TAU BITE – A WEB BASED FOOD ORDERING AND DELIVERY SYSTEM FOR A UNIVERSITY CAMPUS

BY:

AHMED SARDAUNA MAIYAKI

(21/10MSS007)

A PROJECT SUBMITTED

TO

DEPARTMENT OF MATHEMATICAL AND COMPUTING SCIENCES

THOMAS ADEWUMI UNIVERSITY, NIGERIA.

AUGUST | 2025

TAU BITE – A WEB BASED FOOD ORDERING AND DELIVERY SYSTEM FOR A UNIVERSITY CAMPUS

BY:

AHMED SARDAUNA MAIYAKI

(21/10MSS007)

A PROJECT SUBMITTED

TO

THE DEPARTMENT OF MATHEMATICAL AND COMPUTING SCIENCES

FACULTY OF COMPUTING AND APPLIED SCIENCES

THOMAS ADEWUMI UNIVERSITY OKO-IRESE, KWARA STATE, NIGERIA.

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE BACHELOR OF SCIENCE (HONOURS) DEGREE IN SOFTWARE ENGINEEING

AUGUST | 2025

CERTIFICATION

This is to certify that I am responsible for the work submitted in this Project, that the original work is mine, except as specified in acknowledgment and references, and that neither the project nor the original work contained therein has been submitted to this University or any other institutions for the award of a degree.

Ahmed Sardauna Maiyaki (21/10MSS007)

APPROVAL

This project has been approved for the Department of Mathematical and Computing Science, Thomas Adewumi University, Oko, Kwara State, Nigeria

14/10/2025

Name of Supervisor ……………………………
Mr. Omojarabi Olajide **Signature and Date**

14/10.2025

Name of Head of Department …………………………….
Dr. Olabode Omosola **Signature and Date**

14/10/2025

Prof. Ayo Adebiyi …………………………….
Name of External Examiner **Signature and Date**

DEDICATION

This project is dedicated to the almighty Allah, Al Rahman, Al Raheem, Al Halim, Al hakeem, Al Mutakabeer, and to my beloved parents for their prayers and contributions towards this achievement. You will live long to enjoy the fruits of your labor and may Al Muhaymin continue to provide and guide you.

ACKNOWLEDGEMENT

First and foremost, all praise and gratitude belong to Allah (SWT), the Most Merciful, the Most Compassionate, the Source of all strength, wisdom, and guidance. Without His grace and blessings, this project would not have been possible. I am eternally grateful for His endless mercy, patience, and the ability He bestowed upon me to persevere through every challenge.

I would like to express my sincere appreciation to the following individuals and institutions for their invaluable support, guidance, and encouragement throughout the course of this project:

- Mr. Omojarabi Olajide, my dedicated supervisor for his expert guidance, constructive feedback, and unwavering patience. His mentorship went beyond supervision he was a constant source of motivation and clarity at every stage of this work.
- Dr. Olabode Omosola, Head of the Department of Mathematical and Computing Science for his leadership and commitment to academic excellence.
- Mr. Damilare Oladipo, ICT Director for providing essential technical support and access to resources.
- Mr. Paul and Miss Aimas Love Lily for their moral support and encouragement.

My deepest gratitude goes to my beloved family, whose unwavering love, prayers, and sacrifices gave me the strength and determination to see this journey through to completion. Your constant encouragement kept me focused and motivated, even in the most challenging moments.

Finally, I extend my appreciation to Thomas Adewumi University and the Department of Mathematical and Computing Science for providing an enabling academic environment and the necessary infrastructure for this research.

TABLE OF CONTENT

TABLE OF FIGURES

ABSTRACT

This project presents a comprehensive study on the development of a web-based food ordering and delivery system tailored specifically for Thomas Adewumi University cafeteria. The aim is to address the significant challenges posed by the existing manual system, which include long queues, inefficient payment processes, limited menu visibility, and a lack of effective feedback mechanisms. These issues contribute to operational inefficiencies, customer dissatisfaction, and financial losses, particularly in peak hours with a high demand for convenient food services.

The developed system leverages modern web technologies to create a user-friendly platform that enables customers to browse menus, place orders, give feed back and track deliveries in real time. It incorporates a robust cafeteria management system for efficient order processing, a secure payment gateway supporting multiple transaction methods, and a feedback mechanism to facilitate continuous service improvement. By automating key processes, the system is designed to reduce wait times, enhance order accuracy, and streamline delivery logistics for students and staff.

The study's significance lies in its potential to transform the campus dining experience by improving service efficiency, expanding the cafeteria's reach, and mitigating issues such as food waste and system exploitation. While the project focuses on a Minimum Viable Product (MVP) and is limited by time, resources, and testing environments, it provides a scalable model that can be adapted for other institutions. Ultimately, the successful implementation of this system is expected to increase customer satisfaction, optimize cafeteria operations, and drive higher revenues, solidifying the role of technology in modernizing food services within academic institutions.

CHAPTER ONE

INTRODUCTION

1.1 Background of the Study

The food industry has undergone significant transformations over the past few decades driven by technological advancements and the growing demand for convenience in consumer lifestyles (Meuter et al., 2000). The integration of digital solutions into food delivery has improved efficiency and redefined how food is ordered, prepared, and delivered. Web-based systems enable seamless communication between customers, restaurants, and delivery personnel, streamlining the entire process. This evolution supports fast-paced lifestyles and meets rising expectations for personalized, reliable service (Collier & Sherrell, 2010).

Initially, food delivery relied on phone orders which were time-consuming, error-prone, and often led to miscommunication, incorrect orders, and delays. These inefficiencies underscored the need for a more accurate and reliable system. The rise of the internet revolutionized the industry (Huan & Mahdin, 2023), enabling online platforms where users can browse menus, place orders, make payments, and track deliveries in real time. This shift has improved accuracy and allowed businesses to manage higher order volumes without compromising service quality.

Digital platforms have expanded consumer choice and improved operational efficiency. Users can explore multiple restaurants, compare prices, read reviews, and view menus from home (Kim et al., 2016), fostering competition and encouraging restaurants to enhance service and variety. Unlike traditional models limited by operating hours digital systems support 24/7 ordering. Companies like Uber Eats, DoorDash, Chowdeck, and Grubhub have capitalized on this shift by introducing features such as order tracking, meal customization, and customer reviews setting new standards for convenience and user experience.

The growing use of e-commerce and mobile applications has reshaped consumer behavior in the food sector (Tong & Samsudin, 2022). Modern consumers especially in urban areas with tight schedules, prioritize speed, convenience, and reliability. With smartphones central to daily life users now expect instant meal ordering, reinforcing the necessity of efficient web-based systems. This demand has embedded digital food access into modern culture.

The COVID-19 pandemic accelerated the adoption of digital food ordering as lockdowns and safety concerns made contactless delivery a preferred alternative to dining out (Al-Kandari et al., 2021). Digital platforms became essential for business continuity and public safety, highlighting their adaptability during crises. Many restaurants without prior delivery systems quickly adopted digital tools, a shift likely to have long-term effects on how food is consumed and delivered.

University campuses present unique challenges: students, lecturers, and staff often face tight schedules and limited access to meals, especially during peak hours or night classes. Traditional cafeteria visits result in long queues and restricted choices. These issues call for a technology-driven solution (Gumilang & Nurmala, 2018). A dedicated Food Ordering and Delivery System can provide fast, efficient, and accessible services tailored to campus needs. It supports smart campus initiatives, improves student welfare, and empowers student-run food businesses, particularly in contexts like Nigeria.

A well-designed web-based system includes restaurant listings, menu management, real-time tracking, secure payments, and feedback mechanisms. These features enhance user experience and trust. Real-time tracking increases transparency, while secure payment gateways ensure safe, efficient transactions. Customer reviews and ratings allow businesses to respond to feedback, improve service, and increase retention.

Given the digitalization of services and growing demand for efficient delivery, this study aims to develop a web-based system for Thomas Adewumi University. It will serve students, staff, cafeteria workers, and external customers, improving accessibility, user experience, and logistics (Huan & Mahdin, 2023). Unlike large commercial platforms, it will offer affordable, customizable solutions for small and medium-sized enterprises like university cafeterias, which often struggle with high commission fees and lack of control on third-party apps.

The proposed system will feature secure payments, real-time order management, and customized menus. Built on cloud infrastructure with mobile-responsive design, it will be scalable and adaptable to future needs. It will support meal customization, order tracking, and feedback submission offering a level of personalization often missing on broader platforms.

This study will not only develop the system but also evaluate its impact on dining experience by assessing order accuracy, delivery time, and user satisfaction. Findings will help identify success factors and areas for improvement. The research will also examine how the model can be adapted for other universities and small businesses, providing a framework for modernizing food services. The goal is to create a solution that enhances satisfaction, reduces errors, and streamlines operations for all stakeholders.

1.2 Statement of Problem

University campuses like Thomas Adewumi University are dynamic environments with large student populations, tight schedules, and high demand for efficient food services. However, the current cafeteria system relies on manual processes and outdated technology leading to inefficiencies, customer dissatisfaction, and financial losses. This legacy model fails to meet the expectations of a digitally literate user base and hinders operational excellence and service innovation.

The existing food ordering and delivery system at Thomas Adewumi University cafeteria faces several interrelated challenges that affect service quality and efficiency. These problems become more pronounced during peak hours when demand is highest. The key issues include:

1. Long Queues and Service Delays: During peak periods overcrowding at the cafeteria counter causes long waiting times. Students and staff often spend valuable minutes or even hours standing in line between classes or during short meal breaks. This inefficiency disrupts academic and work routines, reduces productivity, and discourages consistent use of the cafeteria.

2. Limited Menu Visibility and Accessibility: There is no real-time digital access to the daily menu forcing users to physically visit the cafeteria or depend on informal, often inaccurate information. This lack of transparency makes it difficult to plan meals or make informed choices. As a result, order placement is delayed, and customer experience suffers.

3. Inefficient and Insecure Payment System: The current system depends on bank transfers which are vulnerable to network failures, transaction delays, and potential fraud. Without integrated secure digital payment options transactions are unreliable and expose users to financial risks. This undermines trust and discourages digital adoption.

4. Absence of Effective Feedback Mechanisms: Feedback can only be given in person to the cafeteria manager which is inconvenient and limits participation. This restricts the volume and quality of input the cafeteria receives from users. Without structured real-

time feedback, it is difficult to identify and address issues like poor service or unpopular menu items.

5. Lack of Communication Channels: There is no system to notify users about promotions, new menu items, or special events. Students and staff remain unaware of discounts or updates reducing engagement and missed sales opportunities. Effective digital communication is essential for building customer relationships and driving patronage.

6. Food Waste Due to Poor Demand Forecasting: The lack of data-driven ordering systems leads to over-preparation of meals, resulting in significant food waste and financial loss. Accurate demand forecasting is hindered by the absence of historical order data and real-time monitoring.

7. Vulnerability to Fraud and System Exploitation: Weak user authentication and lack of order tracking allow students to exploit system flaws for free or unauthorized meals. These fraudulent activities compromise operational integrity and lead to revenue loss. A secure digital platform with verification features is needed to prevent abuse.

8. Limited Access for Broader Community: The current system only serves those physically present on campus, excluding off-campus students and members of the Oko community. This restricts the cafeteria's customer base and limits its revenue potential. Expanding access digitally would open new market opportunities.

9. Inefficient Time Utilization: On average, students and staff spend 9hours in a month accessing food from the cafeteria. This time spent ordering food, waiting in lines, and completing transactions is substantial, particularly when the students' time could be better spent on academic or work-related activities.

10. Lack of Revenue and Operations Tracking: There is no centralized system to monitor sales, inventory, deliveries, or customer satisfaction. This lack of data limits insight into financial performance and operational bottlenecks. A digital solution would enable better decision-making, resource planning, and continuous improvement.

1.3 Aim and Objectives

1.3.1 Aim

The primary aim of this project is to successfully improve service delivery for Thomas Adewumi University by designing and implementing a secure, user-friendly, web-based food ordering and delivery system for the campus cafeteria.

1.3.2 Objectives

To achieve this aim, the project sets the following specific objectives:

1. To design and develop a web-based platform that enables customers to browse menus, place food orders, and track their deliveries in real time.
2. To implement a robust cafeteria management system, allowing cafeteria to manage menus, process orders, and update order statuses efficiently.
3. To integrate a secure and seamless payment gateway that supports multiple payment methods, including debit cards, online banking, and mobile banking.

