

Coding assignment

This development test is used as part of the selection process for Development Engineers. You are requested to develop a simple API that covers all the requirements listed below. To have an indication of the criteria that will be used to judge your submission, all the following are considered as metrics of good development:

- Correctness of the implementation
- Decent test coverage
- Code cleanliness
- Efficiency of the solution
- Careful choice of tools and data formats
- Use of production-ready approaches

The assignment is meant to be completed in a period of 3 hours. After that, a 15 minute presentation + Q&A session will follow where the candidate will present the assignment. You are free to choose any library.

Assignment 1: Currency Conversion Service

Objective:

Build a backend service that allows easy conversion between foreign currencies

- Fixer.io is a simple and lightweight API for and historical foreign exchange (forex) rates.
- [Sign Up Free](#)
- The public API documentation can be found here <https://fixer.io/documentation>

Requirements:

- The service should be a RESTful API built with a framework of your choice.
- The API will have 3 endpoints:
 1. Get the live rate of a single currency
 2. Converting an amount to euros (e.g. 100 dollars to euros)
 3. Converting an amount from euros (e.g. 100 euro to dollars)

Bonus (optional):

- The API should keep track of rates updates at fixer.io, without calling the API each time
- Implement basic error handling to manage potential issues with API requests, such as timeouts or invalid responses. Return appropriate HTTP status codes (e.g., 200 for success, 404 for not found, 500 for server errors).

Documentation:

- Provide a brief README explaining how to set up and run your service, including any dependencies.
- Document the API endpoint with example requests and responses.

Testing:

- Write basic tests for your service to ensure it works as expected.
- Tests should cover at least the core functionality of the endpoints.

Guidance:

- **Time Management:** Focus on building a simple, functional service within the time limit. It's okay if you can't implement the bonus features—focus on delivering the core functionality.
- **Framework Choice:** Use a framework or tools you're comfortable with to save time.

- **Handling Data:** Efficiently handle API responses and ensure that the logic works as expected.

Expected Deliverables:

1. A RESTful API with 3 endpoints
2. A README file with basic (setup) instructions.
3. Basic tests for the API.
4. Clean and well-documented code.

This assignment tests your ability to work with external APIs, aggregate data from multiple sources, and build a RESTful service that provides a unified response. The optional features offer an additional challenge for those who want to demonstrate more advanced skills. In order to assess the assignment, the following criteria will be used:

- Project structure and organization
- Justification of tech stack used
- Client functional suitability
- Code maintainability
- Communication skills

Assignment 2: Front-end

Objective:

Create a simple front-end for the API

Requirements:

- A user can view the EURO conversion rate for a certain currency
- A user can enter an amount and call the API to do a conversion

Guidance:

- **Simple Implementation:** You can use any front-end technology (e.g. <https://open-wc.org/guides/developing-components/getting-started/>)

Expected Deliverables:

1. A front-end that can be viewed in a browser and can do a simple conversion using the API