	Dr D Y Patil Pratishthan's Dr. D.Y. Patil Institute of Engineering, Management and Research, Akurdi, Pune	DI No.: ACAD/DI/72
Academic Year: 2025-26	Weekly Report Format for Internship	Revision : 00 Dated : 20/11/2019
Term - II	Department : Computer Engineering	Date of Preparation : 14/02/2026

Weekly Report Format for Internship

Roll No: 23101

Name of the Student: Ruhan Saad Dave

Name of the Company: CODTECH IT SOLUTIONS PVT.LTD

Domain of Internship: Artificial Intelligence

Name of the Internal Guide: Akanksha Kulkarni

Name of the External Guide: Neela Santhosh Kumar

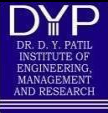
Duration of Internship: 03 Jan 2026 to 14 Feb 2026

Week-III Dates: 17/01/2026 to 24/01/2026

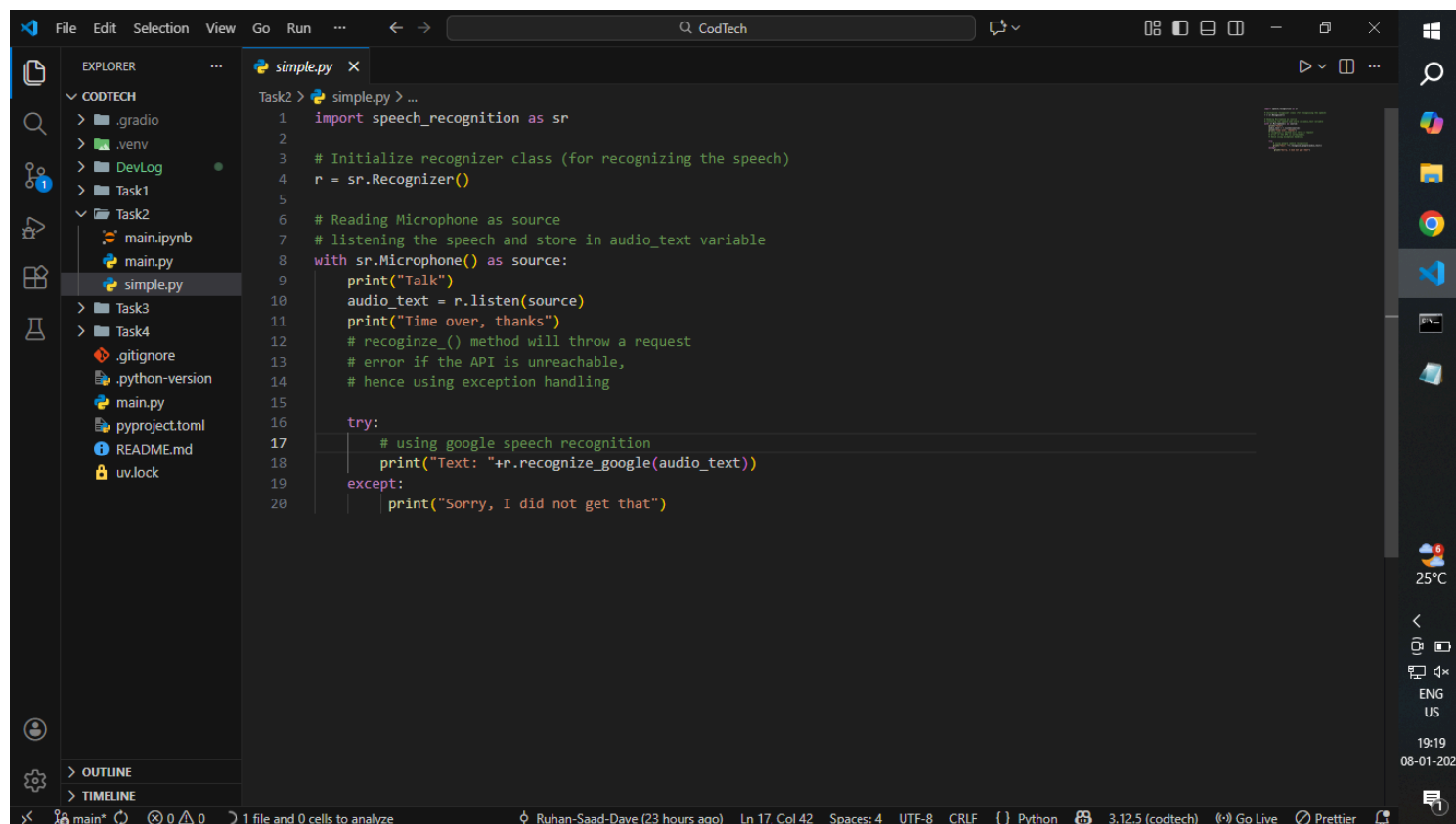
Description of Work Done Till Date (Week 3):

Task 2 & 4 - Speech Recognition and Text Generation (Initial Versions)

- Task 2 (Speech Recognition):
 - Implemented a basic Speech-to-Text system using the 'SpeechRecognition' library.
 - Created a simple script that successfully transcribes spoken words from an audio source.
- Task 4 (Generative Text Model):
 - Developed the initial version of the generative text model.
 - Integrated the LangChain framework with the Google Gemini API (gemini-2.5-flash).
 - Implemented a feature for continuous chat, allowing the model to remember conversation history
 - Used 'python-dotenv' to securely manage the API key.

