

# **SOFTWARE REQUIREMENTS SPECIFICATION**

***ProCollab- A Project Collaboration Management System***

**Prepared by**

Omar Sadat - 2221335042

Md. Alif Bin Turjo - 2221876642

Quazi Sakib Ahmed - 2131936042

Hossain Ahammed - 2222990042

Mahfuzur Rahman - 2221827042

*North South University  
Software Engineering (CSE - 327 Section 4)*

# **CONTENTS**

## **1 Introduction**

1.1 Purpose .....	3
1.2 Intended Audience .....	3
1.3 Intended Use .....	4
1.4 Product Scope .....	7
1.5 Risk Definition .....	8

## **2 Overall Description**

2.1 User Classes and Characteristics .....	10
2.2 User Needs .....	12
2.3 Operating Environment .....	14
2.4 Constraints .....	14
2.5 Assumptions .....	16

## **3 Requirements**

3.1 Functional Requirements .....	18
3.2 Non Functional Requirements .....	29

# **CHAPTER 1 - INTRODUCTION**

## **1.1 PURPOSE**

The Project Collaboration and Task Management System, version 1.0, software requirements are described in this document. By providing a structured platform for user authentication, project ownership, task assignment, and team collaboration, the system is intended to make project coordination safe and effective. Project dashboards, task delegation, email-verified account creation, and real-time progress tracking are some of the main features. Through validated, controlled access, this system aims to give people and groups a dependable setting in which they can oversee tasks, assign tasks, manage projects, and collaborate. Data integrity and user accountability are guaranteed by the system's fundamental integration of authentication and security.

For the platform to be developed, deployed, and maintained successfully, this document offers a thorough overview of the necessary functionalities, performance requirements, limitations, and system interactions. It covers the entire system, not just certain features or modules.

## **1.2 INTENDED AUDIENCE**

All parties involved in the creation, operation, and supervision of the Project Collaboration and Task Management System are meant to read this software requirements specification. Among the target audience are:

- **System Administrators:** System administrators are in charge of managing users and configuring platforms.
- **Business Analysts (BAs):** Based on user and business needs, business analysts (BAs) define and validate system requirements.
- **Project Managers (PMs):** They are in charge of organizing development schedules, materials, and outputs.

- Software developers: responsible for putting the platform into operation in accordance with established guidelines.
- QA/QC engineers: To guarantee system quality, test security, performance, and functionality.
- End Users (Project Creators and Collaborators) : Individuals or groups utilizing the platform for task tracking, project planning, and collaboration.
- UX/UI Designers: Designers of user-centered interfaces and designs.
- Marketing Teams: Educating prospective users about the features and advantages of the platform.
- Stakeholders and possible investors: Examining the functional overview, business value, and system scope.

The SRS guarantees consistent comprehension and communication throughout the product lifecycle by tailoring the document to the particular requirements of these groups.

## **1.3 INTENDED USE**

The intended audience of our SRS has been declared in the previous section. In this section, we will discuss how they can use the SRS for better understanding.

### **Administrators**

- They will use the SRS to understand system requirements, deployment considerations, access control, and potential administrative tasks associated with managing ProCollab.
- It helps them oversee user roles, permissions, and data integrity across the collaboration platform.

## **Business Analysts (BAs)**

- BAs play a critical role in translating business collaboration needs into functional requirements.
- The SRS provides detailed insights into the workflow logic, task management needs, and communication features, aiding BAs in their analysis and documentation.

## **Project Managers (PMs)**

- Project managers use the SRS to understand the scope, features, and limitations of ProCollab.
- It assists them in planning, tracking, and coordinating multiple projects efficiently while ensuring the solution aligns with organizational objectives.

## **Developers**

- Developers refer to the SRS to understand required functionalities, user stories, workflows, and interface requirements.
- They use the documented specifications to implement system features such as task management, notifications, chat, and file sharing.

## **QA/QC Engineers**

- Quality assurance and control engineers focus on verifying the reliability and usability of ProCollab based on the SRS.
- The SRS helps them design test cases and validate whether the implemented system meets quality and performance standards.

## **Users (Team Members and Project Leads)**

- For users, their interaction with ProCollab is primarily guided by functional requirements like task creation, commenting, file uploads, and activity tracking.

