National College of Ireland

Configuration Manual

MSc Research Project
Cloud Computing

Kirthikesh Parthasarathy
Student ID:21195391

School of Computing
National College of Ireland

Supervisor: Shreyas Setlur Arun

National College of Ireland
Project Submission Sheet
School of Computing

National College of Ireland

Student Name:	Kirthikesh Parthasarathy
Student ID:	21195391
Programme:	Cloud Computing
Year:	2023
Module:	MSc Research Project
Supervisor:	Shreyas Setlur Arun
Submission Due Date:	14/12/2023
Project Title:	Configuration Manual
Word Count:	951
Page Count:	8

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:	Kirthikesh Parthasarathy
Date:	29th January 2024

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

Kirthikesh Parthasarathy
21195391

1 Introduction

This configuration manual provides full instructions for the creation and execution of the research project. "An Enhanced Version of Data Classification based on Confidentiality for Cloud Security".The manuals provide a clear and organised instructions for installing all of the dependencies, libraries, and packages needed to implement the project. Then, it discusses the entire architecture from an overview, initially providing the reader with information of each of the parts and how they are connected. The system installation provides an in-depth overview of the execution flow and processes, providing users a precise understanding of the interactions between modules during operation. The book also provides troubleshooting guidance for common mistakes and difficulties that may arise during installation or execution. By following the methods provided in this in-depth manual, the reader will have the required knowledge to successfully implement the project in a different environment. It is recommended that anyone planning to operate the system carefully read this manual full.

2 Hardware and Software Requirements

2.1 Software Requirements

MacOS and Linux has been used for this research. And i have developed this project with Python 3.7 and later version cna be used if its required. To download the python latest version use this url link: https://www.python.org/downloads/**?**

Component	**Specification**
Operating System	Linux
Main Memory	4GB
Number of Cores	8 (Virtual cores)
Storage	30GB
Programming Language	Python3
Python Libraries	Numpy, Pandas, mysqlclient, Keras, TensorFlow, scikit-learn, Flask

Table 1: System Specifications and Software Stack

2.2 Hardware Requirements

Before using this software, it is necessary for users to have a basic knowledge of Amazon Cloud, Anaconda. Navigator, Jupyter Notebook, Python programming, and frequent

machine learning models. Knowing a basic knowledge in Jupyter Notebook is necessary to effectively explore the file system, install packages, and perform commands through the terminal. Users should possess a basic understanding of, including the Jupyter administration of software, users, and processes. Knowledge in the Python programming language is also required. Users must possess a strong knowledge with Python syntax, data structures, control flow, modules, and virtual environments. Also, it is essential for users have knowledge in machine learning algorithms and models training, testing, and performance evaluation.

