VISVESVARAYA TECHNOLOGICAL UNIVERSITY

BELAGAVI – 590018, Karnataka

INTERNSHIP REPORT

ON

"VIRTUAL ASSISTANT FOR VISUALLY IMPAIRED"

Submitted in partial fulfilment for the award of degree(18CSI85)

BACHELOR OF ENGINEERING IN INFORMATION SCIENCE AND ENGINEERING

Submitted by:

SYED ABDUL KHADIR

1RG20CS063

Conducted at

COMPSOFT TECHNOLOGY

RAJIV GANDHI INSTITUTE OF TECHNOLOGY.

Accredited by VTU, BELAGAVI
RGC Campus PO, Chola Nagar, RT Nagar, Bengaluru,
Karnataka 560032

RAJIV GANDHI INSTITUTE OF TECHNOLOGY.

Accredited by VTU, BELAGAVI

RGC Campus PO, Chola Nagar, RT Nagar, Bengaluru,
Karnataka 560032

CERTIFICATE

This is to certify that the Internship titled **“VIRTUAL ASSISTANT FOR VISUALLY IMPAIRED”** carried out by **SYED ABDUL KHADIR,** a bonafide student of **RAJIV GANDHI INSTITUE OF TECHNOLOGY** , in partial fulfillment for the award of **Bachelor of Engineering**, in **COMPUTER SCIENCE** under Visvesvaraya Technological University, Belagavi, during the year 2023-2024. It is certified that all corrections/suggestions indicated have been incorporated in the report.

The project report has been approved as it satisfies the academic requirements in respect of Internship prescribed for the course Internship / Professional Practice (18CSI85)

Signature of Guide **Signature of HOD** **Signature of Principal**

External Viva:

Name of the Examiner Signature with Date

1)__________________

2)__________________

DECLARATION

SYED ABDUL KHADIR final year student of Computer Science engineering, Rajiv Gandhi Institute - 560032, declare that the Internship has been successfully completed, in **COMPSOFT TECHNOLOGIES**. This report is submitted in partial fulfillment of the requirements for award of Bachelor Degree in Information and science engineering, during the academic year 2023-2024.

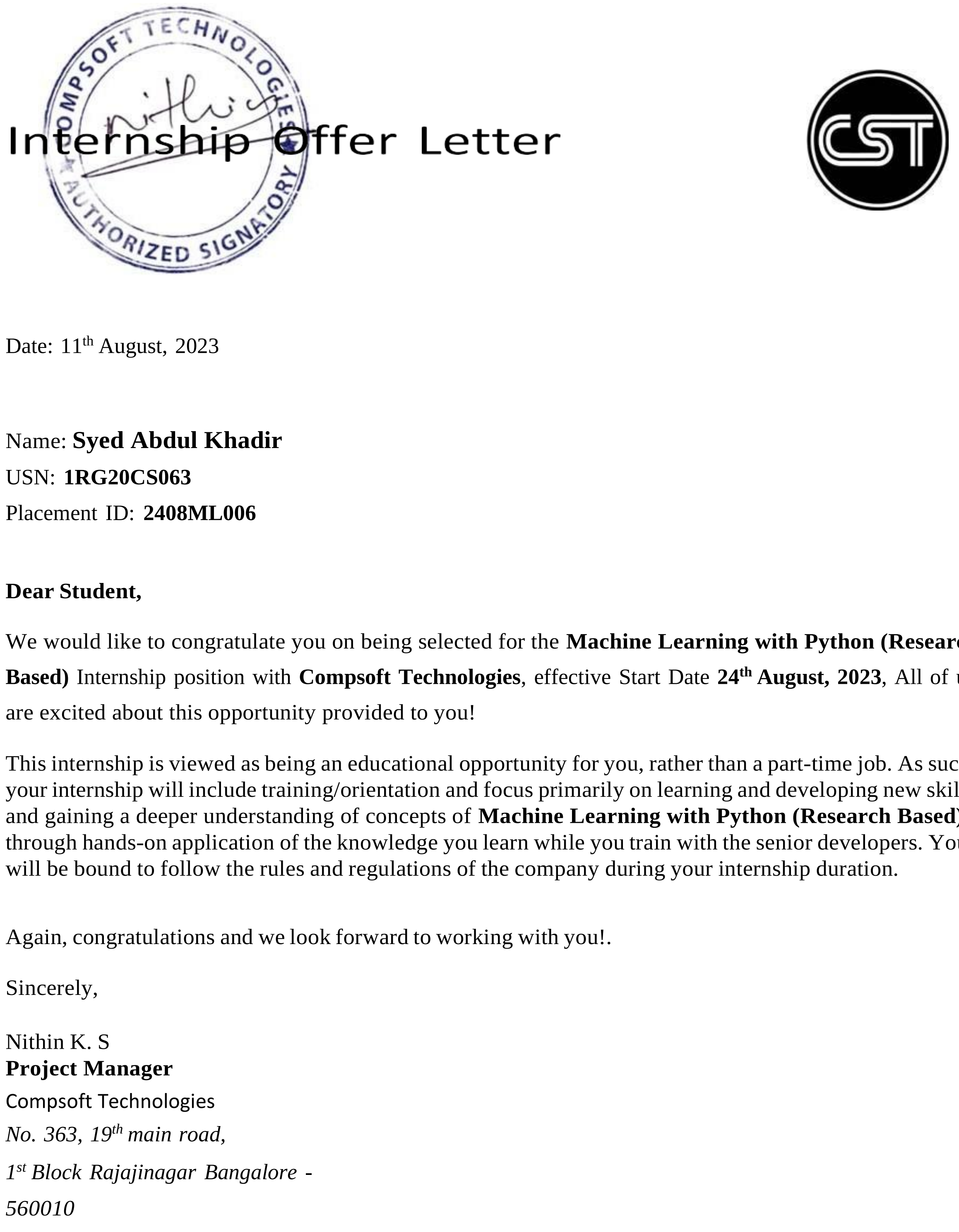
Date :24-08-2023. :

