Tasky

Requirements Specification and Analysis Document

26.03.2021

Oğuz Kaan Yazan
Berke Biten

REQUIREMENTS ANALYSIS DOCUMENT

1. Introduction

1.1 Purpose of the System

Most of the people work on multiple tasks every day in their job, school or even in home. Working on these tasks is getting harder if you try to handle it with a pen and paper. Also, for the people who are responsible for managing a team, it is really hard to monitor the project, tasks and the team. The purpose of the task management system Tasky is making easier and more efficient working and managing the projects and tasks. Users will easily create, assign and prioritize tasks, set deadlines, track how much time spent on tasks and visualize the projects and tasks with the Tasky.

1.2 Scope of the System

Task management system Tasky is designed for making tracking a project easier and more efficient. The system consists of a web site and Mobile application. Users can create projects in the web site and add participants to the project. Once the project is created users can create tasks within the project. Each task has a reporter and assignee. Assignee should be selected when the task is created. The reporter of the tasks is the user that is created the task. Users can also create sub-task which is related with any of the existing tasks. The only difference between task and sub-task is that sub-tasks have a root task. Users can log work under any task and indicate how much time they spent and the description of the work.

There are three roles in a single project. Project manager, watcher and team member. Project manager is the user who is the created the project by default. Project managers can create tasks and assign it to one of the team members. Project managers also can view the reports about the project, tasks and team members. Team members can create tasks too, but they can assign this task to only themselves. They can log work to the tasks. Users whose have watcher role in the project can only monitor the project. They can not do any operation.

1.3 Objectives and Success Criteria of the Project

The objective of the task management system Tasky is providing a strong platform for managing projects and tasks with a web site and Mobile application. By developing this system, we aim to increasing the efficiency of the projects, making easier managing and working on the projects and giving the information about what has been done for the projects and tasks.

The main success criteria is how many projects in the system are successful? Like we mentioned previously we aim to increasing efficiency of the projects. Therefore, if the projects in the system is successful the project is also successful. Also, number of the projects and tasks in the system and the daily usage of the web site and Mobile application are important success criteria for us.

1.4 Definitions, Acronyms, and Abbreviations

Tasky: Name of the proposed task management system.

1.5 Overview

This document contains Introduction, Current System, Proposed System, Glossary and References sections.

In the first part of the introduction, we define the purpose of our system. Then in the section 1.2, we explain the actors and functionalities of the system. In the section 1.3 we talk about the objectives and success criteria of the project. We define some terms that is used in this document in the section 1.4. And for the final part of the introduction, we explained how this document is designed.

In the second part we mention some other systems that is using for task management currently.

In the Proposed System section, we explained our system in detail. First we explain the functional and nonfunctional requirements of the project. In the functional requirements section, we explain all the functions that can be done in the system and which actors will be able to use these functionalities. In the nonfunctional requirements we explained the functionalities that is not directly related with the project.

In the System Models section, we describe the scenarios and use cases of the system in order to indicate the interactions between the actors and system. In these sections we clearly explain how the functions of the system should be done. In the Object Model section, we create a UML class diagram in order to define all the classes and their attributes and operations. We create our dynamic model which consist of sequence and state diagram in the Dynamic Model section. In the final section of the System Models, we design the user interfaces of the system.

In the final sections of the RAD document, first we indicate the terminology of the system in the Glossary. Then in the References section, we list our references.

2. Current System

There are a lot of task management system in use. Jira, Azure Board and Trello are the most popular task management systems. In these systems basically there are projects, tasks and work logs. Monitoring of the projects are handling with these concepts. There are members within a project. Project members logs work under tasks and update the status of the task. By that way, they can track their tasks and plan their works.

In our system the concept is the same. But our main goal is making processes easier and faster than current systems. We aim to ensure simpler and more undersantable interface. Also, with the Mobile application we aim to make accessible our system from everywhere.

3. Proposed System

In this project, a task management system which consists of a web and Mobile application named Tasky is proposed. This system is designed to make easier and more efficient all the processes that can be done in the project by any member of the project team. For project managers, managing and controlling the project, tasks and team members will become easier. Project managers will be able to see the answers of most of the questions about the project such as:

- Which project member is working on which task?
- How much time is spent for a specific task or project?
- How many hours do project members work for the project in a day?

For team members who working on a task of the project; following the tasks that he/she will do, logging work under the tasks and keeping themselves up to date about the other tasks in the project will be easier.

3.1 Overview

The main functions of the Tasky are creating projects and tasks. There are three different roles within each project which defines the accessibility to the system's functionalities for the users.

First role is the project manager. Project managers are the users who created the project. Project managers can invite users to the project. They can create tasks and sub-tasks. Then they can assign these tasks to the any participant of the project. If a project manager work on a task, he/she can log work under this task. Project managers can view reports about the project, tasks and participants to see the how is project going on.

Second role is team member. Team members are the users who work on tasks within the project. They can log work under the tasks. They can also create task and sub-tasks. But they can assign these tasks to only themselves.

Third role is the watcher. Watcher is the who has no permission to do any operation within the project. They can only monitor the project.

3.2 Functional Requirements

Users can register to the system by filling a registration form. They can login to the system with e-mail and password information. They can also logout. After they logged in to the system, they have a profile created by their registration credentials. They can view and edit their profile. Users also can manage their preferences which includes turn on/off notifications and change password functions. These functions are available in both platforms.

Users can create and manage projects via Tasky web application. While creating a project they can invite participants by indicating their e-mails. If there is already a user registered with the given e-mail, the system sends an invitation e-mail and Mobile notification; otherwise, only an e-mail sent. Invited users can accept or decline the invitation. Creator of the project becomes the project manager. Project managers can assign a role to the project participants. Project managers can update the project by changing the status, name or description of the project. Project managers can add or remove participants to/from projects. Project managers can also delete the project.

Participants of the project can view project detail and project board in both platforms. Project detail contains name, description, participants and documents of the project. Project board contains tasks grouped by their status (To-do, active, resolved and closed).

Project managers and team members can create tasks under their projects via only Tasky web application. These tasks can have sub-tasks. The only difference between the task and sub-task is that sub-tasks have a root task.

These tasks can be assigned to a team member. Project managers can assign a task to anyone in the project except watchers. Team members can only assign tasks to themselves. Project managers and assignee of the task can update the task's status, description, files. All participants of the project can view the detail of tasks. Project managers and team members can log work to the tasks. A task can be deleted by only a project manager. These functions are available in both platforms.

Project participants can view the task list of the project via both platforms. In this page/screen they can filter or sort the tasks. Users also can view the last activities of their projects.

Project managers can view the report of their projects via only web application. In the project report they can see the stats of tasks or team members.

Project managers will receive e-mail and Mobile notifications when:

- A user accepts or declines their invitation.
- A task is updated.
- A work is logged into a task they created.
- A team member created a new task.
- A participant left the project.

Team members will receive e-mail and Mobile notifications when:

- A task is assigned to them.
- A work is logged into a task they created.
- A task that is assigned to them is updated or deleted.
- They are removed from a project.

3.3 Nonfunctional Requirements

3.3.1 Usability

- Users should be able to complete their transactions with maximum 5 clicks.
- The components of web and Mobile application such as titles, button names, section names etc. should be descriptive and understandable for people who knows English.

3.3.2 Reliability

- The system should be available 100% of time.
- The system should validate the data that will be stored in the database and if there are errors, users should be notified by the system.

3.3.3 Performance

- The response time of the system should be maximum 30 seconds for heavy functions like preparing a project report, for other functions it should be maximum 10 seconds.

3.3.4 Supportability

- The system should be separated into packages/components to accomplish easy maintenance.

3.3.5 Implementation

- The Mobile application should be implemented with React-native (JavaScript).
- Web application should be implemented with React (JavaScript).
- Web API's that will be used in the Mobile and web application should be implemented with .NET 5 (C#).
- Database should be MSSQL.
- Visual Studio Code should be used for React and React-native. Visual Studio 2019 should be used for .NET 5.

3.3.6 Interface

- There is no interface requirement in this project.

3.3.7 Packaging

- Our web application does not require any installation. It should be run on any internet browser.
- Mobile application should be run on Android and iOS platforms by installing the application.

3.3.8 Legal

- Mobile application should be signed with a signing key for android. For iOS, it should be signed with a certificate.

-

3.4 System Models

3.4.1 Scenarios

Scenario Name: Login (Web + Mobile)
Participating Actors: Ali: Registered User
Flow of Events:

1- Ali clicks on the Login button on the main page/screen of the Tasky web/Mobile application.
2- Ali enters his email and password to login form and clicks to login button.
3- Given login credentials are checked and if they match Ali is redirected to his homepage, otherwise Ali gets an error feedback that declares credentials are wrong.

Scenario Name: Register (Web + Mobile)
Participating Actors: Berke: Guest
Flow of Events:

1- Berke clicks on the Register button on the main page/screen of the Tasky web/Mobile application.
2- Berke enters asked information to the registration form.
3- Given credentials are checked and if they are valid an activation email is sent to Berke's email.
4- Berke clicks on the URL that is sent to his email and activates his account.
5- Registration is completed.

Scenario Name: Logout (Web + Mobile)
Participating Actors: Ali: Registered User
Flow of Events:

1- Ali clicks on the logout button which is visible on every page/screen of Tasky web/Mobile application.
2- Ali is logged out from the Tasky application.

Scenario Name: View Profile (Mobile)
Participating Actors: Ali: Registered User
Flow of Events:

1- In the main screen of Tasky Mobile application, Ali clicks on the Profile button from the drawer menu.
2- System redirects Ali to profile screen.

Scenario Name: View Profile (Web)
Participating Actors: Ali: Registered User

Flow of Events:

1- Ali clicks on his profile image on top right corner of the Tasky web application.
2- Ali clicks on the Profile button on the opened menu.
3- System redirects Ali to profile page.

Scenario Name: Edit Profile (Mobile)
Participating Actors: Ali: Registered User
Flow of Events:

3- Ali clicks on the Profile button from the drawer menu in the main screen.
4- System redirects Ali to profile screen.
5- Ali clicks on the edit icon on the top right corner of the screen.
6- System opens a form that is filled with Ali's information.
7- Ali edits the information that he wants to change.
8- Ali clicks on the save button.

Scenario Name: Edit Profile (Web)
Participating Actors: Ali: Registered User
Flow of Events:

1- Ali clicks on his profile image on top right corner of the Tasky web application.
2- System opens a menu.
3- Ali clicks on the Profile button on the menu.
4- System redirects Ali to profile page. Then Ali clicks on the edit button.
5- System opens a form that is filled with Ali's information.
6- Ali edits the information that he wants to change.
7- Ali clicks on the save button.

Scenario Name: Manage User Preferences (Mobile + Web)
Participating Actors: Ali: Registered User
Flow of Events:

1- Ali clicks on the Preferences button from the drawer/side bar menu in the main screen/page of Tasky Mobile/web application.
2- System opens the preferences form.
3- Ali edits the preferences that he wants to change.
4- Ali clicks on the save button.

Scenario Name: Create Project (Web)
Participating Actors: Ali: Registered User

Flow of Events:

1- Ali clicks on the Create Project button on the projects page of the Tasky web application.
2- Ali fills the Project Creation form and clicks to Create button.
3- Project is created and if Ali added any participants, an invitation email sent to their emails.

Scenario Name: Update Project (Web)
Participating Actors: Sude: Project Manager
Flow of Events:

1- Sude clicks on the Detail tab of the project that she is the manager of.
2- Sude clicks on the Edit button and changes the field(s) that she wants to update.
3- Sude clicks Save button and project is updated.

Scenario Name: Delete Project (Web)
Participating Actors: Sude: Project Manager
Flow of Events:

5- Sude clicks on the Detail tab of the project that she is the manager of.
6- Sude clicks on the Delete button.
7- Sude enters her password to confirm deletion.
8- Project is deleted.

Scenario Name: Invite Participant to Project (Web)
Participating Actors: Ayşe: Project Manager
Flow of Events:

1- Ayşe clicks on the projects button in the side bar menu and opens the list of the projects. Then she navigates to the project detail by clicking on the project name.
2- Ayşe clicks on the add participant button under the project participants section in the project detail.
3- Ayşe writes the e-mail of the person that she wants to invite and clicks on the invite button.
4- System sends an e-mail and Mobile notification to the invited users.

Scenario Name: Remove Participant from Project (Web)

Participating Actors: Ayşe: Project Manager

Flow of Events:

1- Ayşe clicks on the projects button in the side bar menu and opens the list of the projects.
2- She navigates to the project detail by clicking on the project name.
3- Ayşe clicks on the remove button which is next to the name of the participant that she wants to remove.
4- System opens a warning box to ask she is sure to remove that participant.
5- Ayşe confirms that she wants to remove selected user from the project by clicking the remove button in the warning box.

Scenario Name: Accept/Decline Invitation (For Registered User)

Participating Actors: Ahmet: Registered User

Flow of Events:

1- Ahmet receives an e-mail and Mobile notification that tells he is invited to a project.
2- Ahmet clicks on the url in the e-mail or Mobile notification.
3- Ahmet displays the information of the project that he is invited. He accepts or declines the invitation by clicking the accept or decline button.
4- Manager of the project gets a notification about the Fatma's response.

Scenario Name: Accept/Decline Invitation (For Guest)

Participating Actors: Fatma: Guest

Flow of Events:

1- Fatma receives an e-mail that tells she is invited to a project.
2- Fatma clicks on the url in the e-mail.
3- System opens the registration form.
4- Fatma fills that form and clicks on the sign-up button.
5- System opens the project information page.
6- Fatma accepts or declines the invitation by clicking the accept or decline button.
7- Manager of the project gets a notification about the Fatma's response.

Scenario Name: View Project Detail (Web + Mobile)

Participating Actors: Kemal: Project Participant

Flow of Events:

1- Kemal clicks on the project's name that he wants view details in the projects page/screen.
2- Kemal clicks on the Details tab on the project page/screen.

Scenario Name: View Project Report (Web)

Participating Actors: Sude: Project Manager
Flow of Events:

1- Sude clicks on the project's name that she wants view report of, in the projects page.
2- Sude clicks on the Report tab on the project page.

Scenario Name: View Project Board (Web + Mobile)
Participating Actors: Ahmet: Project Participant
Flow of Events:

1- Ahmet clicks on the project's name on projects tab.
2- Ahmet clicks on the Board tab in project page/screen.

Scenario Name: Create Task
Participating Actors: Ayşe: Project Manager, Kemal: Team Member
Flow of Events:

1- In the main page of Tasky web application, Ayşe/Kemal clicks on the name of the project on the side bar menu and navigates to project detail.
2- Ayşe/Kemal clicks on the create new task button.
3- Ayşe/Kemal fills the new task form and clicks on the create button.
4- If Ayşe selects an assignee for the created task, system sends a notification to the assignee about this task is assigned to him/her.

Scenario Name: Create Sub-task
Participating Actors: Ayşe: Project Manager, Kemal: Team Member
Flow of Events:

1- In the main page of Tasky web application, Ayşe/Kemal clicks on the name of the project on the side bar menu and navigates to project detail.
2- Ayşe/Kemal displays the task list and clicks on the name of the task. Then he/she navigates to task detail
3- Ayşe/Kemal clicks on the create sub-task button.
4- Ayşe/Kemal fills the new sub-task form and clicks on the create button.
5- If Ayşe selects an assignee for the created task, that assignee gets a notification about this task is assigned to him/her.

Scenario Name: View Task List (Web + Mobile)
Participating Actors: Ahmet: Project Participant
Flow of Events:

1- Ahmet clicks on the project's name on projects tab.
2- Ahmet clicks on the Tasks tab in project page/screen.

Scenario Name: Filter Task List (Web + Mobile)

Participating Actors: Ahmet: Project Participant

Flow of Events:

1- Ahmet clicks on the project's name on projects tab.
2- Ahmet clicks on the Tasks tab in project page/screen.
3- Ahmet selects which filter(s) he wants to use for filtering and clicks them.
4- Task list is filtered according to selected options.

Scenario Name: Sort Task List

Participating Actors: Ahmet: Project Participant

Flow of Events:

1- Ahmet clicks on the project's name on projects tab.
2- Ahmet clicks on the Tasks tab in project page/screen.
3- Ahmet clicks on the column that he wants to order by.
4- Task list is sorted according to clicked column.

Scenario Name: Delete Task

Participating Actors: Ayşe: Project Manager, Kemal: Team Member, Oğuz: Assignee of the Task

Flow of Events:

1- In the main page of Tasky web application, Ayşe clicks on the name of the project on the side bar menu and navigates to project detail.
2- Ayşe displays the task list and clicks on the name of the task.
3- System opens the task detail.
4- Ayşe clicks on the delete task button.
5- Ayşe confirms that she wants to delete this task. Then system deletes the task.
6- Oğuz gets a notification about the task is deleted.

Scenario Name: View Task Detail (Web)

Participating Actors: Ayşe: Project Manager, Kemal: Team Member, Zeynep: Watcher

Flow of Events:

1- In the main page of Tasky web application, Ayşe/Kemal/Zeynep clicks on the name of the project on the side bar menu and navigates to project detail.
2- Ayşe/Kemal/Zeynep displays the task list and clicks on the name of the task. Then he/she navigates to task detail

Scenario Name: View Task Detail (Mobile)

Participating Actors: Ayşe: Project Manager, Kemal: Team Member, Zeynep: Watcher

Flow of Events:

1- In the main screen of Tasky Mobile application, Ayşe/Kemal/Zeynep clicks on the Projects tab from the bottom tab menu.
2- System opens the projects list.
3- He/she clicks on the project name and navigates to project detail.
4- Ayşe/Kemal/Zeynep displays the task list and clicks on the name of the task. Then he/she navigates to task detail

Scenario Name: Log Work (Mobile)

Participating Actors: Kemal: Team Member, Oğuz: Reporter of the Task
Flow of Events:

1- In the Tasky Mobile application, Kemal navigates to task detail from the main screen or task list screen.
2- He clicks on the Log Work button.
3- System opens a work log form.
4- He fills the form and clicks on submit button.
5- Oğuz gets a notification about the logged work.

Scenario Name: Log Work (Web)
Participating Actors: Kemal: Team Member, Oğuz: Reporter of the Task
Flow of Events:

1- In the Tasky web application, Kemal navigates to task detail from the main page.
2- He clicks on the Log Work button.
3- System opens a work log form.
4- He fills the form and clicks on submit button.
5- Oğuz gets a notification about the logged work.

Scenario Name: Update Task
Participating Actors: Kemal: Team Member, Ayşe: Project Manager
Flow of Events:

1- In the Tasky web application, Ayşe/Kemal navigates to task detail from the main screen or task list screen.
2- He/She clicks on the Edit button.
3- System opens the Task form.
4- He/She edits the information that he/she wants to change. Then he/she clicks on the save button.

Scenario Name: Update Task Status (Web)
Participating Actors: Kemal: Team Member, Ayşe: Project Manager
Flow of Events:

1- In the Tasky web application, Ayşe/Kemal navigates to project board from the project detail page.
2- He/She drags the task and drops it to the status column that he/she wants.

Scenario Name: Update Task Status (Mobile)
Participating Actors: Kemal: Team Member, Ayşe: Project Manager
Flow of Events:

1- In the Tasky Mobile application, Ayşe/Kemal navigates to task detail from the main or task list screen.
2- He/She clicks on the status text and selects the new status of the task.

Scenario Name: View Last Activities (Web)

Participating Actors: Kemal: Team Member, Ayşe: Project Manager, Zeynep: Watcher
Flow of Events:

1- In the Tasky web application, Ayşe/Kemal/Zeynep logs into the system.
2- System opens the last activities list.

Scenario Name: View Last Activities (Mobile)
Participating Actors: Kemal: Team Member, Ayşe: Project Manager, Zeynep: Watcher
Flow of Events:

1- In the Tasky Mobile application, Ayşe/Kemal/Zeynep clicks on the Activities tab.
2- System opens the last activities list.

3.4.2 Use case model

Use case name	**Login (Web + Mobile)**
Participating Actors	Initiated by **Registered User**
Flow of events	1. **Registered User** clicks on the login button on the main page/screen of the Tasky web/Mobile application. 2. **SYSTEM** shows a form. 3. **Registered User** enters his 'email' and 'password' to the form and clicks to login button. 4. **SYSTEM** checks given login credentials and redirects user to the main page/screen.
Entry condition	**Registered User** not logged into the system.
Exit conditions	**Registered User** navigates to other pages.
Exceptions	3a. **Registered User** enters the "email" and "password" fields wrong. 4a. **SYSTEM** gives a feedback that says "email or password is wrong". 3b. **Registered User** doesn't fill the "username" or "password" field or both. 4b. **SYSTEM** gives a feedback that says "both fields must be filled."

Use case name	**Register (Web + Mobile)**
Participating Actors	Initiated by **Guest**
Flow of events	1. **Guest** clicks on the Register Button on the main page/screen of the Tasky web/Mobile application. 2. **SYSTEM** responds by showing the Registration form. 3. **Guest** fills the form by entering his "first name", "last name", "email", "password" to the Registration Form and clicks to Register Button. 4. **SYSTEM** validates the form and sends an activation email to the given email address. 5. **Guest** clicks to the activation link in the email. 6. **SYSTEM** activates user's account.
Entry condition	**Guest** is in the main screen.
Exit conditions	**Guest** navigates to other screens.
Exceptions	3a. **Guest** enters at least one of the fields wrong. 4a. **SYSTEM** gives a feedback that indicates the related field is invalid. 3b. **Guest** enters an email that is already registered. 4b. **SYSTEM** gives a feedback that says "Email is already in use." 5a. **Guest** tries to activate his account after the link is expired. 6a. **SYSTEM** gives a feedback that says, "link has expired".

Use case name	**Logout (Web + Mobile)**

Participating Actors	Initiated by **Registered User**
Flow of events	1. **Registered User** clicks on the Logout button/icon on web/Mobile application 2. **SYSTEM** responds by logging the user out and redirecting the user to the main page/screen. and redirects the Registered User to the profile screen/page.
Entry condition	**Registered User** is logged in.

Use case name	**View Profile (Web)**
Participating Actors	Initiated by **Registered User**
Flow of events	1. **Registered User** clicks on the profile image which is on the top right corner. 2. **SYSTEM** shows a menu. 3. **Registered User** clicks on the Profile button in the menu. 4. **SYSTEM** responds by redirecting the user to the profile page.
Entry condition	**Registered User** logged into the system.
Exit conditions	**Registered User** navigates to other pages.

Use case name	**View Profile (Mobile)**

Participating Actors	Initiated by **Registered User**
Flow of events	1. **Registered User** clicks on the drawer icon which is on the top left corner. 2. **SYSTEM** responds by showing the drawer menu. 3. **Registered User** clicks on the Profile button in the drawer menu. 4. **SYSTEM** responds by redirecting the user to the profile screen.
Entry condition	**Registered User** is in the main screen.
Exit conditions	**Registered User** navigates to other screens.

Use case name	**Edit Profile (Mobile and Web)**
Participating Actors	Initiated by **Registered User**
Flow of events	1. **Registered User** clicks on the Edit icon on the top right corner in the Mobile application / Edit button in the web application. 2. **SYSTEM** responds by showing the edit profile form. 3. **Registered User** edits the one or more than one of the first name, last name, e-mail and profile image information that he/she wants to change and clicks on the save button. 4. **SYSTEM** responds by showing up a toast notification that says “Profile Information Saved.” and redirects the Registered User to the profile screen/page.
Entry condition	**Registered User** is in the profile screen/page.
Exit conditions	**Registered User** clicks on the save button or navigates to other screens/pages.
Exceptions	3a. **Registered User** does not fill one or more than one required input. 4a. **SYSTEM** gives a feedback that says “Please fill all the required information.”.

3b. **Registered User** fill one or more than one input in wrong format.

4b. **SYSTEM** gives a feedback that says “Please check your inputs.”

Use case name	**Manage User Preferences (Mobile)**
Participating Actors	Initiated by **Registered User**
Flow of events	1. **Registered User** clicks on the drawer icon in the top left corner. 2. **SYSTEM** responds by showing the drawer menu. 3. **Registered User** clicks on the user preferences button. 4. **SYSTEM** responds by redirecting the Registered User to the user preferences form. 5. **Registered User** indicates his/her e-mail and Mobile notification preferences and clicks on the Save button. 6. **SYSTEM** responds by showing up a toast notification that says “User Preferences Saved.” and redirects the Registered User to the main screen.
Entry condition	**Registered User** is in the main screen.
Exit conditions	**Registered User** clicks on the save button or navigates to other screens.

Use case name	**Manage User Preferences (Web)**

Participating Actors	Initiated by **Registered User**
Flow of events	1. **Registered User** clicks on the User Preferences button in the side bar menu. 2. **SYSTEM** responds by redirecting the Registered User to the user preferences form. 3. **Registered User** edits the user preferences form and clicks on the save button. 4. **SYSTEM** responds by showing up a toast notification that says “User Preferences Saved.” and redirects the Registered User to the main screen.
Entry condition	**Registered User** is logged in.
Exit conditions	**Registered User** clicks on the save button or navigates to other screens/pages.

Use case name	**Create Project (Web)**
Participating Actors	Initiated by **Registered User**
Flow of events	1. **Registered User** clicks on the Create Project button on the projects page of the Tasky web application. 2. **SYSTEM** responds by showing the project creation form. 3. **Registered User** enters “project name”, “project description” and adds participants (not required). 4. **SYSTEM** validates the form, creates the project and sends invites if any participants have been added.
Entry condition	**Registered User** is in projects page of the web application.
Exit conditions	**Registered User** clicks to the cancel button or navigates to another page.

Use case name	**Update Project (Web)**

Participating Actors	Initiated by **Project Manager**
Flow of events	1. **Project Manager** clicks on the Detail tab of the project that he/she is the manager of. 2. **SYSTEM** responds by redirecting the Project Manager to the detail tab. 3. **Project Manager** clicks on the Edit button and changes the field(s) that he/she wants to update and clicks to Save button. 4. **SYSTEM** validates the Edit form and project is updated.
Entry condition	**Project Manager** is logged in.
Exit conditions	**Project Manager** clicks on the cancel button or navigates to other screens/pages.

