



## **An-Najah National University**

Faculty of Engineering & Information Technology Department of  
Computer Engineering

Graduation Project I  
Software Project  
(GreenSage)



E-commerce application.  
Herbs & Grocery Delivery Application

Done by:  
Alaa Hamayde.  
Haneen Banishamseh

Supervised by:  
Dr.Raed Qadi

Presented in partial fulfillment of the requirements for  
**Bachelor Degree in Computer Engineering**

# Table of content

1. Abstract
2. Introduction
3. Constraint, Standards and earlier work
4. Literature review
5. Methodology
  - 5.1. Tools, Programming Languages, and Technologies.
    - 5.1.1. Mobile Application
    - 5.1.2. Website
    - 5.1.3. Database
    - 5.1.4. IDEs and Code editors
  - 5.2. Architecture
    - 5.2.1. Mobile Application
    - 5.2.2. Website
  - 5.3. Implementation
    - 5.3.1. Mobile Application
      - 5.3.1.1. Dashboard Application.
      - 5.3.1.2. Shopping Dashboard Application.
    - 5.3.2. Website
      - 5.3.2.1. Dashboard Website.
      - 5.3.2.2. Home page.
6. Results and Discussion
  - 6.1. Learnings
  - 6.2. Challenges
7. Conclusion and Recommendation
  - 7.1. Conclusion
  - 7.2. Recommendations
  - 7.3. What I learned
8. References

# Acknowledgment

in the success of this project.

A special acknowledgment goes to my family, the unwavering pillar of support throughout my university journey and beyond. I am immensely grateful to my friends, who stood as a sturdy wall for me to lean on whenever needed. Additionally, I extend my thanks to the computer engineering department at my university for their constant care and support.

I would like to give special appreciation to Dr. Raed Qadi, whose timely assistance and guidance have been invaluable to me.

# 1. Abstract

The landscape of technology has significantly transformed the operations of numerous companies. In the case of many grocery and herbs E-commerce businesses, technology stands out as a pivotal force and a primary marketing tool.

The market for E-commerce apps in the grocery and food supplement industry has witnessed a remarkable surge in recent years, resulting in an increased demand for mobile programming tailored to these businesses. Developing and programming an E-commerce app for groceries not only enhances its market presence but also serves as a powerful online brand promotion tool.

This project empowers online grocery stores to sell their products on sophisticated platforms, providing customers with a comprehensive view of the diverse range of items available in their inventory.

The implementation of dynamic chat. Additionally, the inclusion of a feedback mechanism allows customers to benefit from the experiences and opinions of fellow shoppers.

The notifications feature keeps customers informed about updates and relevant news, while the recommendation feature guides them towards products they may desire or frequently request. For the app development, we plan to use React Native for the front end,. The backend will be powered by NodeJs, and Firebase will handle the notifications.

## 2. Introduction

After the period of the spread diseases, people started to prefer have an advice for health recovers applications and websites while they were in their homes, they need products for that match and help them recover their health.

The effects of the COVID-19 epidemic have made unemployment in Palestine worse. A great deal of difficulty has been faced by many store owners in increasing sales and guaranteeing a steady income. Given these urgent concerns, internet purchasing has become an essential lifeline for many trying to get through the financial hardships caused by the pandemic.

Anyone may now use social media networks to create an online business. This strategy, nevertheless, may be burdensome for both storeowners and patrons. It might be annoying for customers to not be able to see every product in depth and to browse them all at once. Another thing that might be annoying is waiting for a storeowner to reply to your questions about the products and what is the best match one.

This project was started to address these issues by giving users the ease of exploring and choosing the top health-related items via a specialized application.

Initially, the application helps customers to view products from the vendors available on the application, which gives them the advantage of choosing best match product in an easier way, instead of going to the market. In addition, customers can easily compare and view all products, additionally they can see the suggested products that match the health lacks, or the most suitable product for them. Also, it can keep track of all products releases without you having to keep checking. This gives app customers the best shopping experience available in the market.

It also provides an additional feature to make owner (vendor) can easily add/delete/modify products and all changes will be automatically applied to the application database.

In addition to the process of adding, deleting or modifying the vendors personal data. and, the store owner will have a detailed list of the orders received from the customers of the app once they have placed their order. So that when receiving orders from customers.

In addition, there is a process of adding, deleting, or modifying delivery information, so that each delivery person, upon adding it, has a personal account that shows him all orders issued under his name, so that each order shows him the person's information in terms of his name, phone number, and complete order details, the owner can change the status of the delivery (pending, shipped, delivered)

There is also the idea of tracking the customer's orders, so that the person delivering can track customer order status easily. There is a customer communication feature, by using Dynamic Chat to answer customers to all their requests.

In the following chapters I will talk about the limitations that we encountered during the development of this project, the training courses that helped us reach our goal in this work, discuss similar available applications and how this project differs from others, as well as the methodology, techniques and procedures that I followed to complete it, I will discuss the emerging results, and finally I will talk On conclusion, limitations, and future work.

### 3. Constraints, standards and earlier work

I used the NodeJs framework to manage the website application, fetching and inserting data. So I learned how to use it through an online course, as well as the docs, used Mongo DB for databases and JSON format to pass data for some features, I learned by watching videos and searching online.

**Authentication and Authorization:** Examining previous efforts to incorporate best practices for protecting React apps into the implementation of secure authentication and authorization systems in Node.js.

**Deployment Strategies** Knowing how to deploy Node.js apps effectively (e.g., utilizing platforms like Docker).

## 4.Literature review

Nowadays, the majority of people consider access to online shopping platforms to be a necessity rather than a privilege, since e-commerce has become an essential part of everyday life. With about 3 billion people using the internet worldwide, e-commerce has grown dramatically as a result of people's increasing comfort level with making purchases online and the internet's broad use. This is seen by the growing trend of mobile-based transactions and the number of clients who have registered on e-commerce platforms.

I have created an online shopping app that is aimed at a wide range of users, with a specific focus on those who are dealing with health issues, in reaction to this trend. With the goal of improving users' well-being, this cutting-edge program makes product recommendations that are especially pertinent to their health circumstances.

This project is distinct because it places a special emphasis on the health state of the user. In contrast to previous comparable endeavors, this app offers a customized and health-conscious shopping experience by suggesting goods that are directly relevant to the user's health.

This program, which is only accessible in Palestine, helps company owners deliver their orders to clients in a more polished and orderly manner while expediting the process of buying products.

The desire to support people in difficult times—especially those facing health concerns—is what motivated the development of this application. The app acts as a guide, suggesting herbal remedies and natural goods that aid in healing and offer illness prevention. The goal of this project is to improve people's wellbeing by providing personalized recommendations for a healthier lifestyle.

## 5. Methodology

### 5.1 Tools, Programming Languages, and Technologies.

#### 5.1.1. Website.

There are many frameworks and languages that were used to build the mobile application and the website.

- Frameworks

We chose to use NodeJs which is an open-source software development kit which enables smooth development, and we choose, and React for the front end

- Programming Languages

The programming language we use is JavaScript because we use NodeJs as the framework.

#### 5.1.2. Mobile application. Frameworks:

We build the application on react native, is an open-source framework developed by Facebook for building mobile applications using JavaScript and React. It allows developers to create cross-platform apps with a single codebase.

#### Editor:

Expo (React Native Wrapper): Expo is a framework built around React Native that simplifies the development process and provides additional features.

## 5.1.3: Database:

- **Database for website and application**

Mongo DB is known for its ability to handle unstructured or semi-structured data and is particularly well suited for applications with large amounts of diverse data, such as those in web development and big data scenarios. It provides features like horizontal scaling, automatic sharding, and flexible schema design, making it a versatile choice for developers working on modern, dynamic applications.

