Signvertor — A Hearing Aid Module For The Deaf

FINAL REVIEW REPORT

Submitted by

Sharadindu Adhikari 19BCE2105
Soumyadip Mondal 19BCE2107

for

CSE1901 – TECHNICAL ANSWERS FOR REAL WORLD PROBLEMS
PROJECT COMPONENT
C2+TC2

Under the Guidance of

Dr. A. Nagaraja Rao
Associate Professor, SCOPE
VIT University, Vellore

School of Computer Science and Engineering

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

NOVEMBER 2022

DECLARATION

We hereby declare that the report entitled "Signvertor — A Hearing Aid Module For The Deaf", submitted by us, for the CSE1901 Technical Answers for Real World Problems course to Vellore Institute of Technology is a record of bona fide work, carried out by us, under the supervision of Dr. A. Nagaraja Rao.

We further declare that the work reported in this report has not been submitted, and will not be submitted, either in part or in full, for any other courses in this institute, or any other institute or university.

Place: Vellore

Date: November 17, 2022

Soumyadip Mondal

Sharadindu Adhikari

Signatures of the Candidates

ACKNOWLEDGEMENTS

We would like to sincerely express our gratitude to all those who have helped us in the successful completion of this project. Without their support, it would not have been possible to achieve the goal of completing this project work successfully.

Taking this opportunity, we would like to thank our advisor, Dr. A. Nagaraja Rao, for his constant support, guidance and mentorship. Without his suggestions, it would have been extremely challenging for us to complete the project on time.

We would also like to give our thanks to Dean, who provided us with the faculties required and conductive conditions for the project.

Finally, we would like to express our sincere gratitude to Vellore Institute of Technology, which provided us with a platform to hone our skills over a period of last four years.

ABSTRACT

The most natural and expressive method for hearing-impaired people is sign language. Despite the fact that sign language is sufficient for communication between hearing impaired persons, there is an issue when non-disabled people attempt to interact with the hearing impaired. This study demonstrates a mechanism that can translate speech and text from a normal person into sign language using Indian Sign Language (ISL). 3D animations are used to express the Sign Language. The project follows HCI Principles and is created as a Web and Mobile application. The development process employs the Agile methodology, and users provide in-depth feedback and participate in surveys.

Keywords: Sign Language, ISL, User Experience, Natural Language Processing, Application Development

TABLE OF CONTENTS

Parameters	Page No.
Declaration	3
Acknowledgements	4
Abstract	5
Table of Contents	6
1. INTRODUCTION	**9**
2. BENEFICIARY OF THE PROJECT	**9**
3. RELATED WORK	**9**
4. METHODOLOGY	**10**
5. TOOL REQUIREMENTS	**12**
6. EXPERIMENTS AND RESULTS	**12**
7. CODE SCREENSHOTS	**15**
8. CONTRIBUTION	**17**
9. CONCLUSIONS AND FUTURE WORK	**17**
REFERENCES	**18**

1. INTRODUCTION

Every country has an increasing population of deaf people. There is only one way to communicate with this group of people, and that is through sign language. This dialect, if you can call it that, which has its own syntax and usage style helps the deaf learn to converse with the hearing. With an average of 93.7:6.3 in India, the ratio of deaf persons to those who can understand sign language is cause for concern. In the lack of interpreters who can translate between spoken languages and sign language, the project's objective is to assist the deaf, for whom sign language is a must for communication. Therefore, it is very helpful for those who want to interact face-to-face with the hearing-impaired community in everyday life, such as when performing tasks like teaching a class of hard-of-hearing students or delivering news to the public that includes deaf viewers.

This project was made specifically with Indian Sign Language in mind. It is an updated version of a system that translates speech into sign language and has new features and capabilities to increase adaptability to new tasks or semantic domains. It comprises of a speech recognizer that breaks down spoken phrases into utterances and silences and recognises it as text—a series of words—and a 3D film that shows how the spoken sentence is interpreted in sign language. It is a flexible technique that can considerably reduce the time and parallel corpus required for a speech to sign language translation system to be applied to a new domain.

2. BENEFICIARY OF THE PROJECT

Majorly people with hearing disabilities will benefit from our project. For example, let's say, someone is trying to initiate a conversation with another person, who is unfortunately deaf.

Our webapp will take in input (voice), convert the individual words to sign language, and display them for the deaf person to understand what they're being told. Win-win for everyone.

3. RELATED WORK

Deaf Mute Communication Interpreter - A Review [1]: The many deaf-mute communication technologies are discussed in this essay. Wearable communication devices and online learning systems are the two primary categories of communication approaches employed by the deaf-mute. There are three types of wearable communication systems: keypad technique, glove-based system, and Handicom touch-screen. Different sensors, accelerometers, micro-controllers, text-to-speech converters, etc. are used by many wearable technology gadgets. The second option, which uses online learning system technology, can eliminate the need for an external device to translate messages between deaf and non-deaf persons.

