RECGrocery

A MINI PROJECT REPORT

Submitted by

SAI RUTHWIK(220701237)

in partial fulfilment of the award of the degree
of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE

RAJALAKSHMI NAGAR

THANDALAM

CHENNAI – 602 105

NOVEMBER 2024

RAJALAKSHMI ENGINEERING COLLEGE

CHENNAI - 602105

BONAFIDE CERTIFICATE

Certified that this mini project report "**RECGrocery**" is the bonafide work of "**SAI RUTHWIK (220701237)**" who carried out the project work for the subject CS19542-Internet Programming under my supervision.

MR. DEEPAK KUMAR,

Assistant Professor (SG),

Department of

Computer Science and Engineering

Rajalakshmi Engineering College

Rajalakshmi Nagar

Thandalam

Chennai – 602105

Submitted to Project and Viva Voce Examination for the subject CS19542-Internet Programming held on _____.

INTERNAL EXAMINER   **EXTERNAL EXAMINER**

TABLE OF CONTENTS

ABSTRACT

The **Grocery Store Management System** is a web-based platform aimed at streamlining the management and purchasing process of grocery stores. The platform offers users the ability to browse products, add them to a cart, make secure payments, and track their orders.

The system also includes an administrator interface for managing inventory, monitoring orders, and generating sales reports. Designed with a user-friendly interface and a robust backend, the system enhances the shopping experience for customers while simplifying operations for store managers.

Future enhancements could include predictive stock management using AI, mobile app integration, and loyalty reward systems.

ACKNOWLEDGEMENT

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

The Grocery Store Management System is a comprehensive web-based application designed to revolutionize the traditional approach to grocery shopping and inventory management. With the rapid advancement of e-commerce technologies, consumers expect a seamless and efficient shopping experience, while business owners require robust tools to manage their operations effectively. This system addresses these demands by integrating customer and administrative functionalities into a unified platform.

For customers, the system provides an intuitive and user-friendly interface to browse a variety of grocery items, add them to a virtual cart, and complete secure online transactions. Additionally, it allows customers to track their orders in real time, offering them the convenience of shopping from the comfort of their homes while avoiding long queues at physical stores.

On the administrative side, the system empowers grocery store owners and managers with the tools they need to handle inventory, monitor sales, and track customer orders. It automates routine tasks such as stock updates and order fulfillment, reducing human error and increasing operational efficiency. The integration of sales analytics further aids decision-making, allowing businesses to identify trends and optimize their stock and pricing strategies.

The system is built using modern technologies such as React.js for the frontend, Node.js for the backend, and MongoDB for database management.

It is designed to ensure scalability, security, and ease of use, making it suitable for grocery businesses of all sizes.

This project represents a significant leap towards digital transformation in the retail sector, emphasizing customer satisfaction and operational efficiency. By addressing the pain points of both customers and business owners, the Grocery Store Management System aims to set a new standard for grocery shopping experiences in the digital era.

1.2 SCOPE OF THE WORK

- Facilitates an online shopping experience for customers.
- Streamlines inventory and order management for administrators.
- Ensures secure payment processing.
- Provides real-time stock updates and order tracking.

1.3 EXISTING SYSTEM

Traditional grocery store management is time-consuming and error-prone due to manual processes. Customers face difficulties in browsing stock and waiting in long queues for checkout. Store managers face challenges in inventory tracking and order handling.

1.4 AIM AND OBJECTIVES OF THE PROJECT

Aim:

To design and implement a web-based Grocery Store Management System that improves customer shopping experiences and enhances store operations.

Objectives:

- Develop an intuitive user interface for product browsing and purchasing.
- Integrate secure user authentication and payment systems.
- Design a backend for dynamic inventory and sales management.
- Enable real-time order tracking for customers.

CHAPTER 2

SYSTEM SPECIFICATIONS

2.1 HARDWARE SPECIFICATIONS

The system requires basic hardware to host the website, which can be adjusted based on user load. Recommended hardware specifications include:

- Processor: Intel i5 or higher

- RAM: 8 GB

- Storage: 256 GB SSD (preferred)

- Internet: High-speed broadband connection

For local development, any modern laptop or desktop with the above specifications will suffice. However, for production, cloud hosting is recommended to ensure scalability and accessibility.

