

Project for 5th Semester of Bachelor of Information Technology

“DELICIOUS”



By

Dinesh Gurung, Nikesh Chaudhary, Pravat Nagarkoti

KIST college of Information Technology

Faculty of Science and Technology

Purbanchal University, Nepal

February 10, 2025

CERTIFICATE



This Project is entitled “**DELICIOUS**” built and presented by **Dinesh Gurung (6075), Nikesh Chaudhary (6126), Pravat Nagarkoti (6084)** has been evaluated and is accepted as the **5th Semester project of Bachelor of Information and technology** by **Purbanchal University**.

External Examiner

Mr. Sameer Kumar Ram

Project Supervisor

Mr. Bhola Kafle

Department Coordinator

Deepak Khadka

ABSTRACT

This project presents the design and development of **Delicious**, a modern food ordering website that ensures a seamless and user-friendly experience. The primary objective is to provide an intuitive platform where users can browse food items, place orders efficiently, and interact through a dynamic cart system.

Built with a **Node.js** backend and a responsive frontend using **HTML and Tailwind CSS**, the platform integrates key functionalities such as user authentication, database management with **MongoDB**, and a structured backend API architecture. It enables smooth food item retrieval, secure order processing, and an interactive feedback system to enhance user engagement.

This project serves as a practical demonstration of **full-stack web development**, highlighting essential concepts such as API design, database integration, and responsive UI development. It underscores the importance of modern web technologies in enhancing digital food ordering experiences, ensuring efficiency and accessibility for users.

KEYWORDS:

Food Ordering, Web Development, User Authentication, Cart Management, Responsive Design, Database Integration, API Architecture, User Experience, Tailwind CSS, Node.js, MongoDB, Chatbot Support, Food Browsing, Order Management, Performance Optimization.

ACKNOWLEDGEMENT

It is with immense satisfaction and joy that we submit our report on project entitled **Delicious**. We have successfully completed this project as part of the curriculum of **Purbanchal University**.

We would like to take this opportunity to extend our sincere gratitude to our **BIT Coordinator, Mr. Deepak Khadka**, and **Project Supervisor, Mr. Bhola Kafle**, for their unwavering support, insightful guidance, and valuable advice, which played a crucial role in the successful completion of this project at every stage.

Furthermore, we express our heartfelt appreciation to **Purbanchal University** for designing a well-structured curriculum that has broadened our understanding in the field of Information Technology and empowered us to pursue a promising future in this ever-evolving domain.

Table of Contents

CERTIFICATE.....	ii
ABSTRACT.....	iii
ACKNOWLEDGEMENT.....	iv
CHAPTER 1: INTRODUCTION.....	1
1.1 Project Overview	1
1.2 Problem statement	1
1.3 Purpose & Objectives	1
CHAPTER 2: LITERATURE REVIEW.....	2
2.1 Introduction.....	2
2.2 Evolution of Food Delivery Platforms	2
2.3 Technologies Used in Online Food Delivery Systems	2
2.3.1 Frontend Technologies:	2
2.3.2 Backend Technologies:	3
2.3.3 Databases:	3
2.4 Challenges.....	3
2.5 Emerging Trends	4
CHAPTER 3: SYSTEM REQUIREMENTS AND DEVELOPMENT APPROACH.....	5
3.1 Hardware and Software Requirements	5
3.2 Development Methodology and Implementation	6
CHAPTER 4: SYSTEM ARCHITECTURE.....	8
4.1 Overview	8
4.2 Use Case Diagram.....	8
4.3 ER Diagram.....	9
4.4 Gantt Chart	10
CHAPTER 5: CONCLUSON AND FUTURE SCOPE.....	11
CHAPTER 6: REFERENCES.....	12

CHAPTER 1: INTRODUCTION

1.1 Project Overview

The website "**Delicious**" is a dynamic food ordering platform designed to provide a seamless and user-friendly experience for customers. This document aims to provide a comprehensive overview of the project, detailing its development process, key features, and underlying architecture.

