International Scientific Journal of Engineering and Management
Volume: 03 Issue: 04 | April – 2024
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata
ISSN: 2583-6129
DOI: 10.55041/ISJEM01562

Voice Web : Your Intelligent Virtual Assistant

Kushal K., Manikanta H. M.,Mithun D. Raj., Mownith H. S., Dr. Keerti Kumar H. M
Department of CSE, Malnad College of Engineering, Hassan.

Abstract: — Automatic voice-controlled systems have changed the way humans interact with a computer. Voice or speech recognition systems allow a user to make a hands-free request to the computer, which in turn processes the request and serves the user with appropriate responses. After years of research and developments in machine learning and artificial intelligence, today voice-controlled technologies have become more efficient and are widely applied in many domains to enable and improve human-to human and human-to-computer interactions. The state-of-the-art e-commerce applications with the help of web technologies offer interactive and user-friendly interfaces. However, there are some instances where people, especially with visual disabilities, are not able to fully experience the serviceability of such applications. A voice-controlled system embedded in a web application can enhance user experience and can provide voice as a means to control the functionality of doctor's appointment booking websites. In this paper, we propose the integration of Alan AI enabled natural language processing capabilities, allowing users to navigate through tabs and pages using voice commands. This intuitive interface enhanced user experience and accessibility, making it easier for users to find relevant information and schedule appointments seamlessly. Additionally, Annyang, a JavaScript library for speech recognition, was employed to streamline the appointment booking process. Users could verbally input their details into the appointment form, such as name, contact information, and preferred appointment time, reducing manual input and improving efficiency.
Keywords: Alan AI, Annyang, Speech-to-text, Natural Language Processing, Speech Recognition Systems (SRS).

I. INTRODUCTION

The concept of virtual assistants in earlier days is to describe the professionals who provide ancillary services on the web. The job of a voice is defined in three stages: Text to speech; Text to Intention; Intention to action; Voice assistant will be fully developed to improve the current range. Voice assistants are not befuddled with the virtual assistants, which are people, who work casually and can therefore handle all kinds of tasks. Voice Assistants anticipate our every need and it takes action, Thanks to AI based Voice Assistants [1]. One of the goals of Artificial intelligence (AI) is the realization of natural dialogue between humans and machines. In recent years, the dialogue systems, also known as interactive conversational systems are the fastest growing area in AI. Many companies have used the dialogue systems technology to establish various kinds of Virtual Personal Assistants (VPAs) based on their applications and areas [2].

.

Alan AI is an end-to-end conversational platform to build robust and reliable AI assistants and chatbots. The Alan backend takes care of most of the heavy lifting, including creating spoken language models, training speech recognition software, deploying and hosting conversational components. To create a conversational experience, you only need one developer, not a group of Machine Learning and Devops specialists [3].

Over the years, various web assistive technologies like screen readers, special browsers, and screen magnification techniques have been developed to address the problems faced by visually impaired web users. These systems have helped the users either by reading, enabling voice commands, or providing ways for screen magnification to comprehend the contents on a web page. However, most of these solutions have failed in terms of accuracy as the frequency of misinterpretation is often high.

Web applications equipped with voice-enabled systems can not only provide flexibility in terms of users' choice of web interaction but can also increase the usability of the applications for the general users when they are unable to use the traditional human-computer interaction mechanisms. By allowing users to control the functionality of the applications with their voice, SRS can enhance users' browsing experience, and allow users to effectively convey their instructions and requests using natural languages.

This project has a main purpose of helping the people who are differently abled and to help those who face problems navigate through the website while browsing for certain information. Designed solution is an application, based on speech recognition, semantic analysis, user interaction, and braille engine, which can use as a smart programming tool for visually impaired and differently abled people. It' s a method of implementing a new way to help and motivate disable people, to use latest technologies in Information Technology and software engineering aspects [4].

I. RELATED WORK

Kandhari et al. [5], presents a study of the state-of-the-art speech recognition systems and propose a taxonomy of SRS. We also present a voice-controlled e-commerce application using IBM Watson speech-to-text service as a part of a comparative study with other speech-to-text systems such as Google and Amazon.

International Scientific Journal of Engineering and Management
Volume: 03 Issue: 04 | April – 2024
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata
ISSN: 2583-6129
DOI: 10.55041/ISJEM01562

Gaida et al. [6], described a large scale evaluation of open-source speech recognition toolkits. The main contributions are the inclusion of Kaldi to a comprehensive evaluation of open-source ASR systems and the usage of standard corpora making our results comparable to other publications in the field.

