Module Code & Module Title

CS6PO5NT-Final Year Project

Assessment Weightage & Type

Final Year Project Proposal (5%)

Year and Semester

2024-25 Autumn

Student Name: Subin Rai

London Met ID: 22072220

College ID: NP05CP4A220125

Assignment Submission Date: 4 Dec, 2024

Internal Supervisor: Mr. Sagar Basnet

External Supervisor: Mr. Chintan Karki

Title: OnSite App

I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline for my assignment 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

Table of Figures

Table of Tables

1 Introduction

1.1 Introduction to Topic

In today's fast-moving world, many applications make our lives easier by digitizing all types of day-to-day activities. Nevertheless, a lot of operations, particularly in industries like construction, depend extensively on manual processing, giving rise to lots of inefficiencies and errors. To address these problems and drive a fully digital transformation, my final year project focuses on developing an OnSite App.

This application is designed to assist builders and contractors in streamlining their operations at the construction site. The app helps builders track worker attendance, capture budget data, and monitor project progress. Traditionally, many construction site activities are handled manually, which can result in errors, inefficiencies, and delays. By providing a digital solution, this app aims to make the whole workflow easier by reducing paperwork and enhance overall productivity.

1.2 Problem Scenario

When builders and contractors receive a contract, they often rely on paper-based methods to keep records. This manual approach can lead to numerous errors, as they frequently depend on memories the document details after visiting the site. The delay between observing the work and recording it on paper increases the likelihood of forgotten details, resulting in inaccurate or incomplete records, which can create significant issues in tracking project progress and managing resources effectively.

Figure 1: Records of different contracts

Additionally, in construction, builders are typically provided with a budget upfront. As they purchase materials throughout the project, they often lose track of the remaining funds, requiring frequent manual calculations to determine the remaining budget, which can be time-consuming and lead to mistakes. When it comes time to pay workers their weekly salaries, builders must maintain daily attendance records on paper and manually calculate salaries, which can also result in errors. (Planyard, 2024)

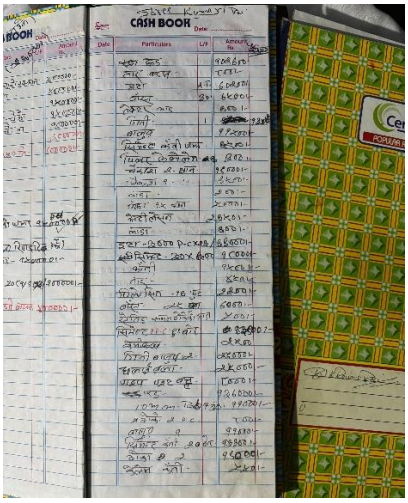

Figure 2: Budget record of the contract

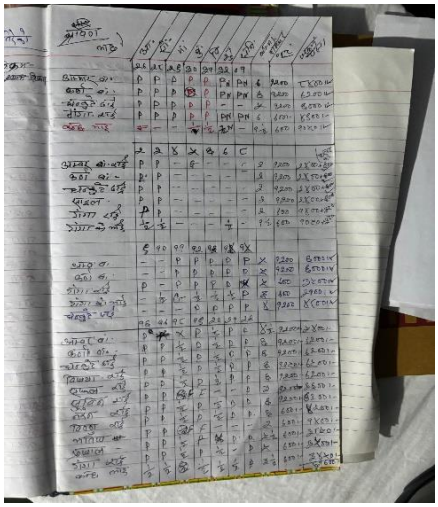

Figure 3: Attendance of the worker

Moreover, builders often have numerous contacts for material suppliers, and they may forget or lose these important details over time.

These issues are not unique to builders even my father as a builder faces similar challenges in his work. This inspired me to develop an app that will assist him in his daily tasks and alleviate the pressure associated with managing these responsibilities. While these may seem like simple problems, I believe many contractors and builders experience similar difficulties, and this app could provide valuable supports.

