possible for those sections to make use of the component. In general, this code generates a reusable component that displays the text "BestValueSlider" on a basic div element.

3.2.1.11 Component EditProduct

src > components > EditProduct.jsx > ...

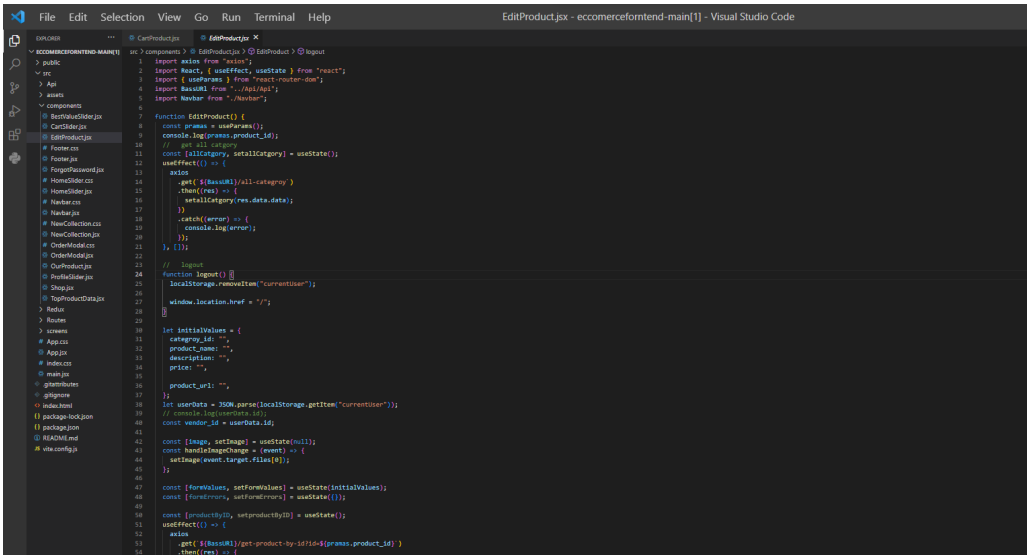


Figure 20 - Sample Code Screenshot of Component Edit Product

To retrieve the product ID from the URL, the component makes use of the useParams hook that is included in the react-router-dom package. After that, it utilises the axios library to send two API requests in order to acquire the specifics of the product as well as all of the categories that are available. After the API queries have been successfully completed, the component will use the useState hook to establish the starting values for the form. The handleChange method is then used to bring about the updating of the form values whenever the user types into the input fields. [6][3]

The handleSubmit method is called whenever the user sends in the form that they have filled out. This function generates a FormData object that contains the recently modified form values as well as the image file that was selected. After that, the axios. Post method is used to transmit the data from the form to the server so that it can be processed. The end result of this code is the creation of a reusable component that gives users the ability to change the product details and submit a new image of the product. In addition to being connected with React Router and the Bootstrap CSS framework, the component was developed with an API backend in mind from the beginning. [6][3]

3.2.1.12 Component NewCollection

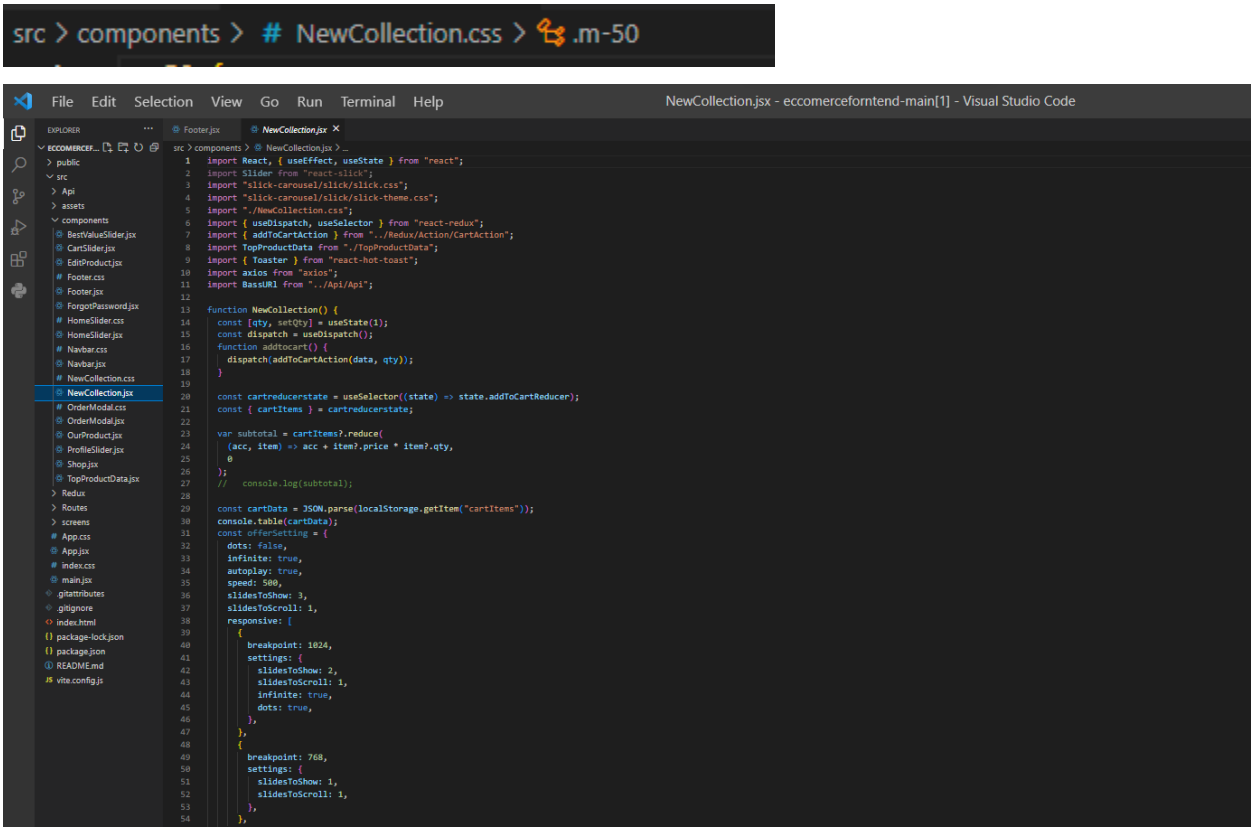


Figure 21 - Sample Code Screenshot of Component New Collection

The NewCollection component is a software module implemented in React, which is responsible for rendering a designated section of a web page. This section showcases a curated selection of high-quality products, accompanied by two promotional banners that advertise fresh vegetables and healthy fruits. The code imports essential dependencies such as 'react-slick', addToCartAction, TopProductData, Toaster, and axios. It also initialises state variables and defines a function named 'addtocart()'. The code retrieves cart items from the Redux store and sets up a responsive carousel configuration for the top products section. Additionally, the code fetches all products using an API call with axios in a useEffect hook and sets the product state variable. This component comprises a Toaster component that is utilized to exhibit notifications, a section that showcases the top products utilising the TopProductData component, a 'show more' button that redirects to the '/shop' page, and two promotional banners that advertise fresh vegetables and healthy fruits. Additionally, the component is exportable. [6][9][3]

3.2.1.13 Component OrderModal

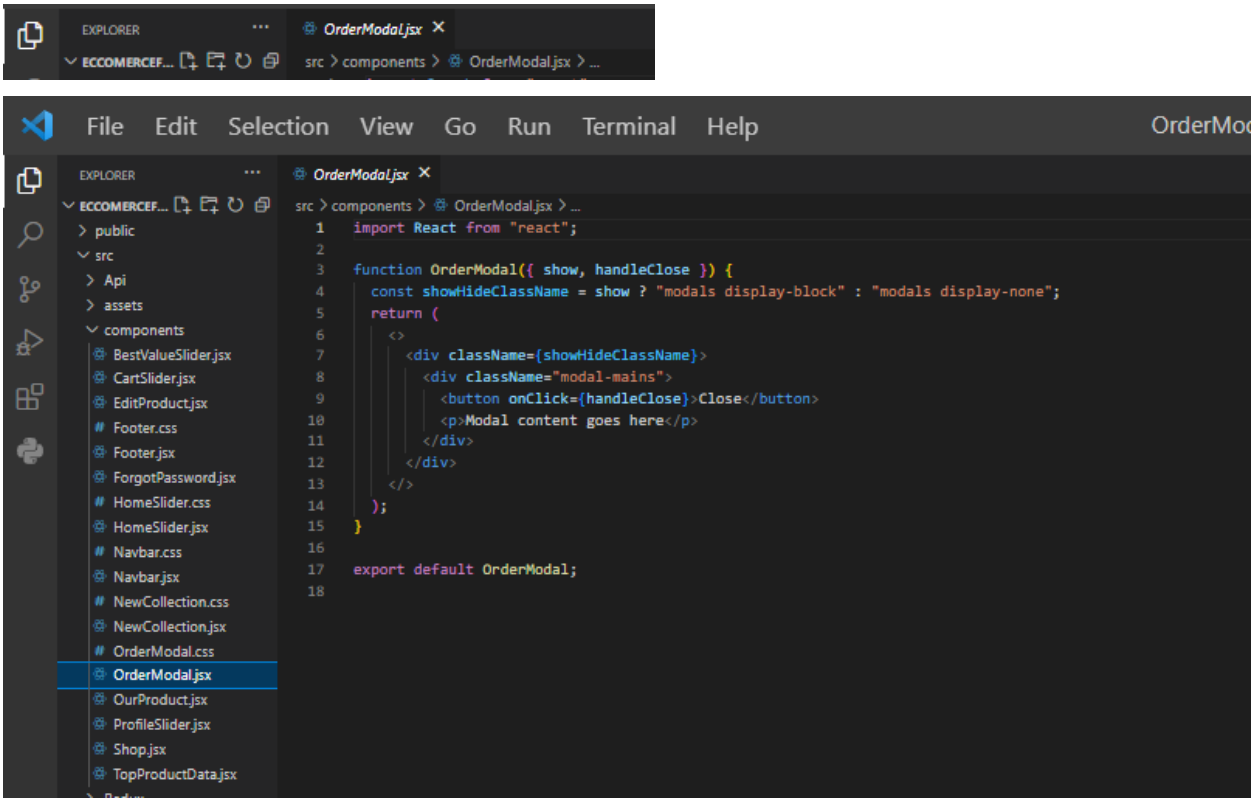


Figure 22 - Sample Code Screenshot of Component Order Modal

The OrderModal component is a functional component in the React framework that generates a basic modal interface featuring a button for closing the modal and a placeholder text for the modal content. The function accepts two parameters, namely show and handleClose, and assigns the showHideClassName variable a value based on the value of the show parameter. The modal component is instantiated through a div element, which is assigned a class name of showHideClassName. Additionally, the component includes a button element that triggers a onClick event, which in turn invokes the handleClose function that has been passed down as a prop. Lastly, the component contains a paragraph element that serves as a placeholder for the modal content. The aforementioned component has the capability to exhibit a modal containing content that is dependent on the show and handleClose props that are supplied to it. [6][3]

3.2.1.14 Component TopProduct

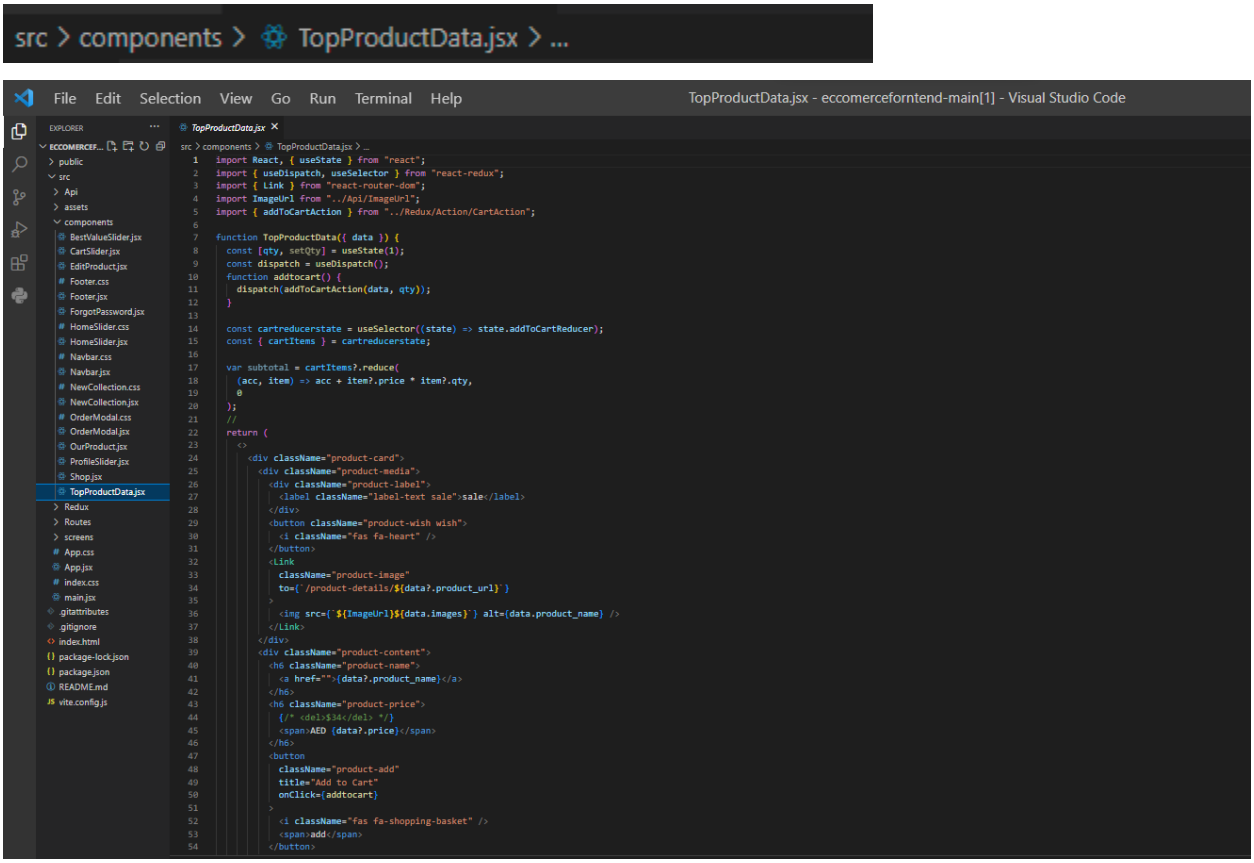


Figure 23 - Sample Code Screenshot of Component Top Product Data

The React functional component, TopProductData, is designed to receive a data prop and display a product card containing pertinent information about a particular product. The code imports essential dependencies, initialises state variables and hooks, computes the subtotal of the items in the cart, generates a product card that includes a product label, a wishlist button, an image of the product, the product name, the product price, and a "Add to Cart" button, and finally exports the component. The product card is commonly utilized to showcase a solitary item within a grid or list arrangement, as an element of a more extensive product inventory or storefront page. [6][7][3] Upon clicking the "Add to Cart" button, the designated quantity of the product is appended to the cart.

3.2.1.15 Cart

Cart Function is the main important function as they are divided in to 2 major part and has 2 buttons in the front end.[6][7][8][9][3]

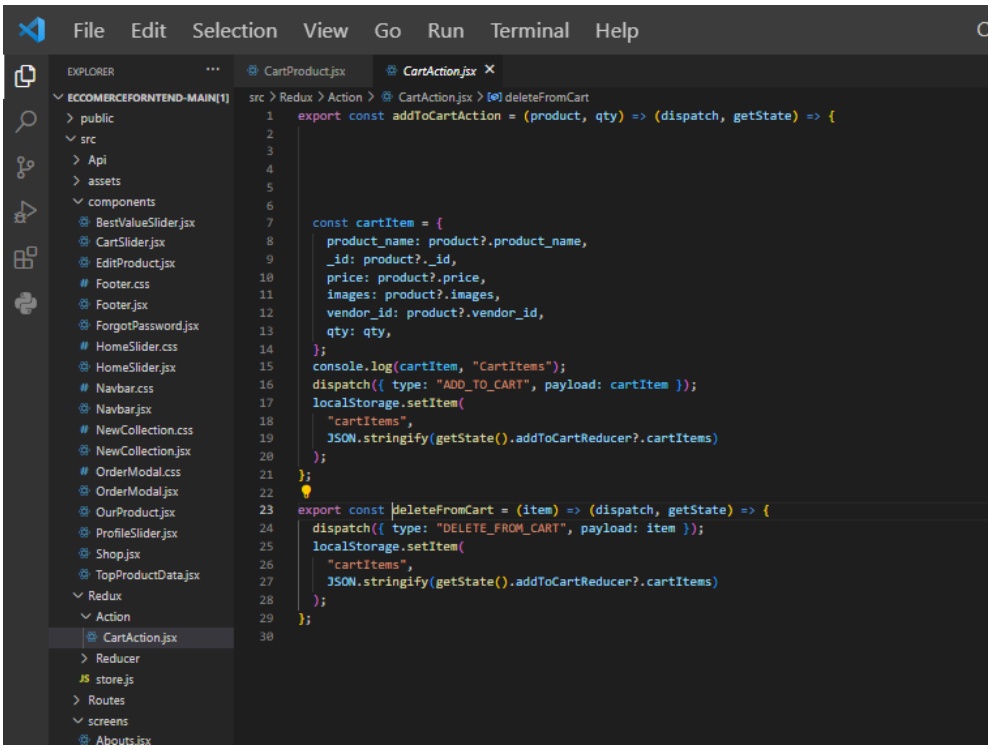


Figure 24 - Sample Code Screenshot of Component Cart Action

3.2.1.16 Add to cart

The add to cart is a button which implements a method called (ADD to cart) which add product from the shop page in the frontend. It also validates the order Quantity placed . the function Uses an if loop which checks if product exists in the cart . [6][7][8][9][3] it increments a value to add another no. after it is added we get a message saying Added to cart.

