



islington college
(इस्लिंग्टन कॉलेज)

Module Code & Module Title

CS6P05NI Project

Proposal

5% Individual Coursework

Submission: Final Submission

AY 2025 2026

Academic Semester: Autumn Semester 2025

Credit: 30 Credit Year Long Module

Student Name: Bhishma Pratap Karki

London Met ID: 23049355

College ID: NP01CP4A230361

Assignment Due Date: Wednesday, December 3, 2025

Assignment Submission Date: Wednesday, December 3, 2025

Internal Supervisor: Shresha Rajbhandari

External Supervisor: Rabin Sapkota

I confirm that I understand my coursework needs to be submitted online via MST 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.

Acknowledgement

I would like to express my sincere gratitude to my internal supervisor, Ms.Shresha Rajbhandari, and my external supervisor, Mr.Rabin Sapkota, for their continuous guidance, valuable suggestions, and encouragement throughout the preparation of this proposal. Their support has been crucial in shaping my ideas, refining the objectives, and providing the right direction for my project.

I am also sincerely thankful to the FYP department for assigning this project and giving me the opportunity to work on a practical and meaningful problem. Preparing this proposal has allowed me to explore the challenges involved in building a Smart Job and Skill Portfolio web application and to plan its development using the MERN stack, along with AI-based job recommendations and application tracking features.

Finally, I would like to acknowledge the chance to work on this project proposal, which has helped me strengthen my technical understanding, project planning skills, and problem-solving abilities. This experience will serve as a valuable foundation for my future studies and career in the IT field.

Abstract

The recruitment process in today's competitive job market is often inefficient, time-consuming, and lacks transparency for both job seekers and recruiters. The proposed Smart Job and Skill Portfolio web application aims to address these issues by providing a platform where job seekers can create detailed portfolios, showcase their skills, certifications, and projects, and track their applications in real time.

The system also includes AI-based job recommendations to match candidates with suitable opportunities based on their skills, portfolios, and quiz results. Recruiters benefit from efficient application management and an integrated ATS resume scanner to evaluate, score, and rank candidates. Admins can manage quizzes, assessments, and overall platform activity to ensure smooth and secure operation.

This project will be developed using the MERN stack (MongoDB, Express.js, React.js, Node.js), following a combination of SDLC and Agile Scrum methodology to ensure structured yet flexible development. The final deliverables will be fully functional, user-friendly web application providing seamless interaction between job seekers, recruiters, and administrators, bridging the gap between candidate's skills and relevant job opportunities.

Table of Contents

| | | |
|------|--|----|
| 1. | Introduction | 1 |
| 1.1. | Problem Scenario | 1 |
| 1.2. | The Project as a Solution..... | 1 |
| 2. | Aims and objectives | 3 |
| 2.1. | Aim:..... | 3 |
| 2.2. | 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 |
| 5. | Methodology..... | 8 |
| 5.1. | Software Development Life Cycle (SDLC)..... | 8 |
| 5.2. | Considered Methodologies | 8 |
| 5.3. | Selected Methodology | 9 |
| 6. | Resource Requirements..... | 10 |
| 6.1. | Hardware Requirements..... | 10 |
| 6.2. | Software Requirements | 10 |
| 6.3. | Datasets and Publications | 10 |
| 6.4. | IT / Access Requirements:..... | 10 |
| 7. | Work Breakdown Structure (WBS)..... | 12 |
| 8. | Milestones | 13 |
| 9. | Gantt Chart..... | 15 |
| 9.1. | Initiation | 15 |
| 9.2. | Planning..... | 15 |

| | |
|--|----|
| 9.3. Execution | 16 |
| 9.3.1. Sprint 1: Core Setup and Authentication | 16 |
| 9.3.2. Sprint 2: Job Seeker Portfolio and Profile | 16 |
| 9.3.3. Sprint 3: Job Discovery and Application Tracking | 17 |
| 9.3.4. Sprint 4: Recruiter Features | 17 |
| 9.3.5. Sprint 5: Admin Dashboard | 18 |
| 9.3.6. Sprint 6: Notifications, Messaging and Networking | 18 |
| 9.3.7. Sprint 7: Testing and Final Documentation | 19 |
| 9.4. Control | 19 |
| 9.5 Closure | 20 |
| 9.6. Full View of Gantt Chart..... | 21 |
| 9.7. List View of Gantt Chart..... | 23 |
| 10. Conclusion | 24 |
| 11. References | 25 |
| 12. Appendix | 28 |
| 12.1. Detailed Explanation of Considered Methodologies | 28 |
| 12.1.1. Prototype Model | 28 |
| 12.1.2. Spiral Model | 29 |
| 12.1.3. Agile Kanban Model | 30 |
| 12.2. Work Breakdown Structure (WBS) | 31 |
| 12.3. Milestones..... | 31 |
| 12.4. Gantt Chart | 32 |

Table of Figures

| | |
|--|----|
| Figure 1: SDLC (Software Development Life Cycle) Image (Sami, 2012)..... | 8 |
| Figure 2: Agile Scrum Image (Hubstaff Tasks, 2025)..... | 9 |
| Figure 3: Work Breakdown Structure (WBS)..... | 12 |
| Figure 4: Project Milestone Map..... | 14 |
| Figure 5: Gantt Chart Initiation | 15 |
| Figure 6: Gantt Chart Planning..... | 15 |
| Figure 7: Gantt Chat Execution (Sprint 1: Core Setup and Authentication)..... | 16 |
| Figure 8: Gantt Chart Execution (Sprint 2: Job Seeker Portfolio and Profile) | 16 |
| Figure 9: Gantt Chart Execution (Sprint 3: Job Discovery and Application Tracking).... | 17 |
| Figure 10: Gantt Chart Execution (Sprint 4: Recruiter Features)..... | 17 |
| Figure 11: Gantt Chart Execution (Sprint 4: Recruiter Features)..... | 18 |
| Figure 12: Gantt Chart Execution (Sprint 6: Notifications, Messaging and Networking) | 18 |
| Figure 13: Gantt Chart Execution (Sprint 7: Testing and Final Documentation)..... | 19 |
| Figure 14: Gantt Chart Control | 19 |
| Figure 15: Gantt Chart Closure | 20 |
| Figure 16: Full View Gantt Chart 2 | 21 |
| Figure 17: Full View Gantt Chart 1 | 21 |
| Figure 18: Full View Gantt Chart 3 | 22 |
| Figure 19: List View of Gantt Chart | 23 |
| Figure 20: Prototype Model Image (Board Infinity, 2025)..... | 28 |
| Figure 21: Spiral Model Image (Boehm, 1988)..... | 29 |
| Figure 22: Agile Kanban Model Image (Radigan, 2025)..... | 30 |

Table of Tables

| | |
|--|----|
| Table 1: Project Risk, Threats and Contingency Plans | 5 |
| Table 2: Milestones | 13 |

1. Introduction

The job market is highly competitive, with over 220 million unemployed worldwide. Many struggle to find jobs that match their skills, and about 70% never receive updates on their applications (International Labour Organization, 2023) (Linkedin, 2022). Recruiters often have to review hundreds of applications manually, which is time-consuming and can lead to mistakes.

Traditional application methods, like sending resumes and cover letters by email, are not transparent and make it hard for recruiters to see a candidate's true abilities (Chavan, et al., 2024). Online portfolios and skill platforms let candidates showcase projects, certifications, and work samples. An application tracker keeps candidates informed at every step, making hiring more transparent and reliable (Peicheva, 2023). Admins manage quizzes, platform activity, and ensure the system runs smoothly and securely.

1.1. Problem Scenario

The recruitment process today is slow and often confusing for both job seekers and recruiters. Using keyword based resume screening may miss skilled candidates (Hatti, et al., 2025). Many companies still use outdated hiring methods that are time-consuming, can lead to mistakes, and sometimes unfair (Tuttle & Critchlow, 2025). Candidates often do not know if their applications were seen or shortlisted (International Labour Organization, 2023) (Kumari, 2025). Portfolios and certificates are often scattered to different platforms, making it hard for recruiters to judge true potential (Tayal, et al., 2023).

1.2. The Project as a Solution

The proposed Smart Job and Skill Portfolio platform helps candidates, recruiters, and admins. Candidates can create profiles to show their projects, certificates, and skills. An application tracker keeps them updated, and a job recommendation suggests jobs based on their skills and quiz results.

Recruiters can manage applications more easily using a resume scanner that ranks candidates based on skills and experience. They can also assign assessments for specific roles to find the right candidate. Admins can manage quizzes, user activities and ensure the platform runs smoothly for all users.

2. Aims and objectives

2.1. Aim:

To develop a Smart Job and Skill Portfolio web application that enables job seekers to showcase their skills, find suitable jobs efficiently, and allow recruiters and admins to manage applications, assessments, and platform activities effectively.

2.2. Objectives:

- i. To allow job seekers to create and update portfolios with project, certificates, and skills (Sprint 2).
- ii. To provide a real-time application tracking for candidates showing status updates (Sprint 3).
- iii. To implement an AI-based job recommendation system suggesting suitable jobs based on skills, portfolio, and quiz results (Sprint 3).
- iv. To enable recruiters to post jobs, search candidates, and manage applications efficiently (Sprint 4).
- v. To develop a local ATS (Application Tracking System) resume scanner to extract, score, and rank candidates based on the scores (Sprint 4).
- vi. To allow admins to manage quizzes, assessments, platform content, and monitor user activity (Sprint 5).
- vii. To provide dashboards with key performance and activity analytics (Sprint 5).
- viii. To enable networking and messaging between candidates and recruiters (Sprint 6).
- ix. To ensure a secure, user-friendly, and accessible interface, and complete final testing and documentation (Sprint 7).

3. Expected Outcomes and Deliverables

3.1. Expected Outcomes

- i. A functional web application supporting Job Seekers, Recruiters, and Admin roles.
- ii. Job Seekers will be able to create portfolios, showcase their skills and certificates, track their applications in real-time, and receive AI-based job recommendations.
- iii. Recruiters will be able to post jobs, search candidates, manage applications, and use an ATS resume scanner to assign scores and rank candidates.
- iv. Admins will be able to create and manage quizzes and assessments, control platform content, monitor user activity, and maintain platform security.
- v. All users will have access to dashboards and analytics to track activity, performance, and insights.
- vi. Candidates and recruiters will be able to communicate and provide feedback through the networking and messaging system.

