APPLIED INFORMATION TECHNOLOGY AND COMPUTER SCIENCE
e-ISSN: 2773-5141

AITCS

Vol. 6 No. 1 (2025) 1049-1064
https://publisher.uthm.edu.my/periodicals/index.php/aitcs

OTAi Burger Ordering Management Application

Nagulan Panir Selwam[1], Suhaila Mohd Yasin[1]*

[1] *Fakulti Sains Komputer dan Teknologi Maklumat,*
Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, 86400, MALAYSIA

*Corresponding Author: ysuhaila@uthm.edu.my
DOI: https://doi.org/10.30880/aitcs.2025.06.01.058

Article Info

Received: 7 January 2025
Accepted: 19 June 2025
Available online: 30 June 2025

Keywords

Food ordering application, Prototyping model, React Native, Order management

Abstract

The OTAi Burger Ordering Management Application modernizes smaller burger stalls like OTAi Burger in Puncak Alam, Selangor. Key challenges included price discrepancies, high staff turnover, and inefficient manual ordering. The Prototyping Model ensures user feedback refines each iteration. Built with React Native for cross-platform development and Firebase for secure data management, the application automates orders, standardizes pricing, provides detailed receipts, and offers modules for menu, sales analytics, feedback, job vacancies, and staff management. By reducing reliance on third-party services, it boosts efficiency, transparency, and customer satisfaction. This solution helps smaller food businesses improve performance and customer loyalty in a competitive market.

1. Introduction

The OTAi Burger Ordering Management Application aims to revolutionize the operational dynamics of smaller burger stalls by leveraging modern technology. Located in Puncak Alam, Selangor, and owned by Mr. Zulfadli, OTAi Burger has built a reputation for its delectable burgers and significant growth in Selangor's competitive food market over the past two years [1]. Despite its success, the application is designed to streamline operations, improve efficiency, and enhance customer satisfaction. It enables customers to place orders via the app, process payments, and simplify asset utilization. This integration ensures consistent pricing, automates order management, provides detailed receipts, and facilitates efficient staff recruitment and management. By organizing data effectively and integrating seamlessly with existing processes, the application aims to position OTAi Burger for sustained success in today's competitive market.

1.1 Problem Statement

The existing system at the OTAi Burger stall has led to several critical issues impacting its operations and viability, including reliance on third-party delivery services like Food Panda, resulting in pricing discrepancies between the app and physical store that decrease customer trust and hinder revenue growth. Additionally, a high staff turnover rate can lead to temporary closures, affecting revenue and stability, while manual order recording consumes valuable time, lacks proper records for monthly sales, and complicates financial management and accountability. The proposed OTAi Burger Ordering Management Application aims to address these challenges by ensuring consistent pricing, rebuilding customer trust, including a job vacancy feature to quickly attract new staff, and automating order management to provide detailed receipts and seamless integration with existing systems, ultimately enhancing operations, transparency, and customer satisfaction.

1.2 Objective

There are three scopes established to guide this project towards its goals. The following is a list of the project objectives:

i. To analyze and design the OTAi Burger Ordering Management Application using an object-oriented approach.
ii. To develop a mobile application for the OTAi Burger Ordering Management System.
iii. To test and evaluate the functionality of the OTAi Burger Ordering Management Application using functional testing and user acceptance testing (UAT).

1.3 Scope

The scope of this project involves developing the OTAi Burger Ordering Management Application to address the specific challenges faced by OTAi Burger. To solve the identified problems, the project will use modern software development techniques, including prototyping model for iterative refinement through continuous stakeholder feedback. The mobile-based application will feature intuitive user interfaces and robust backend systems to streamline ordering, enhance efficiency, and improve the overall customer experience. The primary users of the application will include the admin, staff, and customers. Table 1 shows the modules of the application.

Table 1 *Modules of the OTAi Burger Ordering Management Application*

Modules	Description
Sign Up and Login	New users, admin, and staff can create accounts and log in using email and password.
	Customers can reset passwords via the "forget password" option.
Manage Profile	Admin, staff, and customers can view and edit their information.
Manage Menu	Admin can add, delete, update items' information, and change their status.
	Customers can view the menu list.
Manage Order	Staff can view and update order status.
	Customers can view items, order, update the cart, and proceed to checkout.
Manage Payment	Customers can make payments and view payment receipts.
Manage Feedback	Customers can give feedback, and admin can view the feedback.
Manage Report	Admin can view sales and generate sales reports.
Apply Job Vacancy	Customers can view, fill in, and submit job vacancy forms.
Manage Job Vacancy	Admin can view job vacancy details, approve, or reject staff applications.
Manage Staff	Admin can edit staff information, update their status, and create staff accounts.

2. Literature review

This literature review delves into the essential technologies and methodologies that underpin the OTAi Burger Ordering Management Application. By examining existing research and case studies, it provides a foundation for understanding the current state of technology and best practices, ensuring the application is designed to effectively meet the needs of OTAi Burger and address common industry challenges.

2.1 Technology used

The OTAi Burger Ordering Management Application leverages several key technologies to enhance its functionality. The mobile application framework, particularly React Native, simplifies app development by providing pre-built components and enabling code reuse across platforms. Integrated Development Environments (IDEs) like Visual Studio and Eclipse combine essential development tools into one application, boosting developer productivity with features like auto-completion and error detection. For secure online transactions, the application uses payment gateways such as PayPal and Stripe, which encrypt data and integrate fraud detection systems, though they come with transaction fees and technical complexities. Firebase, owned by Google, provides the database backbone with its Realtime Database and Cloud Firestore, offering real-time data

synchronization, offline capabilities, and advanced querying, essential for managing the app's live updates and complex data needs.

2.2 Study of existing related systems

This section examines existing food ordering systems to understand their functionality, strengths, and weaknesses. By analyzing systems such as Alexandria's Café Food Ordering System, Rosalinda Food Ordering Mobile Application, and Face Food Café Application, we can identify best practices, avoid past mistakes, and leverage proven solutions. This study informs the development of the OTAi Burger Ordering Management Application, ensuring it addresses common challenges and meets user needs effectively.

2.2.1 Alexandria's Café Food Ordering System

The food ordering system at Alexandria's Café operates manually, where customers queue to place orders with staff who then manage and update the menu manually [2]. This traditional method relies heavily on face-to-face interactions, offering immediate communication and personal service. While simple and direct, it can lead to longer wait times, especially during peak hours, due to the need for staff to individually take and process each order. Additionally, manual menu updates can result in customer dissatisfaction if unavailable items are not communicated until the front of the line. The system's reliance on staff for order management increases the likelihood of human error. Unlike automated systems, it does not utilize digital tools to streamline operations or improve order accuracy.