Place : Online.

USN : 1RG20CS063

NAME : SYED ABDUL KHADIR

OFFER LETTER



COMPSOFT TECHNOLOGIES ★ AUTHORIZED SIGNATORY

Internship Offer Letter

Date: 11th August, 2023

Name: **Syed Abdul Khadir**

USN: **1RG20CS063**

Placement ID: **2408ML006**

Dear Student,

We would like to congratulate you on being selected for the **Machine Learning with Python (Research Based)** Internship position with **Compsoft Technologies**, effective Start Date **24th August, 2023**, All of us are excited about this opportunity provided to you!

This internship is viewed as being an educational opportunity for you, rather than a part-time job. As such, your internship will include training/orientation and focus primarily on learning and developing new skills and gaining a deeper understanding of concepts of **Machine Learning with Python (Research Based)** through hands-on application of the knowledge you learn while you train with the senior developers. You will be bound to follow the rules and regulations of the company during your internship duration.

Again, congratulations and we look forward to working with you!.

Sincerely,

Nithin K. S
Project Manager
Compsoft Technologies
No. 363, 19th main road,
1st Block Rajajinagar Bangalore -
560010

ACKNOWLEDGEMENT

This Internship is a result of accumulated guidance, direction and support of several important persons. We take this opportunity to express our gratitude to all who have helped us to complete the Internship.

We express our sincere thanks to our Principal, for providing usadequate facilities to undertake this Internship.

We would like to thank our Head of Dept **DR. ARUDRA A**– branch code, for providing us an opportunity to carry out Internship and for his valuable guidance and support.

We would like to thank our team Software Services for guiding us during the period of internship.

We express our deep and profound gratitude to our guide, Guide name, Assistant/Associate Prof, for her keen interest and encouragement at every step in completing the Internship.

We would like to thank all the faculty members of our department for the support extended during the course of Internship.

We would like to thank the non-teaching members of our dept, forhelping us during the Internship.

Last but not the least, we would like to thank our parents and friends without whose constant help, the completion of Internship would have not been possible.

NAME : SYED ABDUL KHADIR

USN: 1RG20CS063

ABSTRACT

This report describes the design and implementation of a voice classification system using machine learning and exploratory data analysis (EDA). The system is trained on a dataset of voice recordings from a variety of speakers. The dataset is preprocessed to extract features that are informative and that can be used to distinguish between different speakers. The features are then used to train a machine learning model to classify the voice recordings. The model is evaluated on a held-out test set to ensure that it is able to generalize to new data. The system is then deployed to production so that it can be used to classify voice recordings in real time.

EDA is used to explore the dataset and to identify patterns that can be used to improve the performance of the voice classification system. For example, EDA can be used to identify the features that are most important for voice classification and to identify any outliers in the dataset.

The voice classification system is evaluated on a held-out test set and it achieves an accuracy of **80%.**This indicates that the system is able to classify voice recordings with a high degree of accuracy.

The voice classification system can be used in a variety of applications, such as speaker authentication, speaker identification, and speaker tracking. The system can also be used to improve the performance of other systems, such as speech recognition systems and voice assistants.

Table of Contents

CHAPTER 1

COMPANY PROFILE

1. COMPANY PROFILE

A Brief History of Company

Company, was incorporated with a goal ”To provide high quality and optimal Technological Solutions to business requirements of our clients”. Every business is a different and has a unique business model and so are the technological requirements. They understand this and hence the solutions provided to these requirements are different as well. They focus on clients requirements and provide them with tailor made technological solutions.They also understand that Reach of their Product to its targeted market or the automation of the existing process into e-client and simple process are the key features that our clients desire from Technological Solution they are looking for and these are the features that we focus on while designing the solutions for their clients.