- The SRS helps visualize how each module will operate, ensuring a user-friendly and productive collaboration experience.

### **Stakeholders (Leadership, Clients, Product Owners)**

- Stakeholders use the SRS to understand the overall purpose, features, and advantages of ProCollab.
- It helps them align the platform with business goals and make decisions related to adoption, marketing, and budgeting.

### **Testers**

- Testers use the SRS to design detailed test scenarios and expected behaviors across modules like task tracking, notifications, user management, and chat.
- They ensure the system meets functional and non-functional requirements as specified.

### **Investors**

- Investors review the SRS to understand the value proposition, potential market fit, scalability, and associated risks of ProCollab.
- The document aids in making informed decisions about funding and supporting the project.

## **1.4 PRODUCT SCOPE**

The ProCollab system is a web-based collaboration platform designed to help project teams manage tasks, communication, and documentation in a centralized manner. It enables users to create, assign, and track tasks, manage project resources, and collaborate efficiently, reducing the need for scattered communication tools and manual updates.

### **Purpose**

- This system is designed to streamline project collaboration by providing a single platform where team members can create tasks, assign responsibilities, set deadlines, and communicate effectively. It ensures transparency in project progress and supports real-time updates and notifications, enhancing team productivity.

### **Benefits and Objectives**

- Enhancing Team Collaboration: ProCollab provides tools that facilitate clear and consistent communication among project members.
- Task and Time Management: It helps in organizing tasks and deadlines effectively, ensuring timely project delivery.
- Improved Accountability: By assigning tasks to specific users and tracking their completion, it promotes ownership and accountability.
- Centralized Documentation: Keeps all project-related information, files, and discussions in one place.
- Better Visibility and Control: Project managers can monitor progress, manage workload, and make informed decisions.
- Reducing Communication Gaps: Built-in chat and comment features minimize delays and misunderstandings.

### **Alignment with Corporate Goals**

- ProCollab aligns with organizational goals of improving efficiency, reducing project delays, and fostering better team collaboration. It aids in

delivering successful projects by providing clear workflows and reducing overhead from scattered tools.

## **Relating to Business Strategies**

- By offering a unified platform for team collaboration, ProCollab supports the business strategy of digital transformation and remote workforce enablement. It contributes to building a reputation for innovation and process optimization in project management.

## **1.5 RISK DEFINITION**

The Software Requirements Specification (SRS) of ProCollab identifies potential risks that could impact the platform's functionality, user experience, and overall success. These risks must be managed to maintain a smooth operation, protect user data, and ensure the platform delivers on its intended features.

### **User Non-Compliance**

Risk that users (project members, admins) may not adhere to platform policies, guidelines, or role-based responsibilities. This could lead to issues in task management, project collaboration, and potential violations of platform rules, affecting the overall experience.

### **User Inactivity**

Risk that users may not actively participate in project tasks, updates, or discussions, hindering collaboration and impacting project timelines. This could reduce the effectiveness of task assignment and progress tracking features.

### **Administrator Overload**

Risk that administrators may face an overwhelming workload due to managing user accounts, handling permissions, enforcing policies, and resolving technical issues, potentially leading to delays in support or mismanagement of the platform.

## **Miscommunication Among Stakeholders**

Risk that different stakeholders (users, admins, developers) may interpret the SRS differently, resulting in misalignment regarding platform functionality, priorities, and expectations, ultimately leading to delays or feature mismatches.

## **Changes in Project Scope**

Risk that unforeseen changes or updates to the project scope may not be immediately incorporated into the SRS, leading to confusion among developers, testers, and users about new requirements or features.

## **Shifting Stakeholder Needs**

Risk that evolving stakeholder needs (such as feature requests or modifications) may not be consistently updated in the SRS, requiring frequent communication and updates to ensure the platform aligns with changing expectations.

## **System Downtime and Performance Issues**

Risk that platform downtime or performance issues may occur due to server failures, high traffic, or inadequate load handling, which could disrupt user activity and hinder the platform's usability.

## **User Experience Issues**

Risk that the platform design, navigation, or usability may not meet user expectations, leading to confusion, dissatisfaction, and low user engagement. This could hinder collaboration and task management within projects.

