# PHASE 3

# FROM SIRI TO SMARTYPANTS: BUILDING YOUR PERSONALIZED AI ASSISTANT

**COLLEGE CODE: 7100**

**COLLEGE NAME: ANNA UNIVERSITY REGIONAL CAMPUS, COIMBATORE.**

**TECHNOLOGY: ARTIFICIAL INTELLIGENCE**

**TOTAL NUMBER OF STUDENTS IN GROUP: 4**

**STUDENT'S DETAILS WITHIN THE GROUP:**

**1) KATHIR S (710022106005)**

**2) KAAVYASRIE V T (710022106019)**

**3) THANUSHREE M (710022106031)**

**4) HAMSAVENI M (710022106010)**

**IV SEMESTER**

**SUBMITTED BY,**

**KATHIR S (710022106005)**

**aut7100ECE32**

**IV SEMESTER**

# FROM SIRI TO SMARTYPANTS: BUILDING YOUR PERSONALIZED AI ASSISTANT

## ABSTRACT

The project "From Siri to SmartyPants: Building Your Personalized AI Assistant" introduces an innovative virtual assistant tailored explicitly for electronics enthusiasts and learners. This AI-powered assistant integrates cutting-edge technologies such as speech recognition, text-to-speech conversion, and extensive electronic component databases to offer a comprehensive platform for exploring electronics and fostering hands-on learning experiences. One of the primary highlights of "From Siri to SmartyPants: Building Your Personalized AI Assistant" is its comprehensive component database, covering a diverse range of electronic components, including resistors, capacitors, microcontrollers, and sensors. This vast repository of information enables users to delve deep into the functionalities, applications, and connections of each component, providing a solid foundation for understanding electronic circuits and systems.Moreover, "From Siri to SmartyPants: Building Your Personalized AI Assistant" stands out with its unique ability to generate project ideas based on the components users possess. By analyzing the compatibility and interconnections of these components, the assistant suggests innovative project proposals along with detailed connection instructions. This feature not only promotes creativity but also encourages users to apply their knowledge practically, fostering a hands-on learning approach.The interactive learning experience offered by "From Siri to SmartyPants: Building Your Personalized AI Assistant" further enhances its utility. Through natural language processing and speech synthesis capabilities, users can engage in seamless conversations with the assistant, asking questions, seeking explanations, and receiving real-time responses. This interactive dialogue facilitates a deeper understanding of electronics concepts, supports project development, and makes learning electronics more accessible and engaging.Overall, "From Siri to SmartyPants: Building Your Personalized AI Assistant" redefines the learning experience in electronics by providing a user-friendly platform, personalized assistance, and an interactive environment, making it an invaluable tool for electronics enthusiasts, learners, and professionals alike.

# INTRODUCTION

Our project, "Electro AI," represents a comprehensive endeavor aimed at developing an advanced and personalized AI assistant tailored specifically for individuals interested in electronics. Through the utilization of cutting-edge technologies such as artificial intelligence (AI), machine learning (ML), and natural language processing (NLP), we are creating an interactive and intelligent assistant capable of understanding user queries, providing relevant information, offering project suggestions, and assisting with various tasks related to electronics. This project spans across multiple phases, each focusing on enhancing different aspects of the assistant's functionality and intelligence. In the initial phases, we laid the groundwork by implementing fundamental features like speech recognition, basic knowledge integration, and task automation. These functionalities allowed users to interact with the assistant through voice commands, ask general questions about electronics, and receive assistance with basic tasks such as setting reminders and retrieving information from a predefined database. As the project progressed, we delved deeper into advanced technologies to improve the assistant's capabilities. Phase two involved refining the speech recognition system for better accuracy and responsiveness, enhancing the NLP algorithms to understand complex queries and user intents more effectively, and integrating machine learning models to provide personalized recommendations and insights based on user preferences and historical interactions. We expanded the assistant's database to include a wide range of electronic components, circuits, and project ideas. This enabled Electro AI to offer detailed information about specific components, suggest suitable projects based on user-supplied components, and provide step-by-step guidance on circuit connections and configurations. In the current phase, we are enhancing Electro AI's user interface and experience by developing a web-based platform. This platform will enable users to access the assistant from any device with internet connectivity, making it more accessible and convenient for users across different environments. Additionally, we are implementing real-time updates and feedback mechanisms to continuously improve the assistant's performance and relevance to user needs.

# OVERVIEW

"Electro AI" is an innovative virtual assistant designed for electronics enthusiasts and learners. It uses advanced technologies such as speech recognition, text-to-speech conversion, and extensive databases to provide a comprehensive platform for exploring electronic components and generating project ideas. Through interactive voice commands and responses, users can easily delve into the world of electronics. The assistant offers detailed explanations of various electronic components, their functionalities, and applications, making learning accessible and engaging.

1. **Comprehensive Component Database:** "Electro AI" features a vast database covering a wide range of electronic components, from resistors and capacitors to microcontrollers and sensors. This extensive information helps users understand the intricacies of each component and how they contribute to electronic circuits.

2. **Project Idea Generation:** A standout feature is the ability to suggest project ideas based on the components users have. By analyzing the compatibility and connections of these components, the assistant generates project proposals with detailed connection instructions, fostering creativity and hands-on learning.

3. **Interactive Learning Experience:** Using natural language processing and speech synthesis, the assistant creates an interactive learning environment. Users can ask questions, seek explanations, and receive real-time responses, enhancing their understanding of electronics concepts and aiding in project development.

# SOFTWARE INTERFACES

This project utilizes a variety of software interfaces to enhance development and user experience. Python, the core language, manages tasks like speech recognition and data handling. Visual Studio Code (VSCode) serves as the Integrated Development Environment (IDE), providing essential coding features and organization tools. Jupyter Notebook is employed for data analysis and visualization, creating interactive documents for efficient data exploration and reporting.