3.2.1.17 Delete from cart

The Delete from cart is a button on the cart page. The button has a method which delete a product from the cart in the user cart page in the frontend. It also validates the order Quantity placed . the function Uses an if loop which checks if product exists in the cart . [6][7][8][9][3] It decrements a value to remove it.

3.2.3 Index. HTML

This is the only page which doesn't used the JSX whereas it creates basic Website structure and we use Function calling in the body to call it rather than implementing the full code[11].

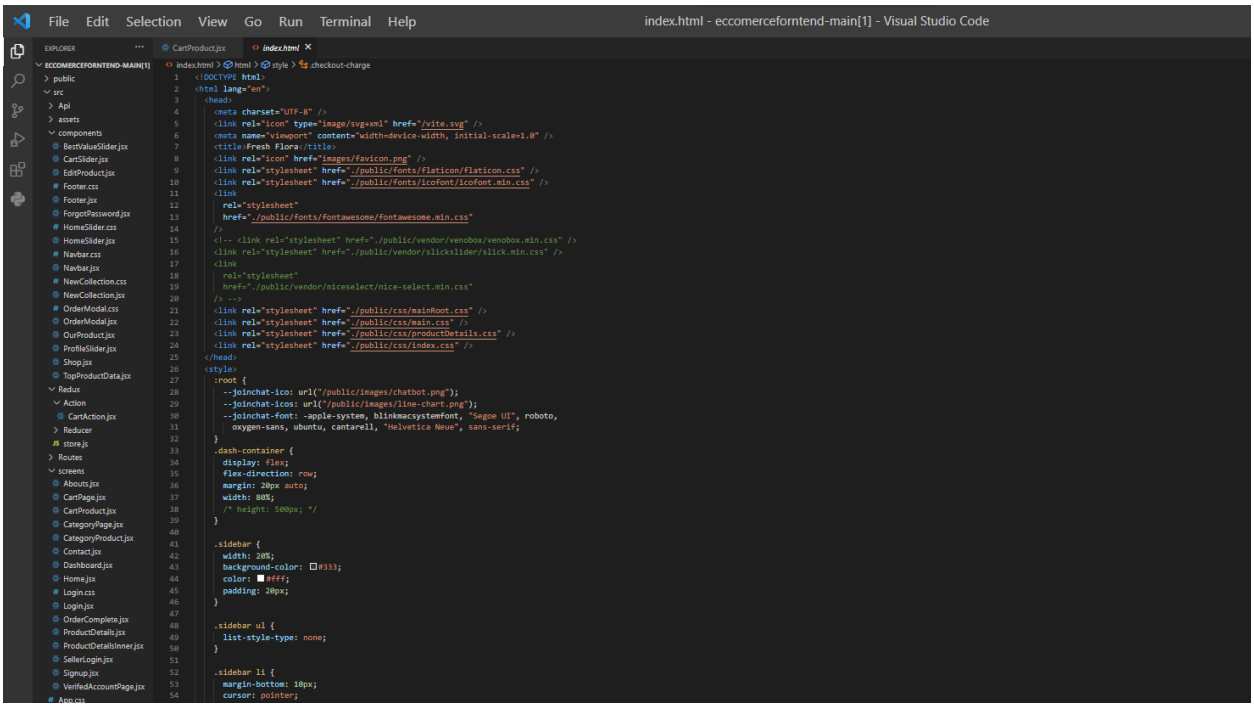


Figure 25 - Sample Code Screenshot of Index HTML

3.2.2 CSS (Cascading Style Sheet)

A Style sheet Language is a computer language that expresses the presentation of structured document [11].

```
<!DOCTYPE html>
<html>
    <head>
        <title>Arduino SD Card Web Page</title>
        <style type="text/css">
            h1 {
                font-family: courier, courier-new, serif;
                font-size: 20pt;
                color: blue;
                border-bottom: 2px solid blue;
            }
            p {
                font-family: arial, verdana, sans-serif;
                font-size: 12pt;
                color: #6B6BD7;
            }
            .red_txt {
                color: red;
            }
        </style>
    </head>
    <body>
        <h1>Arduino SD Card Page with CSS</h1>
        <p>Welcome to the Arduino web page with CSS styling.</p>
        <p class="red_txt">This text is red.</p>
        <p>This paragraph has one word that uses <span class="red_txt">red</span> text.</p>
    </body>
</html>
```

Figure 26 - Sample Code Screenshot of CSS

3.2.3 API

An architectural style known as Representational State Transfer (REST) stipulates rules that must be followed when developing web services. REST API is a protocol for gaining unprocessed, yet versatile, access to web services[6].

In comparison to the more powerful Simple Object Access Protocol (SOAP), REST technology is widely favored because of its lower bandwidth requirements, its simplicity, and its adaptability, all of which make it well-suited for use over the internet. It communicates with a web service in order to receive or send data. The REST API exclusively makes use of HTTP requests for all of its interactions[6].

The HTTP GET, POST, PUT, and DELETE verbs are used to send requests from a client to a server. After that, the server returns a resource, which could be anything from HTML to XML to an image to JSON. However, nowadays JSON is the most often used format for Web Services[6].

Figure 27 - Working of REST API

Throughout the whole web Development all of the classes (React files) have used API for the Connection to different pages.

3.3 Website Backend server side (NoSQL)

Different data models allow for a wide range of database formats, including the non-tabular NoSQL (not exclusively SQL) databases. Large, small, structured, semi-structured, and polymorphic data types.[18]

It is impossible to store unstructured data in a relational database management system (RDBMS) since it does not conform to any predefined data model or schema. Unstructured data accounts for 80-90% of all data created and gathered by businesses, and its quantities are rising at a rate several times faster than that of structured databases.[18]

Here we choose NoSQL for below reasons and to learn latest technologies

- Reduce efforts for development from developer hence cost optimization.
- Storage Cost optimization
- Unstructured data collected through chatbot and other channels to be queried.
- Data input by sellers being more in numbers can't predict in sizing of servers, amount of data to store for this application and query increased.
- To adopt the changes required quickly & rapidly to application till database level NoSQL gives flexibility in creating new queries.
- NoSQL have capability to cloud computing to distribute data to multiple servers & regions.

3.3.1 Databases

A database is a container for collections. Each database gets its own set of files on the host file system. A single MongoDB server typically has multiple databases. [18]

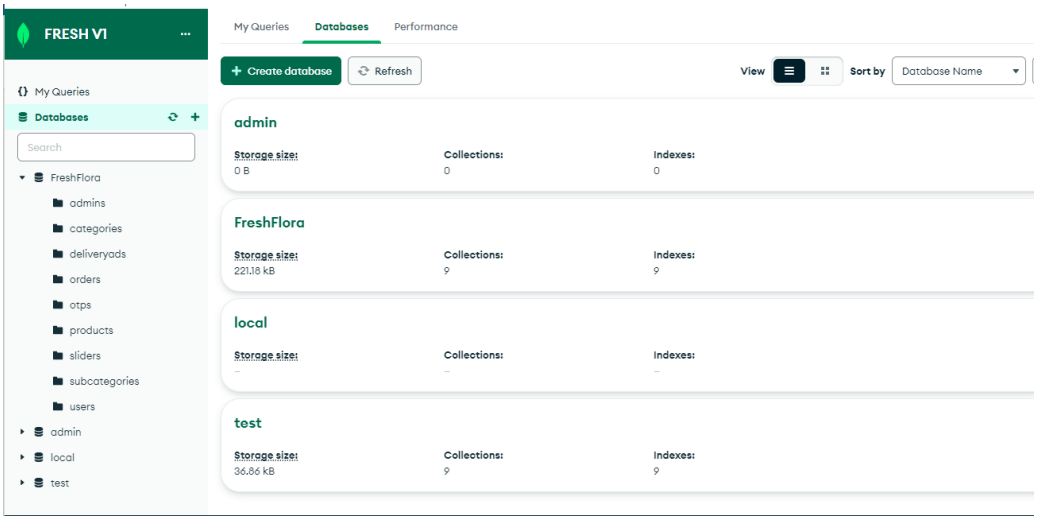


Figure 28 - Screenshot of Database

3.3.2 Collections

In MongoDB, data is organized into collections. A collection's documents may have varying fields. In a database management system, collections serve the same purpose as tables. One database can house many collections. Documents/Aggregations/Schema/Explain pan/Indexes/Validations will be part of each collection. [18]

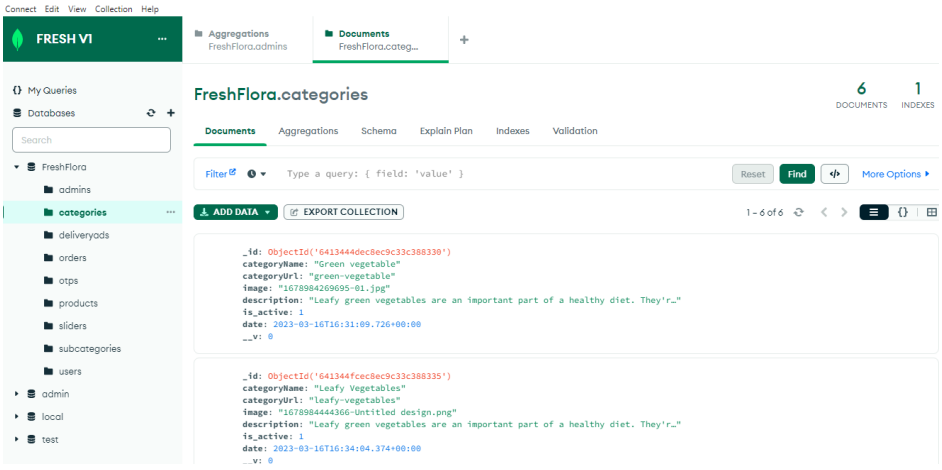


Figure 29 - Screenshot of Database Collection

3.3.3 Documents

The smallest unit of data that can be created, viewed, deleted, and altered is the document. By design, documents in a NoSQL database are not required to have a common schema. Each document in a collection doesn't have to have the same fields, and each field's data type might be different.[18] Update the documents in a collection to reflect the changed structure if you need to add new fields, delete fields, or convert field values to a different type. Even if there is significant diversity across documents in the collection, each document can nonetheless match the data fields of the represented object.[18]

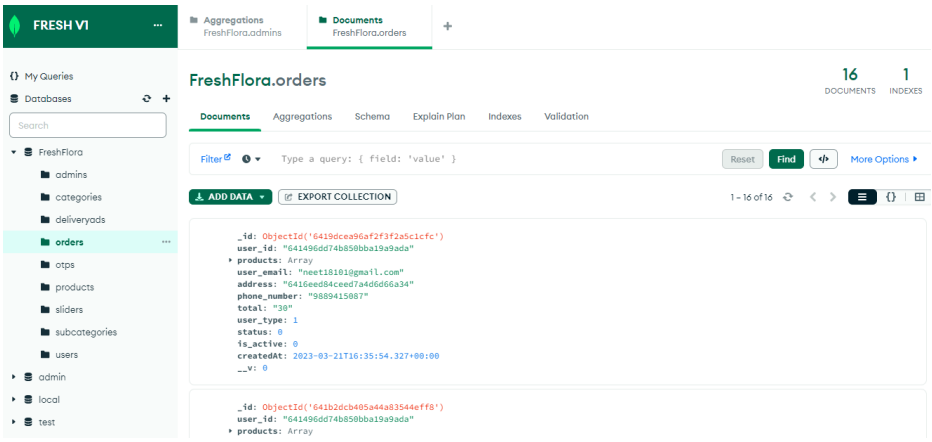


Figure 30 - Screenshot of Database Documents

3.3.4 Aggregations

The Aggregation Pipeline Builder in MongoDB Compass lets you create aggregation to process documents from a collection or view and return computed results.[18][12] For example, you can use aggregation pipelines to:

- Group values from multiple documents together.
- Perform operations on the grouped data to return a single result.
- Analyze data changes over time.

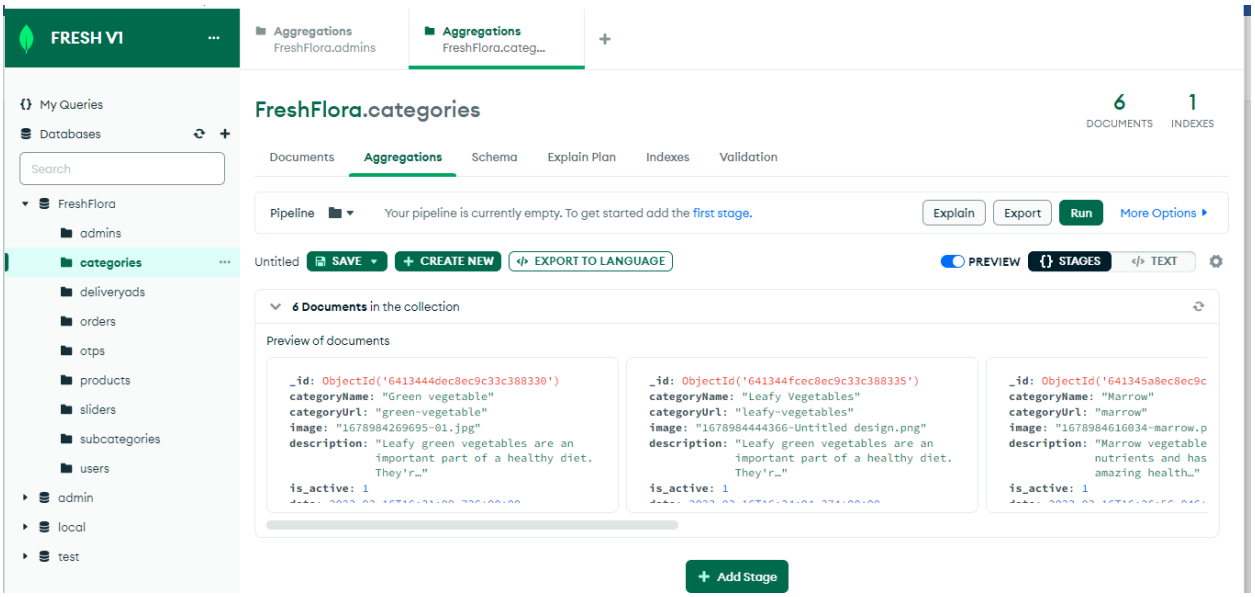


Figure 31 - Screenshot of database Aggregation

3.3.5 Schema

Schema in NoSQL is a JSON object that specifies its format and content. Atlas App Services' BSON schemas, an extension of the JSON Schema standard, may be used to design your app's data model and validate documents upon creation, modification, or deletion.[18][16]

In contrast to actual values, schemas only describe categories of information. The number of predefined schemas in App Services is rather large. Primitive types like texts and integers, as well as structural kinds like objects and arrays, can be combined to represent specialized object types in schemas.[18][16]

3.3.7 Indexes

MongoDB queries can run more quickly thanks to indexes. Without indexes, MongoDB must do a collection scan, which involves reading each and every document in a collection in order to find the ones that are relevant to the query. MongoDB can reduce the number of documents it needs to examine by using an index if one has been created specifically for the query. Indexes are specialized data structures that keep a condensed, easily-navigated subset of the entire collection's information. The index keeps track of the values of a selected field or collection of fields in ascending order according to the fields' respective values.[18][14] The index's entry order facilitates fast equality comparisons and range-based queries. Furthermore, MongoDB may provide sorted results by consulting the index's order. When a collection is created in MongoDB, a unique index is generated for the _id field. [18][14] The _id index ensures that no two documents may be inserted with the same _id value from the same client. The _id field's index is required and cannot be removed.