# **CHAPTER 2- OVERALL DESCRIPTION**

## **2.1 USER CLASSES AND CHARACTERISTICS**

In this collaboration and project management system, user classes are defined to address the different roles and responsibilities of users within projects. These roles enable clear access control, task assignment, and project organization.

### **User Class: Registered Users**

#### **Characteristics:**

- Registered users have an account in the system and can authenticate using email and password.
- They can create new projects and become project owners.
- Registered users can log in and manage their account information securely.
- They can request password changes with email verification to protect their accounts.
- They have access to their personalized dashboard showing projects and tasks.
- They can accept invitations to collaborate on projects and manage their project memberships.

## **User Class: Project Members**

### **Characteristics:**

- Project members are users who belong to one or more projects, either as owners or invited collaborators.
- They can create, assign, and update tasks within their projects.
- Members can set task deadlines and statuses to track progress.
- They have the ability to invite new collaborators via email and manage project team composition.
- Project members receive notifications about task assignments and project updates.
- Access control is enforced to prevent unauthorized users from modifying project data.

## **User Class: Invited Collaborators**

### **Characteristics:**

- Invited collaborators are users who have received email invitations to join a project but have not yet accepted.
- They are marked with a “pending” status in the system until they accept the invitation.
- They have limited access and cannot modify project data until they become project members.
- They can accept or decline invitations through a secure link sent via email.
- Upon acceptance, they gain full member privileges within the project.

## User Class: Unauthorized Users

### Characteristics:

- Unauthorized users are visitors or users who have not logged in.
- They are restricted from accessing any project or task management functionalities.
- Attempts to access restricted pages redirect them to the login page with appropriate messages.
- They can sign up to become registered users and gain access after authentication.

## 2.2 USER NEEDS

This section of the Software Requirements Specification (SRS) outlines the specific requirements and expectations of the end-users within the ProCollab collaboration and project management system.

### Registered Users

- **Account Management:** Registered users need a straightforward interface to manage their accounts, including secure signup, login, password changes with email verification, and password recovery.
- **Project Creation and Ownership:** Users require the ability to create new projects easily, specify project details, and manage projects they own.
- **Dashboard Overview:** Users expect a clear dashboard showing summaries of projects they own, projects they joined, and tasks assigned to them, enabling quick progress tracking.

- **Security:** Users need their personal data and project information to be protected through secure authentication and access controls.

## Project Members

- **Task Management:** Project members require intuitive tools to create, update, assign, and track tasks within projects. This includes setting deadlines, priorities, and statuses.
- **Collaboration:** Members need the ability to invite collaborators via email, accept or decline invitations, and clearly see task assignments and updates.
- **Notifications:** Timely notifications about task assignments, project invitations, and status changes are essential for effective collaboration.
- **Access Control:** Members expect that only authorized users can view or modify project and task data to maintain data integrity.

## Invited Collaborators

- **Invitation Handling:** Invited collaborators need a clear, secure process to accept or decline project invitations, including prompts to log in or sign up if necessary.
- **Pending Status Visibility:** They require visibility into their invitation status before joining a project, with clear instructions on how to proceed.
- **Onboarding:** Once accepted, invited collaborators expect seamless integration into the project with appropriate access to tasks and project resources.

## General User Needs

- **Usability:** The system must provide an intuitive, easy-to-navigate interface suitable for users with varying technical skills.
- **Responsive Design:** The interface should be responsive to different device types, including desktops, tablets, and smartphones.

- **Error Handling and Feedback:** Users need informative error messages and feedback when actions fail or succeed, to guide them through system interactions.
- **Performance:** The system should provide quick load times and real-time updates where applicable, to ensure a smooth user experience.

## 2.3 OPERATING ENVIRONMENT

### Software Environment:

- Web-based platform, accessible via browsers.
- Technologies: Frontend (HTML, CSS, JavaScript), Backend (Node.js), Database (MongoDB).
- Secure authentication email notifications
- Device compatibility: Desktop, mobile, and tablet devices.

### Hardware Environment:

- Internet connectivity is required for accessing the platform.

### User Environment:

- Role-based access for users (Admin, Project Member).
- Real-time notifications, task management, and collaboration tools.
- Secure access with encryption and data privacy.