1.4 Research Questions

1. What are the current challenges faced by students and staff in accessing food services within the campus environment?

2. How can a digital food delivery system improve the efficiency and convenience of food ordering and delivery on campus?

3. What features should be incorporated into a campus food delivery platform to enhance user experience and vendor performance?

4. To what extent can a campus food delivery system reduce waiting times and improve customer satisfaction among users?

1.5 Scope of the Study

The scope of this project is defined by the functionalities and stakeholders involved in the development of a web-based food delivery system for Thomas Adewumi University cafeteria. This system is designed to enhance operational efficiency, improve customer satisfaction, and optimize food service logistics. The system will cater to three primary user groups:

I. Customers: Individuals who use the platform to browse menus, place orders, make payments, and track deliveries.

II. Cafeteria Operators: Businesses (cafeteria staff) who will manage food menus, receive orders, process payments, and update order statuses.

The system will feature the following key functionalities:

1. User Registration and Authentication: Users will create accounts and log in securely to ensure a personalized and secure experience. The system will include secure login and authentication mechanisms to protect user data.

2. Cafeteria and Menu Management: Cafeteria operators will be able to list food items, update menu prices, and modify food availability in real time. This functionality ensures that customers have access to the most current options and pricing.

3. Order Placement and Processing: Customers can browse menus, add items to a cart, and confirm orders. The system will streamline the process, reduce errors and wait times, and ensure a seamless transaction process.

4. Payment Processing: Multiple payment options will be integrated for customer convenience, including debit cards, online banking, mobile banking, options. This will address the payment issues mentioned in the problem statement and provide a secure, reliable solution for transactions.

5. Review and Feedback System: The system will include a mechanism for customers to provide feedback and rate the cafeteria and delivery services. This functionality will allow for continuous improvement in service quality, ensuring customer satisfaction is regularly evaluated and acted upon.

6. System Testing and Performance Evaluation: Extensive system testing will be conducted to ensure the platform's reliability, efficiency, and scalability. Performance evaluations will check the system's ability to handle high volumes of orders and ensure consistent operation during peak times.

1.6 Significance of the Study

The significance of this study lies in its potential to address existing challenges in the food delivery industry and improve overall service efficiency. The following key benefits highlight the importance of the project:

1.6.1 For Customers

1. Provides a user-friendly interface for browsing menus, ordering food, and tracking deliveries in real time.
2. Enhances order accuracy and efficiency, reducing errors that occur in traditional phone-based ordering.
3. Offers secure and convenient payment options, ensuring a hassle-free checkout process.
4. Reduces waiting times by providing real-time updates on order progress.

1.6.2 For cafeteria

1. Automates menu management, order processing, and payment handling, reducing manual workload.
2. Improves order management efficiency, enabling restaurants to handle multiple orders simultaneously.
3. Increase customer reach and sales potential by providing an online ordering system.

1.6.3 For Delivery Personnel

1. Optimizes delivery logistics, ensuring faster and more efficient food delivery.
2. Reduces delivery errors through an automated order assignment system.
3. Improves job satisfaction by providing a structured system for handling deliveries.

1.7 Limitations of the Study

Despite the anticipated benefits of the web-based food delivery system for Thomas Adewumi University cafeteria, the study faces several limitations:

Limited Time and Resources: Due to time constraints, the project will focus on core functionalities for the Minimum Viable Product (MVP). The system will prioritize essential features such as order placement, payment processing, menu access, and customer registration.

Internet Dependency: The system requires a stable internet connection for order placement, tracking, and payment processing. Poor or intermittent connectivity may disrupt the experience for customers and staff.

Limited Access to High Performance Servers: The study lacks access to high-performance servers for scalability testing under heavy loads. System performance during peak traffic may not reflect real-world conditions.

Core Functionalities in MVP: The initial phase will include only basic features like ordering, payment processing, and menu access. Advanced functions such as AI-powered recommendations, real-time delivery tracking, and third-party integrations (e.g., GPS or food safety APIs) are excluded from the MVP and may be added later.

Testing Environment Limitations: The testing setup may not fully mimic real cafeteria conditions, especially during peak hours. Without stress testing under actual high-traffic scenarios, some performance or usability issues may go undetected.

Limited Test User Pool: The number of test users may be too small to uncover all potential problems. A limited sample may not represent the diverse needs of the university community, affecting the reliability of feedback.

System Integration Limitations: The system will not integrate with third-party tools like food safety monitoring systems or real-time GPS tracking. Such features are beyond the current study's scope but could be explored in future versions.

Scope of Deployment: The system is designed specifically for Thomas Adewumi University's cafeteria. While it may be adaptable to other institutions, the study does not address scalability or customization for broader use in other universities or commercial services.

1.8 Definition of Terms

Web-based System: A web-based system is a software application that operates over the internet through a browser interface. Unlike traditional software, it can be accessed from any device with an internet connection and is hosted on a server without requiring local installation (O'Brien & Marakas, 2011).

Food Delivery System: A food delivery system enables the delivery of food from providers to consumers' locations. It includes ordering, payment, and tracking, often supported by web-based platforms (Kimes, 2011).

Campus: A campus is a defined area used by a university, including land, buildings, and facilities for academic, residential, and administrative purposes (Munich Business School, n.d.).

User Interface (UI): The User Interface (UI) is the visual part through which users interact with a system, designed to be intuitive and user-friendly (Interaction Design Foundation, n.d.).

Vendor: In food service, a vendor is a person or business that operates a food outlet and sells meals to the public (Genie AI, n.d., as cited from SEC Filings).

User Authentication: User authentication verifies a user's identity before granting system access, using passwords, biometrics, or multi-factor methods (Wagner, 2016).

Payment Gateway: A payment gateway processes online payments, authorizing transactions and securing payment data between customers and vendors (Shaikh & Padhye, 2019).

Order Management: Order management tracks orders from placement to fulfillment, including delivery and after-sales processes (IBM, n.d.).

Order Tracking: Order tracking allows customers to monitor their food order’s status and location in real time using technologies like GPS (Chen & Zhang, 2019).

Delivery Logistics: Delivery logistics involves planning and executing the transport of food from the vendor to the customer (WEZOM, n.d.).

Demand Forecasting: Demand forecasting predicts future food demand using historical data to reduce waste and ensure adequate supply (Seung & Yongsuk, 2018).

Menu Listing: A menu listing is a digital catalog of food items with descriptions, prices, and categories for customer browsing (Online Food Ordering System, ResearchGate PDF, n.d.).

Feedback System: A feedback system allows users to rate and comment on their experience to help assess service quality and guide improvements (Levy & Ray, 2017).

Automated Order Assignment: Automated order assignment allocates delivery tasks to personnel based on proximity, availability, and volume to improve efficiency (Yuan & Li, 2016).

RealTime Updates: Realtime updates deliver immediate information changes to users, such as order status or delivery progress (Liu & Zhang, 2018).

Customer Satisfaction: Customer satisfaction measures how well a company’s offerings meet or exceed customer expectations (ASQ, n.d.).

Fraud Prevention: Fraud prevention uses technology and protocols to stop unauthorized transactions or misuse of an online ordering platform (Singh & Bansal, 2020).

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

The development of web-based food ordering systems has its roots in the evolution of ecommerce and the digital transformation of food services. The traditional food ordering process, which involved phone calls or in person orders, has increasingly been replaced by digital platforms that provide faster, more efficient services. This shift began with early examples of online restaurant ordering systems, such as Pizza Hut's online ordering platform, which emerged in the late 1990s.

The rapid expansion of mobile devices and internet access has further fueled this trend, with many businesses, including small-scale food providers, adopting web-based platforms to cater to customers' growing demand for convenience and speed. Notably, the rise of platforms like Uber Eats and DoorDash in the 2010s significantly altered the landscape, making online food delivery more accessible and reliable. Studies like those by Chen et al. (2019) have shown how the technological adoption in food services has resulted in both operational efficiencies and changes in consumer expectations.

Over time, food delivery systems have evolved from simple order placement platforms to comprehensive systems that handle everything from menu management to real-time delivery tracking. As consumer preferences shifted towards on demand services, digital systems have had to become more responsive, secure, and user friendly. This transition is crucial for understanding the need for modern web-based delivery systems and their integration of payment systems, order management, and delivery logistics.

2.2 Review of Related Literature

1. Dynamic Mult Objective Balancing in Online Food Delivery Systems

Zheng et al. (2022) proposed a dynamic Mult objective balancing approach for online food delivery systems using a fuzzy logic system to identify supply–demand relationships. Their study addressed the complexities of balancing multiple objectives, such as minimizing delivery time and maximizing customer satisfaction, in a dynamic environment. The fuzzy logic system facilitated the identification of supply demand relationships, enabling more responsive and efficient delivery strategies. This approach demonstrated improved operational efficiency and customer satisfaction compared to traditional methods.

2. Factors Influencing the Use of Internet Based Food Delivery Service Apps

Thakur, Sharma, and Bhatt (2018) investigated the factors affecting the usage of internet-based food delivery service applications. Their research identified key determinants such as perceived ease of use, perceived usefulness, and trust in the service provider. The study highlighted that user-friendly interfaces and reliable services significantly enhance user adoption and continued usage of these applications. Additionally, the perceived usefulness of the app in terms of convenience and timesaving was found to be a critical factor in user satisfaction.

3. Modeling Stochastic Service Time in Complex OnDemand Food Delivery

Zheng et al. (2022) addressed the challenges of modeling stochastic service times in complex food delivery systems. Their study emphasized the importance of accurately predicting service times to optimize delivery routes and schedules. By incorporating stochastic modeling techniques, the research provided insights into handling uncertainties inherent in delivery processes, leading to more efficient and reliable food delivery services.

4. Optimization Framework for OnDemand Meal Delivery Systems

Paul, Rathee, Matthew, and Adusumilli (2020) developed an optimization framework for on demand meal delivery systems. Their research focused on optimizing delivery logistics to enhance efficiency and reduce operational costs. The framework incorporated factors such as delivery time windows, driver availability, and order batching to improve overall system performance. The study demonstrated that implementing such optimization techniques can lead to significant improvements in delivery efficiency and customer satisfaction.

5. Consumer Intentions to Use Online Food Delivery Systems in the USA

Gunden, Morosan, and DeFranco (2020) explored consumers' intentions to use online food delivery systems in the United States. The study identified factors such as convenience, perceived usefulness, and social influence as significant predictors of consumers' intentions to adopt online food delivery services. The findings suggested that enhancing the perceived value and ease of use of these platforms could drive higher adoption rates among consumers.

6. Geographical Information System for Web-based Food Delivery Services

Aprilliyanto, Fauzi, and Nuruzzaman (2015) designed a geographical information system (GIS) for web-based food delivery services. The system aimed to optimize delivery routes and improve service efficiency by integrating spatial data into the delivery process. The implementation of GIS technology enabled more accurate and efficient route planning, reducing delivery times and operational costs.

7. Fairness in Online Service with Multiple Servers: Application to Food Delivery

Singh, Kumar, and Chakraborty (2023) introduced a fairness-oriented approach to online services with multiple servers, specifically applied to food delivery systems. Their research proposed algorithms to ensure fair distribution of delivery tasks among available servers, addressing issues of workload imbalance and service quality. The study highlighted that incorporating fairness considerations can lead to more equitable and efficient service delivery.

8. Impact of Online-to-Offline Food Delivery Platforms on Healthy Food Choices

Zhang et al. (2024) investigated the impacts of online-to-offline (O2O) food delivery platforms on healthy food choices. The study analyzed how the availability and promotion of various food options on these platforms influence consumer dietary behaviors. Findings indicated that O2O platforms could contribute to the proliferation of "cyber food swamps," where unhealthy food options are more accessible and promoted, potentially leading to poorer dietary choices among consumers.

9. Intelligent Online Food Delivery System: Dynamic Model for Delivery Strategy and Tip Advice

Luo, Liufu, and Li (2020) developed an intelligent online food delivery system featuring a dynamic model to generate delivery strategies and provide tip advice to customers. The system utilized genetic annealing algorithms to optimize delivery queues and regression models to offer tipping recommendations based on desired waiting times. This approach is aimed at enhancing delivery efficiency and customer satisfaction by aligning service strategies with customer expectations.

10. Design and Implementation of a Food Delivery Notification System Using Django

Sharma (2023) presented the design and implementation of a food delivery notification system using the Django framework. The system aimed at improving communication between food delivery services and customers by providing real-time notifications about order status. The implementation demonstrated that leveraging web frameworks like Django can enhance the responsiveness and reliability of food delivery systems.