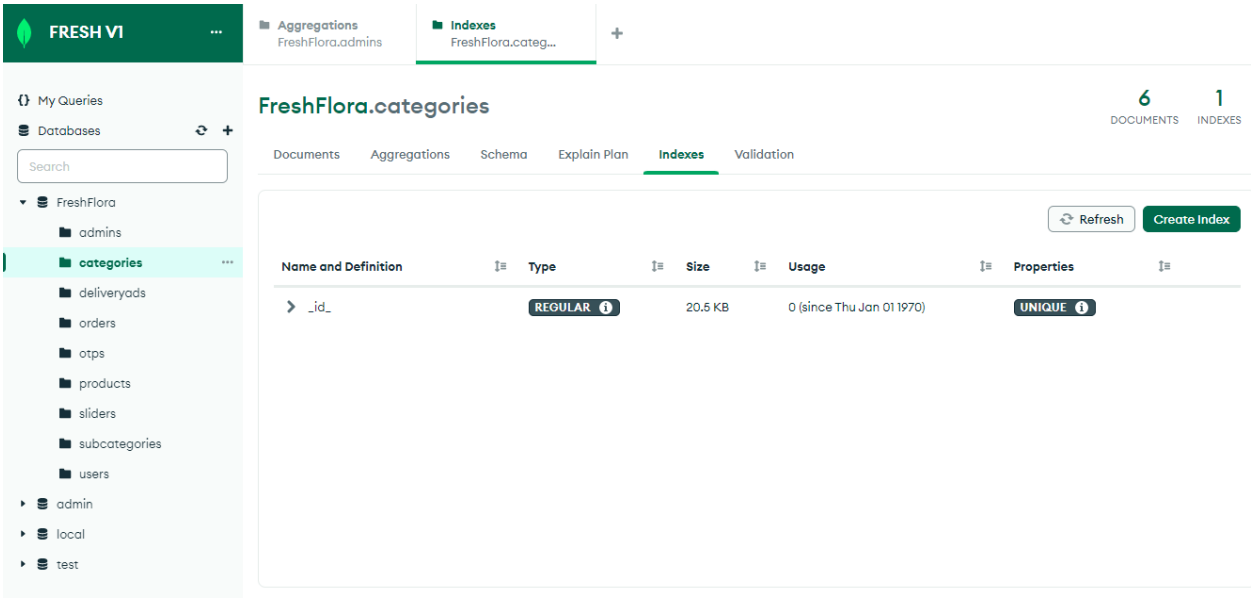


Figure 32 - Screenshot of Database Index

4. CONCLUSION

4.1 Summary

The e-Commerce website (Fresh flora) is a leading producer of fresh products aiming for the health-centric food needs of the future. The main aim of the project is to serve where consumers can directly have contact with sellers (Farmers) who sell their products and also provide a feature to directly purchase fruits & vegetables from the seller (farmers). The main aim of the Project is to provide more information & nutrition facts about Fruits & vegetables in line with a healthy perspective and also provide the right information about the product of what they are looking for should be accurate & genuine, which can't be manipulated by wrong sources, Farmers to get value to their efforts through their Product. Virtual assistant AI Text based chatbot will be used to support consumers with instant information about Fruits & vegetables in large scale where its produced more and types of products.

4.2 Evaluation

As mentioned, the main purpose of the project is to develop e-Commerce website with Virtual assistant chatbot using latest technologies & optimize the solutions depending on functional requirements . The best way to critically analyses the implementation of the website & Chabot is to first look at the completeness of features. To start, the website access 2 main roles registered user called Consumer & seller (Farmer) & unregistered user who just view high-level pass-through site .Currently, Consumer can buy product (order product) seller (Farmer) can add product to gallery upon approval available to consumer to buy. While consumer selecting product same will be reflected to seller (Farmer) dashboard thus direct order to seller (Farmers) and cash on delivery direct to seller (Farmer) .however, it does not contain functionality for online payment intension to root direct purchase from seller (Farmer) . If seller (Farmer) unable to place their products they can directly deal with admin to add their product on behalf Back-end system admin can manage entire process of the application. End to end full stack website development considered.

AI Text Based chatbot : Act as standalone through this AI Text Based chatbot user can get more information on different products , nutrition facts about the product and different types of product names , Availability of the product .At this stage its focused only INDIA based locations. How ever its planned to add more geographic locations including maps etc.

4.3 Future Work

The Project is a proof of concept, demonstrating various user roles & virtual Assistant. In future, more of this project enhanced with especially adding more geolocations and the logistic management, and in build chatbot would be top priority for further implementation. Hosting the Website into the Public Domain (www.Freshflora.in) and making a mobile application.

4.4 Reflection

The project's progress was well managed. If I had the chance to do the project over, I would lay out a more detailed plan and specify more concrete dates for each stage. Due to inexperience and a lack of awareness of the consequences of making changes to components mid-project, basic planning was neglected. I feel like I've grown a lot as a result of these misteps.

The transition from the temporary to the permanent campus was very difficult for me. Given the extensive amount of time I had previously invested in the project before to this transfer, I was unclear of how to contact my supervisor for help. The initiative got off to a good start despite these obstacles. I was able to successfully record the project plan because I had a firm grasp on the big picture. I was going to be the only source of information for development and functional needs, had already established the project's scope and objectives, and had received necessary approvals in a timely manner. I approached lecturers at RAK Campus about helping me test the prototype, and they graciously consented. I appreciate Ms. Preetha's and Dr. Mario support on the guidance, motivation.

I was able to finish the front-end UI interface on time, however working alone and juggling Project work proved challenging. Moving to the large campus from a smaller one made me feel even more alone. Despite these obstacles, I was able to finish the website primary functionality and make it available for testing. My loved ones acted as test subjects, and their honest critiques helped me pinpoint and resolve problems. However, with Dr. Mario's encouragement, I've begun developing a Chatbot (AI Text Based Chatbot) that can function independently. The results of this effort have given a good output and still I would like to improvise and add more functionalities.

5. APPENDIXES

5.1 CHATBOT TEST CASES EXECUTION SCREEN AND SCREENSHOTS

Test Scenario	TAG	Pattern	Responses
General Greetings			
1	greeting	Hi there, how are you, Is anyone there? Hey, Hola, Hello, Good day [29]	Hello, thanks for asking, Good to see you again, hi there, how can I help?[29]
2	goodbye	Bye, see you later, Goodbye, Nice chatting to you, bye, Till next time[29]	See you! Have a nice day, Bye! Come back again soon. [29]
3	thanks	Thanks, Thank you, That's helpful, Awesome, thanks, Thanks for helping me[29]	Happy to help!, Any time!, My pleasure[29]
4	no answer	Na [29]	Sorry, can't understand you, Please give me more info, Not sure I understand [29]
5	options	How you could help me? What you can do? What help you provide? How you can be helpful? What support is offered	I can guide you through website Registration, queries Etc..., Offering support for Account Issues, Product Queries and Info
Product specific			
1	FRESH ONIONS	onions , about onions , facts about onions ,Indian onions , about Indian onions [30]	India is the second-largest onion-growing country in the world. The Indian onions are famous for their pungency and are available round the year. Indian onions have two crop cycles, first harvesting which starts from November to January and the second harvesting from January to May. [30]
1.1	Onion Varieties	what are varieties of onions , onion	Agrifound Dark Red, Agrifound Light Red , NHRDF Red , Agrifound

		varieties found in India , onion types found in India [30]	White , Agrifound Rose , Agrifound Red , Pusa Ratnar, Pusa Red ,Pusa white , Tana F1 , Arad-H ,Suprex , Granex 55 , HA 60 ,Granex 429] [30]
1.2	Onion Areas of Cultivation	where onion cultivated , Indian onion cultivation , best place to cultivate onion , onion producing states [30]	Maharashtra , Karnataka , Madhya Pradesh , Gujarat , Bihar , Andhra Pradesh , Rajasthan , Haryana , Telangana , Maharashtra], [30]
1.3	Onion Major Export Destinations	where is Indian onion exported more , onion export destinations . [30]	Bangladesh , Malaysia , Sri Lanka , United Arab Emirates , Nepal ,Indonesia , U.K , Qatar, Hong Kong, Kuwait] [30]
2	Walnuts	Walnuts , about Walnuts , facts about Walnuts ?,Indian Walnuts , about Indian Walnuts [30]	Walnut (Juglans sp.) is the most important temperate nut fruit in the country. Walnuts in India are found in different size and shapes. The Indian walnuts are categorized into 4 categories viz paper-shelled, thin-shelled, medium-shelled and hard-shelled. Walnuts flourishes at altitudes of 900 to 3000.] [30]
2.1	Walnuts Varieties	what are varieties of Walnuts , Walnuts varieties found in India , Walnuts types found in India [30]	Lake English , Drainovsky , Opex Caulchry, Gobind , Eureka , Placentia , Wilson , Franquetfe,Kashmir Budded ,Chakrata Selections] [30]
2.2	walnuts Areas of Cultivation	where Walnuts cultivated , Indian Walnuts cultivation , best place to cultivate Walnuts , Walnuts producing states [30]	Jammu and Kashmir , Uttaranchal , Himachal Pradesh, Arunachal Pradesh] [30]
2.3	India Facts and Figures Walnuts	Walnuts Facts & Figures , Indian Walnuts fats Results , Indian Walnuts Reports [30]	The country has exported 2,482.59 MT of Walnuts to the world for the worth of Rs. 73.98 crores/ 9.88 USD Millions during the year 2021-22] [30]

2.4	walnut Major Export Destinations	where Indian Walnuts export more , Walnuts export destinations , countries indication Walnuts exports [30]	United Arab EMTs , France, U.K., Djibouti , Germany, New Zealand, Bangladesh] [30]
3	Fresh Vegetables	Fresh Vegetables , about Fresh Vegetables , facts about Fresh Vegetables ?,Indian Fresh Vegetables , about Indian Fresh Vegetables [30]	India grows a large number of vegetables from temperate to humid tropics and from sea level to the snowline. Vegetables are an excellent source of vitamins, particularly niacin, riboflavin, thiamin and vitamins A and C. They also supply minerals such as calcium and iron besides proteins and carbohydrates. Vegetables are known to be the cheapest source of natural beneficial ingredients. Most of the vegetables being short-duration crops fit very well in the intensive cropping system and are capable of giving very high yields with very high economic returns to the growers. Major vegetables grown in India are Potato, Onion, Tomato, Cauliflower, Cabbage, Beans, Egg Plants, Cucumber and Garkin, Frozen Peas, Garlic and Okra] [30]
3.1	Fresh Vegetables Varieties	what are varieties of Fresh Vegetables , Fresh Vegetables varieties found in India , Fresh Vegetables types found in India [30]	Potato ,Kufri Sindhuri, Kufri Chandramukhi, Kufri Badshah, Kufri Bahar, Tomato, Vaishali , Rupali, Rashmi , Rajni Pusa Ruby, Cauliflower ,Pusa Deepali , Early Kunwari, Punjab Giant-26, Pant Shubhra, Dania Kalimpong, Cabbage , Golden Acre, Pusa Mukta, Pusa Drumhead, K-1, Peas ,Asauji , Lucknow Boniya, Alaska, Bonneville , T-19, Okra ,Pusa Makhmali, Punjab Padmini, Pusa Sawani, Parbhani Kranti, Arka Anamika).] [30]
3.2	Fresh Vegetables	where Fresh Vegetables cultivated	West Bengal , Uttar Pradesh , Bihar, Andhra Pradesh , Madhya Pradesh ,

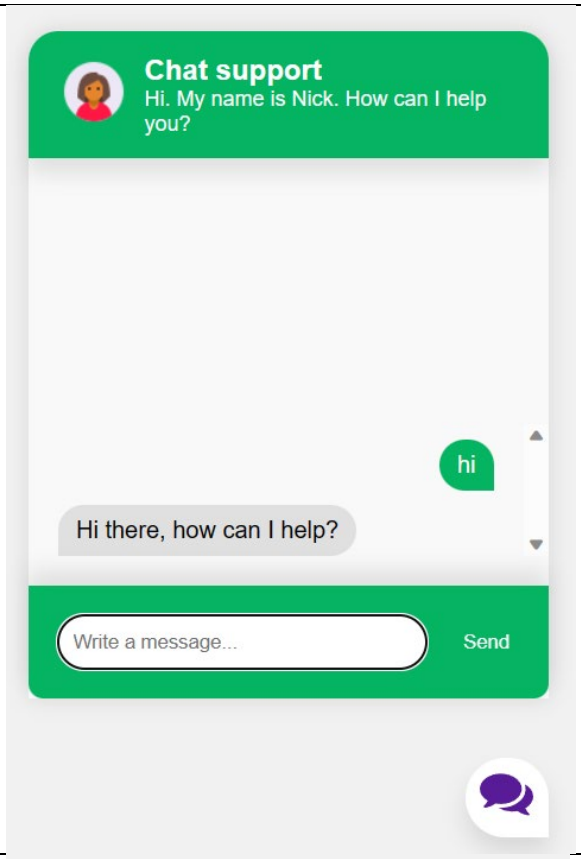
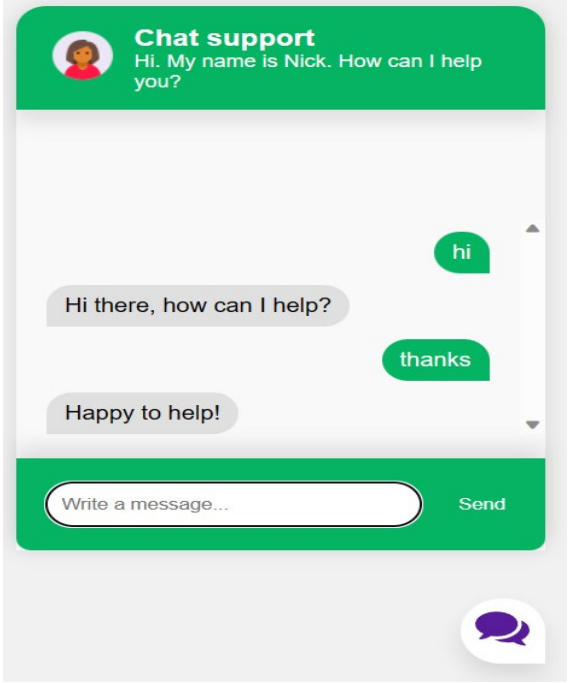
	Areas of Cultivation	, Indian Fresh Vegetables cultivation , best place to cultivate Fresh Vegetables , Fresh Vegetables producing states [30]	Gujarat, Orissa, Tamil Nadu, Maharashtra, Karnataka, Haryana, Chhattisgarh, Jharkhand] [30]
3.3	India Facts and Figures Fresh Vegetables	Fresh Vegetables Facts & Figures , Indian Fresh Vegetables Fats Results , Indian Fresh Vegetables Reports [30]	India is a prominent exporter of fresh vegetables in the world. The country has exported 770,233.19 MT of fresh vegetables other than onion to the world, worth Rs. 2,160.72 crores/ 290.09 USD Millions during the year 2021-22] [30]
3.4	Fresh Vegetables Major Export Destinations	where Indian Fresh Vegetables export more , Fresh Vegetables export destinations , countries Indian Fresh Vegetables exports [30]	United Arab EMTs , Qatar, Oman , New Zealand, Bangladesh , UK] [30]
4	Fresh Fruits	Fresh Fruits , about Fresh Fruits , facts about Fresh Fruits ?,Indian Fresh Fruits , about Indian Fresh Fruits [30]	India is the largest producer of Fruits in the world and is known as the fruit basket of the world. The major fruits grown in India are Mangoes, Grapes, Apples, Apricots, Oranges, Fresh Banana, Avocados, Guava, Litchi, Papaya, Sapota and Water Melons] [30]
4.1	Fresh Fruits Varieties	what are varieties of Fresh Fruits , Fresh Fruits varieties found in India , Fresh Fruits types found in India [30]	Mangoes,Langra, Chausa , Fazli, Krishna Bhog, Himsagar, Neelam , Baneshan , Badami , Grapes , Anabeshahi , CheemaSahebi, Kishmish Chorni, Perlette, Arkavati, Chaubattia,Anupam, Lal Ambri, Golden Delicious, Banana Fresh ,Dwarf Cavendish, Robusta, Rasthali, Poovan, Guava, L49, Allahabad Safeda , Banarasi, Chittidar, Harijha,Papaya] [30]