### Security:

- HTTPS, encryption, and role-based access control.
- Regular audits, secure coding, and proper error handling

## 2.4 CONSTRAINTS

### Technical Constraints:

- Selection of appropriate frontend and backend frameworks (e.g., React, Node.js, Firebase, PostgreSQL) for secure user authentication, real-time collaboration, and task/project management.

- Implementation of secure password management, including hashing and email-based verification for password changes.
- Support for role-based access control (e.g., project member vs. non-member), which restricts sensitive actions such as task creation or assignment.
- Integration of email services for invitation handling, verification links, and notifications.
- Scalability to support growing numbers of users, projects, and tasks with minimal performance degradation.

### **Time Constraints:**

- Development is divided into milestone-based sprints for key modules, including authentication, project creation, task management, collaboration features, and the dashboard.
- Deadline for a functional MVP release to align with academic or target market schedules.
- Sufficient time allocation for testing flows such as invitation acceptance, access restrictions, task updates, and error handling.

### **Budget Constraints:**

- Budget must cover core development (frontend/backend), UI/UX design, testing, deployment, and third-party services (e.g., email notifications, cloud database).
- Additional expenses for hosting (e.g., Firebase, AWS), version control (e.g., GitHub/GitLab), and future marketing or promotional campaigns if commercialized.
- Cost consideration for securing a custom domain, secure sockets (SSL), and potential premium APIs for future scaling.

### **Regulatory and Compliance Constraints:**

- Compliance with relevant data privacy regulations (e.g., GDPR) for handling personal user data (e.g., names, emails, passwords, and project

details).

- All passwords must be securely hashed; no sensitive data should be stored in plain text.
- Data access must be limited to authorized users based on role (e.g., preventing unauthorized task or project access).
- Secure handling of email-based actions such as invitations and password reset to prevent unauthorized access.

### **Resource Constraints:**

- Limited availability of skilled developers may slow down complex features such as drag-and-drop task boards or email-based workflows.
- UI/UX design may be constrained by available tools and the design team's availability.
- Testing and QA may be limited to internal teams without access to large-scale beta testing resources.

## **2.5 ASSUMPTIONS**

- Assumes that users (project owners and members) will actively use the platform to create projects, invite collaborators, assign tasks, and manage deadlines and statuses.
- Assumes that users have a basic understanding of online collaboration tools and can navigate through project dashboards, forms, and dropdowns.
- Assumes that administrators or project owners have the skills and permissions needed to manage users, assign tasks, and monitor activities within their projects.
- Assumes that users will provide valid email addresses to receive invitations, password reset links, and project-related notifications.

- Assumes that all users (including new signups and invitees) have consistent access to the internet to interact with real-time features and receive system updates.
- Assumes that all stakeholders (developers, testers, designers, project owners) have access to the Software Requirements Specification (SRS) and understand its content for unified project alignment.
- Assumes that information documented in the SRS aligns with the expectations of all involved parties and serves as a reliable reference throughout the development lifecycle.
- Assumes effective collaboration and timely communication among team members and stakeholders to avoid delays in implementation.
- Assumes that role-based access and authentication flows are correctly implemented and tested to ensure secure operation.

# CHAPTER 3 - REQUIREMENTS

## 3.1 FUNCTIONAL REQUIREMENTS

### User Story 1: User Signup

As a new user,  
I want to register with my name, email, and password,  
So that I can create and participate in projects using ProCollab.

#### **Description:**

A new user should be able to sign up by providing the necessary information. Duplicate email registrations should be prevented. Passwords should be securely hashed. After a successful sign-up, the user should be logged in automatically and redirected to the dashboard.

#### **Confirmation:**

- Given I am on the signup page, when I load the page, I should see a form with the following fields:

Full Name (required, max 100 characters)  
Email (required, valid format)  
Password (required, minimum 6 characters)  
Confirm Password (required, must match Password)  
Signup button  
"Already have an account? Login" link

2. Given I submit valid information, when I click "Signup", then:

My account is created and stored in the database. I am automatically logged in and redirected to the dashboard. I see a success message: "Signup successful. Welcome to ProCollab!"

3. Given I submit invalid or duplicate information, when I click "Signup", then:

I see appropriate error messages (e.g., "Email already in use", "Password too short"). The form remains filled for correction.

4. Given I am already logged in, when I try to access the signup page, then:

I am redirected to the dashboard automatically.

