PM2 Prototype and Use Cases - Bagel Devs

Zachary Lahlou
Maddie Oehler
Conor Edwards
Arik Scofield
Maceo Cardinale Kwik
Esther Kim
Annie Tran

Prototype:

https://www.figma.com/file/OC1XcNCKS2qxqFmWrU0sNh/Bagel-Devs---To-Do-List-Prototype?type=design&node-id=0-1&mode=design&t=zollBHfkuzUVRbjx-0

Use Case 1: Mark and View Tasks As Completed

1 Preconditions
The user must have logged into the Bagel-To-Do application.
The user has completed a task.

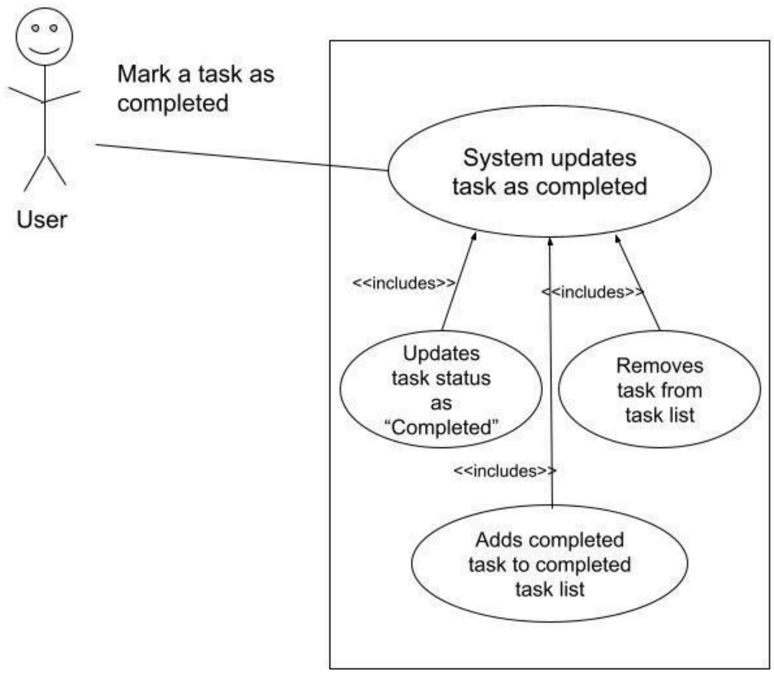
2 Main Flow
User shall navigate to the task list and mark the appropriate task as completed [S1]. System updates the task status as completed and removes the task from the list [S2]. System displays the task in the “Completed Tasks” section of the application [S3].

3 Subflows
[S1] User shall navigate to the task list and marks the task as completed using the appropriate button.
[S2] System receives the request to mark the task as completed. System updates the status of the selected task to "Completed." System removes the task from the task list.
[S3] User navigates to the "Completed Tasks" section. System retrieves the list of tasks marked as completed from the database. System displays the list of completed tasks to the user.

4 Alternative Flows
[E1] After marking the task as completed, the user realizes it was done by mistake. User clicks on an "Undo" button associated with the completed task.System reverts the task status back to "Incomplete" and adds it back to the task list. User sees the task reappear in the list of incomplete tasks.

Postconditions
The selected task is marked as completed in the Bagel-To-Do application.
The task is now visible in the “Completed Tasks” section for future reference.

The user has the option to edit the completed task settings if needed.


Mark a task as completed
User
System updates task as completed
<<includes>>
<<includes>>
Updates task status as "Completed"
Removes task from task list
<<includes>>
Adds completed task to completed task list

Use Case 2: Set Up Recurring Tasks

1 Preconditions

The user must have logged into the Bagel-To-Do application.

2 Main Flow

User shall navigate to the “Add Task” button and create a task [S1]. User shall make the task recurring using the “Recurrence” option [S2]. User shall submit the task and the system shall save the settings to automatically generate the tasks [S3].

3 Subflows

[S1] User shall click on the "Add Task" button and enter in task details such as title, description, priority, due date, etc.
[S2] User shall locate the "Recurrence" section when creating the task. User shall set the recurrence pattern (e.g., daily, weekly, monthly, yearly). User shall set the start and end dates for the recurring task.
[S3] User shall submit the task and the system shall save the recurring task settings.The system shall generate the recurring tasks in the task list based on the specified recurrence pattern.

