

HOME BAKED DELIVERY SYSTEM

PROJECT REPORT

*Submitted in fulfilment for the Course Project of UBCA301L – Full Stack
Application Development*

in

B.C.A

by

ANSEL JACOB AJU (23BCA0223)

DHEVANANDA T V (23BCA0201)

SREE RAM V (23BCA0153)

Under the guidance of

Dr. Brindha.K

SCORE



VIT[®]
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

School of Computer Science Engineering and Information Systems

November 2025



School of Computer Science Engineering and Information Systems

Fall Semester 2025-26

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	
1.	Problem Statement	3
2.	System design	4
3.	Software Requirement Specifications	7
4.	Implementation	
	4.1 Source code	13
	4.2 Screenshots	13

PROBLEM STATEMENT:

In today's digital age, customers increasingly rely on online platforms for purchasing food and bakery products. However, many small and medium-scale bakeries still operate using traditional, manual methods of order-taking and customer management. This approach often results in inefficiency, limited accessibility, and poor customer experience.

Customers must physically visit the bakery or rely on phone calls or messaging apps to place their orders. This manual process leads to miscommunication, delayed responses, and order inaccuracies, especially during peak hours. Furthermore, without an online presence, bakeries miss out on potential customers who prefer convenient online browsing and ordering options.

From a management perspective, maintaining paper-based order records or handling customer requests manually can lead to data loss, duplication, and difficulties in tracking sales and customer preferences. There is also no systematic way to display product details, categorize items, or manage pricing dynamically.

In addition, customers have limited visibility into the range of available products, ongoing offers, or ingredient details, which restricts their ability to make informed choices. In a competitive marketplace, this lack of accessibility and convenience can significantly affect a bakery's customer retention and growth.

Therefore, there is a strong need for a digital solution that provides a user-friendly, responsive web platform allowing customers to easily browse bakery items, place orders, and receive confirmation instantly. The proposed system should also allow the bakery to manage products, track orders, and maintain customer data efficiently in a centralized database.

By addressing these challenges, the project HomeBaked aims to bridge the gap between traditional bakery operations and modern e-commerce standards, ensuring smoother workflows, enhanced customer satisfaction, and improved business efficiency.

SYSTEM DESIGN:

The HomeBaked system has been designed using a modular and layered architecture that ensures flexibility, scalability, and ease of maintenance. It integrates both the frontend user interface and the backend server-side processing to deliver a seamless online ordering experience for customers and efficient data handling for the bakery administrators.

Overview:

The system follows the Client–Server Architecture, where:

The frontend (client side) is responsible for interacting with the users.

The backend (server side) handles business logic, database operations, and communication with the frontend.

This architecture ensures clear separation of concerns, improves performance, and allows for future scalability such as mobile app integration.

System Components:

The HomeBaked system consists of the following major components:

1. Frontend (User Interface)

Technology Used: React.js

The frontend provides an intuitive, responsive, and user-friendly interface.

It allows customers to:

- Browse and view bakery products with images, descriptions, and prices.

- Add or remove items from the shopping cart.
- Enter customer details and place orders through a checkout form.
- View order confirmation and details after successful placement.
- The frontend communicates with the backend using HTTP requests (via Fetch API).

2. Backend (Application Server)

Technology Used: Node.js with Express.js framework

The backend manages the application's business logic and handles client requests.

Key responsibilities include:

- Processing and validating customer order data.
- Generating a unique Order ID for each purchase.
- Communicating with the database to store and retrieve order information.
- Sending responses back to the frontend for order confirmation.

3. Database Layer

Technology Used: MongoDB (NoSQL database)

The database stores all critical information such as:

- Customer details (name, phone number, address).