### Full Database for website:

Collection Name	Documents	Logical Data Size	Avg Document Size	Storage Size	Indexes	Index Size	Avg Index Size
<a href="#">carts</a>	2	354B	177B	36KB	1	36KB	36KB
<a href="#">categories</a>	4	1.56KB	400B	36KB	1	36KB	36KB
<a href="#">chats</a>	13	1.74KB	138B	36KB	1	36KB	36KB
<a href="#">favorites</a>	2	174B	87B	36KB	1	36KB	36KB
<a href="#">notifications</a>	4	288B	72B	36KB	1	36KB	36KB
<a href="#">orders</a>	5	2.54KB	520B	36KB	1	36KB	36KB
<a href="#">products</a>	30	13.75KB	470B	44KB	1	36KB	36KB
<a href="#">ratings</a>	5	754B	151B	36KB	1	36KB	36KB
<a href="#">users</a>	7	4.12KB	604B	36KB	2	72KB	36KB

Figure 1: Database

```
Find Indexes Schema Anti-Patterns Aggregation Search Indexes
INSERT DOCUMENT
Filter Type a query: { field: 'value' }
Reset Apply Options
{
  "_id": ObjectId('65b6b0663868a8b7adfbfe57'),
  "totalPrice": 15,
  "userId": ObjectId('65b6b0663868a8b7adfbfe53'),
  "totalItems": 1,
  "cartItems": Array (1)
  0: Object
    price: 5
    quantity: 3
    productId: ObjectId('65b68fb43868a8b7adfbf28d')
    itemTotalPrice: 15
    ownerName: "Vita Greens"
    _id: ObjectId('65b7b0e2a5052db3b8f76b84')
    __v: 0
  }
  "_id": ObjectId('65b6b18c3868a8b7adfc0009')
  "totalPrice": 0
}
```

System Status: All Good

Figure 2: carts

GreenSageDB

carts

categories

chats

favorites

notifications

orders

products

ratings

users

Find Indexes Schema Anti-Patterns 0 Aggregation Search Indexes

INSERT DOCUMENT

Filter [🔗](#) Type a query: { field: 'value' }

Reset Apply Options ▶

```
_id: ObjectId('65b68766006cc3a5d508fdf2')
name: "Vegetables"
description: "Explore our vibrant selection of nutrient-packed vegetables. From leaf..."
products: Array (9)
  0: ObjectId('65b68fb43868a8b7adfbf28d')
  1: ObjectId('65b6960c3868a8b7adfbf372')
  2: ObjectId('65b696b83868a8b7adfbf392')
  3: ObjectId('65b6977f3868a8b7adfbf4d0')
  4: ObjectId('65b698343868a8b7adfbf4f0')
  5: ObjectId('65b698ea3868a8b7adfbf510')
  6: ObjectId('65b69a5d3868a8b7adfbf5e0')
  7: ObjectId('65b7b9c8a9c80ce7db41c201')
  8: ObjectId('65b7c660a9c80ce7db41c8c3')
__v: 0
```

System Status: All Good

Figure3: categories

GreenSageDB

carts

categories

chats

favorites

notifications

orders

products

ratings

users

Find Indexes Schema Anti-Patterns 0 Aggregation Search Indexes

INSERT DOCUMENT

Filter [🔗](#) Type a query: { field: 'value' }

Reset Apply Options ▶

QUERY RESULTS: 1-13 OF 13

```
_id: ObjectId('65b6c65952e8f5796de53ebe')
user: ObjectId('65b6c5ed3868a8b7adfc006f')
to: "haneen@gmail.com"
from: "Green"
message: "hi haneen"
timestamp: 2024-01-28T21:25:45.985+00:00
__v: 0
```

```
_id: ObjectId('65b6c67352e8f5796de53edc')
user: ObjectId('65b6b18c3868a8b7adfc0005')
to: "greenSage@support.com"
from: "Haneen"
message: "hi "
timestamp: 2024-01-28T21:26:11.298+00:00
```

Figure4: chat

GreenSageDB

- cartes
- categories
- chats
- favorites**
- notifications
- orders
- products
- ratings
- users

Find Indexes Schema Anti-Patterns Aggregation Search Indexes

INSERT DOCUMENT

Filter Type a query: { field: 'value' } Reset Apply Options

QUERY RESULTS: 1-2 OF 2

```

_id: ObjectId('65b6b0663868a8b7adfbfe59')
userId: ObjectId('65b6b0663868a8b7adfbfe53')
items: Array (1)
  0: Object
    productId: ObjectId('65b68fb43868a8b7adfbf28d')
    _id: ObjectId('65b7ee8da14c3e91f3cad5ed')
    __v: 5

```

```

_id: ObjectId('65b6b18c3868a8b7adfc000b')
userId: ObjectId('65b6b18c3868a8b7adfc0005')
items: Array (empty)
__v: 0

```

Figure5: favorites

GreenSageDB

- cartes
- categories
- chats
- favorites
- notifications**
- orders
- products
- ratings
- users

Find Indexes Schema Anti-Patterns Aggregation Search Indexes

INSERT DOCUMENT

Filter Type a query: { field: 'value' } Reset Apply Options

QUERY RESULTS: 1-4 OF 4

```

_id: ObjectId('65b7b9d2a9c80ce7db41c210')
userId: ObjectId('65b6b0663868a8b7adfbfe53')
product: ObjectId('65b7b9c8a9c80ce7db41c201')
__v: 0

```

```

_id: ObjectId('65b7ba95a9c80ce7db41c47c')
userId: ObjectId('65b6b18c3868a8b7adfc0005')
product: ObjectId('65b7b9c8a9c80ce7db41c201')
__v: 0

```

```

id: ObjectId('65b7c660a9c80ce7db41c8cd')

```

System Status: All Good

Figure6: notifications

GreenSageDB

- carts
- categories
- chats
- favorites
- notifications
- orders**
- products
- ratings
- users

Find Indexes Schema Anti-Patterns Aggregation Search Indexes

INSERT DOCUMENT

Filter Type a query: { field: 'value' } Reset Apply Options

QUERY RESULTS: 1-5 OF 5

```

_id: ObjectId('65b6ca8b3868a8b7adfc0459')
userId: ObjectId('65b6b0663868a8b7adfbfe53')
date: 2024-01-28T21:43:39.083+00:00
deliveryFee: 10
shipmentStatus: "pending"
userAddress: Object
  userName: "Alaa Hamayde"
  totalPrice: 15
  cartItems: Array (1)
  __v: 0

```

```

_id: ObjectId('65b7a443ac8b59bbf44cb191')
userId: ObjectId('65b6b0663868a8b7adfbfe53')

```

Figure7: orders

GreenSageDB

- carts
- categories
- chats
- favorites
- notifications
- orders
- products**
- ratings
- users

Find Indexes Schema Anti-Patterns Aggregation Search Indexes

INSERT DOCUMENT

Filter Type a query: { field: 'value' } Reset Apply Options

QUERY RESULTS: 1-20 OF MANY

```

_id: ObjectId('65b68fb43868a8b7adfbf28d')
name: "kale"
description: "Supporting the immune system, regulating blood pressure, full of vitam..."
price: 5
availableInStock: 40
imageUrl: "https://www.veggycation.com.au/siteassets/veggycationvegetable/kale.jp..."
categoryId: ObjectId('65b68766006cc3a5d508fd2')
ratings: Array (1)
cartItems: Array (1)
favorites: Array (1)

```

PREVIOUS 1-20 of many results NEXT

Figure8: products

GreenSageDB

- carts
- categories
- chats
- favorites
- notifications
- orders
- products
- ratings**
- users

Find Indexes Schema Anti-Patterns Aggregation Search Indexes

INSERT DOCUMENT

Filter Type a query: { field: 'value' } Reset Apply Options

QUERY RESULTS: 1-5 OF 5

```

{
  "_id": ObjectId('65b6d9d852e8f5796de54349'),
  "title": "Carrot Crunch",
  "rating": 4,
  "description": "Great!!",
  "userId": ObjectId('65b6b0663868a8b7adfbfe53'),
  "productId": ObjectId('65b696b83868a8b7adfbf392'),
  "__v": 0
}

{
  "_id": ObjectId('65b6da5452e8f5796de5453c'),
  "title": "Carrot Crunch",
  "rating": 5,
  "description": "Nice!",
  "userId": ObjectId('65b6b18c3868a8b7adfc0005')
}