Use case name	**Delete Project (Web)**
Participating Actors	Initiated by **Project Manager**
Flow of events	1. **Project Manager** clicks on the Detail tab of the project that he/she is the manager of. 2. **SYSTEM** responds by redirecting the Project Manager to the detail tab. 3. **Project Manager** clicks on the Delete button and confirms. 4. **SYSTEM** deletes the project.
Entry condition	**Project Manager** is logged in.
Exit conditions	**Project Manager** clicks on the cancel button or navigates to other screens/pages.

Use case name	**Invite Participant to the Project (Web)**

Participating Actors	Initiated by **Project Manager,** Communicates with **Registered User and Guest**
Flow of events	1. **Project Manager** clicks on the Projects button in the side bar menu. 2. **SYSTEM** responds by redirecting the Project Manager to Project List page. 3. **Project Manager** double clicks on the project name**.** 4. **SYSTEM** responds by redirecting the Project Manager to the project detail page. 5. **Project Manager** clicks on the Invite Participant button. 6. **SYSTEM** responds by showing up a participant invitation form. 7. **Project Manager** writes e-mail of the users that he/she wants to invite to project and clicks on the Send Invitation button. 8. **SYSTEM** responds by showing up an acknowledgment that says "Invitation Sent to The Users." and redirects the Project Manager to the project detail page. 9. **Invited Users (Registered User or Guest)** receives an e-mail notification.
Entry condition	**Project Manager** is logged in to the system.
Exit conditions	**Project Manager** clicks on the Send Invitation button or navigates to other pages.
Exceptions	7a. **Project Manager** clicks on the Send Invitation button without writing e-mail. 8a. **SYSTEM** gives a feedback that says "Please fill the e-mail input.". 7b. **Registered User** writes e-mail in wrong format. 8b. **SYSTEM** gives a feedback that says "Please check your e-mail input."
Use case name	**Remove Participant (Web)**

Participating Actors	Initiated by **Project Manager,** Communicates with **Project Member and Watcher**
Flow of events	1. **Project Manager** clicks on the remove icon which is next to name of the project participant in the project detail page. 2. **SYSTEM** shows a warning box that asks “Are you sure to remove this participant ?”. 3. **Project Manager** clicks on the Yes button. 4. **SYSTEM** responds by showing up a toast notification that says “Participant is removed from the project” and redirects the Project Manager to the project detail page. 5. **Project Member / Watcher** which is removed from the project receives a notification.
Entry condition	**Project Manager** is in the project detail page.
Exit conditions	**Project Manager** clicks on the Yes or No button in warning box or navigates to other pages.
Exceptions	3a. **Project Manager** clicks on the No button. 4a. **SYSTEM** closes the warning box.

Use case name	**Accept/Decline Invitation (For Registered Users)**

Participating Actors	Initiated by **Registered User,** Communicates with **Project Manager**
Flow of events	1. **Registered User** receives an invitation mail and clicks on the url in the mail. 2. **SYSTEM** redirects the **Registered User** to the invitation page. 3. **Registered User** views the project information and clicks on Accept or Decline button. 4. **SYSTEM** redirects the **Registered User** to the main page. 5. **Project Manager** receives a notification about the result of the invitation.
Entry condition	**Registered User** gets an invitation and views the invitation mail.
Exit conditions	**Registered User** accept/decline the invitation or navigates to other pages.
Exceptions	1a. **Registered User** views the notifications and clicks on the project invitation notification.

Use case name	**Accept/Decline Invitation (For Guests)**

Participating Actors	Initiated **Guest,** Communicates with **Project Manager**
Flow of events	1. **Guest** receives an invitation mail and clicks on the url in the mail. 2. **SYSTEM** redirects the **Guest** to the registration form. 3. **Guest** registers and logs into the system. 4. **SYSTEM** redirects the user to the invitation page. 5. **Guest** views the project information and clicks on Accept or Decline button. 6. **SYSTEM** redirects the **Guest** to the main page. 7. **Project Manager** receives a notification about the result of the invitation.
Entry condition	**Guest** gets an invitation and views the invitation mail.
Exit conditions	**Guest** does not register to the system or accept/decline the invitation or navigates to other pages.

Use case name	**View Project Detail (Web + Mobile)**
Participating Actors	Initiated by **Project Participant**
Flow of events	1. **Project Participant** clicks on the project's name that he wants view details in the projects page/screen. 2. **SYSTEM** responds by redirecting the Project Participant to Project page. 3. **Project Participant** clicks on the Details tab on the project page/screen**.** 4. **SYSTEM** responds by redirecting the Project Participant to the project detail page.
Entry condition	**Project Participant** is logged into the system.
Exit conditions	**Project Participant** navigates to other pages/screens.

Use case name	**View Project Report (Web)**

Participating Actors	Initiated by **Project Manager**
Flow of events	1. **Project Manager** clicks on the project's name that he wants view Report of, in the projects page. 2. **SYSTEM** responds by redirecting the Project Manager to Project page. 3. **Project Manager** clicks on the Report tab on the project page**.** 4. **SYSTEM** responds by redirecting the Project Participant to the project report page.
Entry condition	**Project Participant** is logged into the system.
Exit conditions	**Project Participant** navigates to other pages/screens.

Use case name	**View Project Board (Web + Mobile)**
Participating Actors	Initiated by **Project Participant**
Flow of events	1. **Project Participant** clicks on the project's name on projects page/screen. 2. **SYSTEM** responds by redirecting the Project Participant to Project page. 3. **Project Participant** clicks on the Board tab on the project page/screen**.** 4. **SYSTEM** responds by redirecting the Project Participant to the project board page/screen.
Entry condition	**Project Participant** is logged into the system.
Exit conditions	**Project Participant** navigates to other pages/screens.

Use case name	**Create Task (Web)**

Participating Actors	Initiated by **Project Manager or Team Member,** Communicates with **Team Member**
Flow of events	1. **Project Manager/Team Member** clicks on the Create Task button in the project detail. 2. **SYSTEM** redirects the **Project Manager/Team Member** to the new task form. 3. **Project Manager/Team Member** indicates name, description, priority, label, due date and assignee for the task and clicks on the create task button. 4. **SYSTEM** checks the given information and creates the task. Then SYSTEM redirects the user to the created task detail page. 5. **Team Member** that is selected as assignee of the task gets a notification.
Entry condition	**Project Manager/Team Member** is in the project detail page.
Exit conditions	**Project Manager/Team Member** clicks on the create task button or navigates to other pages.
Exceptions	3a. **Project Manager/Team Member** does not fill at least one required input. 4a. **SYSTEM** gives a feedback that says “Please fill all required inputs.” 3b. **Project Manager/Team Member** writes at least one input in wrong format. 4b. **SYSTEM** gives a feedback that says “Please check your inputs.”

Use case name	**Create Sub-Task (Web)**

Participating Actors	Initiated by **Project Manager or Team Member,** Communicates with **Team Member**
Flow of events	1. **Project Manager/Team Member** clicks on the Create Sub-Task button in the task detail. 2. **SYSTEM** redirects the **Project Manager/Team Member** to the new task form. 3. **Project Manager/Team Member** indicates name, description, priority, label, due date and assignee for the sub-task and clicks on the create sub-task button. 4. **SYSTEM** checks the given information and creates the task. Then SYSTEM redirects the user to the created task detail page. 5. **Team Member** that is selected as assignee of the task gets a notification.
Entry condition	**Project Manager/Team Member** is in the task detail page.
Exit conditions	**Project Manager/Team Member** clicks on the create sub-task button or navigates to other pages.
Exceptions	3a. **Project Manager/Team Member** does not fill at least one required input. 4a. **SYSTEM** gives a feedback that says “Please fill all required inputs.” 3b. **Project Manager/Team Member** writes at least one input in wrong format. 4b. **SYSTEM** gives a feedback that says “Please check your inputs.”

Use case name	**Delete Task (Web)**

Participating Actors	Initiated by **Project Manager or Team Member,** Communicates with **Team Member**
Flow of events	1. **Project Manager/Team Member** clicks on the Delete Task button in the task detail. 2. **SYSTEM** shows a warning box that asks “Are you sure to delete this task?”. 3. **Project Manager/Team Member** clicks on Yes button. 4. **SYSTEM** responds by showing up a toast notification that says “Task is deleted.” and redirects the **Project Manager/Team Member** to the project detail page. 5. **Team Member** who is the assignee of the deleted task gets a notification.
Entry condition	**Project Manager/Team Member** is in the task detail page.
Exit conditions	**Project Manager/Team Member** clicks on the Yes or No button in the warning box or navigates to other pages.
Exceptions	3a. **Project Manager/Team Member** clicks on No button. 4a. **SYSTEM** closes the warning box.

Use case name	**Edit Task (Web)**
Participating Actors	Initiated by **Project Manager or Team Member**
Flow of events	1. **Project Manager/Team Member** clicks on the Edit Task button in the task detail. 2. **SYSTEM** redirects the **Project Manager/Team Member** to the task form filled by information of the task. 3. **Project Manager/Team Member** edits one or more than one of the name, description, priority, label, due date and assignee information and clicks on the Save button.

	4. **SYSTEM** checks the given information and edits the task. Then SYSTEM shows a toast notification that says “Task is saved”.
Entry condition	**Project Manager/Team Member** is in the task detail page.
Exit conditions	**Project Manager/Team Member** clicks on the Save button or navigates to other pages.
Exceptions	3a. **Project Manager/Team Member** does not fill at least one required input. 4a. **SYSTEM** gives a feedback that says “Please fill all required inputs.” 3b. **Project Manager/Team Member** writes at least one input in wrong format. 4b. **SYSTEM** gives a feedback that says “Please check your inputs.”
Use case name	**View Task List (Web + Mobile)**
Participating Actors	Initiated by **Project Participant**
Flow of events	1. **Project Participant** clicks on the project’s name on projects page/screen. 2. **SYSTEM** responds by redirecting the Project Participant to Project page. 3. **Project Participant** clicks on the Tasks tab on the project page/screen**.** 4. **SYSTEM** responds by redirecting the Project Participant to the project task list page/screen.
Entry condition	**Project Participant** is logged into the system.
Exit conditions	**Project Participant** navigates to other pages/screens.

Use case name	**Filter Task List (Web + Mobile)**
Participating Actors	Initiated by **Project Participant**
Flow of events	1. **Project Participant** clicks on the project's name on projects page/screen. 2. **SYSTEM** responds by redirecting the Project Participant to Project page. 3. **Project Participant** clicks on the Tasks tab on the project page/screen**.** 4. **SYSTEM** responds by redirecting the Project Participant to the project task list page/screen. 5. **Project Participant** selects which filter(s) he wants to use for Filtering and clicks them. 6. **SYSTEM** responds by filtering the list according to selected options.
Entry condition	**Project Participant** is logged into the system.
Exit conditions	**Project Participant** navigates to other pages/screens.

Use case name	**Sort Task List (Web + Mobile)**
Participating Actors	Initiated by **Project Participant**
Flow of events	1. **Project Participant** clicks on the project's name on projects page/screen. 2. **SYSTEM** responds by redirecting the Project Participant to Project page. 3. **Project Participant** clicks on the Tasks tab on the project page/screen**.** 4. **SYSTEM** responds by redirecting the Project Participant to the project task list page/screen. 5. **Project Participant** clicks on the column that he wants to order by.

	6. **SYSTEM** responds by sorting the list according to clicked column.
Entry condition	**Project Participant** is logged into the system.
Exit conditions	**Project Participant** navigates to other pages/screens.

Use case name	**Log Work (Web + Mobile)**
Participating Actors	Initiated by **Team Member,** Communicates with **Team Member or Project Manager**
Flow of events	1. **Team Member** clicks on the Log Work button. 2. **SYSTEM** redirects the **Team Member** to Work Log Form page/screen. 3. **Team Member** start date, duration, description and file information and clicks on the Log Work button. 4. **SYSTEM** checks the given information and logs the work. Then **SYSTEM** shows a toast notification that says "Work is logged" and navigates the **Team Member** to task detail page/screen. 5. **Team Member/Project Manager** who is the reporter of the task gets a notification about the work log.
Entry condition	**Team Member** is in the task detail page/screen.
Exit conditions	**Team Member** clicks on the Log Work button or navigates to other pages/screens.
Exceptions	3a. **Team Member** does not fill at least one required input. 4a. **SYSTEM** gives a feedback that says "Please fill all required inputs." 3b. **Team Member** writes at least one input in wrong format. 4b. **SYSTEM** gives a feedback that says "Please check your inputs."

Use case name	**View Last Activities (Web)**
Participating Actors	Initiated by **Team Member or Project Manager or Watcher**
Flow of events	1. **Team Member/Project Manager/Watcher** logs into the system 2. **SYSTEM** redirects the user to the main page which contains last activities of user's projects.
Entry condition	-
Exit conditions	**Team Member/Project Manager/Watcher** navigates to other pages.

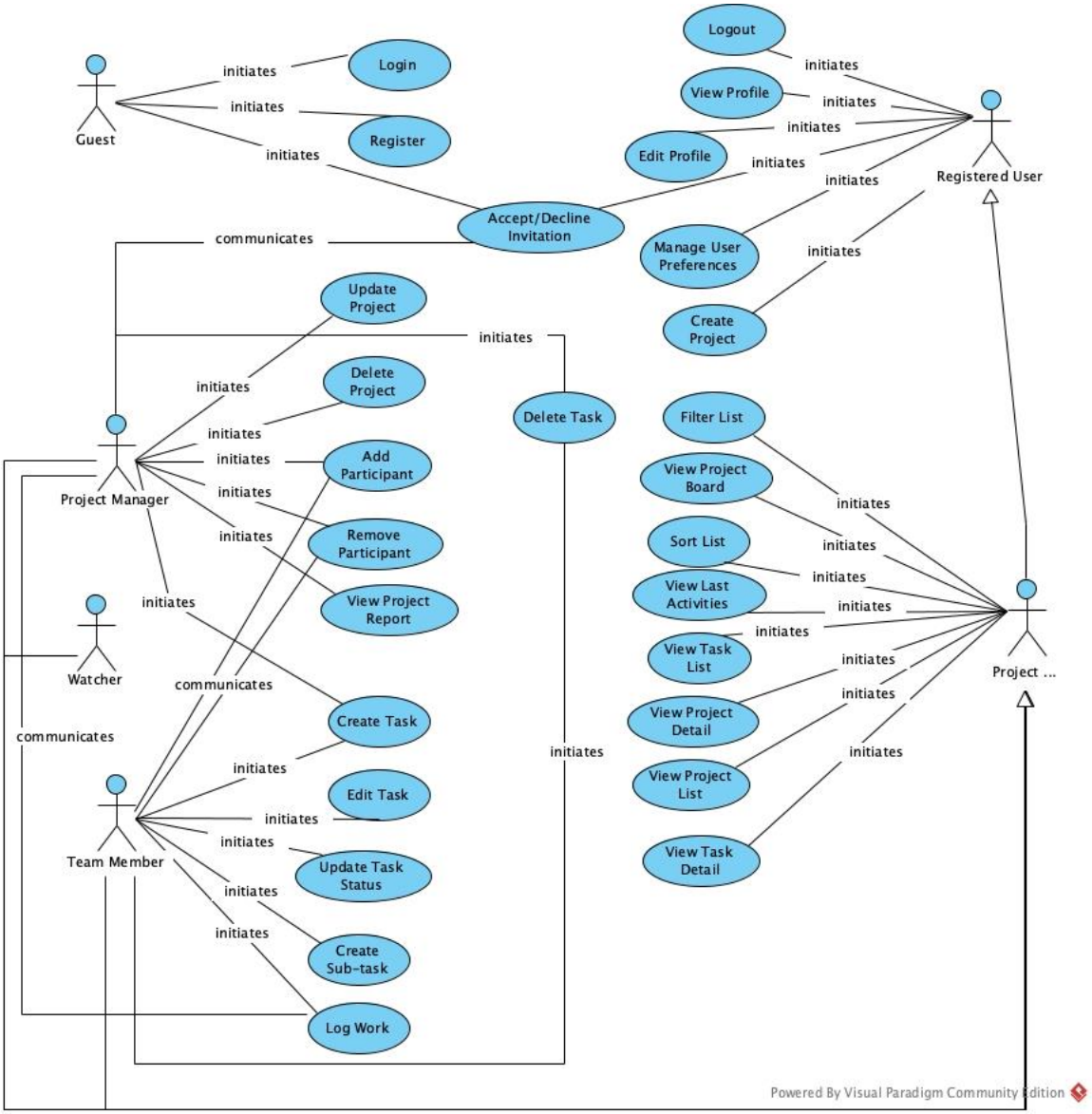
Use case name	**View Last Activities (Mobile)**
Participating Actors	Initiated by **Team Member or Project Manager or Watcher**
Flow of events	1. **Team Member/Project Manager/Watcher** clicks on the activities tab. 2. **SYSTEM** redirects the user to the activities screen which contains last activities of user's projects.
Entry condition	**Team Member/Project Manager/Watcher** is logged into the system.
Exit conditions	**Team Member/Project Manager/Watcher** navigates to other screens.

Use case name	**View Task Detail (Web + Mobile)**
Participating Actors	Initiated by **Team Member or Project Manager or Watcher**
Flow of events	1. **Team Member/Project Manager/Watcher** clicks on the name of the task in the task list. 2. **SYSTEM** redirects the **Team Member/Project Manager/Watcher** to task detail page/screen.
Entry condition	**Team Member** is in the task list page/screen.
Exit conditions	**Team Member** navigates to other pages/screens.

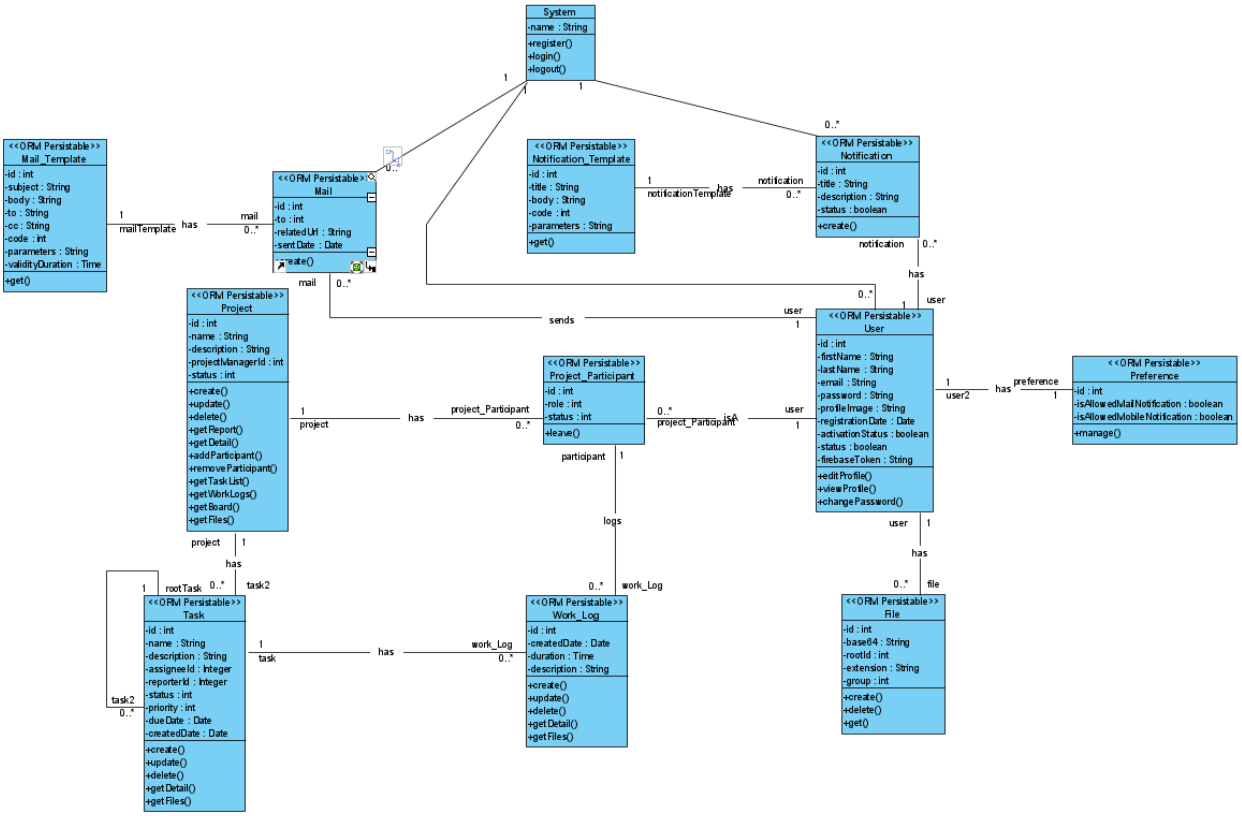
Use case name	**Update Task Status (Web)**
Participating Actors	Initiated by **Team Member or Project Manager**
Flow of events	1. **Team Member/Project Manager** clicks on the project board button in the side bar menu. 2. **SYSTEM** redirects the **Team Member/Project Manager** to project board page. 3. **Team Member/Project Manager** drags the card of the task and drops in the status column that he/she wants. 4. **SYSTEM** updates the status of the task and shows a toast notification that says “Task is updated.”.
Entry condition	**Team Member** is in the project detail page.
Exit conditions	**Team Member** navigates to other pages.

Use case name	**Update Task Status (Mobile)**
Participating Actors	Initiated by **Team Member or Project Manager**
Flow of events	1. **Team Member/Project Manager** clicks on the text status in the task detail. 2. **SYSTEM** opens a dropdown menu which contains possible statuses for tasks. 3. **Team Member/Project Manager** selects the new status of the task. 4. **SYSTEM** updates the status of the task and shows a toast notification that says “Task is updated.”.
Entry condition	**Team Member** is in the task detail page.
Exit conditions	**Team Member** navigates to other screens.

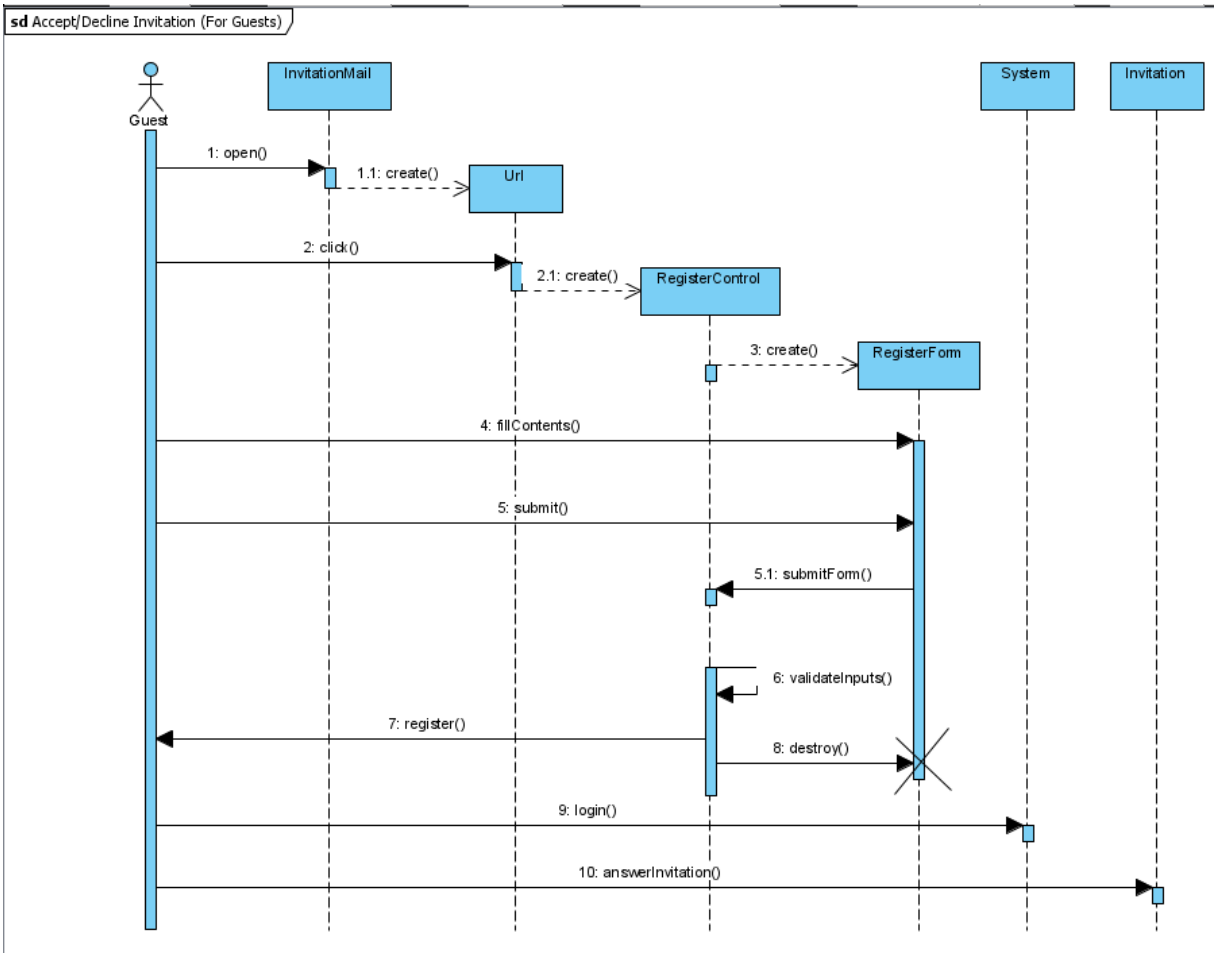
Use Case Model:

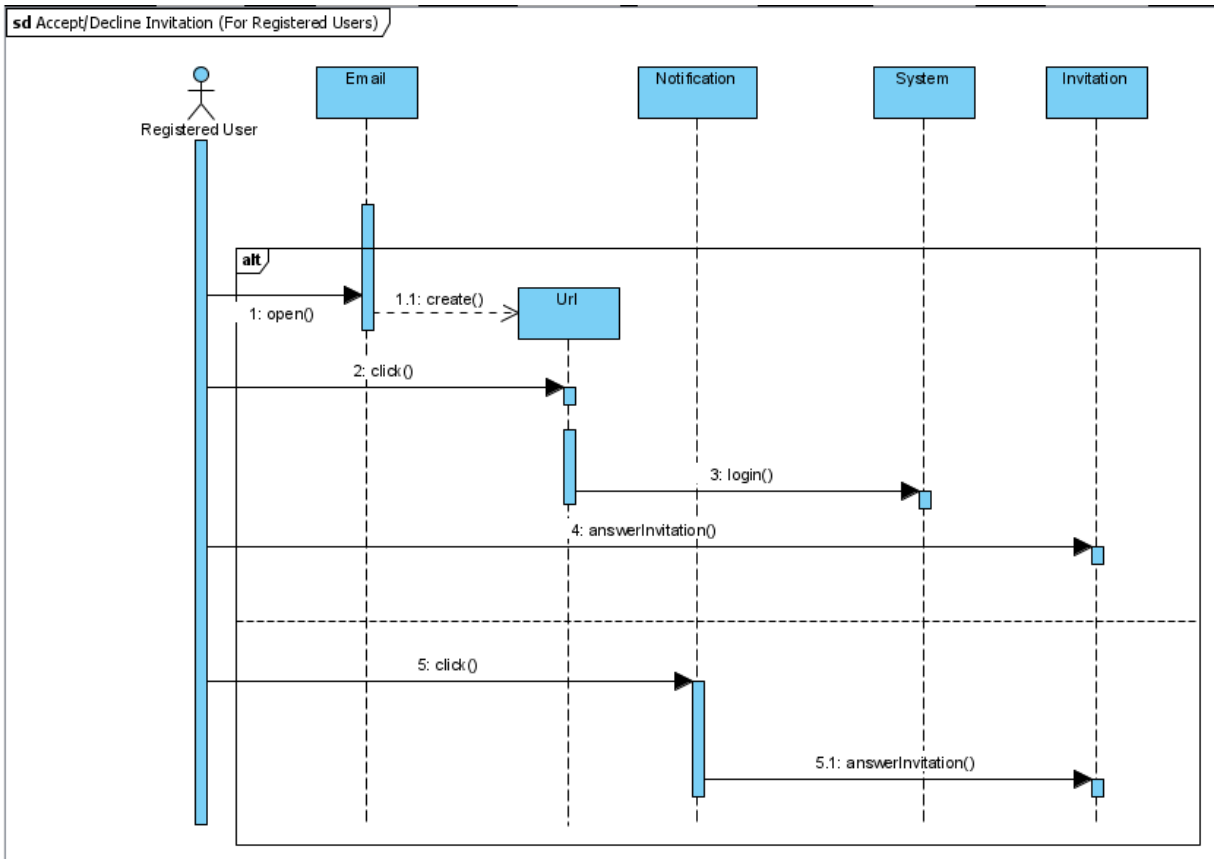


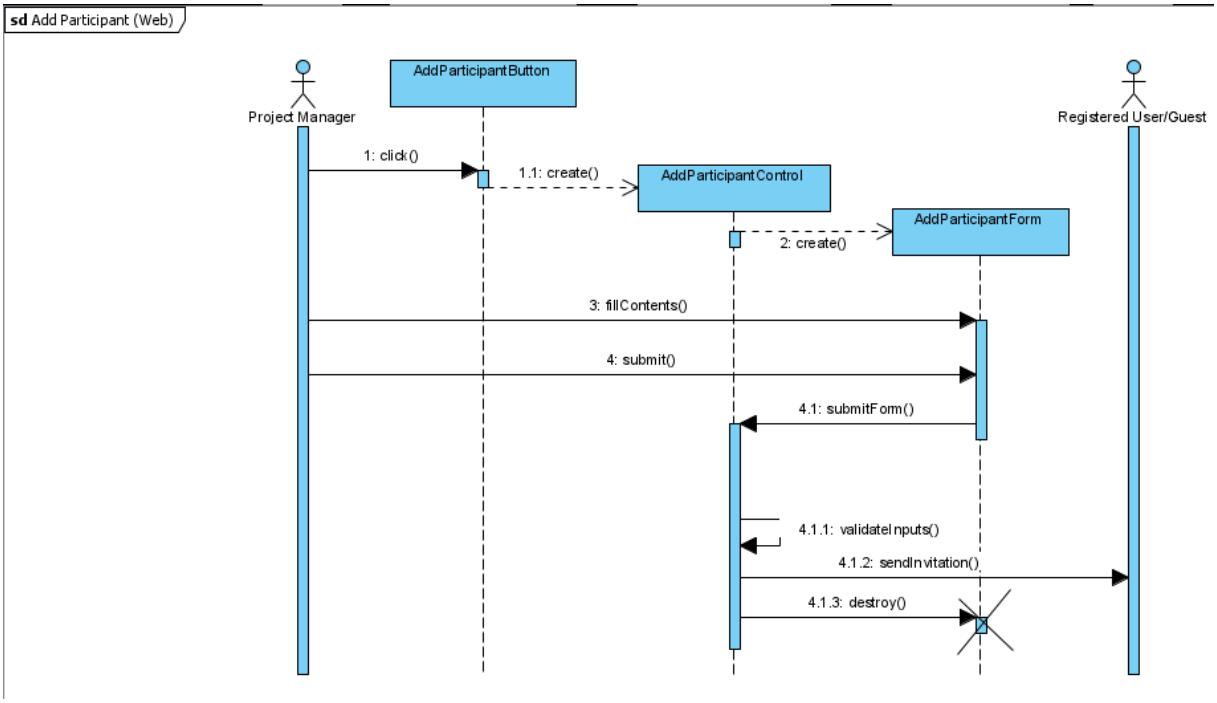
3.4.3 Object model

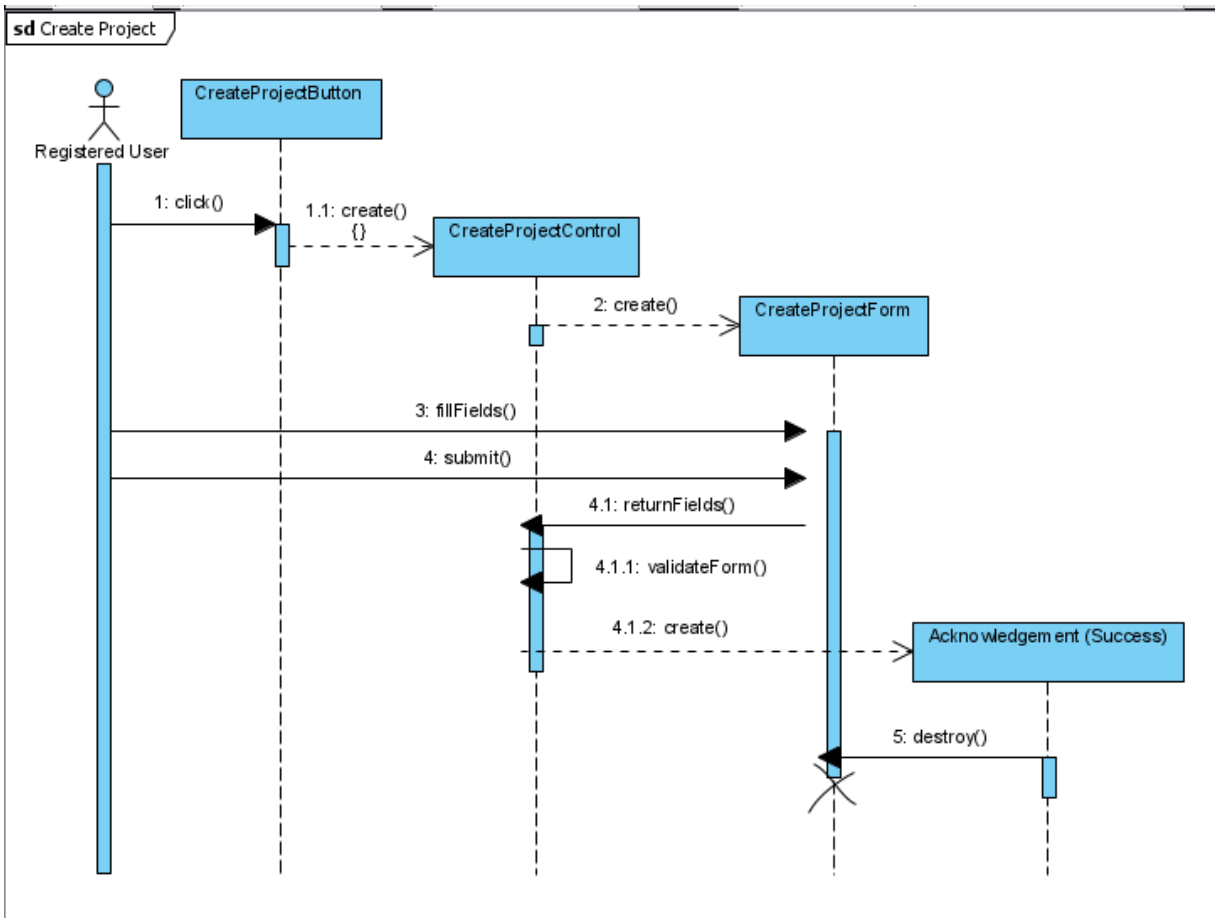


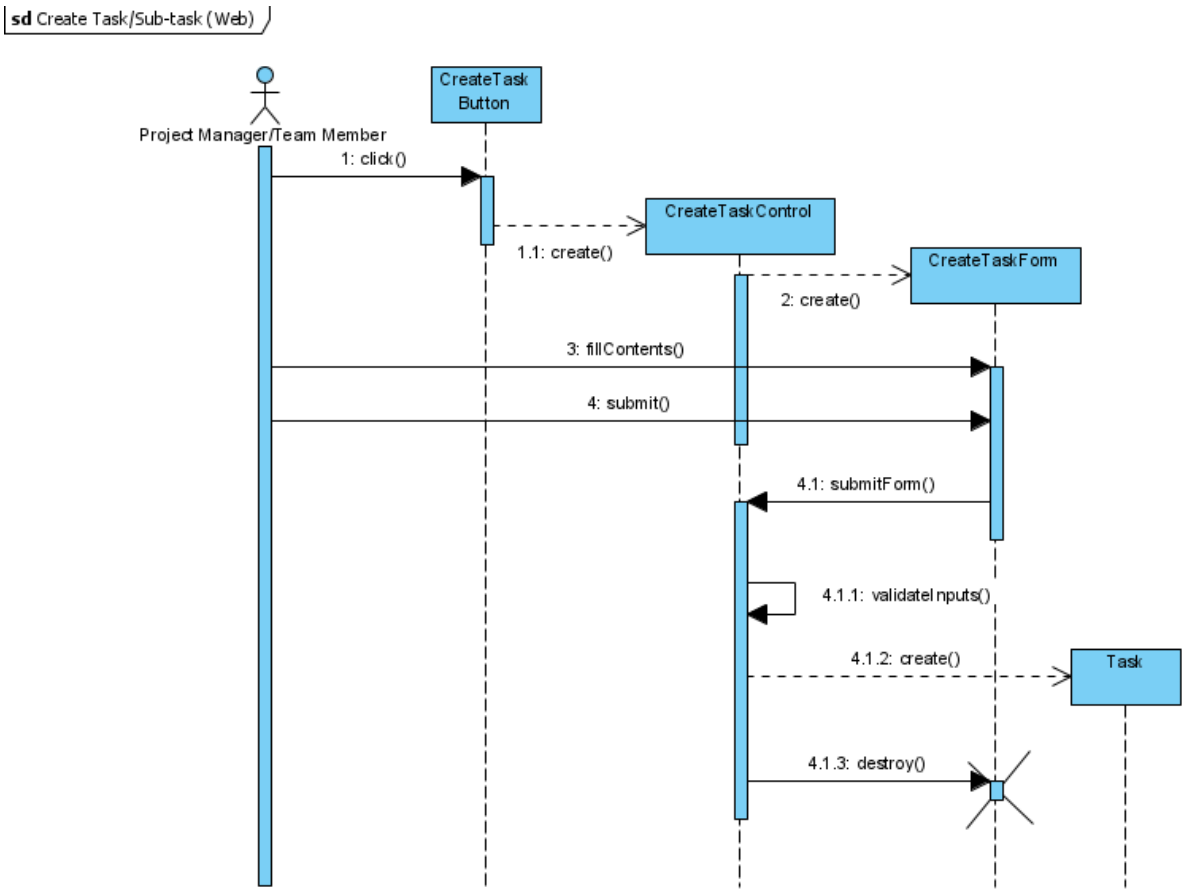
3.4.4 Dynamic model

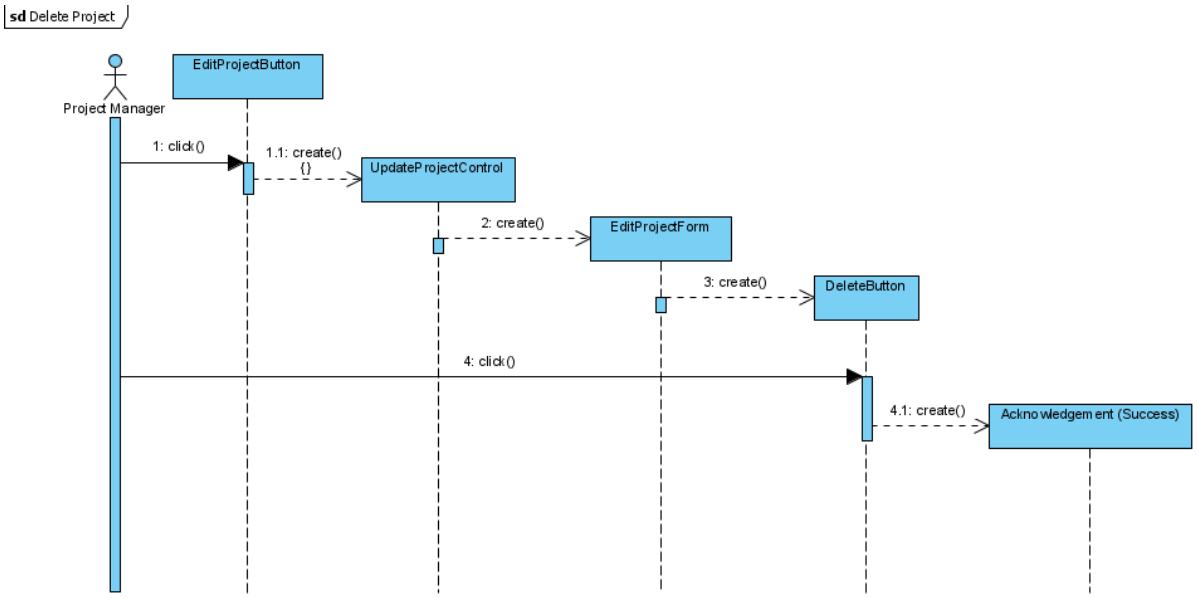


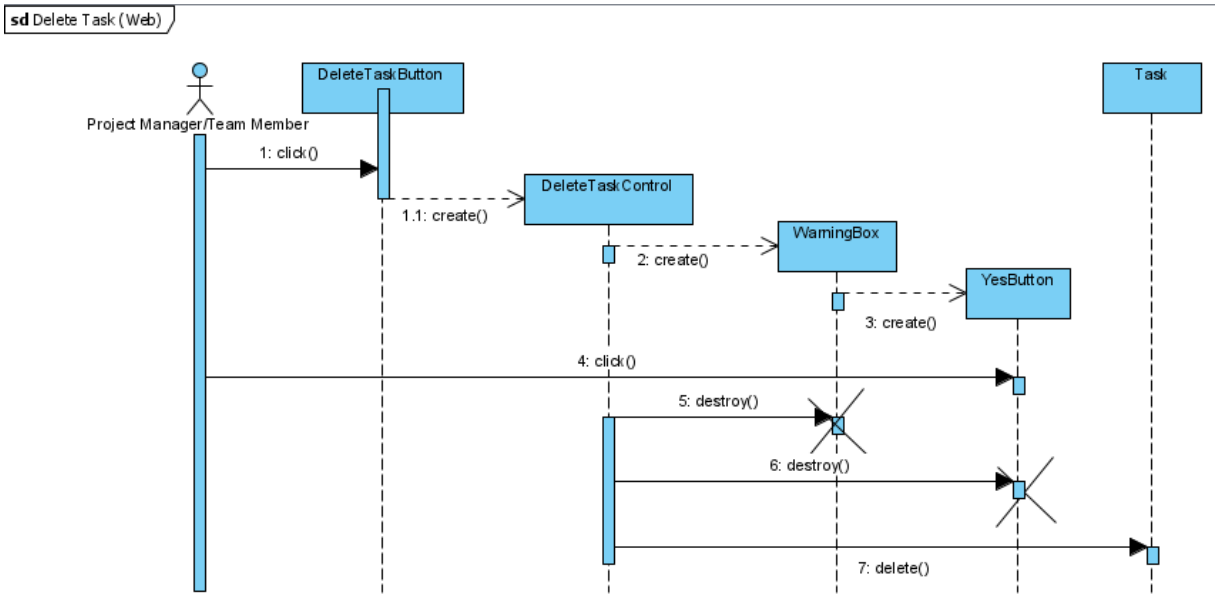

sd Accept/Decline Invitation (For Registered Users)
Registered User
Email
Notification
System
Invitation
alt
1: open()
1.1: create()
Url
2: click()
3: login()
4: answerInvitation()
5: click()
5.1: answerInvitation()

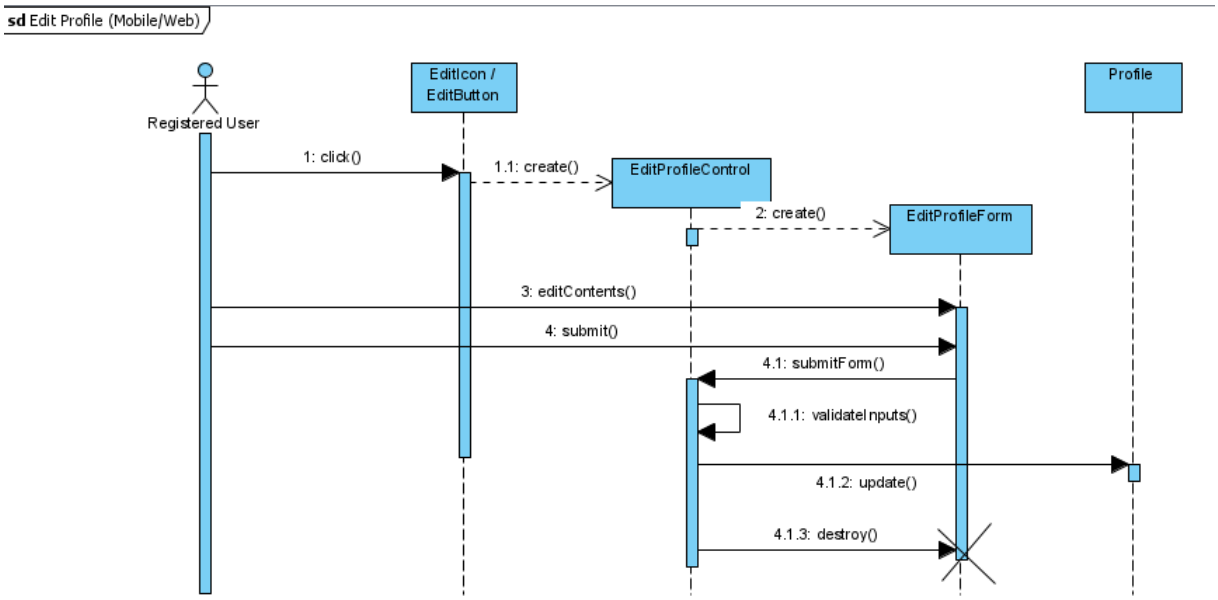

sd Add Participant (Web)
Project Manager
AddParticipantButton
Registered User/Guest
1: click()
1.1: create()
AddParticipantControl
AddParticipantForm
2: create()
3: fillContents()
4: submit()
4.1: submitForm()
4.1.1: validateInputs()
4.1.2: sendInvitation()
4.1.3: destroy()

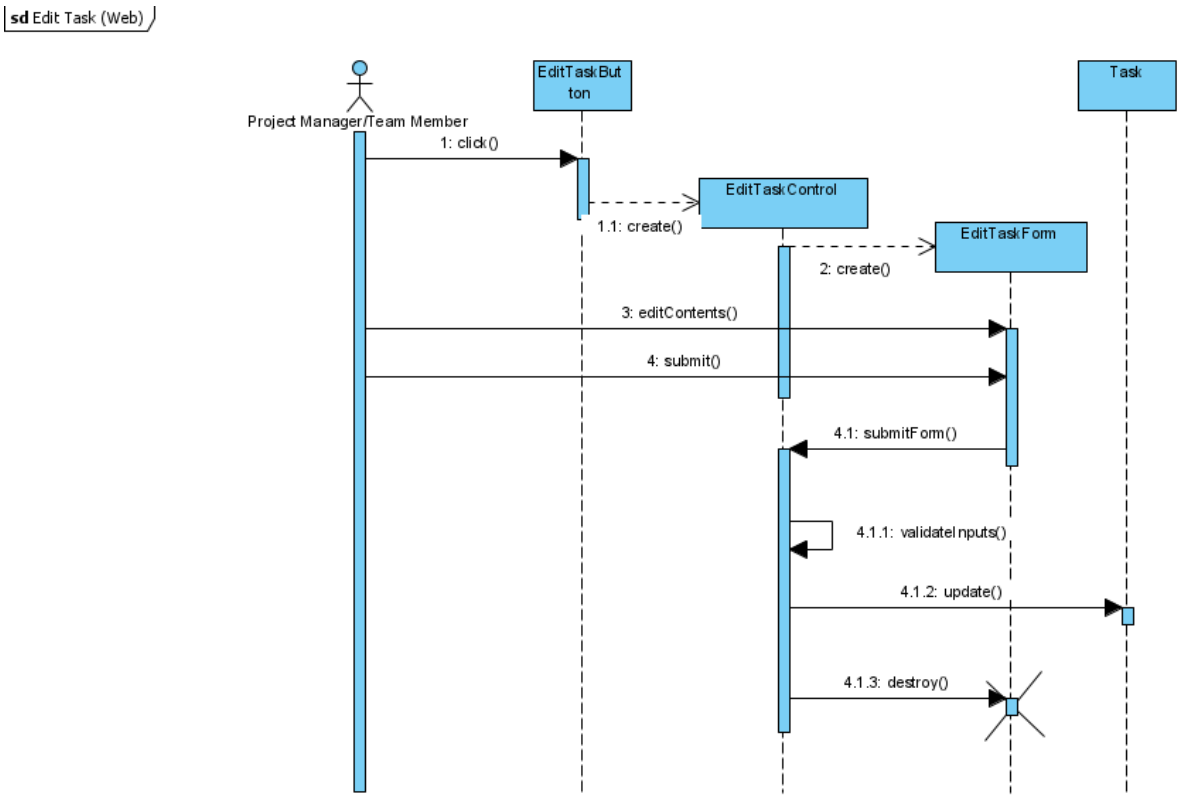

sd Create Project
Registered User
CreateProjectButton
1: click()
1.1: create()
{}
CreateProjectControl
2: create()
CreateProjectForm
3: fillFields()
4: submit()
4.1: returnFields()
4.1.1: validateForm()
4.1.2: create()
Acknowledgement (Success)
5: destroy()

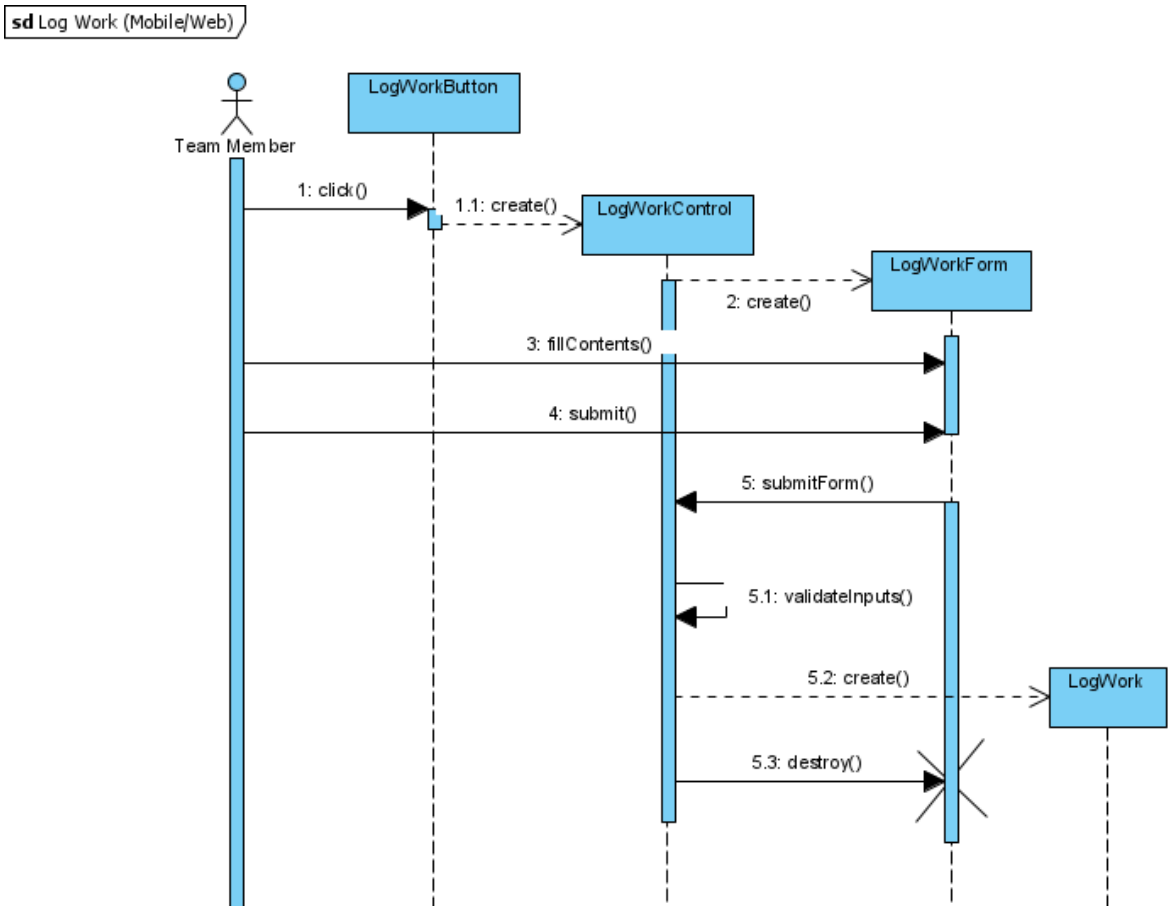

sd Create Task/Sub-task (Web)
Project Manager/Team Member
CreateTask Button
1: click()
CreateTaskControl
1.1: create()
CreateTaskForm
2: create()
3: fillContents()
4: submit()
4.1: submitForm()
4.1.1: validateInputs()
4.1.2: create()
Task
4.1.3: destroy()