2.2.2 Rosalinda Food Ordering Mobile Application

The Rosalinda Food Ordering Mobile Application was developed to replace the old manual system at Restoran Rosalinda, which often led to problems such as unreadable orders and long wait times [3]. The app allows customers to place orders digitally, significantly improving service speed and efficiency. Similar to the OTAi Burger Ordering Management Application, it automates the ordering process, reducing errors from miscommunication and incorrect order entries. Real-time order notifications keep customers informed about their order status, enhancing satisfaction. Additionally, the app includes backend functionalities for staff, such as product management and sales reporting, providing valuable insights into restaurant operations. However, reliance on digital systems can be challenging for customers unfamiliar with technology or without smartphone access, and maintaining the system requires regular updates and can incur operational costs.

2.2.3 Face Food Café Application

The Face Food Café application enhances customer experience by enabling online food ordering through an Android app, replacing the traditional manual process where customers ordered in [4]. The application includes features such as user registration and login, menu browsing, order placement, and order management. Customers can create accounts, log in, browse an updated menu, and place orders directly through the app. The restaurant administrator receives and manages these orders, with the ability to add, delete, and modify menu items as needed. Real-time updates on order status ensure a smooth customer experience. The automation of the order process reduces errors and speeds up service, while digital menu management eliminates the need for printed menus, reducing costs. However, challenges include potential exclusion of customers without smartphones and the need for ongoing maintenance and updates to the app. Despite these challenges, the digital system significantly improves efficiency and customer satisfaction.

2.3 Comparison between related system

This section provides a comparative analysis of the features found in existing food ordering systems and the proposed OTAi Burger Ordering Management Application. By examining the functionalities, user interfaces, and overall user experiences of Alexandria's Café Food Ordering System Mobile Application, Rosalinda Food Ordering Mobile Application, and Face Food Café Application, we can identify the unique advantages and potential improvements offered by the OTAi Burger Ordering Management Application. The following table summarizes the key features and differences between these systems.

Table 2 *Comparison between the existing systems and the proposed system*

Features	Alexandria's Café Food Ordering System Mobile Application	Rosalinda Food Ordering Mobile Application	Face Food Café Application	OTAi Burger Ordering Management Application
Sign Up and Login	Yes	Yes	Yes	Yes

Manage Profile	Yes	Yes	Yes	Yes
Manage Menu	Yes	Yes	Yes	Yes
Manage Order	Yes	Yes	Yes	Yes
Manage Payment	Yes	Yes	Yes	Yes
Manage Feedback	No	No	Yes	Yes
Manage Report	No	Yes	Yes	Yes
Apply Job Vacancy	No	No	No	Yes
Manage Job Vacancy	No	No	No	Yes
Manage Staff	No	Yes	Yes	Yes

3. Methodology

The Prototyping Model's iterative and user-centered approach was ideal for the development of the OTAi Burger Ordering Management Application, particularly for refining requirements through continuous stakeholder feedback [5]. This methodology involved six phases, beginning with the Requirements Gathering and Analysis phase, where the system scope and functionalities were identified through stakeholder interviews and a Gantt chart was developed to guide project activities. The Quick Design phase translated these requirements into visual representations, including use case diagrams, wireframes, and system architecture designs, using React Native for the interface and Firebase for the backend. The Prototype Development phase involved building foundational modules incrementally, such as Sign Up, Login, Manage Menu, and Order Management, with version control tools ensuring smooth collaboration. During the User Evaluation phase, stakeholders tested the prototype and provided feedback, which was incorporated during the Refinement phase to improve system workflows, interface designs, and functionality. The Final System Development phase focused on integrating all modules, conducting rigorous testing, and preparing the application for deployment. This structured yet flexible approach ensured a robust, user-friendly system aligned with business objectives and stakeholder expectations.

Table 3 *Software development activities and their task*

Phases	Tasks	Output
Requirements Gathering and Analysis	- Identify system scope and high-level requirements. - Conduct interviews with the stakeholder. - Analyze requirements through data collection. - Create a project timeline using a Gantt chart.	Project proposal, Gantt chart
Quick Design	- Design use case diagram, sequence diagrams, activity diagrams, and class diagram. - Design use case specifications. - Design user interface. - Design database.	Use case diagram, Use case specifications, Sequence diagrams, Activity diagrams, Class diagram, User interfaces and Schema table
Development	- Develop the system modules incrementally using React Native and Firebase. - Implement key features for testing and feedback.	Fully working system modules, Integrated system and Initial working prototype
User Evaluation	- Share the prototype with stakeholders for feedback. - Identify issues and areas for improvement.	Feedback reports and Usability findings
Refinement	- Update the prototype based on feedback. - Improve system functionalities and workflows. - Conduct iterative testing	Refined prototype
Final System Development	- Fully integrate and finalize all system modules. - Conduct testing to ensure functionally and performance.	Fully functional system

Applied Information Technology and Computer Science Vol. 6 No. 1 (2025) p. 1049-1064

4. Analysis and Design

This section focuses on the Analysis and Design phase of the OTAi Burger Ordering Management Application. It covers both functional and non-functional aspects. Using Unified Modeling Language (UML) diagrams and use case diagrams, it visualizes system interactions and functionalities. Additionally, it details the database design and interface design for user interactions, providing a solid foundation for the application's development and implementation.

4.1 System Requirements Analysis

System requirement analysis identifies, evaluates, validates, and documents customer needs to ensure they are accurately reflected in the final product. This process involves eliciting requirements from stakeholders, detailing them, obtaining approvals, and managing changes to keep the project on track in terms of time, quality, and budget. Functional and non-functional requirements are essential, defining the system's features, functions, and quality attributes. Functional requirements specify the system's modules, inputs, outputs, parameters, data types, and users involved. Table 4 show the functional requirements and Table 5 shows the non-functional requirements.

Table 4 *Functional requirements*

Modules	Description
Sign Up and Login	• This module enables user authentication and account management. • The module allows customers to create new accounts to log in, while admin and staff can log in using their email and password. • The module allows customers to reset their passwords by clicking the “Forget Password” link.
Manage Profile	• This module allows users to manage their personal information. • The module allows admin, staff, and customers can view and edit their own profiles to ensure their information is current and accurate.
Manage Menu	• This module gives the admin full control over the menu items available for ordering. • The module allows admin can add new items, delete existing ones, update item information such as descriptions and prices, and change the status of items.
Manage Order	• This module handles the ordering process. • The module allows staff can view all customer orders and update their statuses, while customers can browse menu items, place orders, update their cart, and proceed to checkout for payment.
Manage Payment	• This module facilitates payment processing. • The module allows customers can make payments for their orders and view receipts to confirm successful transactions.
Manage Feedback	• This module allows customers to provide feedback on their experience and menu items. • The module allows admin can view and assess this feedback to improve service quality and customer satisfaction.
Manage Report	• This module provides tools for generating and viewing sales reports. • The module allows admin can access detailed sales data and performance metrics, allowing them to analyze and improve business operations.
Apply Job Vacancy	• This module streamlines the job application process. • The module allows customers can view job vacancy application, fill out application forms, and submit them for consideration.
Manage Job Vacancy	• This module allows admins to manage job vacancy postings and applications. • The module allows admin can review job vacancy details and approve or reject applications from potential staff.

