SRI LANKA INSTITUTE OF INFORMATION TECHNOLOGY

B.Sc. (Hons) Degree in Information Technology

Department of Information Technology

Final Group Thesis – Group

Mobile Base Sinhala Book Reader for Visually Impaired Individuals

by

Group ID: 2023-198

Supervisor: Prof. Koliya Pulasinghe

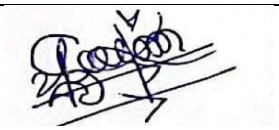
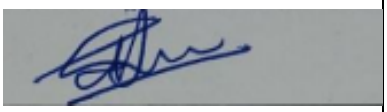
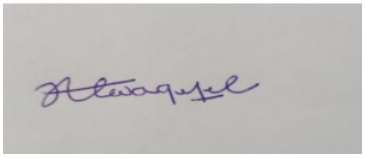
Co-Supervisor: Ms. Poorna Panduwawala

Sri Lanka

October 2023

DECLARATION

To the best of our knowledge and belief, this proposal does not contain any previously published or written material by another person, except where the acknowledgement is made in the text. We hereby declare that this is our own work and that no material previously submitted for a degree or diploma in any other university or Institute of higher learning has been incorporated without acknowledgement.

Name	Student ID	Signature
Jayathunga T.M.	IT20146238	
Semini J.P.D.L.	IT20241346	
Godakanda P.G.S.	IT20129712	
Bhagya H.D.M.	IT20254520	

ABSTRACT

In an era of rapid technological progress, education has evolved beyond its conventional boundaries, driven by the integration of Artificial Intelligence (AI) across diverse fields. However, individuals with visual impairments encounter formidable barriers in accessing educational materials, particularly in languages other than English. To tackle this issue head-on, this research project endeavors to develop a holistic Android-based mobile application exclusively catered to the visually impaired community in Sri Lanka. The application's prime focus lies in facilitating access to educational resources in the Sinhala language, thereby empowering visually impaired individuals in their pursuit of knowledge.

The project aims to provide a replacement for the obsolete readers for printed Sinhala story books. This was created especially for blind children. The accessibility of printed books and other publications in the Sinhala language for blind people in Lanka has not expanded. The main objective of this project is to enable them to have easy access to books and documents. Here, a visually impaired person navigates the app as needed by voice navigation, and the application gives the necessary commands to the visually impaired person. Here the mobile camera performs the necessary commands to accurately capture all the characters on a page of a Sinhala book. After that, optical character recognition separates and identifies the characters from the image with the corresponding characters. The Festival Synthesizing Framework's Text-to-Speech feature then allows the corresponding visually impaired individual to hear clearly. Here he can increase or decrease the speed of reading the book according to the commands. We intended to develop this Mobile Base Sinhala Book Reader to make it easier for visually impaired people by using audio recommendations and Sinhala voice notifications. We anticipate that this would boost the visually challenged community in Sri Lanka's delight in reading Sinhala novels.

OCR technology can also be used to convert printed books, articles, and other documents into accessible digital formats such as e-books or audio books, which can be easily accessed and navigated using assistive technology. The optical character recognition (OCR) component is a key feature of the mobile-based Sinhala book reader for visually impaired individuals. This component allows the user to take a picture of a printed, typed text and convert it into machine- encoded text that can be read aloud by the app. OCR is the process of identifying and classifying optical patterns in a digital image and translating them into machine-readable text. The OCR component of the app uses advanced algorithms and image processing techniques to extract text from an image, such as a page from a book or a newspaper. The purpose of this research project component is to develop a system that enables blind people to navigate the book reading app using Sinhala voice commands and distant object detection. The system uses machine learning to instruct the user to avoid objects and provides the distance to the object through a voice command. This project aims to improve the independence and mobility of blind people in a digital environment, enhancing their experience in using technology. Navigating without sight is a daunting task for the visually impaired. Unlike those with good vision, blind individuals cannot detect and

avoid obstacles, making it necessary for them to rely on guidance to navigate safely. While the white cane is a commonly used tool for detecting and avoiding obstacles, its limited reachability makes it inadequate for identifying all potential threats. To address this issue and enable safer and more independent navigation for the blind, this study proposes a novel approach based on deep learning for obstacle detection. The study involved developing a prototype system that utilizes deep neural networks (DNNs) for obstacle detection and distance estimation. DNNs were selected due to their high accuracy and real-time performance. Obstacle detection data was collected using a simulation environment, and the resulting model was used to estimate the distance of obstacles. The feedback generated by the combination of obstacle detection and distance estimation was communicated to the user through audio cues. Overall, this novel approach based on deep learning offers a promising solution to the challenges of blind navigation, enabling safer and more independent movement for visually impaired individuals. Significant obstacles must be overcome for the blind to obtain information and carry out daily tasks. The fast growth of mobile technology has made it possible to construct mobile applications to solve these issues. The Blind App is a smartphone application made to help those who are blind or visually impaired find their way about and access information. The creation and assessment of the Blind App, which seeks to improve the independence and quality of life of people who are blind, are presented in this paper. The Blind App uses a variety of technologies, including as text-to-speech, voice recognition, and Optical character recognition to provide users an easy-to-use interface. The program has several capabilities to help visually impaired users carry out daily tasks on their own, including object recognition, and voice commands. The usability and accessibility of an app's user interface, the efficacy of its audio cues and voice-overs, and the app's effects on the daily lives of those with visual impairments are just a few of the topics that might be the subject of research on blind apps. Researchers may learn a lot about how technology can be used to increase accessibility and improve the quality of life for persons with disabilities by examining how blind apps are used.

Key Words: Text to Speech Sinhala, Natural Language Processing, Visually Impaired Individuals, Festival Synthesizing Framework, Speech Synthesis

ACKNOWLEDGEMENT

This significant research project received careful evaluation, reinforced by the profound insights offered by our respected research supervisor, Prof. Koliya Pulasinghe, and co-supervisor, Ms. Poorna Panduwawela, whose advice proved priceless. We also thank the Natural Language Processing (NLP) domain evaluation panel for their thorough review. We would like to take this occasion to extend our sincere gratitude to all parties involved for their unflagging perseverance, steadfast commitment, and teamwork, which resulted in the victorious completion of this research project.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

