



# Scheduly – Appointment Booking App Template

**Version:** 1.0

**Author:** John Omeyo

**Framework:** Flutter

---



## Table of Contents

1. Introduction
  2. Features
  3. Folder Structure
  4. Installation
  5. How to Run the App
  6. Customization Guide
  7. API Integration
  8. Troubleshooting
  9. Support
- 



## Introduction

**Scheduly** is a modern and clean Flutter app template for appointment booking services. It's fully designed with reusability in mind, making it easy to plug in any backend via API.

Whether you're building a salon booking app, doctor appointments, or service scheduling — Scheduly gets you started fast.

---



## Features

- 100% Flutter Code (Cross-platform for Android & iOS)

- Clean and scalable architecture
  - All key booking flow screens included
  - Ready-to-integrate data models
  - Easy theme, logo, and asset customization
  - Organized, well-commented code
- 

## Folder Structure

```
scheduly/
├── lib/
│   ├── models/      # All app data models
│   ├── pages/       # All UI screens
│   └── main.dart     # App entry point
├── assets/          # Images, icons, fonts
├── pubspec.yaml     # Dependencies and assets
├── android/         # Android native setup
├── ios/             # iOS native setup
└── documentation.pdf # This file
```

---

## Installation

- 1. Install Flutter**  
If not already installed, follow: <https://flutter.dev/docs/get-started/install>
  - 2. Clone or unzip this project**  
Navigate into the root folder in your terminal or IDE.
  - 3. Install dependencies**  
Run: `flutter pub get`
  - 4. Run on emulator or real device**  
  
Run: `flutter run`
- 

## How to Run the App

You can run the app using either an IDE like **VS Code** or **Android Studio** or via terminal:

```
flutter run
```

To build a release version:

```
flutter build apk --release
```

---

## Customization Guide

### Change App Name, Icon, and Colors:

- **App Name:** Edit `android/app/src/main/AndroidManifest.xml` and `ios/Runner/Info.plist`
- **App Icon:** Use [flutter\\_launcher\\_icons](#)
- **Theme Colors:** Modify the color variables in `lib/utils/constants.dart` or wherever theme data is defined

### Replace Assets:

- Put your own images in `/assets/`
  - Update references in `pubspec.yaml`
- 

## API Integration

The app is built to allow easy API integration.

1. **Locate the Models** in `lib/models/`
2. Each model supports `fromJson()` and `toJson()` methods
3. You can integrate API services using `http`, `dio`, or any preferred method

 **Tip:** Create a new file like `lib/services/api_service.dart` for organizing your API logic

---

## Troubleshooting

Problem	Solution
App doesn't run	Run <code>flutter clean</code> and <code>flutter pub get</code>
Assets not showing	Run <code>flutter pub get</code> after adding new files
Android/iOS build issues	Ensure Flutter and SDKs are up to date

---

## Support

If you encounter any bugs or issues, feel free to reach out via:

- Email: [youremail@example.com](mailto:youremail@example.com)

Please include:

- Your purchase code
- A detailed description of the issue
- Screenshots or logs if possible