BASAVARAJESWARI GROUP OF INSTITUTIONS

BALLARI INSTITUTE OF TECHNOLOGY & MANAGEMENT

Autonomous Institute under VTU, Belagavi | Approved by AICTE,New Delhi Recognized by Govt. of Karnataka

NACC Accredited Institution*

(Recognized by Govt. of Karnataka, approved by AICTE, New Delhi & Affiliated to Visvesvaraya Technological University, Belgavi)

"JnanaGangotri" Campus, No.873/2, Ballari-Hospet Road, Allipur, Ballar1-583 104 (Karnataka) (India)

Ph: 08392 – 237100 / 237190, Fax: 08392 – 237197

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

A Mini Project Report On

"J.A.R.V.I.S. VIRTUAL ASSISTANT"

Project Associates:

APPU	**3BR22CS014**
KARTHIK M K	**3BR22CS066**
MALLIKARJUN	**3BR22CS084**
PRAVEEN KUMAR NAIK	**3BR22CS124**

Under the Guidance of

Dr. Sudhakar Avareddy.
Associate Prof,
Dept of CSE
BITM, Ballari.

Visvesvaraya Technological University

Belagavi, Karnataka

2024-2025

BASAVARAJESWARI GROUP OF INSTITUTIONS
BALLARI INSTITUTE OF TECHNOLOGY & MANAGEMENT
Autonomous Institute under VTU, Belagavi | Approved by AICTE,New Delhi Recognized by Govt. of Karnataka

NACC Accredited Institution*
(Recognized by Govt. of Karnataka, approved by AICTE, New Delhi & Affiliated to Visvesvaraya Technological University, Belgavi)
"JnanaGangotri" Campus, No.873/2, Ballari-Hospet Road, Allipur, Ballar1-583 104 (Karnataka) (India)
Ph: 08392 – 237100 / 237190, Fax: 08392 – 237197

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the project work entitled "J.A.R.V.I.S. VIRTUAL ASSISTANT" is a Bonafide work carried out by **Mallikarjun: 3BR22CS084** in partial fulfillment for the award of degree of **Bachelor Degree in Computer Science & Engineering** in the VISVESVARAYA TECHNOLOGICAL UNIVERSITY, Belagavi during the academic year 2024-2025. It is certified that all corrections and suggestions indicated for internal assessment have been incorporated in the report deposited in the library. The project has been approved as it satisfies the academic requirements in respect of mini project work prescribed for a Bachelor of Engineering Degree.

Signature of project guide
Dr. Sudhakar Avareddy

Signature of Coordinator
Prof. Md. Shafiulla

Signature of HOD
Dr. R. N Kulkarni

Abstract

Jarvis is a voice-controlled virtual assistant built with HTML, CSS, and JavaScript, featuring a sleek design and dynamic interactions. It uses Speech Synthesis for text-to-speech and Speech Recognition to process commands like opening websites or answering queries. Unrecognized commands trigger a Google search, ensuring functionality. The UI includes features like voice animations and microphone buttons, with timeouts to manage silences. Combining voice recognition and responsive design, Jarvis demonstrates innovative, user-friendly web applications.

Acknowledgement

The satisfaction that we feel at the successful completion of Mini project, it would be incomplete if we did not mention the names of people, whose noble gesture, affection, guidance, encouragement and support crowned my efforts with success. I express my deep sense of gratitude to the Management of Ballari Institute of Technology and Management, Ballari, for providing me with the congenial environment in the college. I am deeply indebted to **Dr. Sudhakar Avareddy**, Associate Professor project guide and Project Coordinator **Mr. Md. Shafiulla**, for consistently providing with the required guidance which helped me in the timely and successful completion of this project. Despite his extremely busy schedules in Department, he was always available to share with me his deep insights, wide knowledge and extensive experience. I express my deep gratitude to **Dr. Yadavalli Basavaraj**, Principal, Ballari Institute of Technology and Management, Ballari, for providing me with the congenial environment in the college. I am thankful to **Dr. R. N. Kulkarni**, Head of the Department, Computer Science and Engineering, for providing the facilities in the Department to do this Mini project work. I extended sincere thanks to all the teaching and non-teaching staff members of, Computer Science and Engineering Department of BITM, Ballari for constant support and help during the project work.

