

NAAN MUDHALVAN REPORT

AVOCADO GROCERY WEBAPP

SREE SASTHA INSTITUTE OF ENGINEERING AND TECHNOLOGY

Chembarambakkam

SWETHA T

AI & DS (FINAL YEAR)

212421243035

INDEX PAGE

INTRODUCTION

FEATURES

TECHNOLOGIES USED

PROJECT STRUCTURE

HOW IT WORKS

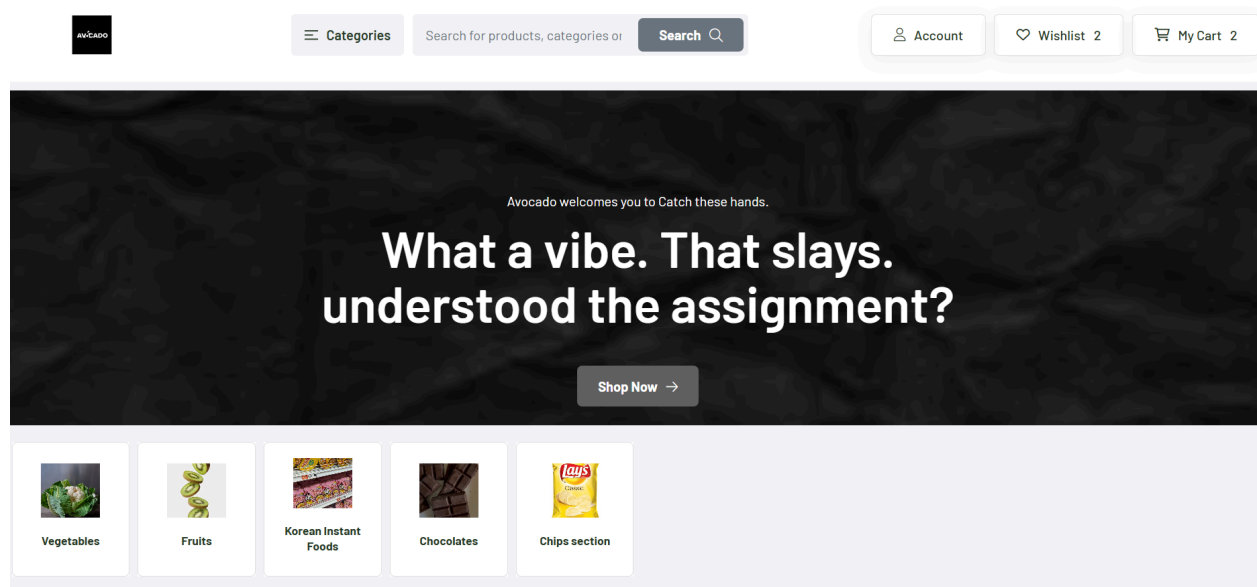
INSTALLATION GUIDE

DATA BASE

CONCLUSION

INTRODUCTION

The **Avocado grocery web app** project is an advanced e-commerce web application developed using the **MERN stack** (MongoDB, Express.js, React, Node.js).