3.2. Deliverables

- i. The fully integrated Smart Job and Skill Portfolio web application with complete features for all three user roles.
- ii. A comprehensive documentation covering system architecture, module descriptions, ATS resume scanning approach, AI recommendation logic, system design diagrams, and database structure.
- iii. User guidance included in the documentation for Job Seekers, Recruiters, and Admins.
- iv. An admin panel with tools to manage quizzes, assessments, and platform activities,
- v. Fully tested and optimized web application, ready for deployment, with secure and user-friendly interfaces.

4. Project Risks, Threats and Contingency Plans

Table 1: Project Risk, Threats and Contingency Plans

| SN | Risk Description | Impact (1-5) | Likelihood (1-5) | Risk Level (Impact × Likelihood) | Threats | Mitigation Strategy | Contingency Plan |
|----|---|-----------------|---------------------|-------------------------------------|----------------------------|--|---|
| 1 | Complexity in AI job recommendations and ATS resume scanning | 4 | 4 | 16 | Development delays, errors | Develop modules separately, use libraries/APIs | Extra testing time, research, advice from supervisors |
| 2 | Data handling for portfolios, certificates, quizzes, applications | 4 | 3 | 12 | Data loss or inconsistency | Clear data structure, best practices | Regular testing, data backups |
| 3 | Performance issues in AI recommendations and ATS resume scanning. | 4 | 3 | 12 | Slow or inaccurate results | Optimize algorithms, validate with dataset | Iterative refinement, research, guidance from supervisors |
| 4 | Integration of job seeker, recruiter, admin modules | 3 | 4 | 12 | Functional issues | Structured design, document workflows | Step-by-step validation, extra integration testing |
| 5 | User Interface/usability problems | 3 | 3 | 9 | Confusing navigation | Regular usability checks, feedback | UI fixes based on feedback |

| | | | | | | | |
|----|---|---|---|---|---|---|--|
| 6 | Third-party library or API failures | 3 | 3 | 9 | Certain features may stop working | Test alternatives, keep updated dependencies | Replace or update libraries, fallback solutions |
| 7 | Internet connectivity issues during testing or pushing code in GitHub | 3 | 3 | 9 | Delay in testing and repository updates | Maintain backup connections | Use alternate internet sources, reschedule testing |
| 8 | Incomplete or unclear requirements | 3 | 3 | 9 | Misaligned features or rework | Regular review with supervisor, clarify requirements | Adjust design, re-plan sprints based on feedback |
| 9 | Hardware or system crashes during development | 3 | 2 | 6 | Loss of progress, corrupted data | Regular backup, maintain stable environment | Restore from backups, spare hardware ready |
| 10 | Time constraints due to academic workload | 3 | 2 | 6 | Delayed milestones, incomplete deliverables | Strictly follow Gantt chart, break tasks according to WBS, prioritize weekly deliverables | Adjust schedule if needed, research time saving methods, seek advice from supervisors. |

Risk Level Indicators:Extreme Risk High Risk Medium Risk 

Low Risk

5. Methodology

5.1. Software Development Life Cycle (SDLC)

This project follows the Software Development Life Cycle (SDLC), providing a structured roadmap through requirement analysis, design, development, testing, and deployment (Sommerville, 2011) (Pressman, 2010). SDLC ensures organized, systematic development, minimizing errors and rework. It is particularly suitable for the Smart Job and Skill Portfolio web application with multiple modules, including authentication, portfolio management, AI recommendations, recruiter tools, ATS resume scanning, and admin features, allowing each component to be planned and developed step by step.

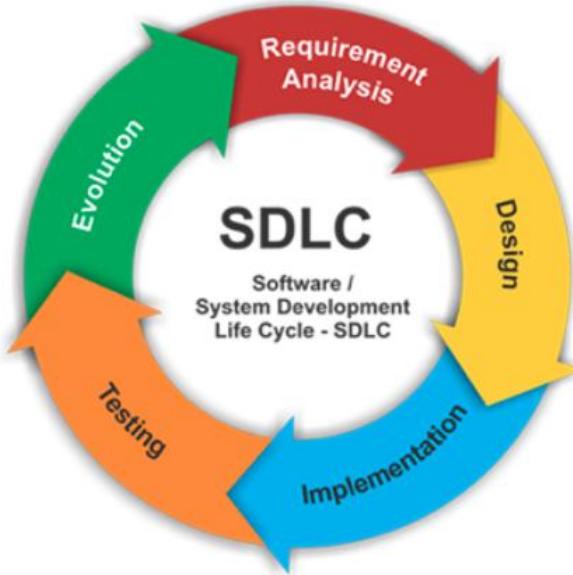


Figure 1: SDLC (Software Development Life Cycle) Image (Sami, 2012)

5.2. Considered Methodologies

Several development approaches were considered, including the Prototype Model, Spiral Model, and Agile Kanban Model, based on factors like project complexity, flexibility, risk management, and iterative feedback.

A detailed explanation of considered methodologies is provided in Appendix.

5.3. Selected Methodology

After reviewing the considered models, the project will follow a combination of SDLC and Agile methodology using the Scrum framework. SDLC ensures structured planning, while Agile Scrum adds flexibility for iterative development of a feature-rich system. Scrum defines roles, ceremonies, and artifacts, allowing incremental feature delivery, early feedback, and continuous improvement. This combination balances planning, risk management, and adaptability, ensuring the platform meets functional requirements for all users (Sommerville, 2011) (Pressman, 2010).

Project Scrum Phases:

- i. **Initiation and Planning:** Define goals, stakeholders, project scope, and plan sprints.
- ii. **Sprint Execution:** Develop and test specific features in each sprint.
- iii. **Control and Improvement:** Review progress after each sprint and apply improvements.
- iv. **Closure:** Final testing, documentation, and presentation of the completed system.

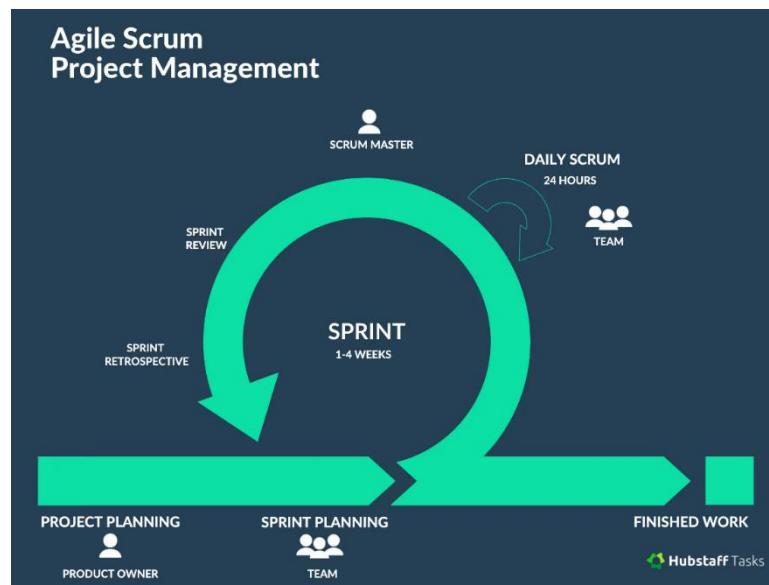


Figure 2: Agile Scrum Image (Hubstaff Tasks, 2025)

6. Resource Requirements

6.1. Hardware Requirements

Personal computer or laptop with 8GB RAM, i5 processor or higher, 256GB SSD for development and testing.

6.2. Software Requirements

i. Development Tools:

- **Code Editor / IDE:** Visual Studio Code
- **Version Control:** Git/GitHub
- **Package Management:** npm for installing and managing dependencies

ii. Frontend Technologies: HTML, CSS, JavaScript, React.js

iii. Backend Technologies: Node.js, Express.js

iv. Database: MongoDB

v. Testing: Manual testing and postman for testing backend APIs

vi. Design Tools: Figma

vii. Project Management Tools: Jira

6.3. Datasets and Publications

- i. **Datasets:** Sample resumes, job listing, and portfolios for testing AI recommendations and ATS resume scanner, sourced from public datasets like Kaggle, GitHub and Hugging Face.
- ii. **Research and Publications:** Relevant journals and articles on AI-based job recommendations, resume parsing, MERN stack best practices, and portfolio management will guide development and design.

6.4. IT / Access Requirements:

- i. High speed internet for research, open-source libraries, and software updates.
- ii. Web browser (Chrome, Firefox) for testing and debugging the application.
- iii. Document editor (MS Word) for preparing project documentation and reports.

- iv. Draw.io for creating the Work Breakdown Structure (WBS) and ClickUp for generating the Gantt chart.

7. Work Breakdown Structure (WBS)

[For an explanation of WBS, refer to Appendix.](#)

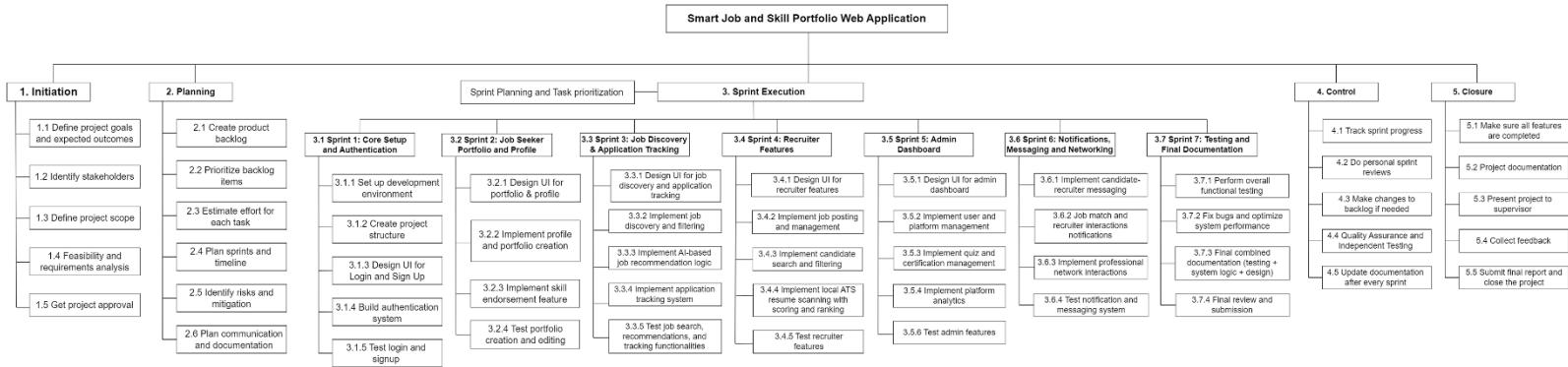


Figure 3: Work Breakdown Structure (WBS)

