Junior Backend Developer Coding Assessment

Objective

Develop a **.NET 8 Web API** that integrates with a **third-party API of your choice** (e.g., weather, news, or open APIs like JSONPlaceholder) while demonstrating knowledge in key technologies and practices.

Technical Requirements

Core Features

1. **Third-Party API Integration**

   - Create at least 1 API controller to:

     - Fetch data from an external API

     - Process/transform responses

     - Persist relevant data to a database

   - Example: *"Fetch weather data → store historical trends"*

   - As an extra, you can include Authentication and Authorization on your application (not mandatory).

2. **Database Implementation**

   - You can use any database for this application. Sql Server and/or MongoDB are encouraged.

   - As an extra, you can perform raw sql/nosql queries for more complex queries for knowledge demonstration (not mandatory)

3. **Documentation & Testing (not mandatory)**

   - Enabling Swagger UI with endpoint descriptions would be welcome.

   - Unit Tests would also be well received.

GitHub Submission Guidelines

- Submit your repository to GitHub.

- Repository must also contain:

  - Commit history showing incremental progress of the application development.

  - README.md with:

    - Project overview

    - Setup instructions

    - Third-party API documentation link

    - .gitignore for sensitive data

- **Do NOT** include:

  - Production credentials

  - Proprietary business logic

- Send us an email with your github repo link once you are done.

Evaluation Criteria

- Code organization

- Good practices

- Error handling

- Database usage

- Documentation

- Extra features beyond requirements.

- Git practices (meaningful commits, branching strategy)

Note: Creativity in API/data selection is encouraged – use public datasets (e.g., NASA APIs, GitHub API) to avoid commercial sensitivities.