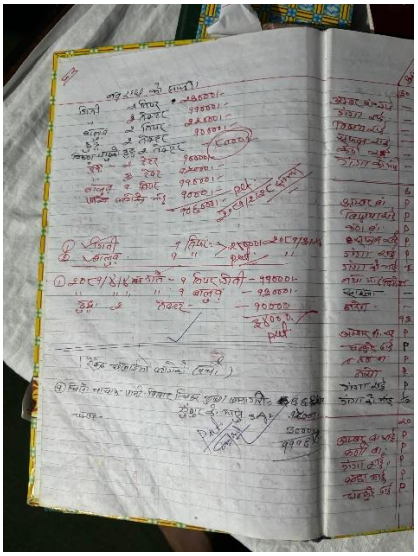

Figure 4: Calculating the material cost

1.3 Project as a solution

The OnSite application designed to offer a comprehensive digital solution to the problems builders and contractors face in managing construction site operations. By automating these processes, the application decreases inefficiencies and the error rates that are most prone to happening when tracking attendance, budget management, and project monitoring. The centralized platform simplifies operations, decreases dependence on memory and paper-based records, and promises to ensure better accuracy and productivity.

By integrating advanced technologies, such as AI and OCR into the app, it enhances decision-making and develops better collaboration between builders and clients. As these challenges are effectively tackled with the OnSite app, operational processes will be smoother and less prone to errors while providing an efficient experience for construction professionals, which again is beneficial not only for a builder like my father but also for many others.

2 Aims and Objectives

2.1 Aims

The aim of this project is to develop a user-friendly construction site app that streamlines worker attendance tracking, project progress monitoring, and budget management, while enhancing communication with clients to improve overall productivity and reduce errors in construction operations.

2.2 Objectives

1. To develop a user-friendly interface that allows builders and contractors to easily navigate and utilize the app's features.
2. To implement a real-time attendance tracking system that records worker presence accurately and efficiently.
3. To integrate a budget management tool that automatically updates remaining funds as expenses are logged, eliminating manual calculations.
4. To provide a secure database for storing supplier contact information, making it easily accessible for builders when needed.
5. To create the feature of visualizing 3D house models for better project understanding among clients.
6. To include a chatting feature that will help builders communicate seamlessly with clients to get problems resolved as fast as possible and collaborate on important notes.
7. To simplify salary payments through an online payment feature, ensuring secure and efficient transactions with automated record-keeping.

3 Expected Outcomes and Deliverables

The OnSite App will be embedded with a set of high-level and user-friendly features that are intended to solve the prevalent problems of builders and contractors. A detailed overview of the deliverables of the app is given below:

Attendance Tracking System

- A feature that allows builders to track worker attendance digitally.
- Attendance records are updated automatically in the system, and they can be looked upon at any time required for reference or salary purposes.

Budget Management Tool

- Gives builders a single interface where they can view, the allocated budget, actual expenses, and remaining balance.
- Updates the budget on its own once material costs or other expenses are added.
- No need to create tedious calculations manually.

Supplier Contact Storage with OCR Integration

- A secure and searchable database where supplier contact information can be stored.
- Utilizes OCR to extract and digitize contact details from business cards, or documents so that no information is lost over time.

3D Project Visualization

- Provides builders and clients with the opportunity to see 3D models of construction projects for better visualization and understanding.
- This includes rotations, zooms, and other features that help make sure things are clear regarding designs.

House Building Cost Prediction (AI)

- Estimates construction costs by using AI to factor in land area, number of floors, and total number of rooms.

- Accurately forecasts the budget for effective financial planning and decision-making.
- Minimizes unexpected expenses by factoring in historical data and current market trends.

House Selling Cost Prediction (AI)

- Predicts the potential selling price of a property by analyzing key elements like construction costs, room count, and additional features.
- Includes market data to provide builders and clients with a solid basis on which to negotiate prices.

Chatting and Realtime Collaboration

- A messenger within the app for real time communication between builders and clients.
- Includes file sharing, and push notifications to ensure that relevant parties are on the same page in terms of project progression and updates.

Salary Payment Integration

- Automates salaries disbursement calculation and dispersal based on attendance records.

Store documentation

- Ensures data about a site are safely kept in digital form-for example, invoices, contracts, and reports.
- No paper is needed hence eliminating loss or damage of core documents.