Manage Staff	• This module handles staff information and account management. • The module admin can edit staff details, update their status, and create new staff accounts to ensure proper staff management and record-keeping.

Table 5 *Non-functional requirements*

Modules	Description
Usability	• The software application must offer an intuitive user interface, ensuring ease of navigation and interaction for all user categories. • The design should be user-friendly, allowing customers, staff, and admins to efficiently perform their respective tasks without requiring extensive training.
Performance	• The system must perform efficiently under expected load conditions. • It should provide fast response times for user interactions and data processing, ensuring that users do not experience significant delays when using the application. • The system should be capable of handling peak traffic periods without performance degradation.
Security	• Robust security measures must be in place to protect user data and ensure secure operations. • This includes data encryption, secure login processes, and protection against security threats.
Compatibility	• The application must be compatible across various devices and operating systems. • It should function seamlessly on different screen sizes and resolutions, ensuring a consistent user experience regardless of the device being used. The system should guarantee high reliability with consistent uptime, ensuring that the services are always available to users.
Reliability	• The system should guarantee high reliability with consistent uptime, ensuring that the services are always available to users.

4.2 System Analysis

System analysis is the foundation of a successful software development lifecycle, where the existing environment is evaluated, user needs are identified, and objectives for the new system are set. This phase involves examining current processes, uncovering potential improvements, and establishing clear requirements that shape the design, development, and implementation stages. In this analysis, we include a use case diagram to clarify user interactions within the OTAi Burger Ordering Management Application and a class diagram to define the system's core structure and relationships, ensuring both functional needs and architectural integrity are addressed.

4.2.1 Use Case Diagram

The use case diagram for the OTAi Burger Ordering Management Application provides a detailed representation of the interactions between the various actors and the system's functions. This diagram helps to identify the different roles users play and the specific functionalities they interact with, ensuring all user requirements are clearly defined and addressed. Fig. 1 shows the interactions between different user types and the system's functions.

Applied Information Technology and Computer Science Vol. 6 No. 1 (2025) p. 1049-1064

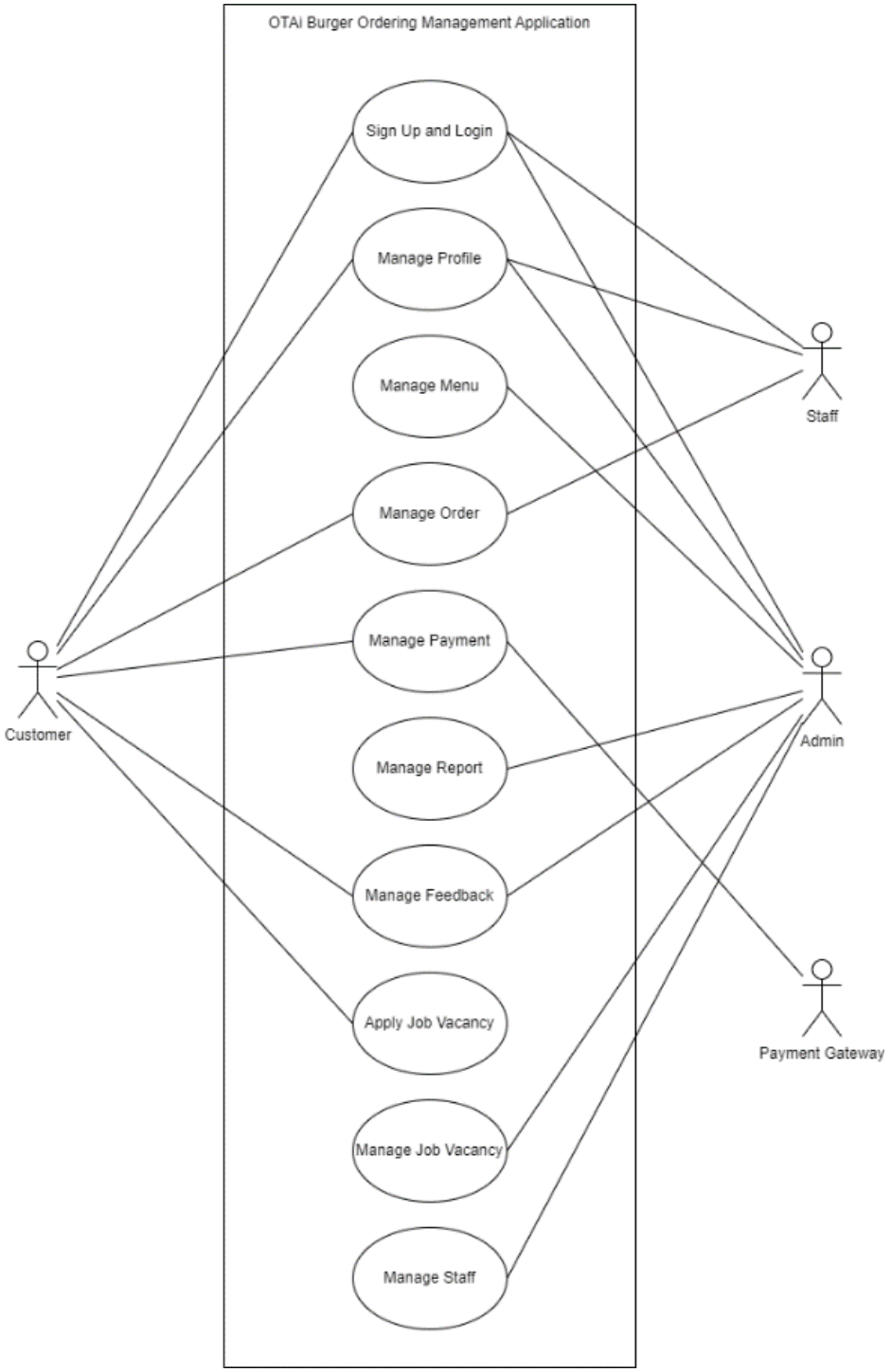


Fig. 1 *Use case diagram of OTAi Burger Ordering Management Application*

4.2.2 Class Diagram

The class diagram for the OTAi Burger Ordering Management Application illustrates a system designed to manage menu, orders, payments, feedback, job vacancies, and reporting. Admins handle menu, reports, job vacancies, feedback, and staff management, while staff manage orders and profiles. Customers can sign up, log in, manage profiles, view the menu, place orders, make payments, view order status, and apply for jobs. The system includes classes such as Admin, Staff, Customer, Menu, Order, Payment, Feedback, JobVacancy, and Report, providing a structured framework for processing transactions, managing orders and payments, receiving feedback, applying for jobs, and generating reports. This cohesive framework ensures efficient management of all processes, enhancing the user experience for admins, staff, and customers. Fig. 2 shows the class diagram of OTAi Burger Ordering Management Application.