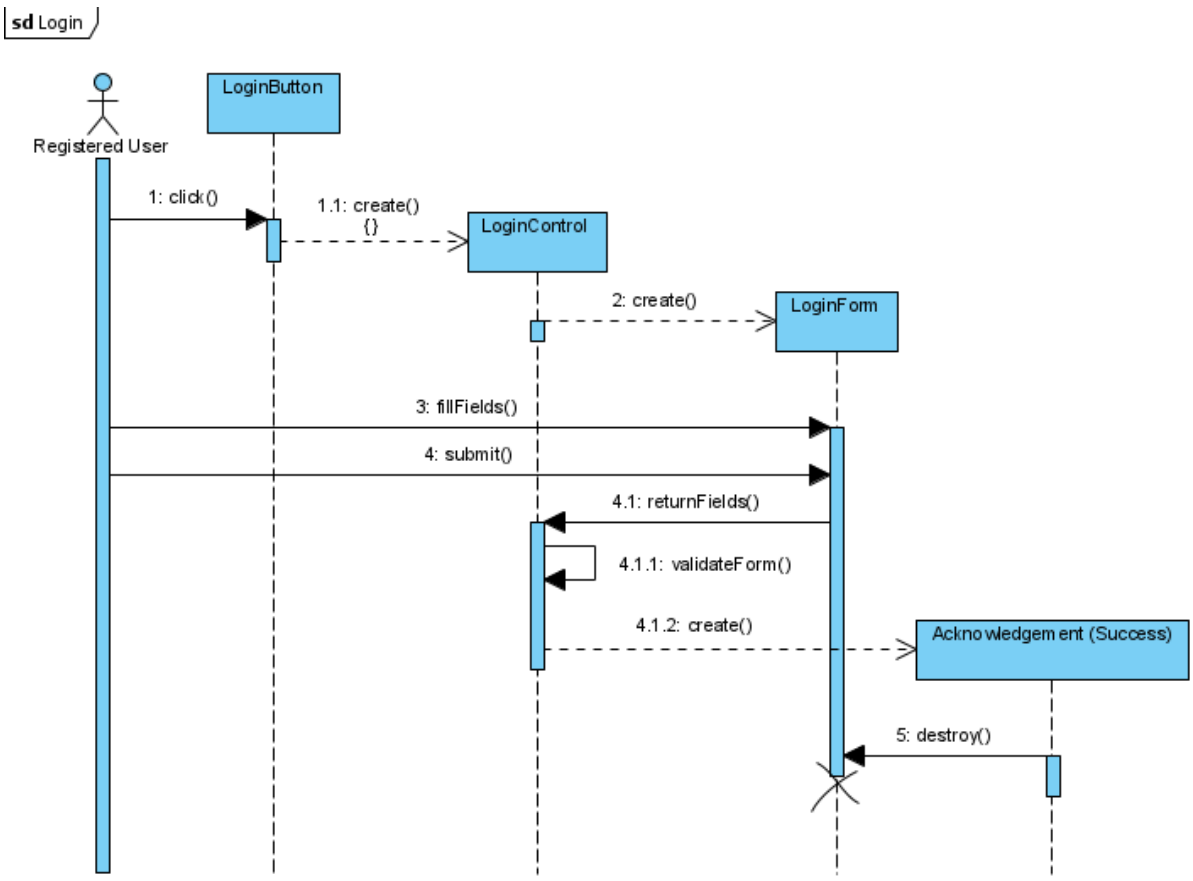

sd Delete Project
Project Manager
EditProjectButton
1: click()
1.1: create()
{}
UpdateProjectControl
2: create()
EditProjectForm
3: create()
DeleteButton
4: click()
4.1: create()
Acknowledgement (Success)

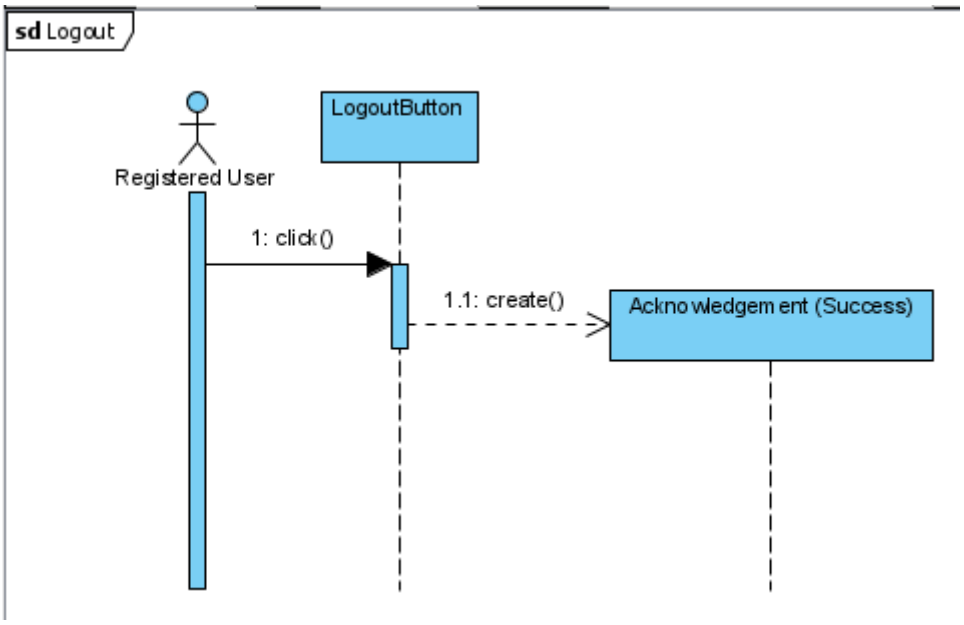

sd Delete Task (Web)
Project Manager/Team Member
DeleteTaskButton
Task
1: click()
1.1: create()
DeleteTaskControl
2: create()
WarningBox
3: create()
YesButton
4: click()
5: destroy()
6: destroy()
7: delete()

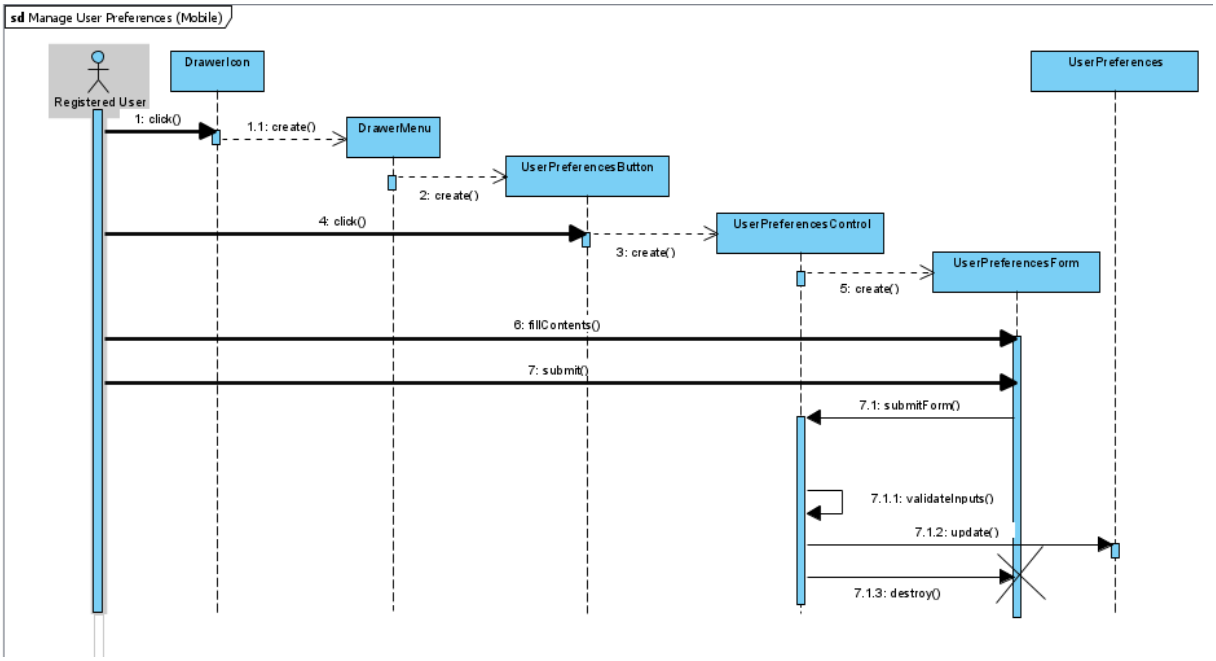

sd Edit Profile (Mobile/Web)
Registered User
EditIcon / EditButton
Profile
1: click()
1.1: create()
EditProfileControl
2: create()
EditProfileForm
3: editContents()
4: submit()
4.1: submitForm()
4.1.1: validateInputs()
4.1.2: update()
4.1.3: destroy()

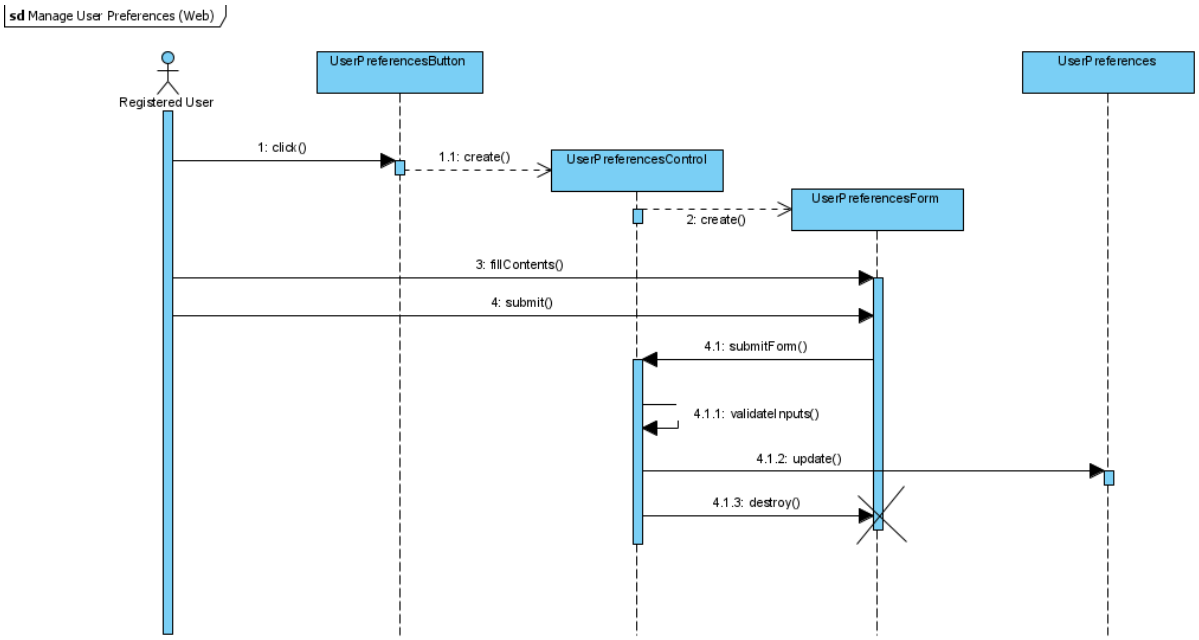

sd Edit Task (Web)
Project Manager/Team Member
EditTaskBut ton
Task
1: click()
EditTaskControl
1.1: create()
EditTaskForm
2: create()
3: editContents()
4: submit()
4.1: submitForm()
4.1.1: validateInputs()
4.1.2: update()
4.1.3: destroy()

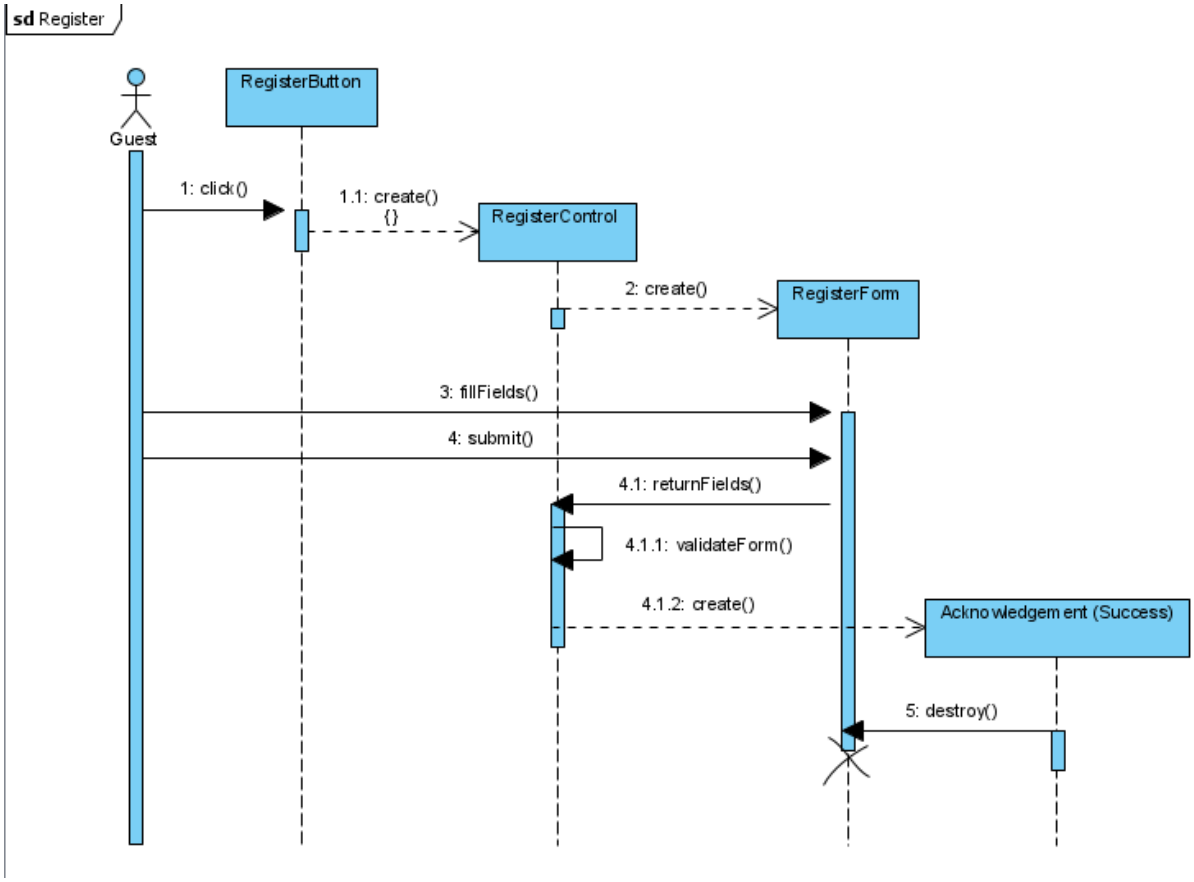

sd Log Work (Mobile/Web)
Team Member
LogWorkButton
LogWorkControl
LogWorkForm
LogWork
1: click()
1.1: create()
2: create()
3: fillContents()
4: submit()
5: submitForm()
5.1: validateInputs()
5.2: create()
5.3: destroy()

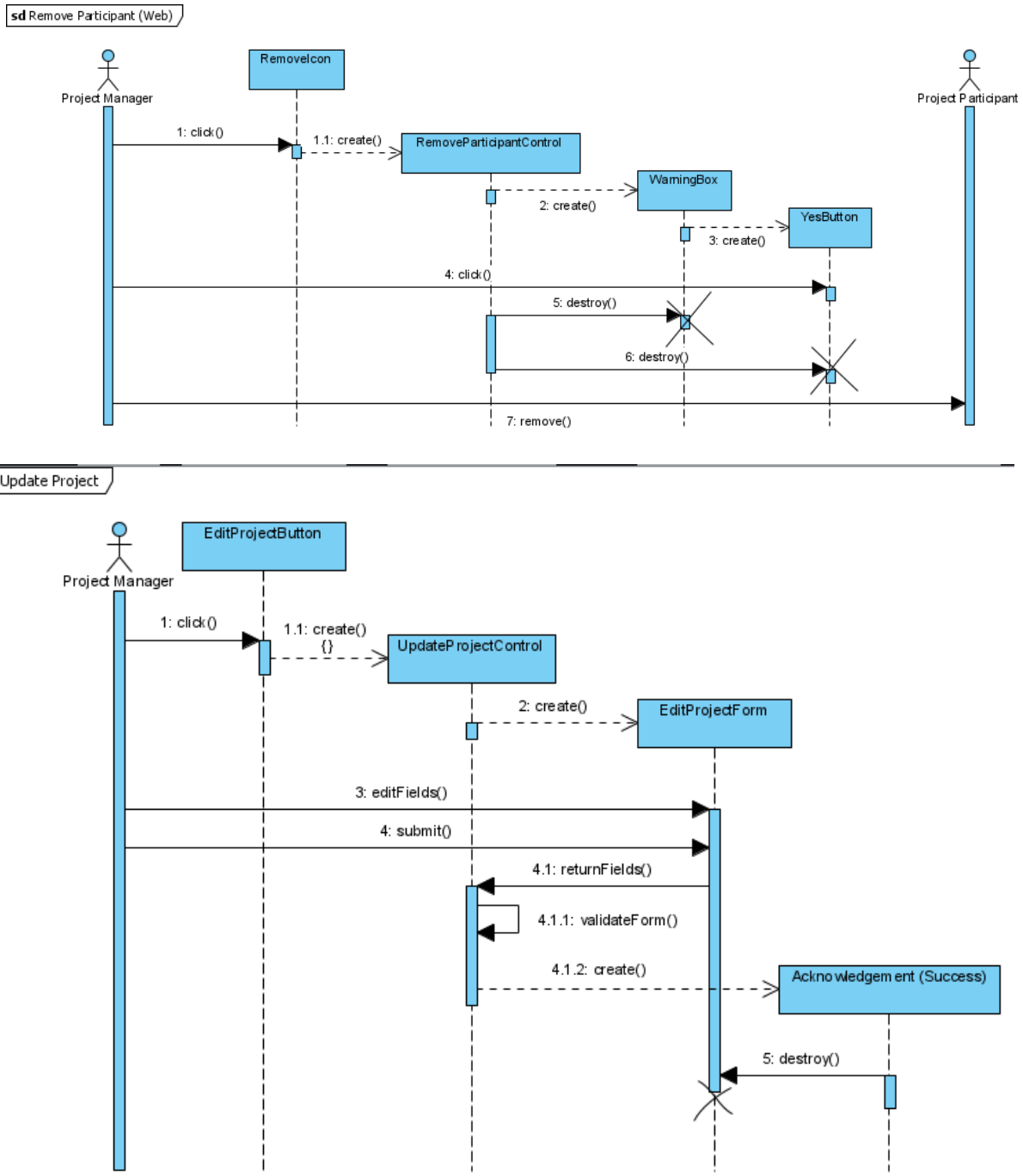

sd Login
Registered User
LoginButton
LoginControl
LoginForm
Acknowledgement (Success)
1: click()
1.1: create()
{}
2: create()
3: fillFields()
4: submit()
4.1: returnFields()
4.1.1: validateForm()
4.1.2: create()
5: destroy()

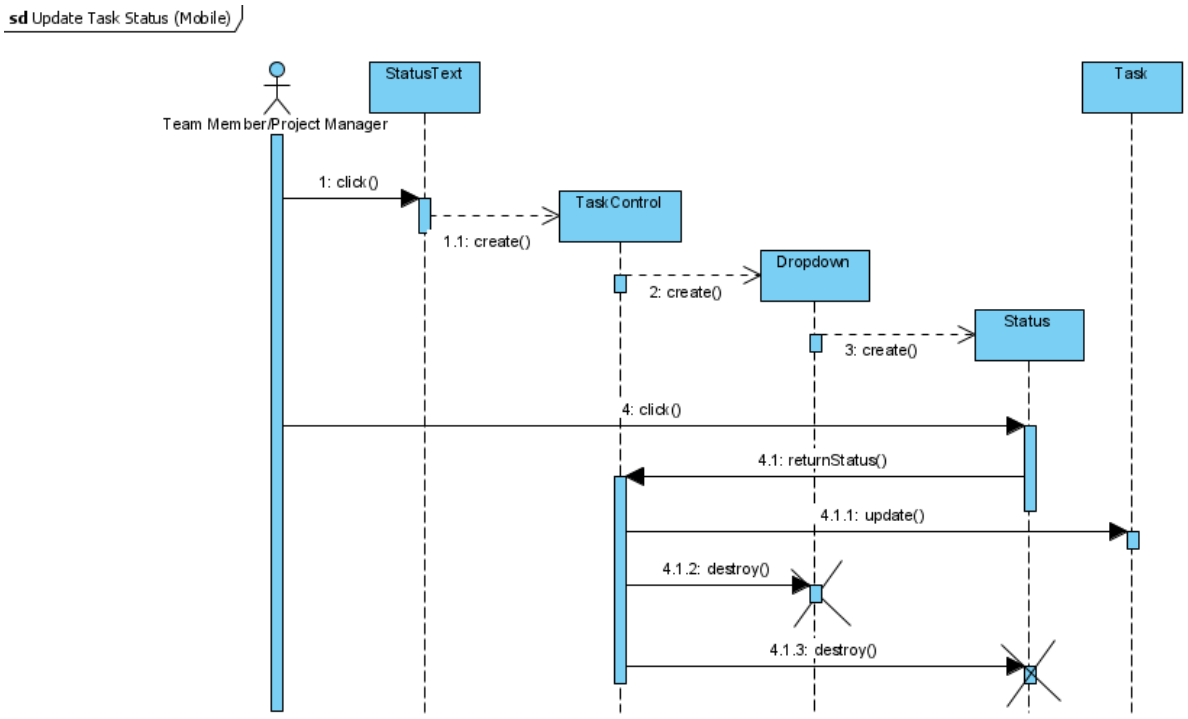

sd Logout
Registered User
LogoutButton
1: click()
1.1: create()
Acknowledgement (Success)

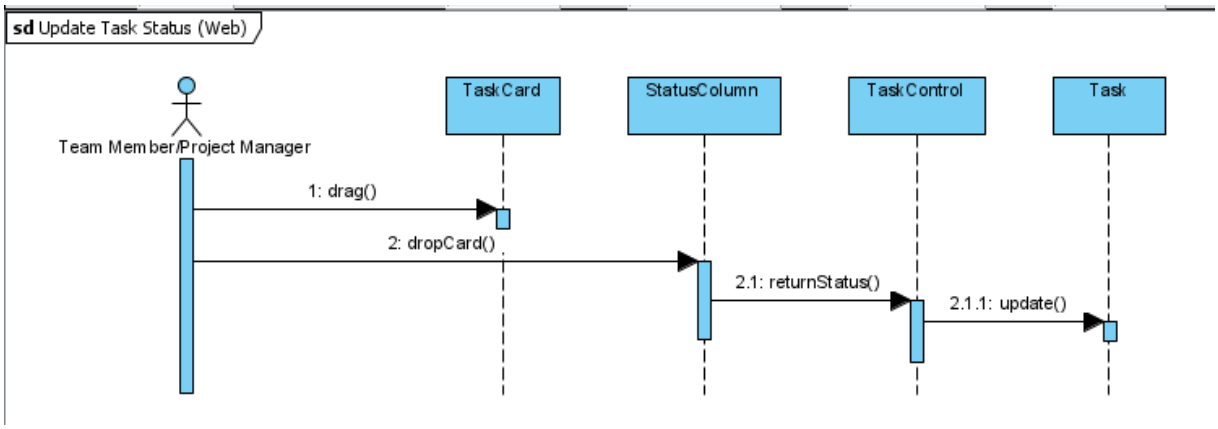

sd Manage User Preferences (Mobile)
Registered User
DrawerIcon
DrawerMenu
UserPreferencesButton
UserPreferencesControl
UserPreferencesForm
UserPreferences
1: click()
1.1: create()
2: create()
3: create()
4: click()
5: create()
6: fillContents()
7: submit()
7.1: submitForm()
7.1.1: validateInputs()
7.1.2: update()
7.1.3: destroy()

