

Augmented reality application with speech recognition for malay spelling

Norizan Mat Diah¹, Akmal Hakim Mohammad Mokhriz¹, Norsafinar Rahim², Suzana Ahmad¹,
Elly Johana Johan³, Zainura Idrus¹

¹Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, Shah Alam, Malaysia

²Centre for Instructional Technology and Multimedia, Universiti Sains Malaysia, Penang, Malaysia

³Department of Computer and Mathematical Sciences, Universiti Teknologi MARA Cawangan Pulau Pinang, Permatang Pauh Campus, Permatang Pauh, Malaysia

Article Info

Article history:

Received Dec 10, 2024

Revised Jun 3, 2026

Accepted Jul 5, 2026

Keywords:

Augmented reality

Malay language

Speech recognition

Spelling

ABSTRACT

This project aims to revolutionize the process of teaching Malay spelling to young learners by using new technologies, notably augmented reality (AR), and voice recognition, on the Android smartphone platform. The objective is to create speech recognition algorithms specifically adapted for mobile AR applications and extensively test their accuracy on mobile devices. The study begins with an initial phase of analysis and rapid design, followed by an iterative prototyping cycle using a systematic approach. This cycle includes developing, testing, and continuously refining the mobile spelling application, ultimately producing an improved version ready for final testing. The result is a fully working smartphone application that smoothly incorporates AR and voice recognition technology, providing children with an engaging and correct spelling learning experience. This study demonstrates the effective adoption of cutting-edge technology in education and highlights a practical and innovative tool for educators and parents seeking innovative approaches to enhance language learning.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Norizan Mat Diah

Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA

40450 Shah Alam, Selangor, Malaysia

Email: norizan289@uitm.edu.my

1. INTRODUCTION

Early childhood education is important for cognitive development and influences educational performance, especially reading proficiency. Studies by [1], [2] highlight the profound effect of early learning experiences on children's reading skills and cognitive development, hence improving their performance in early education. According to studies by [3], [4], children's inherent curiosity during their formative years drives them to explore their surroundings and engage with adults, facilitating a natural process of learning. This early development phase is characterised by crucial brain development that enhances key cognitive functions such as attention, memory, executive skills, and language [5], [6]. Increasing screen-based media exposure during this developmental phase introduces significant factors that can affect children's optimal functional and overall development.

Neuroplasticity, as described by [7], highlights the brain's adaptability during development, where positive experiences promote learning and cognitive progress, while negative experiences may hinder cognitive growth. In today's digital ecosystem, mobile devices and interactive applications have become important aspects of children's lives, significantly influencing their development [8], [9]. Approximately 75% of children have access to smartphones or tablets [10], making educational applications an essential

component of modern educational practices [11]. A wide range of programs focuses on fundamental life skills such as reading and speaking [12]. Recognising the importance of acquiring reading and speaking skills in early childhood, this study aims to develop an educational application focused on spelling, incorporating speech recognition to monitor progress and augmented reality (AR) to improve engagement [13]. This provides a foundation for analysing the convergence of early childhood education, screen-based media, and innovative educational tools. The proposed spelling application is designed to address young learners' developmental needs and technology preferences by utilising AR and speech recognition to offer an engaging and effective educational experience.

2. RESEARCH METHOD

The fundamental skills of reading, writing, and speaking form an important foundation in a child's development, enabling knowledge acquisition as they advance through their education [14], [15]. These skills are frequently developed naturally through interaction with caregivers and the surrounding environment. However, the widespread use of mobile phones among children has raised concerns, as technology often replaces outdoor play and face-to-face communication, potentially affecting their overall development [16]. In response to these challenges, adapting teaching and learning methodologies to align with the digital age has become imperative. Interactive educational tools such as games, applications, and movies provide valuable learning opportunities, even during children's engagement with electronic devices [17]. While excessive smartphone usage might adversely impact learning outcomes, research indicates that utilising these devices for educational reasons can have beneficial consequences [18].

The increasing focus on developing educational applications that integrate modern teaching strategies, such as those targeting reading, writing, speaking, and problem-solving skills, highlights the necessity for innovative approaches [11], [19]. A proposed method involves the integration of AR into educational applications, specifically designed to enhance reading and spelling skills. The interactive and immersive features of AR engage learners, providing a unique educational experience. The integration of AR with reading and spelling instruction represents a revolutionary advancement in the world of education, improving interactivity, engagement, and dynamism in the learning process [20].

AR enhances the reading experience by animating characters and scenes, consequently improving comprehension and fostering a higher enthusiasm for literature [21]. In spelling education, AR revolutionizes the learning process by incorporating interactive games that engage learners in immersive experiences [22]. This technology enables users to interact and navigate artificial three-dimensional (3D) environments in real-time, replicating actual or alternative scenarios. Research indicates that integrating AR into educational applications enhances learning outcomes compared to traditional methods, providing more engaging and immersive educational experiences [23], [24]. Incorporating AR into reading and spelling instruction can transform traditional teaching methods, fostering a dynamic and effective learning environment that fulfills the needs of today's learners. AR applications improve the comprehension and retention of reading material by animating characters, objects, or text. Furthermore, advanced speech recognition technology is important for evaluating student proficiency and offering immediate feedback on pronunciation and spelling [25]-[27]. This incorporation supports a personalised learning experience, enabling students to receive customised feedback that fulfils their specific needs, enhancing more effective skill development.

Multisensory AR integrates many learning approaches, enhancing exercises by linking the exercise with real-world contexts. This comprehensive strategy motivates the students to engage in their educational journey actively, improving engagement and developing a sense of empowerment. Incorporating AR with voice recognition technology allows educators to transform traditional teaching methods and offer new opportunities for student success.