## **User Story 2: User Login**

As a registered user,  
I want to log in using my email and password,  
So that I can securely access my ProCollab dashboard and project data.

### **Description:**

A registered user should be able to log in with their email and password. If the credentials are valid, they should be directed to the dashboard. Invalid login attempts should show appropriate error messages. Unauthenticated users should not access any restricted project pages.

### **Confirmation:**

1. Given I am on the login page, when I load the page, I should see a form with the following fields:

Email (required, valid email format)  
Password (required, at least 6 characters)  
Login button  
"Forgot Password?" link  
"Sign up" link for new users

2. Given I enter valid credentials and click "Login", then:

I am authenticated and redirected to my dashboard. A session is started to maintain my login. I see a success message: "Login successful."

3. Given I enter incorrect credentials, when I click "Login", then:

I see an error message: "Invalid email or password." The form remains filled for correction.

4. Given I am already logged in, when I access the login page again, then:

I am redirected to the dashboard automatically.

### **User story 3: Create project**

As a logged-in user

I want to create a new project by entering relevant fields like name, description  
So that I can start organizing and managing my project tasks and members

#### **Description**

Users will have access to a "Create Project" page or modal where they can enter a Project Name (required), Description (optional), and Visibility (optional: Private/Public).

Once submitted, the project gets stored in the database, and the creator becomes the Project Owner.

#### **Confirmation:**

1. Project Created Successfully

Given: I am logged in.

When: I fill the project form with required details and click Submit.

Then:

The project is saved in the database with me as the owner.

I see a success message: "Project created successfully."

The project appears in my Dashboard under Owned Projects.

## 2. Missing Required Field

Given: I am logged in.

When: I try to submit the form without entering the project name.

Then:

I get a validation error: “Project Name is required.”

The project is not saved.

## 3. Not Logged In

Given: I am not logged in.

When: I try to access the Create Project page or submit the form directly.

Then:

I get an error: “You must log in to create a project.”

I am redirected to the Login page.

## 4. Backend Save Error

Given: I am logged in and have filled the form.

When: The database/server has an issue while saving.

Then:

I see an error: “Unable to save project right now. Please try again later.”

The project has not been created.

## **User story 4: Dashboard**

As a logged-in user

I want to see a summary of my owned projects, joined projects, and tasks assigned to me,

So that I can quickly track my projects and task progress from a single place.

### **Description**

The dashboard will show:

Owned Projects: Projects I created.

Joined Projects: Projects where I am a member.

My Tasks: Tasks assigned to me (across all projects).

## **Confirmation:**

### 1. View Projects

Given: I am logged in.

When: I open my Dashboard page.

Then:

I see two project sections:

“Owned Projects”

“Joined Projects”

Each project shows: Name, Short Description, and Status (Active/Archived).

### 2. View Assigned Tasks

Given: I am logged in.

When: I open the dashboard,

Then:

I see a “My Tasks” section.

It shows a list or board view of tasks assigned to me across all projects.

Each task shows: Task Name, Status (Not Started / In Progress / Completed), Due Date (if any).

### 3. No Projects or Tasks Yet

Given: I just signed up and have no projects or tasks.

When: I open the dashboard.

Then:

I see a message like: “No projects found. Start by creating one!”

And for tasks: “No tasks assigned yet.”

### 4. Not Logged In

Given: I am not logged in.

When: I try to access the Dashboard URL directly.

Then:

I get an error: “Please log in to view your dashboard.”

I am redirected to the Login page.

## 5. Backend Save Error

Given: I am logged in.

When: The database/server has an issue while loading data..

Then:

I see an error: "Unable to load right now. Please try again later."

## **User Story 5: Invite a Collaborator via Email**

As a project member,

I want to invite a collaborator via email,

So that I can invite others to join the project and contribute to the tasks.

### **Description:**

A project member should be able to invite a potential collaborator by entering their email address. An invitation email will be sent, and the user will be marked with a "pending" status in the database until they accept the invitation.

### **Confirmation:**

1. Given that I am a project member,

I navigate to the invite collaborator page and see a form to enter the collaborator's email address (required, valid email format) and an Invite button.

2. Given I enter a valid email address and click "Invite", then:

An email invitation is sent to the provided email address. The collaborator's status is set to "Pending" in the database. Then:

I see a success message: "Invitation sent successfully."