[For clear and larger view of WBS, Click here.](#)

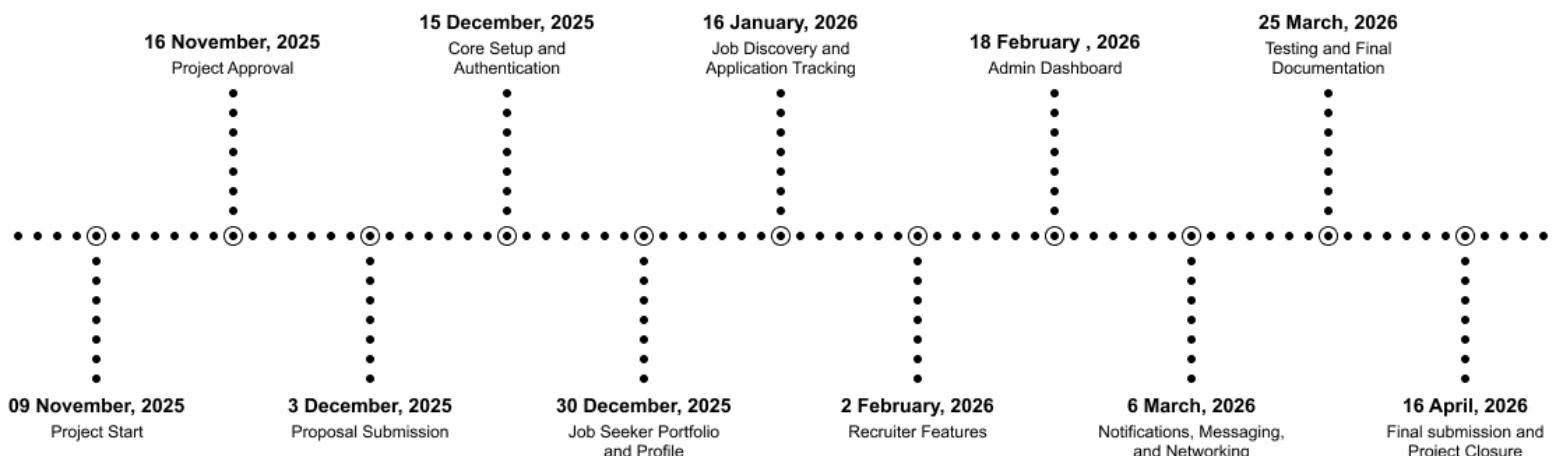
8. Milestones

[For an explanation of milestones, refer to Appendix.](#)

Project Milestones

Table 2: Milestones

| Milestone | Description | Date |
|--|--|-------------------|
| Project Start | Research and requirement gathering | 09 November, 2025 |
| Project Approval | Concept review and approval | 16 November, 2025 |
| Proposal Submission | Submit project proposal | 3 December, 2025 |
| Core Setup and Authentication | Setup backend, frontend, authentication | 15 December, 2025 |
| Job Seeker Portfolio and Profile | Implement profiles and portfolios | 30 December, 2025 |
| Job Discovery and Application Tracking | Job discovery, application tracking and AI-based job recommendations | 16 January, 2026 |
| Recruiter Features | Job posting and ATS resume scanning | 2 February, 2026 |
| Admin Dashboard | Quizzes and platform analytics | 18 February, 2026 |
| Notifications, Messaging, and Networking | Messaging and networking features | 6 March, 2026 |
| Testing and Final Documentation | Final testing and documentation | 25 March, 2026 |
| Final submission and Project Closure | Project closure and submission | 16 April, 2026 |

*Figure 4: Project Milestone Map*

9. Gantt Chart

[For a detailed explanation of Gantt Chart, refer to Appendix](#)

9.1. Initiation

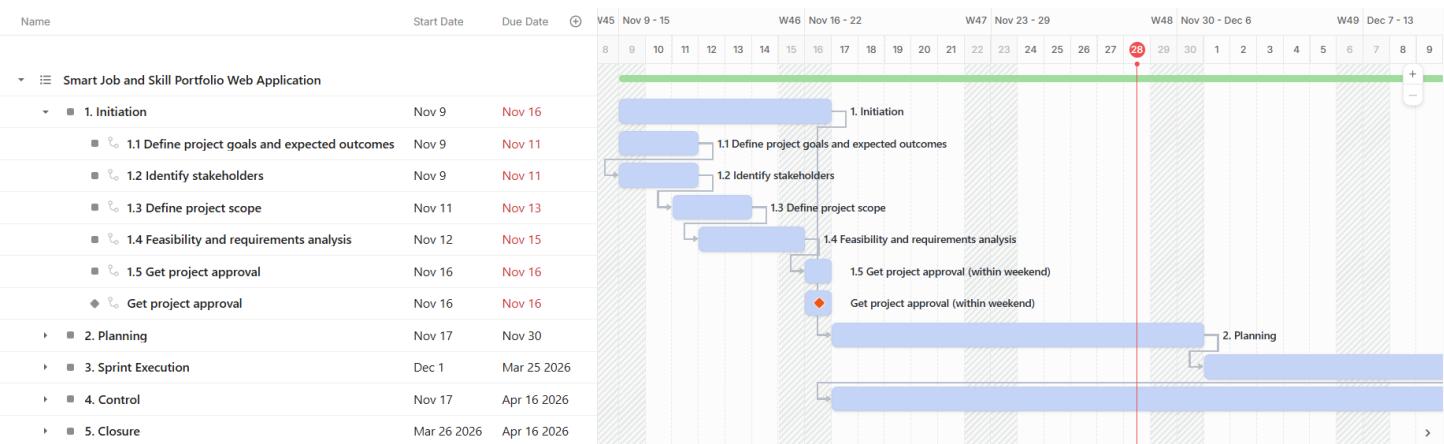


Figure 5: Gantt Chart Initiation

9.2. Planning

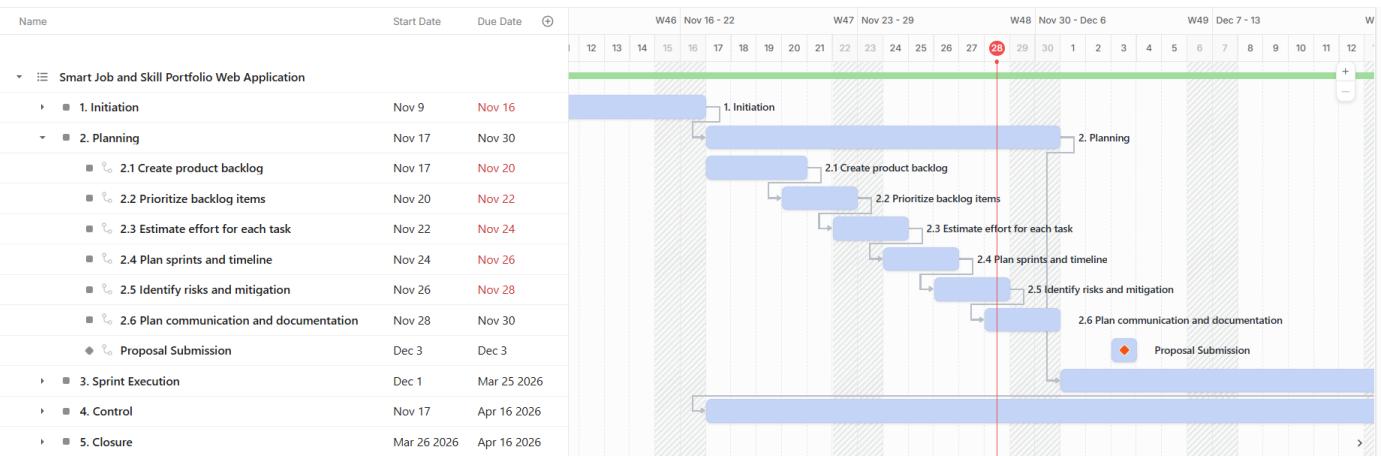


Figure 6: Gantt Chart Planning

9.3. Execution

9.3.1. Sprint 1: Core Setup and Authentication

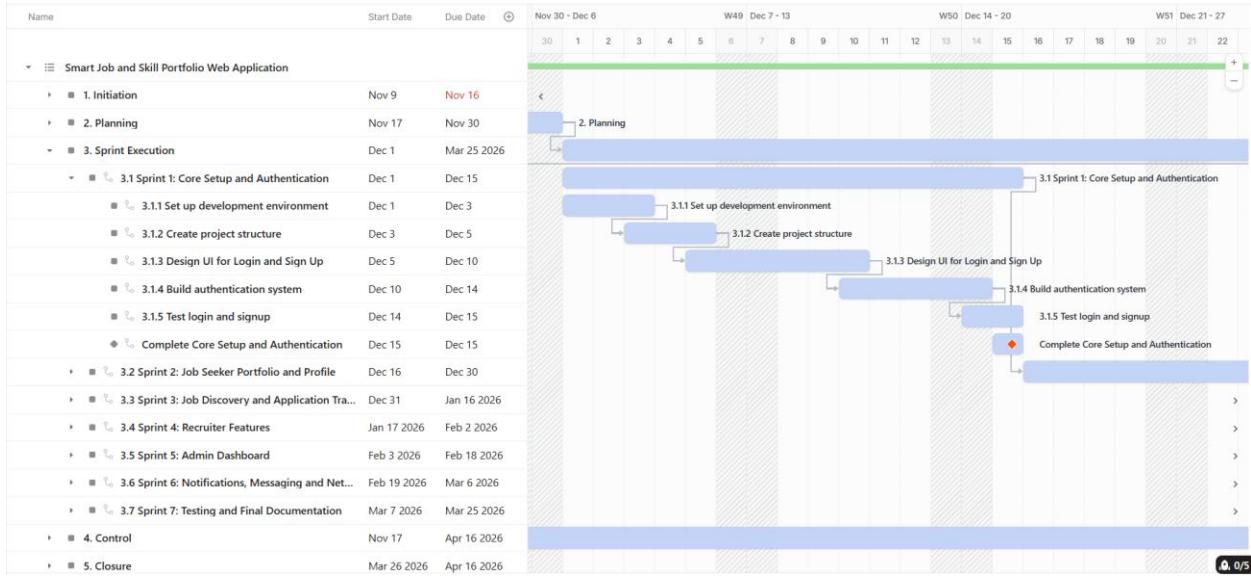


Figure 7: Gantt Chat Execution (Sprint 1: Core Setup and Authentication)

