*[The Test Case ID should be unique. In addition, the name of each Test Case should reflect the intent of the test case, ideally expressed as a Boolean condition.]*

**TG-UCT-001 – All Task Options Entered Will Be Shown True and Adequacy by The System:**

Description: The user will be able to decide all the options according to their preferences while creating new task.

Related Use Case: Create a Task

Pre-conditions: The conditions below must be provided before start to test:

* System must be operational and users can login to the system.
* User’s internet connection must be stable during the process.
* All predefined fields must be provided to the users.
* User needs to start the creating new task process.

Post-conditions: After the test case ends these conditions must be true:

* + The task created successfully by the system.
  + System will be able to show task.
  + User will be able to enter into task that created by the user.
  + The system will be able to saved all the data required\* to enter by the user with successfully.
  + User can save the options without entering optional choices.

Data required: Task Type, Priority Level, Recurrence, Task Name, Start Date, End Date, Detailed Info, Place, Attachment, Reminder

**TG-UCT-002 – When Must Task Fields Not Entered by The User, System Will Be Able to Create Exception:**

Description: The system will be able to create exception if the user doesn’t fill all the necessary options to finish creating task.

Related Use Case: Create a Task

Pre-conditions: The conditions below must be provided before start to test:

* User’s internet connection must be stable during the process.
* All predefined options should be provided to the users.
* User must be started the creating new task process.
* User needs to not fill all the necessary options

Post-conditions: After the test case ends these conditions must be true:

* System should be able to create exception message when user presses the finish button.
* The System should not advance the process before user fills non-optional options.
* System should repeat the message until user fill all the necessary options.

Data required: Task Type, Priority Level, Recurrence, Task Name, Start Date, End Date, Detailed Info, Place, Attachment, Reminder

**TG-UCT-003 – When Character Limits Exceed by The User System Will Be Able to Create Exception**

Description: The system will be able to ensure that the user can’t exceed task name (64 char.) and Detailed Info (128 char.) areas limits.

Related Use Case: Create a Task

Pre-conditions: The conditions below must be provided before start to test:

* User’s internet connection must be stable during the process.
* All predefined fields must be provided to the users.
* User should be started the creating new task process.
* User exceed the character limits of task name and detailed info options.

Post-conditions: After the test case ends these conditions must be true:

* System should be able to create exception message when user presses the finish button.
* System should repeat the message until user enter the information between character limits.

Data required: Task Name, Detailed Info

**TG-UCT-004 – All User Profile Options Entered Will Be Shown True and Adequacy by The System:**

Description: The user will be able to decide all the user information according to themselves.

Related Use Case: Set User Profile

Pre-conditions: The conditions below must be provided before start to test:

* System must be operational and users can login to the system.
* User’s internet connection must be stable during the process.
* All predefined fields must be provided to the users.
* User must press to the update user profile button.

Post-conditions: After the test case ends these conditions must be true:

* + The user profile updated successfully by the system.
  + System will be able to show new profile information.
  + User will be able to enter into task that created by the user.
  + The system will be able to save all the data required\* to enter by the user with successfully.

Data required: Username, Firs Name, Last Name, Country, City, Interests, Birth Date, Biography, Profile Picture.

**TG-UCT-005 – All Updated Task Options Will Be Shown True and Adequacy by The System:**

Description: The user will be able to update all the options according to their preferences.

Related Use Case: Update Task

Pre-conditions: The conditions below must be provided before start to test:

* System must be operational and users can login to the system.
* User’s internet connection must be stable during the process.
* All predefined fields must be provided to the users with prefilled information by the user.
* User needs to start the updating task process.

Post-conditions: After the test case ends these conditions must be true:

* + The task updated successfully by the system.
  + System will be able to show updated task information.
  + User will be able to enter into task that updated by the user.
  + The system will be able to save all the data required\* to enter by the user with successfully.
  + User can save the options without changing all the fields.

Data required: Task Type, Priority Level, Recurrence, Task Name, Start Date, End Date, Detailed Info, Place, Attachment, Reminder

**TG-UCT-006 – Created Task Will Be Deleted by The User:**

Description: The user will be able to delete the task.

Related Use Case: Remove Task

Pre-conditions: The conditions below must be provided before start to test:

* System must be operational and users can login to the system.
* User’s internet connection must be stable during the process.
* System will be able to show task.

Post-conditions: After the test case ends these conditions must be true:

* + The task deleted successfully by the system.

Data required: Task

**TG-UCT-007 – All Users Can Send Add Friend Request Properly by The System:**

Description: The use case describes how the user adds other users as friends from other user’s profile page.