```

Figure9: ratings

GreenSageDB

- carts
- categories
- chats
- favorites
- notifications
- orders
- products
- ratings
- users**

Find Indexes Schema Anti-Patterns Aggregation Search Indexes

INSERT DOCUMENT

Filter Type a query: { field: 'value' } Reset Apply Options

QUERY RESULTS: 1-7 OF 7

```

{
  "_id": ObjectId('65b68e7f3868a8b7adfbf263'),
  "firstName": "Vita",
  "lastName": "Greens",
  "email": "vita@gmail.com",
  "mobile": "0599123456",
  "password": "123456789",
  "imageUrl": "https://media.hanglungmalls.com/custom-import/cmsUpload/6a9b2cd0-0f0d-...",
  "role": "owner",
  "addresses": Array (1),
  "ratings": Array (empty),
  "orders": Array (empty),
  "description": "Embrace vitality with Vita Greens - your one-stop destination for farm...",
  "__v": 0
}

```

Figure10: users

## 5.1.4: IDEs and Code Editors

Visual Studio Code (VSCode): developed by Microsoft, is a lightweight, free, and open-source code editor. It supports a wide range of languages, including JavaScript, Typescript, HTML, CSS, and more.

Key Features:

- IntelliSense for code completion.
- Built-in Git integration.
- Extensions marketplace for additional features.
- Debugging support.
- Integrated terminal.

## 5.2 Architecture (Website)

### 5.2.1 Frontend Architecture (React):

React is based on a component-based architecture. Break down your UI into reusable components, each responsible for a specific part of the UI. Use functional components and, if needed, React Hooks for state management.

#### 5.2.1.1: Component-Based Architecture:

React follows a component-based architecture.

The UI is divided into small, reusable components, each responsible for a specific part of the user interface.

This promotes modularity and reusability.

#### 5.2.1.2: Virtual DOM:

React uses a virtual DOM to efficiently update the actual DOM. When data changes, React updates the virtual DOM first and then calculates the most efficient way to update the real DOM. This minimizes the number of updates and enhances performance.

#### 5.2.1.3: State Management:

React components can have local state, managed using the useState hook. Additionally, global state management can be handled with tools like React Context or state management libraries such as Redux for more complex applications.

#### 5.2.1.4 Routing:

Client-side routing is often implemented using React Router. It allows for navigation within the application without triggering a full page reload, providing a smoother user experience.

## **5.2.2. Backend Architecture (Node.js and Mongo DB):**

### **3.2.2.1: Server Architecture (Node.js):**

Node.js is used on the server side to handle HTTP requests and responses. The application may be organized using the Express.js framework, a popular choice for building web applications with Node.js.

### **3.2.2.2: RESTful API Design:**

The backend exposes a RESTful API that the frontend communicates with. This API defines endpoints for different resources (e.g., users, products) and supports standard HTTP methods (GET, POST, PUT, DELETE) for CRUD operations.

### **3.2.2.3: Middleware:**

Middleware functions are used in Express.js to perform tasks such as authentication, logging, error handling, and more. Middleware can be organized in a pipeline to process requests in a sequential manner.

### **3.2.2.4 : Database (MongoDB):**

MongoDB is used as the database for storing and retrieving data. The application may use Mongoose, an ODM (Object Data Modeling) library for MongoDB and Node.js, to interact with the database and define data models.

**4.2.2.5: Authentication and Authorization:** User authentication is implemented using techniques like JWT (JSON Web Tokens). Middleware is used to secure routes and ensure that only authenticated users have access to certain resources.

### **4.2.2.6: Scalability:**

Node.js is known for its scalability due to its non-blocking, event-driven architecture. The application can be scaled horizontally by adding more Node.js instances behind a load balancer.

**4.2.2.6: WebSocket :** In scenarios where real-time communication is essential, WebSockets can be used to establish a persistent, bidirectional communication channel between the frontend and backend. This architecture provides a scalable, maintainable, and modular foundation for developing web applications.

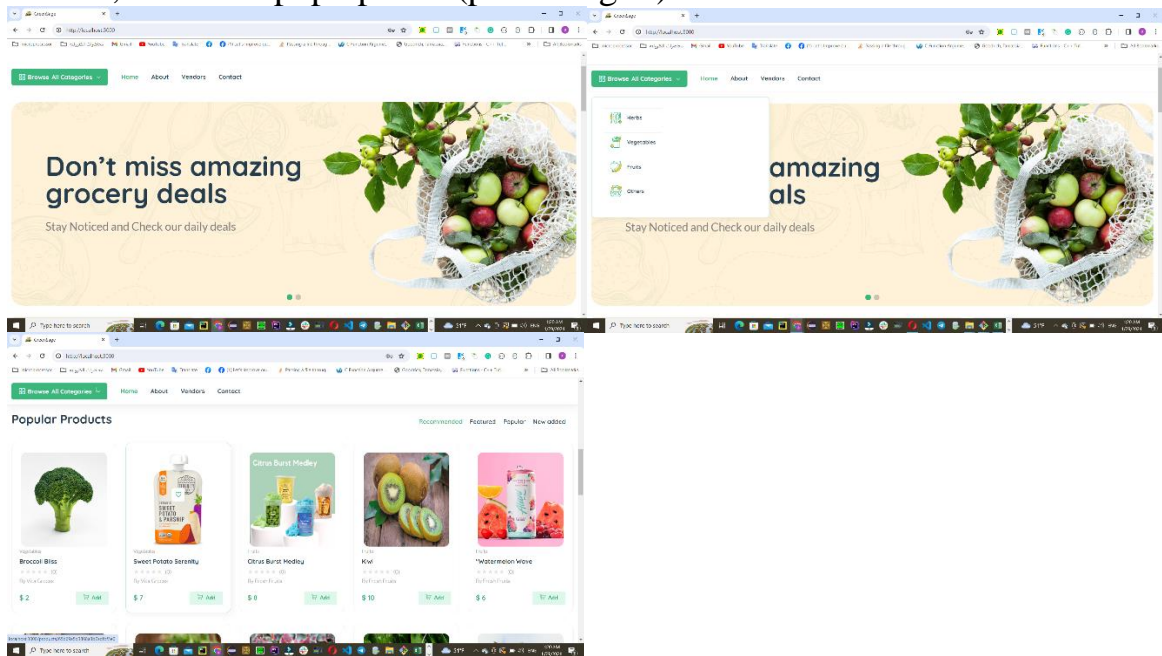
**4.2.2.6: API:** The frontend communicates with the backend through HTTP requests using RESTful API endpoints. This standardizes communication and allows for clear separation of concerns between the client and server. AJAX/Fetch: The frontend utilizes AJAX (Asynchronous JavaScript and XML) or the modern Fetch API to make asynchronous requests to the backend. This enables the application to retrieve and send data without requiring a full page reload.

## 5.3 Implementation

### 5.3.1: Website.

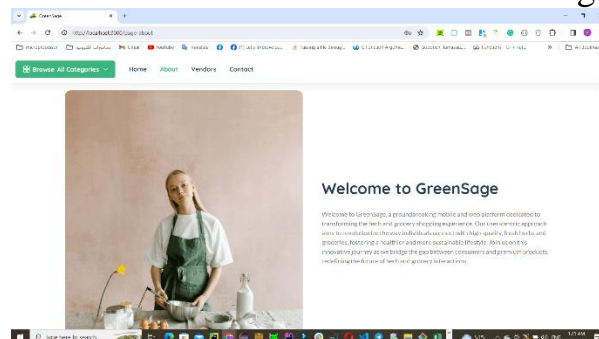
- **Home Page.**

This page is the landing page, you can see all products but you cant add any one unless you log in to the system, if you try to add any product without log in, there is a pop up said (please log in).