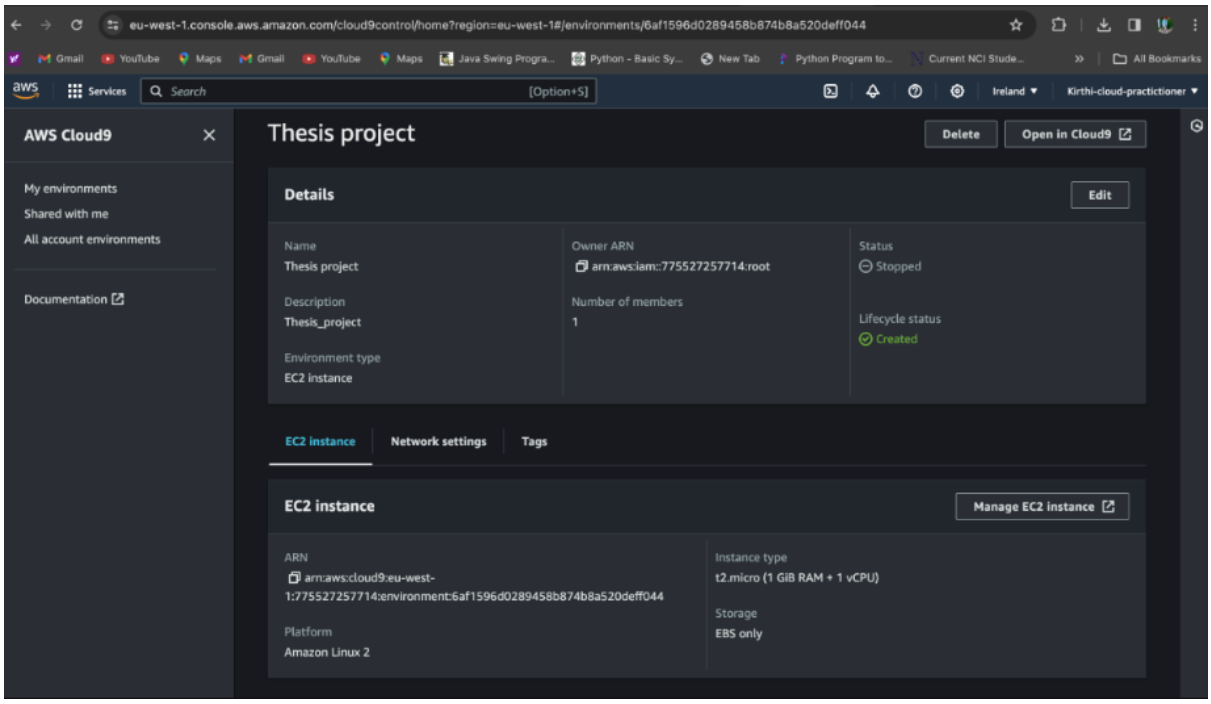


Figure 1: AWS EC2 Instance

3 Software Installation

- Install Anaconda3 using this link: https://www.anaconda.com/download. ?
- Install Python: Make sure you have Python installed on your computer?. You can download the latest version from the official Python website. Jupyter Notebook comes pre-installed with Anaconda distribution as well. https://www.python.org/downloads/

```
(myenv) ec2-user:~/environment/Final_Update_UI $ sudo yum install python3.7
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
246 packages excluded due to repository priority protections
Package python3-3.7.16-1.amzn2.0.4.x86_64 already installed and latest version
```

Figure 2: sudo yum install python3.7

- Install Jupyter: You can install Jupyter Notebook using the following command in your terminal or command prompt: pip install jupyter

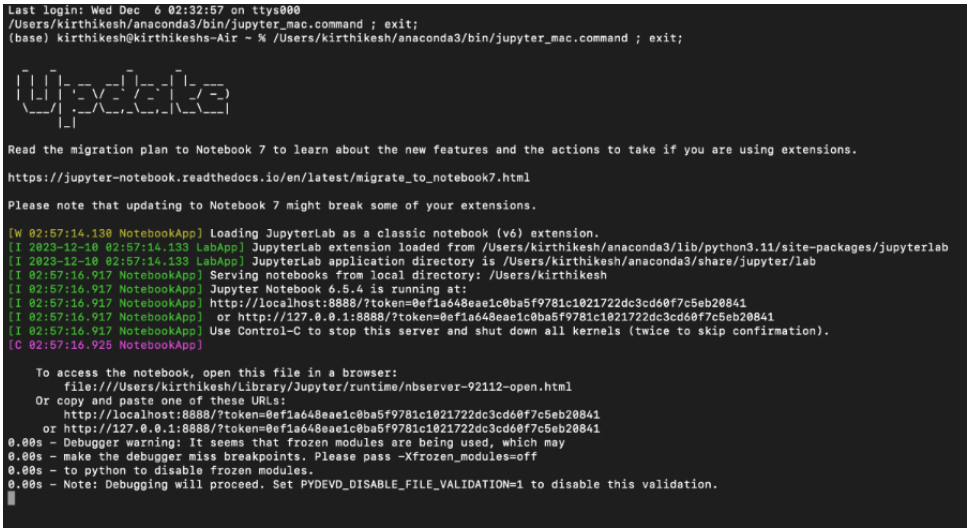


Figure 3: pip install jupyter

- Open Terminal/Command Prompt: Once you have Jupyter installed, open your terminal (Linux/Mac) or command prompt (Windows).
- Navigate to the Desired Directory: Use the cd command to navigate to the directory where you want to create or access your Jupyter Notebook files. To access the notebook, open this file in a browser: `file:///Users/kirthikesh/Library/Jupyter/runtime/nbserver-92112-open.html`.
- Start Jupyter Notebook: In the terminal/command prompt, enter the following command:jupyter notebook
- This will start the Jupyter Notebook server and open a new tab or window in your default web browser.

