

**A**

**PROJECT REPORT**

**ON**

“WEB APPLICATION TO CONVERT YOUTUBE CAPTIONS INTO SIGN LANGUAGE”

For the subject **Lab II - Project Phase II**

Submitted in partial fulfillment of the requirement for the award of

**Bachelor of Engineering**

**In**

**Computer Science and Engineering**

**Solapur University, Solapur**

By

|  |  |  |
| --- | --- | --- |
| Name | Roll. No. | Exam Seat No. |
| Drushti Luhadiya | 13 | 807033 |
| Purvashi Dhakad | 14 | 807077 |
| Shaurya Jain | 15 | 807147 |
| Manjula Hiremath | 16 | 807144 |

Under Guidance Of

**Mr. P.S.R. PATNAIK**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**WALCHAND INSTITUTE OF TECHNOLOGY**

**SOLAPUR - 413006**

**(2018-2019)**

****

**CERTIFICATE**

This is to certify that the Project entitled

**“​**WEB APPLICATION TO CONVERT YOUTUBE CAPTIONS INTO SIGN LANGUAGE**”**

**Is**

**Submitted by**

|  |  |  |
| --- | --- | --- |
| Name | Roll. No. | Exam Seat No. |
| Drushti Luhadiya | 13 | 807033 |
| Purvashi Dhakad | 14 | 807077 |
| Shaurya Jain | 15 | 807147 |
| Manjula Hiremath | 16 | 807144 |

As a part of Project Design Report.

Studying in BE CSE for the subject **Lab II - Project Phase II**

**(Mr. P.S.R. PATNAIK) (Dr.R.V.Argiddi) (Dr.S.A.Halkude)**

***Project Guide* HeadPrincipal**

**Dept of Computer Science & Engg**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**WALCHAND INSTITUTE OF TECHNOLOGY**

**SOLAPUR**

**(2018-2019)**

**Plagiarism Undertaking**

We solemnly declare that project work presented in the report titled “………………………………………………………………………………………………............

WEB APPLICATION TO CONVERT YOUTUBE CAPTIONS INTO SIGN LANGUAGE

…….………………………………………………………………………………………………...”

is solely our project work with no significant contribution from any other person. Small contribution/help wherever taken has been duly acknowledged and that complete report has been written by the members of the project group.

We understand the zero-tolerance policy of the WIT, Solapur and University towards plagiarism. Therefore, we as Authors of the above titled report declare that no portion of the report has been plagiarized and any material used as reference is properly referred / cited.

We undertake that if found guilty of any formal plagiarism in the above titled report even after award of the degree, WIT, Solapur and Solapur University reserves the rights to withdraw/revoke the degree granted and that WIT, Solapur and the University has the right to publish our name on the website on which names of students are placed who submitted plagiarized report.

|  |  |  |  |
| --- | --- | --- | --- |
| Name of Student | Exam Number | University PRN Number | Signature |
| Drushti Luhadiya | 807033 | 2015032500254186 |  |
| Purvashi Dhakad | 807077 | 2015032500254066 |  |
| Shaurya Jain | 807147 | 2015032500254484 |  |
| Manjula Hiremath | 807144 | 2016032500276086 |  |

Date: / / 2019 .

**Acknowledgment**

At the outset, we would like to take this opportunity to express our deep gratitude to

our guide **Mr.P.S.R. Patnaik** of CSE Department for his guidance and moral support throughout our project.

We heartily thank **Prof. Dr.R.V.Argiddi**, Head of CSE Dept for his moral support and

promoting us our project.

We would also like to thank our Principal **Dr. S. A. Halkude** and all staff members

For their whole hearted cooperation in working on this project.

**INDEX**

|  |  |  |
| --- | --- | --- |
| S.no | Topic | Page No. |
| 1 | Abstract | 06 |
| 2 | Introduction | 07 |
| 3 | Existing Systems | 08 |
| 4 | Methodology | 10 |
| 5 | Dependencies and Requirements | 14 |
| 6 | Instructions for deployment | 17 |
| 7 | Summary | 20 |
| 8 | Future scope | 21 |
| 9 | Reference | 22 |

**Abstract**

The core of this project is basically the conversion of text language to sign language. Sign language is used by deaf and dumb people for the purpose of the communication. Sign language is nothing but the means of communication through hand gestures or fingers known as finger spelling. There are many existing systems like Fun translation , Pro deaf translator , Hand talk Translator etc. which provide fingerspelling for every word or just translate text language to sign language or play animation for the given word and thus it is useful for ones who understand it but this is an extended system which converts the YouTube captions into sign language and shows fingerspelling and show the animation for the YouTube videos.