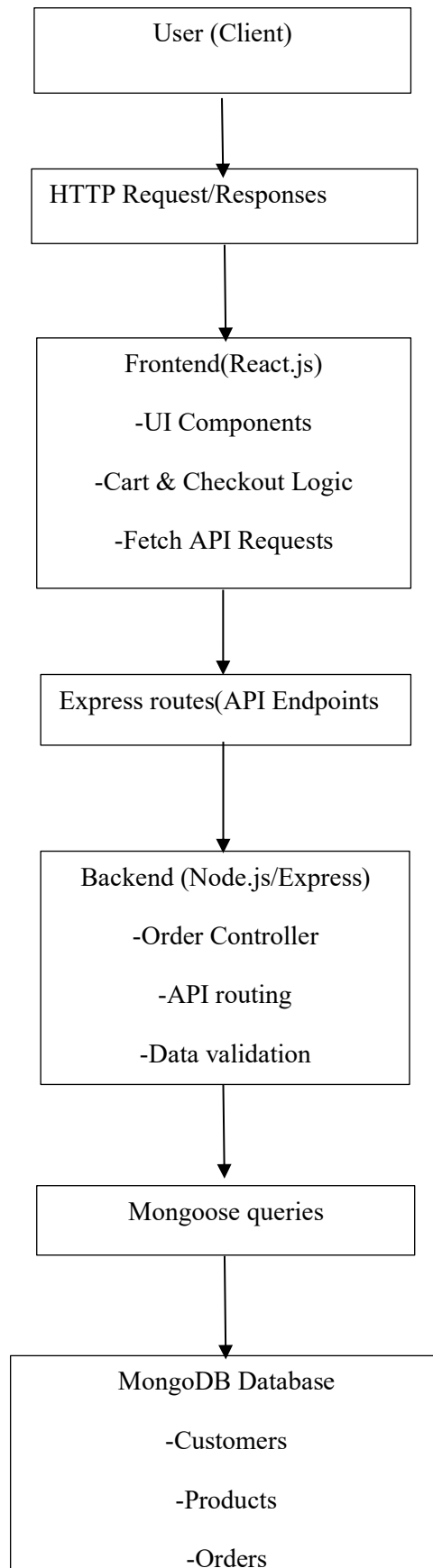
- Product information (name, price, image, description).
- Order details (order ID, items, payment method, total amount, timestamps).
- MongoDB provides a flexible schema, making it easy to store JSON-like documents.

The backend connects to MongoDB using Mongoose ODM for efficient data operations.

Data Flow Diagram (Conceptual Flow):

- ❖ Step 1: The customer visits the HomeBaked website and browses products displayed on the frontend.
- ❖ Step 2: Selected products are added to the cart for checkout.
- ❖ Step 3: The customer fills out their details and submits the order form.
- ❖ Step 4: The frontend sends the order data to the backend via an HTTP POST request.
- ❖ Step 5: The backend validates the data, generates an order ID, and stores it in MongoDB.
- ❖ Step 6: The backend sends a success response to the frontend with the order details.
- ❖ Step 7: The frontend displays the Order Confirmation Page showing all customer and order information.

System Architecture Diagram (Description):



Advantages of the Design:

- **Modular and Scalable:** Each layer (frontend, backend, database) can be updated independently.
- **Secure:** Sensitive data like customer details and payment info are securely handled.
- **Efficient Communication:** Uses RESTful APIs for fast and lightweight data exchange.
- **Responsive Design:** Ensures compatibility across desktops, tablets, and mobile devices.
- **Maintainable Codebase:** Clear separation of concerns allows easier debugging and enhancement.

Software Requirement Specifications:

1. Introduction

1.1 Purpose

The purpose of this Software Requirement Specification (SRS) document is to define the requirements for the HomeBaked web application — an online platform that allows customers to browse, select, and order bakery products from the comfort of their homes.

This document describes both the functional and non-functional requirements, detailing the system's behavior, features, and constraints.

1.2 Scope

The HomeBaked system provides a digital platform for customers to easily explore available bakery items, place customized orders, and receive real-time order confirmations.

The system also assists the bakery management by storing all order details, customer information, and transaction history in a centralized database.

Key Features:

- Browse a variety of bakery items with details (name, image, description, and price).
- Add and remove items from the cart dynamically.
- Provide delivery details through an intuitive checkout form.
- Place orders online and receive confirmation with an order ID.
- Manage all customer orders through a connected backend database (MongoDB).

