

Innovative Product Development

Smart Home Virtual Assistant Integration

Submitted in fulfillment of the requirements of Innovative Product Development (DJ19ILL1) (Semester-VI) in the Department of Artificial Intelligence & Data Science.

by

Rujuta Jariwala (60018210015)

Yukta Saraf (60018210028)

Rishabh Shah (60018210033)

Alan George (60018210093)

Mentor
**Professor Lakshmi
Kurup**

Department of Artificial Intelligence & Data Science



Shri Vile Parle Kelavani Mandal's

DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING

(Autonomous College Affiliated to the University of Mumbai)

NAAC Accredited with "A" Grade (CGPA : 3.18)





CERTIFICATE

This is to certify that, topic entitled "**Smart Home Virtual Assistant Integration**" has been reviewed and evaluated by undersigned members, and is submitted as partial fulfillment of Innovative Product Development (DJ19ILL1) (Sem-VI) in the Department of Artificial Intelligence & Data Science.

Professor Lakshmi Kurup

Examiner Name & Signature

Lakshmi Kurup
(I/c. HOD)

Dr. Hari Vasudevan
(Principal)



DECLARATION

I declare that any and all sources utilized in the preparation of this report have been properly cited and referenced. The ideas, concepts, and research findings presented in this proposal are entirely our own, unless otherwise acknowledged and referenced. This report represents my genuine efforts to contribute to the field of Artificial Intelligence & Data Science Engineering and to advance scholarly knowledge in a meaningful and ethical manner.

- 1) Rujuta Jariwala
- 2) Yukta Saraf
- 3) Rishabh Shah
- 4) Alan George

(Team Member Names)

Date :

Place :



Abstract

This report explores the integration of virtual assistant technology with home automation systems. The voice assistant JARVIS leverages natural language processing to understand user requests and perform tasks such as information retrieval from Wikipedia, video playback on YouTube, and managing reminders and alarms. By simplifying daily routines and enhancing accessibility, JARVIS has the potential to revolutionize how we interact with technology in our homes. We use artificial intelligence (AI) and natural language processing (NLP) methods like Hidden Markov Models (HMM) and Mel-Frequency Cepstral Coefficients (MFCC) for speech recognition. Our setup involves using Raspberry Pi 3B+ to link with IoT devices and sending signals over Bluetooth and Wi Fi. We have prioritized the security of the user by ensuring that the data does not get stored in the cache memory. To maintain functionality when disconnected from the internet, the system will explore alternative communication methods for signals between system components. This report takes a thorough look at how we can blend virtual assistant technology into home automation systems to make living spaces intelligent.

Keywords: Virtual Assistant, Smart Home Automation, Artificial Intelligence, Natural Language Processing, Raspberry Pi 3B+, Speech Recognition, Hidden Markov Model (HMM), Mel-Frequency Cepstral Coefficients (MFCC), Natural Language Understanding, Accessibility, Scalability, User Privacy.



Table of Contents

Chapter	Contents	Page No.
1	INTRODUCTION	7
2	LITERATURE REVIEW	8
	2.1 Literature survey	8-11
	2.2 Outcome of the survey	12
3	PROBLEM DEFINITION	13
	3.1 Factors affecting the accessibility gap	13
	3.2 Objectives	13
	3.3 Scope of the project	14
4	NOVELTY OF PROPOSED WORK	15
5	STUDY DESIGN & METHODOLOGY	16
	5.1 System architecture	16-18
	5.2 Software and Hardware requirements	19
	5.3 Module Implementation details	20-21
6	APPLICATION & RESULTS	22
7	CONCLUSION & FUTURE SCOPE	28
8	REFERENCES	29



List of Figures

Fig. No.	Figure Caption	Page No.
0	Study Design and Architecture of Voice Assistant, Home Automation	17,18
1, 2	Speech Recognition	22
3, 4	Google Search via Voice	22
5	Tokenization	23
6, 7	Wikipedia Summary via Voice	23
8	Youtube Content Creator / Video Search Via Voice	23
9, 10	Query Execution and Implementation in Sqlite3 Database	24
11	LED ON/OFF state	24
12	Sending & Receiving Messages between Iot Hub And Raspberry Pi	24
13	Appealing Modern Graphical Design and UI/UX	26
14	Chatroom	27

Abbreviations

Sr. No.	Abbreviation	Expanded form
1.	ML	Machine Learning
2.	SVM	Support Vector Machines
3.	CLI	Command Line Interface
4.	VHA	Virtual Home Assistant
5.	IoT	Internet of Things
6.	AI	Artificial Intelligence
7.	GUI	Graphical User Interface
8.	NLP	Natural Language Processing
9.	DSP	Digital Signal Processing
10.	ASR	Automatic Speech Recognition



1. Introduction

In the contemporary landscape of technological innovation, the convergence of home automation and virtual assistant technologies stands as a transformative force, promising unparalleled convenience and efficiency in the way individuals interact with and manage their living spaces. The synergy between these two domains holds the potential to redefine the very fabric of daily life, offering a seamless integration of intelligent systems that respond to voice commands, automate routine tasks, and enhance overall living experiences. However, amidst the promises of a connected and automated future, a substantial challenge emerges—one that threatens to impede the widespread adoption and inclusivity of this transformative integration.

The integration of home automation with virtual assistants, while unlocking a realm of possibilities for enhanced living, encounters a multifaceted accessibility gap that extends across various dimensions. This gap is not solely technological but encompasses socioeconomic, educational, and inclusivity-related facets, rendering the benefits of these innovations unevenly distributed among diverse user demographics.

This research embarks on a comprehensive exploration of the challenges posed by the accessibility gap in the integration of home automation with virtual assistants. By delving into the intricate layers of this issue, we aim to unravel the nuanced barriers that hinder the seamless incorporation of these technologies into the daily lives of a broad spectrum of users. Understanding the significance of this integration in modern living is crucial, as it sets the stage for recognizing the far-reaching implications of the accessibility gap in this technological synergy.

The purpose and scope of this research extend beyond a mere analysis of the existing challenges; it aspires to illuminate the path toward a more inclusive and equitable future. Through an in-depth literature review, we explore the evolution of home automation technologies, the current landscape of virtual home assistants, and existing studies on the adoption and impact of smart home devices.

As we progress through the subsequent sections, the focus broadens to dissect the accessibility and inclusivity dimensions in smart living technologies. From understanding the role of virtual home assistants in breaking accessibility barriers to examining customization features for differently-abled individuals, this research aims to unearth practical solutions and recommendations for bridging the accessibility gap. Additionally, we delve into the efficiency and convenience aspects of home management, health and safety integration, energy efficiency, sustainability, and remote monitoring and security, analyzing how these dimensions are impacted by and, in turn, influence the accessibility landscape.

The research also endeavors to provide real-world insights through case studies and practical applications, showcasing successful implementations, user testimonials, and the challenges encountered along the way. The concluding section synthesizes the key findings, offering a comprehensive recapitulation of the research's contributions. It also explores the broader implications for the future of smart living and puts forth recommendations for further research, setting the stage for a continued discourse on creating accessible, inclusive, and technologically advanced living environments for all. Through this research, we endeavor to contribute meaningfully to the ongoing dialogue surrounding the integration of home automation with virtual assistants, ushering in a future where technology transcends barriers and enriches the lives of diverse individuals across the global landscape.



2. Literature Review

2.1. Literature Survey

The literature survey serves as a crucial component in understanding the foundational concepts and trends surrounding the integration of home automation with virtual assistants (VHAs). This survey encompasses an exhaustive exploration of existing research, industry reports, and scholarly publications, providing a comprehensive backdrop for the subsequent research endeavors.

Sr No.	Title of Paper	Methodology	Gaps	Future Scope
1	Intelligent Voice Assistant using Android Platform Author: Vinay Sagar , Kusuma S.	Android development, voice control (recognition, command analysis), data mining.	Voice assistants often require initial training to recognize and adapt to a user's voice patterns and preferences accurately.	Enhance prediction accuracy, integrate more smart functionalities, optimize for diverse user needs, improve compatibility with various devices, and refine the user interface for broader accessibility.
2	Home Automation Using Internet of Things Author: Darpan Raut, Sumedh Salvi, Shrikrishna Salvi, Prof. S. R. Rangari .	Integration of cloud, Wi-Fi, and power-line for remote control in home automation.	Scope of improving the accuracy of voice recognition Functionality of voice assistance is compromised due to dependency on internet connectivity Potential security risks like accidental recordings or unauthorized access to sensitive information.	Strengthen Security, Expand Compatibility by Integrating with diverse appliances for broader use, Optimize Energy Usage, Enhance User Interface, Integrate AI for predictive maintenance, Address scalability challenges.
3	Artificial Intelligence-Based Voice Assistant. Authors: Vigneswaran A, Dr. Gowri J, Aakash B.	Voice recognition, sentiment analysis using AI, pattern matching, text classification with SVM.	IoT Integration: The depth of integration with IoT devices needs elaboration.	Enhance IoT integration for voice assistants, broaden home appliance control, integrate diverse



			User Interaction Personalization Performance Evaluation: There's a lack of detailed performance metrics or evaluation methodologies for the voice assistant's effectiveness and accuracy.	applications, and personalize interactions for heightened user engagement.
4	Home Automation System with Raspberry Pi Authors: Nikolay Valov, Irena Valova.	Hardware Selection: Choose Raspberry Pi 3 B+ for its versatility and compatibility. Sensor Utilization: Integrated sensors using I2C interface for data collection. Software Development: Created modules for data handling, administration, visualization, control, and user interface. Database Management: Employed MariaDB for data storage and retrieval.	Real Device Testing: Limited testing done with LED lights; actual appliance validation pending. Security Testing: Needs thorough testing and refinement for system security. Data Utilization: Opportunity to utilize accumulated data for system improvement and prediction.	Device Integration: Integrate with devices such as IR Transmitter, Microphone, Speakers. Data Utilization: Proper utilization of data for faster and improved prediction, so as to give better as well as faster responses. Addition of encryption and authentication to ensure anonymity of the user's data.



