

Configuration Manual

MSc

Artificial Intelligence focuses on business

Juan Alejandro Casillas Tello

Student ID: 23355328

School

National College of Ireland

Supervisor: Haque R

National College of Ireland
MSc Project Submission Sheet

Student Name: Juan Alejandro Casillas Tello

Student ID: 23355328

Programme: Master of Science in Artificial Intelligence Focusses on business **Year:** 2024-25

Module: Practicum Part 2

Supervisor: Haque q

Submission Due Date: 15th September 2025

Project Title: Research Project Configuration manual

Word Count: 772 **Page Count** 10

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: Juan
Alejandro

Date: 11-08-24

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

Juan Alejandro Casillas Tello

Student ID: 23355328

Table of Contents

1	Introduction.....	4
2	Experimental Setup	4
3	Technologies and Software Used for Implementation.....	4
4	Implementation.....	5

1 Introduction

The configuration manual outlines the tools and technologies utilized during the research implementation. Section 2 covers the experimental setup in detail. In Section 3, the technologies and software tools used are discussed. Section 4 provides a comprehensive, step-by-step guide for setting up the software tools, detailing the implementation process from installing and configuring Suricata, creating custom rules, generating attacks from Kali Linux, setting up email alerts, saving logs to Google Drive, and analysing and visualizing the attack patterns. Lastly, Section 5 contains the references for the software guide.

2 Experimental Setup

The experiment was designed on a personal computer, which was set up specifically for this purpose.

- **Hardware Specifications:**

Processor: 12th Gen Intel(R) Core(TM) i7-1255U (1.70 GHz).

Ram: 16.0 GB

System type: 64 bit

Operating System: Windows 10, Ubuntu 18.04, Kali Linux

Experimental Setup: Windows 10, Python 3.10, VS Code, PythonAnywhere (as the deployment environment), Anaconda for virtual environment management, and FFmpeg for optional media processing.

3 Technologies and Software Used for Implementation

- **Software Used:** Visual Studio Code (VS Code), PythonAnywhere, Anaconda, FFmpeg

Visual Studio Code (VS Code) is a free, open-source code editor developed by Microsoft. It offers helps in debugging, syntax highlighting, intelligent code completion, integrated terminal access, and a vast library of extensions. Making it highly customizable and versatile for various programming languages and frameworks, like Python and Flask

PythonAnywhere is a cloud-based hosting and execution environment for Python applications. It allows developers to deploy web applications without managing physical servers or complex infrastructure, providing an accessible and reliable platform for lightweight applications.

Anaconda is a distribution of Python and R that simplifies package management and deployment. It is particularly useful for creating isolated virtual environments to manage dependencies for machine learning and AI projects.

FFmpeg is a powerful, open-source multimedia framework used to process, convert, and stream audio and video. .

4 Implementation

Step 1: Install all the systems and configurations before mentioned.

Step 2: Run the installer.