9.3.2. Sprint 2: Job Seeker Portfolio and Profile

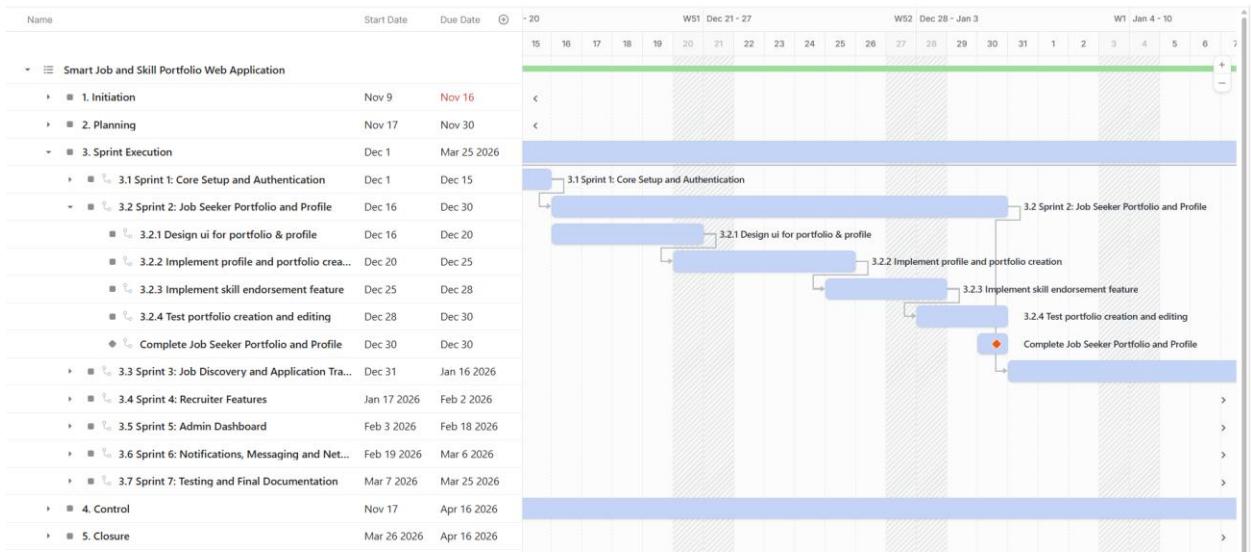


Figure 8: Gantt Chat Execution (Sprint 2: Job Seeker Portfolio and Profile)

9.3.3. Sprint 3: Job Discovery and Application Tracking

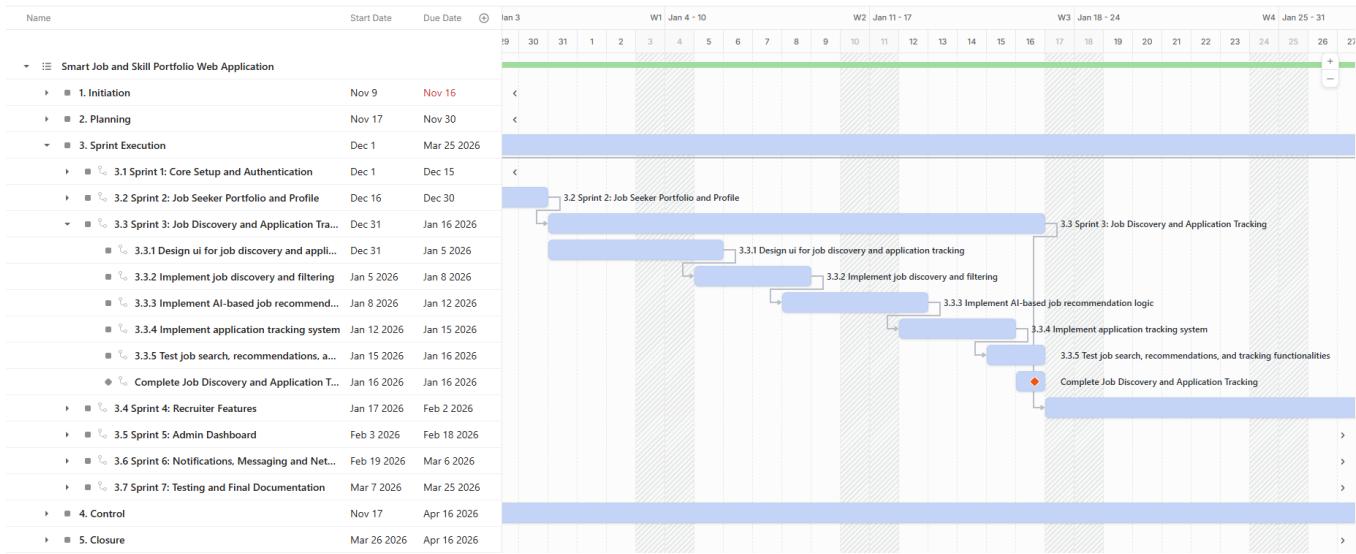


Figure 9: Gantt Chart Execution (Sprint 3: Job Discovery and Application Tracking)

9.3.4. Sprint 4: Recruiter Features

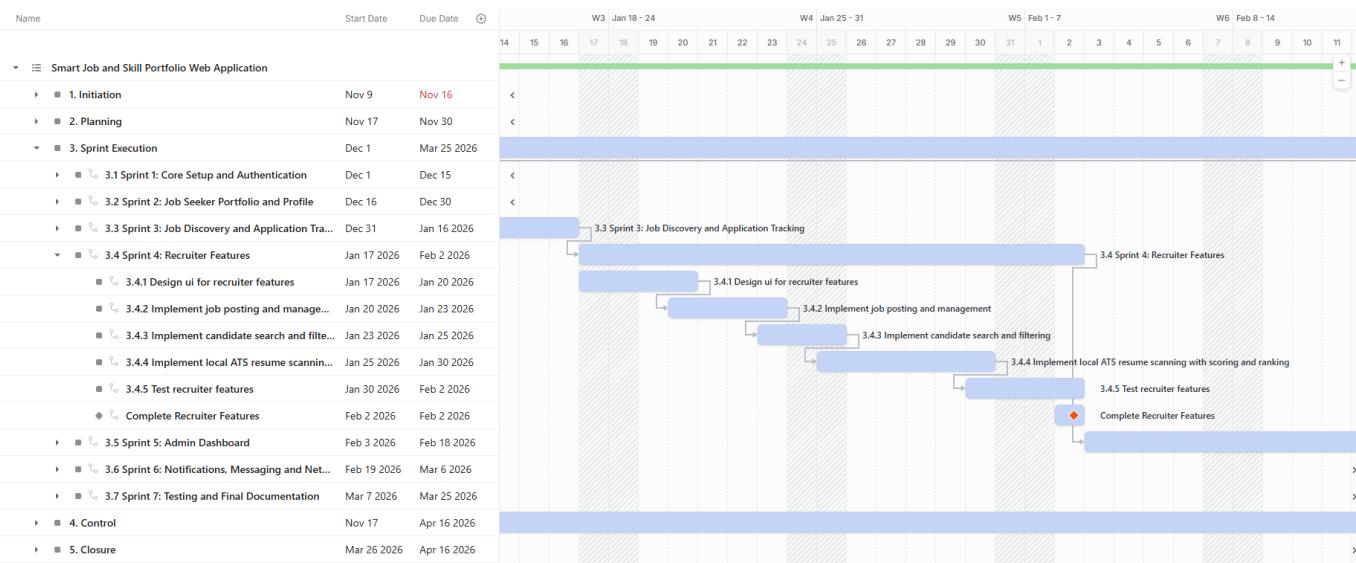


Figure 10: Gantt Chart Execution (Sprint 4: Recruiter Features)

9.3.5. Sprint 5: Admin Dashboard

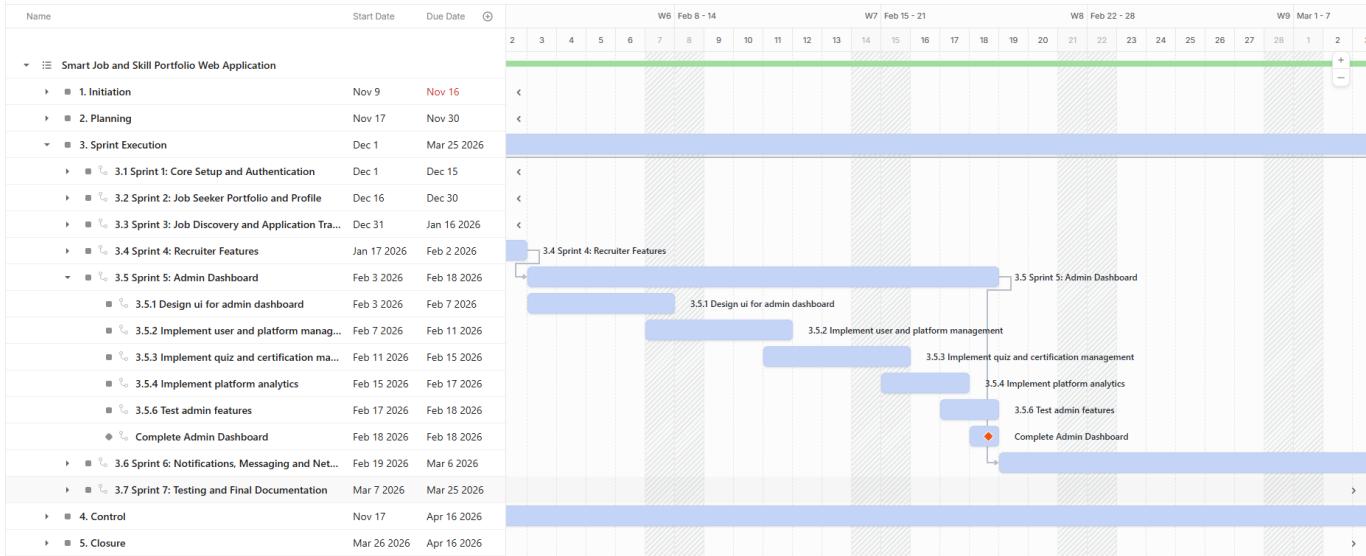


Figure 11: Gantt Chart Execution (Sprint 4: Recruiter Features)

