

[**TASK MANAGEMENT SYSTEM - “ WORKSPACE”**](#_pl97haqn1xk) **3**

[**1. Introduction**](#_au2mk9gflrea) **3**

[1.1 Purpose](#_q44h3f7o1h5t) 3

[1.2 Scope of the Project](#_nc6et9c06eqx) 3

[1.3 Process Model](#_95cd9mbf34dl) 4

[1.4 Intended Audience](#_y30ifahanmnd) 4

[1.5 Definitions, Acronyms, and Abbreviations](#_3sdjn8m3p07) 5

[**2. Overall Description**](#_24sois2bhpr5) **6**

[2.1 Product Perspective](#_3q6axwreh8rg) 6

[2.2 Product Functions](#_doz8j9ctojgc) 6

[2.3 User Characteristics](#_mtj33pfs032a) 7

[2.4 Operating Environment](#_k1akgsiet12p) 8

[2.5 Design and Implementation constraints](#_a307c7doqy9f) 8

[2.6 Assumptions and Dependencies](#_eq5tf7i1v6e4) 9

[**3. Specific Requirements**](#_cv3yno61rguu) **9**

[3.1 Functional Requirements](#_vrzdrvekmrw3) 9

[3.1.1 User Class 1 - The Admin](#_lu0zs64kvcy4) 9

[3.1.1 User Class 2 - The Employee](#_yr1jfuxj9gew) 9

[3.2 External User Requirements](#_6xnlfamrhvpv) 9

[3.2.1 Software Interface](#_km52wq6kqgxi) 10

[3.2.2 Hardware Interface](#_1tgejadjbhse) 10

[3.2.3 Communication Interface](#_nbi6atuxk30v) 10

[3.3 Non - Functional Requirements](#_fsex5sn6w1x4) 10

[3.3.1 Security](#_q8y7vctspkg) 10

[3.3.2 Maintainability](#_rp8lcuo19muu) 11

[3.3.2 Portability](#_93kuo7s95ig9) 11

[3.3.3 Performance](#_vm8zn5uy2oa0) 11

[3.3.4 Availability](#_84tjhkd6acoe) 12

[**5. USE CASE DIAGRAM**](#_ujgtupjdqgnr) **12**

[**6. ER DIAGRAM**](#_g6aas51oaffi) **13**

[6.1 Logical ERD](#_r59a4d7s9zit) 13

[6.2 physical ERD](#_58ovy22i7rtd) 13

[**7. DATA FLOW DIAGRAM**](#_i2yjgu2lehnp) **13**

[7.1 Level 0 - Context Level Diagram](#_ezan6j5ahi8r) 13

[7.2 Level 1](#_r8q9fgiw0uzz) 13

[7.3 Level 2](#_k3l3lkskptkr) 13

[**8. State Diagram**](#_jeuy6mxqb01u) **13**

[**9. Activity Diagram**](#_qsqmkspyej86) **13**

[10. Summary](#_5sls4o6vumkv) **13**

# TASK MANAGEMENT SYSTEM - “ WORKSPACE”

# **Introduction**

This section gives a scope description, requirements, and overview of the Scope Requirement Specification (SRS) document.

## **1.1 Purpose**

The Software Requirements Specification (SRS) will provide a detailed description of the requirements for the Task Management System. This SRS will allow for a complete understanding of what is to be expected from the system which is to be designed and developed. A clear understanding of the system and its functionality will allow for the correct software to be developed for the end user and will be used for the development of the future stages of the project. It is primarily intended to be proposed to a customer for its approval and provide a foundation for developing the first version of the system for the development team. It provides a common reference point for both the developer team and the stakeholder community. The SRS will evolve over time as users and developers work together to validate, clarify and expand its contents.

## **1.2 Scope of the Project**

The Task management system “WORKSPACE” is a web application that helps students and employees to manage their daily work and teamwork in a more organized and simpler way. This application should be accessible to any browser on the desktop. This software includes important features like authentication to the portal, authorization to the data as per requirement in the teams or organization, note taking (in collaboration also), calendar updates after events/ tasks are managed, team rooms, chat rooms, and task list boards in each team room.

The objective of the collaborative Task Management System is to manage tasks smoothly from one platform instead of using a collection of tools and services.

