

DEVELOPING ANIMATION AND INTERACTIVITY IN A WEB-BASED APPLICATION FOR LEARNING MALAYSIAN SIGN LANGUAGE

¹Shamsul Anuar Mokhtar, ²Siti Sarah Shamsul Anuar, ³Siti Mashitah Shamsul Anuar

Universiti Kuala Lumpur, 1016 Jalan Sultan Ismail, 50250 Kuala Lumpur, Malaysia

For Correspondence; ¹shamsulanuar@unikl.edu.my ²sarahdrsam@gmail.com ³sitimashitah@unikl.edu.my

ABSTRACT: Malaysian Sign Language is the native language for the deaf people in Malaysia. However, there is still a lack of web resources available to the public to facilitate the learning of the language. The purpose of the paper is to discuss the development of sign language animation content on the web and how interactivity is built into MalaysianSignLanguage.com, the web application developed by the authors for learning Malaysian Sign Language. The design and development of the web application covers two aspects: the content and the application modules. The web application incorporates character animation and utilizes open source technologies. The modules developed includes sign language representation of 200 words or more, letters of the alphabet, numbers, fingerspelling with user-defined words, and formation of user-defined sentences. Within two years of publishing the web application to the World Wide Web, the web application has attracted more than 18,000 visitors. In terms of search engine ranking, the web application is currently the top ranked web site for Malaysian Sign Language. Usability evaluation conducted on participants found that more than 90% of the respondents gave positive response to the sign language animation content and the features offered by the web application.

Keywords: Animation, interactivity, sign language, web application, learning application

1. INTRODUCTION

According to the World Federation of the Deaf, the number of deaf people in Malaysia was reported to be 29,522 [1]. This is approximately 7 percent of the disabled population in Malaysia. To protect the rights of people with disabilities, Malaysia has ratified the Convention on the Rights of Persons with Disabilities (CRPD) in 2010. The government has adopted the Persons with Disabilities Act 2008 that focuses in areas such as health, rehabilitation and education [2].

Similar to hearing people, the deaf community has a right to a native language of their own. This right is stated in The Salamanca Statement, UN Standard Rule on Equal Opportunity and Participation for the Disabled Person, Agenda of Action [3]. Many people in the deaf communities around the world use sign language as their primary means of communication. Sign language uses gestures, including hand movements and facial expressions and they are interpreted visually. The deaf community in Malaysia adopts a version of sign language which is the Malaysian Sign Language which is adapted from the American Sign Language. Many deaf Malaysians consider sign language as their primary means of communication [4].

Books, which are commonly used in Malaysia as the main learning tool for sign language may not be effective. They lack interaction with the learners and hence, accurate representation of words in sign language through this medium is difficult. Apart from books, videos on sign language are also available. However, they lack interactivity. Several web sites also promote the learning of Malaysian Sign Language. Many of them either use drawings or videos to represent sign language.

One of the means of providing interactivity to the learning of Malaysian Sign Language is by incorporating animation in the learning application. As the most widespread platform to access learning content is still the World Wide Web, the learning application must be web based and it must be also accessible via computers, tablets and mobile devices.

This paper focuses on the MalaysianSignLanguage.com, a web-based learning application developed by the authors of this paper. It is currently the top-ranked Malaysian Sign

Language web-based learning application in the world based on Google, Bing and Yahoo search engines. Within 2 years of publishing the web application to the World Wide Web, it has attracted more than 16,000 visitors. In this paper, the authors discuss the development of sign language animation content on the web and how interactivity is built into the application.

2. LITERATURE REVIEW

Most sign language videos and software learning packages in the market are based on the American Sign Language. They may not be suitable for use in Malaysia as some language representations of American Sign Language is different from the Malaysian Sign Language. They also lack content and may not be compatible with the Malaysian culture. As for local web applications that are available on the World Wide Web, they are very few in numbers.

There are several web learning resources for Malaysian Sign Language. This section reviews these resources in terms of their sign language content and how they relate to concepts that are specific to the Malaysian culture. These web resources, however, lack clarity of the sign language content and do not allow users to explore the use of sign language by forming their own sign language sentences and fingerspelling of words.