An Effective Wavelet Transform Framework for Indian Sign Language Recognition [2]: The suggested ISLR system is regarded as a pattern recognition method that includes the two crucial modules of feature extraction and classification. To recognise sign language, discrete wavelet transform (DWT)-based feature extraction and nearest neighbour classifier are combined. The experimental findings demonstrate that the proposed hand gesture recognition system when using a cosine distance classifier, achieves a maximum classification accuracy of 99.23%.

PCA-based hand gesture recognition in [3]: The authors of this study provided a method for database-driven hand gesture detection based on thresholding and a skin colour model approach, as well as an efficient template matching, which may be used for human robotics applications and related applications. The segmentation of the hand region begins with the use of the YCbCr colour space skin colour model. Thresholding is used in the following stage to distinguish between foreground and background. Principal Component Analysis (PCA) is used to construct a template-based matching technique for recognition.

Hand Gesture Recognition System For Dumb People [4]: The authors demonstrated a digital image processing-based system for recognising static hand gestures. SIFT technique is used to the feature vector for hand gestures. The edges that are resistant to scaling, rotation, and noise addition are where the SIFT features have been computed.

Automated System for Indian Sign Language Recognition in [5]: This work presents a method for automatically recognizing signs using shape-based attributes. Otsu's thresholding approach, which selects an ideal threshold to reduce the within-class variance of thresholder black and white pixels, is used to segment the hand region from the images. Hu's invariant moments are used to partition the hand region and calculate its features, which are then fed into an artificial neural network for categorization. Accuracy, Sensitivity, and Specificity are used as the metrics for measuring the system's performance.

4. METHODOLOGY

The proposed method to translate text or speech into equivalent sign language animation is described in this section. A number of separate procedures are combined to create the overall structure. Fig. 1 shows the system's overall architecture.

A. Getting the Input

This framework's first step is to gather input from the user. The user has two options for entering text: manually into the input field or using the microphone button to record live voice.

B. Speech Recognition

Using JavaScript Web Speech API, audio speech input is recognized if it is audio input, at which point the speech is converted to text and displayed in the input box.

C. Applying NLP Techniques

The free (unstructured) text is converted into normalized, structured data using a variety of Natural Language Processing (NLP) techniques. Tokenization, stop words removal, and lemmatization are all applied to the text sentence in order to remove only inflectional endings and return the base or dictionary form of a word.

D. Displaying ISL Animations

Each tokenized word is processed individually. The database is used to display the appropriate Indian Sign Language animations for each word. Additionally, if a word's animation is missing from the database, the word is spat out before the animations corresponding to each letter are displayed.

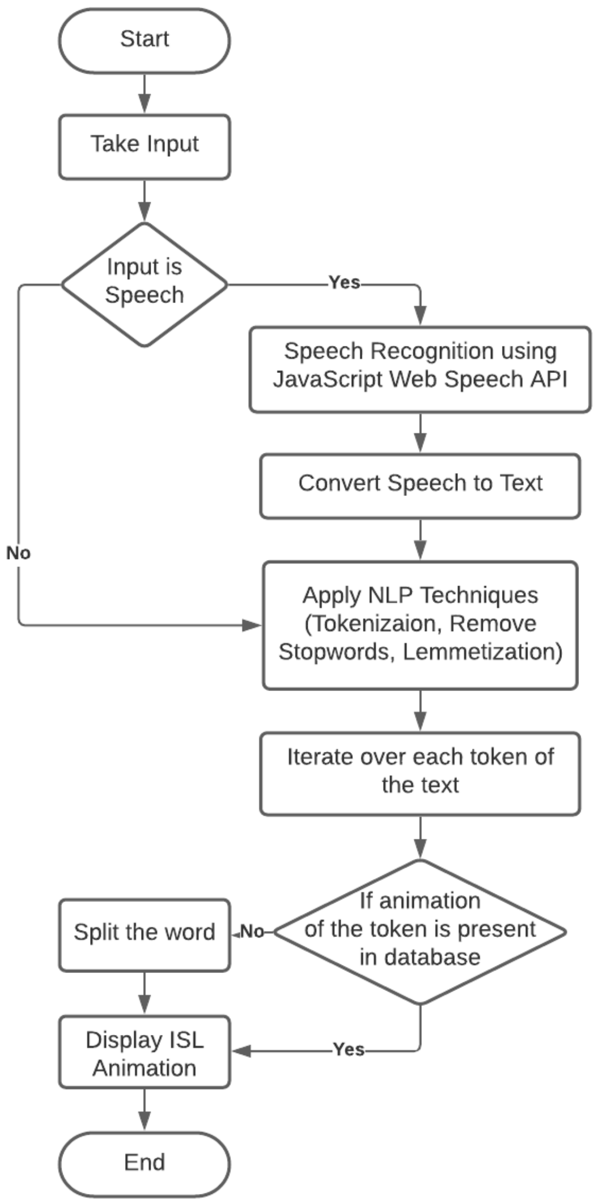


Figure 1: Overview of the proposed solution

5. TOOL REQUIREMENTS

Software Requirements:

- Python 3.7 or above
- Django Module
- WebSpeech Module
- NLTK

Hardware Requirements:

- Intel Core i5 8th gen or above
- 8 GB RAM
- Intel 9400 Quadro GPU or above

6. EXPERIMENTS AND RESULTS

We created the user interface with the following HCI standards in mind:

- Standardization and consistency.
- Beautiful and simplistic design.
- Request verification before performing any crucial, non-trivial actions.
- Permit quick reversal of the majority of activities.
- Allow user-customizable input (text or audio speech).
- Employed common terms, acronyms, and colours that are likely.
- To make decision-making simpler, Hick's Law is applied.
- Produce informative error messages.
- Classified various categories of content using windows (different tabs in the navigation bar).
- A tutorial is available to aid with error diagnosis and recovery.
- Provide enlightening commentary.

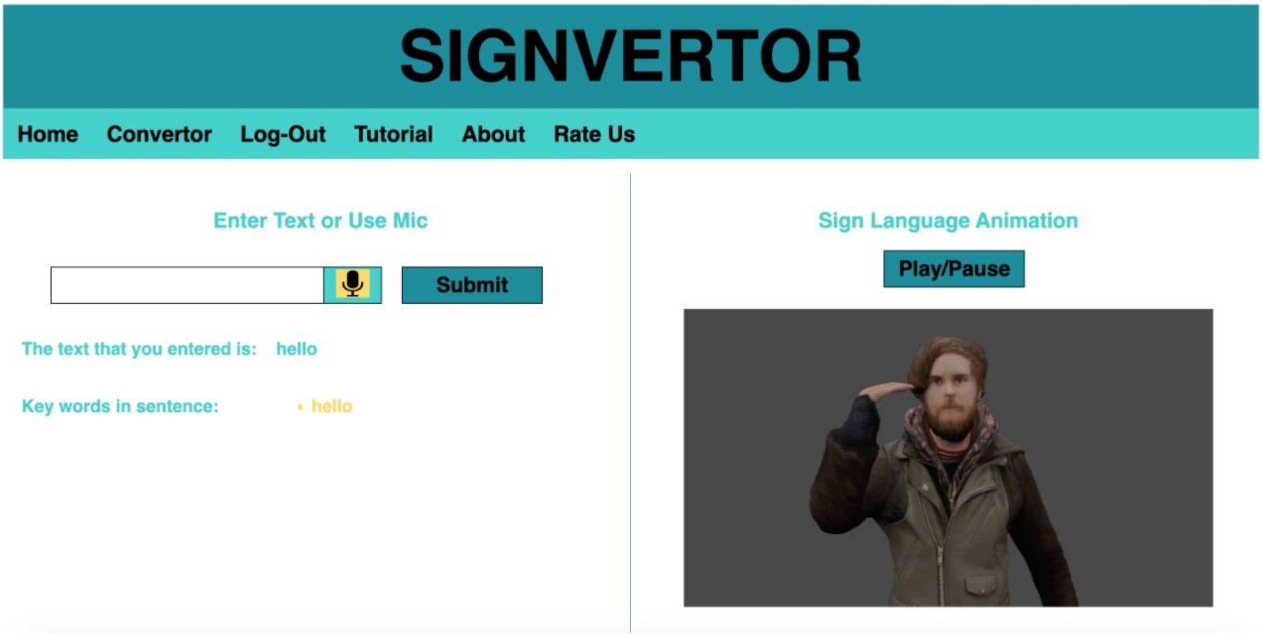


Figure 2: UI Design

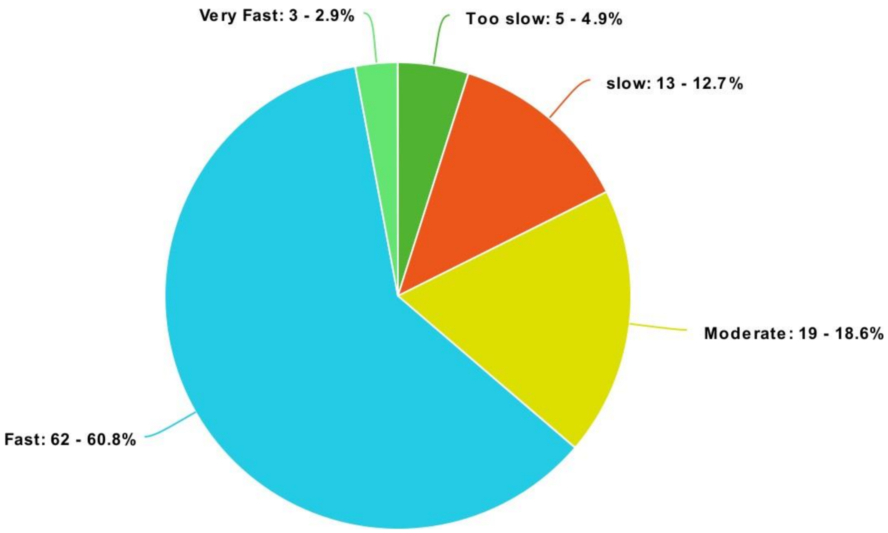


Figure 3: Results for system and web application speed

A screenshot of the animation interface used by Signvertor to display the Indian Sign Language corresponding hand signs is shown in Figure 2. The 3D avatar on the right side of the interface translates the text "Hello" which is displayed on the left side of the interface as speech.

