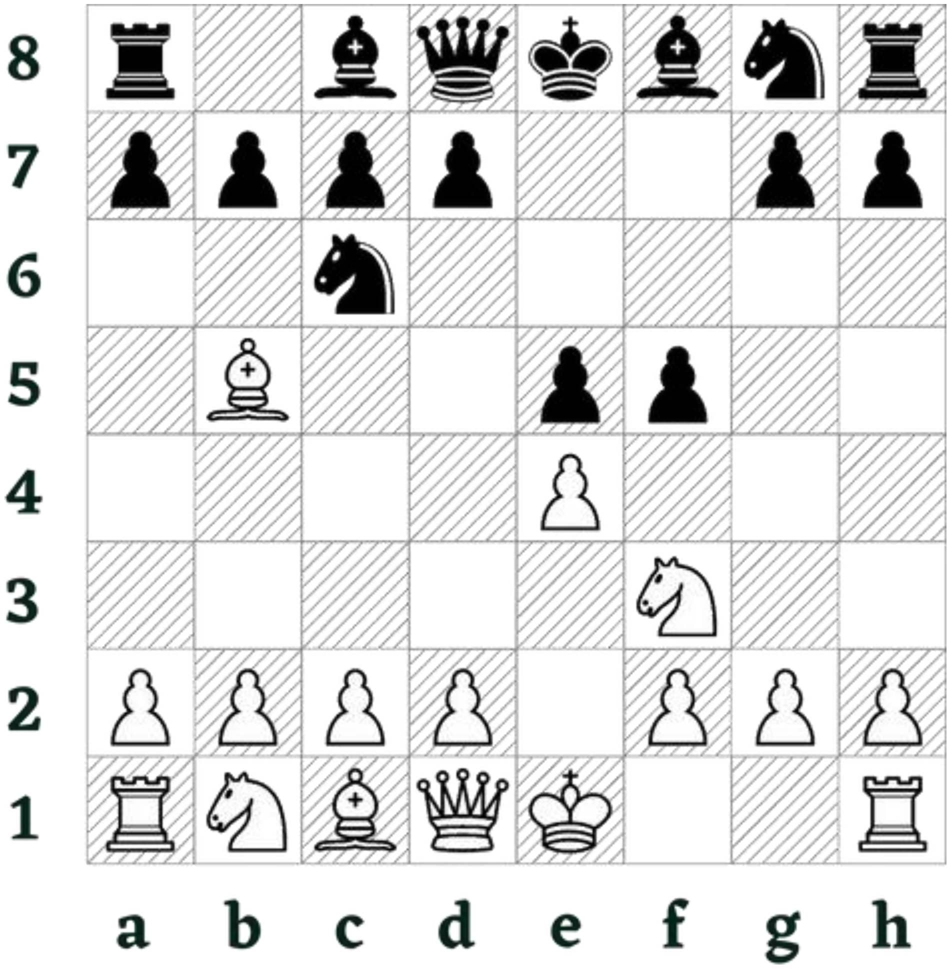
JUMBO MANA –TECHNICAL TEST

BACK-END

1 – CAPTCHA

The purpose of this exercise is to create a server capable of generating random chess position where black and white are equals.



Exemple of a chess game

This exercise is an improvement of the 960 chess game by Bobby Fischer, not limited to the initial position. You are free to define your own rules to make this variation of the 960 chess game more fun.

1.1 BACK-END

To create this service, you will develop an API on Python with the FastAPI framework. The back-end needs to generate a chess position and determine if the game is equal. You will use stockfish to determinate if the position is equal.

1.2 FRONT-END

A simple front-end will be developing to interact with the back-end and show random chess positions.

1.3 - Livrables

You must deliver your project on Github or Bitbucket.

Your code must to be documented and your architecture must to respect the object-oriented programming.

Your repository would have a ReadMe including:

- Steps to deploy and reproduce your results ;
- Your architecture presentation ;
- The reference to code or project you use for inspiration ;
- All information you think to be mentioned.

This technical test aims to assess you on four criteria:

- Your ability to develop your own API ;
- Your skill in web programming ;
- Your notions in software architecture for productive deployment ;
- Your autonomy and strength of proposal