1.2 Problem statement

In today's digital era, a seamless and efficient online food ordering system is essential for enhancing customer convenience. However, many platforms compromise user experience with unnecessary complexities, making navigation and order management less intuitive. "**Delicious**" addresses this by providing a well-structured and user-friendly platform that enables customers to browse food items, manage their cart dynamically, and submit feedback effortlessly. By focusing on core functionalities, project ensures a smooth and engaging ordering experience tailored to user needs.

1.3 Purpose & Objectives

The primary objectives of "**Delicious**" are:

- To provide an efficient and user-friendly food ordering platform that streamlines the food selection and ordering process for customers.
- To eliminate the frustration of complicated ordering systems, offering a simple and quick solution for both customers and restaurant owners.
- To enhance the customer experience by offering dynamic cart functionality, allowing users to easily modify quantities and track the total price in real-time.
- To improve customer interaction and engagement through an integrated chatbot, providing instant answers to general queries like food recommendations, while gathering valuable feedback to enhance service and offerings.

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

The "**Delicious**" project focuses on creating a user-friendly online food ordering platform that streamlines the food selection and ordering process for customers. The platform offers an intuitive interface where users can browse food menus, add items to their cart, and place orders for home delivery. This project aims to leverage modern web technologies to provide a seamless and responsive experience across devices, making it easy for users to explore various food options, customize their orders, and manage their profiles. By integrating essential features like user authentication, cart management, and chatbot support, the platform is designed to meet the growing demand for online food delivery services. Through a combination of front-end and back-end development, "Delicious" ensures both performance and scalability while delivering a simple yet efficient solution to enhance the food ordering experience.

2.2 Evolution of Food Delivery Platforms

Online food delivery platforms have transformed the traditional restaurant business model by introducing digital ordering systems, streamlined customer experiences, and increased delivery efficiency. According to Chauhan & Kapoor (2021), the food delivery industry has grown significantly due to technological advancements in mobile apps, payment systems, and real-time tracking. Platforms like Uber Eats and Grubhub focus on providing quick and user-friendly services, while also integrating features like personalized recommendations and customer reviews. Moreover, integration with third-party delivery services and payment gateways has become essential for these platforms.

2.3 Technologies Used in Online Food Delivery Systems

The success of food delivery platforms depends largely on the integration of various technologies that support frontend interfaces, backend management, and real-time service delivery.

2.3.1 Frontend Technologies:

- **HTML/CSS/JavaScript:** The backbone of any web-based food delivery platform, enabling responsive designs and interactive elements.

- **React:** A JavaScript library used by many food delivery platforms for dynamic user interfaces and quick updates without reloading the page.
- **Flutter:** A UI toolkit by Google for creating natively compiled applications for mobile devices, helping to maintain consistency across different platforms.
- **Swift (iOS) / Kotlin (Android):** Essential for building native mobile applications for iOS and Android devices.

2.3.2 Backend Technologies:

- **Node.js:** A runtime that powers scalable applications, often used in real-time systems such as food delivery apps.
- **Django:** A Python-based web framework that supports rapid development and secure management of food ordering data.
- **Ruby on Rails:** An efficient backend framework that supports fast application development, often chosen for startups in the food delivery domain.
- **GraphQL:** A query language for APIs that allows clients to request specific data, useful for delivering personalized customer experiences based on orders.

2.3.3 Databases:

- **MySQL/PostgreSQL:** Relational databases commonly used to store user data.
- **MongoDB:** A NoSQL database that enables flexible storage and quick access to dynamic data, such as reviews and menu items.
- **Redis:** Used to cache frequent queries and data like restaurant listings and menus, improving response time.

2.4 Challenges

Food delivery platforms face several challenges, including managing delivery logistics and maintaining a reliable customer service experience. High operational costs, especially related to delivery personnel and infrastructure, can affect profitability. Additionally, customer expectations for fast and accurate service put pressure on platforms to optimize route planning and delivery times. Regulatory compliance, data security, and managing customer privacy also remain key concerns in an increasingly competitive market. Balancing these factors while scaling the business continues to be a significant challenge.

2.5 Emerging Trends

Emerging trends in food delivery platforms include the integration of AI for personalized recommendations and optimized delivery routes. The use of autonomous vehicles and drones is expanding to improve delivery speed and efficiency. Contactless delivery options and the growing demand for healthier, sustainable food choices are reshaping consumer expectations. Additionally, subscription-based services and loyalty programs are becoming more common to retain customers. The rise of mobile-first platforms and enhanced data analytics for customer insights are also driving innovation in the industry (Smith & Johnson, 2023; Lee et al., 2022; Williams, 2024; Taylor & Green, 2023; Nguyen, 2023).