Christian et al. [7], describes that the characteristics of web navigation may be advantageous to voice-controlled navigation. A small number of commands can provide the navigation functionality common to visual browsers, and current technology is effective for small vocabularies. Although there is a small set of commands, users must typically speak the text of the link to follow the link.

Bhalerao et al. [8], describes that they have presented a web-based service that is news application using ALAN AI with an interactive voice assistant which gives user a simplified version of application. This helps to the people who have very busy schedule and have difficulty in reading. The main advantage of this application is that it is voice based so it helps to interact with platform by voice commands.

Chaprana et al. [9], describes the proposed idea of the Voice Controlled Web Application provides a simplistic approach and ease to the user. The service provides all the components required for a user to be able to use voice and speech as a medium to find and look for news about his/her choice and the option to go through the news in a very concise or in very detailed manner.

II. Methodology

The project design involves implementing voice-enabled features to enhance user experience. Utilizing HTML, CSS, and JavaScript (with React.js for dynamic components) for frontend development, Node.js with Express.js for backend logic, and Alan AI SDK for voice recognition, the design focuses on voice-based appointment scheduling, FAQ assistance, and website navigation. The user interface design prioritizes intuitive interaction, responsive layout, and clear visual cues. Thorough testing, including usability testing with real users, ensures accuracy and usability across devices and browsers. Documentation, training resources, and a structured deployment plan facilitate smooth integration and maintenance post-deployment. This design emphasizes iterative improvements based on user feedback and technological advancements to continually enhance the website's functionality and user satisfaction.

A. BLOCK DIAGRAM

Phase 1: Speech Command Analysis

In the initial phase, users interact with the virtual assistant by providing speech commands. The virtual assistant selectively captures and analyzes the main and pertinent words from the user's speech input. This analysis involves various natural language processing (NLP) techniques, such as speech-to-text conversion and keyword extraction. By focusing on extracting relevant keywords and phrases, the virtual assistant aims to understand the user's intent accurately. Additionally, contextual information and user history may be considered to enhance the accuracy of speech command interpretation. This phase sets the foundation for further processing by converting the captured speech into a text format that can be easily processed by the system.

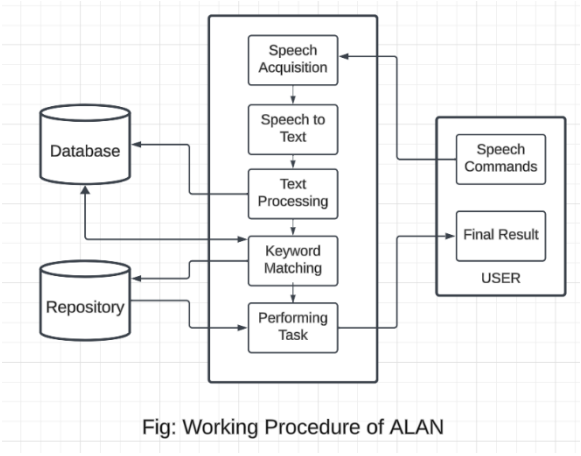

Fig.1 Block Diagram

The methodology for utilizing the virtual assistant encompasses three distinct phases**:-**

Phase 2: Text Processing and Data Comparison

Following the conversion of speech into text format, the system enters the second phase, where the converted text undergoes further processing with Alan AI. This processing involves leveraging Alan's natural language understanding capabilities to interpret the user's query or command. Alan AI employs sophisticated algorithms to analyze the text input and extract relevant keywords and entities. These keywords and entities are then compared against the data stored within the system's database or knowledge base, which contains information related to appointment schedules, medical services, FAQs, and other relevant data.

Phase 3: Action Selection and Execution

In the final phase, the system selects the appropriate action based on the identified keyword or phrase from the previous phase. The identified keyword is cross-referenced within the repository to determine the corresponding action to be executed. This action may involve various tasks such as scheduling appointments, retrieving information, providing responses to queries, or executing specific commands. Once the appropriate process is selected, the system executes the task seamlessly. This execution may involve interacting with external systems or services, updating database records, or generating responses for the user. The culmination of this iterative methodology ensures efficient communication and seamless task execution between the user and the virtual assistant, enhancing the overall user experience.