1.3 Objectives

The objectives of HomeBaked include:

- To simplify the process of ordering bakery products online.
- To create a responsive, user-friendly, and visually appealing web application.
- To ensure secure and efficient data management using a modern backend framework.
- To enhance customer satisfaction through a reliable and fast online experience.

1.4 Definitions, Acronyms, and Abbreviations

- ❖ Term Description
- ❖ UI User Interface
- ❖ API Application Programming Interface
- ❖ DB Database
- ❖ SRS Software Requirement Specification
- ❖ CRUD Create, Read, Update, Delete
- ❖ JSON JavaScript Object Notation
- ❖ MERN MongoDB, Express.js, React.js, Node.js

2. Overall Description

2.1 Product Perspective

HomeBaked is a web-based application built using the MERN stack.

It follows a client-server architecture, where:

- The frontend (React.js) handles user interactions.
- The backend (Node.js + Express.js) processes business logic and communicates with the database.
- The database (MongoDB) stores order and customer details.

2.2 Product Functions

The major functions of the system include:

- Displaying bakery items dynamically on the homepage and menu page.
- Enabling users to add selected items to the shopping cart.
- Allowing users to enter delivery details and payment preferences at checkout.
- Saving the order details into MongoDB.
- Generating a unique order ID for each purchase.
- Displaying an order confirmation page with all relevant order details.

2.3 User Characteristics

Customer:

- Non-technical end-users who browse and order bakery items online.
- Requires a simple, intuitive interface and responsive design.

Administrator (Bakery Owner):

- Responsible for viewing and managing customer orders in the database.
- Requires reliable backend access for monitoring and record-keeping.

2.4 Constraints

- The system depends on internet connectivity for operation.
- The backend must be running for data to be saved in MongoDB.
- The application should be hosted on a server that supports Node.js.
- Data storage and retrieval are limited by MongoDB database capacity.

2.5 Assumptions and Dependencies

- Users have a modern web browser (Chrome, Firefox, Edge).
- MongoDB service must be active for database connectivity.
- Node.js and npm must be properly configured on the server.
- Proper API communication between frontend and backend is required.

3. Specific Requirements

3.1 Functional Requirements

- ❖ ID Requirement Description
- ❖ FR1 The system shall allow users to view all available bakery items.
- ❖ FR2 The system shall allow users to add or remove items from the shopping cart.
- ❖ FR3 The system shall allow users to enter personal and delivery information.
- ❖ FR4 The system shall generate a unique order ID for each placed order.
- ❖ FR5 The system shall store order details in MongoDB.
- ❖ FR6 The system shall send confirmation details to the frontend.
- ❖ FR7 The system shall display an order confirmation page after successful order placement.

3.2 Non-Functional Requirements

- ❖ ID Requirement Description
- ❖ NFR1 Performance: The system must load product pages within 3 seconds.
- ❖ NFR2 Security: Customer data must be transmitted securely between client and server.

- ❖ NFR3 Usability: The interface must be simple, responsive, and intuitive.
- ❖ NFR4 Reliability: The system must handle concurrent users without crashes.
- ❖ NFR5 Scalability: The backend should allow integration of new features such as admin login or payment gateway.
- ❖ NFR6 Availability: The application should remain operational 24/7 when hosted online.

3.3 Hardware Requirements

- | | |
|-------------|---|
| ❖ Component | Minimum Requirement |
| ❖ Processor | Intel Core i3 or higher |
| ❖ RAM | 4 GB or above |
| ❖ Storage | 500 MB (for project files and MongoDB data) |
| ❖ Display | 1366 × 768 resolution minimum |

3.4 Software Requirements

- | | |
|----------------------|-------------------------------------|
| ❖ Software | Description |
| ❖ Operating System | Windows 10 / Linux / macOS |
| ❖ Frontend Framework | React.js |
| ❖ Backend Framework | Node.js with Express.js |
| ❖ Database | MongoDB |
| ❖ Development Tools | Visual Studio Code, MongoDB Compass |
| ❖ Browser | Google Chrome / Mozilla Firefox |
| ❖ Package Manager | npm (Node Package Manager) |

4. Conclusion

The HomeBaked system provides a complete and efficient solution for digitalizing traditional bakery operations. It simplifies the customer experience, enhances order management, and promotes business growth through technology. This SRS defines all key requirements to ensure that the system is robust, reliable, and user-centric, aligning with modern e-commerce standards.

