Team Collaboration Tool

Description

A collaboration tool for teams to manage their projects and divide them into tasks. An admin can create teams, users can create an account and join teams. Users can also add projects and divide them into tasks and complete them and keep track of their projects and their tasks.

User Stories

<p align="center">**USER**</p>

Story add user

As a user, I want to be able to add myself to the system so that I can later be added to teams and work on projects

Wireframe

Acceptance Criteria:

- Users can register themselves in the system

Story update user info

As a user, I want to be able to update my profile information to keep it accurate and personalized.

Wireframe

Acceptance Criteria:

- Users can access and edit their profile information.

Story overview teams

As a user I want an overview of the teams so I can see what teams there are

Wireframe

Acceptance Criteria: User can see the different teams

Story add user to team

As a user, I want to join teams so that I can collaborate with others and contribute to our shared goals.

Wireframe

Acceptance Criteria:
- User can add themselves to a team
- User gets a notification that he joined the selected team

Story remove user from team

As a user I want to be able to remove myself from a team so that I can discontinue my involvement from the team
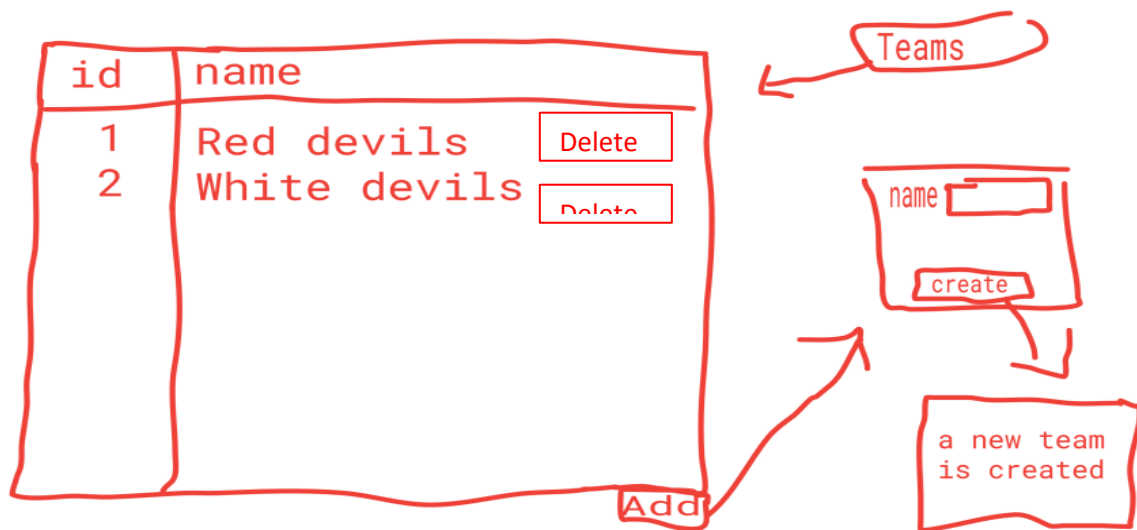
Wireframe

Acceptance Criteria:
- User can remove themselves from a team.
- User gets a notification that he left the selected team

Admin

Story create team

As the admin I want to be able to create a team so users can start the collaboration

Wireframe

Acceptance Criteria:
- Admin can create a team.
- After creating a team the admin gets a confirmation

