

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 COMPUTER SCIENCE

Submitted by:

Bishwajit Paul

1TJ19CS014



COMPSOFT TECHNOLOGIES



Department of computer science and engineering

T. John Institute of Technology

88/1, Gottigere, Bannerghatta Road, bengaluru-560083

CERTIFICATE

This is to certify that the Internship titled “**Virtual Assistant for visually impaired**” carried out by **Mr. Bishwajit Paul**, a bonafide student of T. John Institute of Technology, in partial fulfillment for the award of **Bachelor of Engineering**, in **COMPUTER SCIENCE AND ENGINEERING** under Visvesvaraya Technological University, Belagavi, during the year 2022-2023. 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

DECLARATION

I, Bishwajit Paul, final year student of Computer Science and Engineering, T. John Institute of Technology - 560083, 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 Computer Science and Engineering, during the academic year 2022-2023.

Date: 25-09-2022

Place: Bangaluru

USN: 1TJ19CS014

NAME: BISHWAJIT PAUL

OFFER LETTER



Date: 23rd August, 2022

Name: Bishwajit Paul
USN: 1TJ19CS014

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 **23rd August, 2022**. 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 would like to express our gratitude to our chairman **Dr. Thomas P. John** for providing us with the necessary facilities for the successful completion of the project.

We express our sincere thanks to our principal **Dr. Suresh P. Venugopal** for providing us adequate facilities to undertake this Internship.

We would like to thank our Head of Dept **Prof. Suma R**, for providing us an opportunity to carry out Internship and for his valuable guidance and support.

We would like to thank our (Lab assistant name) 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, for helping 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.

Bishwajit Paul

1TJ19CS014

ABSTRACT

One of the most used forms of communication among people is Email. A lot of confidential and urgent information is exchanged over emails in today's time. There are about 253 million visually impaired people worldwide. These visually impaired people are facing a problem of communication. Since technology is growing day by day, these visually challenged people feel that they are more challenged.

So, we proposed a Voice-based Email System using AI that will make the email system very easily accessible to visually challenged people and also help society. Accessibility is the most important feature that is considered while developing this system.

Table of Contents

Sl no	Description	Page no
1	Company Profile	8
2	About the Company	9
3	Introduction	12
4	System Analysis	15
5	Requirement Analysis	17
6	Design Analysis	18
7	Implementation	20
8	Snapshots	21
9	Conclusion	24
10	References	25

CHAPTER 1

COMPANY PROFILE

A Brief History of Compsoft Technologies

Compsoft Technologies, 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.

Sarvamoola Software Services. 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.

Compsoft Technologies, 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. Compsoft Technologies work with their clients and help them to define 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 it in one sentence” Technology helps you to Delight your customers” and that is what we want to achieve

CHAPTER 2

ABOUT THE COMPANY



Compsoft Technologies 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, 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 stakeholder 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.

Products of Compsoft Technologies.

Android Apps

It is the process by which new applications are created for devices running the Android operating system. Applications are usually developed in Java (and/or Kotlin; or other such option) programming language using the Android software development kit (SDK), but other development environments are also available, some such as Kotlin support the exact same Android APIs (and bytecode), while others such as Go have restricted API access.

The Android software development kit includes a comprehensive set of development tools. These include a debugger, libraries, a handset emulator based on QEMU, documentation, sample code, and zutorials. Currently supported development platforms include computers running Linux (any modern desktop Linux distribution), Mac OS X 10.5.8 or later, and Windows 7 or later. As of March 2015, the SDK is not available on Android itself, but softwaredevelopment is possible by using specialized Android applications.

Web Application

It is a client-server computer program in which the client (including the user interface and client- side logic) runs in a web browser. Common web applications include web mail, online

retail sales, online auctions, wikis, instant messaging services and many other functions. web applications use web documents written in a standard format such as HTML and JavaScript, which are supported by a variety of web browsers. Web applications can be considered as a specific variant of client–server software where the client software is downloaded to the client machine when visiting the relevant web page, using standard procedures such as HTTP. The Client web software updates may happen each time the web page is visited. During the session, the web browser interprets and displays the pages, and acts as the universal client for any web application. The use of web application frameworks can often reduce the number of errors in a program, both by making the code simpler, and by allowing one team to concentrate on the framework while another focuses on a specified usecase. In applications which are exposed to constant hacking attempts on the Internet, security-related problems can be caused by errors in the program.