Undoubtedly, AR possesses tremendous potential in educational applications, improving the learning experience while delivering improved outcomes. This corresponds with current existing evidence of AR's beneficial impacts on education, emphasizing the importance of persistently investigating and integrating new technologies to enhance educational systems. By embracing the changing educational landscape, this strategy leverages technology to actively engage students, accommodate various learning styles, and enhance technical language skills and analytical thinking.

3. RESEARCH METHOD

The proposed research methodology delineates the architectural components (Figure 1) and the systematic flow of the educational application, focusing on incorporating AR and voice recognition technology. This methodology consists of three main components: the user, the Android device, and the AR application, which is further categorised into two subcomponents: Google Speech-to-Text API and Unity 3D.

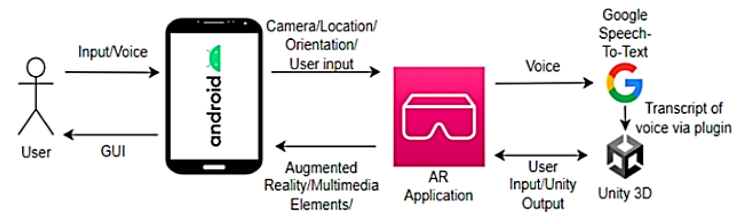


Figure 1. Application architecture

- i. User and Android device interaction
 - a. The user acts as the primary component, engaging with the system by delivering inputs such as audio commands and selecting the levels using the graphical user interface (GUI) on the Android device.
 - b. The Android device is the application platform, receiving user input and supplying vital geolocation data, including camera, location, and orientation information. This information is important for setting up the AR environment.
- ii. AR application processing
 - a. The AR application functions as the central processing component, creating the AR environment and incorporating voice recognition technology.
 - b. Two subcomponents, Google Speech-to-Text API and Unity 3D, work collaboratively to process voice audio from the user. The API transcribes the voice input into a command understood by Unity 3D.
 - c. A Unity 3D plugin facilitates seamless communication, acting as a bridge between the API and Unity 3D. Once the command is understood, Unity 3D processes the necessary steps in the background to generate an appropriate response for display.
- iii. Application flow
 - a. The flow of the application is depicted in a flowchart, illustrating the sequence of actions from initiation to completion (Figure 2).
 - b. The user initiates the system by selecting a category (e.g., animals and colours) that defines the spelling topic. The application responds by displaying an object in AR, accompanied by its name, to prompt the user to read.
 - c. After the user reads the object, the application undergoes processes to verify the correctness of the reading. If correct, a congratulatory message is displayed, and the user can choose to continue. If incorrect, a “try again” message is presented, allowing the user another attempt.
 - d. The flow continues cyclically, with the option for the user to stop after an attempt, concluding the flow.

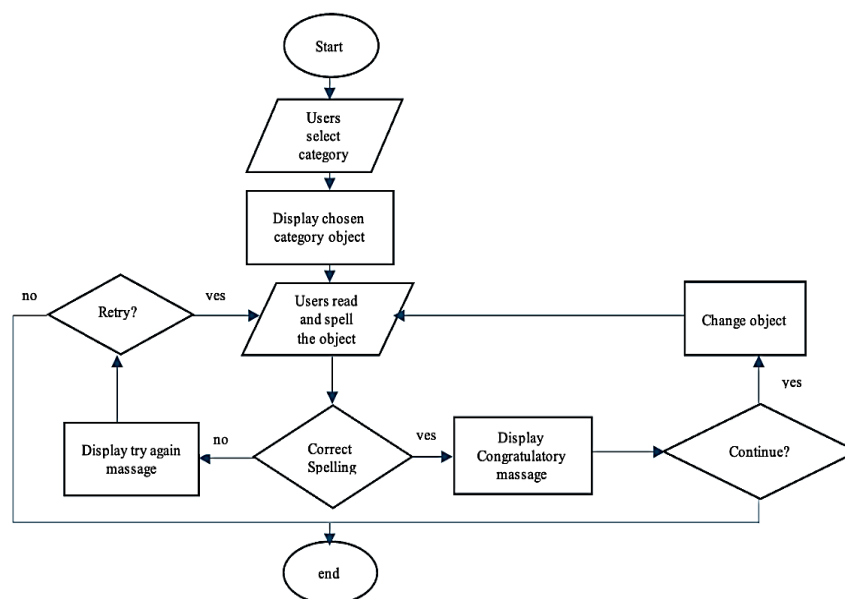


Figure 2. Flowchart of the application

This methodology provides a clear structure for developing and implementing the educational application, ensuring the seamless integration of user interaction, AR technology, and voice recognition capabilities. As the flowchart outlines, the systematic flow guides the user through an engaging learning experience, promoting interaction and feedback within the application.

3.1. Augmented reality development

The user interface is one of the most important components developed in this project, as it serves as the primary medium through which users interact with the application. Since the application is designed for children, the interface incorporates large icons, simple text, and visually appealing graphics to facilitate ease of use and engagement. Figure 3 presents the application interfaces, including Figure 3(a) the main menu, Figure 3(b) category selection, Figure 3(c) the AR interaction screen, and Figure 3(d) the feedback screen.