2.2 SOFTWARE SPECIFICATIONS

- Frontend: HTML, CSS, JavaScript

- Backend: Node.js

- Database: MongoDB

- Framework: Express.js (backend), React.js (frontend)

- Tools: VS Code, GitHub, npm

- Payment Gateway: Razorpay

CHAPTER 3

ARCHITECTURE DIAGRAM
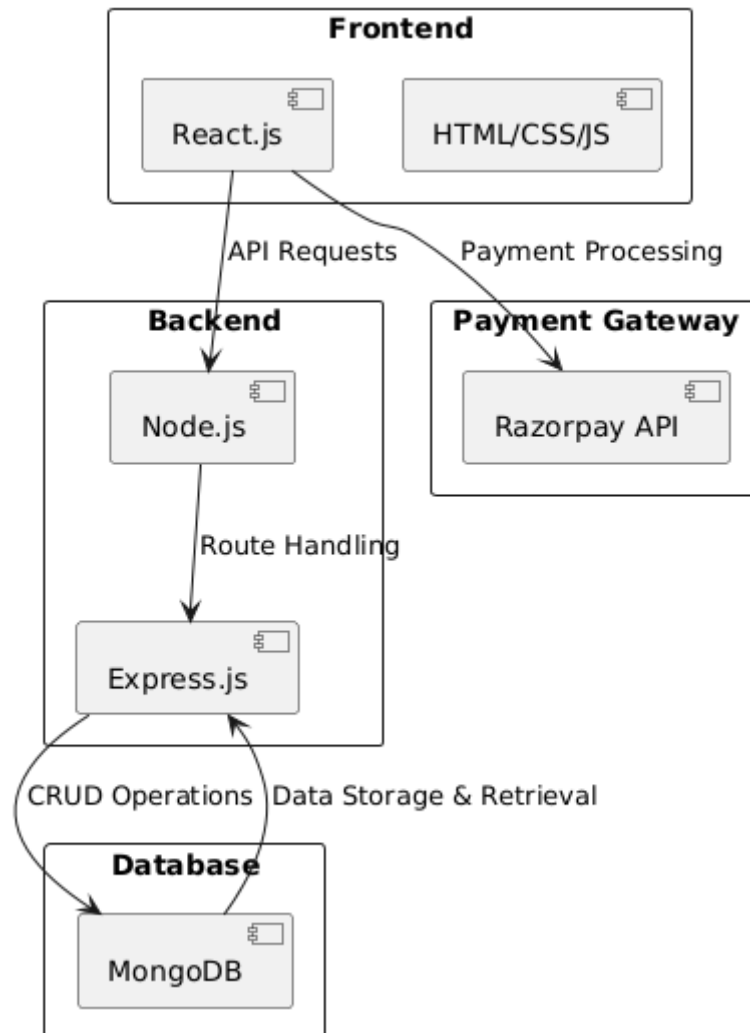
CHAPTER 4

MODULE DESCRIPTION

4.1 USER MODULE

- Browse products by category.
- Add items to the cart.
- Securely checkout using a payment gateway.
- Track orders in real-time.

4.2 ADMIN MODULE

- Login and authentication.
- Add, edit, and delete products.
- Track and manage customer orders.
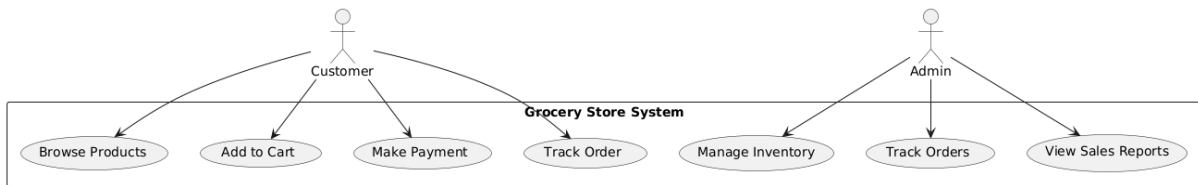- View sales reports and analytics.