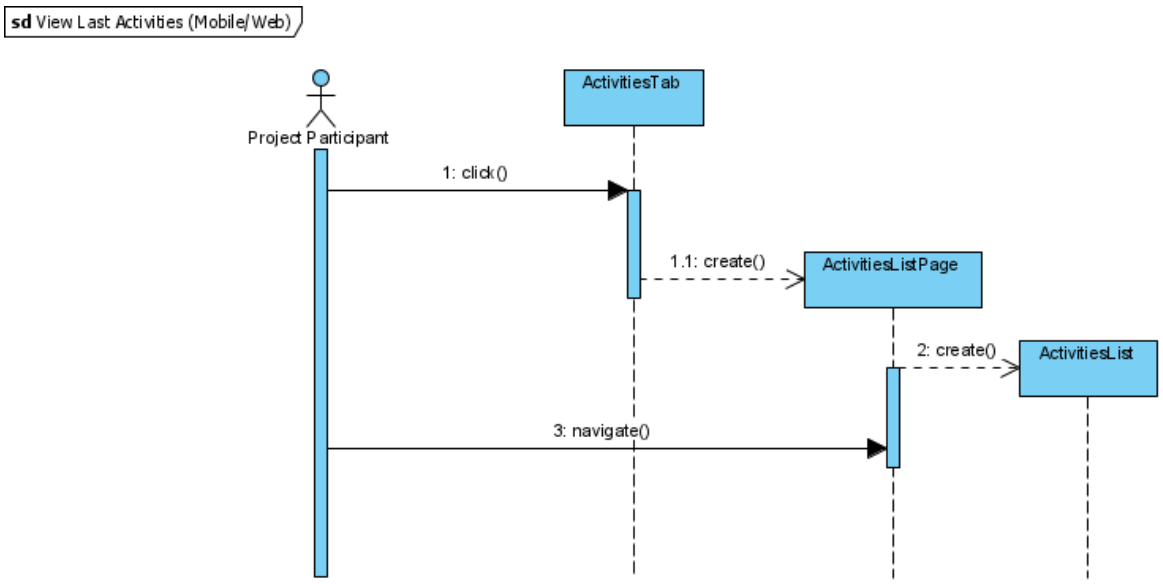

sd Manage User Preferences (Web)
Registered User
UserPreferencesButton
UserPreferences
1: click()
1.1: create()
UserPreferencesControl
UserPreferencesForm
2: create()
3: fillContents()
4: submit()
4.1: submitForm()
4.1.1: validateInputs()
4.1.2: update()
4.1.3: destroy()

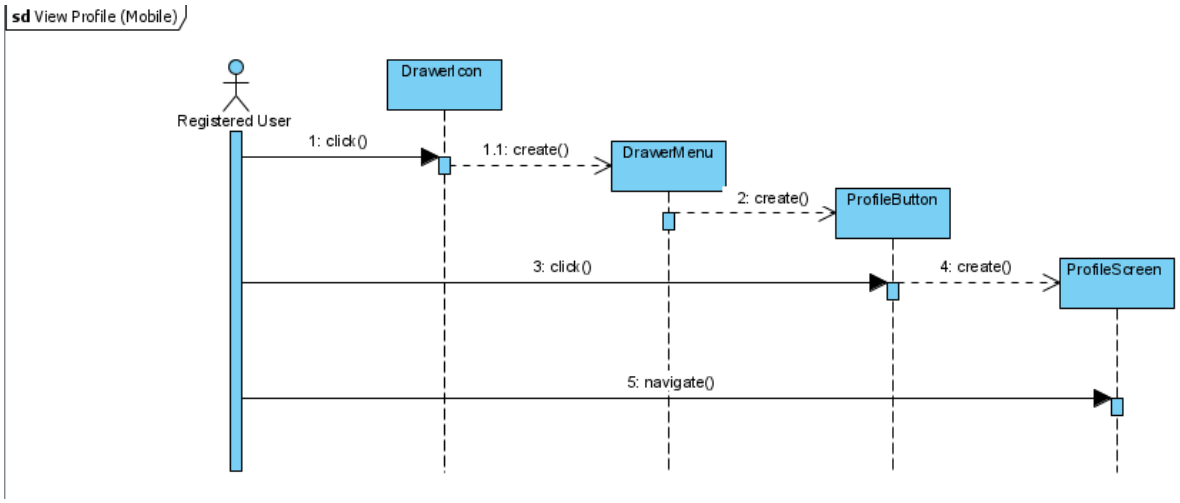

sd Register
Guest
RegisterButton
1: click()
1.1: create()
{}
RegisterControl
2: create()
RegisterForm
3: fillFields()
4: submit()
4.1: returnFields()
4.1.1: validateForm()
4.1.2: create()
Acknowledgement (Success)
5: destroy()

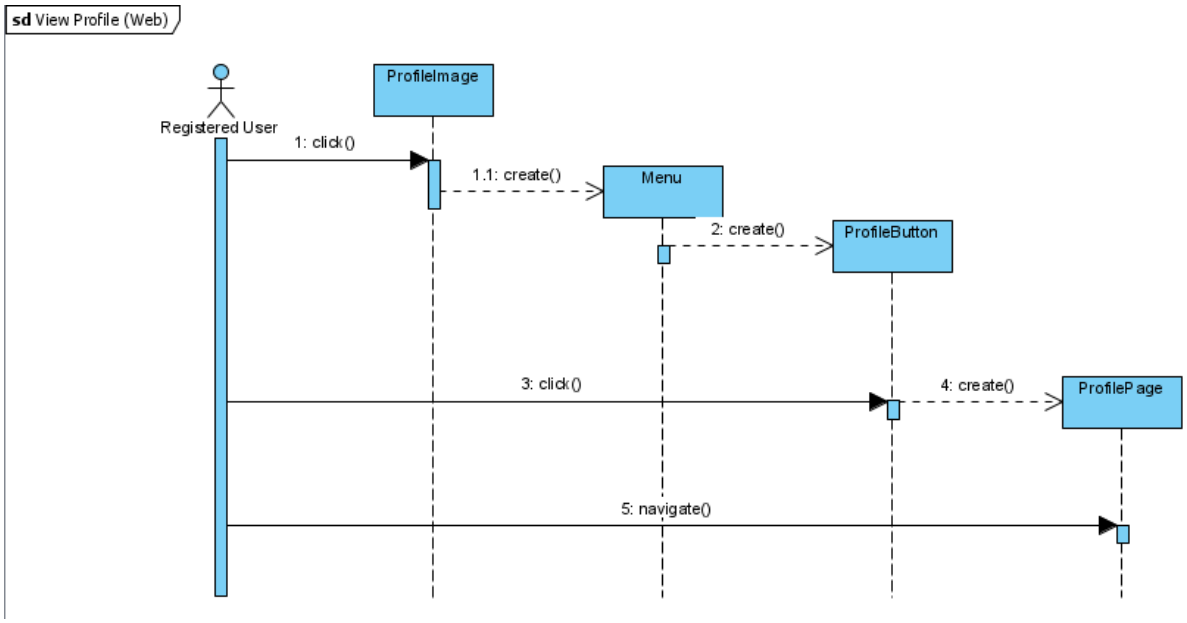

sd Remove Participant (Web)
Project Manager
RemoveIcon
Project Participant
1: click()
1.1: create()
RemoveParticipantControl
WarningBox
2: create()
YesButton
3: create()
4: click()
5: destroy()
6: destroy()
7: remove()
sd Update Project
Project Manager
EditProjectButton
1: click()
1.1: create()
{}
UpdateProjectControl
2: create()
EditProjectForm
3: editFields()
4: submit()
4.1: returnFields()
4.1.1: validateForm()
4.1.2: create()
Acknowledgement (Success)
5: destroy()

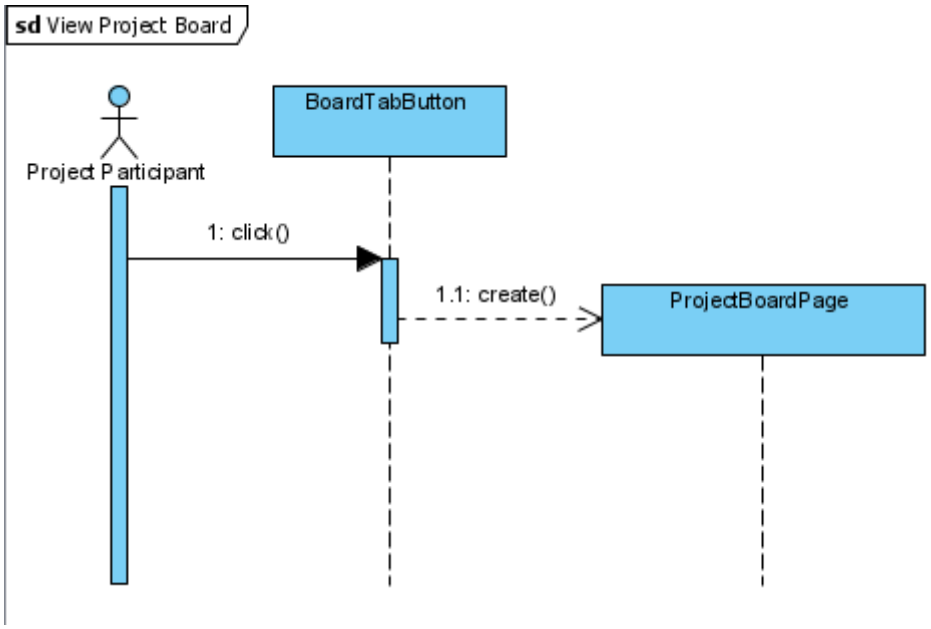

sd Update Task Status (Mobile)
Team Member/Project Manager
StatusText
TaskControl
Dropdown
Status
Task
1: click()
1.1: create()
2: create()
3: create()
4: click()
4.1: returnStatus()
4.1.1: update()
4.1.2: destroy()
4.1.3: destroy()

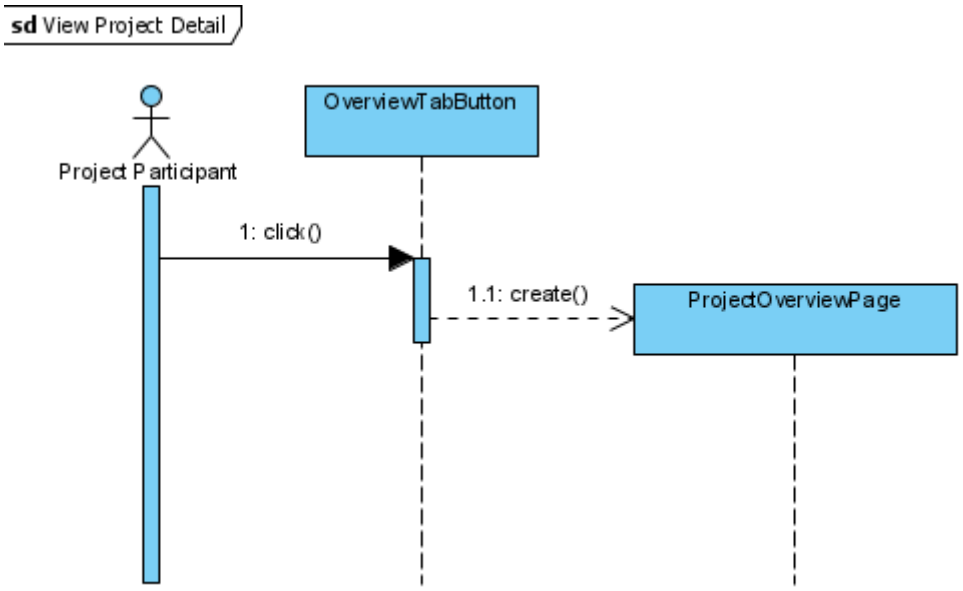

sd Update Task Status (Web)
Team Member/Project Manager
TaskCard
StatusColumn
TaskControl
Task
1: drag()
2: dropCard()
2.1: returnStatus()
2.1.1: update()

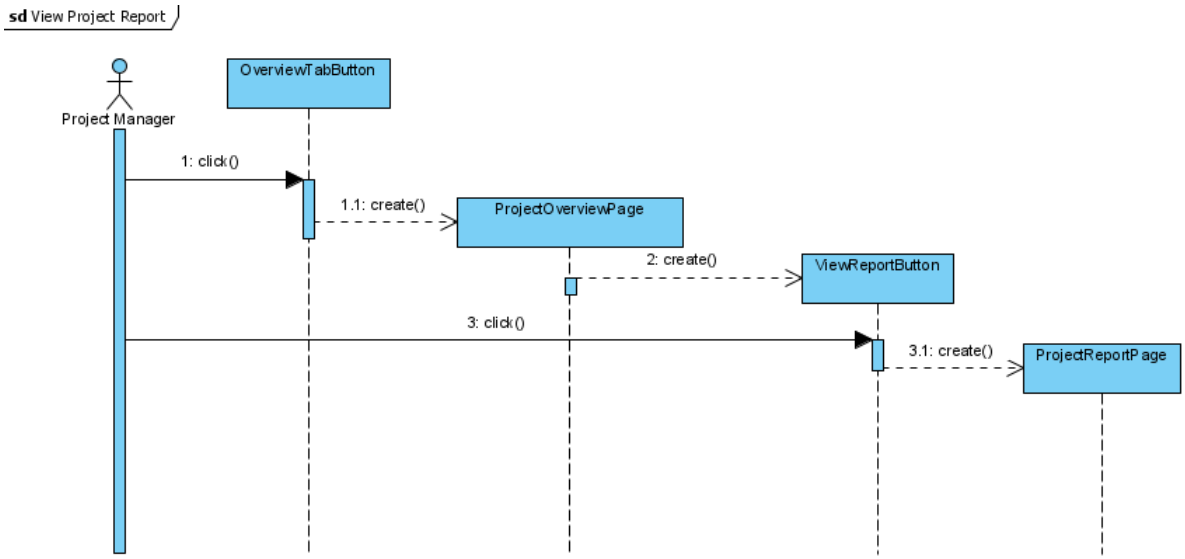

sd View Last Activities (Mobile/Web)
Project Participant
ActivitiesTab
ActivitiesListPage
ActivitiesList
1: click()
1.1: create()
2: create()
3: navigate()

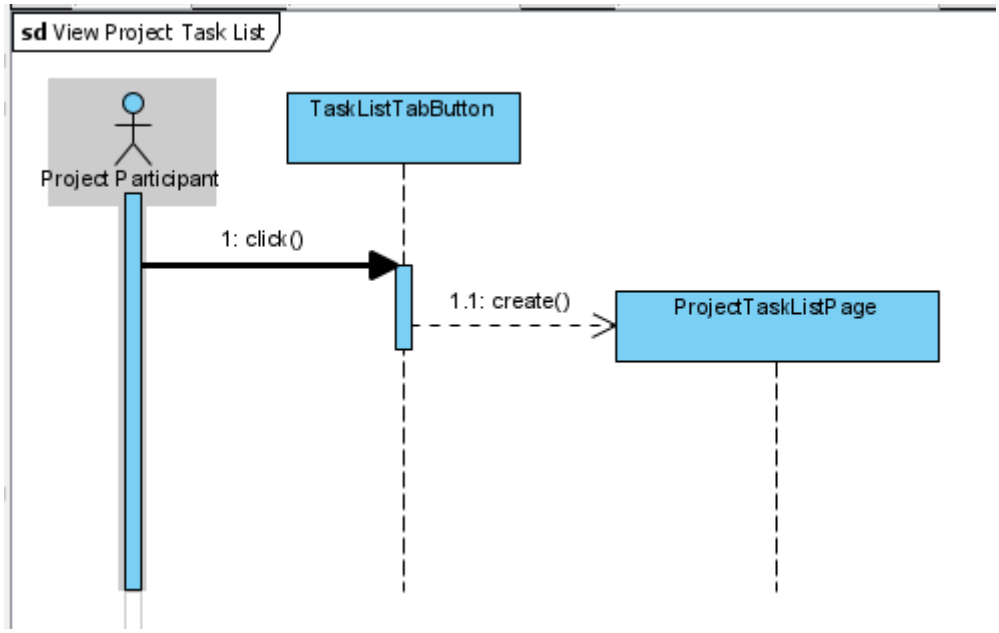

sd View Profile (Mobile)
Registered User
DrawerIcon
DrawerMenu
ProfileButton
ProfileScreen
1: click()
1.1: create()
2: create()
3: click()
4: create()
5: navigate()

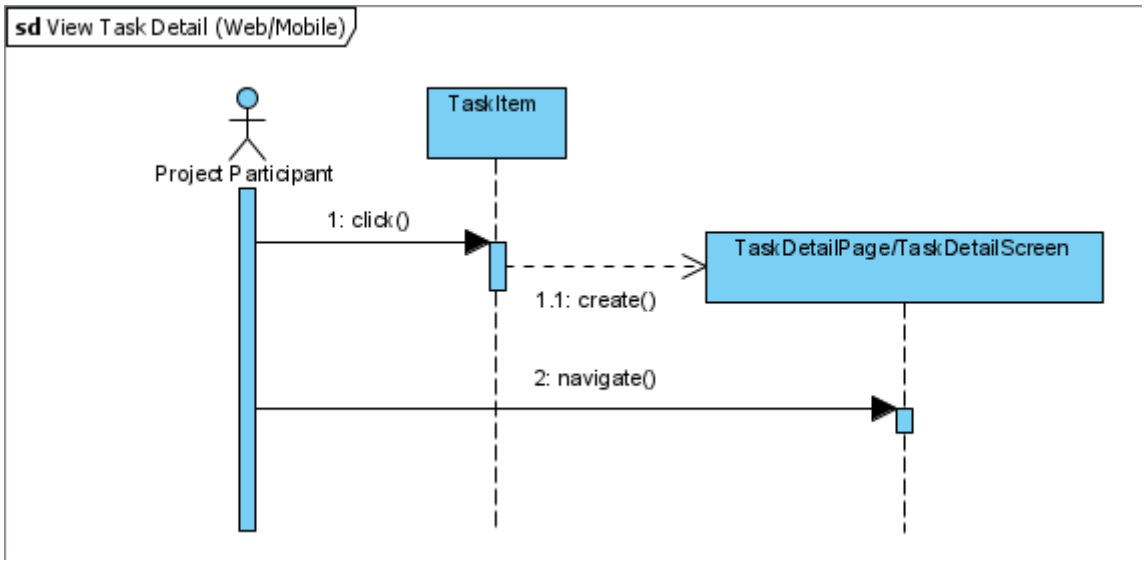

sd View Profile (Web)
Registered User
ProfileImage
Menu
ProfileButton
ProfilePage
1: click()
1.1: create()
2: create()
3: click()
4: create()
5: navigate()


sd View Project Board
Project Participant
BoardTabButton
ProjectBoardPage
1: click()
1.1: create()


sd View Project Detail
Project Participant
OverviewTabButton
1: click()
1.1: create()
ProjectOverviewPage


sd View Project Report
Project Manager
OverviewTabButton
1: click()
1.1: create()
ProjectOverviewPage
2: create()
ViewReportButton
3: click()
3.1: create()
ProjectReportPage

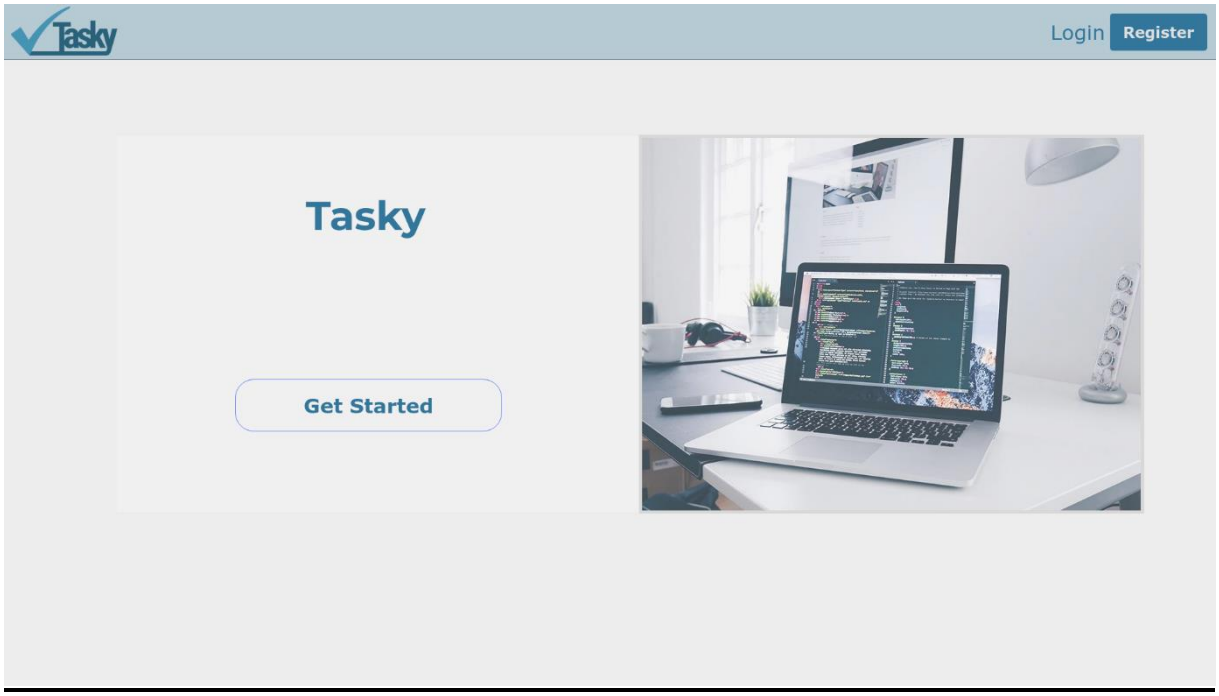

sd View Project Task List
Project Participant
TaskListTabButton
1: click()
1.1: create()
ProjectTaskListPage


sd View Task Detail (Web/Mobile)
Project Participant
TaskItem
1: click()
TaskDetailPage/TaskDetailScreen
1.1: create()
2: navigate()

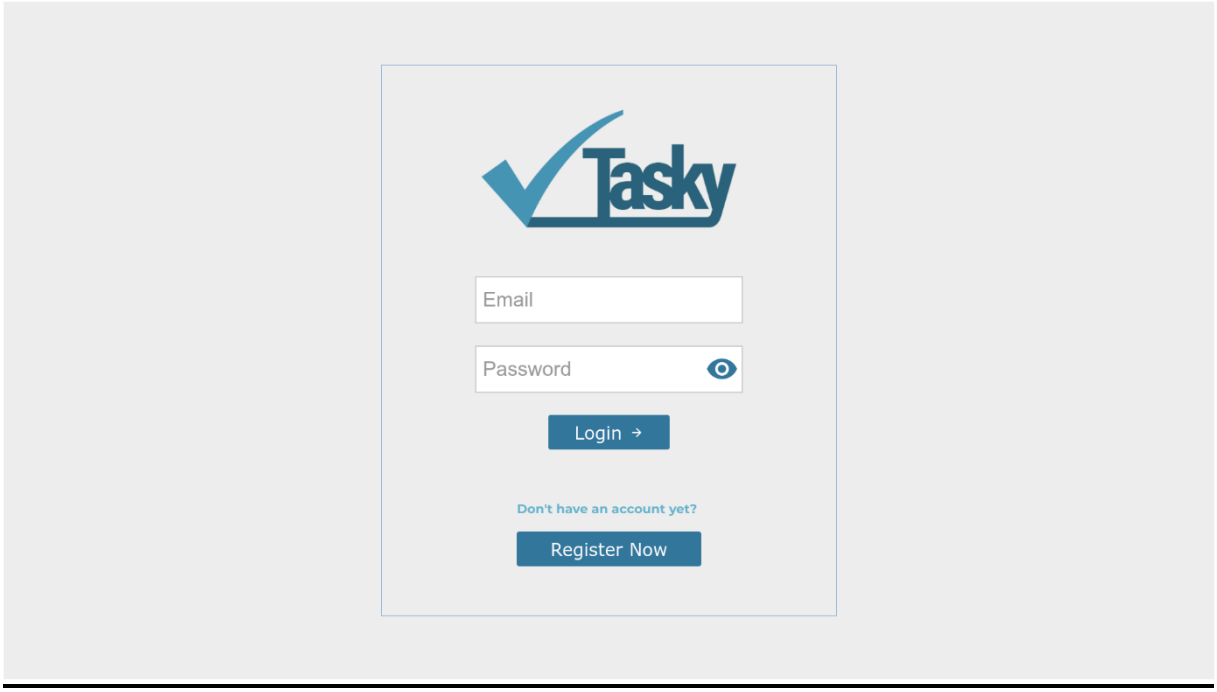
3.4.5 User interface—navigational paths and screen mock-ups

WEB MOCKUPS

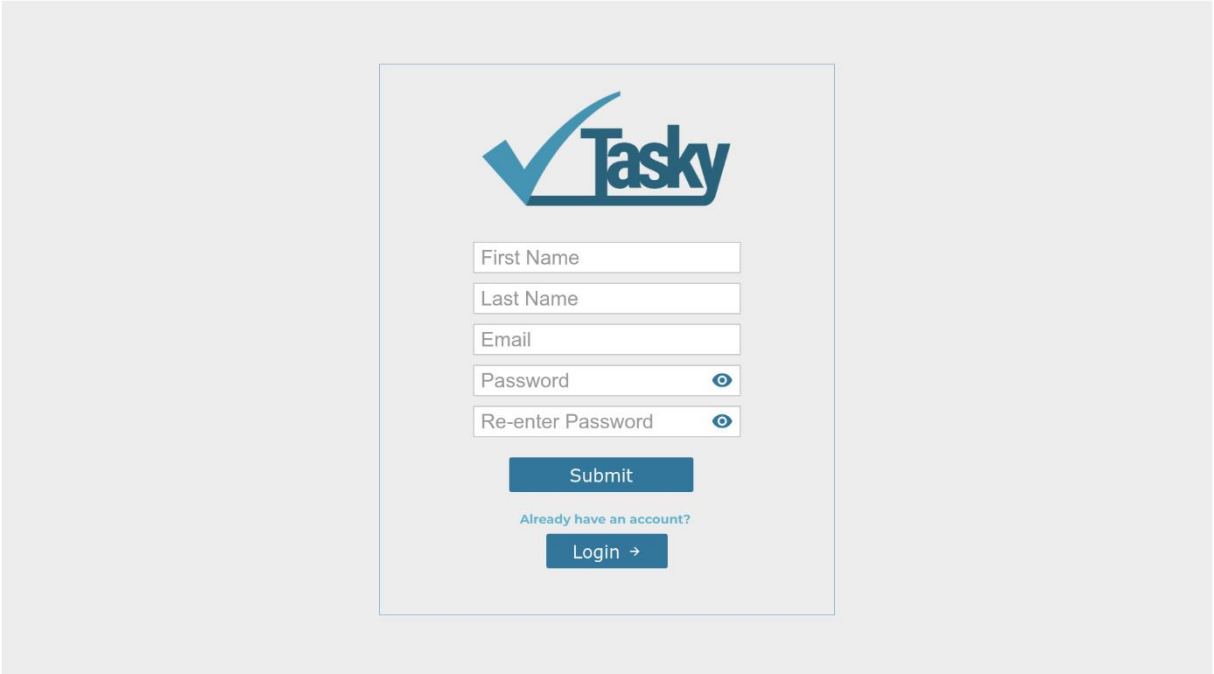
Main Page (Logged Out)