YouTube is a platform where many entertainment and informative videos are present but unfortunately deaf and dumb people are unable to utilize them because of lack of resources like finger spelling captions or play an animation. For such people the system converts the YouTube captions into sign language. It provides a platform for the deaf and dumb people and act as a medium for them to understand the videos with the help of fingerspelling and an animated character. The fingerspelling is mainly used to describe the name of a person or place, though it plays along with the animated character.

Methodology used for adopting this project is that the system will fetch the YouTube URL. The caption tracks are taken from the YouTube API endpoint with the help of YouTube APIs [1]. The system will then make an API call and get the captions from YouTube. The extraction of captions are done using the two API calls ‘caption.list [2] and caption.download [3]’.The system firstly check that whether the captions of a particular video are present or not and if the caption track of the video is not present then an error message will be displayed on the screen. Further it checks that if the captions are in English language and if the captions are not in English then an error message displays on the screen. If captions are present in English then system will start its working and asks the user that whether he wants to use fingerspelling captions or animated character.

In Fingerspelling only, the images of fingers are displayed on screen and if user uses animated character then an animated character is displayed on the screen and it works along with the video. It provides animation for each and every word used in that video, if there is any such word whose animation is not possible to depict then the finger spelling of that word is displayed and animated character keeps on working after it finds the next suitable word for animation.

Thus, this project is helpful for all those who understand the sign language and work as a medium for them to understand the YouTube videos for their knowledge or entertainment which was earlier not possible with the other systems which only works for particular word or sentence i.e. text to sign conversion.

# Introduction

This web application is made with the purpose of converting the YouTube captions of a desired video into the sign language. This system is made for the deaf and dumb people who are unable to understand English. This application helps those people by converting the English caption to sign language. The sign language is displayed to user in the form of fingerspelling [4] or an animated character and displayed on the screen enacting the video with the help of caption tracks.

The idea here is to get the caption track from YouTube API Endpoint for the given video URL using the YouTube APIs [1]. The API calls used are ‘caption.list’ [2] and ‘caption.download’ [3] .If the caption track is present in English language, then the web application gives output to user screen in the form of sign language. The output can be obtained in two ways first is fingerspelling and second is in the form of an animated character.

The data for fingerspelling is images [4] of alphabets ‘a-z’ and numeric values ‘0-9’, while the animated character uses the ‘SiGML player [5]’ and ‘Sigml files’ [6] to display the output on the screen.

The technologies used for front end are bootstrap and CSS for styling and designing of the website and jQuery and JavaScript for front end operations.

For backend we have used php for database operations and implementation. Data is stored in MySQL database.

**Existing Systems**

Earlier people used to shoot and record the videos of an actor performing the fingerspelling actions which can sometimes be inefficient and very expensive. So, to overcome this ‘Hamburg notation system’ and ‘SiGML file player [6] is used’. There are many websites [7][8][9] and applications which can convert the simple English text to any selected sign language. Below is the list of few applications that converts text to sign language.

3.1 **Apps**:

3.1.1) ***Hand talk translator***

You can count on the largest sign language automatic translation platform to take accessibility to everyone with the Hand Talk app you can automatically translate text and audio to Brazilian Sign Language (Libras), for free [10].

3.1.2) ***ASL Translator***

There are two parts to this app.

1. TEXT-TO-SIGN GENERATOR (SIGNED ENGLISH)

2. ASL PHRASES (ASL)

PART ONE

Over 30,000 words translated into Seamless Sign Language Video in real time! Patented technology seamlessly connects each signed word to display a smooth video stream, the only software with this unique feature.

PART TWO

110 ASL Phrases

Learn how to sign ASL idioms and phrases. Jessica Tanner (deaf actress) and Dwight D. Godwin, NIC (certified ASL interpreter), translate 100 idioms and phrases such as “where is the bathroom” and “train go sorry” into ASL. (No internet required) [11]

# 3.1.3) *Mimix3D Sign Language*

# Mimix3D Sign Language Translator is a mobile app. that interprets spoken and written English into sign language using a 3D friendly avatar, new signs are added every week so bear with us.

# The 3D avatar will interpret text/speech input in English and deliver real-time sign language translations, enabling easier communication with the deaf & hard of hearing community without having to know sign language.[12]

# 

# 

# 

# 

# 

# Methodology

4.1 **YouTube URL:**

It all starts when the user gives the input in the form of URL and is sent to backend Where the video id is extracted and the subtitle file is searched if it is already present or not. If the file is found in the database then the parsing of caption file is done. If file is not found then an API call is made to the YouTube API endpoint. If any error or exception is occurred then the error is displayed on the output. If everything goes fine the caption track is downloaded and is stored in the database for future use