5	Development of GUI for Text-to-Speech Recognition using Natural Language Processing Authors: Partha Mukherjee, Soumen Santra, Subhajit Bhowmick, Ananya Paul, Pubali Chatterjee, and Arpan Deyasi.	NLP Integration: Analyzing and processing text inputs using NLP techniques. DSP Conversion: Utilizing DSP technology to convert processed text into synthesized speech. Application	The paper could expand on the specific algorithms and models used in NLP and DSP for a deeper technical understanding. It lacks details on potential limitations or	Language Expansion: Extend language capabilities beyond American English to increase accessibility. Enhanced Accuracy: Continuously improve accuracy for real-life
---	---	---	---	---

		Development: Creating an intuitive GUI with functionalities for text input, speech conversion, and mp3 file saving using C and .Net Framework 3.5.	challenges faced during the development process.	environment application. Web-based Real-time System: Transition the system into a web-based platform to widen its reach and accessibility.
6	Google assistant, Alexa, Microsoft Cortana, Siri Creators: Google, Amazon, Microsoft, Apple.	Can make proactive suggestions, can be integrated to various home appliances, and the dataset has been trained.	Generates pre-built responses to queries, limited to a few specific devices (not generalized across all platforms).	Provide it with a knowledge table that keeps being updated every selected time cycle, so as to make the responses more accurate, for multi-platform compatibility.



7	<p>Development of Voice Control and Home Security for Smart Home Automation.</p> <p>M. Ebrahim Abidi, Ani Liza Asnawi, N.FM. Azmin.</p>	<p>Develop a comprehensive architecture for integrating voice control and home security into smart home automation systems.</p> <p>Define communication protocols between devices to ensure seamless integration.</p> <p>Implement state-of-the-art voice recognition algorithms for accurate and responsive voice commands.</p>	<p>Issues may arise if devices from different manufacturers have interoperability challenges.</p> <p>Standardization efforts are crucial to ensuring seamless integration.</p> <p>The scalability of the system needs careful consideration, especially as the number of connected devices increases.</p> <p>Continuous monitoring for potential security vulnerabilities is critical.</p>	<p>Creating a standardized application for proper interpretability of user's data.</p> <p>Applying various scalability techniques.</p>
---	---	--	--	--



2.2. Outcome of the Survey

Overview of Virtual Home Assistants (VHAs):

The literature unfolds by delving into the categorization, functionalities, and evolution of Virtual Home Assistants. An exploration of these aspects is paramount for understanding the capabilities and limitations of VHAs, offering insights into the technological advancements that have propelled them to the forefront of smart living.

Categorization, Functionalities, and Evolution of VHA's:

The survey commences by delving into the intricate categorization and functionalities of Virtual Home Assistants, painting a vivid picture of their evolution over time. By scrutinizing their capabilities and limitations, the survey establishes a foundational understanding of the technological prowess that VHAs bring to the realm of smart living. This insight serves as a compass guiding subsequent investigations into the practical implications of their integration.

Evolution of Home Automation Technologies:

A historical contextualization of home automation technologies elucidates the trajectory of advancements that have culminated in the contemporary integration with VHAs. This section provides an understanding of the technological landscape and the underlying infrastructure that supports seamless connectivity and automation within living spaces.

Studies on the Adoption and Impact of Smart Home Devices:

A comprehensive review of academic and industry studies sheds light on user perceptions, adoption rates, and the broader impact of smart home devices. Analyzing existing research findings helps in identifying patterns, challenges, and areas of improvement, contributing valuable insights to the current state of smart home technology adoption.

Accessibility and Inclusivity in Smart Home Technologies:

Addressing the core challenge of accessibility, this section scrutinizes existing literature on the barriers faced by diverse user groups. Understanding these barriers, whether socio-economic, educational, or related to physical abilities, is imperative for formulating strategies that enhance inclusivity in the integration of home automation and VHAs.

This literature survey lays the groundwork for the research methodology, providing a basis for the selection of research approaches, data collection methods, and ethical considerations. As we embark on a journey through the existing body of knowledge, the synthesis of information gleaned from diverse sources enriches the research framework, informing subsequent investigations into the practical implications and potential solutions for the accessibility gap in the integration of home automation with virtual assistants.

In summation, the outcome of the literature survey is not merely a compilation of existing knowledge but a nuanced narrative that articulates the intricacies of integrating home automation with virtual assistants. It is a compass that guides the research towards a deeper understanding of challenges, paving the way for innovative solutions and advancements that hold the promise of a more inclusive and accessible technological landscape.



3. Problem Definition

Due to the limitations of current smart home interfaces, which are complex and lack natural interaction, there is a necessity for the development of an AI voice assistant-integrated home automation system that seamlessly integrates with the home environment of the user and allows them to control their appliances through voice commands.

3.1 Factors affecting the accessibility gap

The accessibility gap in home automation with VAs arises from a complex interplay of factors, including:

- Limited sensory input options: VAs primarily rely on auditory and visual cues for interaction. This excludes individuals with hearing or visual impairments who may struggle to effectively communicate with the system.
- Lack of physical control alternatives: Voice commands are the dominant mode of interaction, leaving users with limited dexterity or mobility dependent on others for assistance.
- Unintuitive user interfaces: VAs often employ complex language and require specific phrasing for accurate understanding, creating barriers for users with cognitive or language disabilities.
- Limited compatibility with assistive technologies: Existing assistive devices like screen readers or speech recognition software may not seamlessly integrate with home automation systems, leading to fragmented control and frustrating user experiences.
- Data privacy and security concerns: Concerns about data privacy and security, particularly for vulnerable populations, can deter individuals from adopting home automation technologies.

3.2. Objectives

- 1) The voice assistant recognizes and responds to natural language voice commands
- 2) The voice assistant performs standard voice assistant functions such as answering questions, setting reminders, alarms, timers, and providing personalized recommendations based on user preferences
- 3) Enables users to control and monitor various aspects of their home environment, including lighting, temperature, etc.
- 4) The system is easily adaptable to different types of homes and devices
- 5) Interactions with the IoT devices are confined within the boundaries of the private network of the home
- 6) Only trusted users have access to the voice assistant



3.3. Scope of the Project

Proposed Solutions:

To bridge this gap and create an inclusive home automation experience, several avenues can be explored:

- Multimodal input and output: Develop VAs that accept commands through various modalities, including voice, touch, and gesture recognition. This diversifies interaction options and caters to diverse user needs.
- Physical control alternatives: Design intuitive physical interfaces like tactile buttons or voice-controlled switches to provide alternative control methods for users with limited dexterity.
- Simplified user interfaces: Implement intuitive and user-friendly interfaces that minimize reliance on complex language and specific phrasing. This can be achieved through visual cues, context-aware prompts, and error correction mechanisms.
- Enhanced assistive technology integration: Ensure seamless compatibility between home automation systems and existing assistive technologies like screen readers and speech recognition software.
- Data privacy and security measures: Implement robust data security measures and provide transparent information about data collection and usage to address privacy concerns and build trust among users with disabilities.

Layout of idea:

This innovative project addresses a Smart Home Assistant System that merges advanced voice control with Internet of Things (IoT) technology. It aims to revolutionize smart home interactions by offering a comprehensive and intuitive control system that sets a new standard for user experience.

JARVIS leverages sophisticated Natural Language Processing (NLP) and machine learning algorithms to enable natural language understanding. Users can effortlessly command and monitor their smart home through voice commands, eliminating the need for complex apps or button presses.

This goes beyond standard automation by catering to diverse user needs. Accessibility features, and personalized recommendations based on usage patterns can be implemented. Security, scalability, and adaptability across various devices are also central to the design, promising a truly enhanced smart living experience.

This project involves integrating voice control with home automation systems. The core idea is to create a user-friendly system that empowers users with natural language interactions. By simplifying daily routines and offering intuitive voice control, JARVIS has the potential to transform the way we interact with technology in our homes.



4. Novelty of the proposed work

1) Integration of the assistant with IoT

Novelty - Integration of voice assistants with IoT enhances the functionalities and efficiency of home automation systems.

2) Contextual Continuity

Novelty - Our voice assistant maintains context from previous dialogues, ensuring coherent responses to follow-up questions.

3) Data Security/Privacy

Novelty - Interactions with the IoT devices are confined within the boundaries of the private network of the home.

