Use Cases

for

Task Hawk

Version 1.0 approved

Prepared by Jacob Thieret

Southeast Missouri State University

2/9/2023

Revision History

Name	Date	Reason For Changes	Version
Jacob Thieret	2/17/2023	Initial Documentation	1.0

Task Management

Use Case ID:	1.1		
Use Case Name:	Create Task		
Created By:	Jacob Thieret	Last Updated By:	Jacob Thieret
Date Created:	2/18/2023	Date Last Updated:	2/18/2023

Actor:	User
Description:	The user wants to create a new task to be added to their task list. The task will have a title and description, and may optionally include a due date, category, color, priority, reminder options, and recurrence options.
Preconditions:	1. The user is logged into the task management system and has access to the task creation feature.
Postconditions:	1. The new task is added to the user's task list with the specified details. 2. The task is displayed in weekly and calendar views
Priority:	High
Frequency of Use:	Multiple times a day
Flow of Events:	1. User selects the "Create Task" option from the Task Hawk's user interface. 2. The system displays a form for the user to enter details for the new task, including title, description, due date, category, color, priority, reminder options, and recurrence options. 3. User enters task details into the form. 4. User submits the form to create the new task. 5. The system validates the task details and adds the new task to the user's task list.
Alternative Flows:	1.1.AC.1 If the user does not complete required fields, the system will prompt them to fill in all required fields before submitting the form 1.1.AC.2 If the user enters invalid data into a field, the system will display an error message and prompt them to correct the data before submitting the form. 1.1.AC.3 If the user wants to create a category, the system will prompt the user to enter the name and color of the category.
Exceptions:	1.1.EX.1 If there is an error adding the task to the task list, the system will display an error message and prompt the user to try again.
Includes:	1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 1.10, 2.1, 2.2
Special Requirements:	Once a task is submitted and validated, the user should see the task appear in the calendar and weekly views in under 4-5 seconds, preferably faster.
Assumptions:	The user has access to the task create option in the applications UI
Notes and Issues:	How/where the task data is saved is TBD.

Use Case ID:	1.2		
Use Case Name:	Modify Task		
Created By:	Jacob Thieret	Last Updated By:	Jacob Thieret
Date Created:	2/18/2023	Date Last Updated:	2/18/2023

Actor:	User
Description:	The user wants to modify an existing task in their task list. The task's title, description, due date, category, color, priority, reminder options, and recurrence options may all be modified.
Preconditions:	1. The user is logged into the task management system and has an existing task that they wish to modify.
Postconditions:	1. The modified task is saved with the new details and is displayed in the appropriate views
Priority:	High
Frequency of Use:	Multiple times a day
Flow of Events:	1. User selects the task they wish to modify from their task list. 2. The system displays the task details in an editable form, including the title, description, due date, category, color, priority, reminder options, and recurrence options. 3. User modifies the desired fields in the form. 4. User submits the form to save the changes to the task. 5. The system validates the modified task details and updates the task in the user's task list.
Alternative Flows:	1.2.AC.1 If the user does not complete required fields, the system will prompt them to fill in all required fields before submitting the form 1.2.AC.2 If the user enters invalid data into a field, the system will display an error message and prompt them to correct the data before submitting the form. 1.2.AC.3 If the user selects "cancel" on the confirmation dialog, the system returns to the task list and the task is not modified. 1.2.AC.4 If the user wants to create a category, the system will prompt the user to enter the name and color of the category.
Exceptions:	1.2.EX.1 If there is an error adding the task to the task list, the system will display an error message and prompt the user to try again.
Includes:	1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 1.10, 2.1, 2.2
Special Requirements:	Once modifications are submitted and validated, the user should see the modified task in the calendar and weekly views in under 4-5 seconds, preferably faster.
Assumptions:	The user has an existing task that they wish to modify
Notes and Issues:	None

Use Case ID:	1.3		
Use Case Name:	Delete Task		
Created By:	Jacob Thieret	Last Updated By:	Jacob Thieret
Date Created:	2/18/2023	Date Last Updated:	2/18/2023

Actor:	User
Description:	The user wants to delete a task from their task list.
Preconditions:	1. The user is logged into the task management system and has an existing task that they wish to delete.
Postconditions:	1. The deleted task is removed from the user's task list and is no longer displayed in any views.
Priority:	High
Frequency of Use:	Multiple times a day
Flow of Events:	1. User selects the task they wish to delete from their task list. 2. The system displays a confirmation dialog to ensure the user wants to delete the task. 3. User confirms the deletion by selecting the "delete" button. 4. The system removes the task from the user's task list and all associated data and views.
Alternative Flows:	1.3.AC.1 If the user selects "cancel" on the confirmation dialog, the system returns to the task list and the task is not deleted.
Exceptions:	1.3.EX.1 If there is an error deleting the task, the system will display an error message and prompt the user to try again.
Includes:	2.1, 2.2
Special Requirements:	Weekly and Calendar views should be updated within 4-5 seconds after the task is deleted
Assumptions:	The user has an existing task they'd wish to delete
Notes and Issues:	How to handle user mistakes or accidental deletion is TBD