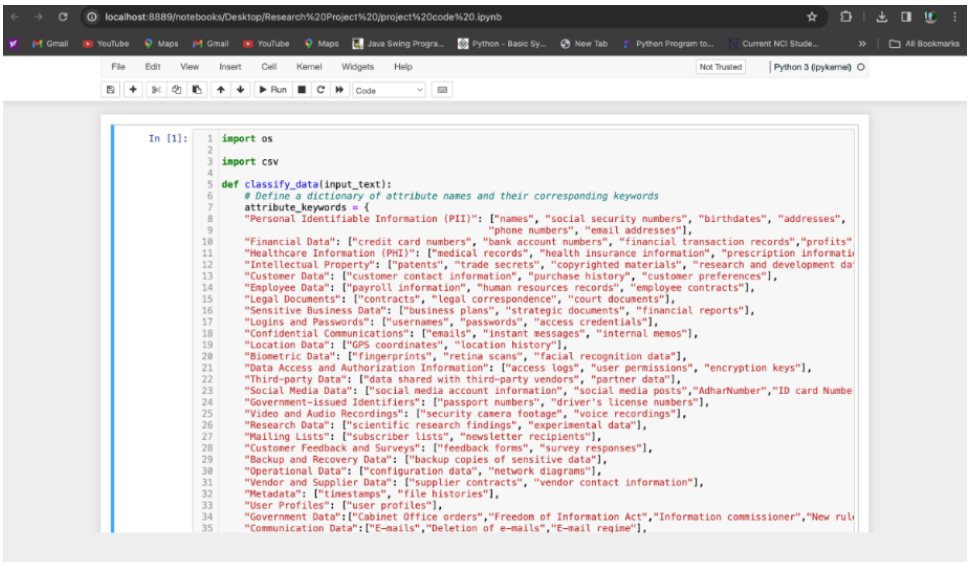


Figure 4: Jupyter Notebook

4 Algorithm Implementation

4.1 DataSet Collection

During this phase, the data was categorized into two main types: confidential and non-confidential. To achieve this, key attributes were utilized for distinguishing confidential data. And the dataSet has been taken from kaggle and It is publicly accessible.

The textual data was collected from the Kaggle datasets which can be accessed using the following link:

`https://www.kaggle.com/datasets/jensenbaxter/10dataset-text-document-classificatio`
`select=business.?`