Frameworks can also promote the use of best practices such as GET after POST. There are some who view a web application as a two-tier architecture. This can be a “smart” client that performs all the work and queries a “dumb” server, or a “dumb” client that relies on a “smart” server. The client would handle the presentation tier, the server would have the database (storage tier), and the business logic (application tier) would be on one of them or on both. While this increases the scalability of the applications and separates the display and the database, it still doesn’t allow for true specialization of layers, so most applications will outgrow this model. An emerging strategy for application software companies is to provide web access to software previously distributed as local applications. Depending on the type of application, it may require the development of an entirely different browser-based interface, or merely adapting an existing application to use different presentation technology. These programs allow the user to pay a monthly or yearly fee for use of a software application without having to install it on a local hard drive. A company which follows this strategy is known as an application service provider (ASP), and ASPs are currently receiving much attention in the software industry.

Security breaches on these kinds of applications are a major concern because it can involve both enterprise information and private customer data. Protecting these assets is an important part of any web application and there are some key operational areas that must be included in the development process. This includes processes for authentication, authorization, asset handling, input, and logging and auditing. Building security into the applications from the beginning can be more effective and less disruptive in the long run.

Web design

It is encompassing many different skills and disciplines in the production and maintenance of websites. The different areas of web design include web graphic design; interface design; authoring, including standardized code and proprietary software; user experience design; and

search engine optimization. The term web design is normally used to describe the design process relating to the front-end (client side) design of a website including writing mark up. Web design partially overlaps web engineering in the broader scope of web development. Web designers are expected to have an awareness of usability and if their role involves creating markup then they are also expected to be up to date with web accessibility guidelines. Web design partially overlaps web engineering in the broader scope of web development.

Departments and services offered

Compsoft Technologies plays an essential role as an institute, the level of education, development of student's skills are based on their trainers. If you do not have a good mentor then you may lag in many things from others and that is why we at Compsoft Technologies gives you the facility of skilled employees so that you do not feel unsecured about the academics. Personality development and academic status are some of those things which lie on mentor's hands. If you are trained well then you can do well in your future and knowing its importance of Compsoft Technologies always tries to give you the best.

They have a great team of skilled mentors who are always ready to direct their trainees in the best possible way they can and to ensure the skills of mentors we held many skill development programs as well so that each and every mentor can develop their own skills with the demands of the companies so that they can prepare a complete packaged trainee.

Services provided by Compsoft Technologies.

- Core Java and Advanced Java
- 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

This is an innovative System for visually impaired people and acts as a voice assistant for them. This system is used to help the visually impaired to have access to the most important features of the phone enhancing the quality of the system making use of different custom layouts and using speech to text. The System has custom messaging feature with inbox and sent items, call log and dialer, notes and battery level checking and reminder. All actions performed by the user the system speaks out and helps the user to know his current position. The System helps the user to also read the contents of the message along with the sender and the date and time, in whole everything .

The system also allows the user to note few things with its custom note pad. The System speaks out the dialer number pressed and called notification also. The System in all is a voice assistant for whatever action the user has performed through a custom app while taking the data from the default application. The custom app doesn't save any data it is dependent on the phones data .

Artificial Intelligence is based on how any device perceives its Environment and takes actions based on the perceived data to achieve the result successfully. It is the study of intelligent agents. Artificial Intelligence gives the ability of thinking and behaving like humans to a computer. A chatbot is also known as a Talbot, chatterbot, Bot, interactive agent etc. is a computer program which conducts a conversation via voice or text methods and thereby it acts as a conversational partner rather than humans. For various practical purposes like customer service or information acquisition, healthcare chatbot is being used now a days. Mostly chatbots use natural language processing for interpreting the user input and generate the corresponding response but certain simpler systems search for the keyword within the text and then provide a reply based on the matching keywords or certain pattern. Trending, chatbots are part of virtual assistants such as Google Assistant, Siri, and Alexa are being used by many organizations' apps, websites, and on instant messaging platforms. Non-assistant applications include chatbots used for entertainment purposes, for research, and social bots which promote a particular product, candidate, or issue. Chatbot's are such kind of computer programs that interact with users using natural languages. For all kind of chatbots the flow is same, though each chatbot is specific in its own area knowledge that is one input from human is matched against the knowledge base of chatbot.