Collectively, these studies contribute to a comprehensive understanding of various aspects of web-based food delivery systems, including technological innovations, consumer behavior, operational optimization, and health implications. The insights gained from this body of literature can inform the development of more efficient, user friendly, and equitable food delivery platforms.

2.3 Evolution of Online Food Delivery Platforms

The progression of online food delivery (OFD) platforms reflects technological advancements, shifting consumer demands, and industry innovations. Over the past two decades, the OFD industry has evolved from basic website-based ordering to advanced mobile apps and, more recently, experimental drone-based delivery. These changes are driven by the growing need for convenience, speed, and improved customer experience. The rise of e-commerce and the digital economy has further accelerated this shift, as businesses aim to meet demand for on-demand services. Advances in logistics, data analytics, and artificial intelligence (AI) have enabled platforms to offer personalized, efficient services. The following subsections outline key phases in OFD evolution, showing how technology has transformed food service operations and enhanced efficiency.

2.3.1 Website-based Ordering

In the early days of online food delivery, restaurants created their own websites to accept orders. This required each business to build and maintain digital infrastructure, including online menus, payment systems, and order management tools. While it gave restaurants direct control over their digital presence, the model was costly and required ongoing technical updates. Many small and medium-sized restaurants struggled to adopt it due to limited resources. Without a unified platform, customers had to visit multiple websites to compare options, making the process inconvenient. Despite these drawbacks, website-based ordering was the first step toward digitalization in food services. It laid the foundation for future innovations and responded to early consumer interest in online transactions. As e-commerce grew, users began demanding more integrated solutions that could centralize multiple restaurant options.

2.3.2 Third Party Aggregators

The rise of third-party platforms like Grubhub, Uber Eats, and DoorDash transformed the OFD industry by consolidating multiple restaurants on a single interface. Consumers could now browse, compare, and order from various eateries through one app, simplifying the process significantly. These aggregators also handled logistics, providing delivery services and reducing the burden on restaurants. This model increased visibility for food vendors, especially smaller ones, helping them reach a broader audience. However, high commission fees—sometimes up to 30% per order—placed financial pressure on restaurants. Many found it difficult to sustain profitability under such costs. Additionally, restaurants lost direct access to customer data, limiting their ability to build marketing strategies. Despite these issues, the aggregator model remains dominant, with ongoing improvements aimed at boosting efficiency and user satisfaction (Li et al., 2020).

2.3.3 Mobile Application Based Delivery

The widespread use of smartphones and mobile internet ushered in the era of app-based food delivery. Mobile apps became the primary channel for ordering, offering features like real-time tracking, AI-powered recommendations, secure payments, and intuitive interfaces. Compared to websites, apps provide a more engaging experience, allowing users to place orders with minimal effort. They also use data analytics and machine learning to personalize suggestions based on past behavior, location, and preferences. Push notifications and loyalty programs further boost engagement and retention. The success of this model stems from increasing reliance on smartphones for everyday tasks and the expectation of instant service. However, challenges include the need for frequent updates, compatibility across operating systems, and concerns about data privacy (Ray et al., 2019). Despite these, mobile-based delivery dominates the market, with companies exploring AI chatbots and voice ordering to improve convenience.

2.3.4 Drone-based Delivery

With advances in logistics, drone-based delivery is emerging as a potential breakthrough in the industry. Companies like Amazon and Uber Eats have tested unmanned aerial vehicles (UAVs) to deliver food quickly by avoiding traffic and cutting delivery costs. Drones offer faster delivery times, reduced dependence on human couriers, and lower carbon emissions compared to traditional vehicles. However, the model faces major regulatory and technical obstacles. Aviation authorities impose strict rules on drone operations, limiting commercial use. Issues such as limited battery life, low payload capacity, and navigation challenges in urban areas hinder large-scale adoption. Still, pilot programs have shown promise, especially in rural and suburban regions where conventional delivery is inefficient. As regulations adapt and technology improves, drone delivery may become a viable mainstream option, offering unmatched speed and efficiency (Hwang et al., 2021).

2.4 Consumer Perceptions and Behavioural Intentions

Consumer perceptions play a crucial role in the adoption and continued use of online food delivery services. Various factors, including convenience, trust, service quality, and risk perception, influence customer decisions regarding OFD platforms. Understanding these factors helps businesses refine their services to enhance user satisfaction and loyalty. The following subsections explore the key drivers and barriers affecting consumer behavior in the OFD industry.

2.4.1 Convenience and Timesaving

One of the primary motivations for using OFD services is the convenience and timesaving benefits they offer. In today's fast paced society, consumers seek solutions that minimize effort and maximize efficiency. OFD platforms allow users to order meals from a variety of restaurants without the need to travel, reducing the time and effort required for meal preparation. This convenience is particularly appealing to working professionals, students, and individuals with busy schedules. Additionally, the ability to schedule deliveries, customize orders, and access real-time updates further enhances the appeal of these services. However, convenience is also influenced by factors such as user interface design, app responsiveness, and the efficiency of delivery personnel. Studies have shown that consumers are more likely to continue using an OFD service if the ordering process is seamless and delivery times are consistently met (Yeo et al., 2017).

2.4.2 Perceived Risk and Trust

Despite the convenience offered by OFD platforms, concerns regarding data security, food quality, and timely delivery present significant barriers to adoption. Consumers are often wary of sharing personal and financial information on digital platforms due to the risk of fraud and data breaches. Additionally, inconsistencies in food quality, delayed deliveries, and poor handling of complaints can erode trust in OFD services. To address these concerns, businesses must implement stringent security measures, transparent order tracking systems, and robust customer support mechanisms. Establishing partnerships with reputable payment gateways and adopting blockchain technology for transaction security can further enhance consumer confidence. Studies indicate that trust is a critical determinant of customer loyalty in the OFD industry, with reliable service providers enjoying higher retention rates (Hwang et al., 2019).

2.4.3 Service Quality and Satisfaction

Service quality plays a pivotal role in shaping consumer satisfaction and loyalty in the OFD industry. Key components of service quality include order accuracy, delivery speed, customer support, and overall user experience. Customers expect their orders to be delivered correctly, promptly, and in good condition. Any deviation from these expectations, such as incorrect orders, late deliveries, or poor packaging, can lead to dissatisfaction and negative reviews. Many OFD platforms have introduced features like real-time tracking, customer ratings, and AI driven support chatbots to improve service quality. Research suggests that platforms that prioritize service excellence enjoy higher customer retention rates and positive word of mouth marketing (Chai & Yat, 2019).

2.5 Technological Advancements in Online Food Delivery Systems

Technological advancements have significantly transformed the online food delivery (OFD) industry, improving efficiency, reliability, and customer satisfaction. The integration of artificial intelligence (AI), blockchain technology, and the Internet of Things (IoT) has enhanced the way food delivery platforms operate, leading to better service quality, optimized logistics, and secure transactions. These innovations address key challenges such as order accuracy, fraud prevention, and last mile delivery efficiency. The following subsections examine how these emerging technologies have shaped modern OFD systems.

2.5.1 Artificial Intelligence and Machine Learning

Artificial intelligence (AI) and machine learning (ML) play a critical role in optimizing food delivery operations, from order recommendations to logistics management. AI driven recommendation systems analyze customer preferences, past orders, and browsing behavior to suggest personalized meal options, enhancing the user experience and increasing order frequency. Machine learning algorithms also assist in demand forecasting, helping restaurants and delivery platforms predict peak ordering times and manage inventory accordingly.

Additionally, AI powered chatbots provide instant customer support, handling inquiries, processing refunds, and resolving complaints without human intervention. Another significant application of AI in OFD is route optimization, where algorithms analyze traffic patterns and weather conditions to determine the most efficient delivery paths. By reducing delivery times and minimizing operational costs, AI and ML enhance both business profitability and customer satisfaction (Ray et al., 2019).

2.5.2 Blockchain Technology

Blockchain technology provides a decentralized and transparent way to manage online transactions, helping to address issues like payment security, fraud prevention, and order verification. In online food delivery (OFD), blockchain can record every stage of the transaction—from order placement to payment and delivery confirmation—creating an immutable and traceable history. This transparency allows restaurants, delivery personnel, and customers to monitor and verify order progress without the risk of tampering or fraudulent activity. Smart contracts, a key feature of blockchain, can automatically process payments when specific conditions are met, such as delivery confirmation, reducing delays and disputes. The technology also enhances food traceability by enabling customers to access information about ingredient sourcing and handling, which is especially important for those with dietary needs or preferences for organic products. By increasing security, accountability, and trust, blockchain supports the reliability and long-term viability of OFD platforms (Li et al., 2020).

2.5.3 Internet of Things (IoT)

The Internet of Things (IoT) has revolutionized the logistics and monitoring aspects of online food delivery. IoT devices, such as GPS trackers and smart sensors, enable real-time tracking of delivery vehicles, allowing customers to monitor the estimated time of arrival of their orders. These devices also enhance food safety by tracking temperature, humidity, and handling conditions during transit, ensuring that perishable food items are delivered in optimal condition. Additionally, IoT integrated kitchen appliances allow restaurants to streamline food preparation by automating cooking processes and managing ingredient inventory more efficiently. The use of IoT in OFD not only improves operational efficiency but also enhances customer confidence in food quality and delivery reliability (Hwang et al., 2021).

2.6 Sustainability Implications of Online Food Delivery

The rapid growth of online food delivery services has raised concerns about their environmental and social sustainability. While OFD platforms offer convenience and economic benefits, they also contribute to environmental challenges such as excessive packaging waste, carbon emissions from delivery vehicles, and inefficient resource consumption. Furthermore, the gig economy model employed by many OFD services has implications for labor rights and fair compensation for delivery personnel. The following subsections explore these sustainability challenges and potential solutions.

2.6.1 Environmental Impact

Online food delivery raises significant environmental concerns, primarily due to the widespread use of single-use plastic packaging. Restaurants often package meals in disposable containers, plastic bags, and cutlery, contributing to growing plastic waste and pollution. The high frequency of deliveries also increases carbon emissions from motorized vehicles, worsening urban air quality and climate change. To address these issues, OFD platforms are adopting eco-friendly measures such as biodegradable packaging, reusable containers, and optimized delivery routes to reduce fuel consumption. Some companies have started using electric bikes and introduced carbon offset programs to lower their environmental impact. However, large-scale adoption of sustainable practices depends on industry-wide cooperation and supportive government policies that encourage greener alternatives (Li et al., 2020).

2.6.2 Social Implications

The gig economy model used by most food delivery platforms has led to debates about labor rights, job security, and fair pay for delivery workers. Many companies classify riders as independent contractors instead of employees, which excludes them from benefits like health insurance, paid leave, and social protections. While this model offers flexible working hours, it often results in unstable income and poor working conditions. In response, some countries have implemented labor laws requiring fair wages, accident insurance, and safety measures for gig workers. Concerns also exist around algorithmic management, where AI systems control task assignments and payment rates, often without transparency. These practices highlight the need for greater accountability and ethical standards in how delivery workers are managed (Li et al., 2020).

2.7 Theoretical Frameworks in Online Food Delivery Research

Understanding consumer behavior in the context of online food delivery (OFD) is pivotal for developing effective platforms. Several theoretical models have been employed to analyze factors influencing consumer adoption and usage of OFD services.

2.7.1 Technology Acceptance Model (TAM)

The Technology Acceptance Model (TAM) posits that perceived usefulness and perceived ease of use are primary determinants of technology adoption. In the realm of OFD, these constructs have been instrumental in understanding user acceptance. Studies have demonstrated that when consumers find OFD platforms user-friendly and beneficial, their likelihood of adoption increases (Davis, 1989).

2.7.2 Theory of Planned Behavior (TPB)

The Theory of Planned Behavior (TPB) suggests that behavioral intentions are influenced by attitudes, subjective norms, and perceived behavioral control. In OFD research, TPB has been utilized to assess how individual attitudes towards online ordering, societal influences, and perceived control over the ordering process affect consumer intentions (Ajzen, 1991).

2.7.3 Unified Theory of Acceptance and Use of Technology (UTAUT)

The Unified Theory of Acceptance and Use of Technology (UTAUT) integrates multiple models to explain user intentions and subsequent usage behavior. Factors such as performance expectancy, effort expectancy, social influence, and facilitating conditions are considered. In OFD studies, UTAUT has provided a comprehensive framework to evaluate consumer adoption behaviors (Venkatesh et al., 2003).

