



ADDIS ABABA UNIVERSITY
ADDIS ABABA INSTITUTE OF TECHNOLOGY
CENTER OF INFORMATION TECHNOLOGY AND
SCIENTIFIC COMPUTING
DEPARTMENT OF INFORMATION TECHNOLOGY

Title of the Project

KiloEats: An Affordable Food Delivery System for AAU Students

PREPARED BY: Group 17

Members

- | | |
|-------------------|-------------|
| 1. Abenezer Elias | UGR/3180/15 |
| 2. Abiy Aragie | UGR/8104/15 |
| 3. Bisrat Dereje | UGR/3229/15 |
| 4. Blen Debebe | UGR/5297/15 |
| 5. Kena Ararso | UGR/9085/15 |
| 6. Tselot Million | UGR/0522/15 |

ADVISOR: Mr. Aderaw

Date: October 28, 2024

ACKNOWLEDGMENT

We would like to begin by expressing our heartfelt gratitude to God for providing us with the strength, inspiration, and guidance throughout the development of this project proposal. His blessings have been a constant source of motivation and encouragement.

Contents

ACRONYMS	i
ABSTRACT	ii
1. INTRODUCTION	1
1.1 Background	1
1.2 The Existing System	2
1.3 Statement of the Problem	3
1.4 Objective of the Project	4
1.4.1 General Objective	4
1.4.2 Specific Objectives	4
1.5 Proposed System	4
1.6 Feasibility Study	5
1.6.1 Economic Feasibility	5
1.6.1.1 Startup Costs	5
1.6.1.2 Revenue Streams	5
1.6.1.3 Market Demand	5
1.6.2 Technical Feasibility	5
1.6.2.1 Technology Stack	5
1.6.2.2 System Integration	6
1.6.2.3 Scalability and Reliability	6
1.6.3 Schedule Feasibility	6
1.6.3.1 Development Timeline	6
1.6.3.2 Milestones	6
1.6.3.3 Resource Management	6
1.7 Scope	7
1.8 Methodology	8
1.8.1 Justification for Choosing the Waterfall Methodology	8
1.8.2 Project Phases	10
1.8.3 Implementation Methodology	11
1.8.4 Testing Plan	12

1.8.5 Success Assessment.....	12
1.9 Project Management Plan.....	13
1.9.1 Time Management Plan.....	13
1.9.2 Quality Management Plan	13
1.9.3 Communication Management Plan	15
1.9.3.1 Internal communication:.....	15
1.9.3.2 External communication:.....	16
APPENDIX	17
REFERENCE	18

List Of Tables

Table 1.....	13
Table 2.....	15
Table 3.....	16

ACRONYMS

AAU - Addis Ababa University

ABSTRACT

KiloEats aims to provide a convenient, affordable food delivery platform specifically targeting Addis Ababa University (AAU) students in 4, 5, and 6 Kilo. With limited affordable delivery services available for students, the goal of KiloEats is to fill this gap by partnering with local restaurants and eateries.

1. INTRODUCTION

1.1 Background

KiloEats is a food delivery system designed to address a pressing need among students at Addis Ababa University (AAU). The platform aims to connect students with affordable food options from local eateries in the 4, 5, and 6 Kilo areas. This initiative fills a critical gap in the market, where existing delivery services prioritize premium restaurants, leaving students without an accessible, budget-friendly alternative. KiloEats will act as both a delivery service and an online marketplace, bridging the needs of students with the capabilities of small local businesses. The following points elaborate on the motivation, target users, outcomes, financial implications, timeline, stakeholders, and interdependencies associated with this project.

The primary motivation for KiloEats is to create a food delivery system that offers low-cost, convenient options tailored to the needs of AAU students. Existing services such as BeU and Deliver Addis either charge high delivery fees or focus on upscale restaurants, making them impractical for budget-conscious students. Many students also face challenges when traveling off-campus to get meals, resulting in lost study time and additional transportation expenses. KiloEats seeks to reduce these inefficiencies by providing a dedicated platform that focuses on affordability, convenience, and accessibility. The project will also support local restaurants by expanding their customer base and creating an ecosystem that benefits both students and businesses.

