



**A Proposal on
Food Ordering Web Application
Web Application Name: Malt**

**Assessment Type
FYP Proposal
5th Semester
2023/24 Autumn**

Student Name: Manish Dhamala

London Met ID: 22069006

College ID: NP04CP4A220070

Proposal Due Date: Friday, August 30, 2024

Proposal Submission Date: Tuesday, August 27, 2024

Submitted To: Informatics College Pokhara

I confirm that I understand my proposal needs to be submitted online via MST Classroom under the relevant module page before the deadline in order for my proposal to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded.

Table of Contents

1. Introduction	1
1.1 Problem Statement.....	2
1.2 Project as Solution	2
2. Aims and Objectives.....	3
3. Expected Outcomes and Deliverables	4
3.1 Expected Outcomes	4
3.2 Deliverables.....	4
4. Project Risks, Threats and Contingency Plans.....	5
4.1 Risks and Threats.....	5
4.2 Contingency Plans.....	5
5. Methodology.....	6
5.1 Software Development Life Cycle (SDLC).....	6
5.2 Selected Methodology (Incremental model)	8
6. Resource Requirements.....	10
7. Work Break Down Structure	11
8. Milestones	12
9. Gantt Chart.....	13
10. Conclusion	14
References.....	15

Table of Figures

Figure 1: Software Development Life Cycle (SDLC)	6
Figure 2: Incremental model.....	8
Figure 3: Work Break Down Structure (WBS)	11
Figure 4: Milestones	12
Figure 5: Gantt Chart.....	13

1. Introduction

Malt is a web application that allows you to order food from restaurants online. The name "**Malt**" comes from combining '**salt**' an ingredient found in almost every dish, with the first letter of my name, **Manish**.

In the quick evolving digital world, online services have become an essential part of everyday life, especially in urban areas. In this world, where technological adoption is rapidly increasing, the demand for online food ordering platforms has grown considerably. Traditional methods of ordering food over the phone or in person are gradually being replaced by more efficient, user-friendly web applications which relate to the fast-paced lifestyle of today's consumers. (Gautam, 2021) However, the existing food ordering solutions often lack the customization, simplicity, and localized features required to meet the specific needs and preferences of the people.

This project aims to provide a comprehensive and user-friendly food ordering web application. The program will not only simplify the ordering process for customers, but it will also give a platform for local restaurants to reach a larger audience and manage orders more efficiently. By using modern web technologies, the project seeks to address the gap in the current market, providing a solution that is both user-friendly and aligned with the technological capabilities and cultural context of Nepal.

1.1 Problem Statement

The problems currently faced by the Nepali people are as follows:

- The Nepalese market has a limited availability of applications for ordering food online. (Shrestha & Shrestha, 2018)
- People still rely on phone calls or messages for ordering food, but these methods can be unreliable and frustrating.
- Restaurants missed out on revenue opportunities because they lacked an online presence.
- Restaurant staffs often forget or write down incorrect orders, leading to inefficient order management.
- Managing a restaurant online is challenging due to a complex user interface and design. (Restolabs, 2023)

1.2 Project as Solution

The development of the food ordering web application "Malt" will provide the following solutions:

- This project "**Malt**" will fill the gap in the Nepalese market by offering a dedicated and efficient platform for ordering food online, addressing the limited availability of such applications.
- "Malt" will eliminate the need for unreliable phone calls or messages by simplifying the food ordering process for customers.
- The application will help restaurants increase their revenue by providing them with a robust online presence, enabling them to reach a broader audience.
- "Malt" will improve order accuracy and efficiency by offering a digital platform that minimizes human errors in order management.
- The platform will have a simple and user-friendly interface, making it easier for restaurant owners to effectively manage their online operations.

2. Aims and Objectives

- The main Aim of this project is:

To create an advanced, user-friendly web application that simplifies the online food ordering process for customers and provides a robust, efficient platform for restaurants to manage and increase their online presence and revenue.