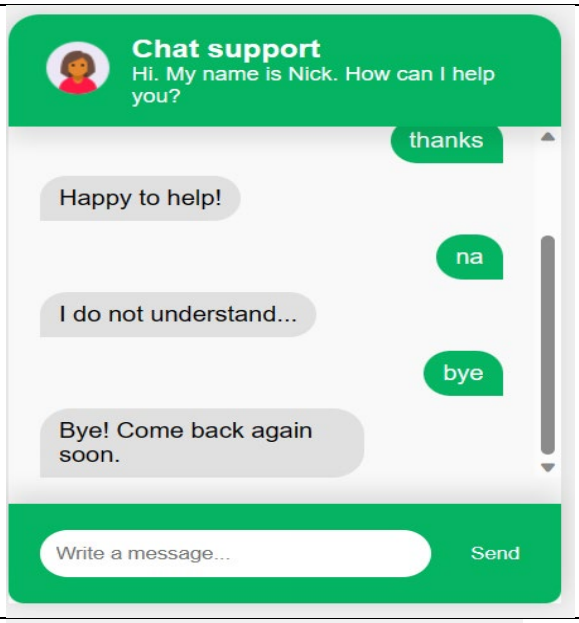
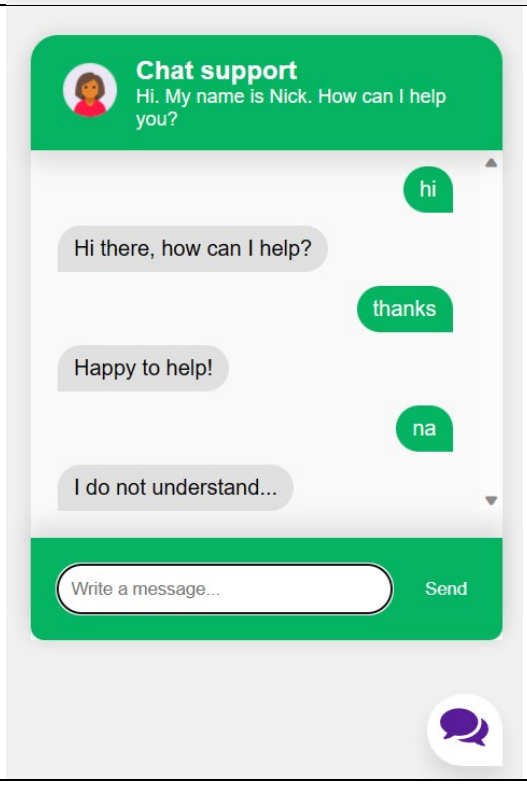
4.2	Fresh Fruits Areas of Cultivation	where Fresh Fruits cultivated , Indian Fresh Fruits cultivation , best place to cultivate Fresh Fruits , Fresh Fruits producing states [30]	West Bengal , Uttar Pradesh , Bihar, Andhra Pradesh , Madhya Pradesh , Gujarat, Orissa, Tamil Nadu, Maharashtra, Karnataka, Haryana, Odissa ,Assam Uttar Pradesh, Kerala, Madya Pradesh] [30]
4.3	India Facts and Figures Fresh Fruits	Fresh Fruits Facts & Figures , Indian Fresh Fruits fats Results , Indian Fresh Fruits Reports [30]	India is a major exporter of fruits to the world. The country has exported 761,031.20 MT of fresh fruits other than Grapes and Mango to the world, worth Rs. 2,900.73 crores/ 388.42 USD Millions during the year 2021-22.] [30]
4.4	Fresh Fruits Major Export Destinations	where Indian Fresh Fruits export more , Fresh Fruits export destinations , countries Indian Fresh Fruits exports [30]	United Arab EMTs , Qatar, Oman , New Zealand, Iran, Nepal [30]
5	Fresh Mangoes	Fresh Mangoes , about Fresh Mangoes , facts about Fresh Mangoes ?,Indian Fresh Mangoes , about Indian Fresh Mangoes [30]	The Indian mango is a special product that substantiates the high standards of quality and bountiful nutrients packed in it. A single mango can provide up to 40 percent of the daily dietary fiber needs - a potent protector against heart disease, cancer and cholesterol build-up. In addition, this luscious fruit is a warehouse of potassium, beta- carotene and antioxidants. In India, mangoes are mainly grown in tropical and subtropical regions from sea level to an altitude of 1,500m. Mangoes grow best in temperatures around 27] [30]
5.1	Fresh Mangoes Varieties	what are varieties of Fresh Mangoes , Fresh Mangoes varieties found in India , Fresh Mangoes types found in India [30]	Banganapalli , Suvarnarekha , Neelum,Totapuri ,Bombay Green, Dashehari, Fazli, Gulabkhas, Kishen Bhog, Himsagar, Zardalu,Langra ,Kesar, Alphonso, Rajapuri, Jamadar, Totapuri, Neelum, Dashehari, Langra] [30]

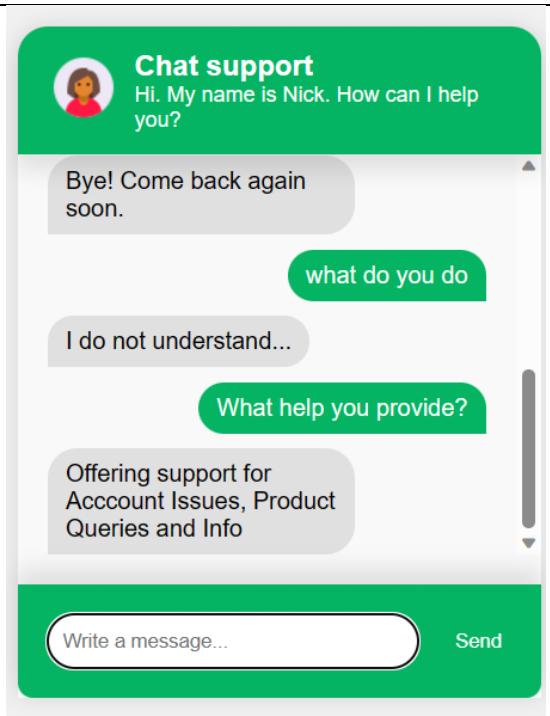
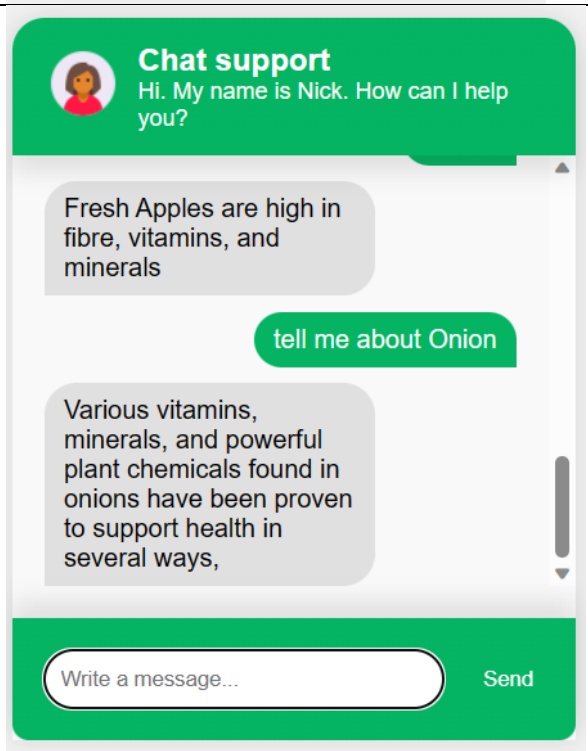
5.2	Fresh Mangoes Areas of Cultivation	where Fresh Mangoes cultivated , Indian Fresh Mangoes cultivation , best place to cultivate Fresh Mangoes , Fresh Mangoes producing states [30]	Uttar Pradesh , Andhra Pradesh , Karnataka, Bihar, Gujarat ,Telangana [30]
5.3	India Facts and Figures Fresh Mangoes	Fresh Mangoes Facts & Figures , Indian Fresh Mangoes Facts Results , Indian Fresh Mangoes Reports [30]	India is also a prominent exporter of fresh mangoes to the world. The country has exported 27,872.78 MT of fresh mangoes to the world for the worth of Rs. 327.45 crores/ 44.05 USD Millions during the year 2021-22.] [30]
5.4	Fresh Mangoes Major Export Destinations	where Indian Fresh Mangoes export more , Fresh Fruits export destinations , countries Indian Fresh Mangoes exports [30]	United Arab EMTs , Qatar, Oman , UK , Iran, Kuwait ,Canada, Japan , Australia] [30]
6	Fresh Grapes	Fresh Grapes , about Fresh Grapes , facts about Fresh Grapes ?,Indian Fresh Grapes , about Indian Fresh Grapes [30]	Grape Vitis vinifera is grown in temperate to warm regions however hot and dry climate is ideal for its cultivation. Indian grapes come in varied characteristics namely colored,white,seeded,unseeded,large and small berries. Indian grapes are successfully grown at and above 250 mean sea levels.] [30]
6.1	Fresh Grapes Varieties	what are varieties of Fresh Grapes , Fresh Grapes varieties found in India , Fresh Grapes types found in India [30]	Bangalore Blue, Thompson Seedless ,GulabiMuscat,Beauty Seedless, Sharad Seedless,Anab -e-Shahi, Dilkhush (clone of Anab-e-Shahi),Perlette, Pusa Seedless ,Thompson Seedless, Tas-A-Ganesh Sonaka,Manik Chaman] [30]
6.2	Fresh Grapes	where Fresh Grapes cultivated , Indian Fresh Grapes	Maharashtra , Andhra Pradesh , Karnataka, Tamilnadu, Mijoram] [30]

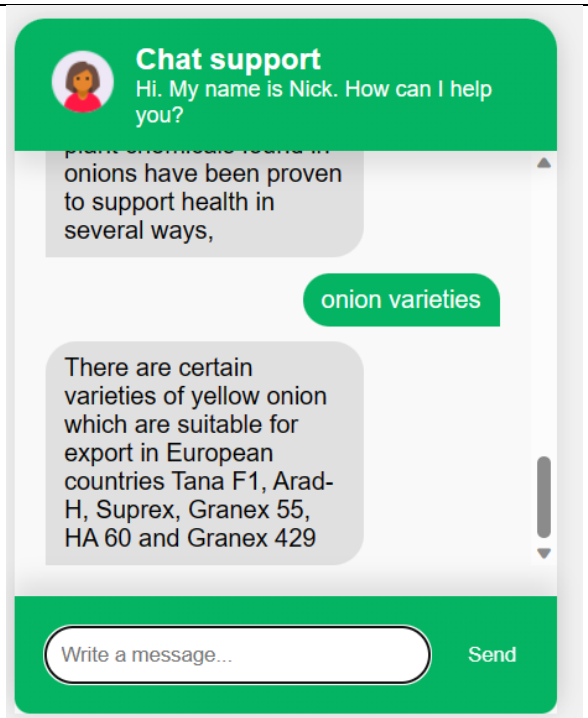
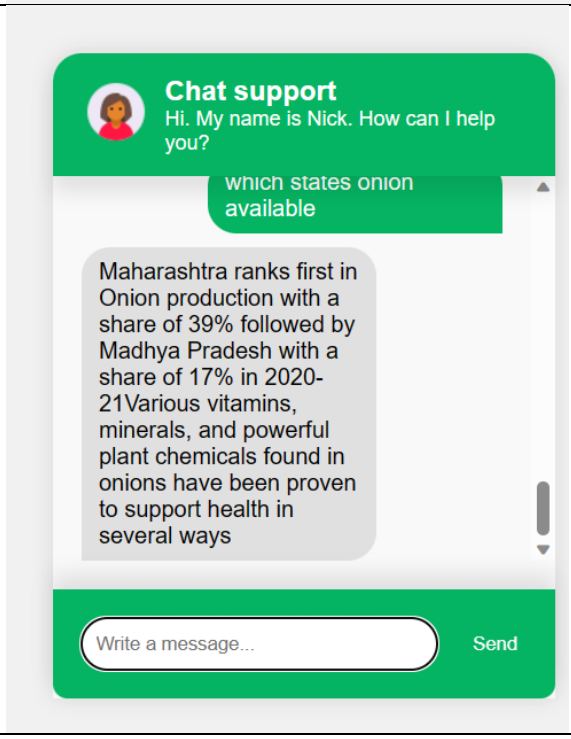
	Areas of Cultivation	cultivation , best place to cultivate Fresh Grapes , Fresh Grapes producing states [30]	
6.3	India Facts and Figures Fresh Grapes	Fresh Grapes Facts & Figures , Indian Fresh Grapes fats Results , Indian Fresh Grapes Reports [30]	Grape is one of the important fruits covering an area of 155.30 thousand hectares occupying 2.24 % of the total area in 2020-21. The country is also a major exporter of fresh Grapes to the world. The country has exported 263,075.67 MT of Grapes to the world, worth Rs. 2,302.16 crores/ 305.66 USD Millions during the year 2021-22] [30]
6.4	Fresh Grapes Major Export Destinations	where Indian Fresh Grapes export more , Fresh Fruits export destinations , countries Indian Fresh Grapes exports [30]	United Arab EMTs , Qatar, Oman , Germany ,UK , Iran, Kuwait ,Canada, Japan , Australia, Russia, Netherlands] [30]
7	BETEL LEAVES	BETEL LEAVES , about BETEL LEAVES , facts about BETEL LEAVES ?,Indian BETEL LEAVES , about Indian BETEL LEAVES [30]	The Betel is the leaf of the vine. In India, it is known as paan Betel vine is a perennial, evergreen climber which grows in tropics and subtropics. Betel leaf is mostly consumed in Asia and elsewhere in the world by some Asian emigrants. Today, betel is grown for local consumption and exports. Major betel leaf growing countries are Sri Lanka, India, Thailand and Bangladesh] The areca nut is the fruit of the areca palm (Areca catechu), which grows in much of the tropical Pacific, Southeast and South Asia, and parts of east Africa. It is commonly referred to as betel nut so it is easily confused with betel (Piper betel) leaves that are often used to wrap it (paan] [30]
7.1	BETEL LEAVES Varieties	what are varieties of BETEL LEAVES , BETEL LEAVES	Karapaku , Chennor, Tellaku, Bangla, Kalli Patti , Assam Patti , Awani pan ,Khasi pan],

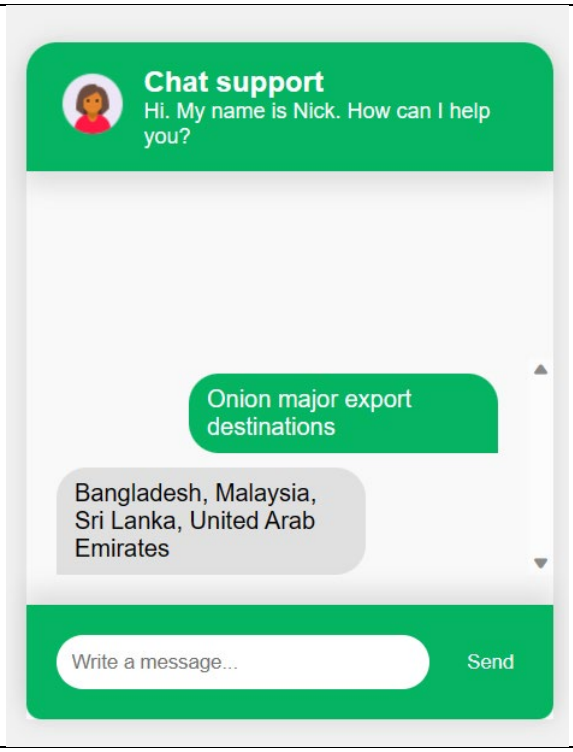
		varieties found in India , BETEL LEAVES types found in India [30]	
7.2	BETEL LEAVES Areas of Cultivation	where BETEL LEAVES cultivated , Indian BETEL LEAVES cultivation , best place to cultivate BETEL LEAVES , BETEL LEAVES producing states [30]	Assam Andhra Pradesh, Gujarat, Odisha Karnataka, Madhya Pradesh, West Bengal, Maharashtra] [30]
7.3	India Facts and Figures BETEL LEAVES	BETEL LEAVES Facts & Figures , Indian BETEL LEAVES fats Results , Indian BETEL LEAVES Reports [30]	The country has exported 6517.26 MT of Betel Leaves to the world, worth Rs. 45.97 crores/ 6.17 USD Millions in 2021-22..] [30]
7.4	BETEL LEAVES Major Export Destinations	where Indian BETEL LEAVES export more , Fresh Fruits export destinations , countries Indian BETEL LEAVES exports [30]	United Arab EMTs , Australia, Canada, Sri Lanka , Indonesia , Singapore] [30]

Category	Screenshot
Greetings	
Thanks	

Goodbye	
No Answer	

Options	
Fresh onions	

Onions varieties	
Onion Areas of Cultivation	

Onions Export	

5.2 WEBSITE TEST CASE SCENARIOS

20 Test cases to be tested for functionality to work

S NO	Process	Use case	Result (Pass/Fail)
1	Account logon	Navigate to logon page Welcome	Pass
2	Login - Enter user ID /Password	Login Success	Pass
3	Login - Enter user ID incorrect	Message about wrong User ID	Pass
4	Login – Enter Password incorrect	Message about wrong User ID	Pass
5	Register	Navigate to Register page Select Type of Type (Consumer /Seller)	Pass
6	Select Consumer Type Enter Name /Email/Phone number /Password	Registration success full	Pass
7	Select Consumer Type Enter Name wrong email format	Error message with missing format for email	Pass
8	Select Consumer Type Enter Name / email / Phone number in Text format	Error message with Numeric format for email	Pass
9	Select Consumer Type Enter Name / same email already Registered	Error message email already registered	Pass
10	Select Consumer Type Enter Name / same Mobile already Registered	Error message Mobile number already registered	Pass
11	Select Consumer Type Enter Name / incorrect mobile number	Error message use correct Mobile number	Pass
12	Without verification email Logon	Error message email account not verified	Pass
14	Forgot Password	Navigate to forgot password page	Pass
15	Forgot user Id	Navigate to forgot user Id page	Pass
		Enter Email/Phone number to retrieve password & user Id	Pass