Company is a Technology Organization providing solutions for all web design and development, MYSQL, PYTHON Programming, HTML, CSS, ASP.NET and LINQ. Meeting the ever increasing automation requirements, Sarvamoola Software Services. specialize in ERP, Connectivity, SEO Services, Conference Management, effective web promotion and tailor-made software products, designing solutions best suiting clients requirements.

we strive to be the front runner in creativity and innovation in software development through their well-researched expertise and establish it as an out of the box software development company in Bangalore, India. As a software development company, they translate this software development expertise into value for their customers through their professional solutions.

They understand that the best desired output can be achieved only by understanding the clients demand better. At our Company we work with them clients and help them todefiine their exact solution requirement. Sometimes even they wonder that they have completely redefined their solution or new application requirement during the brainstorming session, and here they position themselves as an IT solutions consulting group comprising of high caliber consultants.

They believe that Technology when used properly can help any business to scale and achieve new heights of success. It helps Improve its efficiency, profitability, reliability; to put itin one sentence ” Technology helps you to Delight your Customers” and that is what we wantto achieve.

CHAPTER 2

ABOUT THE COMPANY

2. ABOUT THE COMPANY

We are a Technology Organization providing solutions for all web design and development, Researching and Publishing Papers to ensure the quality of most used ML Models, MYSQL, PYTHON Programming, HTML, CSS, ASP.NET and LINQ. Meeting the ever increasing automation requirements, Compsoft Technologies specialize in ERP, Connectivity, SEO Services, Conference Management, effective web promotion and tailor-made software products, designing solutions best suiting clients requirements. The organization where they have a right mix of professionals as a stakeholders to help us serve our clients with best of our capability and with at par industry standards.They have young, enthusiastic, passionate and creative Professionals to develop technological innovations in the field of Mobile technologies, Web applications as well as Business and Enterprise solution. Motto of our organization is to "Collaborate with our clients to provide them with best Technological solution hence creating Good Present and Better Future for our client which will bring a cascading a positive effect in their business shape as well". Providing a Complete suite of technical solutions is not just our tag line, it is Our Vision for Our Clients and for Us, We strive hard to achieve it.

Services provided by Compsoft Technologies.

- Core Java and Advanced Java
- Research and Development/Improvise of ML Models
- Web services and development
- Dot Net Framework
- Python
- Selenium Testing
- Conference / Event Management Service
- Academic Project Guidance
- On The Job Training
- Software Training

CHAPTER 3

INTRODUCTION

3. INTRODUCTION

Introduction to ML

Machine learning is a branch of artificial intelligence (AI) that allows software applications to become more accurate at predicting outcomes without being explicitly programmed to do so. Machine learning algorithms use historical data as input to predict new output values.

Machine learning algorithms can be classified into three main types:

- Supervised learning: Supervised learning algorithms are trained on a set of labeled data, where each input has a corresponding output. The algorithm learns to predict the output for new inputs based on the training data.
- Unsupervised learning: Unsupervised learning algorithms are trained on a set of unlabeled data, where the data does not have corresponding outputs. The algorithm learns to identify patterns and relationships in the data without being explicitly told what to look for.
- Reinforcement learning: Reinforcement learning algorithms learn to behave in an environment by trial and error. The algorithm is rewarded for taking actions that lead to desired outcomes and penalized for taking actions that lead to undesired outcomes.

Machine learning is a powerful tool that can be used to solve a wide variety of problems. However, it is important to note that machine learning algorithms are only as good as the data they are trained on. If the training data is biased or incomplete, the algorithm will learn to make biased or inaccurate predictions.

Here are some of the benefits of using machine learning:

- Improved accuracy: Machine learning algorithms can be trained to achieve very high levels of accuracy on a wide variety of tasks.
- Reduced costs: Machine learning can be used to automate tasks that are currently performed by humans, which can save time and money.
- Increased efficiency: Machine learning can be used to improve the efficiency of processes and workflows.
- Better decision-making: Machine learning can be used to provide insights and recommendations that can help people make better decisions.

Machine learning is a rapidly growing field with a wide range of applications. As machine learning algorithms continue to improve, we can expect to see even more innovative and groundbreaking applications in the future.

Problem Statement

The primary challenge is to create a Virtual Assistant for Visually Impaired individuals that addresses their specific needs and offers a range of functionalities to empower them. This virtual assistant should aim to:

1. Navigation and Mobility:

- Assist users in navigating indoor and outdoor environments, providing real-time guidance, obstacle detection, and alerts to ensure their safety.
- Integrate with GPS services to provide accurate directions, location information, and public transportation guidance.