4) The extension of the voice assistant's capabilities

to various tasks such as communication, AI cooking recommendations, timetable management, and product recommendations to widen its area of work.

Novelty - The multifunctional nature of the AI assistant goes beyond traditional home automation, offering users a holistic solution that simplifies daily life and improves productivity.

Features performed by JARVIS:

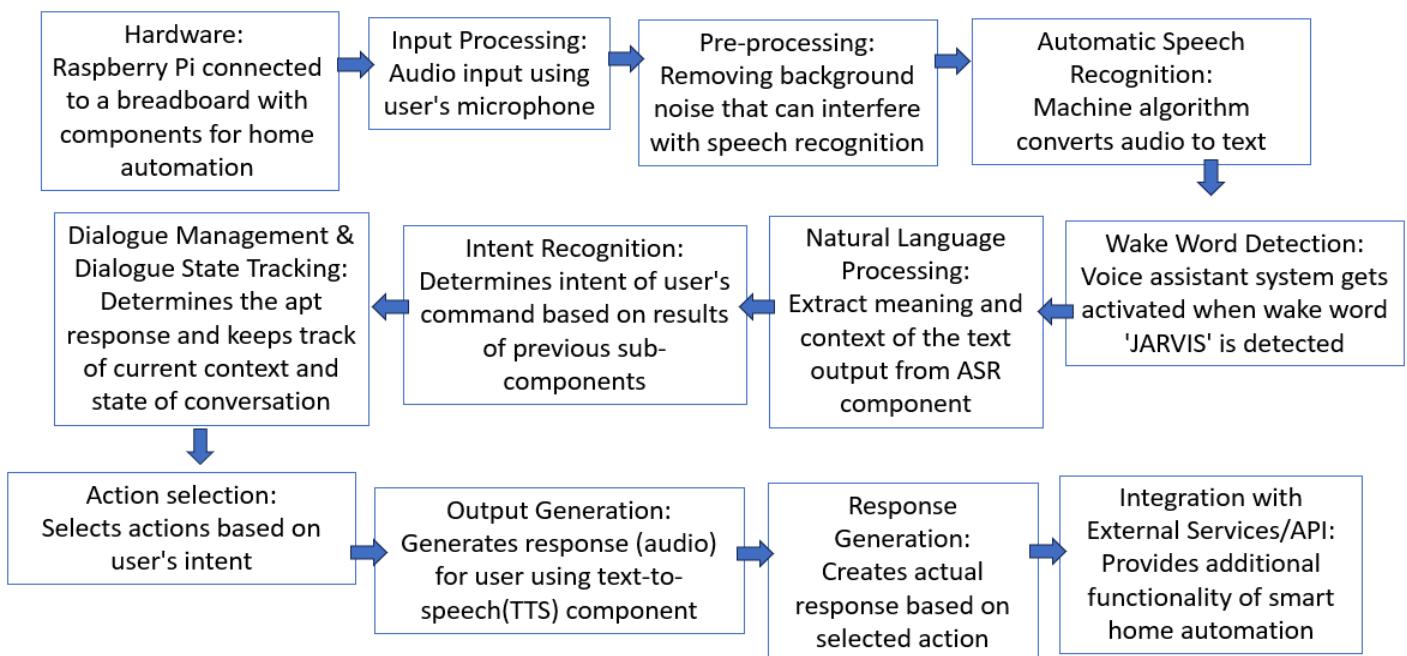
1. The voice assistant JARVIS recognizes and responds to natural language voice commands.
2. JARVIS performs standard voice assistant functions such as using search engines to access sites like Google, YouTube, and others
3. Smart lights/fans/other appliances can be turned ON/OFF via voice activation
4. Answer your doubts easily
5. The system has security features to ensure user privacy and data protection
6. Search on Wikipedia for you
7. Basic calculations can be performed
8. Setting up an Alarm clock and reminders
9. Allows exiting from the prompted programme
10. One can have a basic conversation with JARVIS
11. Only trusted users have access to the voice assistant
12. It can provide recipes based on ingredients in your pantry
13. Provides personalized recommendations based on user preferences



5. Methodology

5.1 System Architecture

Study Design and Methodology (VA)



Input Processing:

- Audio input using user's microphone: This is the first step in the process, where the ASR system captures the user's spoken words using a microphone.
- Pre-processing: The captured audio is then pre-processed to remove background noise, silence, and other unwanted sounds that could interfere with speech recognition. This can involve techniques like spectral filtering and noise gating.
- Wake Word Detection: Some ASR systems use a wake word, such as "Hey Google" or "Alexa," to activate the system. This step involves detecting the presence of the wake word in the audio input.

Dialogue Management & Dialogue State Tracking:

- Dialogue Management: This component determines the appropriate response to the user's input, taking into account the context of the conversation and the user's goals. It also keeps track of the current state of the dialogue, such as what information has been exchanged and what actions have been taken.
- Intent Recognition: This step involves identifying the user's intent, which is the meaning or goal behind their spoken words. For example, the user's intent might be to ask a question, give a command, or make a request.



- Dependency Parsing: This involves analyzing the grammatical structure of the user's spoken words to identify the relationships between words. This can be helpful for understanding the meaning of the sentence and generating an appropriate response.
- Natural Language Processing: This is a broad term that refers to the various techniques used to understand and process human language. In the context of ASR, NLP can be used for tasks such as identifying named entities (e.g., people, places, organizations), resolving pronouns, and interpreting sarcasm.

Action selection & Output Generation:

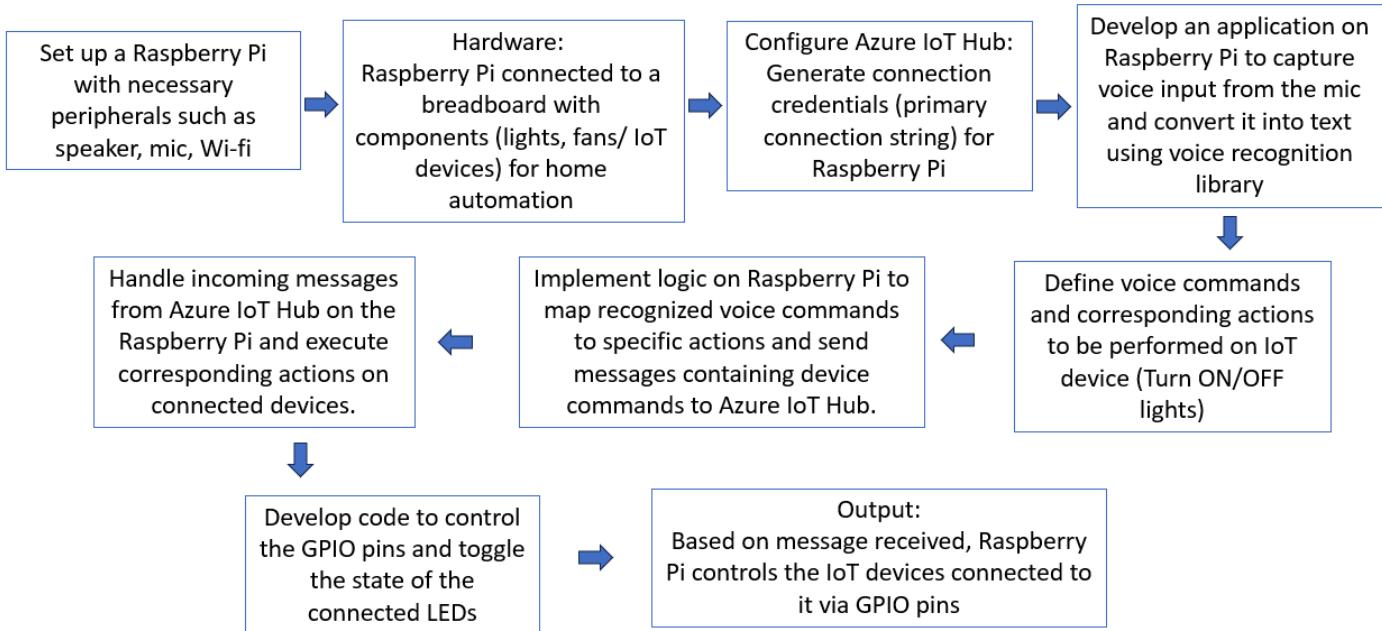
- Action selection: Based on the user's intent, the dialogue manager selects the appropriate action to take. This could involve providing information, completing a task, or asking the user for more information.
- Output Generation: The system then generates a response to the user, which can be in the form of text or speech. For speech output, a text-to-speech (TTS) engine is used to convert the text into spoken language.
- Response generation: This step creates the actual response that will be given to the user, based on the selected action.
- GUI: The user interface (GUI) displays the response to the user, and may also allow the user to interact with the system in other ways, such as through buttons or menus.
- Integration with External Services/API: ASR systems can be integrated with external services and APIs to provide additional functionality, such as access to databases or other web services.

Machine Learning:

Training the ASR/NLP components: ASR and NLP systems are trained on large amounts of data, such as spoken language recordings and text documents. This training data helps the system to learn how to recognize speech patterns and understand the meaning of language.