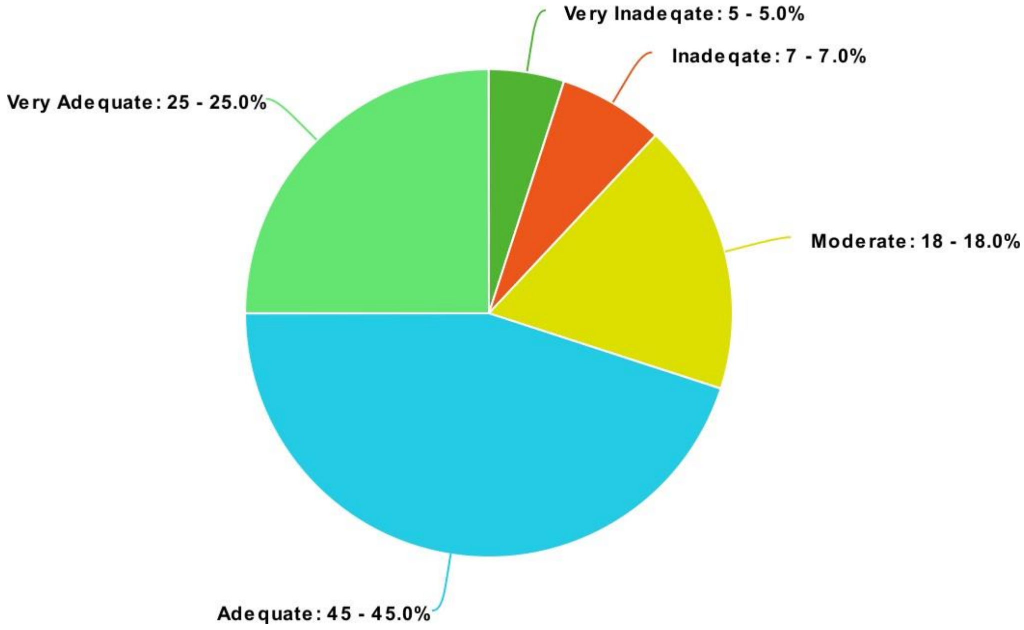


Figure 4: Results for Adequacy of the Animations corresponding to signs

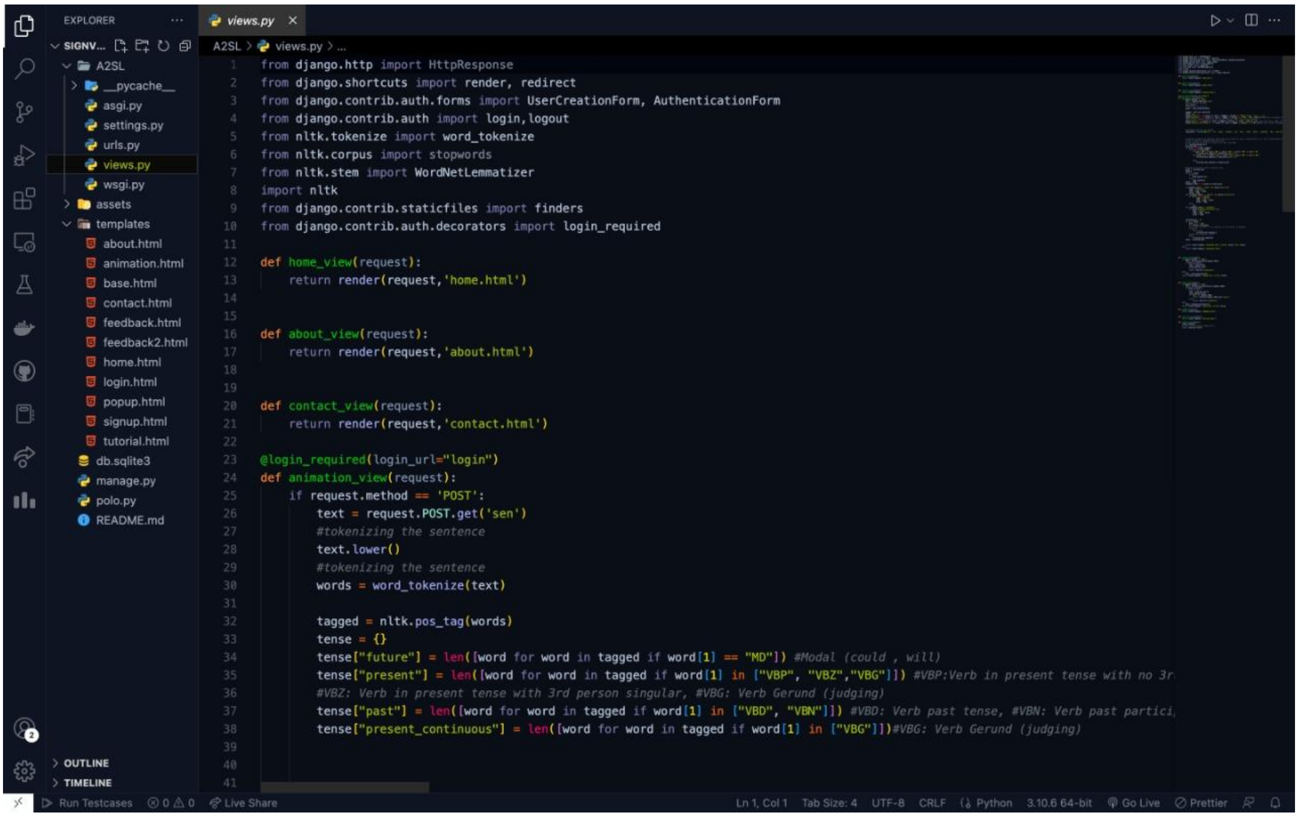
We conducted empirical research analysis by gathering quantitative data from a sample size of respondents to a