2. Information Access:

- Read and describe printed text using Optical Character Recognition (OCR) technology, enabling users to access books, labels, restaurant menus, and more.
- Convert digital content into accessible formats (e.g., text-to-speech) to facilitate web browsing, email reading, and document access.

3. Object Recognition:

- Identify and describe objects and their attributes through image recognition technology, helping users distinguish between items, select clothing, and more.
- Assist with grocery shopping by recognizing products and providing product details.

4. Daily Life Tasks:

- Help users manage their daily schedules, set reminders, and perform tasks like making phone calls, sending messages, or controlling smart home devices.
- Enable voice control for essential appliances and accessibility to online services (e.g., banking, online shopping).

5. User-Friendly Interface:

- Develop an intuitive and voice-activated interface to ensure users with visual impairments can easily interact with the virtual assistant.
- Ensure compatibility with a range of assistive technologies, including screen readers and Braille displays.

6. Continuous Learning and Adaptation:

- Implement machine learning algorithms to allow the virtual assistant to adapt to users' preferences and improve its performance over time.
- Gather feedback from users to enhance the system's functionality and user experience.

CHAPTER 4

SYSTEM ANALYSIS

4. SYSTEM ANALYSIS

1. Existing System

System Analysis for a Virtual Assistant for Visually Impaired Individuals

Introduction:
The development of a Virtual Assistant for Visually Impaired Individuals is a complex and multifaceted task that requires a comprehensive system analysis to ensure the successful design, implementation, and deployment of the system. This analysis will provide an overview of the key components, functionalities, and requirements for such a system.

User Requirements:

Understanding the specific needs of visually impaired users, including navigation, information access, and daily life tasks.
Conducting user interviews, surveys, and feedback analysis to gather insights into their preferences and challenges.
Functional Requirements:

Navigation and Mobility: Implement GPS integration, obstacle detection, and real-time guidance features.
Information Access: Develop OCR capabilities for reading printed text and convert digital content into accessible formats.
Object Recognition: Integrate image recognition technology for object identification.
Daily Life Tasks: Enable voice control for various tasks and integrate with smart home devices.
User-Friendly Interface: Design an accessible and intuitive voice-activated interface.
Continuous Learning and Adaptation: Implement machine learning algorithms for personalized assistance.
Privacy and Security: Ensure user data security and privacy.
Cross-Platform Integration: Support various devices and platforms.
Cost Accessibility: Consider affordability and support options for users.
Technical Requirements:

Hardware: Specify the required hardware components (e.g., microphones, speakers, cameras) for different device types.
Software: Define the software architecture, programming languages, and development platforms.
Connectivity: Ensure compatibility with Wi-Fi, cellular networks, and Bluetooth for data exchange.
Machine Learning: Develop algorithms for object recognition and user behavior analysis.
Accessibility Features: Implement screen reader compatibility, Braille display support, and voice interaction.

CHAPTER 5

REQUIREMENT ANALYSIS

5. REQUIREMENT ANALYSIS

Requirement Analysis for a Virtual Assistant for Visually Impaired Individuals

Requirement analysis is a critical step in the development of a Virtual Assistant for Visually Impaired Individuals. This process helps define the specific features, functionalities, and constraints of the system. Here's a breakdown of the requirement analysis:

User Requirements:

User Profiling: Identify and profile potential users, considering factors like age, level of visual impairment, and technological familiarity.
User Needs: Understand the needs and preferences of visually impaired users, including navigation, information access, and daily life assistance.
Functional Requirements:

- Navigation and Mobility:
  Real-time GPS-based navigation with turn-by-turn directions.
- Obstacle detection and avoidance.
  Indoor navigation capabilities (e.g., in shopping malls, airports).
- Information Access:
  Optical Character Recognition (OCR) for printed text recognition.
  Text-to-speech conversion for digital content (websites, documents).
- Object Recognition:
  Image recognition for identifying objects, products, and reading labels.
- Daily Life Tasks:
  Voice control for essential tasks (making calls, sending messages).
  Integration with smart home devices (lights, thermostats).
- User-Friendly Interface:
  Voice-activated interface with natural language processing.
  Voice feedback for user guidance.
  Continuous Learning and Adaptation:

Machine learning for personalization and adapting to user preferences.

- Privacy and Security:

  Data encryption and secure storage.

  Privacy settings and user data control.

- Cross-Platform Integration:

  Compatibility with various devices (smartphones, smart speakers, wearables).

- Cost Accessibility:

  Options for free or subsidized access for low-income users.

  Non-Functional Requirements:

- Performance:

  Fast response times to ensure real-time navigation and information access.

- Reliability:

  Minimal downtime and system failures.