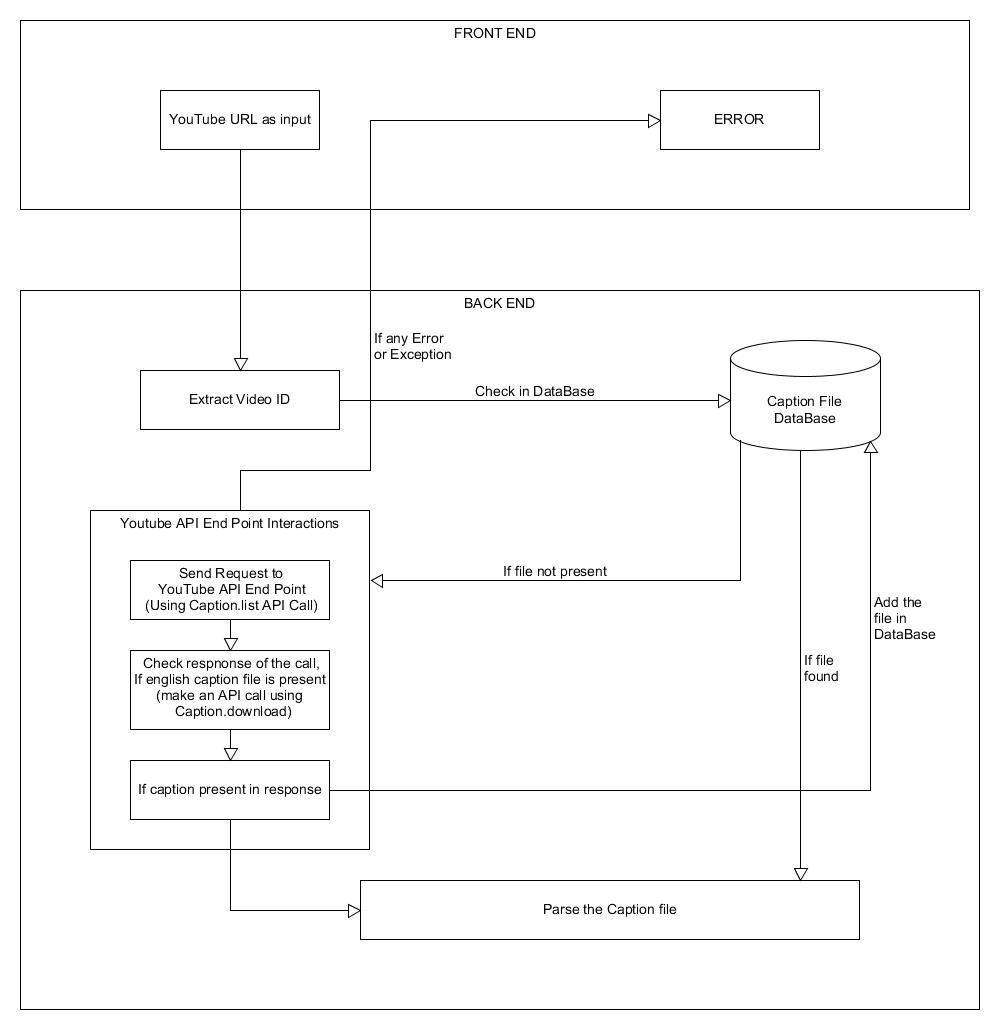


Fig.4.1 System Architecture

4.2 **Parsing of caption File:**

The caption file is parsed such that the all the important information and details in the arrays which include:

1.Array of strings with the all the caption text in it.

2.Array containing the difference time period between the two caption string.

3.Array containing the time period for which the caption will be displayed.

4.3 **Displaying the output:**

The user has two options:

1.Play the video with finger spellings.

2.Play the video with the animation character.

4.3.1) ***Finger Spelling:***

The array of caption track string from parsed caption file is passed to the function and the new array is generated with the path of finger spelling and is displayed to the screen with the time stamp.

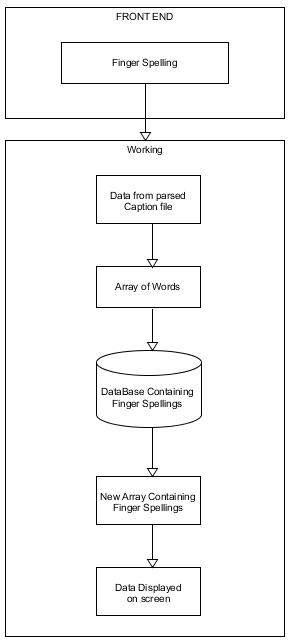


Fig.4.2 Finger Spellings Workflow

4.3.2) ***Animated Character*:**

Here the array of the string with caption track is combined as a single string and then each word is passed to the lemmatizer and the output is sent to the JASigning controller and the word is searched in the sigml data files, if the word is found then the animation is played or else the finger spelling is displayed.