- **About.**

This page define the website and what GreenSage provide.



- **Register**

In this page you can register as a customer or vendor, if you log in as a customer or vendor, the customer register with his information and the health status nad what is the health lacks they have, but when you try to

Register as a vendor the list for health status disabled, because the vendor just for add products or edit them, and change the status of them.

Already have an account? [Log in instead](#)

First Name, Last Name

vita@gmail.com

Mobile Number

Profile Image URL

\*\*\*\*\*

Confirm password

Please Enter your address info

Street, postalCode

State, City

Please Check the Preferred Keywords

- vitasID
- vitas
- vitas@E2
- vitasum
- vitas@E3
- vitas@E
- vitas@C
- vitas
- vitas@um
- vitas

Security code\* 4074

I am a customer

I am a vendor

- **Log in**

You can login here in this page

GreenSage want a success

Search

Account

Browse All Categories

Home About Vendors Contact

Home > Pages > Login & Register

**Login**

Don't have an account? [Create here](#)

vita@gmail.com

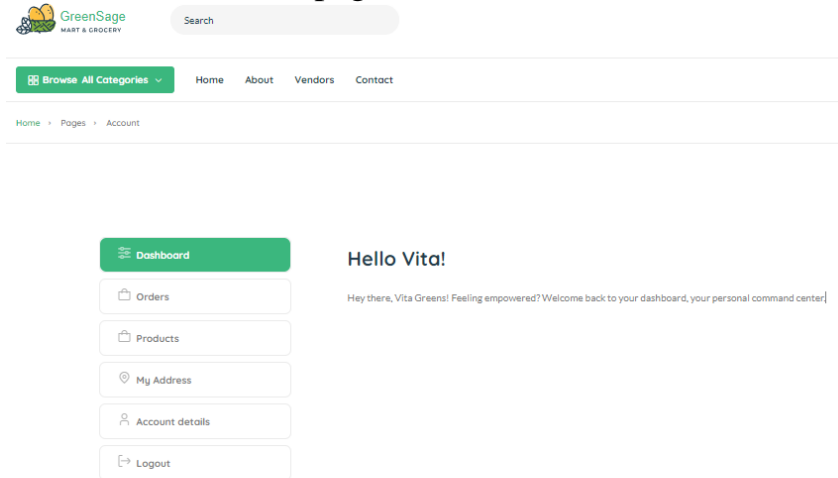
\*\*\*\*\*

Security code... 0278

Log in

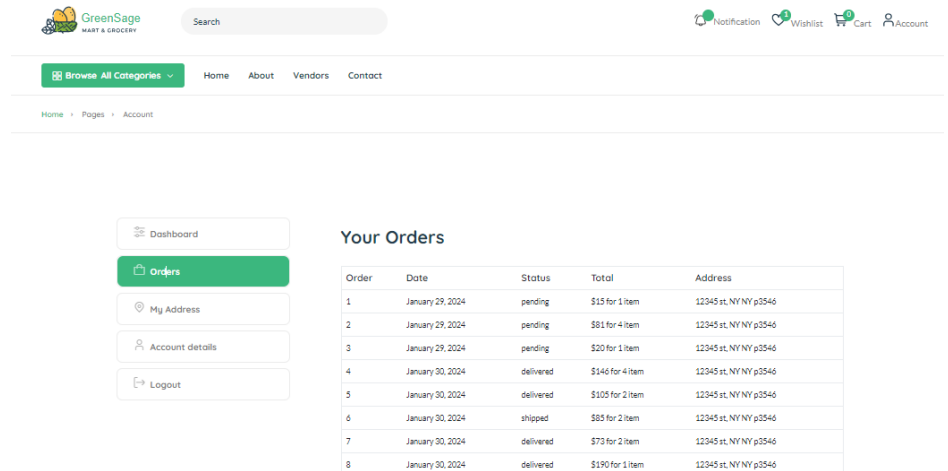
- **Welcome Page (Dashboard)**

This is the welcome page for the user.



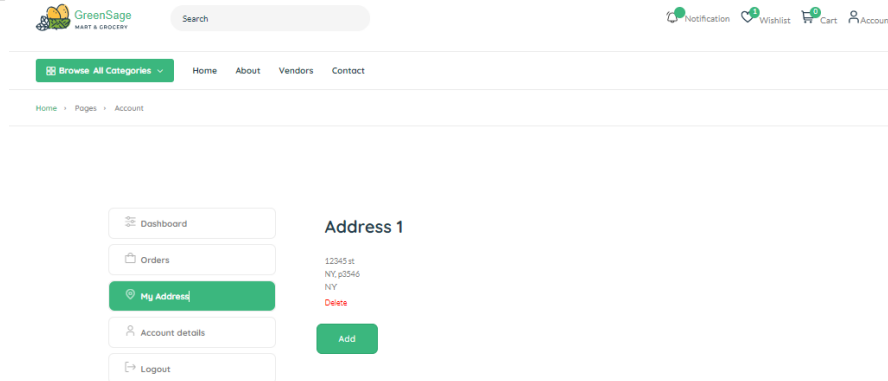
- **Orders**

This page contains all orders details that the customer bought it and pay for it, the customer can follow or monitor the status of the product (Pending, Shipped, Delivered).



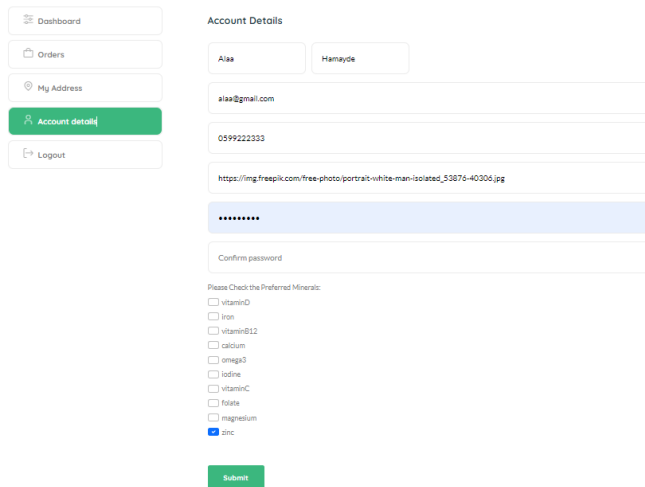
- **My address**

In this page you can delete or add address food you either customer vendor.



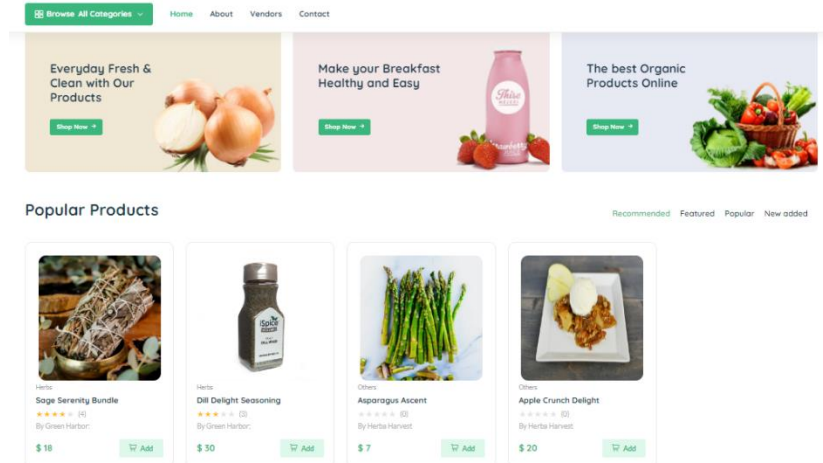
- **Account Details.**

In this page you can edit you information and health status.



- **Popular Products.**

In the home page for a customer there is a section named popular products it contains the products that match the health status for a customer that contain the supplements lacks for a him.