Table of Contents

List of Tables

CHAPTER 1

1.1 Introduction to Project

In the modern era, artificial intelligence (AI) has become an integral part of daily life, streamlining tasks and enhancing productivity. This project, "Jarvis Virtual Assistant," is a practical implementation of AI-driven voice interaction technology, inspired by the concept of intelligent personal assistants like Jarvis from the Marvel Cinematic Universe. The system is designed to assist users with a variety of tasks through voice commands, offering a hands-free and efficient way to interact with technology.

The primary objective of this virtual assistant is to provide a user-friendly interface capable of performing tasks such as playing music, opening websites, retrieving information from the internet, and providing real-time updates like date and time. It also incorporates features for basic calculations and intelligent task execution based on natural language commands.

The development of this project utilizes cutting-edge web technologies such as JavaScript's Speech Synthesis and Speech Recognition APIs for speech output and input. Furthermore, integration with third-party platforms like YouTube enhances its functionality, making it a versatile tool for entertainment and information retrieval.

This report outlines the design, implementation, and features of the Jarvis Virtual Assistant, emphasizing the methodologies and technologies employed in its development. The project demonstrates the potential of voice-controlled systems to revolutionize human-computer interaction by making it more natural and intuitive.

1.2 Problem Statement

To implement a "Virtual Assistant" by using APIs like Speech Synthesis, Speech Recognition for developing a voice-activated platform.

1.3 Objectives

1. To allow users to open online applications like YouTube, Google, Gmail, etc. using simple voice commands.

2. To enable users to search for and play specific songs, videos, or content directly from media platforms via voice.

3. To conduct basic online searches as per user request and deliver quick results.

4. To perform essential tasks such as retrieving the current time or date, making quick calculations, and browsing information on the internet.

5. To ensure the assistant can simplify routine tasks, reducing dependency on manual input and improving user accessibility.

6. To greet users based on the time of day and respond naturally to personal queries like "Who are you?" or "What's your name?".

1.4 Scope of The Project

The scope of this project is to create an interactive and voice-controlled virtual assistant named **JARVIS**. It is designed to enhance user convenience by performing tasks based on voice commands. Jarvis can greet the user based on the time of day, respond to specific queries, and execute various actions such as opening popular websites (YouTube, Instagram, Google, etc.), providing the current time or date, and answering personal or project-related questions. The project integrates advanced browser APIs like **Speech Recognition** for understanding user voice commands and **Speech Synthesis** to provide spoken responses, creating an engaging conversational experience.

1.5 Literature Survey

SL no	Title	Author & Published Year	Findings
1.	J.A.R.V.I.S – an AI Technology Based Personal Windows Assistant system	Susmita Das, Ankita Saha, Aparupa Chakraborty, Swagata Thakur, Sumanta Chatterjee, 2022	The paper presents JARVIS, an AI-based assistant for Windows that uses voice commands to perform tasks like opening applications, web searches, and scheduling. It features speech recognition, text-to-speech, and general conversation capabilities. Future plans include integration with mobile devices, IoT, and enhanced automation.
2.	AI Based Voice Assistant Using Python	Deepak Shende, Ria Umahiya, Monika Raghorte, Aishwarya Bhisikar, Anup Bhange, 2019.	The paper explores local voice assistants without cloud reliance, enhancing security, privacy, and device compatibility. It highlights local data processing and reducing dependency on external services for broader adoption.
3.	Desktop AI Assistant: J.A.R.V.I.S Just A Rather Very Intelligent System	Aishwarya C Maharajpet, Prof. Varsha S Jadhav, Ananya M Pachamukhi, Pranav Adagatti, Varshini. S. Gondkar,	The paper introduces JARVIS, an intelligent Python-based personal assistant using NLP and AI to simplify human-computer interaction. It details system architecture including speech recognition. Highlights include extensibility, modularity, and potential for AI-driven innovation while addressing challenges in stability and security.

1.5 Literature Survey Table

Chapter 2

SYSTEM ANALYSIS

System analysis involves studying and understanding the requirements, functionality, and design of the "Jarvis Virtual Assistant" to ensure its effectiveness and reliability in achieving the desired objectives. This section outlines the statement of the problem, feasibility analysis, system requirements, and overall system architecture.