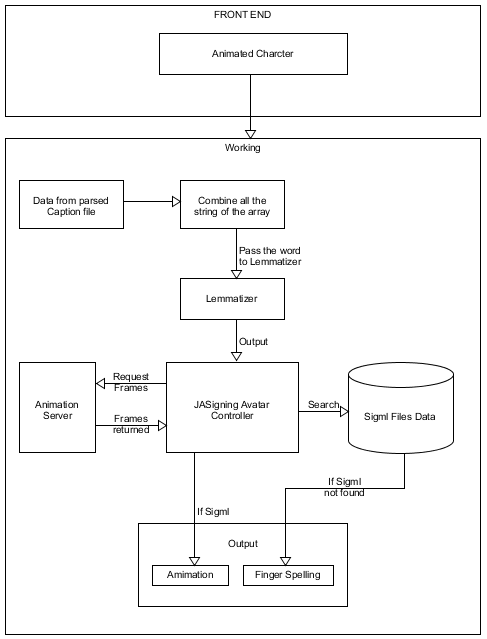


Fig.4.3 Animated Character Workflow

# Dependencies and Requirements

5.1 **Dependencies & Technologies used:**

5.1.1) ***Front end:***

5.1.1.1. HTML:

HTML stands for Hyper Text Markup Language. It describes the structure of Web pages

using markup.HTML elements are the building blocks of HTML pages.[13]

Version: 5

[Policies and Legal Information and](https://www.w3.org/Consortium/Legal/ipr-notice.html) License can be found of the website.[14]

5.1.1.2. JavaScript & jQuery:

JavaScript is the programming language of HTML and the Web. We used it to make

HTML elements functional.[15]

jQuery is a cross platform JavaScript library designed to simplify the client-side scripting

of HTML. We used it for validation purposes.

jQuery version: v3.4.0

License: MIT License

5.1.13. Bootstrap & CSS:

Bootstrap is a free and open source front end library for designing websites and web

applications. It contains HTML and CSS base design templates for typography, fonts

, buttons navigation and other interface components, as well as optional JavaScript

extensions [16].

Bootstrap version: v3.4.1

License:MIT License

5.1.2) ***Back End:***

5.1.2.1 PHP 7:

Hypertext preprocessor is a server-side scripting language design for web development but also used as a general-purpose programming language [17]. All back end is written in it and database operations are done using queries in it.

Version: 7.3.0

License: PHP License v3.01

5.1.3) ***Database:***

5.1.3.1 MySQL**:**

MySQL is used to develop a database [18]. All data is stored in it, using MySQL queries. These queries are written in php to select, insert, update and delete the data in database.

License: GNU Public License

5.1.4) ***YouTube APIs v3:***

5.1.4.1 Caption. List:

Returns a list of caption tracks that are associated with a specified video [2].

5.1.4.2 Caption. Download:

Returns the caption track in the desired format by using caption\_id [3].

Terms of Service License can be found of the website.[19]

5.1.5) ***JASIGNING:***

The [JASigning](http://vh.cmp.uea.ac.uk/index.php/JASigning) software is based on WEBGL framework and is used to play the sigml files and display the animation on the screen.

# Version: vhg.2019

Terms of Service and conditions can be found on the website.[20]

5.1.6) ***PHP-Lemmatizer:***

PHP Lemmatizer is a lemmatization library for PHP to retrieve a base form from an inflected form word in English [21].

License: MIT License

5.2 **Requirements:**

**Browsers**:

\*On OS X project will work fine with Firefox, Chrome, Opera, and Safari.

\*Windows project will work fine with Firefox, Chrome, Opera, Edge, and Internet Explorer.

\*Linux project will work fine with Firefox, Chrome, Opera, Chromium, and Vivaldi.

\*iOS project will work fine with Safari.

Note: For best experience update the browsers to the latest

High speed internet connection is recommended.

**Instructions for Deployment**

**Step 1**: Configuring and setting up the project

1.Put all the files on the server/localhost and import the database file in the MySQL database.

2.Either download the JASigning library package and keep it on the server/localhost or else fetched the required file from using their link of JavaScript file and CSS file.

**Step 2**: Executing the project

1. User will copy the URL from YouTube and provide it as input to the site also both shortened URL and normal URL are supported.