E-Hearme

E-Hearme is a web application that is based on the Malaysian Sign Language. It facilitates the learning of sign language by allowing the user to select or search from a list of 600 to 700 sign language words. Each selected word is represented by a line drawing and description that appears to be scanned from a book as well as a short video clip depicting the sign. The medium of instruction for the website is Bahasa Malaysia. The web application does not include any sign that represents alphabet and numbers in Malaysian Sign Language [5].

E-Isyarat

E-Isyarat is another web application that facilitates the learning of Malaysian Sign Language. It is generally in Bahasa Malaysia and covers approximately 200 to 300 sign language words. The web application uses short video clips to represent the signs. The signs however do not include letters of the alphabet and numbers. Regarding the compatibility of

the web application across different computing platforms, it failed to display the pages when using mobile devices and computer tablets [6].

Virtual Malaysian Sign Language

Virtual Malaysian Sign Language is a website that promotes the learning of Malaysian Sign Language. The website embeds YouTube video clips on how Malaysian Sign Language can be used in sentences. The video clips covers sign language representation of alphabet and numbers. It also includes 12 lessons with the lessons represented by 3 to 5-minute video clips that show simultaneous conversation in sign language, voice narration and text subtitle. The medium of instruction used is Bahasa Malaysia [7].

Facebook Group on Malaysian Sign Language

There is an active group on Facebook that promotes the Malaysian Sign Language. The group goes by the name of Bahasa Isyarat Malaysia (BIM) / Malaysian Sign Language (MySL). In the group, the moderator and members posted links and materials that in some way facilitate the learning of Malaysian Sign Language [8].

3. MALAYSIANSIGNLANGUAGE.COM

MalaysianSignLanguage.com is a web-based learning application developed by the authors of this paper. It is a bilingual web application, in English and Bahasa Melayu. The web-based learning application allows users to learn all the 26 letters of the alphabet (Figure 1). The letters are represented by graphics. Users can also learn selected numbers in sign language and they are represented by graphics and animation (Figure 2). In addition, users can form and learn custom spellings (animation) in sign language (Figure 3). As for everyday words, the users can learn more than 200 signs (Figure 4). Based on these words, users can form and learn custom phases and sentences in sign language.



Fig. 1: Alphabet in sign language

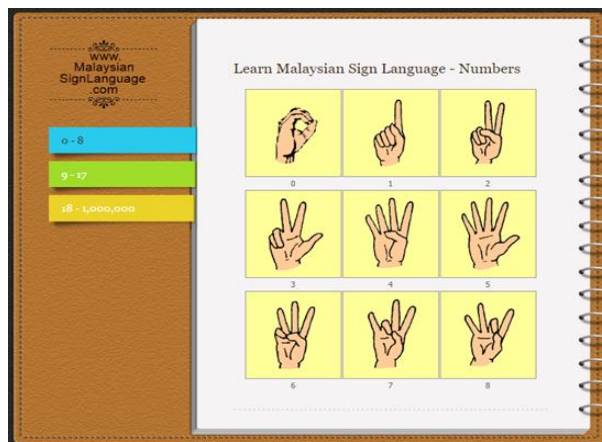


Fig. 2: Numbers in sign language

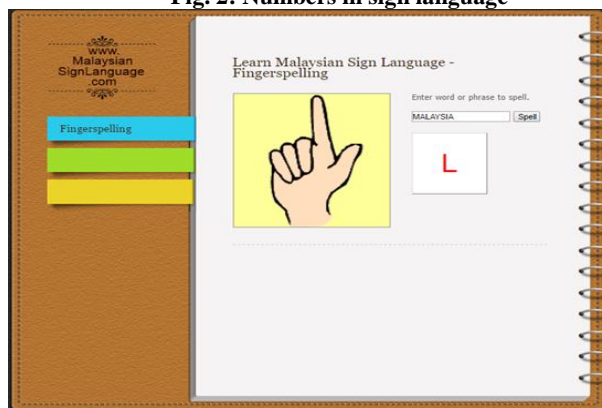


Fig. 3: Fingerspelling



Fig. 4: Words in sign language



Fig. 5: Sentences in sign language

4. DEVELOPMENT OF SIGN LANGUAGE ANIMATION CONTENT

As the web application is heavy on content, much time and effort were spent to ensure the appropriateness and level of comprehensiveness of the content that need to be included in the learning package. Most of the content of the learning package are based on the book entitled "Belajar Bahasa Isyarat Dalam Sepuluh Jam" [10] and "Bahasa Isyarat Malaysia" that was published by the Malaysian Federation of the Deaf [9]. Besides the books, the learning package is also based on the two-day course conducted by the book author himself.