Use Case ID:	1.4		
Use Case Name:	Categorize Task		
Created By:	Jacob Thieret	Last Updated By:	Jacob Thieret
Date Created:	2/18/2023	Date Last Updated:	2/18/2023

Actor:	
Description:	The user wants to categorize a task in order to better organize and filter their task list.
Preconditions:	1. The user has an existing task that they wish to categorize, or the user is in the task creation menu
Postconditions:	1. The task is assigned to a category and is associated with that category in the task list and other views.
Priority:	Medium
Frequency of Use:	Multiple times a day
Flow of Events:	1. User selects the task they wish to categorize and clicks the modify option, or clicks the create task option. 2. The system displays a list of available categories or allows the user to create a new category. 3. User selects an existing category or creates a new category. 4. The system assigns the task to the selected category and updates the task list and other views.
Alternative Flows:	1.4.AC.1 If the user does not select a category, the task remains uncategorized. 1.4.AC.1 If the user cancels out of the category selection screen, the task remains uncategorized
Exceptions:	If there is an error categorizing the task, the system will display an error message and prompt the user to try again.
Includes:	1.1, 1.2, 1.8, 1.10, 2.1, 2.2
Special Requirements:	Filtered calendar views should refresh within 4-5 seconds.
Assumptions:	The user has an existing task that they wish to categorize. Users are able to create categories from the task create/modify screens.
Notes and Issues:	None

Use Case ID:	1.5		
Use Case Name:	Set due date for task		
Created By:	Jacob Thieret	Last Updated By:	Jacob Thieret
Date Created:	2/18/2023	Date Last Updated:	2/18/2023

Actor:	User
Description:	The user wants to assign a due date to a task in order to track its progress and prioritize their workload.
Preconditions:	1. User has an existing task that they wish to assign a due date to, or is in the task creation screen
Postconditions:	1. The task is assigned a due date and is displayed in the user's task list and other views, with an indicator of the deadline.
Priority:	Medium
Frequency of Use:	Multiple times a day
Flow of Events:	1. User selects the task they wish to assign a due date to from their task list, or opens the task creation menu. 2. The system displays a date picker or allows the user to enter the due date manually. 3. User selects a due date or enters a due date manually. 4. The system assigns the due date to the task and updates the task list and other views to display the deadline.
Alternative Flows:	1.5.AC.1 If the user does not select or enter a due date, the task remains without a deadline. 1.5.AC.2 If the user cancels out of the due date selection screen, the task remains without a deadline.
Exceptions:	1.5.EC.1 If there is an error assigning the due date to the task, the system will display an error message and prompt the user to try again.
Includes:	1.1, 1.2, 1.6, 2.1, 2.2
Special Requirements:	Tasks should appear on the day selected as the due date in all associated views.
Assumptions:	The user has an existing task that they wish to assign a due date to.
Notes and Issues:	None

Use Case ID:	1.6		
Use Case Name:	Set reminder for task		
Created By:	Jacob Thieret	Last Updated By:	Jacob Thieret
Date Created:	2/18/2023	Date Last Updated:	2/18/2023

Actor:	User
Description:	The user wants to set a reminder for a specific task in their task list. The reminder should notify the user before the task's due date and time. The user should be able to choose the notification method, such as email or push notification.
Preconditions:	1. The user has access to the task with a due date and time 2. The user is in the task create menu or task modify menu
Postconditions:	1. The reminder is set for the selected task, and the user receives a notification based on their preferred method.
Priority:	Medium
Frequency of Use:	Multiple times per week
Flow of Events:	1. The user selects a task from their task list that they want to set a reminder for, or creates a task. 2. The user opens the task details screen and selects the "Set Reminder" option. 3. The user selects an option for the amount of time for the reminder to be sent before the tasks data and time values. 4. The user selects a preferred method for the reminder, such as email or push notification. 5. The user saves the reminder settings, or submits the task. 6. The system confirms that the reminder has been set for the task.
Alternative Flows:	1.6.AC.1 If the user selects a reminder time that has already passed, the system will display an error message and prompt the user to choose a new time. 1.6.AC.2 If the user doesn't select a reminder option, no reminders will be set.
Exceptions:	1.6.EX.1 If the system is unable to set the reminder, the system will display an error message and prompt the user to try again.
Includes:	1.1, 1.2, 1.5
Special Requirements:	None
Assumptions:	The task to be assigned a reminder has a value for date and time.
Notes and Issues:	None