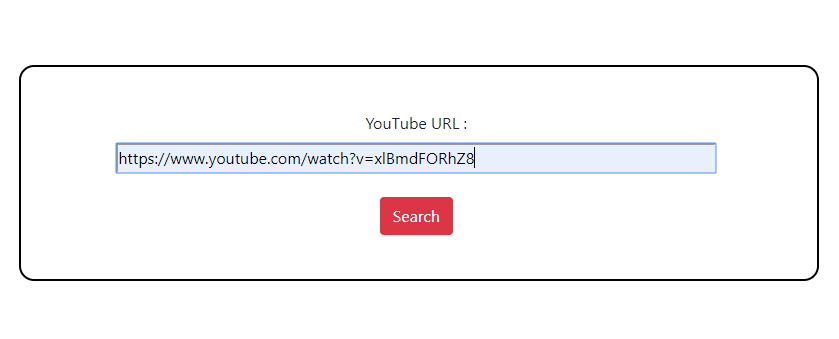


Fig.6.1 Input as YouTube URL

2. If any error or exception occur error will be displayed on the screen.

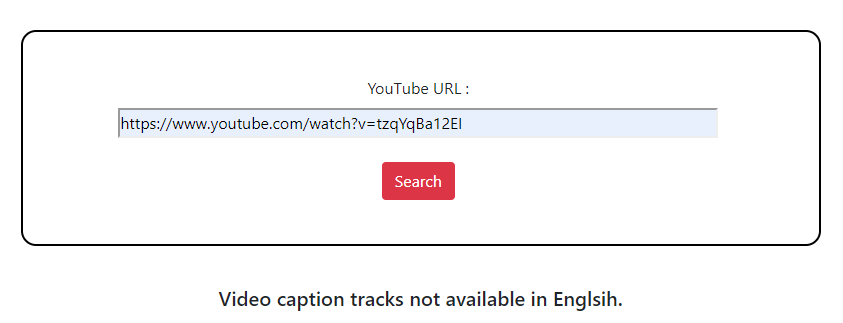


Fig.6.2 Error Message when English Caption not available

3. After this the user will have 2 option to play video either in fingerspelling mode or with the help of animated character.

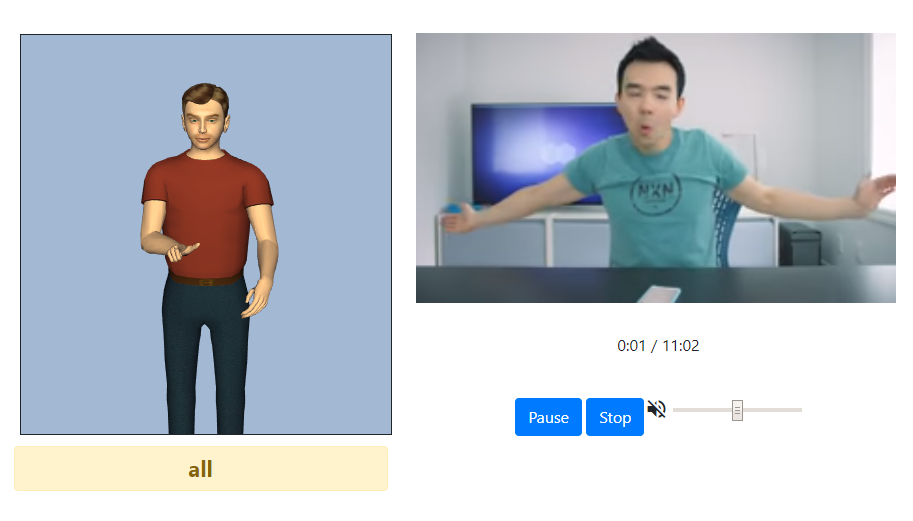


Fig .6.3 Output on Screen With animated character

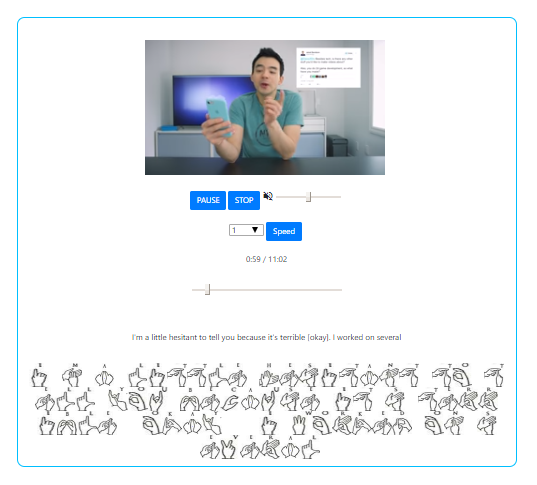


Fig.6.4 Output on Screen with Fingerspelling

# Summary

This is a web application for the conversion of text language into sign language. Here the web application converts YouTube captions into sign language. The system combines all the necessities of a deaf person in one platform. The application is very easy to handle by the deaf people and they can easily access the YouTube videos by just copying the URL of YouTube videos in the search bar. This application assists them in understanding the YouTube videos in a simplified manner.

Here the user will be able to play the YouTube videos along with finger spellings or with animation character depending on his/her choice. The finger spellings are perfectly sync with the video and only sometimes mismatches if the YouTube video buffers. The only thing which lacks in the system is the syncing of animated character with the YouTube video ,some level of syncing can be achieved with animation character by slowing the playback speed of the video but its inconvenient and still there are issues with the syncing of animated character with YouTube video .It is due the animation library because the speed of animation can’t be controlled and a particular word will take its time to play its animation and then only next word can animate so due to this animated character is unable to sync with the video.