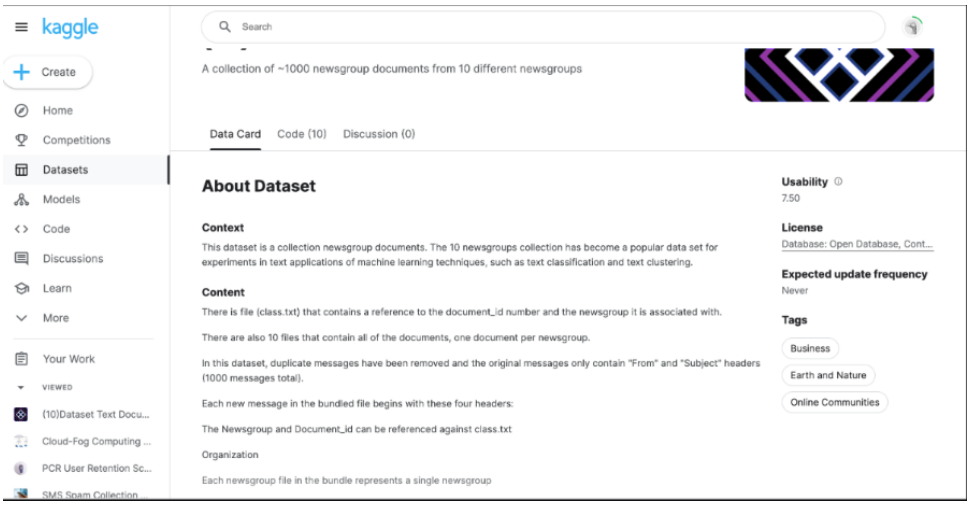


Figure 5: Kaggle datasets

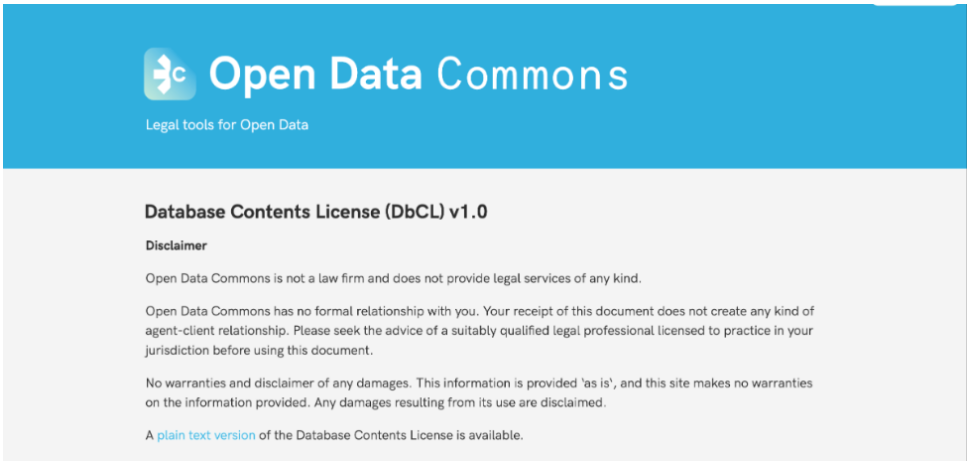


Figure 6: Proof: Database Contents Licence (DbCL) v1.0

4.2 CNN-LSTM with Random Forest Algorithm

The model is defined using TensorFlow's Keras API. It starts with a 1D convolutional layer (Conv1D) with 256 filters, a kernel size of 5, and a rectified linear unit (ReLU)

activation function. This layer is designed for causal padding, meaning it considers only the past values in the input sequence.Followed by two LSTM layers: one with 128 units and another with 64 units. Both use the hyperbolic tangent (tanh) activation function, and the first LSTM layer returns sequences.Subsequently, there are three dense layers with 32, 16, and 8 units, respectively, using ReLU activation. The last layer is a lambda layer that scales the output by a factor of 100. This scaling compensates for the fact that the LSTM's tanh activation function outputs values between -1 and 1 After applying the CNN-LSTM architecture, we obtain a set of eight features from the input data.

Figure 7: Code Snippet for TensorFlow's Keras API

The chosen method for secure data transfer is through the use of the Speck lightweight encryption algorithm.

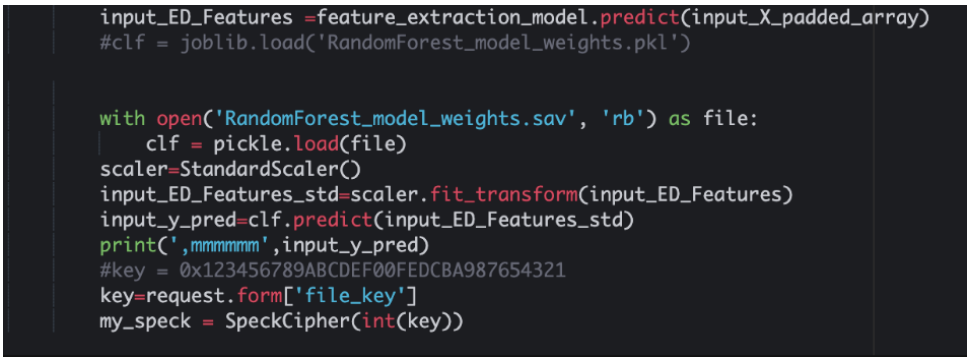
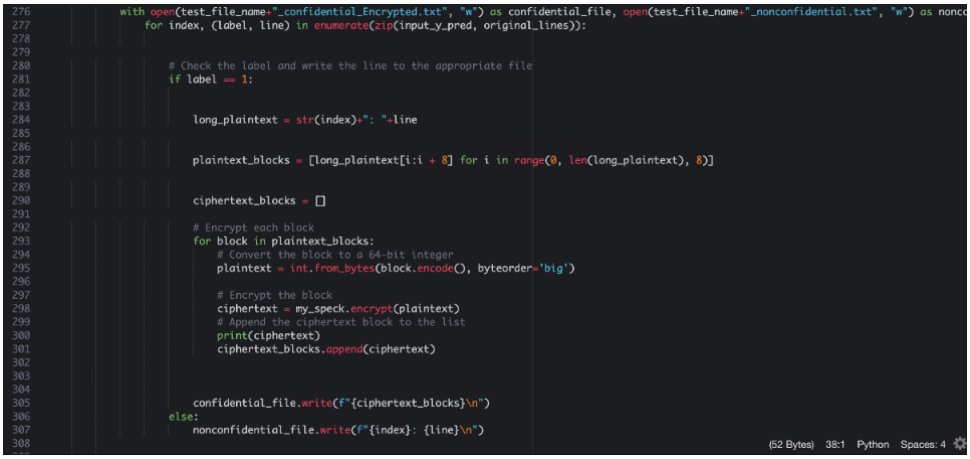


Figure 8: Code Snippet for Random Forest Algorithm Implementation

After completing model development, a secure data transfer technique is implemented.The entire framework is then designed into a user interface (UI).Within the UI, users can register under the Container as a Service (CaaS) platform. They are granted the ability to store and download their own data securely.

