Sea To-do

Organizer app

Content

1. Introduction

Focus on important things while staying calm

The app has been designed as an assistant for long-term and calm movement towards your goals, or the acquisition of new habits.

Tasks

You can create long-term tasks or habits and track progress each day for complete tasks. The end date is dynamically calculated for each task.

Sea Calendar

Each day can be judged by the state of calmness. While using the calendar, you will have the opportunity to clearly see how calmly and efficiently everything turns out.

Prospects for the development and expansion of the application

1. There is currently no synchronization in the application. For example, when adding synchronization with a Google account it will be possible to use the application simultaneously on different devices.
2. At the moment, the application implements the creation of long-term tasks. Similarly with such tasks, you can add the ability to create one-time tasks. For example, scheduling a to-do list for the day.
3. An additional aspect of the development of the application can be a Sea Calendar. At the moment, the Sea Calendar can be supplemented with various functions. Additional features may include notifications, etc.

The project was created on the Unity engine. (Unity 2019.2.14f1)

The features of the project are:

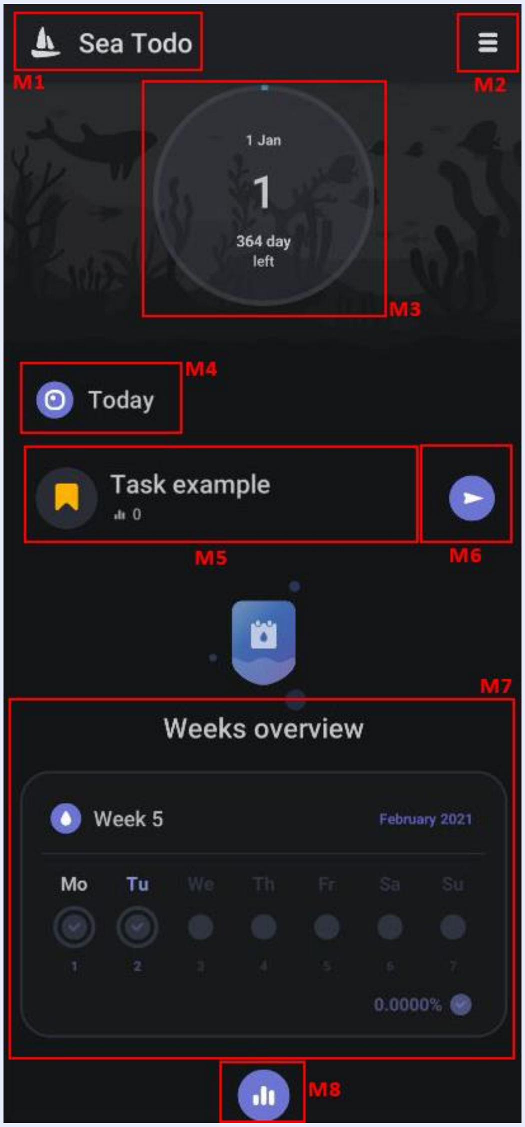
1. Using the UI.
2. Using vector graphics (images in SVG format).
3. Writing classes with minimal use of Unity MonoBehaviour classes.
4. Relatively small size of the application.
5. Storage of all visual objects at the same time on one scene.
6. The project can be packaged for various platforms such as iOS, Android, etc.

2. Description of application pages

Generalized description of the functional

Home page (M)

The main page contains the current tasks that the user is performing. In addition: year overview, week overview (weeks can be flipped horizontally), statistics button and calendar button (to view the status of tasks for the required day).

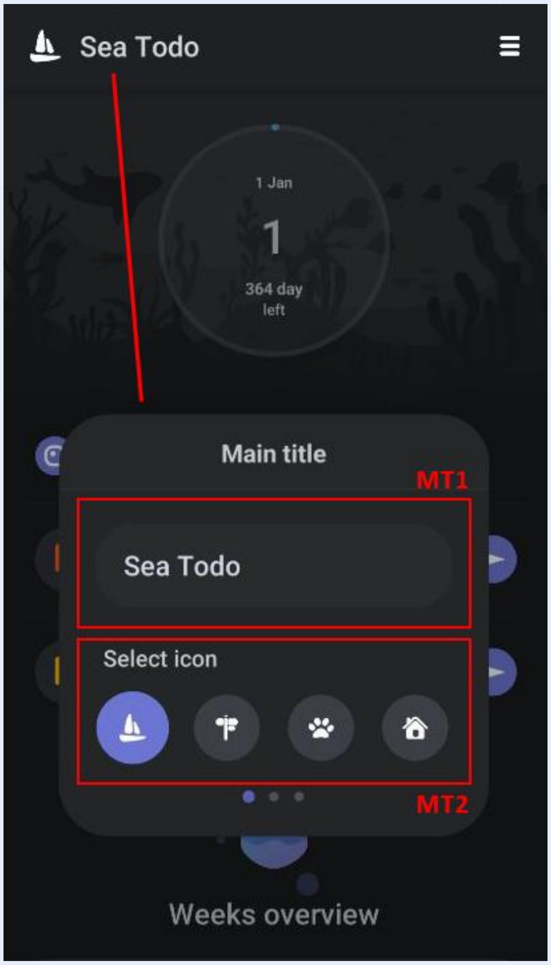


- **M1** – Title name and page icon. When you click on this area, you can edit text and icons;
- **M2** – Menu opening button;
- **M3** – A small overview of the current year. Clicking on this area opens the activity page;
- **M4** – Display of the day for which tasks that are at the bottom are shown;

- **M5** – Task display;
- **M6** – Clicking on this button opens a panel for recording the progress for the current day;
- **M7** – Review of weeks. For each week, the percentage of tasks completed daily is shown. The percentage of total completion is also displayed below. The weeks are flipped horizontally;
- **M8** – Activity page open button.

Editing the title of the application (MT)

To close the window, click on the shaded free area.



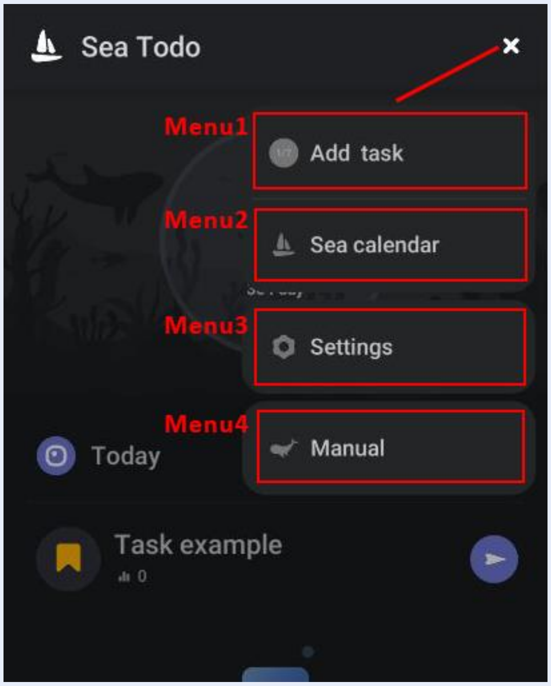
- **MT1** – Input field for the title name of the main page;
- **MT2** – Choosing an icon for the main page. Icons can be flipped horizontally.

Menu (Menu)

Opening is carried out by pressing button **M2**.

You can close the menu:

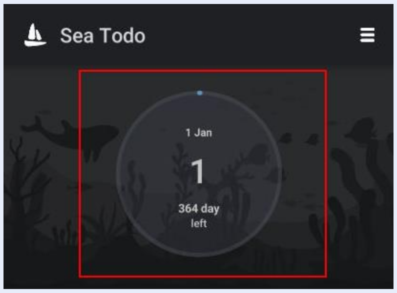
- by clicking on a free area;
- by moving the menu itself to the right outside the screen area;
- by press the close button.



- **Menu1** – Open the page for creating a new task. If the maximum number of tasks has already been created, then when you click a panel it`s will appear with information about why the number of tasks is limited;
- **Menu2** – Go to the Sea calendar;
- **Menu3** – Go to the Settings page;
- **Menu4** – Opening the manual. The button will only be active a certain number of times after installing the application. Then it will no longer appear, because the manual will not be relevant to the user.

Current year overview

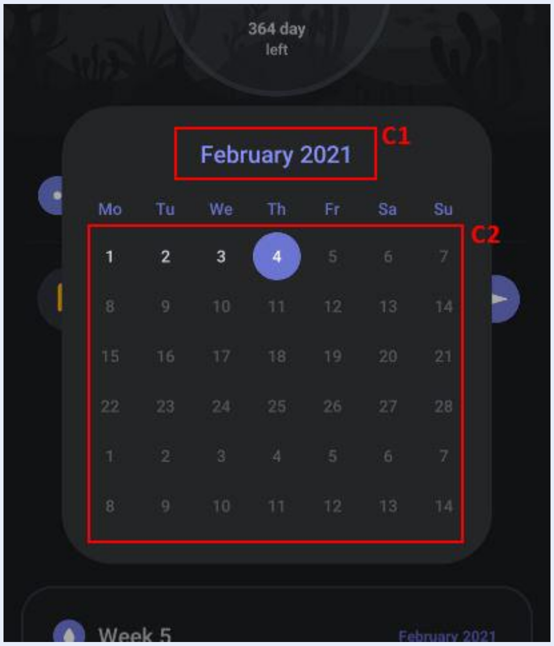
When clicked, the activity page opens.



An additional line along the border of the circle indicates what percentage of the current year has passed.

Calendar window (C)

To turn over the months, you need to move them horizontally. To close the calendar, click on the free shaded area.



364 day left

February 2021

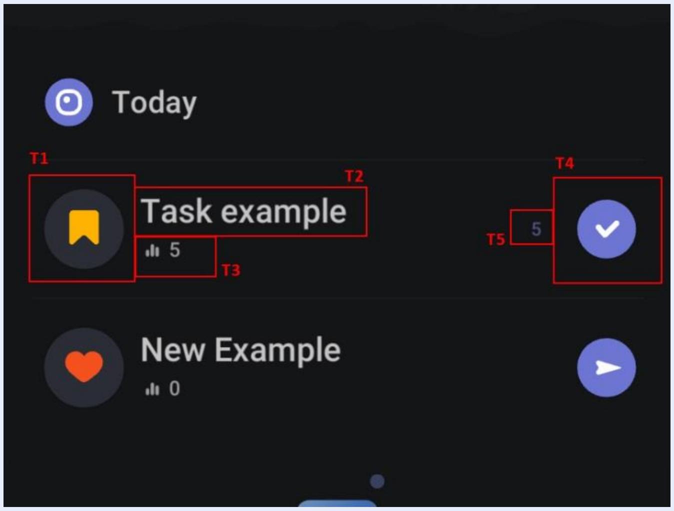
Mo	Tu	We	Th	Fr	Sa	Su
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
1	2	3	4	5	6	7
8	9	10	11	12	13	14

Week 5 February 2021

- **C1** – Name of the month and year;
- **C2** – Days of the month. You can select only the current day or those that have already passed.