In designing the animation content for the sign language, specific body parts involved in the animation are individually drawn on the computer. These drawings are then assembled to create a storyboard. Each storyboard represents the important points of a particular animated sign language word. Some of these body parts are reused in other sign language words. These body parts are then assembled to create a sequence of movements.

The images and animation was created using photo editing application and Adobe Flash. Animation templates were created and used to ensure standardisation among all animation files. The individual animation consists of reusable hand and finger objects in a variety of pose as well as the main head and body parts. Each of the objects was drawn and coloured individually based on a live model. The manipulated graphics were then converted into symbols and were stored into the library. The symbols were reused in multiple occasions, and the instances of the symbol were manipulated without changing the master symbol or increasing the file size of the animation. Figure 6 shows samples of the main animation character design objects created for the web application. In addition, other reusable objects being created are for the alphabet, numbers and various hand shapes, as shown in Figure 7.



Fig. 6: Main character reusable design objects (sample)

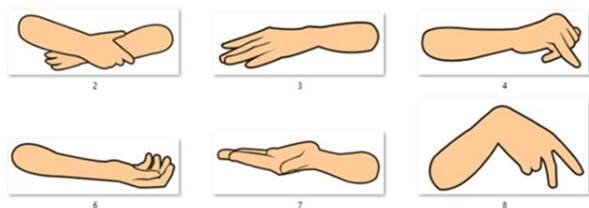


Fig. 7: Reusable hand shape objects (sample)

The stage was created to enable the arrangement of the imported symbols. The animation was coordinated and assembled on separate layers. The animation was then created by changing the content of successive frames based on a specified timeline. In this research, the frame-by-frame and

tweened methods were both used in creating the animation sequence. After the animation has been finalised, it was published into animated GIF format, which is an open standard and the main file format for distributing web content. The animation files were then stored in a specific project folder. Information about and referenced to all the animation files is stored in a MySQL database tables.

5. DEVELOPMENT OF ANIMATION IN FINGERSPELLING

Users can form and learn custom spellings in sign language. To achieve this, JavaScript functions were developed for use in this web application. One strong characteristic of using JavaScript is that it allows images and animation to be loaded very fast without the need to refresh the page. The control enables each character that makes up a letter or number to be individually activated and highlighted in a specified time interval.

Figure 8 illustrates how the control is used to animate the sign language equivalent of the spelling "L-O-V-E". Individual character is activated and highlighted based on a specific time interval. The animation for the current active character is played.

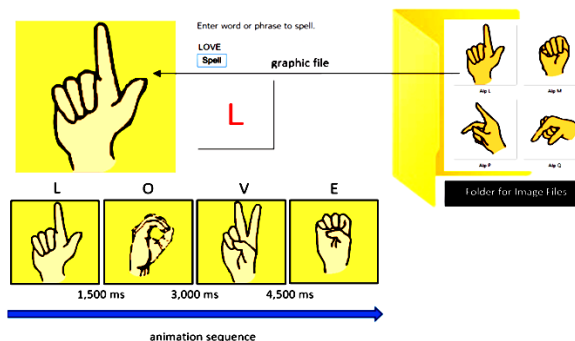


Fig. 8: Main process for animating fingerspelling

Appendix 1 lists the JavaScript function to animate an individual letter or number in the fingerspelling of a word or phrase. The *alphanumeric* function first validates the input text to only allow letters of the alphabets and numbers. Once validated, the function calls the *spellWord* function. The function first converts the letters of the user input text to uppercase so that a uniform display of uppercase letters be shown together with their respective representation in sign language. Each of these letters or numbers of the text is stored in a string array. The function then gets the first element of the array and concatenates it with a partial path to form a specific file location of the particular sign language image file that represents the letter or number. The image container then displays the sign as background image. For a blank space or a full stop character, the location points to a blank image file. The index is then increased by one and a time interval of 1,500 milliseconds is set. The process loops and the function calls itself and points to the next element of the array. The next sign language image file is then displayed. The process repeats until the index of the array equals the length of the array. Here, the set time interval is cleared and the process where the function calls itself is terminated. The animation stops with the display of the final letter of the input text.