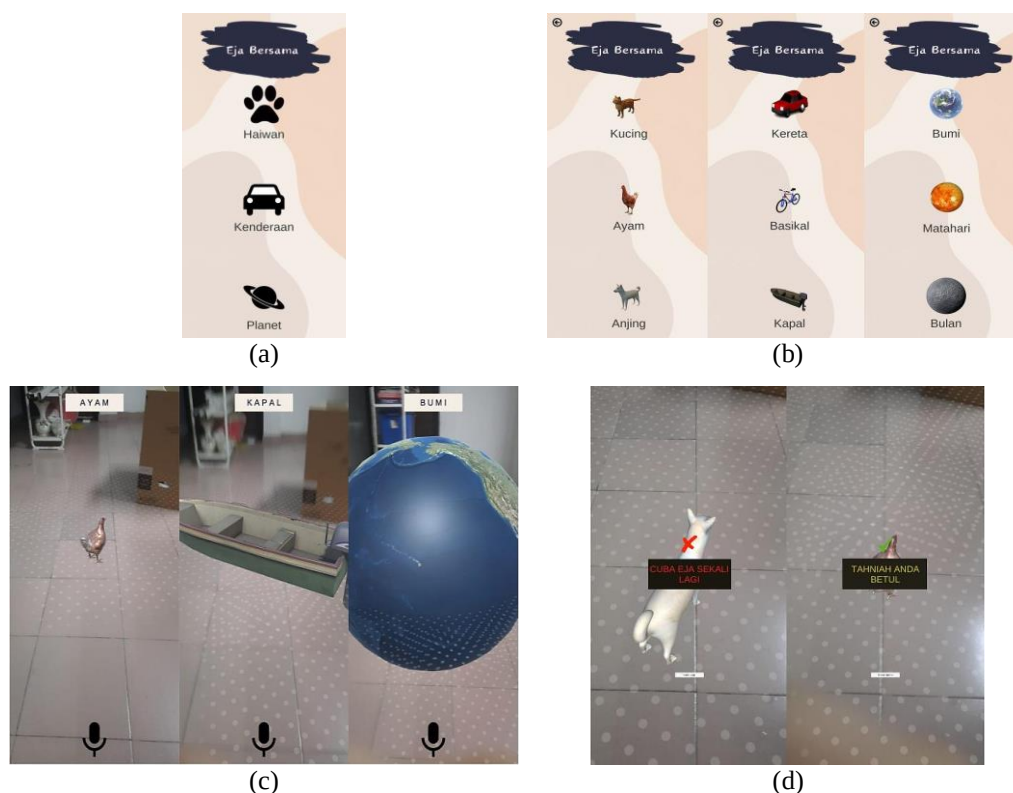


Figure 3. User interfaces of the augmented reality application: (a) main menu, (b) category selection, (c) AR interaction screen, and (d) feedback screen

The main menu is displayed when the application is launched and provides access to the available learning categories. Users can select one of three categories: Animal, Vehicle, or Planet. Each category contains three learning objects represented by images and text labels, allowing users to easily identify and select the desired object.

After an object is selected, the AR interaction screen is displayed. This interface includes a back button for navigation, a text box showing the name of the selected object, and a microphone button used to activate the speech recognition feature. To render the AR object, users must point the device camera toward a flat surface until the application detects a plane, indicated by a white dotted floor pattern. Users can then tap the detected surface to place the 3D object into the real-world environment.

Once the object is displayed, users can activate the speech recognition feature by pressing the microphone button. The system then allows approximately three seconds for users to verbally spell the displayed object's name. Based on the user's pronunciation, the application provides immediate feedback. If the spelling is incorrect, an error message accompanied by a cross icon is displayed, prompting the user to try again. Conversely, if the spelling is correct, a success message with a checkmark icon is shown to congratulate the user. The feedback screen also provides navigation options, allowing users either to repeat the activity or return to the main menu.

3.2. Augmented reality object placement implementation

The AR application was developed using Unity and AR Foundation to enable real-time object placement within the physical environment. The overall implementation is illustrated in Figure 4. As illustrated in Figure 4(a), the Awake() function is used to initialize and reference the required AR Foundation components, including the ARRaycastManager, ARPlaneManager, and ARAnchorManager. These components are automatically assigned using the GetComponent() method, allowing the system to detect surfaces, perform raycasting operations, and manage anchors for object placement.

Once the AR components have been initialized, the object placement process is executed through user interaction. Figure 4(b) presents the object instantiation mechanism used in the application. When the user touches a detected AR plane, the system performs a raycast operation to identify the touched location and obtain the corresponding pose and plane information. An anchor is then created at the detected position, serving as the reference point for placing the selected 3D object. The application ensures that only a single instance of the object is maintained within the AR environment. If no object exists, a new instance is generated at the detected location; otherwise, the existing object is repositioned according to the user's selection. This mechanism improves resource efficiency while providing a seamless and intuitive interaction experience. The implementation provides a stable and intuitive interaction mechanism, enabling users to place and reposition learning objects seamlessly in real time.

```
// On Awake(), we obtain a reference to all the required components.
// The ARRaycastManager allows us to perform raycasts so that we know where to place an anchor.
// The ARPlaneManager detects surfaces we can place our objects on.
// The ARAnchorManager handles the processing of all anchors and updates their position and rotation.
void Awake()
{
    m_RaycastManager = GetComponent<ARRaycastManager>();
    m_AnchorManager = GetComponent<ARAnchorManager>();
    m_PlaneManager = GetComponent<ARPlaneManager>();
    m_AnchorPoints = new List<ARAnchor>();
}
```