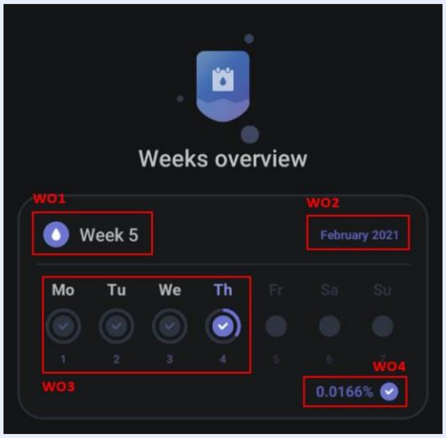
Task list (T)

In order to open the task page, you need to click on its main area. In order to mark the progress of the task, you need to click on the button on the right side.
If you need to change the priority of tasks, then for this you need to hold the task for a while, and after unpinning it, move it to the desired position vertically.



- **T1** – Task icon;
- **T2** – Task name;
- **T3** – General progress of the task;
- **T4** – Button to record the current progress;
- **T5** – Reported progress for the current day.

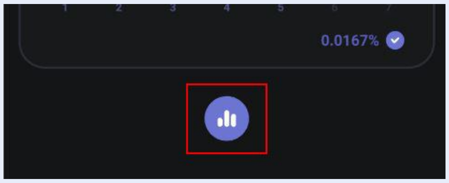
Week overview (WO)



- **WO1** – Week number;
- **WO2** – Month and year of the week;
- **WO3** – Days of the week showing the percentage of completion of the number of tasks;
- **WO4** – The overall percentage of completion for all tasks.

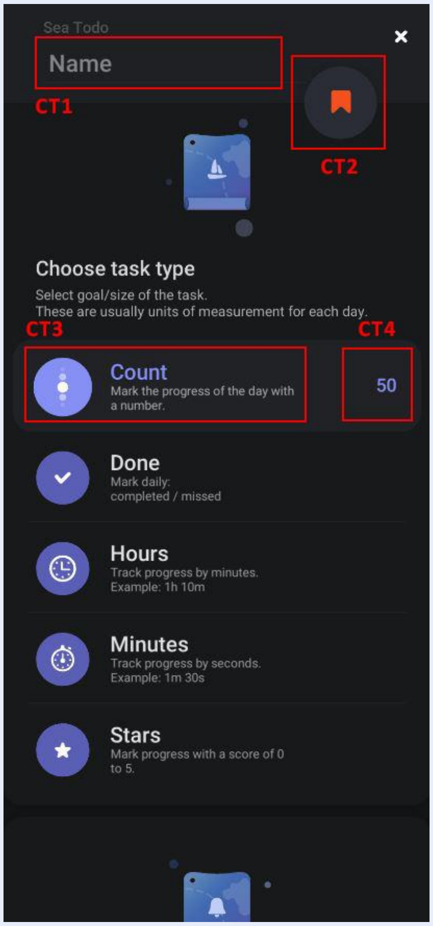
Statistics button

The button is responsible for the transition to the activity page (the same as the overview of the current year).



The page for adding a new task (CT)

On this page, you can create a new task: firstly, you can choose icon and write name of task. After that, you can select the type of task (what progress will be measured in). When you click on the area on the right of selected type, a track window for selecting the amount of the goal will open (according to the type of task). After that, you can select the days of the week and time to receive local notifications. To activate a day of the week, click on that day. To deactivate, you should press the day again. When the day is active, it becomes possible to set the time for receiving the notification.



Choose task type

Select goal/size of the task.
These are usually units of measurement for each day.

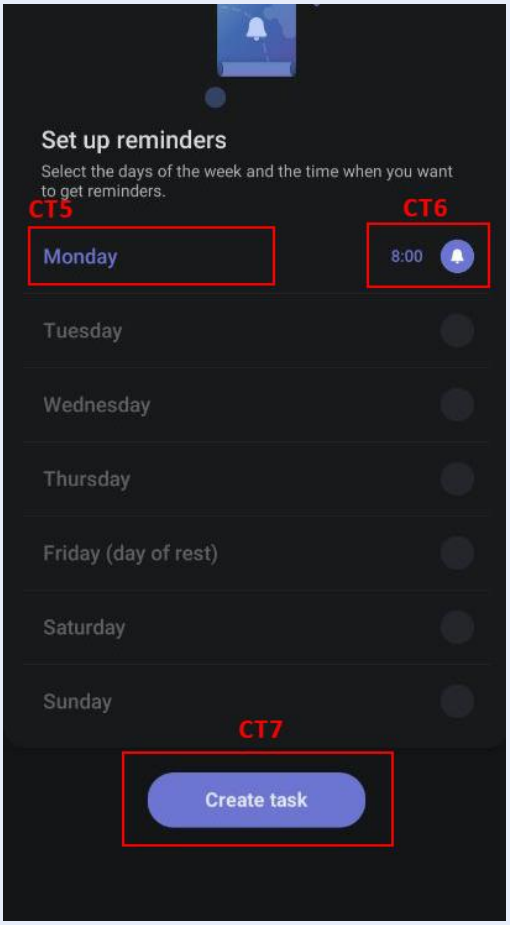
Count – Mark the progress of the day with a number.

Done – Mark daily: completed / missed

Hours – Track progress by minutes. Example: 1h 10m

Minutes – Track progress by seconds. Example: 1m 30s

Stars – Mark progress with a score of 0 to 5.



Set up reminders

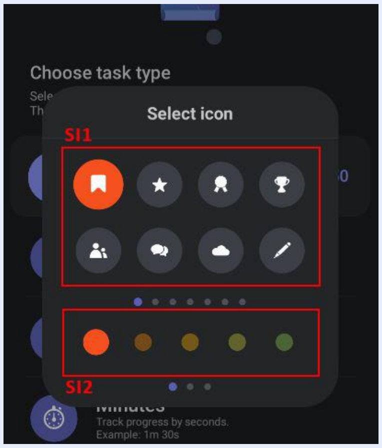
Select the days of the week and the time when you want to get reminders.

- **CT1** – Task name;

- **CT2** – Task icon;
- **CT3** – Task type and description;
- **CT4** – Target size;
- **CT5** – Name of the day of the week to receive reminders;
- **CT6** – Time of receipt of notifications;
- **CT7** – Task creation button.

Task icon selection window (SI)

In this window you can customize the task icon. Icons and icon colors are flipped horizontally. To close the window, click on an empty area.

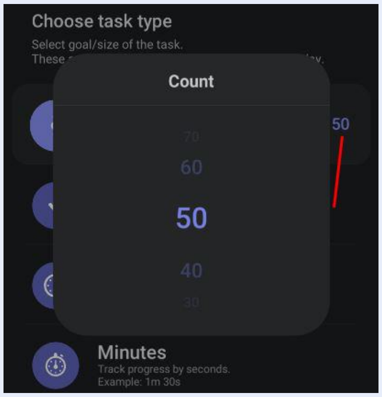


Track progress by seconds.
Example: 1m 30s

- **SI1** – List of icons;
- **SI2** – Palette for icons.

Task goal selection window

To select the goal of a task, it is necessary to move these values vertically. The number that is in the center will be the current goal of task. To close the window, click on the shaded free area.

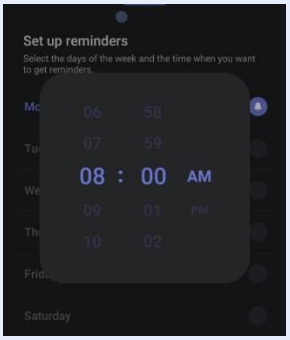


Select goal/size of the task.

Track progress by seconds.
Example: 1m 30s

Reminder time selection window (ST)

In this window, you can select the time to get notifications for a specific day of the week. To adjust the time, you need to move the values vertically. To close the window, click on the shaded free area.



Set up reminders

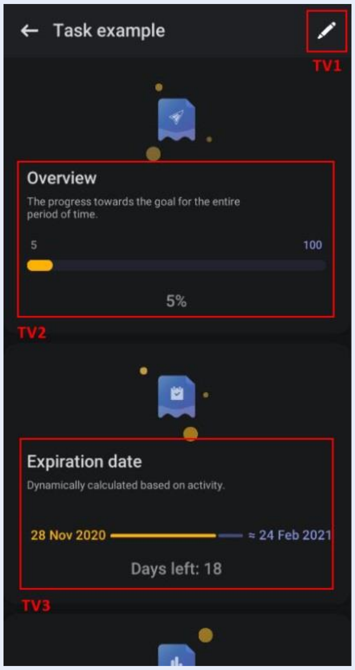
Select the days of the week and the time when you want to get reminders.

Task view page (TV)

This page opens when you click on a task (on the home page). Here you can view the progress indicators of the task. In the upper right corner there is a button to go to the task editing page.

The page includes:

- Percentage of task completion;
- The estimated finish date of a task based on the user's activity;
- Generalized statistics of daily progress of a month. The flipping of the months is carried out using horizontal movements;
- Overview of active reminders by day of the week.



Task example

Overview

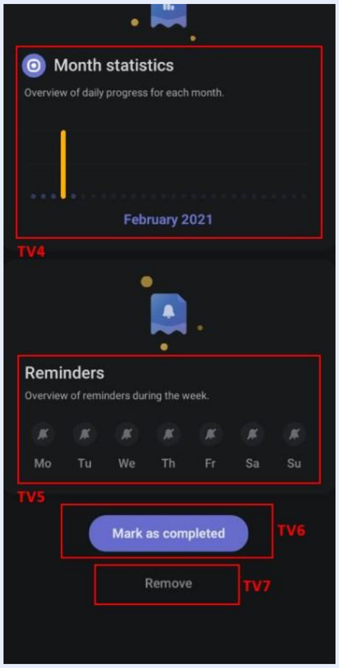
The progress towards the goal for the entire period of time.

Expiration date

Dynamically calculated based on activity.