Story disband team

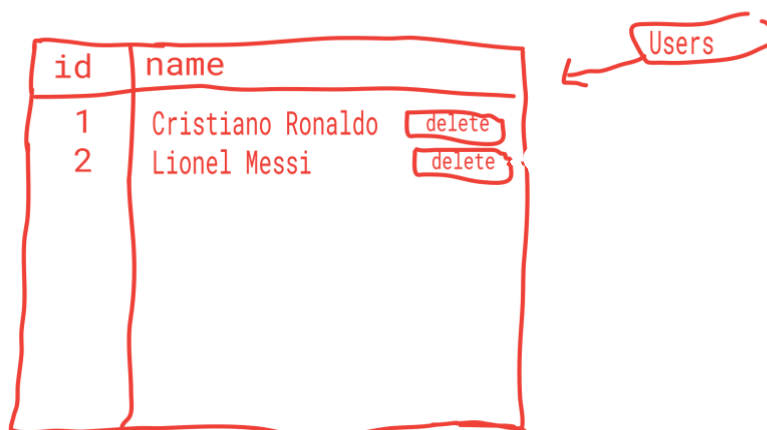As the admin I want to be able to disband a team so it doesnt show up anymore

Wireframe

Acceptance Criteria:
- Admin can disband a team.
- Admin gets asked to confirm before deleting a team
- Disbanding a team removes it from the system.

Story overview users

As the admin I want an overview of users to keep track of who's in the system

Wireframe

Acceptance Criteria: Admin can see the different users

Story remove user
As the admin I want to be able to remove users from the system so that ex-employees can't access the tool

Wireframe

Acceptance Criteria:
- Admin can remove users
- Admin gets asked to confirm before deleting a user
- A removed user should be disassociated from any team

Story Admin Project Overview
As an admin i want an overview of all projects so i can see all active projects

Wireframe
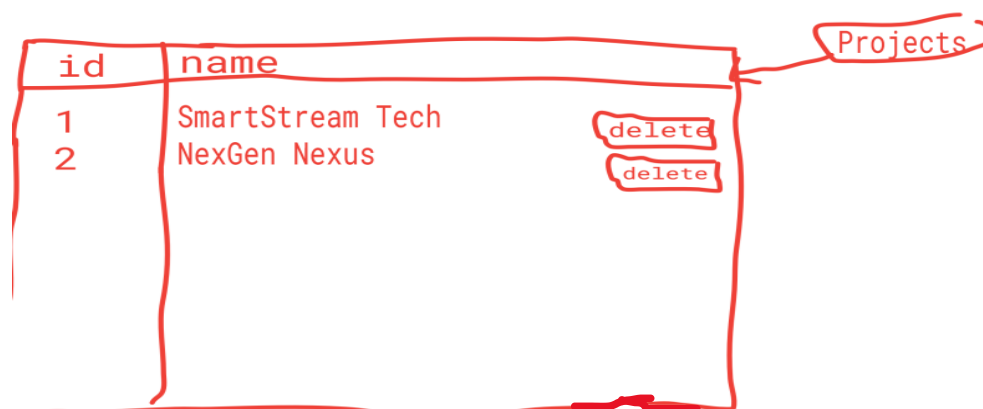
Acceptance Criteria:
- Admin can see an overview of all projects

Story remove project

As the admin i want to be able to remove projects so that completed or unneeded projects are removed

Wireframe
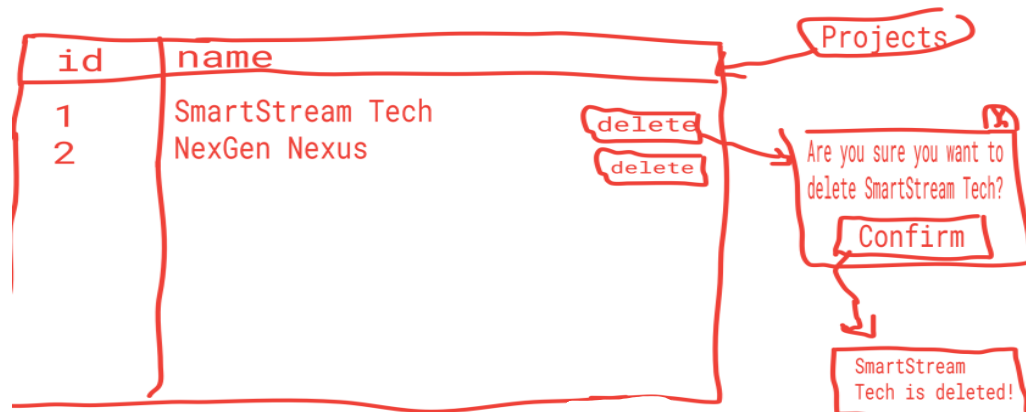
Acceptance Criteria:
- Admin can delete projects
- Deleting a project also deletes related tasks