Apart from this, as developers we have learnt to work as a team. With the help of our guide, tutorials and online help we have implemented this idea.

# Future Scope

In future system can be modified by following ways.

1. Adding a login system so that users can create their account. After creating an account user will also be able to create playlists.

2. Also a separate section can be added where the most watched videos will be displayed.

3.Also the accuracy of the syncing of animation and video can future be improved.

# References

[1] <https://en.wikipedia.org/wiki/YouTube_API>

[2] <https://developers.google.com/youtube/v3/docs/captions/list>

[3] <https://developers.google.com/youtube/v3/docs/captions/download>

[4] <https://www.signcommunity.org.uk/finger-spelling.html>

[5] <https://en.wikipedia.org/wiki/Hamburg_Notation_System>

[6] <http://vh.cmp.uea.ac.uk/index.php/Configuring_JASigning_for_HTML5_web_pages>

[7] <https://funtranslations.com/sign-language>

[8] https://lingojam.com/SignLanguageTranslator

[9] <http://www.islfromtext.in/>

[10] <https://play.google.com/store/apps/details?id=br.com.handtalk&hl=en_IN>

[11] <https://play.google.com/store/apps/details?id=com.asltranslator>

[12] <https://play.google.com/store/apps/details?id=com.mindrocketsinc.mimix&hl=en_US>

[13] https://en.wikipedia.org/wiki/HTML

[14] <https://www.w3.org/Consortium/Legal/2015/copyright-software-and-document>

[15] https://opensourceforu.com/2016/07/basics-javascript-jquery/

[16] https://en.wikipedia.org/wiki/Bootstrap\_(front-end\_framework)

[17] <http://php.net/manual/en/intro-whatis.php>

[18] https://www.mysql.com/

[19] <https://developers.google.com/youtube/terms/api-services-terms-of-service>

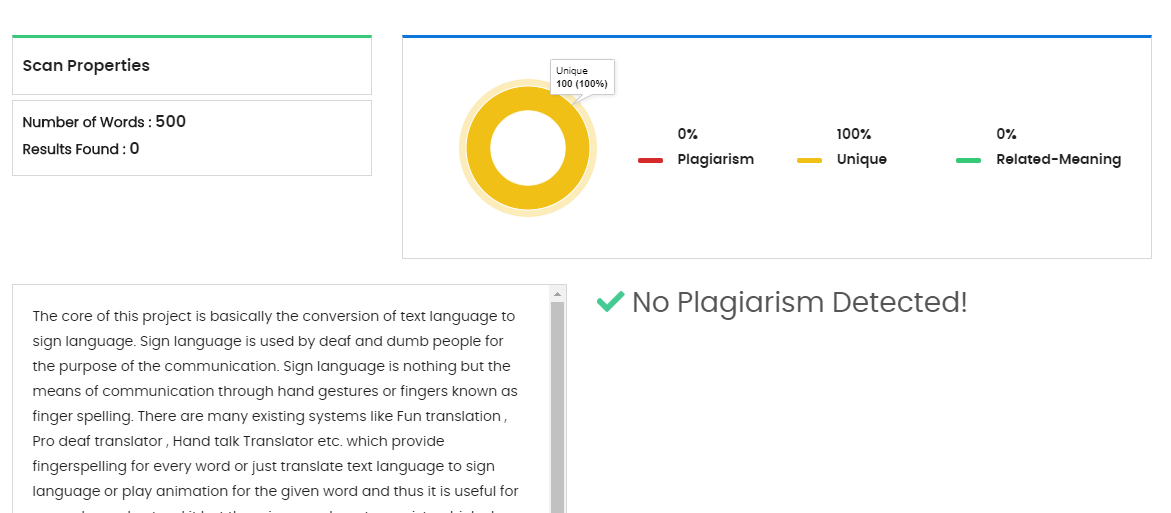
[20] <http://vh.cmp.uea.ac.uk/index.php/JASigning_Conditions>