28 Nov 2020 ≈ 24 Feb 2021

Days left: 18



Month statistics

Overview of daily progress for each month.

February 2021

Reminders

Overview of reminders during the week.

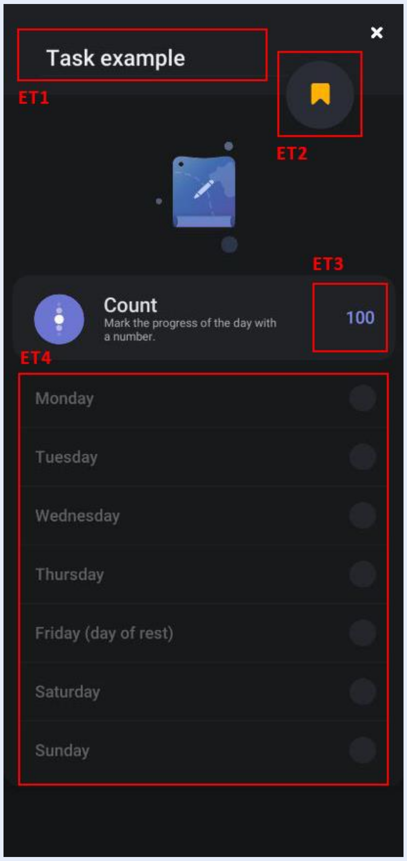
- **TV1** – Button to go to the task editing page;

- **TV2** – The percentage of completion of this task;
- **TV3** – Estimated finish task date;
- **TV4** – Statistics of this task;
- **TV5** – Overview of notification activity by day of the week;
- **TV6** – Button for transfer the task to the completed state;
- **TV7** – Delete task button.

When you click on the “Mark as completed” button, you go to a special page for reviewing the completed task. When you click on the “Remove” task button, a confirmation window appears.

Task edit page (ET)

On the task editing page, it is possible to update the goal value, update the name, update the icon, and update task reminders. However, you cannot change the type of the task (because when you change the type of the task, the past progress cannot be converted to the new type). Changes are automatically applied when you exit the edit page.



Task example

Count

Mark the progress of the day with a number.

100

Monday

Tuesday

Wednesday

Thursday

Friday (day of rest)

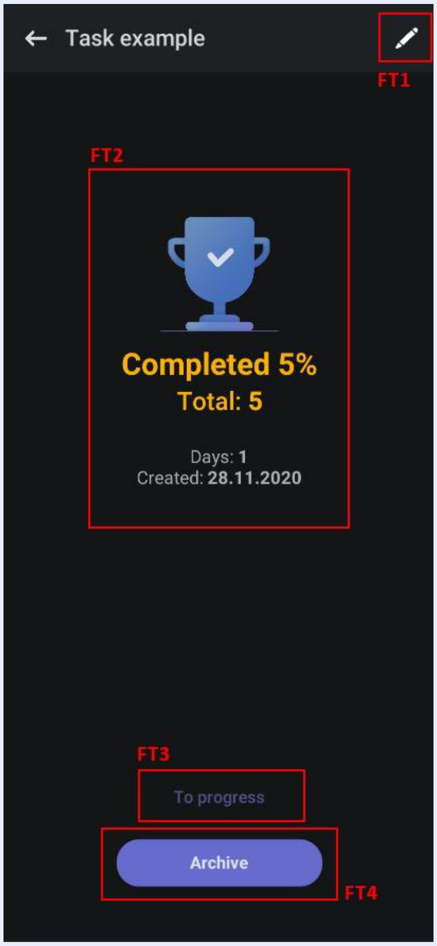
Saturday

Sunday

- **ET1** – The editable field with the task name;
- **ET2** – Task icon (when pressed the icon editing window appears);
- **ET3** – Displaying the value of the goal of the task;
- **ET4** – Editing task reminders.

Finished task page (FT)

On this page you can view the results of the task. The “To progress” button moves the task back to the active state. If the task is 100 percent or more complete, then this button will not be displayed. When you click on the “Archive” button, the task will no longer be displayed in active tasks, but will be available for viewing in an activity page.



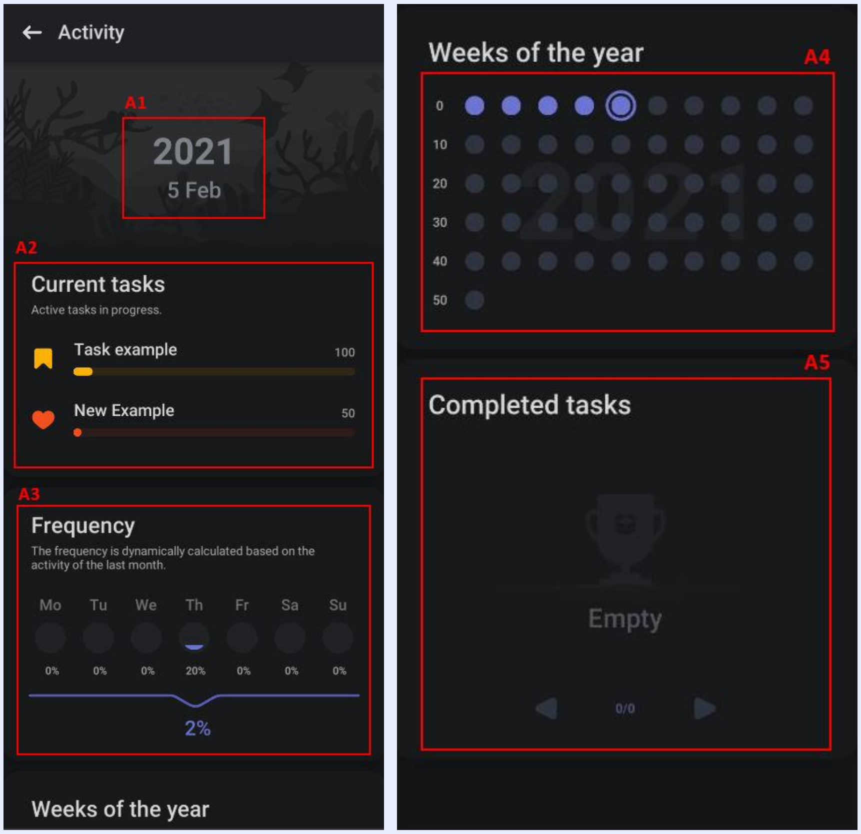
- **FT1** – Button to go to the task editing page;
- **FT2** – Displaying results;
- **FT3** – Button to return to the activity state;
- **FT4** – Archive task button.

Activity page (A)

On this page you can view general activity. Main information:

- The top area displays a list of active tasks with an view of the percentage of their completion;
- Frequency of task execution by day of the week. Shows how actively the user is recording progress. This characteristic allows for more analysis of activity.

- The overview of weeks shows the weeks that have passed. This part can be as additional motivation.
- At the bottom there is a list of archived tasks, which can be flipped using horizontal movements. When you click on a task, a window will appear with the ability to return this task to an active state or remove.



Activity

2021
5 Feb

Current tasks

Active tasks in progress.

Task example 100

New Example 50

Frequency

The frequency is dynamically calculated based on the activity of the last month.

Mo	Tu	We	Th	Fr	Sa	Su
0%	0%	0%	20%	0%	0%	0%

2%

Weeks of the year

Completed tasks

Empty

0/0

- **A1** – The current date;
- **A2** – Displaying a list of active tasks;
- **A3** – Displaying the frequency of tasks progress by each day of the week;
- **A4** – Overview of the weeks of the year;
- **A5** – Review of archived tasks.

Sea Calendar page (SC)

What is Sea Calendar?

Sometimes it is very important to stay calm in the midst of a lot of unnecessary information and noisy environments. Such things can distract from the main things and interfere with calm decision-making. Sea Calendar is an addition to the main purpose of the application: to help you achieve long-term goals or acquire new habits.

How to use?

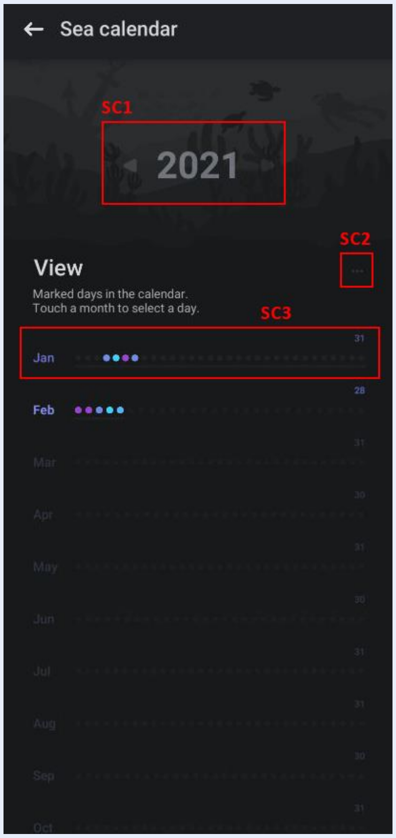
Every day, you can evaluate the quality of the day. There are 5 categories for rating: Full calm, Calm, Waves, Noise, and Storm. Each day can be designated one of these categories. In the future, you will have the opportunity to review how calmly and efficiently everything turns out. Based on this you can better understand if something needs to be changed.

Quality of days:

- Full calm. Days dedicated to complete rest and recovery.
- Calm. Calm and fulfilling days.
- Waves. Days with awkward moments.
- Noise. Days in which disturbing moments or events may have occurred.
- Storm. Days in which very difficult moments or events could occur that could affect mental health.

Description of the page

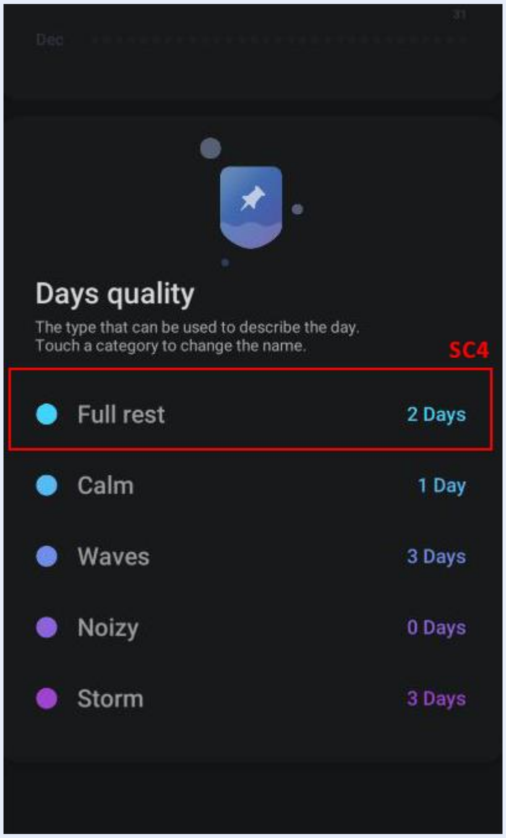
The main part of the page shows a list of the months of the year and the highlighted days. When you click on any month, a window appears with a list of days. The second part shows the characteristics of the days with which you can mark the days. When you click on any characteristic, a window appears with the ability to rename it.