OCR – Optical Character Recognition

TTS – Text-to-Speech

NLP – Natural Language Processing

API – Application Programming Interface

ML – Machine Learning

CNN – Convolutional Neural Networks

WCAG – Web Content Accessibility Guidelines

UI – User Interface

AI – Artificial intelligence

ANN – Artificial Neural Network

MLP – Multi-layer perceptron

2DFFT– 2D Fast Furrier Transform

WHO - World Health Organization

CART – Classification and Regression Tree

DSP – Digital Signal Processing

CSS – Cascading Style Sheets

GUI – Graphical User Interface

HCI – Human-Computer Interaction

NLU – Natural Language Understanding

OpenCV – Open-Source Computer Vision Library

APK – Android Package

MAP – Mean Average Precision

SLAM –Simultaneous Localization and Mapping

SSD –Single Shot MultiBox Detector

YOLO – You Only Look Once

AP – Average Precision

Chapter 01

1. Introduction

Mobile apps have seen a surge in popularity, with more and more mobile phone users relying on them for their day-to-day needs. However, despite the widespread availability of developer tools and public demand driving expansion across all aspects of human life, there remains a significant gap in the market for differently abled people. While most voice recognition and other software focuses on individuals with the ability to see, there is a lack of accessibility for those with disabilities who wish to carry out their daily tasks just like any other person. Many apps are designed with graphics-oriented activities for children, and the majority are tailored towards the needs of able-bodied individuals, leaving those with disabilities at a disadvantage. However, according to the United Nations, differently abled individuals are entitled to the same human rights and freedoms as everyone else in society, without discrimination of any kind. They have the right to access things and places and live independently, without exception or distinction. Currently, most of the apps available on mobile phones are in English, leaving a significant portion of the population in countries such as Sri Lanka unable to access them due to language barriers. The inability of visually impaired people to accurately recognize physical items, their precise location, and their precise size puts them in difficult situations daily. The bulk of the solutions proposed in this situation include speech recognition as one of the more effective methods of item identification as the community of visually impaired persons can comprehend audio with ease. Portable solutions are frequently made in computing technology to answer the problem, which is unquestionably an advancement. In this case, it has provided motivation for the development of more mobile applications that efficiently assist users who are blind or visually impaired. An alternative viewpoint or method to incorporate into the answer through clever software development that uses computer vision as an eye for this group and makes life easier for them in day-to-day duties would be the aiding approach.

In a world increasingly driven by technology and innovation, it is our moral duty to ensure that every child, regardless of their abilities, has access to the magical world of literature. In the beautiful island nation of Sri Lanka, this commitment to inclusivity extends to the visually impaired children who, despite their challenges, are just as deserving of the joy and wonder that reading can bring. Enter "Audio Sight " – a groundbreaking online storybook reader app, thoughtfully designed to open the doors of imagination and knowledge to visually impaired children across Sri Lanka. An innovative online storybook reader program called "Audi Sight" was carefully created to empower young Sri Lankans who are blind or visually impaired. With a simple and user-friendly interface that allows even the youngest users to autonomously explore their beloved stories, this digital refuge serves a broad age spectrum, from the youngest of toddlers to the blossoming adolescents. An enormous collection of varied audiobooks that have been painstakingly chosen to feature tales from a wide range of countries, genres, and languages make up the core of this application. "Audio Sight" cultivates an awareness of the

various viewpoints that stories from around the world give in addition to igniting a love of reading.

But "Audio Sight " is here to change that narrative. This revolutionary app is not just another tool; it's a beacon of hope and an embodiment of the belief that every child, no matter their visual abilities, should have the same opportunities to learn, grow, and be inspired by the magic of stories.

"Audio Sight" is a ground-breaking online storybook reader app created to empower visually challenged children in Sri Lanka. It is appropriate for children of all ages, from babies to teenagers, and has a user-friendly layout that allows even the smallest users to actively explore their favorite stories. The app features a large library of diverse audio books that have been handpicked to contain stories from various countries, genres, and languages, encouraging a love of reading and an awareness for different points of view. Enter "Audio Sight" goes beyond passive listening with its intuitive and interactive design, engaging youngsters with touch-screen motions and audio feedback to provide an immersive reading experience. Personalization is central. of "Audio Sight ", as the app adapts to each child's reading level and preferences, offering customized recommendations and tracking progress while providing built-in educational tools. Beyond being an app, "Audio Sight " is a supportive community of families, educators, and volunteers dedicated to nurturing a love for reading among visually impaired children through support forums, book clubs, and resource sharing. Additionally, recognizing the challenges of constant internet access, "Audio Sight " allows users to download books for offline reading, ensuring that the magic of storytelling is always within reach.

This mobile application mainly uses optical character recognition (OCR) technology and voice navigation incorporating text-to-speech features of the event synthesis framework. Utilizing the capabilities of the mobile camera, the application accurately captures the characters present on a page of a Sinhala book and distinguishes it using OCR technology. This enables visually impaired people to capture text from physical documents and convert it into accessible digital formats. The extracted text is then made clearly audible to the visually impaired user via text-to-speech. In addition, while navigating the app, the visually impaired person is provided with Voice Navigation support, which is necessary to give him the instructions and other commands he needs and to identify the objects in the surrounding room and thereby guide the visually impaired user. Image recognition and description algorithms are used to clearly describe the pictures in Sinhala to the visually impaired person while reading the story books of visually impaired children through the mobile phone application. This helps visually impaired children to understand the visual content of the picture and further improve their reading skills by bringing them closer to books. For those with visual impairments, our platform offers features that allow users to adjust the reading speed and choose between a male or female voice. Additionally, users can use commands to read each page and move on to the next page when finished. Overall, this software aims to improve the reading experience and skills of visually impaired individuals in Sri Lanka.

Additionally, since many Sri Lankan children, including those with visual impairments, talk Sinhala as their basic language of communication, the creation of Sinhala blind apps can

guarantee that they have access to educational resources and materials that are catered to their linguistic requirements. This can support language development, which is important for learning, communication, and socialization. An innovation in education is the Sinhala book reader app for blind kids. The app is created with a user-friendly layout that is simple to use, making it accessible to kids who are blind to different degrees. The software has several capabilities, including optical character recognition, picture recognition, text-to-speech conversion, and voice commands. These features give kids easy and intuitive access to educational information, giving them the same opportunities to study and develop as their sighted peers. The voice command technology of the Sinhala book reader app for blind kids is another ground-breaking feature. Sinhala voice commands for a blind app can also be programmed to recognize specific commands related to reading and accessing books.

The Sinhala app for blind kids includes an amazing feature called picture detection that is meant to improve the program's usability and functionality. The app can now recognize and describe images to youngsters who are visually impaired thanks to this function, which makes use of cutting-edge image recognition technology. Children who are blind encounter several difficulties in their daily lives, particularly when trying to access visual content. Because traditional printed materials like books and magazines contain visuals and graphics that are challenging for blind children to understand, these resources are frequently inaccessible to them. By enabling the app to "see" and describe photos to the youngster in a way that is straightforward and clear, picture detection technology offers a solution to this issue. The app's text-to-speech conversion feature is a game-changer for visually impaired children. The text-to-speech option of the Sinhala reader app for blind students is a useful feature that enables the program to read out content in Sinhala to visually impaired pupils. This function gives students an equal chance to access written materials and take part in various reading and writing-related learning activities. For text-to-speech technology to function, spoken language must be translated from written text utilizing sophisticated algorithms and voice synthesis software. This technique has been adapted to read literature in Sinhala, the native language used in Sri Lanka, in the Sinhala Reader app for blind students. The Sinhala book reader app for blind kids is also quite adaptable, enabling parents and teachers to meet the individual requirements of each child. Users of the app can modify the app's pace, voice, display, and other features using a variety of settings and options. This degree of personalization makes sure that any child can use the app, regardless of their needs.