CHAPTER 3: SYSTEM REQUIREMENTS AND DEVELOPMENT APPROACH

3.1 Hardware and Software Requirements

The hardware and software required for the successful development and deployment of the system are described below:

Hardware Requirements:

- **Processor:** Minimum of 2.5 GHz multi-core processor for efficient computation.
- **RAM:** At least 8 GB of RAM to support the smooth running of the development tools and frameworks.
- **Storage:** 256 GB SSD for fast data processing and ample storage for development files and dependencies.
- **Networking:** Stable internet connection for accessing cloud-based services and API integrations.

Software Requirements:

- **IDE:** Visual Studio Code, IntelliJ IDEA, or any code editor of preference.
- **Programming Languages:** JavaScript (Node.js for backend), HTML, CSS (for frontend.)
- **Version Control:** Git (GitHub, GitLab, or Bitbucket for repository management.)
- **Database:** MongoDB (NoSQL database for storing user and product data.)
- **Backend Framework:** Express.js (Node.js framework for backend API.)
- **Frontend Framework:** Tailwind CSS (for responsive and flexible frontend design.)
- **Chatbot Integration:** OpenAI API (for chatbot functionality.)

3.2 Development Methodology and Implementation

The approach and methodology used in developing the system, along with the tools and practices followed during the implementation is:

Spiral Methodology:

The project follows the Spiral Model, focusing on iterative development with continuous risk assessment. Each cycle includes planning, design, implementation, and evaluation, allowing for the gradual refinement of features while addressing potential risks. This approach ensures flexibility and effective risk management throughout the development process.

