**PROJECT MANAGEMENT SYSTEM**

**SRS**

BY

V.AKASH

M.RAKSHAN

P.S.LOGESH

P.NARESH

S.JANA KRISHNA

**Table of content**

1.Introduction

2 .Functional Requirements

2.1 Project Creation and Management

2.2 Resource Management

2.3 Collaboration and Communication

2.4 Reporting and Analytics

2.5 Security and Privacy

3.Non-Functional Requirements

3.1 Usability and User Experience

3.2 Performance and Scalability

3.3 Compatibility and Integration

3.4 Maintenance and Support

4.Conclusion

5.Diagram

5.1 usecase diagram

5.2 Class diagram

5.3 State diagram

5.4 Data flow diagram

1.Introduction

The purpose of this Software Requirements Specification (SRS) document is to outline the requirements for a project management software. The software will be used to manage projects, tasks, resources, and timelines for businesses and organizations. The target audience for this software includes project managers, team members, and stakeholders.

2 .Functional Requirements

The following are the functional requirements of the project management software:

2.1 Project Creation and Management

- The software should allow users to create and manage projects

- Each project should have a unique name and description

- The software should allow users to add tasks to projects

- The software should allow users to assign tasks to team members

- The software should allow users to set start and end dates for projects and tasks

- The software should allow users to set project budgets and track expenses

2.2 Resource Management

- The software should allow users to manage resources such as team members, equipment, and materials

- The software should allow users to assign resources to tasks

- The software should allow users to view the availability of resources

2.3 Collaboration and Communication

- The software should allow team members to communicate and collaborate on tasks

- The software should allow users to share files and documents related to the project

- The software should provide a messaging system to communicate with team members and stakeholders

2.4 Reporting and Analytics

- The software should provide reporting and analytics features to track project progress and performance

- The software should allow users to view reports on project status, budget, and timeline

- The software should provide visualizations such as charts and graphs to display data

2.5 Security and Privacy

- The software should have measures in place to ensure the security and privacy of user data

- The software should require user authentication to access the system

- The software should encrypt sensitive user data such as passwords and financial information

- The software should have backup and disaster recovery mechanisms in place to protect against data loss

3.Non-Functional Requirements

The following are the non-functional requirements of the project management software:

3.1 Usability and User Experience

- The software should have a user-friendly interface

- The software should be intuitive and easy to use

- The software should be accessible to users with disabilities

3.2 Performance and Scalability

- The software should be responsive and perform well under heavy loads

- The software should be scalable to accommodate increasing numbers of users and projects

3.3 Compatibility and Integration

- The software should be compatible with various operating systems and devices

- The software should integrate with other software and tools used by the organization

3.4 Maintenance and Support

- The software should be maintainable and easy to update

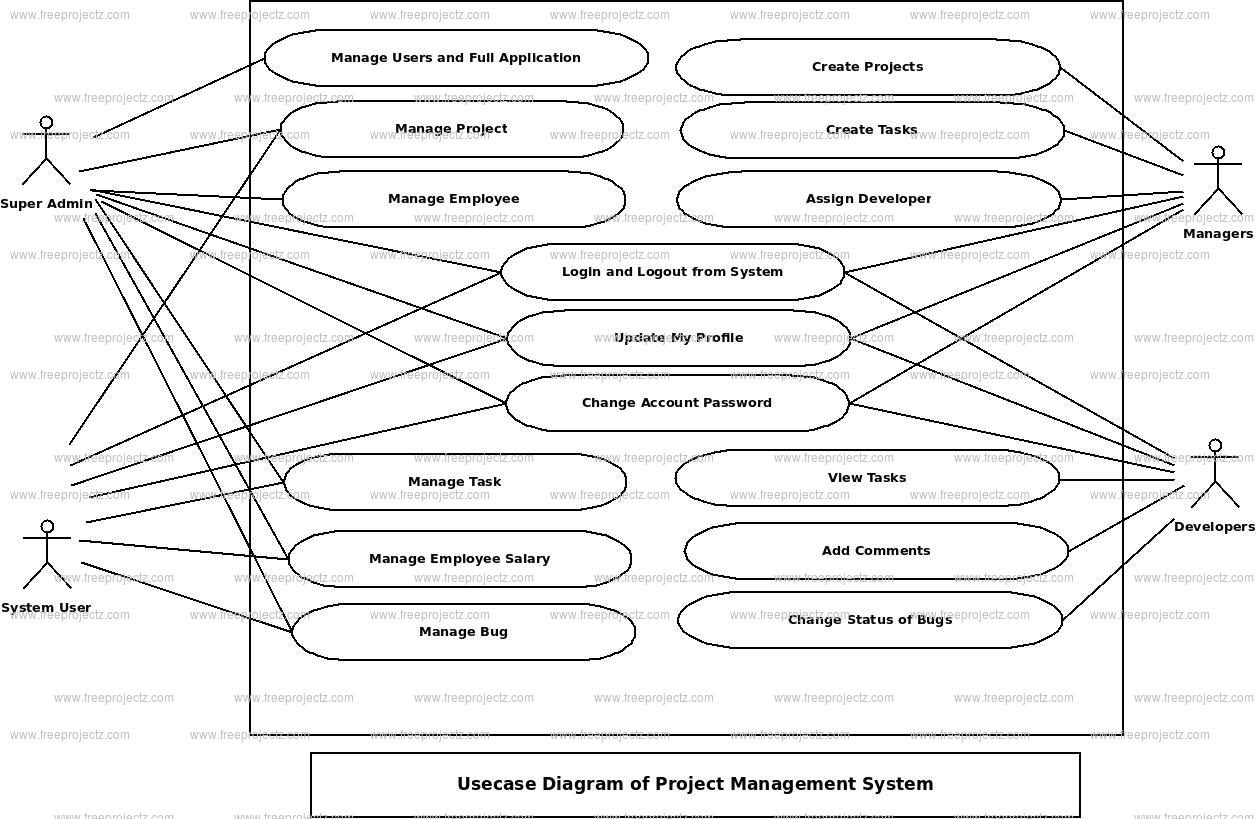
- The software should have a support system in place to assist users with issues and concerns

4.Conclusion

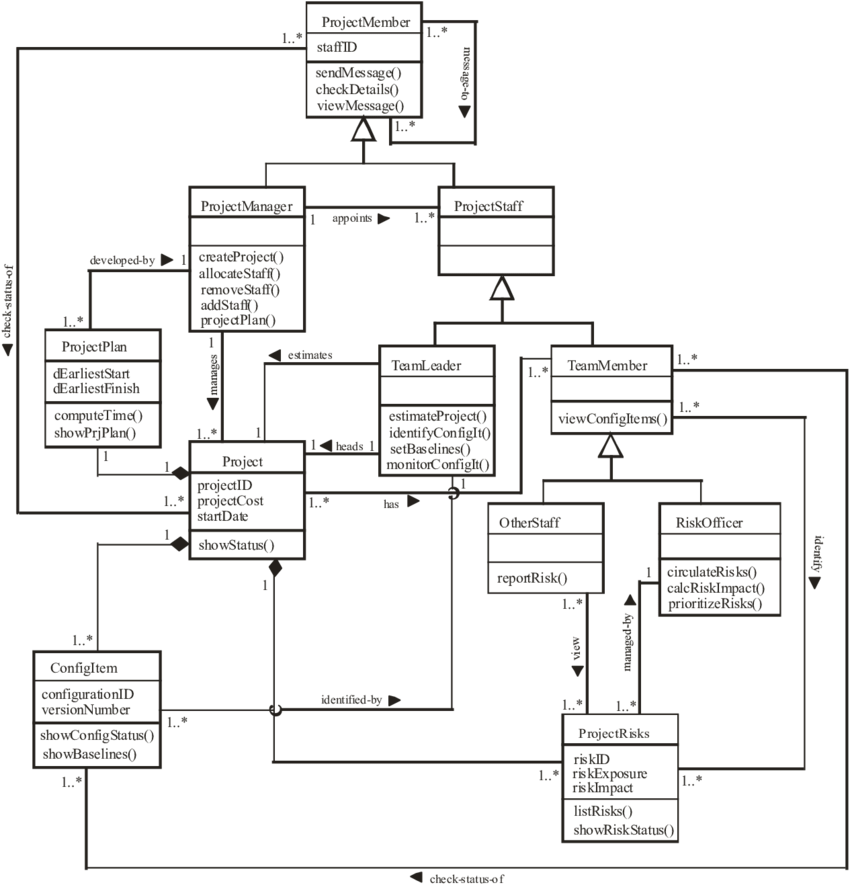
This Software Requirements Specification document outlines the requirements for a project management software. The software should provide functionality for project creation and management, resource management, collaboration and communication, reporting and analytics, security and privacy, as well as meet non-functional requirements for usability, performance, compatibility, and maintenance and support.

5.Diagram

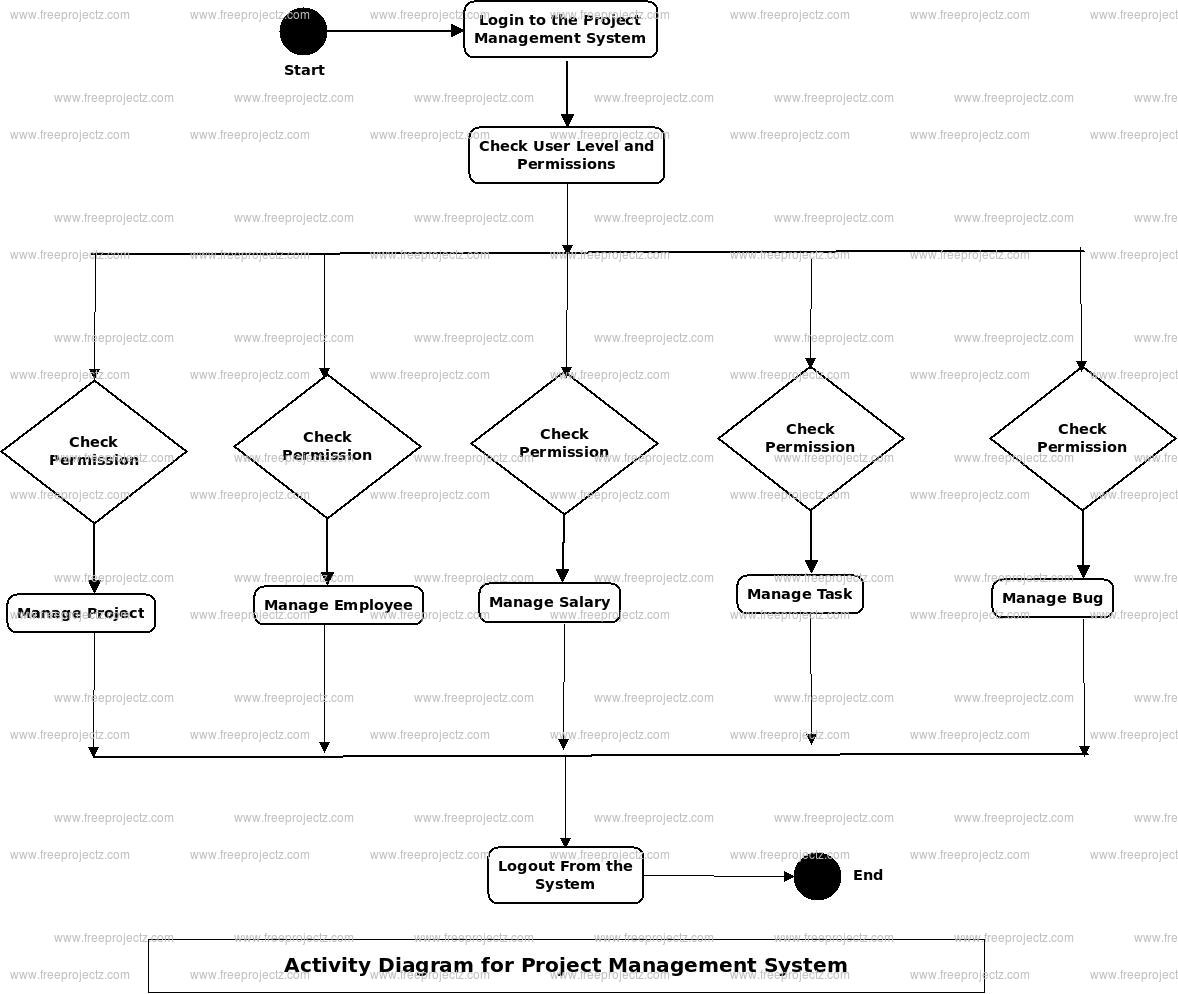
5.1 usecase diagram



5.2 Class diagram



5.3 State diagram



5.4 Data flow diagram

