

Code Documentation for AdoraWellness

Overview

AdoraWellness is a comprehensive iOS mobile application designed to connect wellness enthusiasts with certified yoga and pilates instructors. The application provides a seamless platform for booking sessions, accessing guided video lessons, and managing wellness routines. Built using the **Model-View-ViewModel (MVVM)** architecture with **Swift** and **SwiftUI**, it integrates **Firebase Authentication**, **Firestore**, **EventKit**, **MapKit**, and **AVFoundation**.

Key Features

1. User Authentication & Role Management:

- Support for **Email and Password** and **Google Sign-In**.
- Dual user roles: **Students** and **Instructors** with role-based access control.
- Secure password reset functionality through Firebase.
 - [Firebase Authentication Setup](#)
 - [Google Sign-In Integration](#)

2. Firebase Firestore Database:

- Stores user profiles, session schedules, booking data, and instructor information.
- Supports real-time CRUD operations for session management and user interactions.
- Handles secure payment transaction records and user preferences.
 - [Firestore Database Schema](#)
 - [Firestore CRUD operations](#)

3. Session Booking & Management:

- Real-time session creation and booking system.
- Calendar integration using **EventKit** for reminders and scheduling.
- Push notifications for booking confirmations and session reminders.
 - [Calendar Integration](#)
 - [Reminder Integration](#)
 - [Push Notification](#)

4. Video Content Delivery:

- **AVFoundation** integration for smooth video streaming.

- [AVFoundation Video Player](#)
- [Core Data Favorites System](#)

5. Location Services & Maps:

- **MapKit** integration for yoga and pilates studio discovery.
- Interactive maps with custom annotations for studios.
 - [MapKit Studio Locations](#)
 - [Custom Map Annotations](#)

Dependencies

Dependency	Purpose
Firebase Auth	Secure user authentication and role management
Firestore	Real-time cloud database for sessions, users, and session bookings
Core Data	Local storage for favorite lessons
EventKit	Calendar integration and reminder management
MapKit	Yoga and pilates studio discovery Location services
AVFoundation	Video streaming and audio/video playbook
Google Sign-In SDK	Third-party Google authentication
UserNotifications	Push notifications for booking and session reminders

Architecture Pattern

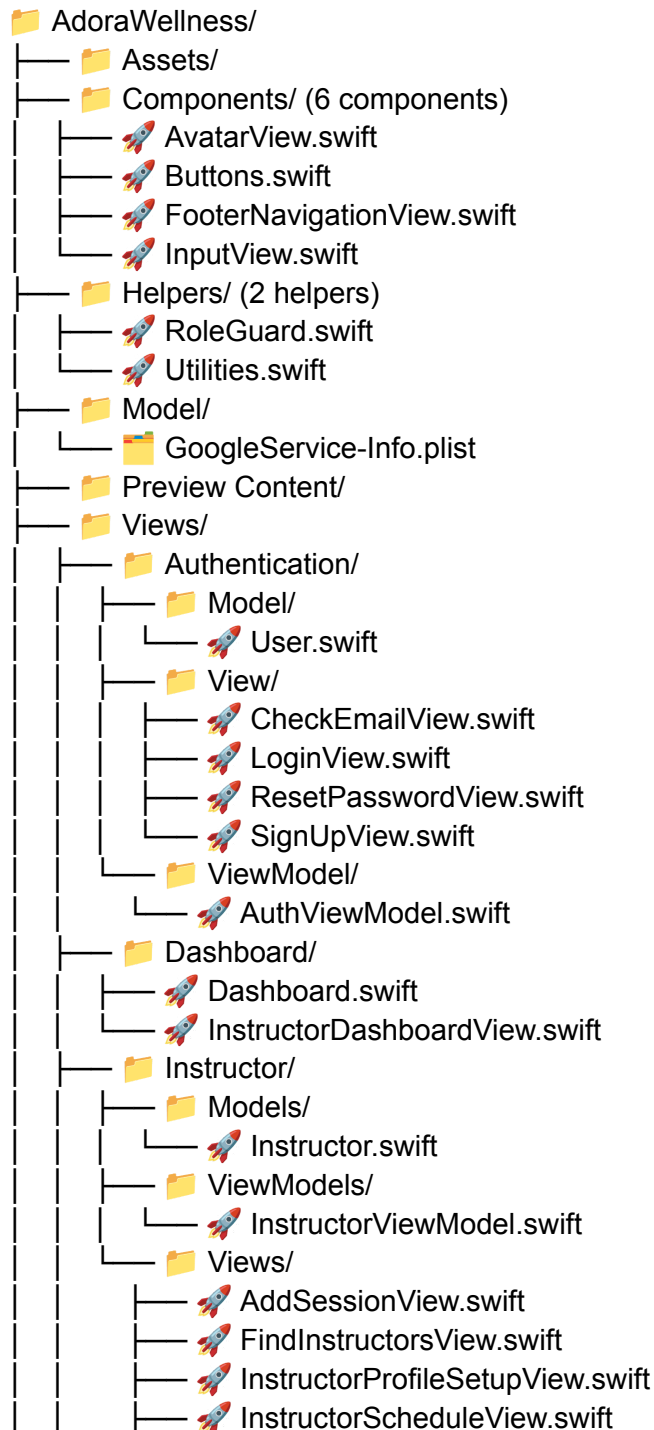
MVVM (Model-View-ViewModel) Implementation