(a)

```
void Update()
{
    // If there is no tap, then simply do nothing until the next call to Update().
    if (Input.touchCount == 0)
        return;

    var touch = Input.GetTouch(0);
    if (touch.phase != TouchPhase.Began)
        return;

    if (m_RaycastManager.Raycast(touch.position, s_Hits, TrackableType.PlaneWithinLayers))
    {
        // Raycast hits are sorted by distance, so the first one
        // will be the closest hit.
        var hitPose = s_Hits[0].pose;
        var hitTrackableId = s_Hits[0].trackableId;
        var hitPlane = m_PlaneManager.GetPlane(hitTrackableId);

        // This attaches an anchor to the pose on the plane corresponding to the request hit,
        // and afterwards instantiates an instance of your chosen prefab at that point.
        // This prefab instance is parented to the anchor to make sure the position of the prefab is consistent
        // with the anchor. Once an anchor is attached to a surface it will be updated automatically by the ARAnchorManager as the surface's exact position is refined.
        var anchor = m_AnchorManager.AttachAnchor(hitPlane, hitPose);
        Instantiate(m_AnchorPrefab, anchor.transform);

        if (m_AnchorPrefab == null && m_IsPressed() == false)
        {
            m_AnchorPrefab = Instantiate(m_AnchorPrefabInt, hitPose.position, m_AnchorPrefabInt.transform.rotation);
        }
        else
        {
            m_AnchorPrefab.transform.position = hitPose.position;
        }

        if (anchor == null)
        {
            Debug.Log("Error creating anchor.");
        }
        else
        {
            // Store the anchor so that it may be removed later.
            m_AnchorPoints.Add(anchor);
        }
    }
}
```

(b)

Figure 4. AR object placement implementation (a) AR component initialization and (b) object placement mechanism

3.3. Voice recognition development

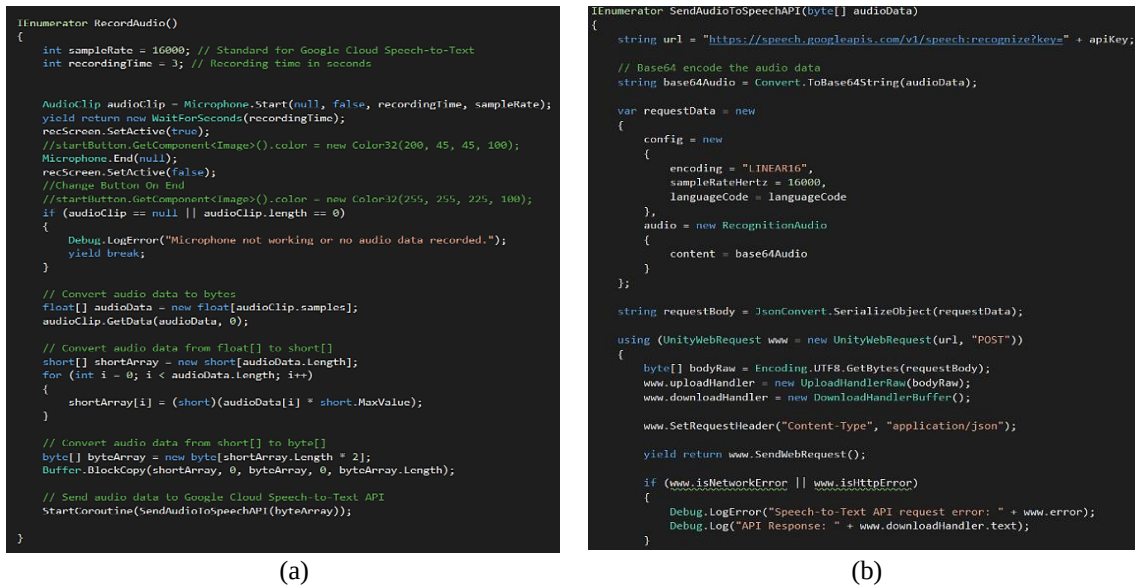
The voice recognition functionality was implemented using the Google Cloud Speech-to-Text API due to its native support for the Malay language and its ease of integration with the Unity development environment. The implementation consists of two main stages: audio recording and speech recognition processing. Figure 5 presents the implementation of the voice recognition functionality, consisting of (a) the audio recording process and (b) the integration with the Google Cloud Speech-to-Text API.

As shown in Figure 5(a), the audio recording process begins when the user presses the microphone button. The application records the user's voice using Unity's microphone functionality at a sampling rate of 16,000 Hz, which is the recommended sampling rate for Google Cloud Speech-to-Text services. The recording duration is limited to three seconds to ensure a responsive user experience while capturing

sufficient speech input for recognition. Once recording is completed, the captured audio is validated and converted into the appropriate format for transmission to the speech recognition service.

The second stage involves transmitting the recorded audio to the Google Cloud Speech-to-Text API for processing. As illustrated in Figure 5(b), the recorded audio data is converted into Base64 format and embedded within a JSON request structure. The request includes the required configuration parameters, such as the audio encoding type (Linear16), sampling rate (16,000 Hz), and language code (ms-MY) to enable Malay language recognition. The request is then sent to the Google Cloud Speech-to-Text server through a secure web service call.

Upon successful processing, the API returns the recognized speech as text, which is subsequently compared with the target word displayed in the application. If the recognized text matches the expected spelling, the application provides positive feedback to the user. Otherwise, corrective feedback is displayed, encouraging the user to repeat the spelling activity. This implementation enables real-time speech recognition and supports interactive language learning through immediate feedback.



