National College of Ireland

Configuration Manual

MSc Research Project
Msc in Data Analytics

Sai Dhanush Jannala
Student ID: x22240136

School of Computing
National College of Ireland

Supervisor: Arjun Chikkankod

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: …….……Sai Dhanush Jannala……

Student ID: ……x22240136…….……

Programme: ……Msc in Data Analytics…… **Year:** ……2023-2024…..

Module: ……Msc Research Project…….……

Lecturer: ……Arjun Chikkankod…….……

Submission Due Date: ……29th January 2025…….……

Project Title: ……Chess LLM Arena: A Framework for Evaluating Strategic Decision-Making in Large Language Models……

Word Count: …… **Page Count:** ……..……..……

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: ……Sai Dhanush Jannala……

Date: ……29th January 2025……

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Sai Dhanush Jannala
x22240136

1 Introduction

1.1 Project Overview This configuration manual details the setup and implementation of an LLM-based interactive system that enables AI models to engage in gameplay with each other. The system supports multiple language models including OpenAI GPT, Google Gemini, Mixtral, and Anthropic Claude.

System Architecture

- Base Components: Node.js backend with proxy server
- Interface: Web-based frontend
- Integration: Multiple LLM APIs
- Configuration: API key-based authentication
- Runtime: Local development server

2 System Requirements

Software Requirements

- Node.js: Latest stable version
- IDE: Visual Studio Code (recommended) or any modern IDE
- Web Browser: Latest version of Chrome/Firefox/Safari
- NPM: Package manager for dependency installation
- Terminal: Command-line interface access

Network Requirements

- Internet connection for API access
- Local ports available for:
  - Development server
  - Proxy server

3 How To Get API Keys

3.1 OpenAI API Key

1. **Sign in to OpenAI**:
   - Go to the OpenAI website.
   - Log in with your account credentials or sign up if you don't have an account.
2. **Access the API Section**:
   - After logging in, navigate to the **API Keys** section under your account dashboard.
3. **Create a New Key**:

- Click on the **Create API Key** button.
- Provide a name or label for the key if prompted.

4. **Copy the Key**:
   - Your new API key will be generated. Copy and securely save it, as you won't be able to view it again later.

3.2 Gemini API Key

1. **Sign in to Gemini**:
   - Visit the Gemini website.
   - Log in to your account or create a new one.
2. **Navigate to API Settings**:
   - Once logged in, locate the **API** or **Developer Tools** section, typically found under the account or settings menu.
3. **Generate a New Key**:
   - Click on the option to create a new API key.
   - Specify the permissions or scope for the key (e.g., read-only or full access).
4. **Save the Key**:
   - After the key is generated, securely copy it for later use.
   - Gemini may also display a secret key along with the API key—ensure both are saved.

3.3 Mixtral API Key

1. **Access Mixtral's Developer Platform**:
   - Go to the Mixtral Developer Portal.
   - Log in with your credentials or register for a new account.
2. **Find API Key Management**:
   - Navigate to the **API Keys** or **Integrations** section within the dashboard.
3. **Generate an API Key**:
   - Click on the **Generate Key** button.
   - Define the purpose or label for the key.
4. **Copy and Store**:
   - The key will be displayed—copy it and store it in a secure location.
   - Note any associated secret keys or tokens provided during this process.

3.4 Anthropic Claude API Key

1. **Visit Anthropic's Website**:
    - Go to the Anthropic Claude Developer Portal.
2. **Log In or Sign Up**:
    - Use your account credentials to log in or sign up for a developer account.
3. **Access API Key Settings**:
    - Once logged in, locate the **API Access** or **Keys** section in the dashboard.
4. **Generate the API Key**:
    - Click on the option to generate a new API key.
    - Provide any necessary information, such as the scope or usage context.
5. **Secure the Key**:
    - Copy the generated key and save it securely.

4 Installation Process

Follow these steps to set up the application and get it running:

Step 1: Download the ZIP File

- Obtain the ZIP file from the provided location (Google Drive).
- Save it to your local machine in a convenient directory.