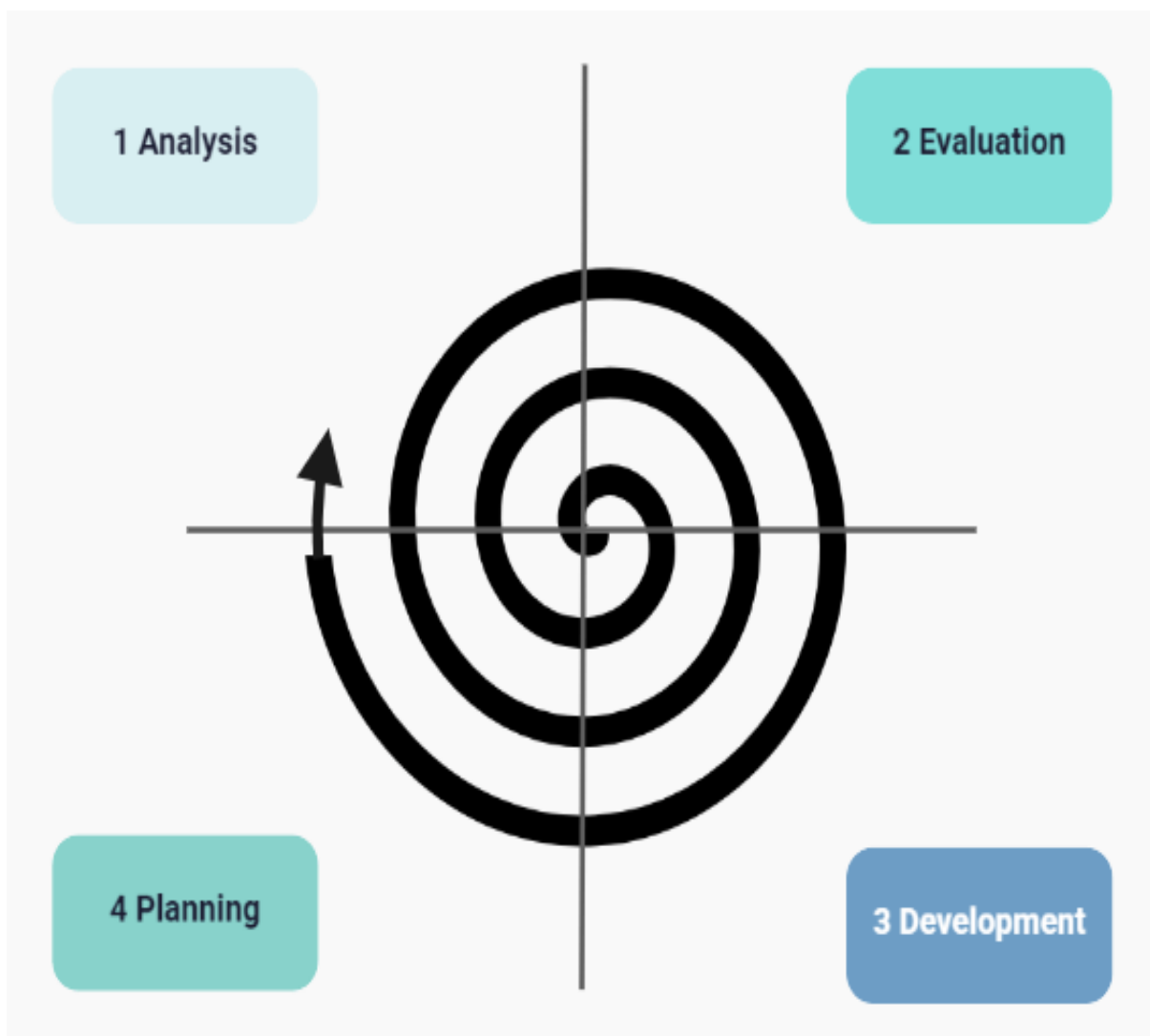


Fig: Spiral Model

"Delicious" follows the Spiral Model as follows:

1. Analysis

- Identify core features: food item listing, cart functionality, user authentication (only when adding to cart), and chatbot.
- Determine additional features like feedback submission and admin panel for managing food items.
- Assess feasibility regarding performance, database structure, and scalability.

2. Evaluation

- Identify potential risks, such as performance issues with database queries and UI responsiveness.
- Create prototypes for core functionalities (cart system, chatbot integration).
- Conduct initial user testing to refine UI/UX and system behavior.

3. Development

- Develop the system iteratively, implementing core features first.
- Continuously refine the system based on feedback, improving responsiveness and user experience.

4. Planning

- Analyze feedback from the previous iteration.
- Adjust the development plan to prioritize feature enhancements or fixes.
- Allocate resources (e.g., optimizing API calls, improving UI responsiveness) for the next cycle.

CHAPTER 4: SYSTEM ARCHITECTURE

4.1 Overview

The system architecture defines the structure and components of the delicious food ordering system, outlining how different modules are designed to interact with each other ensuring smooth functionality and performance.