survey to assess the usability of our product. The survey asked a series of questions from the following sections:

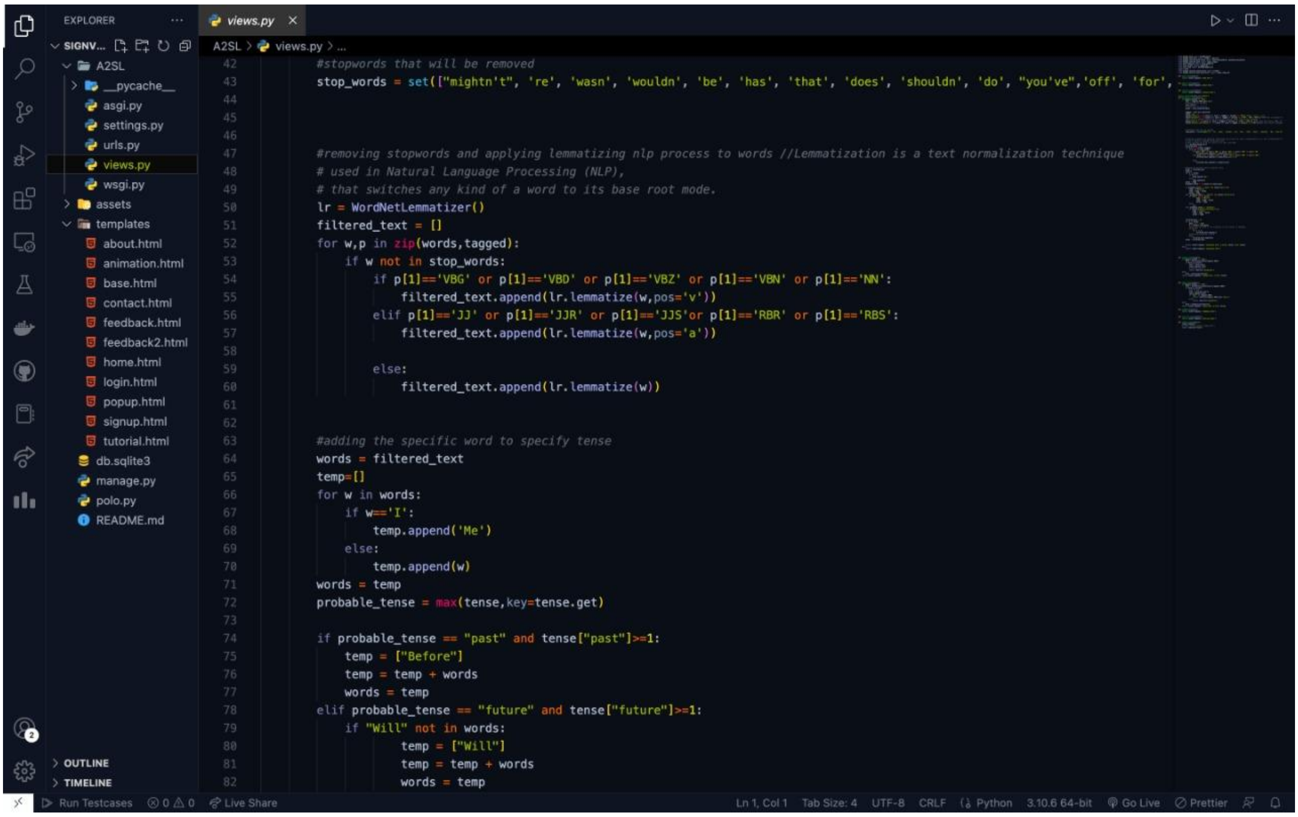
- Screen
- Terminology
- System Details
- Learning
- System features
- User interface (UI)
- Overarching Feelings