[21] <https://github.com/skyeng/php-lemmatizer>

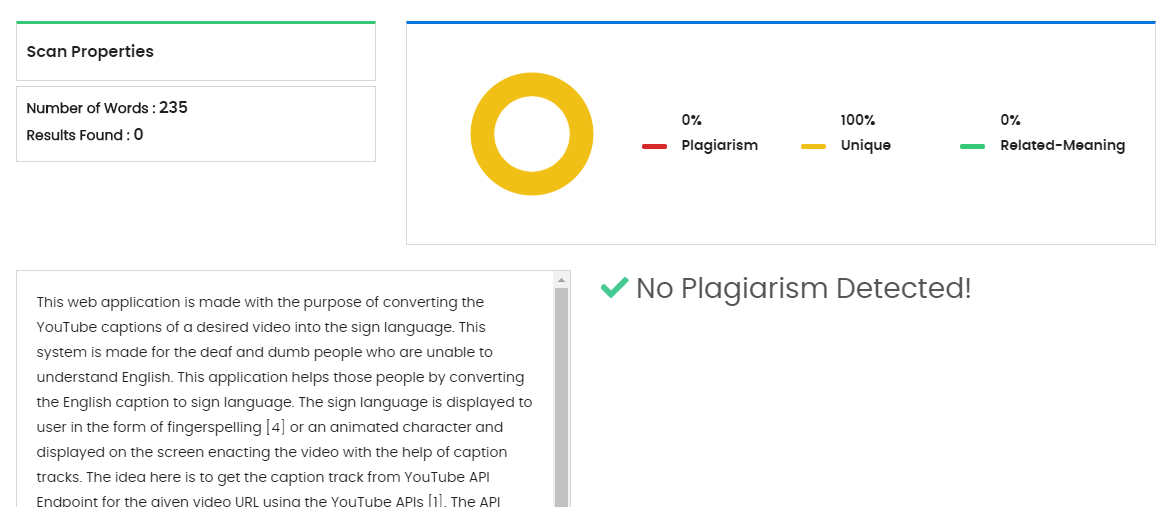
**Plagiarism Check**

**Source:** [**https://www.duplichecker.com/**](https://www.duplichecker.com/)

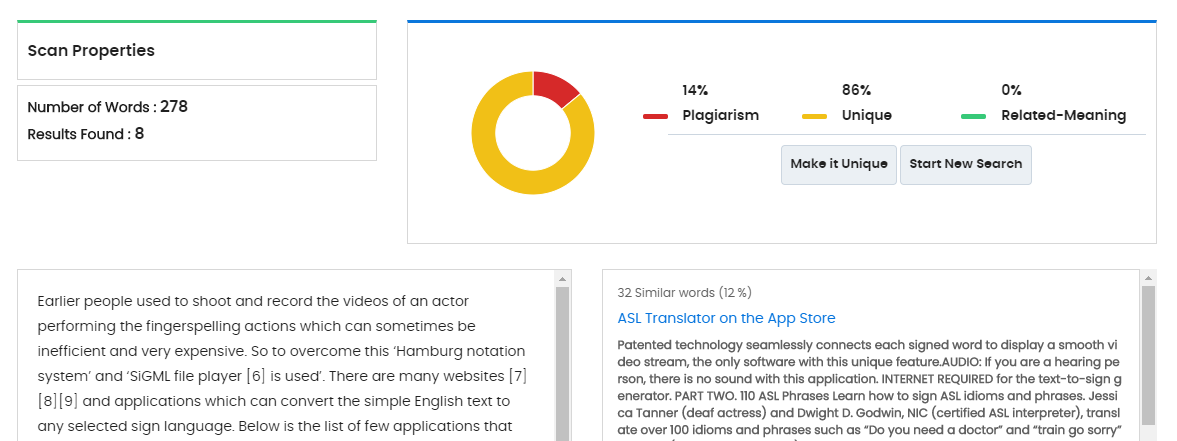
**Abstract:**

****

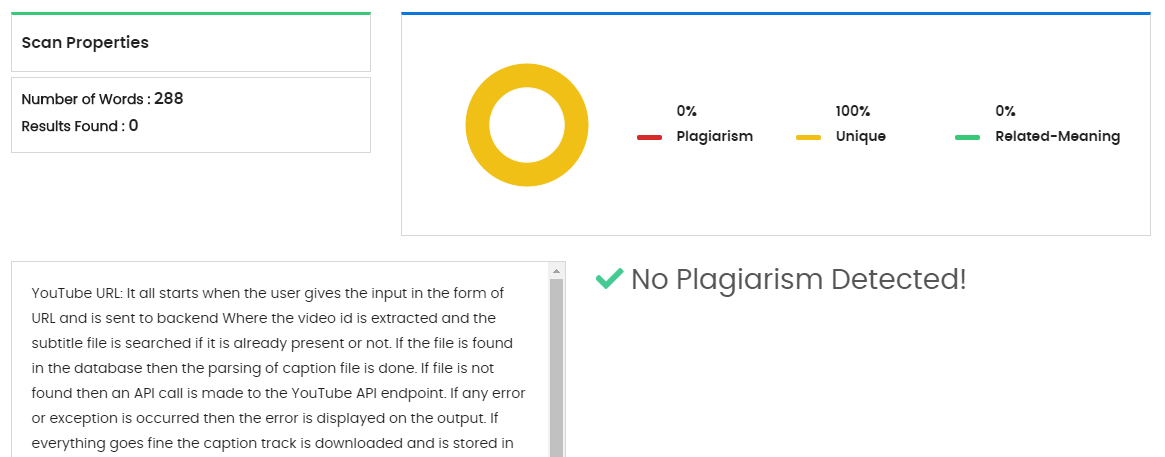
**Introduction:**

****

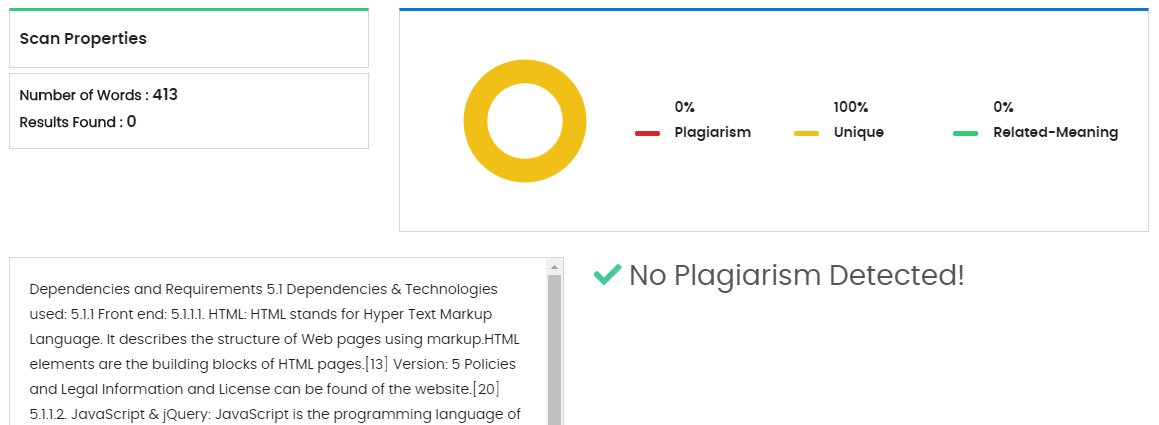
**Existing Systems:**

****

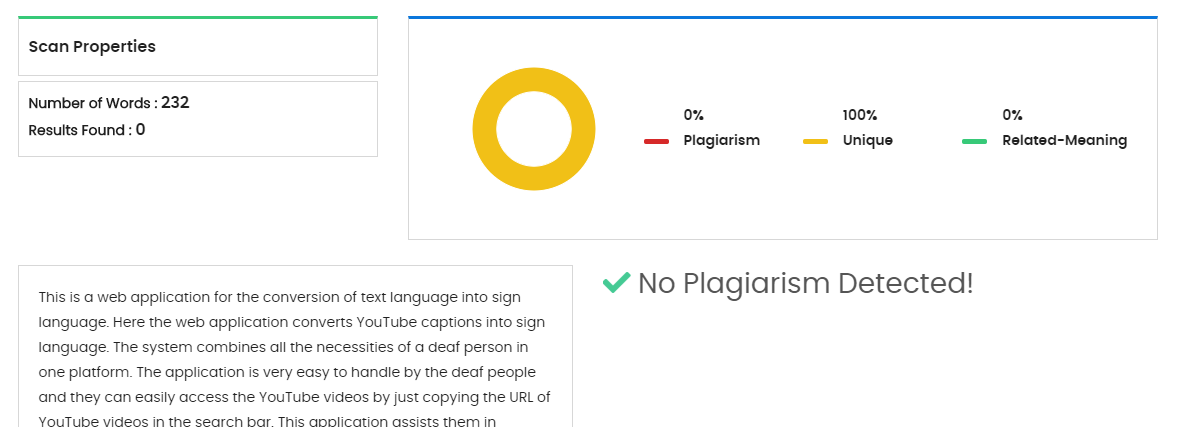
**Methodology:**

****

# Dependencies and Requirements:



**Summary:**

****

**PROJECT GROUP MEMBERS**

|  |  |  |
| --- | --- | --- |
| Name of Student | University PRN Number | E-MAIL |
| Drushti Luhadiya | 2015032500254186 | drushtiluhadiya@gmail.com |
| Purvashi Dhakad | 2015032500254066 | purvashidhakad.pd@gmail.com |
| Shaurya Jain | 2015032500254484 | smartyj44@gmail.com |
| Manjula Hiremath | 2016032500276086 | hmanjulah1997@gmail.com |



Pictured left to right: Manjula Hiremath, Purvashi Dhakad, Drushti Luhadiya and Shaurya Jain

**Project GitHub Link:** <https://github.com/smartyj44/WEB-APPLICATION-TO-CONVERT-> YOUTUBE-CAPTIONS-INTO-SIGN-LANGUAGE