Study Design and Methodology (Smart Home Automation)



Set Up Raspberry Pi:

Set up Raspberry Pi with the necessary peripherals such as a microphone, speaker, and internet connectivity (Wi-Fi or Ethernet)

Configure Azure IoT Hub:

Generate connection credentials (device keys or certificates) for your Raspberry Pi to securely connect to Azure IoT Hub

Implement Voice Recognition:

Develop a script or application on Raspberry Pi to capture voice input from the microphone and convert it into text using the chosen voice recognition library

Map Voice Commands to Actions:

Define a set of voice commands and corresponding actions that you want to perform on your devices. For example, "Turn on lights," "Set thermostat to 25 degrees Celsius," etc. Implement logic on Raspberry Pi to map recognized voice commands to specific actions, including control of GPIO-connected devices such as LEDs

Communicate with Azure IoT Hub:

Develop code on Raspberry Pi to send messages containing device commands or updates (e.g., "turn on lights") to Azure IoT Hub

Device Control and Monitoring:

Implement logic to handle incoming messages from Azure IoT Hub on the Raspberry Pi and execute corresponding actions on connected devices

Control GPIO-connected LEDs:

Connect LEDs to the GPIO pins of the Raspberry Pi according to your hardware setup and develop code to control the GPIO pins and toggle the state of the connected LEDs based on commands received from Azure IoT Hub or voice commands recognized by the Raspberry Pi



5.2 Software and Hardware Requirements

Software Stack:

- Speech Recognition: Google Cloud Speech-to-Text
- Text-to-Speech (TTS): Python Text-to-Speech
- Azure for hosting the website
- Azure PostgreSQL for database
- Machine Learning Framework: PyTorch
- Groq for hosting the LLM
- Programming Language: Python (selected for its extensive libraries and programming ease)
- Project IDX from google used for writing and maintaining code (Central Repo)
- Operating System: Raspbian (for Raspberry Pi)
- Natural Language Processing (NLP): Hugging-Face [Zero-Shot Classification]
- Web Services/APIs: Hub API Endpoints

Hardware Stack:

- Single Board Computer: Raspberry Pi 3 B Plus (various models)
- Connectivity: Wi-Fi or Ethernet for internet connectivity Bluetooth for peripheral devices
- Storage: MicroSD card for the operating system and data
- Breadboard and connecting wires
- LEDs (Light Emitting Diode)
- Beeper
- Microphones and Speakers: USB or I2S microphones, USB or I2S speakers



5.3 Module Implementation details

1. Speech Recognition:

Function: Converts spoken language into text in real-time.

Utility: Enables applications to transcribe audio content, facilitating voice-controlled interfaces, transcription services, and speech analytics in various domains like customer service, healthcare, and accessibility.

2. Text-to-Speech (Python Text-to-Speech):

Function: Converts written text into spoken words or audio output.

Utility: Useful in various applications such as voice assistants, accessibility tools, audiobooks, and automated customer service, enhancing user interaction and accessibility to information.

3. Machine Learning Framework (PyTorch):

Function: Deep learning framework for building and training neural network models & LLM.

Utility: PyTorch offers flexibility, ease of use, and strong support for research prototyping and production deployment in fields like computer vision, natural language processing, and reinforcement learning.

4. Database (PostgreSQL):

Function: PostgreSQL, also known as Postgres, is a free and open-source relational database management system emphasizing extensibility and SQL compliance.

Utility: Maintains the valuable data of users, sessions, state of devices, connections to raspberry pi.

5. Project IDX (Google):

Function: Makes use of a central repository for writing and maintaining code within an organization.

Utility: Facilitates quick collaboration among the developers on the same project, makes changes in real-time while collaborating with other developers.

6. Programming Language (Python):

Function: General-purpose programming language known for its simplicity and extensive libraries.

Utility: Widely used for web development, scientific computing, data analysis, and machine learning due to its readability, ease of learning, and rich ecosystem of libraries and frameworks.

7. Operating System (Raspbian):

Function: Lightweight operating systems suitable for embedded devices like Raspberry Pi.

Utility: Optimal for resource-constrained devices, supporting various applications in IoT, robotics, and embedded systems development.

8. Natural Language Processing (Hugging Face's Zero-Shot Classification API):

Function: Classifies text into predefined categories without requiring training data for each category.

Utility: Facilitates quick categorization of text inputs without the need for extensive labeled datasets, enabling applications in sentiment analysis, content moderation, and text classification.



9. Web Services/APIs (Hub API Endpoints and Custom APIs):

Function: Provide interfaces for accessing system functionalities.

Utility: Enables interaction between different applications, devices, or services, facilitating integration and communication within a software ecosystem.

The fusion of Google's Speech-to-Text and Hugging Face's Zero-Shot Classification API enables seamless audio transcription and text categorization without exhaustive training data. Leveraging Python's Text-to-Speech and PyTorch's advanced machine learning, the system attains diverse functionality. Raspbian and Debian-based Linux serve as optimal, lightweight OS choices for embedded devices. Integrated Hub API Endpoints and custom APIs facilitate smooth integration and access across applications and devices. This versatile ecosystem epitomizes adaptability, poised as an invaluable solution across diverse industries.



6. Results

6.1 Application Demonstration of JARVIS

Speech Recognition:

```
Listening...
Recognizing....
User said: {'alternative': [{}{'transcript': 'hello'}], 'final': True}

{'alternative': [{}{'transcript': 'hello'}], 'final': True}

Listening...
Recognizing....
User said: {'alternative': [{}{'transcript': 'hi how are you doing'}, {}{'transcript': 'hay how are you doing'}], 'final': True}

{'alternative': [{}{'transcript': 'hi how are you doing'}, {}{'transcript': 'hay how are you doing'}], 'final': True}
(env) PS D:\Coding\IPD> & d:/Coding/IPD/env/Scripts/python.exe d:/Coding/IPD/testbench.py
Listening...
Recognizing....
User said: {'alternative': [{}{'transcript': 'Jarvis could you tell me who is Mark Zuckerberg'}], 'final': True}

{'alternative': [{}{'transcript': 'Jarvis could you tell me who is Mark Zuckerberg'}], 'final': True}
(env) PS D:\Coding\IPD> & d:/Coding/IPD/env/Scripts/python.exe d:/Coding/IPD/testbench.py
Listening...
Recognizing....
User said: {'alternative': [{}{'transcript': 'what are the headlines today'}], 'final': True}

{'alternative': [{}{'transcript': 'what are the headlines today'}], 'final': True}
```

Tokenization:

- PS D:\Coding\IPD> & d:/Coding/IPD/env/Scripts/Activate.ps1
- (env) PS D:\Coding\IPD> & d:/Coding/IPD/env/Scripts/python.exe d:/Coding/IPD/tester.py
Enter the command: search mrbeast on youtube
search
mrbeast
on
youtube
- (env) PS D:\Coding\IPD>

Wikipedia:

- PS D:\Coding\IPD> & d:/Coding/IPD/env/Scripts/Activate.ps1
- (env) PS D:\Coding\IPD> & d:/Coding/IPD/env/Scripts/python.exe d:/Coding/IPD/test1.py
Enter the command: jarvis
Enter the command: bill gates wikipedia
William Henry Gates III (born October 28, 1955) is an American businessman, investor, philanthropist, and writer best known for co-founding the software giant Microsoft, along with his childhood friend Paul Allen. During his career at Microsoft, Gates held the positions of chairman, chief executive officer (CEO), president, and chief software architect, while also being its largest individual shareholder until May 2014.

```
(env) PS D:\Coding\IPD> & d:/Coding/IPD/env/Scripts/python.exe d:/Coding/IPD/test1.py
Enter the command: jarvis
Enter the command: bill gates wikipedia
title: Bill Gates
Content (limited to 1 lines):
William Henry Gates III (born October 28, 1955) is an American businessman, investor, philanthropist, and writer best known for co-founding the software giant Microsoft, along with his childhood friend Paul Allen. During his career at Microsoft, Gates held the positions of chairman, chief executive officer (CEO), president, and chief software architect, while also being its largest individual shareholder until May 2014. He was a major entrepreneur of the microcomputer revolution of the 1970s and 1980s.
```



Database:

```
● PS D:\Coding\Practice\Python\GUI> & d:/Coding/Practice/Python/GUI/env/Scripts/Activate.ps1
(env) PS D:\Coding\Practice\Python\GUI> & d:/Coding/Practice/Python/GUI/env/Scripts/python.exe d:/Cod
ing/Practice/Python/GUI/testdb.py
User data already exists.
Query data already exists.
UserDetails data already exists.
```

User Table:

```
[(1, 'JohnDoe'), (2, 'JaneDoe')]
```

Query Table:

```
[(1, 'How does photosynthesis work?', 'Photosynthesis is...', '2023-12-08 13:28:20'), (2, 'What is th
e capital of France?', 'The capital is Paris.', '2023-12-08 13:28:20')]
```