9.3.6. Sprint 6: Notifications, Messaging and Networking

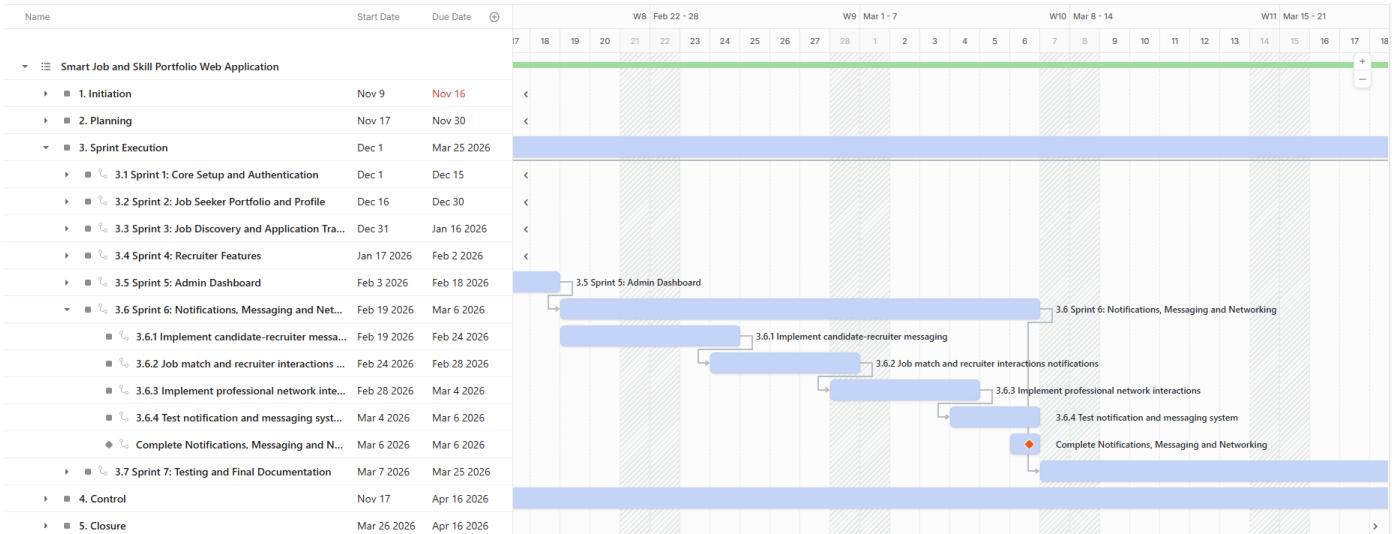


Figure 12: Gantt Chart Execution (Sprint 6: Notifications, Messaging and Networking)

9.3.7. Sprint 7: Testing and Final Documentation

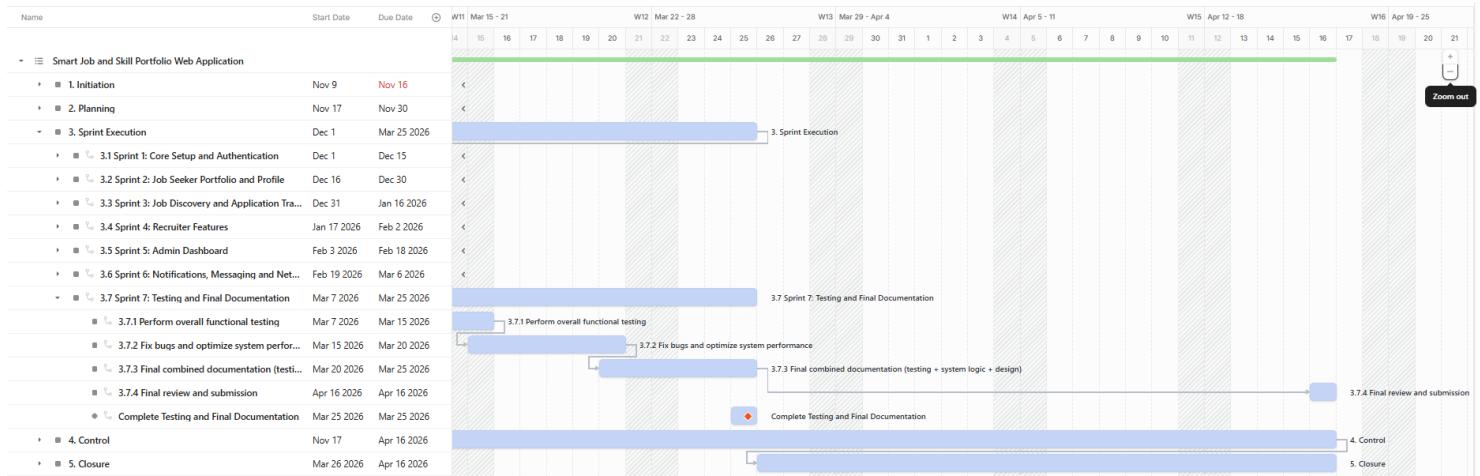


Figure 13: Gantt Chart Execution (Sprint 7: Testing and Final Documentation)