- **Operating System:** Windows 10 (64-bit), macOS (recent version), or Linux (e.g., Ubuntu)
- **Python (version 3.6 or later):**https://www.python.org/downloads/
- **Python Libraries:**
  1. SpeechRecognition:www.github.com/Uberi/speech_recognition
  2. pyttsx3:www.pyttsx3.readthedocs.io/en/latest/
  3. datetime: (Built-in with Python)
  4. wikipedia: www.pypi.org/project/wikipedia/
  5. webbrowser: (Built-in with Python)
  6. random: (Built-in with Python)

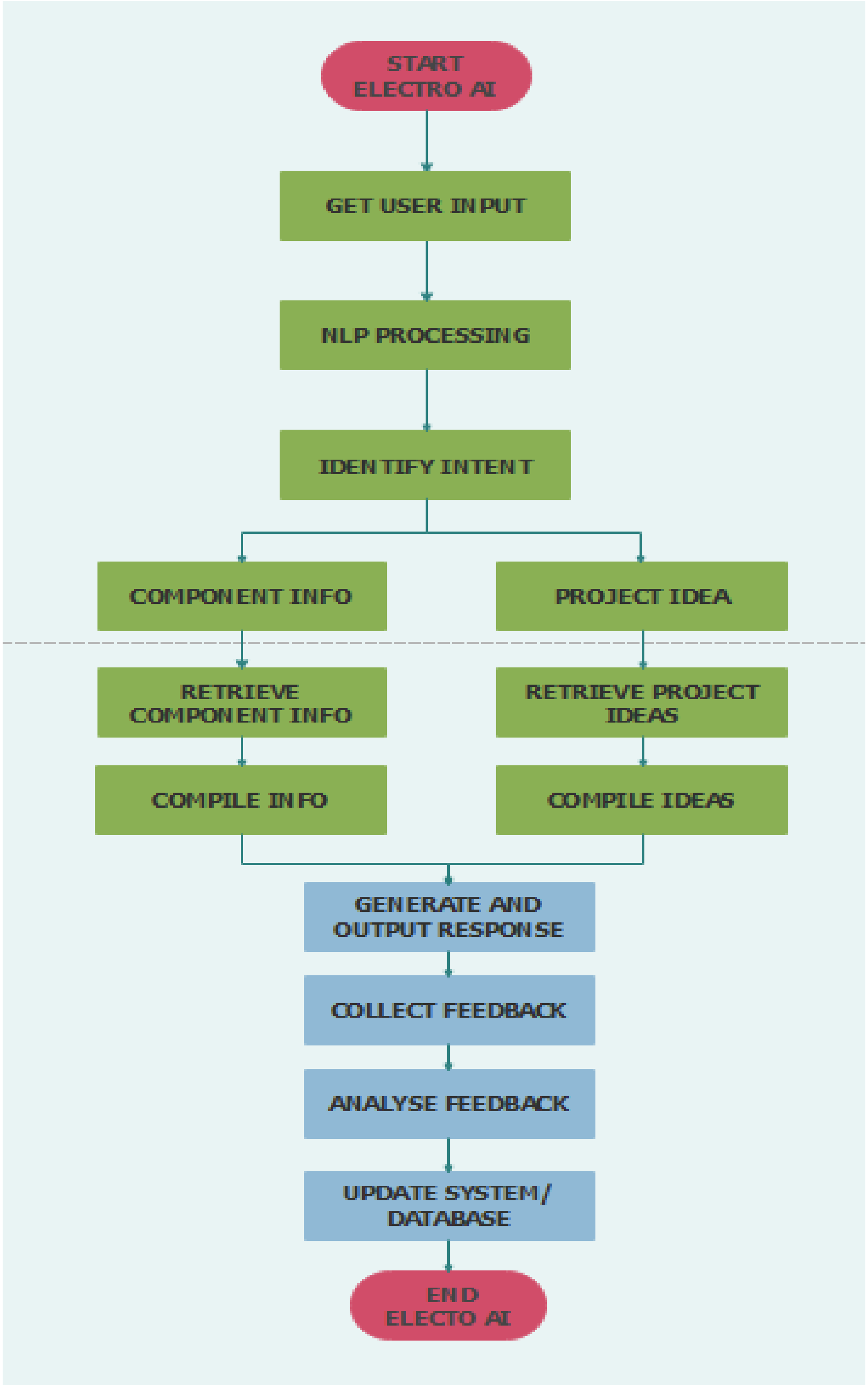
- **Text Editor or IDE (Integrated Development Environment) with Python Support:**

o Visual Studio Code: https://code.visualstudio.com/ (cross-platform)
o Jupyter Notebook: https://jupyter.org/ (web-based) System

## HARDWARE REQUIREMENTS:

- **Processor:** Intel Core i3 or equivalent (i5 or better recommended
- **RAM:** 4 GB minimum (8 GB or more recommended for larger datasets)
- **Hard Drive:** 20 GB free space (more space may be needed depending on dataset size)
- **Operating System:** Windows 10 (64-bit), macOS (recent version), or Linux (e.g., Ubuntu)
- **Internet Connection:** Optional, for downloading libraries and documentation

## FLOW CHART:


START
ELECTRO AI
GET USER INPUT
NLP PROCESSING
IDENTIFY INTENT
COMPONENT INFO
PROJECT IDEA
RETRIEVE COMPONENT INFO
RETRIEVE PROJECT IDEAS
COMPILE INFO
COMPILE IDEAS
GENERATE AND OUTPUT RESPONSE
COLLECT FEEDBACK
ANALYSE FEEDBACK
UPDATE SYSTEM/ DATABASE
END
ELECTO AI