UserDetails Table:

```
[(1, 'john.doe@example.com', '123-456-7890', '1990-05-15', 32), (2, 'jane.doe@example.com', '987-654-
3210', '1985-08-22', 37)]
(env) PS D:\Coding\Practice\Python\GUI>
```

```
User data already exists.
Query data already exists.
UserDetails data already exists.
```

```
User Table:
[(1, 'JohnDoe'), (2, 'JaneDoe')]
```

```
Query Table:
[(1, 'How does photosynthesis work?', 'Photosynthesis is...', '2023-12-08 13:28:20'), (2, 'What is the capital of France?', 'The capital is Paris.', '2023-12-08 13:28:20
')]
```

UserDetails Table:

```
[(1, 'john.doe@example.com', '123-456-7890', '1990-05-15', 32), (2, 'jane.doe@example.com', '987-654-3210', '1985-08-22', 37)]
```

Django administration

Site administration

ACCOUNTS	
Email addresses	+ Add Change
AUTHENTICATION AND AUTHORIZATION	
Groups	+ Add Change
Users	+ Add Change
BASE	
Connection strings	+ Add Change
Devices	+ Add Change
Messages	+ Add Change
Rooms	+ Add Change
SITES	
Sites	+ Add Change
SOCIAL ACCOUNTS	
Social accounts	+ Add Change
Social application tokens	+ Add Change
Social applications	+ Add Change

Recent actions

My actions

None available



Sending & Receiving Messages Between IoT Hub And Raspberry Pi

```
+ Code + Text
▶ def send_message(device_client):
    try:
        message = Message("Hello from Raspberry Pi!")
        device_client.send_message(message)
        print("Message sent:", message)
    except Exception as e:
        print("Error:", e)

def main():
    try:
        client = IoTHubDeviceClient.create_from_connection_string(CONNECTION_STRING)
        client.connect()
        while True:
            send_message(client)
            time.sleep(5)
    except KeyboardInterrupt:
        print("Exiting...")
    finally:
        client.disconnect()

if __name__ == "__main__":
    main()

→ Message sent: Hello from Raspberry Pi!
Exiting...
```

```
▶ import time
from azure.iot.device import IoTHubDeviceClient, Message

CONNECTION_STRING = "HostName=iothubipd.azure-devices.net;DeviceId=raspberry;SharedAccessKey=ebnEBI2BTnGC9RkLtw7NqBsDV026Q7fDMAIoTCqCHuU="
DEVICE_ID = "raspberry"

def message_received_listener(message):
    print("Message received from IoT Hub:", message.data)

def main():
    try:

        client = IoTHubDeviceClient.create_from_connection_string(CONNECTION_STRING)
        client.connect()
        client.on_message_received = message_received_listener

        print("Waiting for messages from IoT Hub...")

        while True:
            time.sleep(1)

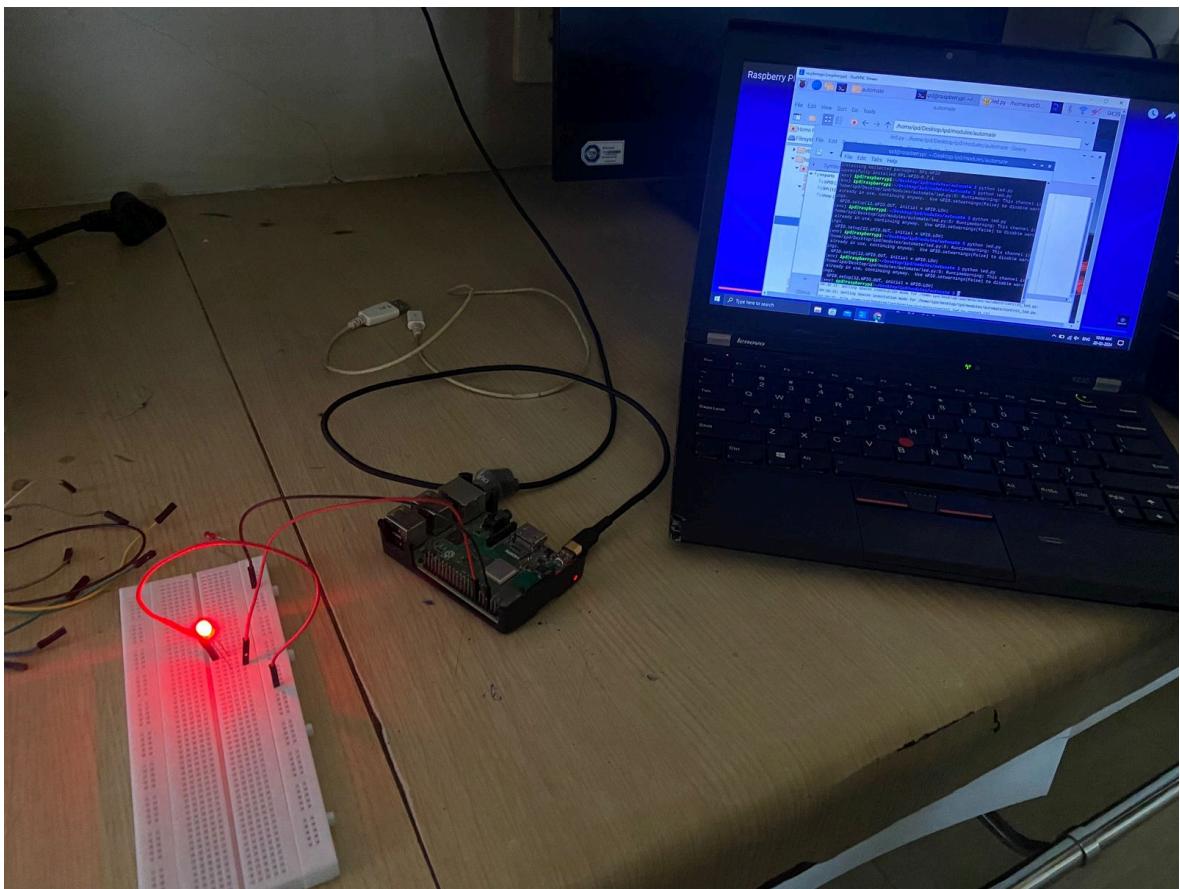
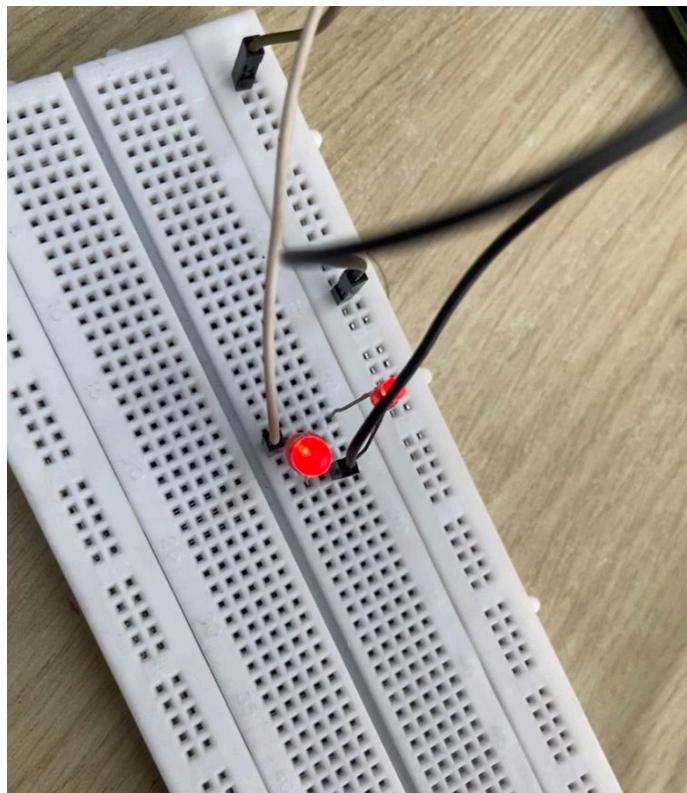
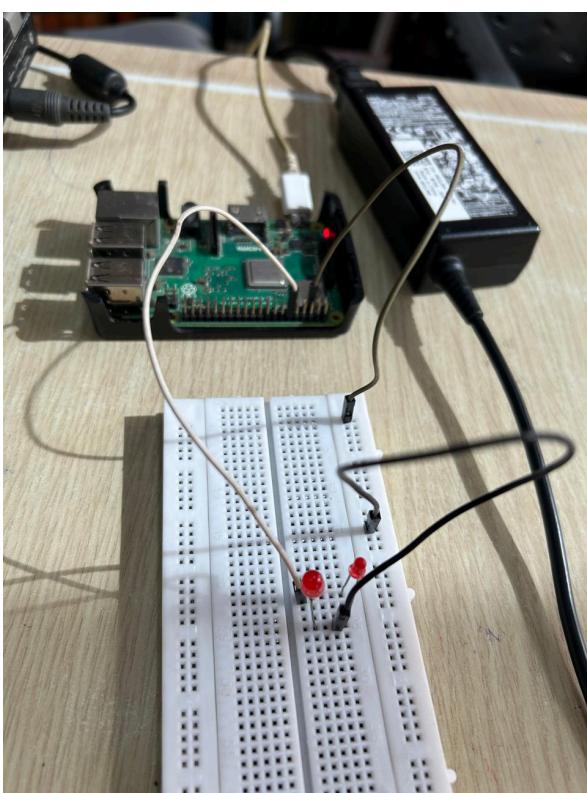
    except KeyboardInterrupt:
        print("Exiting...")
    finally:
        client.disconnect()

if __name__ == "__main__":
    main()

→ Message received from IoT Hub: b'hello yuku and ruju'
Waiting for messages from IoT Hub...
Exiting...
```