- The objectives of the project are as follows:

- **Develop a user-friendly interface** that makes it easy for customers to browse, select, and order food online.
- **Create a reliable and secure platform** for handling online orders, payments, and customer data.
- **Enable restaurants to easily manage orders** through a simple dashboard.
- **Implement features for order tracking**, so customers can follow their order status in real-time.
- **Provide customization options for restaurants**, allowing them to update menus and prices.
- **Optimize the application for fast loading times** to enhance user experience, even on slower internet connections.

3. Expected Outcomes and Deliverables

3.1 Expected Outcomes

These are the outcomes that can be expected after completion of the project:

- **Enhanced Accessibility:** Users in Nepal will have easy access to a reliable online platform for ordering food, reducing dependency on traditional methods like phone calls.
- **Increased Restaurant Revenue:** Restaurants will benefit from a broader online presence, leading to increased revenue opportunities.
- **Improved Order Management:** Restaurants will experience fewer errors in order processing, leading to more efficient operations and higher customer satisfaction.
- **User-Friendly Experience:** Both customers and restaurant owners will find the application easy to use, because of its simple interface.

3.2 Deliverables

The following deliverables will be provided upon completion of the project:

- **Functional Web Application:** A fully operational web app, "Malt," with features like food browsing, order placement, payment processing, and order tracking.
- **Restaurant Dashboard:** A user-friendly dashboard for restaurants to manage orders, update menus and price.
- **Project Documentation:** A well-documented report detailing the development process, design, and system architecture of the application.
- **Source code:** A complete source code of the application.

4. Project Risks, Threats and Contingency Plans

4.1 Risks and Threats

These are the risks and threats associated with the project:

- **Technical Challenges:** The development process might encounter bugs or technical difficulties that delay progress.
- **Slow Internet Connections:** Users in Nepal may experience slow internet speeds, affecting the performance of the application.
- **Limited Technical Skills:** Lack of advanced technical skills might hamper the implementation of complex features.
- **Cybersecurity Risks:** The platform could be targeted by cyber-attacks, leading to data breaches or service disruptions.
- **No Backup:** Failure to maintain backups could result in data loss, especially during unexpected failures or accidents.

4.2 Contingency Plans

These are the contingency plans I've developed to address or minimize risks and threats:

- **For Technical Challenges:** I've allocated extra time for testing and debugging, and I will actively seek assistance from online forums or my supervisor if needed.
- **For Slow Internet Connections:** I'll optimize the application for low-bandwidth environments and ensure it's tested on various internet speeds to maintain performance consistency.
- **For Limited Technical Skills:** My approach is to prioritize building core features first, gradually introducing more advanced functionalities as I enhance my skills.
- **For Cybersecurity Risks:** I'll implement strong security measures, including encryption, and prepare a response plan to quickly address any potential cyberattacks.
- **For Backup:** I will use **GitHub** regularly for version control and backups, ensuring easy restoration in case of data loss.

5. Methodology

5.1 Software Development Life Cycle (SDLC)

Software Development Life Cycle (SDLC) is a framework defining processes that are aimed to produce software with the lowest cost, highest quality, and in the shortest time. (Kukhnavets, 2019)