4.2 Use Case Diagram

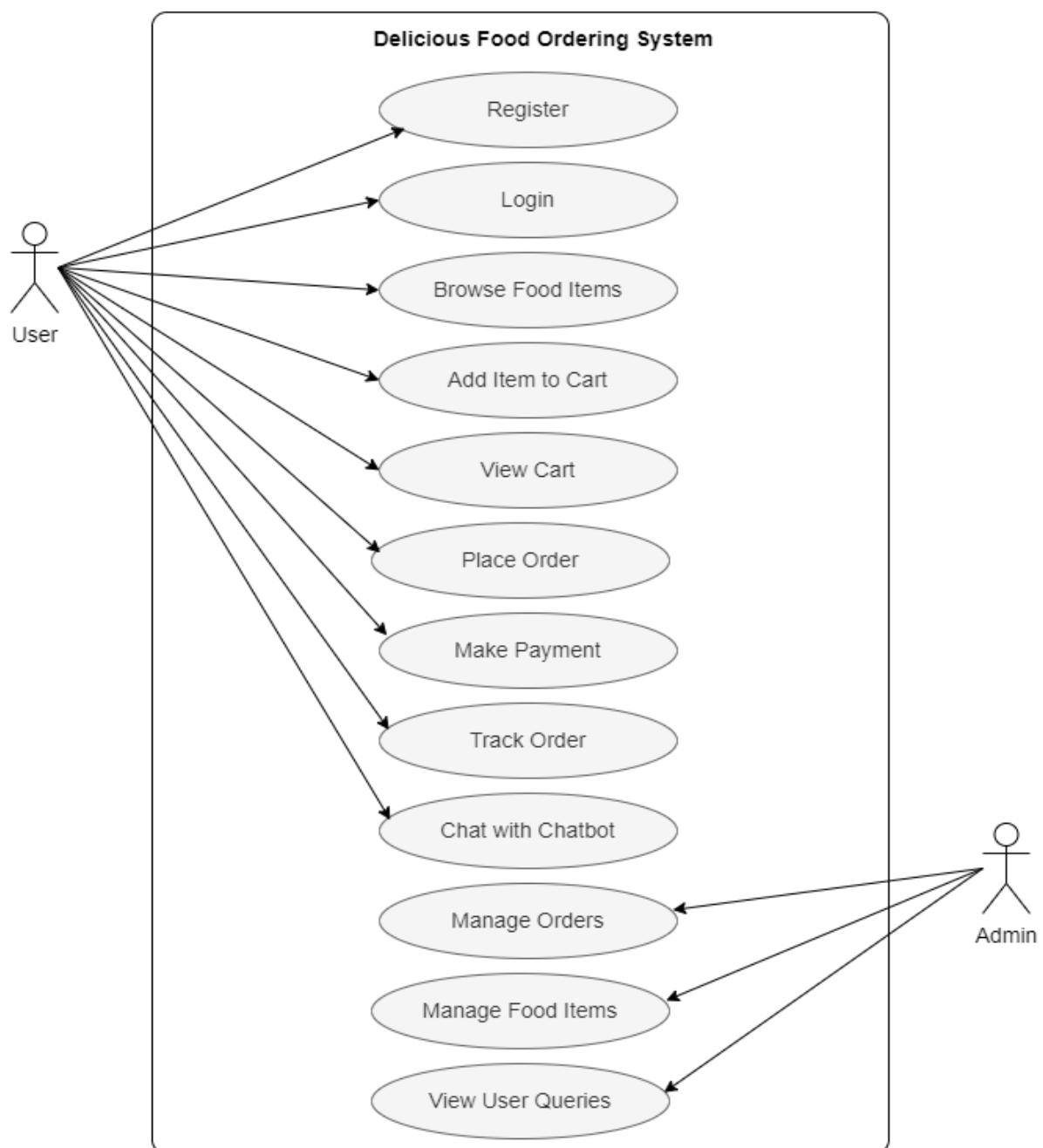


Fig: Use Case Diagram

4.3 ER Diagram

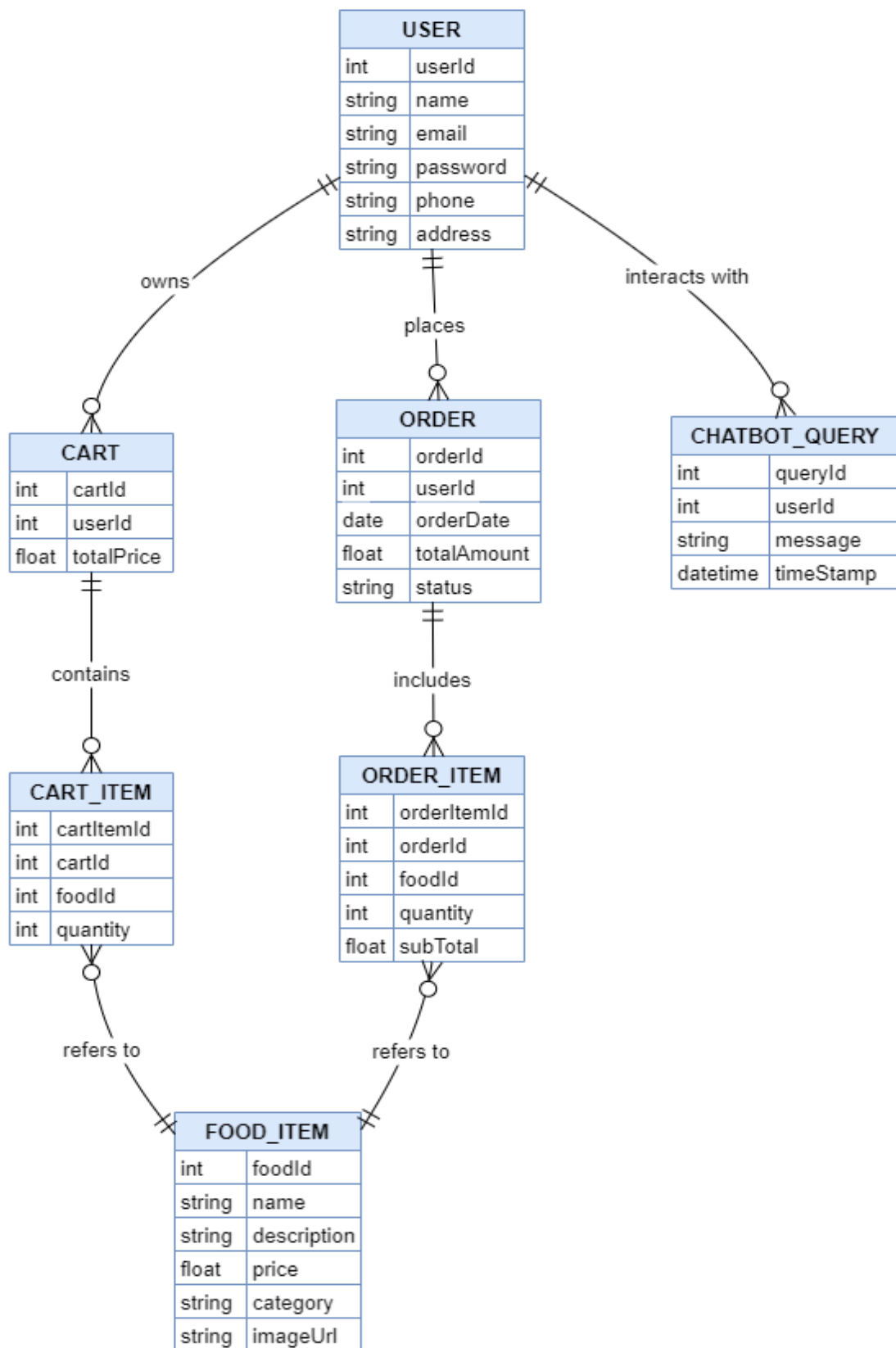


Fig: ER Diagram

4.4 Gantt Chart

Task	Start	End	Duration
Topic Selection	1/14/2025	1/15/2025	1
Conduct Research	1/15/2025	1/18/2025	3
Setup Development Environment	1/18/2025	1/20/2025	2
Create Homepage Interface	1/20/2025	1/22/2025	2
Implement Food Item Listing	1/22/2025	1/24/2025	2
Develop Cart Functionality	1/24/2025	1/27/2025	3
Implement User Authentication	1/27/2025	1/29/2025	2
Integrate Chatbot	1/29/2025	1/31/2025	2
Implement Feedback System	1/31/2025	2/1/2025	1
Test and Debug	2/1/2025	2/3/2025	2
Documentation	2/3/2025	2/4/2025	1