1. INTRODUCTION

Introduction to ML

Machine learning (ML) is a branch of artificial intelligence (AI) that enables computers to “self-learn” from training data and improve over time, without being explicitly programmed. Machine learning algorithms are able to detect patterns in data and learn from them, in order to make their own predictions. In short, machine learning algorithms and models learn through experience.

In traditional programming, a computer engineer writes a series of directions that instruct a computer how to transform input data into a desired output. Instructions are mostly based on an IF-THEN structure: when certain conditions are met, the program executes a specific action.

Machine learning, on the other hand, is an automated process that enables machines to solve problems with little or no human input, and take actions based on past observations.

While artificial intelligence and machine learning are often used interchangeably, they are two different concepts. AI is the broader concept – machines making decisions, learning new skills, and solving problems in a similar way to humans – whereas machine learning is a subset of AI that enables intelligent systems to autonomously learn new things from data.

Instead of programming machine learning algorithms to perform tasks, you can feed them examples of labelled data (known as training data), which helps them make calculations, process data, and identify patterns automatically.

Put simply, Google’s Chief Decision Scientist describes machine learning as a fancy labelling machine. After teaching machines to label things like apples and pears, by showing them examples of fruit, eventually they will start labelling apples and pears without any help – provided they have learned from appropriate and accurate training examples.

Machine learning can be put to work on massive amounts of data and can perform much more accurately than humans. It can help you save time and money on tasks and analyses, like solving customer pain points to improve customer satisfaction, support ticket automation, and data mining from internal sources and all over the internet.

Problem Statement

Imbalance between health workforce and patients: It is a major challenge in healthcare sector, as imbalances between health workforce or lower health professionals leads to lower quality and productivity of medical services, increasing wait time, diversion of emergency department of patients.

Human errors in medical diagnosis: Diagnostic Error is concern with the failure of establish an accurate and timely explanation of the patient's health problem or that explanation to the patient. This can cause serious health risk.

Increasing individual healthcare expenses: This is the most common issue in healthcare sector

for consulting a healthcare professional diagnosis and medication cost are too high.

Real time health monitoring: There is not any system available that will monitor our health in real time because a smaller number of healthcare workforce that leads to lack of personal assistance of the patient health.

Unnecessary diagnostic tests: unnecessary diagnostic tests lead to expensive medical services. Not all people can afford that much high cost of health consultation.

Continuous shortage of nursing and technician staff: lack of access to healthcare, especially in rural areas. Currently available physicians for the general population have not been able to keep up with the current healthcare demand. Healthcare service providers have been pressured to take care of more patients than they can handle, leading to lower quality of care.

CHAPTER 4

4.1 SYSTEM ANALYSIS

1. Existing System:

Ask Question

- Brief Description: User or chatbot can ask any kind of questions.
- User: Any active user, chatbot.
- Input Data: question asked by the user.
- Output Data: reply by the chatbot
- Pre-conditions: chatbot should be active and connected to datafile.

Reply

- Brief Description: the reply chatbot or user is framed upon the question asked by the user or chatbot.
- User: Chatbot any active user.
- Input Data: question asked by the user or chatbot.
- Output Data: reply by the chatbot or user.
- Pre-conditions: chatbot should be active and connected to datafiles.

2. Proposed System

The proposed idea is to create a virtual assistant using Artificial Intelligence. Artificial intelligence integrated healthcare chatbot which will diagnose and provides the common health issues details before consulting a doctor. Healthcare chatbot system will use natural language processing and neural networks to train the medical database. To reduce the healthcare overall costs and improve accessibility to medical knowledge the healthcare chatbot is built. There are certain chatbots are existing that acts as medical reference books that helps the patients to know additional information about their disease and helps to improve their health. This chatbot system engages with patients using text-to-text conversation about their health issues and provides the personalized diagnosis based on their symptoms. Hence, individuals will have the knowledge about their health status and will get to know about right treatments

3. Objective of the System

Overall, we feel that working on this project provided us with a great learning experience. This project has completely solely by us so it taught us a lot in terms of time management and work load management. our favorite part of this project was working on an area of

computing that is really starting to grow and break into the mainstream market. With so many companies getting on board by creating their own chatbots, in a few years we will see it as an everyday thing. It has been great to work with soft Bots Framework that is still in development and more and more features being added each day. Regrettably, we feel that we did spend a lot of time working with chatbot python library before I moved on to Bot Framework, we feel that if we have been working with Bot Framework from the start, we would've achieved a lot more. The feedback from our project mentor at our mid-point presentation really set us in the right direction. In the future, we will remember to be realistic with the goals we set ourselves and to explore every available option to us before committing to one framework. Although this project was great to work on, we are disappointed that we did not get to implement any of the security features that we have learned about in my other modules completed in previous semesters. We have learned a lot in the security aspect of computing from our other modules but it has been a learning experience developing this application as it touched on areas that we have never worked with before which gave me a challenging but beneficial experience.