4.3 PAYMENT MODULE

- Secure payment processing using Razorpay.
- Supports credit/debit cards, UPI, and wallets.
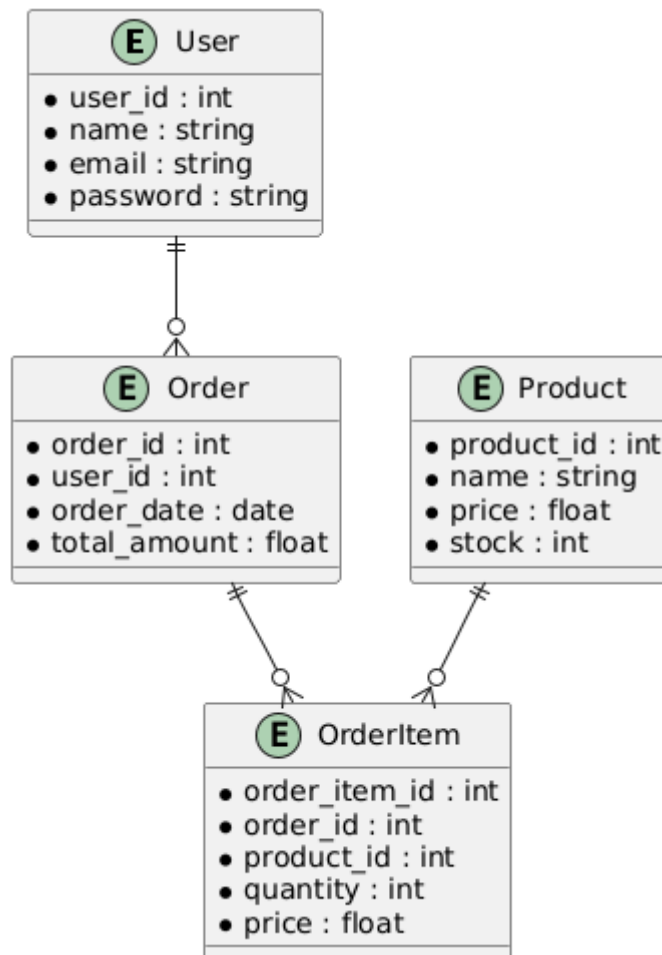- Sends order confirmation after successful payment.