	Dr D Y Patil Pratishthan's Dr. D.Y. Patil Institute of Engineering, Management and Research, Akurdi, Pune	DI No.: ACAD/DI/72
Academic Year: 2025-26	Weekly Report Format for Internship	Revision : 00 Dated : 20/11/2019
Term - II	Department : Computer Engineering	Date of Preparation : 14/02/2026

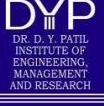
Supporting Documents:

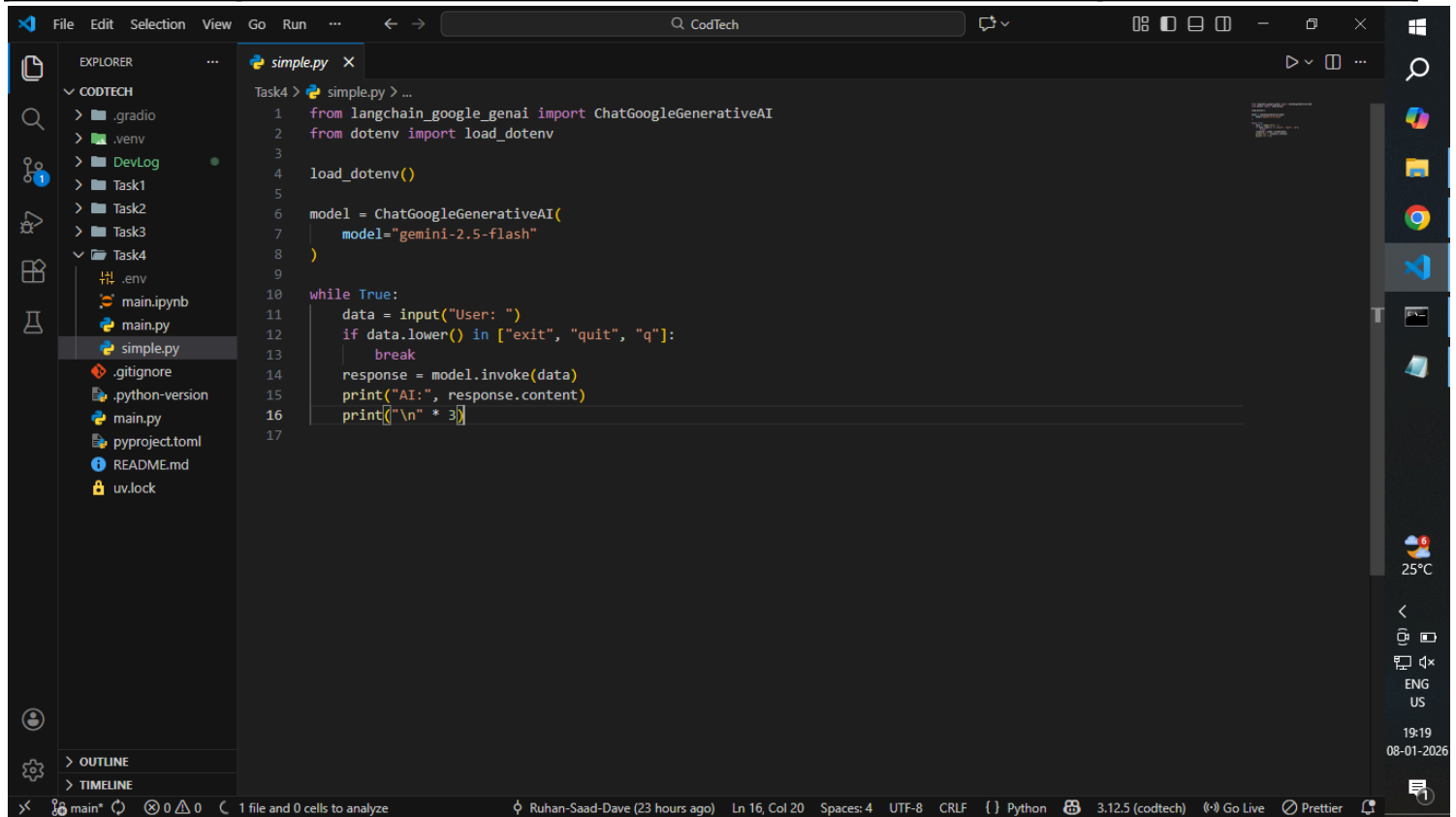


```

1  import speech_recognition as sr
2
3  # Initialize recognizer class (for recognizing the speech)
4  r = sr.Recognizer()
5
6  # Reading Microphone as source
7  # listening the speech and store in audio_text variable
8  with sr.Microphone() as source:
9      print("Talk")
10     audio_text = r.listen(source)
11     print("Time over, thanks")
12     # recognize_() method will throw a request
13     # error if the API is unreachable,
14     # hence using exception handling
15
16     try:
17         # using google speech recognition
18         print("Text: "+r.recognize_google(audio_text))
19     except:
20         print("Sorry, I did not get that")