Main Page (Login)

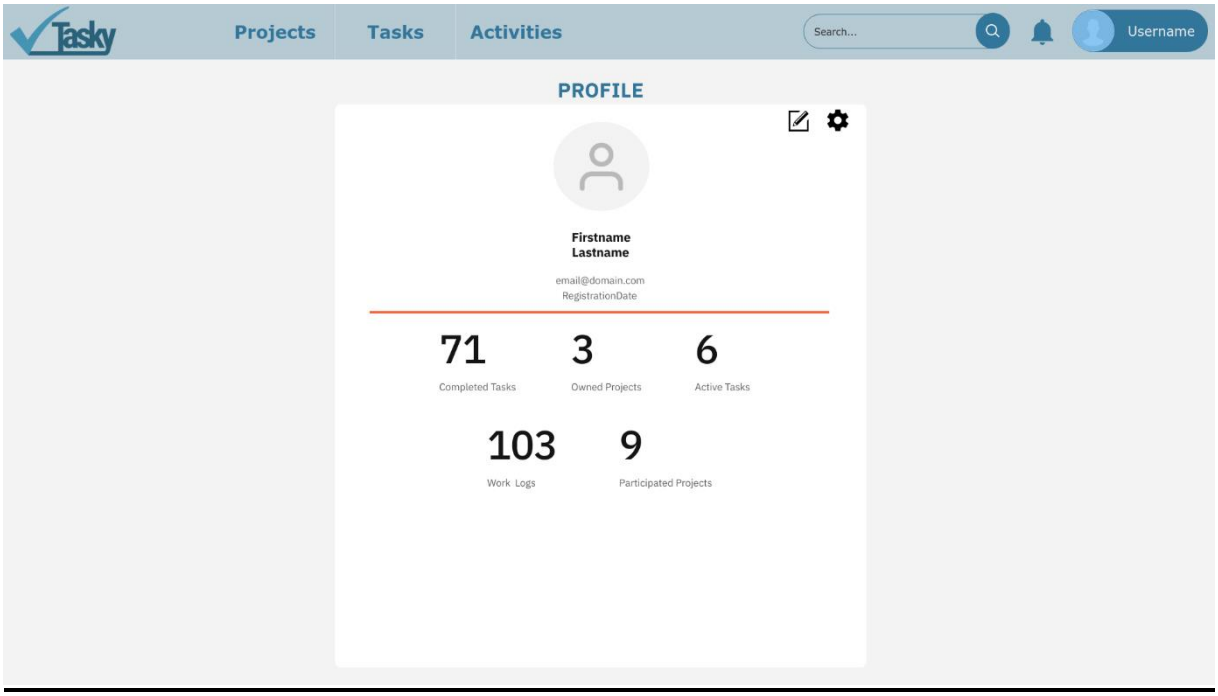


Register

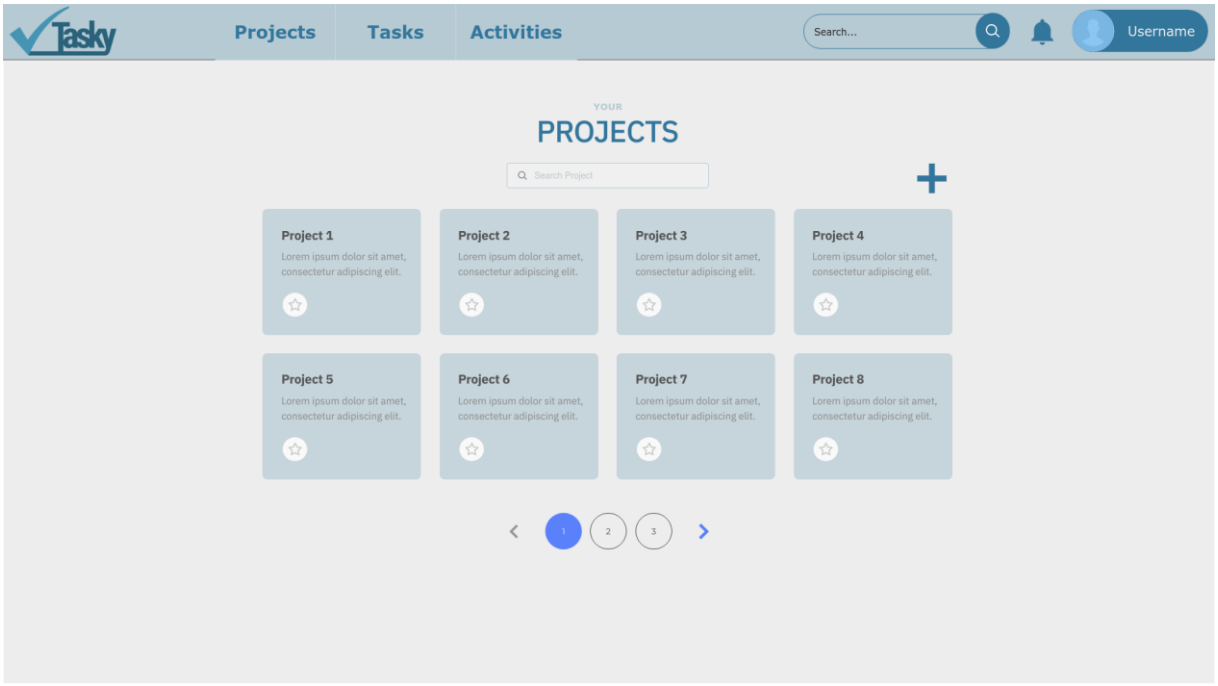


PP

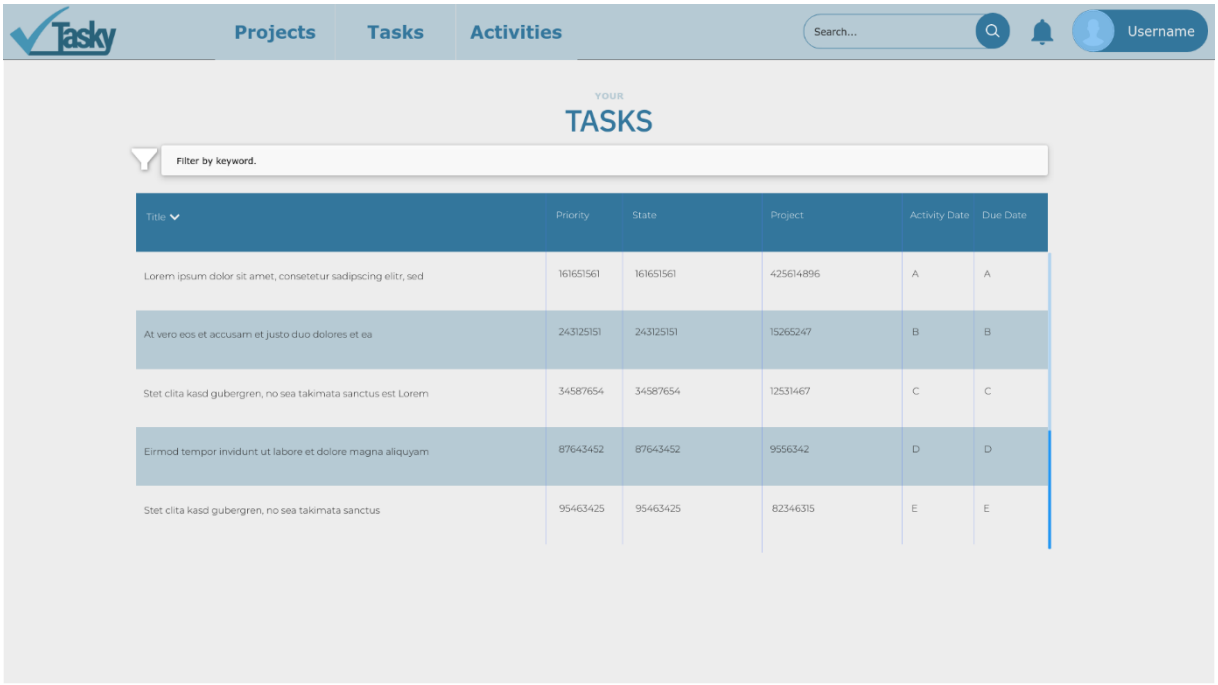
Profile



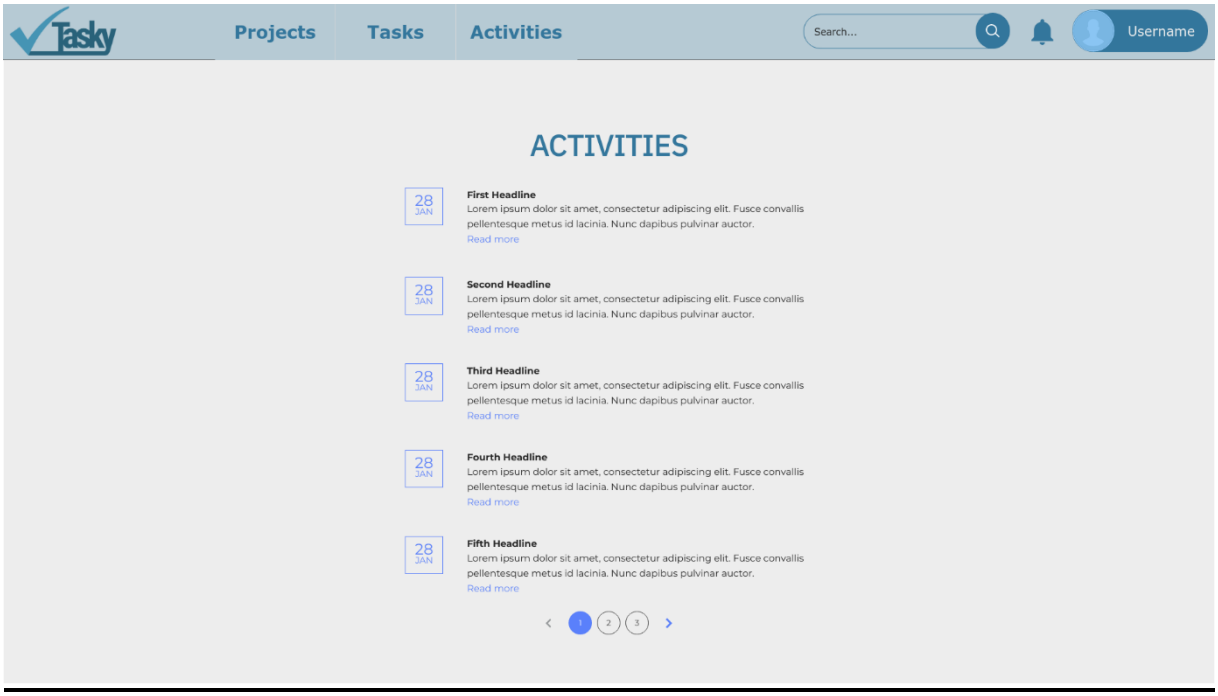
Projects



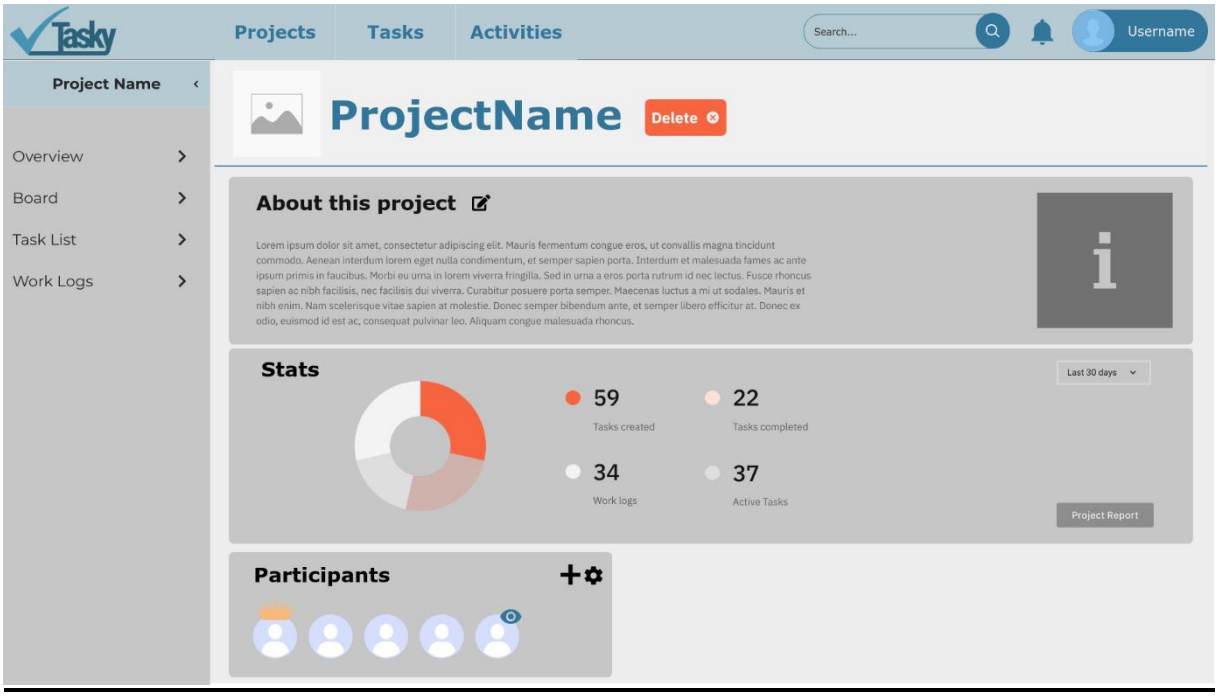
Tasks



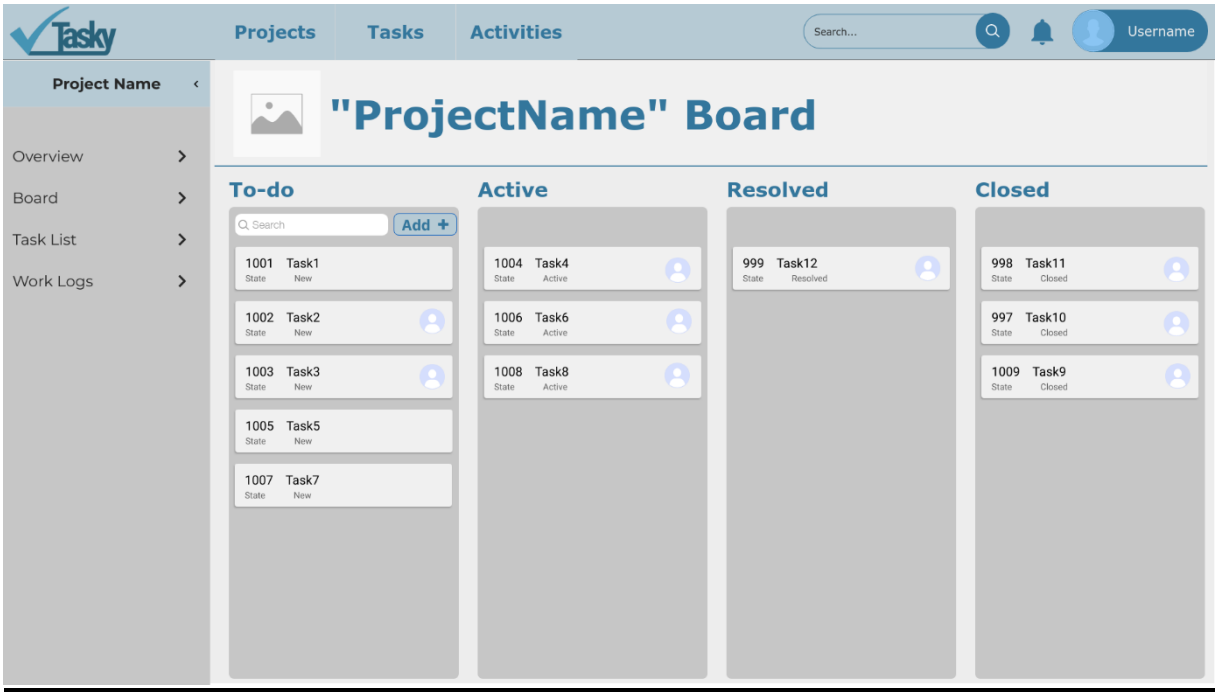
Activities



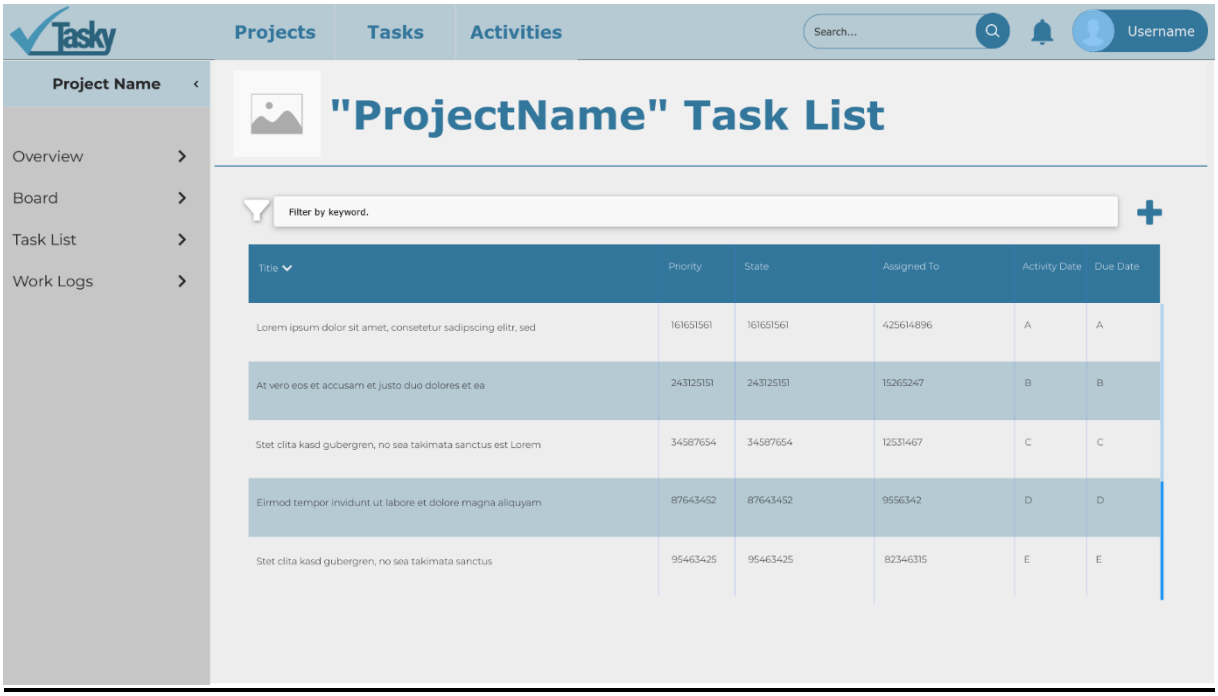
Project Overview



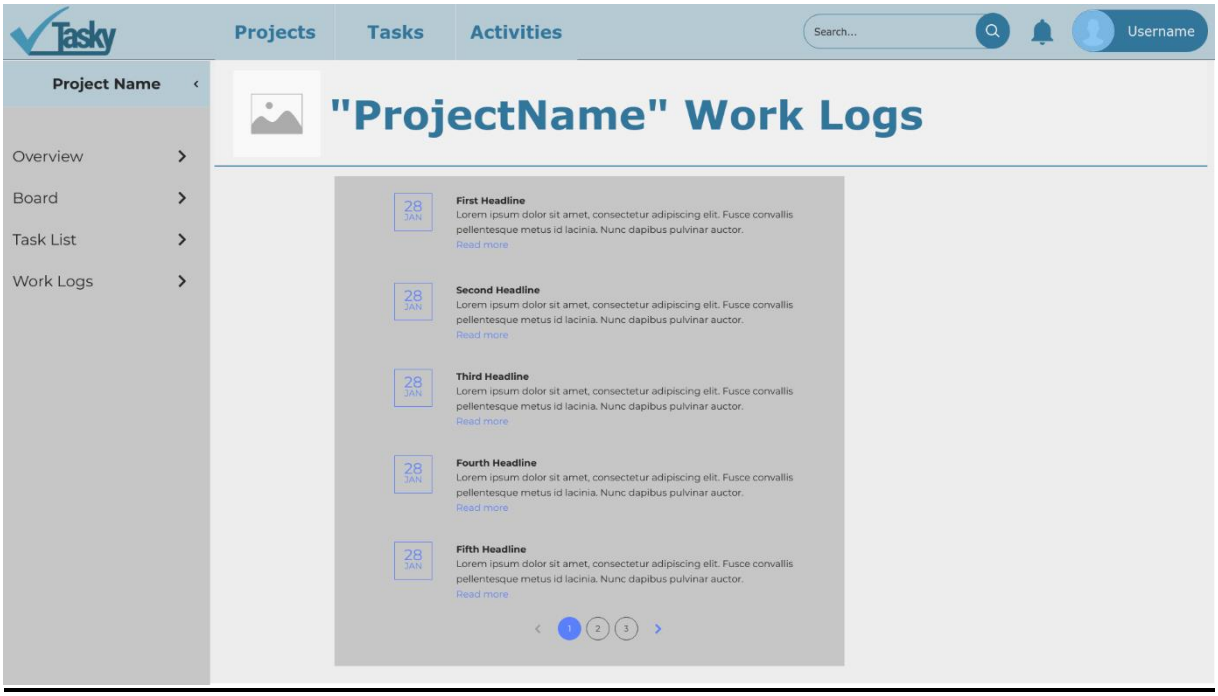
Project Board



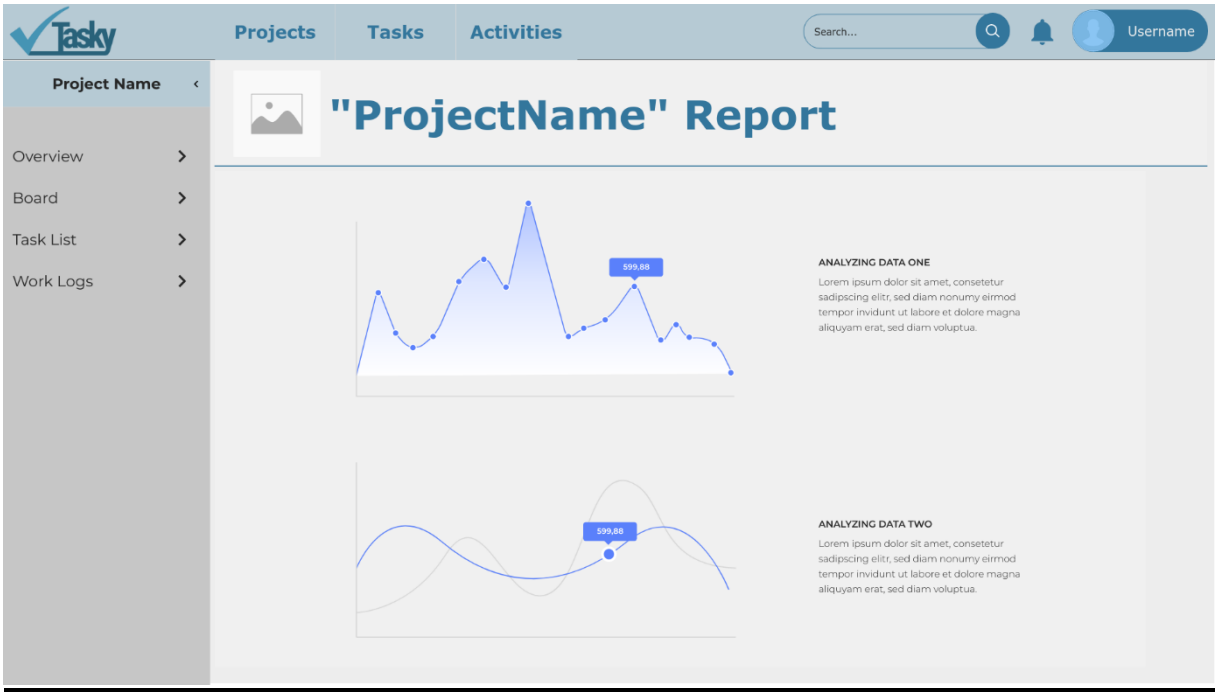
Project Task List



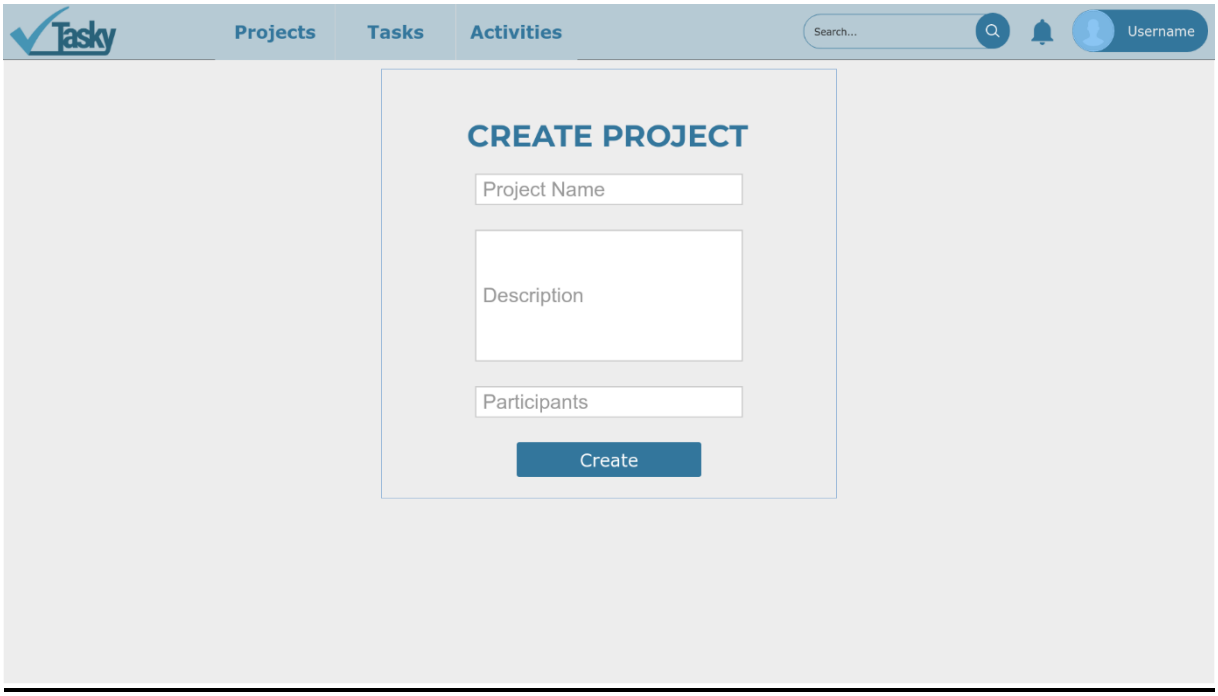
Project Work Logs



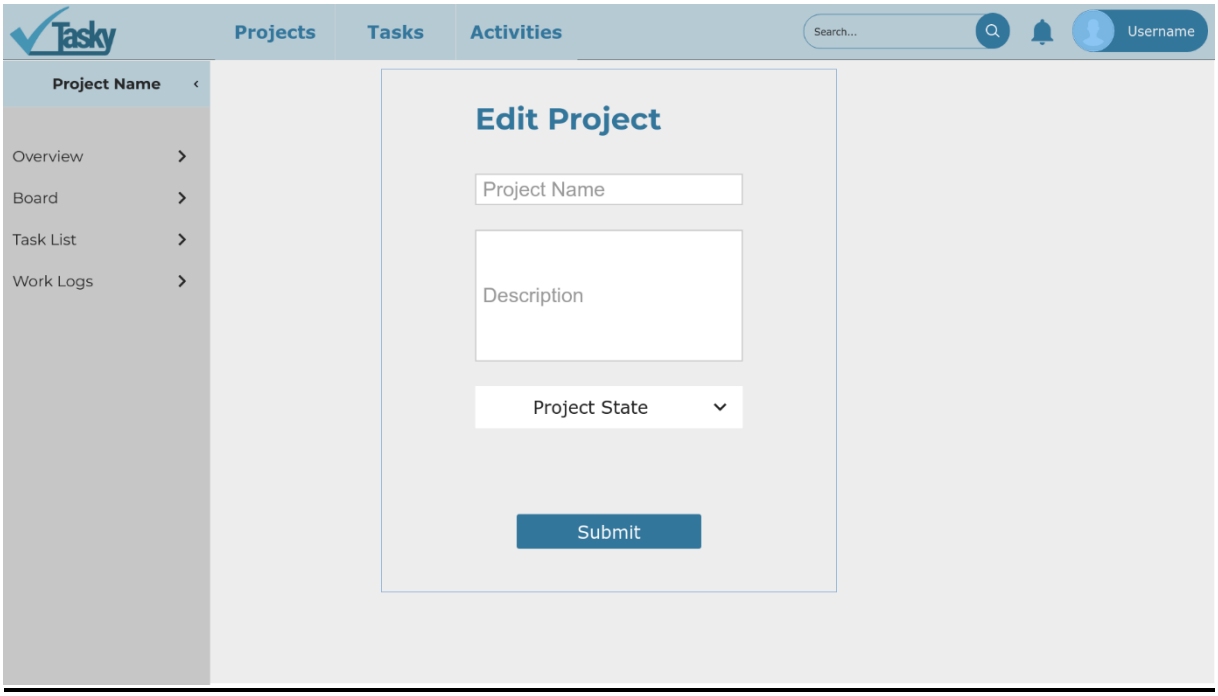
Project Report



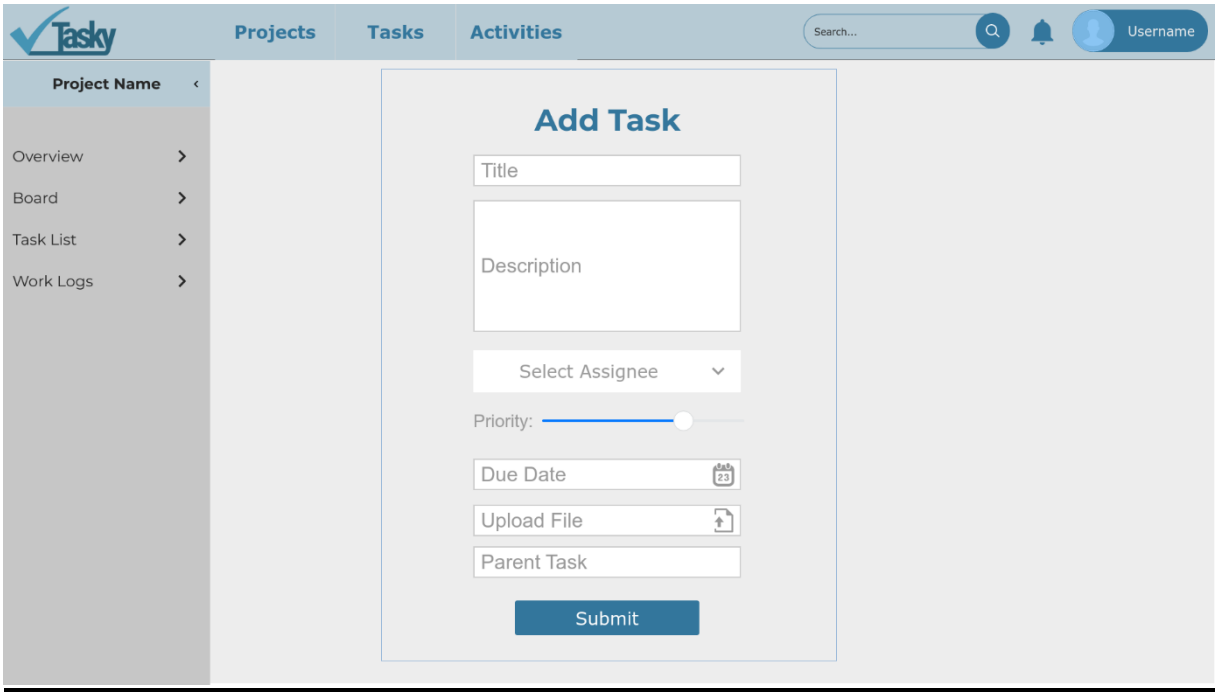
