



Polytechnic University of Puerto Rico

Electrical & Computer Engineering and Computer Science Department

CECS 4204-80 Software Engineering

Prof. Carlos M. De La Torre Ugarte

Software Project Management Plan

Version: 1.0

Latest revision:



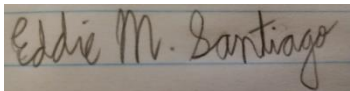
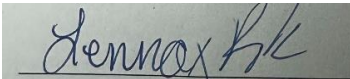
10/21/2024

Prepared by:

Coral S. Schmidt Montilla

For:

Prof. Carlos M. De La Torre Ugarte

Member	ID#	Program	Signature
Coral S. Schmidt Montilla	#148830	Computer Science	
Taishali N. Jimenez Quinones	#148017	Computer Engineering	
Eddie M. Santiago Malave	#116933	Computer Engineering	
Lennox Rivera	#140874	Computer Engineering	

History of Changes

The following table represents the work done by each member and the version of the SRS where the revision was made.

Version	Description	Modified by	Date
1.0	Finished the Introduction	Coral S. Schmidt Montilla	09/20/2024
1.0	Finished the Project Organization	Coral S. Schmidt Montilla	09/22/2024
1.0	Finished Managerial Process	Coral S. Schmidt Montilla	09/27/2024
1.0	Finished Technical Process	Coral S. Schmidt Montilla	09/29/2024
1.0	Finished Work Packages, Budget, and Schedule	Coral S. Schmidt Montilla	10/04/2024
1.0	Finished Tables	Coral S. Schmidt Montilla	10/06/2024
1.0	Finished Figures	Coral S. Schmidt Montilla	10/12/2024
1.0	Finished Scope and References	Coral S. Schmidt Montilla	10/14/2024
1.0	Finished Revising Every Section, Table, and Figure	Coral S. Schmidt Montilla	10/21/2024

Table of Contents

Scope and References	6
Scope	6
References.....	6
1. Introduction.....	7
1.1 Project Overview.....	7
1.2 Project Deliverables	7
1.3 Evolution of the SPMP	8
1.4 Definitions and Acronyms.....	9
2. Project Organization	11
2.1 Process Model.....	11
2.2 Organizational Structure	12
2.3 Organizational Boundaries and Interfaces.....	13
2.4 Project Responsibilities	13
3. Managerial Process	15
3.1 Management Objectives and Priorities.....	15
3.2 Assumption, Dependencies and Constraints	15
3.3 Risk Management	16
3.4 Monitoring and Controlling Mechanisms.....	16
3.4.1 Weekly Group Meetings	18
3.4.2 Weekly Editing by All Team Members	18
3.5 Staffing Plan	19
4. Technical Process	20
4.1 Methods, Tools, and Techniques	21
4.1.1 Development Methodologies	21
4.1.2 Team Structure.....	21
4.1.3 Programming Languages	22
4.2 Software Documentation.	22
4.3 Project Support Functions	23
5. Work Packages, Budget, and Schedule.....	24
5.1 Works Packages	24
5.2 Dependencies	26
5.3 Resource Requirements.....	27
5.4 Budget and Resource Allocation.....	28
5.5 Schedule	29

List of Tables

Table 1: Project Deliverables.....	7
Table 2: Definitions	9
Table 3: Acronyms.....	10
Table 4: Abbreviations.....	10
Table 5: Team Roles	12
Table 6: Risk Assessment	16
Table 7: Monitoring and controlling mechanisms	17
Table 8: Member roles and required skills	19
Table 9: Technical Process	20
Table 10: Software documentation	22
Table 11: Additional Documents.	23
Table 12: Work Packages	25
Table 13: Resource Requirements	28
Table 14: Budget and Resource	29
Table 15: Gantt Chart of Project Schedule	30
Table 16: Project Manager Gantt Chart	31
Table 17: Front-End Developer Gantt Chart	32
Table 18: Back-End Developer Gantt Chart	32
Table 19: Quality Manager Gantt Chart	33
Table 20: Business Analyst/System Analyst Gantt Chart.....	34

List of Figures

Figure 1: Process Model	11
Figure 2: Organizational Structure.....	12
Figure 3: Agile D.M.....	21
Figure 4: Dependencies.....	26
Figure 5: Gantt Chart of Project Schedule	30
Figure 6: Project Manager Gantt Chart.....	31
Figure 7: Front-End Developer Gantt Chart	31
Figure 8: Back-End Developer Gantt Chart.....	32
Figure 9: Quality Manager Gantt Chart	33
Figure 10: Business Analyst/System Analyst Gantt Chart	33

Scope and References

Scope

The Resume Builder project aims to deliver a web-based application that allows users to create professional resumes with ease. The application will guide users through a series of steps to input personal, educational, and professional information, select from a range of resume templates, and generate a downloadable PDF document. This tool is targeted at job seekers, students, and professionals who require assistance in building their resumes. The project will be developed using Agile methodologies, with iterative feedback from stakeholders, ensuring that the product evolves based on user needs. Key features include template selection, resume customization, and integration with external APIs for job-specific recommendations.

References

The following references are used in preparing this SPMP. These references are cited using the IEEE format.

[1]

S. Laoyan, “What Is Agile Methodology? (A Beginner’s Guide) [2024] • Asana,” *Asana*, Feb. 02, 2024. Available: <https://asana.com/resources/agile-methodology>. [Last Accessed: Oct. 21, 2024]

[2]

“Application Risk Management,” *Snyk*, 2024. Available: <https://snyk.io/learn/application-security/application-risk-management/>. [Last Accessed: Oct. 11, 2024]

[3]

“Sample Gantt Chart Project,” *Teamgantt.com*, Sep. 25, 2024. Available: <https://www.teamgantt.com/what-is-a-gantt-chart>. [Last Accessed: Oct. 21, 2024]

1. Introduction

Following a survey conducted with over 200 students at the Polytechnic University of Puerto Rico, San Juan Campus, it was found that a significant number of students lack the skills necessary to create professional resumes tailored to specific job applications. This highlights a clear need for an accessible and user-friendly solution. Our project aims to address this by developing a web-based application that simplifies the resume-building process for users with limited or no prior experience. By providing customizable templates and step-by-step guidance, the application ensures that users can create high-quality resumes efficiently and effectively for a variety of job opportunities.

1.1 Project Overview

This project focuses on developing a user-friendly web application designed to assist users in creating professional resumes. The application will guide users through customizable templates, helping them input their personal, educational, and professional information to generate a PDF resume. The tool will target a wide audience, including students, professionals, and job seekers, offering various templates tailored to different industries. The project will be developed using the Agile Development Methodology (Agile D.M.), ensuring flexibility and continuous improvements based on user feedback and testing. Key functionalities include resume template selection, customization, and integration with external APIs for job-specific recommendations.

1.2 Project Deliverables

This table outlines the key project deliverables, their delivery methods, recipients, and deadlines, ensuring progress is tracked and communicated effectively to the Project Manager, QA team, and Stakeholders.

Deliverable (amount)	Delivery Method	Person to Deliver to	Date to deliver
Front-End UI Prototype (1)	Presentation/Demo	Project Manager & Stakeholders	November 2024
Back-End API Integration (1)	Code Review	Project Manager & QA Team	January 2025
Alpha Testing Report (1)	Report	Project Manager & Stakeholders	March 2025
Beta Testing Report (1)	Report	Project Manager & Stakeholders	May 2025
Final Documentation (1)	PDF Document	Project Manager & Stakeholders	May 2025

Table 1: Project Deliverables.