Use Case ID:	1.7		
Use Case Name:	Mark task as complete		
Created By:	Jacob Thieret	Last Updated By:	Jacob Thieret
Date Created:	2/18/2023	Date Last Updated:	2/18/2023

Actor:	User
Description:	The user wants to mark a task as complete. Once the task is marked as complete, it will no longer appear in the user's task list, any associated reminders will be disabled, and removed from any associated views. The task will be sent to a 'completed' category, which displays recent tasks completed within the past week, where the user can also unmark the task as complete, and will reappear in the users task list.
Preconditions:	1. Users have at least one task in their task list.
Postconditions:	1. The selected task is marked as complete and is removed from the user's task list. 2. The completed task appears in the completed tasks list
Priority:	High
Frequency of Use:	Multiple times a day
Flow of Events:	1. User selects the task they wish to mark as complete from their task list. 2. User clicks on the "Mark as Complete" button. 3. The system moves the task to the completed tasks list 4. The system removes the task from the user's task list and disables any associated reminders.
Alternative Flows:	None
Exceptions:	1.7.EX.1: If there is an error marking the task as complete, the system displays an error message and prompts the user to try again.
Includes:	2.1, 2.2
Special Requirements:	None
Assumptions:	The user has access to the "Mark as Complete" button for each task in their task list.
Notes and Issues:	None

Use Case ID:	1.8		
Use Case Name:	Assign color to category		
Created By:	Jacob Thieret	Last Updated By:	Jacob Thieret
Date Created:	2/18/2023	Date Last Updated:	2/18/2023

Actor:	User
Description:	The user wants to assign a color to a category to help visually distinguish tasks that belong to that category.
Preconditions:	1. The user has at least one category created
Postconditions:	1. The selected category is displayed with the assigned color
Priority:	Low
Frequency of Use:	Multiple times a month
Flow of Events:	1. User selects the "Categories" option from the Task Hawk's user interface. 2. The system displays a list of categories. 3. User selects the category for which they want to assign a color. 4. The system displays the details of the selected category. 5. User selects the "Assign Color" option. 6. The system displays a color palette. 7. User selects a color from the palette. 8. The system updates the selected category to display the assigned color.
Alternative Flows:	1.8.AC.1 If the user selects an invalid category, the system displays an error message and prompts the user to select a valid category. 1.8.AC.2 If the user cancels the color selection, the system does not assign a new color to the selected category.
Exceptions:	1.8.EX.1 If there is an error updating the category color, the system displays an error message and prompts the user to try again.
Includes:	1.1, 1.2, 1.10, 1.11
Special Requirements:	User's should be able to select from a color palette or enter their own custom hex values.
Assumptions:	The user has at least one category
Notes and Issues:	None

Use Case ID:	1.9		
Use Case Name:	Set priority of task		
Created By:	Jacob Thieret	Last Updated By:	Jacob Thieret
Date Created:	2/18/2023	Date Last Updated:	2/18/2023

Actor:	User
Description:	The user wants to set a priority level for a task they have created in the task management system. The priority level will help the user to better organize and manage their tasks based on their importance or urgency.
Preconditions:	1. The user has one or more task created 2. The user is in the task create menu or task modify menu
Postconditions:	1. The priority level for the task is updated and displayed in the task list and other views
Priority:	Medium
Frequency of Use:	Multiple times a day
Flow of Events:	1. The user selects the task they want to set a priority level for. 2. The user selects the "Set Priority" option from the task create/modify menus. 3. The system displays a dropdown menu or other input method to allow the user to select a priority level for the task. 4. The user selects the desired priority level from the options available. 5. The system saves the updated priority level for the task. 6. The system refreshes the task list and other views to display the updated priority level.
Alternative Flows:	1.9.AC.1 If the user cancels the priority level selection, or leaves it blank, the system does not update/assign the task priority level.
Exceptions:	1.9.EX.1 If the , the system displays an error message and prompts the user to try again.
Includes:	1.1, 1.2
Special Requirements:	None
Assumptions:	User is able to access the create/modify task options
Notes and Issues:	None

Use Case ID:	1.10		
Use Case Name:	Create Category		
Created By:	Jacob Thieret	Last Updated By:	Jacob Thieret
Date Created:	2/18/2023	Date Last Updated:	2/18/2023

