ToGather   
Use-Case: Create a Task

# Brief Description

# The use case describes how the users create a task on calendar application.

# Actor Brief Descriptions

## Professionals or University Student (Users)

# Preconditions

Users must have stable internet connection for the device they are using the application.

Users must be Logged-in to the ToGather App.

# Basic Flow of Events (Main Success Scenario)

1. The application user interface is displayed to the User
2. User selects current schedule.
3. User presses “+” button to add new task additional to the existing tasks.
4. System displays two options: “Manual Task (Marked as Default)” and “Automated Task”
5. User clicks “Next” button.
6. System displays a menu including the necessary fields to be filled.
7. User fills the needed information which are Task Name (Necessary), Start Date (Necessary), End Date (Necessary), Task Type (Necessary), Priority Level (Optional), Detailed Info (Optional), Place (Optional), Attachment (Optional), Reminder (Optional), Recurrence (Optional), Card Color (Optional) and Other People (Optional).
8. Finally, User presses Finish button to see scheduled task on the calendar.
9. The use case ends successfully.

# Alternative Flows (Extensions)

\*a. At any time, User cancels the operation

1. User presses cancel button.

2. System returns to the current calendar.

## Create Automated Task (4a)

If in step 4 of the basic flow, the User wants to add an automated task then

1. User selects “Automated Task” from the dropdown menu.

2. System displays a menu including the necessary fields to be filled.

3. User fills the needed information which are Task Name (Necessary), Automation Start Date (Necessary), Automation End Date (Necessary), Recurrence (Optional), Duration (Necessary), Priority Level (Optional), Reminder (Optional), Event Category (Optional), Event Location (Optional), and Other People (Optional). Then, the user presses “Next” button.

3a. In step 3, User wants to add a recurrent task then:

1. User clicks “Recurrence” button.

2. System displays a menu including the necessary fields to be filled.

3. User fills the needed information which are Recurrence Start Date (Necessary), Recurrence End Date (Necessary) and Recurrence Pattern (Necessary)

4. User presses “OK” button to complete the operation.

4. System asks for automation preferences.

5. User fills in necessary fields of automation preferences form.

6. User presses Finish button to see scheduled task(s) on the calendar.

## Set Recurrence (7a)

If in step 7 of the basic flow, User wants to add a recurrent task then:

1. User clicks “Recurrence” button.

2. System displays a menu including the necessary fields to be filled.

3. User fills the needed information which are Recurrence Start Date (Necessary), Recurrence End Date (Necessary) and Recurrence Pattern (Necessary)

4. User presses “OK” button to complete the operation.

## Add Attachment (7b)

If in step 7 of the basic flow, the User wants to add attachment into the task then:

1. User presses “+” button besides to Attachment string.
2. User chooses the file which is desired to attached.
3. Finally, User presses to the Upload.
4. The use case resumes at step 7 again.

## Missed Information (8a)

If in step 8 of the basic flow, the User leaves the necessary information field blank and presses Finish button then:

1. System displays “All necessary information must be filled before forward” exception.

2. The use case resumes at step 8 again.

## Conflict with Existing Task (8b)

If in step 8 of the basic flow the User selects a date that overlaps an existing task, then:

1. System displays “Time conflict is occurred” exception.

2. The use case resumes at step 8 again.

# Post-conditions (Success Guarantee)

## Successful Task Addition

## User will see task on the calendar with defined time interval and in the detailed task view, app shows all the task details correct.

# Special Requirements

[SpReq:CT-1] Only .pdf, .doc, .docx, .xlsx, .jpg, .img, .png, .eml, and .msg formats are applicable for the attachment.