Sea calendar

View

Marked days in the calendar.
Touch a month to select a day.



Days quality

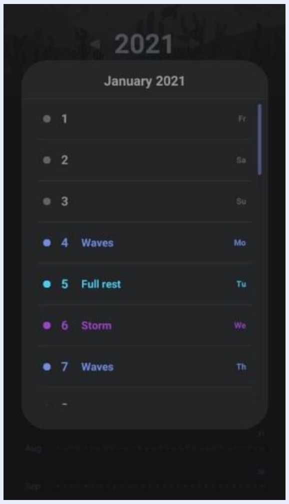
The type that can be used to describe the day.
Touch a category to change the name.

Full rest	2 Days
Calm	1 Day
Waves	3 Days
Noizy	0 Days
Storm	3 Days

- **SC1** – Display of months for the shown year;
- **SC2** – Guide button;
- **SC3** – Review of the month;
- **SC4** – Quality characteristic of the day.

Day overview window in Sea Calendar (MSC)

When you select a month on the main Sea Calendar page, this window appears. In this window, you can select a specific day of the month. To close the window, click on the shaded free area.



2021

January 2021

1 Fr

2 Sa

3 Su

4 Waves Mo

5 Full rest Tu

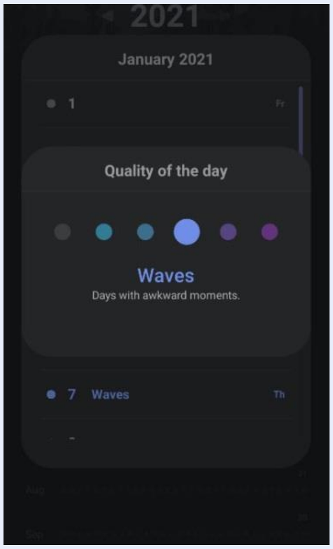
6 Storm We

7 Waves Th

-

Day characteristic assignment window (SDQ)

In this window, you select the characteristic of the quality of the day for the selected day.



2021

January 2021

1 Fr

Quality of the day

Waves

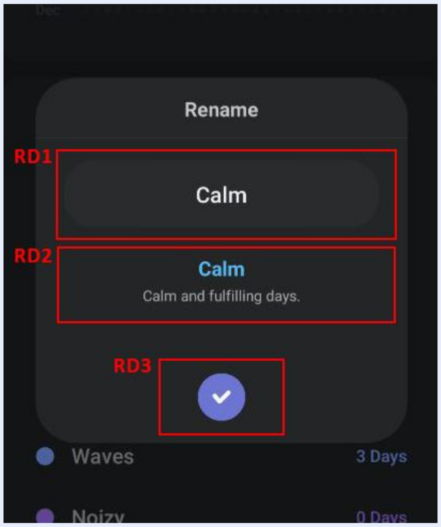
Days with awkward moments.

7 Waves Th

-

Properties renaming window (RD)

When you click on the name of the characteristic under the input field, the default name will be placed in the input field itself. To close the window, click on the shaded free area. To save, press the lower button.



Rename

Calm

Calm

Calm and fulfilling days.

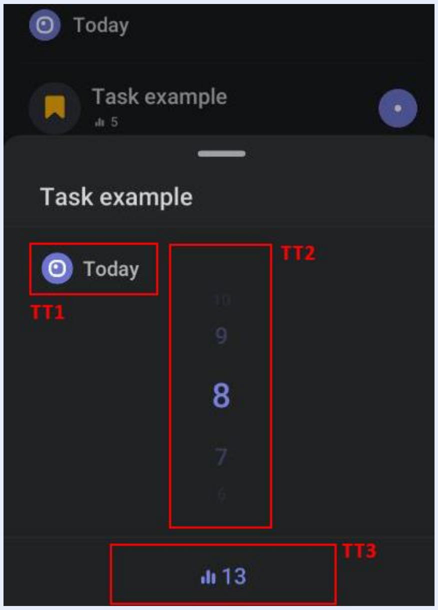
Waves 3 Days

Noizy 0 Days

- **RD1** – Input field of name;
- **RD2** – Description of the day;
- **RD3** – Save button.

Task progress recording window (TT)

When you click on the button of track progress, this window pops up, where by moving the values vertically, you select the necessary progress value. To close the window, you need to click on the darkened free area, or drag the window down from the top.



Today

Task example 5

Task example

Today

10 9 8 7 6

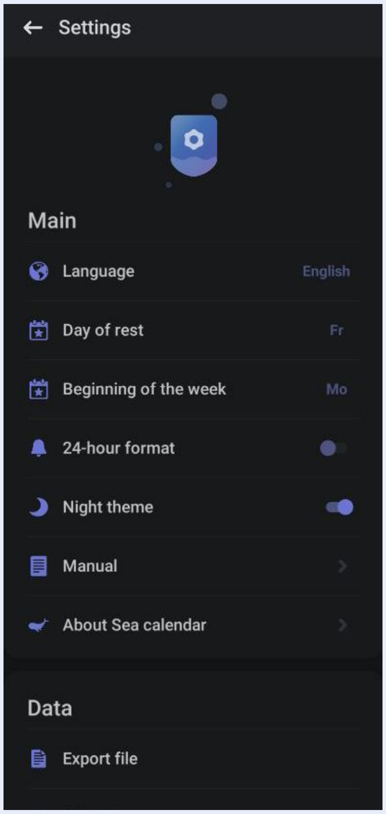
13

- **TT1** – Day for track progress;
- **TT2** – Selectable progress for the day;
- **TT3** – Overall task progress.

Settings page

This page contains 3 groups of application settings:

- The first group is responsible for basic settings;
- The second group is responsible for working with saving and application files. This group has a different set of features for the iOS and Android platforms;
- The third group is responsible for additional information about the application.



← Settings

Main

Language English

Day of rest Fr

Beginning of the week Mo

24-hour format

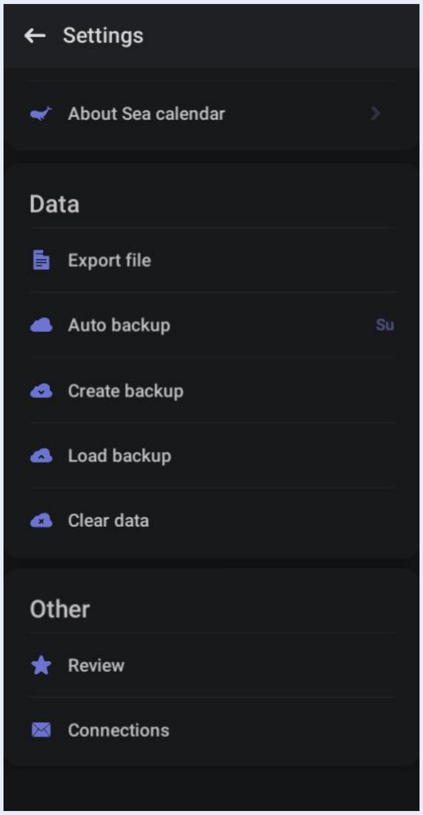
Night theme

Manual

About Sea calendar

Data

Export file



← Settings

About Sea calendar

Data

Export file

Auto backup Su

Create backup

Load backup

Clear data

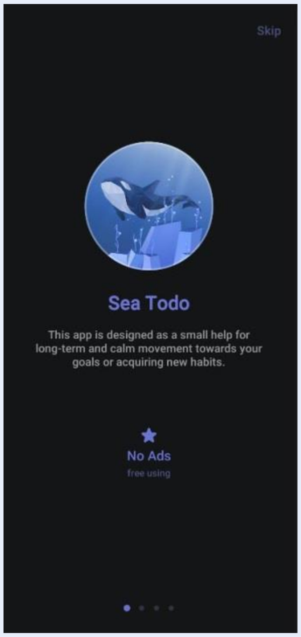
Other

Review

Connections

Manual

The manual consists of 4 pages. To scroll, you need to move them horizontally. In addition to such movement, pages move automatically after a certain period of time.

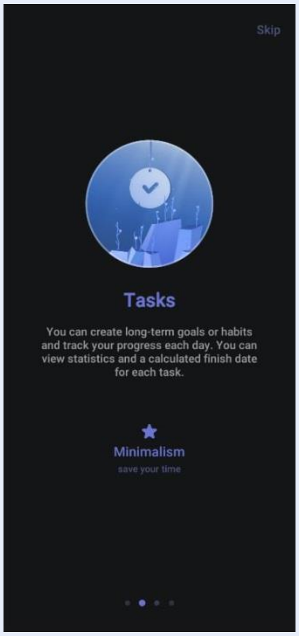


Sea Todo

This app is designed as a small help for long-term and calm movement towards your goals or acquiring new habits.

★ No Ads

free using

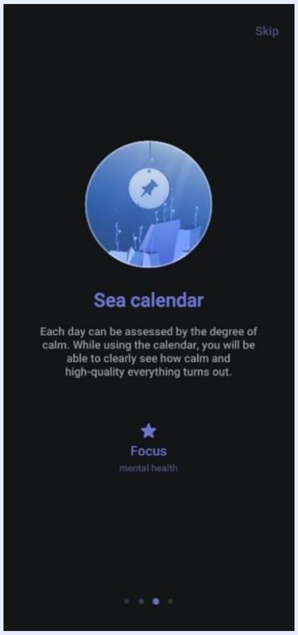


Tasks

You can create long-term goals or habits and track your progress each day. You can view statistics and a calculated finish date for each task.