4 Alternative Flows

[E1] After setting up a recurring task, the user decides to remove the recurrence. User accesses the task settings and navigates to the "Recurrence" section. User unselects a button indicating the recurrence.

[E2] After setting up a recurring task, the user decides to adjust the recurrence pattern. User accesses the task settings and navigates to the "Recurrence" section. User modifies the recurrence pattern. System updates the recurring task settings accordingly.

Future instances of the recurring task are adjusted based on the new recurrence pattern.

Post Conditions

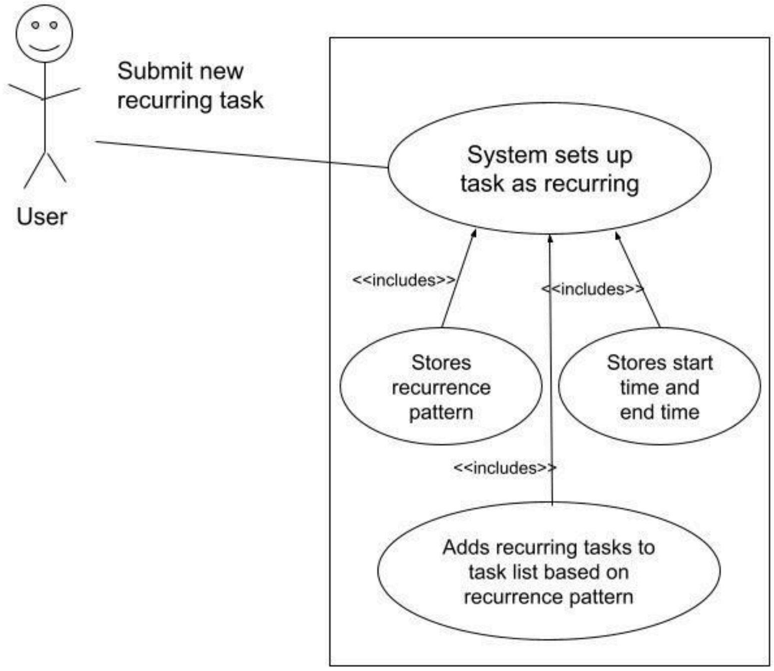
The recurring task is successfully created and saved in the Bagel-To-Do application.

Recurring tasks are automatically generated and added to the task list according to the specified recurrence pattern.

The system has stored the recurrence settings specified by the user for the task.

The start and end dates for the recurring task are recorded and implemented by the system.

The user has the option to edit or delete the recurring task settings if needed.



Submit new recurring task
User
System sets up task as recurring
<<includes>>
<<includes>>
Stores recurrence pattern
Stores start time and end time
<<includes>>
Adds recurring tasks to task list based on recurrence pattern

Use Case 3: Categorize and Filter Tasks

1 Preconditions

The user must have logged into the Bagel-To-Do application.

2 Main Flow

User shall navigate to the “Add Task” button and create a task [S1]. User shall assign a category to the task [S2]. User shall submit their task and the system shall display task labeled with its categories [S3]. User shall select filters for the task list based on category and the system shall filter the task list [S4].