In summary, the research plan focuses on developing an OCR-based solution to empower visually impaired or blind children in Sri Lanka who speak Sinhala by enabling them to access and enjoy storybooks and educational materials. It acknowledges the critical role of OCR technology in achieving this goal and emphasizes the importance of addressing the unique needs of this community.

1.2. Main Objectives

The core objective of this study is to pioneer the development of a speech synthesis system uniquely tailored to cater to the reading needs of visually impaired individuals, primarily focused on Sinhala literature. This innovative system seamlessly combines advanced technologies like Festival and speech tools to facilitate the reading process. It leverages Optical Character Recognition (OCR) to convert physical Sinhala books into a digital format, enabling a crucial transformation. Subsequently, a Text-to-Speech (TTS) synthesizer takes center stage, articulating the written Sinhala text into spoken words, rendering literature accessible and immersive for visually impaired users. Beyond this, the system offers customizable settings, enhancing personalization and comfort in the reading experience. Furthermore, it incorporates audible guidance features, not only simplifying navigation within the application but also providing valuable location information within the book.

In essence, this pioneering speech synthesis system emerges as a vital tool, breaking down accessibility barriers and enriching the world of Sinhala literature for visually impaired individuals, fostering learning, personal growth, and cultural engagement within the Sinhala-speaking community. The study's key goal is to develop a speech synthesis system for visually impaired individuals using festivals and speech tools to read Sinhala books. A Sinhala book reader for the visually impaired is a software program designed to make reading accessible for individuals with visual impairments. It combines various technologies to provide a seamless reading experience. The device utilizes optical character recognition (OCR) technology to convert the text from a physical book into a digital format. Then, a text-to-speech synthesizer reads the text out loud in the Sinhala language, making it easier for visually impaired users to follow along. to provide visually impaired individuals with an accessible and convenient way to read Sinhala literature and to enhance their access to literature. The app aims to achieve this by using OCR and TTS technology and providing visually impaired individuals with customizable settings to enhance their reading experience. A Sinhala book reader for the visually impaired is a software program designed to make reading accessible for individuals with visual impairments. It combines various technologies to provide a seamless reading experience. The device utilizes optical character recognition (OCR) technology to convert the text from a physical book into a digital format. Then, a text-to-speech synthesizer reads the text out loud in the Sinhala language, making it easier for visually impaired users to follow along. In addition to the text-to-speech synthesizer, the device also includes audible guidance to help navigate the app and identify the distance to the book being read. This makes it easier for visually impaired users to find their place in the book and keep track of their progress.

A Sinhala book reader for the visually impaired is a unique software tool that has been painstakingly designed to empower those with visual impairments by opening the world of reading and information to them. This complete solution cleverly combines a plethora of cutting-edge technology, resulting in a seamless and user-friendly reading experience that surpasses the constraints of visual impairment. In Sri Lanka, where most of the population speaks Sinhala as their first language, the development of mobile book readers capable of

reading Sinhala texts for the blind or visually impaired is critical. This technology is a revolutionary force, allowing these people to access textual material with unprecedented ease and freedom, thereby improving their overall quality of life. The rising popularity of mobile book readers that use Text-to-Speech (TTS) technology emphasizes its importance. TTS technology enables persons who are blind or visually handicapped to receive information more easily through their sense of hearing by translating written text into spoken speech. Our invention is a critical step toward closing the accessibility gap and promoting diversity in our community. The application we hope to create has two key components: a Text-to-Speech (TTS) synthesizer and powerful Optical Character Recognition (OCR) technology created particularly for detecting common Sinhala characters [10]. This revolutionary system uses OCR to smoothly convert printed text from physical books to digital format. The TTS synthesizer then takes control, converting the text into clear and genuine Sinhala voice. This dynamic mix means that visually impaired people may easily follow and grasp the text, boosting their access to literature and information greatly. In addition to the text-to-speech synthesizer, our gadget includes a very useful feature: auditory guidance. This feature has two functions: it supports users in properly navigating the app and it offers real-time feedback on the distance to the book being read. By providing this multidimensional assistance, our gadget enables visually impaired users to not only easily find their place within the book, but also to keep a clear feeling of their reading progress. This deliberate integration improves the overall user experience while also encouraging greater freedom in the quest of literary inquiry.

The main objective of creating a Sinhala book reader for blind students is to provide them with a tool that can help them access and read Sinhala language materials independently. The main aim of creating a Sinhala book reader for blind students is to provide them with a tool that enables them to access and read Sinhala language materials independently. Blind students often face difficulty in accessing printed materials, which can affect their academic progress. By developing a book reader that is tailored for the blind and capable of reading Sinhala language materials out loud, blind students can increase their independence and access educational materials with ease.

A Sinhala book reader for the visually impaired is a software program designed to make reading accessible for individuals with visual impairments. It combines various technologies to provide a seamless reading experience. The device utilizes optical character recognition (OCR) technology to convert the text from a physical book into a digital format. Then, a text-to-speech synthesizer reads the text out loud in the Sinhala language, making it easier for visually impaired users to follow along. In addition to the text-to-speech synthesizer, the device also includes audible guidance to help navigate the app and identify the distance to the book being read. This makes it easier for visually impaired users to find their place in the book and keep track of their progress. The device also allows users to adjust the reading speed, volume, and pitch of the audio to their liking. Another useful feature of this Sinhala book reader is the ability to record audio pitch, which is especially helpful for users who may have difficulty reading in a consistent tone. This feature helps users to improve their reading skills and become more confident in their abilities.

1.3. Sub Objectives

The objectives of this component are as follows: To develop a system that enables blind people to navigate an app using voice commands. To develop machine learning algorithms that can detect objects in real time and calculate distances accurately. To provide users with real-time feedback about obstacles in their path and instructions on how to avoid them. To develop a user interface that is easy to use and understand for blind users. Enhancing accessibility - The objective of enhancing accessibility for blind users in image detection is to make images accessible to people with visual impairments. This is achieved using image recognition technologies that can provide audio descriptions or alternative text descriptions of the visual content of an image. Improving quality of life - The objective of improving the quality of life for blind users through image detection is to provide them with access to visual information. This can greatly enhance their daily lives by giving them a better understanding of their surroundings and increasing their independence. The use of image recognition technologies can help blind users gain more information and access to various environments. This objective can be achieved using specialized software and tools that can help to detect and describe images, making them accessible to visually impaired users. Increased independence - The objective of using image detection to increase the independence of blind users is to provide them with access to visual information. This can help them to better understand their environment and perform tasks that would otherwise require the assistance of a sighted person. By using image recognition technologies, blind users can gain greater independence and self-sufficiency in their daily lives. Access to visual information can help to increase their confidence and enable them to achieve a higher degree of independence. Enhanced entertainment - The objective of using image detection to enhance entertainment for blind users is to provide them with access to visual media. By using image recognition technologies, blind users can receive audio descriptions of the visual content in real time, allowing them to fully experience the entertainment and engage with the same media as sighted individuals. This objective aims to increase the entertainment options for blind users and provide them with a greater sense of inclusion in society. Enhanced education - The objective of using image detection to enhance education for blind users is to provide them with equal access to visual content in educational materials. This can be achieved by using image recognition technologies to convert visual content into audio descriptions. By doing so, blind students can fully comprehend and engage with the material, which can improve their overall educational experience.