## CODE IMPLEMENTATION:(SAMPLE CODE)

```
    import speech_recognition as sr
    #from flask import Flask, render_template, request, jsonify
     #This coding will be run normally if we comment the flask and if we uncomment,
     this will create a web server
    import pyttsx3
    import datetime
    import wikipedia
    import webbrowser
    import random
    import schedule
    from intentstrain import intents
    from projecttrain import projects
    import os
    import subprocess
#app = Flask(__name__)
# Initialize the text-to-speech engine
    engine = pyttsx3.init()
# Setting up properties for female voice
    voices = engine.getProperty('voices')
    engine.setProperty('voice', voices[1].id)
    #Uncomment the below if we uncomment flask in above
     """
     @app.route("/")
     def index():
         return render_template('index.html')
     @app.route("/get", methods=["POST"])
     """
```

```
 # Function Sections
 # Function to generate a response based on user input
#Function Sections
# Function to generate a response based on user input
def get_response(user_input):
    for intent, data in intents.items():
        for pattern in data["patterns"]:
            if pattern.lower() in user_input.lower():
                return random.choice(data["responses"])
    return "I'm sorry, I don't understand that."

# Function to suggest a project based on user components
def suggest_project(components):
    for project, details in projects.items():
        if all(comp.lower() in components.lower() for comp in details["components"]):
            response = f"{project}:\n{details['connection']}"
            return response
    return "I couldn't find a suitable project based on the components you provided."

# Function to speak out and print text
def speak_and_print(text, rate=150):
    print(text)
    engine.setProperty('rate', rate)
    engine.say(text)
    engine.runAndWait()
```

```
# Function to recognize speech
def take_command():
    r = sr.Recognizer()
    with sr.Microphone() as source:
        print("Listening...")
        r.pause_threshold = 1
        audio = r.listen(source)
    try:
        print("Recognizing...")
        query = r.recognize_google(audio, language='en-us')
        print(f"User said: {query}\n")
    except Exception as e:
        print(e)
        speak_and_print("Say that again please...")
        return "None"
    return query
# Function to wish user as per the current time
def wish_me():
    hour = datetime.datetime.now().hour
    if 0 <= hour < 12:
        speak_and_print("Good Morning!")
    elif 12 <= hour < 18:
        speak_and_print("Good Afternoon!")
    else:
        speak_and_print("Good Evening!")
    speak_and_print("I am Electro AI. How may I assist you today?")
# Main function to execute the program
if __name__ == "__main__":
    wish_me()
    while True:
        query = take_command().lower()
        # Logic for executing tasks based on query
        if 'what is' in query:
            query = query.replace("what is", "")
            try:
       results = wikipedia.summary(query, sentences=3)
                speak_and_print(results)
            except wikipedia.exceptions.PageError:
                speak_and_print("Sorry, I couldn't find any information on that topic.")
            except wikipedia.exceptions.DisambiguationError:
                speak_and_print("There are multiple possible meanings for this term.
Please be more specific.")
        elif query.startswith('open '):
            website = query.split('open ')[1]
            if not website.endswith('.com'):
                website += '.com'
            webbrowser.open(website)
            speak_and_print(f"Opening {website.capitalize()}")
        elif 'tell about you' in query:
            speak_and_print("I am Electronics Assistant Robot made by Kathir
Kaavyasrie Thanushree Hamsaveni...I will be very useful if you are a ECE
engineer")
        elif 'what will you do' in query:
            speak_and_print("I will help you to understand the concepts of Basics in
Electronics..But I am still in under development...But still I will give you the
connections of any circuits if you give or show just the components...")
```

```
    elif 'the time' in query:
        str_time = datetime.datetime.now().strftime("%H:%M:%S")
        speak_and_print(f"The time is {str_time}")
    elif 'shutdown' in query:
        os.system("shutdown /s /t 1")  # Shutdown the system
    elif 'project helper' in query:
        # Code for initiating the project suggestion process
        speak_and_print("Sure, I can help you with project ideas based on the components you have. Please list the components you have.")
        user_components = take_command()
        project_idea = suggest_project(user_components)
        speak_and_print(project_idea)
    elif 'app' in query:
        app_name = query.split('app ')[1].lower()  # Extract app name and convert to lowercase
        app_executables = {
  'calculator': 'C:\\Windows\\System32\\calc.exe',
  'notepad': 'C:\\Windows\\System32\\notepad.exe',
  'Edge': 'C:\\Program Files (x86)\\Microsoft\\Edge\\Application\\msedge.exe',
  'word': 'C:\\Program Files\\Microsoft Office\\root\\Office16\\WINWORD.EXE',
  'excel': 'C:\\Program Files\\Microsoft Office\\root\\Office16\\EXCEL.EXE',
  'powerpoint': 'C:\\Program Files\\Microsoft Office\\root\\Office16\\POWERPNT.EXE',
  'proteus': r'C:\Program Files (x86)\Labcenter Electronics\Proteus 8 Professional\BIN\PDS.EXE',
  'matlab': r'C:\Program Files\MATLAB\R2013a\bin\matlab.exe',
   'arduino':   r'C:\Users\Home\AppData\Local\Programs\Arduino   IDE\Arduino
 IDE.exe',
   'kicad': r'C:\Program Files\KiCad\8.0\bin\kicad.exe',
   'pcb': r'C:\Program Files\KiCad\8.0\bin\kicad.exe',
}
            if app_name in app_executables:
                app_executable = app_executables[app_name]
                subprocess.Popen(app_executable)  # Open the specified appelse
                speak_and_print(f"Opening {app_name.capitalize()} application")
            else:
                speak_and_print("App not found. Please specify a valid app name.")
        elif 'exit' in query or 'bye' in query:
            speak_and_print("Goodbye!")
            exit()
        else:
            response = get_response(query)
            speak_and_print(response)
```