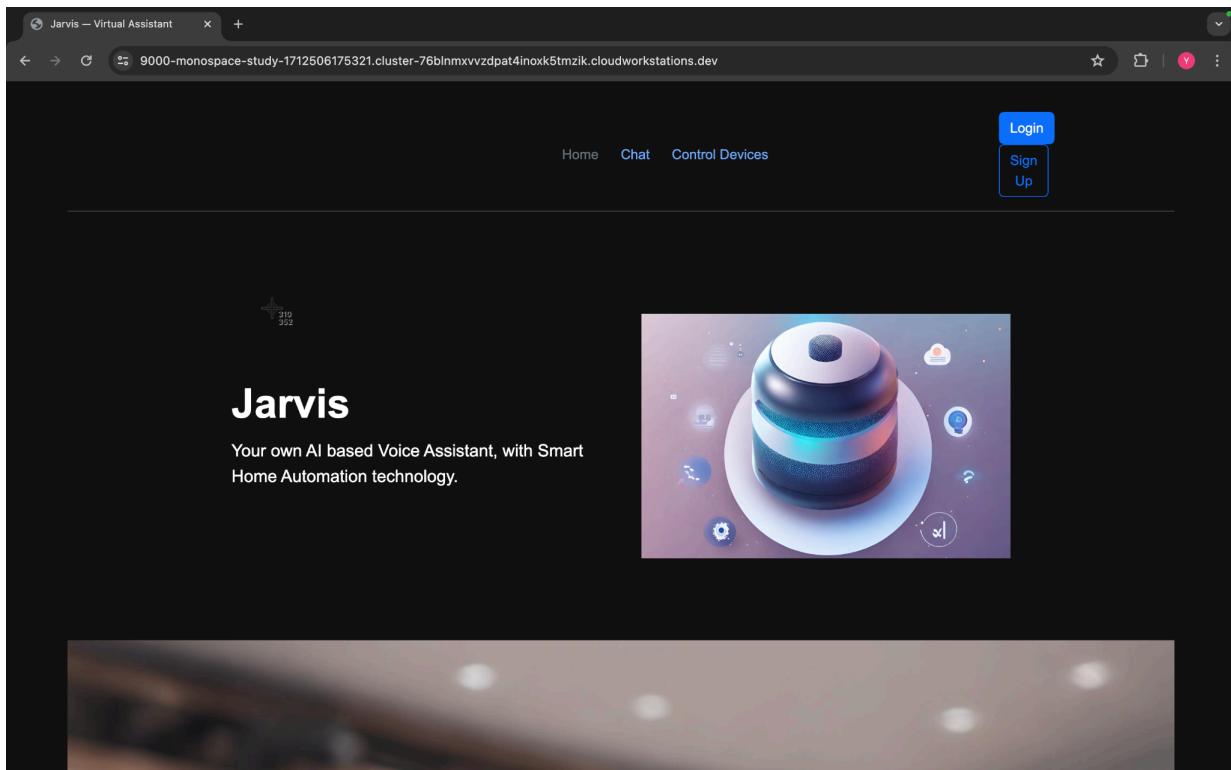


LED ON/OFF STATE:





GUI:



EXPLORER

- STUDY
 - admin.py
 - apps.py
 - forms.py
 - models.py
 - tests.py
 - urls.py
 - views.py
 - env
 - mysite
 - __pycache__
 - __init__.py
 - asgi.py
 - settings.py
 - urls.py
 - wsgi.py
 - static
 - css
 - fonts
 - img
 - js
 - templates
 - index_new.html
 - index.html
 - manage.py
 - requirements.txt
 - .gitignore
 - devserver.sh
 - README.md

Index_new.html X index.html home.html

```
ntainer > header.d-flex.flex-wrap.align-items-center.justify-content-center.justify-content-md-between.py-3.mb-4.border-bottom > div.nav-auth > div.col-md-3.text-end
1  {% load static %}
2  <!DOCTYPE html>
3  <html lang="en" data-bs-theme="dark">
4  <head>
5    <meta charset="UTF-8">
6    <title>Home Page</title>
7    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-Y
8    <link rel="stylesheet" href="{% static 'css/index.css' %}">
9    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-Y
10   </head>
11   <body>
12     {% block content %}
13     <div class="container">
14       {% load account %}
15       <header class="d-flex flex-wrap align-items-center justify-content-center justify-content-md-between py-3 mb-4 border-bottom">
16         <div class="col-md-3 mb-2 mb-md-0">
17           <a href="/" class="d-inline-flex link-body-emphasis text-decoration-none">
18             <img class="bi" width="40" height="32" role="img" aria-label="Bootstrap" data-bs-toggle="button" data-bs-
19           </a>
20         </div>
21       <ul class="nav col-12 col-md-auto mb-2 justify-content-center mb-md-0">
22         <li><a href="/" class="nav-link px-2 link-secondary">Home</a></li>
23         <li><a href="{% url 'chat-room' %}" class="nav-link px-2">Chat</a></li>
24         <li><a href="{% url 'control-devices' %}" class="nav-link px-2">Control Devices</a></li>
25       </ul>
26       <div class="nav-auth">
27         {% if request.user.is_authenticated %}
28           <a href="#" class="d-block link-body-emphasis text-decoration-none dropdown-toggle" data-bs-toggle="dropdown">
29             
30           </a>
31         {% else %}
32           {% if page != 'account_login' and page != 'account_signup' %}
33             <div class="col-md-3 text-end">
34               <div class="buttons-container">
```

Gemini Ln 43, Col 1 Spaces: 2 UTF-8 LF HTML R Go Live □



Chat-Room:

test

Conversation

rishi

Difference between linear and logistic regression

Jarvis

A great question! Linear regression and logistic regression are both types of regression analysis, but they differ in their goals and assumptions. Linear regression is a type of regression analysis where the dependent variable (target variable) is continuous or numerical. The goal of linear regression is to establish a linear relationship between the independent variables (features) and the dependent variable, with the aim of predicting the value of the dependent variable based on the values of the independent variables. The relationship is modeled using a linear equation, where the dependent variable is a linear combination of the independent variables. On the other hand, logistic regression is a type of regression analysis where the dependent variable is categorical or binary (0 or 1, yes or no, etc.). The goal of logistic regression is to predict the probability of an event occurring (e.g., 1) based on the values of the independent variables. The relationship is modeled using a logistic function, which maps the input variables to a probability between 0 and 1. Some key differences between linear and logistic regression are:

1. Dependent variable: Linear regression has a continuous dependent variable, while logistic regression has a categorical dependent variable.

2. Goal: Linear regression aims to predict the value of the dependent variable, while logistic regression aims to predict the probability of an event occurring.

3. Model: Linear regression uses a linear equation, while logistic regression uses a logistic function.

4. Assumptions: Linear regression assumes a linear relationship between the independent and dependent variables, while logistic regression assumes a non-linear relationship between the independent variables and the probability of the event occurring.

When to use each:

* Use linear regression when the dependent variable is continuous and you want to predict its value.

* Use logistic regression when the dependent variable is categorical and you want to predict the probability of an event occurring.

I hope that helps clarify the difference between linear and logistic regression!

rishi

Give a comprehensive overview of cyber security, and various types to security threats.

Jarvis

I'd be happy to provide a comprehensive overview of cyber security and the various types of security threats. Cyber security refers to the practices, technologies, and processes designed to protect digital information, computer systems, and networks from unauthorized access, use, disclosure, disruption, modification, or destruction. This includes protection against malicious attacks, unauthorized access, and other types of threats. There are several types of security threats, including:

1. Malware: Malware is a type of malicious software that is designed to harm or exploit a computer system or network. This includes viruses, worms, Trojan horses, spyware, adware, and ransomware.

2. Phishing: Phishing is a type of social engineering attack where attackers send fraudulent emails, texts, or messages that appear to be from a legitimate source, aiming to trick victims into revealing sensitive information such as passwords, credit card numbers, or personal data.

3. Ransomware: Ransomware is a type of malware that encrypts a victim's files or locks their device and demands a ransom in exchange for the decryption key or unlock code.

4. SQL Injection: SQL injection is a type of web application security vulnerability that allows an attacker to inject malicious SQL code into a web application's database, potentially allowing them to access or modify sensitive data.

5. Cross-Site Scripting (XSS): XSS is a type of web application security vulnerability that allows an attacker to inject malicious scripts into a website, potentially allowing them to steal user data or take control of the user's session.

6. Denial of Service (DoS) and Distributed Denial of Service (DDoS) Attacks: DoS and DDoS attacks involve flooding a network or system with traffic in an attempt to overwhelm it and make it unavailable to users.

7. Insider Threats: Insider threats occur when an individual with authorized access to a system or network intentionally or unintentionally causes a security breach.

8. Advanced Persistent Threats (APTs): APTs are sophisticated, targeted attacks that are designed to evade detection by traditional security controls.

