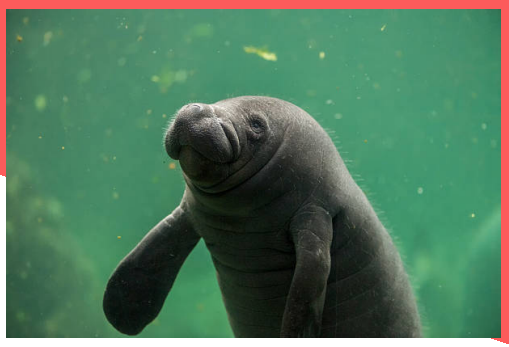
CYBERNETICA

Internship Programme 2023
Project Manatee



Introduction

You have been given a test assignment for a hiring management platform called Manatee, which is currently in the prototype stage. The platform allows recruiters to initiate a new hiring process with candidates and manage the process until successful onboarding. Your task is to implement a new feature for scheduling interviews, specifying the interview time, interviewer name, and interview type. The interview types include informal, technical, behavioural, and final.

Requirements

While you are permitted to add dependencies to the project to meet your requirements, please do not remove or modify the provided dependencies. Any added dependencies must be justified and documented in the README.md file.

You are supposed to follow the OpenAPI standard. We initialised the repository with the API specification *(manatee/spec/manatee-api.yml)* and DTO-generating capabilities.

For more information on the OpenAPI standard, please visit:
https://swagger.io/specification/

You are allowed to make any changes in the source code.

CYBERNETICA

Tasks

1. Create an endpoint specification (manatee/spec/manatee-api.yml) for scheduling interviews.
2. Design a database entity to store information related to interviews.
3. Implement the scheduling interview endpoint and write a state transition for it.
4. Develop at least one unit test for the newly added endpoint and another for validating the state transition.
5. Modify the GET /applications endpoint to retrieve relevant information about scheduled interviews.
6. Seed the in-memory database during runtime with 4-6 different applications. Some applications must contain scheduled interview data. You can use any tooling for this task.

Bonus Task: Create a simple React client that consumes the GET /applications endpoint. The desired outcome is a Single-Page Application (SPA) that includes a single well-designed table, retrieving the backend applications. Submit the client as a separate repository using a similar format.

Interesting fact! You may use the same generator to create any JavaScript/TypeScript-based client. This means that all you need to do is design a user-friendly interface using a library of your choice.

CYBERNETICA

Submission

The assignment must be completed within a single workday (8 person-hours). The workload can be spread over multiple calendar days. Once the designated time has expired, you must submit your solution. While it does not need to be perfect, most functionality should be operational. Plan your workload accordingly so that you can begin wrapping up your work as the deadline approaches to ensure that you have a consistent and functional solution.

The code repository should contain working code, OpenAPI specification and a summary in the README.md file.

In summary, describe your overall experience with the topic, what you learned, and any technical challenges you encountered. Your answer should be between 50-100 words.

The deadline for submitting the assignment is **April 14th**.

To submit your solution, create a private git repository in GitHub, GitLab, or BitBucket and give read access to internship@cyber.ee. Be sure to ask if you have any questions.

Grading

Don't feel guilty when you don't finish some tasks. You can still submit the assignment for grading. The best submissions are as follows:

- The execution order is structured logically.
- Demonstrates familiarity with OOP design principles.
- Minimises unnecessary repetition.
- Classes and methods are small, predictable, and maintainable.
- Includes relevant, consistent, and comprehensive unit tests.
- Variable and method names are meaningful and descriptive.
- Provides a summary in the README.md file.
- Possesses general knowledge of React (for bonus task submissions).

Disclaimer: This test assignment aims to get a general understanding of an applicant's skills. Cybernetica will not use the results of this task for commercial purposes and will not compensate for the time spent on this assignment.