**Full Documentation for FastAPI Project with Docker**

**1. Project Overview**

This project is a FastAPI web application that predicts the type of Olympic medal (Gold, Silver, or Bronze) based on the provided input features such as discipline and event. The application is containerized using Docker for easy deployment and portability.

**2. Problem Faced**

**Issue:**

When trying to access the FastAPI application through http://0.0.0.0:8000/, the browser returned the error ERR\_ADDRESS\_INVALID.

**Root Cause:**

The IP address 0.0.0.0 is a special address used to bind the application to all available interfaces but is not meant to be accessed directly from a browser.

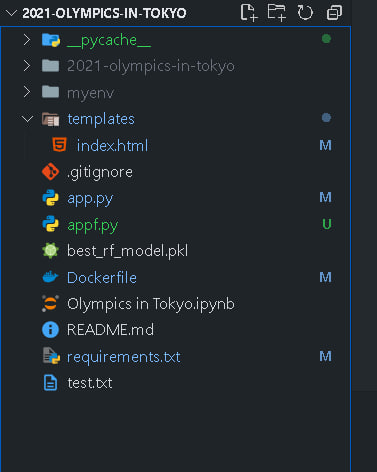
**Solution:**

Instead of using 0.0.0.0, you should use localhost or 127.0.0.1 to access the application in your browser:

http://localhost:8000/

or

<http://127.0.0.1:8000/>



**3. Project Structure**

The folder structure of the project is as follows:

**4. Virtual Environment Setup**

1. **Activate virtual environment** (assuming you're using one like myenv):

myenv\Scripts\activate # For Windows



This command generates a requirements.txt file that lists all the packages and their versions currently installed in your environment. It is necessary for installing dependencies inside the Docker container.

**A screenshot of a computer

Description automatically generated5. FastAPI Application Code**

The FastAPI application (appf.py) handles predictions using a pre-trained Random Forest model. The application serves an HTML form where users can input the discipline and event IDs, and it returns a prediction of the medal type.

A screenshot of a computer

Description automatically generated**6. Docker Setup**

Docker is used to containerize the FastAPI application, ensuring it can run consistently across different environments.

A screenshot of a computer program

Description automatically generated**7. Running the Application**

**Steps:**

**Build the Docker image:**

**docker build -t fastapi-app .**

A screenshot of a computer

Description automatically generatedA screenshot of a computer program

Description automatically generatedA screenshot of a computer

Description automatically generated

**A screenshot of a computer

Description automatically generated**

**8. Problem Solved**

**Initial Issue:**

Attempting to access the application via http://0.0.0.0:8000/ resulted in an error (ERR\_ADDRESS\_INVALID).

**Solution:**

Accessing the app through http://localhost:8000/ or http://127.0.0.1:8000/ solved the problem. The 0.0.0.0 address is for binding to all interfaces and not meant for direct access in a browser.

**9. Conclusion**

The project demonstrates how to containerize a FastAPI application using Docker and use machine learning models for predictions. The issue of accessing the application was resolved by understanding the purpose of the 0.0.0.0 address and switching to localhost for browsing.

This setup ensures that the application can be deployed on any system using Docker, with consistent behavior across environments.