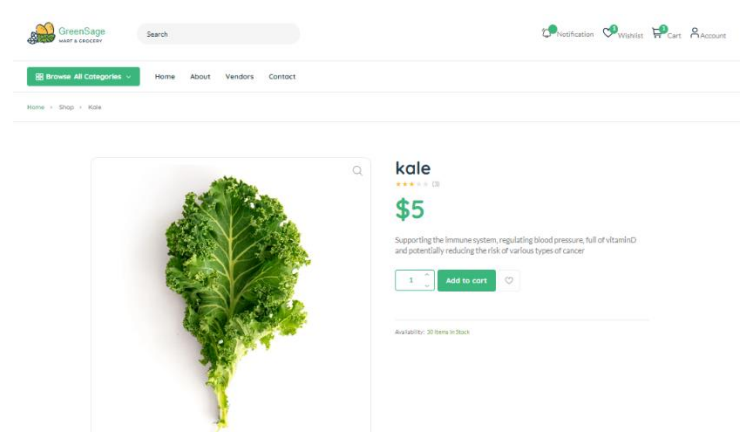
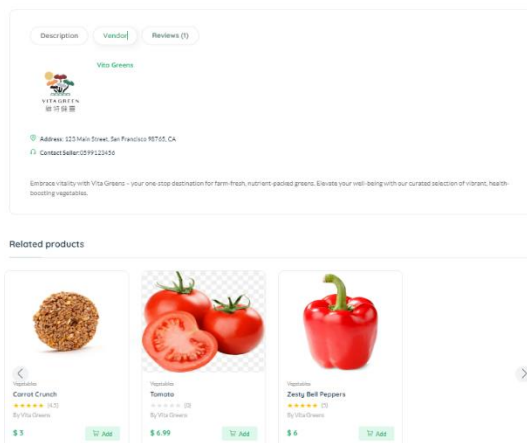


- **Product details.**

This page appeared when I click on the product, it contains the average rating and description, you can add it to cart as quantity you need and you can add it to the wish list, also it contains the availability quantity in the stock.

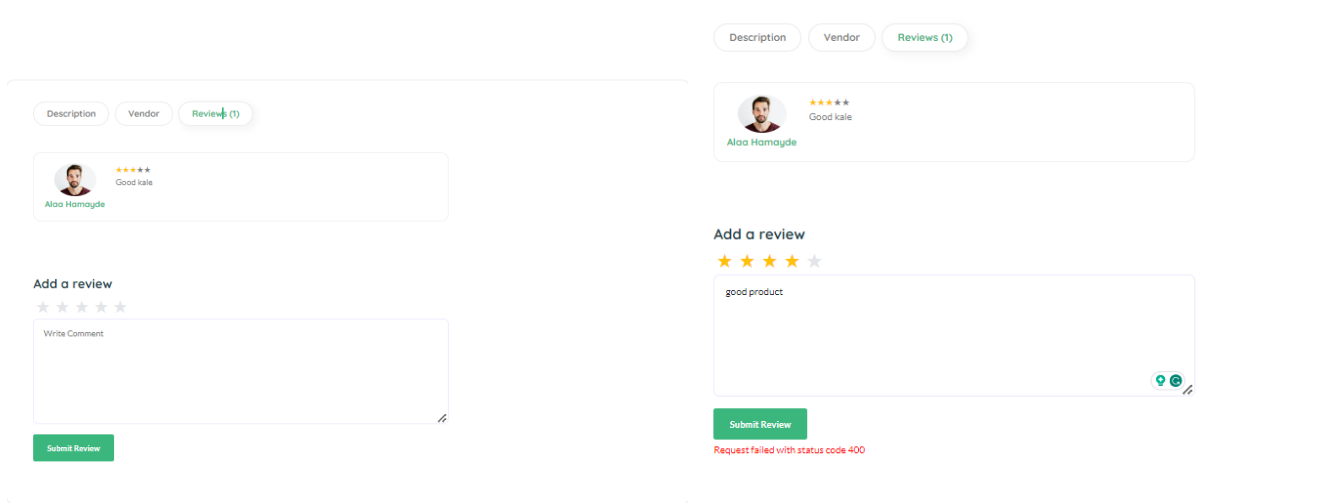
The below section of the same page it contains the description also, Vendor that supply me the product, and the review section.

In addition, it contains the Related products fetched based on the products that have the same category and the same supplement which matches the clicked product.



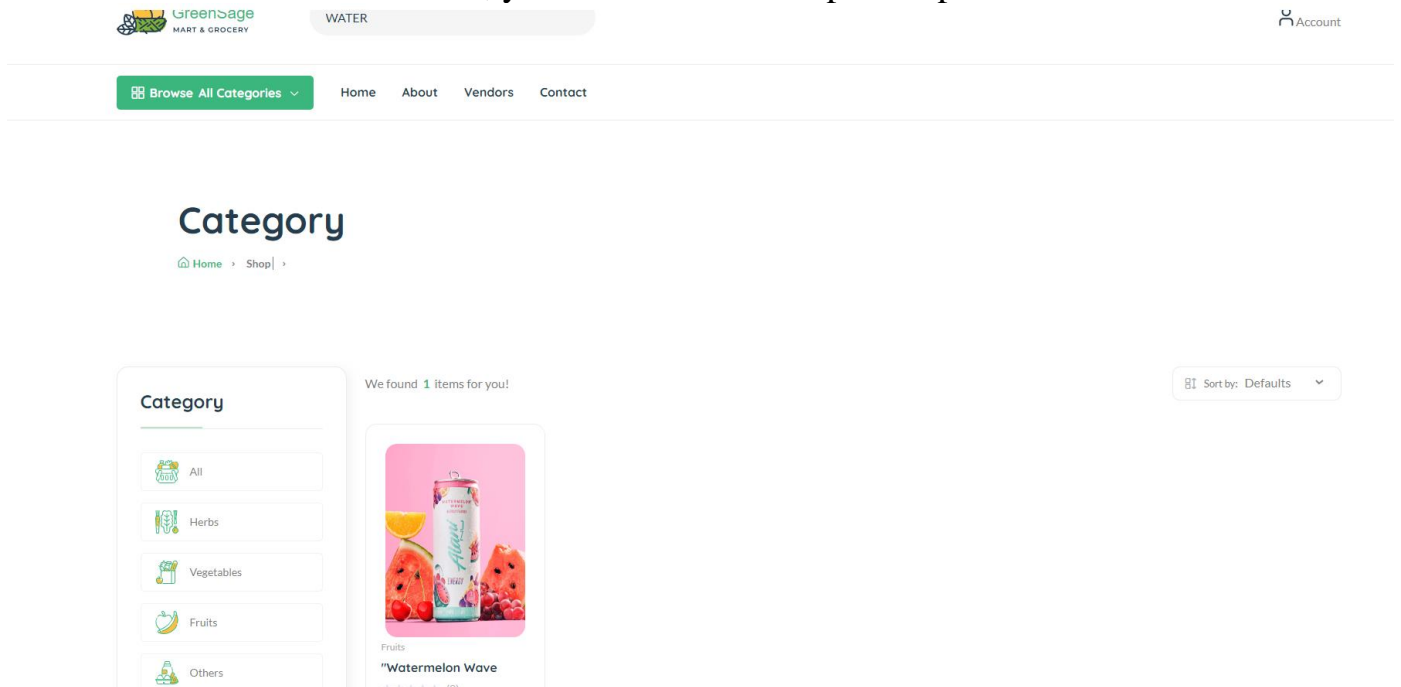
- **Review**

In this section you can review the product and rate it, then leave a comment, but you can't review the same product more than one time.















- **Search.**

In the search section, you can search for a specific product.



