# **[Project Logo (optional)]**

**Project Trireme**

**Software Requirements Specification**

**Version 1**

**Team Number: 12**

**Project Manager: Gustavo Santos Santana**

**Team Members:**

**Patrick Brown****, John Diveris,** **Yifan Gao****,** **Fredrick Hernandez****,** **Michael Lowell**

**Revisions**

| **Version** | **Primary**  **Author(s)** | **Description of Version** | **Date Completed** |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

**Review History**

| **Reviewer** | **Version Reviewed** | **Date** |
| --- | --- | --- |
|  |  |  |
|  |  |  |

**Table of Contents (version 1)**

1. Introduction
   1. Project Objectives
   2. Project Scope
   3. Project Overview
2. Project Description
   1. Project Features / Functions
   2. User Stories
   3. Use Cases
   4. Project Assumptions and Dependencies
3. Project Collaboration and Documentation
4. Project Management
5. Requirements Specification
   1. Business Requirements
   2. User Requirements
   3. Functional Requirements
   4. Non-Functional Requirements
   5. Implementation (Performance) Requirements (Optional)
6. **Introduction**
   1. Project Objectives

< Define the primary goals of the software being developed and set clear expectations for what the system aims to achieve.>

* 1. Project Scope

< Define the boundaries of the project, including what is included and excluded. >

* 1. Project Overview

< Describe the project's goals, the steps to achieve these goals, and the expected outcomes. Outline the project schedule, necessary resources, and status.>

1. **Project Description**
   1. Project Features / Functions

< List at least 3 main features and functionalities to be developed.>

* 1. User Stories

< Simple, concise description of a feature or requirement from the perspective of an end user. User stories are often used in Agile development methodologies to capture user needs and facilitate communication between stakeholders. List at least 3 user story statements to briefly describe the type of user, what they want and why. >

* + 1. As a < type of user >, I want < some goal > so that < some benefits>.
    2. As a < type of user >, I want < some goal > so that < some benefits>.
    3. As a < type of user >, I want < some goal > so that < some benefits>.
  1. Use Case

< Describes a specific interaction between a user (or "actor") and the application to achieve a particular goal. It outlines the steps involved in this interaction and can help to clarify functional requirements. Use cases are especially useful in understanding how users will interact with the application and are typically presented in a structured format. List at least 3 user cases to describe a set of interactions between a system and one or more actors. >

* + 1. Use Case 1 (either Use Case Diagram or Use Case Specification)
    2. Use Case 2 (either Use Case Diagram or Use Case Specification)
    3. Use Case 3 (either Use Case Diagram or Use Case Specification)
  1. Project Assumptions and Dependencies

< Outlines the factors that are assumed to be true for the project and the external elements that the project depends on. These assumptions and dependencies help manage risks by identifying potential challenges or factors that could impact the project’s success.>

1. **Technical Assumption**: The software assumes that it will be given factual data from which it can formulate a study schedule from.
2. **Operational Assumption**: The software assumes that users will have a basic understanding of the time it takes to do a task and familiarity with similar applications, reducing the need for extensive training.
3. **Business Assumption**: The software assumes a stable internet connection will be available to all users for accessing the application at all times.
4. **Resource Constraint**: The project is subject to a fixed budget of $256,342 and must be completed within a 6 month time frame.
5. **Technological Constraint**: The software must be able to run and appear correctly on mobile and traditional computer devices.
6. **External Constraint**: The software must comply with the current data protection regulatory laws and protect user data by handling it securely and in adherence to data protection laws.
7. **Operational Constraint**: The software must support multiple user accounts with different schedules and task assignments, requiring proper user authentication and secure schedule and personal info storage.
8. **Performance Constraint**: The software must handle multiple concurrent requests from a minimum of 680 users as well as maintaining a response time of less than 2 seconds under normal operating conditions.
9. **Project Collaboration and Documentation**

< Specify what collaboration and documentation tools / platforms that will be used in this project development and how your group will use these tools / platforms.>

1. **Project Management**

< Specify what project management methods and tools will be used in this project development and how your group will use these methods and tools. >

1. **Requirements Specification**
   1. **Business Requirements**

< Outline the high-level needs of the business or organization that the application must fulfill. List at least 3 business requirements with MOSCOW code.>

| Requirement ID | Requirement Description | MOSCOW |
| --- | --- | --- |
| BR1 | e.g., The application must provide tools for users to interact with the company, such as customer portals, live chat, and automated notifications. | M |
| BR2 |  |  |
| BR3 |  |  |

* 1. **User Requirements**

< Describe what the end users need or expect from the application. These requirements focus on the functionality and usability aspects of the software from the user's perspective. They outline what the application should enable users to do in order to achieve their goals. List at least 3 user requirements with MOSCOW code.>

| Requirement ID | Requirement Description | MOSCOW |
| --- | --- | --- |
| UR1 | e.g., Users must be able to create an account and log in securely using their email and password. | M |
| UR2 |  |  |
| UR3 |  |  |

* 1. **Functional Requirements**

< Detail the specific behaviors, functions, and capabilities that the application must provide. These requirements are derived from user and business requirements and outline how the application will fulfill those needs. List at least 3 functional requirements with MOSCOW code.>

| Requirement ID | Requirement Description | MOSCOW |
| --- | --- | --- |
| FR1 | e.g., The application shall provide user authentication mechanisms to ensure secure access. | M |
| FR2 |  |  |
| FR3 |  |  |

* 1. **Non-Functional Requirements**

< Describe the attributes, qualities, and constraints of the application that are not directly related to specific functionalities. These requirements focus on how the system performs its tasks rather than what tasks it performs. They can include aspects such as performance, usability, reliability, security, and scalability. List at least 3 non-functional requirements with MOSCOW code.>

| Requirement ID | Requirement Description | MOSCOW |
| --- | --- | --- |
| NFR1 | e.g., The application shall provide a user-friendly interface that allows users to complete tasks without prior training. | M |
| NFR2 |  |  |

* 1. **Implementation (Performance) Requirements (Optional)**

< Outline the technical aspects and constraints that need to be considered during the development of the application. These requirements typically include details about the development environment, programming languages, software tools, frameworks, hardware specifications, and deployment processes.>

| Requirement ID | Requirement Description | MOSCOW |
| --- | --- | --- |
| IR1 | The application should be developed using Python or JavaScript. | S |
| IR2 | The project should use Oracle as the primary database management system. | S |
| IR3 | The should be developed using Visual Studio Code and Github. | M |