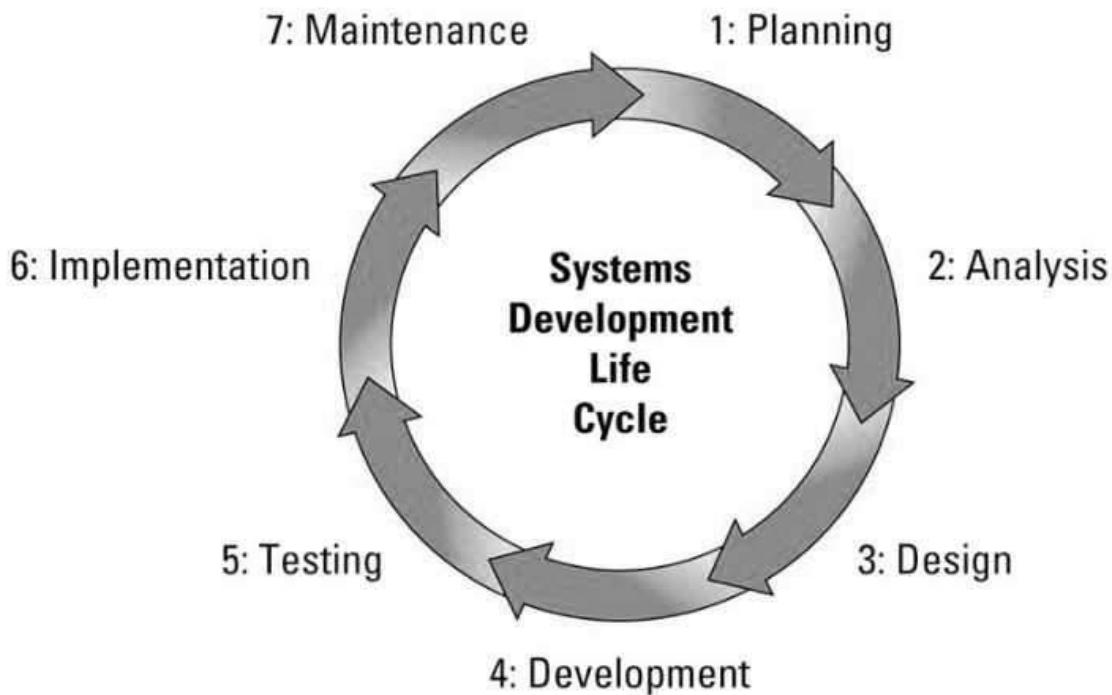


Figure 1: Software Development Life Cycle (SDLC)

Image source: <https://welldoneby.com/blog/wp-content/uploads/2019/10/SDLC-cycle.jpg>

The 7 phases of the Software Development Life Cycle (SDLC) are explained below:

1. **Planning:** Identifying the goals and requirements of the project, deciding on the scope, and creating a basic plan.
2. **Analysis:** Gathering detailed information about what the system should do and analyze the needs of the users.

3. **Design:** Creating a blueprint of the software. This includes designing the system's architecture, user interfaces, and data models.
4. **Development:** Writing the actual code based on the design. This is where the software is developed.
5. **Testing:** Checking for any bugs or issues in the software. Making sure that it works as expected and meets the requirements.
6. **Deployment:** Releasing the software to users. It's now ready to be used in the real world.
7. **Maintenance:** After deployment, any new issues that arise are fixed, and the application is updated or improved as needed.

This cycle helps to ensure that the software is well-planned, meets user needs, and is reliable.

5.2 Selected Methodology (Incremental model)

After evaluating various software methodologies, the **incremental model** seems most suitable for this project. The Incremental Model in software engineering is a method where the project is divided into smaller, manageable parts or increments (Sachan, 2024). Each increment goes through the processes of gathering requirements, designing, testing, and implementation in this approach. These increments are built upon one another, gradually adding functionality to the software until the final product is complete.