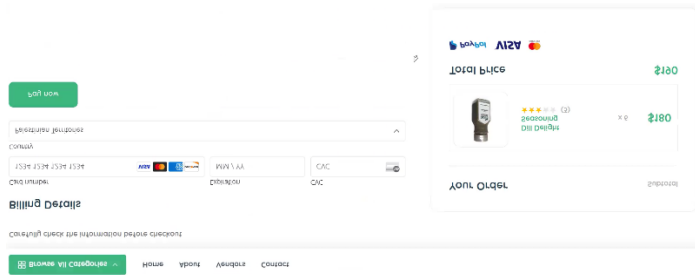
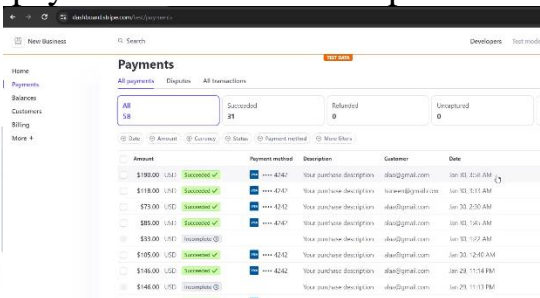
- **Top Rated**

In this section, any product on the system rated more than 3.5 it listed on the top rated.

Top Selling	Trending Products	Recently added	Top Rated
 <p><b>kale</b> ★★★★★ (3) \$ 5</p>	 <p><b>kale</b> ★★★★★ (3) \$ 5</p>	 <p><b>kale</b> ★★★★★ (3) \$ 5</p>	 <p><b>Carrot Crunch</b> ★★★★★ (4.5) \$ 3</p>
 <p><b>Carrot Crunch</b> ★★★★★ (4.5) \$ 3</p>	 <p><b>Carrot Crunch</b> ★★★★★ (4.5) \$ 3</p>	 <p><b>Spinach Boost</b> ★★★★★ (0) \$ 5</p>	 <p><b>Zesty Bell Peppers</b> ★★★★★ (5) \$ 6</p>
 <p><b>Zesty Bell Peppers</b> ★★★★★ (5) \$ 6</p>	 <p><b>Broccoli Bliss</b> ★★★★★ (0) \$ 2</p>	 <p><b>Broccoli Bliss</b> ★★★★★ (0) \$ 2</p>	 <p><b>Cucumber Crisp</b> ★★★★★ (4) \$ 1</p>

- **Payment(Stripe)**

In this page after click on proceed to checkout, it requires to add the visa card number, we use the developer test number provided from stripe to developers 4242 4242 4242 4242 then click on pay now to confirm the operation.

Amount	Status	Payment method	Description	Customer	Date
\$180.00	Successful	4242	Your purchase description	ibad@gmail.com	Jan 03, 12:01 AM
\$118.00	Successful	4242	Your purchase description	ibad@gmail.com	Jan 03, 12:01 AM
\$75.00	Successful	4242	Your purchase description	ibad@gmail.com	Jan 03, 2:00 AM
\$85.00	Successful	4242	Your purchase description	ibad@gmail.com	Jan 03, 1:01 AM
\$33.00	Successful	4242	Your purchase description	ibad@gmail.com	Jan 03, 1:01 AM
\$105.00	Successful	4242	Your purchase description	ibad@gmail.com	Jan 03, 12:40 AM
\$180.00	Successful	4242	Your purchase description	ibad@gmail.com	Jan 02, 11:14 PM
\$140.00	Successful	4242	Your purchase description	ibad@gmail.com	Jan 29, 11:13 PM

- **Invoice**

This page show all products that added to cart with the total fees named grand total that contains the products fees with the delivery fees, in addition in contains the customer details and the number of order, date of order and the vendor(supplier) for each product, after the invoice appeared the product counts down from the stock availability

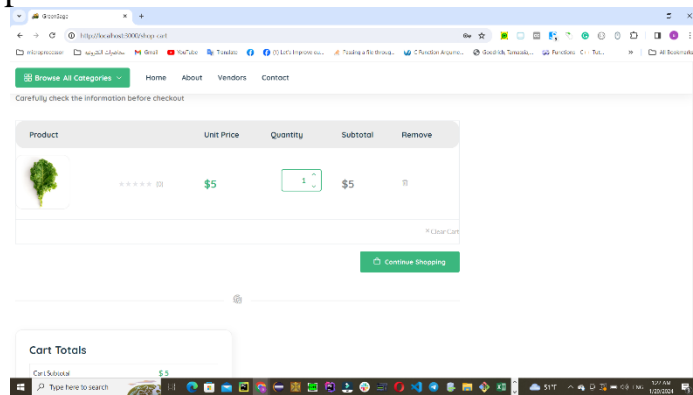
<b>User Details</b> Alaa Hamayde alaa@gmail.com 0599222333		<b>Address</b> 12345 st NY,p3546,NY	
<b>Status:</b> pending		<b>Payment Method</b> Bank Card	

Item name	Unit Price	Quantity	Amount
Dill Delight Seasoning Green Harbor:	\$30	6	\$180
		SubTotal	\$180
		Delivery Fee	\$10
		Grand Total	\$190

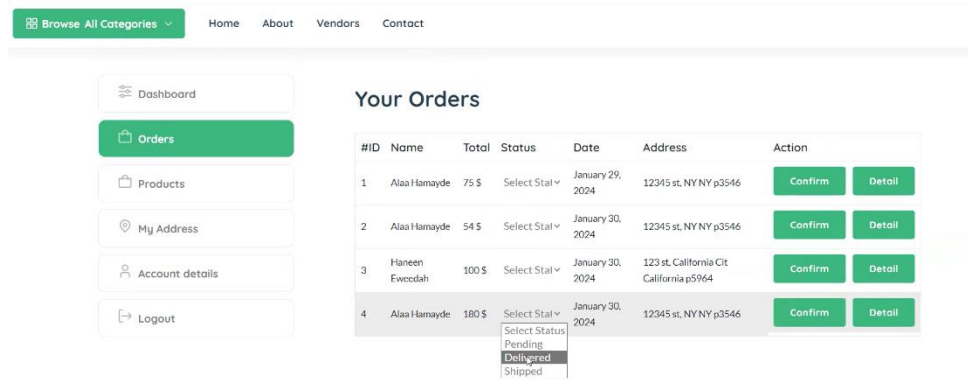
- **Cart.**

This cart contains the products which customers added to their cart they can add the quantity or delete an item or clear the whole cart ,in addition you can continue shopping or proceed to checkout.



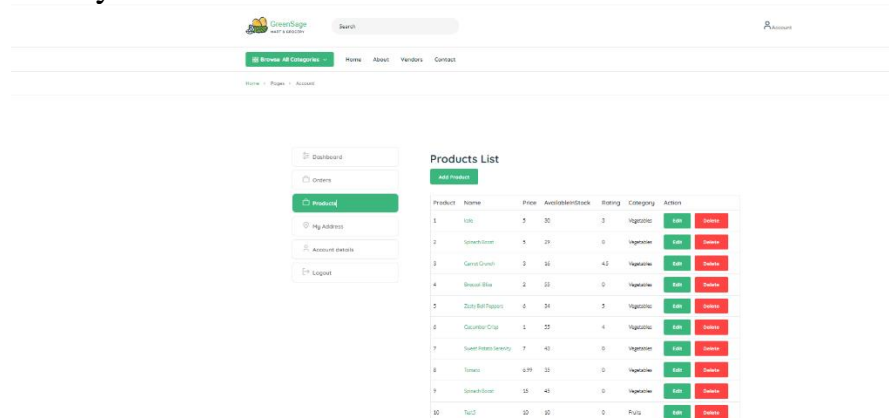
- **Change Status.**

In the orders section for the vendor account, the vendor can update the status of the order for specific customer, the order page shows all orders ordered from him.