Create Project



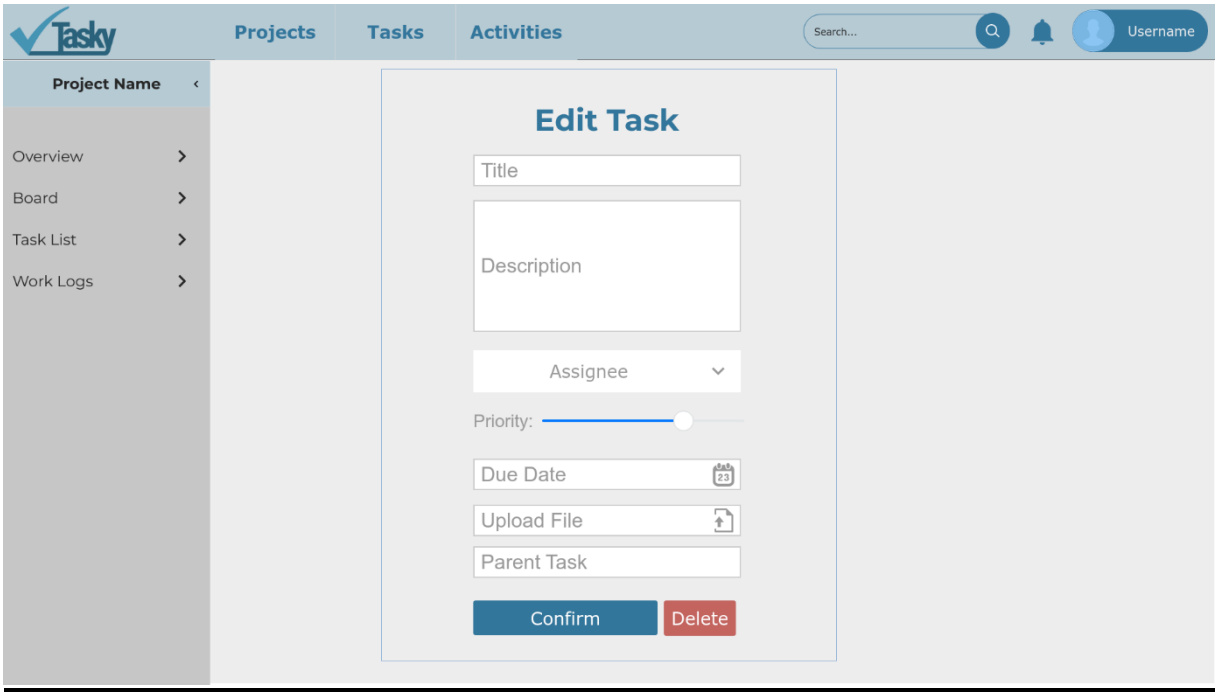
Edit Project



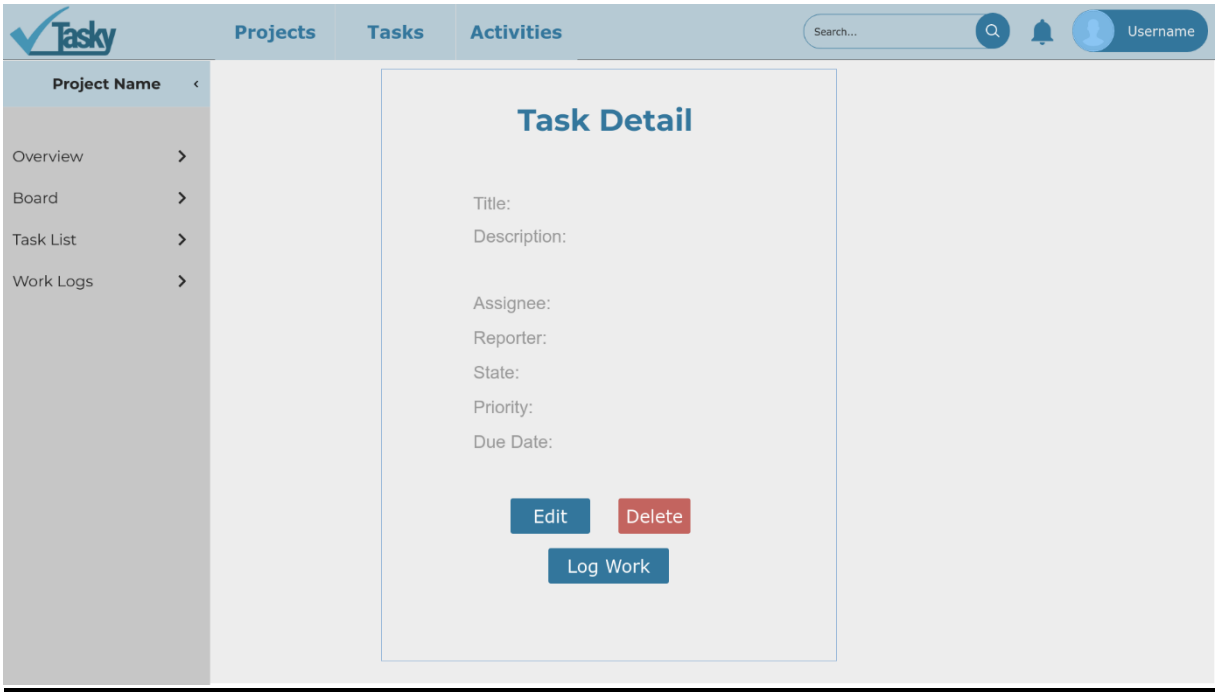
Add Task



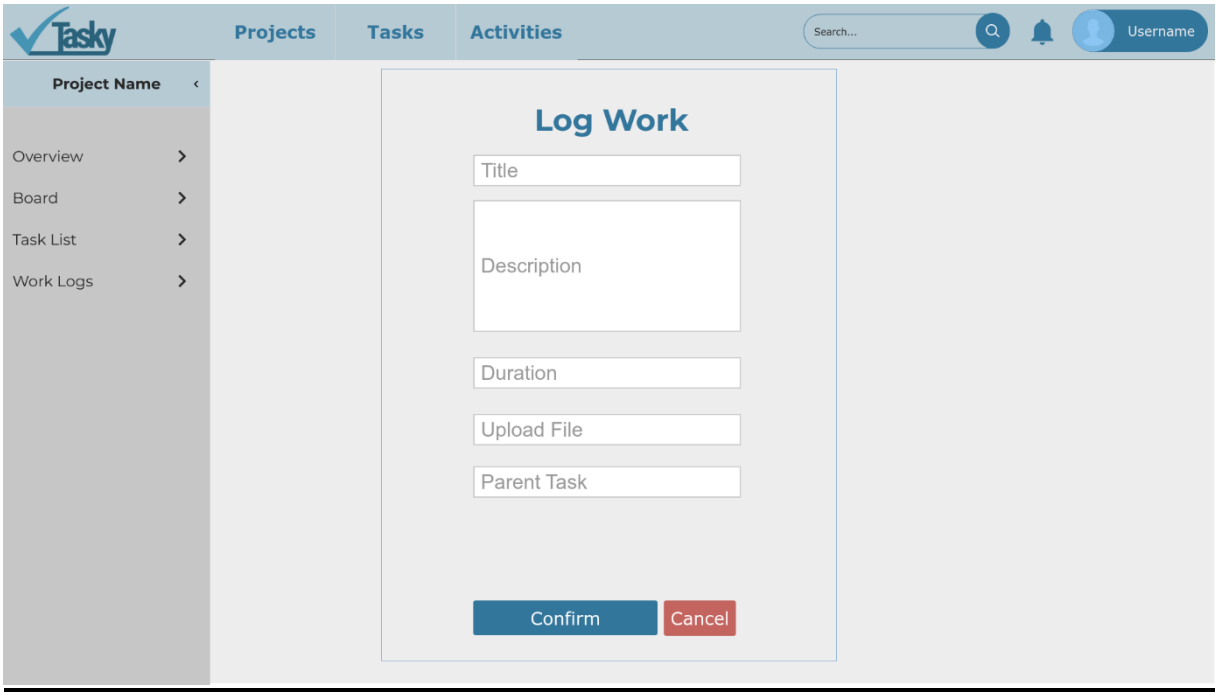
Edit Task



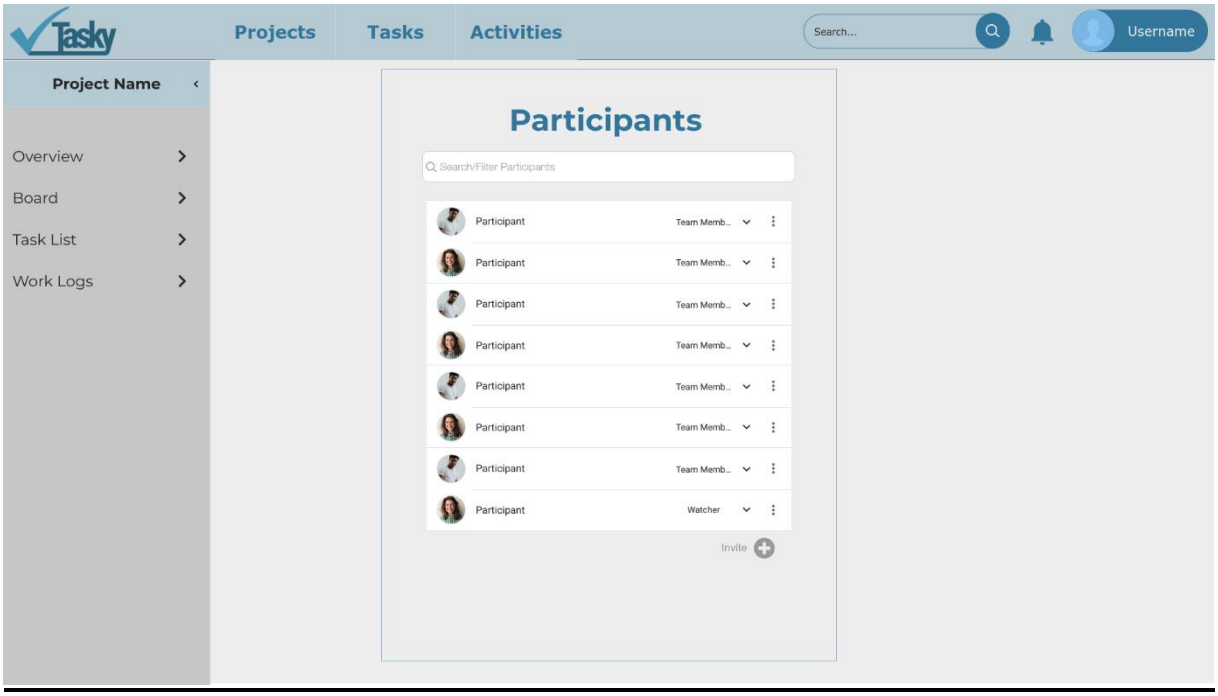
Task Detail



Log Work

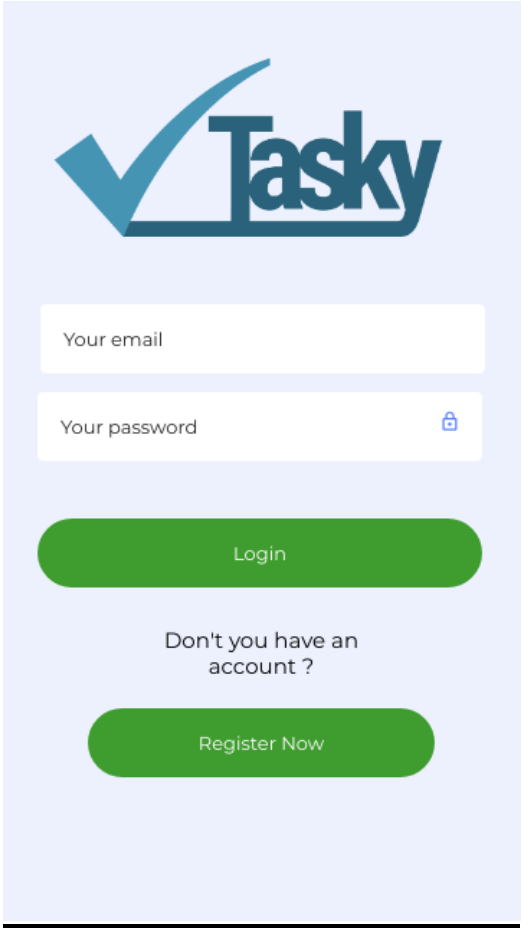


Participants

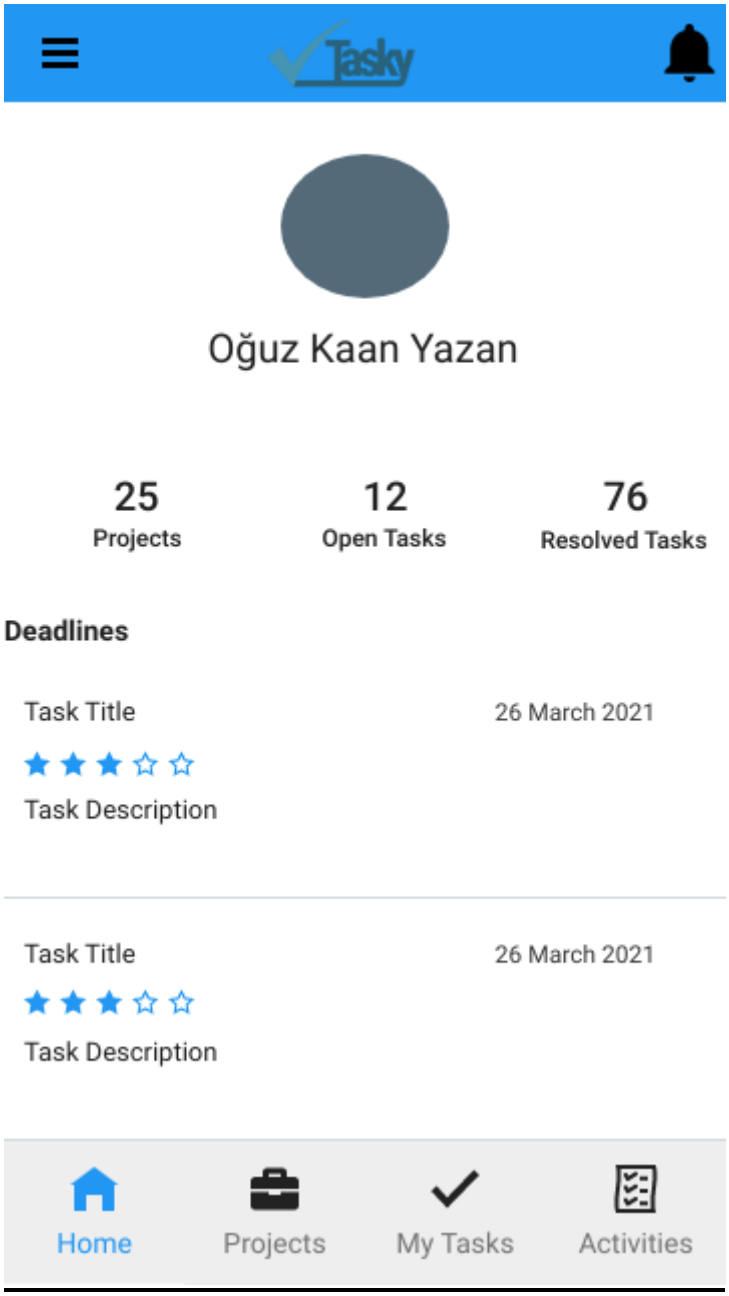


MOBILE MOCKUPS

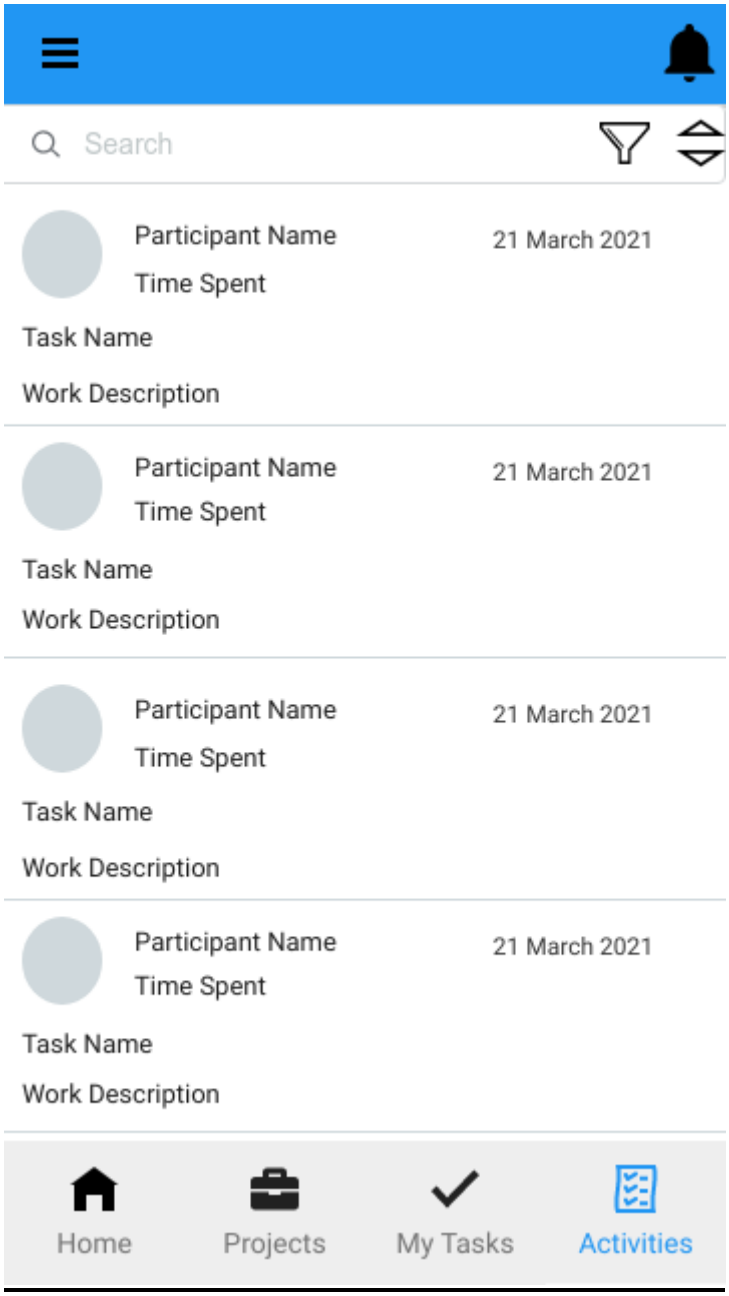
Login



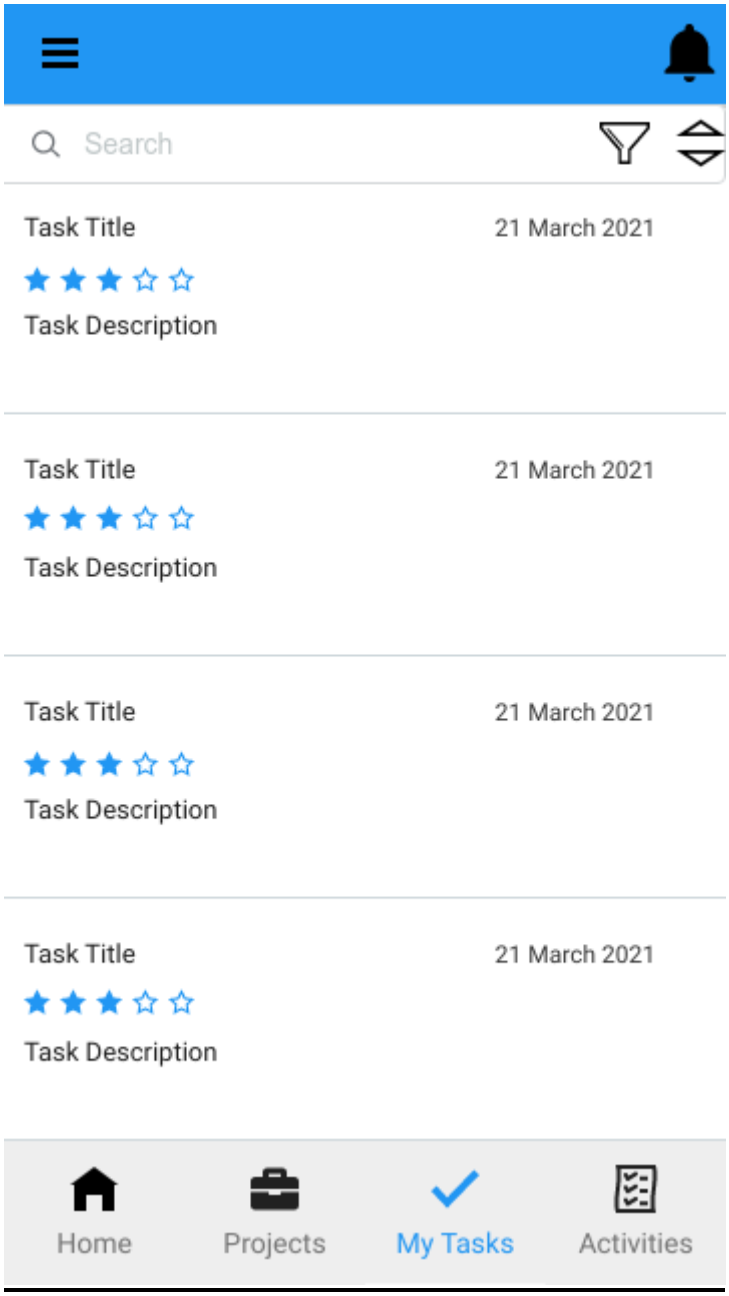
Home Screen


Tasky
Oğuz Kaan Yazan
25
Projects
12
Open Tasks
76
Resolved Tasks
Deadlines
Task Title
26 March 2021
Task Description
Task Title
26 March 2021
Task Description
Home
Projects
My Tasks
Activities