## INTENTSTRAIN.PY:

The intent code encompasses various functions like recognizing user queries about components, suggesting project ideas, addressing general electronics questions, and gathering feedback.

```
"Resistor": {
        "patterns": ["resistor", "resistors"],
        "responses": ["A resistor is a passive two-terminal electronic component
that resists the flow of electric current. It is used to control the amount of
current in a circuit, voltage division, and as a load."]
```

```
    },
    "Capacitor": {
        "patterns": ["capacitor", "capacitors"],
        "responses": ["A capacitor is a passive two-terminal electronic component that stores electrical energy in an electric field. It is used for filtering, energy storage, coupling, and timing applications."]
    },
    "Inductor": {
        "patterns": ["inductor", "inductors"],
        "responses": ["An inductor is a passive electronic component that stores energy in a magnetic field when current flows through it. It is used in filtering, energy storage, and impedance matching."]
    },
    "Diode": {
        "patterns": ["diode", "diodes"],
        "responses": ["A diode is a semiconductor device that allows current to flow in one direction only. It is used in rectification, signal demodulation, and voltage regulation."]
    },
    "Transistor": {
        "patterns": ["transistor", "transistors"],
        "responses": ["A transistor is a semiconductor device used to amplify or switch electronic signals and electrical power. It is fundamental in digital and analog circuits, amplifiers, and switching applications."]
    },
```

As our AI Model is trained for more than 200 intents, here we specify only 5 intents for simplicity and efficiency.

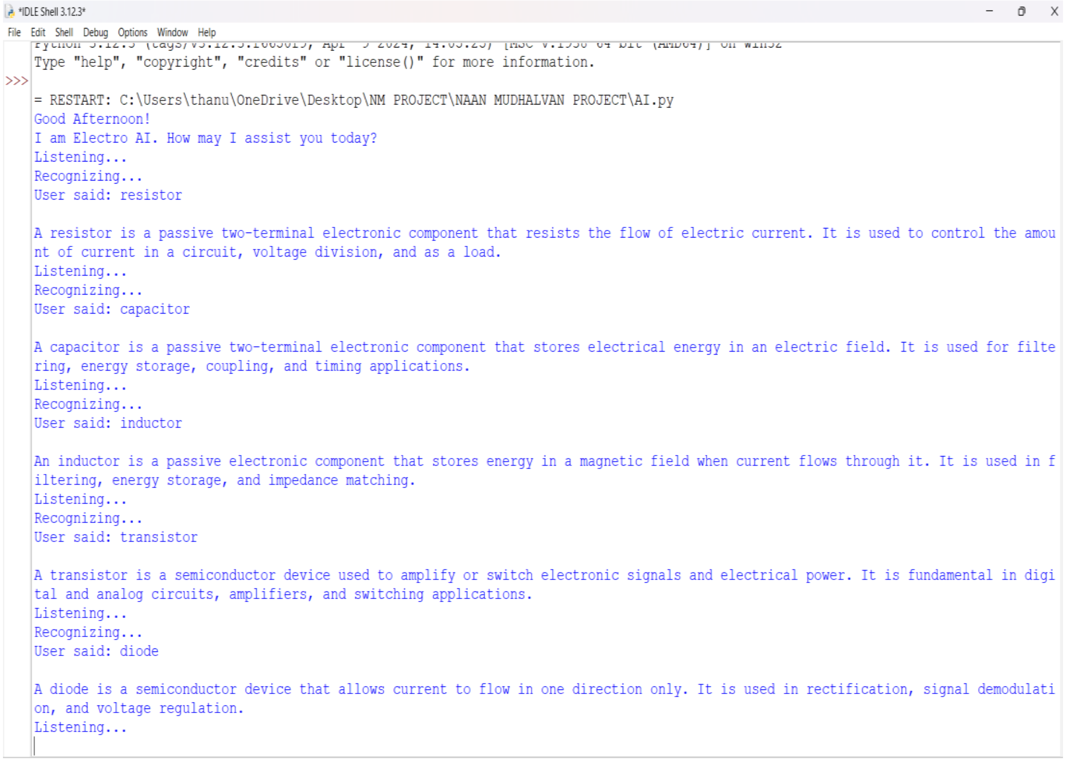
## PROJECTTRAIN.PY :

```
projects.update({
    "arduino sound sensor project": {
        "components": ["sound sensor", "arduino", "led"],
        "connection": "Components:\n- sound sensor: Connect the sound sensor's analog output pin to an analog pin (e.g., a0) on the Arduino.\n- led: Connect the LED's longer leg (anode, +) to a digital pin (e.g., pin 13) on the Arduino and the shorter leg (cathode, -) to a current-limiting resistor connected to ground (gnd).\n- arduino: Program the Arduino to respond to sound levels detected by the sensor."
    },
    "arduino ir remote control project": {
        "components": ["ir receiver module", "arduino", "ir remote"],
        "connection": "Components:\n- ir receiver module: Connect the module's vcc to Arduino's 5v pin, gnd to gnd pin, and out pin to a digital pin (e.g., pin 11) on the Arduino.\n- ir remote: Use the IR remote to send commands to the receiver module.\n- arduino: Program the Arduino to interpret IR signals from the remote and perform corresponding actions."
    },
    "arduino lcd display project": {
        "components": ["lcd display", "arduino", "potentiometer"],
        "connection": "Components:\n- lcd display: Connect the LCD display's vcc to Arduino's 5v pin, gnd to gnd pin, scl to a digital pin (e.g., pin 13), and sda to another digital pin (e.g., pin 12) on the Arduino.\n- potentiometer: Connect one end of the potentiometer to 5v, the other end to ground (gnd), and the wiper (middle pin) to the LCD display's v0 pin for contrast adjustment.\n- arduino: Program the Arduino to display text or data on the LCD."
```