2.1 Existing Systems

In the existing system, virtual assistants were primarily limited in their functionality and adaptability. They relied on basic voice recognition for executing predefined commands, such as opening a few websites, retrieving the current date or time, or providing simple responses to user queries. These systems lacked the ability to handle diverse or complex queries and offered no personalization or advanced task automation features. The lack of integration with a wide range of applications or platforms limited their usefulness to basic interactions.

Lastly, the existing systems were not extensible or scalable, lacking features to integrate additional functionality or adapt to evolving user needs and expectations.

2.2 Proposed Systems

The proposed system builds on these limitations and introduces several advanced features and capabilities. This enhanced system includes a highly interactive voice-controlled virtual assistant named Jarvis. It employs **AI-powered speech recognition and text-to-speech synthesis** to provide seamless and responsive user experience. Jarvis can now handle complex commands, perform advanced tasks like opening a broad range of applications (e.g., YouTube, Instagram, LinkedIn, and Gmail), and even execute voice-driven web searches for unrecognized commands.

It provides detailed explanations, personalized greetings, and tailored responses to enhance user interaction. While it still requires an internet connection, this ensures real-time access to its expanded functionalities, such as scheduling tasks, playing YouTube videos, and searching the web efficiently. These enhancements make the proposed system more robust, versatile, and capable of addressing the limitations of the previous virtual assistant designs.

Chapter 3

System Design

The development and operation of the "Jarvis Virtual Assistant" require a set of hardware and software resources to ensure its functionality and efficiency. These requirements are categorized into hardware, software, and environmental specifications.

3.1 Functional Requirements

1. **Voice Interaction:** Jarvis should greet the user based on the time of day. The system should process voice commands using speech recognition.
2. **Command Execution:** Ability to open specific websites like YouTube, Google, Instagram, and Gmail upon receiving commands. Provide the current time and date upon request. Answer personal queries like "Who are you?" or "What is your name?". Perform Basic Calculations and Play user interested songs and videos in YouTube.
3. **Fallback Behavior:** For unrecognized commands, Jarvis should search the web and provide relevant results.
4. **Interactive Elements:** Show/hide a microphone button dynamically based on interaction.
5. **Error Handling:** Handle unrecognized speech gracefully by suggesting internet searches.
6. **Extensibility:** Support integration with additional platforms like WhatsApp, Twitter, Spotify, etc.

3.2 Non – Functional Requirements

1. **Usability:** Maintain a user-friendly interface with visually appealing designs, such as the futuristic and sleek black background. Ensure responsive behavior across different devices and screen sizes.
2. **Performance:** Speech recognition and text-to-speech should work with minimal latency. Handle multiple commands efficiently without noticeable delays.
3. **Reliability:** Jarvis should respond accurately to commands and maintain consistent functionality.
4. **Scalability:** Future-proof the assistant for more commands, integrations, and functionalities.
5. **Accessibility:** Ensure that the voice and visual elements are comprehensible for all users, including those with limited technical skills.
6. **Maintainability:** Keep the JavaScript, CSS, and HTML code modular and well-documented for easier updates or bug fixes.
7. **Compliance:** Adhere to web standards and best practices, ensuring compatibility with modern browsers.

3.3 Software Requirements

- **Backend:** JavaScript for an interactive user interface.
- **Frontend:** HTML, CSS and JavaScript.
- **Development Tools:** VS Code or any IDE.

3.4 Hardware Requirements

- **Processor:** Intel Core / Ryzen processors
- **RAM:** 4GB or higher
- **Storage:** 1GB or higher
- **Operating System:** Compatible with Windows
- **Network Connectivity:** Required for real time voice recognition, online search capabilities, and app access.

Chapter 4

Methodology

4.1 LLM to perform ‘Natural language processing (NLP)’

1. Speech signal processing

- **Objective:** Prepare raw speech input for feature extraction by enhancing its quality and eliminating noise.

- **Steps:**
    - Apply filtering techniques to remove background noise and irrelevant frequencies.
    - Normalize the amplitude of the speech signal for consistent processing.
    - Segment the speech signal into smaller frames for further analysis.
- **Outcome:** A clean and normalized speech signal ready for feature extraction.