1.3 Evolution of the SPMP

The SPMP will evolve iteratively throughout the project lifecycle. In its initial version, the document will outline the project framework, including the scope, organizational structure, and high-level objectives. Subsequent versions will reflect changes in project requirements, progress, and any adjustments in the scope. Each update will incorporate feedback from stakeholders and users, as well as new developments or risks identified during the project. The final version of the SPMP will provide a comprehensive overview of the entire project, including its planning, execution, and delivery stages.

1.4 Definitions and Acronyms

It was determined that the following definitions, acronyms and abbreviations are important for the full understanding of project development.

Terms	Definition
Application	A program or software designed to fulfill a particular purpose for the user, often accessed through a web or mobile interface.
User Interface	The visual part of the app that users interact with to input information or control the program.
Resume	A formal document listing a person's professional and educational track record and achievements.
Application Programming Interface	A set of functions that allow the app to communicate with external services.
Portable Document Format	A file format that ensures documents maintain consistent formatting for easy sharing and printing.
Agile Development Methodology	A project management framework that breaks projects down into several dynamic phases, commonly known as sprints.
Cloud Storage	A service that allows users to save data on remote servers, accessed over the internet.
Testing Framework	A tool or set of tools used to automate the testing of software.
Version Control	A system that records changes to files or projects over time so that you can recall specific versions later.
Stakeholder	A person, group, or organization that has an interest in the project and can affect or be affected by the outcome.
Deployment	The process of making the application available for use by end-users.
Sprint	A set period during which specific work must be completed in Agile development.

Table 2: Definitions

Acronym	Meaning
UI	User Interface
API	Application Programming Interface
PDF	Portable Document Format
DB	Database
SPMP	Software Project Management Plan
SRS	Software Requirements Specification
STD	Software Test Document
SDD	Software Design Descriptions
QA	Quality Assurance
AWS	Amazon Web Services

Table 3: Acronyms

Abbreviations	Meaning
App	Application
Agile D.M.	Agile Development Methodology
T.B.D.	To be determined
U.I.	User Interface
P.M.	Project Manager

Table 4: Abbreviations

2. Project Organization

The Project Organization outlines the structure and roles within the Resume Builder project. It defines the responsibilities of each team member, including the Project Manager, Front-End Developer, Back-End Developer, Quality Manager, and Business Analyst. The section also describes the Agile development process being followed, emphasizing collaboration, regular communication, and iterative progress.

2.1 Process Model

The project will follow an Agile development methodology. This model is ideal for our Resume Builder project due to its interactive nature, allowing regular feedback and adjustments. Agile D.M. focuses on incremental development, where features are delivered in small functional portions known as sprints. This allows the development team to adapt to feedback from users and stakeholders quickly. The following figure outlines how each part of Agile D.M. is implemented into our Resume builder project.

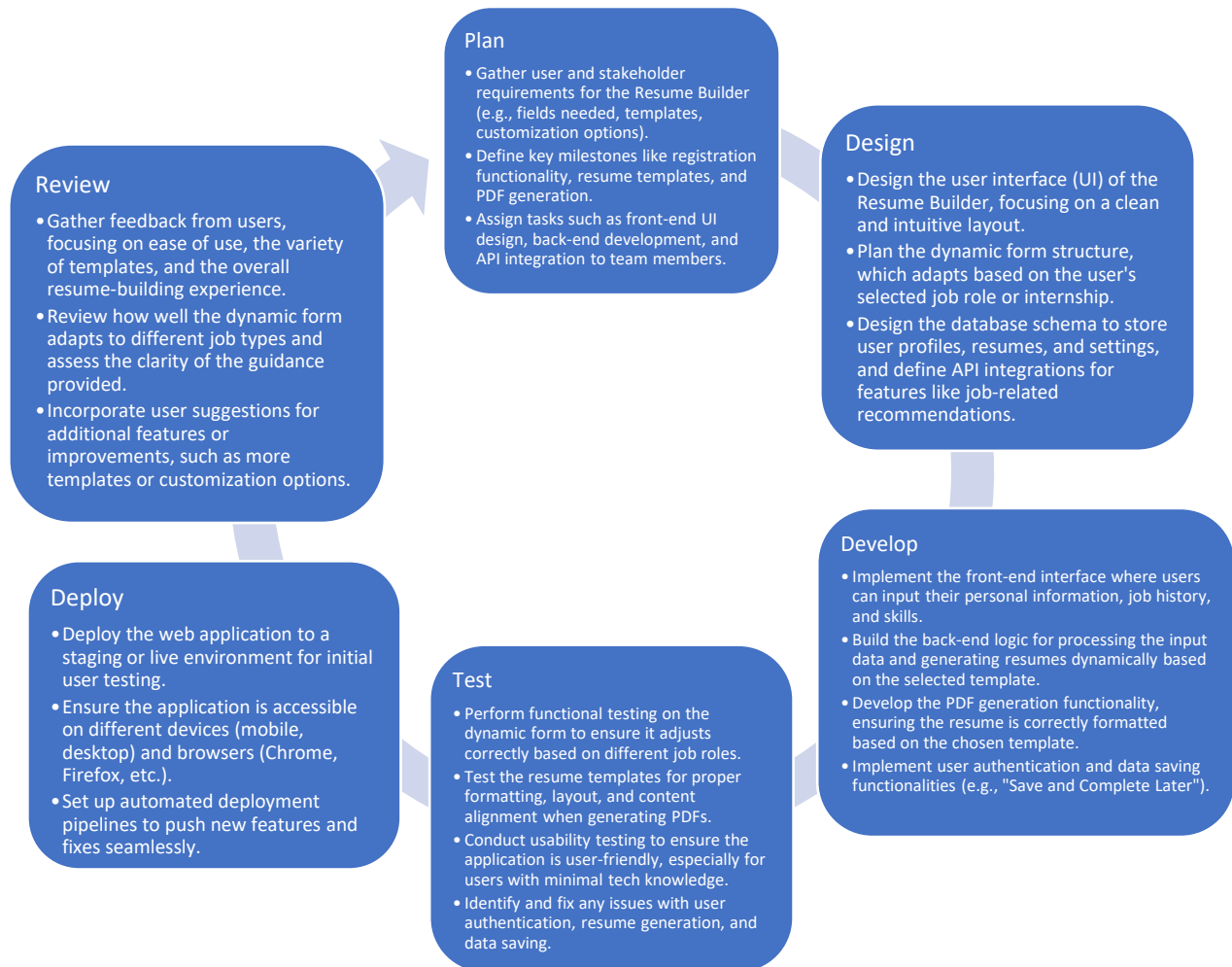


Figure 1: Process Model

2.2 Organizational Structure

The project is structured with clearly defined roles to ensure efficiency and accountability. Each team member has specific responsibilities to support the project's success. The following table and figure outline the roles and responsibilities of the project members:

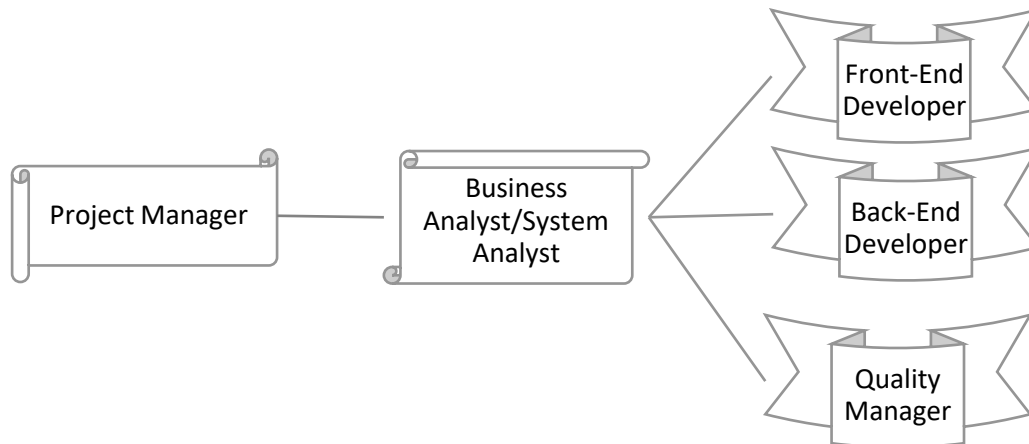


Figure 2: Organizational Structure

This table outlines the roles and responsibilities of the project team members, ensuring clear accountability and defining each member's contributions to the development of the Resume Builder.

Name	Role	Definition
Coral S. Schmidt	Project Manager	Oversees the project's timeline, manages resources, and ensures clear communication between the team and stakeholders.
Taishali N. Jimenez	Software Engineers (Front-End Developer)	Designs and implements the user interface (UI) for the Resume Builder, ensuring responsiveness across devices and a smooth user experience.
Lennox Rivera	Software Engineers (Back-End Developer)	Handles server-side operations, database management, and API integration, ensuring that the resume data is processed, stored, and delivered correctly.
Eddie M. Santiago	Quality Manager	Conducts tests to ensure that the Resume Builder meets all functional and performance criteria, ensuring a bug-free user experience.
Coral S. Schmidt	Business Analyst/System Analyst	Collects requirements from users and stakeholders, translating them into functional specifications for the development team. They ensure that the Resume Builder meets user needs by continuously refining features based on feedback.

Table 5: Team Roles

2.3 Organizational Boundaries and Interfaces

The Resume Builder project includes collaboration across several boundaries:

Development Team:

- The Front-End and Back-End Developers collaborate with the Project Manager to ensure feature implementation aligns with project timelines.
- The Back-End Developer interfaces with the front-end components to ensure proper data flow, resume generation, and template application.
- Works closely with the Quality Manager to resolve any bugs or issues detected during testing.

Quality Assurance:

- Works alongside the Development Team to test the user interface, form adaptability, and the resume generation process (e.g., PDF download functionality).
- Provides continuous feedback to the Development Team and Business Analyst regarding user experience and software functionality.

Business Analyst:

- Interfaces with the stakeholders and users to gather feedback on the Resume Builder.
- Collaborates with the Project Manager to prioritize features and ensure the application remains user-centric.
- Provides insights into market trends for resume design and job application processes, which are then integrated into the app's templates and functionality.

2.4 Project Responsibilities

This section defines the specific responsibilities of each project role, ensuring that tasks are clearly assigned and that team members understand their duties in managing, developing, testing, and analyzing the Resume Builder project.

Project Manager

- Ensure the project is progressing according to the planned timeline, with appropriate resource allocation and stakeholder communication.
- Coordinate between the Business Analyst and Development Team to ensure features are implemented according to user requirements.

Software Engineers (Front-end Developer)

- Develop and maintain the Resume Builder's user interface, ensuring that the dynamic form adjusts based on the user's job selection and inputs.
- Implement features such as template selection and preview, ensuring the application is intuitive and user-friendly.

Software Engineers (Back-end Developer)

- Develop server-side logic, managing user data securely and ensuring proper resume generation.
- Handle API integrations that provide job-specific recommendations or dynamic content for resume sections (e.g., skills, experiences).

Quality Manager

- Test the Resume Builder across different platforms (mobile, desktop) to ensure responsive design.
- Test functionalities like form adaptability, PDF generation, and user login to guarantee smooth operations.
- Document bugs and issues, working with developers to resolve them.

Business Analyst/ System Analyst

- Gather detailed requirements from users, particularly related to the types of resumes (e.g., job-specific or general-purpose).
- Ensure that the resume templates provided are aligned with industry standards and cater to the various needs of users (e.g., students, professionals).
- Provide suggestions for improvement based on user feedback and market trends, ensuring the product stays competitive.

3. Managerial Process

The Managerial Process outlines the methods for overseeing the Resume Builder project, focusing on management objectives, risk management, monitoring mechanisms, and staffing. It ensures the project remains on schedule, addresses risks proactively, and maintains clear communication and accountability among the team through regular reports and meetings.

3.1 Management Objectives and Priorities

The primary objective of the management team is to ensure the successful delivery of the Resume Builder application within the set timeframe and budget. The project follows the Agile D.M., allowing for iterative progress and continuous feedback from stakeholders. Priorities include meeting the functional and performance requirements outlined in the SRS, maintaining user-centric development, and ensuring that critical features, such as resume generation and template selection, are implemented early in the development cycle. The management will prioritize risk management by addressing potential issues like external API downtimes and data security vulnerabilities promptly. Additionally, delivering a user-friendly interface and maintaining regular communication between team members and stakeholders will be central to project success.

3.2 Assumption, Dependencies and Constraints

This section outlines the key assumptions, dependencies, and constraints for the Resume Builder project, clarifying the project's foundational conditions, external dependencies, and regulatory or budgetary limitations that must be addressed to ensure successful completion.

a. Assumptions

- Users will access the application via modern web browsers (Chrome, Firefox, Safari, DuckDuckGo, ...).
- External APIs (e.g., job databases) will be available and reliable for integration into the app.
- Users will require minimal technical expertise to use the Resume Builder.

b. Dependencies

- The successful deployment of the Resume Builder depends on the integration of external job databases to suggest relevant resume sections (skills, experience, ...).
- Cloud storage for saving user data must be reliable and secure to allow users to save and complete resumes at a later time.

c. Constraints

- The Resume Builder must comply with data protection regulations (e.g., GDPR, ...) to ensure the safe handling of user data.
- The project must be completed within the pre-determined budget and timeframe.

3.3 Risk Management

This table identifies potential risks for the Resume Builder project, their effects, and contingency plans to mitigate these risks, ensuring proactive management of issues that could impact the project's success.

Risk Scenario	Effect	Contingency plan
External API service downtime	Users cannot retrieve job-specific resume suggestions.	Provide fallback options with general suggestion; inform users of the service status.
Security vulnerability in the app	Users' personal data could be compromised.	Conduct regular security audits; implement encryption for all data transactions.
Delayed deployment	Project delivery falls behind schedule.	Implement agile sprints to ensure incremental delivery; prioritize key functionalities first.
Browser incompatibility	Users may experience issues using the app.	Preform cross-browser testing to ensure compatibility with all major browsers.