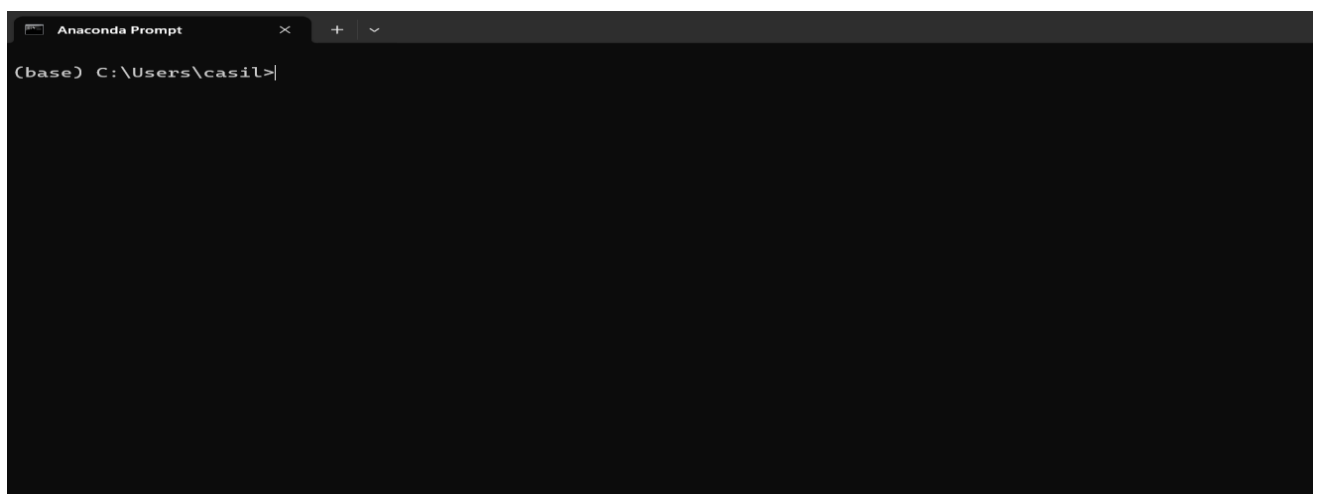


Figure (1): anaconda prompt

Step 3: : Set Up Virtual Environment

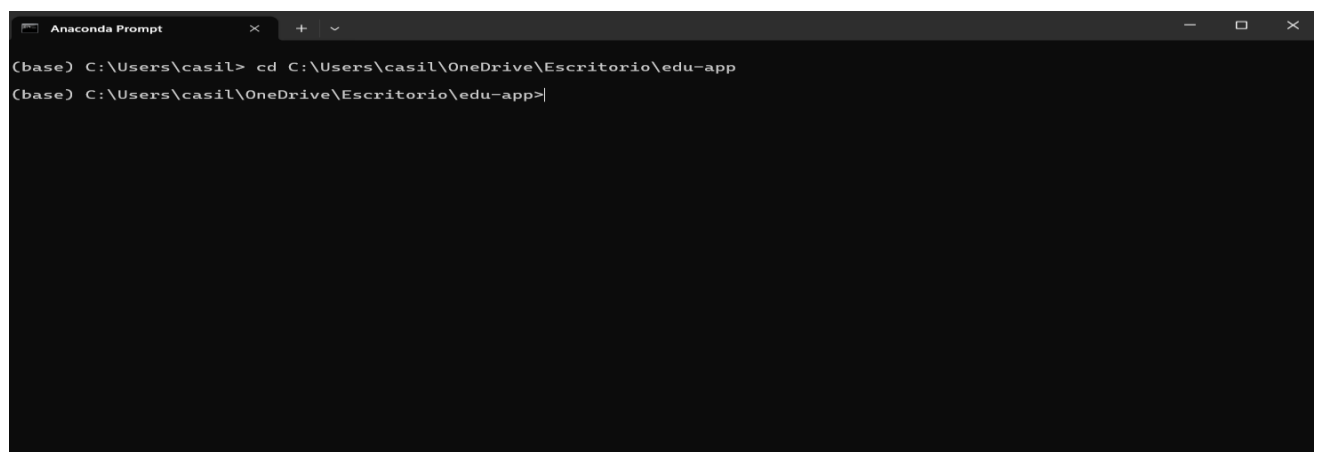
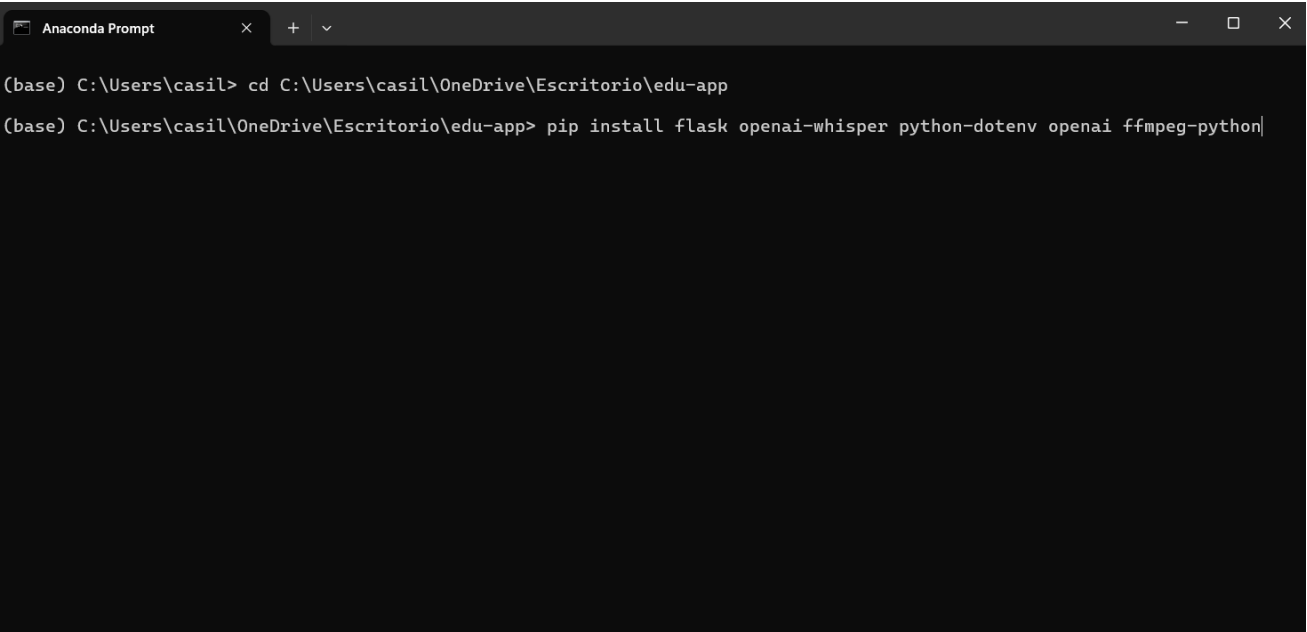