2. Feature Extraction

- **Objective:** Extract meaningful features from the speech signal that represent its acoustic properties

- **Techniques Used:**
    - **MFCC (Mel-Frequency Cepstral Coefficients):** Capture the short-term power spectrum of the speech signal.
    - **Spectrogram Analysis:** Represent the signal’s frequency over time.
    - **Log Energy Analysis:** Extract information about the intensity of speech signals.
- **Outcome:** Feature vectors that encapsulate the speech signal’s essential characteristics.

3. Acoustic Modeling

- **Objective:** Map the extracted features to phonetic units or acoustic representations of speech.
- **Implementation:**

- Use pre-trained acoustic models based on Hidden Markov Models (HMMs) or Deep Neural Networks (DNNs) to identify phonemes.
- Align the input speech features with the corresponding phonetic units.

- **Outcome:** Probabilistic mappings of acoustic features to phonemes, aiding in recognition.

4. Phonetic Unit Recognition

- **Objective:** Recognize and group phonetic units to form coherent words or syllables.

- **Implementation:**
  - Use statistical algorithms or neural networks to combine phonetic units.
  - Account for co-articulation and contextual variations in speech sounds.
- **Outcome:** Detected phonetic units form the foundation for constructing words

5. Language Modeling

- **Objective:** Construct meaningful sentences by applying linguistic rules and context to the recognized phonemes and words.

- **Implementation:**
  - Use pre-trained language models, such as n-gram models or LLMs, to analyze word sequences.
  - Employ contextual understanding to resolve ambiguities and improve accuracy.
- **Outcome:** A decoded message representing the user's spoken input in textual form.

6. Decoded Message

- **Objective:** Produce the final text output that reflects the user's speech input.
- **Outcome:** The decoded message is passed to the system for intent recognition and further action.

4.2Block Diagram

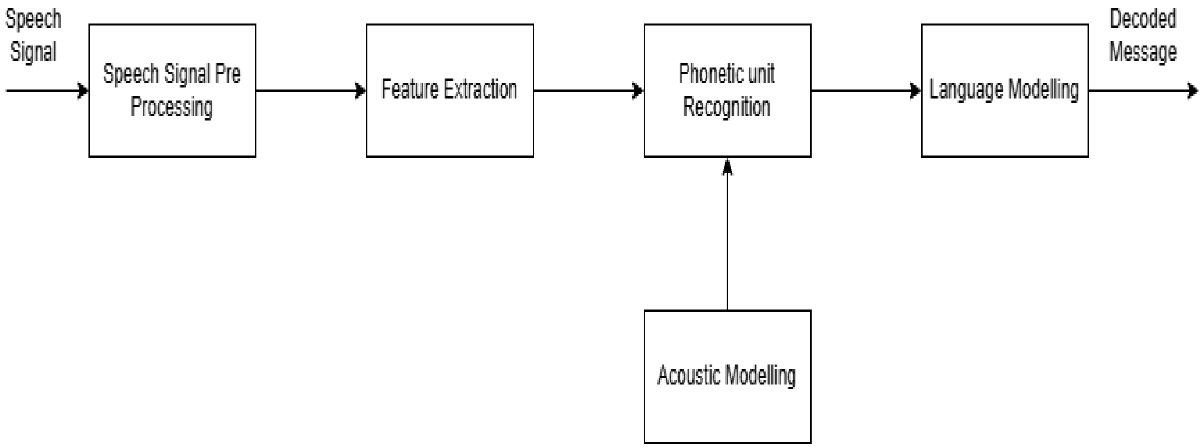

Fig 4.1 Block Diagram representing Voice assistant

Chapter 5

Design

5.1 Sequence Diagram

User Interaction:

- The process begins with the user providing an input speech signal and ends with the user receiving the decoded message.

 Speech Signal Pre-processing:

- The system processes the raw speech input to enhance its quality, removing noise and normalizing the signal for feature extraction.

 Feature Extraction:

- Extracts key acoustic features from the processed speech signal, representing its characteristics for further analysis.