    },
    "arduino temperature control project": {
        "components": ["lm35 temperature sensor", "arduino", "relay module"],
        "connection": "Components:\n- lm35 temperature sensor: Connect the sensor's vcc to Arduino's 5v pin, gnd to gnd pin, and output pin to an analog pin (e.g., a0) on the Arduino.\n- relay module: Connect the relay module's vcc to Arduino's 5v pin, gnd to gnd pin, and in1 (or control pin) to a digital pin (e.g., pin 2) on the Arduino.\n- arduino: Program the Arduino to read temperature data and control a device (e.g., fan, heater) using the relay based on temperature thresholds."
    },
    "arduino motor speed control project": {
        "components": ["dc motor", "arduino", "motor driver (l298n)", "potentiometer"],
        "connection": "Components:\n- dc motor: Connect the DC motor to the motor driver's output terminals (out1, out2 or out3, out4).\n Connect the motor driver's input pins (e.g., in1, in2) to digital pins on the Arduino.\n- motor driver (l298n): Supply power to the motor driver (vcc1, vcc2) and connect the ground (gnd) to the Arduino's ground.\n- potentiometer: Connect one end of the potentiometer to 5v, the other end to ground (gnd), and the wiper (middle pin) to an analog pin (e.g., a0) on the Arduino.\n- arduino: Program the Arduino to control the speed of the motor using the potentiometer."
    },

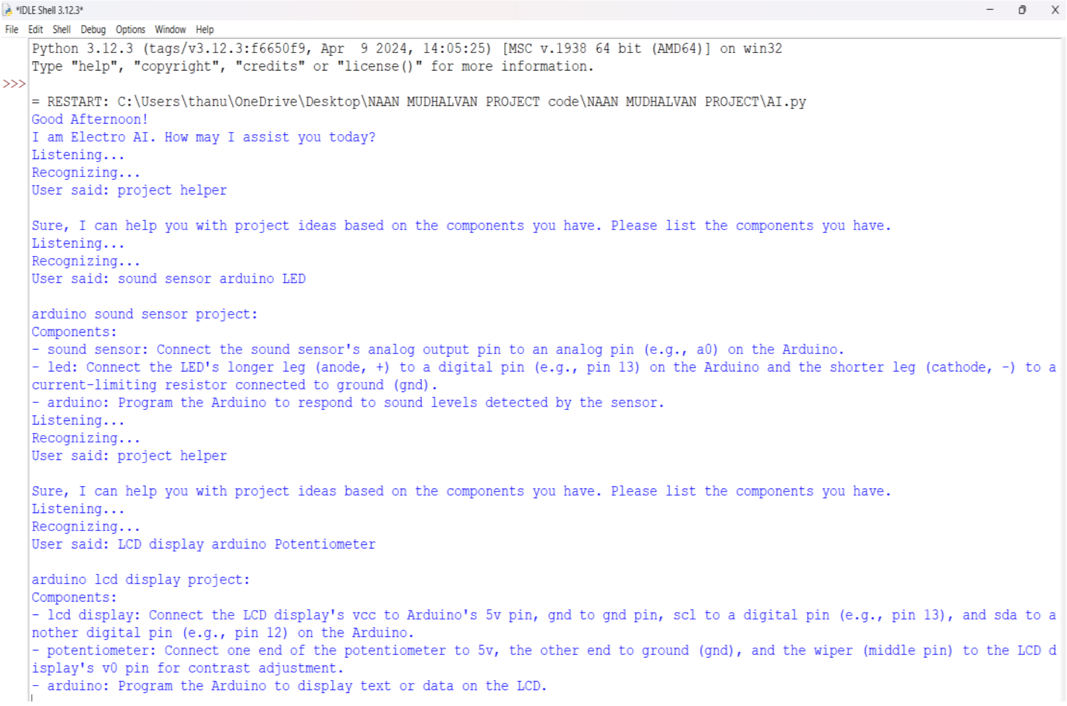
As our AI Model is trained for more than 300 projects, here we specify only 5 intents for simplicity and efficiency.