(a)

```
IEnumerator RecordAudio()
{
    int sampleRate = 16000; // Standard for Google Cloud Speech-to-Text
    int recordingTime = 3; // Recording time in seconds

    AudioClip audioClip = Microphone.Start(null, false, recordingTime, sampleRate);
    yield return new WaitForSeconds(recordingTime);
    recScreen.SetActive(true);
    //startButton.GetComponent<Image>().color = new Color32(200, 45, 45, 100);
    Microphone.End(null);
    recScreen.SetActive(false);
    //Change Button On End
    //startButton.GetComponent<Image>().color = new Color32(255, 255, 225, 100);
    if (audioClip == null || audioClip.length == 0)
    {
        Debug.LogError("Microphone not working or no audio data recorded.");
        yield break;
    }

    // Convert audio data to bytes
    float[] audioData = new float[audioClip.samples];
    audioClip.GetData(audioData, 0);

    // Convert audio data from float[] to short[]
    short[] shortArray = new short[audioData.Length];
    for (int i = 0; i < audioData.Length; i++)
    {
        shortArray[i] = (short)(audioData[i] * short.MaxValue);
    }

    // Convert audio data from short[] to byte[]
    byte[] byteArray = new byte[shortArray.Length * 2];
    Buffer.BlockCopy(shortArray, 0, byteArray, 0, byteArray.Length);

    // Send audio data to Google Cloud Speech-to-Text API
    StartCoroutine(SendAudioToSpeechAPI(byteArray));
}
```

(b)

```
IEnumerator SendAudioToSpeechAPI(byte[] audioData)
{
    string url = "https://speech.googleapis.com/v1/speech:recognize?key=" + apiKey;

    // Base64 encode the audio data
    string base64Audio = Convert.ToBase64String(audioData);

    var requestData = new
    {
        config = new
        {
            encoding = "LINEAR16",
            sampleRateHertz = 16000,
            languageCode = languageCode
        },
        audio = new RecognitionAudio
        {
            content = base64Audio
        }
    };

    string requestBody = JsonConvert.SerializeObject(requestData);

    using (UnityWebRequest www = new UnityWebRequest(url, "POST"))
    {
        byte[] bodyRaw = Encoding.UTF8.GetBytes(requestBody);
        www.uploadHandler = new UploadHandlerRaw(bodyRaw);
        www.downloadHandler = new DownloadHandlerBuffer();

        www.SetRequestHeader("Content-Type", "application/json");

        yield return www.SendWebRequest();

        if (www.isNetworkError || www.isHttpError)
        {
            Debug.LogError("Speech-to-Text API request error: " + www.error);
            Debug.Log("API Response: " + www.downloadHandler.text);
        }
    }
}
```

Figure 5. Voice recognition implementation (a) audio recording process and (b) Google Speech-To-Text API integration

4. RESULTS AND DISCUSSION

An accuracy test was conducted using a sample size of 100 simulations to ensure the voice recognition works as intended. Figure 6 shows the number of test requests done, which is 103 times, with an error of 0 and an average latency of 0.6 seconds. This indicates that the API has not failed to respond to a request within the sample time frame above 5 hours.

Based on Table 1, 103 tests were conducted amongst nine words. Each word has a minimum test count of 11, while certain words such as “Kucing”, “Basikal”, and “Matahari” have more due to errors occurring during testing. As seen in the table, every word with a test count of 11 has an accuracy of 100%, meaning the API understands what the user is saying and relays that message accurately and successfully back to Unity to be used in the scripts. On the other hand, words such as “Kucing” and “Basikal” have an accuracy of 91%, meaning they got one test wrong and “Matahari” has 84% accuracy with two tests wrong. It is due to multiple variables such as environmental, where there is too much noise while speaking, and speech error, where speaking too fast or slow will make it difficult for the API to understand.

The application testing uses the system usability scale (SUS) method. SUS has been widely applied as a rapid way of measuring the usability of technology-based applications [28]. The SUS questionnaire consists of ten items containing statements representing the feelings or responses of the users to a product interface. The statements in the SUS questionnaire look like those shown in Figure 7. This questionnaire uses a five-point Likert scale, with the criteria of “strongly disagree,” “disagree,” “neutral,” “agree,” and “strongly agree.” The ten statements are divided into positive and negative categories [28]-[30]. The positive statements have odd numbers (numbers 1, 3, 5, 7, and 9), and the negative ones have even

numbers (numbers 2, 4, 6, 8, and 10). Subtracting one score from each score that the respondent provided allows for calculating the score for each affirmative statement. The score for each negative statement is calculated by subtracting a score given by a respondent from five; hence, every item of statements has a minimal score of 0 and a maximal score of 2.5 so that the final score of each respondent ranges from 0 to 100. When it is written in a mathematical formula, the calculation of the SUS score for each respondent is as follows, in (1) [31], [32]:

$$SUS\ Score = \{(S_1 - 1) + (5 - S_2) + (S_3 - 1) + (5 - S_4) + (S_5 - 1) + (5 - S_6) + (S_7 - 1) + (5 - S_8) + (S_9 - 1) + (5 - S_{10})\} * 2.5 \quad (1)$$

Note: S_i = the- i item statement



Figure 6. Successful API usage

Table 1. Voice recognition accuracy

Word	Number of tests	Test accuracy
<i>Kucing</i>	12	91%
<i>Anjing</i>	11	100%
<i>Ayam</i>	11	100%
<i>Kapal</i>	11	100%
<i>Basikal</i>	12	91%
<i>Basikal</i>	12	91%
<i>Kereta</i>	11	100%
<i>Bumi</i>	11	100%
<i>Matahari</i>	13	84%
<i>Bulan</i>	11	100%

Question	1	2	3	4	5
	Strongly disagree			Strongly agree	
1. I think that I would like to use this system frequently.	☐	☐	☐	☐	☐
2. I found the system unnecessarily complex.	☐	☐	☐	☐	☐
3. I thought the system was easy to use.	☐	☐	☐	☐	☐
4. I think that I would need the support of a technical person to be able to use this system.	☐	☐	☐	☐	☐
5. I found the various functions in this system were well integrated.	☐	☐	☐	☐	☐
6. I thought there was too much inconsistency in this system.	☐	☐	☐	☐	☐
7. I would imagine that most people would learn to use this system very quickly.	☐	☐	☐	☐	☐
8. I found the system very cumbersome to use.	☐	☐	☐	☐	☐
9. I felt very confident using the system.	☐	☐	☐	☐	☐
10. I needed to learn a lot of things before I could get going with this system.	☐	☐	☐	☐	☐

Figure 7. System usability scale form

The SUS results obtained from six respondents indicate a high level of usability for the application. The highest SUS score recorded was 97.5, corresponding to grade A, while the lowest score was 70, corresponding to grade D. The application achieved an average SUS score of 85, which falls within grade B, indicating that users generally found the application easy to use, effective, and satisfactory in supporting their interaction with the system.