2.8 Empirical Studies on Online Food Delivery Systems

Empirical research provides valuable insights into the effectiveness, challenges, and future trends of online food delivery services. Studies have examined various aspects of OFD, including consumer behavior, operational efficiency, and sustainability. The following subsections summarize key empirical findings in these areas.

2.8.1 Consumer Behavior Studies

Empirical studies on consumer behavior in online food delivery reveal that convenience, variety, and ease of use are the primary drivers of adoption. Consumers appreciate the ability to browse diverse menu options, customize orders, and receive meals at their preferred locations. However, barriers such as delivery delays, high service fees, and concerns about food hygiene influence user satisfaction. Research indicates that platforms that offer personalized recommendations, loyalty programs, and high service reliability are more likely to retain customers. Additionally, studies highlight the importance of customer reviews and ratings in shaping purchase decisions, with positive reviews boosting order frequency and platform credibility (Yeo et al., 2017).

2.8.2 Operational Efficiency Research

Studies focusing on the operational aspects of OFD emphasize the role of technology in improving efficiency. AI driven demand forecasting helps platforms predict peak hours and allocate resources, accordingly, reducing waiting time and improving order fulfillment rates. Research also highlights the benefits of integrating automated dispatch systems and real-time tracking to enhance delivery coordination. However, operational challenges such as traffic congestion, delivery route inefficiencies, and restaurant order preparation times continue to impact service reliability. Some studies suggest that hybrid delivery models, combining human couriers with autonomous vehicles or drones, could improve last mile delivery efficiency (Ray et al., 2019).

2.8.3 Sustainability Research

Empirical Empirical studies on the sustainability of OFD platforms show a complex trade-off between economic gains and environmental harm. While online food delivery boosts local businesses and generates jobs, it also contributes to plastic waste and high carbon emissions. Policy measures like taxes on single-use packaging and incentives for green logistics could promote more sustainable practices in the sector. Research also indicates that AI-driven route optimization can reduce fuel use and lower environmental impact. Rising consumer preference for eco-friendly options gives platforms a chance to stand out by adopting sustainable initiatives (Li et al., 2020).

The online food delivery industry has advanced significantly due to technology, shifting consumer needs, and new business models. From basic websites to AI-powered apps, the sector continues adapting to demand for convenience, speed, and reliability. Yet key challenges around data security, service quality, and environmental sustainability persist. Future research should focus on innovative approaches to improve the resilience and long-term sustainability of OFD systems, ensuring lasting benefits for businesses, users, and society.

2.9 Conclusion

The literature on web-based food delivery systems encompasses a broad spectrum of topics including theoretical frameworks, technological advancements, consumer perceptions, and sustainability concerns. As the online food delivery (OFD) industry continues to evolve ongoing research is essential to address emerging challenges and leverage new opportunities for enhancing service delivery. Numerous studies have explored the adoption, design, and impact of digital food platforms providing valuable insights into user behavior, system efficiency, and technological integration.

However, despite the wealth of existing research, a critical gap remains in the development of localized web-based food ordering platforms tailored for Nigerian university campuses. Most studies focus on commercial third-party applications in urban or developed settings systems that are often too costly, complex, or infrastructure dependent for small-scale institutional use. Furthermore, there is limited academic and practical research on digital solutions that integrate secure payments and real-time tracking within a closed trusted academic environment.

This study addresses that gap by designing and implementing Tau-Bite, a scalable, low-cost web-based food ordering and delivery system specifically for Thomas Adewumi University. The project contributes to knowledge by demonstrating how context-specific digital solutions can improve campus food accessibility, reduce waste, and enhance service efficiency in resource-constrained environments.

CHAPTER THREE

SYSTEM DESIGN AND METHODOLOGY

3.1 Review of the Proposed System

The system was developed using the Laravel PHP framework. The structure follows Laravel's conventions, with clear separation of concerns across controllers, models, views, configuration, and routes. The application is designed for food ordering and delivery, with capabilities to handle Menus, Restaurants, Orders, and Carts.

3.1.1 Core Functionality

1. User Management & Authentication
    a. User Registration & Login: Users can create accounts, log in, and manage their profiles.
    b. Password Management: Features for password reset and change are present (ForgotPasswordController, PasswordController).
    c. Role Based Access: There are separate controllers for admin and regular users, indicating different access levels and dashboards.
2. Restaurant & Menu Management
    a. Restaurant Listings: Users can browse a list of restaurants, view details, and see available menus.
    b. Menu Management: Each restaurant can have multiple menu items, which users can view and add to their cart.
    c. Categories & Types: Menu items and restaurants are organized by categories and types for easier navigation.
3. Cart & Order Processing

a. Shopping Cart: Users can add menu items to a cart, update quantities, and remove items.

b. Order Placement: Users can place orders for selected menu items from one or more restaurants.

c. Order Tracking: Users and admins can view order details and statuses.

4. Payment System

a. Online Payments: Integration with payment gateways (e.g., Paystack) allows users to pay for their orders online.

5. Admin Panel

a. Dashboard: Admins have access to a dashboard for system overview and management.

b. Restaurant & Menu Management: Admins can add, edit, or remove restaurants, menu items, and categories.

c. Order Management: Admins can view, update, and manage all orders in the system.

d. User Management: Admins can manage user accounts, including viewing, editing, or disabling users.

3.2 System Requirements

The System requirements are as follows

3.2.1 Server Requirements

A. Software

1. Web Server:

   a. Apache (via laragon)

   b. Nginx

2. PHP:
   a. Version 8.1 or higher
3. Database:
   a. MySQL 5.7+
4. Composer:
   a. Latest stable version (for dependency management)
5. Node.js & npm:
   a. Node.js 12.x or higher
   b. npm 6.x or higher
6. Framework:
   a. Laravel 10
7. Payment Gateway:
   a. Paystack API
8. Development Environment:
   a. Laragon (local stack)

B. PHP Extensions Required:

1. OpenSSL
2. PDO
3. Mbstring
4. Tokenizer
5. XML

6. Ctype

7. JSON

8. BCMath

9. Fileinfo

10. GD or Imagick (for image processing)

3.2.2 Hardware Requirements

1. Minimum:
   a. CPU: 1 GHz or faster (dual core recommended)
   b. RAM: 1 GB (2 GB or more recommended for development)
   c. Disk Space: 250 MB for codebase + additional for database, logs, and uploads
2. Production (Recommended):
   a. CPU: 2+ cores
   b. RAM: 4 GB or more
   c. Disk Space: 10 GB or more (to accommodate user uploads, images, and logs)

3.3 System Architecture

A multi-tier web application following the Model-View-Controller (MVC) paradigm, leveraging the Laravel PHP framework. This architecture ensures a clear separation of concerns, maintainability, and scalability.

A. Presentation Layer

1. Views: Utilizes Laravel's Blade templating engine (resources/views/) to render dynamic HTML for both user-facing and admin interfaces.

a. User Views: For browsing restaurants, menus, and placing orders

b. Admin Views: For managing restaurants, menus, users, and orders.

2. Assets: Static resources (CSS, JS, images, fonts) are organized under public/, with separate directories for admin and site assets.

B. Application Layer

1. Controllers: Located in app/Http/Controllers/, controllers handle HTTP requests, orchestrate business logic, and interact with models.

   a. Admin Controllers: Manage administrative functions (e.g., AdminController, CategoriesController).

   b. User Controllers: Handle user actions such as authentication, cart management, and payments.

2. Middleware: Implements request filtering, authentication, CSRF protection, and other cross-cutting concerns (app/Http/Middleware/).

C. Domain/Business Logic Layer

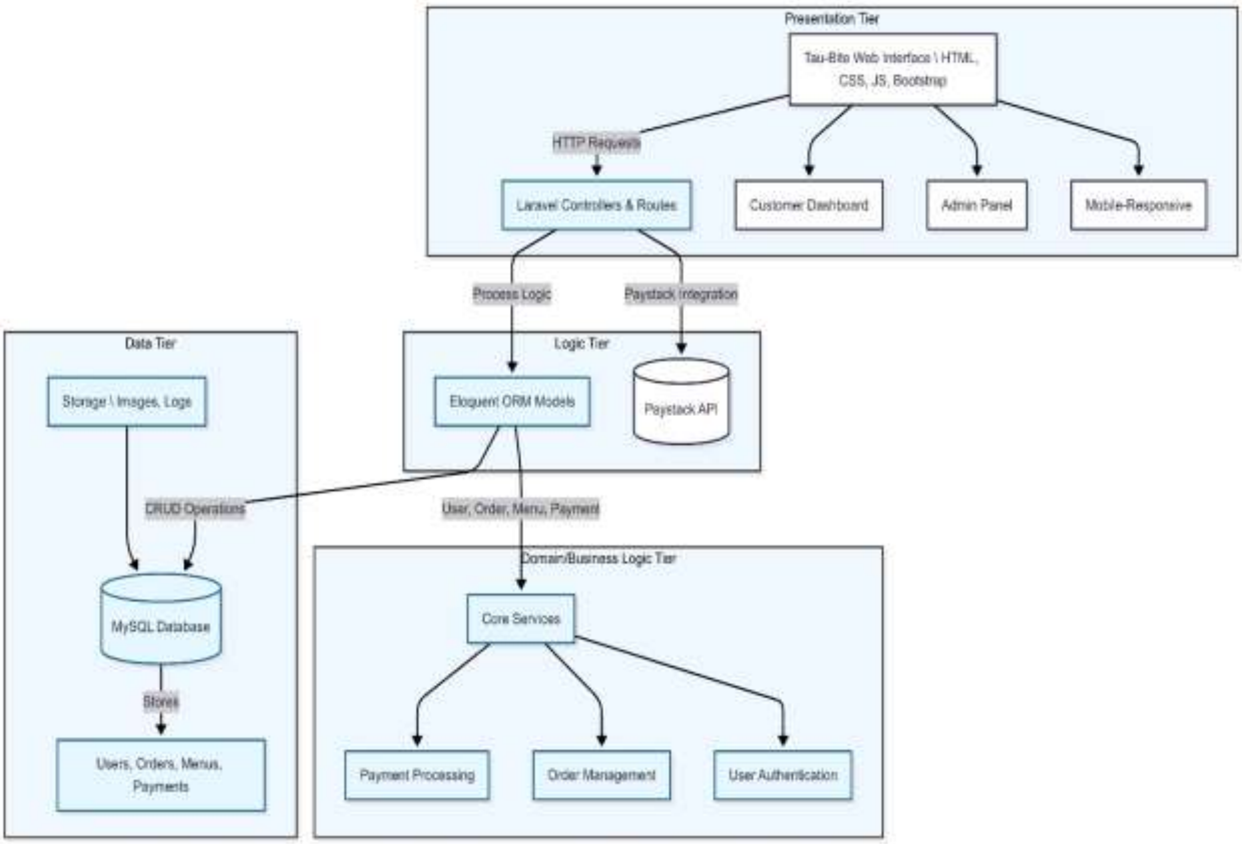
1. Models: Eloquent ORM models (app/Models/) represent core business entities such as User, Order, Menu, and Transaction. Models encapsulate business rules and data access logic.

2. Services & Helpers: Additional business logic and utility functions are provided via service providers and helper files.

D. Data Access Layer

1. Database Migrations & Seeders: Located in database/migrations/ and database/seeders/, these files define and initialize the database schema and seed data.