IMPLEMENTATION

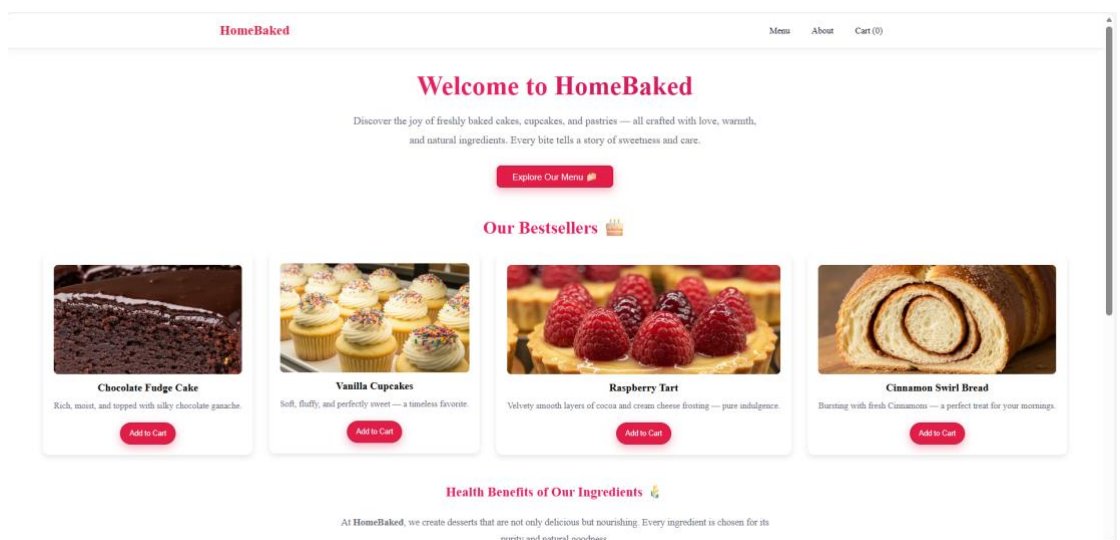
4.1 Source code

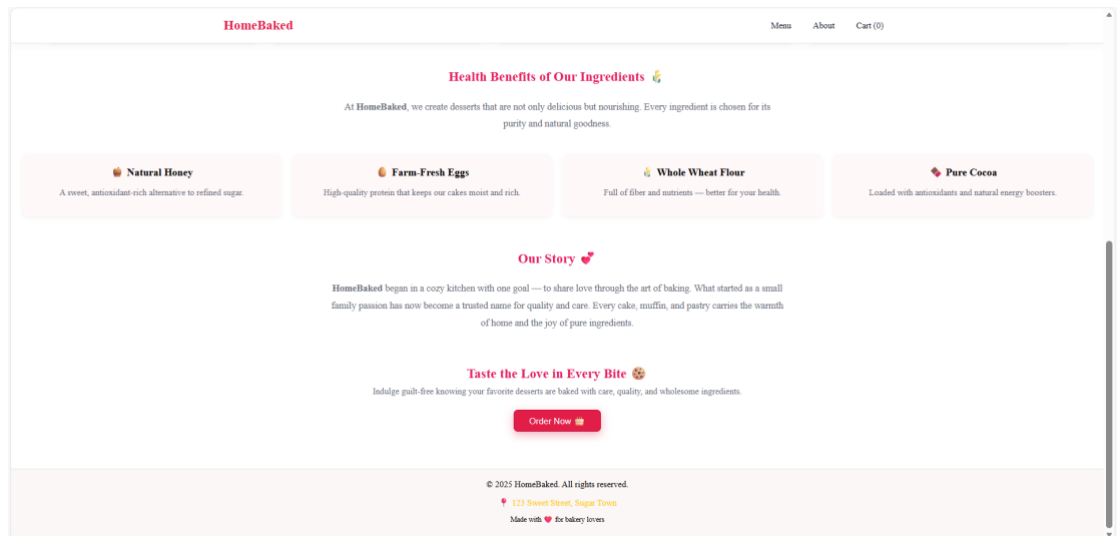
<https://drive.google.com/file/d/1kcVnSOgz66JWjKQxxNZabgML4jFAQxGn/view?usp=sharing>