 Phonetic Unit Recognition:

- Maps extracted features to phonetic units (syllables or phonemes) by interacting with the **Acoustic Modelling** module.

 Acoustic Modelling:

- Supplies acoustic models that help identify phonetic units accurately based on the extracted features.

 Language Modelling:

- Converts recognized phonetic units into meaningful words or sentences by applying contextual and grammatical rules.

 Decoded Message Output:

- The system delivers the final decoded text message back to the user.

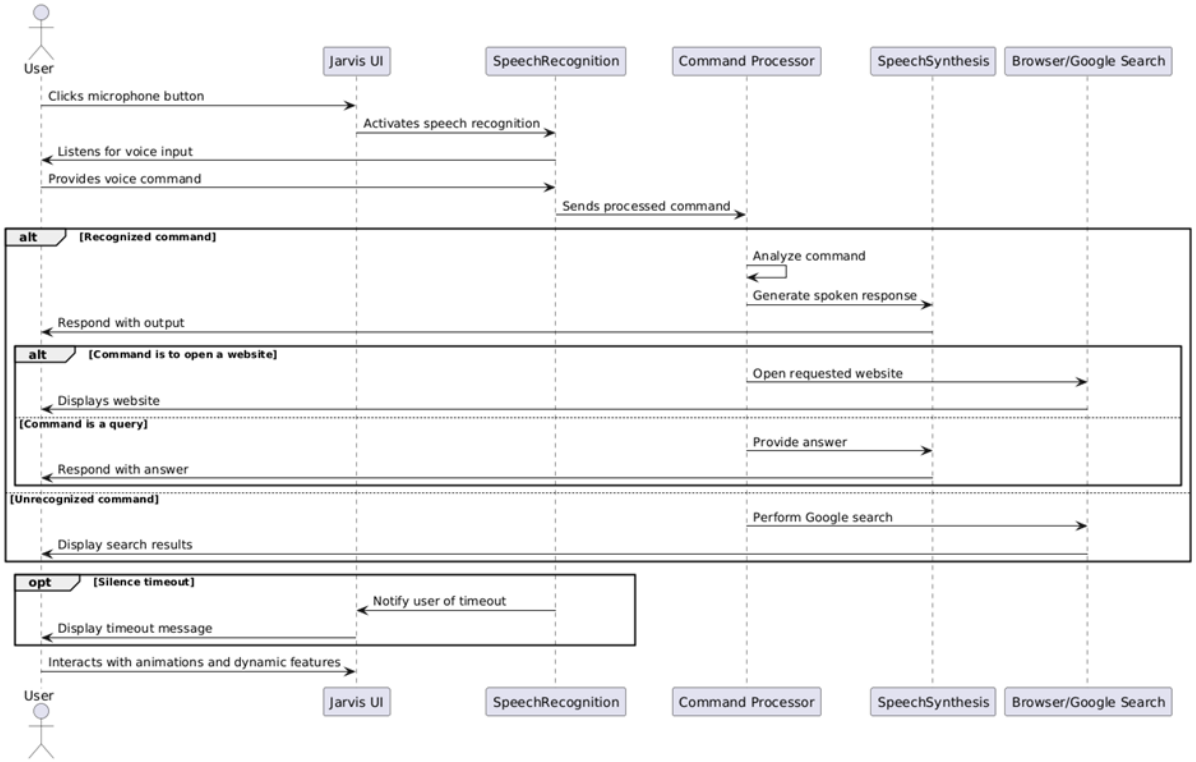

Fig: 5.1 Sequence Diagram

5.2 Use Case Diagram

1. Speech Signal Preprocessing

- Input: **The actor provides a speech signal as the primary input.**
- Process: **The system preprocesses the speech signal to enhance quality by removing noise, normalizing amplitude, and preparing it for feature extraction.**
- Output: **A processed signal suitable for further analysis.**

2. **Feature Extraction**

- **Input:** The processed signal from the preprocessing stage.
- **Process:** The system extracts essential acoustic features using techniques like MFCC or spectrogram analysis, capturing meaningful patterns in the speech data.
- **Output:** Feature vectors representing the speech signal's characteristics.

3. Phonetic Unit Recognition

- **Input:** Feature vectors from the extraction stage.