6. DEVELOPMENT OF ANIMATION IN SIGN LANGUAGE WORDS

The development of individual sign language words has been explained in the previous section in this paper. To display word options to be selected, a drop-down list box is used on the page. Using PHP, the displayed options in the drop-down list box is populated from the available word values fetched from the database table. Several custom JavaScript functions were developed for use in the web application. One strong characteristic of using JavaScript is that it allows images and animation to be loaded very fast without the need to refresh the page. The main process of selecting and displaying the animation for a sign language word is shown in Figure 9. The PHP function for populating the drop-down list box is listed in Appendix 2 (a). On selected, the drop-down list box calls a JavaScript function to play an individual animation file for the selected sign language word, as listed in Appendix 2 (b). The *On Change* function gets the index and consequently the value of the selected option of the drop-down list box. The function then concatenates it with a partial path to form a specific file location of the particular sign language animation file that represents the word. The image container that displays the animated graphic file as background image. The function also sets the value of the label with value of the selected option.

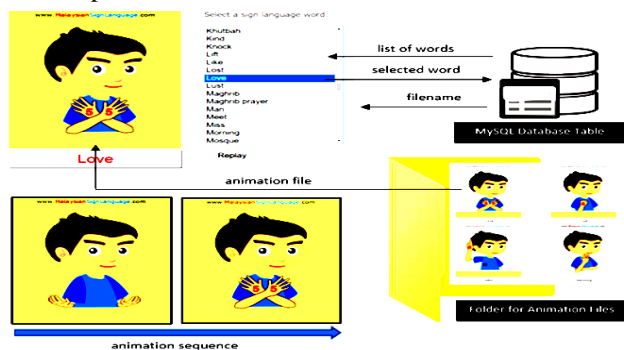


Fig. 9: Main process for displaying animation for sign language words

7. DEVELOPMENT OF ANIMATION IN SIGN LANGUAGE SENTENCES

Users can form and learn custom phrases and sentences (animation) from the sign language words within the database. The main process of forming user-selected sentence in sign language is shown in Figure 10.

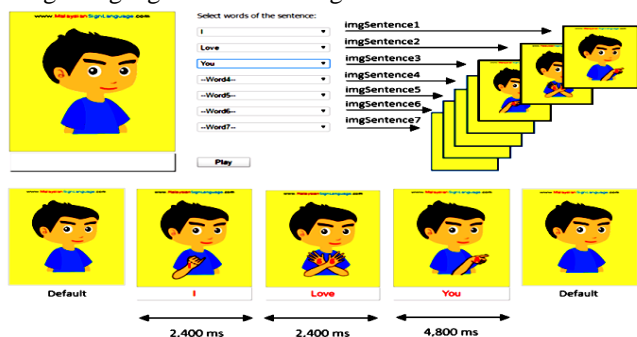


Fig. 10: Main process for displaying animation for sign language sentences

The application allows a maximum of seven words to form the sentence. The selection of each word is done using seven

drop-down list boxes. The drop-down list boxes are populated from the available word values fetched from the database table.

To animate sentences in sign language, an HTML table cell is used as a container for displaying animation as the background image. Altogether there are eight table cells each is given a specific id. The first cell displays a default animation before the sign language sentence being played. The other seven cells correspond to the seven drop-down list boxes for the words. A “display none” style is assigned to the seven cells; therefore making it invisible and only one of the eight cells appear at any one time. These processes are reflected by the *playSentence* function partially listed in Appendix 3.