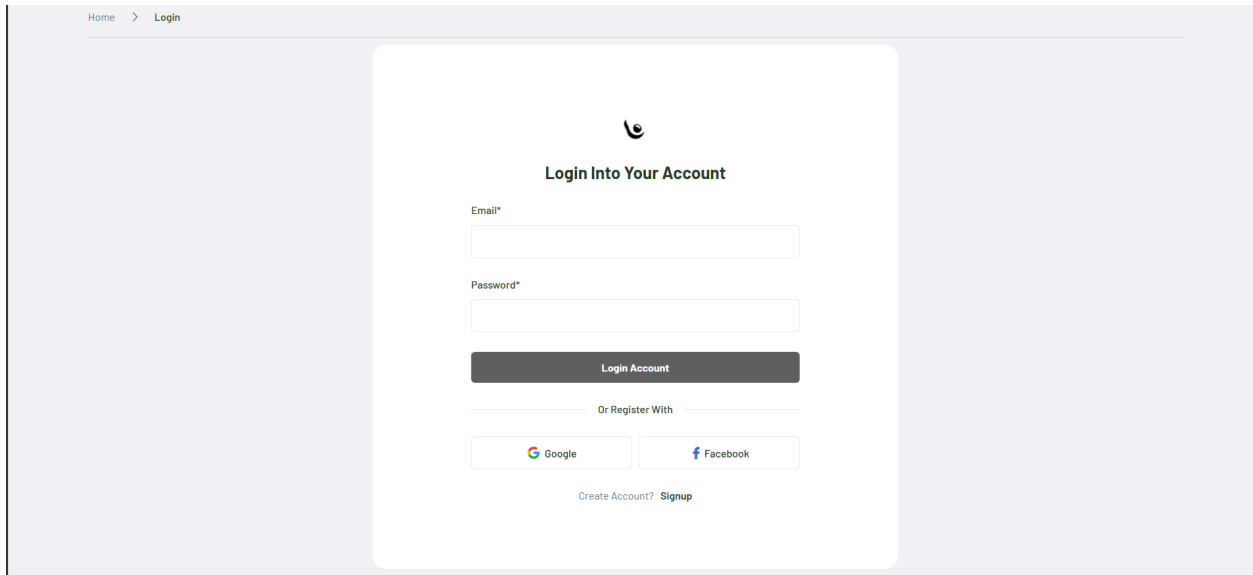


It provides users with a seamless shopping experience, allowing them to browse, manage, and purchase products online.

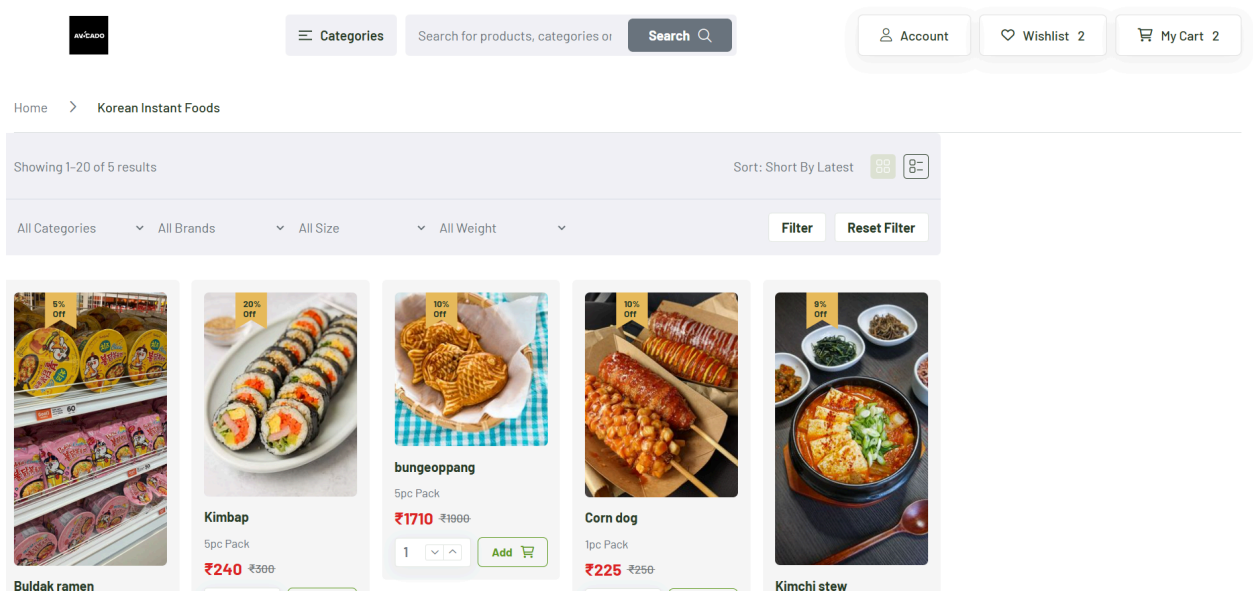
This application highlights full-stack development with multiple features, including a cart system, user authentication, and admin control for product management.