Figure (2): new environment

Step 4: install libraries

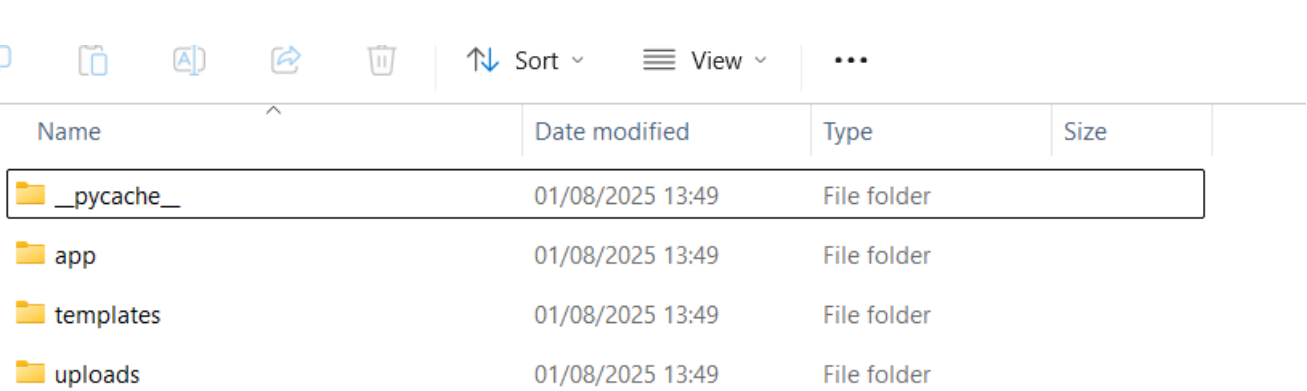


The screenshot shows an Anaconda Prompt window with a dark background. The terminal text is as follows:

```
(base) C:\Users\casil> cd C:\Users\casil\OneDrive\Escritorio\edu-app
(base) C:\Users\casil\OneDrive\Escritorio\edu-app> pip install flask openai-whisper python-dotenv openai ffmpeg-python
```

Figure (3): install libraries

Step 5: create folder structure



The screenshot shows a Windows File Explorer window with a light blue header bar. The address bar shows the path 'C:\Users\casil\OneDrive\Escritorio\edu-app'. The toolbar includes icons for back, forward, and search, along with 'Sort' and 'View' dropdown menus. The main area displays a table of files and folders.