Actor:	User
Description:	The user wants to create a new category to group similar tasks in their task list. The category will have a name and a color.
Preconditions:	The user has access to the category creation feature
Postconditions:	The new category is added to the user's category list with the specified details.
Priority:	Low
Frequency of Use:	As needed
Flow of Events:	1. User selects the "Create Category" option from the Task Hawk's user interface . 2. The system displays a form for the user to enter details for the new category, including name and color. 3. User enters category details into the form. 4. User submits the form to create the new category. 5. The system validates the category details and adds the new category to the user's category list.
Alternative Flows:	1.10.AC.1 If the user does not complete required fields, title and color, the system will prompt them to fill in all required fields before submitting the form. 1.10.AC.2 If the user enters invalid data into a field, the system will display an error message and prompt them to correct the data before submitting the form.
Exceptions:	1.10.EX.1 If there is an error adding the category to the category list, the system will display an error message and prompt the user to try again.
Includes:	2.1, 2.2
Special Requirements:	None
Assumptions:	The user has access to the create category option
Notes and Issues:	None

Use Case ID:	1.11		
Use Case Name:	Modify Category		
Created By:	Jacob Thieret	Last Updated By:	Jacob Thieret
Date Created:	2/18/2023	Date Last Updated:	2/18/2023

Actor:	User
Description:	The user wants to modify an existing category. The category may have a name, color, and description. Modifying the category should change its name, color, or description as necessary.
Preconditions:	1. Users have access to the category modification feature.
Postconditions:	1. The modified category is updated in the user's category list.
Priority:	Medium
Frequency of Use:	As needed
Flow of Events:	1. User selects the "Modify Category" option from the Task Hawk's user interface. 2. The system displays a list of existing categories. 3. User selects the category they want to modify. 4. The system displays a form with the existing category details, including name, color, and description. 5. User modifies the details for the category. 6. User submits the form to save the changes. 7. The system validates the changes and updates the category in the user's category list.
Alternative Flows:	1.11.AC.1 If the user does not complete required fields, the system will prompt them to fill in all required fields before submitting the form. 1.11.AC.2 If the user enters invalid data into a field, the system will display an error message and prompt them to correct the data before submitting the form.
Exceptions:	1.11.EX.1 If there is an error updating the category in the category list, the system will display an error message and prompt the user to try again.
Includes:	1.1, 1.2
Special Requirements:	Once the changes to the category are submitted and validated, the user should see the updated category in their category list within 4-5 seconds, preferably faster
Assumptions:	The user has access to the category modification feature in the application's UI.
Notes and Issues:	None

Use Case ID:	1.12		
Use Case Name:	Delete Category		
Created By:	Jacob Thieret	Last Updated By:	Jacob Thieret
Date Created:	2/18/2023	Date Last Updated:	2/18/2023

Actor:	User
Description:	The user wants to delete a category from their task management system. This category may contain zero or more tasks.
Preconditions:	1. The user has access to the category deletion feature. 2. The category to be deleted exists in the user's task management system.
Postconditions:	1. The category is removed from the user's task management system. 2. If the category contained any tasks, those tasks are now uncategorized.
Priority:	Medium
Frequency of Use:	As needed
Flow of Events:	1. The user selects the "Delete Category" option from the user interface. 2. The system displays a confirmation message asking the user to confirm the deletion. 3. The user confirms the deletion. 4. The system removes the category and any associated tasks from the user's task management system.
Alternative Flows:	1.12.AC.1: If the user cancels the deletion, the system returns the user to the previous screen without deleting the category.
Exceptions:	1.12.EX.1: If the system is unable to delete the category, an error message is displayed and the system does not delete the category.
Includes:	None
Special Requirements:	None
Assumptions:	The user has access to their category list and the "delete category" option
Notes and Issues:	None

Use Case ID:	1.13		
Use Case Name:	Set recurrence for task		
Created By:	Jacob Thieret	Last Updated By:	Jacob Thieret
Date Created:	2/18/2023	Date Last Updated:	2/18/2023

Actor:	User
Description:	The user wants to set up a recurring task, where the task will automatically repeat at a specified interval. This feature is helpful for tasks that need to be completed on a regular basis such as taking medication or paying bills.
Preconditions:	1. The user is modifying an existing task, or is in the process of creating a new task
Postconditions:	1. Task is set to recur at the specified interval.
Priority:	Medium
Frequency of Use:	Multiple times per week
Flow of Events:	1. User selects a task they want to set up for recurrence, or is prompted with the option when creating a task. 2. User clicks on the 'Set Recurrence' button in the task details screen. 3. User is presented with a list of possible recurrence intervals, such as daily, weekly, monthly, etc. 4. User selects the desired recurrence interval. 5. Users are prompted to select a start and end date for the recurring task. 6. User confirms the recurrence details. 7. System sets the task to recur at the specified interval.
Alternative Flows:	1.12.AC.1: If the user decides not to set a recurrence, they can cancel the operation, or leave the field blank and return to the task details screen or submit the task without recurrence options.
Exceptions:	1.12.EX.1: If the system is unable to delete the category, an error message is displayed and the system does not delete the category.
Includes:	1.1, 1.2
Special Requirements:	1. Once the task is marked as completed, the task should not be removed from the user's task list. The recurring task should appear on the calendar/list for the remaining recurrences on its specified recurrence dates. 2. Recurring tasks should be clearly marked in the task list and details screen. 3. Recurring tasks should be modifiable in case the user wants to adjust the recurrence interval or end date.
Assumptions:	1. The system's date and time are accurate.
Notes and Issues:	None