The primary users of KiloEats will be the students of Addis Ababa University residing or studying around 4, 5, and 6 Kilo campuses. These students often live on tight budgets, and many rely on nearby affordable food options. Secondary customers include local eateries and restaurants that want to reach a larger student customer base through delivery services. KiloEats aims to act as a bridge between these two groups, offering students access to budget-friendly meals while enabling restaurants to increase their sales.

KiloEats is a new product designed specifically for AAU students. The key deliverables will be:

1. A mobile and web application
2. Integration with local restaurants
3. A delivery management system
4. A payment gateway
5. A feedback system

The development of KiloEats will primarily leverage open-source tools to minimize costs.

1. Development Costs
2. Operational Costs
3. Marketing and Promotion
4. Delivery Network Costs

Tele Birr will be the primary payment method used on the platform. Restaurants will receive the full payment for each order directly through Tele Birr. At the end of each day or shift, they will pay the delivery personnel their respective delivery fees. The company's earnings will be held at the restaurant, with payments accumulated and settled on a monthly basis to streamline financial management.

Timeline for Completion

The development and deployment of KiloEats will follow an estimated a little more than three-month timeline. The initial three weeks will focus on comprehensive requirements gathering, followed by another three weeks dedicated to UI design to ensure a user-friendly interface. A one-month & ten days development phase will allow for the robust creation and refinement of core functionalities. Finally, the testing, deployment, and post-deployment phases will each be allocated ten days to ensure a smooth rollout and prompt resolution of any issues, enabling an efficient and successful launch.

Involved Organizations

The primary stakeholders involved in the project include:

- Addis Ababa University (AAU)
- Local Restaurants students.
- Software Development Team

KiloEats can leverage knowledge and partnerships from other delivery services that operate in Addis Ababa. Insights gained from the logistics of platforms like BeU and ZMall can inform the development of efficient delivery routes and operational strategies. Additionally, the platform will build on best practices from mobile payment services already in use at local restaurants, ensuring reliable and familiar transaction experiences for both customers and partners.

1.2 The Existing System

The current food delivery landscape in Addis Ababa is dominated by services such as BeU Delivery and Deliver Addis. While these platforms offer the convenience of online ordering and home delivery, they primarily cater to high-end or mid-tier restaurants, which makes them less accessible to students at Addis Ababa University (AAU). These platforms charge relatively high delivery fees and focus on delivering meals from premium restaurants. For students, this often means that either the meal options are too expensive, or the added delivery fees make it impractical to order food regularly.

Most delivery services operate with a minimum order requirement, further limiting the flexibility for students who may only want to buy small meals or snacks. Other delivery services also tend to prioritize deliveries within the central business districts of the city, resulting in longer wait times for areas like 4, 5, and 6 Kilo, where AAU students reside. In cases where delivery is available, the wait time often exceeds 45 minutes, which makes it inconvenient for students who need quick access to meals between classes or during study hours.

Another limitation is the lack of student-friendly pricing. Existing platforms rarely offer discounts, loyalty programs, or meal bundles tailored to student budgets. Furthermore, smaller, local eateries that sell affordable meals are often excluded from these delivery platforms, as they may not meet the operational requirements or cannot afford the commission fees imposed by these services.

The absence of a dedicated food delivery solution focused on affordability, convenience, and quick service for students creates a significant gap in the market. This gap not only affects students but also limits business opportunities for local restaurants that could benefit from a larger, consistent student customer base. KiloEats aims to address these challenges by building a platform that prioritizes student needs, offering affordable meals, fast delivery, and partnerships with local eateries.

1.3 Statement of the Problem

Addis Ababa University (AAU) students, particularly those residing in the 4 Kilo, 5 Kilo, and 6 Kilo campuses, face significant challenges in accessing affordable and convenient meal options. While several small, budget-friendly local restaurants exist around these areas, they lack the digital infrastructure, such as mobile or web applications, to provide efficient services to students. This absence of technology-driven solutions disproportionately affects students who often do not have the financial means to dine at upscale restaurants that typically offer app-based delivery services.