The *playSentence* function clears the animation container by making all the table cells invisible. The function then gets the first word selected from the first drop-down list and concatenates it with a partial path to form a specific file location of the particular sign language animation file that represents the word. The function then makes the animated graphic file as the background image of the table cell. However, as the table cell's properties is display none, the animation is not visible on the page. At this point, the function appends a counter for the number of words in the formed sentence by one if the drop-down list box is selected.

The process repeats for drop-down list boxes 2 to 7 and each of the table cells which form the animation container now has specific animated graphic file as background image. The function will then parse each of the drop-down list box to determine the number of words that are selected to form the sentence. The *playSentence* function then calls the *cycleImage* function, listed in Appendix 4.

What the function does is that it goes to the first word in the sentence and changes its table cell properties to display inline-block, which makes the animated graphic file visible to the users. The function also uses a timer function to go the second word in the sentence. The timer is set to 2,400 milliseconds before the function calls itself. The display properties for the table cell for the first word are again changed to display none and the second is made visible by changing its properties to inline-block. The process repeats until the last word in the sentence. Once there is no more word in the sentence, the function breaks the loop after 4,800 milliseconds and reverts the background image of the animation table cell to the default animation.

8. USABILITY EVALUATION

For this research, the public were invited to try the learning application. Participants were from a mixture of different age groups, gender, race, computer literacy level, sign language knowledge and relationship to deaf family member or friend. The participants were then asked to complete a questionnaire that was designed to gather information on their background and the usability and effectiveness of the Malaysian Sign Language learning application. 162 participants completed the questionnaire. The analysis of the usability evaluation is tabled in Table 1. As a whole, they reacted positively where at least 90% of the participants gave a positive feedback to the survey questions. In terms of animation incorporated to represent different aspects of sign language, 93% agree that the animation is clear and effective. More than 90% think that

the learning of sign language using the application is easy and enjoyable, and the application helps in the learning of sign language. Similarly, the majority of participants of the usability evaluation agree that the application allows them to learn sign language interactively.

Table 1: Analysis of participants' evaluation.

1 - Strongly disagree; 2 – Disagree; 3 - Neutral; 4 - Agree; 5 - Strongly agree					
Evaluation Criteria	Result in %				
	1	2	3	4	5
1. Easy to use.	0	3	0	39	58
2. Enjoy using it.	0	3	6	42	48
3. Helps in learning alphabets in sign language	0	0	10	32	58
4. Helps in learning numbers in sign language.	0	0	3	32	65
5. Helps in learning everyday words in sign language.	0	0	13	55	32
6. Helps in learning Islamic words in sign language.	0	0	3	43	53
7. Helps in learning spelling in sign language.	0	0	6	39	55
8. Helps in learning spelling of selected Islamic related words/names in sign language.	0	0	6	48	45
9. Helps learn how to form phrases/sentences in sign language.	0	3	6	48	42
10. The embedded video page of sign language is an interesting feature.	0	3	10	45	42
11. The index page makes it easy to view multiple words in sign language.	0	0	19	35	45
12. The bilingual feature of Malay and English is helpful.	0	3	6	42	48
13. The address to the web application is easy to remember.	0	3	10	39	48
14. The content is organised in a structured and orderly way.	0	3	6	39	52
15. The navigation from one page to another is easy.	0	3	3	39	55
16. The character in the animation is pleasant looking.	0	3	16	29	52
17. Uses suitable colour combination.	0	0	10	42	48
18. Uses attractive background graphics.	0	0	16	48	35
19. The wordings are clear and easy to read.	0	0	0	39	61
20. The animation is clear and effective.	0	0	6	45	48
21. Every instruction given is easy to understand.	0	0	0	39	61
22. The sign language content reflects Malaysia.	0	0	10	45	45
23. Allows me to learn sign language interactively.	0	0	6	32	61

24. Allows me to learn sign language at any time and place.	0	0	16	42	42
25. Would use the web application again if I want to learn about sign language.	0	0	10	45	45
26. Would recommend the application to anyone interested in learning sign language.	0	0	3	45	52
27. Would like to see more sign language words added.	0	0	6	29	65
28. Would like to see more features added.	0	0	10	26	65
29. Would be interested in a version of the user interface customised for mobile devices.	0	0	3	32	65
30. This web application is a good and beneficial corporate social responsibility (CSR) initiative by MARA and Universiti Kuala Lumpur.	0	0	3	35	61

9. CONCLUSION

The web application succeeded in providing the public easy access to the learning of sign language. The web application is unique as it a bilingual learning web application where the mediums of instruction used are Malay and English. It uses an adorable two-dimensional animated cartoon character to perform the signs. The animation gives a much clearer representation of sign language and more appealing especially to young users. It also uses text narration in synchronisation with the animation. This helps users to learn sign language more effectively.