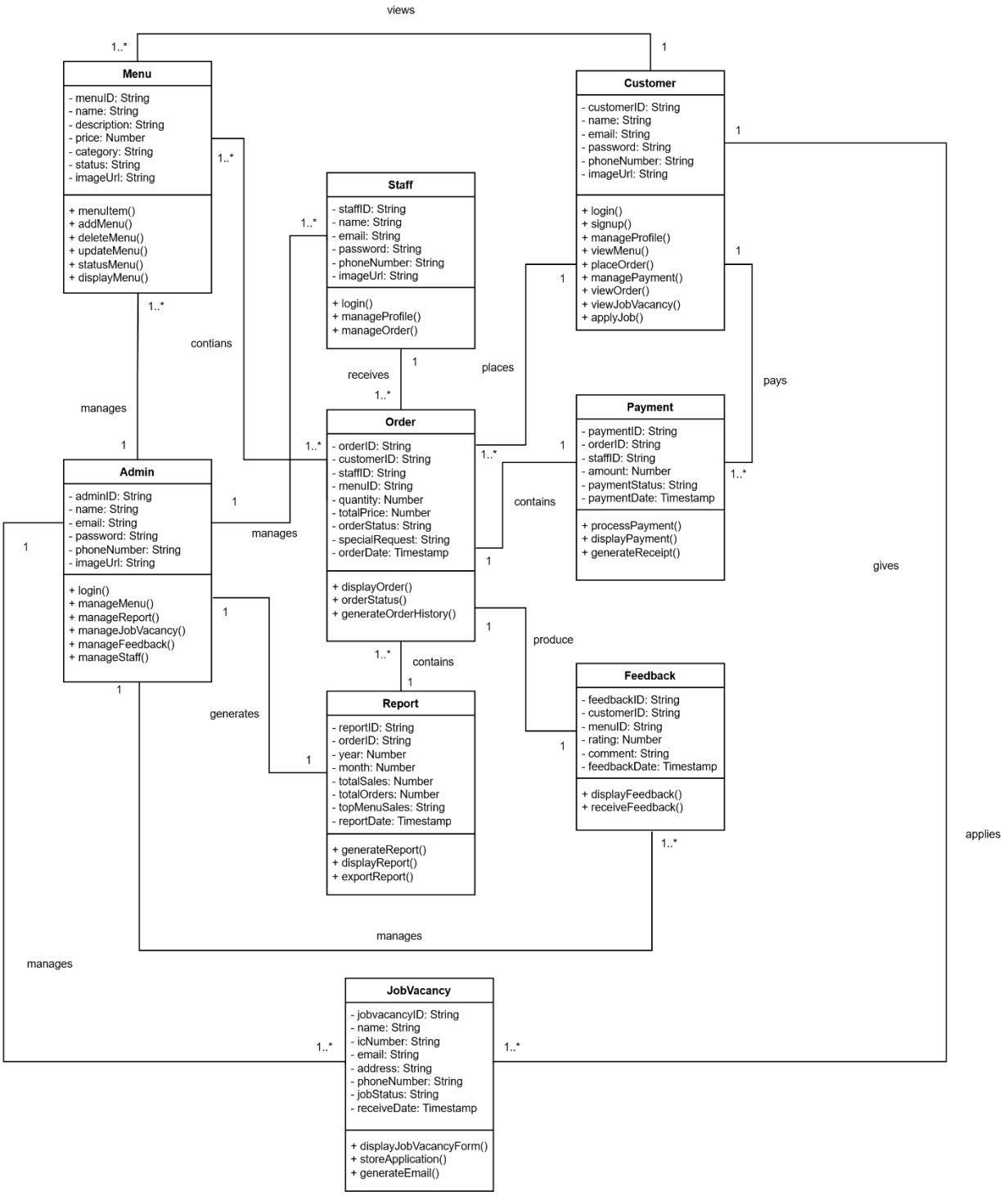


Fig. 2 *Class diagram of OTAi Burger Ordering Management Application*

5. Results and Discussion

The OTAi Burger Ordering Management Application was developed as a mobile application using React Native for the front-end to ensure a dynamic and user-friendly interface. The back-end is powered by Firebase, which provides real-time database management, authentication, and secure storage services [6]. The combination of these technologies ensures a scalable and efficient application tailored to address the needs of small-scale burger stalls like OTAi Burger. Development and testing were carried out using Visual Studio Code as the code editor and Expo for deployment and device testing, ensuring smooth integration of modules and cross-platform functionality. The application is equipped with essential modules, including Sign Up and Login, Menu Management, Order Management, Payment Processing, Feedback Handling, and Job Vacancy Management. These modules were designed to streamline operations, improve customer satisfaction, and support efficient staff management. Local

Applied Information Technology and Computer Science Vol. 6 No. 1 (2025) p. 1049-1064

testing was conducted iteratively during the development phase to ensure each module met the functional requirements and performed seamlessly under different scenarios.

5.1 Interface Design

The interface design for the OTAi Burger Ordering Management Application was implemented using React Native, ensuring a responsive and user-friendly experience for both Android and iOS users. The design focuses on simple navigation, accessibility, and consistent functionality for customers, staff, and admins. Each interface was crafted to provide a smooth and efficient user experience while supporting the application's features and operations. The following sections highlight the design and functionality of each module, showcasing how the application delivers a seamless experience for all users.