4 Projects risks, Threats and Contingency plans

SN	Risk Description	Probability	Impact	Contingency Plans
1	Technical Challenges with 3D Features.	High	High	- Allocate extra time for research. - Take help from domain expert.
2	Data Loss or Corruption	Low	High	- Regular backups of all databases. - Use of version control tools for backup.
3	Performance Issues on Mobile Devices	Medium	Medium	- Performance testing on various mobile devices. - Optimize code and resources for better performance.
4	Data Collection	Medium	High	- Manually collect data.
5	Unexpected bugs or software problems	High	High	- Maintain detailed documentation of the codebase.
6	Third-Party Dependency Failures	Medium	Medium	- Evaluate and choose reliable third-party APIs and tools. - alternative libraries or APIs.

Table 1: Projects risks, Threats and Contingency plans

5 Methodology

5.1 Considered Methodologies

5.1.1 Waterfall methodology

The Waterfall methodology is a linear, successive approach to project development wherein every phase, including requirements, design, implementation, testing, and deployment, needs to be finished before moving on to the next one.

- Advantages: Clearly structured, well-defined milestones, and predictable timelines.
- Disadvantages: rigidly inflexible to change, hence, it is unsuitable for projects whose requirement keeps changing.

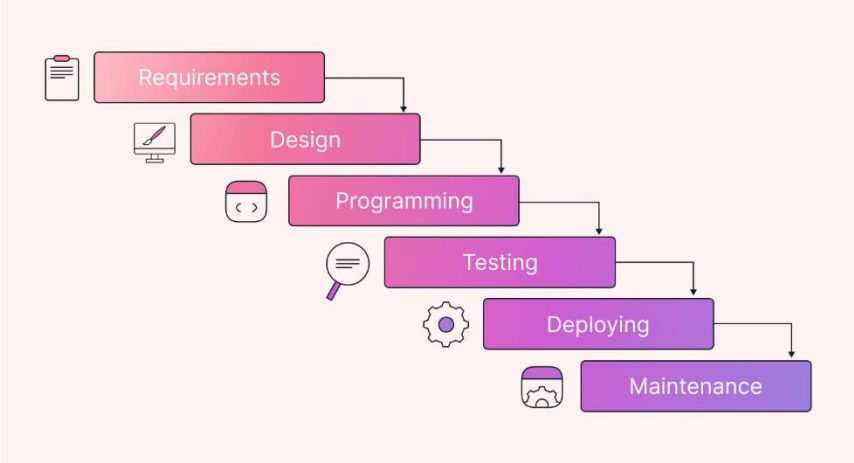

Figure 5: Waterfall methodology

5.1.2 RUP methodology

RUP is an iterative approach that divides the project lifecycle into phases such as inception, elaboration, construction, and transition. (Kroll & Kruchten, 2004)

- Advantages: Focuses on risk mitigation and enables iterative development
- Disadvantages: Complexity in implementation and resource-intensive.

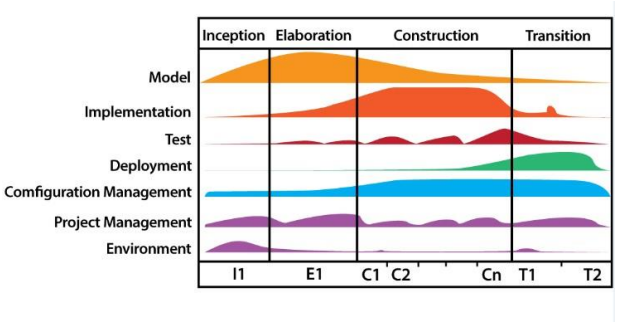

Figure 6: RUP methodology (testbytes, 2019)

5.2 Selected Methodology

In this project, the Scrum methodology has been chosen because it will go well with Agile. Scrum focuses on flexibility, adaptability, and collaboration, which goes hand in hand with the dynamic nature of software development. This project will be divided into 'sprints' lasting 2–4 weeks. These sprints are for continuous progress and regular feedback, hence the evolution in the project efficiently. Towards the end of each sprint, deliveries are made, and refining in light of feedback from the supervisor or stakeholders allows improvement in continuity of the development process. (Scrum Guides, 2020)