It would be a one-stop platform for task and team organization with ease, that includes various management tools like event management, managing teams, team notes, task assignments, and more. This would ease the process of sharing and maintaining tasks efficiently. The system should be user appropriate, easy to use, provide easy recovery of errors, and have an overall end-use higher satisfaction.

## **1.3 Process Model**

Understanding the work scope of the project, the time interval for project work, and team requirements, an interactive as well as an incremental process model like the agile model can be implemented for the processes.

* **Agile model** - Agile model encourages continuous iteration on the development of the small incremental parts of the software and keeps the door open for changes if requirements we can incorporate changes up to some extent as per user requirement.
* Agile Model is a very advanced model [does not follow a step-by-step process unlike other models] and supports parallel modular processes at a time which could be helpful to us in developing independent features at a time.
* **Methodology of the agile model: Kanban + Scrum model** - Using the scrum model we can divide our work into group members for a week or two with proper planning (clear representation of all project activities and their progress). This will help us achieve the project deadline in the span of 3-3.5 months. Boards can be prepared for every sprint similar to the Kanban model to keep a track of the work accomplished, improvements required, efficiency, and backlogs.

## **1.4 Intended Audience**

This project is intended to be used by a particular organization - employees, managers or students, faculty, and so on. This SRS document is intended for several audiences including the customer, as well as the project manager, designers, developer, and tester.

● The customer will use this SRS to verify that the developer team has created a product that is acceptable to the customer.

● The project manager of the developer team will use this SRS to plan milestones and delivery dates and ensure that the developing team is on track during the development of the system.

● The designer will use this SRS as a basis for creating the system’s design. The designer will continually refer back to this SRS to ensure that the system they are designing will fulfill the customer’s needs.

● The developer will use this SRS as a basis for developing the system’s functionality. The developer will link the requirements defined in this SRS to the software they create to ensure that they have created software that will fulfill all of the customer’s documented requirements.

● The tester will use this SRS to derive test plans and test cases for each documented requirement. When portions of the software are complete, the tester will run his tests on that software to ensure that the software fulfills the requirements documented in this SRS. The tester will again run his tests on the entire system when it is complete and ensure that all requirements documented in this SRS have been fulfilled.

## **1.5 Definitions, Acronyms, and Abbreviations**

| **TERM** | **DEFINITION** |
| --- | --- |
| User | Someone who interacts with the web application |
| Team Lead/Manager | Someone who is given specific permission for managing the tasks, and assigns the task to the team members. The team Lead or Manager has this additional functionality along with what an individual user has. |
| Stakeholder | Any person who has interaction with the system who is not a developer |
| SRS | Software Requirement Specification |
| SQL | Structured Query Language |
| DEP | Dependency |
| DESC | Description |
| RAT | Rational |
| TAG | A unique, persistent identifier |
| GIST | A short, simple description of the concept |
| SCALE | The scale of measure used by the requirement |
| METER | The process or device used to establish a location on a SCALE |
| MUST | The minimum level required to avoid failure |
| PLAN | The level at which good success can be claimed |
| WISH | A desirable level of achievement that may not be attainable through current approaches |
| DEFINED | The official definition of a term |

# **Overall Description**

## **2.1 Product Perspective**

The Task management system is a new self-contained software product that will be produced by the project team. A newly introduced system will provide easy accessibility and will have user-friendly features and interfaces. This system will handle all the information related to managing tasks in collaboration and for individuals as well. The platform will provide all the necessary information and will help in preventing miscommunication and missing out on any tasks. The system consists of web applications with different interfaces to facilitate the required functionalities for managing tasks and teams. The platform is a data-centric platform that requires a huge amount of memory to store the data For that a database has to be linked with the platform, that can easily communicate to the portal and help in managing (adding, deleting, updating) the data easily.

## **2.2 Product Functions**

The main aim of the product is to provide a one-stop platform for managing tasks as effectively as possible.

The platform will have an authentication module for signup and login functionality with basic details like name/user id, email, and password.

The user will get a dashboard will all the work and scheduled deadlines and meetings.

The user will have a calendar view as well to save time switching to the calendar before planning any new meeting/work.

The user can categorize their work into personal, group, and others as per requirements.

Each project will have project users and will create a space for these users to share work details, and documents and schedule meetings.

While scheduling any of the work, the user can add another user as well and assign them the work. This can help project admins to well plan the work to meet deadlines.