Table 6: Risk Assessment

3.4 Monitoring and Controlling Mechanisms

The following mechanisms are implemented to monitor and control the development process of the Resume Builder project, ensuring progress is on track and any issues are promptly addressed:

1. Weekly Reports
 - a. Team members will submit a weekly progress report every Friday detailing the work completed during the week, challenges encountered, and any assistance required for the next phase.
 - b. These reports will be consolidated by the Project Manager and shared with stakeholders on Monday to provide a clear status update.

2. Group Meetings

- a. Weekly meetings will be held every Monday to review the previous week's progress, address any unresolved issues, and plan tasks for the upcoming week.
- b. Emergency meetings may be arranged mid-week if any critical issues arise that could impede project progress or deadlines.

3. Binnacle

- a. A project binnacle will be maintained to document all significant discussions, decisions, and actions from weekly meetings, particularly those affecting the project's scope or timeline.
- b. The binnacle will be reviewed in Monday meetings to ensure accountability and adherence to planned activities, as well as to monitor any changes to project requirements.

4. Itinerary

- a. A detailed project itinerary will be updated bi-weekly, outlining the key deliverables and their respective deadlines.
- b. The Project Manager will ensure that the itinerary aligns with the progress reports and make necessary adjustments as needed to avoid delays and scope creep.

Mechanisms	Day	Hour
Weekly Reports	Friday	5:00 pm
Group Meetings	Monday	10:00 am
Binnacle Updates	Monday	11:00 am
Itinerary Review	Bi-weekly Monday	9:00 am

Table 7: Monitoring and controlling mechanisms

3.4.1 Weekly Group Meetings

Weekly group meetings will take place every Monday at 10:00 AM, during which team members will discuss the past week's progress, any obstacles encountered, and the goals for the upcoming sprint. These meetings will serve as checkpoints to review project deliverables, address critical issues, and adjust timelines if necessary. Each team member will present their completed tasks, and any unresolved issues will be logged in the project binnacle. Emergency meetings may be scheduled mid-week if urgent problems arise, that could impact the project's timeline or quality. The Project Manager will ensure that these meetings remain focused and productive, facilitating open communication among all team members.

3.4.2 Weekly Editing by All Team Members

Each week, all team members will be responsible for reviewing and editing their code, documentation, and deliverables. This ensures consistency across the project and early detection of potential issues. All edits must be submitted on Fridays for review, and any conflicting changes will be resolved collaboratively during Monday's group meeting. Weekly editing includes code and documentation such as the SPMP, SRS, STD, and SDD, ensuring that these documents are continuously updated to reflect the latest project developments. This practice helps maintain high-quality deliverables and ensures alignment with the project objectives.

3.5 Staffing Plan

This table lists the roles of project team members and the required skills for each role, ensuring that the team has the necessary expertise to complete the Resume Builder project.

Member	Role	Required Skills (by role)
Coral S. Schmidt	Project Manager	<ul style="list-style-type: none">• Leadership• Communication• Resource management• Agile D.M.• Project coordination
Taishali N. Jimenez	Front-End Developer	<ul style="list-style-type: none">• HTML/CSS• JavaScript• UI/UX design• React or Angular• Debugging
Lennox Rivera	Back-End Developer	<ul style="list-style-type: none">• API integration• Database management• Server-side scripting (Node.js, MySQL, Oracle)• Security protocols
Coral S. Schmidt	Business Analyst/System Analyst	<ul style="list-style-type: none">• Requirement gathering• Process documentation• Communication with stakeholders• Market research• Industry trends
Eddie M. Santiago	Quality Manager	<ul style="list-style-type: none">• Testing strategies• QA methodologies• Cross-platform testing• Debugging• Bug tracking tools

Table 8: Member roles and required skills

4. Technical Process

The Technical Process section outlines the tools, methods, and techniques for developing the Resume Builder project. It includes version control, project management platforms, programming languages, and testing frameworks. Following Agile methodologies, this ensures efficient collaboration, high code quality, and a structured approach to both front-end and back-end development.

Service used	Purpose
Git/GitHub	Version control to manage code, track changes, and enable team collaboration
Trello/Jira	Project management tools to organize tasks, assign responsibilities, and track progress
Slack/Microsoft Teams	Internal communication platform for team collaboration and updates
Jest/Mocha	Automated testing frameworks for unit and integration testing of code
MySQL/Oracle	Database management for storing user information and resume data
Node.js	Server-side scripting for handling back-end logic and API integration
Outsystems	Front-end framework for building dynamic and responsive user interfaces

Table 9: Technical Process

4.1 Methods, Tools, and Techniques

The Methods, Tools, and Techniques section describes the development methodologies, software, and frameworks used in the Resume Builder project. It includes Agile development, tools like GitHub for version control, Outsystems for front-end development, Node.js for back-end logic, and automated testing frameworks to ensure high-quality deliverables.

4.1.1 Development Methodologies

The project follows the Agile D.M., ideal for managing iterative progress and allowing for regular feedback and adjustments. Each development cycle is divided into short sprints where small, functional portions of the Resume Builder are developed, tested, and refined based on stakeholder input. This methodology ensures that high-priority features, such as resume template selection and PDF generation, are implemented early and enhanced iteratively.



Figure 3: Agile D.M.

4.1.2 Team Structure

- Project Manager: Oversees the overall timeline, manages resources, and ensures communication between stakeholders and the team.
- Front-End Developer: Focuses on the user interface (UI) design, ensuring responsiveness and ease of use across devices.
- Back-End Developer: Manages server-side development, including database management and API integration.
- Quality Manager: Conducts testing to ensure functionality and performance criteria are met.
- Business Analyst/System Analyst: Collects requirements and translates them into functional specifications, ensuring that the Resume Builder aligns with user needs.

4.1.3 Programming Languages

- Front-End: HTML, CSS, JavaScript with the Outsystems framework for creating responsive and dynamic user interfaces.
- Back-End: Node.js for server-side scripting, Oracle or MySQL for database management, and REST APIs for external job data integration.
- Other Technologies: JSON for data interchange, PDF generation libraries for creating downloadable resume documents, and Git for version control.

4.2 Software Documentation.

This table lists the critical software documentation required for the project, outlining each document's purpose and the corresponding IEEE format, ensuring comprehensive planning, design, and testing processes.

Document	Description	Format
S.P.M.P.	Software Project Management Plan: Document management refers to the planning of the project, its main stages, and its organization.	IEEE 1058-1998
S.D.D.	Software Design Description: Document describing the product design.	IEEE 1016-1998
S.R.S.	Software Requirement Specifications: Document defining the requirements and details requested by the client.	IEEE 830-1998
S.T.D.	Software Test Documentation: Document defining each test and the tester.	IEEE 829-1998

Table 10: Software documentation

In addition of the software documents there were other documents to complete this document are:

Document	Description
Weekly Reports	The purpose for the weekly reports is to have documented each member's progress.
Group Logs	The purpose of the group logs is to have a brief description of everything discussed within the groups.
Individual Logs	The purpose of the individual logs is to include the completed tasks of each member with a brief description of the task done, what was used and the date.
Presentations	The purpose of the presentations is to give a brief overview of the development process.

Table 11: Additional Documents.