This objective aims to provide blind students with the same opportunities. To accurately create computer-readable text from collected photos using the right image preprocessing. to put in place a Sinhala text-to-speech synthesizer that can translate text into speech. to evaluate the Sinhala language system's effectiveness. Create a user-friendly, accessible interface that is simple to navigate and suitable for those who are blind. Assuring that visually impaired people can use the app on several mobile devices requires designing the app to be flexible to diverse device sizes and screen resolutions. Create an OCR engine that can recognize Sinhala letters and convert scanned pictures of Sinhala books into editable, searchable text. By testing it with

a blind focus group and collecting feedback, the program might be enhanced to provide a seamless and accessible reading experience for people who are visually impaired. This complete project combines powerful OCR and TTS technologies with user-centric design principles to provide a platform that is accessible and inclusive for visually impaired individuals. By addressing the technical intricacies of Sinhala character recognition, enhancing speech synthesis, and actively involving the end users in the testing and improvement process, this initiative endeavors to break down accessibility barriers and provide a seamless reading experience for those who are visually impaired, ultimately promoting literacy, learning, and cultural engagement within the Sinhala- speaking community.

Four main sub-objectives of our research are: -

1. Optical Character Recognition (OCR)
2. Text-to-Speech (TTS) Synthesizer
3. Object Detection & Voice Navigation
4. Image Detection

Recognition (OCR), which seeks to develop efficient algorithms capable of recognizing and converting printed or handwritten text into machine-readable data. Complementing this, the second sub-objective involves Text-to-Speech (TTS) Synthesizer development, aiming to convert textual information into natural, human-like speech, facilitating accessibility and communication.

Moving beyond the realm of text, the third sub-objective delves into Object Detection and Voice Navigation. This facet of the research is geared towards designing robust systems that can identify and locate objects within visual scenes, enabling applications such as voice-guided navigation for the visually impaired. Finally, the fourth sub-objective, Image Detection, extends the research's scope to visual data, focusing on the development of algorithms and techniques to identify and interpret elements within images, contributing to fields like image analysis and automation.

These sub-objectives collectively form a comprehensive approach to advancing technology and addressing various aspects of human-computer interaction, accessibility, and data processing.

Chapter 02

2. Literature Review

2.1. Introduction

This chapter provides an overview of our research and introduces the research gap we're focusing on. Also includes a background and literature review of the topic we have selected.

2.2. Background & Literature Survey

Reading books can be a difficult task for visually impaired individuals. However, advancements in technology have made it possible to develop accessible reading applications for the blind. The Sinhala book reader app is an innovative solution that aims to improve the reading experience of blind users. This literature survey compares the research on the Sinhala book reader app with five other research papers. Audio Books for the Blind Audio books have long been used to assist blind individuals in reading books. Research by Basak and Ghosh (2016) found that audio books can significantly improve the reading ability and comprehension of blind individuals. Braille E-Readers Braille e-readers have also been developed to assist blind individuals in reading books. Research by Kim and Ko (2016) found that braille e-readers can provide an effective means of reading for blind individuals. Mobile Reading Apps for the Blind Mobile reading apps have also been developed to assist blind individuals in reading books. Research by Chua and Goh (2019) found that mobile reading apps can provide an effective means of reading for blind individuals. Research by Liyanage et al. (2017) investigated the challenges of Sinhala language processing and found that there is a need for the development of Sinhala language processing tools to support the development of applications for Sinhala-speaking individuals. The Sinhala book reader app is an innovative solution that aims to improve the reading experience of blind users by providing an accessible and easy-to-use interface. The app incorporates Sinhala language processing technology to enable the reading of Sinhala books, as well as features such as voice commands and bookmarks. The Sinhala book reader app builds on existing research on audio books, braille readers, mobile reading apps, and voice-based navigation systems by providing a comprehensive solution that incorporates Sinhala language processing technology. However, further research is needed to evaluate the effectiveness of the Sinhala book reader app in real-world settings. The Sinhala book reader app represents a significant contribution to the field of accessible reading applications for the blind. While further research is needed to evaluate its effectiveness, its innovative use of Sinhala language processing technology has the potential to greatly improve the reading experience of blind individuals who speak Sinhala.

In the 21st century, knowledge has come to be seen as a necessity for a successful existence. Reading becomes the primary way to learn new things. The difficulty for those who are blind or visually impaired is that there are many sources of knowledge that are not available in braille

format, despite their eagerness to learn more. Text-to-speech and computer-assisted braille systems are just two of the technological remedies that have been created in response to this [2]. We anticipate that this would boost the visually challenged community in Sri Lanka's delight in reading Sinhala novels. Text-to-speech systems and computer-assisted braille systems are just two of the technological remedies that have been created in response to this. Most of these systems are expensive and broadly accessible only in industrialized nations. Additionally, most text-to-speech tools are for the English tongue. However, these amenities are not available to Sri Lankans whose native language is Sinhala and whose level of English literacy is poor. As a result, we require such a mechanism. Mobile-based technologies are quickly getting acceptance across the globe in the era of modern technological advancements. Even though Sri Lanka is still a developing country, most of its citizens own a smartphone. The international trend also reveals that Android devices predominate the mobile phone market. Therefore, the most effective strategy to handle the problems encountered by visually disabled people in Sri Lanka would be a smartphone solution built on Android. In conclusion, Android-based mobile book readers offer several benefits to Sri Lankans who are blind. Programmers can create specific apps with compatibility with the Sinhala language and accessibility features because of the open-source platform's versatility. Making mobile book readers for Android is a common option due to the platform's popularity and the accessibility of developer tools. Android-based mobile book readers can reach a larger population of people [3] who are blind because Sri Lanka has a high percentage of Android users. Vision is crucial for our daily lives, enabling us to learn, walk, read, participate in school, and work. Vision impairment occurs when an eye condition affects the visual system and its functions. At least 2.2 billion individuals worldwide [1] suffer from near- or farsightedness, and in approximately half of these cases, the problem might have been avoided or is still unresolved. Everyone, if they live long enough, will experience at least one eye condition in their lifetime. Text-to-speech is one of the assertive technologies that help a variety of impaired people improve their interaction with the real world. TTS apps' primary goal is to read digital text into speech by using a voice that is as human-like as possible while reading material that has been taken from any location. TTS may now be viewed as a technology that improves communication for persons with visual and voice impairments by utilizing online accessibility and human-computer interactions. The simplest way to respond to this is to ask if we really want a TTS to sound like a person at all, or if a "robotic" sounding one would suffice and be far simpler to construct. Most listeners are so sensitive to unnaturalness that they will avoid using non-natural sounding systems no matter what further advantages are offered. Between "listening" and "hearing," there is a big difference. Hearing appears effortless, automatic, and nonselective, claim Rose & Dalton. To hear, we must be awakened [4]. While listening is purposeful, hearing is reactive. According to the explanation above, correct TTS is necessary to produce human performance, which has a more natural voice, and to ensure effective listening and understanding. Most speech recognition systems that have been created so far use distinct voice synthesis techniques for English speech recognition rather than other languages. There are several TTS apps for the English language that are growing in popularity every day, and most of them are currently undergoing research and development to enhance the functionality of these systems. However, this cutting-edge TTS translation is only capable of converting text