Delicious Project Development Timeline

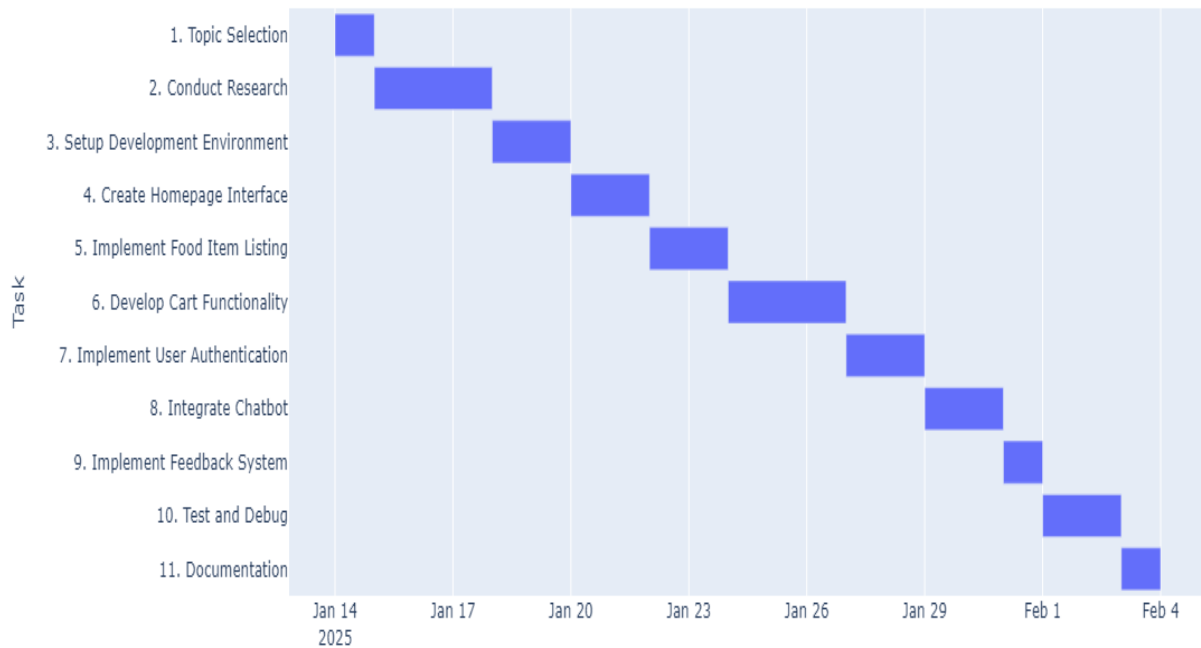


Fig: Gantt Chart

CHAPTER 5: CONCLUSION AND FUTURE SCOPE

This project serves as a learning model for understanding the key principles of online food ordering systems. By integrating essential features such as food selection, cart management, and user authentication, it provides a structured yet practical foundation for web application development.

This project goes beyond technical implementation; it acts as a stepping stone for aspiring developers looking to explore full-stack development.

- **Educational Value:** Enhances knowledge of modern web technologies and their real-world applications.
- **Industry Relevance:** Lays the groundwork for scaling into a fully operational food ordering service.
- **Career Growth:** Equips developers with hands-on experience in building and optimizing e-commerce platforms.

With future enhancements, this platform has the potential to evolve into a sophisticated and user-centric food ordering system.

- **Expanded Menu & Promotions:** Introduce new food categories, special deals, and time-limited discounts to enhance user engagement.
- **Personalized User Experience:** Leverage AI-driven recommendations, order history insights, and user preferences for tailored suggestions.
- **Enhanced Payment Flexibility:** Implement seamless transactions through multiple payment gateways, including digital wallets and instant UPI payments.
- **Real-Time Order Tracking:** Enable customers to monitor their orders from preparation to doorstep delivery with live updates.
- **Mobile Accessibility:** Develop dedicated mobile applications for Android and iOS to improve reach and usability.
- **Customer Loyalty & Rewards:** Establish a reward system where users earn points for purchases, unlocking discounts and exclusive offers.

CHAPTER 6: REFERENCES

Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1994). *Design patterns: Elements of reusable object-oriented software*. Addison-Wesley.

Martin, R. C. (2008). *Clean code: A handbook of agile software craftsmanship*. Prentice Hall.

Flanagan, D. (2020). *JavaScript: The definitive guide* (7th ed.). O'Reilly Media.

Freeman, E., & Robson, E. (2020). *Headfirst design patterns* (2nd ed.). O'Reilly Media.

Mozilla Developer Network (MDN). (2025, February 15). *Web APIs documentation*. Retrieved from <https://developer.mozilla.org>

World Wide Web Consortium (W3C). (2025, February 15). *HTML and CSS standards*. Retrieved from <https://www.w3.org>

MongoDB Inc. (2025, February 15). *MongoDB documentation*. Retrieved from <https://www.mongodb.com/docs>

Node.js Foundation. (2025, February 15). *Node.js documentation*. Retrieved from <https://nodejs.org/en/docs>

Tailwind CSS. (2025, February 15). *Tailwind CSS documentation*. Retrieved from <https://tailwindcss.com/docs>

Stack Overflow. (2025, February 15). *Common issues in Node.js and MongoDB integration*. Retrieved from <https://stackoverflow.com>