CHAPTER 5

5.1 REQUIREMENT ANALYSIS

Hardware Requirement Specification

- Pentium 200-MHz computer with a minimum of 64 MB of RAM
- Monitor with a refresh rate of at least 40Hz for a smooth GUI experience (optional).
- Intel core i5
- Ram :8 GB
- Hard disk: 1TB

Software Requirement Specification

- Jupyter Notebook.
- Google Collab.
- Sklearn
- Google Chrome or Microsoft Edge of latest version.
- Windows XP 11.

CHAPTER 6

6.1 DESIGN ANALYSIS

Technology selection:

We used the following tools to implement the project

- o Pandas
- o Sklearn
- o Matplotlib
- o Chatterbot, a conversational dialogue engine (for chatbot processing)
- o Natural language processing

Natural Language Processing:

Natural Language Processing (NLP) is the study of computer programs to understand human Language. The ultimate goal of NLP is to read, translate, understand and make sense of the human languages. Natural language processing will be used to understand the input (syntax) given by the user for automatic analysis and representation of human language that machine can understand. We will use datasets to train the model. It will perform the sentimental analysis of trained datasets and will diagnose the diseases based on input symptoms.

Chatterbot:

Chatterbot is a conversational dialog engine based on machine-learning algorithms. Chatterbot is built in Python which generates responses based on collections of already known conversations. The language independent design of Chatterbot be trained to speak any language. It is simple and easy to use for building a chatbot.

Python:

Python is an interpreted, high-level, general-purpose programming language. It provides good support with a lot of modules which could be used as a cookbook to use and implement to in further. It gets integrate with functionalities of AI quite well.

NumPy:

NumPy is a Python library used for working with arrays. It also has functions for working in domain of linear algebra, fourier transform, and matrices. NumPy was created in 2005 by Travis Oliphant. It is an open-source project and you can use it freely. NumPy stands for Numerical Python.

Sklearn:

Sklearn is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction via a consistence interface in Python.

CHAPTER 7

7.1 IMPLEMENTATION

Implementation is the stage where the theoretical design is turned into a working system. The most crucial stage in achieving a new successful system and in giving confidence on the new system for the users that it will work efficiently and effectively.

The system can be implemented only after thorough testing is done and if it is found to work according to the specification. It involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the change over and an evaluation of change over methods as a part from planning.

Two major tasks of preparing the implementation are education and training of the users and testing of the system. The more complex the system being implemented, the more involved will be the system analysis and design effort required just for implementation.

The implementation phase comprises of several activities. The required hardware and software acquisition is carried out. The system may require some software to be developed. For this, programs are written and tested. The user then changes over to his new fully tested system and the old system is discontinued.

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 the objectives 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.

CHAPTER 8

8.1 SNAPSHOTS

```
main.py - C:\Users\Thanushree shivaram\AppData\Local\Programs\Python\Python310\Scripts\DobaraMatPuchana\main.py (3.10.7)
File Edit Format Run Options Window Help

import functions
import yolopy
import speech
import cv2
import os
import detect
import datetime
os.environ["GOOGLE_APPLICATION_CREDENTIALS"] = "dfKey.json"

labelsPath = "yolo/coco.names"
weightsPath = "yolo/yolov3.weights"
configPath = "yolo/yolov3.cfg"
args = {"threshold":0.3, "confidence":0.5}
project_id = "blindbot-4f356"
#project_id = "blindbot-286ed"
engine = speech.speech_to_text()

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 != None):
            intent, text = detect.detect_intent_texts(project_id, 0, [resp], 'en')
            if (intent == 'Jyoti' and resp!=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!=None):
            print(resp)
            intent, text = detect.detect_intent_texts(project_id, 0, [resp], 'en')
            if intent == 'Describe':
                detect.describeScene(cam, model, engine)
            elif intent == 'endconv':
                print(text)
                listening = False
                engine.text_speech(text)
            elif intent == 'Brightness':
                engine.text_speech("It is {} outside".format((functions.getBrightness(cam)) [0]))
```