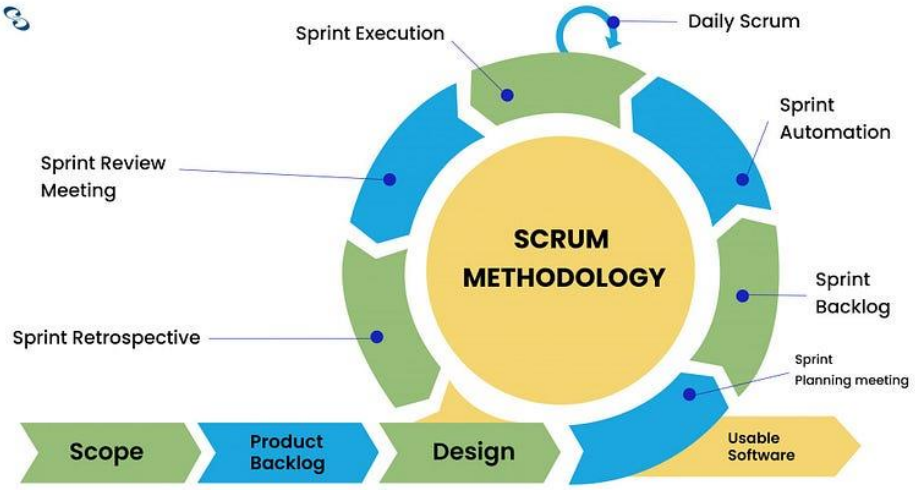

Figure 7: Scrum Methodology (korkut, 2023)

It can also be done as an individual project that focuses on sustaining effective communication with the supervisor and stakeholders through updates and feedback sessions that ensure consistency in maintaining the goals of the project, including improvement integration. The key components of Scrum in the Project are:

1. Product Backlog:

- It acts like a living, digital to-do list for the whole project. Of it all are features, enhancements, and tasks to be developed, like attendance tracking, budgeting management, AI-powered prediction capabilities, and 3D modeling capabilities.

- Backlog items will be prioritized relative to each other based on their value and relevance towards meeting the project objectives. This means the status of the items is updated, and refactoring is done regularly throughout the project course. (Scrum Guides, 2020)

2. Sprint Backlog:

- A sprint backlog can be thought of at the beginning of each sprint; one will select a subset from the product backlog. Those elements represent the work that the Development Team will do during that particular sprint.

- This is an assured way of ensuring that the development team focuses on only the highest priority features in a certain period, without diversion of focus, hence increasing productivity.

3. Sprint Review:

- A sprint review is done at the end of each sprint in order to present the work that has been completed to the supervisor or stakeholder. The meeting enables feedback and thus ensures conformance to the goals of the project.

- Feedback from the sprint review will help refine the product backlog and guide subsequent sprints.

4. Sprint Testing:

- In each sprint, extensive testing will be performed to ensure that developed features are functional and work reliably. For example, attendance tracking will be checked for accuracy.

- This iterative testing approach helps identify and resolve bugs or issues early in the development cycle, reducing potential risks.

5. Retrospective:

- After each sprint, a retrospective will be performed to reflect on what was done well, what can be improved, and any lessons learned. This practice assures continuous improvement in the development process and in the way the team collaborates. (Scrum Guides, 2020)

The main reasons for using Scrum include the high level of adaptability of this methodology and its possibility of supporting ongoing feedback through iterative sprints, making it ideal in dynamic software development projects like OnSite. Unlike the Waterfall model, where there is a rigid, step-by-step flow from start to finish, needing a restart when changes are encountered, Scrum lets there be a flexible consideration of changeability. Similarly, while the RUP involves iterative phases, it is very often as structured as Waterfall, hence not that dynamic when it comes to frequent updates. Scrum, with its emphasis on regular updates and adjustments, ensures a user-focused, flexible development process-just what projects in a constantly changing environment need.

6 Resource Requirements

6.1 Hardware Requirements

- **Laptop:** It will be used in the development of the application. This device will serve as the main one for coding activities, application testing, and operation of development tools.
- **Mobile:** Since, I am developing a mobile app, a mobile device is essential for testing and ensuring the app functions properly on real devices, allowing for accurate performance and usability assessments.