4.3 Project Support Functions

A range of tools and services supports developing and managing the Resume Builder project to facilitate collaboration and ensure smooth progress. Git and GitHub are used for version control, enabling the team to track code changes, manage development branches, and collaborate efficiently. For project management, tools such as Trello or Jira are employed to organize tasks, assign responsibilities, and monitor the progress of each sprint, in line with the Agile D.M. Internal communication is maintained through platforms like Slack or Microsoft Teams, ensuring team members stay connected and aligned with project goals. Additionally, automated testing frameworks like Jest or Mocha perform unit and integration tests, ensuring the application meets quality standards and functions as intended before deployment.

5. Work Packages, Budget, and Schedule

In Work Packages, Budget, and Schedule, the project is broken down into distinct work packages that represent specific tasks or deliverables. Each package is defined by its scope, objectives, and the resources required to complete it. The Budget outlines the financial resources allocated to each work package, ensuring that the project stays within its financial limits. The Schedule provides a timeline for all tasks, as detailed in the Gantt Chart, helping to coordinate the timing of each phase. This section is crucial for ensuring that the project is well-organized, financially controlled, and delivered on time according to the established plan.

5.1 Works Packages

This table outlines the work packages for the Resume Builder project, detailing the tasks, resources, duration (in months), and the associated costs. While the table includes the costs for development and testing individually, it needs to account for the ideal scenario where team members would be paid for maintaining the project after completion, which is not covered in this work package.

Package	Description	Resources	Quantity	Price (per unit)	Price (total)
Planification	Gather user and stakeholder requirements for the Resume Builder, define key milestones, and assign tasks to team members.	Project Manager & Business Analyst/System Analyst	20 days each employee	\$144/day	\$5,760
Front-End Development	Develop the user interface using Outsystems	Front-End Developer	40 days each employee	\$144/day	\$5,760
Back-End Development	Implement server-side logic with Node.js	Back-End Developer	40 days each employee	\$144/day	\$5,760
Database Integration	Set up MySQL/Oracle for user data and resume storage	Back-End Developer	20 days each employee	\$144/day	\$2,880
API Integration	Integrate external job databases for resume suggestions	Front-End Developer & Back-End Developer	20 days each employee	\$144/day	\$5,760
Testing	Perform unit and integration tests with Jest/Mocha	Quality Manager	40 days each employee	\$144/day	\$5,760
Documentation	Maintain project documentation (SPMP, SRS, SDD, STD) (continuous throughout the project)	Business Analyst/System Analyst	140 days each employee	\$144/day	\$20,160
Deployment	Deploy the web application	All employees	10 days each employee	\$144/day	\$7,200
			Total	\$1,152/day	\$59,040

Table 12: Work Packages

5.2 Dependencies

Dependencies illustrate the hierarchy of tasks in the Resume Builder project, where each level depends on completing the one above it. The project starts with planification, the groundwork for the entire process. Documentation runs throughout the project and relies on the planning phase to establish guidelines. Once documentation is in place, Front-End and Back-End Development can proceed. After development, API Integration & Database Setup depends on the established architecture to ensure proper connections and data flow. Testing, a crucial step, follows to verify that all components function correctly, providing reassurance about the quality of our work. Finally, deployment can occur after successful testing and application launch. Each phase builds upon the successful completion of the prior stage.

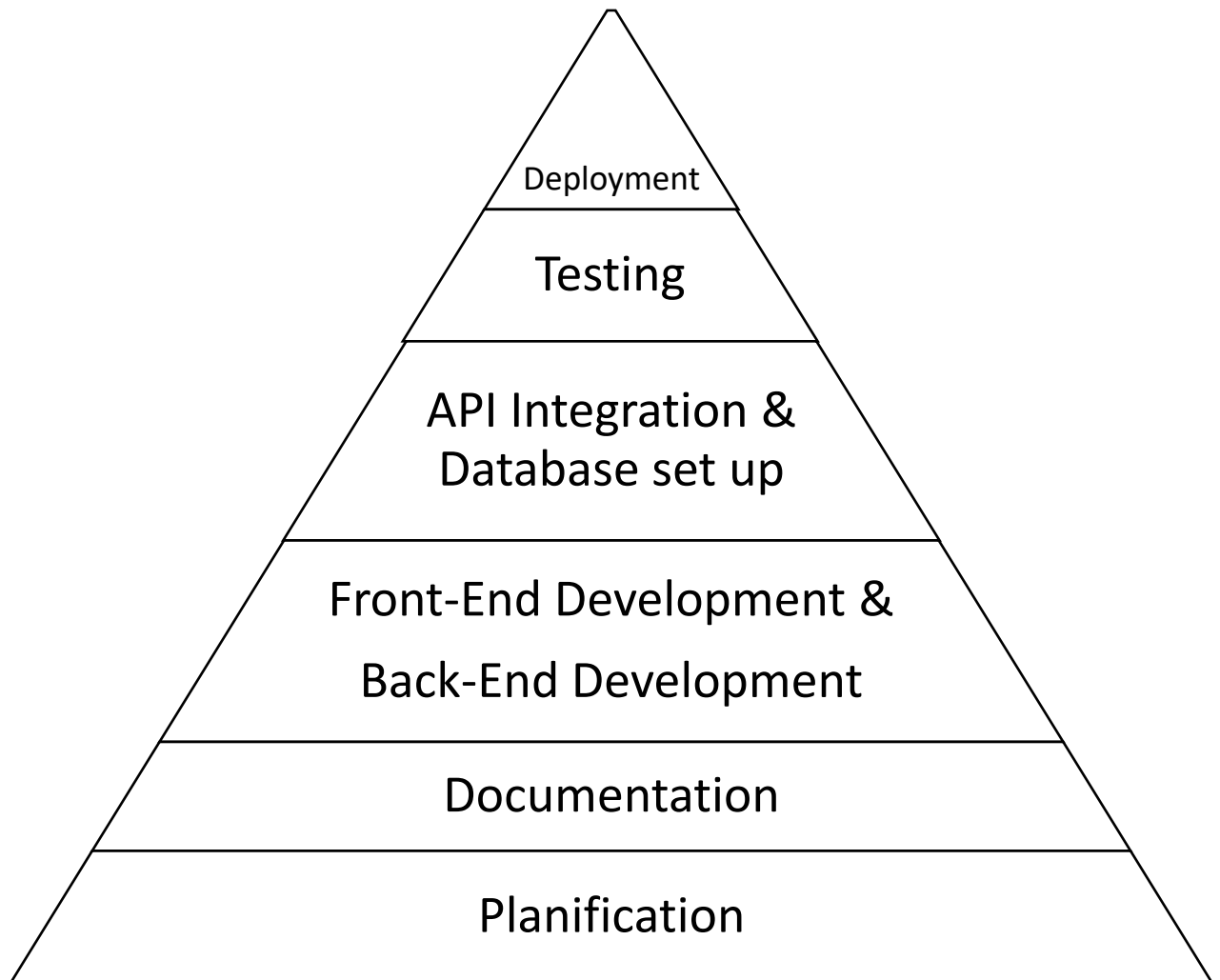


Figure 4: Dependencies

5.3 Resource Requirements

The Resource Requirements section outlines the personnel, software, hardware, and other resources necessary to complete the Resume Builder project successfully. It includes critical roles such as Front-End and Back-End Developers, QA Tester, Project Manager, and Business Analyst. Software tools include Outsystems, Node.js, MySQL/Oracle, GitHub, and Jest/Mocha for testing. Hardware includes developer workstations and cloud servers, while other resources cover hosting services and project management tools like Trello or Jira.