★ Minimalism

save your time

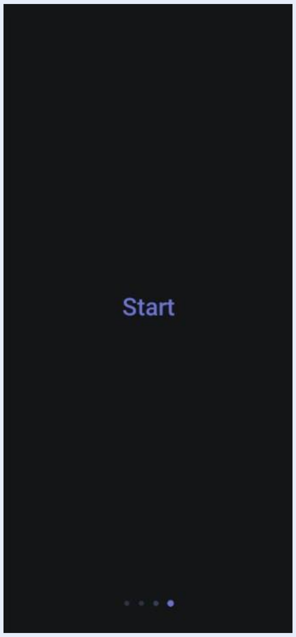


Sea calendar

Each day can be assessed by the degree of calm. While using the calendar, you will be able to clearly see how calm and high-quality everything turns out.

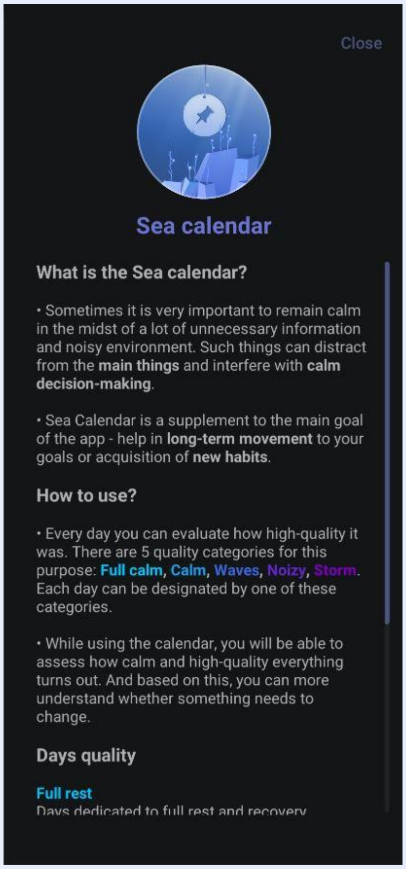
★ Focus

mental health



Description of Sea Calendar

This page contains information on how to use the Sea Calendar.



Sea calendar

What is the Sea calendar?

• Sometimes it is very important to remain calm in the midst of a lot of unnecessary information and noisy environment. Such things can distract from the **main things** and interfere with **calm decision-making**.

• Sea Calendar is a supplement to the main goal of the app - help in **long-term movement** to your goals or acquisition of **new habits**.

How to use?

• Every day you can evaluate how high-quality it was. There are 5 quality categories for this purpose: **Full calm**, **Calm**, **Waves**, **Noizy**, **Storm**. Each day can be designated by one of these categories.

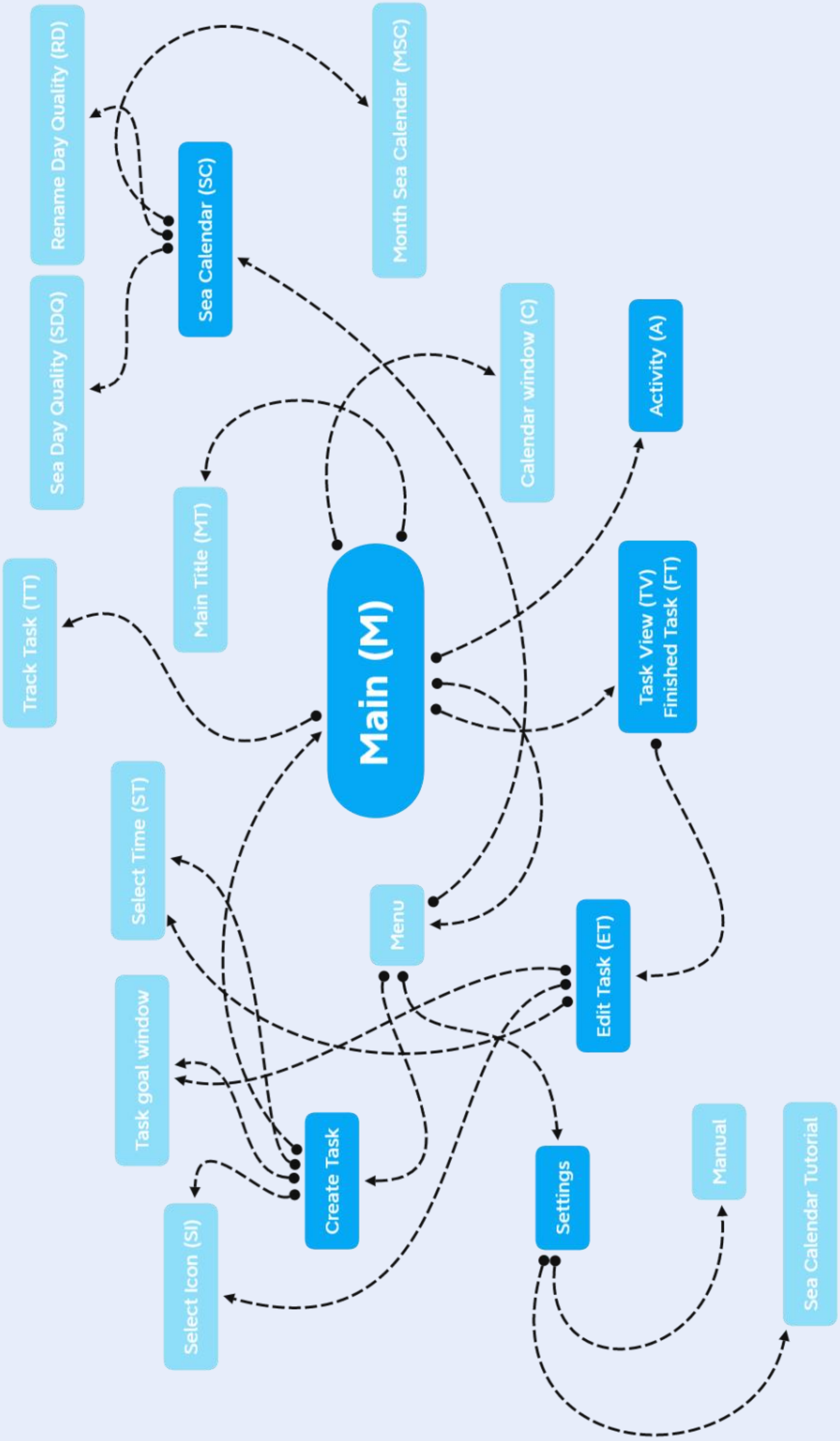
• While using the calendar, you will be able to assess how calm and high-quality everything turns out. And based on this, you can more understand whether something needs to change.

Days quality

Full rest

Days dedicated to full rest and recovery

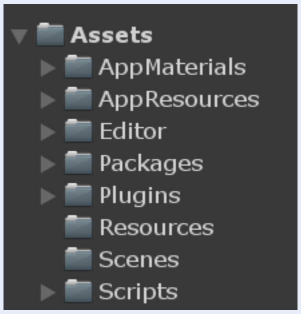
3. Application page structure



4. Description of the project structure in Unity

The description of the project structure consists of 2 parts: the folder structure in Assets and the content of elements on the scene.

Assets



- **AppMaterials** – contains application icons;

  AppMaterials/Other – contains icons for notifications;

- **AppResources** – contains all visual files of the application (Animations, UI, etc.);

  AppResources/Animations – contains UI animations. Animations are divided into folders depending on which page they belong to;

  AppResources/Content – contains additional pictures for the application (backgrounds, icons, etc.);

  AppResources/Materials – contains materials for text, sprites, etc.

  AppResources/Resources – contains icons for instantiate sprites when the application is running;

  AppResources/Resources/ActiveResources/CirclePercentage – contains SVG images (which represent a circle);

  AppResources/Resources/FlowIcons – contains icons for tasks (divided thematically into folders);

  AppResources/Resources/GroupIcons – contains icons for the title bar of the main page;

  AppResources/UI – contains visual elements for displaying the interface;

  AppResources/UI/Other – contains some elements in bitmap format;

  AppResources/UI/SVG – contains elements in vector format. All elements are divided into folders according to application pages;

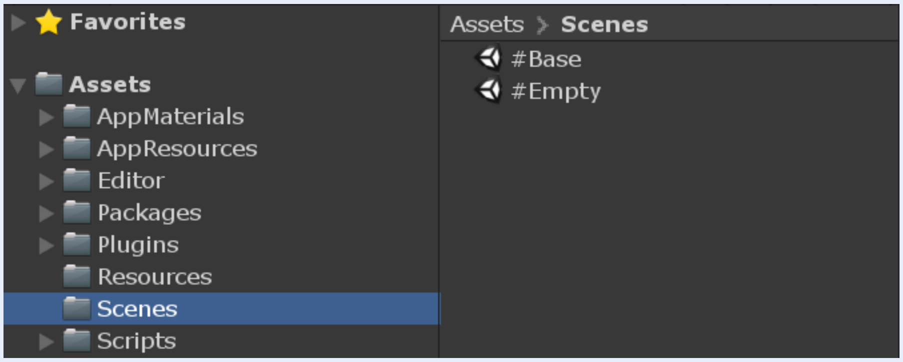
- **Editor/com.unity.mobile.notifications** – contains a notification settings file. Notifications are used from Unity, so you can read about Unity notifications in the official documentation.
- **Packages** – contains external packages that are used in the application;

  Packages/Fonts – contains the Roboto font, as well as its updated version RobotoMenium_Updated, where additional characters have been added;

Packages/HomeTools – contains the set of scripts, which are the main tools for organizing scripting systems;

- **Plugins** – contains additional manifests;
- **Resources** – contains Scriptable Object with translation of all interface strings;
- **Scenes** – contains scenes (#Empty - loading scene and #Base - main scene);
- **Scripts** – contains the main scripts of the application architecture.

Scenes



Favorites
Assets
AppMaterials
AppResources
Editor
Packages
Plugins
Resources
Scenes
Scripts

Assets > Scenes
#Base
#Empty

Content of elements in the scene #Empty

Because the #Empty scene is loaded at the beginning and plays the role of a splash screen, it contains a background and several elements in the center of the screen. The animation of the appearance and disappearance of logo elements occurs using the Animator.

There are also 2 scripts on the Canvas object:

- **LogoLoad** – to load the main scene in parallel way;
- **Localization Initialize** & Setup Localization Object – scripts for initialize language setting and additional setup scriptable object instance;

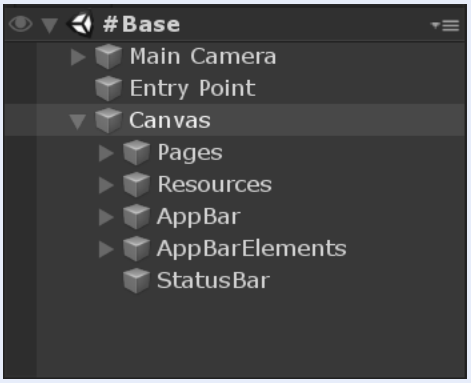
Content of elements in the scene #Base

The filling of the scene has its own characteristics. It simultaneously stores the entire interface of the pages. All UI elements are stored in the Canvas/Resources/...