4.2 Screenshot

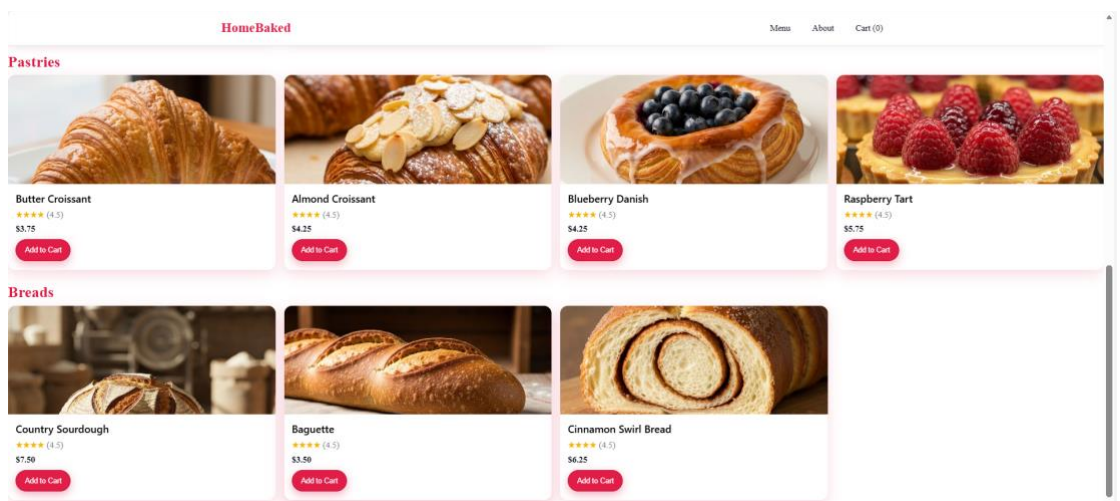
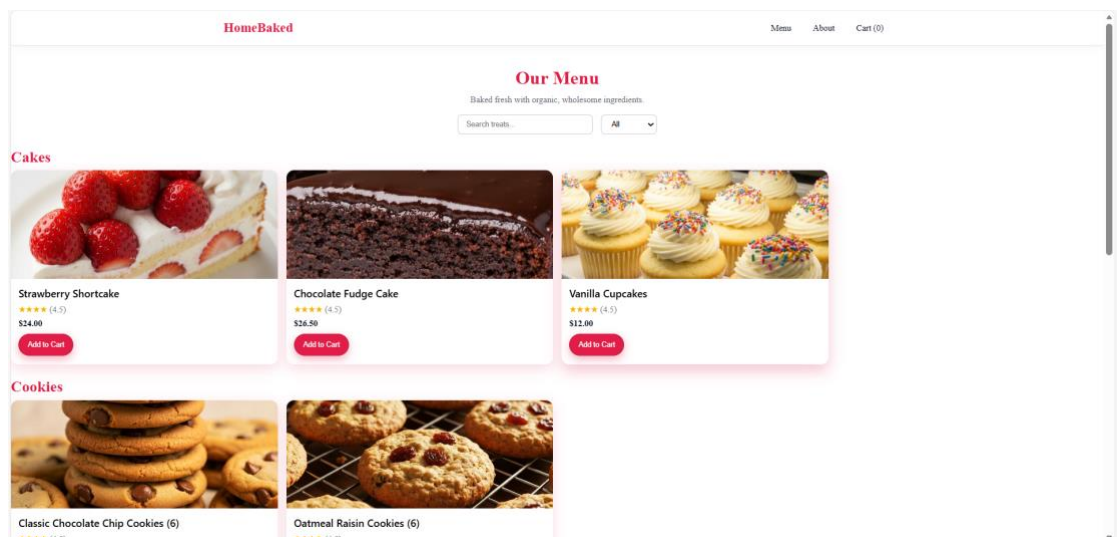
4.2.1 Frontend

