English School Mate

Dhammika de silva
Faculty of Computing
Sri Lanka Institute of Information Technology
Malabe,Sri Lanka
dhammika.d@sliit.lk

Anjali Gamage
Faculty of Computing
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
anjalie.g@sliit.lk

Sasitha Shehan
Faculty of Computing
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
sasithashehan.d@gmail.com

Bhashana Sankalpa
Faculty of Computing
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
bhashanasankalpa@gmail.com

Sapna Divyanjali
Faculty of Computing
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
sapnadivyanjalie221@gmail.com

Hashini Madushani
Faculty of computing
Sri Lanka Institute of information Technology
Malabe, Sri Lanka
hashininanayakkara83@gmail.com

***Abstract**— This research paper aims to develop an English language learning application for the students of Sri Lanka to overcome their difficulties of learning the English language in their secondary school education. Students can easily engage with English materials. According to that research in rural areas, students are not able to learn English because of many basic environmental issues. Here students will be able to improve their English spoken, writing abilities, and improve their brain improvements, and also there is an activity controller and prediction system to student's guide. Machine learning technologies, speech recognition, bitmap, and unity 3d are highly used here. This paper proposes a unified methodology to develop English knowledge and productively to improve the student's brain.*

Keywords - Machine Learning, speech recognition, bitmap, unity 3d, brain improvement

I. Introduction

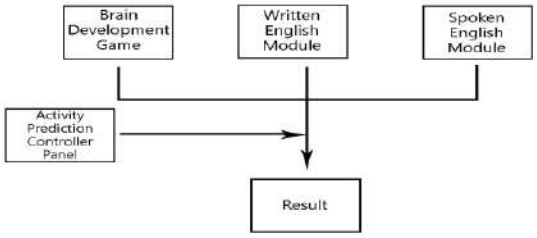
Nowadays knowing the English language increases your chances of getting a good job in a multinational company within your home country or for finding work abroad. It's also the language of international communication, the media, and the internet, so learning English is important for socializing and entertainment as well as work. So for that reason, English learning is so important.

Language is one of the mediums of expressing our ideas, feelings, and emotions. And if we think about language in the present world then English is one of the most used languages in the world and English is used as a second language in Sri Lanka. English is introduced here at the primary level and its inclusion continues till the tertiary level of education. Most of the students in rural areas are weak in the English language due to a lack of skilled and trained teachers who are familiar with the modern methods and approaches of teaching and lack of materials for teaching in the classroom. Primary level English curriculum implementation is essential to achieve the set English language competency in rural areas. Students in rural areas are performing poorly in English compared to their urban counterparts[9].

Methodology

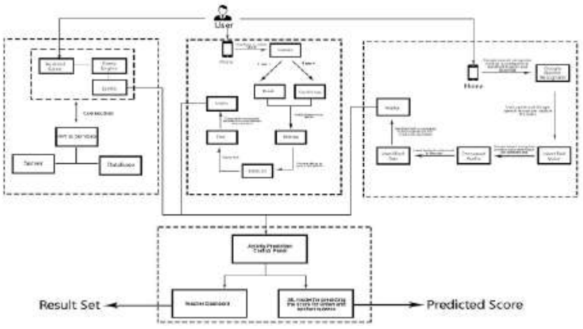
Researchers have broken down the methodology into 4 main components to cover up this research problem.

1. Brain Development Game
2. Written English Module
3. Spoken English Module
4. Activity Prediction Controller Panel



Main Implementation

The main implementation of the Smart Student system has 2 major aspects as the Authentication segment and the Backend and Database. The Authentication segment is implemented using Firebase Authentication as a service. This is composed of a Sign-in screen and a Sign-Up functionality. Further, a Navigation Drawer update option and a user profile picture functionality are added to make the Authentication process complete. The backend and the database are implemented as a REST API using Java Spring-Boot and MongoDB as the database. Moreover, CRUD endpoints for written and spoken questions are also implemented. All the analyses of the question results will be performed here.



Brain Development Game

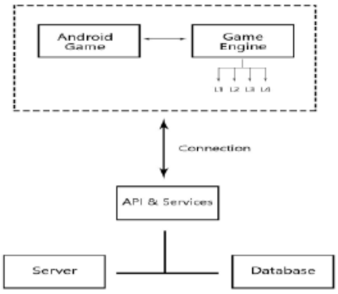
The brain development game is mainly focused on increasing the thinking and analyzing power of students. It is important to note that the Implementation was done using both Android Studio and Unity3D to test the performance of the game since the logical complexity is high at each level. When compared to Android Studio, Unity3D is easier to use if both the software are not familiar. Furthermore, regardless of whether you should know Android Studio and not Unity 3D, you would just profit by that in a specific way.