9.4. Control

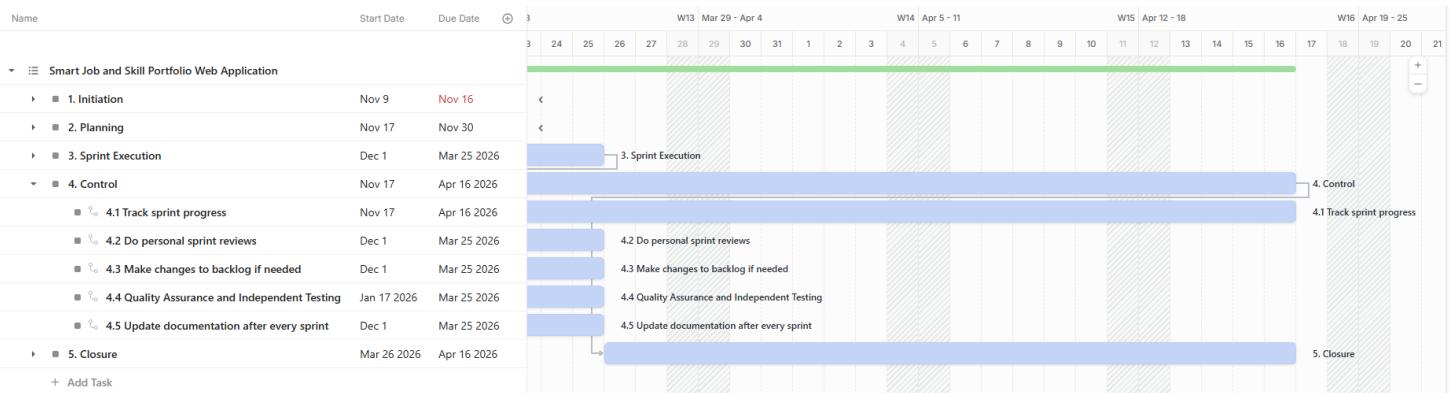


Figure 14: Gantt Chart Control

9.5 Closure

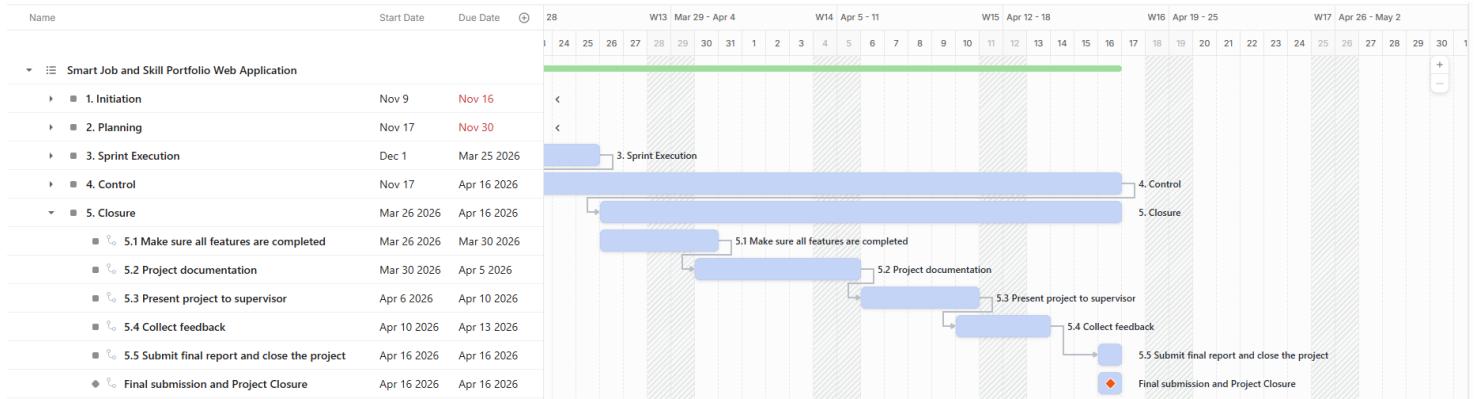


Figure 15: Gantt Chart Closure

9.6. Full View of Gantt Chart

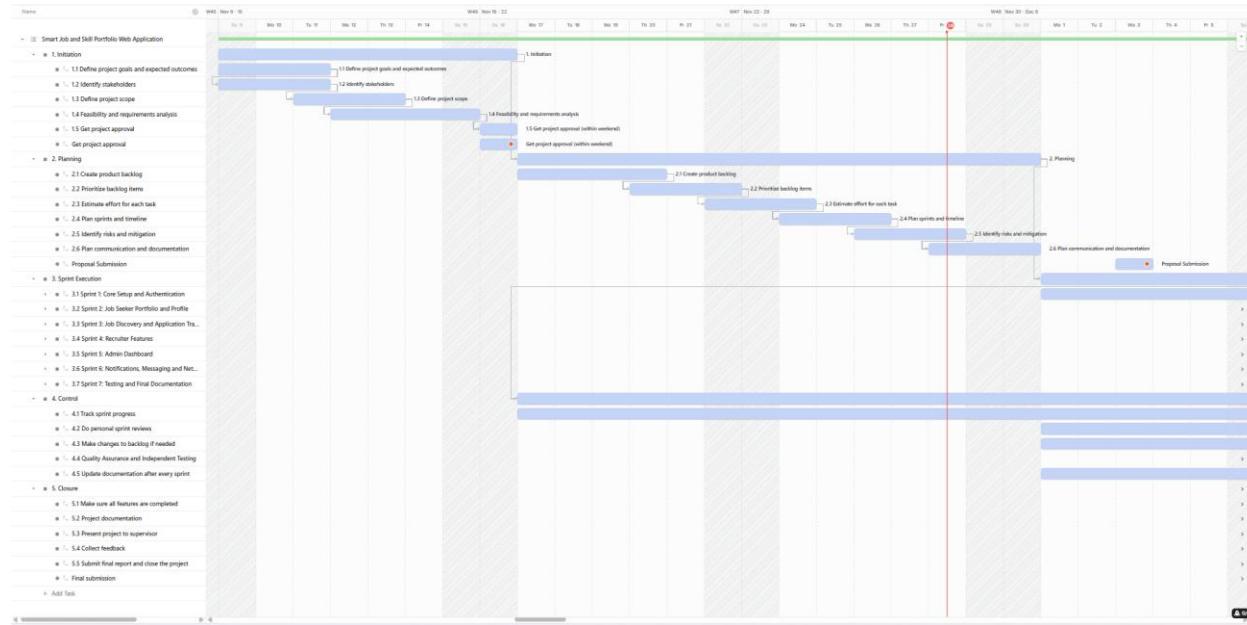


Figure 17: Full View Gantt Chart 1

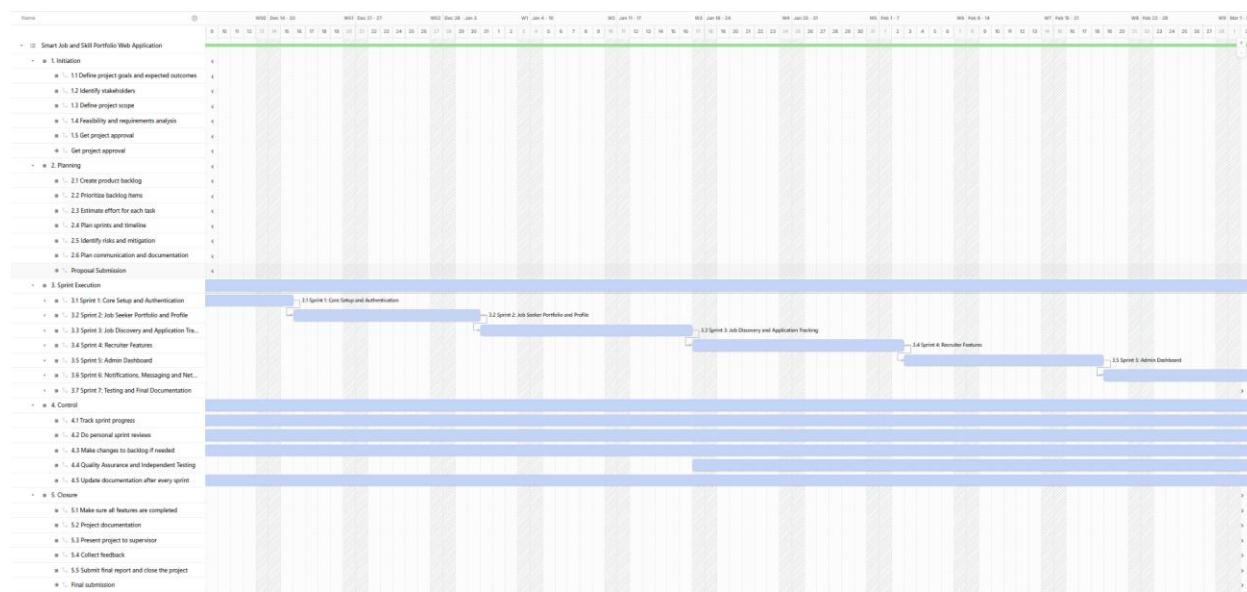


Figure 16: Full View Gantt Chart 2

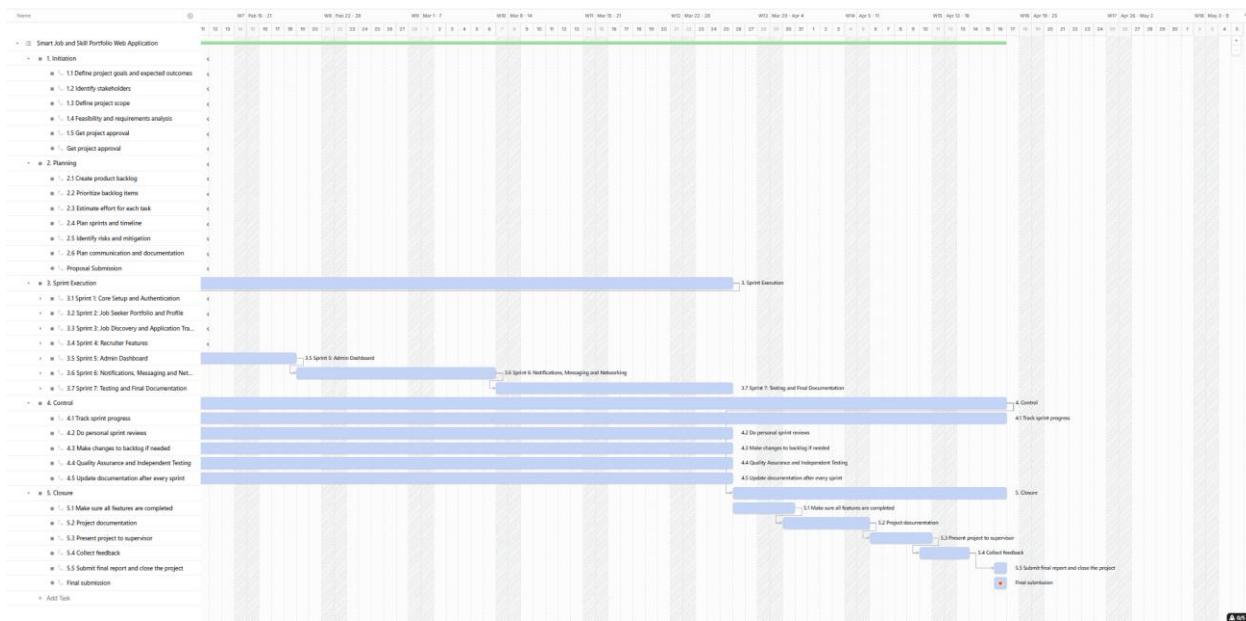


Figure 18: Full View Gantt Chart 3

[For clear and large view of Gantt Chart, Click here](#)