Figure 8 summarises the responses of all participants to each question, with a distinct pattern emerging from their feedback. Notably, odd-numbered questions are framed as positive statements, while even-numbered ones are presented as negative statements.

For positive statements in questions 1, 3, and 7, most respondents expressed a positive sentiment, with the majority opting for “strongly agree.” It suggests a unanimous agreement among participants regarding certain positive aspects of the system. Meanwhile, for negative statements in questions 2, 6, and 10, most respondents leaned towards the “strongly disagree” option. This consistent response implies a strong disagreement with the negative statements posed in these questions. For mixed responses, question 5 received a majority of “agree” responses, indicating a generally positive sentiment, while question 9 elicited an even split between “strongly agree” and “agree.” It suggests a more balanced view regarding certain aspects of system usability. Regarding the responses to questions 4 and 8, most respondents chose “disagree,” indicating a disagreement with the negative statements posed in these questions. This aligns with the overall positive sentiment observed in the participants’ responses.

The general pattern in the respondent’s answers suggests that the system is easy to use, requiring minimal effort to operate. A significant number of “strongly agree” and “agree” responses to positive statements, coupled with “strongly disagree” and “disagree” responses to negative statements, highlight a positive user experience. The collective responses provide a consistent and clear view of the system’s usability and overall user satisfaction. This analysis underscores a largely positive perception of the system and provides valuable insights that can guide future enhancements based on user feedback.

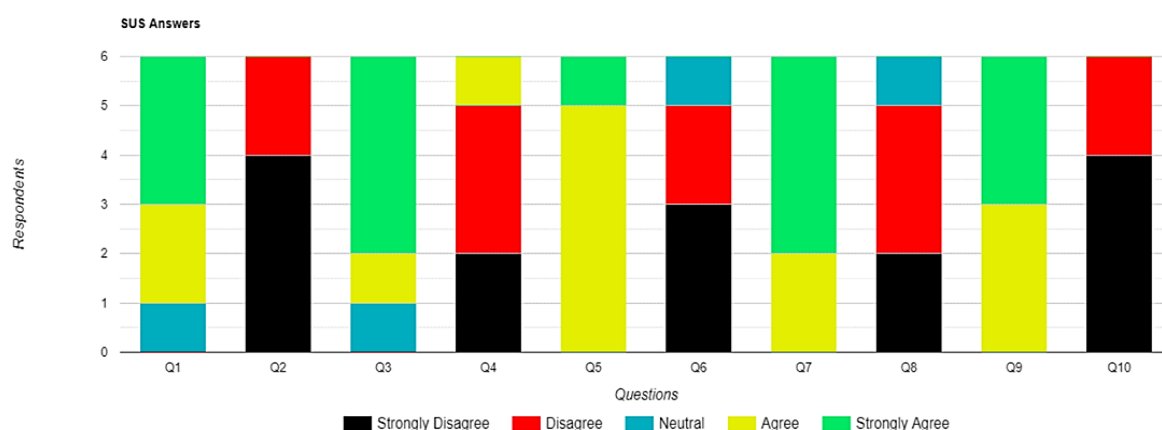


Figure 8. System usability scale answer spread graph

5. CONCLUSION

Lastly, this research effectively met its goals by employing new technologies, including AR and voice recognition, to improve the teaching of spelling in Malay for young learners on the Android smartphone platform. The creation of specialised speech recognition algorithms and thorough testing on mobile devices were critical components of the research methodology, which emphasised a methodical and iterative process. These efforts resulted in a fully working mobile spelling application that smoothly combines AR and voice recognition technology. This application satisfies the requirement of offering an engaging and amusing spelling experience for young learners while also demonstrating a high degree of accuracy throughout their learning journey. The study’s success demonstrates the potential influence of AR and voice recognition in educational technology, providing an important addition to the area. The resultant mobile application is a useful resource for instructors, parents, and students looking for novel ways to language instruction. Overall, the research highlights the need to combine cutting-edge technologies to build dynamic and effective learning environments, hence broadening the landscape of digital learning technology. This study not only demonstrates the transformational potential of technology in education, but it also paves the way for future developments and improvements in language teaching and beyond.

ACKNOWLEDGEMENTS

The authors would like to express their appreciation to the Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA for the support and facilities provided in facilitating this work.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Norizan Mat Diah	✓	✓		✓		✓		✓	✓	✓		✓	✓	✓
Akmal Hakim	✓		✓			✓		✓	✓					
Mohammad Mokhriz														
Norsafinar Rahim										✓	✓			✓
Suzana Ahmad		✓						✓		✓				
Elly Johana Johan								✓			✓			
Zainura Idrus				✓							✓			✓

C : Conceptualization

M : Methodology

So : Software

Va : Validation

Fo : Formal analysis

I : Investigation

R : Resources

D : Data Curation

O : Writing - Original Draft

E : Writing - Review & Editing

Vi : Visualization

Su : Supervision

P : Project administration

Fu : Funding acquisition

CONFLICT OF INTEREST STATEMENT

The author declares no conflict of interest.