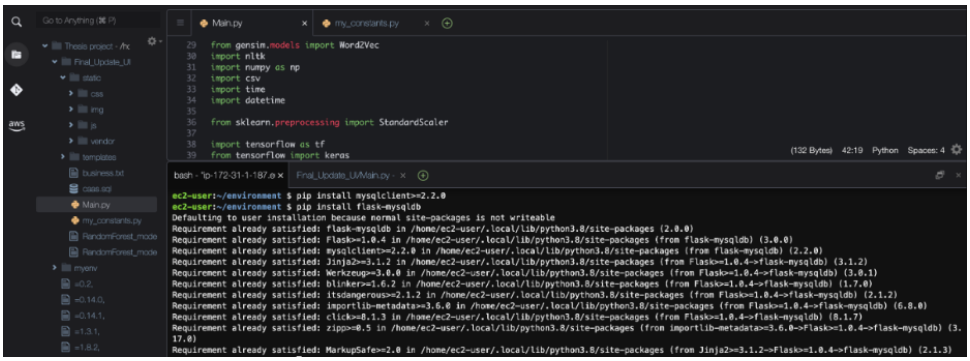
The AUG_ConvoLSTM-RF trained model weights are used to categorise data into confidential and non-confidential segments. Notably, encryption is selectively applied only to confidential data to ensure secure storage and transfer.



```
with open(test_file_name+"_confidential_Encrypted.txt", "w") as confidential_file, open(test_file_name+"_nonconfidential.txt", "w") as nonc
    for index, (label, line) in enumerate(zip(input_y_pred, original_lines)):


        # Check the label and write the line to the appropriate file
        if label == 1:


            long_plaintext = str(index)+": "+line


            plaintext_blocks = [long_plaintext[i:i + 8] for i in range(0, len(long_plaintext), 8)]


            ciphertext_blocks = []

            # Encrypt each block
            for block in plaintext_blocks:
                # Convert the block to a 64-bit integer
                plaintext = int.from_bytes(block.encode(), byteorder='big')

                # Encrypt the block
                ciphertext = my_speck.encrypt(plaintext)
                # Append the ciphertext block to the list
                print(ciphertext)
                ciphertext_blocks.append(ciphertext)



            confidential_file.write(f"{ciphertext_blocks}\n")
        else:
            nonconfidential_file.write(f"{index}: {line}\n")
```

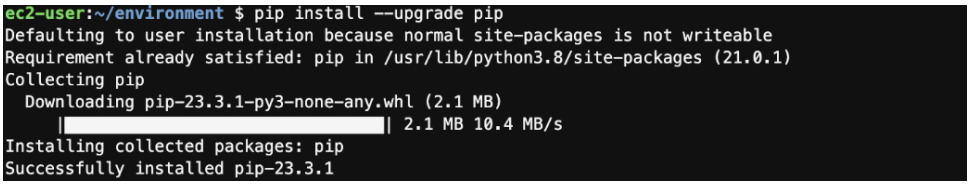
Figure 9: Code snippet for Encryption and Decryption Using Speck