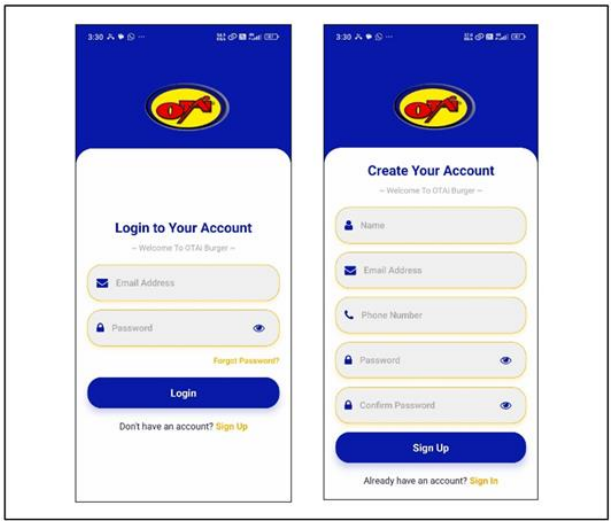


Fig. 3 *Sign Up and Login interface*

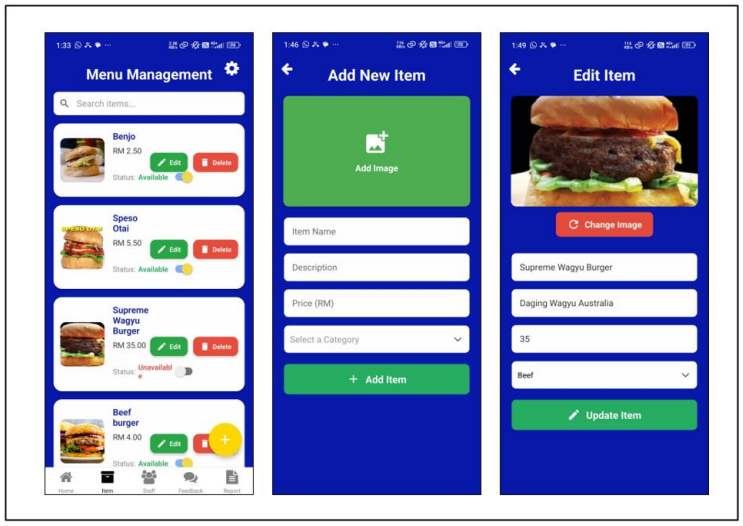


Fig. 4 *Admin's manage menu interfaces*

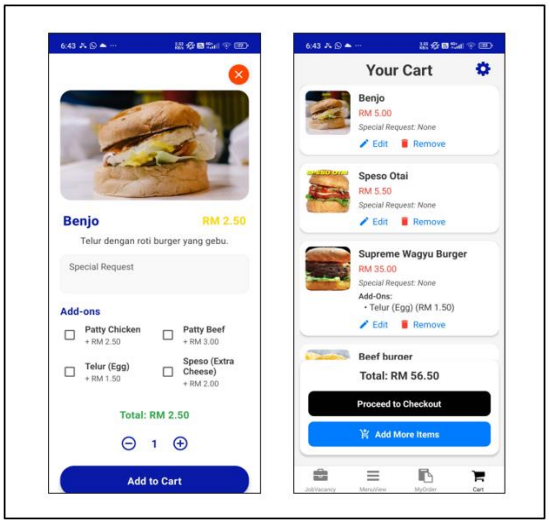


Fig. 5 *Customer's manage order interfaces*

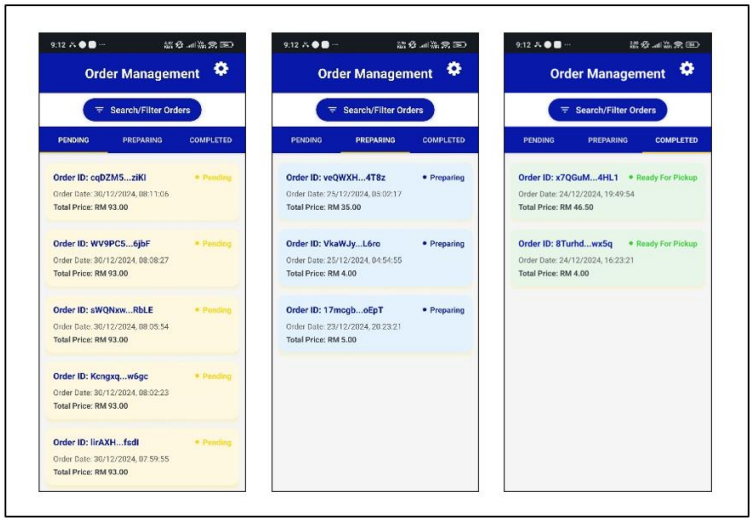


Fig. 6 *Staff's manage order interfaces*

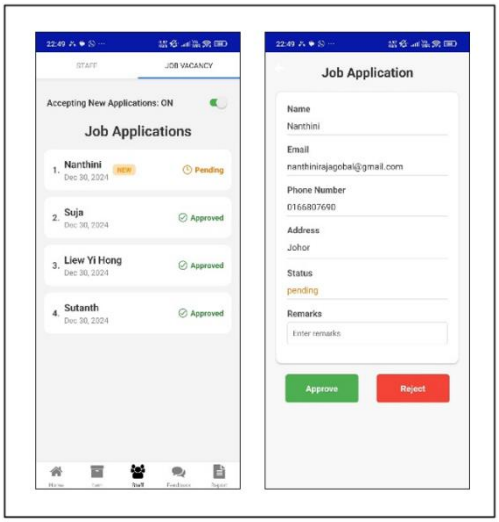


Fig. 7 *Admin's manage job vacancy interfaces*

5.2 Testing

Testing is a crucial phase in the software development lifecycle, ensuring the system operates as intended, meets user requirements, and is free of defects. For the OTAi Burger Ordering Management Application, testing was carried out through functional testing and user acceptance testing to evaluate its performance, usability, and

reliability. These processes were essential to validate the application's functionality and ensure a seamless experience for all users.