	Admin Role			Result (Pass/Fail)
1	Create Category	Add name /Category image /Description	Add category	Pass
2	Create Sub Category	Add name /Category image /Description	Add Sub category	Pass
3	Create Product	Add Title /Description/Category/Subcategory/ General Info/Location /Image	Add Product	Pass
4	Edit Product	Edit Title /Description/Category/Subcategory/ General Info/Location /Image	Edit Product	Pass
5	Delete Product	Delete Title /Description/Category/Subcategory/ General Info/Location /Image	Delete Product	Pass
6	Manage Seller	Manage seller data		Pass
7	Manage seller product	Action Approve /Reject /Delete		Pass
8	Manage order	Approve /Reject /Delete		Pass
9	Manage Order status	New/Process/Delivered		---
10	User data management	Able to view complete data of Seller & Consumer		pass
11	Responsive	Use mobile views		

	Seller Role		Result (Pass/Fail)
1	Account logon	Navigate to logon page Welcome	Pass
2	Login - Enter user ID /Password	Login Success	Pass
3	Login - Enter user ID incorrect	Message about wrong User ID	Pass
4	Login – Enter Password incorrect	Message about wrong User ID	Pass
5	Register	Navigate to Register page Select Type of Type (Seller)	Pass
6	Select seller Type Enter Name /Email/Phone number /Password	Registration success full	Pass
7	Select seller Type Enter Name/ wrong email format	Error message with missing format for email	Pass
8	Select seller Type Enter Name / email / Phone number in Text format	Error message with Numeric format for phone number	Pass
9	Select Seller Type Enter Name / same email already Registered	Error message email already registered	Pass
10	Select seller Type Enter Name / same Mobile already Registered	Error message Mobile number already registered	Pass
11	Select seller Type Enter Name / incorrect mobile number	Error message use correct Mobile number	Pass
12	Without verification email Logon	Error message email account not verified	Pass
14	Forgot Password	Navigate to forgot password page	Pass
15	Forgot user Id	Navigate to forgot user Id page	Pass
16	Update Profile	Enter address & other details	Pass
17	Update Product	Enter Product details /place of availability	Pass

	Consumer Role		Result (Pass/Fail)
1	Account logon	Navigate to logon page Welcome	Pass
2	Login - Enter user ID /Password	Login Success	Pass
3	Login - Enter user ID incorrect	Message about wrong User ID	Pass
4	Login – Enter Password incorrect	Message about wrong User ID	Pass
5	Register	Navigate to Register page Select Type of user (Seller(farmer))	Pass
6	Select seller Type Enter Name /Email/Phone number /Password	Registration success full	Pass
7	Select seller Type Enter Name/ wrong email format	Error message with missing format for email	Pass
8	Select seller Type Enter Name / email / Phone number in Text format	Error message with Numeric format for phone number	Pass
9	Select Seller Type Enter Name / same email already Registered	Error message email already registered	Pass
10	Select seller Type Enter Name / same Mobile already Registered	Error message Mobile number already registered	Pass
11	Select seller Type Enter Name / incorrect mobile number	Error message use correct Mobile number	Pass
12	Without verification email Logon	Error message email account not verified	Pass
14	Forgot Password	Navigate to forgot password page	Pass
15	Forgot user Id	Navigate to forgot user Id page	Pass
16	Update Profile	Enter address & other details	Pass
17	Buy product	Select product add to cart & Buy product	Pass
18	Search functionality	Search for Product	Pass