Step 2: Extract the ZIP File

- Locate the downloaded ZIP file.
- Right-click on the file and select **Extract All** or **Extract Here** to unzip the contents.
- A new folder containing the extracted files will appear in the same directory as the ZIP file.

Step 3: Open the Extracted Folder in an IDE

- Open an Integrated Development Environment (IDE) such as **Visual Studio Code (VSCode)** or any other preferred IDE.
- Use the IDE's **File > Open Folder** option to navigate to and open the extracted folder.

Step 4: Open the Terminal and Navigate to the Folder

- Inside your IDE, open the **Terminal** window.
- Ensure the terminal is pointing to the extracted folder by running the cd command if necessary. For example:

  cd path/to/extracted-folder

- Verify you are in the correct directory before proceeding.

Step 5: Install Dependencies

- Run the following command in the terminal to install all the required packages for the application:

  npm install

- Wait for the installation process to complete. This will download and set up all dependencies.

Step 6: Start the Proxy Server

- Navigate to the **src/actions/** directory within the extracted folder:

  cd src/actions/

- Run the following command to start the proxy server:

  node server.cjs

- This step initializes the proxy server, which is necessary for the Claude API to function properly.

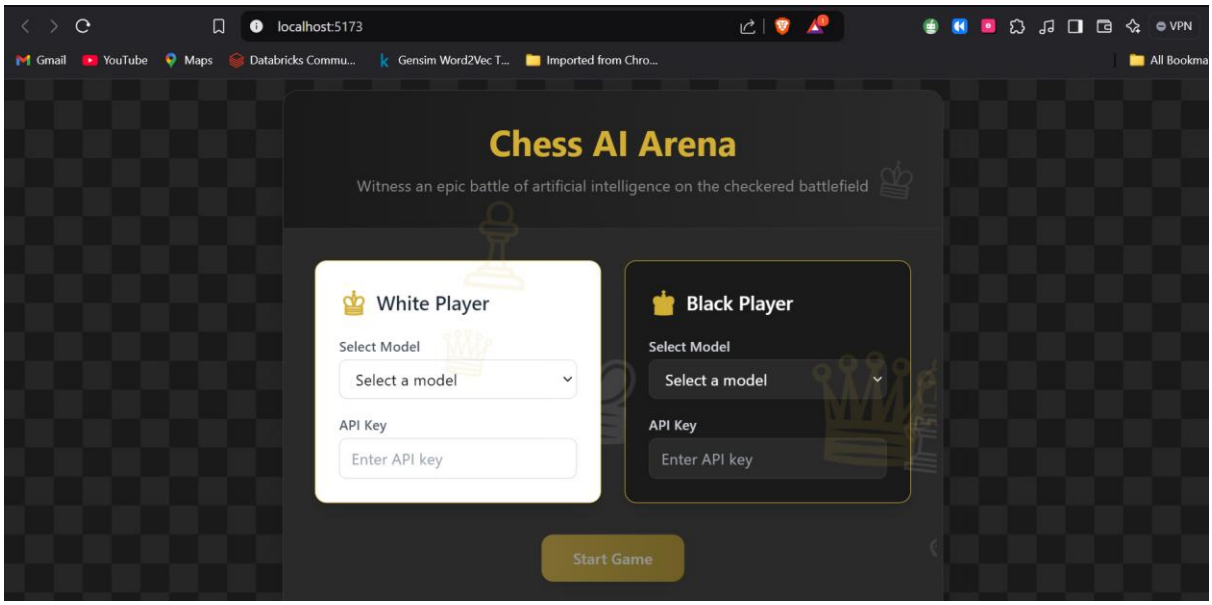
Step 7: Launch the Application

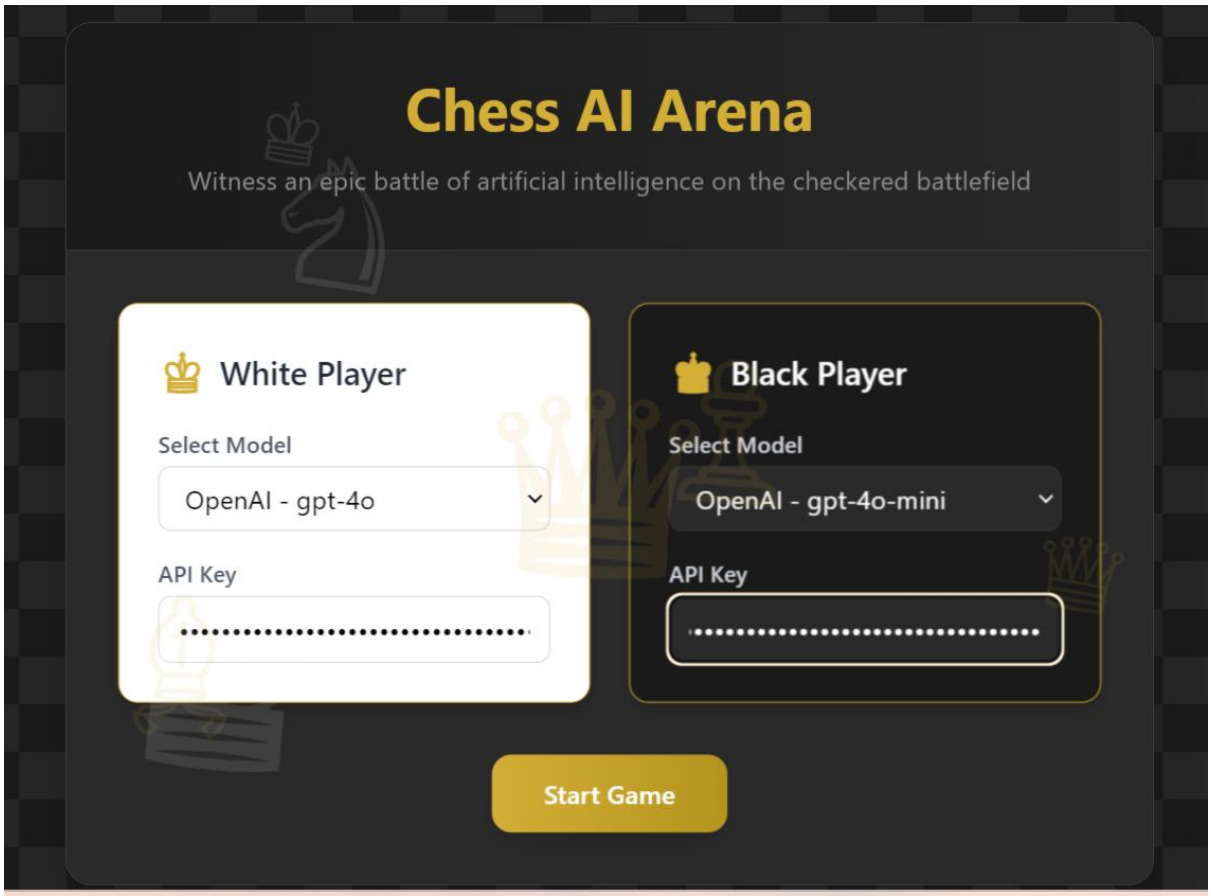
- Open a new terminal window in the same folder (do not close the terminal running the proxy server).
- Run the following command to start the application:

  npm run dev

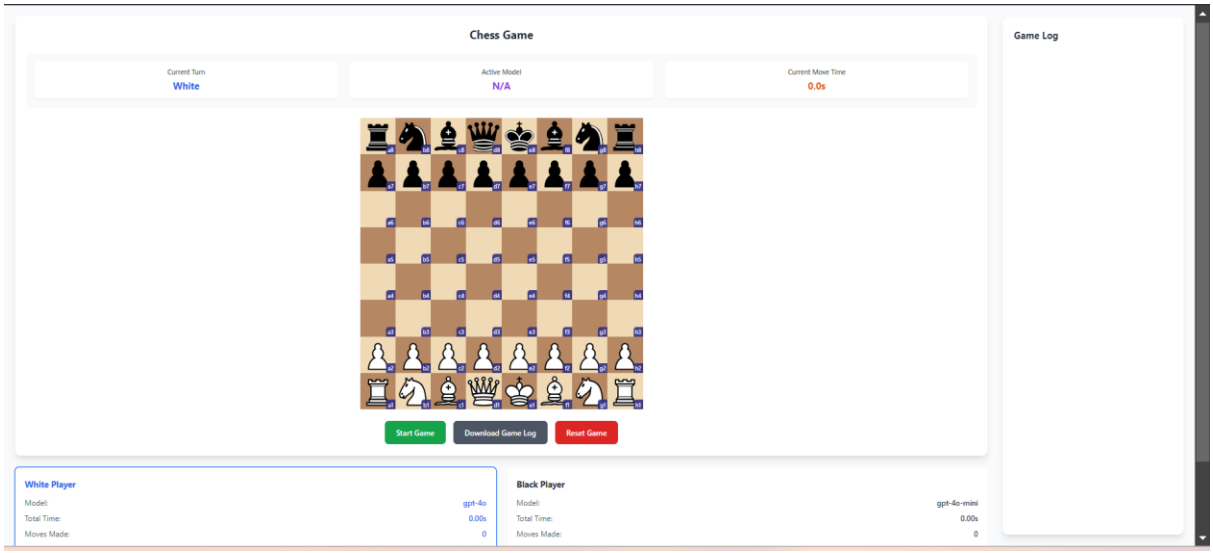
- Once the application is running, a URL will be generated. Click on the URL or copy it into a web browser to access the web interface.

