



Phishing Identification and Decision Support System (PIDSS)

The **PIDSS** Project:

The **Phishing Identification and Decision Support System** (PIDSS for short) is a tool developed as part of the Master Workforce Internship 2023. This tool was made to help with identifying Phishing emails by properly scanning for commonly known strategies and has an easy to understand Interface where users can see the "Phishing Percentage" based on how likely it is to be a Phishing attack. We hope to bring an easy to use and viable tool for users to detect, prevent and help identify one of the most common cyber attacks known.

Mentor:

- Prof. Hacniel Cardona

Developers:


- Nordiel E. Martinez
- Carlos Negron
- Edwin M. Vazquez

Python Setup:

- To get started with the Project we will first need to download the latest version of Python at **NOTE:** make sure to add Python to path:

Download Python

The official home of the Python Programming Language

 <https://www.python.org/downloads/>



- After Python is installed, head over to Visual Studio code and make sure to download the “Python” and “Pylance” extensions.
- Once the extensions are Installed, head over to the “Source Control” section at left sidebar, it should be the third icon shown. There we will clone the GitHub repository “PIDSS” which should have all the necessary files to start working on the project.


Chat GPT API and OpenAI package Setup:

- Login to the OpenAI platform using the project credentials:
- To set up the OpenAI Chat GPT API, we need to first sign up for OpenAI and obtain our API key:

OpenAI Platform



Explore developer resources, tutorials, API docs, and dynamic examples to get the most out of OpenAI's platform.

 <https://platform.openai.com/account/api-keys>

Gradio Package setup:

- For a quick test, we need to **run the code below** which is a "Hello World" using gradio:

```
import gradio as gr

def greet(name):
    return "Hello " + name + "!"

demo = gr.Interface(fn=greet, inputs="text", outputs="text")


demo.launch()
```

- Once gradio is all set up, please **read the documentation** to further explore the different methods for gradio GUI creation:

Gradio Interface Docs



Interface is Gradio's main high-level class, and allows you to create a web-based GUI / demo around a machine learning model (or any Python function)

 <https://www.gradio.app/docs/interface>

How to get the Development Environment set up:

To get started with PIDSS development, make sure to clone the current repository:

```
https://github.com/nordiel/PIDSS
```

After the repository is cloned, open the folder inside Visual Studio Code (or the code editor of your choice) and initialize a python virtual environment inside the repository folder. To initialize a python virtual environment, run the following command inside the repository directory:

```
python3 -m venv .venv
```

or

```
python -m venv .venv
```

Once the virtual environment is initialized, to get all the required packages for our project, make sure to have the `"requirements.txt"` file in the repository folder and run the following command:

```
pip3 install -r requirements.txt
```

or

```
pip install -r requirements.txt
```

Make sure your python environment is activated correctly. If not, use the command:

```
#Mac  
source .venv/bin/activate
```

```
#Windows  
.venv\Scripts\activate
```

Last but not least, create a `.env` file where we can define our environment variables. In this case, we will be defining our OpenAI API key. Your `.env` file should follow the correct naming conventions and look like this:

```
API_KEY='your API key here'
```

NOTE: Make sure to create a `.gitignore` file and include both the `.env` file and `.venv` folder as well as any other sensitive information.