The learning package is also distinctively Malaysian. The use of the Malay language and the content reflect the Malaysian society. It is interactive and promotes self-paced learning. This is important especially for those who do not have the time to attend sign language course. It also complements the existing conventional classroom-based learning of sign language. Furthermore, it allows new signs to be included in the future.

Currently, the web application has attracted more than 16,000 visitors. It has achieved top ranking for Google, Bing and Yahoo search engine for the keywords Malaysian Sign Language and Bahasa Isyarat Malaysia. As such, the web application can be considered a primary resource for learning Malaysian Sign Language on the web.

Although the web application has successfully achieved the stated objective of this research, there is always room for improvement. New words and signs may be added to the existing vocabulary. The modules may be upgraded to generate user-defined numbers in sign language. In addition, a mobile version of the web application can be developed.

10. REFERENCES

- [1] World Federation of the Deaf and Swedish National Association of the Deaf. Global Survey Report. Helsinki: World Federation of the Deaf and Swedish National Association of the Deaf, 2008.
- [2] UNICEF. (2014). State of the World's Children. Retrieved June 19, 2014, from http://www.unicef.org/malaysia/sowc_sowc_13-situation_in_malaysia.html
- [3] Che Omar, H. Penterjemahan Dan Bahasa Isyarat. Kuala Lumpur: Universiti Sains Malaysia & Institut Terjemahan Negara Malaysia, 2009.
- [4] National Institute on Deafness and Other Communication Disorders. (2014). American Sign Language. Retrieved May 3, 2014, from <http://www.nidcd.nih.gov/staticresources/health/hearing/NIDCD-American-Sign-Language.pdf>
- [5] e-Hearme. (2014). Retrieved June 12, 2014, from <http://www.e-hearme.net>
- [6] Maxis Cyberkids. (2014). e-Isyarat. Retrieved June 12, 2014, from <http://cyberkids.my/e-isyarat/>
- [7] Virtual Malaysian Sign Language. (2014). Retrieved June 10, 2013, from <http://www.vmsl.com>
- [8] Bahasa Isyarat Malaysia (BIM) / Malaysian Sign Language (MySL). (2016). Retrieved June 20, 2014, from Facebook: <https://www.facebook.com/groups/BahasaIsyaratMalaysia/>
- [9] Persekutuan Orang Pekak Malaysia. (2000). Bahasa Isyarat Malaysia. Kuala Lumpur: Persekutuan Orang Pekak Malaysia.
- [10] Tan, Y. Belajar Bahasa Isyarat Dalam Sepuluh Jam. Kuala Lumpur: Persatuan Orang-Orang Pekak Selangor Dan Wilayah Persekutuan, 1998.

APPENDIX 1: FUNCTION FOR ANIMATING FINGERSPELLING

```
// Function to check letters and numbers
function alphanumeric(inputtxt) {
    var letterNumber = /^[0-9a-zA-Z .]+$/;
    if(inputtxt.value.match(letterNumber)) {
        spellWord('signImage');
        return true;
    } else {
        alert("Please enter letters or numbers only.");
        return false;
    }
}
```

```
// Function to Load Letter images
function spellWord(id) {
    var index=0;
    var word = document.getElementById("word")
        .value.toUpperCase();

    var intObject= setInterval(function() {
        var filelocation = './Alphabet/Alp '+word[index]+'.png';
        document.getElementById('signImage').style
            .backgroundImage='url('+filelocation+')';

        if (word[index]==' ' || word[index]=='.') {
            document.getElementById('signImage').style
                .backgroundImage='url("./Alphabet/Alp.png")';
            document.getElementById('textAlpha').value=word[index];
        } else {
            document.getElementById('signImage').style
                .backgroundImage='url('+filelocation+')';
            document.getElementById('textAlpha').value=word[index];
        }

        index++;
        if(index==word.length){
            clearInterval(intObject);
        }
    }, 1500);
}
```