AAU students also have demanding academic schedules, which leaves them with limited time to visit these small local restaurants in person, wait for their meals, and then either eat on-site or take their food back to campus. The space constraints in many of these eateries also exacerbate the issue, as the restaurants often do not have enough seating capacity to accommodate large numbers of students during peak hours, leading to rushed dining experiences.

Compounding these issues is the lack of sufficient food delivery platforms in Addis Ababa and Ethiopia at large, even for high-end restaurants. As a result, there is no readily available solution for AAU students to access affordable meals from nearby small restaurants through a convenient and time-efficient platform.

KiloEats aims to address these issues by providing a dedicated food delivery system that connects AAU students with affordable local eateries, ensuring access to budget-friendly meals without the need for students to leave campus or disrupt their busy schedules.

1.4 Objective of the Project

1.4.1 General Objective

To provide a convenient, affordable, and accessible food delivery platform that connects Addis Ababa University (AAU) students with local restaurants in the 4 Kilo, 5 Kilo, and 6 Kilo areas, addressing their meal needs efficiently.

1.4.2 Specific Objectives

- **To develop a mobile and web application** that allows students to browse menus, place orders, and track deliveries from local restaurants, ensuring a user-friendly experience.
- **To integrate local restaurants** into the platform, enabling them to manage their menus and pricing, thus expanding their customer base and supporting their growth.
- **To create a delivery management system** that ensures reliable and fast delivery within the designated areas, minimizing delays and improving student satisfaction.
- **To establish a feedback system** where students can rate restaurants and delivery services, improving service quality through continuous feedback.
- **To reduce the time students, spend off-campus** searching for affordable meals by offering a digital alternative that allows them to stay focused on their academic activities.
- **To support small local businesses** by creating a new revenue stream through delivery orders, helping them grow and thrive in a competitive market.
- **To promote student awareness and use of the platform** through marketing campaigns targeting the AAU campuses, ensuring broad adoption.

1.5 Proposed System

KiloEats aims to address the need for an affordable and efficient food delivery system for Addis Ababa University students. The focus is on connecting students with local restaurants in the Arat Kilo, 5 Kilo, and 6 Kilo areas.

User-Friendly Interface

The website will feature:

- **Intuitive Design:** A clean and simple layout that makes navigation easy for users of all tech levels.
- **Mobile Responsiveness:** Ensures accessibility on smartphones and tablets, crucial for students on the go.
- **Search and Filter Options:** Allows users to quickly find restaurants based on type of food, price, and distance.

Real-Time Updates

- **Integration with Restaurants:** Direct links to restaurant systems will provide live updates on menu items, availability, and pricing.
- **Order notification:** Students will be notified after their orders have arrived in real time, enhancing transparency and trust.

Efficient Delivery Network

- **Local Partnerships:** Utilizes local delivery personnel familiar with the area for faster service.
- **Optimized Routes:** Uses algorithms to determine the quickest delivery routes, minimizing wait times.

1.6 Feasibility Study

1.6.1 Economic Feasibility

1.6.1.1 Startup Costs

- **Technology Investment:** Minimal costs due to the use of open-source technologies and existing digital infrastructure.
- **Operational Costs:** Includes website maintenance, delivery logistics, and minimal staffing needs.

1.6.1.2 Revenue Streams

- **Service Fees:** Small fee per transaction, providing a steady income stream.
- **Restaurant Partnerships:** Restaurants can pay for premium listings or promotional features, offering additional revenue.

1.6.1.3 Market Demand

- **Target Audience:** Primarily students who need affordable and convenient food options.
- **Growth Potential:** Potential to expand services to other university areas or cities in the future.

1.6.2 Technical Feasibility

1.6.2.1 Technology Stack

- **React:** Used for the frontend to create a dynamic, responsive user interface.
- **Nest.js:** Handles backend operations efficiently, supporting asynchronous tasks.
- **MongoDB:** Manages data, offering flexibility and scalability.