```

	<p align="center">Dr D Y Patil Pratishthan's Dr. D.Y. Patil Institute of Engineering, Management and Research, Akurdi, Pune</p>	<p align="center">DI No.: ACAD/DI/72</p>
<p>Academic Year: 2025-26</p>	<p align="center">Weekly Report Format for Internship</p>	<p align="center">Revision : 00 Dated : 20/11/2019</p>
<p align="center">Term - II</p>	<p align="center">Department : Computer Engineering</p>	<p align="center">Date of Preparation : 14/02/2026</p>

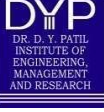


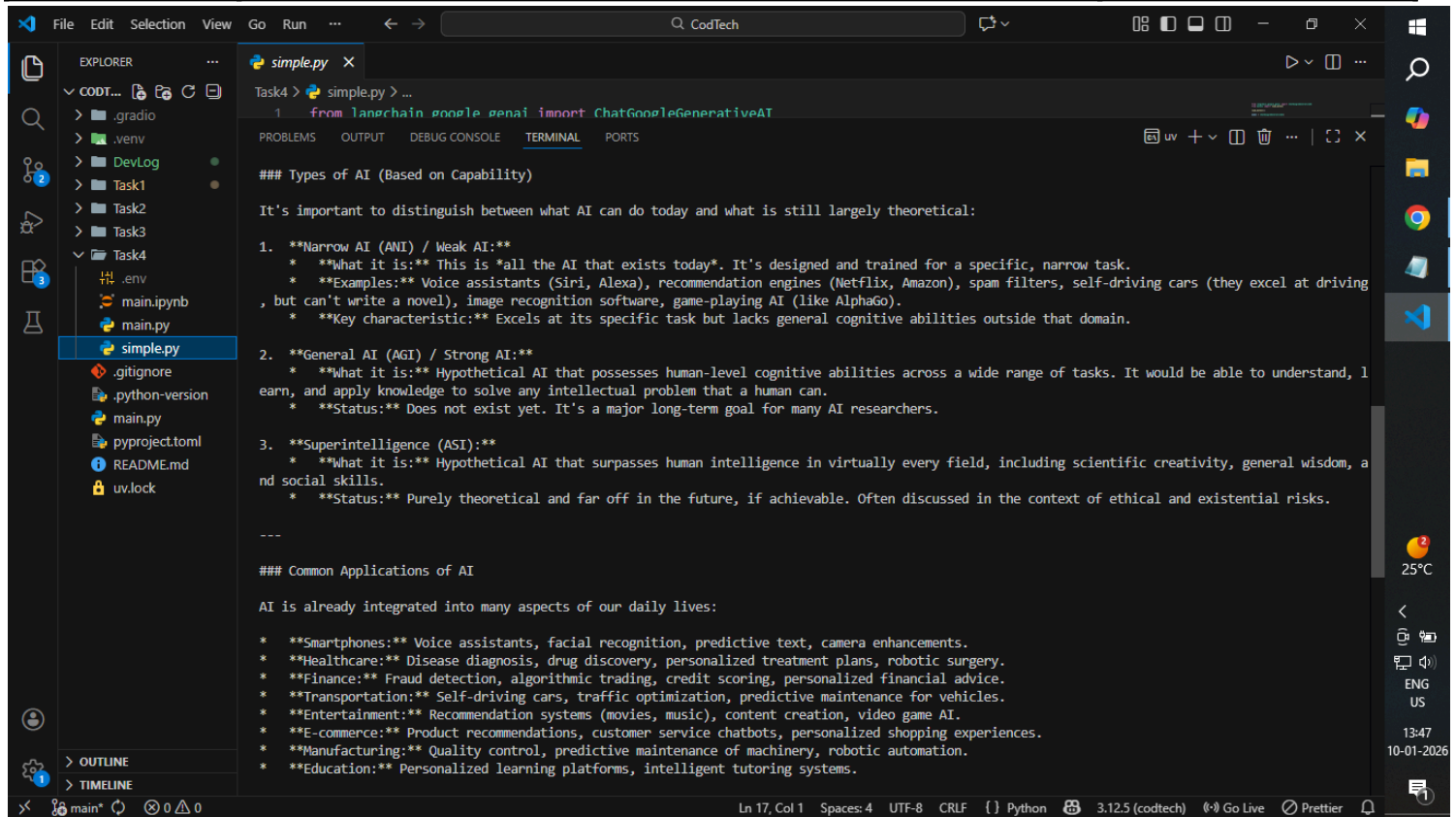
```

1  from langchain_google_genai import ChatGoogleGenerativeAI
2  from dotenv import load_dotenv
3
4  load_dotenv()
5
6  model = ChatGoogleGenerativeAI(
7      model="gemini-2.5-flash"
8  )
9
10 while True:
11     data = input("User: ")
12     if data.lower() in ["exit", "quit", "q"]:
13         break
14     response = model.invoke(data)
15     print("AI:", response.content)
16     print("\n" * 3)
17