9. Zero-Day Exploits: Zero-day exploits are attacks that take advantage of previously unknown vulnerabilities in software or hardware.

10. Man-in-the-Middle (MitM) Attacks: MitM attacks occur when an attacker intercepts communication between two parties, potentially allowing them to eavesdrop, modify, or inject malicious data.

11. Drive-by Downloads: Drive-by downloads occur when a user visits a compromised website or clicks on a malicious link, resulting in the download of malware onto their device.

12. Botnets: Botnets are networks of compromised devices that are controlled remotely by an attacker, often used to conduct DDoS attacks or spread malware.

13. Social Engineering: Social engineering involves manipulating individuals into divulging sensitive information or performing certain actions that can compromise security.

14. Advanced Evasion Techniques (AETs): AETs are sophisticated techniques used by attackers to evade detection by security controls.

15. Cloud Security Threats: Cloud security threats include unauthorized access, data breaches, and other security risks associated with cloud computing.

To protect against these threats, it's essential to implement a comprehensive cyber security strategy that includes:

* Firewalls and intrusion detection/prevention systems

* Encryption and secure communication protocols

* Strong passwords and multi-factor authentication

* Regular software updates and patch management

* Network segmentation and isolation

* Incident response and disaster recovery plans

* Employee education and awareness training

* Continuous monitoring and threat intelligence

I hope this comprehensive overview helps! If you have any further questions or would like more information, feel free to ask. By the way, I'm Jarvis, also known as Just Another Rather Very Intelligent System, and I'm here to help with any questions or topics you'd like to discuss!

Write message here...



rishi

explain mathematical induction with formulae

Jarvis

Mathematical induction is a fundamental concept in mathematics, and I'd be happy to explain it with formulae. Mathematical induction is a technique used to prove that a statement $P(n)$ is true for all positive integers n . The principle of mathematical induction consists of two steps:

Base Case: Prove that the statement $P(n)$ is true for the smallest possible value of n , usually $n = 1$.

Inductive Step: Assume that the statement $P(n)$ is true for some arbitrary integer k , and then prove that it is also true for $k + 1$. The formula for mathematical induction can be represented as:

1. Base Case: $P(1)$ is true 2. Inductive Step: $\forall k \in N, P(k) \Rightarrow P(k + 1)$ Here, N represents the set of natural numbers, and \forall is the universal quantification symbol, meaning "for all". Let's break it down further: **Base Case**.

We need to show that the statement $P(n)$ is true for the smallest possible value of n , which is usually $n = 1$. This is often trivial and involves a simple calculation or verification. **Inductive Step**: We assume that the statement $P(n)$ is true for some arbitrary integer k . This is called the induction hypothesis. Then, we need to prove that if $P(k)$ is true, then $P(k + 1)$ is also true. The key idea is that if the statement $P(n)$ is true for some k , then it must also be true for $k + 1$. This is often done by using the induction hypothesis to prove that $P(k + 1)$ is true. Here's a simple example to illustrate this: **Example**: Prove that the sum of the first n positive integers is given by the formula: $1 + 2 + 3 + \dots + n = n(n + 1)/2$ **Base Case**: For $n = 1$, the formula becomes $1 = 1(1 + 1)/2$, which is true. **Inductive Step**: Assume that the formula is true for some k , i.e., $1 + 2 + 3 + \dots + k = k(k + 1)/2$ Now, we need to prove that the formula is true for $k + 1$: $1 + 2 + 3 + \dots + k + (k + 1) = (k + 1)(k + 1 + 1)/2$ Using the induction hypothesis, we can rewrite the left-hand side as: $(k(k + 1)/2) + (k + 1) = (k + 1)(k + 2)/2$ which is equal to the right-hand side. Therefore, we have shown that if the formula is true for k , then it is also true for $k + 1$. By mathematical induction, we can conclude that the formula is true for all positive integers n . I hope this explanation helps!

Do you have any specific questions about mathematical induction or would you like more examples?

rishi

Tell me why world war 1 and world war 2 happened?

Jarvis

What a great question! World War I, also known as the Great War, was a complex and multifaceted conflict that involved many countries and was sparked by a series of events. One of the main causes was the assassination of Archduke Franz Ferdinand, the heir to the Austro-Hungarian throne, by Gavrilo Princip, a Serbian nationalist, in Sarajevo, Bosnia, on June 28, 1914. This event triggered a chain reaction of diplomatic crises and military mobilizations that eventually led to the outbreak of war. Other factors that contributed to the war include:

1. Nationalism: The rise of nationalist movements in various countries, particularly in the Balkans, led to an increase in tensions between nations.

2. Imperialism: The competition for colonies and resources in Africa and Asia led to rivalries between European powers.

3. Militarism: The buildup of military forces and the glorification of war in many countries created an atmosphere in which war was seen as a viable solution to conflicts.

4. Alliances: The complex system of alliances between European powers, such as the Triple Entente (France, Britain, and Russia) and the Triple Alliance (Germany, Austria-Hungary, and Italy), created a situation in which a small conflict could quickly escalate into a larger war.

As for World War II, it was also a complex and multifaceted conflict, but some of the main causes include:

1. The Treaty of Versailles: The harsh penalties and reparations imposed on Germany after World War I led to widespread resentment and a sense of injustice among the German people.

2. The Rise of Nationalism: The rise of nationalist movements, particularly in Germany and Italy, led to an increase in tensions between nations.

3. Fascist and Nazi Ideologies: The aggressive militarism and expansionist policies of fascist Italy and Nazi Germany, led by Benito Mussolini and Adolf Hitler, respectively, contributed to the outbreak of war.

4. Appeasement Policy: The policy of appeasement, pursued by Britain and France, which involved giving in to Germany's demands in the hope of avoiding war, ultimately emboldened Hitler and led to further aggression.

These are just some of the main causes of World War I and World War II. If you'd like more information or clarification, feel free to ask! By the way, I'm Jarvis, also known as Just Another Rather Very Intelligent System. I'm an AI designed to provide information and answer questions to the best of my abilities.

rishi

Tell me about transformers the movie series

Jarvis

I'd be happy to tell you about the Transformers movie series! The Transformers film series is a series of science fiction action films based on the Transformers franchise, which began with the 2007 film *Transformers*, directed by Michael Bay. The series has since grown

to include five live-action films, as well as an animated film and several spin-offs. The main storyline of the series revolves around the Transformers, sentient robots from the planet Cybertron who can transform into various forms of vehicles, animals, and other objects.

The films typically feature the Autobots, a group of heroic Transformers who come to Earth to protect humanity from the Decepticons, a rival group of evil Transformers. The live-action films in the series include:

1. *Transformers* (2007) - The first film in the series, which introduces the main characters and sets the stage for the franchise.

2. *Transformers: Revenge of the Fallen* (2009) - The sequel to the first film, which sees the Autobots facing off against the Decepticons in a battle to protect the Earth.

3. *Transformers: Dark of the Moon* (2011) - The third film in the series, which explores the history of the Transformers and their connection to the Apollo 11 moon landing.

4. *Transformers: Age of Extinction* (2014) - The fourth film in the series, which takes place four years after the events of the third film and introduces new human characters.

5. *Transformers: The Last Knight* (2017) - The fifth film in the series, which sees the Autobots and humans working together to uncover the secrets of the Transformers' past.

The animated film, *Transformers: Animated*, was released in 2007 and serves as a standalone story within the franchise. The series has received mixed reviews from critics, but has been commercially successful, with the films grossing over \$4.8 billion worldwide. Would you like to know more about a specific aspect of the Transformers franchise?

Write message here...





7. Conclusion and Future Scope

This report explores the potential of voice assistants within the realm of smart homes and as a transformative technology shaping how we interact with the world around us. The voice assistant Jarvis offers a user-friendly way to adjust lights, fans, and appliances, creating a convenient and comfortable environment for the user.

However, creating a truly seamless experience requires overcoming certain problems including privacy concerns regarding data collection and security vulnerabilities.

Technical limitations in compatibility across various smart home devices necessitate further development.

To build a smarter future, robust security protocols, anonymized user data, and user control over data collection are crucial for trust. Standardized protocols and open-source platforms could improve compatibility between smart home devices and voice assistants, while advancements in speech recognition algorithms will ensure understanding of diverse languages and accents. Offline functionality through local processing can further reduce reliance on a stable internet connection.

These advancements pave the way for evolutionary innovations beyond the smart home.

Voice-controlled smart homes of the future could utilize AR overlays activated by voice commands, allowing for virtual furniture placement and real-time appliance repair instructions. Voice assistants could connect with smart city infrastructure, report issues. Reminding users to take medication, or even connecting them with healthcare professionals in case of emergencies.

Finally, voice assistants can become powerful tools for increased productivity and accessibility for users with physical limitations, allowing them to manage calendars, schedule appointments, or dictate documents using just their voice. By addressing current limitations and embracing these future trends, voice assistants have the potential to transform our living spaces, revolutionize how we interact with technology, and enhance our lives in countless ways.

There could be an option of local inference where the voice assistant can process basic tasks and respond to user queries even without a stable internet connection. This could offer offline responses but it comes with a trade-off, the pre-trained model's knowledge and capabilities will be limited to the data it was trained on, potentially hindering its ability to respond to new information. Future research can explore techniques for enabling local inference.