III. IMPLEMENTATION DETAILS

International Scientific Journal of Engineering and Management
Volume: 03 Issue: 04 | April – 2024
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata
ISSN: 2583-6129
DOI: 10.55041/ISJEM01562

In this section, we describe about the implementation details of our website.

A. Software Components

The front-end development is based on the React JS framework, which is a library from the component-based JavaScript libraries and frameworks. It enables the creation of fast and vivid user interfaces without reloading the entire page. The structure of React JS allows ensuring both intuitive and smooth user experience while enabling content updates on various gadgets and across a vast array of existing platforms. Front-end involves implementing interactive components and the proper management of state, which means responding quickly to users' actions and sending requests to backend services and external APIs. It also includes adding pictures and other media assets to the pages.

Back-end development is a combination of various technologies that support vocal interaction and scalable server-side operations. Alan AI's conversational AI capabilities are advanced enabling natural language processing and voice-based personalized interactions. Annyang is a lightweight JavaScript library which further enhances voice-controlled functionality through simple integration and easy command definition. Node.js is used for such general server side tasks as data processing, server management, API integrations among others making possible to have scalable and efficient backend operations. These back- end parts function together in order to create strong interactive system facilitating both normal as well as voice based interfaces.

B. Initialize ALAN

```
npm install @alan-ai/alan-sdk-web --save
```

```
import alanBtn from '@alan-ai/alan-sdk-web';
```

```
useEffect(() => {
  alanBtn({
      key: '8ac6e4f9f57426f15e0de645666aaf8e2e956eca572e1d8b807a3e2338fdd0dc/stage',
      onCommand: (commandData) => {
        if (commandData.command === 'go:back') {
          // Call the client code that will react to the received command
        }
      }
  });
}, []);
```

Fig. 2 Initializing ALAN

Alan AI can be initialized in a React application by setting up the `alanBtn` function with a unique API key so as to establish connection with Alan AI service. With `useEffect` hook, the developers ensure that it runs when the component is first mounted to enable voice assistant listen for voice commands immediately. The `onCommand` event handler helps in reacting to certain commands thereby allowing voice controlled interactions within the app. For this reason, establishing this kind of environment is necessary if one wants to integrate conversational AI capabilities in a frontend developed using React, which will also provide users with seamless vocal interface for carrying out tasks or navigating through an app.

C. Setting-up ALAN to navigate to different pages

```
function App() {
  const navigate = useNavigate(); // Initialize navigate
  const [alanBtnInstance, setAlanBtnInstance] = useState(null);
  const [alanBtnVisible, setAlanBtnVisible] = useState(false);
  useEffect(() => {
    const alanKey = '8ac6e4f9f57426f15e0de645666aaf8e2e956eca572e1d8b807a3e2338fdd0dc/stage';

    const instance = AlanBtn({
      key: alanKey,
      onCommand: (commandData) => {
        if (commandData.command === 'goTo') {
          const route = commandData.route;
          // Navigate to the specified route using React Router's navigate method
          navigate(`/${route}`);
          // Return a voice message confirming the action
          return `Opening ${route} page.`;
        }
      }
    });

    // Enable audio output
    instance.textChat.setAudioOutputEnabled(true);

    setAlanBtnInstance(instance);

  }, [navigate]); // Update useEffect dependency to include navigate
```

Fig. 3 Adding ALAN for navigation

This code snippet initializes Alan AI within a React application, enabling voice-based interactions and navigation. In the `App` component, React Router's useNavigate` hook is used for programmatic navigation, while `useState` manages the Alan AI instance and its visibility. The `useEffect` hook initializes the Alan AI instance using a unique key and sets up an `onCommand` callback to handle voice commands. If a 'goTo' command is detected, the app navigates to the specified route using `navigate`, providing a hands-free navigation experience.
Additionally, the Alan AI instance is configured to enable audio output, allowing the application to respond with voice feedback, thereby enhancing user engagement and interactivity. This setup demonstrates how to create a voice-responsive interface in a React environment, offering users a more natural and intuitive way to interact with the application.