3 Subflows

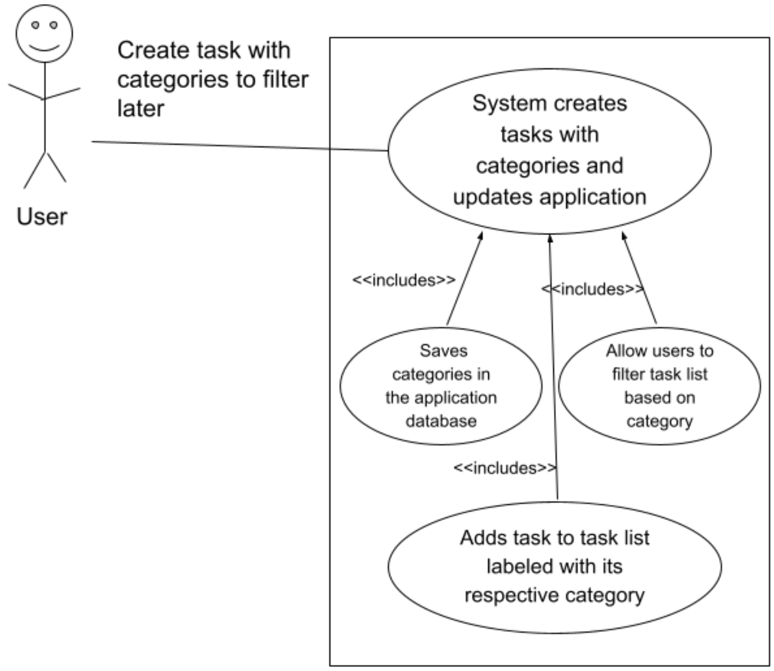
[S1] User shall click on the "Add Task" button and enter in task details such as title, description, priority, due date, etc.
[S2] User shall locate the "Category" option when creating the task. User shall select existing categories or create new categories (e.g., "Work," "Personal," "Project A," "Project B").
[S3] User shall submit the task and the system shall display task in the task list labeled with its categories.
[S4]. User shall navigate to the "Filters" feature of the task list and select one or more categories to filter the task list. System shall display only the tasks belonging to the selected categories on the task list.

4 Alternative Flows

[E1] User accesses the task details of the specific task. User locates the "Category" option associated with the task. User modifies the category selection by either choosing a different existing category or creating a new one. System updates the task's category label accordingly. Task is now displayed in the task list with a newly assigned category.
[E2] User navigates to the "Filters" feature of the task list. User selects desired categories along with additional criteria such as due date, priority, or status.System processes the combined filters and displays only the tasks that meet all selected criteria. User can view the refined task list containing tasks filtered by category as well as other specified criteria.

Post Conditions

Tasks with categories are successfully created and saved in the Bagel-To-Do application.
Each task is labeled with its respective category, as specified by the user during task creation.
Users can apply filters to the task list based on categories and other criteria, resulting in a refined task view.


Create task with categories to filter later
User
System creates tasks with categories and updates application
<<includes>>
<<includes>>
<<includes>>
Saves categories in the application database
Allow users to filter task list based on category
Adds task to task list labeled with its respective category

Use Case 4: Creating Team Tasks

1 Preconditions

The user must have logged into the Bagel-To-Do application.
The user has created a team with other users.

2 Main Flow

User shall navigate to the “Team” section of the application and click on the “Add Team Task” button to create a task [S1]. User shall assign team members to the task [S2]. User shall submit their task and the system shall display the task on the task lists of the assigned team members task list and a team task list [S4].

3 Subflows

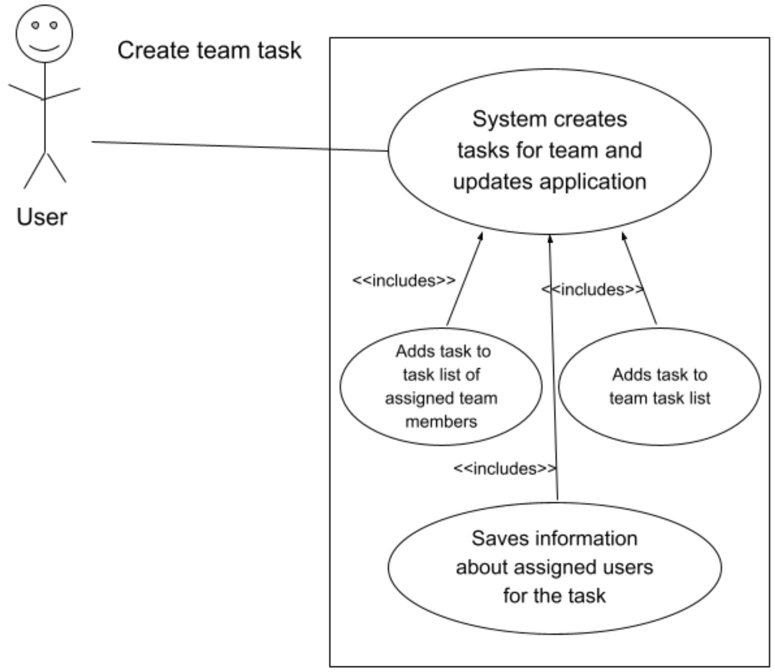
[S1] User shall navigate to the “Teams” section of the application and click on the "Add Task" button and enter in task details such as title, description, priority, due date, etc.
[S2] User shall locate the "Users" section when creating the task. User shall select a team and mark team members to be assigned to the task.
[S3] User shall submit the task. System shall display the task on the task list of all the assigned members. System shall display the task on the appropriate team task list in the “Teams” section, the task labeled with the team members icons.