5.3 WEBSITE STATIC PAGES

a. Homepage

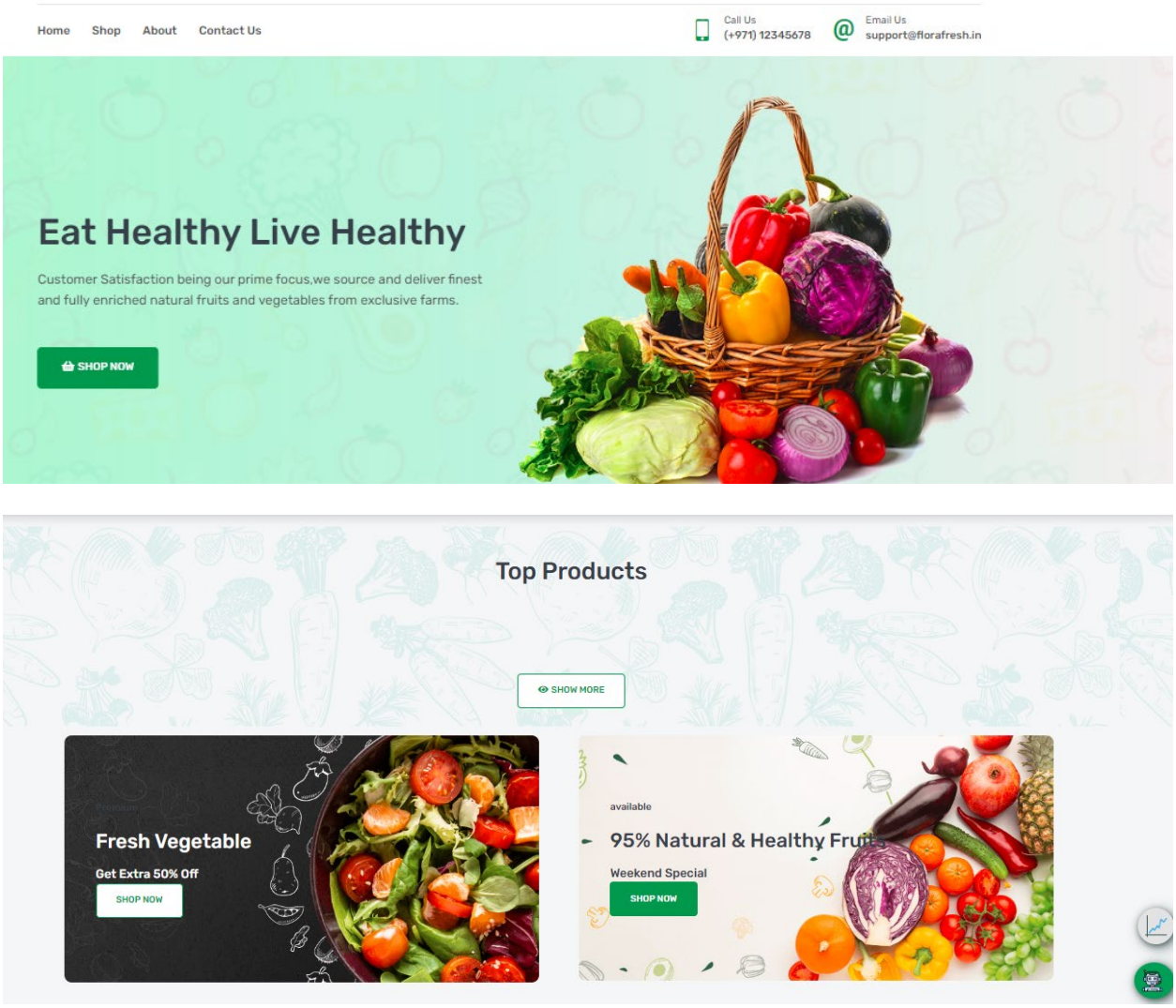


Figure 33 – Screenshot of Homepage

b. Shop Page

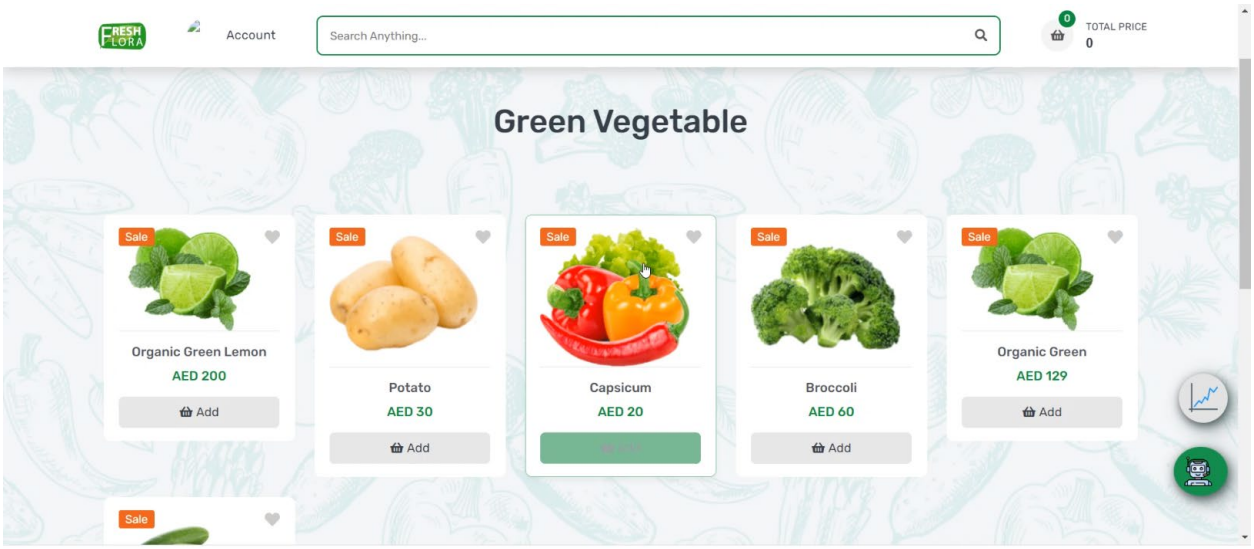


Figure 34 – Screenshot of shop Page

c. Contact Us Page

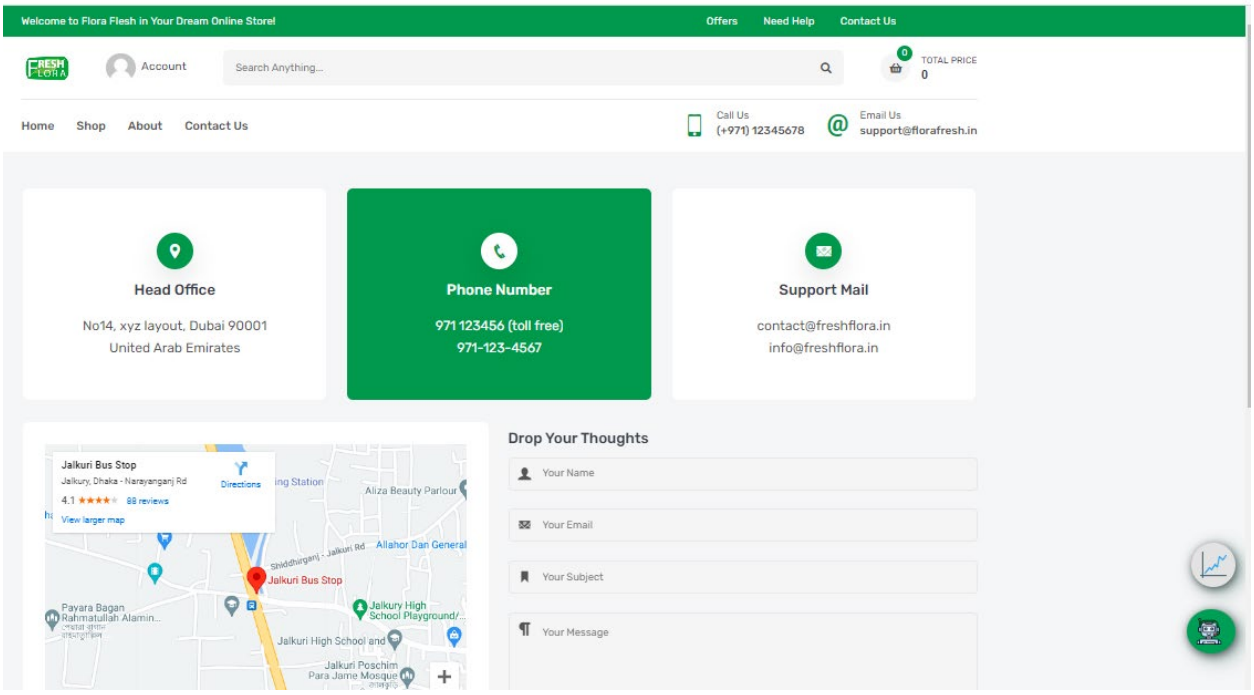


Figure 35 – Screenshot of Contact Us Page

d. Cart Page

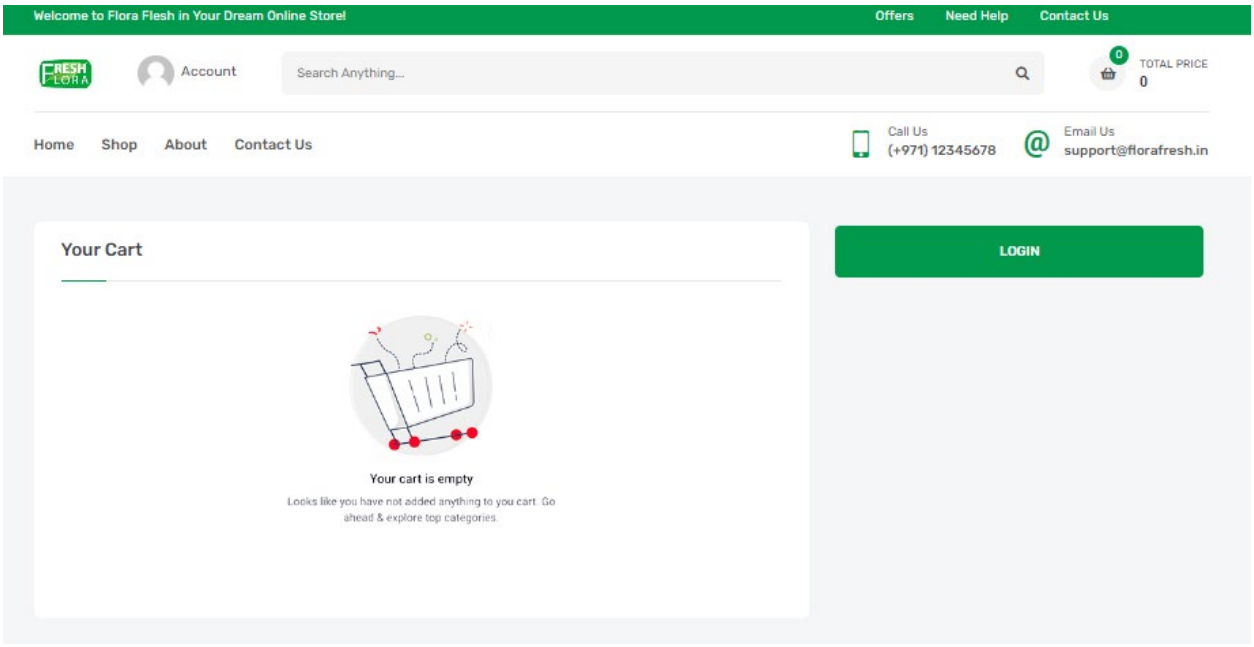


Figure 36 – Screenshot of Cart

5.4 WEBSITE CONSUMER SCREENS

5.4.1 Registration

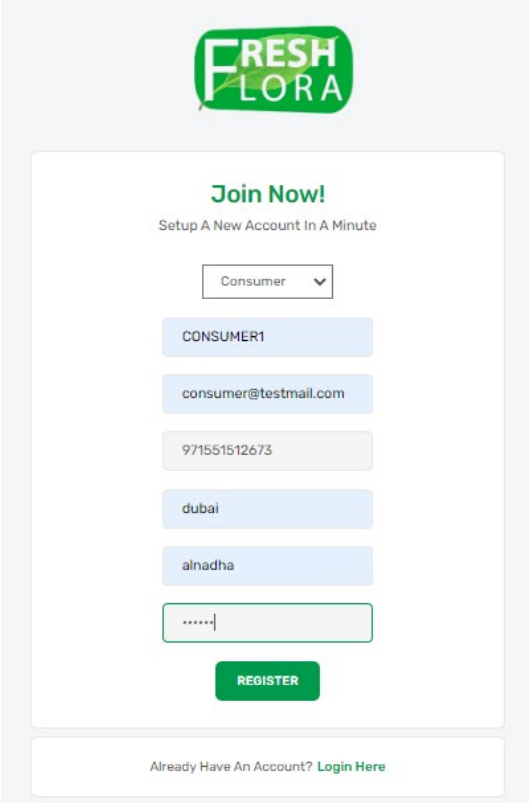


Figure 37 – Sign up Form Consumer

5.4.2 Login

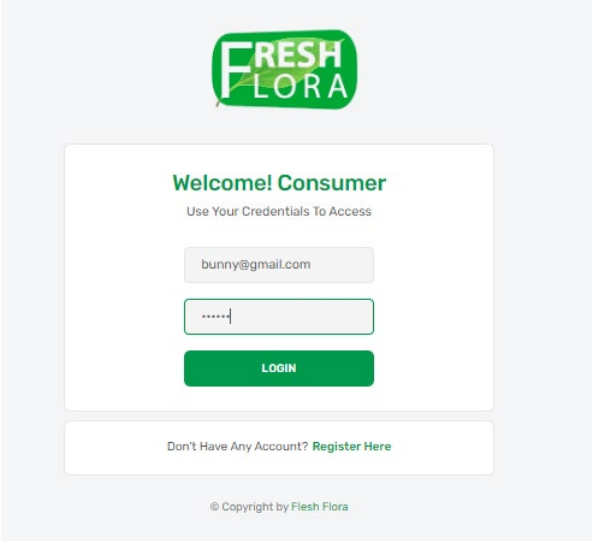


Figure 38 – login page Consumer

5.4.3 Profile

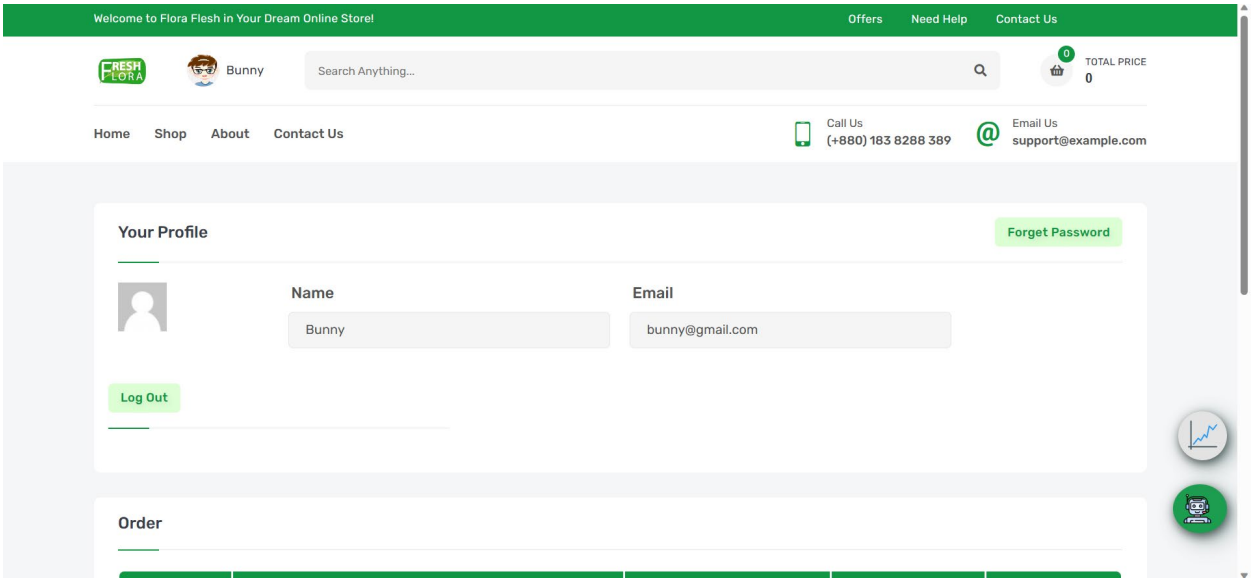


Figure 39 – profile page Consumer

5.4.5 Order

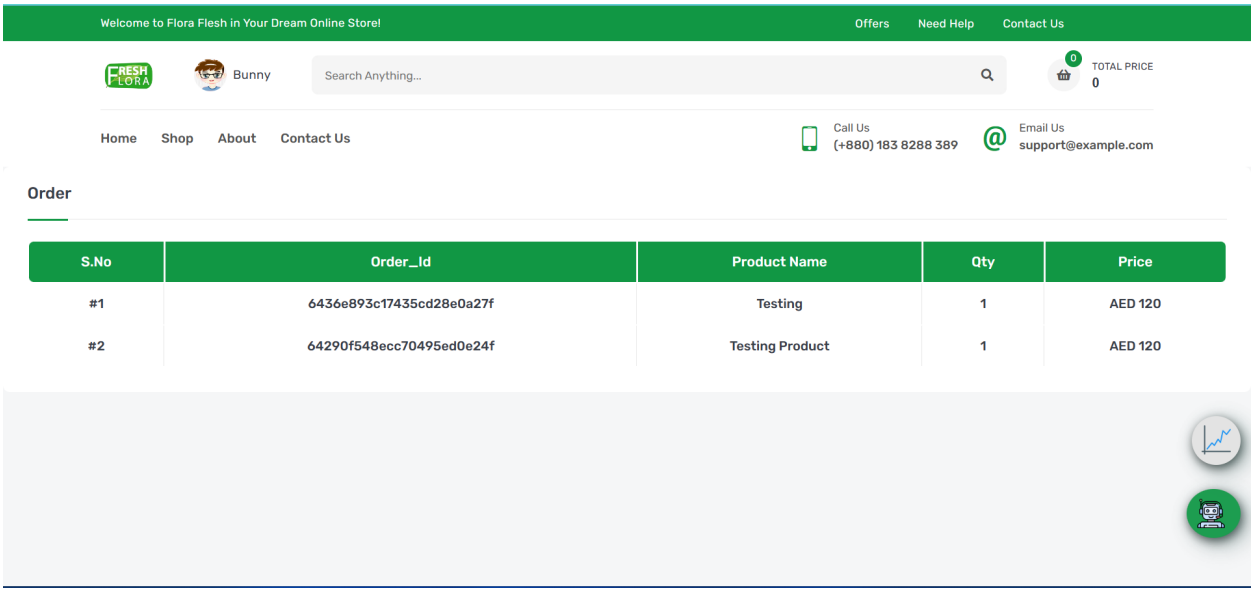


Figure 40 – Order Section Consumer

5.5 WEBSITE SELLER SCREENS

5.5.1 Registration

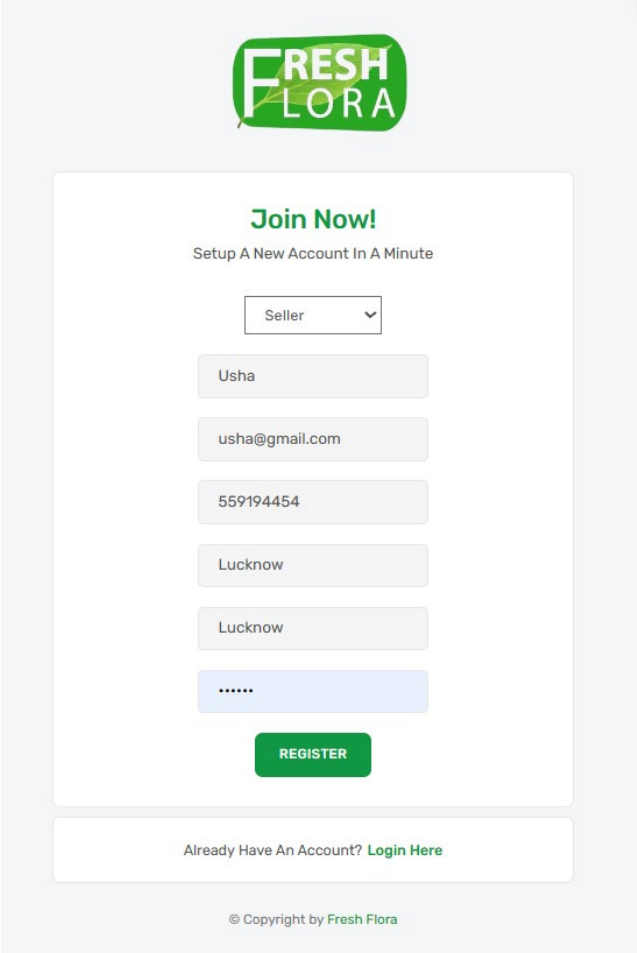


Figure 41 – Registration page Seller (Farmer)

5.5.2 Login

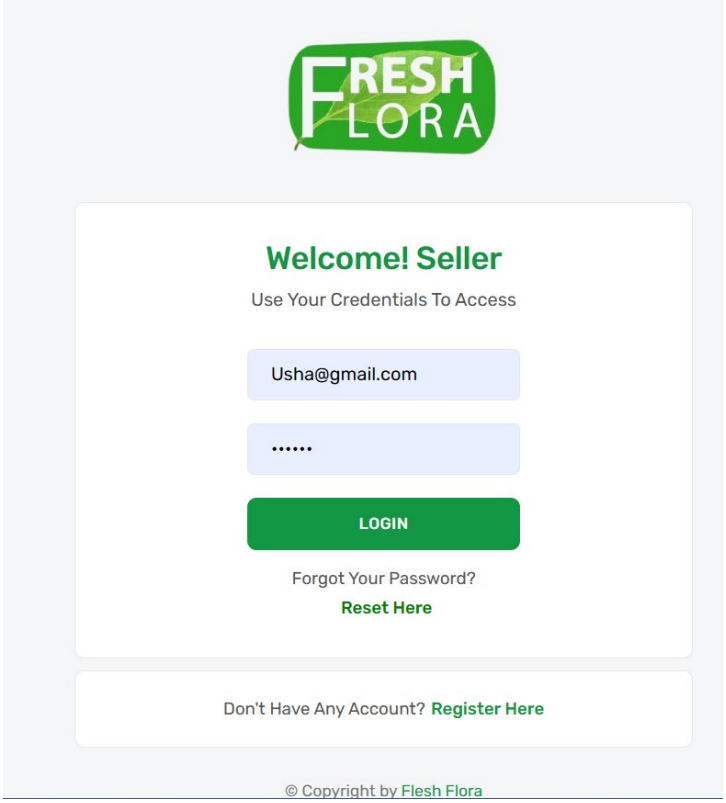


Figure 42 – login page Seller (Farmer)

5.5.3 Dashboard

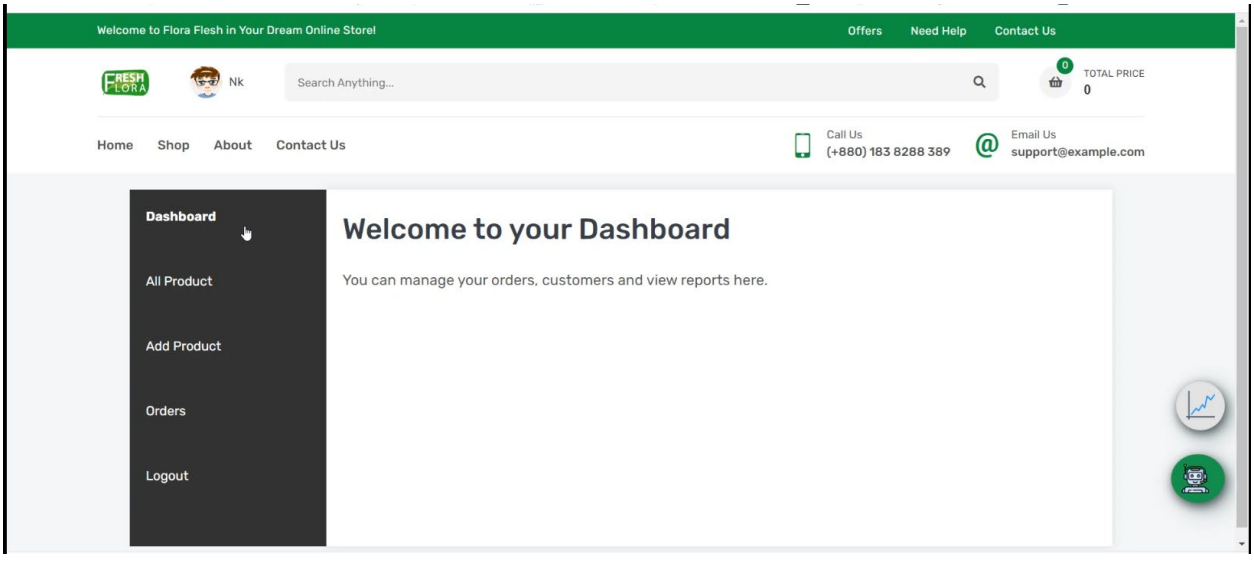


Figure 43 – Dashboard Page

5.5.4 Profile

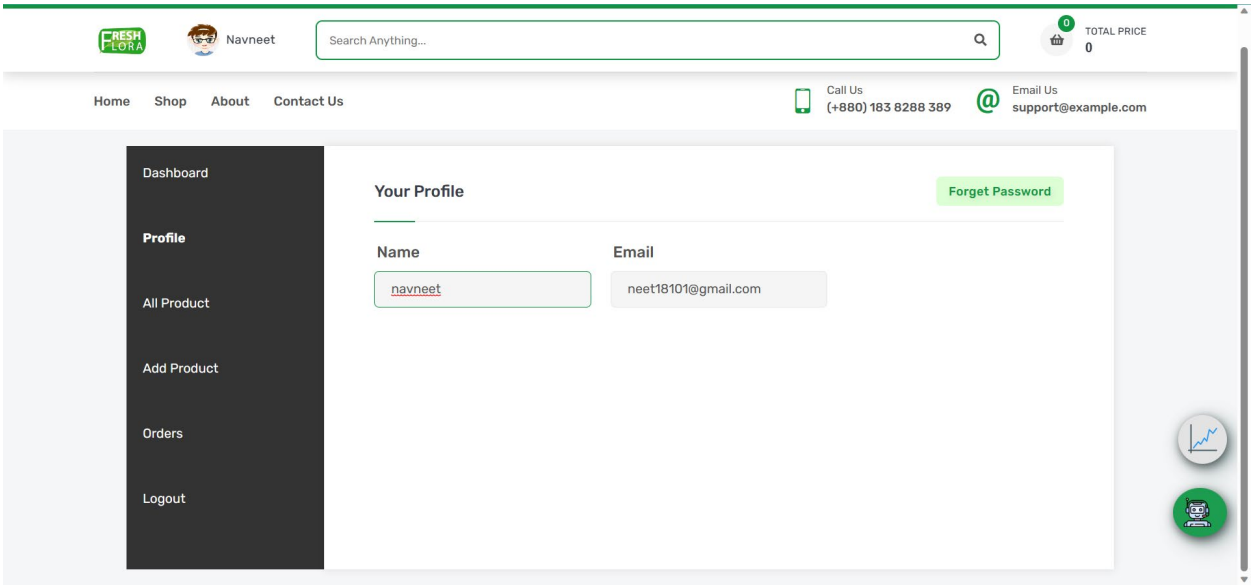


Figure 44 – Profile Page Seller

5.5.5 Add Product

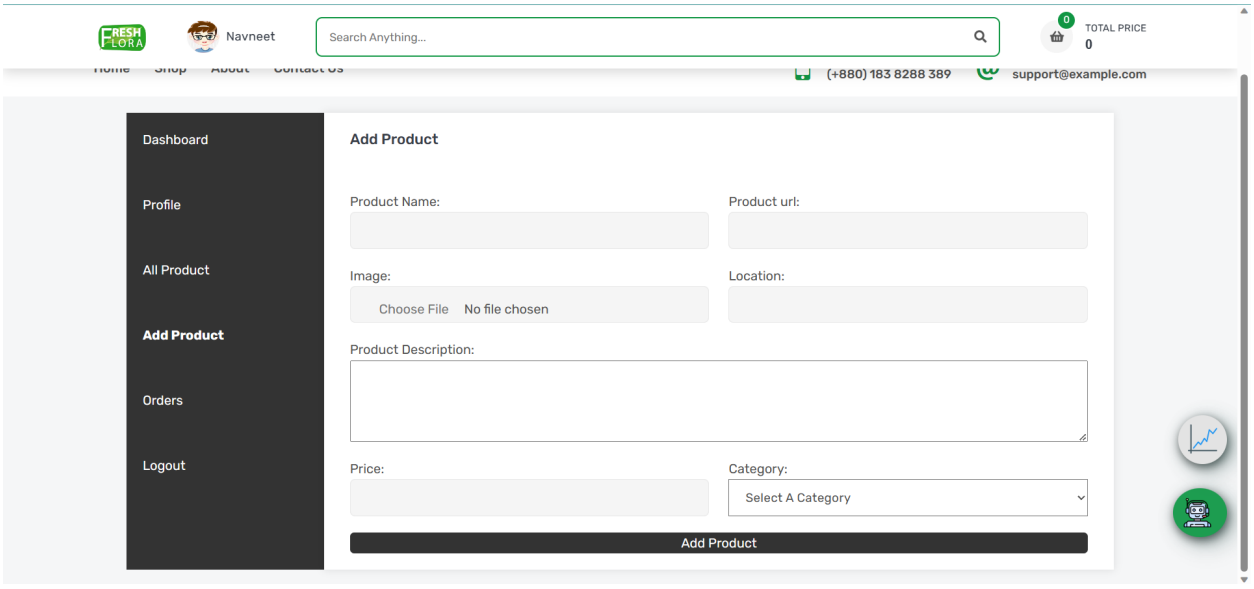


Figure 45 – Add Product Seller (Farmer)