1.6.2.2 System Integration

- **POS Integration:** Connects with restaurant point-of-sale systems to ensure accurate data flow.
- **API Development:** Custom APIs will facilitate smooth communication between different system components.

1.6.2.3 Scalability and Reliability

- **Cloud Hosting:** Ensures the platform can handle increased traffic and data loads.
- **Redundancy and Backups:** Implemented to prevent data loss and ensure service continuity.

1.6.3 Schedule Feasibility

1.6.3.1 Development Timeline

- **Phase 1 (Weeks 1 & 2):** Planning and prototype development, including basic features and design.
- **Phase 2 (Weeks 3, 4 & 5):** Beta testing and refinement based on user feedback, focusing on usability and functionality.

1.6.3.2 Milestones

- **Prototype Completion:** End of week 2, allowing for initial evaluations.
- **Beta Testing:** By Week 5, involving real users to fine-tune the platform.
- **Launch:** Scheduled for Month 2, with marketing efforts to attract users.

1.6.3.3 Resource Management

- **Team Structure:** Includes developers, UI/UX designers, project managers, and marketing specialists.
- **Budget Allocation:** Funds distributed across technology, marketing, and operations to ensure balanced development.

KiloEats is set to transform food accessibility for students, fostering strong community ties and supporting local businesses.

1.7 Scope

The KiloEats initiative focuses on creating a web application designed to provide affordable food delivery services tailored for students at Addis Ababa University (AAU). This project seeks to meet the specific needs of students by connecting them with local small restaurants, often referred to as "mother bets," that offer budget-friendly meal options.

Project Deliverables

KiloEats will encompass several essential features:

- **User Registration and Profile Management:**
Students will have the ability to register on the platform using their email addresses. Each user will maintain a profile to manage personal details conveniently.
- **Restaurant Listings:**
The application will present a carefully curated list of local small restaurants surrounding the university. Each restaurant will have a dedicated page showcasing its menu, including food items, descriptions, and prices.
- **Ordering System:**
Students will benefit from a streamlined interface that allows them to explore menus, select food items, and place orders for delivery. Customization options will also be available to cater to individual meal preferences and dietary needs.
- **Delivery Coordination:**
The application will feature a system for coordinating deliveries, ensuring that food reaches students at designated locations, such as dormitories or study areas. Students will also have access to tracking features to monitor their order status in real time.
- **Feedback and Rating System:**
To facilitate continuous improvement, students will be able to leave reviews and ratings for the small restaurants and their offerings. This feedback mechanism will be integral to enhancing service quality.
- **Payment Mechanism (Tele-birr Integration)**
Integration of a payment system through Tele-birr will enable students to complete transactions securely within the app. This will streamline the payment process, allowing for a more seamless and efficient user experience.

Project Exclusions

To maintain clarity and focus, certain aspects will not be included in the initial scope of the KiloEats project:

- **High-End Restaurant Partnerships:**
The platform will exclude listings for upscale restaurants or premier restaurants that do not align with the affordability expected by students.

- **Mobile Application Development:**
The initial phase will concentrate solely on the web application. The development of a mobile app may be considered in future phases based on student interest and feedback.
- **Inventory Management for Restaurants:**
KiloEats will not provide inventory management solutions for participating small restaurants; each restaurant will independently manage its stock and menu updates.
- **International Food Delivery:**
The service will focus exclusively on local small restaurants within the vicinity of AAU, with no plans to include international cuisine or deliveries from outside the local area at this time.
- **Advanced Features:**
Features such as advanced analytics for restaurants, loyalty programs, or personalized marketing strategies will not be part of the initial launch but may be evaluated for future updates based on user feedback.

This scope of KiloEats aims to establish a clear framework for the development process, ensuring that the project remains dedicated to providing a valuable service that directly addresses the needs of AAU students.

1.8 Methodology

The KiloEats project will adopt the Waterfall methodology, which provides a structured and sequential framework for development over the next three months. This approach is particularly well-suited to our project as it emphasizes distinct phases, allowing us to clearly define each stage of the development process. By following this methodology, we can effectively track our progress and ensure that all critical components of the project are comprehensively addressed. The Waterfall model's focus on thorough documentation and clear deliverables will help guide our team as we work to create a successful and impactful food delivery platform for students.

