

## Introduction

For this assignment, you are required to create a book inventory application using **ANAPIOFICEANDFIRE** API, in ReactJS (Typescript).

## Ground Rules

- Use ReactJS and Typescript.
- Your solution should be submitted as a link to a GitHub repository containing a README file written in English.
- The repository should have public access.
- We should be able to verify your submission by cloning your repository, configuring the environment, and running the application. Please document all details about the project setup in the README file.

## Requirements to Implement

Your solution should provide an implementation for each of the following requirements. The tasks will be tested by running your code on the web browser.

### Requirement 1

#### Read

Use the Ice and Fire API to enable the user to see a list of books available on the platform.

**GET** <https://www.anapioficeandfire.com/api/books>

Your List UI should have the following.

- Publisher
- Name
- ISBN
- Authors
- End Date

In your User Experience do implement infinite loading by using the pagination function of the **Ice and Fire API**

**Please use appropriate components when rendering your book page. Don't write everything in a single component. Try to make your code readable.**

### Requirement 2

#### Read

We want to be able to search for a book using the following parameters.

- Publisher
- Name

- ISBN
- Authors
- End Date
- Characters Name
- Character Culture

**GET** <https://www.anapioficeandfire.com/api/characters>

Use the API above to fetch a list of characters currently.

The goal of this search is to see how you optimize your system by handling the API calls between the two endpoints i.e (book endpoint and character endpoint). Please try to make it as efficient as possible.

## Evaluation Instructions

When evaluating the submitted solution, we will put attention to the following:

- **Creative User Interface:** We want to see how creative your user interface is. Please take into perspective UI and UX.
- **Functional solution:** we will test the API endpoints and see if they return the expected results based on the provided input.
  - Make sure to have things well documented in the README, so we can easily set up the project and test it. If we have to contact you for more information, that will count against your score.
- **Good code design:** we put special attention to the design of the code. Make sure your code is clean and readable. It complies with the SOLID design principles and makes use of design patterns where it makes sense.

**We appreciate your effort in completing this assessment and encourage everyone to strive to finish the task before the due date. However, if you are unable to meet the deadline, we strongly advise you to submit your work regardless.**