The code snippet of library packages that i have installed in this project



```
from gensim.models import Word2Vec
import nltk
import numpy as np
import csv
import time
import datetime

from sklearn.preprocessing import StandardScaler

import tensorflow as tf
from tensorflow import keras

ec2-user:~/environment $ pip install mysqlclient>=2.2.0
ec2-user:~/environment $ pip install flask-mysqldb
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: flask-mysqldb in /home/ec2-user/.local/lib/python3.8/site-packages (2.0.0)
Requirement already satisfied: Flask>=1.0.4 in /home/ec2-user/.local/lib/python3.8/site-packages (from flask-mysqldb) (3.0.0)
Requirement already satisfied: mysqlclient>=2.2.0 in /home/ec2-user/.local/lib/python3.8/site-packages (from flask-mysqldb) (2.2.0)
Requirement already satisfied: Jinja2>=3.1.2 in /home/ec2-user/.local/lib/python3.8/site-packages (from Flask>=1.0.4->flask-mysqldb) (3.1.2)
Requirement already satisfied: Werkzeug>=3.0.0 in /home/ec2-user/.local/lib/python3.8/site-packages (from Flask>=1.0.4->flask-mysqldb) (3.0.1)
Requirement already satisfied: blinker>=1.6.2 in /home/ec2-user/.local/lib/python3.8/site-packages (from Flask>=1.0.4->flask-mysqldb) (1.7.0)
Requirement already satisfied: itsdangerous>=2.1.2 in /home/ec2-user/.local/lib/python3.8/site-packages (from Flask>=1.0.4->flask-mysqldb) (2.1.2)
Requirement already satisfied: importlib-metadata>=3.6.0 in /home/ec2-user/.local/lib/python3.8/site-packages (from Flask>=1.0.4->flask-mysqldb) (6.8.0)
Requirement already satisfied: click>=8.1.3 in /home/ec2-user/.local/lib/python3.8/site-packages (from Flask>=1.0.4->flask-mysqldb) (8.1.7)
Requirement already satisfied: zipp>=0.5 in /home/ec2-user/.local/lib/python3.8/site-packages (from importlib-metadata>=3.6.0->Flask>=1.0.4->flask-mysqldb) (3.17.0)
Requirement already satisfied: MarkupSafe>=2.0 in /home/ec2-user/.local/lib/python3.8/site-packages (from Jinja2>=3.1.2->Flask>=1.0.4->flask-mysqldb) (2.1.3)
```

Figure 10: Code snippet for installing Flask-mysqldb



```
ec2-user:~/environment $ pip install --upgrade pip
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: pip in /usr/lib/python3.8/site-packages (21.0.1)
Collecting pip
  Downloading pip-23.3.1-py3-none-any.whl (2.1 MB)
     |████████████████████████████████| 2.1 MB 10.4 MB/s
Installing collected packages: pip
Successfully installed pip-23.3.1
```

Figure 11: sudo yum Upgrade

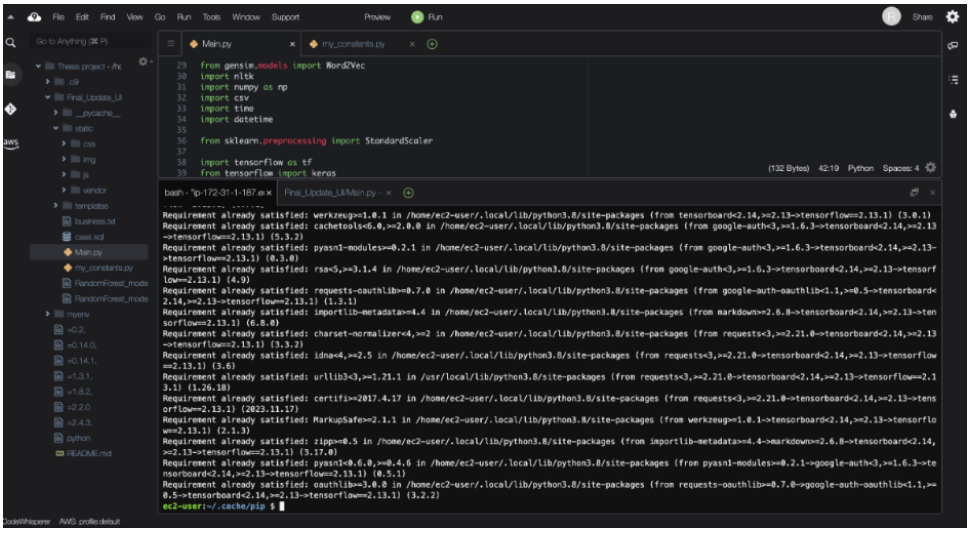


Figure 12: Code snippet for installing TenserFlow

XAMPP is widely used in projects for its simplicity and versatility. It provides an integrated environment with Apache for web hosting, MySQL for database management, PHP for server-side scripting, and more. This all-in-one package streamlines the setup of a local development server, enabling developers to create, test, and debug web applications seamlessly.