1.8.1 Justification for Choosing the Waterfall Methodology

The decision to utilize the Waterfall methodology as our primary development framework is informed by several critical factors that align with our current capabilities, project objectives, and learning goals.

1. Structured Approach

The Waterfall model is characterized by its linear and sequential phases, which include:

- Requirements gathering
- Design
- Implementation
- Testing
- Deployment

This structured approach is particularly beneficial for our team as it provides a clear roadmap to follow. Each stage must be completed before moving on to the next, reducing the likelihood of confusion or overlap in tasks. For beginners, this clarity is essential in ensuring that we focus on mastering one aspect of development at a time.

2. Clear Documentation

One of the significant advantages of the Waterfall model is its emphasis on documentation at each phase. This focus allows us to create comprehensive records of our requirements, design decisions, and testing results. As first-time developers, having detailed documentation will serve multiple purposes:

- **Knowledge Retention:** Aids in retaining the knowledge we gain throughout the project, making it easier for team members to refer back to decisions and rationale.
- **Communication:** Enhances communication among team members and stakeholders, ensuring everyone is aligned with the project goals and progress.
- **Future Projects:** Serves as a reference for future projects, helping us to improve and build upon our initial experiences.

3. Focus on Learning

Given our current programming skills and the necessity to learn new technologies and frameworks, the Waterfall model allows us to dedicate specific timeframes to each phase of development. This focused approach is advantageous for several reasons:

- **Skill Development:** Concentrating on one phase at a time enables us to allocate our learning efforts effectively. For example, during the requirements phase, we can focus on understanding user needs and gathering insights without the distraction of simultaneous coding tasks.
- **Foundation Building:** The sequential nature of the Waterfall model enables us to build a solid foundation in programming languages, frameworks, and tools before moving on to more complex tasks. This is crucial as we aim to develop a robust application.

4. Risk Management

Although the Waterfall model is less flexible compared to iterative methodologies, its structured phases allow for early identification of potential risks. By completing thorough requirements gathering and design before implementation, we can address significant issues upfront. This reduces the likelihood of encountering major problems later in the development process, which is particularly beneficial for a team with limited experience.

5. Simplicity in Progress Tracking

The Waterfall model facilitates straightforward progress tracking through its defined phases. Each phase has specific deliverables and timelines, enabling us to monitor our advancement toward project completion. This clarity will help in managing deadlines and ensuring that we remain on schedule throughout the three-month development period.

1.8.2 Project Phases

1. Requirements Gathering Phase

In the first month of the KiloEats project, our primary focus will be on defining the project requirements. This critical phase will involve comprehensive data collection to ensure that we fully understand the needs and expectations of our target audience.

➤ Interviews and Surveys

We will conduct structured interviews and surveys with a diverse group of stakeholders to gather valuable insights. This will include:

- **Students:** We aim to engage approximately 25 students from various departments within Addis Ababa University. These discussions will help us identify their food delivery needs, preferences, and pain points. By gathering a range of perspectives, we can ensure that the platform is tailored to meet the diverse requirements of our student body.
- **Local Restaurant Owners:** We will engage with around 3 local restaurant owners to understand their operational capabilities, willingness to participate in the KiloEats platform, and any specific requirements they might have for integration. This dialogue will be essential for building partnerships that are mutually beneficial.

➤ Timeline

We plan to conduct these interviews and surveys within the first two weeks of the project. Following this initial data collection, we will analyze the gathered information to identify common themes, specific user needs, and functional requirements that will guide the development of the KiloEats platform.

➤ Documentation

The requirements identified during this phase will be meticulously documented in a detailed Requirements Specification Report. This report will outline user needs, functional requirements, and system constraints, serving as a foundational document for the subsequent phases of the project. By ensuring clarity and comprehensiveness in our documentation, we aim to align all stakeholders on project objectives.

2. Design Phase

Following the requirements gathering phase, we will transition to the design phase in the second month. This phase is crucial for visualizing the application and ensuring that it meets the expectations of our users.