## **2.3 User Characteristics**

The people will interact with the system with the following goals:

1. Effectively manage personal tasks - with deadlines, related documents, and scheduled meetings.
2. Effectively communicate with the team for group/team tasks - define and assign tasks, add relevant documents, schedule meetings, and define tasks.

* Under a team project, there would be an admin to manage tasks effectively.

**Personal Task Management**

Personal task management includes users who wish to manage their personal tasks in an effective manner. They can add tasks with deadlines, document, assign priority and track them on the dashboard which helps the user to avoid missing any of the tasks or scheduled meetings.

**Team Task Management**

Team task management includes users who wish to effectively manage and communicate as a team on a particular project. They can add users to the team, they can add tasks with deadlines, priorities, descriptions, and documents. Also, the tasks can be assigned to the particular user so they can see that on their dashboard as a priority.

## **2.4 Operating Environment**

This is a web-based system and hence will require the operating environment for a client and server GUI.

**Software**

● Website is designed to run on any platform which supports web browsers like Google Chrome and Safari.

● A running XAMPP server if wanted to host it on localhost.

● Database is installed on PHPMyAdmin, therefore, requires a server to host it.

**Hardware**

● Operating System Supports all known operating systems, such as Windows, Linux, or Mac.

● Computer 512MB+ RAM, monitor with a minimum resolution of 1024x768, keyboard, and mouse.

## **2.5 Design and Implementation constraints**

The storage and cross-connections among tasks and projects are the constraints for the application. Since the software holds large data from the database, it is crucial that there is an appropriate database designed for the software to function.

Decisions regarding which database to use should be taken considering the fact that the data being exchanged or stored is large, and the appropriate data management system will yield efficient performance.

This system has a language constraint and is intended to be executed in the ‘English’ language. Hence, it does not provide multilingual support to users.

## **2.6 Assumptions and Dependencies**

Assumptions about the product are that it would be accessed by an internet browser on any device. The dependency is that administrators' details would be verified and filled in beforehand in order to authorize the web portal. Users should know the basics about accessing the system. Users and Admins will only be able to access the system by the registered email id to make sure the person is authorized and an eligible user that is part of the organization. In the case of database and information storage, it depends on SQL servers for information management of the system. It is also assumed that the client could change their decision on the next phases of software development. According to agile software engineering methodologies, clients and developers have the flexibility to incorporate changes in the system design and implementation as and when required.

Another technical assumption is that the admin will first authorize the project/task/event/user information from the head of the department and then update it on the web portal for the employed to access. This requirement of validating the work and the content can be an additional feature for future versions of the software.

# **3. Specific Requirements**

## **3.1 Business Requirements**

**ID: BR1**

**TITLE: Ease of use**

**DESC:** The web portal should be easy to use, with easy navigation, understandable design, could handle the load and fast to perform tasks.

**ID: BR2**

**TITLE: Data Security**

**DESC:** The web portal should ensure organization data is secure.

**ID: BR3**

**TITLE: Customizable**

**DESC:** The system should be easily customizable as per requirement.

**ID: BR4**

**TITLE: Easy system integration**

**DESC:** The system should be easy to integrate with other substitutes or the main system of the organization.

## **3.2 Functional Requirements**

**ID: FR1**

**TITLE: View Web Portal**

**DESC:** A user should be able to view the web portal through any browser on a mobile phone, laptop, and desktop.

## **Login Module**

**ID: FR2**

**TITLE: Login**

**DESC:** A user and admin should be able to login into the system by signing in with his/her registered Email id.

**ID: FR3**

**TITLE: Store Password in encrypted mode**

**DESC:** The password is stored in the encrypted mode for each user in the system to ensure the safety and security of the software.

**ID: FR4**

**TITLE: Logout**

**DESC:** A user and admin should be able to log out from the portal.

## **User Interaction Module**

**ID: FR5**

**TITLE: Add New Project**

**DESC:** An admin user can edit, update and add new projects. Details required for new projects are name, the status of the project, start date, end date, project manager, project team members, description, resources, and tasks.

**ID: FR6**

**TITLE: Add contribution details to the Project / Add comment for the team**

**DESC:** An user, as well as an admin, can add their progress/productivity update on the project tasks. This feature can be used to add comments and notes on the same.