D. ALAN speech recognition data

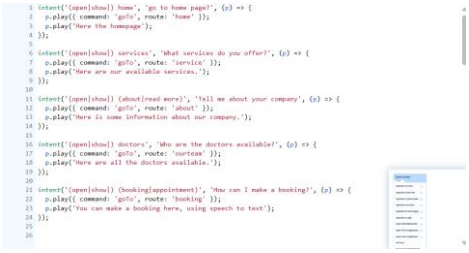

Fig. 4 ALAN studio code

This code snippet defines several voice command intents for a conversational AI system. Each `intent` function listens for specific phrases and executes an action when a matching phrase is detected. The intents cover different application sections, enabling users to navigate through voice commands. The first intent listens for variations of "open" or "show" along with "home," navigating to the home page and providing voice

International Scientific Journal of Engineering and Management
Volume: 03 Issue: 04 | April – 2024
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata
ISSN: 2583-6129
DOI: 10.55041/ISJEM01562

feedback. Similarly, other intents guide users to various sections, such as services, company information, doctors, and booking. The system responds with a `goTo` command to navigate to the appropriate route and plays a confirmation message to give users auditory feedback. This setup provides an intuitive voice interface, allowing users to interact with the application through natural language and enabling a seamless and accessible navigation experience.

E. ANNYANG Implementation

```
useEffect(() => {
  if (annyang) {
    annyang.addCallback('result', (phrases) => {
      setVoiceInput(phrases[0]); // Update voice input text
    });

    annyang.addCommands({
      'enter name *tag': (variable) => fillFormField('patientName', variable),
      'enter age *tag': (variable) => fillFormField('age', variable),
      'select gender *tag': (variable) => fillFormField('gender', variable),
      'enter phone number *tag': (variable) => fillFormField('mobile', variable),
      'enter address *tag': (variable) => fillFormField('address', variable),
      'enter email *tag': (variable) => fillFormField('email', variable),
      'select doctor *tag': (variable) => fillFormField('doctor', variable),
      'enter date *tag': (variable) => function(variable: any): void e),
      'select time *tag': (variable) => fillFormField('time', variable),
      'book now': handleSubmit,
      'reset': handleReset,
    });
    if (voiceEnabled) {
      annyang.start();
    } else {
      annyang.abort();
    }
  }
}, [voiceEnabled]); // added voiceEnabled as dependency

const fillFormField = (field, value) => {
  setFormData(prevState => ({
    ...prevState,
    [field]: value
  }));
};
```

Fig. 5 Annyang implementation

The code snippet initializes voice recognition in a React component using Annyang, a lightweight JavaScript library for speech recognition. It utilizes the `useEffect` hook to configure Annyang when the component mounts or when the `voiceEnabled` state changes. The setup adds a callback to capture the spoken phrases, updating the `setVoiceInput` state with the first recognized phrase. Additionally, a set of voice commands is defined to interact with a form. Commands include entering values for various fields (like name, age, gender, phone number, address, email, doctor, date, time) and triggering specific actions like booking an appointment (`book now`) or resetting the form (`reset`). Each command maps to a function that fills the relevant form field or handles specific actions. The component conditionally starts or stops Annyang based on the `voiceEnabled` state, allowing users to control voice interaction. This approach integrates voice commands into a form-based application, enhancing accessibility and user interaction through speech recognition.

IV. RESULT AND ANALYSIS

Home Page

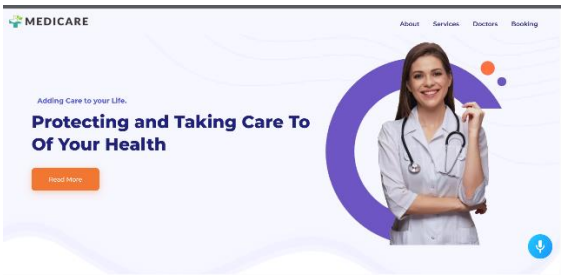

Fig. 6 Home Page

The above Fig.2 shows the picture of the home page of our doctor's appointment booking website where we have multiple tabs to navigate at the top which includes about, which provides the details about the hospital. Next we have Services, which provides the information about all the services offered by the hospital. Coming up next we have Doctors, where we can find the information about all the doctors available and their respective details. Lastly we have Appointment, where we can book an appointment for any specific doctors by entering the required details about the user.
There is also the ALAN button present at the bottom right corner of every page which can be used to enable\disable the speech recognition system which helps to navigate through different tabs just by using the voice commands.

Doctors Page

Fig. 7 Doctors Page

Here is the Doctors page shown below in the Figures.3 where the user can navigate through the doctors available and also get details regarding each and every doctor.
There are three different categories of specialized doctors, those are Dentist, Cardiologists and Neurologists. Each of these categories consist of three specialized doctors respectively. In this page, we can see that we have mentioned each doctor's qualification in their respective fields to make it clearly understandable

ISSN: 2583-6129
DOI: 10.55041/ISJEM01562

for the users. The users can also go through the details about each doctors.