➤ System Design

We will create detailed wireframes and UI mockups to visualize the application's interface and user experience. This process will involve iterative design reviews to refine our visuals and ensure user-friendliness. Our goal is to create an intuitive interface that facilitates easy navigation for students.

➤ **Defining Constraints**

During the design phase, we will identify key constraints related to usability, performance, and maintainability. For example:

- Usability: The application must be intuitive, allowing students to navigate quickly and efficiently. We will focus on minimizing the number of steps required to place an order.
- Performance: The application should load within three seconds to ensure a smooth user experience. We will set performance benchmarks to guide our development efforts.

➤ **Comparative Analysis**

To inform our design decisions, we will conduct a comparative analysis of alternative design strategies. This analysis will help us choose the most effective approach based on functionality and user feedback gathered from our initial research. By evaluating existing solutions, we can leverage best practices and avoid common pitfalls.

➤ **Documentation**

All design decisions, constraints, and methodologies will be documented thoroughly to ensure clarity and alignment with the project goals. This documentation will serve as a reference throughout the development process, facilitating communication among team members.

1.8.3 Implementation Methodology

In the third month, we will shift our focus to the implementation of the KiloEats platform.

1. Technology Stack

We plan to utilize the following tools and frameworks to ensure a robust and scalable application:

- Frontend: We will use React.js to develop a responsive user interface that enhances the user experience across devices.
- Backend: Nest.js with Express.js will be employed to handle server-side logic, providing a solid foundation for our application's functionality.
- Database: MongoDB will be used to manage user data and restaurant menus, allowing for efficient data retrieval and storage.

2. Incremental Development

We will adopt an incremental approach to implementation, developing the application in stages. This will facilitate continuous integration of feedback from our initial testing, enabling us to make necessary adjustments in real-time.

3. Version Control

To manage our codebase effectively and facilitate collaboration among team members, we will utilize Git for version control. This will help us track changes, manage branches, and ensure that all team members are aligned on the latest code updates.

1.8.4 Testing Plan

Testing will be integral throughout the project, particularly in the final month, to ensure the KiloEats platform meets quality standards.

Types of Testing

We will implement various types of testing to comprehensively assess the application:

- **User Acceptance Testing:** We will conduct usability testing with a group of 25 students to ensure that the application meets their needs and is easy to use. Feedback from this testing will be critical for refining the user experience.
- **System Testing:** Comprehensive testing will be performed to validate the functionality of the application, including all user interactions. This will ensure that all components work as intended.
- **Integration Testing:** We will test the interaction between the frontend and backend systems to ensure seamless data flow. This will help us identify any issues in the communication between different parts of the application.
- **Unit Testing:** In addition to the above testing types, we will implement unit testing to validate individual components of the application. This involves testing each function or method in isolation to ensure that it behaves as expected.

1.8.5 Success Assessment

The success of the KiloEats platform will be assessed based on user feedback, application performance metrics (such as load times), and the overall satisfaction of both students and restaurant owners. We will establish criteria for success that align with our project goals, ensuring that we deliver a valuable service to our users.

By following this structured methodology, we aim to develop KiloEats efficiently and effectively, ultimately providing a valuable food delivery service tailored to the needs of students at Addis Ababa University. This comprehensive approach will ensure that we address all critical aspects of the project, from initial requirements gathering to final testing and deployment.

1.9 Project Management Plan

1.9.1 Time Management Plan

The following Gantt chart outlines the project's timeline, highlighting the duration of each event. By following this plan, we can ensure that the project stays on track and meets all deadlines on time.