**ID: FR7**

**TITLE: Edit contribution details to the Project / Edit comment for the team**

**DESC:** An user, as well as an admin, can edit their already posted progress/productivity update on the project tasks. This feature can be used to add comments and notes on the same.

## **Admin Interaction Module**

**ID: FR8**

**TITLE: Update Project Information**

**DESC:** The project manager or administrator should be allowed to edit the project information like project description, start-end date, resources, tasks, action details, etc.

**ID: FR9**

**TITLE: Delete Project**

**DESC:** An admin should be able to delete the project, Delete the project should remove the project from the user list as well.

**ID: FR10**

**TITLE: Add New Project Task Description**

**DESC:** An admin should be able to add a project task description along with the project task name and status.

**ID: FR11**

**TITLE: Edit Project Task Description**

**DESC:** An admin should be able to edit the project task description along with the project task name and status.

**ID: FR12**

**TITLE: Delete Project Task**

**DESC:** An admin should be able to delete the project tasks.

**ID: FR13**

**TITLE: Change the status of the task**

**DESC:** An admin should be able to change the status of a task by editing the project task details.

**ID: FR14**

**TITLE: Assign Task**

**DESC:** An admin/project manager should be able to assign tasks to different users.

**ID: FR15**

**TITLE: Add New User**

**DESC:** An admin should be able to add a new userby adding details like the first name, last name, email, and user role, creating and confirming the password, and adding an avatar.

**ID: FR16**

**TITLE: View Users List**

**DESC:** The admin can see the list of all the registered users.

**ID: FR17**

**TITLE: Edit Users Details**

**DESC:** This should include editing/updating details like the first name, last name, email, user role, changing password, and avatar.

**ID: FR18**

**TITLE: Add Event**

**DESC:** An admin should be able to add events on a particular date by referring to the calendar.

**ID: FR19**

**TITLE: View Event**

**DESC:** An admin should be able to view the added events on a particular date in the calendar.

**ID: FR20**

**TITLE: Delete Event**

**DESC:** An admin should be able to delete the added events on a particular date from the calendar.

**ID: FR21**

**TITLE: Download Report**

**DESC:** The admin can download the project's progress report from the portal which can be further shared if required.

**ID: FR22**

**TITLE: Search Project or task**

**DESC:** The admin or user should be able to search project and task.

**ID: FR23**

**TITLE: Profile Update**

**DESC:** The admin or user should be able to update his/her profile.

## **View Module**

**ID: FR24**

**TITLE: View User Dashboard**

**DESC:** A user can see the project progress, complete tasks, and complete tasks with relevant details.

**ID: FR25**

**TITLE: View all Project**

**DESC:** A user can view the list of enrolled projects and their details.

**ID: FR26**

**TITLE: View Project Description**

**DESC:** This allows the user to view all the relevant details like project name, description, project manager, status, start-end date, task list, team members, and their contribution updated by the project manager or team members.

**ID: FR27**

**TITLE: View Project Task Description**

**DESC:** This allows the user to view all the relevant details related to the project task like task , task description, status, and specific actions to view, update and delete tasks.

**ID: FR28**

**TITLE: View Project Progress**

**DESC:** A user should be able to view the project progress whether it is pending, on hold, or done from the status option.

**ID: FR29**

**TITLE: View all the Tasks**

**DESC:** A user should be able to view all the tasks of a project along with the project starting name date, due date, project status, and task status from the task option in the navbar.

**ID: FR30**

**TITLE: View Task Description**

**DESC:** A user should be able to view the task description from the action option.

**ID: FR31**

**TITLE: View Events**

**DESC:** A user should be able to view the events.

## **Account Module**

**ID: FR32**

**TITLE: Change Password**

**DESC:** This allows users and admins to change their existing passwords and update their credentials to log in to the web portal.

## 

## **3.3 External User Requirements**

This section provides a detailed description of all the inputs and outputs from the system. It also gives a description of the software communication interfaces and provides a basic prototype of the user interface. The communication required between the database and the web portal will be required in form of reading, adding, deleting, and modifying the data.

## **3.3.1 Software Interface**

The software interface should be easy to use, fast, easy to understand, and with an intuitive navigation path for a better user experience of working on the web portal.