Table 2 *Test criteria*

Test Case ID	Requirements ID	Descriptions	Status
TC_01	**REQ_100**	**Sign Up and Login**	
TC_01_01	REQ_101	The system shall allow the user to Login with correct email and password.	PASS
TC_01_02	REQ_102	The system shall allow the user to Sign Up with user details and valid email, password and confirm password.	PASS
TC_01_03	REQ_103	The system shall validate the email during registration.	PASS
TC_01_04	REQ_104	The system shall update the new password in the database if the user forgets their password.	PASS
TC_02	**REQ_200**	**Manage Profile**	
TC_02_01	REQ_201	The system shall allow the user to access profile management options from the settings page.	PASS
TC_02_02	REQ_202	The system shall allow the user to update personal details such as name, phone number, and email address in personal details page.	PASS
TC_02_03	REQ_203	The system shall allow the user to change their password in change password page.	PASS
TC_03	**REQ_300**	**Manage Menu**	
TC_03_01	REQ_301	The system shall allow the user to access menu management page.	PASS
TC_03_02	REQ_302	The system shall display options to add, update, delete, or change the status of menu items.	PASS
TC_03_03	REQ_303	The system shall allow the admin to add new menu items by entering item details.	PASS
TC_03_04	REQ_304	The system shall allow the admin to update existing menu items.	PASS
TC_03_05	REQ_305	The system shall allow the admin to delete menu items.	PASS
TC_03_06	REQ_306	The system shall allow the admin to update the status of menu items to available or out of stock.	PASS
TC_04	**REQ_400**	**Manage Order**	
TC_04_01	REQ_401	The system shall allow staff to log in and navigate to the Orders page.	PASS
TC_04_02	REQ_402	The system shall allow staff to update the order status.	PASS
TC_04_03	REQ_403	The system shall allow customers to log in and navigate to the Home page.	PASS
TC_04_04	REQ_404	The system shall allow customers to add items to their cart.	PASS
TC_04_05	REQ_405	The system shall redirect customers to the payment page after they click the checkout button.	PASS
TC_04_06	REQ_406	The system shall allow customers to view the ordered list and order status on the My Order page.	PASS
TC_05	**REQ_500**	**Manage Payment**	

TC_05_01	REQ_501	The system shall allow users to initiate payments through a payment gateway.	PASS
TC_05_02	REQ_502	The system shall process and record payment details after a successful transaction.	PASS
TC_05_03	REQ_503	The system shall generate a receipt for the completed order.	PASS
TC_05_04	REQ_504	The system shall allow users to view their receipt on the receipt page.	PASS
TC_06	**REQ_600**	**Manage Report**	
TC_06_01	REQ_601	The system shall display the report page.	PASS
TC_06_02	REQ_602	The system shall allow the admin to generate reports.	PASS
TC_06_03	REQ_603	The system shall allow the admin to export generated reports.	PASS
TC_07	**REQ_700**	**Manage Feedback**	
TC_07_01	REQ_701	The system shall allow customers to rate item on a scale of 1 to 5 stars after purchasing them.	PASS
TC_07_02	REQ_702	The system shall allow customers to provide text feedback for purchased items.	PASS
TC_07_03	REQ_703	The system shall allow admins to view customer feedback.	PASS
TC_08	**REQ_800**	**Apply Job Vacancy**	
TC_08_01	REQ_801	The system shall display the job vacancy application form.	PASS
TC_08_02	REQ_802	The system shall allow the customer to submit job applications.	PASS
TC_09	**REQ_900**	**Manage Job Vacancy**	
TC_09_01	REQ_901	The system shall display the job vacancy page to the admin.	PASS
TC_09_02	REQ_902	The system shall display the list of applied job vacancies.	PASS
TC_09_03	REQ_903	The system shall allow the admin to view detailed information of any job application	PASS
TC_09_04	REQ_904	The system shall provide options for the admin to approve or reject job applications.	PASS
TC_09_05	REQ_905	The system shall save the information of approved applications in the staff database.	PASS
TC_09_06	REQ_906	The system shall send email to the application sender.	PASS
TC_10	**REQ_1000**	**Manage Staff**	
TC_10_01	REQ_1001	The system shall display the staff management page to the admin.	PASS
TC_10_02	REQ_1002	The system shall display the list of all staff members.	PASS
TC_10_03	REQ_1003	The system shall allow the admin to view detailed information of any staff member.	PASS
TC_10_04	REQ_1004	The system shall provide an option for the admin to generate new accounts for staff members.	PASS
TC_10_05	REQ_1005	The system shall provide an option for the admin to terminate staff members.	PASS

User Acceptance Testing (UAT) is the final phase before deployment, where real users evaluate the system to ensure it meets their needs and functions as expected [7]. For the OTAi Burger Ordering Management Application, user acceptance testing involved 11 participants which are one admin, one staff member, and nine random customers. Testing was conducted using a Google Form, divided into two sections which are perceived usefulness and perceived ease of use [8]. Results showed over 90% positive feedback, indicating the system is effective, user-friendly, and enhances productivity. Both admin and staff users found the system intuitive and easy to use, while customers appreciated its efficiency and simplicity.

6. Conclusion

In conclusion, the OTAi Burger Ordering Management Application tackles the burger stall's key challenges by providing a single platform for ordering, payments, staff hiring, and reporting. Using a prototyping approach with ongoing stakeholder input, the team created a reliable, user-friendly mobile application. React Native ensures smooth cross-platform performance, while Firebase offers real-time data updates and secure authentication. Testing showed high user satisfaction, proving the system's ease of use and operational benefits. Overall, this application represents a significant advancement in managing burger stall operations,
ensuring sustained growth and competitiveness in the food service industry.

Acknowledgement

The authors extend their heartfelt appreciation to the Faculty of Computer Science and Information Technology at Universiti Tun Hussein Onn Malaysia for their indispensable support and encouragement throughout the course of this study.

Conflict of Interest

Authors declare that there is no conflict of interests regarding the publication of the paper.

Author Contribution

The authors confirm contribution to the paper as follows: ***study conception and design:*** *Nagulan Panir Selwam, Suhaila Binti Mohd Yasin;* ***data collection:*** *Nagulan Panir Selwam, Suhaila Binti Mohd Yasin;* ***analysis and interpretation of results:*** *Nagulan Panir Selwam, Suhaila Binti Mohd Yasin;* ***draft manuscript preparation:*** *Nagulan Panir Selwam, Suhaila Binti Mohd Yasin. All authors reviewed the results and approved the final version of the manuscript.*

References

[1] Google Maps. (2024). Google.com. https://www.google.com/maps/search/fasa+3+puncak+alam/@3.244193,101.4246874,17z/data=!3m1!4b1?authuser=0&entry=ttu

[2] S. S. K. Lau and R. Ibrahim, "Food Ordering System Mobile Application", *aitcs*, vol. 3, no. 1, pp. 964–974, Jun. 2022, Accessed: Apr. 14, 2024. https://publisher.uthm.edu.my/periodicals/index.php/aitcs/article/view/2178

[3] M. Z. Y. Diri and Nurezayana Zainal, "Rosalinda Food Ordering Mobile Application", *aitcs*, vol. 4, no. 2, pp. 1539–1557, Nov. 2023, Accessed: Apr. 14, 2024. https://publisher.uthm.edu.my/periodicals/index.php/aitcs/article/view/12258

[4] N. F. Soble and Ts. Dr. Azizul Azhar Ramli, "Mobile Application FaceFood Cafe: Aplikasi Restoran Face Food Cafe ", *aitcs*, vol. 3, no. 1, pp. 902–911, Jun. 2022, Accessed: Apr. 14, 2024. https://publisher.uthm.edu.my/periodicals/index.php/aitcs/article/view/2561