Name	Date modified	Type	Size
__pycache__	01/08/2025 13:49	File folder	
app	01/08/2025 13:49	File folder	
templates	01/08/2025 13:49	File folder	
uploads	01/08/2025 13:49	File folder	

Figure (4): Folders

Step 6: install ffmpeg



Figure (5): ffmpeg page download

Step 7 get API key

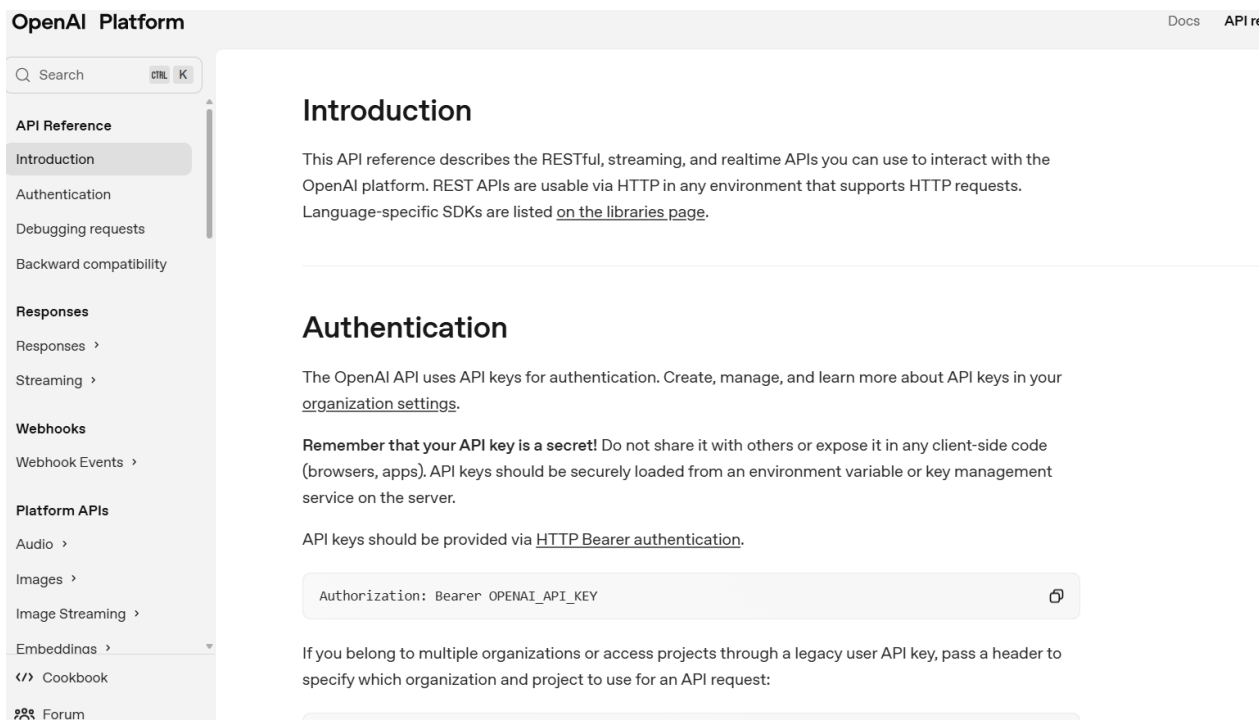


Figure (6): official page API GPT

Step 8 generate files .py and save in the folders

 _init_	29/07/2025 08:15	Python File	1 KB
 quiz_generator	28/07/2025 22:20	Python File	4 KB
 video_processing	27/07/2025 22:57	Python File	2 KB

Figure (7): ffmpeg page download

Step 9 create .env file

 __pycache__	01/08/2025 13:49	File folder	
 app	08/08/2025 14:58	File folder	
 templates	01/08/2025 13:49	File folder	
 uploads	01/08/2025 13:49	File folder	
 .env	28/07/2025 20:49	ENV File	1 KB