CHAPTER 5
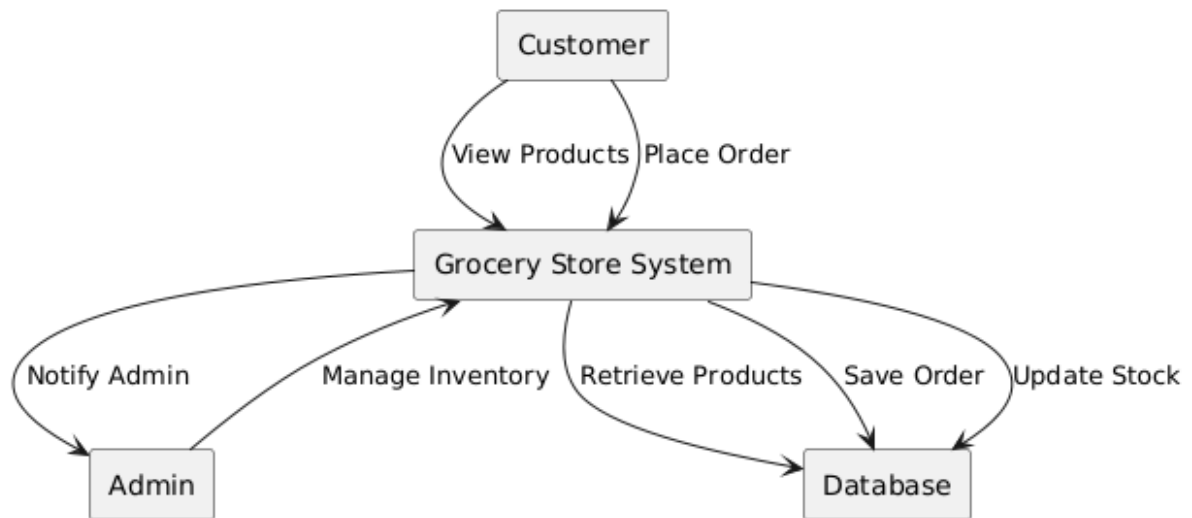
SYSTEM DESIGN

5.1 USE CASE DIAGRAM

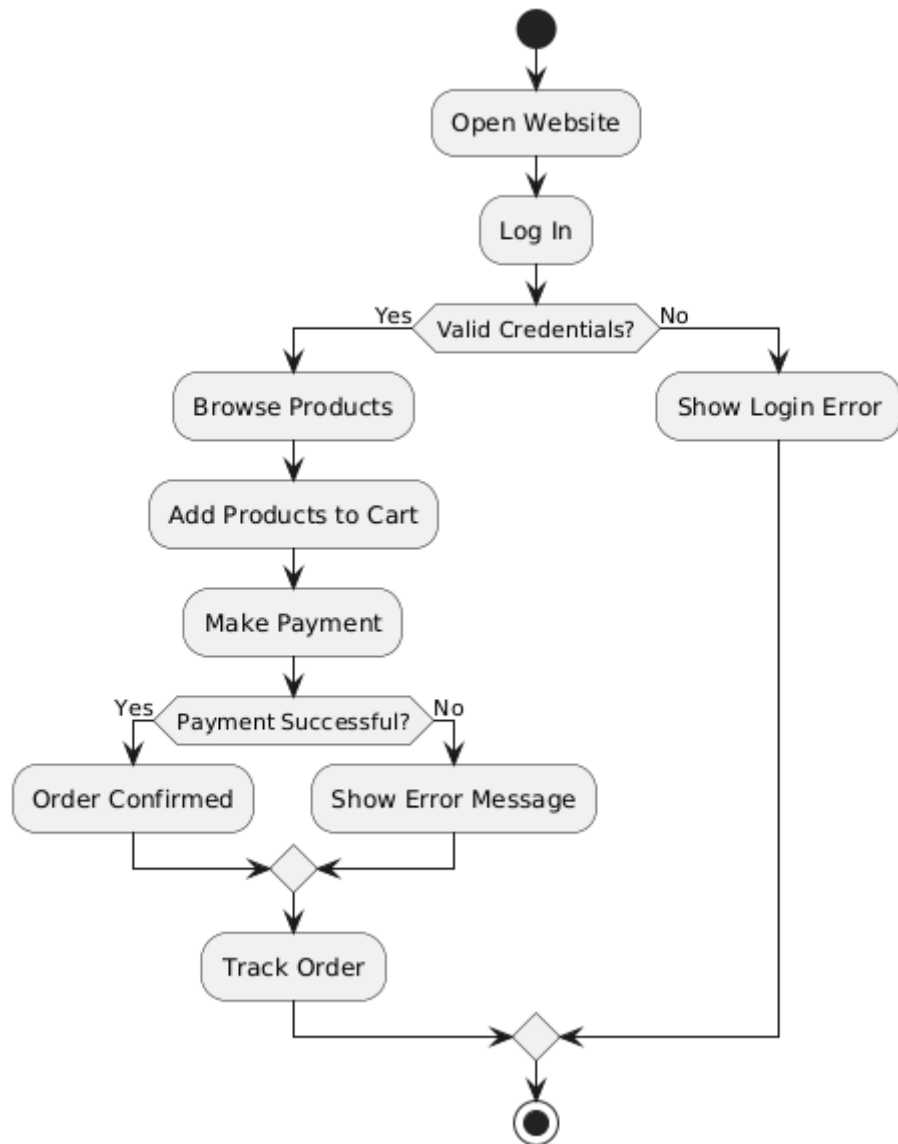## 5.2 E-R DIAGRAM

5.3 DATA FLOW DIAGRAM

5.4 ACTIVITY DIAGRAM
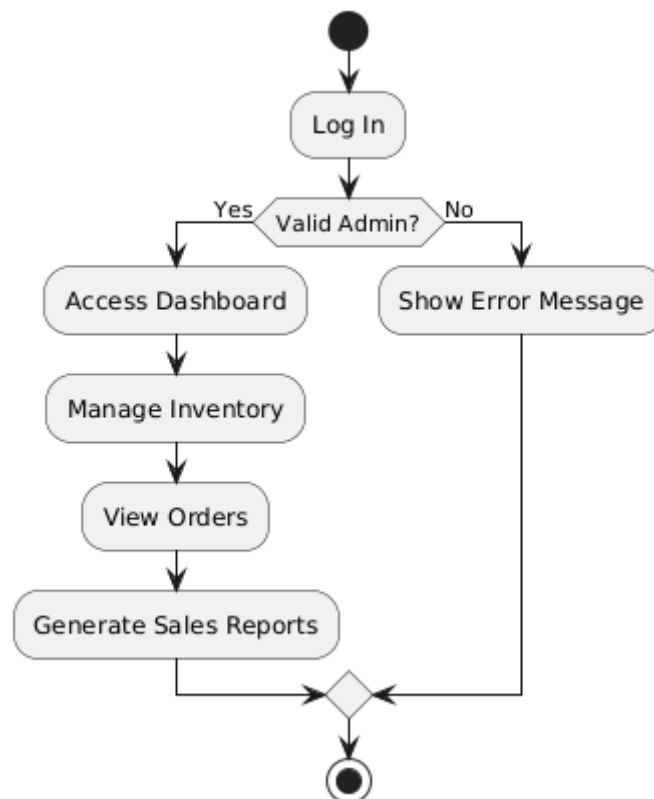
CHAPTER 6

SCREENSHOTS

LOGIN NOW

enter your email

enter your password

Login Now

don't have an account? register now

USER ACCOUNTS

user id : 31

username : admin

email :
sierrahotel777@gmail.com

user type : user

Delete

REGISTER NOW

enter your name

enter your email

enter your password

confirm your password

Choose File   No file chosen

Register Now

already have an account? login now

REC Gro.   Home Shop My Orders About Support

don't panic, go organice

REACH FOR A HEALTHIER YOU WITH ORGANIC FOODS

Lorem ipsum dolor sit amet consectetur adipisicing elit. Iusto natus culpa officia quasi, accusantium explicabo?

About Us

SHOP BY CATEGORY

Home   Shop   My Orders   About   Support

👤 🔍 ❤(0) 🛒(0)

PRODUCTS ADDED

Your Cart Is Empty

grand total : Rs. 0/-

Continue Shopping

Delete All

Proceed To Checkout

AdminPanel

home   products   orders   users   messages   👤

DASHBOARD

$0/-	$0/-	0	0
total pendings	completed orders	orders placed	products added
See Orders	See Orders	See Orders	See Products

1	1	2	0
total users	total admins	total accounts	total messages
See Accounts	See Accounts	See Accounts	See Messages

ADD NEW PRODUCT

enter product name

enter product price

select category ⌄

Choose File   No file chosen

enter product details

Add Product

PRODUCTS ADDED

Now Products Added Yet!

CHAPTER 7

CONCLUSION

The **Grocery Store Management System** has successfully achieved its core objectives of simplifying and digitizing grocery store operations, improving both the user experience and administrative management. By offering an easy-to-use platform for customers to browse and purchase items, the system enhances the shopping experience, providing convenience, flexibility, and efficiency. Additionally, real-time order tracking and a secure payment system help create a seamless and reliable shopping experience, fostering greater customer satisfaction.