- **Personnel:**

- a. Front-End Developer: Responsible for building the user interface using Outsystems.
- b. Back-End Developer: Manages server-side scripting with Node.js and database integration.
- c. QA (Quality Assurance) Tester: Tests the application using Jest/Mocha frameworks.
- d. Project Manager: Oversees the project timeline, coordinates resources, and manages communication.
- e. Business Analyst/System Analyst: Gathers requirements and provides continuous feedback from stakeholders.

- **Software components:**

- a. Outsystems: For front-end development.
- b. Node.js: For back-end server-side logic.
- c. MySQL/Oracle: For database management and user data storage.
- d. Git/GitHub: For version control.
- e. Jest/Mocha: For automated testing.
- f. API Services: External job databases for resume suggestions.

- **Hardware components:**

- a. Developer Workstations: Computers used by developers for coding and testing.
- b. Servers: Cloud-based hosting for the application (e.g., AWS, Azure).

- **Other Resources:**

- a. Hosting Services: To deploy the application on a web server (e.g., AWS, Azure).
- b. Project Management Tools: Trello or Jira for task management.

Resource	Software	Hardware	Usage	Roles
Outsystems	✓		Front-end development platform	Front-End Developer
Node.js	✓		Server-side scripting	Back-End Developer
MySQL/Oracle	✓		Database for storing user and resume data	Back-End Developer
Jest/Mocha	✓		Unit and integration testing frameworks	QA Tester
Git/GitHub	✓		Version control	Development Team
Developer Workstations		✓	Workstations for coding and testing	All Developers
Servers (AWS, Azure)		✓	Hosting the web application	Back-End Developer
Project Management Tools	✓		Task management (Trello/Jira)	Project Manager

Table 13: Resource Requirements

5.4 Budget and Resource Allocation

This table presents the budget and resource allocation for the Resume Builder project, including estimated costs for the OutSystems license, API integration, MySQL/Oracle data storage, team salaries, and miscellaneous expenses. The estimated cost for team salaries amounts to \$100,800, due to the plan to pay all team members \$18 per hour for full-time work (40 hours per week) over the 8-month and 2-week duration of the project. The total estimated cost is \$138,275. These prices have been overestimated to ensure sufficient coverage for any unexpected increases in base costs, such as higher platform fees or additional storage requirements for the application.

Resource	Details	Estimated Cost
OutSystems License	Cost of the platform	\$36,300/yr
API Integration	Fees for accessing external job databases	\$100
MySQL/Oracle	Cost for data storage (depends on usage plan)	\$75/mo
Development Tools	Software tools for coding, testing, and deployment	N/A (open source)
Team Salaries	Compensation for development and testing teams	\$100,800
Miscellaneous	Additional expenses (e.g., marketing, maintenance)	\$1000
	Total	\$138,275

Table 14: Budget and Resource

5.5 Schedule

The Schedule section of the project provides an organized timeline for the different phases of development, clearly detailing the start and end dates, as well as the duration of each task. It ensures that the project stays on track by defining key milestones such as planification, front-end and back-end development, database and API integration, testing phases, deployment, and documentation. This section is critical for planning the workflow, assigning tasks to team members, and managing resources efficiently. By following this structured schedule, the project team can monitor progress, anticipate potential delays, and ensure that each phase is completed within the designated timeframe, contributing to the successful delivery of the project.

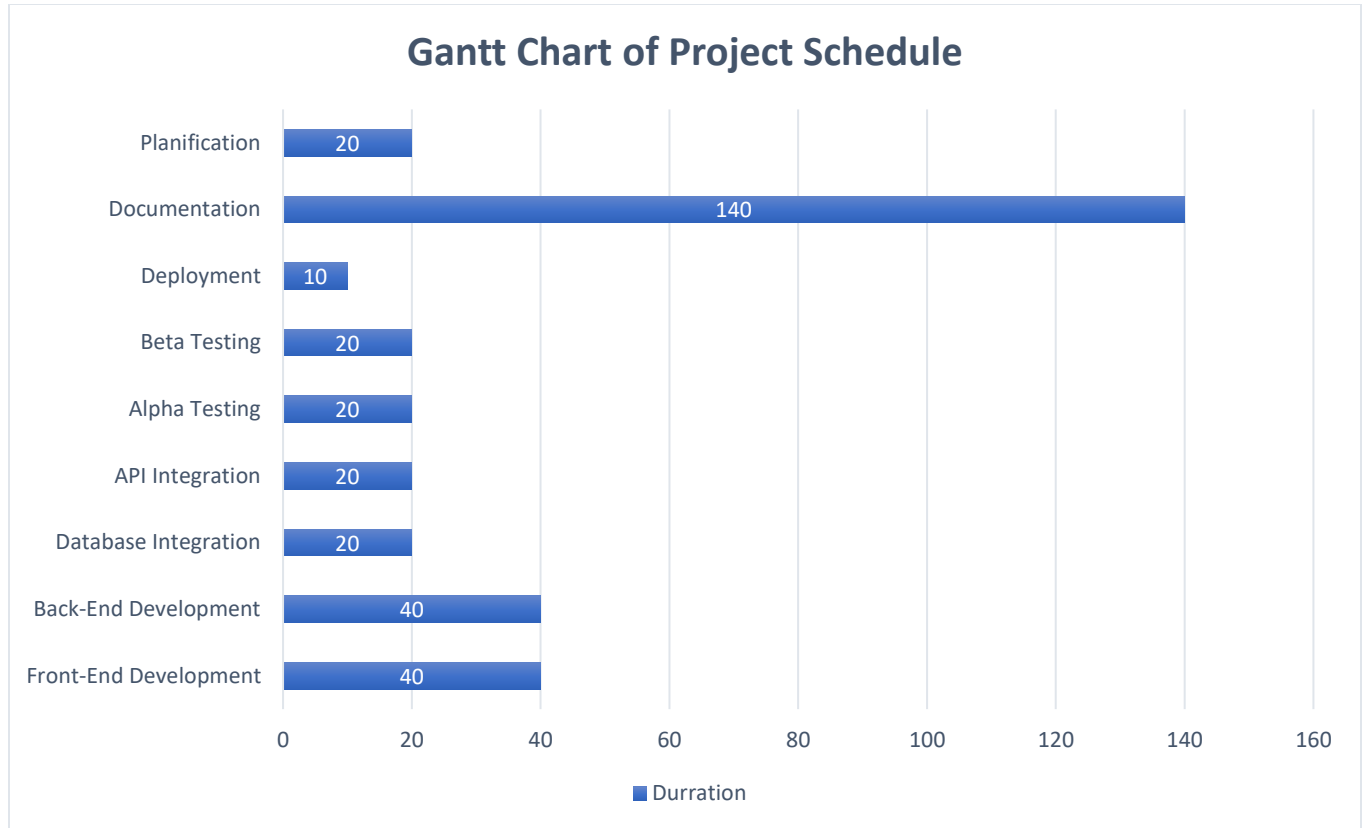


Figure 5: Gantt Chart of Project Schedule

Gantt Chart of Project Schedule			
Task	Start Date	End Date	Duration (Days)
Planification	9/30/2024	10/25/2024	20
Front-End Development	10/28/2024	12/20/2024	40
Back-End Development	10/28/2024	12/20/2024	40
Database Integration	1/6/2025	1/31/2025	20
API Integration	1/6/2025	1/31/2025	20
Alpha Testing	2/3/2025	2/28/2025	20
Beta Testing	3/3/2025	3/28/2025	20
Deployment	3/31/2025	4/11/2025	10
Documentation	9/30/2024	4/11/2025	140