DATA AVAILABILITY

The data of this research are available upon request from the corresponding author.

REFERENCES

- [1] E. Bodrova and D. Leong, *Tools of the Mind: The Vygotskian Approach to Early Childhood Education: Third Edition*. New York: Routledge, 2024. doi: 10.4324/9781003164920.
- [2] M. H. Bornstein et al., *Parenting and Child Development in Low- and Middle-Income Countries*. New York: Routledge, 2022. doi: 10.4324/9781003044925.
- [3] A. L. Bjerknes, T. Wilhelmsen, and E. Foyn-Bruun, "A systematic review of curiosity and wonder in natural science and early childhood education research," *Journal of Research in Childhood Education*, vol. 38, no. 1, pp. 50–65, Jan. 2024, doi: 10.1080/02568543.2023.2192249.
- [4] S. Syarif, I. Nisaa, and V. Fitriani, "Exploring the strategies and environmental factors that foster curiosity in early childhood education," *Educia Journal*, vol. 2, no. 1, pp. 1–10, 2025, doi: 10.71435/610400.
- [5] E. Swider-Cios, A. Vermeij, and M. M. Sitskoorn, "Young children and screen-based media: the impact on cognitive and socioemotional development and the importance of parental mediation," *Cognitive Development*, vol. 66, p. 101319, Apr. 2023, doi: 10.1016/j.cogdev.2023.101319.
- [6] R. Segundo-Marcos, A. M. Carrillo, V. L. Fernández, and M. T. D. González, "Development of executive functions in late childhood and the mediating role of cooperative learning: A longitudinal study," *Cognitive Development*, vol. 63, p. 101219, Jul. 2022, doi: 10.1016/j.cogdev.2022.101219.
- [7] C. Mucignat-Caretta and G. Soravia, "Positive or negative environmental modulations on human brain development: the morpho-functional outcomes of music training or stress," *Frontiers in Neuroscience*, vol. 17, Nov. 2023, doi: 10.3389/fnins.2023.1266766.
- [8] K. Liskai-Peres et al., "Association between the use of mobile touchscreen devices and the quality of parent-child interaction in preschoolers," *Frontiers in Child and Adolescent Psychiatry*, vol. 3, Mar. 2024, doi: 10.3389/frcha.2024.1330243.
- [9] K. Bittner, "Engagement and immersion in digital play: supporting young children's digital wellbeing," *International Journal of Environmental Research and Public Health*, vol. 18, no. 19, p. 10179, Sep. 2021, doi: 10.3390/ijerph181910179.
- [10] J. S. Radesky et al., "Young children's use of smartphones and tablets," *Pediatrics*, vol. 146, no. 1, Jul. 2020, doi: 10.1542/peds.2019-3518.
- [11] J. Calleja and P. Camilleri, "Teachers' learning in extraordinary times: shifting to a digitally facilitated approach to lesson study," *International Journal for Lesson and Learning Studies*, vol. 10, no. 2, pp. 118–137, 2021, doi: 10.1108/IJLLS-09-2020-0058.
- [12] S. Varaksina, "Educational game development: how to make learning games for kids," themindstudios. Accessed: Dec. 20, 2023. [Online]. Available: <https://themindstudios.com/blog/educational-game-development/>
- [13] A. Daniel and I. A. Suleiman, "Enhancing pupil engagement and learning through augmented reality-based interactive phonetics education," *World Journal of Advanced Engineering Technology and Sciences*, vol. 9, no. 1, pp. 260–271, Jun. 2023, doi: 10.30574/wjaets.2023.9.1.0131.