- **Process:** The system maps feature to phonetic units (e.g., syllables or phonemes) with the help of acoustic modeling.
    - **Acoustic Modeling:** Supplies pre-trained acoustic models that aid in identifying phonetic units accurately by considering variations in tone, pitch, and pronunciation.
- **Output:** A sequence of phonetic units for further processing.

4. Language Modelling

- **Input:** Recognized phonetic units.
- **Process:** The language model analyzes the phonetic units to form meaningful words and sentences. It applies contextual and grammatical rules to ensure accurate decoding.
- **Output:** A fully decoded message reflecting the actor's speech input.

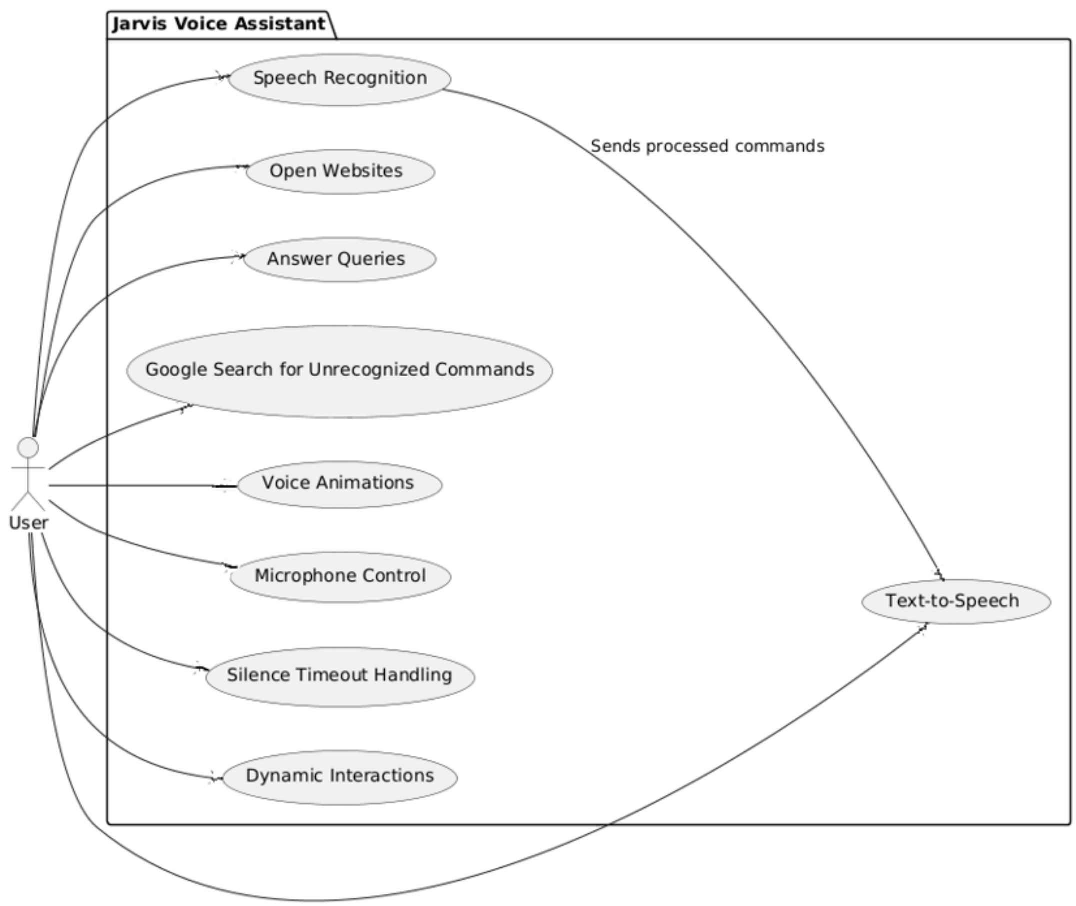

Fig 5.2 Use Case Diagram

Chapter 6

Implementation

The speak (text, callback) function uses the SpeechSynthesisUtterance API to convert text into speech with adjustable properties like rate, pitch, volume, and language. A callback is triggered once the speech ends, enabling further actions after the spoken message. The wishMe () function greets the user based on the time of day, introducing itself as "Jarvis" and using the speak function for the greeting. After the greeting, it displays a button that allows the user to start interacting with the assistant.