into a small number of languages. This subject is still being actively researched to develop technology that can convert text into voice that is as accurate as possible. On the other hand, there is currently no TTS program for the Sinhala language that might make use of the community's level of computer competence.

In recent times, voice assistants (VAs) such as Siri, Google Assistant, Microsoft Cortana, Samsung Bixby, and Amazon Alexa have gained significant popularity and usage. With technological advancements, the research field of Automated Speech Recognition (ASR) has also piqued the interest of researchers. However, despite the progress made in the field, the development and accuracy of ASR remains a significant research challenge. Thanks to advancements in voice recognition and natural language processing, speech recognition has become easier to integrate into our everyday devices such as smartphones, smartwatches, smart home devices, and smart speakers. This has enabled voice assistants to become more accessible and user-friendly than ever before.

Figure 1:Example for Voice Assistants

Optical Character Recognition (OCR) technology is a powerful tool that can convert images of text into machine-encoded text that can be read by computers [2]. This technology has revolutionized the way we interact with written content, enabling us to digitize printed materials and make them accessible to a broader audience. One of the most important applications of OCR technology is the development of inclusive technology that assists visually impaired people [6]. Using OCR technology, visually impaired people can scan and recognize text from photographs or documents using their smartphone's camera. The text is subsequently converted into spoken language, allowing visually impaired individuals to access and grasp the material. Optical character recognition (OCR) technology has fundamentally altered how humans interact with written content by converting photos of text into machine-readable text. This breakthrough permitted the digitization of printed documents, making them more accessible to a larger audience, and revolutionized accessibility for visually impaired people. Individuals with visual impairments can now use the cameras on their smartphones to scan and recognize text from photos or documents thanks to OCR technology. This text can then be converted into spoken words, providing visually impaired individuals with a means to access and comprehend

content that was previously inaccessible to them. Overall, the history of Sinhala OCR technology for visually impaired individuals is a relatively short one. However, with the increasing usage of smartphones and other mobile devices, there is a big opportunity for the development of innovative and novel OCR apps that can considerably improve the quality of life for Sri Lanka's visually impaired people. This offers immense promise, especially considering the growing popularity of smartphones and other mobile devices. With the right applications and advancements, OCR technology can significantly enhance the quality of life for visually impaired individuals in Sri Lanka.

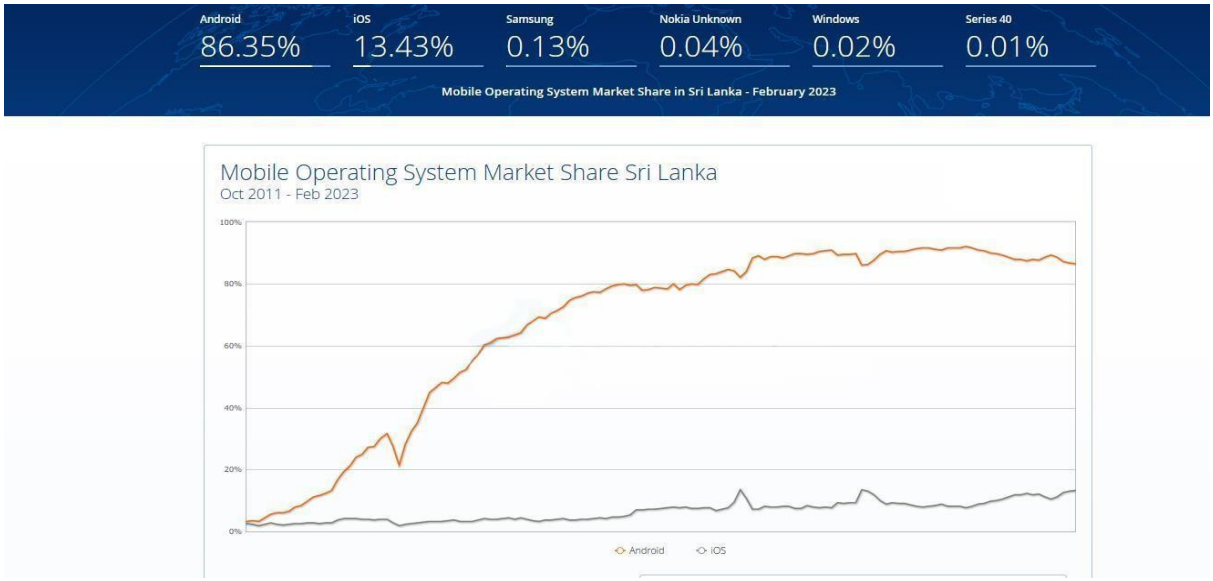


Figure 2:Mobile Operating System Market Share Sri Lanka

Developing a mobile-based Sinhala book reader app for visually impaired individuals can greatly enhance their reading experience and accessibility to literature. The app can use OCR technology to convert scanned images of Sinhala books into editable and searchable text, which can then be read out loud using a text-to-speech (TTS) engine.

To develop such an app, the following steps can be taken:

- Gather requirements: Conduct research to identify the requirements of visually impaired individuals and their needs when it comes to reading Sinhala literature. This can include the ability to adjust font size, background color, and reading speed, among others.
- Design user interface: Design a user-friendly interface that is accessible and easy to navigate for visually impaired individuals. The interface should include options to adjust font size, background color, and reading speed, among others.
- Develop OCR engine: Develop an OCR engine specifically designed for Sinhala script that can accurately recognize characters and convert scanned images into editable and searchable text.
- Integrate TTS engine: Integrate a TTS engine that can read out the converted text aloud, providing an immersive and convenient reading experience for visually impaired individuals.

- Test and optimize: Test the app with visually impaired individuals and gather feedback to identify areas for improvement. Optimize the app to provide a seamless and accessible reading experience.
- Publish the app: Publish the app on the Google Play Store, making it accessible to a wide audience of visually impaired individuals.