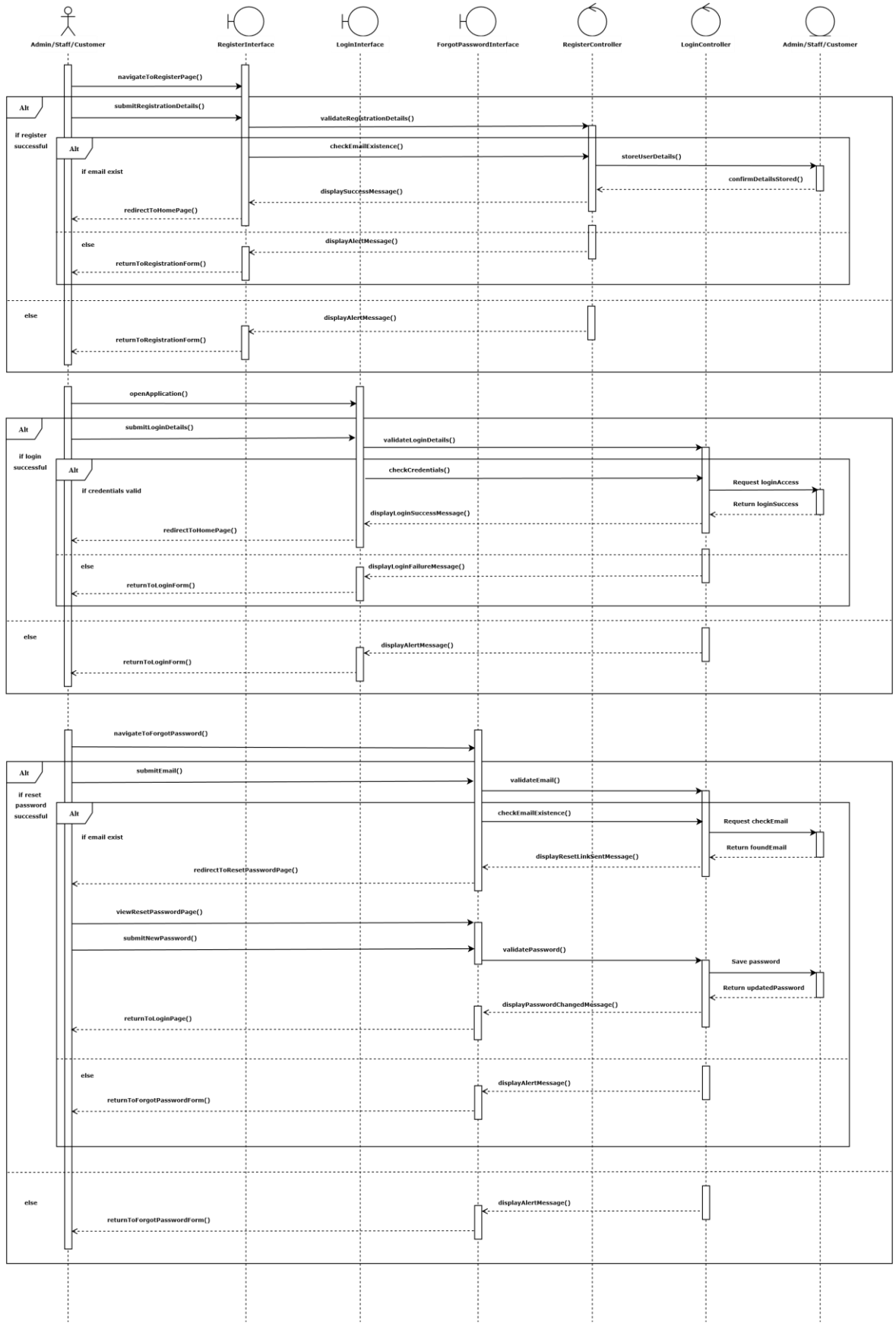
[5] Sommerville, I. (2011). SOFTWARE ENGINEERING Ninth Edition. https://engineering.futureuniversity.com/BOOKS%20FOR%20IT/Software-Engineering-9th-Edition-by-Ian-Sommerville.pdf

[6] Moroney, L. (2017). The definitive guide to Firebase: Build Android apps on Google's mobile platform. Apress. DOKUMEN.PUB. (2017). https://dokumen.pub/the-definitive-guide-to-firebase-build-android-apps-on googles-mobile-platform-1st-edition-9781484229422-9781484229439 1484229428-1484229436.html
Conway, A. (2023). *Introduction To This Guide Bugwolf | The Definitive Guide To User Acceptance Testing (UAT)*. (2017). https://cdn2.hubspot.net/hubfs/1544139/TOFO/The%20Definitive%20Guide%20To%20User%20Acceptance%20Testing%20(UAT)%20-%20Bugwolf.pdf