Task Viewing

Use Case ID:	2.1		
Use Case Name:	Weekly view		
Created By:	Jacob Thieret	Last Updated By:	Jacob Thieret
Date Created:	2/18/2023	Date Last Updated:	2/18/2023

Actor:	User, System
Description:	The user wants to view their tasks in a weekly format, organized by the day of the week. This will be one of the two main views to display tasks. The top bar should contain the current week days(mon-sun) with the associated days of the month. Top bar should be horizontally scrollable, and the vertically scrollable task list should update according to the week selected in the top bar.
Preconditions:	1. The user is logged into the task management system and has access to the weekly view feature
Postconditions:	2. The user can see their tasks displayed in a weekly format, organized by day of the week.
Priority:	High
Frequency of Use:	Multiple times a day
Flow of Events:	1. User selects the "Weekly View" option from the Task Hawk's user interface. 2. The system will display the current week in the top bar, highlighting the current day. On the lower half of the screen the user is presented the tasks for the selected week in the top bar 3. The user can scroll through the week and view tasks for other days of the week. 4. The user can select a task to view its details, modify or delete the task, or mark the task as completed.
Alternative Flows:	2.1.AC.1 If the user has no tasks scheduled for a particular day of the week, the system will display a message indicating that there are no tasks for that day. 2.1.AC.2 If the user has a large number of tasks scheduled for a particular day of the week, the system may display a scrollable list of tasks for that day.
Exceptions:	2.1.EX.1 If there is an error displaying the weekly view, the system will display an error message and prompt the user to try again.
Includes:	1.1, 1.2, 1.3, 1.7, 2.2
Special Requirements:	The weekly view should load quickly, preferably in under 2-3 seconds, and should be easy to navigate.
Assumptions:	The user has access to the weekly view option in the application's UI.
Notes and Issues:	None

Use Case ID:	2.2		
Use Case Name:	Calendar view		
Created By:	Jacob Thieret	Last Updated By:	Jacob Thieret
Date Created:	2/18/2023	Date Last Updated:	2/18/2023

Actor:	User, system
Description:	The user wants to view their tasks in a calendar format. The calendar view will display all tasks with their due dates and times for the selected month. The boxes for every day for the month should have some sort of visual input indicating the number of tasks for the day, as well as the color of the category(s). The user should be able to show/hide tasks based on category and/or priority. User can tap on a day to bring up the list of tasks for that day
Preconditions:	1. The user has created tasks with due dates.
Postconditions:	1. The user can see all tasks with their due dates in a calendar format.
Priority:	High
Frequency of Use:	Multiple times a day
Flow of Events:	1. The user selects "Calendar View" from the task management system's menu. 2. The system displays a calendar with the current month and all tasks with due dates marked on the corresponding dates. 3. The user can click on a task to see additional information or to mark the task as complete.
Alternative Flows:	2.2.AC.1 If there are no tasks with due dates, the calendar will be displayed but there will be no tasks marked on it.
Exceptions:	None
Includes:	1.1, 1.2, 2.1
Special Requirements:	1. The default calendar view should display all tasks with due dates, regardless of their priority or category. 2. The calendar should be updated in real-time as the user creates, modifies, or deletes tasks. 3. The calendar should allow the user to easily navigate between months.
Assumptions:	1. The user has created tasks with due dates. 2. The user is familiar with navigating a calendar interface.
Notes and Issues:	None

Use Case ID:	2.3		
Use Case Name:	Filter Tasks by category		
Created By:	Jacob Thieret	Last Updated By:	Jacob Thieret
Date Created:	2/18/2023	Date Last Updated:	2/18/2023