JARVIS also faces a latency issue i.e. time lag between the user asking the question and the response generated, which could be improved over time by faster processing and smarter algorithms.



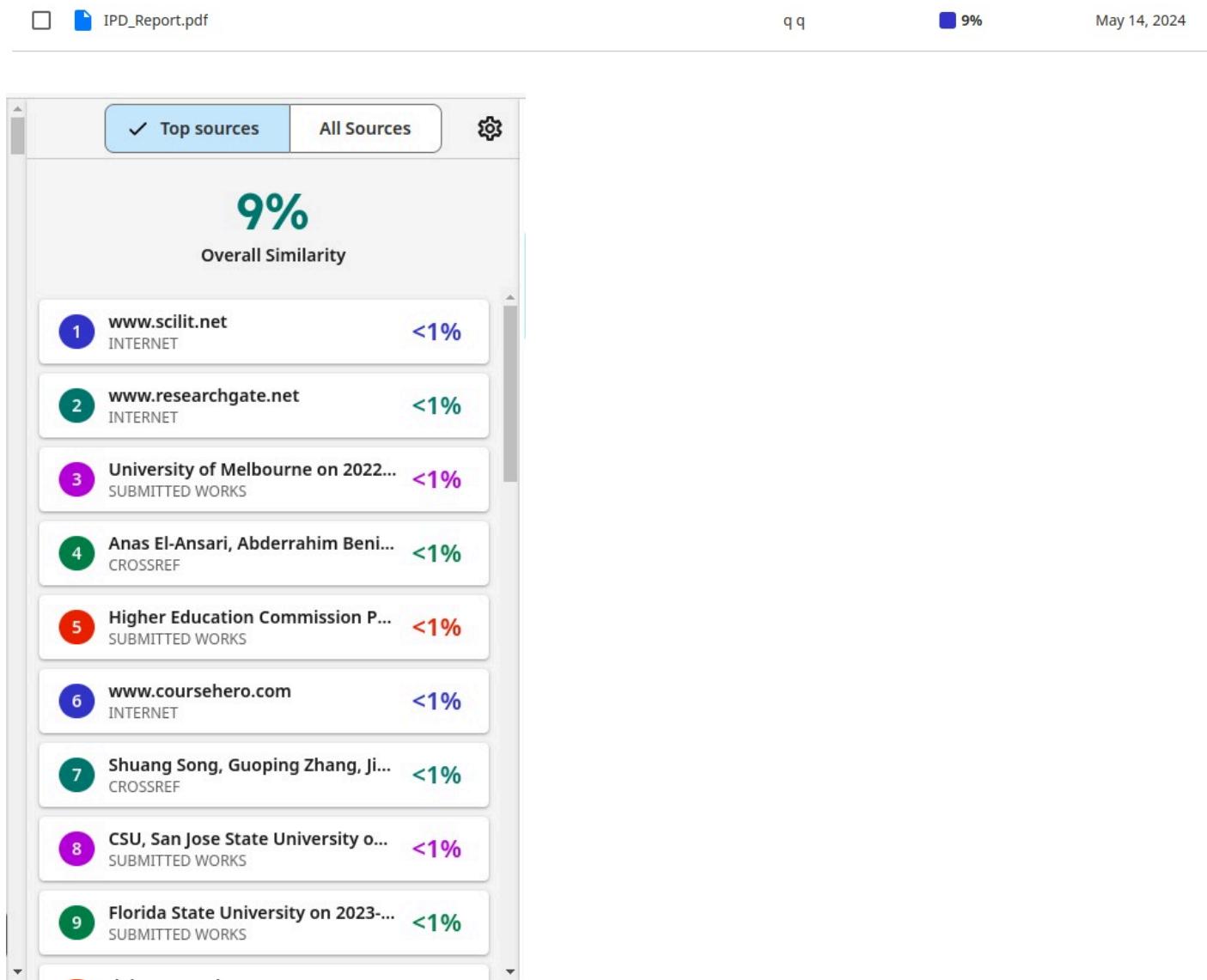
8. References

- [1] N. Valov and I. Valova, "Home Automation System with Raspberry Pi," 2020 7th International Conference on Energy Efficiency and Agricultural Engineering (EE&AE), Ruse, Bulgaria, 2020, pp. 1-5, doi: 10.1109/EEAE49144.2020.9278998.
- [2] R. B. A. Tonoy, Mahmudunnabi, H. M. Zilany and R. M. Rahman, "A Smart and Intelligent Home Automation System," 2021 IEEE 12th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), New York, NY, USA, 2021, pp. 0446-0452, doi: 10.1109/UEMCON53757.2021.9666614.
- [3] Y. Wang, C. Chen and W. Zhang, "Design of Smart Home Control System Based on Internet of Things," 2023 IEEE 6th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Chongqing, China, 2023, pp. 1680-1683, doi: 10.1109/ITNEC56291.2023.10082440. keywords: {Wireless communication; Wireless sensor networks; TV; Microcontrollers; Web pages; Smart homes; Control systems; smart home; Internet of things; microcontroller; wireless communication}
- [4] P. S. Nagendra Reddy, K. T. Kumar Reddy, P. A. Kumar Reddy, G. N. Kodanda Ramaiah and S. N. Kishor, "An IoT based home automation using android application," 2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES), Paralakhemundi, India, 2016, pp. 285-290, doi: 10.1109/SCOPES.2016.7955836. keywords: {Pins; Androids; Humanoid robots; Temperature sensors; Home appliances; Rain; Arduino; IoT; Android; Wi-Fi; Home Automation},
- [5] Iqbal, Salman, et al. "Internet-of-Things based Smart Home Automation System using Android Phone." Sir Syed University Research Journal of Engineering & Technology (SSURJET) 11.2 (2021).
- [6] P. Dubey, H. Chourasia and S. Ghode, "A Novel Approach to Design Home Automation using IoT Applications," 2023 International Conference on Advancement in Computation & Computer Technologies (InCACCT), Gharuan, India, 2023, pp. 801-806, doi: 10.1109/InCACCT57535.2023.10141754. keywords: {Home appliances; Fans; Home automation; Wheelchairs; Operating systems; Lighting; Programming; sensor; microcontroller; Internet of Things (IoT); relay; blynk app; Raspberry pi etc},
- [7] Tonoy, Rahat & Mahmudunnabi, & Zilany, Hasan & Rahman, Mohammad. (2021). A Smart and Intelligent Home Automation System. 0446-0452. 10.1109/UEMCON53757.2021.9666614.
- [8] Y. Wang, S. Rawat and F. Metze, "Semi-automatic audio semantic concept discovery for multimedia retrieval," 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Florence, Italy, 2014, pp. 1375-1379, doi: 10.1109/ICASSP.2014.6853822. keywords: {Semantics; Acoustics; Vocabulary; Videos; Multimedia communication; Streaming media; Speech; audio semantic concept discovery; semiautomatic; multimedia retrieval},
- [9] I. Feki, A. Ben Ammar and A. M. Alimi, "Query sound-by-example video retrieval framework," 2014 14th International Conference on Hybrid Intelligent Systems, Kuwait, Kuwait, 2014, pp. 297-302, doi: 10.1109/HIS.2014.7086165. keywords: {Speech; Support vector machines; Indexing; Music; Silicon; Hidden Markov models; Accuracy; audio concept; sound indexing; query sound; video retrieval},
- [10] R. Chengalvarayan and Li Deng, "HMM-based speech recognition using state-dependent, discriminatively derived transforms on mel-warped DFT features," in IEEE Transactions on Speech and Audio Processing, vol. 5, no. 3, pp. 243-256, May 1997, doi: 10.1109/89.568731.



9. Appendix

9.1 Plagiarism Report



(Report)

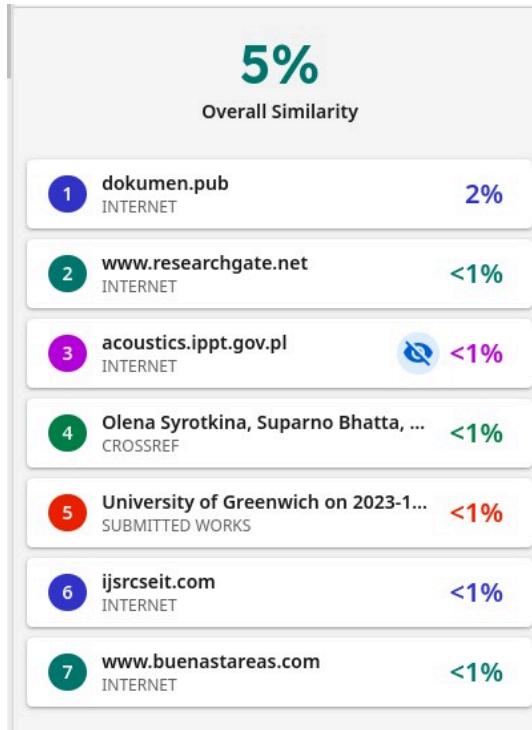


IPD_rearch_paper_final1 (1).pdf

q q

5%

May 14, 2024



(Research Paper)