On the administrative side, the system has streamlined inventory management, order processing, and sales reporting. The backend's interaction with the **MongoDB database** ensures that product information, order details, and customer data are efficiently stored and retrieved, minimizing the chances of human error. Administrators can also generate reports to track sales, optimize stock levels, and manage customer orders, all through a unified dashboard.

In conclusion, this system not only addresses the needs of grocery store owners and managers but also enhances the overall shopping experience for users, offering a complete, modern solution for grocery management. Its future versions can incorporate machine learning algorithms for dynamic pricing, predictive analytics for inventory management, and even support for various payment methods.

REFERENCES

- **Node.js Documentation**
  Node.js is the backbone for server-side JavaScript execution. Its documentation provides extensive resources for server management and API development.
  Link: https://nodejs.org/en/docs/

- **MongoDB Documentation**
  MongoDB is the NoSQL database chosen for this project, providing flexibility and scalability for managing product and customer data.
  Link: https://www.mongodb.com/docs/

- **React.js Documentation**
  React.js is a JavaScript library used for building user interfaces, and its documentation helps developers understand the core principles behind component-based architecture.
  Link: https://reactjs.org/docs/getting-started.html

- **Razorpay API Documentation**
  Razorpay is the integrated payment gateway that ensures secure online transactions. This documentation details how to integrate payment solutions in web applications.
  Link: https://razorpay.com/docs/

- **Express.js Documentation**
  Express.js is a web application framework for Node.js, used for building APIs and handling server requests efficiently.
  Link: https://expressjs.com/en/starter/installing.html

- **GitHub Repository**
  The code for this project can be found on GitHub, where various versions and commits are available for review and collaboration.
  Link: https://github.com/sierrahotel777/220701251-CS19542-Internet-Programming-Lab/tree/main/grocery%20store

- **MongoDB Atlas**
  MongoDB Atlas is a fully-managed cloud database that provides a robust environment for deploying MongoDB databases in the cloud.
  Link: https://www.mongodb.com/cloud/atlas