

# Assignment 2

Due Date: November 11, 2024 at 7:00 PM

## Overview

This assignment will be worth **10%** of your total grade.

## Deployment Target

Please note that the “Minimum Deployments” within the target should be set to 17.5. Anything other than 17.5 will result in a 10% deduction. You will also receive a 10% deduction if you have an incorrect submission format.

## Specifications

The goal of this assignment is to create a location-based task manager app in SwiftUI, integrating MapKit, Core Location, and some form of persistent storage. The app should enable users to create, manage, and track tasks based on their current location. Tasks should be saved locally so that when the app is re-run, the tasks will still appear. You cannot import any 3rd party libraries (via SPM) to complete this assignment.

Design an iOS app named “LocationTasks” that allows users to create, view, and manage tasks associated with specific locations. The app should consist of the following components:

## Map View - 25 points

### Home Screen

Your user interface will contain a list of the following pieces (a NavigationView with the entire view being a map):

- In a NavigationView, have a title named “LocationTasks” with a button in the top right that states “View Tasks”.
- Display the user’s current location as a marker on the map.
- Allow users to zoom in/out and pan across the map freely.
- Enable long-press gesture recognition for adding new tasks at specific locations.

- When a task is added, it should appear as a marker on the map.

## Task Creation Pop-up - 25 points

- A sheet popup triggered by a long-press gesture on the map.
- Include input fields for the task name, description, and due date (make this a Date data type).
  - All fields must have at least one character filled out, otherwise, the confirm button should be disabled.
  - A past date cannot be added, it must be set for a future date.
- Option to confirm or cancel task creation.
  - Both options should dismiss the view.
  - If a user confirms the task, it will be added in persistent storage (it is up to you how this is stored).

## Task List View - 15 points

Appears when the “View Tasks” Navigation button is clicked

- Display all created tasks, including their name, description, due date (day, month, and year), and lat/lng of the task.
- Allow users to swipe to remove a task from the list.
  - If a task is removed, it should be removed from the list, the map when the user hits the back button, and persistent storage.

## Persistent Storage - 15 points

- Efficient use of data storage, removal, and retrieval.

## Coding Style - 20 points

You will be evaluated on:

- Cleanliness of code (both the SwiftUI view abstractions and the Swift code)
- Directory structure of files
- The data structure(s) used

## Notes

- Long press on MapKit, this will help point you in the right direction: <https://stackoverflow.com/questions/71399626/anybody-got-swiftuimapkitlongpress-working>
- Remove item from a list (UI): <https://www.hackingwithswift.com/quick-start/swiftui/how-to-let-users-delete-rows-from-a-list>
- Date components for pulling out day/month/year: <https://www.hackingwithswift.com/books/ios-swiftui/working-with-dates>

## Submission

Please submit the following on BlackBoard:

- .zip of the entire Xcode project