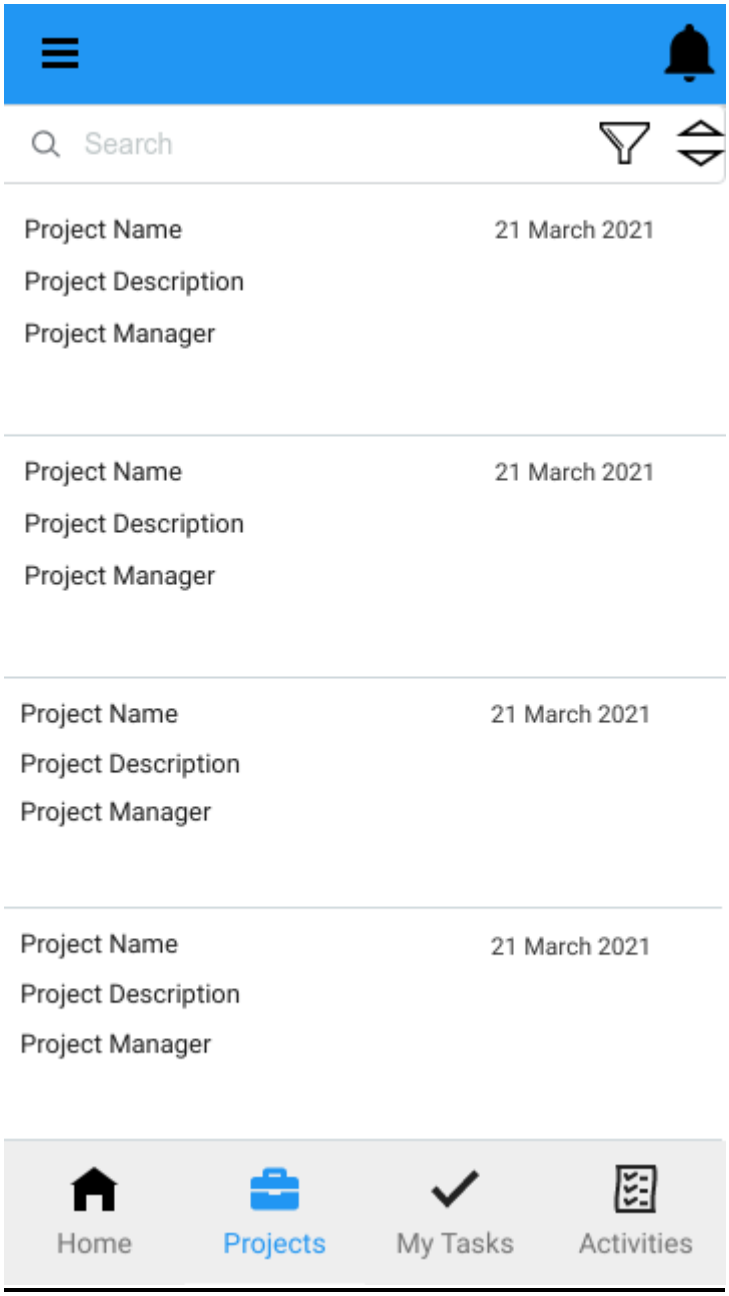
Activities


Search
Participant Name
21 March 2021
Time Spent
Task Name
Work Description
Participant Name
21 March 2021
Time Spent
Task Name
Work Description
Participant Name
21 March 2021
Time Spent
Task Name
Work Description
Participant Name
21 March 2021
Time Spent
Task Name
Work Description
Home
Projects
My Tasks
Activities

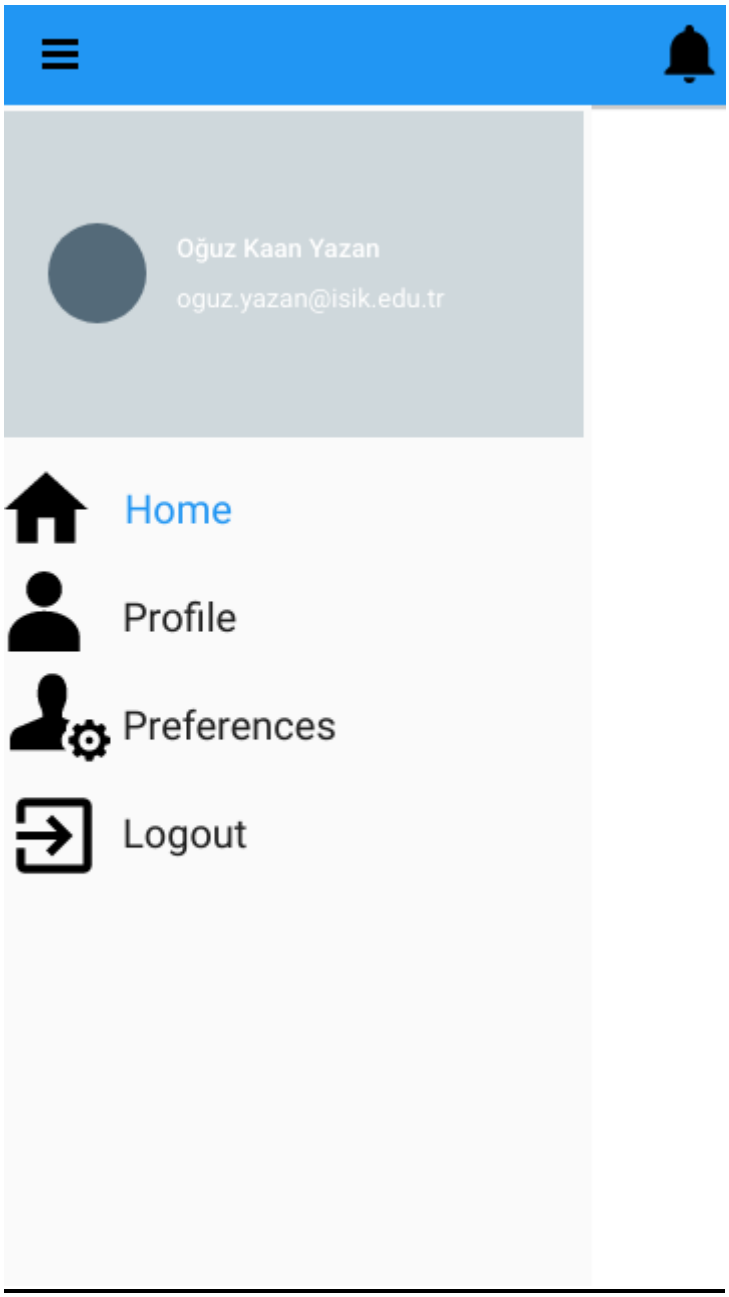
Tasks Screen



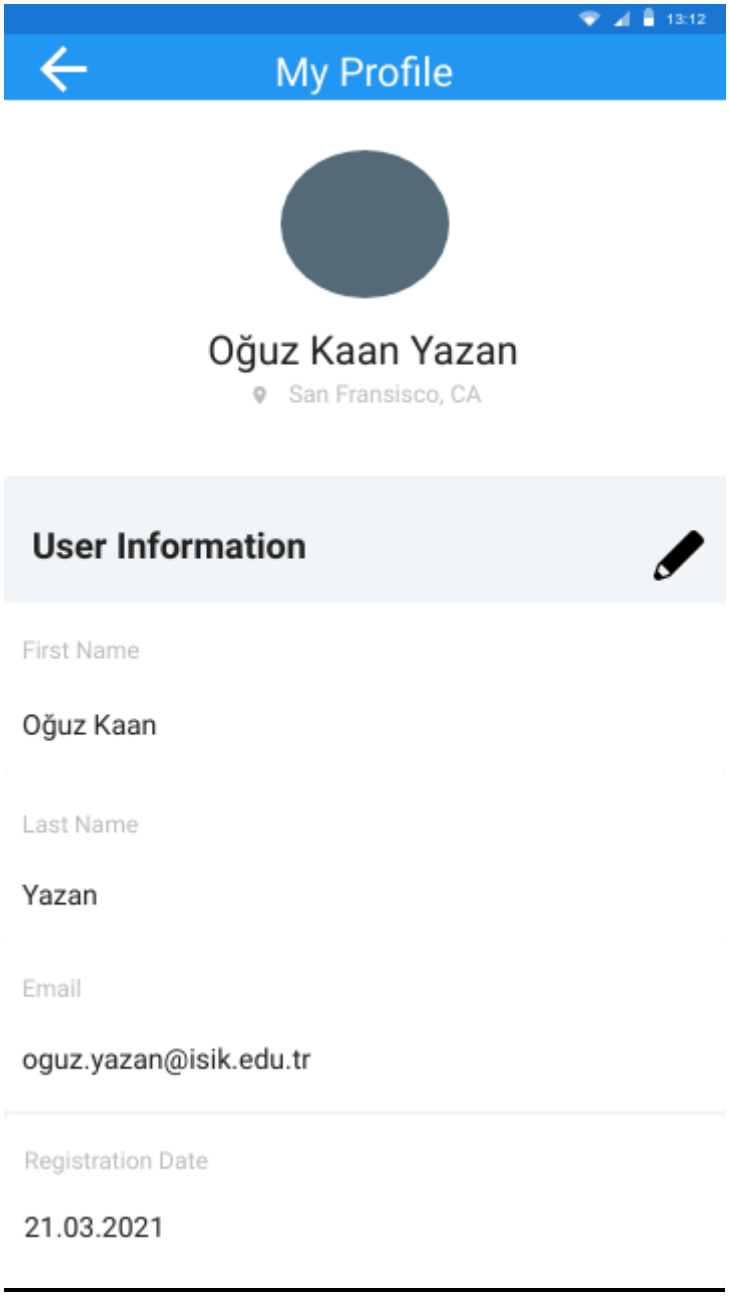
Projects Screen



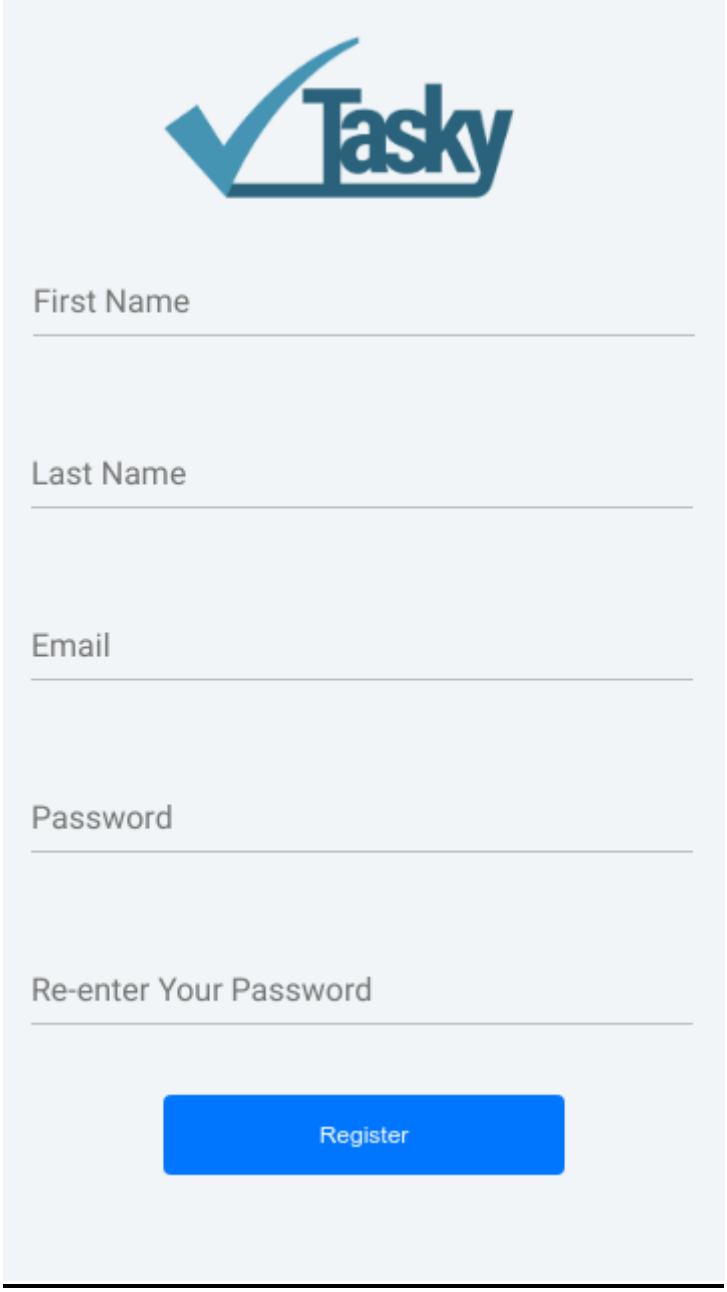
Drawer Menu


Oğuz Kaan Yazan
oguz.yazan@isik.edu.tr
Home
Profile
Preferences
Logout

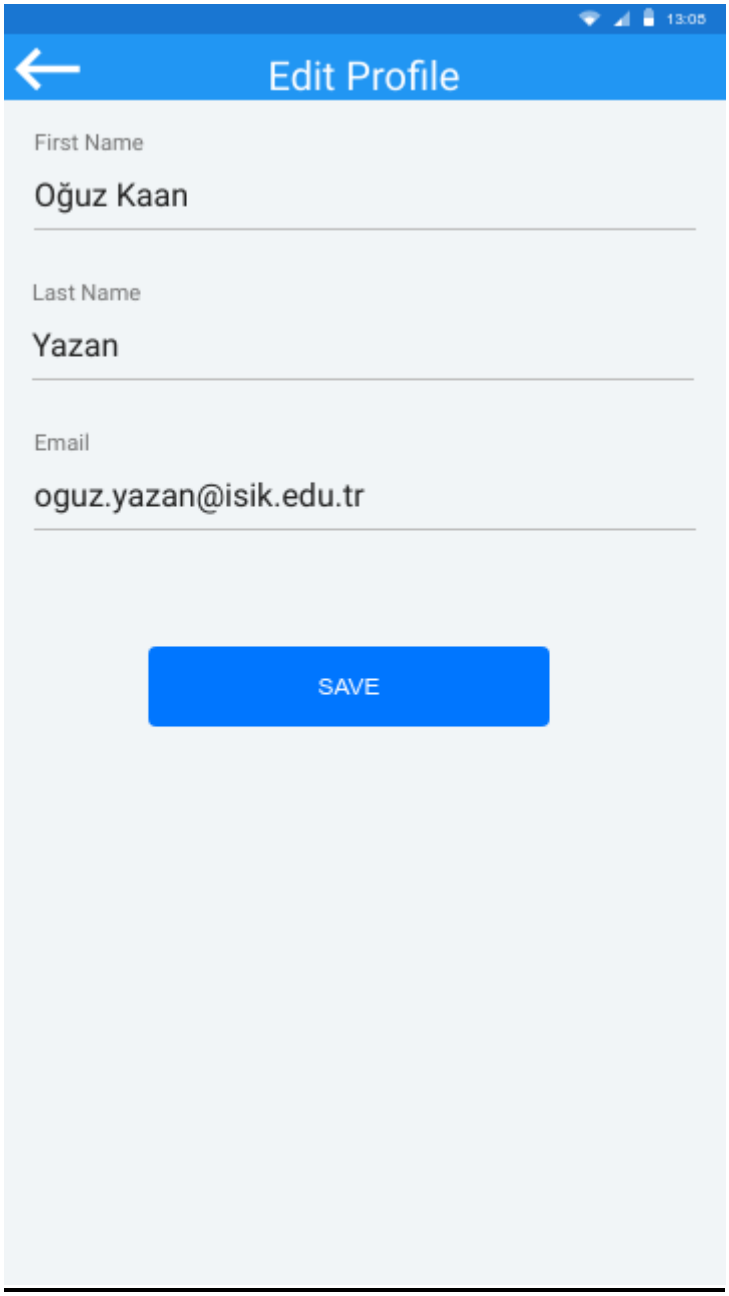
Profile


13:12
My Profile
Oğuz Kaan Yazan
San Fransisco, CA
User Information
First Name
Oğuz Kaan
Last Name
Yazan
Email
oguz.yazan@isik.edu.tr
Registration Date
21.03.2021

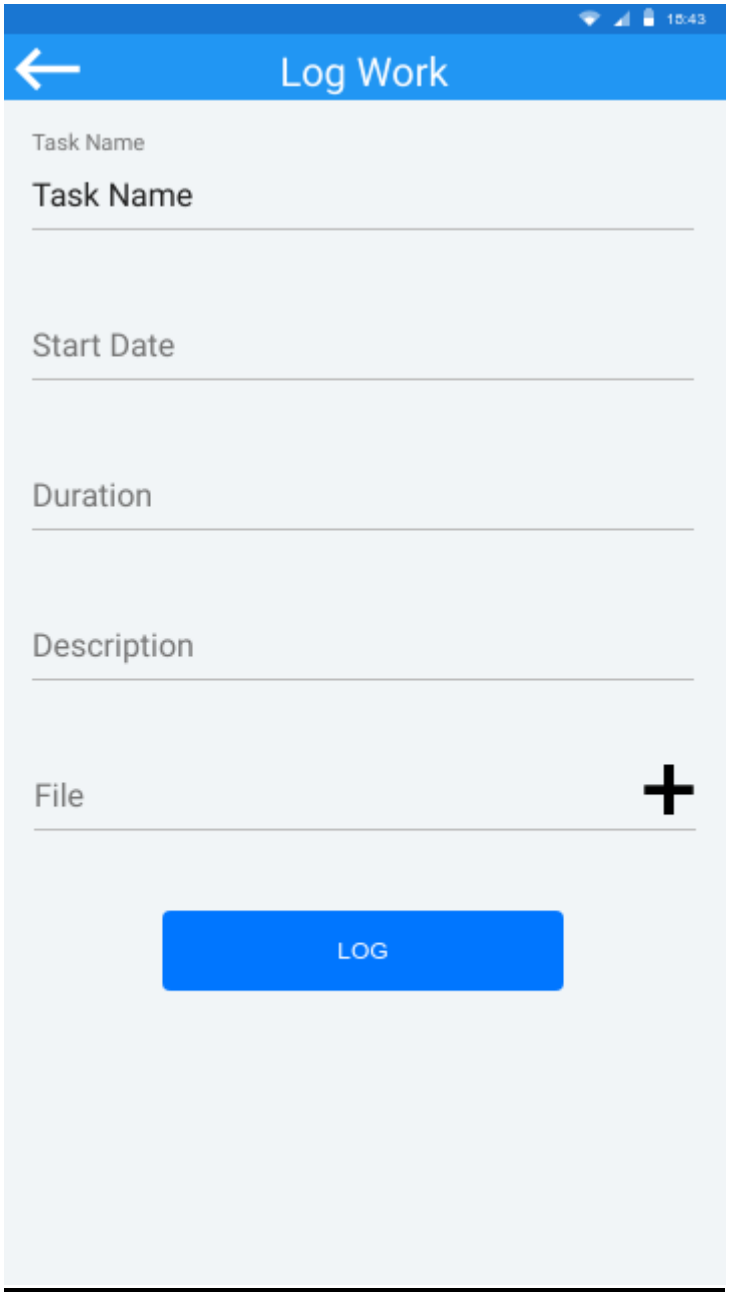
Register


Tasky
First Name
Last Name
Email
Password
Re-enter Your Password
Register

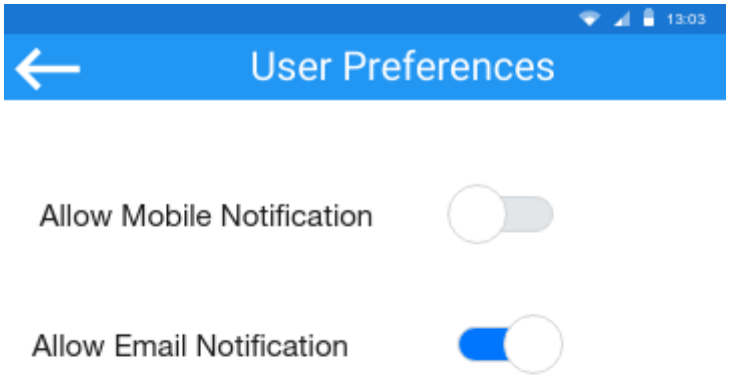
Edit Profile



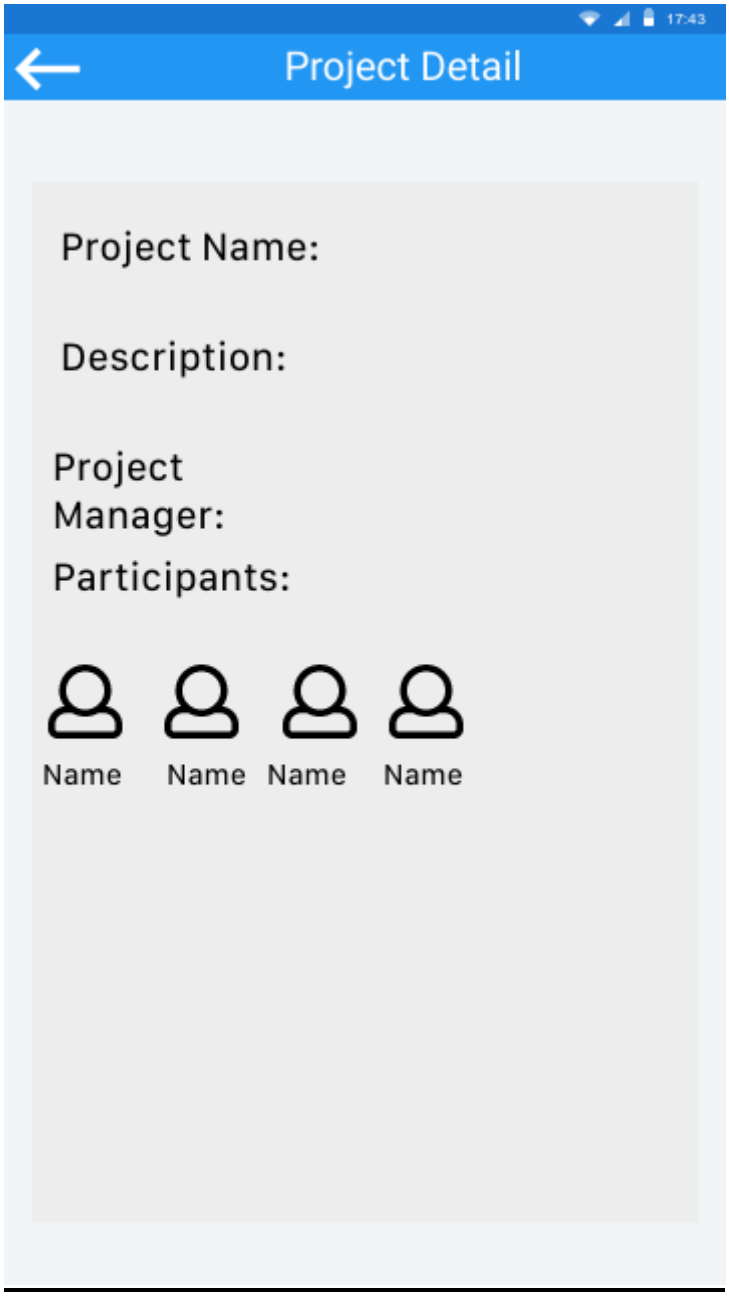
Log Work


15:43
Log Work
Task Name
Task Name
Start Date
Duration
Description
File
LOG

User Preferences



Task Detail


17:43
Project Detail
Project Name:
Description:
Project
Manager:
Participants:
Name
Name
Name
Name

4. Glossary

Tasky: Name of the proposed task management system.

Registered User: People who is registered to our system.

Guest: People who visit our system without registering.

Project Participant: User who is a member of a project.

Project Manager: User who is created a project.

Watcher: User who has read only access to the project.

Team Member: User who works on tasks under a project.

Work Log: Users log their work under tasks and they are called “work logs”.

5. References

1. Bruegge B. & Dutoit A.H.. (2010). *Object-Oriented Software Engineering Using UML, Patterns, and Java*, Prentice Hall, 3rd ed.