```

The screenshot shows a Visual Studio Code editor window with a file explorer on the left. The file explorer shows a project named 'CODTECH' with a folder 'Task4' containing 'simple.py'. The main editor area displays the Python code for 'simple.py'. The code imports 'ChatGoogleGenerativeAI' from 'langchain_google_genai' and 'load_dotenv' from 'dotenv'. It then loads the environment variables and initializes a 'ChatGoogleGenerativeAI' model with the name 'gemini-2.5-flash'. A while loop is used to repeatedly prompt the user for input. If the user enters 'exit', 'quit', or 'q', the loop breaks. Otherwise, the model's response is printed, followed by three blank lines. The status bar at the bottom indicates the file is 'main.py' and shows various settings like 'Ln 16, Col 20', 'Spaces: 4', 'UTF-8', 'CRLF', 'Python', '3.12.5 (codtech)', 'Go Live', and 'Prettier'.

	<p align="center">Dr D Y Patil Pratishthan's Dr. D.Y. Patil Institute of Engineering, Management and Research, Akurdi, Pune</p>	<p align="center">DI No.: ACAD/DI/72</p>
<p>Academic Year: 2025-26</p>	<p align="center">Weekly Report Format for Internship</p>	<p align="center">Revision : 00 Dated : 20/11/2019</p>
<p align="center">Term - II</p>	<p align="center">Department : Computer Engineering</p>	<p align="center">Date of Preparation : 14/02/2026</p>



```

Task4 > simple.py > ...
1 from lanchain google genai import ChatGoogleGenerativeAI

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
uv + v | | | | |

### Types of AI (Based on Capability)

It's important to distinguish between what AI can do today and what is still largely theoretical:

1. **Narrow AI (ANI) / Weak AI:**
   * **What it is:** This is "all the AI that exists today". It's designed and trained for a specific, narrow task.
   * **Examples:** Voice assistants (Siri, Alexa), recommendation engines (Netflix, Amazon), spam filters, self-driving cars (they excel at driving, but can't write a novel), image recognition software, game-playing AI (like AlphaGo).
   * **Key characteristic:** Excels at its specific task but lacks general cognitive abilities outside that domain.

2. **General AI (AGI) / Strong AI:**
   * **What it is:** Hypothetical AI that possesses human-level cognitive abilities across a wide range of tasks. It would be able to understand, learn, and apply knowledge to solve any intellectual problem that a human can.
   * **Status:** Does not exist yet. It's a major long-term goal for many AI researchers.

3. **Superintelligence (ASI):**
   * **What it is:** Hypothetical AI that surpasses human intelligence in virtually every field, including scientific creativity, general wisdom, and social skills.
   * **Status:** Purely theoretical and far off in the future, if achievable. Often discussed in the context of ethical and existential risks.

---

### Common Applications of AI

AI is already integrated into many aspects of our daily lives:

* **Smartphones:** Voice assistants, facial recognition, predictive text, camera enhancements.
* **Healthcare:** Disease diagnosis, drug discovery, personalized treatment plans, robotic surgery.
* **Finance:** Fraud detection, algorithmic trading, credit scoring, personalized financial advice.
* **Transportation:** Self-driving cars, traffic optimization, predictive maintenance for vehicles.
* **Entertainment:** Recommendation systems (movies, music), content creation, video game AI.
* **E-commerce:** Product recommendations, customer service chatbots, personalized shopping experiences.
* **Manufacturing:** Quality control, predictive maintenance of machinery, robotic automation.
* **Education:** Personalized learning platforms, intelligent tutoring systems.

```

Student Sign

Internal Guide Sign