**A purple square with a pen and a notebook

AI-generated content may be incorrect.**

**Software Requirements Specification**

Prepared By:

|  |  |  |  |
| --- | --- | --- | --- |
| **Member** | **ID#** | **Program** | **Signature** |
| Coral S. Schmidt Montilla | #148830 | Computer Science |  |
| Jaime Cuebas | #121703 | Computer |  |
| Christopher Ortiz | #130607 | Computer |  |
| Christian Brito | #9734 | Computer |  |
| Taishali Jimenez | #148017 | Computer |  |

**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 |
| 1.2 | Finished the Project Organization | Coral S. Schmidt Montilla | 09/22/2024 |
| 1.3 | Finished Managerial Process | Coral S. Schmidt Montilla | 09/27/2024 |
| 1.4 | Finished Technical Process | Coral S. Schmidt Montilla | 09/29/2024 |
| 1.5 | Finished Schedule | Coral S. Schmidt Montilla | 10/04/2024 |
| 1.6 | Finished Tables | Coral S. Schmidt Montilla | 10/06/2024 |
| 1.7 | Finished Figures | Coral S. Schmidt Montilla | 10/12/2024 |
| 1.8 | Finished Scope and References | Coral S. Schmidt Montilla | 10/14/2024 |
| 1.9 | Finished Revising Every Section, Table, and Figure | Coral S. Schmidt Montilla | 10/21/2024 |
| 2.0 | Modified information to align better | Christian Brito | 1/22/2025 |
| 2.1 | Added some finishing touches and edits | Coral S. Schmidt Montilla | 1/29/2025 |
| 2.2 | Updated the Gantt Chart. | Coral S. Schmidt Montilla | 4/22/2025 |
| 2.3 | Update logo. | Coral S. Schmidt Montilla | 5/4/2025 |

**Table of Contents**

[Scope and References 5](#_Toc197247552)

[Scope 5](#_Toc197247553)

[References 5](#_Toc197247554)

[1. Introduction 6](#_Toc197247555)

[1.1 Project Overview 6](#_Toc197247556)

[1.2 Project Deliverables 6](#_Toc197247557)

[1.3 Evolution of the SPMP 7](#_Toc197247558)

[2. Project Organization 9](#_Toc197247559)

[2.1 Process Model 9](#_Toc197247560)

[2.2 Organizational Structure 10](#_Toc197247561)

[2.3 Project Responsibilities 10](#_Toc197247562)

[3. Managerial Process 11](#_Toc197247563)

[3.1 Management Objectives and Priorities 11](#_Toc197247564)

[3.2 Assumption, Dependencies and Constraints 12](#_Toc197247565)

[3.3 Risk Management 12](#_Toc197247566)

[3.4 Monitoring and Controlling Mechanisms 13](#_Toc197247567)

[3.4.1 Weekly Group Meetings 14](#_Toc197247568)

[3.4.2 Weekly Editing by All Team Members 14](#_Toc197247569)

[3.5 Staffing Plan 14](#_Toc197247570)

[4. Technical Process 15](#_Toc197247571)

[4.1 Methods, Tools, and Techniques 16](#_Toc197247572)

[4.1.1 Development Methodologies 17](#_Toc197247573)

[4.1.2 Programming Languages 17](#_Toc197247574)

[4.2 Project Support Functions 18](#_Toc197247575)

[5. Dependencies, Resource Requirements, and Schedule 18](#_Toc197247576)

[5.1 Dependencies 18](#_Toc197247577)

[5.2 Resource Requirements 19](#_Toc197247578)

[5.3 Schedule 21](#_Toc197247579)

**List of Tables**

[Table 1: Project Deliverables. 6](#_Toc196248257)

[Table 2: Definitions 8](#_Toc196248258)

[Table 3: Acronyms 8](#_Toc196248259)

[Table 4: Abbreviations 8](#_Toc196248260)

[Table 5: Team Roles 10](#_Toc196248261)

[Table 6: Risk Assessment 13](#_Toc196248262)