5.5.6 Order

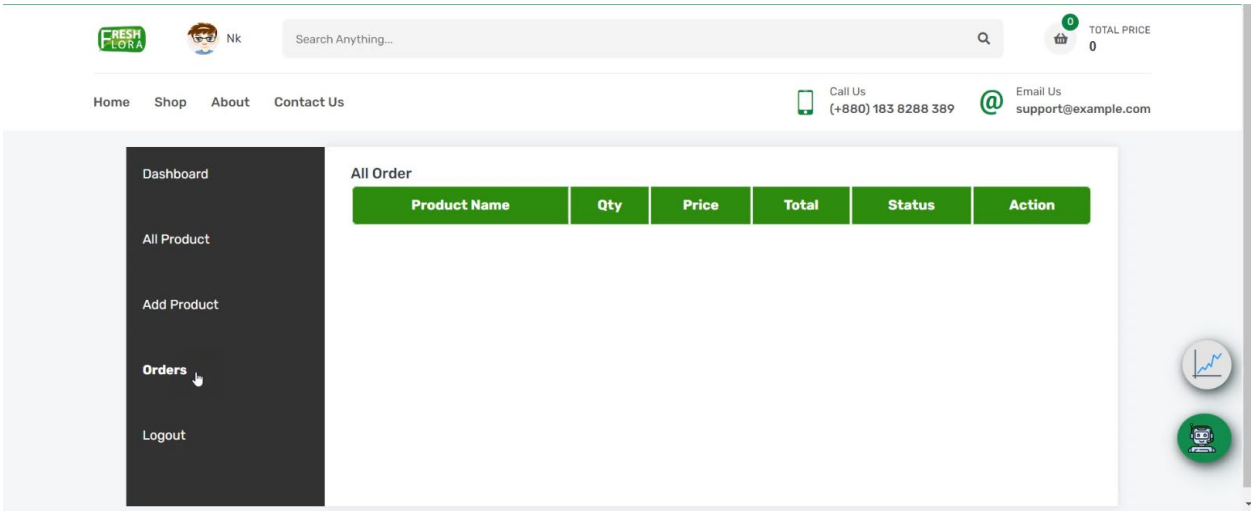


Figure 46 – Order Section Seller (Farmer)

5.5.7 All Product

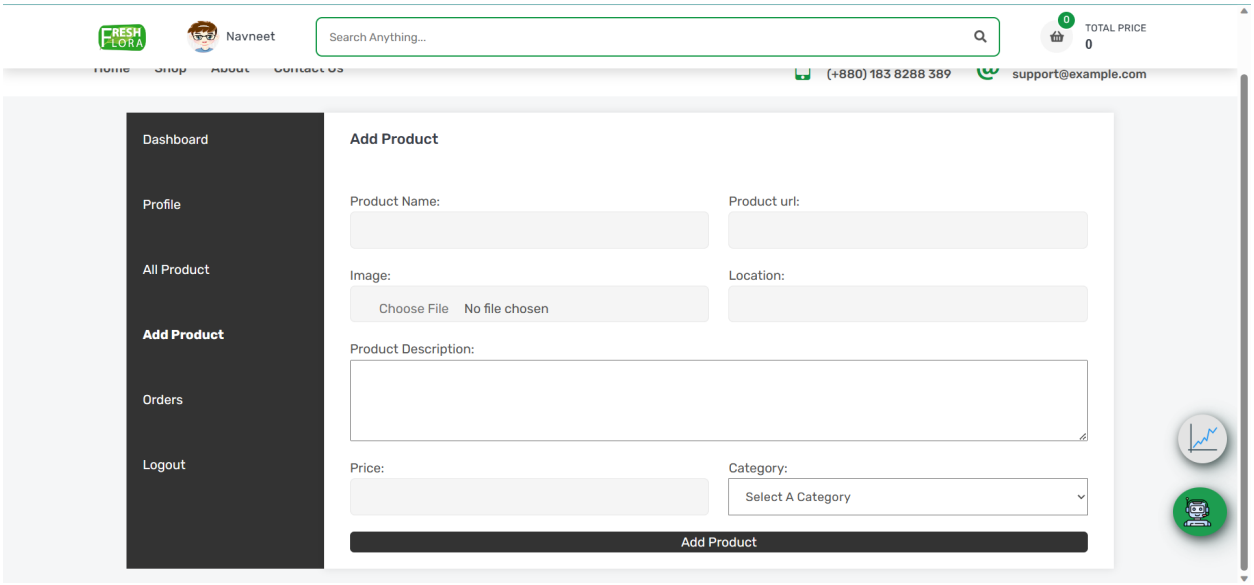


Figure 47 – All Product Section Seller (Farmer)