APPENDIX 2 (A) FUNCTION FOR POPULATING DROP-DOWN LIST BOX

```
<?php
echo("<select name='$selectName' id='$selectName'
    style='width: 200px' size='$listSize'
    onchange=\"OnChange(this.form.$selectName);\">");
if($listSize<=1){
    echo("<option value='999'>--$selectName--</option>");
}
while($wordArray = mysqli_fetch_array($result)){
    $wordId = $wordArray['word_id'];
    $wordEnglish = $wordArray['word_english'];
    $wordMalay = $wordArray['word_malay'];
    $wordFilename = $wordArray['word_filename'];
    echo "<option value = '$wordEnglish'>$wordEnglish
    </option>"; // $wordMalay for the Malay version
}
echo "</select>";
$num_of_rows = mysqli_num_rows($result);
?>
```

APPENDIX 2 (B) FUNCTIONS FOR DISPLAYING SIGN LANGUAGE WORDS

```
function OnChange(dropdown)
{
    var myindex = dropdown.selectedIndex
    var SelValue = dropdown.options[myindex].value
    var filelocation = './Word/' + SelValue + ".gif";
    document.getElementById('embedDiv').style
        .backgroundImage='url('+filelocation+')';
    document.getElementById('textWord0').value=SelValue;
    return true;
}
```

APPENDIX 3: FUNCTION FOR PREPARING THE ANIMATION CONTAINER FOR WORDS IN SENTENCE

```
function playSentence() {

    displayImgCount = 0;

    document.getElementById("imgSentence0").style.display = "none";

    document.getElementById("imgSentence1").style.display = "none";
    document.getElementById("imgSentence2").style.display = "none";
    document.getElementById("imgSentence3").style.display = "none";
    document.getElementById("imgSentence4").style.display = "none";
    document.getElementById("imgSentence5").style.display = "none";
    document.getElementById("imgSentence6").style.display = "none";
    document.getElementById("imgSentence7").style.display = "none";

    var wordCount = 0;

    var Word1 = document.getElementById('Word1');
    var myindex1 = Word1.selectedIndex;
    var SelValue1 = Word1.options[myindex1].value;
    var filelocation1 = "./Word/" + SelValue1 + ".gif";
    document.getElementById('imgSentence1').style.backgroundImage
        = 'url("' + filelocation1 + '")';

    if (myindex1 != 0) {
        wordCount++;
    }
    .
    .
    .
}
```

APPENDIX 4: FUNCTION FOR DISPLAYING SIGN LANGUAGE WORD ANIMATION IN SENTENCE

```
function cycleImage(){

    displayImgCount = displayImgCount === wordCount ?
        wordCount : displayImgCount + 1;
    document.getElementById("imgSentence" + displayImgCount).
        style.display = "inline-block";
    document.getElementById("dot" + displayImgCount)
        .style.display = "inline-block";
    document.getElementById("imgSentence" + displayImgCount)
        .style.border = "1px groove";
    document.getElementById("imgSentence" + displayImgCount)
        .style.margin = "3px 30px 0 0";
    document.getElementById("imgSentence" + displayImgCount)
        .style.padding = "4px";

    .
    .
    .

    if (displayImgCount >= wordCount) {
        SelValueEnd = '999';
        setTimeout(function () {
            filelocationend = "./Word/" + SelValueEnd + ".gif";
            document.getElementById("imgSentence" + displayImgCount)
                .style.backgroundImage = 'url("' + filelocationend + '")';
            document.getElementById("imgSentence" + displayImgCount)
                .style.backgroundColor = "#FFFF99";
        }, 4800); // End FUNCTION TIMEOUT
        return;
    } // End if

    setTimeout(cycleImage, 2400);
} // End Function cycleImage()
```