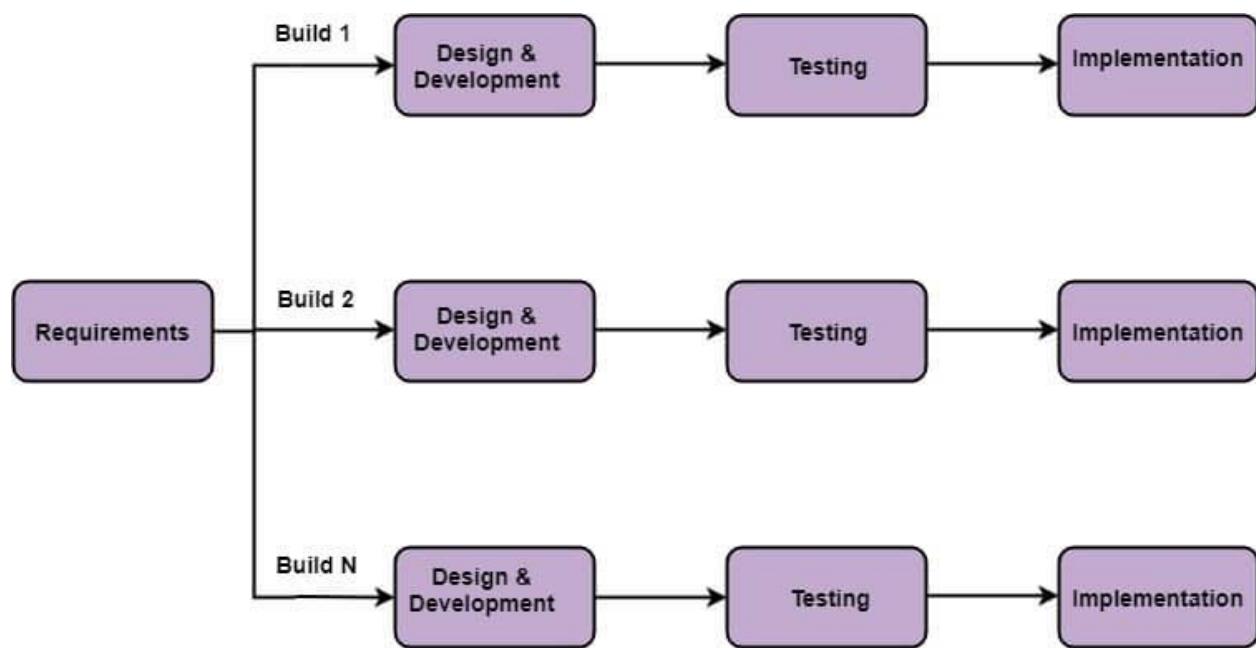


Figure 2: Incremental model

Image Source: <https://static.javatpoint.com/tutorial/software-engineering/images/software-engineering-incremental-model.jpg>

The reason for choosing incremental model is because it allows for adjustments at any point in the development process. If something needs to be added, it can be included in the next version or increment. Each iteration adds new functionality, allowing for the completion of essential features first. To achieve the final product, the project must implement several functionalities. Thus, completing a few features in each iteration is the most effective approach.

According to my plan, the project will be completed in three increments. The initial requirement gathering must be finished before any increments can begin. The first increment will include the completion of the wireframe design, database design, user login and registration page, home page, and the release of the initial version. The second version will be released after the cart page, user profile page, restaurant detail page, and payment gateway integration have been built in the second increment. The admin panel will be developed in the final increment. Then, the full version will be released along with the final documentation.

The various phases of the incremental model that I will be following while developing the application are as follows:

- i. **Requirement Analysis:** First, I will gather all the necessary details about what the web app needs to do. This includes features like user registration, browsing menus, placing orders, and payment options. I will break these requirements down into smaller, manageable parts that I can develop one by one.
- ii. **Design and Development:** Once I have a clear set of requirements, like the user registration feature, I will move on to design. I will plan out how this feature will look and function, including the layout and user interactions. After that, I will start coding to develop the feature. I will repeat this process for each feature.
- iii. **Testing:** After developing a feature, such as user registration, I will thoroughly test it to ensure that it works as expected. This includes checking for bugs and making sure all data is correctly handled. If there are any issues, I'll fix them before moving forward.
- iv. **Implementation:** Once a feature is tested and working properly, I'll implement it by adding it to the live application. This means it's now ready for users to interact with. I'll then move on to the next feature, like menu browsing, and continue the process until the entire app is complete.