5.6 WEBSITE ADMIN PROFILE SCREENS

5.6.1 Login

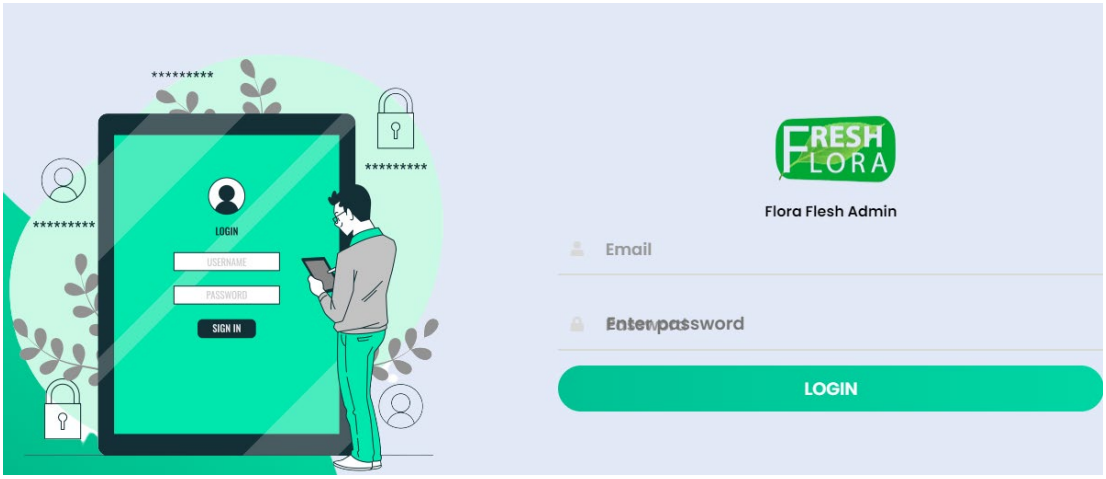


Figure 48 – login page Admin

5.6.2 Dashboard

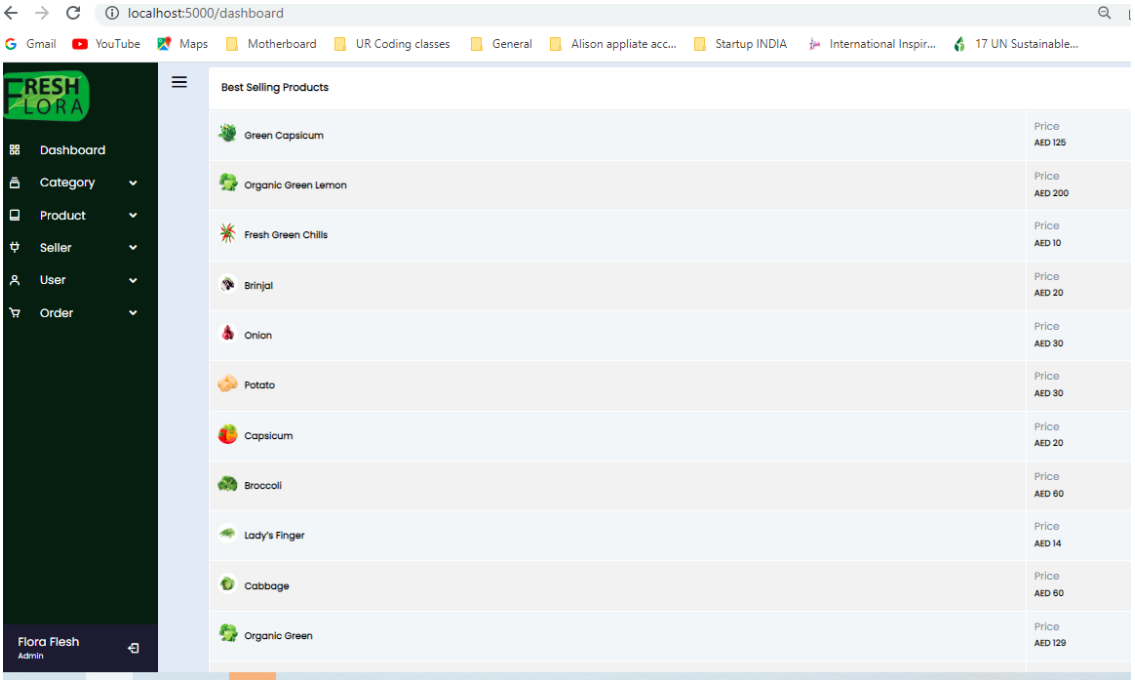


Figure 49 – Dashboard Admin

5.6.3 Categories

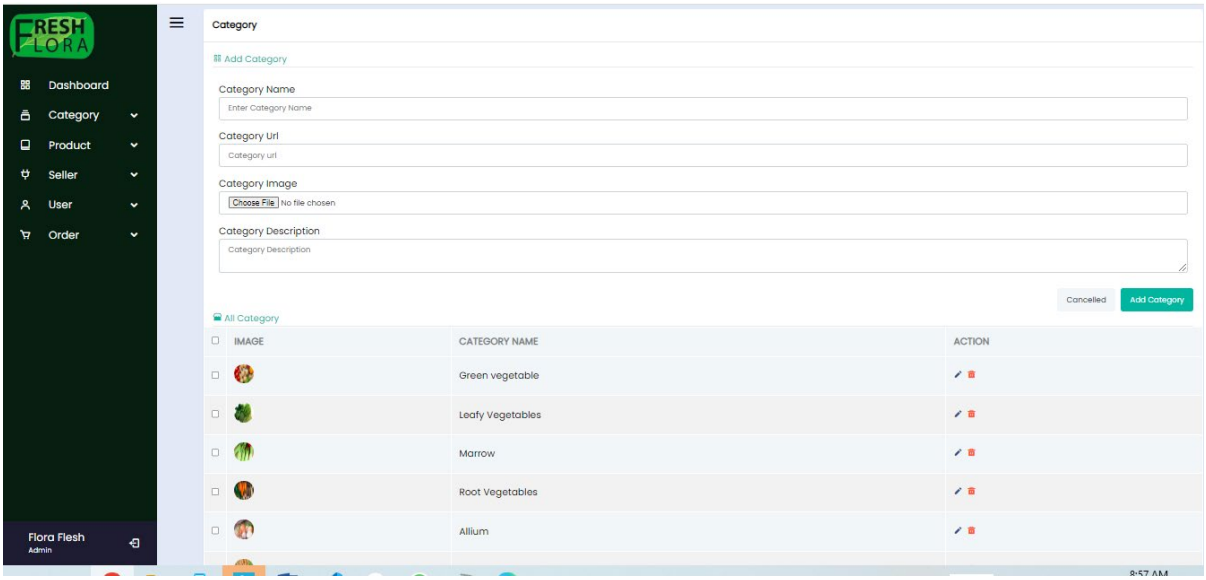


Figure 50 – Category Admin

5.6.4 Subcategories

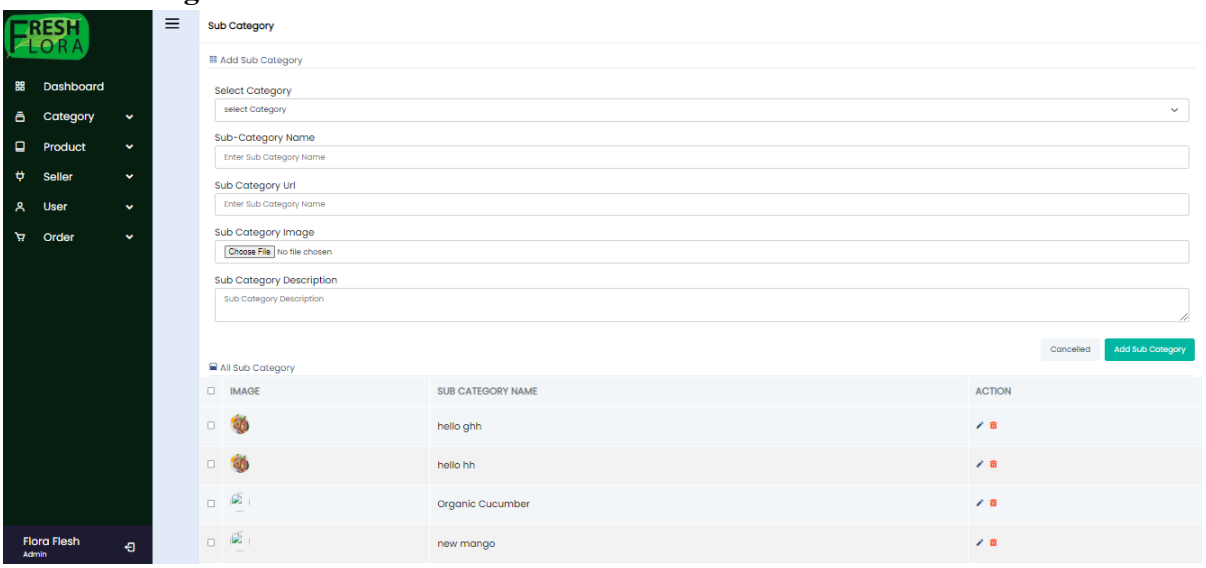


Figure 51 – Sub- Category Admin

5.6.5 Products

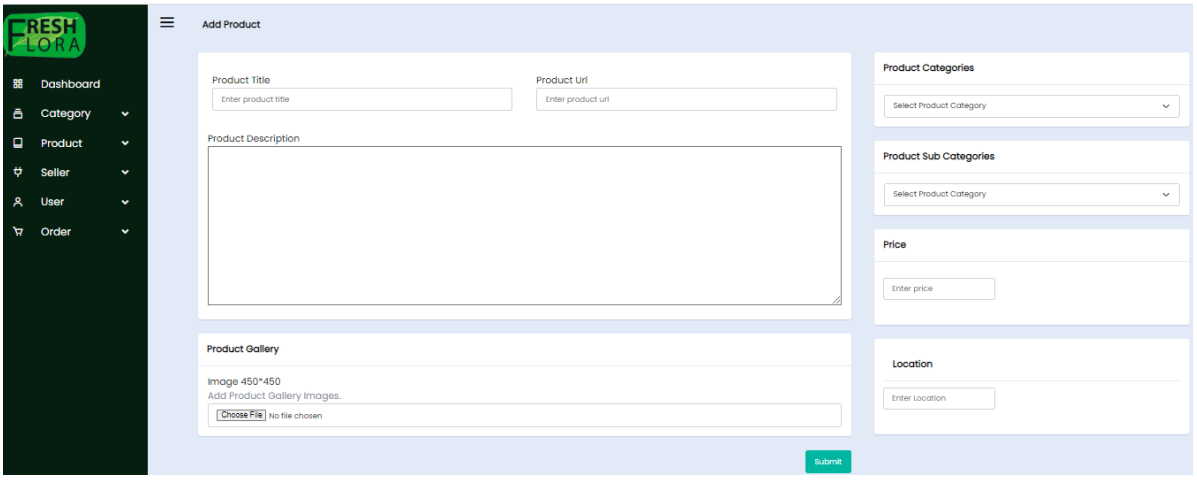


Figure 52 – Add Product Admin

Figure 53 – products View Admin

5.6.6 Sellers & seller products

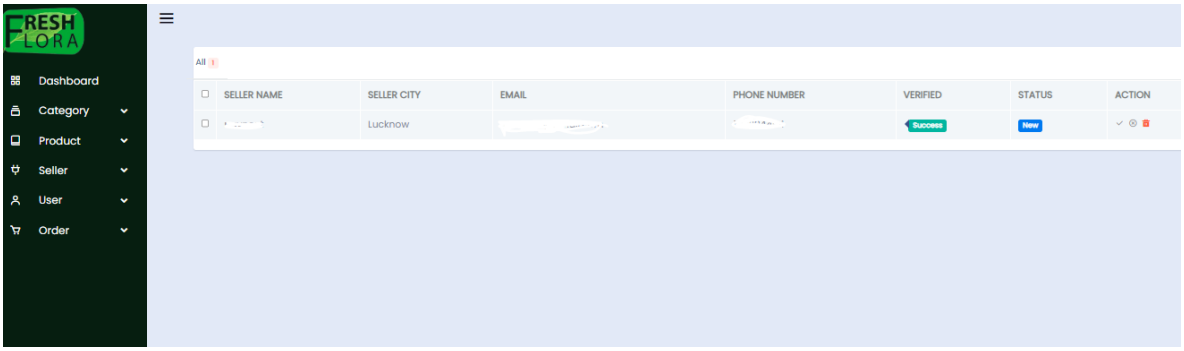


Figure 54 – Sellers View Admin

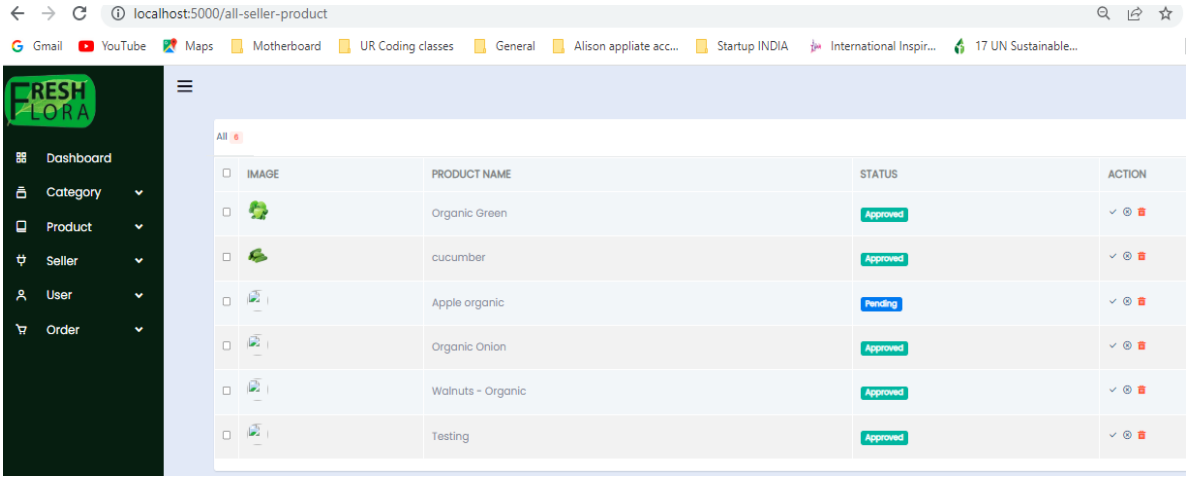


Figure 55 – Seller Products Admin

5.7 PROJECT MILESTONE TABLE

Week	Project Task	Deliverables
Autumn: 7 - 8	Understanding of Project requirements & study & validate available resources with Project finalize Scope & Prepare Project plan	Scope of the work
Autumn: 9 - 10	Prepare Prerequisites for hardware & software for Project & submit use cases collected & get approval from Project	Finalize scope & get access to test Systems requirement provided to Project
Autumn: 11 - 12	Prepare Prototype & showcasing the basic prototype using the use case scenarios.	Test system (non-interactive website)
Autumn: 13 - 15	No project work - exams	

Week	Project Task	Deliverables
Pre-Spring	Continue developing prototypes and testing. Deploy the navigation and the profile menu on the website.	
Spring: 1-2	prepare and present navigation and profile section on the website. Test website response using the use case Scenario created.	Technical Implementation
Spring: 3-4	Creation and linking of Database to the Website for data storing.	Database integration
Spring: 5-6	Testing of the DB connectivity and verify data storing commands. Creation and integration of chatbot on the website. Testing the chatbot using the use case Scenario created.	Verification of Information storage. Testing the chatbot Interactivity.
Spring: 7-8	Final test of the Project with verification of all the functionalities.	Complete final Website testing for release.
Spring: 9-10	Prepare and deliver presentation to Supervisor (Mrs Preetha VK & Dr Mario) and also creation of Draft of the Project.	Presentation and Dissertation: technical development of the website
Spring: 11	Draft evaluation and conclusions of the Project. Submit draft to Supervisor (Mrs Preetha VK & Dr Mario) for feedback	Complete dissertation draft
Spring: 12 - 13	Revise draft, submit dissertation. Final code demo to be presented to the supervisor (Mrs Preetha VK).	Dissertation and Final application

5.8 SOFTWARE INSTALLATION

For the Development of the Entire project (Website) & Chatbot the required Software are below

1. Visual Studio Code – VS Code
2. Python
3. Mongo DB
4. Node JS
5. React
6. Express

5.8.1 Visual Studio Code Installation –

To install Visual Studio Code on Windows, it is recommended to follow the following steps:

Windows:

1. Download the installer: Visit the official Visual Studio Code website at https://code.visualstudio.com/ and click on the "Download for Windows" button to download the latest version of VS Code for Windows.

2. Run the installer: Double-click on the downloaded file (VSCodeUserSetup-{version}.exe) to start the installation process.

3. Accept the license agreement: Read the license terms and click on the "I accept the agreement" checkbox, then click "Next."

4. Choose the installation location: Select the folder where you want to install VS Code or leave the default location, then click "Next."

5. Select additional tasks: Choose options like creating a desktop icon, adding "Open with Code" in the file context menu, and registering the code as an editor for supported file types, then click "Next."

6. Confirm the installation: Review the selected settings, then click "Install" to begin the installation.

7. Complete the installation: After the installation is finished, you can choose to launch VS Code by checking the "Launch Visual Studio Code" checkbox, then click "Finish."

5.8.2 Python Installation -

To install Python on Windows, proceed as follows:

Windows:

1. Download the Python setup program: Visit the Python website at https://www.python.org/downloads/windows/ for more information. Download the latest Windows version of Python 3.x. Choose the 32-bit or 64-bit installer based on your operating system.
2. Utilize the installer: To initiate the installation procedure, double-click on the downloaded file.
3. Customize installation (optional): You can customize the installation by selecting the "Customize installation" button; however, the default settings are typically adequate.
4. Before selecting "Install Now", be sure to check the box "Add Python to PATH" at the bottom of the installer window. This will make it simple to execute Python from the command prompt.
5. Click "Install Now" and wait for the installation to complete to install Python.
6. Confirm installation: Open a Command Prompt or PowerShell prompt and enter python --version. You should see the version of Python that was deployed.
7. After installing Python on macOS and Windows, it is prudent to update pip, the Python package installer. Execute the specified command:
   python3 -m pip install --upgrade pip on macOS
   Python -m pip install --upgrade for Windows
8. Now that Python is installed on your macOS or Windows computer, you can begin executing scripts, installing packages, and developing applications.

5.8.3 Mongo DB Installation -

To install Mongo DB on Windows, it is recommended to follow the following steps:

Windows :

1. Download the MongoDB installer: Visit the official MongoDB website at https://www.mongodb.com/try/download/community and download the latest version of MongoDB Community Server for Windows. Choose the appropriate installer based on your system (MSI package for Windows).
2. Run the installer: Double-click on the downloaded file to start the installation process.
3. Follow the installation wizard: Accept the default settings, or customize the installation as needed.
4. Install MongoDB: Click "Install" and wait for the installation to complete.
5. Set up the MongoDB environment:

- Open the Command Prompt as an administrator.
- Run the following command to create the data directory:

  “mkdir %HOMEDRIVE%%HOMEPATH%\data\db “
- Add the MongoDB binary folder to the system PATH:

“Set x /M PATH "%PATH%; C:\Program Files\MongoDB\Server\<version>\bin" “

Replace <version> with the version number of MongoDB you installed.

6. Start MongoDB: Open a new Command Prompt as an administrator and run: “mongod “
7. Verify the installation: Open another Command Prompt window and type mongo. If the MongoDB shell starts, the installation was successful.

5.8.4 Node Js Installation -

To install Node.js on Windows, it is recommended to follow the following steps:

Windows :

1. To obtain Node.js, please download the installer. To obtain the most recent LTS (Long Term Support) version of Node.js for Windows, please visit the official Node.js website at https://nodejs.org/en/download/.
2. Select the suitable installer according to the specifications of your system, either 32-bit or 64-bit.
3. Execute the installation process: To initiate the installation process, it is necessary to double-click on the downloaded file.
4. Proceed with the installation wizard. One may choose to either adhere to the default settings or tailor the installation according to their specific requirements.
5. To install Node.js, the user should click on the "Install" button and patiently wait for the installation process to finish.
6. Confirm the installation: To verify the installed versions of Node.js and npm (Node Package Manager), access the Command Prompt or PowerShell and input the commands "node --version" and "npm --version".

5.8.5 React Installation -

To install React on Windows, it is recommended to follow the following steps:

Windows :

Once you have Node.js and npm installed, you can create a new React project using the following steps:

1. Open the Command Prompt (Windows).
2. Install the **create-react-app** command-line tool globally by running: “npm install -g create-react-app “
3. Create a new React project by running: “npx create-react-app my-react-app”
4. Replace **my-react-app** with the name you want for your project. This command will create a new folder with the specified name, and it will set up a new React application inside that folder.
5. Change to the project directory by running: “cd my-react-app”
5. Start the development server by running: “npm start”

5.8.6 Express Installation -

To install Express on Windows, it is recommended to follow the following steps:

Windows :

1. Open the Command Prompt (Windows).
2. Create a new directory for your Express project: “mkdir my-express-app && cd my-express-app”
3. Replace **my-express-app** with the name you want for your project.
4. Initialize a new Node.js project by running: “npm init”
5. This command will prompt you for some information about your project, such as name, version, description, and entry point. You can press Enter to accept the default values or provide your own. Once you're done, a **package.json** file will be created in your project directory.
6. Install Express as a dependency by running: “npm install express”
7. This command will download the Express package and add it to your project's **package.json** file.
8. Create a new file called **app.js** (or any other name you prefer) in your project directory. This file will contain your Express application code. You can create the file using a text editor or by running the following command:

6. REFERENCES

- **Website and DB Creation**

1. "The Ultimate Guide to Machine-Learning Chatbots and conversational AI: IBM Watson Advertising," *IBM*. [Online]. Available: https://www.ibm.com/watson-advertising/thought-leadership/machine-learning-chatbot. [Accessed: 27-Apr-2023].

2. "Node Js Tutorial," Node.js tutorial. [Online]. Available: https://www.w3schools.com/nodejs/default.asp. [Accessed: 15-Aug-2022].

3. What is JavaScript? [Online]. Available: https://learn.saylor.org/mod/book/tool/print/index.php?chapterid=20094&id=36725. [Accessed: 14-Apr-2023].

4. Admin, "JavaScript VS NodeJS: What is the difference?," Java Assignment Help, 07-Mar-2022. [Online]. Available: https://www.javaassignmenthelp.com/blog/javascript-vs-nodejs/. [Accessed: 07-Apr-2023].

5. "Responsive Dropdown Menu Bar with HTML & CSS | coding Nepal," YouTube, 13-Mar-2020. [Online]. Available: https://www.youtube.com/watch?v=Iyx_809qwoc. [Accessed: 11-Oct-2022].

6. "Introduction · REACT NATIVE," React Native RSS, 17-Jan-2023. [Online]. Available: https://reactnative.dev/docs/getting-started. [Accessed: 27-Feb-2023].

7. "Core components and api · REACT NATIVE," React Native RSS, 12-Jan-2023. [Online]. Available: https://reactnative.dev/docs/components-and-apis. [Accessed: 12-Jan-2023].

8. “React Tourtial,” React tutorial. [Online]. Available: https://www.w3schools.com/react/default.asp. [Accessed: 14-Dec-2022].

9. “Platform specific code · REACT NATIVE,” React Native RSS, 12-Jan-2023. [Online]. Available: https://reactnative.dev/docs/platform-specific-code. [Accessed: 27-Apr-2023].

10. “Rest api (introduction),” GeeksforGeeks, 25-Apr-2023. [Online]. Available: https://www.geeksforgeeks.org/rest-api-introduction/. [Accessed: 15-Apr-2023].

11. W. A. S. (c) 2023, “CSS introduction,” Starting Electronics, Electronics for beginners and beyond. [Online]. Available: https://startingelectronics.org/tutorials/arduino/ethernet-shield-web-server-tutorial/CSS-introduction/. [Accessed: 01-Apr-2023].

12. “https://www.mongodb.com/docs/manual/aggregation/,” Aggregation Operations - MongoDB Manual. [Online]. Available: https://www.mongodb.com/docs/manual/aggregation/. [Accessed: 27-Mar-2023].

13. “https://www.mongodb.com/docs/manual/crud/,” MongoDB CRUD Operations - MongoDB Manual. [Online]. Available: https://www.mongodb.com/docs/manual/crud/. [Accessed: 27-Apr-2023].

14. “https://www.mongodb.com/docs/manual/indexes/,” Indexes - MongoDB Manual. [Online]. Available: https://www.mongodb.com/docs/manual/indexes/. [Accessed: 20-Feb-2023].

15. “https://www.mongodb.com/docs/manual/administration/install-community/,” Install MongoDB Community Edition - MongoDB Manual. [Online]. Available: https://www.mongodb.com/docs/manual/administration/install-community/. [Accessed: 27-Apr-2023].

16. “https://www.mongodb.com/docs/atlas/app-services/schemas/types/#std-label-schema-types,” Schema Types - Atlas App Services. [Online]. Available: https://www.mongodb.com/docs/atlas/app-services/schemas/types/#std-label-schema-types. [Accessed: 15-Feb-2023].

17. “JSON schema,” JSON Schema. [Online]. Available: https://json-schema.org/. [Accessed: 13-Feb-2023].

18. “Mongo DB Tutorial,” MongoDB tutorial. [Online]. Available: https://www.w3schools.com/mongodb/index.php. [Accessed: 27-Apr-2023].

- **Chatbot References**

19. “Python Programming,” Python tutorial. [Online]. Available: https://www.w3schools.com/python/default.asp. [Accessed: 20-Oct-2022].

20. “The Complete Guide to preparing for an AI chatbot project,” UMNI, 12-Apr-2022. [Online]. Available: https://umni.bg/en/blog/the-complete-guide-to-preparing-for-an-ai-chatbot-project/. [Accessed: 13-Apr-2023].

21. “How do chatbots work? an architecture of Chatbots,” Rakebots blog, 12-Dec-2019. [Online]. Available: https://rakebots.com/blog/how-do-chatbots-work-an-architecture-of-chatbots/. [Accessed: 12-Dec-2022].

22. “Chat bot with PyTorch - NLP and Deep Learning - Python Tutorial (Part 1),” YouTube, 08-Jun-2020. [Online]. Available: https://www.youtube.com/watch?v=RpWeNzfSUHw. [Accessed: 20-Jan-2023].

23. “Chat bot with PyTorch - NLP and Deep Learning - Python Tutorial (Part 2),” YouTube, 08-Jun-2020. [Online]. Available: https://www.youtube.com/watch?v=8qwowmiXANQ. [Accessed: 20-Jan-2023].

24. “Chat bot with PyTorch - NLP and Deep Learning - Python Tutorial (Part 3),” YouTube, 08-Jun-2020. [Online]. Available: https://www.youtube.com/watch?v=Da-iHgrmHYg. [Accessed: 20-Jan-2023].

25. “Chat bot with PyTorch - NLP and Deep Learning - Python Tutorial (Part 4),” YouTube, 08-Jun-2020. [Online]. Available: https://www.youtube.com/watch?v=k1SzvvFtl4w. [Accessed: 20-Jan-2023].

26. “Build & integrate your own custom chatbot to a website (Python & JavaScript),” YouTube, 23-Sep-2021. [Online]. Available: https://www.youtube.com/watch?v=a37BL0stIuM. [Accessed: 25-Apr-2022].

27. “QuickStart¶,” QuickStart - Flask Documentation (2.2.x). [Online]. Available: https://flask.palletsprojects.com/en/2.2.x/quickstart/#a-minimal-application. [Accessed: 25-Feb-2023].

28. Patrickloeber, “Patrickloeber/PyTorch-chatbot: Simple chatbot implementation with PyTorch.,” GitHub. [Online]. Available: https://github.com/patrickloeber/pytorch-chatbot. [Accessed: 25-Feb-2023].

29. Apeda, “Fruits and Vegetable Data India,” Fresh fruits and vegetables. [Online]. Available: https://apeda.gov.in/apedawebsite/six_head_product/FFV.htm. [Accessed: 23-Oct-2022].