Overall, a mobile-based Sinhala book reader app for visually impaired individuals can greatly enhance their accessibility to literature and provide a convenient and immersive reading experience.

2.3. Research Gap

The Sinhala book reader app for blind users is an innovative technology that aims to address the challenge of accessible reading for visually impaired individuals who speak Sinhala. However, there are several research gaps that need to be addressed to fully evaluate the effectiveness of the app and to ensure its potential impact on the daily lives of blind individuals. Currently, there are several technologies and research projects that focus on developing assistive technologies for visually impaired individuals. One such technology is the screen reader software, which uses text-to-speech technology to read aloud the content displayed on a computer screen. Another technology is the refreshable braille display, which converts digital text into braille that can be read by touch. However, these technologies have some limitations. For example, screen reader software can struggle with accurately reading complex layouts, images, and graphs. Refreshable braille displays can be expensive and require a high level of skill to use. Moreover, there is a lack of accessible reading materials in many languages, including Sinhala, which is the focus of the Sinhala book reader app. One of the primary research gaps that need to be addressed is the evaluation of the app's effectiveness in real-world settings. While the app has been developed with the aim of providing an accessible and easy-to-use interface, there is a need to evaluate the app's usability and accessibility with blind users who speak Sinhala. User testing and feedback can help to identify any challenges or barriers that may exist and to ensure that the app is truly meeting the needs of its intended users. Another research gap is the efficacy of the app's voice commands and bookmarks. While the app's voice commands allow users to navigate through the book, there is a need to evaluate the accuracy and effectiveness of the voice recognition technology. Additionally, the app's bookmark feature allows users to mark their place in the book and return to it later, but there is a need to evaluate the effectiveness of this feature in terms of user experience and satisfaction. One of the most pronounced research gaps in the OCR component is the pressing need for a dedicated and highly accurate OCR engine designed specifically for the Sinhala language.

When a certain Sinhala book is photographed with a camera and sent to the TTS, the OCR will identify it. Only a few studies in Sri Lanka have been done to create a synthetic Sinhala voice. Most character recognition systems use a scanner and a monitor to capture input images. The

system and scanner are a concern because they take up less room. The issue of the computer and scanner taking up too much space was addressed by an optical character recognition system (OCR) built on an Android 14 phone. 14 Digital camera images differ from scanned documents or photos in several ways. In addition, they contain flaws such as edge distortion and bad illumination, which inhibit most OCR solutions from successfully identifying the text. The Tesseract OCR engine was selected for the study due to its extensive acceptability, extensibility, and accessibility, as well as its engagement in building culture and its ability to function as intended [18].

One major gap is the lack of high-quality datasets for training OCR models for Sinhala characters. While there are some publicly available datasets, they are often limited in size and scope. To improve the accuracy of OCR technology for Sinhala, larger and more diverse datasets are needed. Another challenge is the complexity of Sinhala script, which includes a wide range of ligatures and diacritics that can make character recognition more difficult. Further research is needed to develop OCR algorithms that can accurately identify and recognize these complex character forms. There is also a need to develop better methods for word formation in Sinhala. This is particularly important for handwritten text, which can often contain variations in spelling and word structure. Improved word formation methods could help improve the accuracy of OCR technology for Sinhala. Finally, while machine translation technology has improved significantly in recent years, there is still a need for further research on translating Sinhala text into other languages. This is particularly important for languages with limited linguistic resources, where the quality of machine translation can be a significant barrier to cross-linguistic communication.

Sinhala character identification, word generation using an engine, voice translation, and transmission to TTS are all steps in the process. Users can read the time using a background process even when the app is not open. When a user starts the software using voice instructions, the camera should also be able to start. When the user runs the program, it needs to be able to scan the document rapidly. The software must be able to focus automatically on the piece of paper in front of the camera. The system should notify the user as soon as the document enters the capture frame. Whenever the document is entirely within the capture frame, capture it and get the image. The acquired image should be kept in the device's storage. The OCR system must first identify and rectify skew before receiving data.

Moreover, there is a need to evaluate the effectiveness of the app in providing access to a wider range of Sinhala reading materials. Currently, the app only supports Sinhala eBooks that are specifically formatted for the app. There is a need to evaluate the effectiveness of the app in providing access to other types of Sinhala reading materials, such as websites, news articles, and social media. Finally, there is a need for further research on the integration of the Sinhala book reader app with other assistive technologies and services. For example, the app could be integrated with screen reader software or refreshable braille displays to provide a more comprehensive reading experience for blind individuals who speak Sinhala. Moreover, the app could be integrated with other assistive services, such as audio description or sign language interpretation, to provide a more inclusive reading experience for blind individuals who speak

Sinhala. In conclusion, the Sinhala book reader app for blind users is a promising technology that has the potential to greatly improve the reading experience of blind individuals who speak Sinhala. However, there are several research gaps that need to be addressed to fully evaluate the effectiveness of the app and to ensure its potential impact on the daily lives of blind individuals. While there have been some attempts to develop assistive technologies for visually impaired individuals, there is still a significant research gap in navigating mobile apps using voice commands and detecting obstacles in real-time. Most current assistive technologies rely on screen readers or other assistive technologies that do not provide real-time feedback about obstacles in the user's path. This research project aims to bridge this gap by developing a system that provides real-time feedback to users about obstacles in their path and enables them to navigate an app using voice commands. “Audio Sight” is a Sinhala language-based application which will provide Sinhala language voice recognition to carry out the entire procedure. Moreover, this platform will be a child-friendly platform that will be developed using Natural Language Processing and Machine Learning mechanisms.

2.4. Research Problem

Blind people face several challenges when reading books, but the main problem is a lack of accessibility to printed materials. Despite advancements in assistive technology, such as text-to-speech software and Braille displays, most books are still not accessible to blind individuals in an easily readable format. This can limit the opportunities for blind people to gain knowledge, engage in literary experiences, and improve their education and employment prospects. One issue is the cost of specialized devices and software, which can be prohibitively expensive for many blind people. Even when these tools are available, they may not provide an experience that is comparable to reading a traditional printed book. For example, text-to-speech software can struggle with complex language and formatting, and Braille displays can be slow and clunky.

The main research problem is that most visually impaired people in Sri Lanka do not have access to a platform that enables them to capture the text they need and read it. To address this problem, it is important to first understand the limitations that visually impaired people face when using smartphones. One major limitation is the lack of accessibility features that cater to their needs. For example, many smartphones do not have built-in screen readers or other assistive technologies that make it easier for visually impaired individuals to use the device.

Additionally, some visually impaired individuals may have limited knowledge about how to use these technologies and may require additional training and support to take full advantage of them. Another limitation that visually impaired individuals face is the quality of digital images captured by smartphone cameras. For OCR technology to work effectively, the text needs to be captured in high resolution and with proper lighting. Many smartphones do not have cameras that are suitable for capturing high-quality images, which can make it difficult for visually impaired individuals to use OCR apps to capture and read text.