Since most games contain things like realistic resources, sound resources, an explicit treatment of client input, liveliness and so forth and so on, and a game motor like Unity 3D is extraordinarily evolved to be acceptable at taking care of those things, whereas increasingly "nonexclusive" improvement devices like Android Studio, Eclipse, Visual Studio, etc. All things considered, nonexclusive. Solidarity 3D additionally lets you run the game inside the game supervisor, empowering you to change the properties of your game resources legitimately and see the outcomes right away. You would need to grow a large portion of such prospects yourself should you make your game in an apparatus like Android Studio[1].

The coding portion of building up a game is best made in a device for coding, and Unity 3D in and without anyone else isn't a coding instrument. For that part, you need an extra tool like MonoDevelop (which you get packaged with Unity 3D) or Visual Studio, whereas the Unity 3D editorial manager is made for building the game scenes and taking care of the game resources.

It is exceptionally lightweight and asset the executives is done entirely well in this IDE. It boots up quickly when contrasted with other improvement IDEs. It's anything but a memory hungry IDE. Even beginners can work with this IDE very easily. Emulator begins extremely delayed in Android Studio, possibly that is where it needs improvement. It takes a great deal of memory space to introduce the IDE. Heavy application and will hinder PC execution[2].

The motor is exceptionally favored for its all-inclusive help to 27 stages. The application created and conveyed can be effortlessly shared between PC, web, and versatile stages. Moreover, the spry philosophy empowers quick prototyping and steady discharges, which thus accelerate game development. The content manager is given by IDE to compose the code, however once in a while, a particular code supervisor is likewise utilized by the designers to mitigate disarray. Moreover, the incorporated improvement supervisor support JavaScript and C# for scripting and offers eminent highlights that are perfect for the game development. The excellent sound and special visualizations are upheld by the motor that facilitates the game turn of events. The visuals are versatile on each screen and gadget with no twisting or bargain with the picture quality. It's an unquestionable requirement. The fledgling designers need the straightforward documentation that is given in detail by the Unity motor. The definite documentation incorporates the clarification of each little topic. The troubleshooting and tweaking are incredibly simpler with Unity game advancement since all the game factors are shown during interactivity, which thusly permits the designers to investigate the procedure at runtime. Not to state, yet the motor lingers behind from a graphical perspective. It doesn't offer a variety of apparatuses to make staggering illustrations instead of other game advancement engines. In Unity 5 motor, the inherent help for the PhysX material science motor has some presentation issues and comes up short on some significant functionalities which should be added to make the incredible game app. The designers need to have licenses for the best illustrations, organization, and execution upgrades. These licenses are costly to buy. Besides, the utilization of rendering, cradle support, stencil support, and truly more highlights scale up the advancement costs because of costly licenses. The code is steady in Unity rather than different motors and pressed with extraordinary engineering that improves the game application execution. Be that as it may, inaccessibility of the source code makes discovering, tending to, and fixing the exhibition issues difficult. The game created utilizing Unity motor devours more memory, which thus makes OOM blunders and investigating issues in the applications[3].



Written English Module

The Written English module is implemented based on several advanced logics. Initially, the product component has various types of exercises built within it covering most of the learning techniques including questions to underline, questions to mark true or false, and questions to fill in the blanks. Further, it is important to note that in some instances the questions are organized into lessons where a lesson contains questions of all the types.

The Written English module follows a unique method to load the questions into the front end which is known as fetched HTTP calls. The Hypertext Transfer Protocol (HTTP) is designed to enable communications between clients and servers. HTTP works as a request-response protocol between a client and a server. Here the questions are stored on a remote database and fetched through HTTP calls [10]. The Fetch API is a modern replacement for XHR; it was introduced in browsers recently to make asynchronous HTTP requests easier to do in JavaScript, both for developers and other APIs that build on top of Fetch[11].