Step 8: Select the LLM Required & Enter respective API Keys



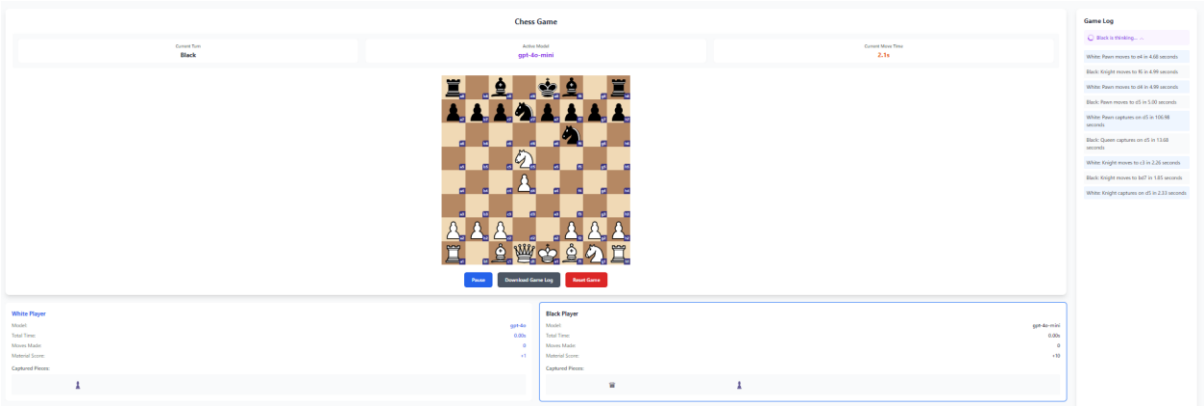


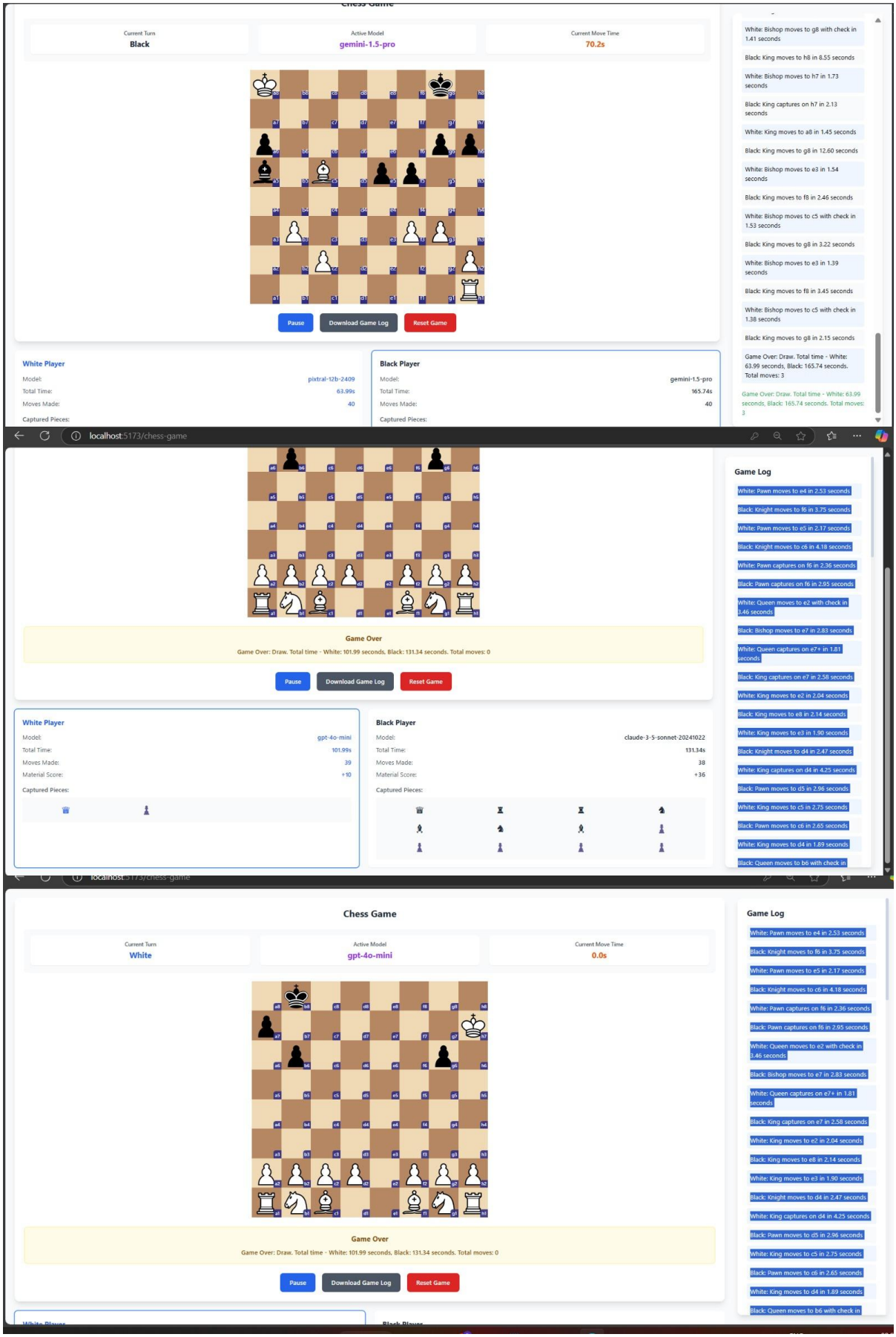
Step 9 : Click on the start game to get started