4 Alternative Flows

[E1] User accesses the task details of the specific task in the “Team” section of the application. User locates the "Users" option associated with the task. User modifies the team member assignments. System updates the individual and team task lists appropriately.

Post Conditions

Team tasks that are assigned to individual or multiple team members are successfully created and saved in the Bagel-To-Do application.
Assigned tasks are visible to the designated team members in their task list and on the team task list.


User
Create team task
System creates tasks for team and updates application
<<includes>>
<<includes>>
Adds task to task list of assigned team members
Adds task to team task list
<<includes>>
Saves information about assigned users for the task

Use Case 5: Editing a task

1 Preconditions

The user must have logged into the Bagel-To-Do application.
The user has created a task.

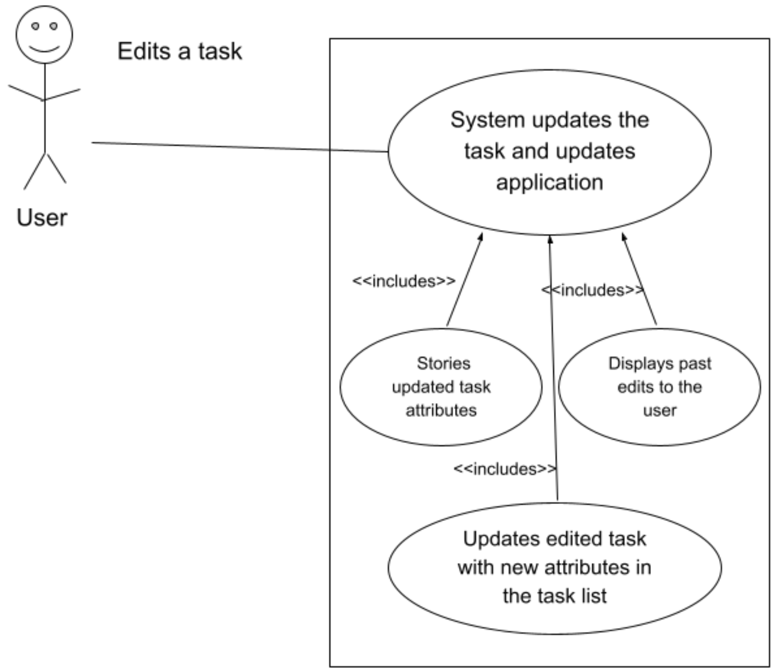
2 Main Flow

The user shall navigate to the task list and click edit task [S1]. The system updates the task attributes such as the due date. [S2]. Users shall view updated task attributes [S3].

3 Subflows
[S1] User shall navigate to the task list and edit the task using the appropriate button.
[S2] The system receives the request to edit the task attributes. The system updates the attributes of the selected task. The system shows the task's changes when viewing the task.
[S3] User navigates to the "Tasks" section. The system retrieves the list of tasks and shows the attributes and edit button. The system displays the changes when viewing the task.

4 Alternative Flows
[E1] After editing the task, the user realizes it was done by mistake. The user clicks on an "Undo" button associated with the edited task. The system reverts the task attribute changes and updates the changes in the task list.
.

Postconditions
The selected task has the updated attributes shown in the task list when viewing the task.
The user has the option to edit the task further if needed.
The user has the option to view past edits.


Edits a task
User
System updates the task and updates application
<<includes>>
<<includes>>
Stories updated task attributes
Displays past edits to the user
<<includes>>
Updates edited task with new attributes in the task list