2. Configuration: Database and service configurations are managed in the config/ directory.



System Architecture

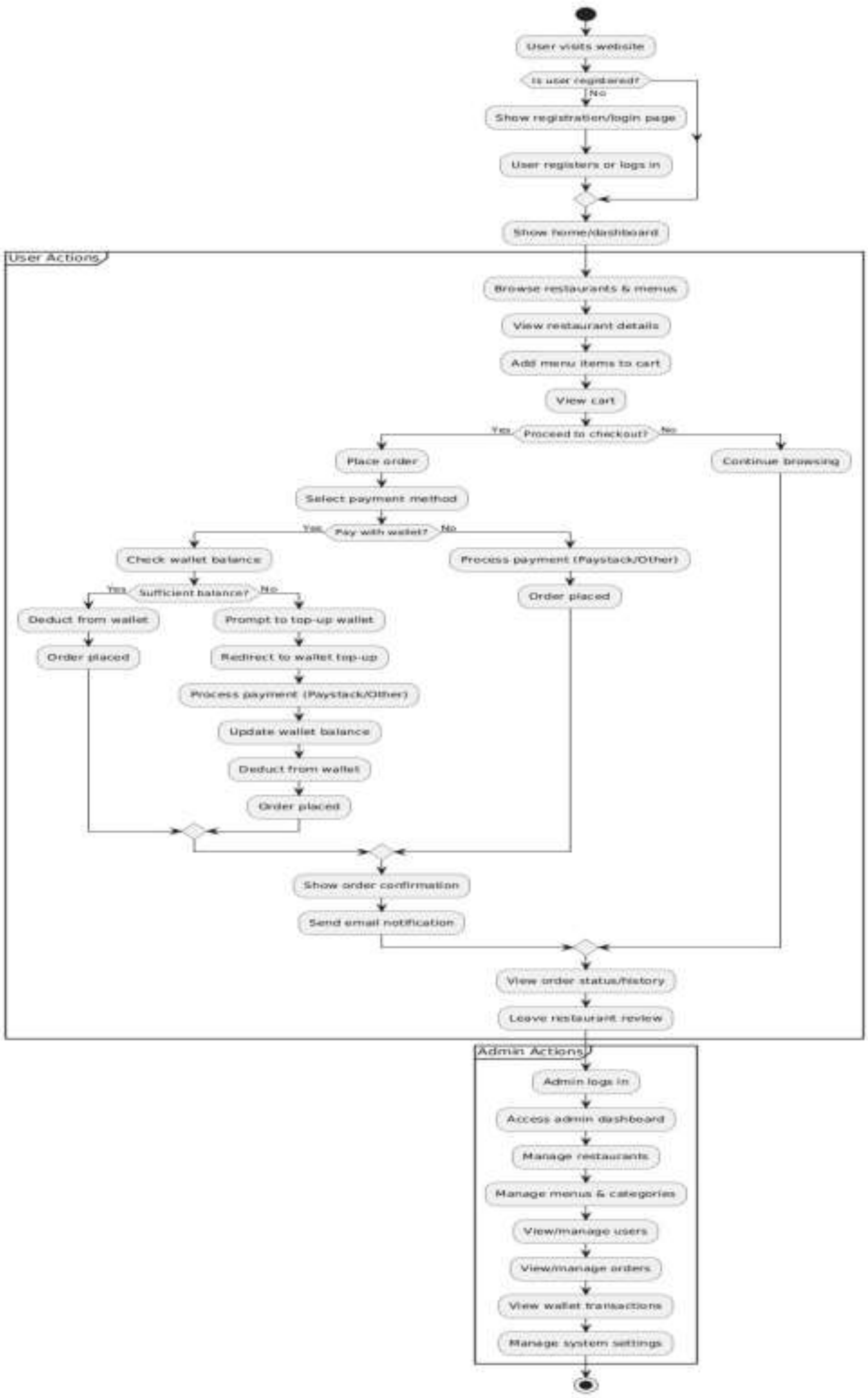


Figure 3.1 Flowchart of Tau-Bite

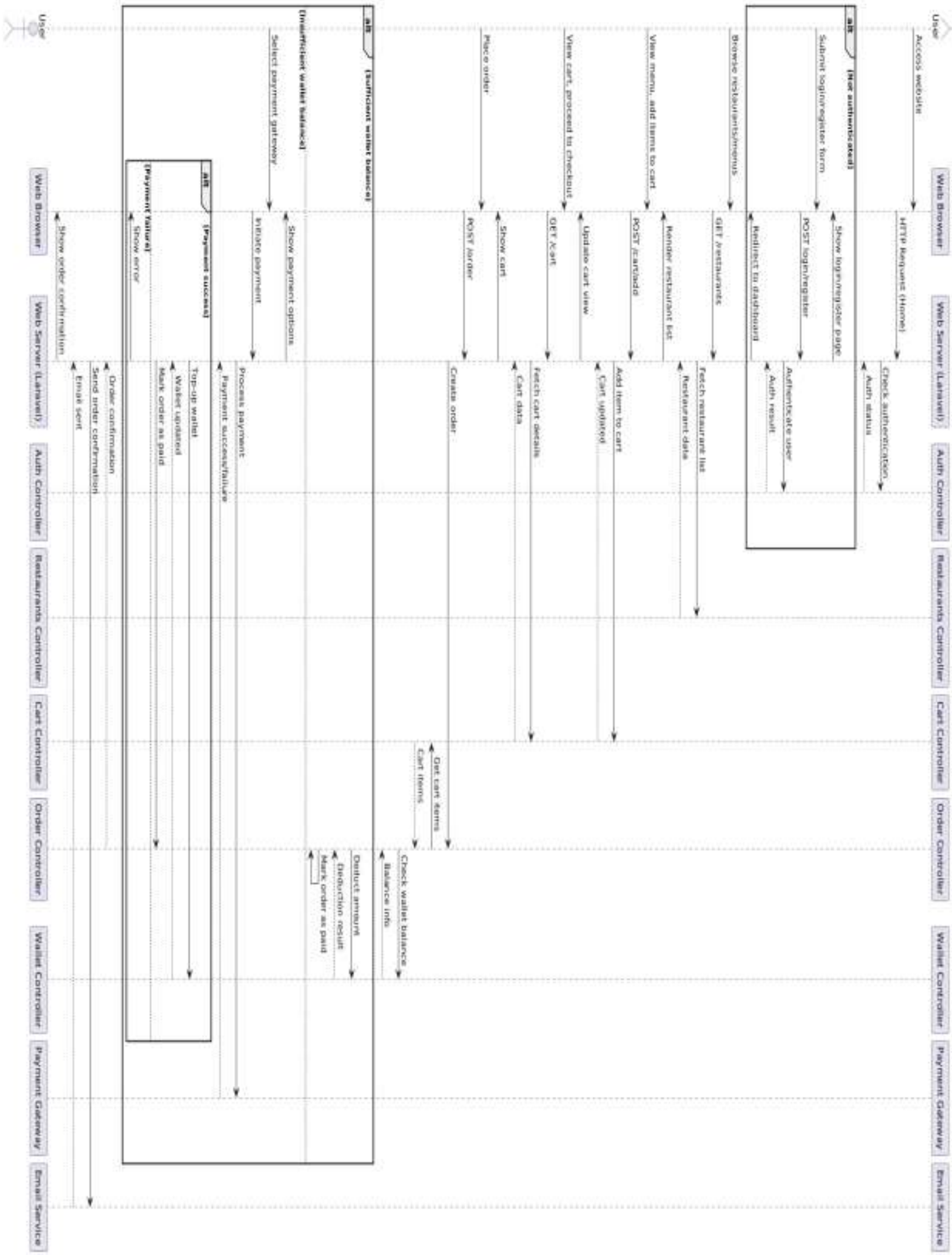


Figure 3.2 Sequence of Tau-Bite

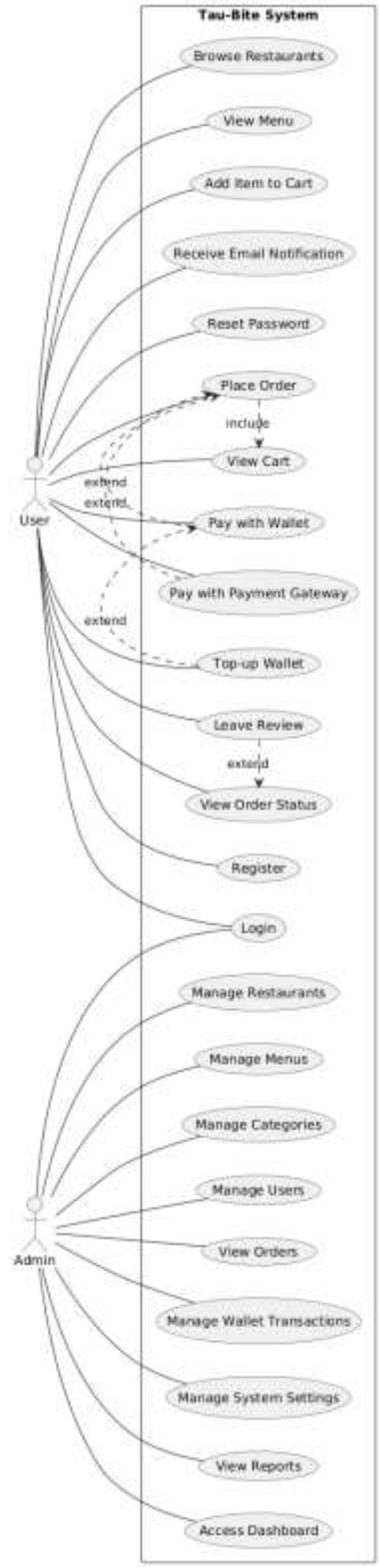


Figure 3.3 Case of Tau-Bite

3.4. System Development Methodology

The application appears to have been developed using an Agile software development methodology. Agile is a widely adopted, iterative, and incremental approach that emphasizes flexibility, collaboration, and customer feedback throughout the development lifecycle.

Key Characteristics:

1. Iterative Development: The system is built in small, manageable increments (sprints), allowing for regular reassessment and adaptation of requirements and solutions.
2. Continuous Feedback: Stakeholders and end-users are engaged throughout the process, providing feedback on each increment, which helps ensure the product meets business needs and user expectations.
3. Cross-functional Teams: Development involves collaboration among developers, designers, testers, and business stakeholders, ensuring a holistic approach to problem-solving.
4. Prioritized Backlog: Features and enhancements are managed in a prioritized backlog, allowing the team to focus on delivering the most valuable functionality first.

3.4.1 Phases of Development

1. Requirements Gathering & Analysis
    a. Initial requirements are collected from stakeholders.
    b. User stories and use cases are defined to capture functional and non-functional needs.
2. Planning
    a. The project is broken down into sprints or iterations.
    b. Tasks are estimated and assigned based on priority and complexity.

3. Design

   a. System architecture is designed using the MVC pattern.

   b. Database schema, user interface wireframes, and API contracts are defined.

4. Implementation

   a. Features are developed incrementally with each sprint delivering a potentially shippable product increment.

   b. The code is written following Laravel best practices ensuring maintainability and scalability.

5. Testing

   a. Automated and manual testing are performed at each iteration.

   b. Unit, feature, and integration tests are created to ensure code quality and reliability.

6. Deployment

   a. The application is deployed to staging and production environments using version control and deployment tools.

   b. Continuous Integration/Continuous Deployment (CI/CD) pipelines.

3.5. Programming Languages and Tools

Programming Languages:

1. PHP

   a. Primary backend language.

   b. Used for business logic, controllers, models, and Laravel framework code.

2. JavaScript

a. Used for frontend interactivity and dynamic behavior.

b. Present in custom scripts and possibly for asset compilation.

3. HTML

a. Used in Blade templates for structuring web pages.

4. CSS / SCSS / SASS

a. Used for styling the frontend.

SCSS and SASS files are present for advanced styling and theming.

Backend

1. Laravel

a. PHP web application framework (MVC architecture).

2. Composers

a. Dependency manager for PHP.

Frontend

1. Blade: Laravel's templating engine for PHP.

2. Bootstrap: CSS framework for responsive design (inferred from asset files).

3. jQuery: JavaScript library for DOM manipulation and AJAX (present in assets).

4. Font Awesome: Icon library (present in assets).

5.

Database

1. MySQL: Relational database management system (inferred from Laravel's default and migration files).

3.6. Database Design

A relational database design, most likely using MySQL or MariaDB, is standard with Laravel applications. The schema is normalized and organized around the core business entities of a food ordering and restaurant management platform.

3.6.1 Core Entities and Tables

1. Users and Authentication
   a. Users: Stores user account information, including authentication credentials, profile data, and role (e.g., customer, admin).
   b. Password_resets: Supports password recovery workflows.
   c. Personal_access_tokens: Manages API tokens for secure access.
2. Restaurants and Menus
   a. Restaurants: Contains details about each restaurant (name, address, contact, etc.).
   b. Menu: Stores menu items offered by restaurants, including name, description, price, and associated restaurant.
   c. Categories: Organizes menu items into logical groups (e.g., appetizers, main courses).
   d. Types: Classifies restaurants or menu items by type (e.g., cuisine, service style).
3. Orders and Cart
   a. Cart: Temporary storage for items a user intends to order.
   b. Restaurant_order: Records finalized orders, linking users, restaurants, and menu items.
   c. Restaurant_review: Stores user reviews and ratings for restaurants.
4. Payments

a. Payments: Tracks payment transactions, including method, status, and references to orders.

b. Transactions: General ledger for all financial transactions, possibly including direct payments.