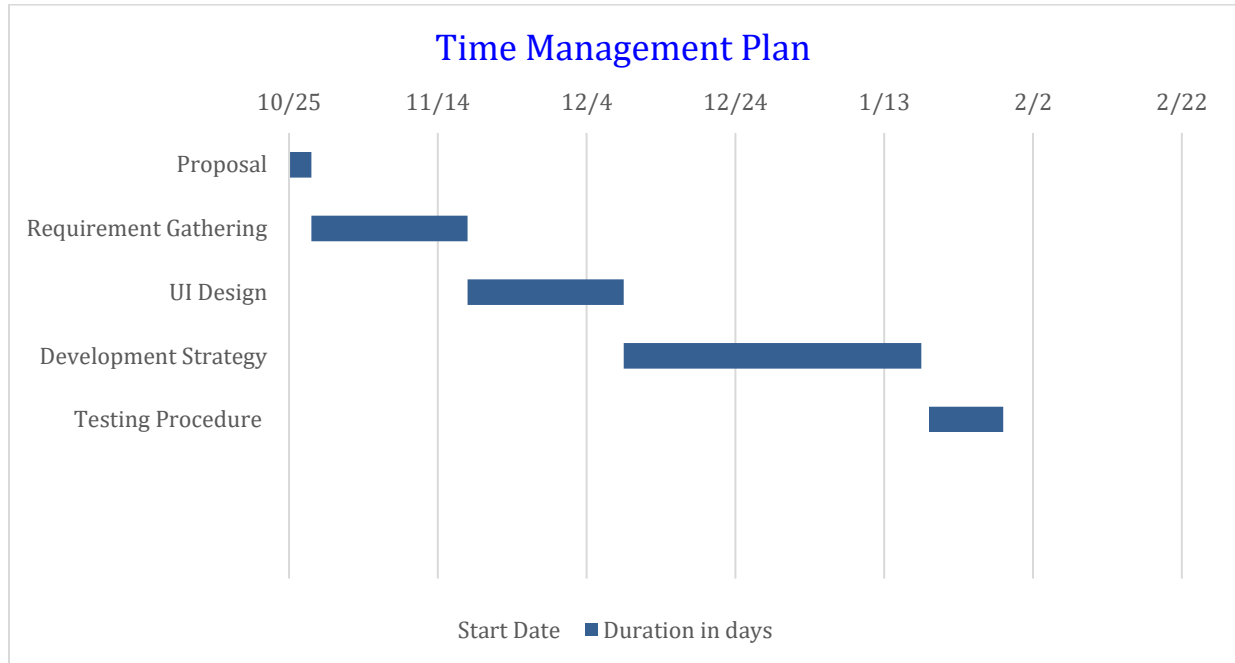


Table 1

1.9.2 Quality Management Plan

Objective: Guarantee that the KiloEats platform consistently adheres to high quality standards, while also delivering an intuitive, effortless user experience for every customer. This includes regular updates, rigorous testing, and continuous monitoring to identify and address any issues promptly. Through these measures, KiloEats ensures not only satisfaction but delight for all its users.

The quality standards we work towards for KiloEats are:

- ✓ Established on the industry's best methods and employ efficient service
- ✓ Ensure the platform is designed to be secure, scalable and user-friendly.

Potential risks associated to KiloEats are:

- **Data Breaches:** The protection of sensitive customer information is paramount. Any breach could result in severe consequences, ranging from financial losses to irreversible damage to KiloEats' reputation.
- **Scalability and Performance Issues:** As KiloEat's popularity grows, a sudden increase in users during peak hours could strain the platform, leading to the system crashing, slowing down and causing poor user experience.

- **Delivery Delays:** Insufficient logistics, traffic congestion or other unforeseen circumstances could result in significant delays in the food delivery. This negatively impacts customer satisfaction.
- **Vendor Reliability:** The performance and reliability of partner restaurants and vendors are crucial to KiloEats' success. If these partners fail to meet quality standards or timing commitments, it reflects poorly on the platform.
- **Regulatory Compliance:** Adhering to local food safety regulations and data privacy laws is essential to avoid legal challenges and fines.