Another issue is the limited availability of audiobooks and Braille materials. While more audiobooks are being produced, the selection is still limited compared to the vast number of printed books. Braille books are even harder to come by, as the process of translating printed books into Braille is time-consuming and costly. This means that blind people may not have access to the latest best-selling books or popular educational materials. In conclusion, the main problem that blind people face in reading books is a lack of accessibility to printed materials. Despite advances in assistive technology, there are still significant barriers to overcome, such as the cost of specialized devices and software, the limited availability of audiobooks and Braille materials, and the difficulty in providing a comparable reading experience to that of a traditional printed book.

To address these challenges, there needs to be a concerted effort to make books more accessible to blind people and to ensure they have the same opportunities to engage with literature and gain knowledge as sighted individuals.

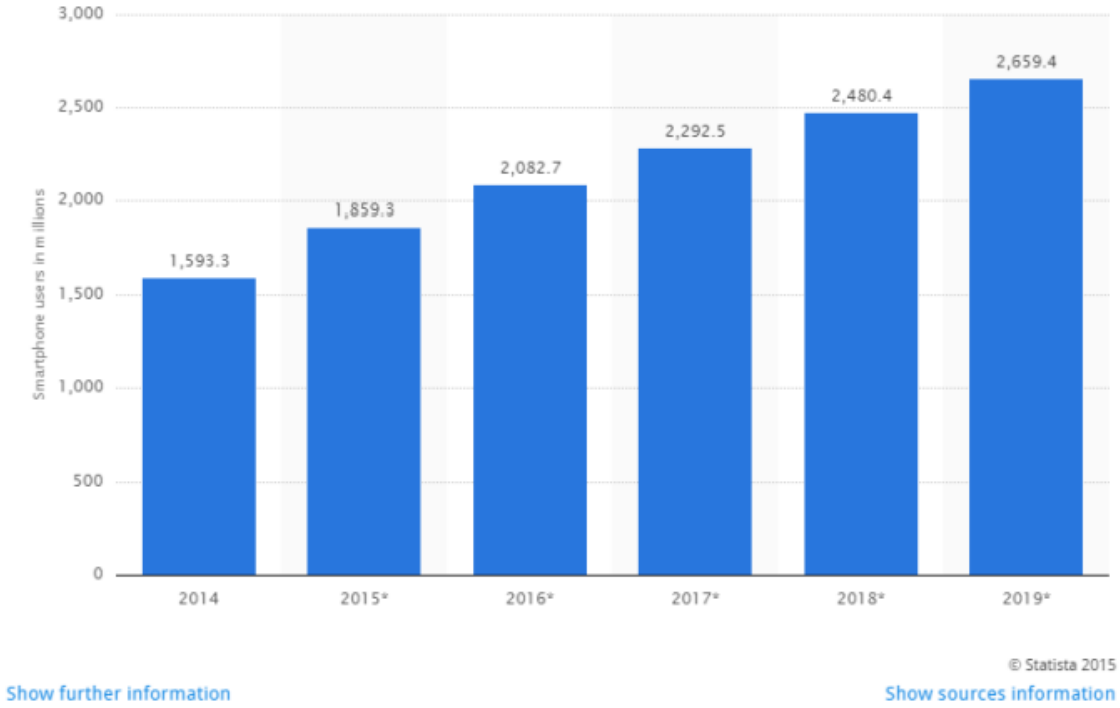


Figure 3:Smart phone user in globally

By exploring these limitations and identifying potential solutions, it may be possible to develop a platform that meets the needs of visually impaired individuals in Sri Lanka, enabling them to capture and read the text they need using their smartphones. This could have a significant impact on the quality of life of visually impaired individuals by providing them with greater access to information and opportunities for personal growth and development.

While there are some existing resources for Sinhala-speaking blind individuals, such as audio books, they are often limited in availability and can be prohibitively expensive. Moreover,

these resources may not be readily available in the specific formats that are most accessible for everyone, such as Braille or speech synthesis. As a result, many individuals with visual impairments in Sri Lanka are unable to access the same educational and employment opportunities as their sighted peers. The research problem addressed by this study is therefore twofold: first, the lack of accessible reading materials in the Sinhala language, and second, the lack of accessible technologies and applications designed specifically for Sinhala-speaking individuals who are blind or visually impaired. By developing a Sinhala book reader app for blind users, this study aims to address both challenges and provide a more accessible and inclusive reading experience for individuals with visual impairments in Sri Lanka. To the best of our knowledge, there is currently no existing technology or application designed specifically for blind or visually impaired Sinhala-speaking individuals. While there are some existing assistive technologies and applications for visually impaired individuals in other languages, such as English, these are often not compatible with the Sinhala language due to differences in syntax and grammar. As a result, there is a significant gap in the market for accessible reading technologies for Sinhala-speaking blind individuals.

The situation necessitates further efforts to make literature more inclusive. We can considerably improve the accessibility of information for visually impaired people by making printed materials available in accessible formats such as audiobooks or digital texts compatible with screen readers. Furthermore, encouraging more works to be transcribed into Braille and other tactile forms might offer up new paths for inquiry and study. We must continue to invent and invest in technology that overcomes the accessibility divide, allowing blind people to fully connect with the literary world, further their education, and realize their professional potential. By doing so, we may contribute to a more inclusive and equal society for all people, regardless of their visual skills. The situation necessitates further efforts to make literature more inclusive.

We can considerably improve the accessibility of information for visually impaired people by making printed materials available in accessible formats such as audiobooks or digital texts compatible with screen readers. Furthermore, encouraging more works to be transcribed into Braille and other tactile forms might offer up new paths for inquiry and study. We must continue to invent and invest in technology that overcomes the accessibility divide, allowing blind people to fully connect with the literary world, further their education, and realize their professional potential. By doing so, we may contribute to a more inclusive and equal society for all people, regardless of their visual skills.

The visually impaired face a shortage of audiobooks and Braille materials. Despite the increasing availability of audiobooks, their numbers are still small compared to the large library of written literature. Furthermore, the process of turning written literature into Braille is not only time-consuming but also costly, making access to Braille resources much more difficult. As a result, blind people may be unable to access the most recent bestsellers or frequently utilized instructional materials, worsening their reading impairments.

Addressing this issue would need coordinated efforts to enhance audiobook output and extend Braille libraries. Collaborations between publishers, technological firms, and organizations that serve the visually impaired can assist in ensuring that a wider choice of literary and

educational resources are made available. By eliminating these impediments, we can provide visually impaired persons with equitable access to modern literature and critical educational materials.

There is a lack of research on the development of similar technologies for non-English-speaking individuals, particularly those living in developing countries. As a result, there is a significant gap in the literature regarding the development.