5. System and Miscellaneous

a. Settings: Stores application-wide configuration options.

b. Failed_jobs: Logs failed background jobs for troubleshooting.

3.6.2. Relationships and Integrity

1. One-to-Many:

    a. A restaurant has many menu items.

    b. A user can have many orders, reviews, and transactions.

2. Many-to-Many:

    a. Orders may involve multiple menu items (handled via pivot tables or order details).

3. Foreign Keys:

    a. Enforced between related tables (e.g., menu.restaurant_id references restaurant).

4. Indexes:

    a. Primary keys on all tables with additional indexes on foreign keys and frequently queried columns for performance.

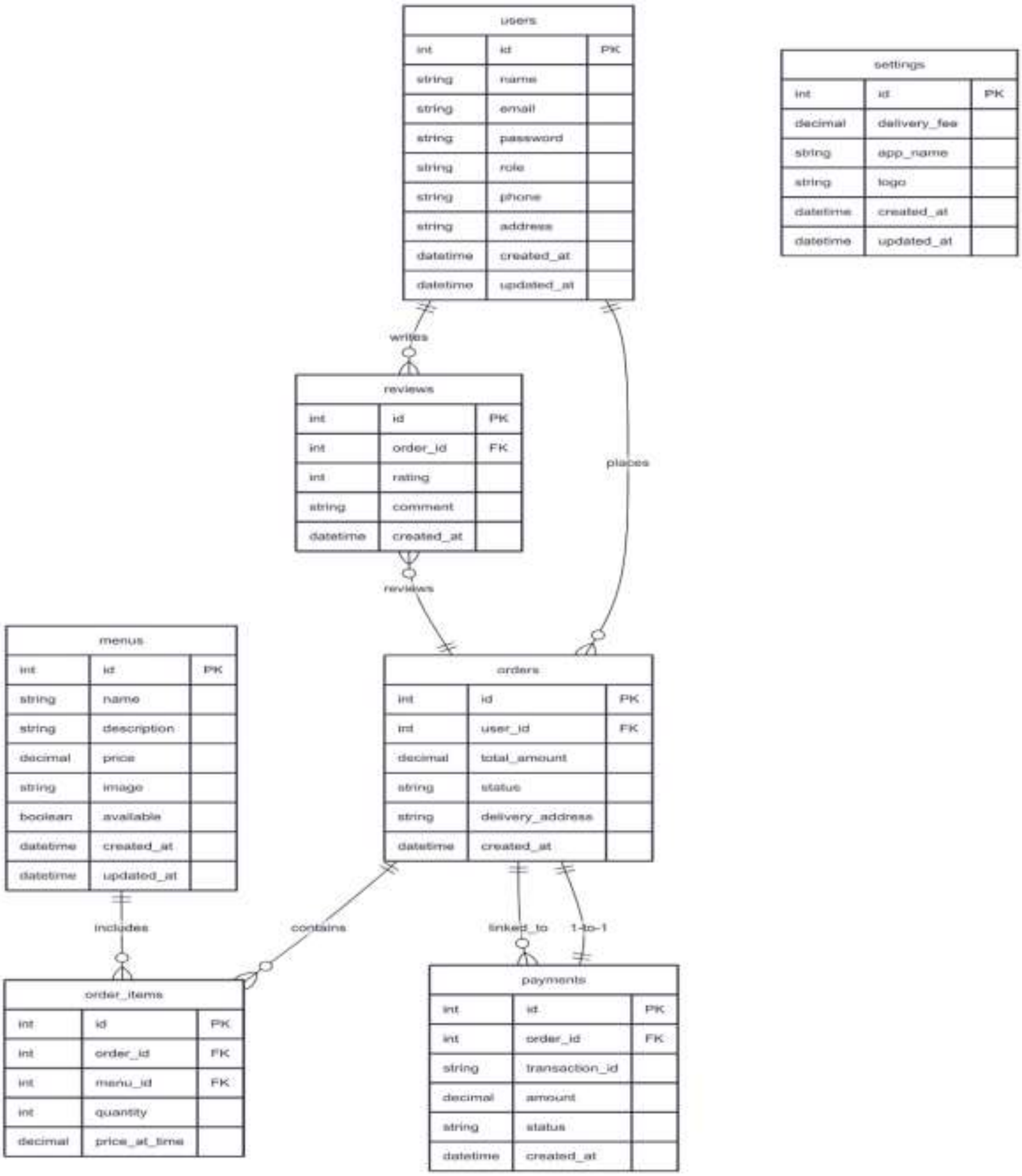


Figure 3.4 Entity-Relationship (ER)

3.7 System Security

Security is a critical aspect of any web-based application, especially one that handles user authentication, personal data, and financial transactions. The Tau-Bite food ordering and delivery system incorporates multiple layers of security to protect user information, prevent unauthorized access, and ensure data integrity throughout the system.

The following security measures have been implemented:

1. Authentication and Session Management
   a. Secure User Authentication: The system uses Laravel's built-in authentication scaffolding with hashed passwords (using Bcrypt) to ensure that plaintext passwords are never stored in the database.
   b. Email Verification: New users must verify their email addresses before gaining full access, reducing the risk of fake accounts.
   c. Session Protection: Laravel manages user sessions securely, with automatic session expiration and protection against session fixation attacks.
2. Cross-Site Request Forgery (CSRF) Protection
   a. Laravel automatically generates CSRF tokens for every active user session.
   b. These tokens are embedded in all forms and validated on submission, preventing malicious third-party websites from submitting requests on behalf of authenticated users.
3. Role-Based Access Control (RBAC)
   a. The system supports three user roles: Customer, Cafeteria Operator, and Administrator.
   b. Middleware is used to enforce access control, ensuring that users can only access features relevant to their role.

I. Example: Only administrators can manage users or view system settings.

4. Data Encryption and Secure Communication

   a. Environment Variables: Sensitive data such as database credentials, API keys, and Paystack secrets are stored in the .env file, which is excluded from version control.

   b. HTTPS Recommendation: While the system was tested locally over HTTP, it is designed to be deployed over HTTPS in production to encrypt data in transit.

   c. Paystack Integration: All payment transactions are processed through Paystack's secure API, which uses SSL/TLS encryption and tokenization to protect cardholder data.

5. Input Validation and Sanitization

   a. Laravel's form request validation is used to validate all user inputs before processing.

   b. This prevents common vulnerabilities such as SQL Injection and Cross-Site Scripting.

   c. Example: Order placement requires validated input for delivery address, quantity, and user ID.

6. SQL Injection Prevention

   a. The system uses Eloquent ORM and parameterized queries, which automatically escape user input, eliminating the risk of SQL injection attacks.

   b. Raw SQL queries are avoided unless necessary, and when used, they are parameterized.

7. Error Handling and Logging

   a. Debug mode is disabled in production (via .env configuration) to prevent sensitive system information from being exposed in error messages.

   b. Laravel's logging system records critical events (e.g., failed login attempts) for monitoring and auditing purposes.

8. Data Privacy and GDPR Compliance (Principles)

a. Users have control over their personal data, including the ability to update or delete their profiles.
b. Data is only collected for functional purposes (e.g., order delivery) and is not shared with third parties.
c. While the system is not subject to GDPR, it follows its principles by minimizing data collection and ensuring user consent.

CHAPTER FOUR

IMPLEMENTATION AND TESTING

4.1. Overview of the Overall System Development

The system is designed to serve both end-users (customers) and administrators, providing seamless restaurant discovery, browsing, order placement, payment processing, and administrative oversight. The system was developed with modular controllers, models, and views for all major features: authentication, restaurant/menu management, cart and order processing, and payment integration, reviews, and admin dashboard. Integrated third-party services such as payment gateways (e.g., Paystack) and email notifications.

4.2. System Implementation

The implementation of the system was guided by modular design principles, leveraging Laravel's MVC (Model-View-Controller) architecture to ensure maintainability, scalability, and clear separation of concerns. The development process was iterative, with features built, tested, and refined in cycles to ensure alignment with business requirements and user needs.

4.2.1 Backend Implementation

1. Controllers
   a. Controllers were developed to handle HTTP requests, orchestrate business logic, and interact with models.
   b. Separate controllers were implemented for user-facing features (e.g., authentication, cart, orders, payments) and administrative functions (e.g., restaurant, menu, user, and order management).
2. Models and ORM

a. Eloquent ORM models represent core business entities such as User, Restaurant, Menu, Order, and Transaction.
b. Models encapsulate data access logic and relationships, enabling efficient querying and data manipulation.

3. Middleware
    a. Middleware components were implemented for authentication, authorization, CSRF protection, and request validation, ensuring secure and reliable request handling.

4. Services and Helpers
    a. Service providers and helper functions were used to encapsulate reusable business logic, such as payment processing, and notification delivery.

5. Database Integration
    a. Database schema was defined and managed using Laravel migrations, ensuring version control and repeatability.
    b. Seeders were used to populate the database with initial data for development and testing.

4.2.2 Frontend Implementation

1. Blade Templating
    a. The user interface was constructed using Laravel's Blade templating engine, enabling dynamic rendering of content and layouts.
    b. Views were organized for both user and admin interfaces, with reusable partials for headers, footers, and navigation.

2. Styling and Responsiveness
    a. CSS, SCSS, and Bootstrap were used to ensure a responsive and visually appealing design.

b. JavaScript and jQuery provided interactivity and enhanced user experience, such as dynamic cart updates and form validation.

3. Asset Management

a. Frontend assets were managed and compiled using Node.js, npm, and Vite, optimizing performance and maintainability.

4.3. Application Manual

1. Getting Started

    a. Visit the Tau-Bite website using your web browser.

    b. On the homepage, you can browse featured cafeteria/vendors and menu items.

    c. For the best experience, create an account or log in.

2. Account Management

    a. Registering a New Account

        i. Click Register or Sign Up on the homepage.

        ii. Fill in your details (name, email, password, etc.).

        iii. Submit the form. You'll receive a confirmation email.

        iv. Click the link in your email to activate your account.

    b. Logging In

        i. Click Login.

        ii. Enter your email and password.

        iii. Click Sign In to access your dashboard.

    c. Editing Your Profile

        i. After logging in, go to my account section to update your information.

ii. Browsing cafeteria/vendors & Menus

iii. Use the cafeteria tab to view all available cafeteria/vendors. iv. Click on a cafeteria/vendor to see its menu, address, and reviews.

3. Browse menu categories (e.g., Noodles, Amala, rice) for easy navigation.

a. Using the Cart

i. Adjust the quantity as needed.

ii. Once you are done, click order now.

iii. To view your cart, click my account then click on my order at the top of the page.

iv. You can remove and cancel orders.

b. Placing an Order

i. click order now.

ii. Confirm your delivery address and contact details. iii. Choose your preferred payment method (Payment Gateway).

iv. Click Place Order to complete your purchase.

4. Making Payments

a. Paying with Payment Gateway

i. Select payment gateway (e.g., Paystack).

ii. Select preferred payment method (e.g., card, bank transfer, USSD). iii. Enter your payment details and follow the on-screen instructions.

iv. Upon successful payment, you'll receive an order confirmation.

5. Tracking Orders

a. Go to My Orders in your dashboard to view all your orders.

b. Check the status of each order (e.g., Processing, Out for Delivery, Completed).

c. Click on an order to see detailed information.