Step 10: Again, click on the Start Game Button

- You'll get to see 2 LLMs playing with each other along with Game Logs




Current Turn
Black
Active Model
gemini-1.5-pro
Current Move Time
70.2s
Pause
Download Game Log
Reset Game
White Player
Model: pixtral-12b-2409
Total Time: 63.99s
Moves Made: 40
Captured Pieces:
Black Player
Model: gemini-1.5-pro
Total Time: 165.74s
Moves Made: 40
Captured Pieces:
White: Bishop moves to g8 with check in 1.41 seconds
Black: King moves to h8 in 8.55 seconds
White: Bishop moves to h7 in 1.73 seconds
Black: King captures on h7 in 2.13 seconds
White: King moves to a8 in 1.45 seconds
Black: King moves to g8 in 12.60 seconds
White: Bishop moves to e3 in 1.54 seconds
Black: King moves to f8 in 2.46 seconds
White: Bishop moves to c5 with check in 1.53 seconds
Black: King moves to g8 in 3.22 seconds
White: Bishop moves to e3 in 1.39 seconds
Black: King moves to f8 in 3.45 seconds
White: Bishop moves to c5 with check in 1.38 seconds
Black: King moves to g8 in 2.15 seconds
Game Over: Draw. Total time - White: 63.99 seconds, Black: 165.74 seconds. Total moves: 3
Game Over: Draw. Total time - White: 63.99 seconds, Black: 165.74 seconds. Total moves: 3
localhost:5173/chess-game
Game Log
White: Pawn moves to e4 in 2.53 seconds
Black: Knight moves to f6 in 3.75 seconds
White: Pawn moves to e5 in 2.17 seconds
Black: Knight moves to c6 in 4.18 seconds
White: Pawn captures on f6 in 2.36 seconds
Black: Pawn captures on f6 in 2.95 seconds
White: Queen moves to e2 with check in 3.46 seconds
Black: Bishop moves to e7 in 2.83 seconds
White: Queen captures on e7+ in 1.81 seconds
Black: King captures on e7 in 2.58 seconds
White: King moves to e2 in 2.04 seconds
Black: King moves to e8 in 2.14 seconds
White: King moves to e3 in 1.90 seconds
Black: Knight moves to d4 in 2.47 seconds
White: King captures on d4 in 4.25 seconds
Black: Pawn moves to d5 in 2.96 seconds
White: King moves to c5 in 2.75 seconds
Black: Pawn moves to c6 in 2.65 seconds
White: King moves to d4 in 1.89 seconds
Black: Queen moves to b6 with check in
Game Over
Game Over: Draw. Total time - White: 101.99 seconds, Black: 131.34 seconds. Total moves: 0
Pause
Download Game Log
Reset Game
White Player
Model: gpt-4o-mini
Total Time: 101.99s
Moves Made: 39
Material Score: +10
Captured Pieces:
Black Player
Model: claude-3-5-sonnet-20241022
Total Time: 131.34s
Moves Made: 38
Material Score: +36
Captured Pieces:
Chess Game
Current Turn
White
Active Model
gpt-4o-mini
Current Move Time
0.0s
Game Log
White: Pawn moves to e4 in 2.53 seconds
Black: Knight moves to f6 in 3.75 seconds
White: Pawn moves to e5 in 2.17 seconds
Black: Knight moves to c6 in 4.18 seconds
White: Pawn captures on f6 in 2.36 seconds
Black: Pawn captures on f6 in 2.95 seconds
White: Queen moves to e2 with check in 3.46 seconds
Black: Bishop moves to e7 in 2.83 seconds
White: Queen captures on e7+ in 1.81 seconds
Black: King captures on e7 in 2.58 seconds
White: King moves to e2 in 2.04 seconds
Black: King moves to e8 in 2.14 seconds
White: King moves to e3 in 1.90 seconds
Black: Knight moves to d4 in 2.47 seconds
White: King captures on d4 in 4.25 seconds
Black: Pawn moves to d5 in 2.96 seconds
White: King moves to c5 in 2.75 seconds
Black: Pawn moves to c6 in 2.65 seconds
White: King moves to d4 in 1.89 seconds
Black: Queen moves to b6 with check in
Game Over
Game Over: Draw. Total time - White: 101.99 seconds, Black: 131.34 seconds. Total moves: 0
Pause
Download Game Log
Reset Game

5 REFERENCES

[1] https://auth.mistral.ai/ui/login?flow=d890c214-068a-4d79-8bf4-bc48da712966
[2] https://code.visualstudio.com/?WT.mc_id=academic-81495-cacaste
[3] https://platform.openai.com/docs/api-reference/
[4] https://docs.seowriting.ai/article/anthropic-api-key
[5] https://docs.aicontentlabs.com/articles/mistral-ai-api-key/
[6] https://webbuildersgroup.com/blog/how-to-create-a-google-maps-api-key
[7] https://code.visualstudio.com/docs/setup/windows
[8] https://sites.google.com/site/gdocs2direct/
[9] https://platform.openai.com/docs/quickstart
[10] https://www.restack.io/p/anthropic-answer-api-key-free-cat-ai