3. Given I attempt to invite an email address that already belongs to an active project member, when I click "Invite", then:

I receive an error message: "User is already a member of the project," and the invitation is not sent.

4. Given I am not a project member, when I try to invite a collaborator, then:

I am denied access with an error message: "Unauthorized access."

### **User Story 6: Accept Invitation**

As an invited user,

I want to accept an invitation to join the project,

So that I can collaborate with the project members and contribute to tasks.

#### **Description:**

An invited user should be able to accept the invitation sent via email. Upon accepting, they should be added to the project, and their status will be updated to "Project Member."

#### **Confirmation:**

1. Given I receive an email invitation, when I click on the acceptance link, then:

I am redirected to a page where I can accept or decline the invitation.

I see a button labeled "Accept Invitation."

2. Given I click "Accept Invitation", then:

My status is updated to "Project Member" in the database and access the project's task list and resources.

I see a success message: "You have successfully joined the project."

3. Given I am not logged in, and I try to accept the invitation after clicking on the invitation link,

I am prompted to log in first, and after logging in, I can successfully accept the invitation.

4. Given I attempt to accept an invitation after the project has been archived or deleted, when I click the acceptance link, then:

I see an error message: "The project is no longer available, cannot join the project."

- Given I am not invited to the project, when I attempt to access the invitation link, then:

I see an error message: "Invitation not found."

### **User Story 7: Create a Task**

As a project member,  
I want to create a task with basic information,  
So that I can document and track specific work that needs to be done within the project.

#### **Description:**

A project member should be able to create a task by providing essential details such as title, description, and optional fields like priority or category. The task should be associated with the project and visible to all project members.

#### **Confirmation:**

- Given that I am a project member, when I navigate to the task creation page,  
Then I should see a form with fields for:

Task title (required, max 100 characters)  
Task description (optional, max 500 characters)  
Priority (optional, dropdown: Low, Medium, High)  
Category (optional, predefined list, "Development", "Design")

- Given I submit a valid form, when the task is created, then:

The task is saved in the database with a unique ID, linked to the project.  
The task appears in the project's task list for all members.  
I receive a confirmation message ("Task created successfully").

- Given I submit an invalid form (like an empty title), when I click submit, then:

I receive an error message ("Task title is required").  
The form remains populated with my input for correction.

4. Given I am not a project member, when I attempt to create a task, then  
I will be denied access with an error message showing ("Unauthorized access").

## **User Story 8: Assign a Task**

As a project member,  
I want to assign a task to one or more collaborators involved in the project,  
So that responsibilities are clearly defined and team members know their tasks.

### **Description**

A project member should be able to assign a created task to one or more collaborators who are part of the project. The assignment process should allow selecting users from a list of project members and notify them of their new responsibilities.

### **Confirmation:**

1. Given I am a project member, when I view a task, then

I should see an option to assign it to collaborators (multi-select list of project members).

2. Given I select one or more collaborators, when I save the assignment, then:

The task is updated in the database with the assigned collaborators' IDs.  
Assigned collaborators receive a notification ( in-app or email) about the task assignment.

The task reflects the assignees in the project's task list or dashboard.

3. Given I attempt to assign a task to a user not in the project, when I try to save, then:

I receive an error message ( "User is not a project member").  
The assignment is not saved.

4. Given that I am not a project member, when I attempt to assign a task, then

I am denied access with an error message ("Unauthorized access").

- Given a task is assigned, when a collaborator views their dashboard, then

They should see the task listed under their assigned tasks.

### **User Story 9: Set a Task Deadline and Status**

As a project member,

I want to set a deadline and status for a task,

So that I can track its timeline and progress within the project.

#### **Description:**

A project member should be able to assign a deadline (due date) and update the status of a task to reflect its current progress (e.g., "Not Started," "In Progress," "Completed"). This helps in better project planning and tracking.

#### **Confirmation:**

- Given that I am a project member, when I view or edit a task,

Then I should see options to:

Set/modify a deadline (date + time picker, optional).

Update the status (dropdown: "Not Started," "In Progress," "Completed")

OR, Update status via dropdown or Trello-style drag-and-drop cards ("Not Started"/"In Progress"/"Completed").

- Given I set a deadline and status for a task, when I save the changes,

then:

The task is updated in the database with the new deadline and status.