Actor:	User, System
Description:	The user wants to filter their task list by a specific category to only see tasks that are associated with that category.
Preconditions:	1. The user has at least one category assigned to their tasks.
Postconditions:	1. The user's task list is filtered to display only tasks associated with the selected category.
Priority:	Low
Frequency of Use:	Multiple times a week
Flow of Events:	1. User navigates to the "Filter Tasks" option in the weekly or calendar views. 2. The system prompts user to select "Filter by Category" and "Filter by Priority" 3. The system displays a list of available categories. 4. User selects the category they want to filter by. 5. The system updates the task list to display only tasks that are associated with the selected category.
Alternative Flows:	2.3.AC.1 If the user has not assigned any categories to their tasks, the system will display an error message informing them of the need to categorize their tasks before using the filter option.
Exceptions:	2.3.EX.1 If there is an error filtering the task list, the system will display an error message and prompt the user to try again.
Includes:	2.1, 2.2
Special Requirements:	The system should allow the user to easily clear the filter and return to the unfiltered task list.
Assumptions:	The user has assigned categories to their tasks.
Notes and Issues:	None

Use Case ID:	2.4		
Use Case Name:	Filter Tasks by priority		
Created By:	Jacob Thieret	Last Updated By:	Jacob Thieret
Date Created:	2/18/2023	Date Last Updated:	2/18/2023

Actor:	User, System
Description:	The user wants to filter their task list to display only tasks of a certain priority level.
Preconditions:	1. The user has access to the task list with priority information.
Postconditions:	1. The system displays only the tasks that meet the selected priority level.
Priority:	Low
Frequency of Use:	Multiple times a week
Flow of Events:	1. User navigates to the "Filter Tasks" option in the weekly or calendar views. 2. The system prompts user to select "Filter by Category" and "Filter by Priority" 3. User selects the "Filter by Priority" option. 4. The system displays a list of priority levels. 5. User selects a priority level to filter by. 6. The system filters the task list to display only tasks of the selected priority level.
Alternative Flows:	2.4.AC.1 If there are no tasks with the selected priority level, the system will display a message indicating that there are no tasks to display.
Exceptions:	2.4.EX.1 If there is an error filtering the task list, the system will display an error message and prompt the user to try again.
Includes:	2.1, 2.2
Special Requirements:	None
Assumptions:	The user has access to the task list with priority information in the application's UI
Notes and Issues:	None

Calendar Objects

Use Case ID:	3.1		
Use Case Name:	Export Calendar		
Created By:	Jacob Thieret	Last Updated By:	Jacob Thieret
Date Created:	2/18/2023	Date Last Updated:	2/18/2023

Actor:	User, System
Description:	The user wants to export their task data to external formats such as CSV for backup purposes or to share with others.
Preconditions:	1. The user has a calendar with events and tasks. 2. The user has access to an external storage or email inbox.
Postconditions:	1. The user has successfully exported their calendar data to an external format. 2. The exported file is saved in a secure manner.
Priority:	High
Frequency of Use:	Multiple times a month
Flow of Events:	1. The user selects the "Export" option in the application. 2. The user is presented with options for file format and location to save the file. 3. The user selects the desired format and location to save the file. 4. If password protection is desired, the user enters a password to encrypt the file. 5. The application exports the calendar data to the selected format and saves it to the specified location. 6. The application notifies the user that the export was successful.
Alternative Flows:	3.1.AC.1 If the user chooses not to password protect the exported file, the application skips the password entry step.
Exceptions:	3.1.EX.1 If there is an error during the export process, the application notifies the user of the error and returns to the file format selection step.
Includes:	2.1, 2.2, 3.3, 3.4
Special Requirements:	1. The exported file must be secure and not readable in plain-text. 2. The exported file should be easily importable to other applications. 3. If password protection is used, the user must be prompted to enter the password upon importing the file.
Assumptions:	1. The user has basic knowledge of how to save and share files on their device. 2. The user has access to an external storage medium or email inbox.
Notes and Issues:	The method/file format for calendar objects is TBD. Method to save files to mobile devices is unknown currently. It is not known if it's possible to password lock an exported file, and how to handle secure export/import of data.

Use Case ID:	3.2		
Use Case Name:	Import Calendar		
Created By:	Jacob Thieret	Last Updated By:	Jacob Thieret
Date Created:	2/18/2023	Date Last Updated:	2/18/2023

Actor:	User, System
Description:	The user imports external calendar data and saves it in the users calendar list where they are provided with options to merge/overlap one or more calendars.
Preconditions:	1. The user must have access to the external calendar data they wish to import. 2. The user must have an existing account in the application.
Postconditions:	1. The imported calendar is stored as a separately viewable list of tasks
Priority:	High
Frequency of Use:	Multiple times a month
Flow of Events:	1. User selects the option to import external calendar data from the application's menu. 2. The application prompts the user to select the file they wish to import. 3. The user selects the file they wish to import. 4. The application reads the file and verifies that it is in a supported format. 5. The user selects the calendar object they wish to merge the imported data with. 6. The application imports the data and merges it with the selected calendar object. 7. The application displays a message confirming the successful import and merge of the external calendar data.
Alternative Flows:	3.2.AC.1 If the selected file is not in a supported format, the application displays an error message and prompts the user to select a different file. 3.2.AC.2 If the user cancels the import process, the application returns the user to the previous screen.
Exceptions:	3.2.EX.1 If the imported data contains errors, the application displays an error message and prompts the user to select a different file.
Includes:	2.1, 2.2, 3.4, 4.2, 4.3
Special Requirements:	If the imported file is password protected, the user should be prompted to enter the password.
Assumptions:	The imported calendar data is in a supported format.
Notes and Issues:	Unsure of how to detect whether an imported file is password protected or not. Unsure of how to handle importing/exporting tasks with categories. Will categories/priority/color etc be carried over?