Fig 1: main.py

```
detect.py - C:\Users\Thanushree shivaram\AppData\Local\Programs\Python\Python310\Scripts\DobaraMatPuchana\detect.py (3.10.7)
File Edit Format Run Options Window Help

import io
from google.oauth2 import service_account
from google.cloud import vision
import dialogflow_v2 as dialogflow
import cv2

def detect_text(cam, engine):
    credentials = service_account.Credentials.from_service_account_file('aj.json')
    client = vision.ImageAnnotatorClient(credentials=credentials)
    ret, content = cam.read()
    cv2.imwrite('op.jpg', content)
    with io.open('op.jpg', 'rb') as image_file:
        content = image_file.read()
    image = vision.types.Image(content=content)
    response = client.text_detection(image=image)
    texts = response.text_annotations
    print(len(texts))
    print('Text:')
    textm = ""
    for i, text in enumerate(texts):
        engine.text_speech(text.description)
        textm += text.description
        textm = textm + " "
    print(textm)

def detect_form(cam, engine):
    credentials = service_account.Credentials.from_service_account_file('aj.json')
    client = vision.ImageAnnotatorClient(credentials= credentials)
    #content = cam.read()
    path = 'bank.jpg'
    with io.open(path, 'rb') as image_file:
        content = image_file.read()
    image = vision.types.Image(content=content)
    response = client.text_detection(image=image)
    texts = response.text_annotations

    print('Text:')
    textm = ""
    for i, text in enumerate(texts):
        if (i==0):
            engine.text_speech("The form is entitled as")
        if (i==1):
            engine.text_speech("The form asks about these details")
            engine.text_speech(text.description)
            if ("Official" in text.description):
                break
```

Fig 2: detect.py

```
yolopy.py - C:\Users\Thanushree shivaram\AppData\Local\Programs\Python\Python310\Scripts\DobaraMatPuchana\yolopy.py (3.10.7)
File Edit Format Run Options Window Help

import cv2
import time
import numpy as np

class yolo:
    def __init__(self, labelsPath, weightsPath, configPath):
        # load the COCO class labels our YOLO model was trained on
        self.LABELS = open(labelsPath).read().strip().split("\n")
        # initialize a list of colors to represent each possible class label
        np.random.seed(42)
        self.COLORS = np.random.randint(0, 255, size=(len(self.LABELS), 3),
                                                    dtype="uint8")
        # derive the paths to the YOLO weights and model configuration
        # load our YOLO object detector trained on COCO dataset (80 classes)
        print("[INFO] loading YOLO from disk...")
        self.net = cv2.dnn.readNetFromDarknet(configPath, weightsPath)
        # determine only the 'output' layer names that we need from YOLO
        ln = self.net.getLayerNames()
        self.ln = [ln[i] - 1 for i in self.net.getUnconnectedOutLayers()]

    def detectYolo(self, frame, args):
        (H, W) = frame.shape[:2]
        # construct a blob from the input image and then perform a forward
        # pass of the YOLO object detector, giving us our bounding boxes and
        # associated probabilities
        blob = cv2.dnn.blobFromImage(frame, 1 / 255.0, (416, 416),
                                    swapRB=True, crop=False)
        self.net.setInput(blob)
        start = time.time()
        layerOutputs = self.net.forward(self.ln)
        end = time.time()
        print("[INFO] YOLO took {:.6f} seconds".format(end - start))
        # initialize our lists of detected bounding boxes, confidences, and
        # class IDs, respectively
        boxes = []
        confidences = []
        classIDs = []

        # loop over each of the layer outputs
        for output in layerOutputs:
            # loop over each of the detections
            for detection in output:
                # extract the class ID and confidence (i.e., probability) of
                # the current object detection
                scores = detection[5:]
                classID = np.argmax(scores)
                confidence = scores[classID]
```

