

User Guide

Overview

Priority is a task management application designed to help users organize, prioritize, and complete their to-do's efficiently. With automatic sorting, completion tracking, and smooth iOS integration, it's an ideal tool for managing daily responsibilities.

Installation & Setup

System Requirements

- **Device:** Computer running MacOS
- **Development Environment:** Xcode 15+
- **Dependencies:**
 - [Auth0.swift](#)
 - Core Data
 - SwiftUI

Setup Instructions

1. Clone the repository from GitHub.
2. Open Priority folder in Xcode.
3. Install dependencies if not already integrated via Swift Package Manager:
 - a. Auth0 is already declared in project.pbxproj and linked.
 - b. Core Data and SwiftUI come with XCode
4. Run the app using a connected iOS device or simulator.
5. Login
 - a. **You MUST login using the iOS device or app simulator. Logging in from the preview will not work, as it is not the full app.**
 - b. Credentials
 - i. Username: tester@gmail.com
 - ii. Password: testerPassword1!

Features & Functionality

Task Creation & Management

- Create new tasks with:
 - Title
 - Due date
 - Estimated time to complete
 - Priority Level
- All fields are optional
- All data is stored and persisted via **Core Data**.

Task Organization

- Tasks are organized by their Priority Score which is based on the due date and biased by the time to complete and the Priority score. This score is shown on the list view on the right of a given task.
- Displayed tasks are always shown as:
 - Incomplete tasks first
 - Completed tasks last

Completion Tracking

- **Completion percentage** dynamically calculates how many tasks have been finished out of the total.
- Visual progress tracking is supported via view models.

Edit & Delete Tasks

- Tasks can be updated with new titles or details by swiping left and selecting the pen icon.
- Tasks can be deleted individually by swiping left and selecting the trashcan icon.

Auth0 Integration

- The app uses **Auth0** for authentication and credential management.
- The .entitlements file includes support for **web credentials**, allowing seamless sign-in experiences.

Navigating & Using the App

Main Views

1. Content View

- a. This is the app's home screen
- b. Shows the completion percentage at the top with a circle graphic, which is the Progress Ring View
- c. Holds the Task List View below that
- d. Contains a menu at the bottom to open side menus or add a task

2. Task List View

- a. Displays all tasks based on their Priority Score and completion.
- b. Allows toggling completion status by swiping right or tapping the card.

3. Task Editor

- a. Triggered when creating (Plus button at bottom of the screen) or editing a task (Swipe left on a task and select pen icon).
- b. Provides form-style inputs for task metadata.
- c. Can cancel or save tasks with buttons at top.

4. Options/Navigation Panels

- a. Triggered by the Hamburger (Navigation) or 3 dot (Options) buttons at the bottom of the screen.
- b. Both are menu views.

5. Calendar View

- a. Accessed through the navigation panel.
- b. Shows a calendar with due dates highlighted. If no due date is provided for a task, it is included in the current (today's) date.
- c. Click on a date to see the assignments due on the given date.

6. Profile View

- a. A login screen is shown when opening the app if there is no user signed in.
- b. Otherwise, a view displaying user info and a log-out button can be accessed through the navigation pane.

7. Widget Extension (Optional)

- a. The app includes a `PriorityWidgetExtension.appex` target that can display selected tasks on the home screen as a widget.
- b. Built using `WidgetKit + SwiftUI`.
- c. Right now, there is example data plugged in. For a fully functional widget, the developer would need a paid apple developer subscription because app groups are not available with the free tier. App groups are required for information sharing between that app and the widget.

Advanced: Developer Notes

- **Core Data Stack:** Managed via a singleton TaskManager.
- **Architecture:** MVVM with TaskViewModel and EditTaskViewModel.
- **Model:** Includes Task and Category entities defined in TasksDataModel.
- **Environment Setup:** Swift version 5, gnu++20 for C++ interoperability, full AppIcon and AccentColor support.

Summary

The **Priority App** offers an elegant and functional solution for personal productivity. Designed with modern iOS frameworks and a clean MVVM architecture, it allows users to take control of their tasks with ease and clarity.