

Senior Backend Developer (Python) - Code Challenge

We are a global digital marketing service provider who specializes in affiliate marketing & publishing. We are digital natives, data obsessed and focused on measurable outcomes. We are proud of our people and we have some of the most talented individuals you'll ever meet working with us. Our values are at the heart of all decisions we make, from business goals to people initiatives and they have helped us to develop a world class team of experts, we are proud of. We've grown considerably in the past months and continually focus on growth via our global talent – you would be joining us at the most exciting time in our history.

Code challenge designed to evaluate technical skills of our [Senior Backend Developer \(Python\) candidates](#).

- [Senior Backend Developer \(Python\) - Code Challenge](#)
 - [Introduction](#)
 - [The Challenge](#)
 - [Scenario](#)
 - [Solution](#)
 - [Hints/Advices](#)
 - [Nice to have](#)
 - [Submission Guidelines](#)
 - [License and Software Information](#)
 - [License](#)
 - [ATTENTION](#)

Introduction

This challenge helps us to understand how you think through technical problems and how you might architect and implement solutions to those problems. It also helps us to assess your comfort level with various technologies and determine which projects you might start on, if you join our team.

We understand that everyone has existing personal and work commitments, so this challenge is not intended to use up a lot of your time. As such, there is no specific deadline or pre-set time allotment. Send us your submission once you're comfortable with it. Some people prefer to dive right in and submit something right away, and others prefer spread out the time over a week or more - whatever works best for you and your schedule! Please note however, that our hiring process moves fast, so **please send your submission as soon as possible**.

If you have any questions or require any clarification, please [email](#) us and we'll get back to you as soon as we can.

The Challenge

Scenario

Our *fake client*, **ACME Sports**, wants to develop a RESTful API to return a list of NFL events in JSON format. The events data must be dynamic because the process will pull the NFL event data from a remote API that is frequently updated.

The remote API has been described using OpenAPI. You can see the OpenAPI specification opening this [file](#).

In summary, the API provides us two endpoints for retrieving information events.

You can assume the remote API was well-documented so it will be returned the data in the exact same format, with the exact same field names and datatypes.

Solution

For figuring out this challenge, you must implement the expected API, which has been described using OpenAPI as well, and you can see in [file](#)

We are intentionally forgetting details about program languages or framework in order to write the solution, because we want that you can use the framework/object-oriented program language you are very familiar. For example, we know different frameworks that you could use for doing this challenge:

- **Python:** FastAPI, Django, Flask, Tornado, ...
- **Java / Kotlin:** Spring Boot / Spring Framework, Quarkus, Micronauts, Microprofile, ...
- **NodeJS:** Nest.JS, Express.JS, Fastify, ...
- **.NET:** .NET core, .NET framework 4.x, 5.x...

Of course, a working implementation in code will be the best indicator of your abilities, however, if you prefer, you can describe how you would solve the problem in a PDF doc, you can attach drawings, doodles or screenshots, or you can write pseudo code - it's up to you! Include any third-party libraries or frameworks you want to save time. **We like saving time too!**

Hints/Advices

- You should create a mock/stub of this [remote API](#) for typing your code. You can use [Postman](#), [Microcks](#) or whatever you would rather with. Anyway, we have provided to you a small mock API, that you can run executing the next command in the current directory:

```
docker compose build -d
```

Once you have executed the command, the port **9000** will be busy in your laptop, and you can call the two endpoints that is going to be exposed by the Mock API:

- **HTTP GET** <http://localhost:9000/NFL/scoreboard>
- **HTTP GET** <http://localhost:9000/NFL/team-rankings>

So, yup! Let us go ahead to use this mock in order to type your code! 😊

Nice to have

Below points are not mandatory, but we would like to see in your code 😊 :

- You can include in your implementation some **architecture diagrams** in order to show us that you technically understood the challenge. In the case you are going to paint diagrams, you can use [Draw.io](#) or [C4 models](#) or whatever you prefer.
- Unit / integrations tests in order to check the APIs works fine! Here, you can use the Test Framework you are very familiar. Let us provide us some known frameworks (*):
 - **Java / Kotlin**: JUnit, TestNG
 - **NodeJS**: Jest, Mocha, Jasmine
 - **Python**: Pytest, ...

NOTE: *If you dedice to implement unit tests, please, keep in mind you would need to mock objets for ensuring really are unit tests.*

NOTE: *If you dedice to implement integration tests, you can use 3rd party tools such as [\[Testcontainers\]](#).

Submission Guidelines

We strongly recommend to use [Git](#) in order to do the challenge, so maybe the best way for sending your submission would be creating a fork of this project, and pushing your code in a public/private [GitHub](#) repository. If you would dedice to push your code into a private Git repository, you must give right access to such repository to the [medidachallenges](#) GitHub username.

However, you can email to challenges@medida.com your submission. Please, do not attach any file in the email due to our SPAM restrictions. So, it would be better to send your code using [WeTransfer](#) link or something like this.

Regardless what would it be the way you will choose, please use Git! We would like to see though the [Git Log](#) the development process you have been done.

License and Software Information

© Medida

Medida publishes this software and accompanied documentation (if any) subject to the terms of the MIT license with the aim of opening some challenge wordings to the community. You will find a copy of the MIT license in the root folder of this package. All rights not explicitly granted to you under the MIT license remain the sole and exclusive property of Medida.

NOTICE: The software has been designed solely for the purpose of analyzing the code quality by checking the coding guidelines. The software is NOT designed, tested or verified for productive use whatsoever, nor or for any use related to high risk environments, such as health care, highly or fully autonomous driving, power plants, or other critical infrastructures or services.

If you want to contact Medida related to the software, you can mail us at challenges@medida.com.

License

[MIT](#)

ATTENTION

Do **NOT** try to PUSH direct to THIS repository!