Table 15: Gantt Chart of Project Schedule

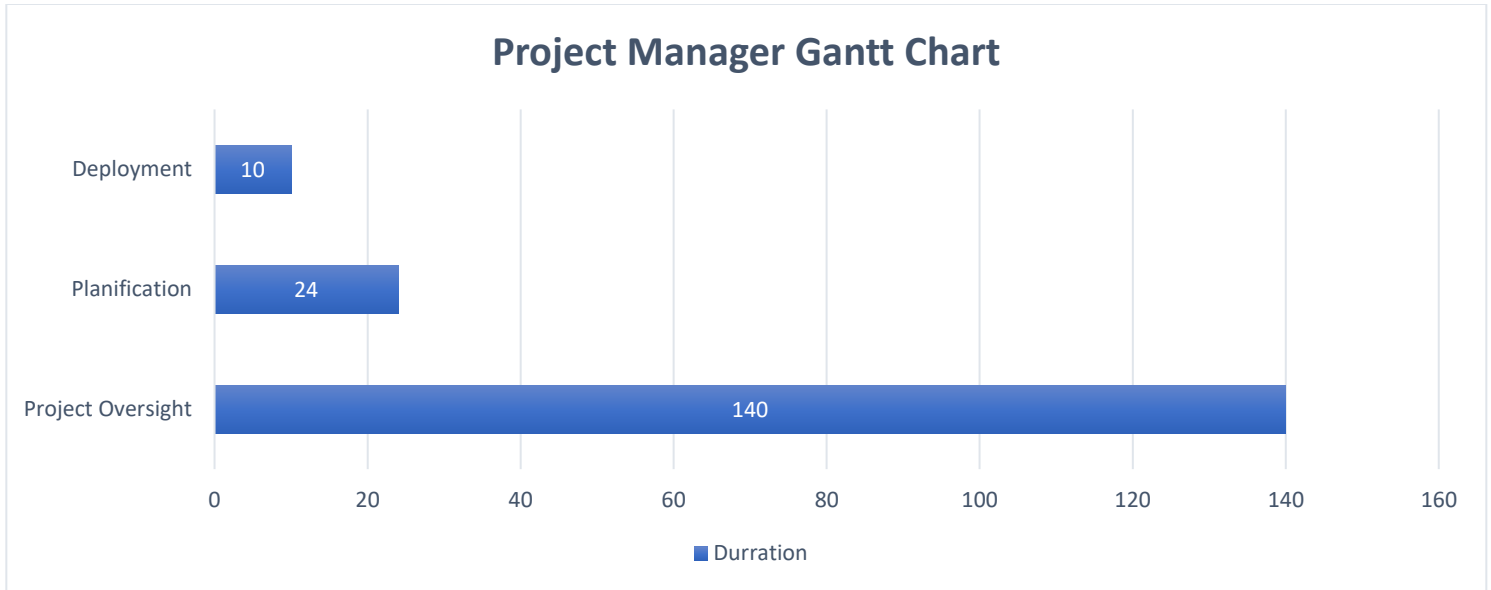


Figure 6: Project Manager Gantt Chart

Project Manager Gantt Chart			
Task	Start Date	End Date	Duration (Days)
Project Oversight	9/30/2024	4/11/2025	140
Planification	10/1/2024	11/1/2024	24
Deployment	3/31/2025	4/11/2025	10

Table 16: Project Manager Gantt Chart

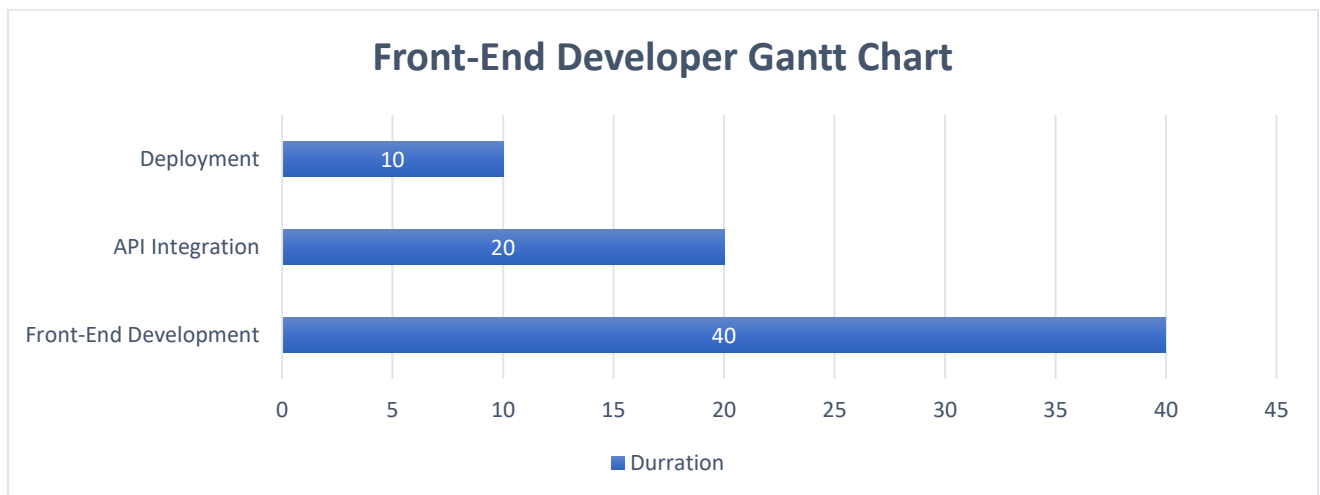


Figure 7: Front-End Developer Gantt Chart

Front-End Developer Gantt Chart			
Task	Start Date	End Date	Duration (Days)
Front-End Development	10/28/2024	12/20/2024	40
API Integration	1/6/2025	1/31/2025	20
Deployment	3/31/2025	4/11/2025	10