Related Use Case: Add Friend

Pre-conditions: The conditions below must be provided before start to test:

* System must be operational and users can login to the system.
* User’s internet connection must be stable during the process.
* System should show all results when user browses a user.

Post-conditions: After the test case ends these conditions must be true:

* + Friend request will be sent to the selected user successfully by the system.
  + User will be able to see request sent label on the requested user profile page.

Data required: User Name

**TG-UCT-008 – All Users Can Send Add Friend Request Properly by The System (from Browsing Page):**

Description: The use case describes how the user adds other users as friends from browser page.

Related Use Case: Add Friend

Pre-conditions: The conditions below must be provided before start to test:

* System must be operational and users can login to the system.
* User’s internet connection must be stable during the process.
* System should show all results when user browses a user.

Post-conditions: After the test case ends these conditions must be true:

* + Friend request will be sent to the desired search result user successfully by the system.
  + User will be able to see request sent label under the requested user.

Data required: User Name

**TG-UCT-009 – Friendship Requests Are Handling by The Users:**

Description: The use case describes how the user handles other users’ friendship requests.

Related Use Case: Handle Friendship Request

Pre-conditions: The conditions below must be provided before start to test:

* System must be operational and users can login to the system.
* User’s internet connection must be stable during the process.
* System should show all friendship requests when user goes to activities page.

Post-conditions: After the test case ends these conditions must be true:

* + Friendship connection will be accepted with the desired user successfully by the system.
  + User will see user as friend in user’s friend list.

Data required: User Name, Requested User

**TG-UCT-010 – Users Can Review Their Tasks:**

Description: The use case describes how the users review an existing task on calendar application.

Related Use Case: Review Existing Task

Pre-conditions: The conditions below must be provided before start to test:

* System must be operational and users can login to the system.
* User’s internet connection must be stable during the process.
* Users must have existing tasks in their schedule.

Post-conditions: After the test case ends these conditions must be true:

* + User will see up-to-date task details including the defined parameters that were filled while creating or updating the task.

Data required: Task

**TG-UCT-011 – All Users Can Remove Friend Properly by The System:**

Description: The use case describes how the user removes other users from friendship from other user’s profile page.

Related Use Case: Remove Friend

Pre-conditions: The conditions below must be provided before start to test:

* System must be operational and users can login to the system.
* User’s internet connection must be stable during the process.
* System should show all results when user browses a user.

Post-conditions: After the test case ends these conditions must be true:

* + Friendship connection will be removed with the selected user successfully by the system.
  + User will be able to see Add Friend label again after removed him/her on the user’s profile page.

Data required: User Name

**TG-UCT-012 – All Users Can Remove Friend Properly by The System (from Friend List):**

Description: The use case describes how the user removes other users from friendship from friend list page.

Related Use Case: Remove Friend

Pre-conditions: The conditions below must be provided before start to test:

* System must be operational and users can login to the system.
* User’s internet connection must be stable during the process.
* System should show all friends when user goes to friend list page.

Post-conditions: After the test case ends these conditions must be true:

* + Friendship connection will be removed with the desired user successfully by the system.
  + Removed friend will be disappeared from user’s friend list.

Data required: User Name

**TG-UCT-013 – All Users Can Create Community:**

Description: The use case describes how the users create community on calendar application.

Related Use Case: Create Community

Pre-conditions: The conditions below must be provided before start to test:

* System must be operational and users can login to the system.
* User’s internet connection must be stable during the process.

Post-conditions: After the test case ends these conditions must be true:

* + User will see the community in his/her communities, app shows all the community details correct.

Data required: Community Name, Description

**TG-UCT-014 – All Users Can Update Community:**

Description: The use case describes how the users update community information on calendar application.

Related Use Case: Update Community

Pre-conditions: The conditions below must be provided before start to test:

* System must be operational and users can login to the system.
* User’s internet connection must be stable during the process.
* User must be in the community.

Post-conditions: After the test case ends these conditions must be true:

* + User will see the community in his/her communities, app shows all the community details correct.

Data required: Community Name, Description

Revision Table

|  |  |  |
| --- | --- | --- |
| **Revision** | **Description** | **Date** |
| 1.0 | First revision | 25/11/2023 |
| 1.1 | New Test Cases are added for Add Friend, Handle Friendship Request and Review Existing Task use cases. (From TG-UCT-007 to TG-UCT-010) | 16/12/2023 |
| 1.2 | Related Use Case sections are added to the test cases. | 23/12/2023 |
| 1.3 | Related Use Case sections are added to the test cases. | 01/01/2024 |