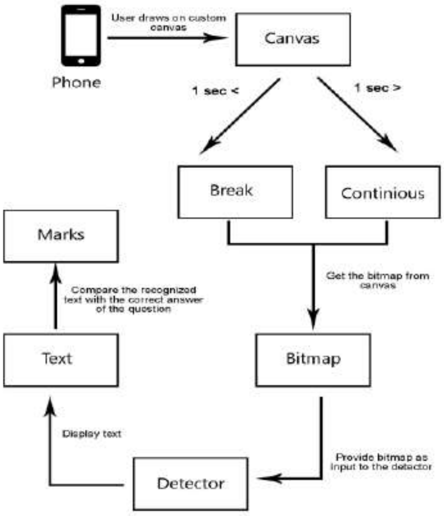
The Written module also allows the users to see their progress, rank, scores, and other statistics for written quizzes which will provide the user as well as the mentor a productive environment to work with. The user-friendly interfaces further make the system more usable to children making the product more interesting. The answer validation process of the Written English module is implemented using Firebase ML vision and a custom drawing canvas. Firebase ML Kit is a mobile SDK that makes it easier for mobile developers to include machine learning capabilities in their applications. ML Kit is a wrapper over the complexities of including and using machine learning capabilities in your mobile app [7][8]. The complete mechanism of the Written English module can be divided into 6 main phases.

- The user draws on custom canvas
- If the user does not draw for 1s recognize it as break

- Get the bitmap from canvas
- Provide bitmap as input to the detector
- Display text
- Compare the recognized text with the correct answer of the question

The diagram below illustrates the system overview of the Written Module.

Spoken English Module


Phone
User draws on custom canvas
Canvas
1 sec <
1 sec >
Break
Continious
Get the bitmap from canvas
Bitmap
Provide bitmap as input to the detector
Detector
Display text
Text
Compare the recognized text with the correct answer of the question
Marks

The logic of the Spoken English module is equal to the Written English module. Within the Spoken English component, the users are allowed to attempt the build-in exercises provided under different categories as levels. Several versatile techniques are used throughout the exercise development process to improve the effectiveness of the product including conversations, dialogs, model speeches, poems, and pronunciations. Further, it is important to note that in some instances the questions are organized into lessons where a lesson contains questions of all the types[5].

The Spoken English module follows a unique method to load the questions into the front end which is known as fetched HTTP calls. The Hypertext Transfer Protocol (HTTP) is designed to enable communications between clients and servers. HTTP works as a request-response protocol between a client and a server. Here the questions are stored on a remote database and fetched through HTTP calls[10]. The Fetch API is a modern replacement for XHR; it was introduced in browsers recently to make asynchronous HTTP requests easier to do in JavaScript, both for developers and other APIs that build on top of Fetch[11].

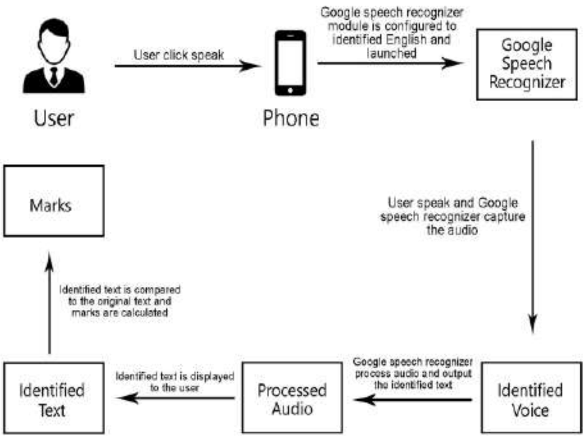
The next most important feature implemented is the progress status and the score related to the component. Users will be able to see their progress, rank, scores, and other statistics for spoken quizzes. The score calculation process is based on the identification of the answers based on the given voice output to the product. This is Implemented using Google speech recognizer. Google has a great Speech Recognition API. This API converts spoken text into written text, briefly Speech to Text[5]. You can simply speak in a microphone and Google API will translate this into written text. The API has excellent results for the English language. The complete mechanism of the Spoken English module can be divided into 6 main phases[12]. This can also be referred to as speech recognition and answer validation process in technical terms.

