

Motivation

TeamMotive is an online app with the intent of providing a user friendly, and simple interface for members of a large-scale project to monitor and track progress of the project as a whole. It allows for members to create new projects and add or assign them to other members. It also lets the user keep track of and prioritize their individual tasks

Purpose

- **Provide a means for different departments of a company to keep track of progress of a project.** In the case of large scale projects, it is essential to be able communicate between departments to efficiently gauge and monitor the progress across disciplines.
- **Serve as a tool for teams to stay organized.** By having an interface that allows users to keep track of what tasks they have to complete within different projects, TeamMotive seeks to clarify each user's responsibilities and help users prioritize their work.
- **Allow members to see how their work fits in to the rest of the project.** The app helps users understand what part of the project they are contributing to and how the completion of their part allows other parts of the project to progress.

Context Diagram

The scope of this project would be a large company setting, with multiple departments that need to communicate effectively to accomplish a task.



Concepts

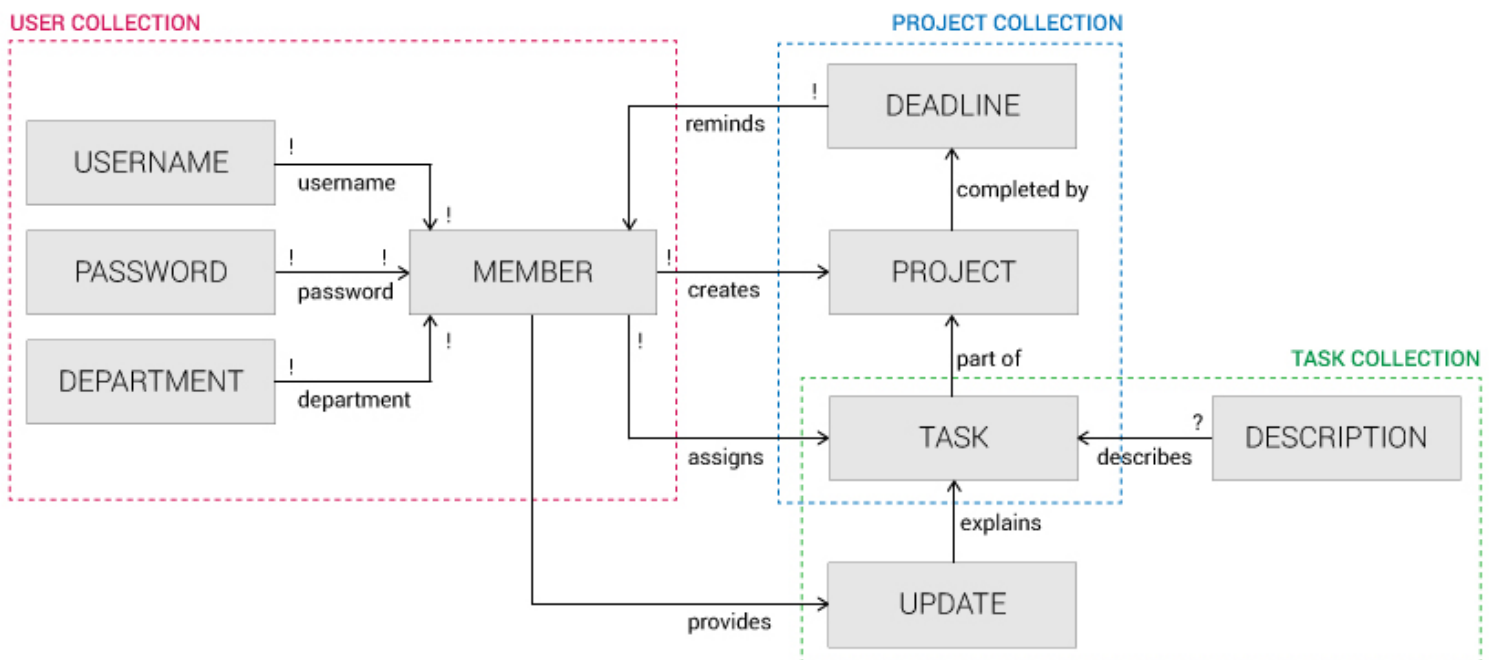
Project: This is the key concept our application is based on. Our application runs on the existence and completion of large scale projects. A project keeps the tasks organized and allows users to see how their work fits in with everything else.

Tasks: Each project can be split up in to multiple tasks that may be assigned and kept track of individually. Tasks help members stay organized.

Deadline: A deadline may simply be thought as an attribute of a project or task. However, in the case of large scale projects, deadline is a pivotal concept that determines the success of a project. The idea of a strict deadline is utilized: if a project is not completed by this deadline, repercussion will be enforced. Deadlines help users keep organized by knowing which projects need to be done first.

Updates: As a member is completing his tasks, he should make comments and provide updates in a paragraph format to explain his progress. This will help others understand the issues encountered and provide feedback throughout the process. This will also let others know the effort and time put into completing a task. Updates overall help users keep track of and have a better understanding of progress.

Data Model



Design Challenges

Creation and priority assignment of tasks

How are projects and tasks created? Similarly, how is priority assigned? The project manager should keep track of all projects created under his main project. However, it is useful for members to create projects that suit their organizational needs in implementation.

Potential solutions:

- *Only project manager can create projects or tasks and assign priority:* power is solely in the hands of the project manager, which reduces proactivity from other members.
- *All users can create projects or tasks and assign priority:* project manager oversees the whole project, so it is important for them to have the final say on the choice of projects and tasks to be completed.
- *Other members can propose new projects or tasks and priority assignments for the project manager to approve. Project manager can create projects and assign priority:* encourages other members to actively participate in the overall process of completing the project, while giving the ultimate control to the project manager.

Dropping out of a project

If a member drops out of a project, he is removed from the database, and steps are taken to ensure that the tasks do not disappear with him. The tasks must be reassigned to another member without breaking the application.

Potential Solutions:

- One way to prevent this is when a member decides to drop out, the project manager reassigns the tasks to people with proper qualifications and are least busy before actually removing the member from the database.
- Another less feasible way is to prevent a member from dropping out before completing all his assigned tasks. Doing this will conveniently result in not needing to change anything within our application, but there are probably legal reasons why this cannot be implemented.

Completion of tasks and projects

After tasks and projects are completed, should they be deleted right away or archived?

Potential solutions:

- *Delete completed tasks and projects:* keeps the pending tasks and projects free of clutter, but at the same time it becomes hard for users to determine which tasks are done and which have yet to be added.
- *Remove tasks and projects from the hierarchy and store them in an archived section:* keeps the pending tasks and projects free of clutter, but users would have to check the archived section to verify the completion of a project.

- *Update their status to be completed but keep them in the hierarchy:* users can continue to keep track of what has been done for a project.

Owner of Tasks in Data Model

One design challenge was to determine what should own tasks in the database.

Potential solution:

- *Have tasks owned directly by projects, and have the tasks store which users are assigned to them.* This approach places the project at the highest level of importance.
- *Let tasks be its own top-level collection, and to have everything that uses tasks store a list of task id's.* This approach was chosen because many different things depend on tasks, so giving tasks its own collection provides the best protection against performance bottlenecks and explicitly states its importance.

Data Design Justification

Most of the key points of the data design model have been explained in the concepts section.

One subtle design element is that a department attribute was included in a member. This is a key focus for the problem that is addressed, because the problem that large scale projects often have is the lack of communication between departments. The department attribute is set here as a reminder of our purpose.