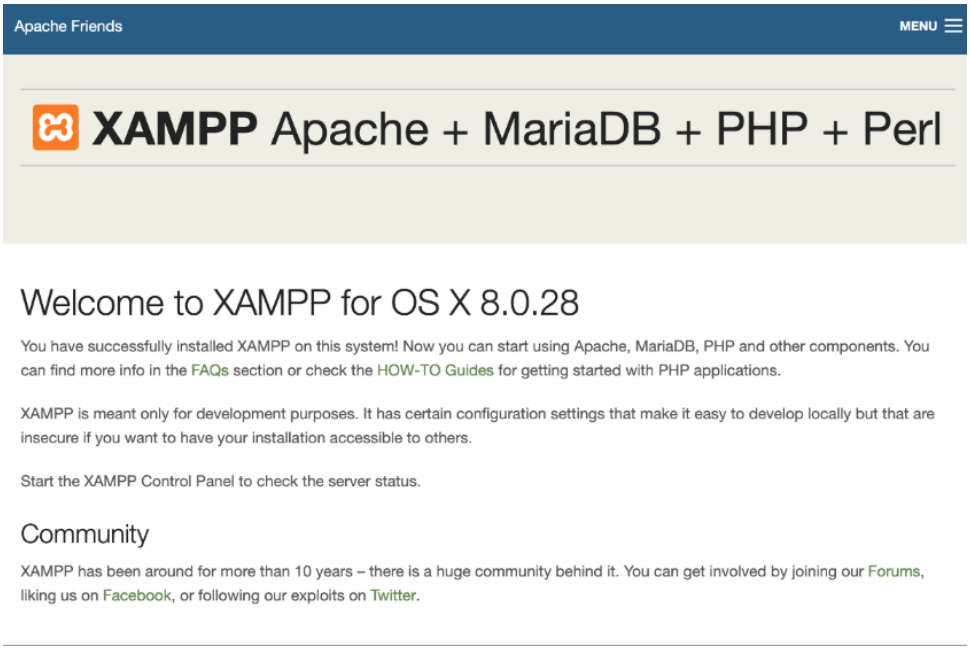


Figure 13: Code snippet for installing TenserFlow

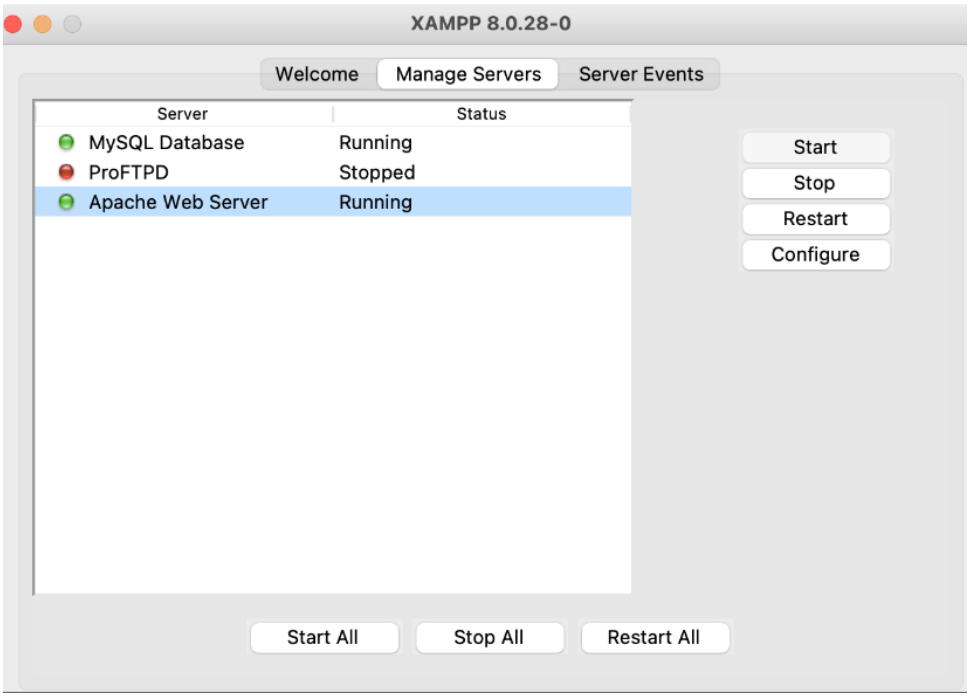


Figure 14: Code snippet for installing TenserFlow

‘

5 Presentation and Demo Video Link

Click on the following link to watch the video: https://youtu.be/pSSNWkzr9pE