- User click speak button
- Google speech recognizer module is configured to identified English and launched
- User speak and Google speech recognizer capture the audio
- Google speech recognizer process audio and output the identified text
- Identified text is displayed to the user
- Identified text is compared to the original text and marks are calculated

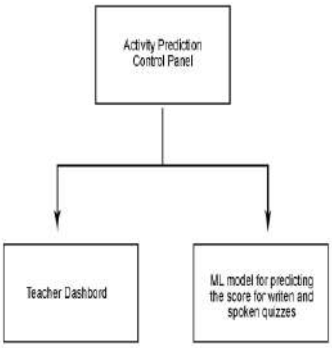
Speech-to-Text has three main methods to perform speech recognition. These are listed below:

- **Synchronous Recognition** (REST and gRPC) sends audio data to the Speech-to-Text API, performs recognition on that data, and returns results after all audio has been processed. Synchronous recognition requests are limited to audio data of 1 minute or less in duration.
- **Asynchronous Recognition** (REST and gRPC) sends audio data to the Speech-to-Text API and initiates a *Long-Running Operation*. Using this operation, you can periodically poll for recognition results. Use asynchronous requests for audio data of any duration up to 480 minutes.
- **Streaming Recognition** (gRPC only) performs recognition on audio data provided within a gRPC bi-directional stream. Streaming requests are designed for real-time recognition purposes, such as capturing live audio from a microphone. Streaming recognition provides interim results while audio is being captured, allowing results to appear, for example, while a user is still speaking[6].

The diagram below illustrates speech recognition and answer the validation process for the Spoken English module.


User
User click speak
Phone
Google speech recognizer module is configured to identified English and launched
Google Speech Recognizer
User speak and Google speech recognizer capture the audio
Identified Voice
Google speech recognizer process audio and output the identified text
Processed Audio
Identified text is displayed to the user
Identified Text
Identified text is compared to the original text and marks are calculated
Marks

Activity Prediction Controller Panel


Activity Prediction Control Panel
Teacher Dashbord
ML model for predicting the score for writen and spoken quizzes

The Activity Prediction Controller Panel is composed of 2 major segments as shown in the figure above. Unlike the other components, this component is implemented as a web application to increase the efficiency and speed of processing.

Teacher Dashboard

The teacher dashboard contains information on the overall performance of a set of students assigned to the teacher. The graphical distributions shown under this component provide the mentor with a piece of analyzed information set about the students who are registered to the system. Several graphical techniques such as bar charts, pie charts, and tables are used in this component reducing the time required by the guiding party to analyze the results of each student. This is one of the unique features of this product which makes the Smart Student System more valuable.

ML model for predicting the score for written and spoken quizzes

In the modern world teachers or parents do not have enough time to spend on individual students most of the time. This machine learning model for score prediction is developed to overcome this problem. In the module, the score for a particular student is predicted based on several factors including hometown, school, family background, previous marks, and individual interests. The data for this is collected through Google Forms and the data is cleaned several times to create the dataset. Data preprocessing and developing the model is using Jupyter notebook as the IDE. Sklearn and pandas libraries are used for models and utilities needed for data preprocessing, training models, model evaluation.

Results and Discussion

The vocabulary skill and brain improvement game section are entirely based on unity 3D. We implemented four levels to develop the game. The game gave me an extra interest in English with huge knowledge. The 1st level is Word categorization. Here we develop the vocabulary level of the students. The 2nd level is Counting the cubes and brain improvement is the main target of this level. The 3rd one is the words match puzzle. The vocabulary level is highly increased in that level. The 4th one is the solve equations level so this is also a brain improvement one. In here totally successful the main objective of brain improvement.

Writing session implemented using Firebase ML vision and a custom drawing canvas and questions stored on a remote database and fetched through HTTP calls. whole module lessons broke into three question types. Those are the true-false questions, underline, and fill in the blank questions. So users can see their progress, rank, scores, and other statistics for written quizzes. The handwriting recognition and answer validation flow are users draw on the custom canvas, If the user does not draw for 1s recognize it as break, then get the bitmap from canvas, provide bitmap as input to the detector, display text and finally compare the recognized text with the correct answer of the question. So in this case the section covered written English exercises and correction models. Here it has a user-friendly interface and cost-effectiveness. Those are the several objectives achieved.

Google speech recognizer used to implement the spoken English section and it categorized into four-lesson sections. English conversations (dialogues), matching with readings (pronunciations), poems and model speeches are those four categories. Questions stored on a remote database and fetched through HTTP calls. Users can see their progress, rank, scores, and other statistics for spoken quizzes. The casual flow of this section is user click speak button, google speech recognizer module is configured to identified English and launched, user, speak and Google speech recognizer capture the audio, google speech recognizer process audio and output the identified text, the identified text is displayed to the user and finally identified text is compared to the original text and marks are calculated. Scores and question numbers displayed per question. So the student's guide can analyze the student's improvement question by question. Finally achieved the main objective of the spoken English Exercises and Correction model implementation.