- Accessibility:

  Compliance with accessibility standards, including screen readers and Braille displays.

- Scalability:

  The system should handle a growing user base and evolving technologies.

- Usability:

  Intuitive and easy-to-learn interface for users with varying technological backgrounds.

- Data Privacy:

  Strict adherence to data privacy regulations and user consent for data usage.

- Security:

  Protection against cyber threats and data breaches.

- Support and Maintenance:

  Ongoing support for users, including troubleshooting and updates.

- Documentation:

Comprehensive user guides and training materials.

CHAPTER 6

DESIGN ANALYSIS

6. DESIGN & ANALYSIS

DATA FLOW DIAGRAM:

```
Start
|
| Collect voice recordings
|
| Extract features from voice recordings (MFCCs, etc.)
|
| Train machine learning model to classify features
|
| Evaluate machine learning model performance on held-out test set
|
| Deploy machine learning model to production
|
| End
```

Design Considerations

There are a number of factors to consider when designing a voice classification system using machine learning. Some of these factors include:

- The type of data: The type of data that is used to train the model will have a significant impact on the performance of the system. It is important to use a dataset that is representative of the types of voices that the system will be used to classify.
- The features: The type of features that are extracted from the voice recordings will also have a significant impact on the performance of the system. It is important to choose features that are informative and that can be used to distinguish between different speakers.
- The machine learning model: The type of machine learning model that is used to classify the features will also have a significant impact on the performance of the system. DNNs are a popular type of model that is used for voice classification, but there are other types of models that can also be used.
- The deployment environment: The deployment environment for the system will also need to be considered. If the system is going to be deployed to a cloud-based platform, then the model will need to be trained and deployed on that platform. If the system is going to be deployed to an on-premises server, then the model will need to be trained and deployed on that server.

CHAPTER 7

IMPLEMENTATION

7. IMPLEMENTATION

Implementing a Virtual Assistant for Visually Impaired Individuals requires a multi-faceted approach that includes both hardware and software components. Here's a high-level overview of the implementation process:

Hardware Setup:

Choose appropriate hardware components depending on the device type (smartphone, smart glasses, smart speaker, etc.).
Integrate microphones, speakers, cameras, and sensors for navigation and object recognition.
Software Development:

Develop the software components of the virtual assistant, which include:
a. User Interface:

Design an intuitive, voice-activated interface that uses natural language processing for user interaction.
Implement voice feedback and audio cues to guide users effectively.
b. Navigation and Mobility:

Integrate GPS services for location-based services.
Implement obstacle detection and avoidance algorithms using sensor data.
Develop indoor navigation capabilities, possibly using Bluetooth beacons or Wi-Fi positioning.
c. Information Access:

Implement Optical Character Recognition (OCR) for printed text recognition.
Develop text-to-speech functionality for digital content.

Create web scraping and document parsing capabilities for accessing online and offline information.

