

Flutter Developer Intern – Assignment

Objective

Build a **custom Flutter UI component** that renders an **Astrological Kundali (North Indian style preferred)** using real astrological data fetched from a public Astrology API.

This task is designed to evaluate: - Flutter UI & custom painting skills - API integration & data handling - Attention to detail and validation - Code quality and documentation

Task Overview

You will build a **Flutter app** that: 1. Takes **birth details as user input** 2. Fetches kundali data from an Astrology API 3. Plots a **visual kundali chart** using a **custom Flutter component** (no static images) 4. Validates the output against app.atri.care

Functional Requirements

1. User Input Screen

Create a screen to collect the following inputs: - **Date of Birth** (DD/MM/YYYY) - **Time of Birth** (HH:MM, 24-hour format) - **Latitude** (decimal) - **Longitude** (decimal)

Basic validation is expected (empty fields, valid ranges, etc.).

2. Astrology API Integration

- Create a **free trial account** on any public **Astrology API** (example: AstrologyAPI, Prokerala, etc.)
- Fetch kundali / chart / planetary position data using the user inputs
- Clearly document:
 - API used
 - Endpoint name
 - Request & response structure

 Do NOT hardcode responses. API integration must be real.

3. Kundali UI Component

You must implement a **custom Flutter widget** to render the kundali.

UI Expectations:

- Draw the kundali using **CustomPainter** or equivalent
 - Display:
 - 12 houses
 - Zodiac signs per house
 - Planet abbreviations inside houses (e.g. Su, Mo, Ma, Me, Ju, Ve, Sa, Ra, Ke)
 - Layout should be clean, readable, and scalable
-  Do NOT use images, SVG downloads, or webviews
-

4. Validation Requirement (Very Important)

You must validate your output kundali with:

 <https://app.atri.care>

For the **same birth details**: - Cross-check house placements - Planet positions - Zodiac signs

Add a short note in README: - Which birth details you validated with - Whether the output matched exactly or had differences

Technical Requirements

- Flutter (latest stable)
- Clean folder structure
- Reusable widgets
- Proper naming conventions
- No crashes on invalid input

Bonus (optional): - Light animations - Dark mode support - Responsive layout

Submission Requirements

You must submit **all of the following**:

1. **APK file** (debug or release)
2. **Screen recording** showing:
 - Inputting birth details
 - API call
 - Kundali rendering

3. **GitHub repository** (public)
 - Complete source code
 - Proper commit history
 4. **README.md** containing:
 - Setup instructions
 - API used
 - Screenshots
 - Validation notes vs app.atri.care
-

Evaluation Criteria

- Accuracy of kundali rendering
 - Quality of custom UI implementation
 - Code readability & structure
 - Correct API usage
 - Validation effort & honesty
-

Final Notes

This assignment mirrors **real production work** at Atri. We care deeply about: - UI precision - Astrological correctness - Engineering discipline

Good luck 