The screen design includes the following tasks: Login Page, Dashboard, Add Project Page, View Project, Edit Project, Add Task, View Task, Edit task (status changing - for progress tracking), Add new user, Edit user, Manage events, Manage Notes, Manage Profile, User Section - Add Project/Task progress notes.

## **3.3.2 Hardware Interface**

The current web portal design doesn’t require any hardware interfaces. The internet connection required for accessing and managing the database and web portal can be managed using wif and the underlying operating system.

## **3.3.3 Communication Interface**

The communication between the different parts of the system like database and web server, user and web server, etc is an important part of the system as they depend on each other for functionalities to work efficiently.

## **3.3 Non - Functional Requirements**

## **3.3.1 Security**

**ID: NR1**

**TITLE: Security of accounts for admin login**

**DESC:** If an admin tries to log in to the web portal with a not defined account then the admin should not be logged in and the admin should be notified with an appropriate error message about login failure.

**MUST:** 100% of the time.

**ID: NR2**

**TITLE: Security of accounts for admin login**

**DESC:** If a user tries to log in to the web portal with a not defined account then the user should not be logged in and the user should be notified with an appropriate error message about login failure.

**MUST:** 100% of the time.

**ID: NR3**

**TITLE: Security of data**

**DESC:** The data regarding company projects and work progress should be stored with utmost security measures.

**MUST:** 100% of the time.

## **3.3.2 Maintainability**

**ID: NR4**

**TITLE: Software extensibility**

**DESC:** The application should be easy to extend. The code should be written in a way that it can easily incorporate future required functionalities.

**RAT:** To easily implement future requirements.

**ID: NR5**

**TITLE: Application testability**

**DESC:** Test environments should be built for the application to allow testing of the applications' different functions.

**RAT:** In order to test the application.

## **3.3.2 Portability**

**ID: NR6**

**TITLE: Application portability**

**DESC:** If hosted, the application should be able to run in any browser.

**RAT:** The adaptable platform for the application to run on.

## **3.3.3 Performance**

**ID: NR7**

**TITLE: Response time**

**SCALE:** The response time of the different outputs, processed transactions, and screen refresh for the system.

**MUST:** No more than 5 seconds 100% of the time.

**WISH:** No more than 2 seconds 100% of the time.

**ID: NR8**

**TAG: System Dependability**

**SCALE:** If the system gets some strange input or loses the connection to the Internet, the user should be informed and the system should have a high tolerance to the faults.

**MUST:** 100% of the time.

## **3.3.4 Availability**

**ID: NR9**

**TITLE: System Availability**

**DESC:**  Accounting the availability of the system of the times it is used. The average system availability without taking into consideration the network failure.

**MUST:** More than 98% of the time.

**WISH:** 100% of the time

**ID: NR10**

**TITLE: Internet Connection Requirement** - Xampp works without an internet connection

**DESC:** The web application should be connected to the Internet while using it, in order to build a connection between the database and the web portal.

**ID: NR11**

**TITLE: Ease of use**

**DESC:** The required training time and the number of help frames should be as minimum as possible.

**ID: NR12**

**TITLE: Robustness and Reliability**

**DESC:** The mean time to failure and probability of unavailability, rate of failure occurrence and availability.

# **5. USE CASE DIAGRAM**

# **6. ER DIAGRAM**

## **6.1 Logical ERD**

## **6.2 physical ERD**

# **7. DATA FLOW DIAGRAM**

## **7.1 Level 0 - Context Level Diagram**

## **7.2 Level 1**

## **7.3 Level 2**

# **8. State Diagram**

# **9. Activity Diagram**

User and admin login

Dashboard

Projects and tasks

Add project - update and delete

Add task update and delete

Add event update and delete

Notes/Resources

User add productivity comment.

# **10. Summary**

The SRS provides an overall description and an overview of the system functionality and system interaction with other systems. It also describes the probable user interfaces for a better understanding of the appearance of the system to the end users and stakeholders. It also provides the requirements specification in detailed terms and a description of the different system interfaces. Different specification techniques are used in order to specify the requirements more precisely for different audiences. It also depicts the system with user case scenario - use case diagrams, shows how data in the system will flow with - data flow diagram, changing of states due to action of events - using state diagram, the relationship among entities of the system for the database - Entity Relationship diagram and Flow of activities using- Activity diagram. The SRS document helps the developer to gain a better understanding of the system and a strong reference point for the development and implementation stage of the project.