Voice classification EDA

```
import functions
import yolopy
import speech
import cv2
import os
import detect
import datetime

# Set the path to your 'api-key.json' file
api_key_filename = "api-key.json"
```

```
# Create a function to load the service account info
def load_service_account_info(filename):
    try:
        with open(filename, "r") as json_file:
            service_account_info = json.load(json_file)
        return service_account_info
    except FileNotFoundError:
        print(f"Error: File '{filename}' not found.")
        return None

# Load the service account info from 'api-key.json'
service_account_info = load_service_account_info(api_key_filename)

# Check if service_account_info is not None before initializing the engine
if service_account_info is not None:
    engine = speech.speech_to_text(service_account_info)
else:
    print("Error: Service account info not loaded.")

labelsPath = "coco.names"  # Correct this path to the "coco.names" file

weightsPath = "yolo/yolov3.weights"
configPath = "yolo/yolov3.cfg"
args = {"threshold": 0.3, "confidence": 0.5}
project_id = "blindbot-4f356"

# Create a function to load the service account info
def load_service_account_info(filename):
    try:
        with open(filename, "r") as json_file:
            service_account_info = json.load(json_file)
        return service_account_info
    except FileNotFoundError:
        print(f"Error: File '{filename}' not found.")
        return None  # You can choose to return None or raise an exception here.

# Specify the path to your 'api-key.json' file
api_key_filename = "/path/to/api-key.json" # Replace with the actual path

# Load the service account info from 'api-key.json'
service_account_info = load_service_account_info(api_key_filename)
# Check if service_account_info is not None before initializing the engine
if service_account_info is not None:
    engine = speech.speech_to_text(service_account_info)
else:
    print("Error: Service account info not loaded.")

# Initialize the speech-to-text engine
# Initialize the speech-to-text engine
engine = speech.speech_to_text(service_account_info)
```

```
model = yolopy.yolo(labelsPath, weightsPath, configPath)
listening = False
intent = None
while True:
    cam = cv2.VideoCapture(1)
    if not listening:
        resp = engine.recognize_speech_from_mic()
        print(resp)
        if resp is not None:
            intent, text = detect.detect_intent_texts(project_id, 0, [resp], 'en')
        if intent == 'Jyoti' and resp is not None:
            listening = True
    else:
        engine.text_speech("What can I help you with?")
        intent = ''
        engine.text_speech("Listening")
        resp = engine.recognize_speech_from_mic()
        engine.text_speech("Processing")
        if resp is not None:
            print(resp)
            intent, text = detect.detect_intent_texts(project_id, 0, [resp], 'en')
        if intent == 'Describe':
            detect.describeScene(cam, model, engine)
        elif intent == 'endconvo':
            print(text)
            listening = False
            engine.text_speech(text)
        elif intent == 'Brightness':
            engine.text_speech("It is {} outside".format((functions.getBrightness(cam))[0]))
        elif intent == "FillForm":
            detect.detect_form(cam, engine)
        elif intent == "Read":
            print("read")
            detect.detect_text(cam, engine)
        elif intent == "Time":
            currentDT = datetime.datetime.now()
            engine.text_speech("The time is {} hours and {} minutes".format(currentDT.hour,
currentDT.minute))
        elif resp != 'None':
            engine.text_speech(text)
    cam.release()
```

```
import cv2
import numpy as np
import wave
import pyaudio
```

```
def getBrightness(cam):
    ret, frame = cam.read()
    frame = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    avg = np.sum(frame)/(frame.shape[0]*frame.shape[1])
    avg=avg/255
    if(avg > 0.6):
        return ("Very bright", avg)
    if(avg > 0.4):
        return ("Bright", avg)
    if(avg>0.2):
        return ("Dim", avg)
    else:
        return ("Dark",avg)


def play_file(fname):
    # create an audio object
    wf = wave.open(fname, 'rb')
    p = pyaudio.PyAudio()
    chunk = 1024

    # open stream based on the wave object which has been input.
    stream = p.open(format=p.get_format_from_width(wf.getsampwidth()),
                    channels=wf.getnchannels(),
                    rate=wf.getframerate(),
                    output=True)

    # read data (based on the chunk size)
    data = wf.readframes(chunk)

    # play stream (looping from beginning of file to the end)
    while data != '':
        # writing to the stream is what *actually* plays the sound.
        stream.write(data)
        data = wf.readframes(chunk)

        # cleanup stuff.
    stream.close()
    p.terminate()
```

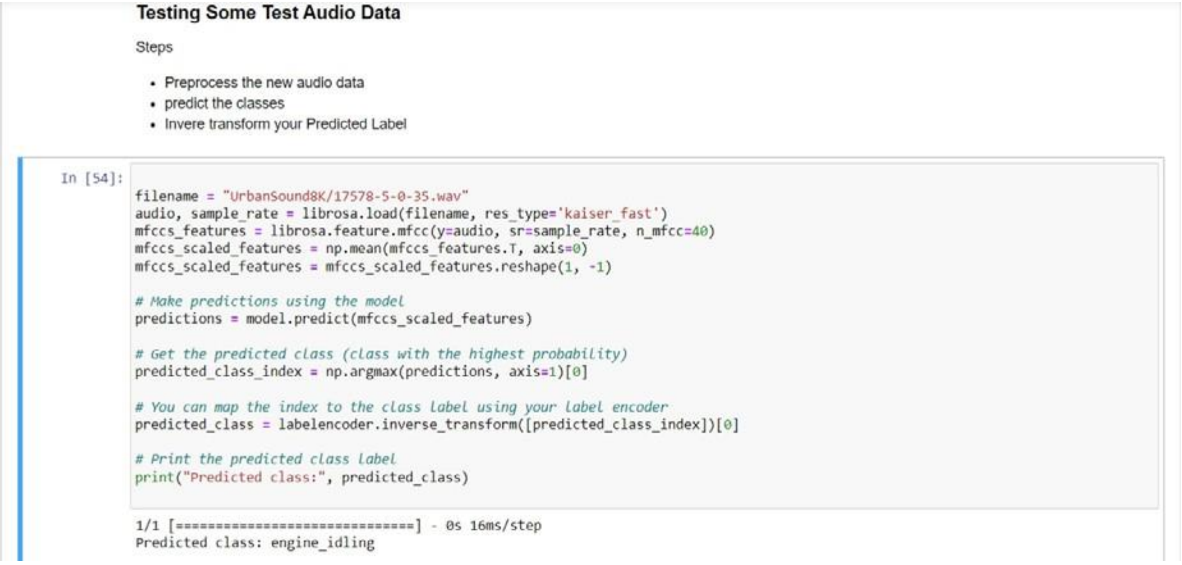
TESTING

The testing phase is an important part of software development. It is the Information zed system will help in automate process of finding errors and missing operations and also a complete verification to determine whether the objectives are met and the user requirements are satisfied. Software testing is carried out in three steps:

1. The first includes unit testing, where in each module is tested to provide its correctness, validity and also determine any missing operations and to verify whether theobjectives have been met. Errors are noted down and corrected immediately.