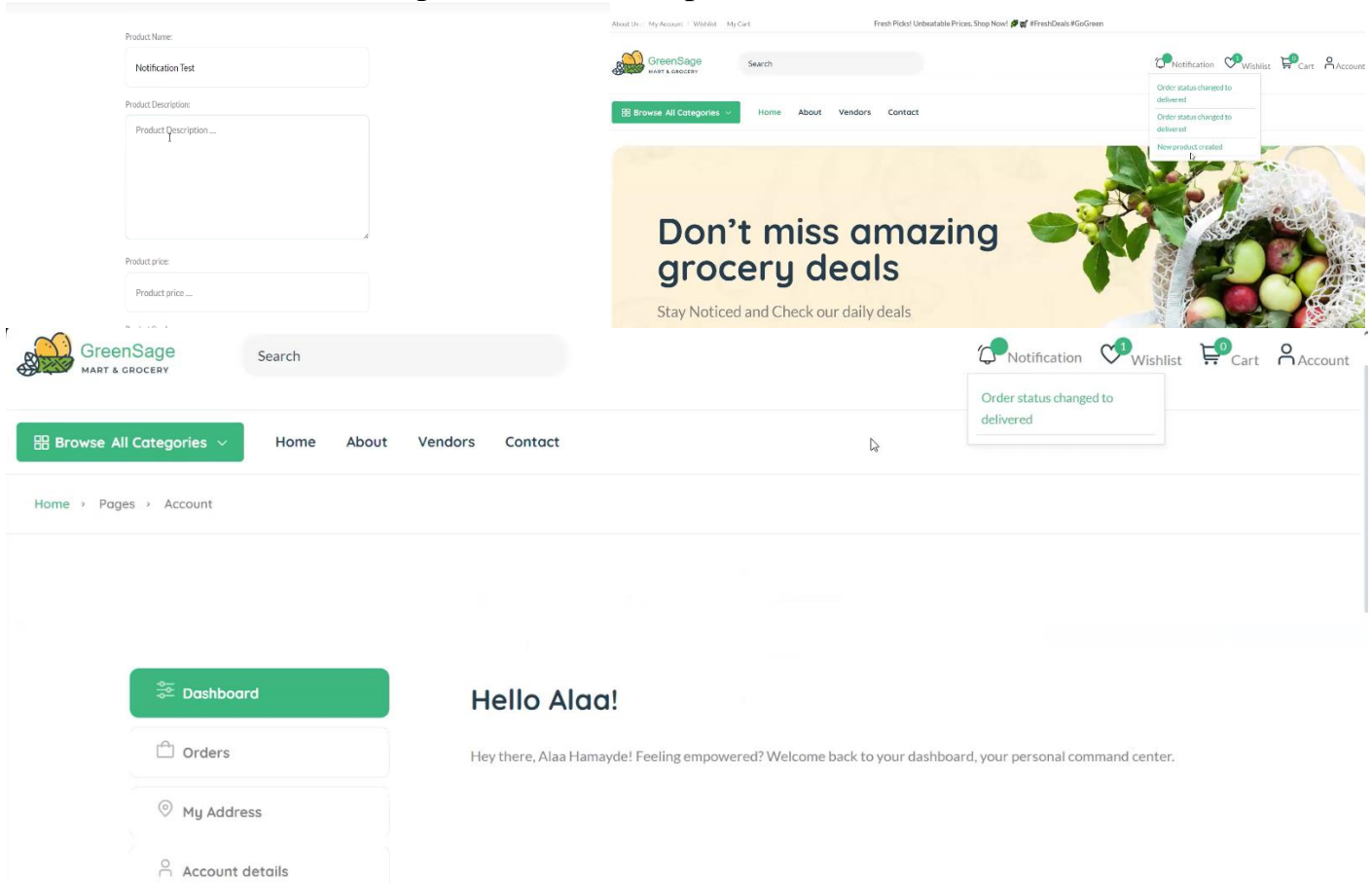
- **Product**

In the vendor account, it also contains the products section that contains all products added by the vendor and he can see all vendors' products, but he cannot see any notifications.



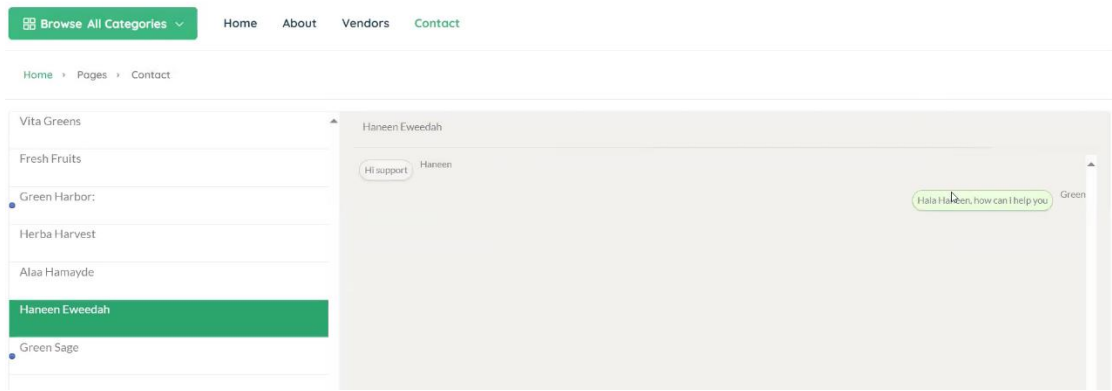
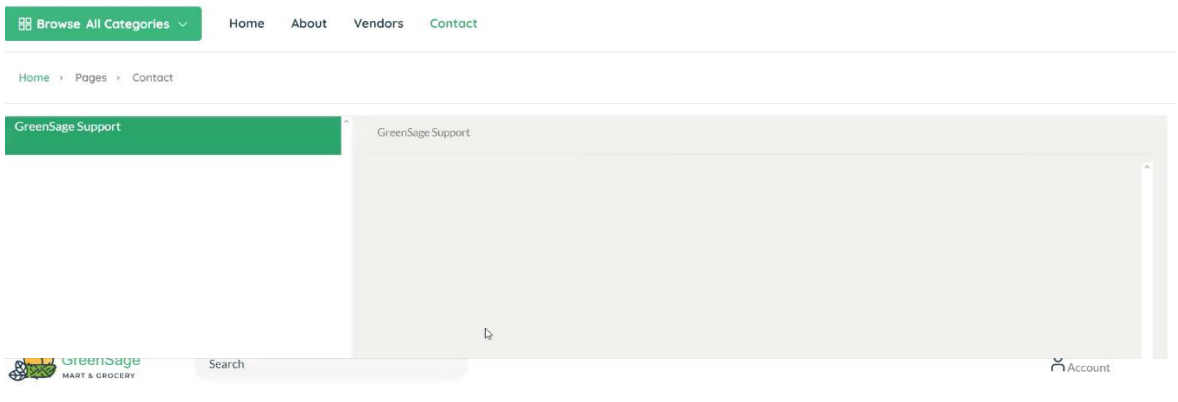
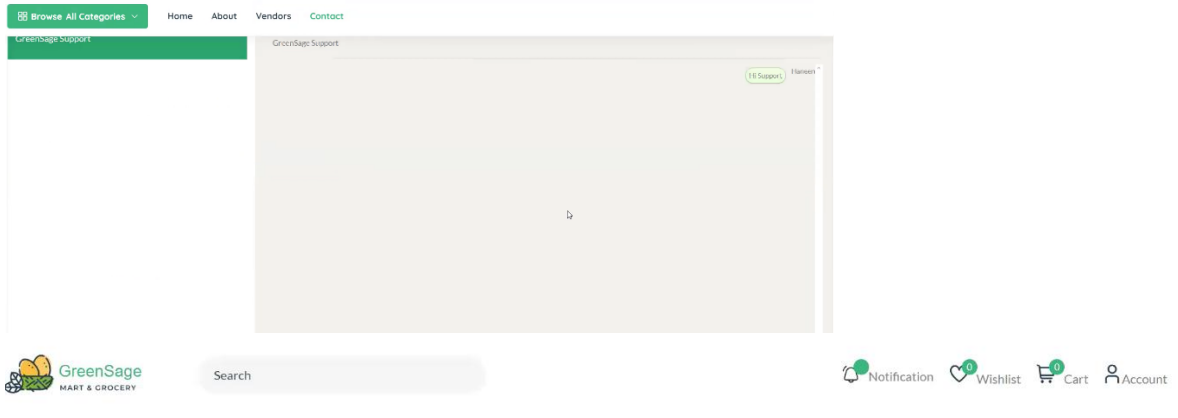
- **Notifications.**

This section to notify the customers if a new product added or the status of the order for specific customer updated.



