Charan Konduri

Fit Streak 

July 13, 2025

Overview

FitStreak is a comprehensive Android fitness tracking application. It lets its users track multiple activities like Step count, Water consumed, Calories burnt, Sleep hours etc. Users can set goals across these activities and monitor their progress throughout the day. They will also be presented with reports on their performance and goals for that entire week. Users can also search for healthy recipes through the app. The main objective of the app is to encourage and motivate its users to maintain consistency in their fitness journey. Built using modern Android development practices, the app integrates with Google Fit API for hardware sensor data, Firebase for authentication, and Spoonacular API for nutritional content

Functional Description

- The app lets the new users sign up and existing users sign in using their Email/Phone number/Google accounts.
- Once the user is in, he/she is welcomed with a couple of onboarding screens(only in the first run) to help setting up the goals to track.
- After the set up, he/she is taken to the home screen where the progress of goals achieved for the day can be seen. All the updates to the goals throughout the day can be done here.
- Users can choose to navigate to the Recipes screen, where recipes across different categories can be selected.
- At any given time, users can go to their dashboard to check their performance and consistency in achieving their goals across activities for a period of week. They can always edit their goals here on this screen.

Specifications

Technical Architecture

Design Pattern: MVVM (Model-View-ViewModel)
- Clear separation of concerns with reactive programming
- StateFlow/SharedFlow for lifecycle-aware data observation
- ViewModelScope for proper resource management

Dependency Injection: Dagger 2
- Modular component architecture with subcomponents for each feature
- Factory pattern implementation for ViewModels
- Scoped dependencies for efficient resource utilization

Data Layer
- Room database with coroutines for local storage
- Repository pattern with clean data source abstraction
- DataStore for user preferences and settings
- Retrofit with Moshi for network operations

UI Framework
- Material Design 3 with comprehensive theming
- Data Binding for declarative UI updates
- MotionLayout for complex animations and transitions
- RecyclerView with ViewHolder pattern for efficient list rendering

API Integrations

Google Fit API
- Runtime permission handling for health data access
- Activity recognition and sensor data collection
- Background data synchronization

Spoonacular API
- RESTful service integration with Retrofit
- JSON parsing with proper error handling
- Rate limiting and quota management

Firebase Services
- Authentication with Google Sign-In
- Real-time user state management
- Secure configuration management

Performance Specifications

Memory Management
- Efficient image loading with Glide and caching
- Proper lifecycle management preventing memory leaks
- Background thread operations for all I/O operations

Responsive Design
- Support for multiple screen densities and orientations
- Adaptive layouts using ConstraintLayout
- Smooth 60fps animations using MotionLayout

Milestones

- ☑ ~~Google Fit integration~~

  Application with all the login and fitness tracking(home screen) functionality integrated with Google Fit API is in place without much emphasis on styling and animations.

- ☑ ~~Recipes API integration~~

  Application now connects to Recipes Api(Spoonacular) and Recipes screen is implemented and presented.

- ☑ ~~Dashboard Presentation~~

  Dashboard screen is implemented and integrated with other screens to present the user specific information in form of charts and graphs.

- [x] ~~Animations~~

  Identify the opportunities for beautiful animations across the app and implement them.

- [x] ~~Testing~~

  Write unit and integration tests for the app.

- [x] ~~Consistent Styling and Accessibility~~

  Make sure the styling is consistent across the app and implement the accessibility features.

- [x] ~~Support landscape orientation~~

  Ensure the layout design works seamlessly in landscape orientation.