Scripts search for UI elements by their name on the scene, so renaming objects without similar renaming in scripts will result in errors.

There is one main camera in the scene with orthographic mode. All other elements are Unity UI. In addition, there is an Entry Point object on the scene, which stores 2 scripts: "EntryPointSeaTodo" and "Purchaser". When the scene is launched in the first script, the main systems for the application are created accordingly. The second script implements in-app purchases.

The main object that contains the UI is Canvas.



- #Base
  - Main Camera
  - Entry Point
  - Canvas
    - Pages
    - Resources
    - AppBar
    - AppBarElements
    - StatusBar

The “Pages” object in the running application stores the designed pages of the application. When moving from one page to another (Page 1 > Page 2, when open another app page), the UI is constructed in Page 2 and the animation of the transition between pages is played. When the transition is made to the main page (Page 1), the elements from Page 2, after the page transition animation, are moved to the additional Page Pool (disabled and needed to store elements that are not currently needed). Pages are filled with UI elements from the Resources object when the scene starts, as well as for other needs.

- **Page 1** – stores the main page UI in the running application;
- **Page 2** – stores the UI of secondary pages in the running application. Dynamically changes its content;
- **Page Pool** – is created when the scene starts and stores previously created elements for reuse. This object is disabled during application execution.

When the user moves the page up/down, the movement is due to the movement of the Page 1 and Page 2 objects.

The Resources object stores prefabs for the UI, which will change and display in Pages when the application is running. In addition, elements of additional windows are stored there (time picker, calendar, progress bar, etc.). These windows in the running application are located through scripts and then stored in the Canvas object itself.

Items in the Resources object:

- **GroupBackground** – template for background UI for groups for pages. This item is copied from resources. Additional UI elements are constructed into it. After that, the generated from the UI GroupBackground is stored in the Page Pool and, if necessary, is reused in the Page 2 when it is necessary to display it. Overall, it is a small, constructed block of UI elements.
- **Year Stats** – it is the circular object that places on the top of the main page.
- **WorkArea Main** – this is the part of the home page that contains the calendar icon and the current day's label. There is also an icon that indicates to the user that there are no tasks to perform (Empty Flow).
- **WorkArea Flow** – this is the dividing line that is used in various pages to delimit in the UI (for example, a dividing line between tasks on the main page).
- **WorkArea Second** – this is the object where the part of the main page (Weeks Overview and icon) is stored.

- **WorkArea Graphic –** this is the template for constructing a graph in the Weeks Overview.
- **WorkArea Background** – this is the background (picture) for the main page, which will be located under the Year Stats elements part.
- **TrackArea** – this is a pop-up area that is needed to track the progress of a task. Appears when you click on a track task button. Elements within this panel are automatically adjusted by the scripting system.
- **MenuArea** – this menu bar that appears when you click on the menu button (located at the top right of the main page).
- **CreateFlow Content 1** & **CreateFlow Content 2** – these are the icons that are displayed on the task creation page.
- **CreateFlow Name Line** – this is the line that places on the top of app bar under the line for input the task name (on the page for creating or editing a task).
- **CreateFlow Name Input** – this is the text input used to enter the name of the task. When the application is running, it also places on the app bar (in the page for creating or editing a task).
- **CreateFlow Icon** – this is the task icon that is displayed on the app bar and displays the currently selected icon (in the task creation or editing page).
- **CreateFlow Title** – this is the text object, instances of which are displayed as title descriptions in task creation blocks.
- **CreateFlow Description** – this is the text object that is copied and pasted into the block descriptions on the task creation page.
- **Track Module –** this is the object with vertical scrolling of numbers. It is used in the pop-up window for recording progress (on the main page), and in the panel for selecting the goal value (on the page for creating and editing a task).
- **Reminders Module** – this is the empty object for configuring notifications (filled with elements at startup). The final result is a module for creating notifications (used in the pages for creating and editing a task).
- **Reminders Line** – this is the line element that is used when constructing the Reminders Module.
- **Reminders Day View** – This is the display element for the day of the week string for the Reminders Module.
- **TrackTime Module** – this is the panel for choosing the time for reminders. When the scene starts, a full-fledged window is created, where you can determine the required time of day.
- **ChooseFlow Type** – this is the element that displays the task type on the task creation page.
- **EmptySignal** – this is the icon representing a handle that appears when the user tries to create a task with an empty title.
- **CreateFlow Create Button** – this is the object of the button that is used to create the final task. Located at the bottom of the task creation page.
- **ViewFlow Progress Part** – this is the part of the assembled UI that displays the percentage of the task completed on the task view page.
- **ViewFlow Statistics Part** – this is the part of the assembled UI that represents the display of statistics on the task view page.
- **ViewFlow Estimate Part** – this is the part of the assembled UI that represents the task end date display on the task view page.
- **ViewFlow Reminders Part** – this is a part of the assembled UI that displays notification activity on the task view page.
- **ViewFlow Content Finished** – this is the reward image on the completed task overview page.
- **ViewFlow Finished Title** – this is the object with text that appears on the completed task overview page.

- **ViewFlow Finished Description** – this is the object with task description text that is displayed on the completed task page.
- **ViewFlow Finished Info** – this is the object with additional information text that is displayed on the completed task page.
- **ViewFlow Mark Completed Button** – this is the task completion button that appears at the bottom of the task overview page.
- **ViewFlow Remove Button** – this is the remove task button that appears at the bottom of the task overview page.
- **ViewFlow To Progress Button** – this is the button to return the task to the execution stage. The button is on the completed task page.
- **ViewFlow Archive Button** – this is the button for archiving the task, which is located on the completed task page.
- **EditFlow Content** – these are the images for the task edit page.
- **TrackGoal Module** – this is a container for **Track Module**. This container displays the task target selection panel.
- **EditFlow Goal** – this is the pre-built part of UI that displays the goal on the task edit page.
- **Statistics InProgress Part** – this is the part of UI for displaying run-time tasks on the activity page.
- **Statistics Frequency Part** – this is the part of UI for displaying the frequency of user activity.
- **Statistics Weeks Part** – this is the part of UI for displaying the weeks of the year on the activity page.
- **Statistics Archived Flows Part** – this is the part of UI for displaying archived tasks on an activity page.
- **TrackCalendar Module** – this is the calendar window for selecting a day. This window is automatically constructed using scripting systems.
- **ChooseIcon Module** – this is the choose task icon window. Used on the pages for creating and editing a task.
- **UpdateTitle Module** – this is the update title window for the current week on the home page.
- **Accept Module** – this is the window that is used for the final confirmation of any user action.
- **Tip Module** – this is the window that is used in places where needs to explain or tell something to a user. Can be used many times with different content.
- **Tasks Limit Tip** – this is the text object for the **Tip Module**. Shown in the prompt window when the user tries to create a task while the number of created tasks has reached a limit.
- **Archived Update Module** – this is the window for selecting actions for an archived task. Appears when you click on an archived task in the activity page.
- **DaysMarker Days Part** – this is the part of UI that is generated in the running application. This part is used to display months on the Sea Calendar page.
- **DaysMarker Color Part** – this is the part of UI that displays a list of the qualities of the days on the Sea Calendar page.
- **UpdateMarker Module** – this is the window used to edit the names of the qualities of the day. Appears when you click on a day characteristic on the Sea Calendar page.
- **Similar Marker Tip –** this is the text object for the Tip Module. Shown in the window when the user tries to rename a day characteristic that already exists.
- **MonthMarkers Module** – this is the window for viewing the days of the month that appears when you click on any month in the main part of the Sea Calendar page.
- **ChooseDay Module** – this is the day characteristic selection window that appears when you select a day in the MonthMarkers Module.

- **Tutorial** – this is the user manual. You can call it viewing from the settings page (always) or from the menu (if available).
- **Settings Main Part** – this is the main part of UI for displaying application settings. Displayed on the settings page.
- **Settings Data Part** – this is the part of UI for displaying application settings that are responsible for creating save files. This part is only active on the Android platform.
- **Settings Data Part Short** - this is the part of UI for displaying application settings, which are responsible for the data on the settings page. This part is only active on the iOS platform.
- **Settings Other Part** - this is the part of UI for displaying additional settings and information about the application.
- **ChooseItems Panel** – this is the window for selecting a list of text elements, which is actively used on the settings page. Used to select language, day of the week, etc.
- **About Sea Calendar** – this is the window that displays information about using Sea Calendar. This is a small tutorial for the Sea Calendar page.
- **Create Backup Module** – this is the save file creation window used on the application settings page.
- **Create Backup Result Module** – this is the window that informs about the success of the save file creation.
- **Single Page View** – this is the module for displaying information in full window. Allows you to scroll vertically any content. It is actively used on the application settings page.
- **Connections View** – this is the UI object that displays information about connections. Used with Single Page View.

The **AppBar** element in Canvas is the background object for the app bar.
The **AppBarElements** in Canvas is the object with UI elements for the app bar.
The **StatusBar** element in the Canvas is the background of the status bar of the device itself.

5. Plugins and add-ons used in the project

1. Json Dot Net;
2. Unity Vector Graphics;
3. Unity Mobile Notification;

6. Project in other languages

The project has been translated into 8 languages:

- English;
- Русский;
- Français;
- Deutsche;
- Italiano;
- Português;
- Español;
- Türk.

The translation was done using the I2 Languages plugin.

7. Principles of writing scripting systems

1. The basic principle of building application architecture is that most classes do not inherit from MonoBehaviour, and therefore, in a running application, scripts are linked to MonoBehaviour scripts only to call the Update method.
2. Also, the basic principle of work of classes with UI is that scripts search for the necessary objects in the scene by name. Therefore, when renaming objects in the scene, you must rename them in classes.
3. Basically, for each page, its own group of classes is created, which is initialized when the main scene starts.