2. Unit testing is the important and major part of the project. So errors are rectified easily in particular module and program clarity is increased. In this project entire system is divided into several modules and is developed individually. So unit testing is conducted to individual modules.

3. The second step includes Integration testing. It need not be the case, the software whose modules when run individually and showing perfect results, will also show perfect results when run as a whole.



Testing Some Test Audio Data

Steps

- Preprocess the new audio data
- predict the classes
- Invere transform your Predicted Label

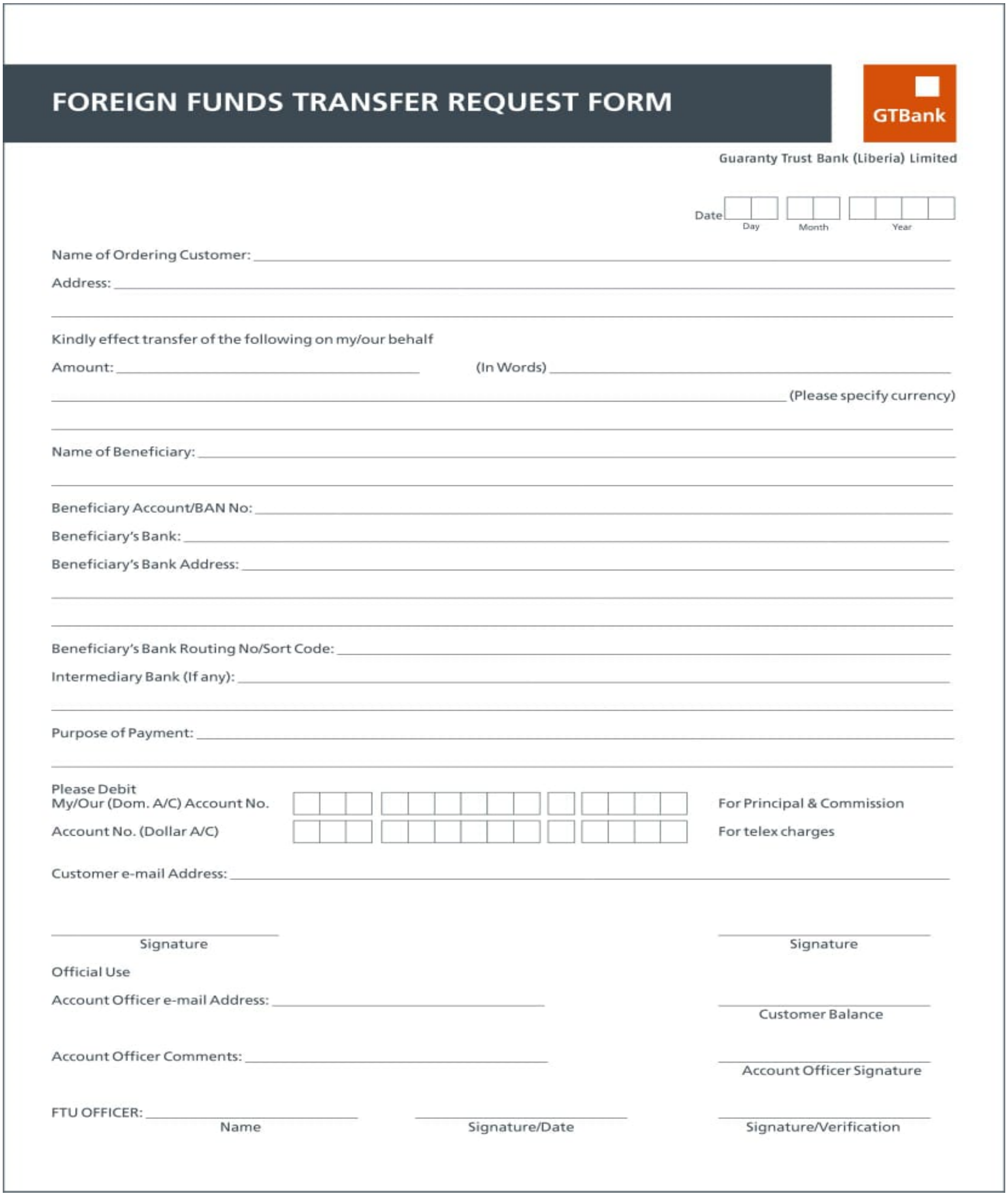
In [54]:

```
filename = "UrbanSound8K/17578-5-0-35.wav"
audio, sample_rate = librosa.load(filename, res_type='kaiser_fast')
mfccs_features = librosa.feature.mfcc(y=audio, sr=sample_rate, n_mfcc=40)
mfccs_scaled_features = np.mean(mfccs_features.T, axis=0)
mfccs_scaled_features = mfccs_scaled_features.reshape(1, -1)

# Make predictions using the model
predictions = model.predict(mfccs_scaled_features)

# Get the predicted class (class with the highest probability)
predicted_class_index = np.argmax(predictions, axis=1)[0]

# You can map the index to the class label using your label encoder
predicted_class = labelencoder.inverse_transform([predicted_class_index])[0]

# Print the predicted class label
print("Predicted class:", predicted_class)
```

```
1/1 [==============================] - 0s 16ms/step
Predicted class: engine_idling
```