- **Contact**

In this page, the customer can contact the GreenSage support if he have any problems or queries.



## 5.3.2: Mobile Application.

- **Home screen**

This screen consists of products and food supplements Welcome in GreenSage App.

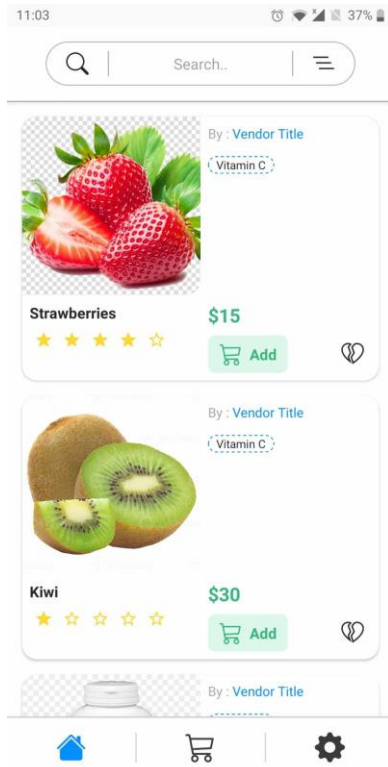
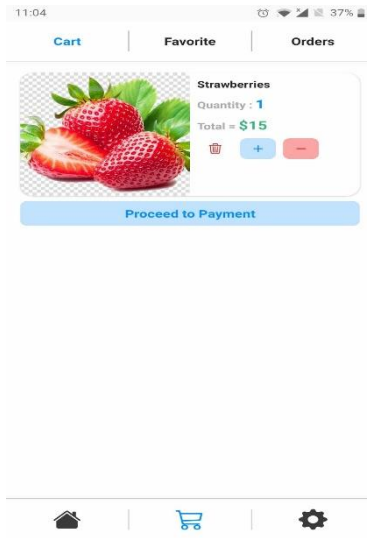


Figure home scre

- **Cart**

Show the items that have been added to the cart to be bought



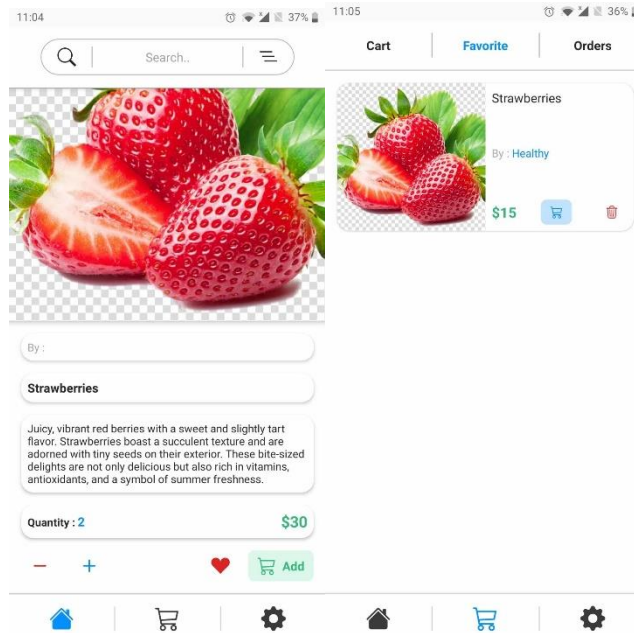
- **Product details.**

Show the product details and what contains of food supplements.



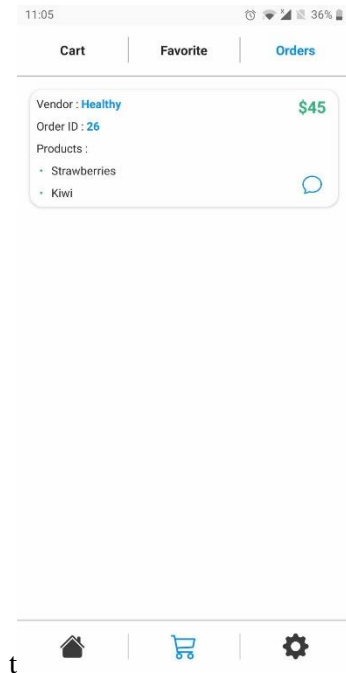
- **Favorite.**

It contains the products that have been added to favorite by the customer.



- **Orders.**

It contains the orders information about the brought products.



- **Login Screen.**

This form for the login information for the user by filling his details.

The image shows a mobile application login screen. At the top, the status bar displays the time 11:05, signal strength, Wi-Fi, and 36% battery. The form consists of several input fields: 'Name' with a placeholder 'Enter Name ..', 'Phone Number' with the value '0594114601', 'City' with the value 'Jenin', and 'Address' with a placeholder 'Street or location'. Below these is a 'Deficiency in:' section with a search bar containing 'Search..' and a selected item 'Vitamin C' highlighted with a dashed blue border. A blue 'Save' button is positioned below the search bar. At the bottom, a navigation bar contains three icons: a home icon, a shopping cart icon, and a settings gear icon.

Figure: Login Screen

## 6 Results and Discussion

During the process of creating my supermarket app, I have developed a strong platform that benefits grocery shop owners and customers alike. The goal of this application is to improve grocery store management's operational efficiency and simplify the shopping experience for consumers.

### **Customer-focused features:**

- **Grocery shopping made easy with intuition**

The grocery item selection and browsing experience is made easier for customers by the app's user-friendly and straightforward layout. As a result, customers are more equipped to make decisions and design unique purchasing experiences.

- **Effective Order Monitoring:**

Within the app, users may effortlessly monitor the progress of their orders. Customers may be updated about the status of their grocery delivery and order transparency is ensured by this real-time order monitoring tool.

- **Client Interaction:**

The grocery business and its customers may communicate more effectively thanks to the app. Store owners may increase consumer engagement and loyalty by notifying customers about promotions, new products, or exclusive deals.

## 7 Learning

Acquiring new skills is a journey that demands patience, dedication, and extensive research. Throughout this project, I embarked on a significant learning experience, delving into various techniques that were entirely unfamiliar to me. The decision to embrace and master Node.js, Express.js, React, and React Native for both client and server-side development proved to be highly rewarding. Learning from scratch required a commitment to understanding these technologies, and the vast wealth of educational content available on the internet played a pivotal role. The abundance of tutorials, guides, and online courses, coupled with the extensive documentation for each technology, provided invaluable resources. This wealth of information not only facilitated a deeper understanding but also enabled me to plan, develop, and implement the project with greater ease and confidence. In essence, the learning process was a dynamic exploration, and the project served as a practical canvas for applying newfound knowledge. Embracing Node.js and React technologies has not only expanded my skill set but has also opened up possibilities for more efficient and innovative development in the ever-evolving landscape of web and mobile applications.

## 7.1 Challenges.

I faced a problem while building the live chat communication, where I build it using websocket, which is different from the RESTfull API's

Where it uses websocket protocol rather than HTTP protocols.

I surpassed these problem by learning websocket and sought guidance from senior developers about it, and eventually I was able to implement it successfully

## 8 Conclusions and Recommendation.

### 8.1 Conclusion.

I managed to create a food supplements website which helps users to pick up the suitable products that provide them with the nutritional supplements in regard to the health status provided in the account info.

Recommendations

In order to improve the application we can add an advanced recommendation system using AI.

### 8.2 What we learned

- NodeJS(backend) ,Express(backend)React (frontend)and ReactNative(mobile app) languages to implement the project.
- NoSQL Database MongoDB.

## 9 References

<https://nodejs.org/docs/latest/api/>

<https://expressjs.com/>

<https://legacy.reactjs.org/docs/getting-started.html>

<https://reactnative.dev/docs/getting-started>

<https://www.mongodb.com/docs/>