Appointment Page

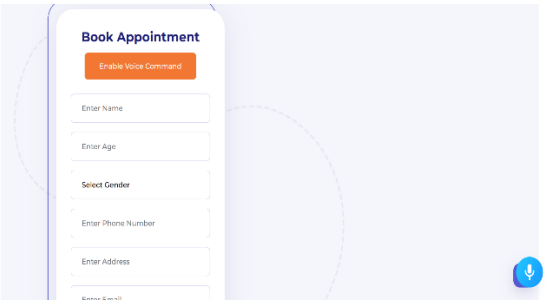

Fig. 8 Appointment Page

The above Fig.4 displays where the user can make a booking even manually or they can enable the annyang library with the button which helps them to fill the details of the form using speech to text. Here the user needs to just use his voice for the filling the contents of the form and also submit it using the voice command itself.

V. CONCLUSION

This project is a pioneering initiative in contemporary web development. It accomplishes an amazing combination of imagination and practicality by blending such cutting-edge technologies as React, Alan AI, Annyang, and JavaScript. By tactically merging these tools, VoiceWeb redefines user interaction, creating a critical departure from traditional web interfaces.

The core foundation of VoiceWeb revolves around the dynamic component-based architecture provided by React that creates an interface which highly adapts to a user's input and preference thereby ensuring smooth user experience across all available devices. The inclusion of Alan AI and Annyang takes the project into the arena of intelligent agents enabling visitors to navigate through site content using spoken commands as well as voice recognition software.

In addition, this VoiceWeb has JavaScript on both its front end and back end making it highly flexible and versatile than any other application ever had. Being extensively deployed by developers due to its ubiquity and flexibility allows intricate logic implementation as well as real-time data processing thus facilitating seamless communication between a frontend interface and backend services. Taking VoiceWeb and other similar projects into consideration, it is safe to assume that this is only the formation stage of a new trend in web development, which in the future will allow creating projects richer and more human-oriented than the current web . This is a significant leap in web development and it will stay with us for a long time.

ACKNOWLEDGMENT

The authors are grateful to Dr. Keerti Kumar H.M, Associate Professor, Department of Computer Science and Engineering, MCE Hassan for guiding at every stage of the work.

REFERENCES

[1]. Chowdhury, S. S., Talukdar, A., Mahmud, A., & Rahman, T. (2018). Domain specific intelligent personal assistant with bilingual voice command processing. In TENCON 2018-2018 IEEE Region 10 Conference (pp. 0731–0734). IEEE.
[2]. Kepuska, V., & Bohouta, G. (2018). Next-generation of virtual personal assistants (Microsoft Cortana, Apple Siri, Amazon Alexa, and Google Home). In 2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC) (pp. 99–103). IEEE.
[3]. Alan AI Documentation. (2017). Alan AI Documentation. Retrieved from https://alan.app/docs/
[4]. Lunuwilage, K., Abeysekara, S., Witharama, L., Mendis, S., & Thelijjagoda, S. (2017). Web-based programming tool with speech recognition for visually impaired users. In 2017 11th International Conference on Software, Knowledge, Information Management and Applications (SKIMA) (pp. 1–6). IEEE.
[5]. Kandhari, M. S., Zulkemine, F., & Isah, H. (2018). A voice-controlled e-commerce web application. In 2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON) (pp. 118–124). IEEE.
[6]. Gaida, C., Lange, P., Petrick, R., Proba, P., Malatawy, A., & Suendermann-Oeft, D. (2014). Comparing open-source speech recognition toolkits. In 11th International Workshop on Natural Language Processing and Cognitive Science*.
[7]. Christian, K., Kules, B., Shneiderman, B., & Youssef, A. (2000). A comparison of voice-controlled and mouse-controlled web browsing. In Proceedings of the Fourth International ACM Conference on Assistive Technologies (pp. 72–79).
[8]. Bhalerao, A., Singh, A., Goupale, M., Vaidya, S., & Gangwani, P. (2023). News application using alan ai. In Journal of Analysis and Computation (JAC), ISSN 0973-2861, Volume XVII, Issue I. ijaconline.
[9]. Chaprana, A., Kumar, R., Saini, A., & Kumar, A. (2021). Voice controlled news web application with speech recognition using alan studio. International Journal of Computer Applications, 183(2), 31–37.

★★★

ISSN: 2583-6129
DOI: 10.55041/ISJEM01562