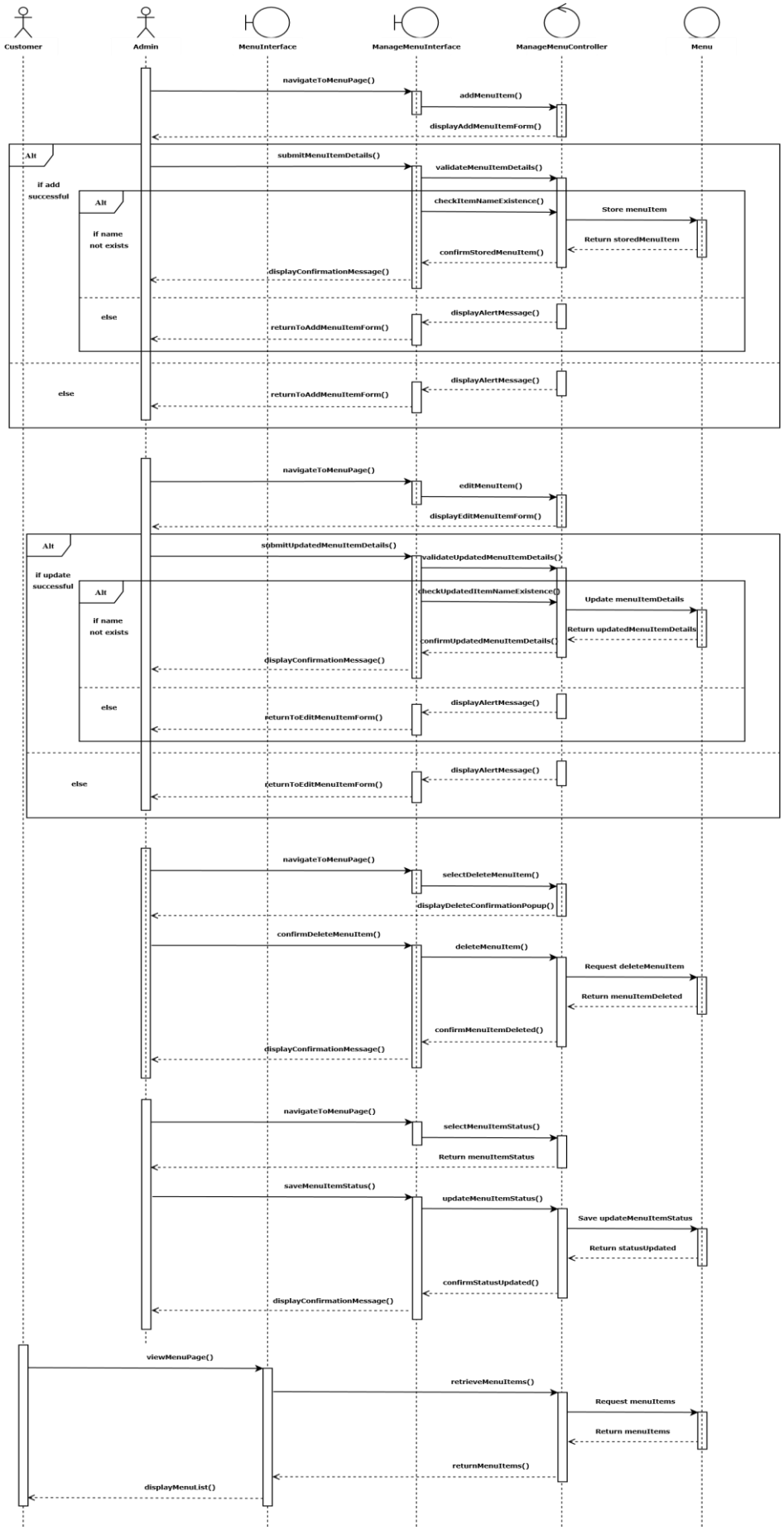
[7] Davis, F. D. (1989). Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology. MIS Quarterly, 13(3), 319–340. Retrieved from https://garyperlman.com/quest/quest.cgi?form=PUEU.

Applied Information Technology and Computer Science Vol. 6 No. 1 (2025) p. 1049-1064

Appendix A: Sequence Diagrams

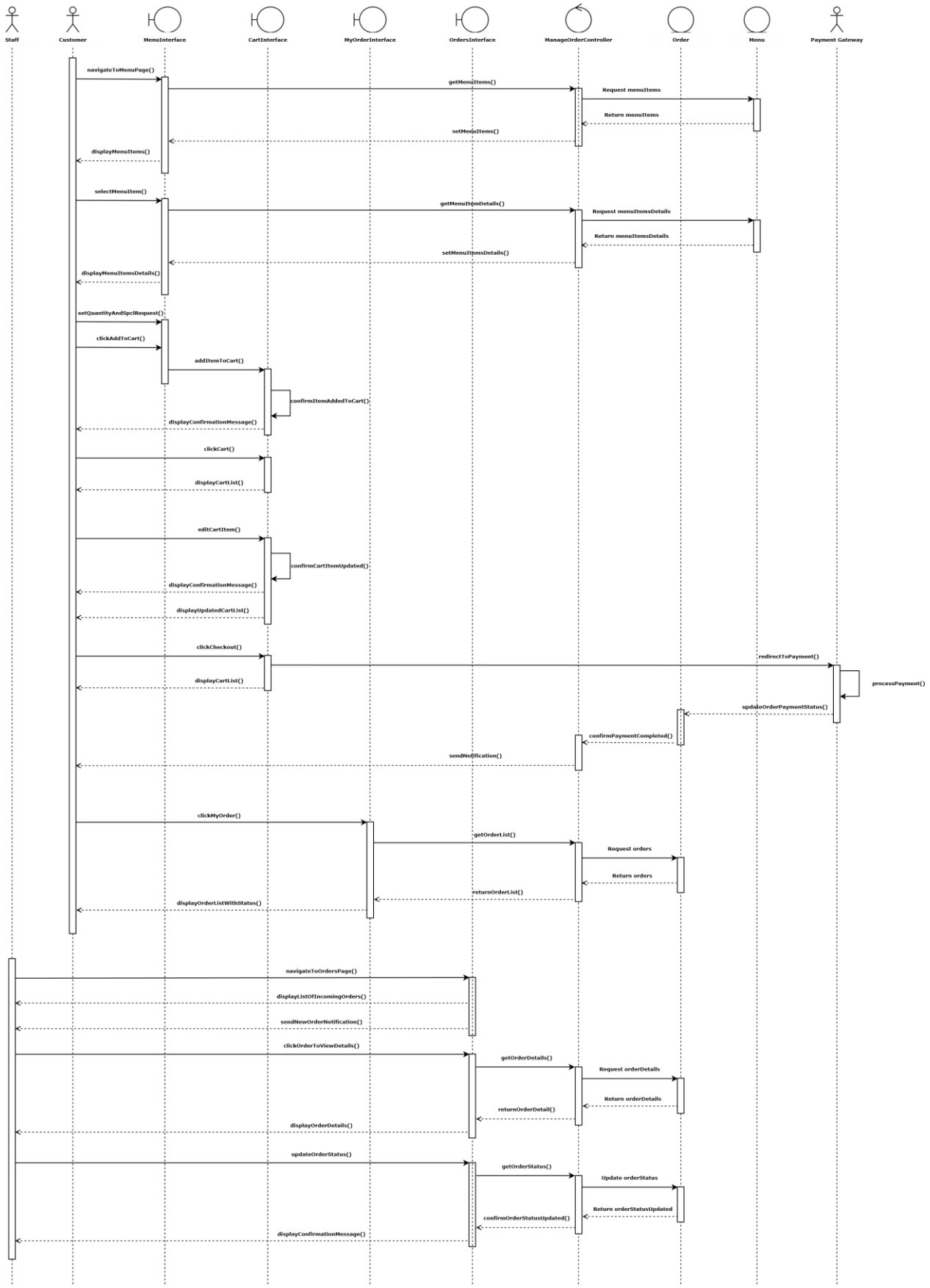


A.1: Sequence diagram for Login



A.2: Sequence diagram for Manage Menu

Applied Information Technology and Computer Science Vol. 6 No. 1 (2025) p. 1049-1064



A.3: Sequence Diagram for Manage Order