# OUTPUT:



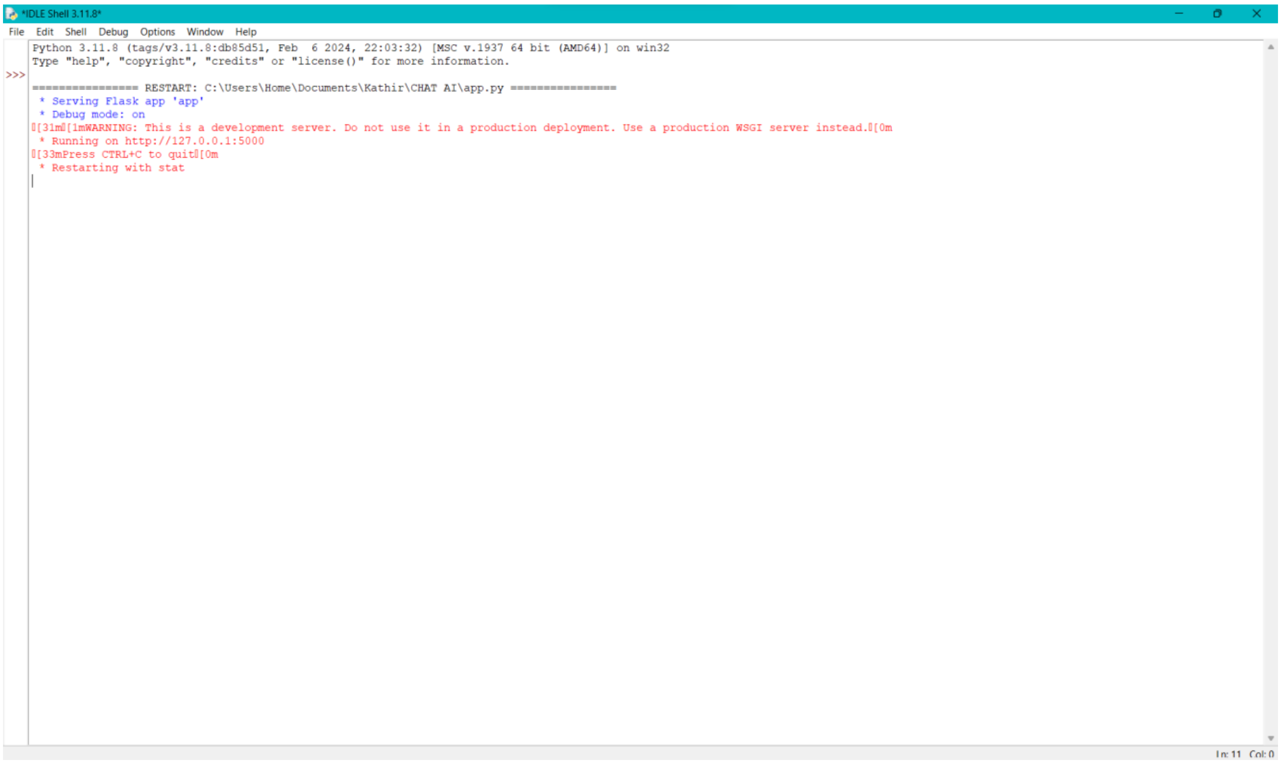
```
Python 3.12.3 (tags/v3.12.3:f6650f9, Apr  9 2024, 14:05:25) [MSC v.1938 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\thanu\OneDrive\Desktop\NM PROJECT\NAAN MUDHALVAN PROJECT\AI.py
Good Afternoon!
I am Electro AI. How may I assist you today?
Listening...
Recognizing...
User said: resistor

A resistor is a passive two-terminal electronic component that resists the flow of electric current. It is used to control the amou
nt of current in a circuit, voltage division, and as a load.
Listening...
Recognizing...
User said: capacitor

A capacitor is a passive two-terminal electronic component that stores electrical energy in an electric field. It is used for filte
ring, energy storage, coupling, and timing applications.
Listening...
Recognizing...
User said: inductor

An inductor is a passive electronic component that stores energy in a magnetic field when current flows through it. It is used in f
iltering, energy storage, and impedance matching.
Listening...
Recognizing...
User said: transistor

A transistor is a semiconductor device used to amplify or switch electronic signals and electrical power. It is fundamental in digi
tal and analog circuits, amplifiers, and switching applications.
Listening...
Recognizing...
User said: diode

A diode is a semiconductor device that allows current to flow in one direction only. It is used in rectification, signal demodulati
on, and voltage regulation.
Listening...
```

## Fig 1: Intentstrain output



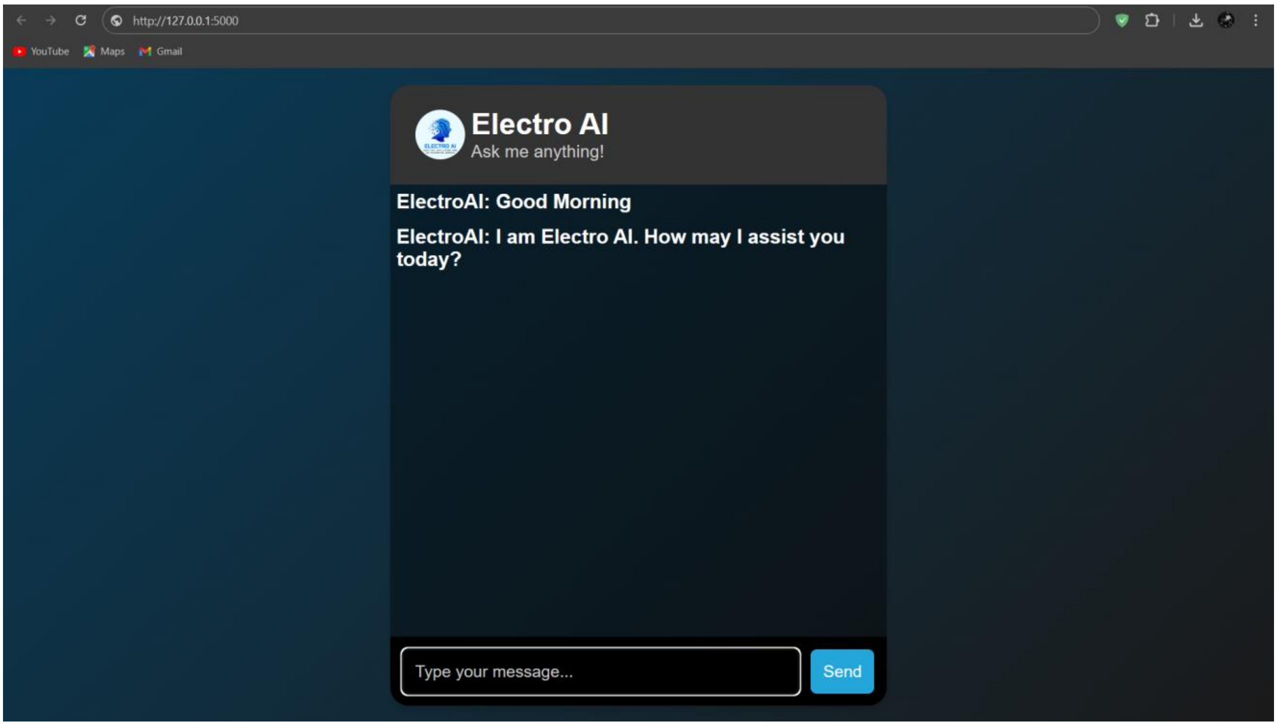
```
Python 3.12.3 (tags/v3.12.3:f6650f9, Apr  9 2024, 14:05:25) [MSC v.1938 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\thanu\OneDrive\Desktop\NAAN MUDHALVAN PROJECT code\NAAN MUDHALVAN PROJECT\AI.py
Good Afternoon!
I am Electro AI. How may I assist you today?
Listening...
Recognizing...
User said: project helper

Sure, I can help you with project ideas based on the components you have. Please list the components you have.
Listening...
Recognizing...
User said: sound sensor arduino LED

arduino sound sensor project:
Components:
- sound sensor: Connect the sound sensor's analog output pin to an analog pin (e.g., a0) on the Arduino.
- led: Connect the LED's longer leg (anode, +) to a digital pin (e.g., pin 13) on the Arduino and the shorter leg (cathode, -) to a
current-limiting resistor connected to ground (gnd).
- arduino: Program the Arduino to respond to sound levels detected by the sensor.
Listening...
Recognizing...
User said: project helper

Sure, I can help you with project ideas based on the components you have. Please list the components you have.
Listening...
Recognizing...
User said: LCD display arduino Potentiometer

arduino lcd display project:
Components:
- lcd display: Connect the LCD display's vcc to Arduino's 5v pin, gnd to gnd pin, scl to a digital pin (e.g., pin 13), and sda to a
nother digital pin (e.g., pin 12) on the Arduino.
- potentiometer: Connect one end of the potentiometer to 5v, the other end to ground (gnd), and the wiper (middle pin) to the LCD d
isplay's v0 pin for contrast adjustment.
- arduino: Program the Arduino to display text or data on the LCD.
```

