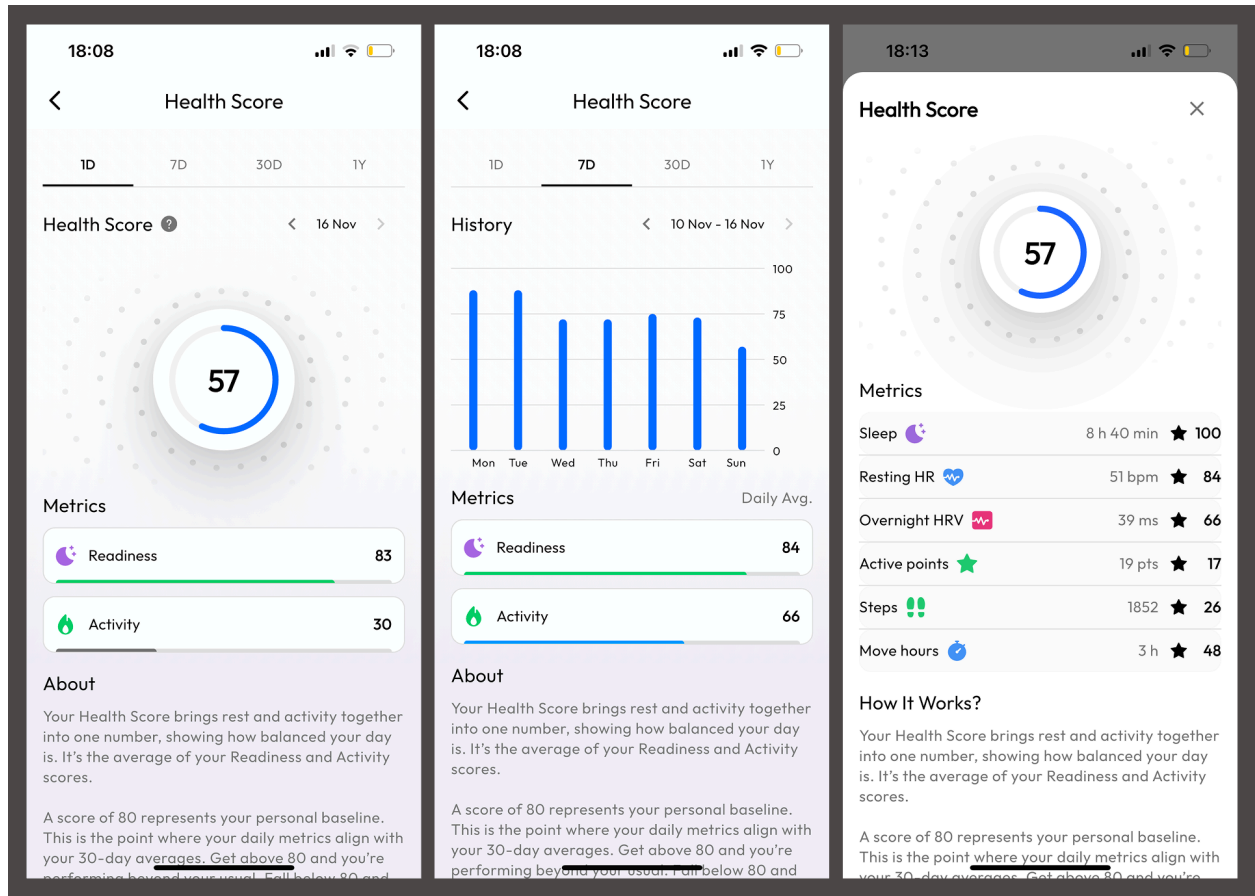# Score Detail Preview — Project Requirements



Build a small Flutter app that recreates the Score Detail screens from the provided UI references. The emphasis is on **reusable UI components**, **clean architecture**, **smooth interactions,** and **local JSON-based data simulation**.

**Reference UI:**
 Figma:
https://www.figma.com/design/L3X4gAEcmMFiopRdG8JBH2/Rolla-UI-Demo?node-id=0-1&t=p
APIILKY76HmQNjp-1

# 1. Tech Stack & Architecture

## Architecture Pattern

- Feature-first module with Clean Architecture.

Directory structure:

```
lib/features/scores/
├── data
├── domain
└── presentation
```

## State Management & Core Libraries

- `flutter_bloc`

- `freezed`

- `get_it`

- `dartz`

- `equatable`

- `intl`

## Error Handling

- No exceptions for control flow.

- Use `Either<Failure, T>` for domain responses.

## Dependency Injection

- Register all dependencies in `scores_injection.dart`.

- Use:

  - **Factories** → blocs

  - **Singletons** → services, caches

## Linting

- Must pass `flutter_lints` (or equivalent) with **zero new warnings**.

---

# 2. Scope & Features

## Home Screen

- Display 3 score cards (0–100 range):

  - **Health**

  - **Readiness**

  - **Activity**

- Each card shows:

  - Title

  - Value

  - Score value

  - Score progress indicator

- Tapping a card opens its **Score Detail** page.

- The same card widgets will be reused inside the Score Detail page.

---

## Score Detail Screen (per score type)

### General Requirements

- Unique accent color per score type (for charts).

- Pull-to-refresh, loading skeletons, empty/error states.

- Supports timeframes:

    - **1D / 7D / 30D / 1Y**

    - All sections must update accordingly.

### Charts

- **1D view:** custom radial gauge

- **7D, 30D, 1Y views:** line/bar charts

- Missing data → visual gaps

### Metrics Section

- Score-specific metric groups:

    - **Activity:** Active Points, Steps, Active Calories, Move Hours

    - **Readiness:** Sleep, Resting HR, HRV

    - **Health:** Activity + Readiness summary tiles

- Metric card displays:

- Title

- Value

- Progress bar with conditional coloring:

  - 0–50 → neutral

  - 50–79 → blue

  - 80+ → green

- Timeframe behavior:

  - **1D:** show daily totals

  - **7D / 30D / 1Y:** show daily averages + "Avg" label

- Metric tile with empty data → neutral style + "No data".

### Insights Section

- Short contextual insights (e.g., "Lower HRV decreased readiness today").

### Info Drawer / Accordion

- Definitions & short explanations for each metric.

---

## 3. UI & UX Requirements

- Match the Figma: layout, spacing, typography, and colors.

- Respect theme variables; support **light and dark mode**.

- Use **Google Font: Outfit**.

- Timeframe switch must update:

    - Charts

    - Gauge

    - Metrics

    - Trend visuals

    - Subtitle text consistently

---

# 4. Deliverables

## Repository

- Public GitHub repo containing full implementation.

## README Requirements

Include:

- Architecture overview (diagram optional but recommended)

- Setup & run instructions

- Notes on assumptions and component usage

## Screen Recording (60–90 seconds)

Must demonstrate:

1. Home → Score Detail navigation

2. Timeframe switching

3. Info overlay drawer

4. Loading states + an empty/error example

---

## 5. Additional Notes & Guidance

- Use `fl_chart` or custom painters for radial gauge and trend charts.

- Prefer small, focused functions.

- Prioritize reusable widgets over one-off UI.

- Keep business logic out of widgets: use blocs/cubits + use cases.