6. Resource Requirements

These are the list of resources that are required for this project:

1. Software:

- **Frontend Development:** I will use **React** for building the user interface.
- **Backend Development:** **Spring Boot** will be used for the backend, handling business logic, APIs, and connecting to the database.
- **Database Management System:** **PostgreSQL** will be used to manage and store data for the application.
- **Documentation:** **MS Word** will be used for writing and organizing project documentation, including reports and design details.
- **Development Environments:**
 - **IntelliJ IDEA:** This will be my primary IDE for developing the backend with Spring Boot.
 - **Visual Studio Code:** I will use this editor for frontend development with React.
- **Version Control:** Git, with GitHub for version control and backup.
- **Diagrams:** **draw.io** will be used for creating diagrams like flowcharts, ER diagrams, and system architecture.
- **Design Tools:** Figma or Canva for UI/UX design.

2. Publications and References:

- **Research Papers and Journals:** Access to publications that discuss best practices in React, Spring Boot, and database management.
- **Datasets:** Sample datasets to test the application's functionality, including user information, menu items, and orders.

3. Company's IT Resources:

- If partnering with a restaurant or food service, I may need access to their IT resources, like their menu database or payment system APIs, for integration purposes.

7. Work Break Down Structure

A Work Breakdown Structure (WBS) is a project management tool that breaks down a project into smaller, more manageable parts. (Organ, 2024) Here is a Work Break Down Structure of my Project:

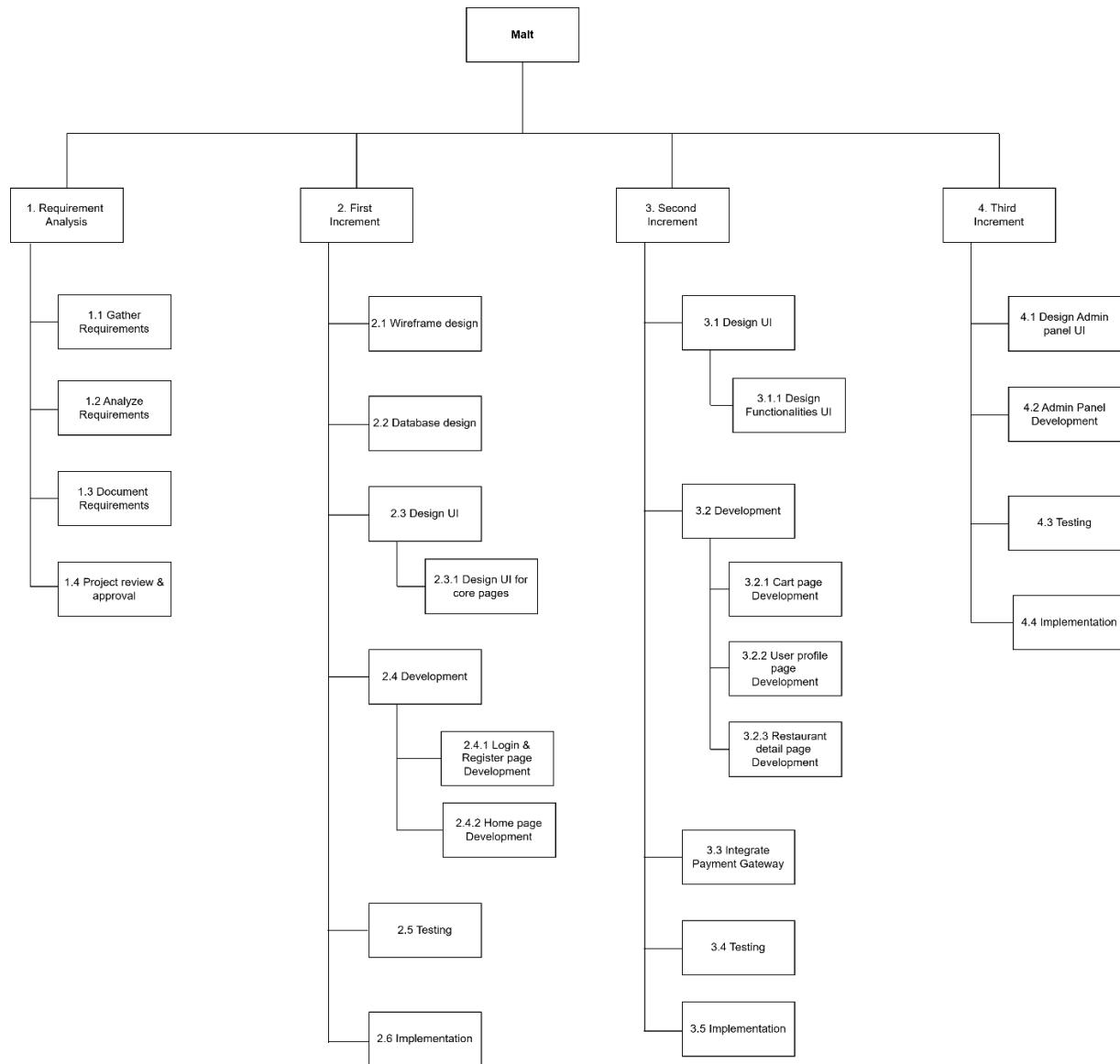


Figure 3: Work Break Down Structure (WBS)

The Work Breakdown Structure (WBS) for the '**Malt**' project provides a well-organized approach to incremental development. By dividing the project into different phases such as Requirement Analysis, First, Second, and Third Increment the WBS ensures clear planning, efficient execution, and complete testing.

8. Milestones

A milestone is a specific point within a project's life cycle used to measure the progress toward the ultimate goal. Milestones in project management are used as signal posts for a project's start or end date, submission of a major deliverable, etc. (shopdev, 2023)