Use Case ID:	3.3		
Use Case Name:	Merge Calendar View		
Created By:	Jacob Thieret	Last Updated By:	Jacob Thieret
Date Created:	2/18/2023	Date Last Updated:	2/18/2023

Actor:	User, System
Description:	The user wants to merge or overlap one or more calendar objects. Users can select from their list of imported or created calendars, and mark the calendars to be overlapped with a checkbox.
Preconditions:	1. User has one or more calendar objects that they want to merge
Postconditions:	1. Selected calendar objects are overlapped in a new calendar view that displays all tasks 2. Users have the option to save the merged calendar as a new calendar, or click "Ok" to leave the merged view.
Priority:	High
Frequency of Use:	Multiple times a month
Flow of Events:	1. User opens the application and navigates to the "Calendar" section 2. User selects the "Merge view" option 3. User is prompted to select which calendar objects they want to merge 4. User selects the desired calendar objects 5. Application merges the selected calendar objects and displays them to the user 6. The user can either save the merged calendars to a new calendar object, or can click "Ok" to leave the overlapped view.
Alternative Flows:	3.3.AC.1 If the user does not have multiple calendar views, the "Merge view" option will not be available 3.3.AC.2 If the user exits the application before completing the merge view process, any selected calendar views will not be merged
Exceptions:	3.3.EX.1 If there is an error in merging the calendar views, the application will display an error message and prompt the user to try again later
Includes:	3.1, 3.2, 3.4
Special Requirements:	1. Merged calendar view should clearly display which events come from which calendar view 2. User should be able to toggle individual calendar views on/off within the merged view
Assumptions:	1. User has multiple calendar views that they want to merge 2. User has access to all calendar views they want to merge 3. User is familiar with navigating the application to the "Calendar" section
Notes and Issues:	Application should have clear instructions for how to select which calendar objects to marge

Use Case ID:	3.4		
Use Case Name:	Manage Calendar List		
Created By:	Jacob Thieret	Last Updated By:	Jacob Thieret
Date Created:	2/18/2023	Date Last Updated:	2/18/2023

Actor:	User
Description:	This use case describes the functionality to manage the list of calendar objects. The user should be able to create, modify, and delete calendar objects in the list.
Preconditions:	1. User has created at least one calendar object.
Postconditions:	2. The list of calendar objects is modified as per the user's actions
Priority:	Medium
Frequency of Use:	As needed
Flow of Events:	1. User navigates to the list of calendar objects 2. User imports/exports calendar objects, or create a new calendar object with no tasks by default. 3. User can modify existing calendar objects, i.e. the calendar name and accent color 4. User can delete existing calendars
Alternative Flows:	3.4.AC.1 If the user tries to create two calendar objects with the same name, the system will notify the user of the issue and prompt the user to enter a different name.
Exceptions:	None
Includes:	3.1, 3.2, 3.3
Special Requirements:	None
Assumptions:	User has a basic understanding of how to create/modify/delete tasks, same as task creation.
Notes and Issues:	None

Security

Use Case ID:	4.1		
Use Case Name:	User Data Encryption		
Created By:	Jacob Thieret	Last Updated By:	Jacob Thieret
Date Created:	2/18/2023	Date Last Updated:	2/18/2023

Actor:	User, System
Description:	Security measures required to ensure the encryption of user data.
Preconditions:	1. The app must have access to user data that needs to be encrypted. 2. The encryption method must be decided and implemented.
Postconditions:	1. The user data must be encrypted. 2. The user data should only be accessible to authorized personnel.
Priority:	High
Frequency of Use:	Continuous
Flow of Events:	1. User data is collected by the app. 2. App encrypts user data using the designated encryption method. 3. Encrypted user data is stored in the app's database. 4. Encrypted user data is only accessible by authorized personnel.
Alternative Flows:	4.1.AC.1 If encryption fails for any reason, the app should notify the user and not store the unencrypted data.
Exceptions:	4.1.EX.1 If the encryption key is lost, the user data may become permanently inaccessible.
Includes:	
Special Requirements:	1. The encryption key must be secure and not be accessible to unauthorized personnel. 2. The encryption method should be reviewed regularly to ensure it meets current security standards.
Assumptions:	1. The app is running on secure and trusted hardware. 2. The app is developed with security as a high priority.
Notes and Issues:	The app should provide an easy and secure way for the user to reset their encryption key if needed.