## Fig 2: Projecttrain output



```
Python 3.11.8 (tags/v3.11.8:db85d51, Feb  6 2024, 22:03:32) [MSC v.1937 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
================ RESTART: C:\Users\Home\Documents\Kathir\CHAT AI\app.py ================
 * Serving Flask app 'app'
 * Debug mode: on
[31m[1mWARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.[0m
 * Running on http://127.0.0.1:5000
[33mPress CTRL+C to quit[0m
 * Restarting with stat
```

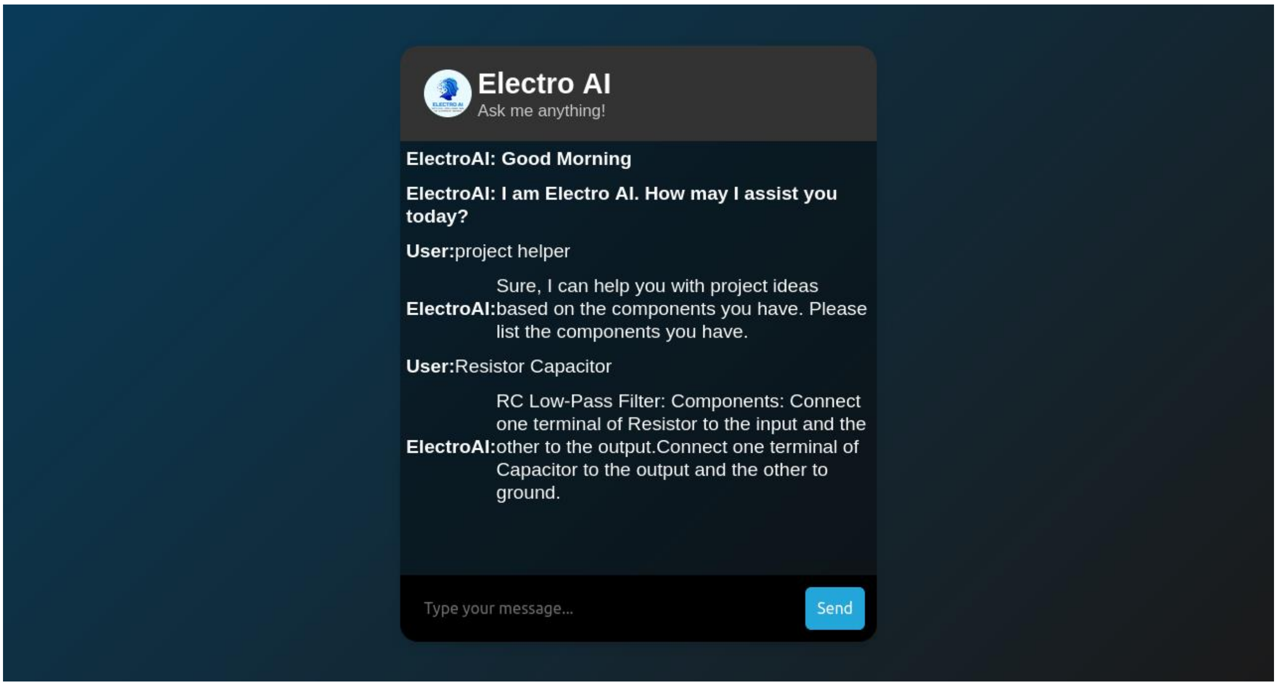
Fig 3: **Python Output with flask**



ElectroAI: Good Morning

ElectroAI: I am Electro AI. How may I assist you today?