User

Story Project Overview

As a user I want an overview of the projects I'm involved in so I can keep track of them

Wireframe

Acceptance Criteria:
- User can see an overview of his projects

Story add project

As a user I want to be able to create projects so that the team can start collaborating

Wireframe

Acceptance Criteria:
- User can create projects
- User can appoint team for a project

Story Task overview

As a user I want to see all the tasks per project I'm involved in so that I can see what needs to be done

Wireframe
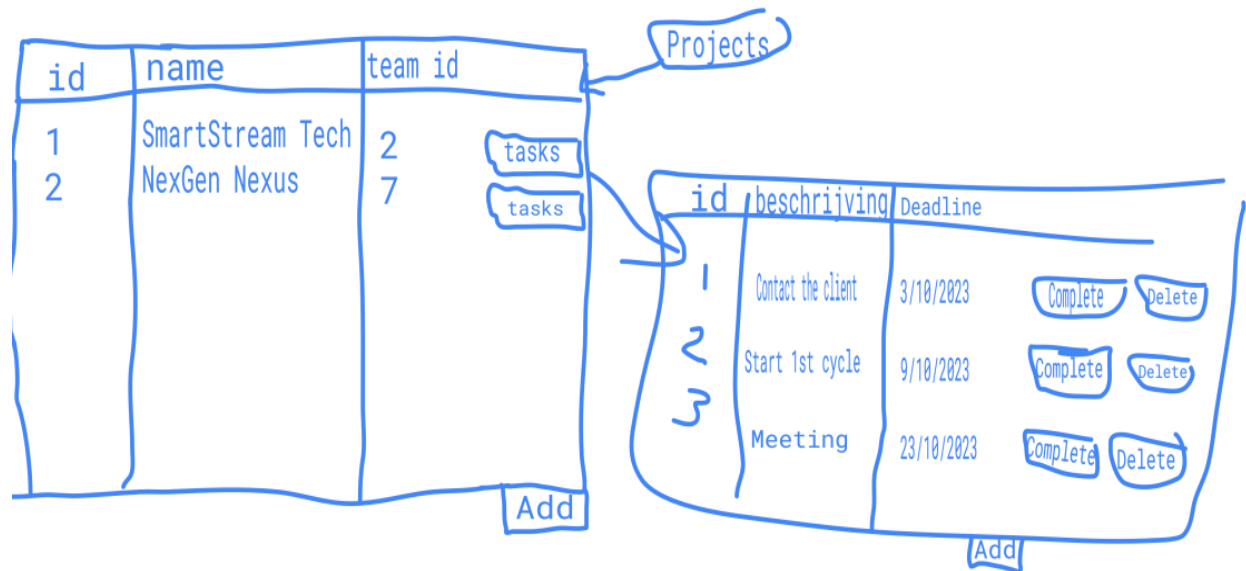
Acceptance Criteria:
- User can see all tasks of his projects

Story add tasks

As a user I want to be able to add tasks to the projects my team is doing so that the work can be divided into small tasks