- [14] B. K. Nath, "Pedagogical impact of teachers on reading preparedness and reading attainment among scheduled tribe children: Insights from Field observations," *Technolearn An International Journal of Educational Technology*, vol. 13, no. 1, Jun. 2023, doi: 10.30954/2231-4105.01.2023.8.
- [15] R. Paul, C. Norbury, and C. Gosse, "Language disorders from infancy through adolescence," in *Elsevier Health Sciences*, Elsevier, 2024, pp. 1–812. doi: 10.1016/C2015-0-04093-X.
- [16] S. S. Dutta, "Does screen time affect children's development?" [Online]. Available: <https://www.news-medical.net/health/Does-Screen-Time-Affect-Childrene28099s-Development.aspx>
- [17] Z. Y. Liu, Z. A. Shaikh, and F. Gazizova, "Using the concept of game-based learning in education," *International Journal of Emerging Technologies in Learning*, vol. 15, no. 14, pp. 53–64, Jul. 2020, doi: 10.3991/ijet.v15i14.14675.
- [18] J. C. Wang, C. Y. Hsieh, and S. H. Kung, "The impact of smartphone use on learning effectiveness: a case study of primary school students," *Education and Information Technologies*, vol. 28, no. 6, pp. 6287–6320, 2023, doi: 10.1007/s10639-022-11430-9.
- [19] F. Hamzah, A. H. Abdullah, and W. Ma, "Advancing education through technology integration, innovative pedagogies and emerging trends: a systematic literature review," *Journal of Advanced Research in Applied Sciences and Engineering Technology*, vol. 41, no. 1, pp. 44–63, Mar. 2024, doi: 10.37934/araset.41.1.4463.
- [20] M. Cascella, A. Cascella, F. Monaco, and M. N. Shariff, "Envisioning gamification in anesthesia, pain management, and critical care: basic principles, integration of artificial intelligence, and simulation strategies," *Journal of Anesthesia, Analgesia and Critical Care*, vol. 3, no. 1, p. 33, Sep. 2023, doi: 10.1186/s44158-023-00118-2.
- [21] C. E. L. L. Yin and I. Yang, "Augmented reality in read aloud: an english language interactive engagement," *Innovative Teaching and Learning Journal*, vol. 7, no. 2, pp. 51–59, Dec. 2023, doi: 10.11113/itlj.v7.134.
- [22] F. Carrión-Robles, V. Espinoza-Celi, and A. Vargas-Saritama, "The use of augmented reality through Assemblr Edu to Inspire writing in an Ecuadorian EFL distance program," *International Journal of Engineering Pedagogy*, vol. 13, no. 5, pp. 121–141, Jul. 2023, doi: 10.3991/ijep.v13i5.38049.
- [23] I. Fitrianto and A. Saif, "The role of virtual reality in enhancing experiential learning: a comparative study of traditional and immersive learning environments," *International Journal of Post Axial: Futuristic Teaching and Learning*, vol. 2, no. 2, pp. 97–110, 2024, doi: 10.59944/postaxial.v2i2.300.
- [24] N. M. Alzahrani, "Augmented reality: a systematic review of its benefits and challenges in e-learning contexts," *Applied Sciences (Switzerland)*, vol. 10, no. 16, p. 5660, Aug. 2020, doi: 10.3390/app10165660.
- [25] C. S. C. Dalim, M. S. Sunar, A. Dey, and M. Billingham, "Using augmented reality with speech input for non-native children's language learning," *International Journal of Human Computer Studies*, vol. 134, pp. 44–64, Feb. 2020, doi: 10.1016/j.ijhcs.2019.10.002.
- [26] N. K. Dennis, "Using AI-powered speech recognition technology to improve english pronunciation and speaking skills," *IAFOR Journal of Education*, vol. 12, no. 2, pp. 107–126, Aug. 2024, doi: 10.22492/jje.12.2.05.
- [27] L. Ping and N. Tao, "Innovative approaches to English pronunciation instruction in ESL contexts: integration of multi-sensor detection and advanced algorithmic feedback," *Frontiers in Psychology*, vol. 15, Jan. 2025, doi: 10.3389/fpsyg.2024.1484630.
- [28] J. Brooke, "SUS - A quick and dirty usability scale," *Usability Eval. Ind.*, vol. 189, pp. 189–194, 1995.
- [29] J. Brooke, "SUS: a retrospective," *Journal of usability studies*, vol. 8, no. 2, pp. 29–40, 2013.
- [30] J. An, L. P. Poly, and T. A. Holme, "Usability testing and the development of an augmented reality application for laboratory learning," *Journal of Chemical Education*, vol. 97, no. 1, pp. 97–105, Jan. 2020, doi: 10.1021/acs.jchemed.9b00453.
- [31] K. Setemen, L. J. E. Dewi, and I. K. Purnamawan, "PAON usability testing using system usability scale," *Journal of Physics: Conference Series*, vol. 1165, no. 1, p. 012009, Feb. 2019, doi: 10.1088/1742-6596/1165/1/012009.
- [32] Y. Thamilarasan, R. R. R. Ikram, M. Osman, L. Salahuddin, W. Y. W. Bujeri, and K. Kanchimalay, "Enhanced system usability scale using the software quality standard approach," *Engineering, Technology and Applied Science Research*, vol. 13, no. 5, pp. 11779–11784, Oct. 2023, doi: 10.48084/etasr.5971.

BIOGRAPHIES OF AUTHORS



Norizan Mat Diah    is an associate professor at the School of Computing Sciences, College of Computing, Informatics, and Mathematics, Universiti Teknologi MARA, Shah Alam, Selangor, Malaysia. She holds a Ph.D. in information sciences from the National University of Malaysia. Her research focuses are on gamification, game engines, real-time feedback (algorithms), and machine learning. She can be contacted at email: norizan289@uitm.edu.my.



Akmal Hakim Mohammad Mokhriz    received his bachelor's degree in computer science (Hons.) Multimedia Computing from Universiti Teknologi MARA (UiTM), Shah Alam, Malaysia. His research interests include computer vision, multimedia computing, and gaming technology. He can be contacted at email: hakim20109@gmail.com.



Norsafinar Rahim    holds a Ph.D. in Information Technology from Universiti Utara Malaysia and currently serves as a Senior Lecturer at the centre for Instructional Technology and Multimedia at Universiti Sains Malaysia (USM). Her research expertise spans user experience (UX) and user interface (UI) design, human-computer interaction, game-based learning, artificial intelligence, and instructional technology, with a focus on their applications in education. She can be reached at email: norsafinar@usm.my.



Suzana Ahmad    is a senior lecturer at Computing Sciences Department, Faculty Computer and Mathematical Sciences, Universiti Teknologi MARA, Shah Alam, Malaysia. She has graduated with Ph.D. in Information Technology and Quantitative Sciences at from Universiti Teknologi MARA. Her research focuses on Interactive Computing, Data Policies and Collaborative System. She can be contacted at email: suzana235@uitm.edu.my.



Elly Johana Johan    is a senior lecturer at the Department of Computer and Mathematical Sciences, Universiti Teknologi MARA, Permatang Pauh Campus, Pulau Pinang, Malaysia. She holds a Ph.D. in Computational Sciences from Universiti Kebangsaan Malaysia (UKM). Her research areas include persuasive technology and educational programming. She can be contacted at email: ellyjohana@uitm.edu.my.



Zainura Idrus    is a senior lecturer at the Computing Sciences Department, Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA. She has graduated with Ph.D. in Information Technology and Quantitative Sciences at from Universiti Teknologi MARA. Her research interests include data visualization, data analytics, collaborative systems, web technology, and machine learning applications. She can be contacted at email: zainura501@uitm.edu.my.