Fig 4: **Webserver of ElectroAI**



**ElectroAI: Good Morning**

**ElectroAI: I am Electro AI. How may I assist you today?**

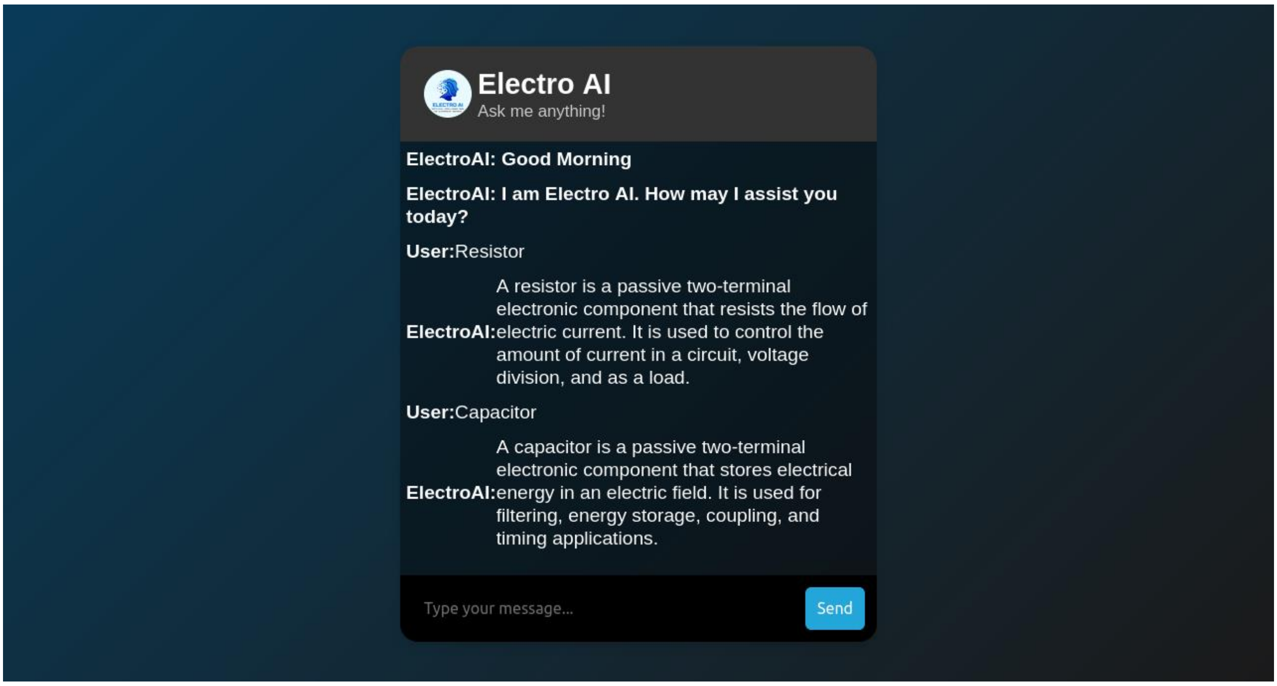
**User:**project helper

**ElectroAI:**Sure, I can help you with project ideas based on the components you have. Please list the components you have.

**User:**Resistor Capacitor

**ElectroAI:**RC Low-Pass Filter: Components: Connect one terminal of Resistor to the input and the other to the output.Connect one terminal of Capacitor to the output and the other to ground.

Fig 5: **Projecttrain output in Webserver**



**ElectroAI: Good Morning**

**ElectroAI: I am Electro AI. How may I assist you today?**

**User:**Resistor

**ElectroAI:**A resistor is a passive two-terminal electronic component that resists the flow of electric current. It is used to control the amount of current in a circuit, voltage division, and as a load.

**User:**Capacitor

**ElectroAI:**A capacitor is a passive two-terminal electronic component that stores electrical energy in an electric field. It is used for filtering, energy storage, coupling, and timing applications.

Fig 6: **Intentstrain output in Webserver**

## PROJECT HURDLES:

1. **Speech Recognition Accuracy:** Overcoming challenges related to speech recognition accuracy, especially in diverse accents and noisy environments.
2. **Natural Language Understanding:** Enhancing the assistant's ability to understand complex and context-rich queries for improved interaction.
3. **Knowledge Integration:** Ensuring seamless integration of domain-specific knowledge and maintaining an updated database of electronic components and projects.
4. **Personalization and Recommendations:** Developing algorithms for personalized recommendations and project suggestions based on user preferences and historical interactions.
5. **User Interface Design:** Designing an intuitive and user-friendly interface for the AI assistant across different devices and platforms.
6. **Real-time Assistance:** Implementing real-time assistance features for instant responses and task completion.
7. **Security and Privacy:** Addressing concerns related to data security, privacy, and confidentiality in handling user information.
8. **Machine Learning Model Training:** Training and fine-tuning machine learning models for improved accuracy, performance, and relevance.
9. **Integration with IoT Devices:** Integrating with IoT devices for enhanced functionality and hands-on experimentation opportunities.
10. **Continuous Improvement:** Establishing mechanisms for continuous improvement, feedback gathering, and updating the assistant's capabilities based on user feedback and evolving technologies.

## CONCLUSION:

Electro AI is an innovative tool combining natural language processing (NLP), information retrieval, and machine learning to assist users with electronic components and project ideas. It interprets user queries accurately, providing detailed information and inspiring project suggestions based on specified components. The system's database is continually updated, ensuring the relevance and accuracy of the information. A robust feedback loop allows for continuous improvement and adaptation to user needs. Electro AI fosters creativity and learning, making it an invaluable resource for hobbyists, engineers, and students in electronics. Its thoughtful design and adaptive capabilities position it as a transformative tool in the field.