Table 17: Front-End Developer Gantt Chart

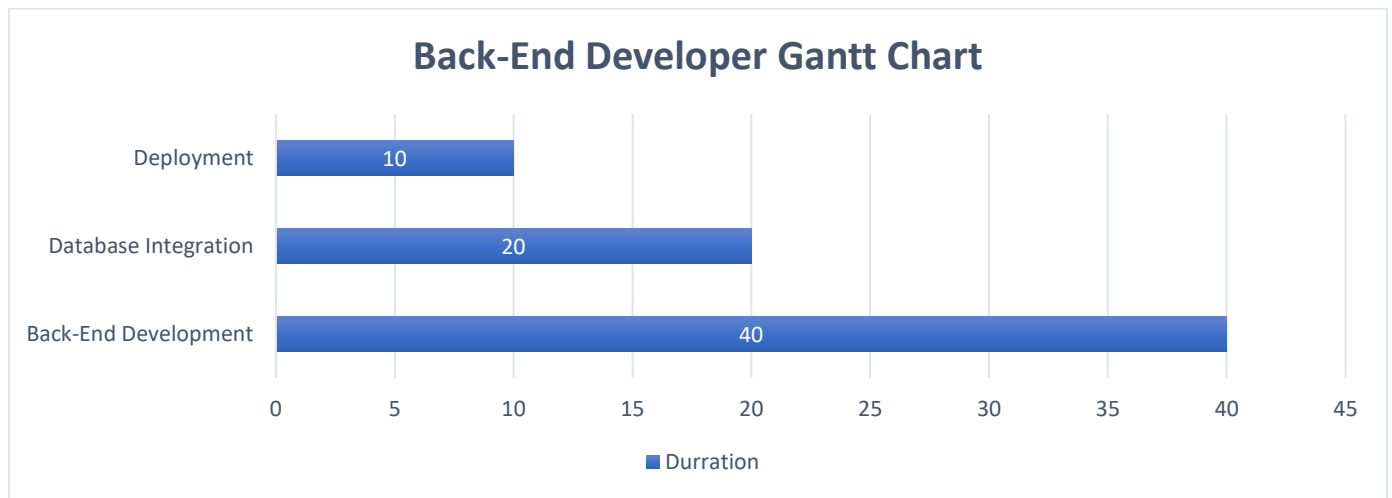


Figure 8: Back-End Developer Gantt Chart

Back-End Developer Gantt Chart			
Task	Start Date	End Date	Duration (Days)
Back-End Development	10/28/2024	12/20/2024	40
Database Integration	1/6/2025	1/31/2025	20
Deployment	3/31/2025	4/11/2025	10

Table 18: Back-End Developer Gantt Chart

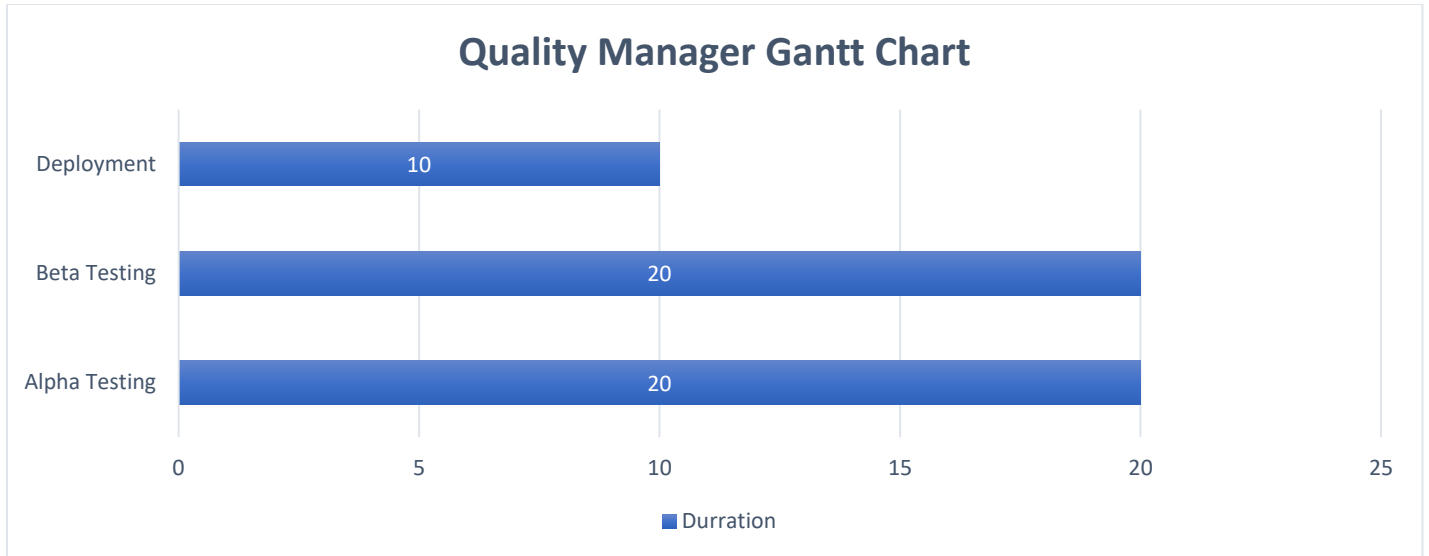


Figure 9: Quality Manager Gantt Chart

Quality Manager Gantt Chart			
Task	Sart Date	End Date	Duration (Days)
Alpha Testing	2/3/2025	2/28/2025	20
Beta Testing	3/3/2025	3/28/2025	20
Deployment	3/31/2025	4/11/2025	10

Table 19: Quality Manager Gantt Chart

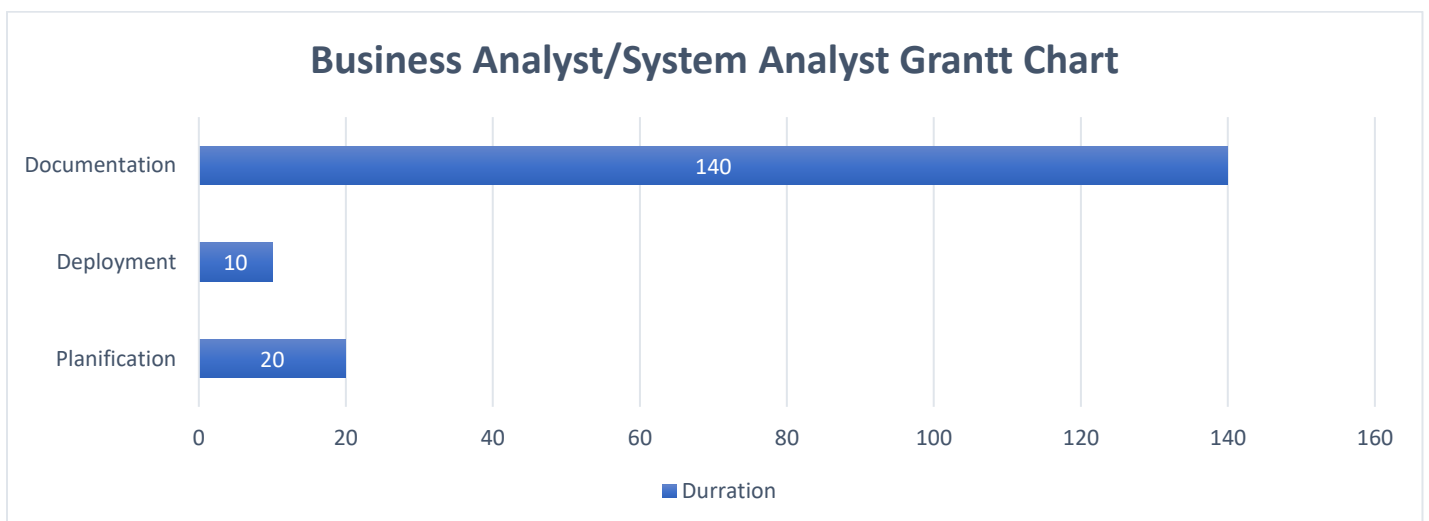


Figure 10: Business Analyst/System Analyst Gantt Chart

Business Analyst/System Analyst Grantt Chart			
Task	Start Date	End Date	Duration (Days)
Planification	9/30/2024	10/25/2024	20
Documentation	9/30/2024	4/11/2025	140
Deployment	3/31/2025	4/11/2025	10

Table 20: Business Analyst/System Analyst Gantt Chart