!pip install SpeechRecognition

!pip install gTTS

!pip install transformers

!pip install tensorflow

!pip install pipwin

!pipwin install pyaudio

import speech\_recognition as sr

from gtts import gTTS

import transformers

import os

import datetime

import numpy as np

class ChatBot():

    def \_init\_(self, name):

        print("--- starting up", name, "---")

[self.name](http://self.name/) = name

    def speech\_to\_text(self):

        recognizer = sr.Recognizer()

        with sr.Microphone() as mic:

            print("listening...")

            audio = recognizer.listen(mic)

        try:

            self.text = recognizer.recognize\_google(audio)

            print("me --> ", self.text)

        except:

            print("me -->  ERROR")

    @staticmethod

    def text\_to\_speech(text):

        print("ai --> ", text)

        speaker = gTTS(text=text, lang="en", slow=False)

        speaker.save("res.mp3")

        os.system("afplay res.mp3")  #mac->afplay | windows->start

        os.remove("res.mp3")

    def wake\_up(self, text):

        return True if [self.name](http://self.name/) in text.lower() else False

    @staticmethod

    def action\_time():

        return datetime.datetime.now().time().strftime('%H:%M')

if \_name\_ == "\_main\_":

    ai = ChatBot(name="maya")

    nlp = transformers.pipeline("conversational", model="microsoft/DialoGPT-medium")

    os.environ["TOKENIZERS\_PARALLELISM"] = "true"

    while True:

        ai.speech\_to\_text()

        if ai.wake\_up(ai.text) is True:

            res = "Hello I am Rohith the AI, what can I do for you?"

        elif "time" in ai.text:

            res = ai.action\_time()

        elif any(i in ai.text for i in ["thank","thanks"]):

            res = np.random.choice(["you're welcome!","anytime!","no problem!","cool!","I'm here if you need me!"])

        else:

            chat = nlp(transformers.Conversation(ai.text), pad\_token\_id=50256)

            res = str(chat)

            res = res[res.find("bot >> ")+6:].strip()

        ai.text\_to\_speech(res)