Using machine learning techniques implemented activity results prediction and controller panel.it categorized teacher dashboard and ML model for predicting the result/activity for written and spoken quizzes . in the Teacher Dashboard Will contain information on the overall performance of a set of students assigned to the teacher. We collect Data through Google Forms get near 1000 data sets. Then Data preprocessing and developing a model using Jupyter notebook as the IDE. Sklearn and pandas libraries will be used for models and utilities needed for data preprocessing, training models, model evaluation. Here we get all the basic information and the results, the teacher can predict the students' states and the teacher can be able to know what activities must give to their students.all of this achieved the main objectives of this component.

Conclusion and Future Work

The overall objective of this research is to develop an integrated hybrid solution to English language learning application for the students of Sri Lanka to overcome their difficulties of learning the English language in their secondary school education. This also provides a solution for the mentor to analyze individual performances of students individually along with a prediction function to check how many marks a student can obtain before doing the quiz. With this, the mentor can get an idea about the status of the student.

As future work, this system can be upgraded to test students with more advanced learning techniques such as online learning, peer to peer interaction learning and also can add a brain development game with more advanced interactive features. Further, it is possible to add more languages, subjects and also make this available for more grades. However, as a learning management system, Smart Student can be considered as a modern and advanced solution which has addressed several unaddressed aspects of learning.

References

[1] "Is it easier to develop Android games in Unity3D or in Android Studio? - Quora", *Quora.com*, 2020. [Online]. Available: https://www.quora.com/Is-it-easier-to-develop-Android-games-in-Unity3D-or-in-Android-Studio. [Accessed: 20- Jul- 2020].

[2] A. Studio, "Pros and Cons of Android Studio 2020", *TrustRadius*, 2020. [Online]. Available: https://www.trustradius.com/products/android-studio/reviews?qs=pros-and-cons. [Accessed: 20- Jul- 2020].

[3] "5 Rarely Known Advantages And Disadvantages Of Unity Game Development | Potenza Global Solutions", *Potenzaglobalsolutions.com*, 2020. [Online]. Available: https://www.potenzaglobalsolutions.com/blogs/5-rarely-known-advantages-and-disadvantages-of-unity-game-development. [Accessed: 20- Jul- 2020].

[4]Google Cloud. 2020. *Speech-To-Text Basics | Cloud Speech-To-Text Documentation*. [online] Available at: <https://cloud.google.com/speech-to-text/docs/basics> [Accessed 19 July 2020].

[5]Google Cloud. 2020. *Speech-To-Text: Automatic Speech Recognition | Cloud Speech-To-Text*. [online] Available at: <https://cloud.google.com/speech-to-text> [Accessed 19 July 2020].

[6]Hincks, R., 2001. Using speech recognition to evaluate skills in spoken English.

[7]dzone.com. 2020. *Machine Learning In Android Using Firebase ML Kit - Dzone AI*. [online] Available at: <https://dzone.com/articles/machine-learning-in-android-using-firebase-ml-kit> [Accessed 19 July 2020].

[8]Hathibelagal, A., 2018. *Getting Started With Firebase ML Kit For Android*. [online] Code Envato Tuts+. Available at: <https://code.tutsplus.com/tutorials/getting-started-with-firebase-ml-kit-for-android--cms-31305> [Accessed 19 July 2020].

[9] Englishfor2day. 2020. *A Diologue On The Importance Of Learning English | Englishfor2day*. [online] Available at: <http://www.englishfor2day.com/article/dialogue/187> [Accessed 21 July 2020].

[10]W3schools.com. 2020. *HTTP Methods GET Vs POST*. [online] Available at: <https://www.w3schools.com/tags/ref_httpmethods.asp> [Accessed 21 July 2020].

[11] MDN Web Docs. 2020. *Fetching Data From The Server*. [online] Available at: <https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Client-side_web_APIs/Fetching_data> [Accessed 26 June 2020].

[12]Ono, Y., 2015. Construction of a Voice-based Asynchronous Communication System Utilizing Speech Recognition and Its Potential for EFL Learners' Speaking Ability: A Pilot Study. In: *15th International Conference on Advanced Learning Technologies*.