Each question was rated on a 5-point Likert scale. We deduced from the data that it didn't take a user long to become accustomed to the Signvertor. For 53% of users, it only took them roughly 10 minutes to complete. Figure 3 shows that 61% of users considered the system and web application to be fast. When we evaluated the suitability of the animations displayed in accordance with each sign, we discovered that, as shown in figure 4, about 25% thought they were extremely adequate and 45% thought they were adequate. These were the answers to some of the survey questions. We deduced from the survey's overall results that our product did well in all areas of HCI.

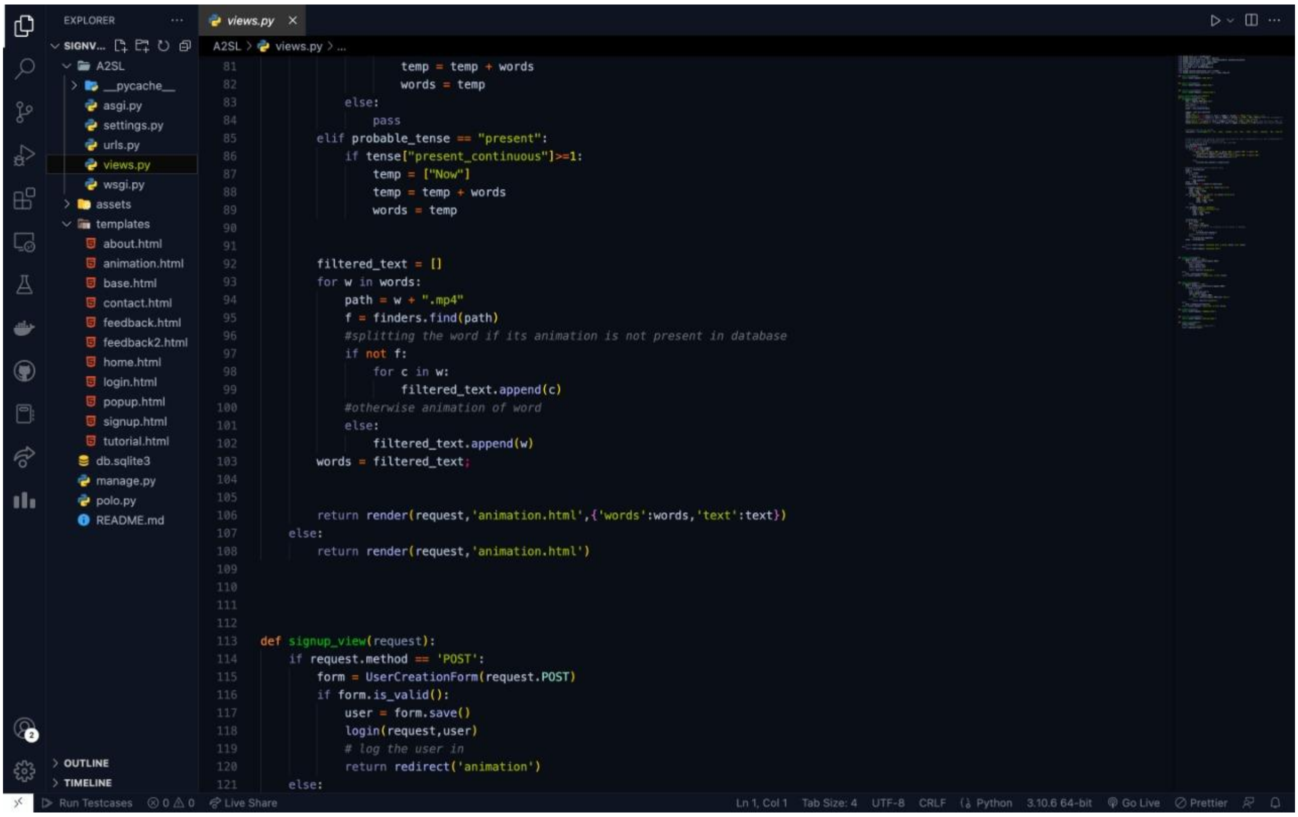
We also conducted a cost and benefit analysis of our project over an eight-year time frame, taking into account that it was introduced to the market as a scalable, all-inclusive solution. We looked at the Cost Profiles for the categories of Contracts, Labour, Infrastructure, Hardware, Software, Telecommunication Equipment, Training Costs, and Other Post Implementation Costs. The estimated total cost for 8 years was ₹12,80,000. Similar to this, we took into account liquid revenues, cost reductions, reimbursements from governmental agencies, outsourced funds, and venture capital funds while determining the benefits profile. The final figure for the benefit was ₹2,18,00,000.