[Table 7: Monitoring and controlling mechanisms 14](#_Toc196248263)

[Table 8: Member roles and required skills 15](#_Toc196248264)

[Table 9: Technical Process 16](#_Toc196248265)

[Table 10: Additional Documents. 17](#_Toc196248266)

[Table 11: Resource Requirements 20](#_Toc196248267)

**List of Figures**

[Figure 1: Process Model 9](#_Toc196248253)

[Figure 3: Agile D.M. 17](#_Toc196248254)

[Figure 4: Dependencies 19](#_Toc196248255)

[Figure 5: Project Schedule Gantt Chart 21](#_Toc196248256)

# 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.

|  |  |  |  |
| --- | --- | --- | --- |
| **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. 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 figure outlines the roles of the project members

| **Name** | **Role** | **Definition** |
| --- | --- | --- |
| Taishali N. Jimenez  Coral S. Schmidt  Christian Brito | 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. |
| Jaime Cuebas  Christopher Ortiz  Christian Brito | 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. |

Table 5: Team Roles

## 2.3 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.

**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.
* 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.
* Ensure that the resume templates provided are aligned with industry standards and cater to the various needs of users (e.g., students, professionals).

**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).
* 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.
* Ensure that the resume templates provided are aligned with industry standards and cater to the various needs of users (e.g., students, professionals).

# 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
   1. 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.
2. Group Meetings
   1. Weekly meetings will be held every Tuesday to review the previous week’s progress, address any unresolved issues, and plan tasks for the upcoming week.
   2. Emergency meetings may be arranged mid-week if any critical issues arise that could impede project progress or deadlines.
3. Binnacle
   1. 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.
   2. 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.

|  |  |  |
| --- | --- | --- |
| **Mechanisms** | **Day** | **Hour** |
| Weekly Reports | Tuesday | 7:00 pm |
| Group Meetings | Tuesday | 7:00pm |

Table 7: Monitoring and controlling mechanisms

### 3.4.1 Weekly Group Meetings

Weekly group meetings will take place every Tuesday at 7:00pm, 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 on the project's timeline or quality.

### 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 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)** |
| --- | --- | --- |
| Taishali N. Jimenez  Coral S. Schmidt  Christian Brito | Front-End Developer | * HTML/CSS * JavaScript * UI/UX design * React or Angular * Debugging |
| Jaime Cuebas  Christopher Ortiz  Christian Brito | Back-End Developer | * API integration * Database management * Server-side scripting (Node.js, MySQL, Oracle) * Security protocols |

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 |
| Out systems | 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

### 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.

A diagram of a process

Description automatically generated

***Figure 3: Agile D.M.***

### 4.1.2 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.

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

| **Document** | **Description** |
| --- | --- |
| Weekly Reports | The purpose of the weekly reports is to have documented each member’s progress. |

Table 10: Additional Documents.

## 4.2 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. Dependencies, Resource Requirements, and Schedule

This section outlines the critical elements required to ensure the structured and timely execution of the Resume Builder project. It begins by detailing the dependencies between different phases of the project, ensuring a logical workflow. Next, the resource requirements necessary for development, including personnel, software, hardware, and additional tools, are specified. Finally, the schedule provides an organized timeline for the project's execution, ensuring each phase is completed within the allocated timeframe.

## 5.1 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

## 5.2 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:**

1. Front-End Developer: Responsible for building the user interface using Outsystems.
2. Back-End Developer: Manages server-side scripting with Node.js and database integration.

* **Software components:**

1. Outsystems: For front-end development.
2. Node.js: For back-end server-side logic.
3. MySQL/Oracle: For database management and user data storage.
4. Git/GitHub: For version control.
5. Jest/Mocha: For automated testing.
6. API Services: External job databases for resume suggestions.

* **Hardware components:**

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

* **Other Resources:**

1. Hosting Services: To deploy the application on a web server (e.g., AWS, Azure).
2. 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 11: Resource Requirements

## 5.3 Schedule

A screen shot of a graph

AI-generated content may be incorrect.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: Project Schedule Gantt Chart