Project Management Plan to Enable Quality:

- Implement advanced encryption protocol to secure sensitive data, and utilize multi-factor authentication to bolster account security. Conduct regular security audits and penetration testing to proactively identify and address vulnerabilities. Additionally, establish an incident response plan to quickly mitigate any breaches that may occur.
- Design the platform with a scalable architecture using cloud-based services to accommodate varying loads. Implement load balancing to distribute traffic efficiently and maintain performance during peak times. Conduct capacity planning to ensure infrastructure can handle future growth.
- Partner with reliable delivery services and create contingency plans for unexpected challenges. Enhance communication with customers to provide accurate delivery updates and address any delays promptly.
- Establish stringent criteria for vendor selection, focusing on quality, reliability and timely delivery. Develop strong partnerships with vendors through regular communication and performance-based incentives.
- Stay informed about local and international regulations related to food safety and data privacy. Implement compliant processes and conduct regular training sessions for staff to ensure understanding and adherence to regulations. Engage with legal experts to review and update compliance measures as needed.

By combining these strategies and implementing multiple rigorous tests to ensure final product, the KiloEats development process will be reliable, secure and perform as intended. The tests will include:

- ✓ **Unit Testing:** Perform detailed testing of individual components to ensure they function correctly in isolation. This involves verifying that each unit of code operated as expected and identifying any defects early in the development process.
- ✓ **Integration Testing:** Conduct tests to verify that different components of the system work together seamlessly. This step ensures that the interaction between integrated units does not cause unexpected issues and that the system operated correctly and ensure smooth communication between modules.
- ✓ **System Testing:** Validate the functionality of the entire system by testing it as a complete integrated product. This phase involves checking the system against the

specified requirements to ensure it meets all functional and non-functional criteria. System testing covers various aspects such as performance, security and usability, ensuring the software is ready for deployment.

- ✓ **User Acceptance Testing (UAT):** Engage a group of target users to conduct testing to ensure the system meets their needs and expectations. UAT involves real-world scenarios and end-user involvement to validate that the software delivers the intended value and user experience. Feedback from UAT helps identify any final adjustments needed before the system goes live.

1.9.3 Communication Management Plan

1.9.3.1 Internal communication:

Frequent meetings among members.

Purpose	Method	Frequency	End goal
Team kick-off	Google meet and in person	Twice using both the methods before submitting our proposal.	Preparing our proposal and team member introduction.
Project start up week	In person	Twice in the first week after the proposal is submitted	Analyzing our resources
Weekly coordination	In person	Once a week on Mondays after class	Clarify responsibilities and expectations for the end of the week.
Demo meeting	Google meet	Once a week on Sundays	Update our week progress
Code review	Online	Once a week	Review our codes before merging
Testing	Online and in person	Thrice after the development of the software has ended.	Test our software

Table 2

1.9.3.2 External communication:

Contacting stakeholders.

While doing our project, stakeholders include restaurant owners, primary users or customers, delivery personnel and our instructor Mr. Aderaw.

Purpose	Frequency	End goal
Submitting report	Once in three weeks	Presenting our well done job to our instructor.
Contacting restaurant owners	On significant updates	Registering restaurants and food providers.
Addressing users	A week before the product is launched	Promoting our product online.
Seeking candidates	The week after the restaurants have registered	Recruiting delivery personnel

Table 3

APPENDIX

Table 1

Requirement Gathering (3 Weeks)

- **Objective:** Establish a clear understanding of user needs and project specifications.
- **Activities:**
 - Conduct interviews and surveys with stakeholders.
 - Analyze existing systems and gather requirements.
 - Document functional and non-functional requirements.

UI Design (3 Weeks)

- **Objective:** Create a user-friendly interface that enhances user experience.
- **Activities:**
 - Develop wireframes and prototypes.
 - Conduct usability testing sessions with potential users.
 - Refine designs based on feedback.

Development Phase (6 Weeks)

- **Objective:** Build and refine core functionalities of the application.
- **Activities:**
 - Implement frontend and backend components.
 - Conduct regular code reviews and integration sessions.
 - Ensure alignment with requirements and design specifications.

Testing Procedures (10 Days)

- **Objective:** Validate the application through various testing methodologies.
- **Activities:**
 - Perform unit testing, integration testing, system testing, and user acceptance testing.
 - Address any identified issues and ensure functionality meets requirements.

REFERENCE

- Addis Ababa University guidelines for project proposals
- [GeeksforGeeks - Design a webpage for online food delivery system](#)