Home:





Menu:



About:

HomeBaked

MenuAboutCart (0)

About HomeBaked

Every morning, we bake small batches of cakes, cookies, pastries, and breads using family recipes, organic flours, pasture-raised eggs, and real butter.

We keep it simple: no preservatives, no shortcuts — just fresh, homemade goodness delivered to your door with care.


- Freshly baked daily
- Organic, high-quality ingredients
- Local delivery and pick-up

Location

123 Sweet Street, Sugar Town


Contact

hello@homebaked.example
(555) 123-4567



© 2025 HomeBaked. All rights reserved.

123 Sweet Street, Sugar Town

Made with  for bakery lovers


Cart:

HomeBaked

MenuAboutCart (1)

Your Cart

Review your goodies before checkout.



Vanilla Cupcakes

\$12.00 each

-

1

+

\$12.00

Remove

Order Summary

Subtotal


Clear Cart

\$12.00

Proceed to Checkout

© 2025 HomeBaked. All rights reserved.

123 Sweet Street, Sugar Town

Made with  for bakery lovers

Checkout:

HomeBaked

MenuAboutCart (1)

Checkout

Provide your details to place your order.

Full Name*

DHEWANANDA TV

Phone Number*

+18943640980

Delivery Address*

123 Sweet Street, Sugar Town

Message on Cake (optional)

Happy Birthday!

Payment Method

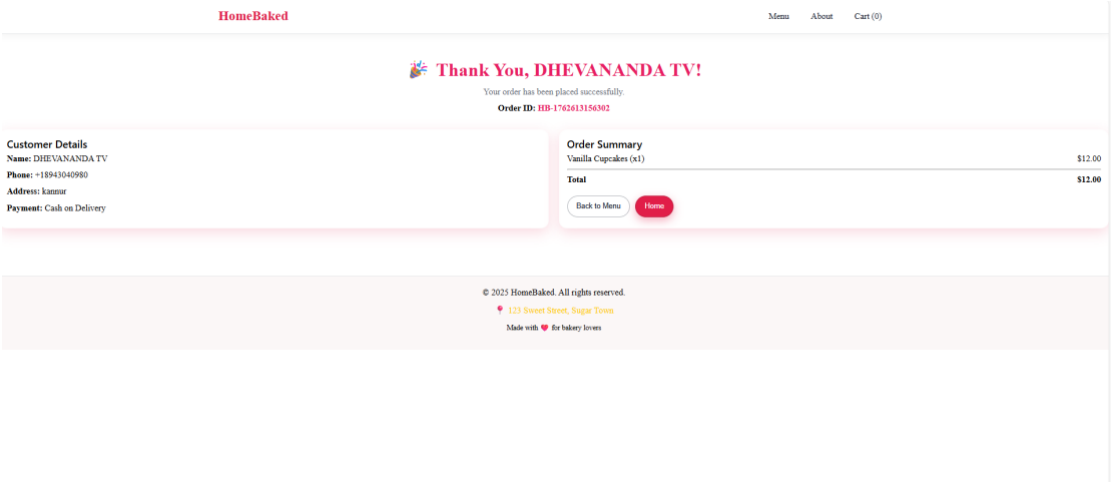
Cash on Delivery

Total: \$12.00

Submit Order

© 2025 HomeBaked. All rights reserved.

Order confirmation:



4.2.2 Backend:

