Software Requirements Specification (SRS) Document

Productivity Management Tool

Team 40
Harshita Upadhyay
Prateek Sancheti
Prince Varshney
Snehal Ranjan

Brief problem statement:

The product is related to managing employees and progress related to software projects. Based on the requirements of the project, employees with relevant skillset are shortlisted from a directory of available employees. The actual hours committed to the project will be compared against the original estimate to track its progress.

System requirements:

The project will be a web-based application with separate client and server modules. The client would run in a web browser and would require a system capable of doing so.
The server module would require a machine that can serve requests from the client based on the number of simultaneous users of the system.

Based on current understanding, MERN stack seems to be a suitable choice for this project.

User's profile:

- Manager: Need a UI that allows smooth management of projects with minimal friction.
- Employee: Require simple setup to confirm/reject project allocations and submit work hours.
- Head of Department: Simple dashboard to approve/deny projects and track their progress.

All the users are directly or indirectly involved in software development so are considered technically competent to use a standard User Interface.

Feature Requirements (described using use cases)

The release schedule of the features will be specified during the release planning phase.

No.	Use Case Name	Description	Release
1.	Registration	Registration of new users, based on their types, asks for relevant data.	R1
2.	Login	Common login portal for the users	R1
3.	Project Creation	Creation of new project by Manager	R1
4.	Employee Search	Based on requirements of the project, search for employees with relevant skills and availability.	R1
5.	Employee Allocation	Manager can request allocation of an employee to a project.	R1
6.	Allocation Confirmation	Employees must confirm their allocation.	R1
7.	Project Confirmation	After all employees have been confirmed the project itself must be approved by the Head of Department.	R1
8.	Commit Hours	Employees must submit the hours committed to the project on a weekly/daily basis.	R1
9.	Hours Confirmation	The work hours of the employees must be confirmed by the Manager/Head.	R1
10.	Project Overview	A dashboard for Manager/Head that provides information about the expected and actual progress	R1
11.	Allocation Modification	Based on the progress of the project, employees can be added or removed	R1

Use case diagram:

The Use Case Diagram is available as a separate document

Use case description:

Use Case Number:	UC-01
Use Case Name:	Registration
Overview:	To access any of the functionalities of the system, a user must register as one of the three user types
Actors:	User
Pre condition:	No active session that is already logged in
Flow:	Main (Success) Flow: 1. The user is requested for their user type 2. Based on the type, relevant fields are presented 3. The user fills the fields 4. The fields are validated 5. The information is sent to the backend 6. It is checked to avoid duplication of key attributes 7. User is registered
	Alternate Flows: Validation Failed: 1. The user is requested for their user type 2. Based on the type, relevant fields are presented 3. The user fills the fields 4. The fields are validated 5. Validation fails Post Condition: The user is requested to check the fields with a message about validation violation Unique Test Failed: 1. The user is requested for their user type 2. Based on the type, relevant fields are presented 3. The user fills the fields 4. The fields are validated 5. The information is sent to the backend 6. It is checked to avoid duplication of key attributes 7. Duplicate Found in value of attributes like email Post Condition: The user is informed about the existence of the duplicate
Post Condition:	A new user is created

Use Case Number:	UC-02
Use Case Name:	Login
Overview:	A common login portal for all the different types of users. Will authenticate the user and take them to their relevant homepage
Actors:	User
Pre condition:	No active session that is already logged in
Flow:	Main (success) Flow: 1. User enters their email address and password 2. The frontend sends an authenticate request to the backend 3. The backend checks user's credentials 4. Backend sends confirmation of authentication and a session token 5. The user is granted access to the system and taken to their respective homepage
	Alternate Flow: Login Failed: 1. User enters their email address and password 2. The frontend sends an authenticate request to the backend 3. The backend checks user's credentials 4. The backend sends an authentication failed response 5. The frontend asks user to check their credentials Post condition: The user is still at the login page
Post Condition:	The user ends up at a homepage depending on their type with a session token that will be used to authenticate their future requests

Use Case Number:	UC-03
Use Case Name:	Project Creation
Overview:	Creation of a new project by providing relevant requirements like skills required, man-hours etc.
Actors:	Users of type Manager
Pre condition:	Users must be of type Manager and must be logged in.
Flow:	1. Manager fills required fields for project creation. 2. Fields are validated 3. The information is sent to the backend 4. It is checked to avoid duplication of key attributes
	Alternate Flows:

	Validation Failed 1. Manager fills required fields for project creation. 2. Fields are validated 3. Validations fails Post Condition: The user is requested to check the fields with a message about validation violation
Post Condition:	Project is created and an appropriate message is displayed.

Use Case Number:	UC-04
Use Case Name:	Employee Search
Overview:	When creating a project, manager can add employees by searching based on skills required and selecting a subset from the results
Actors:	Manager, Employee
Pre condition:	User must be manager, User must be logged in, User is adding employees to a project
Flow:	Main (success) Flow: 1. User enters a list of skills required 2. The frontend sends a request with requirement parameters 3. The backend searches through a database of employees and filters them on the basis of parameters. 4. The results are sorted on the grounds of availability 5. Backend sends the compiled results to the frontend 6. The frontend renders the results received from the backend
	Alternate Flow: No results found: 1. User enters a list of skills required 2. The frontend sends a request with requirement parameters 3. The backend doesn't find any result that match the parameters 4. A response with this information is sent to the frontend 5. The user is asked to check the query Post Condition: Since there are no results, the user is presented with an empty list and an info message
Post Condition:	The user is presented with a list of employees to choose from based on the query

Use Case Number:	UC-05
Use Case Name:	Employee Allocation
Overview:	Manager can request allocation of an employee to a project.
Actors:	Manager
Pre condition:	The person should be registered in the system as one of the employees in order to be allocated with the project.
Flow:	Main (success) Flow: 1. Manager selects employees to be allocated from the search results 2. The backend receives the list of selected employees 3. Confirmation Requests are sent to everyone in the list
	Alternate Flow: None
Post Condition:	The user will be allocated the most suitable project according to his/her skills.

Use Case Number:	UC-06
Use Case Name:	Allocation Confirmation
Overview:	Employees must confirm their allocation.
Actors:	Users of type Employee
Pre condition:	Users must be of type Employee and must be logged in.
Flow:	1) User opens the dashboard to view all requested allocations. 2) Users can then accept/reject the request.
Post Condition:	If the User accepts, he/she is allocated to the project.

Use Case Number:	UC-07
Use Case Name:	Project Confirmation
Overview:	After a manager is done setting up a project, they must request the Head of Department (HoD) to review and confirm the project

Actors:	Head of Department, Manager
Pre condition:	User must be a Head of Department, User must be logged in, There must be projects in review stage
Flow:	Main (success) Flow: 1. HoD selects one of the projects from the “review requests” section of the dashboard 2. HoD reviews the details of the project 3. The project is approved
	Alternate Flows: Review Failed: 1. HoD selects one of the projects from the “review requests” section of the dashboard 2. HoD reviews the details of the project 3. They are not satisfied with something about the project 4. The project is rejected with some comments which must be addressed before requesting another review Post Condition: The state of the projects remains pending approval
Post Condition:	The state of a project is changed to approved in the database

Use Case Number:	UC-08
Use Case Name:	Commit Hours
Overview:	Employees must submit the hours committed to the project on a weekly/daily basis.
Actors:	Employees
Pre condition:	The employee must be allocated to the specific project
Flow:	Main (success) Flow: 1. Employee will select whether the committed hours are on the daily basis or weekly basis and then enter the number of committed hours 2. The entered data is verified as if the entered hours are numeric or not. 3. If the entered data is fine, the frontend send the request to save the data to backend 4. Backend saves the data in the database 5. Backend shows the success message
	Alternate Flows: 1. Employee will select whether the committed hours are on the daily basis or weekly basis and then enter the number of committed hours 2. The entered data is verified as if the entered hours are numeric or not.

	3. If the entered data is not fine, the frontend shows the error message and asks to enter valid input.
Post Condition:	The number of committed hours gets saved in the database.

Use Case Number:	UC-09
Use Case Name:	Hours Confirmation
Overview:	The work hours of the employees must be confirmed by the Manager
Actors:	Users of type Manager
Pre condition:	Users must be of type Manager and must be logged in.
Flow:	1. Open the project 2. For each employee confirm the weekly hours committed.
Post Condition:	Weekly progress is analyzed.

Use Case Number:	UC-10
Use Case Name:	Project Overview
Overview:	A dashboard for Manager/Head that provides information about the expected and actual progress
Actors:	Manager, Head of Department
Pre condition:	1. The user must be logged into the system. 2. There should be one or more than one projects in the approved state in order to view.
Flow:	Main (success) Flow: 1. The user selects any one project from the list of all the projects associated with him/her. 2. The frontend sends a request to view the selected project 3. The backend retrieves all the necessary information about the project from the database. 4. The backend sends the response to the frontend.
Post Condition:	The user will be directed to the dashboard of the selected project.

Use Case Number:	UC-11
Use Case Name:	Allocation Modification
Overview:	Based on the progress of the project employees can be added or removed
Actors:	User of Type Manager
Pre condition:	Users must be of type Manager and must be logged in.
Flow:	1. Find an employee to remove 2. Go to Employee Search use case for adding new employee
Post Condition:	Modifications made to the project must be approved.