6.2 Software Requirements

1. **Node.js:** Node.js is an environment that runs JavaScript outside the browser and on the server. It's for developing the application backend, handling API requests, and maintaining server-side logic. Because it is asynchronous and event-driven, it goes very well with scalability.

2. **React Native:** React Native is a well-known open-source framework by Facebook for building cross-platform applications. This means the mobile developer can design the mobile application for iOS and Android from a single codebase. Development will be much quicker with a native-like user experience.

3. **MySQL:** MySQL is a relational database management system that is used to store data for applications. It is designed with a structure and efficiency in carrying out its functions, thus allowing the handling of complex inquiries and relationships.

4. **Prisma:** In the project of the OnSite app, Prisma will be used to simplify database operations, such as those with MySQL, and speed up development while reducing possible errors in handling complex database queries.

5. **Postman:** Among API development and testing tools, it happens to be one of the favorites because it simplifies so much in creating, testing, and debugging APIs. In

this OnSite app, this would be used during development for testing server-side endpoints in communication with the backend to the frontend.

6. **Spline**: In this project, 3D modeling will be created or, if already created, upload in Spline and integrated to the app. From Spline, a link to be generated will enable Builders and Property Owners to easily go through these 3D models via the app through the improvement of visualization.

7. **Git and GitHub:** Git will track code changes, while GitHub will securely host the project repository and enable collaboration and version management for the OnSite app.