9.7. List View of Gantt Chart

| Name | Start Date | Due Date |
|--|-------------|-------------|
| Smart Job and Skill Portfolio Web Application | | |
| 1. Initiation | | |
| 1.1 Define project goals and expected outcomes | Nov 9 | Nov 11 |
| 1.2 Identify stakeholders | Nov 9 | Nov 11 |
| 1.3 Define project scope | Nov 11 | Nov 13 |
| 1.4 Feasibility and requirements analysis | Nov 12 | Nov 15 |
| 1.5 Get project approval | Nov 16 | Nov 16 |
| Get project approval | Nov 16 | Nov 16 |
| 2. Planning | | |
| 2.1 Create product backlog | Nov 17 | Nov 20 |
| 2.2 Prioritize backlog items | Nov 20 | Nov 22 |
| 2.3 Estimate effort for each task | Nov 22 | Nov 24 |
| 2.4 Plan sprints and timeline | Nov 24 | Nov 26 |
| 2.5 Identify risks and mitigation | Nov 26 | Nov 28 |
| 2.6 Plan communication and documentation | Nov 28 | Nov 30 |
| Proposal Submission | Dec 3 | Dec 3 |
| 3. Sprint Execution | | |
| 3.1 Sprint 1: Core Setup and Authentication | Dec 1 | Dec 15 |
| 3.1.1 Set up development environment | Dec 1 | Dec 3 |
| 3.1.2 Create project structure | Dec 3 | Dec 5 |
| 3.1.3 Design UI for Login and Sign Up | Dec 5 | Dec 10 |
| 3.1.4 Build authentication system | Dec 10 | Dec 14 |
| 3.1.5 Test login and signup | Dec 14 | Dec 15 |
| 3.1.6 Complete Core Setup and Authentication | Dec 15 | Dec 15 |
| 3.2 Sprint 2: Job Seeker Portfolio and Profile | Dec 16 | Dec 30 |
| 3.2.1 Design UI for portfolio & profile | Dec 16 | Dec 20 |
| 3.2.2 Implement profile and portfolio creation | Dec 20 | Dec 25 |
| 3.2.3 Implement skill endorsement feature | Dec 25 | Dec 28 |
| 3.2.4 Test portfolio creation and editing | Dec 28 | Dec 30 |
| 3.2.5 Complete Job Seeker Portfolio and Profile | Dec 30 | Dec 30 |
| 3.3 Sprint 3: Job Discovery and Application Tracking | Dec 31 | Jan 16 2026 |
| 3.3.1 Design UI for job discovery and application tracking | Dec 31 | Jan 5 2026 |
| 3.3.2 Implement job discovery and filtering | Jan 5 2026 | Jan 8 2026 |
| 3.3.3 Implement AI-based job recommendation logic | Jan 8 2026 | Jan 12 2026 |
| 3.3.4 Implement application tracking system | Jan 12 2026 | Jan 15 2026 |
| 3.3.5 Test job search, recommendations, and tracking functionalities | Jan 15 2026 | Jan 16 2026 |
| 3.3.6 Complete Job Discovery and Application Tracking | Jan 16 2026 | Jan 16 2026 |
| 3.4 Sprint 4: Recruiter Features | Jan 17 2026 | Feb 2 2026 |
| 3.4.1 Design UI for recruiter features | Jan 17 2026 | Jan 20 2026 |
| 3.4.2 Implement job posting and management | Jan 20 2026 | Jan 23 2026 |
| 3.4.3 Implement candidate search and filtering | Jan 23 2026 | Jan 25 2026 |
| 3.4.4 Implement local ATS resume scanning with scoring and ranking | Jan 25 2026 | Jan 30 2026 |
| 3.4.5 Test recruiter features | Jan 30 2026 | Feb 2 2026 |
| 3.4.6 Complete Recruiter Features | Feb 2 2026 | Feb 2 2026 |
| 3.5 Sprint 5: Admin Dashboard | Feb 3 2026 | Feb 18 2026 |
| 3.5.1 Design UI for admin dashboard | Feb 3 2026 | Feb 7 2026 |
| 3.5.2 Implement user and platform management | Feb 7 2026 | Feb 11 2026 |
| 3.5.3 Implement quiz and certification management | Feb 11 2026 | Feb 15 2026 |
| 3.5.4 Implement platform analytics | Feb 15 2026 | Feb 17 2026 |
| 3.5.5 Test admin features | Feb 17 2026 | Feb 18 2026 |
| 3.5.6 Complete Admin Dashboard | Feb 18 2026 | Feb 18 2026 |
| 3.6 Sprint 6: Notifications, Messaging and Networking | Feb 19 2026 | Mar 6 2026 |
| 3.6.1 Implement candidate-recruiter messaging | Feb 19 2026 | Feb 24 2026 |
| 3.6.2 Job match and recruiter interactions notifications | Feb 24 2026 | Feb 28 2026 |
| 3.6.3 Implement professional network interactions | Feb 28 2026 | Mar 4 2026 |
| 3.6.4 Test notification and messaging system | Mar 4 2026 | Mar 6 2026 |
| 3.6.5 Complete Notifications, Messaging and Networking | Mar 6 2026 | Mar 6 2026 |
| 3.7 Sprint 7: Testing and Final Documentation | Mar 7 2026 | Mar 25 2026 |
| 3.7.1 Perform overall functional testing | Mar 7 2026 | Mar 15 2026 |
| 3.7.2 Fix bugs and optimize system performance | Mar 15 2026 | Mar 20 2026 |
| 3.7.3 Final combined documentation (testing + system logic + design) | Mar 20 2026 | Mar 25 2026 |
| 3.7.4 Final review and submission | Apr 16 2026 | Apr 16 2026 |
| 3.7.5 Complete Testing and Final Documentation | Apr 16 2026 | Mar 25 2026 |
| 4. Control | | |
| 4.1 Track sprint progress | Nov 17 | Apr 16 2026 |
| 4.2 Do personal sprint reviews | Dec 1 | Mar 25 2026 |
| 4.3 Make changes to backlog if needed | Dec 1 | Mar 25 2026 |
| 4.4 Quality Assurance and Independent Testing | Jan 17 2026 | Mar 25 2026 |
| 4.5 Update documentation after every sprint | Dec 1 | Mar 25 2026 |
| 5. Closure | | |
| 5.1 Make sure all features are completed | Mar 26 2026 | Apr 16 2026 |
| 5.2 Project documentation | Mar 30 2026 | Apr 5 2026 |
| 5.3 Present project to supervisor | Apr 6 2026 | Apr 10 2026 |
| 5.4 Collect feedback | Apr 10 2026 | Apr 13 2026 |
| 5.5 Submit final report and close the project | Apr 16 2026 | Apr 16 2026 |
| Final submission | Apr 16 2026 | Apr 16 2026 |

Figure 19: List View of Gantt Chart

10. Conclusion

The proposed Smart Job and Skill Portfolio web application addresses the current inefficiencies and lack of transparency in the recruitment process. By providing job seekers with detailed portfolios, verified skill displays, AI-based job recommendations and real-time application tracking, the platform ensures that candidates can show their abilities effectively. At the same time, recruiter benefit from efficient application management, ATS resume scanning, skill assessments, and analytics to make informed hiring decisions.

The project follows a structured methodology combining SDLC and Agile Scrum, ensuring organized development with iterative improvements. With clear resource planning, defined milestones, and a systematic work breakdown, the project is designed to deliver a fully functional, user-friendly, and reliable platform. Overall, the proposal demonstrates a well-defined roadmap for creating a tool that bridges the gap between candidate's skills and appropriate job opportunities.

11. References

- Board Infinity, 2025. *A Quick Guide to Prototype Model in Software Engineering*. [Online]
Available at: <https://www.boardinfinity.com/blog/a-quick-guide-to-prototype-model-in-software-engineering/>
[Accessed 30 November 2025].
- Boehm, B. W., 1988. *A Spiral Model of Software Development and Enhancement*. New York, NY: ACM SIGSOFT Software Engineering Notes.
- Chavan, P. R., Chandurkar, Y., Tidake, A. H. & Lavankar, G., 2024. *Enhancing recruitment efficiency: An advanced Applicant Tracking System (ATS)*. [Online]
Available at:
[https://www.researchgate.net/publication/382099903 Enhancing recruitment efficiency
An advanced Applicant Tracking System ATS](https://www.researchgate.net/publication/382099903_Enhancing_recruitment_efficiency_An_advanced_Applicant_Tracking_System_ATS)
[Accessed 22 November 2025].
- Hatti, S. K., Rahaman, S. K. U., V, T. T. & Touqhir, S. M., 2025. *Efficient Resume Screening Tool*. [Online]
Available at:
[https://www.researchgate.net/publication/387930797 Efficient Resume Screening Tool](https://www.researchgate.net/publication/387930797_Efficient_Resume_Screening_Tool)
[Accessed 22 November 2025].
- Highsmith, J., 2009. *Agile Project Management: Creating Innovative Products*. 2nd ed. Boston, MA: Addison-Wesley.
- Hubstaff Tasks, 2025. *A guide to Agile Scrum project management*. [Online]
Available at: <https://hubstaff.com/tasks/scrum-project-management>
[Accessed 30 November 2025].
- International Labour Organization, 2023. *World Employment and Social Outlook Trends 2023*. [Online]
Available at: <https://www.ilo.org/publications/flagship-reports/world-employment-and-social-outlook-trends-2023>
[Accessed 22 November 2025].

Kumari, R., 2025. *5 Strategic Ways to Improve Candidate Experience in Recruitment.* [Online]
Available at: <https://www.reccopilot.com/blogs/5-strategic-ways-to-improve-candidate-experience-in-recruitment>
[Accessed 22 November 2025].

Linkedin, 2022. *2022 Global Talent Trends.* [Online]
Available at: <https://business.linkedin.com/talent-solutions/global-talent-trends/archival/global-talent-trends-january-2022>
[Accessed 22 November 2025].

Peicheva, M., 2023. *DATA ANALYSIS FROM THE APPLICANT TRACKING SYSTEM.* [Online]
Available at:
<https://www.researchgate.net/publication/375517346 DATA ANALYSIS FROM THE APPLICANT TRACKING SYSTEM>
[Accessed 22 November 2025].

Pressman, R. S., 2010. *Software Engineering: A Practitioner's Approach.* 7th ed. New York: McGraw-Hill.

Project Management Institute, 2017. *A Guide to the Project Management Body of Knowledge (PMBOK Guide).* 6th Edition ed. Newtown Square, PA: Project Management Institute.

Radigan, D., 2025. *Kanban.* [Online]
Available at: <https://www.atlassian.com/agile/kanban>
[Accessed 30 November 2025].

Sami, M., 2012. *Software Development Life Cycle Models and Methodologies.* [Online]
Available at: <https://melsatar.blog/2012/03/15/software-development-life-cycle-models-and-methodologies/>
[Accessed 30 November 2025].

Sommerville, I., 2011. *SOFTWARE ENGINEERING.* 9th ed. Boston: Addison-Wesley.

Tayal, R., Dutta, P. & Goyal, S., 2023. *ADOPTION OF TECHNOLOGY ON RECRUITMENT PROCESS A STUDY OF PERCEPTIONS AND ATTITUDES OF HR PROFESSIONALS.* [Online]
Available at: <https://ijcrt.org/papers/IJCRT2307426.pdf>
[Accessed 22 November 2025].

Tuttle, L. & Critchlow, K., 2025. *Digital transformation in talent acquisition: Modern approaches to recruitment and selection.* [Online]
Available at:
[https://www.researchgate.net/publication/391851238 Digital transformation in talent acquisition Modern approaches to recruitment and selection](https://www.researchgate.net/publication/391851238_Digital_transformation_in_talent_acquisition_Modern_approaches_to_recruitment_and_selection)
[Accessed 22 November 2025].

12. Appendix

12.1. Detailed Explanation of Considered Methodologies

12.1.1. Prototype Model

The Prototype Model is a software development approach where a simplified version of the system, called a prototype, is built first to understand requirements and gather feedback from users (Pressman, 2010). Instead of starting development with a fully detailed plan, developers create a basic version of the system that demonstrates key features. This helps identify missing or unclear requirements early and reduces misunderstandings between developers and users.

Prototyping is particularly useful in projects where requirements are not fully known or are likely to change during development. Users can interact with the prototype, provide suggestions for improvements. However, the Prototype Model may not provide enough structure for a complex, feature-rich system like the Smart Job and Skill Portfolio application (Sommerville, 2011). Therefore, it was not selected as the main methodology.