The changes are reflected in the task list for all project members.

I receive a confirmation message ("Task updated successfully").

- Given I leave the deadline empty but update the status, when I save the changes,

then:

Only the status is updated, and the deadline remains unchanged (if previously set) or unassigned.

- Given that I am not a project member, when I attempt to modify a task's deadline or status,

Then I am denied access with an error message ("Unauthorized access").

### **User Story 10: Change Password with Email Verification**

As a registered user,  
I want to change my password securely with email verification,  
So that my account remains protected from unauthorized access.

#### **Description:**

A user should be able to request a password change, receive a verification email, and securely set a new password. This ensures that only the legitimate account owner can modify the password.

#### **Confirmation:**

- Given I am logged in, when I navigate to the "Change Password" page,  
Then I see a form with fields for:

Current password (required).

New password (required, must meet security criteria: min 8 chars, special characters).

Confirm new password (required, must match new password).

- Given I submit a valid password change request, when I confirm,  
then:

A verification email is sent to my registered email address.

I see a message ("Please check your email to complete the password change").

- Given I click the verification link in the email, when I enter and confirm the new password,  
then:

My password is updated in the system.

I receive a success message ("Password changed successfully").

I am logged out and must log in again with the new password.

4. Given I enter an incorrect current password, when I submit the form, then:

I see an error message ("Current password is incorrect").

The form retains my input for correction.

5. Given I am not logged in, when I try to access the password change page, Then

I am redirected to the login page with a message ("Please log in first").

6. Given I am not logged in and forget my password, when I click "Forgot Password?" on the login page or somewhere in my profile page,

Then I see a form to enter my registered email address to reset the password.

## 3.2 NON-FUNCTIONAL REQUIREMENTS

### Performance Requirements

- **Response Time:** The system should respond to user interactions (e.g., dashboard load, task creation, login) within 2 seconds under standard usage.
- **Scalability:** The system should be able to support more users, projects, and tasks without slowing down or crashing.
- **System Growth:** The system should support an increasing number of users, projects, and tasks while maintaining stable performance and responsiveness.

### Safety Requirements

- **Data Backup:** The system should perform automatic database backups every 24 hours to prevent data loss.

- **Crash Recovery:** In the event of system or server failure, critical data (including tasks, user sessions, and project links) must not be lost. Recovery mechanisms should restore the last consistent state of user actions.
- **Consistent Data Handling:** All user-entered data (such as tasks, comments, and deadlines) must be stored persistently and remain intact even during unexpected system interruptions.

## Security Requirements

- **User Authentication:** Users must log in securely using encrypted passwords. Support for 2-Factor Authentication (2FA) should be included for enhanced security.
- **Password Storage:** The system must hash and salt all user passwords before storing them in the database to protect against password theft.
- **Access Control:** The system must enforce strict role-based access to ensure that only authorized users can view or modify project data.
- **Data Encryption:** All sensitive information, including credentials and invitation tokens, must be encrypted during transmission (via HTTPS) and securely stored.

## Software Quality Attributes

- **Usability:** The system interface should be clean, consistent, and intuitive, requiring minimal learning for new users. A minimum of 90% of beta testers should rate it as "easy to use" in post-test surveys.
- **Reliability:** The platform should maintain 99.9% uptime to ensure consistent access for collaboration, especially during critical periods.
- **Maintainability:** The backend codebase should follow modular and clean architecture standards to simplify debugging and future enhancements.

- **Portability & Browser Compatibility:** The system should work seamlessly across all modern web browsers, including Google Chrome, Mozilla Firefox, Apple Safari, and Microsoft Edge.
- **Responsiveness:** The UI should adapt fluidly to various screen sizes for use on desktops, tablets, and smartphones.

## Business Rules

- **Project Ownership:** Only registered users can create projects. A user can own multiple projects and also be a collaborator in others.
- **Invitation Expiry:** Invitations sent to potential collaborators will expire after 7 days if not accepted. Expired invitations are automatically removed from the pending list.
- **Task Visibility:** A user can only view or interact with tasks in projects they are a member of. Unauthorized access attempts should be blocked and logged.
- **Archived Projects:** Once a project is archived, it becomes read-only for all members. No new tasks or changes can be made unless reactivated by the project owner.