Local application modules:

1. The application has the ability to change the color of the interface. It is carried out using the Color Theming module. The basic principle of operation is that when the scene starts, the UI elements are collected into a list with the corresponding color identifiers. When it is necessary to change the color theme of the application, the UI elements updates by a color identifiers.
2. In the application settings, it is possible to form data of user activity into a table into a file. This module is called CSVCreator.
3. Since the application has the ability to receive scheduled local notifications, the package of scripts for the Unity Notifications has been written for this. These helper scripts are located in the Scripts / Modules / Notifications folder.
4. To implement the flexible architecture of the application, a set of scripts was written, which is located in the Packages / HomeTools folder. Main features implemented:
   - **Debug/ScreenDebug** – this is the ability to display messages on the screen from code;
   - **Device/ApplicationChrome** – this is the ability to work with the status bar of the device itself;
   - **Device/Vibration** – this is a script with the implemented ability to enable vibration on a mobile device running Android;
   - **Handling/HandleObject** – this is the wrapper for adding Event Triggers to UI objects;
   - **Handling/HandleMotionSystem** – this is the class that allows you to define user swipes;
   - **HPrefs** – these are scripts that implement saving information similarly to PlayerPrefs. Saving occurs in a separate file on the device;
   - **Input/InputHS** – this is a static class with actual information about the position of the click and the speed of movement of this click;
   - **Messenger** – this is the class for storing a set of Actions that can be called on a transmitted message;
   - **Other** – this is the set of additional methods for working with RectTransform, text, days, etc.;

- **Source** – this is the set of tools for synchronizing classes with MonoBehaviour, as well as tools for working with color and transparency of UI elements, etc.

8. Description of scripting systems

The main application scripts are located in the Scripts folder.

List of systems in the Scripts / Architecture folder:

- **AboutSeaCalendar** – this is a set of scripts that are responsible for displaying the description of the Sea Calendar;
- **AcceptModule** – this is a set of scripts that are responsible for displaying and operating the confirmation window;
- **CalendarModule** – this is a set of scripts that are responsible for the display and behavior of the calendar window;
- **ChooseIconModule** – this is a set of scripts that are responsible for generating, displaying and behavior of the choose task icon window (on the task creation and editing pages);
- **Components** – this is a set of additional scripts for moving pages, the main buttons and working with events;
- **CreateTasksArea** – this is a set of scripts that are responsible for the display and behavior of the task creation page;
- **Data** – this is a set of scripts that are responsible for storing, transforming, and serializing data;
- **DaysMarkerArea** – this is a set of scripts that are responsible for the display and behavior of the Sea Calendar page;
- **EditTaskArea** – this is a set of scripts that are responsible for displaying and working of the task edit page;
- **Elements** – this is a folder for additional components;
- **MenuArea** – this is a set of scripts that are responsible for the operation and display of the menu (which appears in the upper right corner);
- **ModuleReminders** – this is a set of scripts that are responsible for the creation and operation of the UI part of the notification settings;
- **ModuleTrackFlow** – this is a set of scripts that are responsible for the scroll wheel for recording the progress of a task;
- **ModuleTrackTime** – this is a set of scripts that are responsible for displaying the time scroll wheel. This window is displayed when need to setup time for receiving notifications;
- **Other** – this is a set of scripts that are responsible for additional functions for loading task icons from resources. In addition, it contains a script that is responsible for the names of the days of the week and a script that is responsible for the delay before closing the window;
- **Pages** – this is a set of scripts that are responsible for scrolling pages, as well as animating the transition between pages;
- **PurchaserView** – this is a set of scripts that are responsible for displaying the purchase page;
- **SettingsArea** – this is a set of scripts that are responsible for the operation of the settings page;
- **Statistics** – this is a set of basic scripts for creating statistics windows that can be scrolled horizontally;

- **StatisticsArea** – this is a set of scripts that are responsible for the operation of the activity page in the application;
- **TaskViewArea** – this is a set of scripts that are responsible for displaying and working of task pages (task in progress and task in finished state);
- **TextHolder** – this is a set of scripts that are responsible for displaying the localization of the application and linking with the I2 Languages plugin;
- **TipModule** – this is a set of scripts that are responsible for displaying and operating the tip window;
- **TrackArea** – this is a set of scripts that are responsible for displaying and operating the track progress window;
- **TutorialArea** – this is a set of scripts that are responsible for the display and operation of the manual in the application;
- **WorkArea** – this is a set of scripts that are responsible for the operation of the application's main page;
- **MainActivity** – this is a set of scripts that are responsible for loading and running the top bar.

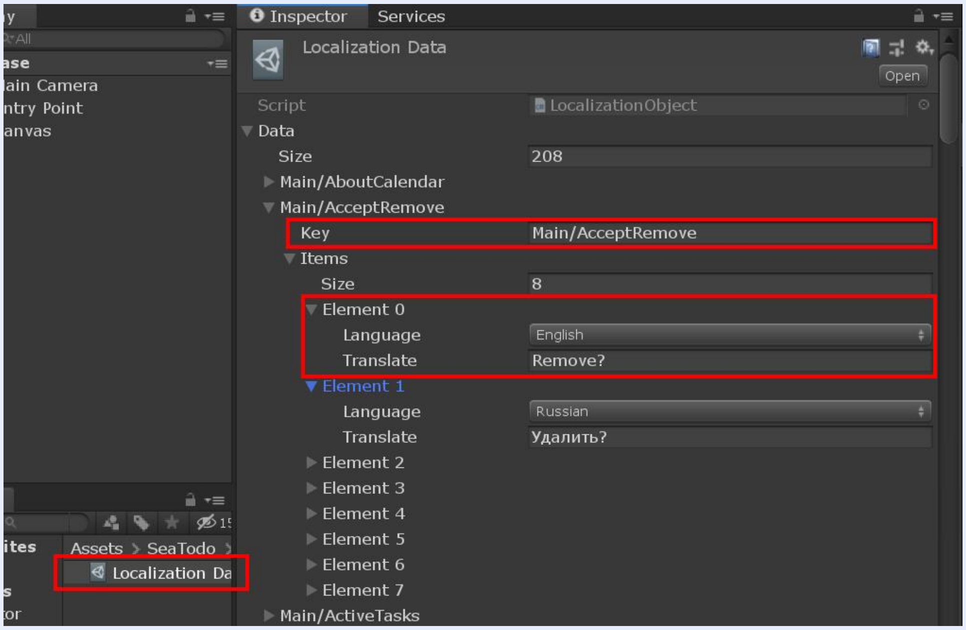
List of systems in the Scripts/Modules folder:

- **AppSettings** – these are the scripts that are responsible for creating the status of the device bar;
- **ColorPackage** – this is the script that is responsible for grouping UI elements and quickly adjusting their color;
- **ColorTheming** – this is a set of scripts that are responsible for organizing the setting of color themes for the UI;
- **CSVCreator** – this is a set of scripts that are responsible for creating a file with a table that contains information about user activity;
- **Notifications** – this is a set of scripts that are responsible for working with Unity Notifications;
- **Localization** – this is a set of scripts that are responsible for app localization;

9. Additionally

How to add text localization?

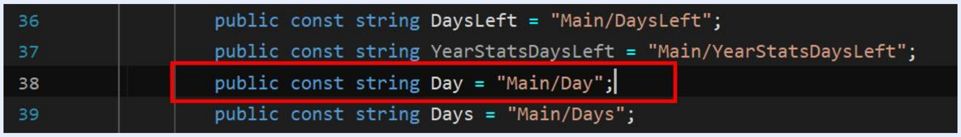
1. To add localization of any text, you need to add a key and text fields for languages in the scriptable object in Resources folder (Localization Data).



Localization Data

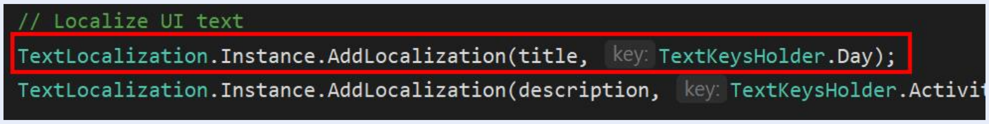
```
Script              LocalizationObject
▼ Data
    Size            208
  ▶ Main/AboutCalendar
  ▼ Main/AcceptRemove
      Key           Main/AcceptRemove
    ▼ Items
        Size        8
      ▼ Element 0
          Language  English
          Translate Remove?
      ▼ Element 1
          Language  Russian
          Translate Удалить?
      ▶ Element 2
      ▶ Element 3
      ▶ Element 4
      ▶ Element 5
      ▶ Element 6
      ▶ Element 7
  ▶ Main/ActiveTasks
```

2. It is necessary to create a field with a localization key in the **TextKeysHolder** class**.**



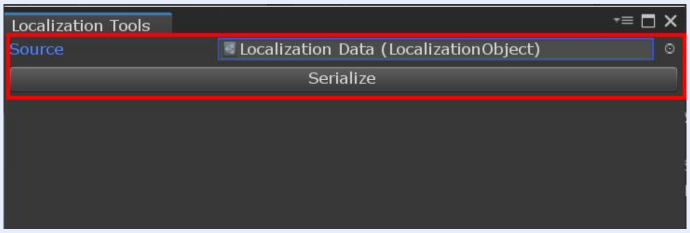
```
        public const string DaysLeft = "Main/DaysLeft";
        public const string YearStatsDaysLeft = "Main/YearStatsDaysLeft";
        public const string Day = "Main/Day";
        public const string Days = "Main/Days";
```

3. Add localization of the text component once in the script from anywhere.



```
// Localize UI text
TextLocalization.Instance.AddLocalization(title,  key: TextKeysHolder.Day);
TextLocalization.Instance.AddLocalization(description,  key: TextKeysHolder.Activi
```

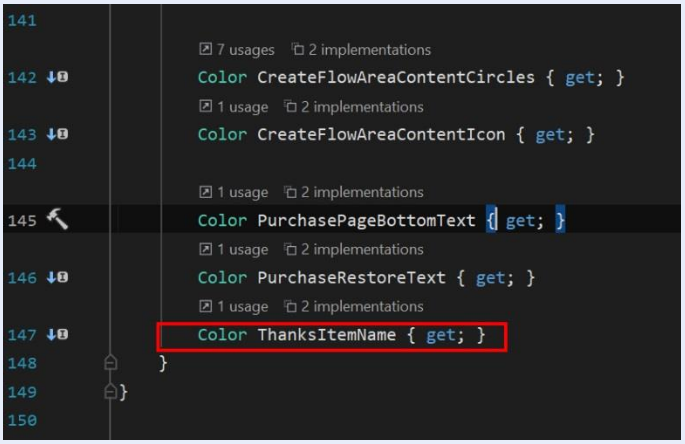
4. After edit Scriptable Object you should serialize file by tool (Tools / Localization Serialize):



Source   Localization Data (LocalizationObject)

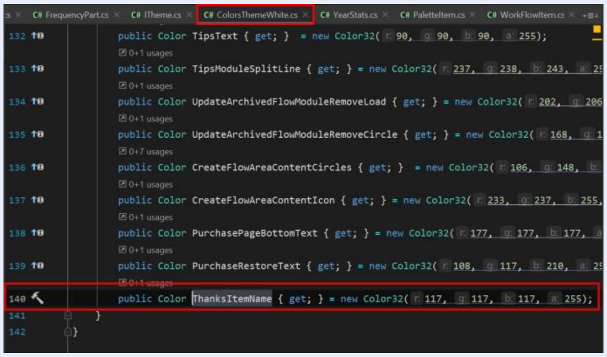
How to add new colors to the Color Theming module?