Wireframe
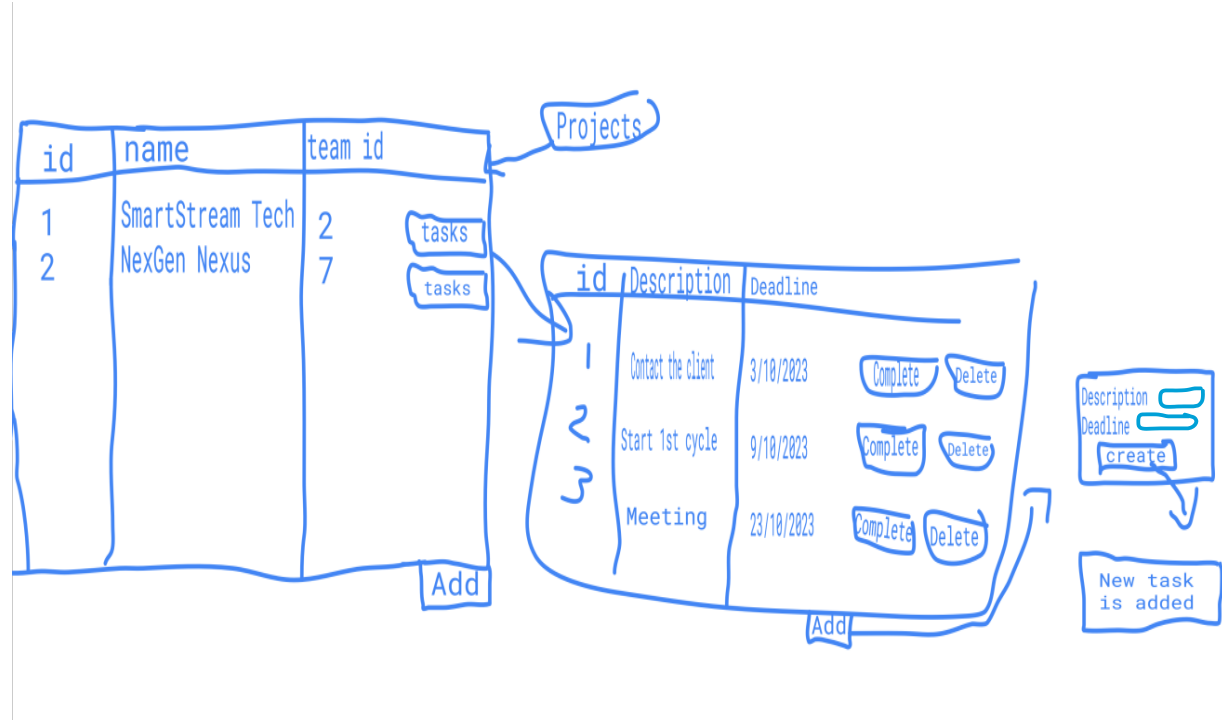
Acceptance Criteria: User can create tasks

Story complete tasks

As a user I want to mark tasks completed so the team can keep track of the progress

Wireframe
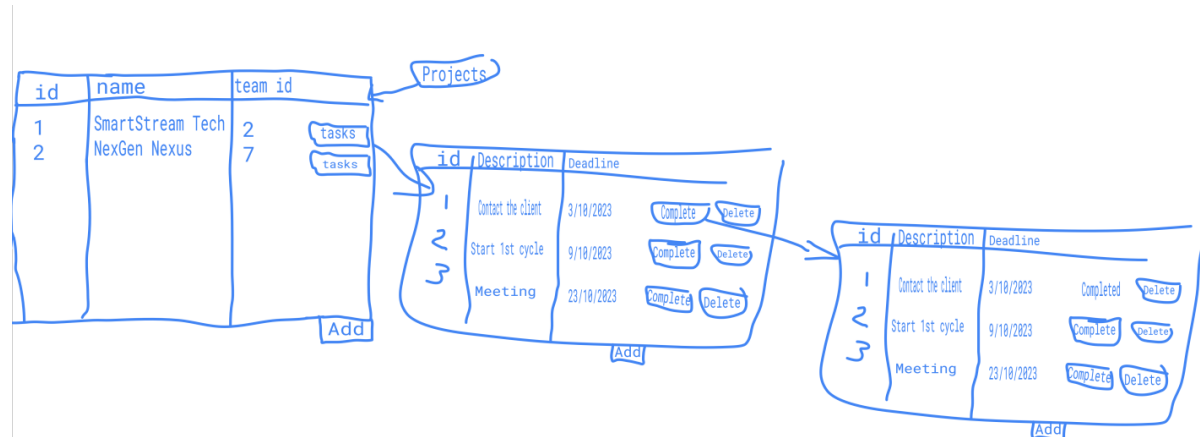
Acceptance Criteria: User can mark a task completed.