4.4. Application Screenshots

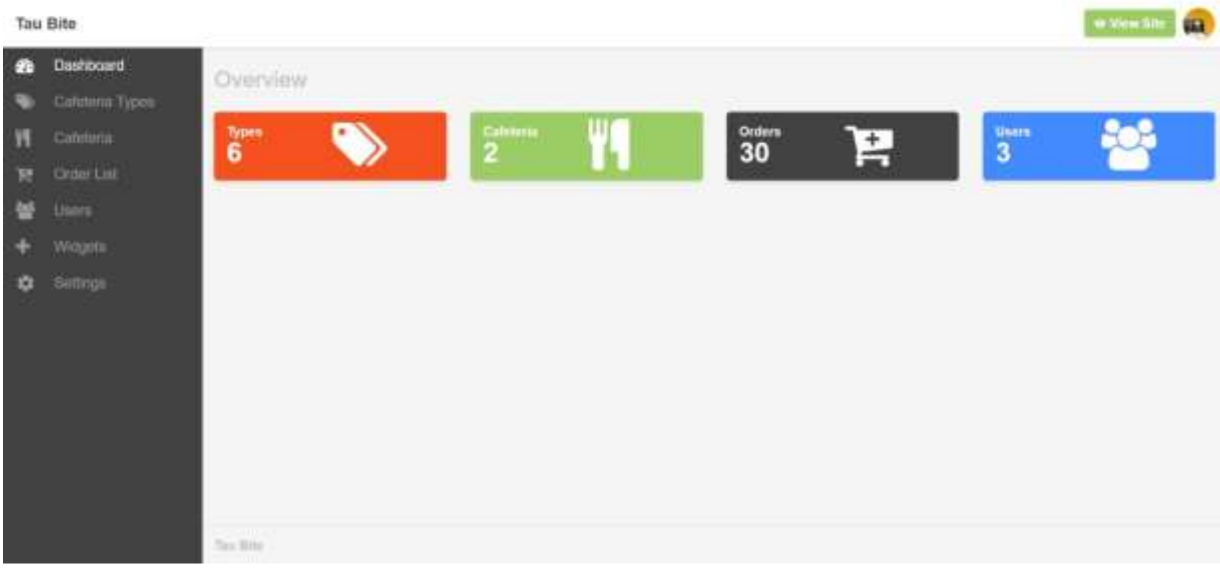


Figure 4.1 Application Dashboard

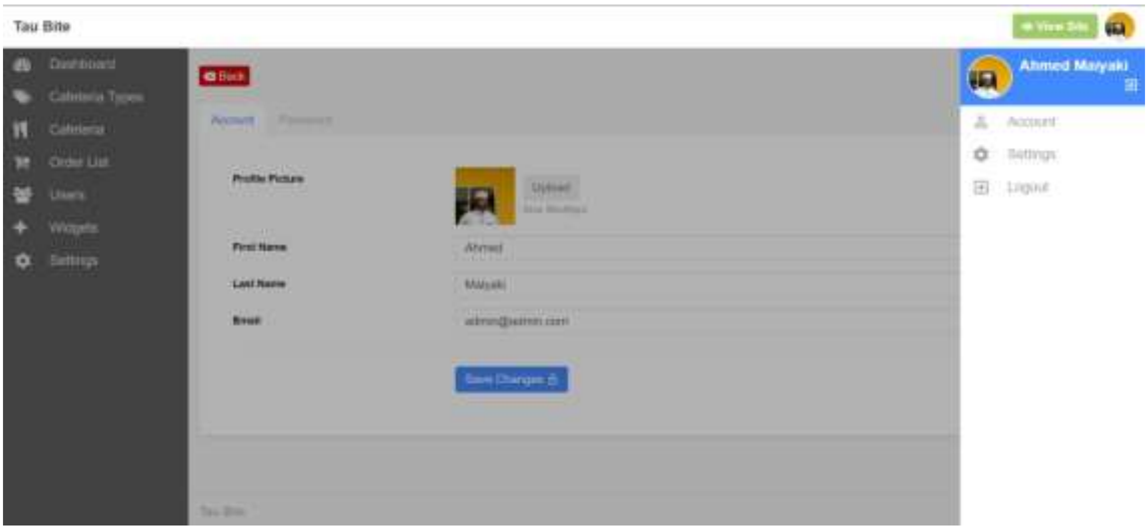


Figure 4.2 Edit Profile

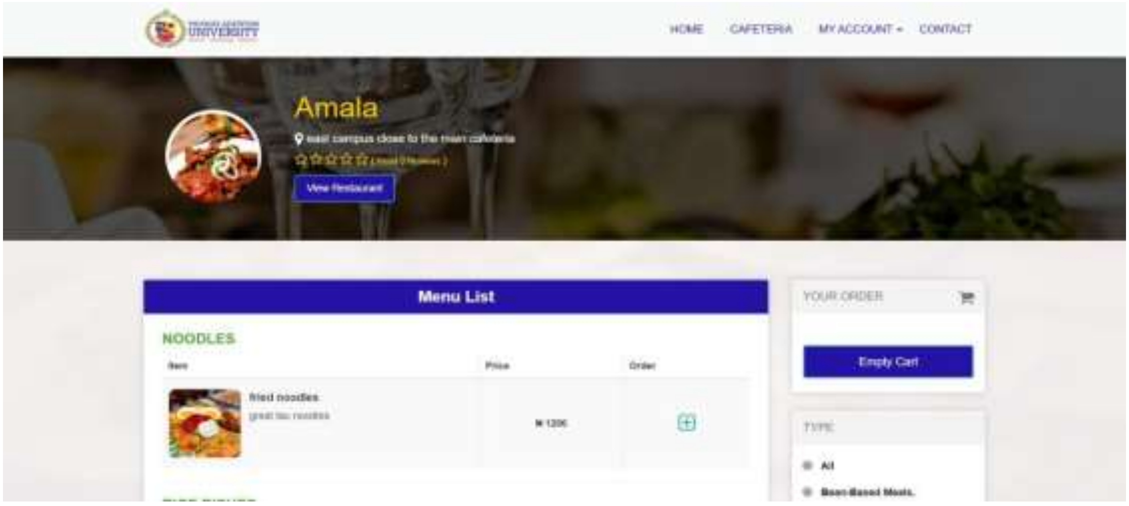


Figure 4.3 Cafeteria Menu

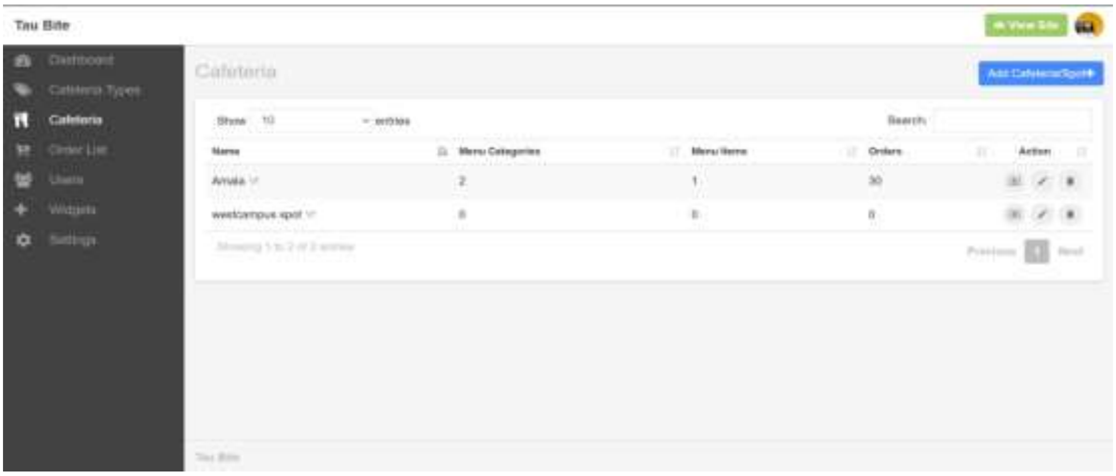


Figure 4.4 Admin Menu

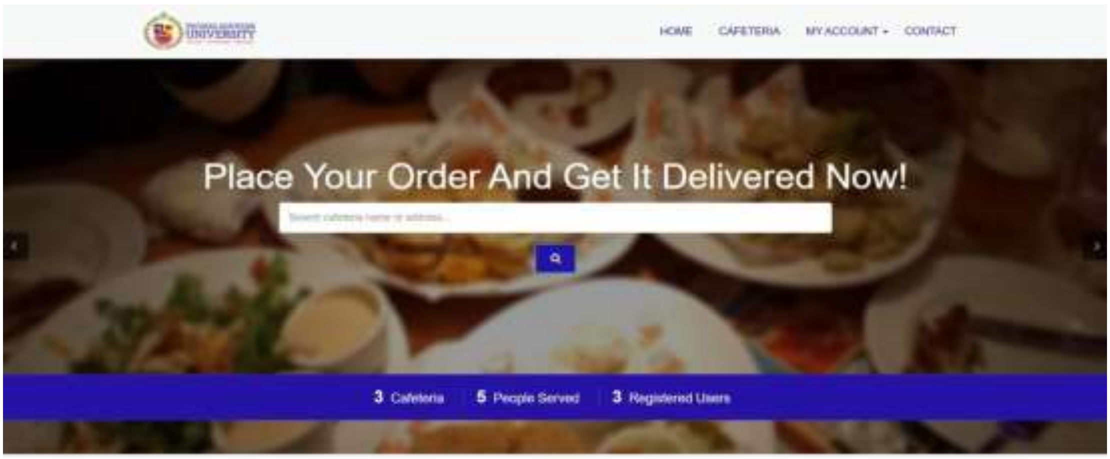


Figure 4.5 Home Page

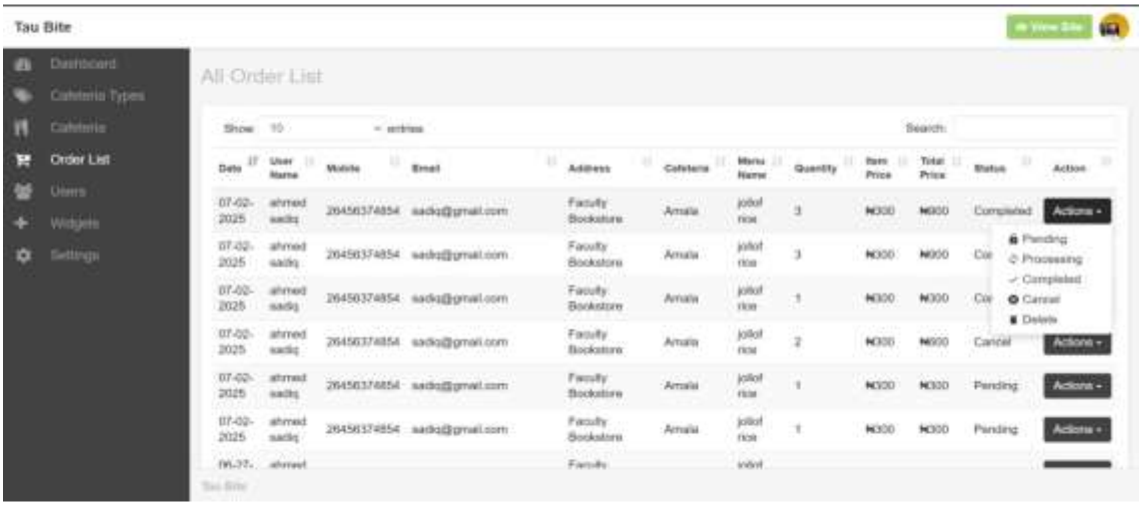


Figure 4.6 Admin Order List

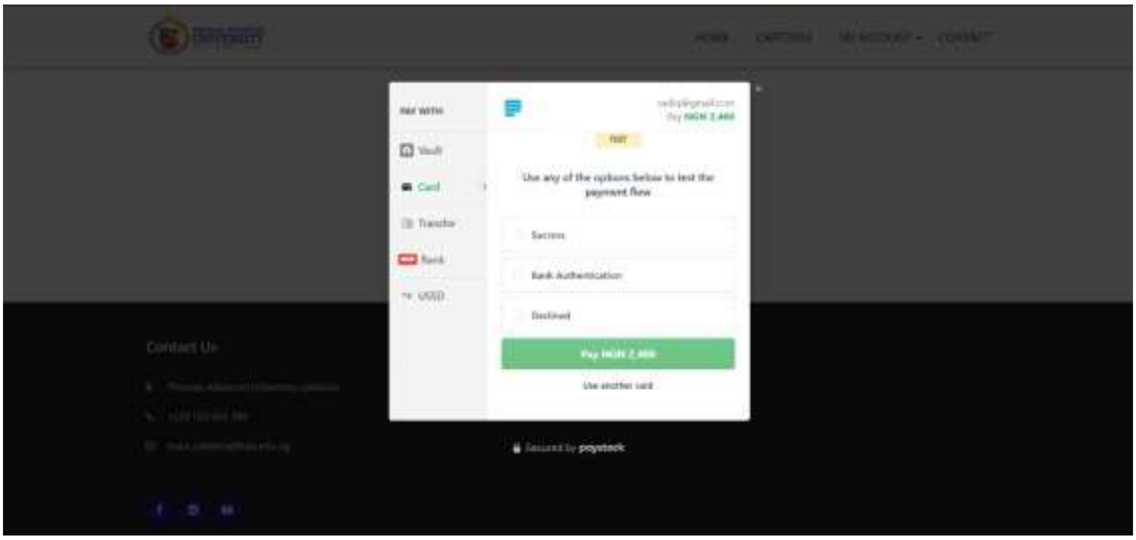


Figure 4.7 Payment Gateway

4.5 Testing Strategies

To ensure the reliability, functionality, and usability of the Tau-Bite system, a comprehensive testing approach was adopted encompassing Unit Testing, Integration Testing, and User Acceptance Testing (UAT). These strategies were selected to validate the system at different levels from individual components to real-world user experience.