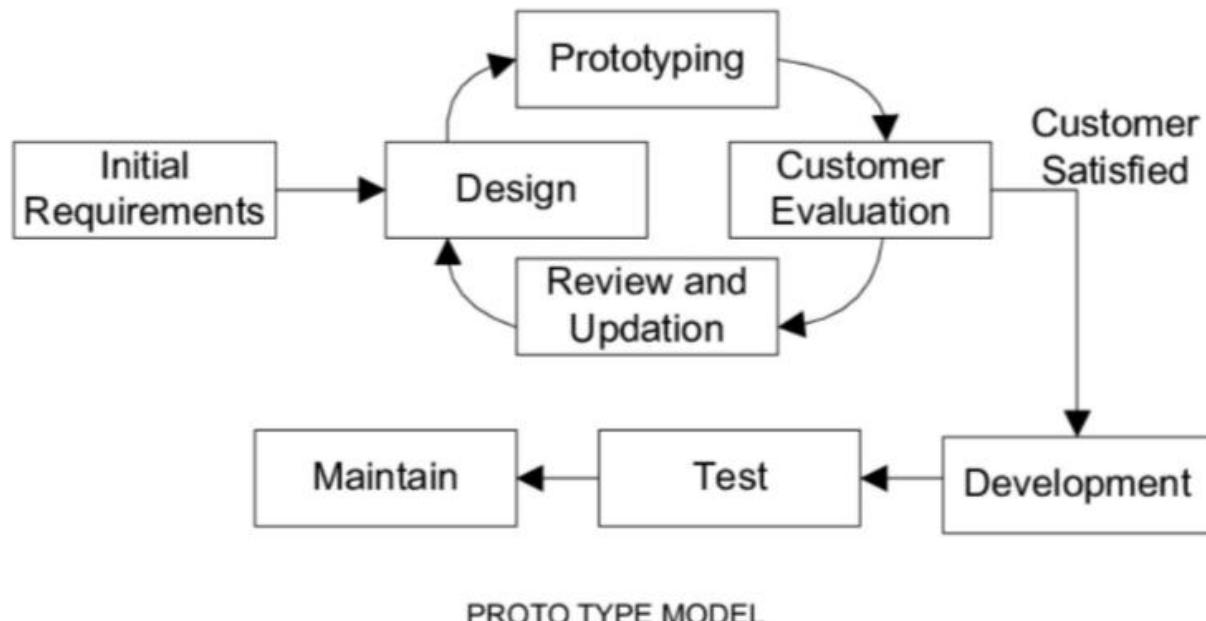


Figure 20: Prototype Model Image (Board Infinity, 2025)

12.1.2. Spiral Model

The Spiral Model is a risk-driven approach that combines iterative development with systematic planning. Development is carried out in a series of cycles or spirals, with each spiral including planning, risk analysis, engineering, and evaluation (Boehm, 1988). This allows for early identification and mitigation of risks while gradually developing a complete system.

The Spiral Model is suitable for large, complex projects with changing requirements or significant risks. Each iteration delivers a working version of the system, which can be evaluated and refined (Sommerville, 2011). However, the model requires careful management and may be more complex than necessary for this web application project. Therefore, it was not selected as the main methodology.

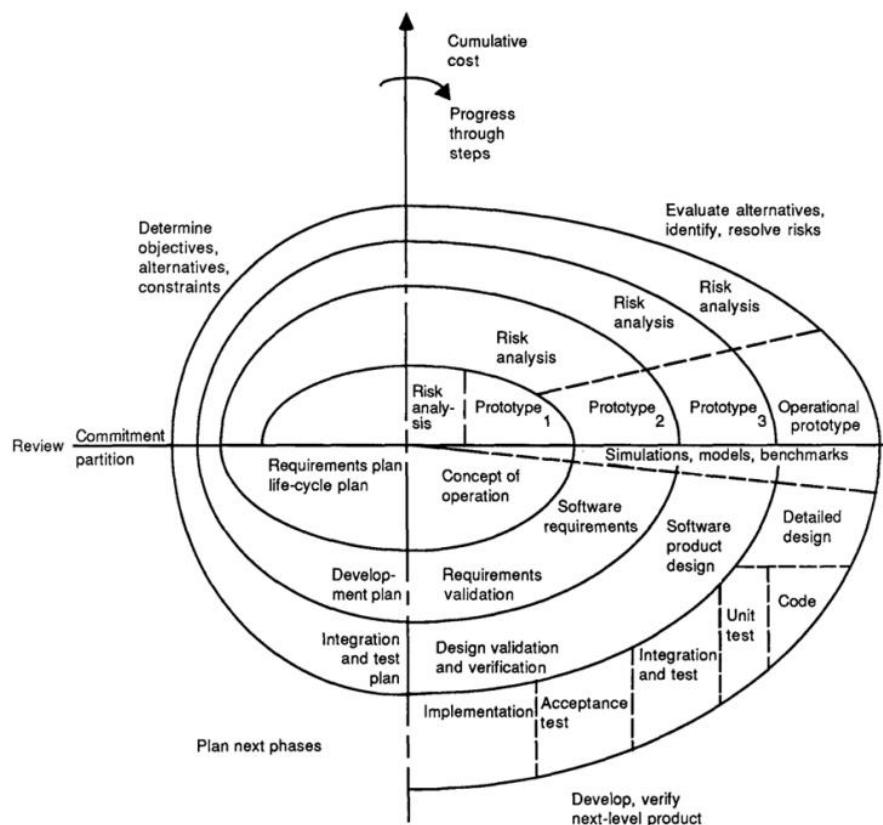


Figure 21: Spiral Model Image (Boehm, 1988)

12.1.3. Agile Kanban Model

Agile Kanban is an iterative, visual approach to managing software development using a task board to track work progress. Task move across columns such as “To Do”, “In Progress”, and “Done”, allowing the team to monitor workflow (Radigan, 2025). This method focuses on continuous delivery, flexibility, and efficiency.

Kanban is especially useful for projects with changing priorities or multiple parallel tasks. It allows the team to focus on high priority works, adjust plans dynamically, and deliver features incrementally. While Kanban is flexible, it lacks the structured planning and clearly defined phases provided by SDLC. For this project, Kanban was considered but not chosen because the combination of SDLC and Agile Scrum provides both structure and iterative flexibility (Highsmith, 2009).

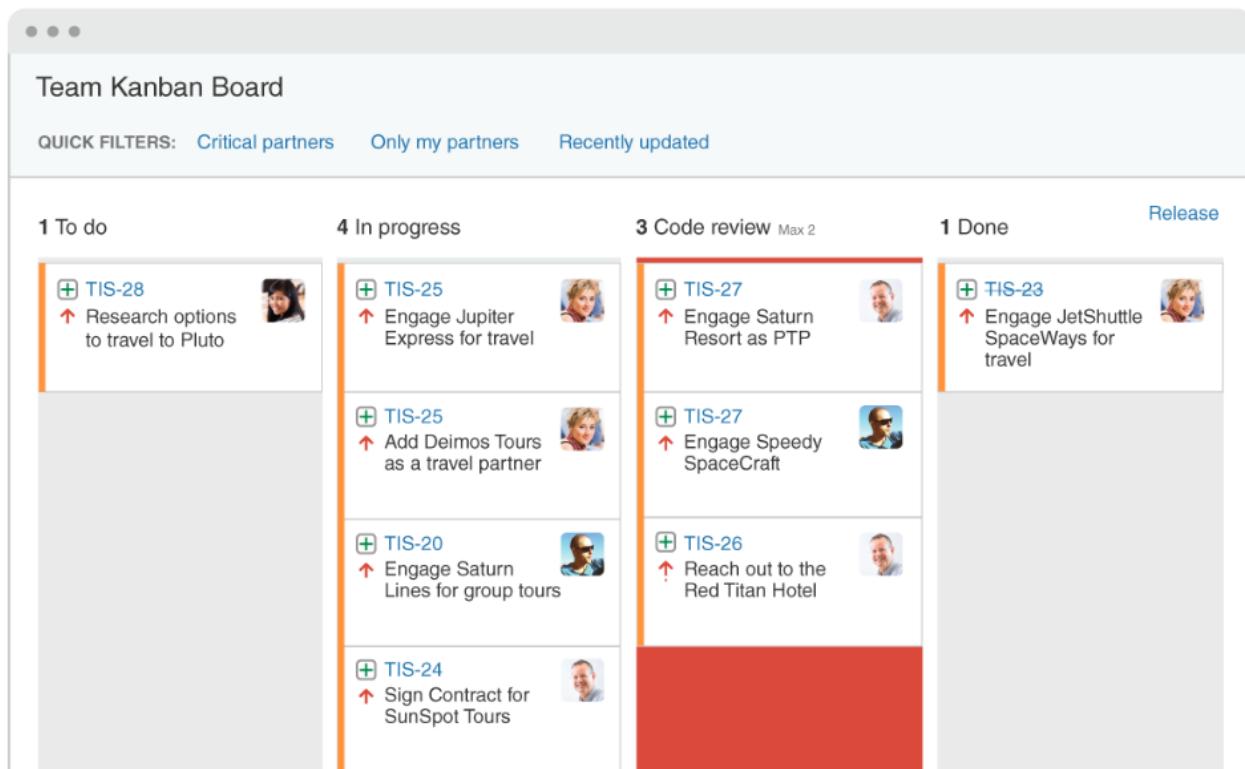


Figure 22: Agile Kanban Model Image (Radigan, 2025)

[Back to Considered Methodology](#)

12.2. Work Breakdown Structure (WBS)

A Work Breakdown Structure (WBS) is a way to break project into smaller, manageable parts. It starts with the main goal and divides it into major deliverables, sub-deliverables, and small tasks. This helps teams plan clearly, assign work, and track progress, ensuring nothing is missed (Project Management Institute, 2017).

In projects like a Smart Job and Skill Portfolio web application, WBS organizes complex modules such as authentication, portfolios, AI job recommendations, ATS resume scanning, and admin features. It improves planning, resource allocation, and monitoring, making development more structured and manageable.

[Back to Work Breakdown Structure \(WBS\)](#)

12.3. Milestones

Milestones are key points or checkpoints in a project that help track progress and ensure major tasks are completed on time. They act as markers to show when important phases or deliverables are finished. In this solo project, milestones help monitor progress, manage time effectively, and identify any delays early. Milestones do not represent detailed tasks but rather significant achievements that indicate the project is moving forward as planned.

[Back to Milestones](#)

12.4. Gantt Chart

A Gantt chart is a visual project management tool that shows tasks or activities on a timeline, indicating when each task starts and ends, their duration, and how tasks overlap or depend on each other. For the Smart Job and Skill Portfolio project, the Gantt chart helps track the progress of features like authentication, portfolio creation, AI-based job recommendations, ATS resume scanning, and admin dashboards. It allows efficient time management, helps identify potential delays early, and ensures that each development milestone is completed on schedule (Pressman, 2010).

[Back to Gantt Chart](#)