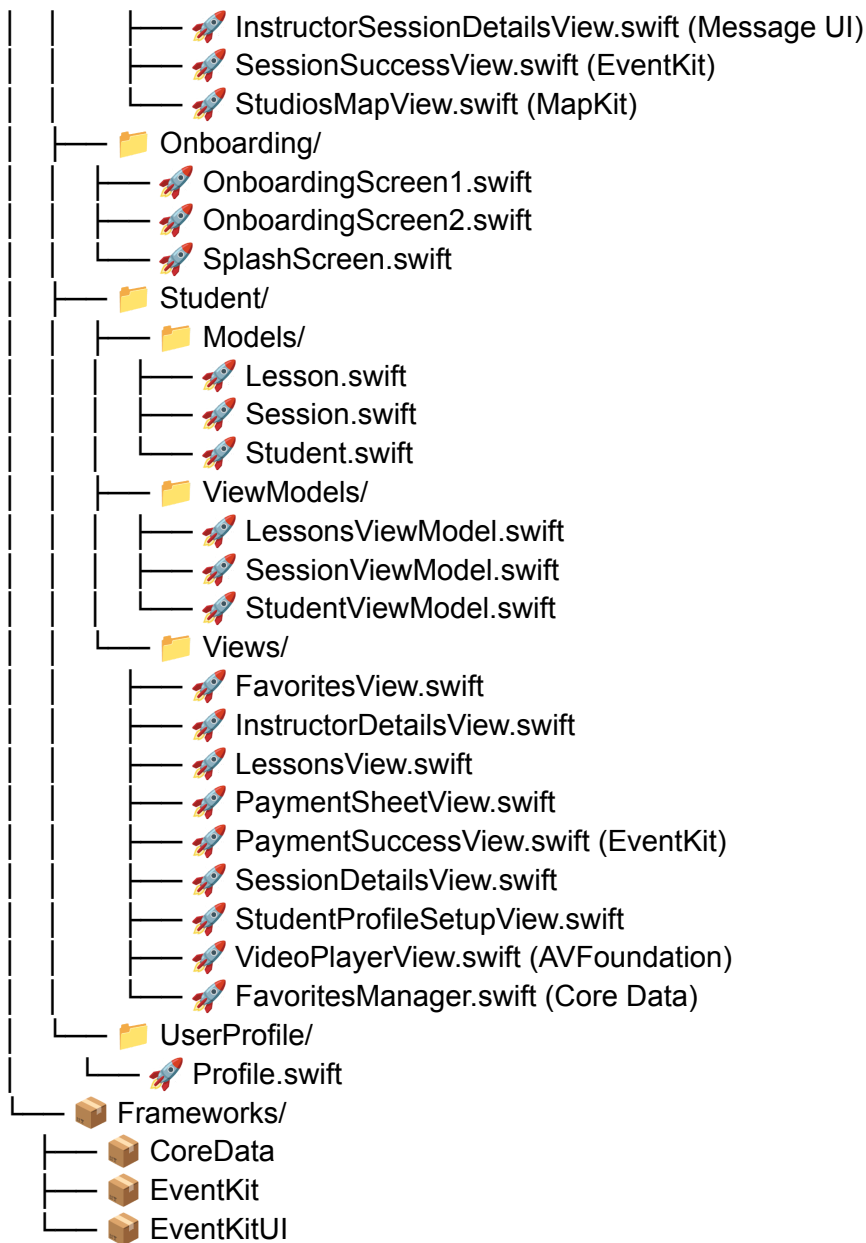
The project uses the **Model-View-ViewModel (MVVM)** pattern to separate concerns and maintain clean code organization:

- **Models:** Core data structures including [User](#), [Instructor](#), [Session](#), [Lesson](#), [Student](#)
- **Views:** SwiftUI interfaces for authentication, dashboards, session management, video player, and profile settings
- **ViewModels:** Business logic controllers, including:
 - [AuthViewModel](#) - Authentication and role management

- `InstructorViewModel` - Instructor-specific operations
- `SessionViewModel` - Session booking and management
- `LessonsViewModel` - Video content management
- `StudentViewModel` - Student-specific functionality

AdoraWellness Project Structure





Setup Steps

1. Clone the Repository

git clone <https://github.com/hashiniRanasinghe/AdoraWellnessCW2.git>

2. Open Project

Open [AdoraWellness.xcodeproj](#) or [AdoraWellness.xcworkspace](#) in Xcode.

3. Add Dependencies

Use Swift Package Manager to add the following dependencies:

- Firebase iOS SDK: <https://github.com/firebase/firebase-ios-sdk>
- Google Sign-In SDK: <https://github.com/google/GoogleSignIn-iOS>

4. Configure Firebase

- Create a new Firebase project in the Firebase Console
- Add `GoogleService-Info.plist` to the Xcode project root
- Enable Authentication, Firestore Database, and Cloud Messaging
- Configure authentication providers (Email/Password, Google)

5. Set Up Authentication

Follow the guides for Google sign-in integration.

6. Run the App

- Use a physical device for full functionality testing
- Simulator can be used for UI development and basic testing

Documentation References

Topic	Implementation Area	Code Reference
Firebase Authentication Setup	User login/signup system	AuthViewModel.swift, User.swift
Google Sign-In Integration	Third-party authentication	AuthViewModel.swift (Google integration)
Firebase Firestore Configuration	Database setup and CRUD operations	SessionViewModel.swift, StudentViewModel.swift
Real-time Session Booking	Session management system	SessionViewModel.swift, SessionDetailView.swift
Core Data Integration	Offline favorites storage	FavoritesManager.swift
EventKit Calendar Integration	Session reminders and scheduling	SessionSuccessView.swift, PaymentSuccessView.swift

MapKit Studio Locations	Location services and maps	StudiosMapView.swift
AVFoundation Video Streaming	Video content delivery	VideoPlayerView.swift, LessonsViewModel.swift
Push Notifications Setup	Booking and session reminders	Integrated in ViewModels
MVVM Architecture Implementation	Project structure and organization	All ViewModels/ folders
Role-Based Access Control	Student/Instructor differentiation	AuthViewModel.swift, RoleGuard.swift