Figure (8): .env file

Step 10 create .html templates

					Sort 	View 	...
Name					Date modified	Type	Size
 index					28/07/2025 22:27	Microsoft Edge HT...	4 KB
 quiz					28/07/2025 22:29	Microsoft Edge HT...	6 KB

Figure (9): html files

Step 11 cream main.py where connect all the files

Name	Date modified	Type	Size
 __pycache__	01/08/2025 13:49	File folder	
 app	08/08/2025 14:58	File folder	
 templates	01/08/2025 13:49	File folder	
 uploads	01/08/2025 13:49	File folder	
 main	01/08/2025 13:41	Python File	3 KB
 .env	28/07/2025 20:49	ENV File	1 KB

Figure (10): main.py

Step 12 run main in anaconda prompt

```
(base) C:\Users\casil\OneDrive\Escritorio\edu-app>python main.py
* Serving Flask app 'main'
* Debug mode: on
INFO:werkzeug:WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
INFO:werkzeug:Press CTRL+C to quit
INFO:werkzeug: * Restarting with watchdog (windowsapi)
WARNING:werkzeug: * Debugger is active!
INFO:werkzeug: * Debugger PIN: 104-494-348
```

Figure (11): run main.py

Step 13 check if the app works

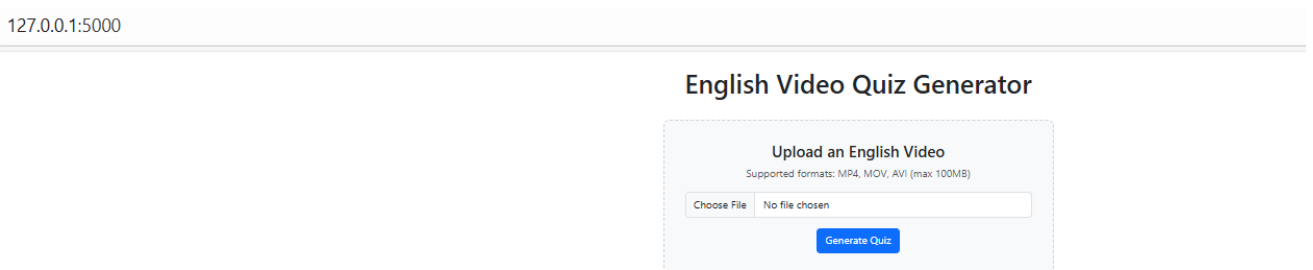


Figure (12): local app

Step 14 generate Pythonanywhere account and upload files

The screenshot shows the Pythonanywhere dashboard for user JuaCasTel. The top navigation bar includes links for Dashboard, Consoles, Files, Web, Tasks, and Databases. The dashboard displays system usage: CPU Usage is 0% (0.00s of 2,000s, resets in 2 hours, 53 minutes) and File storage is 43% full (8.6 GB of 20.0 GB quota). A 'Welcome, JuaCasTel' message is on the right. Below the usage stats are four panels: 'Recent Consoles' with a list of four active consoles; 'Recent Files' with a list of files including main.py, .env, __init__.py, video_processing.py, quiz_generator.py, requirements.txt, and quiz.html; 'Recent Notebooks' which is empty with buttons to 'Add new (Python 3.13)' and 'Browse all files'; and 'All Web apps' showing the user's profile and an 'Open Web tab' button. A 'New console' section at the bottom left offers options for Bash, Python, or More...

Figure (13): run main.py

Step 15 check if the app works well

The screenshot shows a web browser at the URL https://juacastel.pythonanywhere.com. The page features a large heading 'English Video Quiz Generator'. Below this is a light blue box titled 'Upload an English Video' with the text 'Supported formats: MP4, MOV, AVI (max 100MB)'. Inside the box is a file upload area with a 'Choose File' button and a text field showing 'No file chosen'. Below the upload area is a blue 'Generate Quiz' button.

Figure (14): run main.py