7. CODE SCREENSHOTS

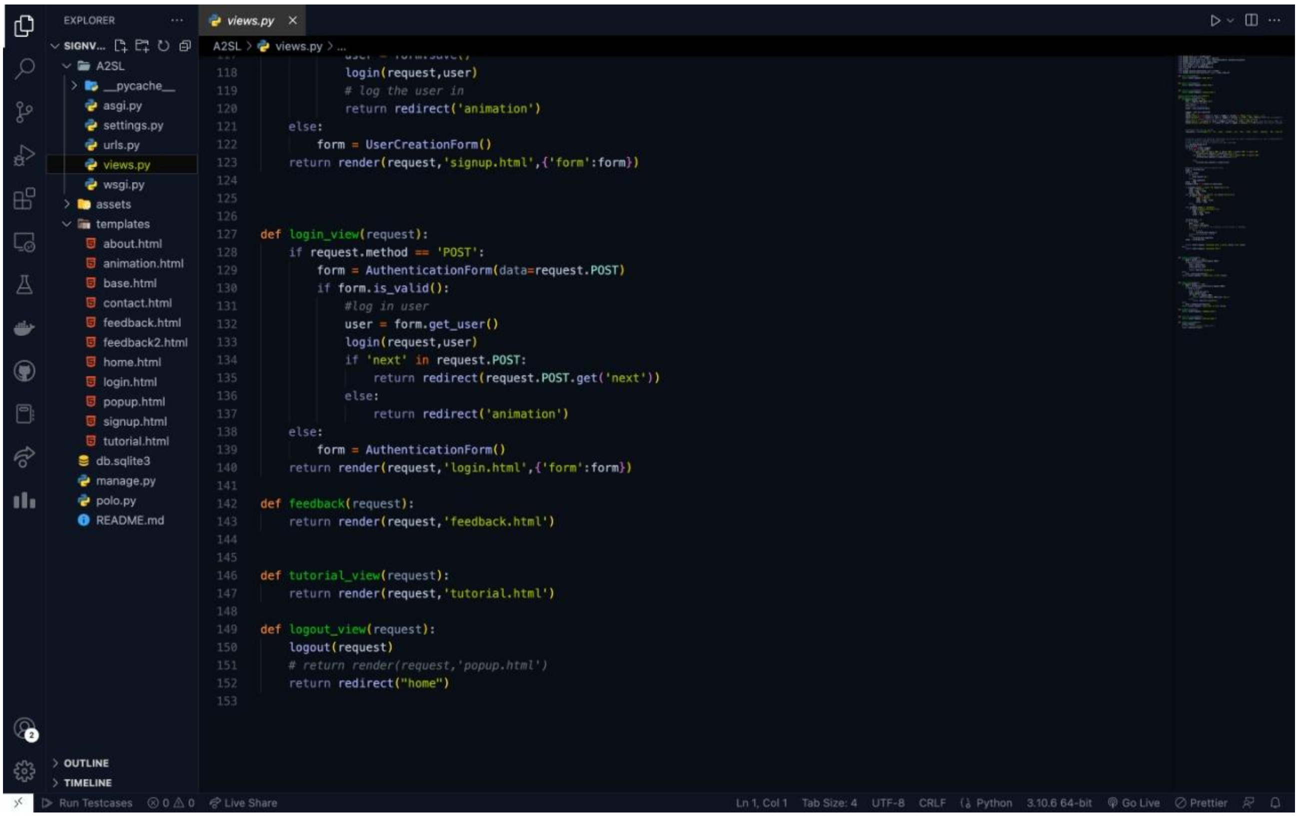




```python
        #stopwords that will be removed
        stop_words = set(["mightn't", 're', 'wasn', 'wouldn', 'be', 'has', 'that', 'does', 'shouldn', 'do', "you've",'off', 'for',



        #removing stopwords and applying lemmatizing nlp process to words //Lemmatization is a text normalization technique
        # used in Natural Language Processing (NLP),
        # that switches any kind of a word to its base root mode.
        lr = WordNetLemmatizer()
        filtered_text = []
        for w,p in zip(words,tagged):
            if w not in stop_words:
                if p[1]=='VBG' or p[1]=='VBD' or p[1]=='VBZ' or p[1]=='VBN' or p[1]=='NN':
                    filtered_text.append(lr.lemmatize(w,pos='v'))
                elif p[1]=='JJ' or p[1]=='JJR' or p[1]=='JJS'or p[1]=='RBR' or p[1]=='RBS':
                    filtered_text.append(lr.lemmatize(w,pos='a'))

                else:
                    filtered_text.append(lr.lemmatize(w))


        #adding the specific word to specify tense
        words = filtered_text
        temp=[]
        for w in words:
            if w=='I':
                temp.append('Me')
            else:
                temp.append(w)
        words = temp
        probable_tense = max(tense,key=tense.get)

        if probable_tense == "past" and tense["past"]>=1:
            temp = ["Before"]
            temp = temp + words
            words = temp
        elif probable_tense == "future" and tense["future"]>=1:
            if "Will" not in words:
                    temp = ["Will"]
                    temp = temp + words
                    words = temp
```



```python
                    temp = temp + words
                    words = temp
            else:
                pass
        elif probable_tense == "present":
            if tense["present_continuous"]>=1:
                temp = ["Now"]
                temp = temp + words
                words = temp


        filtered_text = []
        for w in words:
            path = w + ".mp4"
            f = finders.find(path)
            #splitting the word if its animation is not present in database
            if not f:
                for c in w:
                    filtered_text.append(c)
            #otherwise animation of word
            else:
                filtered_text.append(w)
        words = filtered_text;


        return render(request,'animation.html',{'words':words,'text':text})
    else:
        return render(request,'animation.html')




def signup_view(request):
    if request.method == 'POST':
        form = UserCreationForm(request.POST)
        if form.is_valid():
            user = form.save()
            login(request,user)
            # log the user in
            return redirect('animation')
    else:
```