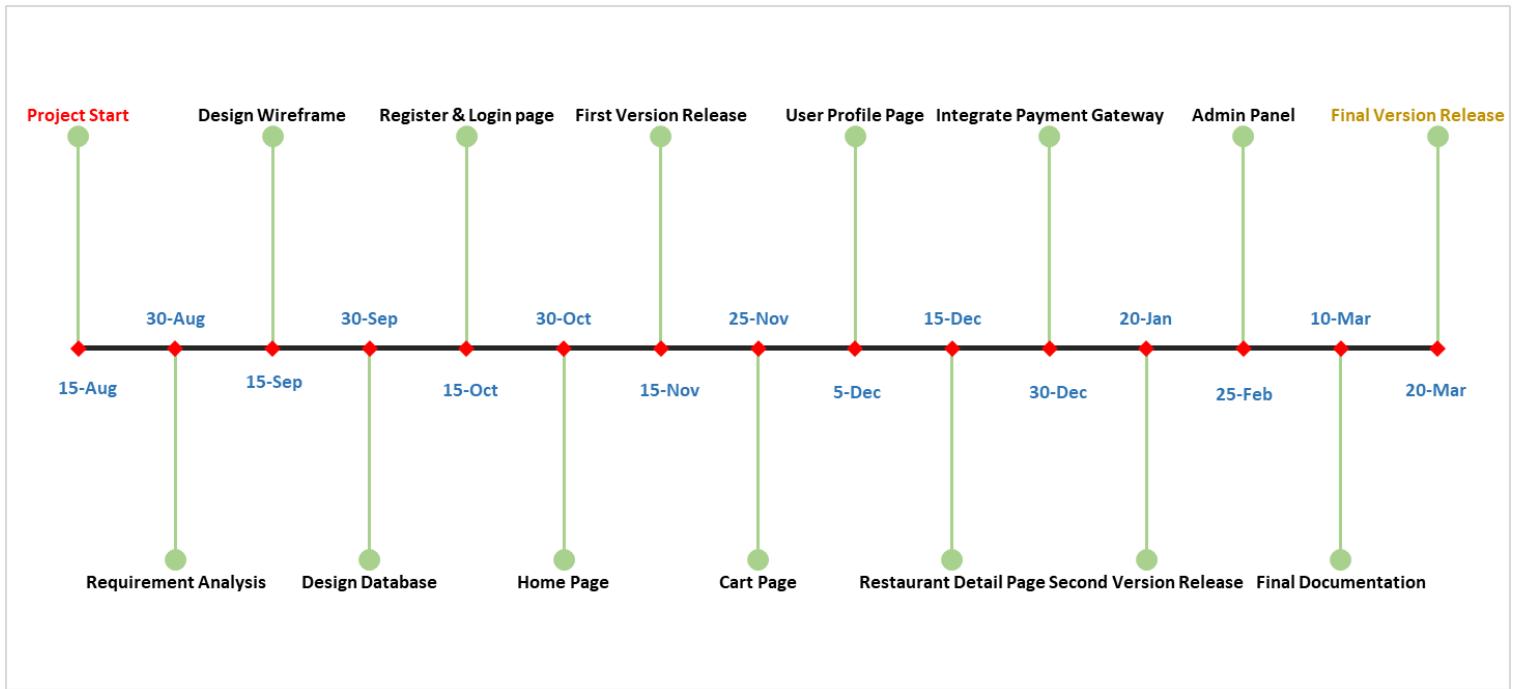


Figure 4: Milestones

The above project milestones provide a clear roadmap for the development of the 'Malt' food ordering web application. Starting from August 15, 2024, and concluding on March 20, 2025, these milestones ensure a structured approach for completing each phase of the project.

9. Gantt Chart

A Gantt chart is a visual project plan that lists tasks and milestones on the vertical axis, with time plotted on the horizontal axis. Gantt charts are commonly used in project management to schedule, track, and communicate deliverables, deadlines, dependencies, and resource assignments. (Adobe Communications Team, 2022)