FEATURES

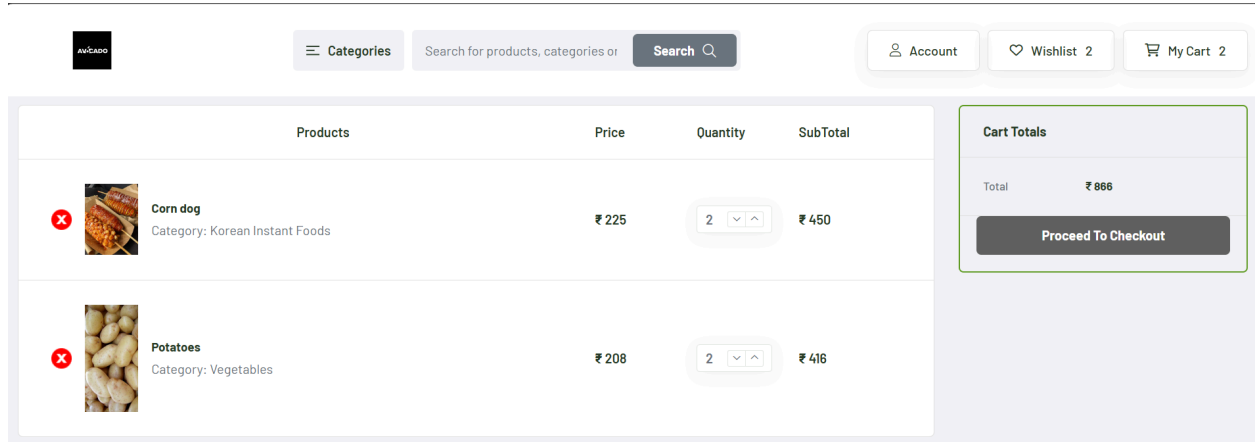
User Authentication: Secure sign-up, login, and purchase functionalities.



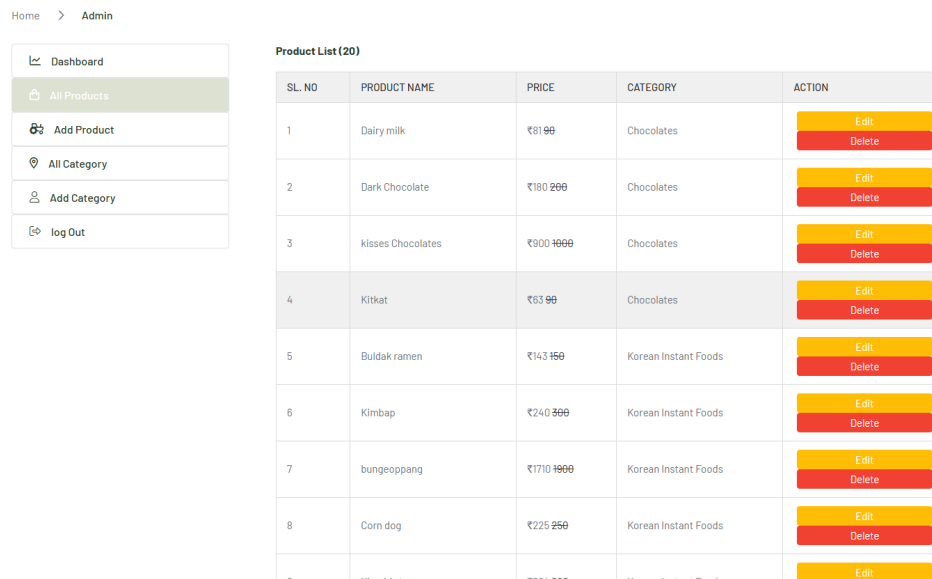
Product Page: Categorized product listings for easy browsing.









Cart System: Add, update, and checkout items in the cart.



Admin Control: Manage products, categories, and orders.



Payment Gateway Integration: Secure payments for users.

Billing Details	Your Order								
<p>Email Address*</p> <input type="text" value="twetha3188@gmail.com"/>	<table><thead><tr><th>Products</th><th>Price</th></tr></thead><tbody><tr><td> Corn dog</td><td>₹ 450</td></tr><tr><td> Potatoes</td><td>₹ 416</td></tr><tr><td colspan="2">Total Price: ₹ 866</td></tr></tbody></table>	Products	Price	 Corn dog	₹ 450	 Potatoes	₹ 416	Total Price: ₹ 866	
Products	Price								
 Corn dog	₹ 450								
 Potatoes	₹ 416								
Total Price: ₹ 866									
<p>First Name*</p> <input type="text" value="SWETHA"/>	<p>Cash On Delivery <input checked="" type="radio"/> Online</p>								
<p>Last Name*</p> <input type="text" value="t"/>	<p>Your personal data will be used to process your order, support your experience throughout this website, and for other purposes described in our privacy policy.</p>								
<p>Country / Region*</p> <input type="text" value="India"/>	<p><input checked="" type="checkbox"/> I have read and agree terms and conditions *</p>								
<p>Street Address*</p> <input type="text" value="dsdfdsfd"/>	<p>Place Order</p>								
<p>Town / City*</p> <input type="text" value="dfd"/>									
<p>State*</p> <input type="text" value="e"/>									
<p>Zip Code*</p> <input type="text" value="32234"/>									
<p>Phone*</p> <input type="text"/>									