4.5.1 Unit Testing

Unit testing focused on verifying the correctness of individual components particularly Laravel controllers, models, and helper functions. Each unit was tested in isolation to ensure it performed as expected.

a. Controllers: Tested for correct HTTP response codes, redirections, and data handling.
b. Models: Verified Eloquent relationships (e.g., User → Orders) and attribute mutators.
c. Helper Functions: Validated logic for order total calculation, delivery fee computation, and status updates.

Although Laravel supports automated unit testing via PHP Unit, this project used manual functional testing due to time and resource constraints. Each function was tested in the development environment to confirm expected behaviour.

4.5.2 Integration Testing

Integration testing verified that combined modules work together seamlessly. This is critical in a system like Tau-Bite where multiple components interact such as the cart, order, payment, and notification systems.

Key integration points tested:

a. Cart to Order Flow: Adding items to cart → Checkout → Order creation
b. Order to Payment: Order placement → Redirect to Paystack → Payment confirmation → Order status update
c. Admin to User Interaction: Admin updates order status → Customer receives real time update

These tests ensured that data flows correctly between components and that the system behaves as a unified whole.

4.5.3 User Acceptance Testing (UAT)

User Acceptance Testing involved 10 real user's students and staff who interacted with the system in a controlled environment. The goal was to evaluate usability, satisfaction, and realworld functionality.

Participants were asked to:

1. Register and log in
2. Browse the menu
3. Add items to cart
4. Place an order
5. Make a payment
6. Track the order
7. Provide feedback

Feedback was collected via observation and a short survey, focusing on ease of use, clarity of interface, and overall experience.

4.6 Test Cases and Results Table

TC-01	User Registration	New user can register with valid email and password	User registered successfully; verification email sent	Pass
TC-02	User Login	Registered user can log in with correct credentials	Login successful; redirected to dashboard	Pass
TC-03	Add to Cart	User can add menu items to cart and adjust quantity	Items added; cart updates dynamically	Pass
TC-04	Checkout Process	User can proceed to checkout after filling delivery details	Checkout form submits; user redirected to payment page	Pass
TC-05	Paystack Payment	Payment is processed and order confirmed	Payment successful; order status updated to "Processing"	Pass
TC-06	Order Tracking	User can view order status in dashboard	Order status displayed correctly	Pass
TC-07	Admin Login	Admin can access admin panel with admin credentials	Access granted; admin dashboard loaded	Pass
TC-08	Menu Update (Admin)	Admin can edit or delete menu items	Changes saved; updated menu visible to users	Pass
TC-09	Role-Based Access	Customer cannot access admin panel	Access denied; redirected to user dashboard	Pass

CHAPTER FIVE

SUMMARY, CONCLUSION AND RECOMMENDATION

5.2. Summary

This study focused on the design, development, and evaluation of a web-based food ordering and delivery system tailored for Thomas Adewumi University. The project was motivated by the persistent challenges associated with the current manual system including long queues, inefficient payment processes, limited menu visibility, food waste, and vulnerability to fraud.

The system, named Tau-Bite is a comprehensive web-based platform built using the Laravel PHP framework and follows the Model-View-Controller (MVC) architectural pattern. Its development adhered to modern software engineering principles including modular design, separation of concerns, and secure coding practices. The Agile methodology was adopted to enable iterative refinement and continuous feedback from stakeholders.

The backend leverages Laravel's Eloquent ORM for robust data modelling, while the frontend uses Blade templating, Bootstrap, and JavaScript to deliver a responsive and interactive user experience. Key features include secure user authentication, role-based access control, dynamic menu management, integration with the Paystack payment gateway, and a real-time order tracking system. The database schema is normalized and managed through Laravel migrations ensuring data integrity, maintainability, and scalability.

Security best practices were implemented throughout the system including CSRF protection, password hashing, and environment-based configuration. The codebase is structured for extensibility, and both automated and manual testing were conducted to ensure reliability and support future deployment.

Overall, Tau-Bite demonstrates how a context-specific locally developed digital solution can effectively modernize campus food services, improve user satisfaction, and enhance operational efficiency in a Nigerian university setting.

5.1. Conclusion

The successful development and testing of Tau-Bite confirm that a customized web-based food ordering and delivery system can effectively address the inefficiencies of the manual food service model at Thomas Adewumi University.

The study achieved its primary aim of enhancing accessibility, efficiency, and user satisfaction through automation and digital integration. By enabling real-time menu access, secure online payments, and order tracking, the system reduces waiting times, minimizes human error, and improves service transparency.

Built on the Laravel PHP framework, Tau-Bite is a robust feature-rich and scalable platform designed to meet the needs of both end-users and administrators. It follows modern web development best practices including modular Model-View-Controller (MVC) architecture, secure authentication, and seamless integration with the Paystack payment gateway. The system's intuitive interface, comprehensive menu management, and streamlined order processing workflows ensure a superior user experience, while features like real-time notifications enhance accessibility and convenience.

From an administrative perspective Tau-Bite provides a centralized dashboard for managing users, orders, menus, and transactions. Its extensible design and adherence to industry standards make it well-suited for future enhancements and integration with additional services.

The reliability of the platform is underpinned by rigorous testing thorough documentation, and a strong commitment to data security and integrity. Furthermore, the use of Agile development methodologies enabled continuous improvement and rapid adaptation to stakeholder feedback ensuring that the final product remains aligned with both business objectives and user expectations.

5.3. System Improvements and Recommendations

1. Security Enhancements

   a. Two-Factor Authentication (2FA): Implement 2FA for both users and administrators to strengthen account security, especially for sensitive operations such as payments and profile changes.

   b. OAuth/Social Login Integration: Allow users to register and log in using third-party providers (Google, Facebook, Apple), improving convenience and reducing password fati +. Comprehensive Audit Logging: Track and log all critical actions (e.g., logins, payments, admin changes) for compliance, security monitoring, and troubleshooting.

2. Performance and Scalability

   a. Caching Strategies: Integrate Redis or Memcached for caching frequently accessed data (e.g., restaurant lists, menu items) to reduce database load and improve response times.

   b. Asynchronous Processing: Offload resource-intensive tasks (such as email notifications, payment processing, and report generation) to background jobs using Laravel Queues and Supervisor.

c. Database Optimization: Regularly review and optimize database indexes, queries, and schema to ensure efficient data retrieval as the user base grows.

3. User Experience (UX) Improvements

   a. Progressive Web App (PWA) Features: Enable offline access, push notifications, and home screen installation for a more app-like experience on mobile devices.

   b. Real-Time Order Tracking: Provide users with real-time updates on order status, including preparation, dispatch, and delivery, possibly using Web Sockets or Laravel Echo.

   c. Personalized Recommendations: Implement recommendation algorithms to suggest restaurants or menu items based on user preferences and order history.

4. Feature Expansion

   a. Loyalty and Rewards Program: Introduce a points-based system to reward frequent users, incentivizing repeat business and increasing user engagement.

   b. Multi-Restaurant Cart and Split Payments: Allow users to order from multiple restaurants in a single transaction, with clear handling of split payments and delivery logistics.

   c. Advanced Search and Filtering: Enhance search capabilities with filters for cuisine, dietary preferences, ratings, price range, and delivery time.

5. Administrative and Business Tools

   a. Advanced Analytics Dashboard: Provide administrators and restaurant owners with detailed analytics on sales, customer behavior, and operational metrics.

b. Role-Based Access Control (RBAC): Implement granular permissions for different admin roles (e.g., super admin, restaurant owner, support staff) to improve security and operational efficiency.
c. Bulk Data Import/Export: Enable bulk upload and download of menu items, restaurant data, and user lists for easier onboarding and data management.

5.4 Contribution to Knowledge

This project contributes to academic and practical knowledge in the following ways:

a. It provides a context-specific digital solution for Nigerian university campuses, addressing a gap in existing literature.
b. It demonstrates how low-cost, locally developed web applications can solve institutional challenges without relying on commercial platforms.
c. It offers a replicable model for digitizing campus services, supporting the growth of smart university initiatives in developing countries.

REFERENCES

Ajzen, I. (1991). The theory of planned behavior. *Organizational Behavior and Human Decision Processes*, *50*(2), 179–211.

Al-Kandari, A. M., Al-Musallam, Z., & Al-Qattan, M. (2021). The Impact of COVID-19 Pandemic on Online Food Delivery Applications: A Study in Kuwait. *Journal of Business Management and Economics*, *2*(1), 1-12.

Aprilliyanto, F., Fauzi, M. A., & Nuruzzaman, A. (2015). Geographical information system for webbased food delivery services.

ASQ. (n.d.). What is Customer Satisfaction?. *Quality Glossary Definition*.

Chai, L. T., & Yat, D. N. C. (2019). Online food delivery services: Making food delivery the new normal. *Journal of Marketing Advances and Practices*, *1*(1), 62–77.

Collier, J. E., & Sherrell, D. L. (2010). Examining the influence of control and convenience in a self-service setting. *Journal of the Academy of Marketing Science*, *38*(4), 490-509.

Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly*, *13*(3).

Genie AI. (n.d.). Food Vendor Definition: 125 Samples. *Law Insider*.

Gumilang, S. F. S., & Nurmala, M. D. I. (2018). Web-based Food Delivery Management System. In *International Conference on Industrial Enterprise and System Engineering (IcolESE 2018)* (pp. 1-5). Atlantis Press.

Gunden, C., Morosan, C., & DeFranco, A. (2020). Consumer intentions to use online food delivery systems in the USA.

Huan, K. J., & Mahdin, H. (2023). Design and Development of Food Ordering System on Webbased Technology. *Applied Information Technology And Computer Science*, *4*(1), 446-465.

Hwang, J., & Kim, H. (2021). Consequences of a green image of drone food delivery services: The moderating role of gender and age. *Business Strategy and the Environment*, *30*(7), 3271–3285.

IBM. (n.d.). What is Order Management?.

Interaction Design Foundation. (n.d.). What is User Interface (UI) Design?.

Luo, Q., Liufu, X., & Li, Z. (2020). Intelligent online food delivery system: Dynamic model for delivery strategy and tip advice.

Li, C., Mirosa, M., & Bremer, P. (2020). Review of online food delivery platforms and their impacts on sustainability. *Sustainability*, *12*(14), 5528.

Meuter, M. L., Ostrom, A. L., Roundtree, R. I., & Bitner, M. J. (2000). Self-service technologies: understanding customer satisfaction with technology-based service encounters.

Journal of Marketing, *64*(3), 50-64.

Munich Business School. (n.d.). The Campus: A Place for Living & Learning. *Glossary*.

Paul, R., Rathee, R., Matthew, P., & Adusumilli, P. (2020). Optimization framework for ondemand meal delivery systems.

Ray, A., Dhir, A., Bala, P. K., & Kaur, P. (2019). Why do people use food delivery apps (FDA)? A uses and gratification theory perspective. *Journal of Retailing and Consumer Services*, *51*, 221–230.

ResearchGate. (n.d.). (PDF) Online Food Ordering System.

Sharma, R. (2023). Design and implementation of a food delivery notification system using Django.

Singh, A., Kumar, M., & Chakraborty, T. (2023). Fairness in online service with multiple servers: Application to food delivery.

Thakur, A., Sharma, M., & Bhatt, R. (2018). Factors influencing the use of internetbased food delivery service apps.

Tong, B. L., & Samsudin, N. A. (2022). Development Of Food Ordering Web-based System For Restaurant. *Applied Information Technology And Computer Science*, *3*(1), 975-994.

WEZOM. (n.d.). Food Delivery Logistics: Challenges and How to Overcome Them.

Yeo, V. C. S., Goh, S. K., & Rezaei, S. (2017). Consumer experiences, attitude and behavioral intention toward online food delivery (OFD) services. *Journal of Retailing and Consumer Services*, *35*, 150–162.

Zhang, Y., Liu, F., Li, W., & Chen, H. (2024). Impact of onlinetooffline food delivery platforms on healthy food choices.

Zheng, Y., Wang, X., & Chen, Z. (2022). Dynamic multiobjective balancing in online food delivery systems.

Zheng, X., Li, J., & Wong, T. (2022). Modeling stochastic service time in complex ondemand food delivery systems.