

Activity Finder Installation Guide for macOS

This guide will walk you through the complete process of setting up and running the Activity Finder app on a new Mac. Follow these steps carefully to ensure a successful installation.

Prerequisites

Before starting, make sure you have:

- A Mac running macOS 10.15 (Catalina) or newer
- An Apple Developer account (free or paid)
- Xcode 13 or newer installed
- Basic knowledge of iOS development
- Internet connection

Step 1: Install Xcode

If you don't have Xcode installed:

- Open the App Store on your Mac
- Search for "Xcode"
- Click "Get" or "Download" to install
- Wait for the installation to complete
- Launch Xcode and accept the license agreement
- Allow Xcode to install additional components if prompted

Step 2: Download project files

- Download the project ZIP file
- Extract the contents to your preferred location (e.g., Documents folder)

Step 3: Configure Firebase

The Activity Finder app uses Firebase for authentication and data storage. You'll need to set up your own Firebase project:

- Go to Firebase Console
- Sign in with your Google account
- Click "Add project"
- Name your project (e.g., "Activity Finder")
- Follow the setup wizard (you can disable Google Analytics if not needed)
- Click "Create project"

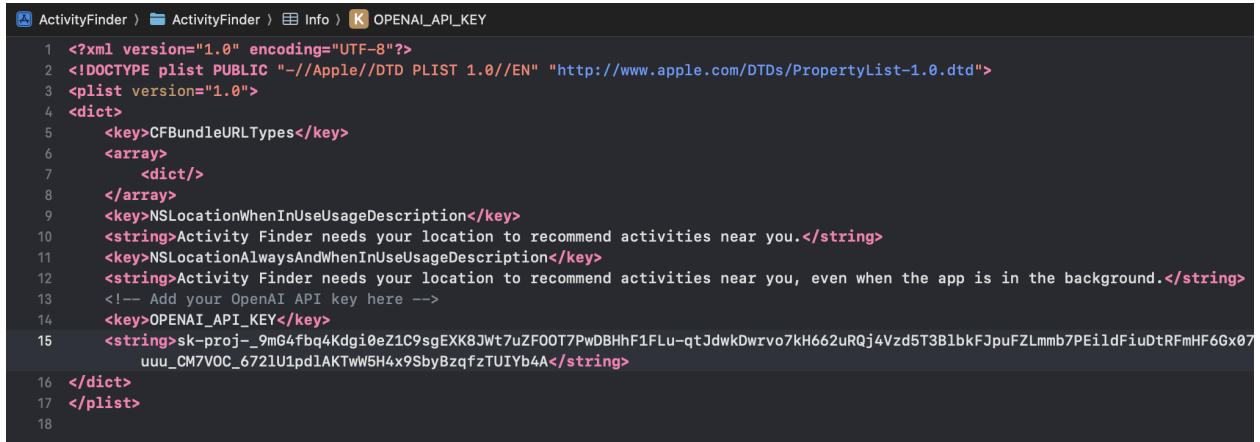
Once your project is created:

- Click "iOS" icon to add an iOS app
- Enter your Bundle ID (e.g., "ID.Activity-Finder")
- Enter the app name and nickname

Register the app

Download the GoogleService-Info.plist file

In Xcode, drag the downloaded file into the project navigator (make sure "Copy items if needed" is selected)



```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
3 <plist version="1.0">
4 <dict>
5 <key>CFBundleURLTypes</key>
6 <array>
7 <dict/>
8 </array>
9 <key>NSLocationWhenInUseUsageDescription</key>
10 <string>Activity Finder needs your location to recommend activities near you.</string>
11 <key>NSLocationAlwaysAndWhenInUseUsageDescription</key>
12 <string>Activity Finder needs your location to recommend activities near you, even when the app is in the background.</string>
13 <!-- Add your OpenAI API key here -->
14 <key>OPENAI_API_KEY</key>
15 <string>sk-proj-9mG4fbq4Kdgi0eZ1C9sgEXK8JWt7uZF00T7PwDBHhF1FLu-qtJdwkDwrvo7kH662uRQj4Vzd5T3B1bkFJpuFZLmmb7PEildFiuDtrFmHF6Gx07
uuu_CM7VOC_672lU1pd1AKTw5H4x9SbyBzqfzTUIYb4A</string>
16 </dict>
17 </plist>
18
```

Step 4: Install Dependencies

The app uses Firebase SDK, which can be installed using Swift Package Manager:

Open the project in Xcode

Go to File > Add Packages...

In the search bar, enter: <https://github.com/firebase/firebase-ios-sdk>

Select the Firebase iOS SDK package

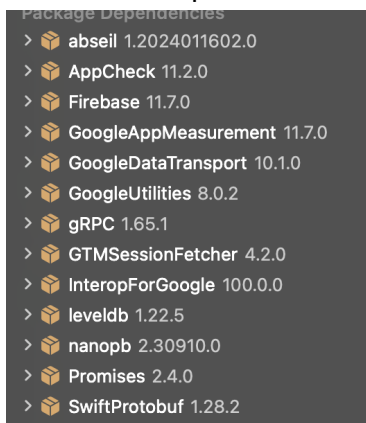
Choose the appropriate dependencies:

FirebaseAuth

FirebaseFirestore

Click "Add Package"

Wait for the dependencies to download and configure



```
Package Dependencies
> abseil 1.2024011602.0
> AppCheck 11.2.0
> Firebase 11.7.0
> GoogleAppMeasurement 11.7.0
> GoogleDataTransport 10.1.0
> GoogleUtilities 8.0.2
> gRPC 1.65.1
> GTMSessionFetcher 4.2.0
> InteropForGoogle 100.0.0
> leveldb 1.22.5
> nanopb 2.30910.0
> Promises 2.4.0
> SwiftProtobuf 1.28.2
```

Step 5: Configure Xcode Project

Open the project in Xcode

Click on the project file in the navigator (the blue icon at the top)

Select your target under "TARGETS"

Go to the "Signing & Capabilities" tab

Sign in with your Apple ID in the Xcode preferences if you haven't already

Check "Automatically manage signing"

Select your team from the dropdown

Step 6: Update Firebase Configuration

Make sure the Firebase configuration matches your project:

Check that GoogleService-Info.plist is correctly added to the project

Verify the Bundle ID in Xcode matches the one you registered with Firebase

Make sure the `FirebaseApp.configure()` call is in your AppDelegate (it should already be there)

Step 7: Build and Run

Connect an iOS device or use the iOS Simulator

In the top-left of Xcode, select your run destination (device or simulator)

Click the play button or press `Cmd+R` to build and run

The app should launch and show the login screen

Step 8: Set Up Firestore Database Rules

To ensure proper data storage and security:

In the Firebase console, go to "Firestore Database"

Click "Create database" if you haven't created one yet

Start in test mode for development purposes

Go to the "Rules" tab and set up appropriate security rules:

```
rules_version = '2';
service cloud.firestore {
  match /databases/{database}/documents {
    match /user_preferences/{userId} {
      allow read, write: if request.auth != null && request.auth.uid == userId;
    }
  }
}
```

Step 9: Test the Application

To verify everything is working:

Create a new user account from the app

Test the login functionality

Navigate to the preferences section and select some activities

Save your preferences and verify they persist when you log out and back in

Troubleshooting

If you encounter issues:

Build Errors: Make sure all Firebase dependencies are properly installed

Login Failures: Verify Firebase Authentication is enabled in the Firebase Console

Database Issues: Check Firestore rules and ensure the database is created

Permission Errors: For location functionality, ensure the proper usage descriptions are in Info.plist