Use Case ID:	4.2		
Use Case Name:	Exported files password protection		
Created By:	Jacob Thieret	Last Updated By:	Jacob Thieret
Date Created:	2/18/2023	Date Last Updated:	2/18/2023

Actor:	User
Description:	The user wants to add password protection to the exported task data files to ensure the security and privacy of the user's data.
Preconditions:	1. User must have a selected calendar to export
Postconditions:	2. Exported calendar is password protected, and can only be accessed with users with the password. Exported calendars with password protection will be encrypted using that password, and non-password protection exports will be saved in plaintext.
Priority:	High
Frequency of Use:	As needed
Flow of Events:	1. The user exports task data to a file format. 2. The user selects an option to add password protection to the exported file. 3. The user is prompted to enter a password for the exported file. 4. The user confirms the password by entering it a second time. 5. The exported file is saved with the password protection and can only be opened with the password.
Alternative Flows:	4.2.AC.1 If the user enters a weak password, they will be prompted to enter a stronger password. 4.2.AC.2 If the user enters two different passwords, they will be prompted to re-enter the same password to confirm.
Exceptions:	4.2.EX.1 If the exported file format does not support password protection, the user will be notified and the option to add password protection will not be available.
Includes:	3.1, 3.2
Special Requirements:	The password entered by the user must meet certain criteria such as minimum length and complexity to ensure security.
Assumptions:	The user has an existing calendar object they want to export
Notes and Issues:	File formats supporting password protection are unknown

Use Case ID:	4.3		
Use Case Name:	Exported files encryption		
Created By:	Jacob Thieret	Last Updated By:	Jacob Thieret
Date Created:	2/18/2023	Date Last Updated:	2/18/2023

Actor:	User, System
Description:	The system should encrypt files exported by the user.
Preconditions:	1. User has access to the task management app's "Export" functionality 2. User has selected the option to export their task data to a file
Postconditions:	1. The exported file is encrypted and saved to the specified location
Priority:	High
Frequency of Use:	As needed
Flow of Events:	1. User selects the option to export their task data to a file 2. User is prompted to specify the export file location 3. User is prompted to enter a password for the exported file 4. The task management app encrypts the exported file with the password specified by the user 5. The encrypted file is saved to the specified location 6. The task management app displays a message to the user indicating that the export was successful and the location of the exported file
Alternative Flows:	4.3.AC.1 If the user cancels the export, the system will cancel the export and return the user to the task list view
Exceptions:	4.3.EX.1 If the user enters an incorrect password, the system will display an error message indicating that the password is incorrect and prompts the user to enter the correct password
Includes:	4.2
Special Requirements:	1. The encryption algorithm used must be secure 2. The app should not store the password entered by the user in plain text 3. The app should display a warning to the user that they should not forget the password, as the encrypted file cannot be accessed without it
Assumptions:	1. The user has a basic understanding of encryption and password protection 2. The user has access to the export functionality and knows how to use it 3.
Notes and Issues:	The app should include documentation on the encryption algorithm used and how it secures the data in the exported file. Encryption method is TBD

Use Case ID:	4.4		
Use Case Name:	Password protected task list		
Created By:	Jacob Thieret	Last Updated By:	Jacob Thieret
Date Created:	2/18/2023	Date Last Updated:	2/18/2023

Actor:	User
Description:	The user wants to create a password-protected task list, where a user can set a password to protect access to the task list. This feature can help keep task information private and secure, especially if using a shared device.
Preconditions:	1.　User has created a task list
Postconditions:	1.　Task list is password-protected 2.　Only users who enter the correct password can access the task list
Priority:	Medium
Frequency of Use:	Multiple times a week
Flow of Events:	1.　User selects the task list they want to password-protect 2.　User selects the option to set a password 3.　User enters and confirms a password 4.　Task list is now password-protected
Alternative Flows:	4.4.AC.1 If the user selects to cancel setting the password, the task list will remain unprotected.
Exceptions:	4.4.EX.1 If the user enters an incorrect password, they will not be able to access the task list.
Includes:	1.1, 1.2
Special Requirements:	1.　The password should be masked as the user enters it. 2.　The user should be able to change the password at any time. 3.　The password should be stored securely, such as using hashing or other encryption methods.
Assumptions:	User is the only person who knows the password. The app is capable of storing encrypted passwords securely.
Notes and Issues:	Users should be reminded to remember their password as it cannot be recovered or reset by the app.