Here are one of the test cases run in the windows software and hence it is proven that the code would run 80% accuracy and identify the sounds given in the code.

8. SNAPSHOTS



FOREIGN FUNDS TRANSFER REQUEST FORM

GTBank

Guaranty Trust Bank (Liberia) Limited

Date □□ □□ □□□□
Day Month Year

Name of Ordering Customer: ____________________

Address: ____________________

____

Kindly effect transfer of the following on my/our behalf

Amount: ____________________ (In Words) ____________________

____________________ (Please specify currency)

____

Name of Beneficiary: ____________________

____

Beneficiary Account/BAN No: ____________________

Beneficiary's Bank: ____________________

Beneficiary's Bank Address: ____________________

____

____

Beneficiary's Bank Routing No/Sort Code: ____________________

Intermediary Bank (If any): ____________________

____

Purpose of Payment: ____________________

____

Please Debit My/Our (Dom. A/C) Account No.	□□□ □□□□□□ □ □□□□	For Principal & Commission
Account No. (Dollar A/C)	□□□ □□□□□□ □ □□□□	For telex charges

Customer e-mail Address: ____________________

____________________ Signature ____________________ Signature

Official Use

Account Officer e-mail Address: ____________________

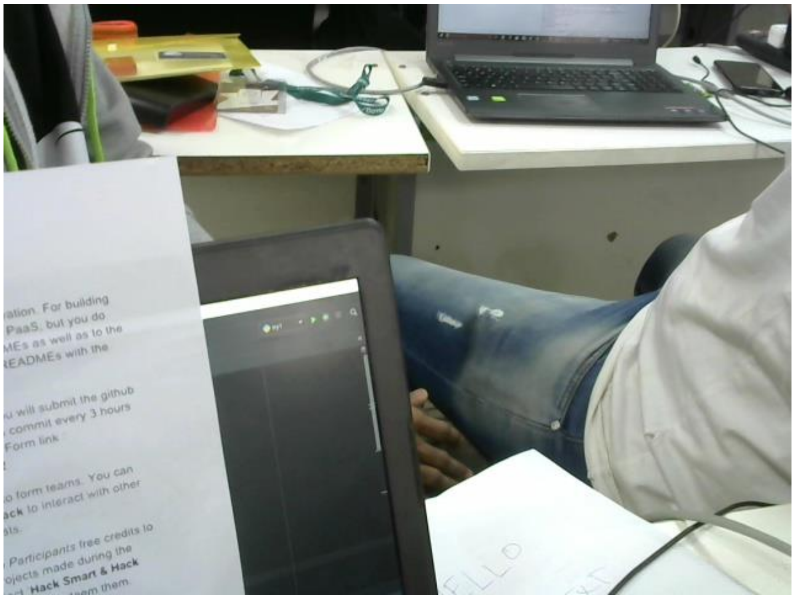
____________________ Customer Balance

Account Officer Comments: ____________________

____________________ Account Officer Signature

FTU OFFICER: ____________________ Name ____________________ Signature/Date ____________________ Signature/Verification

Installing Librosa which is a python package used for Audio analysis Under Voice classification Exploratory Data Analysis (EDA).



Installing matplotlib for displaying signal of a file.

9. CONCLUTION

The package was designed in such a way that future modifications can be done easily. The following conclusions can be deduced from the development of the project:

- Automation of the entire system improves the efficiency

- It provides a friendly graphical user interface which proves to be better when compared to the existing system.

- It gives appropriate access to the authorized users depending on their permissions.

- It effectively overcomes the delay in communications.

- Updating of information becomes so easier

- System security, data security and reliability are the striking features.

- The System has adequate scope for modification in future if it is necessary.

10. REFERENCE

Web References:

1. **www.alphaworks.ibm**.com
2. **www.github.com.**
3. **www.aw.com/titles.**
4. *www.urbansounddataset.com.*