Fig 3: fuctions.py

```
speech.py - C:\Users\Thanushree shivaram\AppData\Local\Programs\Python\Python310\Scripts\DobaraMatPuchana\speech.py (3.10.7)
File Edit Format Run Options Window Help

import speech_recognition as sr
import pyttsx3
from google.oauth2 import service_account
import json

from nltk.stem.porter import PorterStemmer
from nltk.stem.wordnet import WordNetLemmatizer
import re
import nltk
from nltk.corpus import stopwords

from nltk.tokenize import RegexpTokenizer
from nltk.stem.wordnet import WordNetLemmatizer

class speech_to_text():
    def __init__(self):
        en_voice_id = "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Speech\Voices\Tokens\TTS_MS_EN-US_ZIRA_11.0"
        ru_voice_id = "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Speech\Voices\Tokens\TTS_MS_RU-RU_IRINA_11.0"
        self.recognizer = sr.Recognizer()
        self.microphone = sr.Microphone()
        self.engine = pyttsx3.init()
        self.engine.setProperty('voice', en_voice_id)
        self.credentials = service_account.Credentials.from_service_account_file('api-key.json')

    def recognize_speech_from_mic(self):
        print("Start...")
        with self.microphone as source:
            self.recognizer.adjust_for_ambient_noise(source)
            audio = self.recognizer.listen(source)
            print("Found mic")
            response = {
                "success": True,
                "error": None,
                "transcription": None
            }
            try:
                response["transcription"] = self.recognizer.recognize_google(audio)
            except sr.RequestError:
                # API was unreachable or unresponsive
                response["success"] = False
                response["error"] = "API unavailable"
            except sr.UnknownValueError:
                # speech was unintelligible
                response["error"] = "Unable to recognize speech"
            if (response["transcription"] == "None"):
                print("Speech not detected! Pls try again!")
```

Fig 4: speech.py

```
yolopy.py - C:\Users\Thanushree shivaram\AppData\Local\Programs\Python\Python310\Scripts\DoBarMatPuchana\yolopy.py (3.10.7)
File Edit Format Run Options Window Help

import cv2
import time
import numpy as np

class yolo:
    def __init__(self, labelsPath, weightsPath, configPath):
        # load the COCO class labels our YOLO model was trained on
        self.LABELS = open(labelsPath).read().strip().split("\n")
        # initialize a list of colors to represent each possible class label
        np.random.seed(42)
        self.COLORS = np.random.randint(0, 255, size=(len(self.LABELS), 3),
                                                dtype="uint8")
        # derive the paths to the YOLO weights and model configuration
        # load our YOLO object detector trained on COCO dataset (80 classes)
        print("[INFO] loading YOLO from disk...")
        self.net = cv2.dnn.readNetFromDarknet(configPath, weightsPath)
        # determine only the 'output' layer names that we need from YOLO
        ln = self.net.getLayerNames()
        self.ln = [ln[i][0] - 1 for i in self.net.getUnconnectedOutLayers()]

    def detectYolo(self, frame, args):
        (H, W) = frame.shape[:2]
        # construct a blob from the input image and then perform a forward
        # pass of the YOLO object detector, giving us our bounding boxes and
        # associated probabilities
        blob = cv2.dnn.blobFromImage(frame, 1 / 255.0, (416, 416),
                                     swapRB=True, crop=False)
        self.net.setInput(blob)
        start = time.time()
        layerOutputs = self.net.forward(self.ln)
        end = time.time()
        print("[INFO] YOLO took {:.6f} seconds".format(end - start))
        # initialize our lists of detected bounding boxes, confidences, and
        # class IDs, respectively
        boxes = []
        confidences = []
        classIDs = []

        # loop over each of the layer outputs
        for output in layerOutputs:
            # loop over each of the detections
            for detection in output:
                # extract the class ID and confidence (i.e., probability) of
                # the current object detection
                scores = detection[5:]
                classID = np.argmax(scores)
```

Fig 5: yolopy.py

CHAPTER 9

9.1 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.

CHAPTER 10

10.1 REFERENCE

1. HameedullahKazi,B.S. Chowdhry,ZeeshaMemon, "MedChatBot: An UMLS based Chatbot for Medical Students", International Journal of Computer Applications (0975 – 8887)Volume 55– No.17, October 2016.
2. BenildaEleonor V. Comendador, Bien Michael B. Francisco, Jefferson S. Medenilla, Sharleen Mae T. Nacion, and Timothy Bryle E. Serac, "Pharmabot: A Pediatric Generic Medicine Consultant Chatbot ", Journal of Automation and Control Engineering Vol. 3, No. 2, April 2015.
3. <https://stackoverflow.com>
4. Gillian Cameron, David Cameron, Gavin Megaw, Raymond Bond,MauriceMulvenna ,Siobhan O'Neill, Cherie Armour, Michael McTear,"Towards a chatbot for digital counselling",Journal of Medical Internet Research, 4(1)
5. <https://github.com>