Story remove tasks

As a user I want to be able to remove tasks from the projects my team is doing so that we can streamline our project and maintain clarity in our goals.

Wireframe
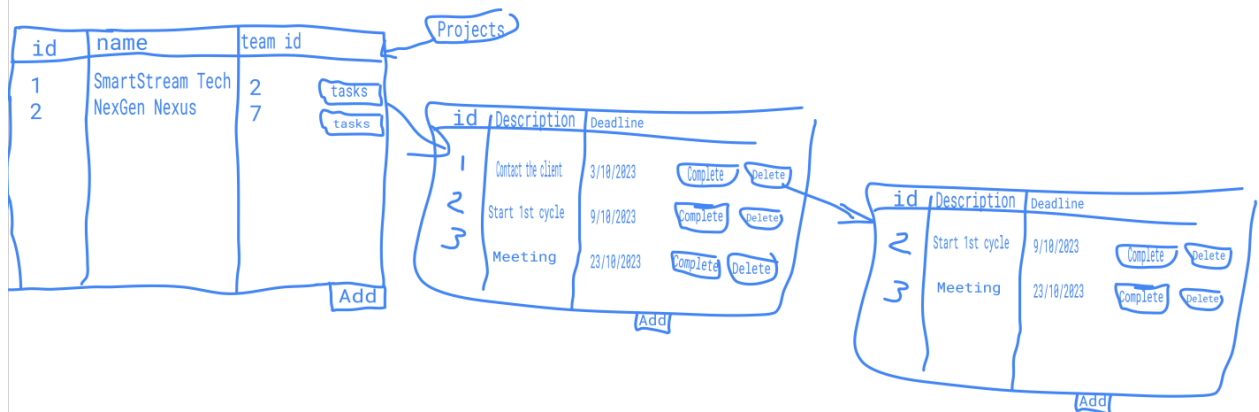
Acceptance Criteria:

- User can remove tasks

Domain Model

User
- password: string
- specialization: string
- name: string
- email: string

1..*
users

1..*
teams

Team
- name: string
- id: int

1
team

1..*
projects

Task
- name: string
- beschrijving: string
- Deadline: Date
- id : int

1..*
tasks

1
project

Project
- name: string
- id : int

Class Diagram (Extra)

User

- password: string
- specialization: string
- name: string
- email: string

+ setName(string): void
+ getName(): string
+ setPassword(string): void
+ getPassword(): string
+ getSpecialization(): string
+ setSpecialization(string): void
+ setEmail(string): void
+ getEmail(): string

1..*
users

1..*
teams

Team

- name: string
- id: int

+ setName(string): void
+ getName(): string
+ getId(): int
+ setId(): int

1
team

1..*
projects

Project

- name: string
- id: int

+ setName(string): void
+ getName(): string

1
project

1..*
tasks

Task

- name: string
- beschrijving: string
- Deadline: Date
- id: int

+ setName(string): void
+ getName(): string
+ setBeschrijving(string): void
+ getBeschrijving: string
+ setDeadline (Date): void
+ getDeadline (): Date

Entity Relationship Diagram

User

PK	email
	name
	specialisation
	password

User_In_Team

PK,FK1	user_email
PK,FK2	team_id

Team

PK	id
	name

Task

PK	id
	name
	beschrijving
	Deadline
FK	project_id

Project

PK	id
	name
FK	team_id