8. CONTRIBUTION

Soumyadip: Worked at the backend part which is the NLTK and normalizing the speech text.

Sharadindu: Worked on the front end part like connecting all the video animations with the respective speech text through the web speech recognition API.

If any word is not present in our database then it is split into alphabets and then their animation will be combined (like the divide and conquer algorithm) We have also taken care of the past present and future tense of the speech text.

9. CONCLUSIONS AND FUTURE WORK

We created Signvertor as part of this project, a web application that can convert any spoken language into Indian sign language as a 3D animation. To communicate with the deaf in public settings such as schools, hospitals, and meetings, this method might be useful in place of an interpretation. Future studies can concentrate on more accurately improving the real-time and continuous speech input. Multiple 3D avatars can be used to personalise and enhance the user experience on the app. The issue of communicating with the deaf can also be resolved for several regional languages in India, even for small regional language communities.

REFERENCES

[1] Sunitha K. A, Anitha Saraswathi.P, Aarthi.M, Jayapriya. K, Lingam Sunny, “Deaf Mute Communication Interpreter- A Review”, Interna- tional Journal of Applied Engineering Research ,Volume 11, pp 290-296, 2016.

[2] Mathavan Suresh Anand, Nagarajan Mohan Kumar, Angappan Kumare- san, “ An Efficient Framework for Indian SignLanguage Recognition Using Wavelet Transform” Circuits and Systems, Volume 7, pp 1874- 1883, 2016.

[3] Mandeep Kaur Ahuja, Amardeep Singh, “Hand Gesture Recognition Using PCA”, International Journal of Computer Science Engineering and Technology (IJCSET), Volume 5, Issue 7, pp. 267-27, July 2015.

[4] Sagar P.More, Prof. Abdul Sattar, “Hand gesture recognition system for dumb people”. International Journal of Science and Research (IJSR).

[6] Chandandeep Kaur, Nivit Gill, “An Automated System for Indian Sign Language Recognition”, International Journal of Advanced Research in Computer Science and Software Engineering.

[7] Pratibha Pandey, Vinay Jain, “Hand Gesture Recognition for Sign Language Recognition: A Review”, International Journal of Science, Engineering and Technology Research (IJSETR), Volume 4, Issue 3, March 2015.

[8] Nakul Nagpal,Dr. Arun Mitra.,Dr. Pankaj Agrawal, “Design Issue and Proposed Implementation of Communication Aid for Deaf & Dumb People”, International Journal on Recent and Innovation Trends in Computing and Communication ,Volume: 3 Issue: 5,pp- 147 – 149.

[9] Neelam K. Gilorkar, Manisha M. Ingle, “Real Time Detection And Recognition Of Indian And American Sign Language Using Sift”, International Journal of Electronics and Communication Engineering Technology (IJECET), Volume 5, Issue 5, pp. 11-18 , May 2014.

[10] Ashish Sethi, Hemanth ,Kuldeep Kumar,Bhaskara Rao ,Krishnan R, “Sign Pro-An Application Suite for Deaf and Dumb”, IJCSET , Volume 2, Issue 5, pp-1203-1206, May 2012.