The play YouTube Song (song Name) function enables Jarvis to search for songs on YouTube using the YouTube Data API. It fetches the first video result based on the user's input and opens it in a new tab. If no result is found or there's an error, it provides a voice response informing the user. The take Command(message) function processes user commands, interpreting various tasks like playing songs, opening websites, performing calculations, telling the time or date, or responding to personal queries. For unrecognized commands, it defaults to searching the query on Google.

The calculation (msg) function extracts mathematical expressions from the user's message and evaluates them using JavaScript's eval function. It supports basic arithmetic operations, powers, and square root calculations. Event listeners are set up to manage interactions; when the page loads, the wishMe () function is automatically triggered. The btn. addEventListener ("click", ...) initializes speech recognition when the user clicks the button, and toggles the display of the button and voice animation.

Finally, the recognition. On result function handles the speech recognition results, displaying the transcript and passing it to take Command for processing. To manage silences, recognition. On speech end recognition stops after five seconds of inactivity, while recognition. On speech start resets the timer when the speech resumes. The recognition. On end function resets the interface when recognition stops, ensuring a smooth interaction. Together, these functions form a seamless virtual assistant capable of voice-controlled interactions.

Snapshots

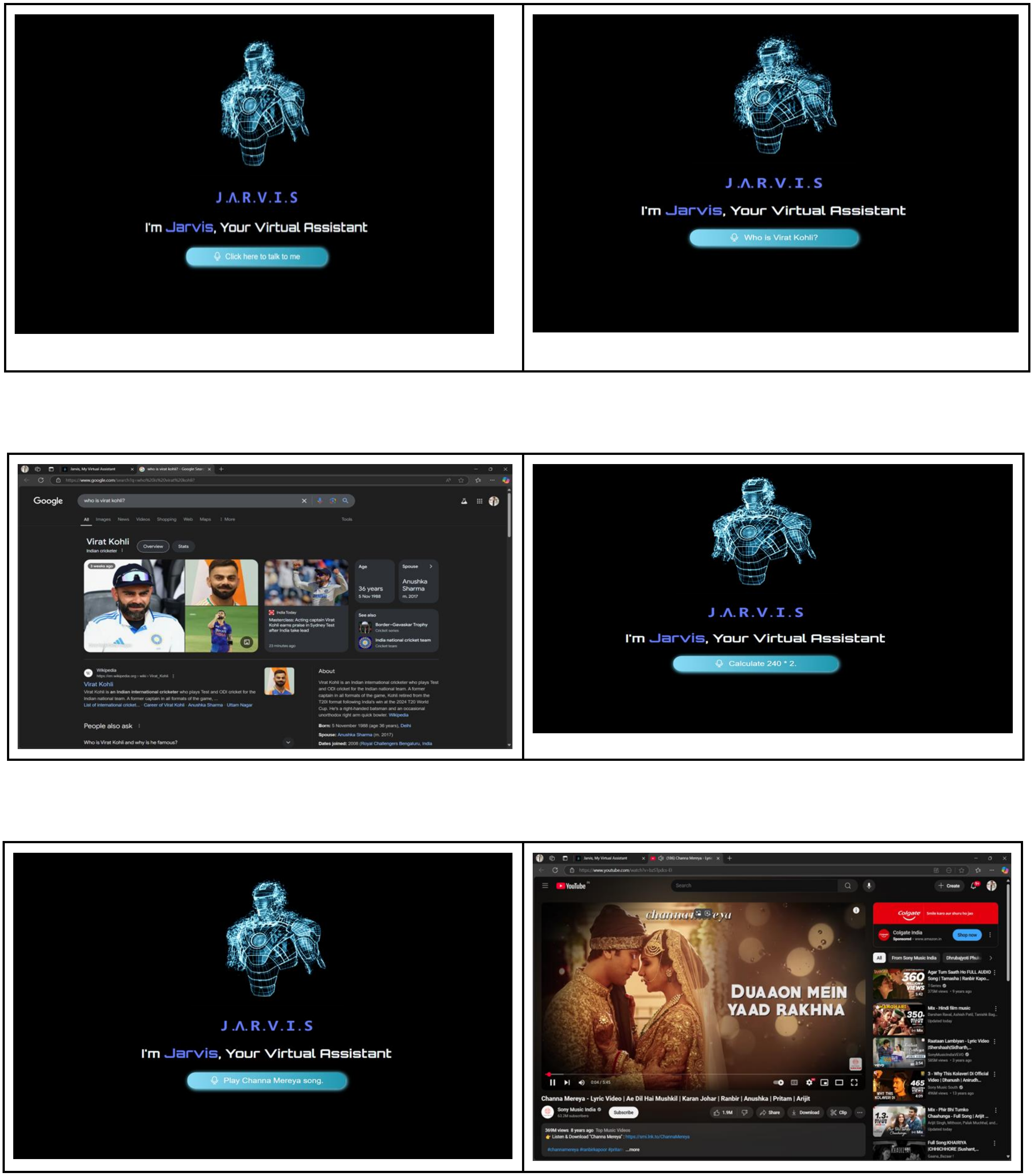

Fig 6.1 Snapshots

Chapter 7

Testing

Test Case ID	Test Scenario	Input	Expected Output	Pass/Fail
TC01	Play video / song	“Play name of song”	Assistant starts playing song.	Pass.
TC02	Web search Functionality	“Search live cricket score”	Assistant performs a web search and displays relevant results.	Pass.
TC03	Basic Calculation	“What is 23 multiplied by 7?”	Assistant responds with the correct answer (161).	Pass.
TC04	Opening Applications	“Open WhatsApp”	Assistant launches the WhatsApp application.	Pass.

Fig 7.1 Test Cases with scenarios, Input, Expected Output and Results.

Conclusion