TECHNOLOGIES USED

- **Frontend:** React.js
- **Backend:** Node.js, Express.js
- **Database:** MongoDB
- **Image Storage:** Cloudinary
- **Deployment:** Render (Backend), Netlify (Frontend)

PROJECT STRUCTURE

- **Frontend:** React and Redux for a dynamic UI.
- **Backend:** Node.js and Express.js to manage API requests and business logic.
- **Database:** MongoDB Atlas for data storage.
- **Image Management:** Cloudinary for product image storage

HOW IT WORKS

Home Page: Displays a list of products with filtering options.

User Authentication: Users register or log in for personalized features.

Product Page: Displays detailed product information.

Admin Controls: Enables admins to manage products and categories.

Cart & Checkout: Users add products to the cart and proceed to payment.

INSTALLATION GUIDE

1. Clone the repository:

```
git clone https://github.com/zukachin/Grocery-webapp.git
```

2. Install dependencies:

```
cd Grocery-webapp  
npm install
```

3. Configure environment variables for MongoDB and Razorpay in .env.

```
DBURI="your-mongodb-uri"  
CLOUDINARY_CLOUD_NAME="your-cloudinary-cloud-name"  
CLOUDINARY_API_KEY="your-cloudinary-api-key"  
CLOUDINARY_API_SECRET="your-cloudinary-api-secret"  
SECRET="your-jwt-secret"  
KEY_ID="razorpay-id"  
KEY_SECRET="razorpay-secret-key"
```

4. Start the server: Frontend:

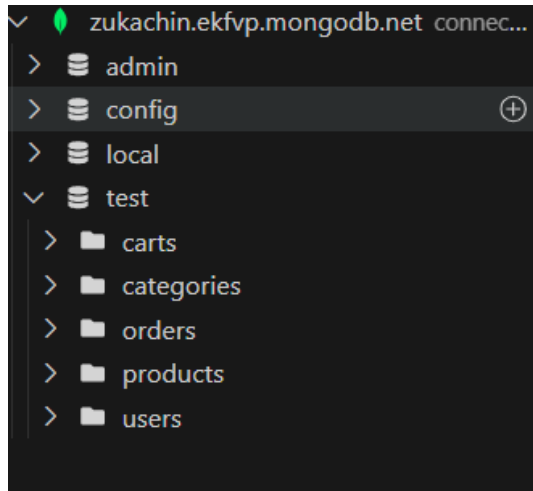
```
cd frontend  
npm run dev
```

Backend:

```
cd backend  
npm start
```

DATA BASE

MONGODB (VS CODE) – COLLECTIONS



DOCUMENTS

```
test.orders:{"_id":"67305c7497fdf1de121cde69"},json test.categories:{"_id":"67303acfc40f5d874e2330cd"},json test.carts:{"_id":"67305c6197fdf1de121cd5d9"},json > {} user
1 {
2   "_id": {
3     "$oid": "67305c6197fdf1de121cd5d9"
4   },
5   "products": [
6     {
7       "product": {
8         "$oid": "6730579797fdf1de121ccfb6"
9       },
10      "quantity": 2,
11      "_id": {
12        "$oid": "67305c6197fdf1de121cd5da"
13      }
14    },
15    {
16      "product": {
17        "$oid": "6730599b97fdf1de121cd0a6"
18      },
19      "quantity": 2,
20      "_id": {
21        "$oid": "67305c6597fdf1de121cd6c3"
22      }
23    }
24  ],
25  "user": {
26    "$oid": "67303a05c40f5d874e2330ac"
27  },
28  "_v": 1
29 }
```


CONCLUSION

The development and deployment of the **Avocado Grocery Web Application** proved that a robust, full-featured e-commerce platform can be built using the MERN stack.

This project demonstrates the potential of the MERN stack for scalable e-commerce solutions and provides a strong foundation for future enhancements.

REFERENCES

1. <https://github.com/Bharath136/grocery-webapp/>
2. <https://mongodb.com>
3. React Developer Tools – React
4. Image and Video Upload, Storage, Optimization and CDN
5. Best Payment Gateway in India - Manage Online Payments - Free Demo