7 Work Breakdown Structure

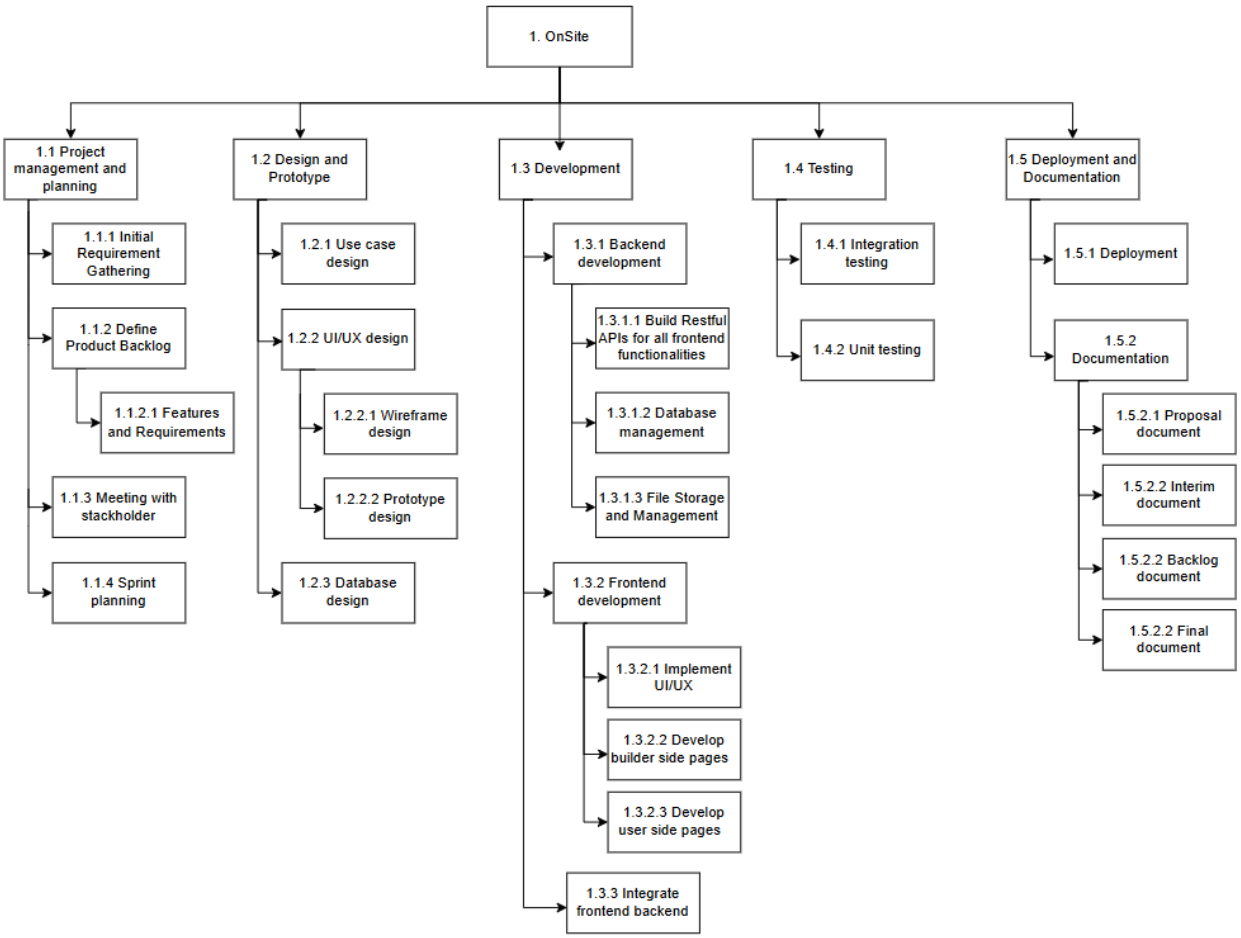

Figure 8: Work Breakdown Structure

8 Milestones Chart

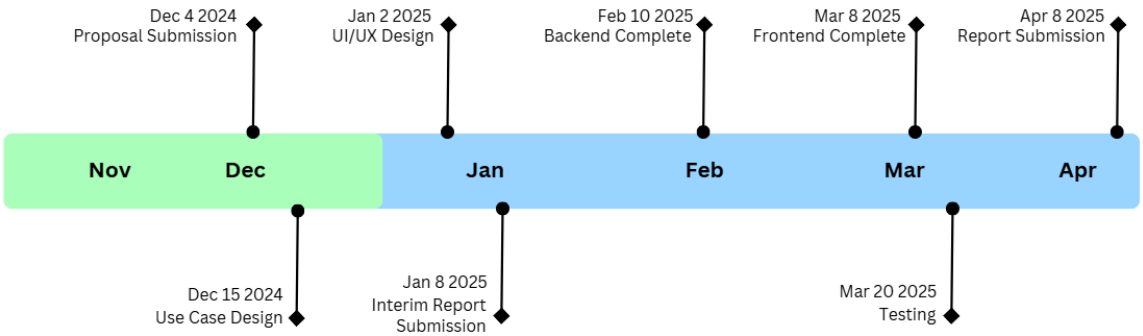

Figure 9: Milestones chart

9 Project Gantt Chart

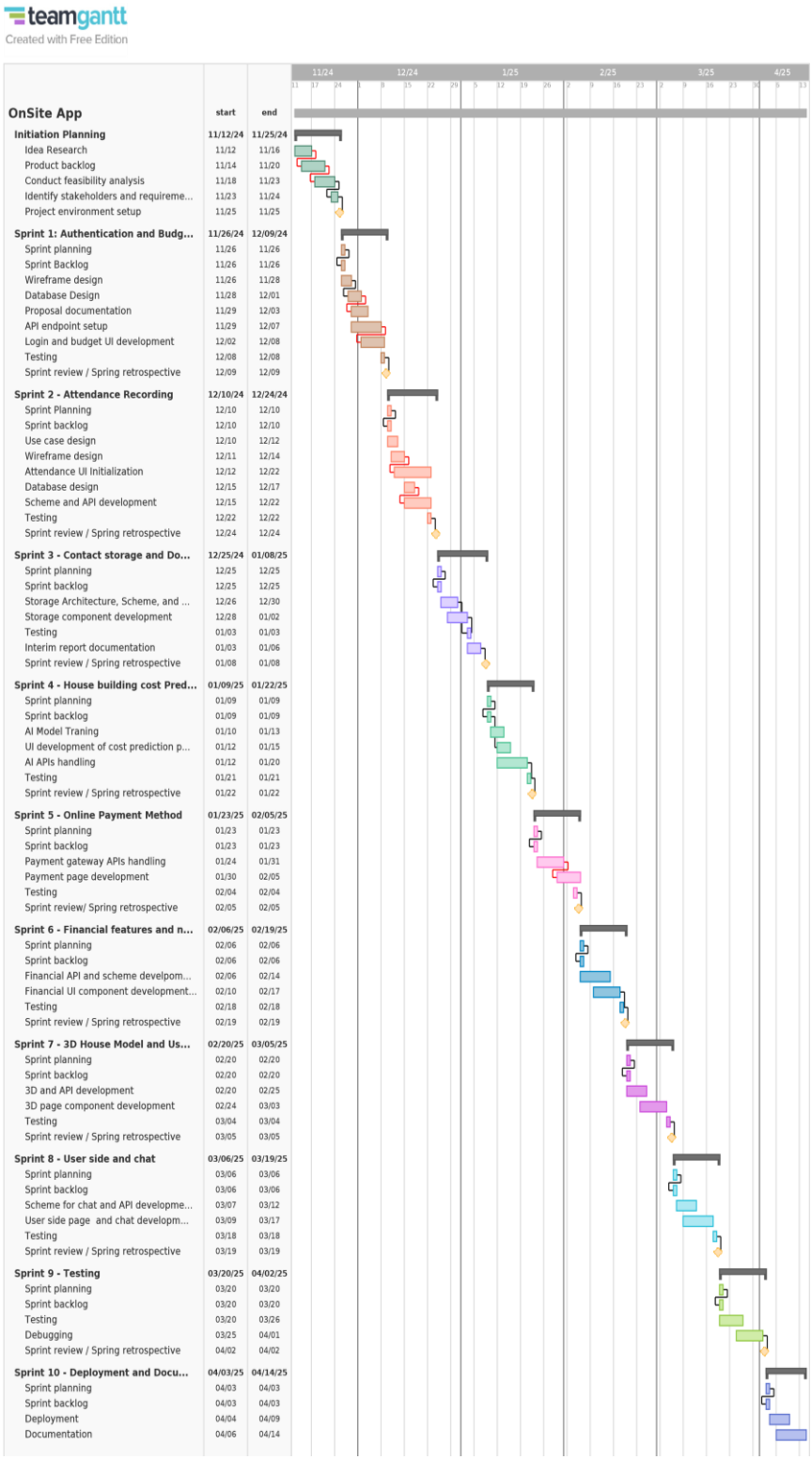

Figure 10: Gantt Chart

10 Conclusion

In Conclusion, the OnSite app is to solve the builder's and contractor's problems efficiently and in a very user-friendly manner. The app digitizes the operations, such as attendance, budgeting, and project progress, to reduce manual errors and enhance productivity. With the integration of advanced features like AI-driven predictions, AR/3D models, the app ensures a seamless workflow for users.

The development of this project involves rigorous planning, continuous testing, and regular feedback for the delivery of a quality product. Every phase of the project will be carried out systematically with utmost effort and dedication. This project addresses real-world problems that builders face and is also an invaluable learning experience that will contribute to my professional growth and skill as a developer.

11 References

korkut, T., 2023. *Excelling in Software Development with Scrum Methodology Part 2.* [Online]
Available at: https://blog.stackademic.com/excelling-in-software-development-with-scrum-methodology-part-2-e2d0b29437ce
[Accessed 30 11 2024].

Kroll, P. & Kruchten, P., 2004. *The rational unified process made easy.* 4th ed. s.l.:s.n.

Planyard, 2024. *Construction Budget Tracking: Long-Term Consequences of Poor Budget Tracking.* [Online]
Available at: https://planyard.com/blog/construction-budget-tracking-long-term-consequences-of-poor-budget-tracking
[Accessed 24 11 2024].

Scrum Guides, 2020. *The 2020 Scrum GuideTM.* [Online]
Available at: https://scrumguides.org/
[Accessed 30 11 2024].

testbytes, 2019. *What is Rational Unified Process Methodology?.* [Online]
Available at: https://www.testbytes.net/blog/rational-unified-process/
[Accessed 30 11 2024].