The J.A.R.V.I.S. project demonstrates the effective use of modern web technologies to create an intuitive and user-friendly virtual assistant voice-controlled experience. Jarvis efficiently integrates speech recognition and synthesis to facilitate seamless interaction, responding dynamically to user commands and queries.

Through its intuitive design, vibrant UI, and robust functionality, Jarvis showcases the potential of web-based AI tools for enhancing everyday tasks. The project highlights the importance of combining advanced technologies with creativity to provide practical, engaging, and futuristic solutions.

Future improvements could include adding custom integrations, expanding the assistant's capabilities, and incorporating machine learning for even smarter interactions.

References

1. Deepak Shende, Ria Umahiya, Monika Raghorte, Aishwarya Bhisikar, Anup Bhange, "AI Based Voice Assistant Using Python", Journal of Emerging Technologies and Innovative Research(JETIR), Published in: Volume 6 | Issue 2 | Year February – 2019 | page number 506 – 509 e-ISSN: 2349-5162, Country: Nagpur, Maharashtra, India.
2. Susmita Das, Susmita Das, Ankita Saha, Aparupa Chakraborty, Swagata Thakur, Sumanta Chatterjee, "J.A.R.V.I.S - An AI Technology Based Personal Windows Assistant System", International Journal of Research and Analytical Reviews (IJRAR.ORG) 2022 IJRAR, Published in: Volume 9 | Issue 2 | Year May – 2022 | page number 385 – 391, ISSN Approved Journal No: E-ISSN 2348-1269, P- ISSN 2349-5138, Country: Kalyani, India, JIS College of Engineering (An Autonomous Institution), Kalyani, India.
3. Rajat Sharma , Adweteeya Dwivedi. "JARVIS - AI Voice Assistant". International Journal of Science and Research (IJSR) Published in: Volume 11 | Issue 5 | Year May – 2022 | page number 386 – 393, SSN: 2319-7064 Paper ID: SR22503183839, Country: SRM Institute of Science &Technology, Delhi NCR Campus, U.P., India
4. Aishwarya C Maharajpet, Prof. Varsha S Jadhav, Ananya M Pachamukhi, Pranav Adagatti, Varshini. S. Gondkar, "DESKTOP AI ASSISTANT: J.A.R.V.I.S JUST A RATHER VERY INTELLIGENT SYSTEM", Journal of Emerging Technologies and Innovative Research(JETIR), Published in: Volume 11 | Issue 3 | Year March 2024 | page number f685 – f691 Country: Dharwad, India.