1. Add color field to **ITheme** interface**.**



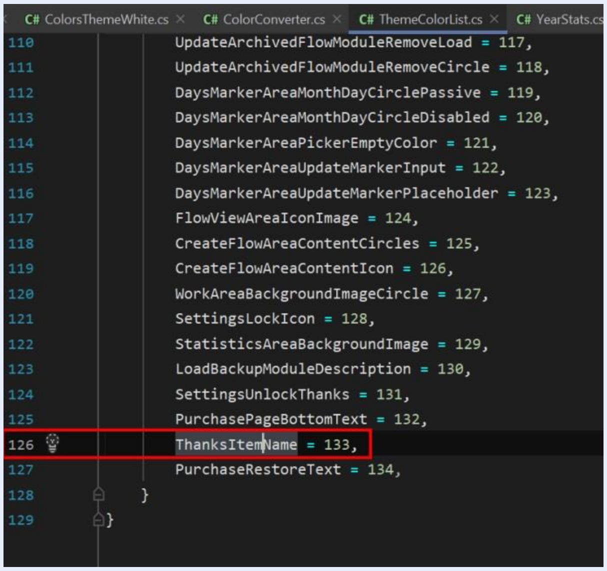
```
        7 usages   2 implementations
        Color CreateFlowAreaContentCircles { get; }
        1 usage   2 implementations
        Color CreateFlowAreaContentIcon { get; }

        1 usage   2 implementations
        Color PurchasePageBottomText { get; }
        1 usage   2 implementations
        Color PurchaseRestoreText { get; }
        1 usage   2 implementations
        Color ThanksItemName { get; }
    }
}
```

2. Implement a new field in the dark and light classes of the application.



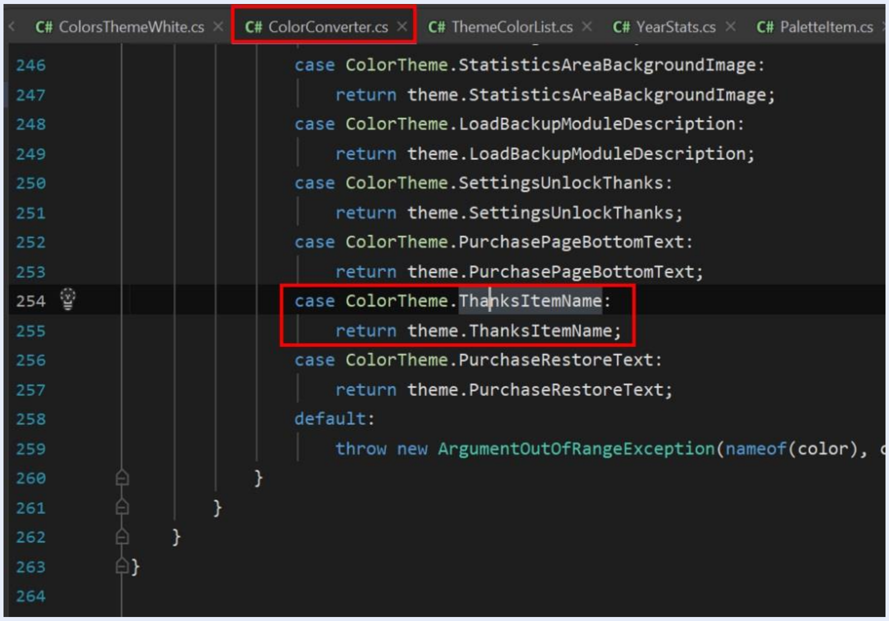
```
        public Color TipsText { get; }  = new Color32( r: 90, g: 90, b: 90, a: 255);
        0+1 usages
        public Color TipsModuleSplitLine { get; } = new Color32( r: 237, g: 238, b: 243, a: 25
        0+1 usages
        public Color UpdateArchivedFlowModuleRemoveLoad { get; } = new Color32( r: 202, g: 206
        0+1 usages
        public Color UpdateArchivedFlowModuleRemoveCircle { get; } = new Color32( r: 168, g: 1
        0+7 usages
        public Color CreateFlowAreaContentCircles { get; }  = new Color32( r: 106, g: 148, b:
        0+1 usages
        public Color CreateFlowAreaContentIcon { get; } = new Color32( r: 233, g: 237, b: 255,
        0+1 usages
        public Color PurchasePageBottomText { get; } = new Color32( r: 177, g: 177, b: 177, a
        0+1 usages
        public Color PurchaseRestoreText { get; } = new Color32( r: 108, g: 117, b: 210, a: 25
        0+1 usages
        public Color ThanksItemName { get; } = new Color32( r: 117, g: 117, b: 117, a: 255);
    }
}
```

3. Add field identifier to **ThemeColorList.**



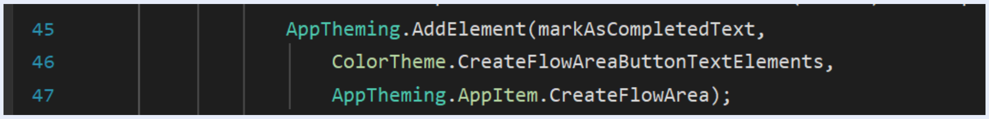
```
        UpdateArchivedFlowModuleRemoveLoad = 117,
        UpdateArchivedFlowModuleRemoveCircle = 118,
        DaysMarkerAreaMonthDayCirclePassive = 119,
        DaysMarkerAreaMonthDayCircleDisabled = 120,
        DaysMarkerAreaPickerEmptyColor = 121,
        DaysMarkerAreaUpdateMarkerInput = 122,
        DaysMarkerAreaUpdateMarkerPlaceholder = 123,
        FlowViewAreaIconImage = 124,
        CreateFlowAreaContentCircles = 125,
        CreateFlowAreaContentIcon = 126,
        WorkAreaBackgroundImageCircle = 127,
        SettingsLockIcon = 128,
        StatisticsAreaBackgroundImage = 129,
        LoadBackupModuleDescription = 130,
        SettingsUnlockThanks = 131,
        PurchasePageBottomText = 132,
        ThanksItemName = 133,
        PurchaseRestoreText = 134,
    }
}
```

4. Add search for the interface field by the created identifier to **ColorConverter**.



```
                case ColorTheme.StatisticsAreaBackgroundImage:
                    return theme.StatisticsAreaBackgroundImage;
                case ColorTheme.LoadBackupModuleDescription:
                    return theme.LoadBackupModuleDescription;
                case ColorTheme.SettingsUnlockThanks:
                    return theme.SettingsUnlockThanks;
                case ColorTheme.PurchasePageBottomText:
                    return theme.PurchasePageBottomText;
                case ColorTheme.ThanksItemName:
                    return theme.ThanksItemName;
                case ColorTheme.PurchaseRestoreText:
                    return theme.PurchaseRestoreText;
                default:
                    throw new ArgumentOutOfRangeException(nameof(color),
            }
        }
    }
}
```

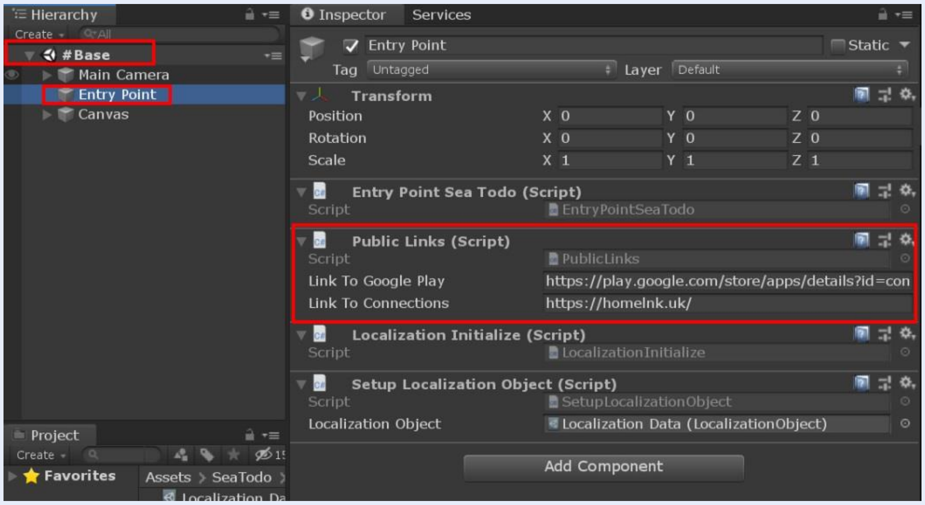
5. Add an UI element to the module with a new color identifier for assigning a color using the class **AppTheming**.



```
AppTheming.AddElement(markAsCompletedText,
    ColorTheme.CreateFlowAreaButtonTextElements,
    AppTheming.AppItem.CreateFlowArea);
```

Where to setup links to site and to market?

For part of settings "Other" you can setup your own links in:



Entry Point ☑ □ Static
Tag Untagged Layer Default

Transform
Position X 0 Y 0 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1

Entry Point Sea Todo (Script)
Script EntryPointSeaTodo

Public Links (Script)
Script PublicLinks
Link To Google Play https://play.google.com/store/apps/details?id=con
Link To Connections https://homelnk.uk/

Localization Initialize (Script)
Script LocalizationInitialize

Setup Localization Object (Script)
Script SetupLocalizationObject
Localization Object Localization Data (LocalizationObject)

Thanks for your purchase!

If you have any questions or suggestions, please write to me.

Mail:
andrew.olenk@gmail.com

Visit:
https://homelnk.uk

UpWork:
https://www.upwork.com/freelancers/~014feeec050fe93daa

Andrew