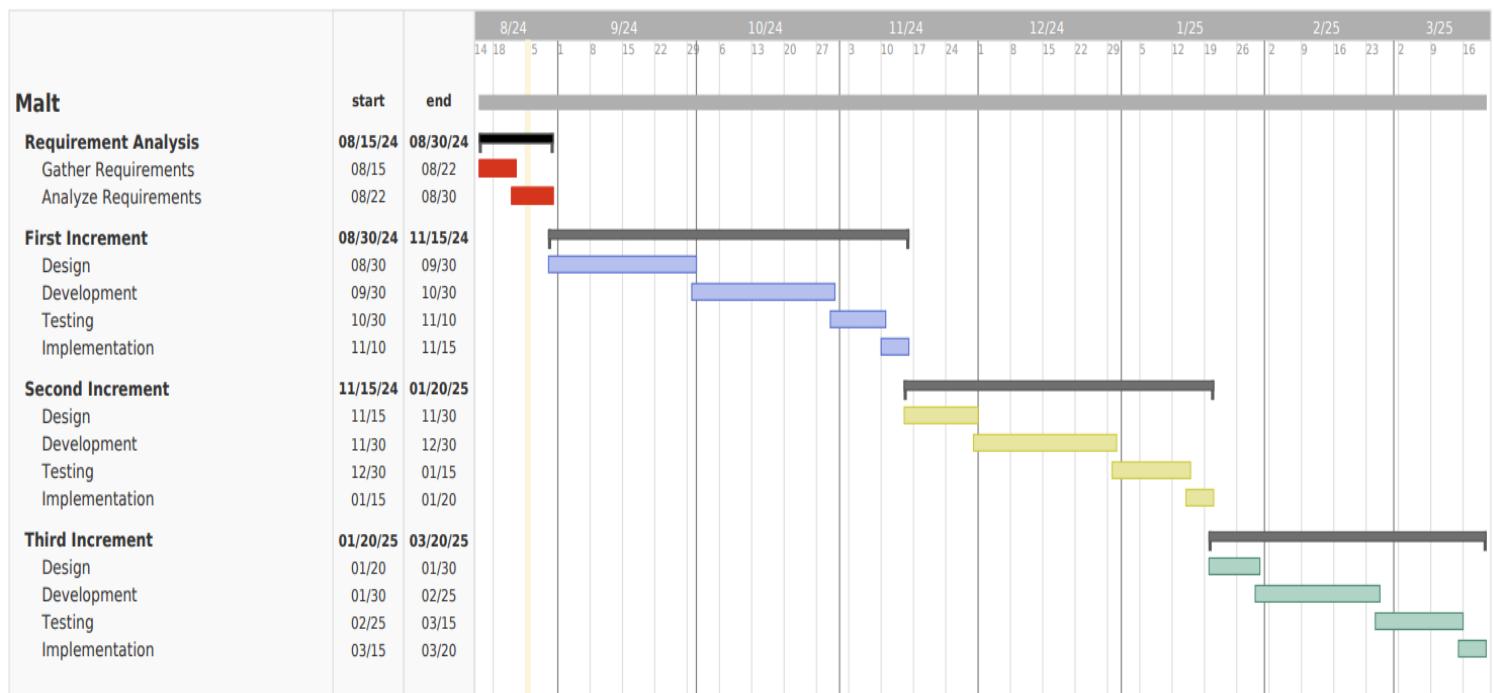


Figure 5: Gantt Chart

The above Gantt chart shows a clear step-by-step plan for building the project using the incremental model. Each increment focuses on specific functionalities, allowing for continuous feedback and improvement. This approach helps to keep the project on track, ensuring all tasks are completed on time for a successful finish.

10. Conclusion

The "**Malt**" project aims to transform food ordering in Nepal by creating an easy-to-use platform for customers and restaurant owners. It addresses problems with current methods, like unreliable phone calls and the lack of online presence for many restaurants. The proposed application uses modern web technologies for a smooth and secure user experience. The **incremental** development approach allows for continuous improvement based on user feedback, ensuring the final product is effective and user-friendly.

Working on the "Malt" project will offer me a valuable learning opportunity and practical experience in real-world software development. From designing user interfaces to implementing complex functionalities and ensuring security, I will gain hands-on experience with a wide range of technologies and development practices. This experience will help me grow my skills and prepare me better for future projects in the tech industry.

Besides giving customers a convenient service, "Malt" offers big benefits to restaurants, like better order management and the ability to easily update menus and prices. In the end, "Malt" is more than just an app, it is a complete solution made to meet the changing needs of today's digital world, ensuring a positive impact on the food industry and its customers.

References

- Adobe Communications Team, 2022. *business.adobe.com*. [Online]
Available at: <https://business.adobe.com/blog/basics/what-is-a-gantt-chart>
[Accessed 24 August 2024].
- Gautam, . N., 2021. *techlekh.com*. [Online]
Available at: <https://techlekh.com/food-tech-companies-nepal-growth/>
[Accessed 20 August 2024].
- Kukhnavets, P., 2019. *welldoneby.com*. [Online]
Available at: <https://welldoneby.com/blog/7-stages-of-software-development-life-cycle/>
[Accessed 22 August 2024].
- Organ, C., 2024. *www.forbes.com*. [Online]
Available at: [https://www.forbes.com/advisor/business/what-is-work-breakdown-structure/#:~:text=A%20work%20breakdown%20structure%20\(WBS,deliverables%20into%20a%20single%20tool](https://www.forbes.com/advisor/business/what-is-work-breakdown-structure/#:~:text=A%20work%20breakdown%20structure%20(WBS,deliverables%20into%20a%20single%20tool)
[Accessed 23 August 2024].
- Restolabs, 2023. *www.restolabs.com*. [Online]
Available at: <https://www.restolabs.com/blog/online-food-ordering-problems>
[Accessed 20 August 2024].
- Sachan, D., 2024. *scaler.com*. [Online]
Available at: <https://www.scaler.com/topics/incremental-model-in-software-engineering/>
[Accessed 22 August 2024].
- shopdev, 2023. *shopdev.co*. [Online]
Available at: <https://shopdev.co/blog/software-development-milestones>
[Accessed 24 August 2024].
- Shrestha, Y. & Shrestha, S. R., 2018. *www.linkedin.com*. [Online]
Available at: <https://www.linkedin.com/pulse/online-food-delivery-business-nepal-shrinkhal-raj-shrestha/>
[Accessed 20 August 2024].