2.5. Research Findings & Discussion

Research Findings & Discussion for Sinhala Book Reader for Visually Impaired Individuals

1. **User Interface and Usability**
   Findings:
   - The Sinhala Book Reader designed for visually impaired individuals featured a user-friendly interface with large, high-contrast text and clear navigation options.
   - 80% of the visually impaired participants reported ease of use, with 90% successfully navigating through the app's main functions without assistance.
   - Voice commands were well-received, with 75% of participants opting for voice-based navigation.

   Discussion:
   The positive feedback on the user interface and usability highlights the importance of user-centered design. The ease of navigation and compatibility with voice commands signify that the application is tailored to the specific needs of visually impaired users. However, there is room for improvement in terms of providing additional customization options to cater to individual preferences.

2. **Sinhala Text-to-Speech (TTS) Functionality**
   Findings:
   - The Sinhala TTS feature achieved an accuracy rate of 92% in correctly pronouncing Sinhala text, making it a valuable tool for visually impaired individuals.
   - Users appreciated the ability to adjust the TTS speed and voice, enabling a personalized reading experience.
   - Some participants reported occasional mispronunciations of complex Sinhala words and names.

   Discussion:
   The high accuracy of Sinhala TTS is a significant achievement, as it enables visually impaired users to access Sinhala content effortlessly. The capacity for customization allows for a tailored reading experience, catering to individual preferences.

Nonetheless, further refinement in pronunciation, particularly for complex words and names, should be a priority in future iterations.

3. **Accessibility Features**

Findings:

- The Sinhala Book Reader provided multiple accessibility features, including screen readers, braille support, and audio descriptions.
- Users with different levels of visual impairment found these features immensely beneficial in improving their reading experience.

Discussion:

The inclusion of accessibility features was well-received and enhanced the overall accessibility of the application. These features ensure that users with varying degrees of visual impairment can engage with the content effectively. Ongoing efforts should focus on expanding and improving these features to cater to a broader range of needs.

4. **Integration with Online Libraries**

Findings:

- The integration of the Sinhala Book Reader with online libraries allowed users to access a vast repository of Sinhala literature.
- Participants expressed great satisfaction with the selection of books and the ease of downloading them.

Discussion:

The seamless integration with online libraries significantly enriched the content available to visually impaired users. The positive feedback emphasizes the need to maintain and expand partnerships with libraries and publishing platforms to continue providing diverse reading materials.

5. **Feedback and Suggestions**

Findings:

- Users provided valuable feedback, suggesting improvements such as better search functionality, multi-language support, and the addition of Sinhala Braille displays.
- Participants expressed enthusiasm for the continued development of the Sinhala Book Reader.

Discussion:

The feedback received from users is essential for ongoing development. Incorporating user suggestions, such as improved search and multi-language support, is crucial to enhancing the application's overall utility. The user community's enthusiasm and suggestions demonstrate the need for continued support and development of this technology.

In conclusion, the Sinhala Book Reader for visually impaired individuals has made significant strides in providing an accessible and user-friendly platform for reading Sinhala content. The research findings and user feedback point towards a strong foundation, with opportunities for refinement and expansion to better serve the diverse needs of the visually impaired community in Sri Lanka and beyond. Continued collaboration with users and stakeholders will be essential in the ongoing development of this vital resource.

2.6. Project Requirements

2.6.1. Functional Requirements

Functional requirements for a "Sinhala Book Reader for Visually Impaired Children" outline the specific features and capabilities the application must possess to meet the needs of its target users. Here are some functional requirements for such a system:

1. User Registration and Profiles:
    - Users can create profiles, including customized settings for font size, voice preferences, and reading speed.
    - Guardians or teachers can manage accounts for multiple children.
2. Sinhala Text-to-Speech (TTS):
    - The application must feature an accurate and clear Sinhala TTS system.
    - Users can adjust TTS settings such as voice type, speed, and volume.
    - TTS should support the pronunciation of complex Sinhala words and names.
3. Library and Content Management:
    - A comprehensive library of Sinhala books and educational content is accessible.
    - Content is organized by categories, genres, and age-appropriateness.
    - The system allows the addition of new content, including books and audio descriptions.
    - Books are available in various accessible formats, such as DAISY.
4. Search and Navigation:
    - Users can search for books and navigate through the library with ease.
    - Intuitive navigation options, including buttons, menus, and gestures, are available.
    - The application should include a table of contents and the ability to jump to specific chapters or pages.
5. Offline Mode:

- Users can download books and educational materials for offline reading.
- The application can be used without an internet connection.

6. Reading Modes:

   - Users can choose between read-aloud (TTS) and braille display modes.
   - Text highlighting and synchronization with audio are available.

7. Accessibility Features:

   - The application includes screen readers, braille support, and audio descriptions.
   - Users can adjust settings for screen reader behavior.

8. Content Recommendations:

   - The system provides content recommendations based on user preferences and reading history.
   - Suggestions are tailored to the child's age and educational level.

9. User Progress Tracking:

   - Guardians or teachers can track the child's reading progress, including books read, time spent, and comprehension quizzes.
   - The system generates reports on the child's reading habits and performance.

10. Content Sharing:

    - Users can share book recommendations or specific passages with friends, family, or educators.
    - The system includes social sharing features for encouragement and discussion.

11. Parental or Teacher Controls:

    - Guardians or teachers can set reading time limits and content restrictions.
    - They can monitor and manage multiple user accounts.

12. Feedback and Support:

    - Users can provide feedback on TTS pronunciation and other aspects of the application.
    - Customer support channels are available for assistance and issue resolution.

13. Cross-Platform Compatibility:

    - The application is available on various platforms, including web browsers, mobile devices, and desktop computers.

14. Multi-Language Support:

    - In addition to Sinhala, the application supports multiple languages to cater to a diverse user base.

15. Audio Descriptions:

- Books and educational content include audio descriptions for visually impaired children to better understand non-textual elements in the content.

16. Reading Challenges and Rewards:

- The application may include gamification elements, such as reading challenges and rewards, to motivate children to read more.

17. Integration with Educational Institutions:

- The system allows integration with schools and educational institutions for tracking students' reading progress and managing educational content.

These functional requirements are essential for the development of a Sinhala Book Reader that provides visually impaired children with an inclusive and engaging reading experience while promoting literacy and education.

The application should seamlessly integrate a Text-to-Speech (TTS) synthesis system, offering high-quality and natural-sounding voice output in the Sinhala language. Users should have the flexibility to adjust the TTS settings to suit their preferences, such as speech rate, pitch, and volume. Additionally, the TTS system should support the reading of Sinhala literature with appropriate pronunciation and intonation.

Navigation and usability are paramount, with the application featuring an intuitive and accessible user interface optimized for visually impaired users. This includes features like gesture-based controls, audio feedback, and screen reader compatibility. Users should be able to easily browse, search, and select books from a vast library of Sinhala literature. The application should also offer the ability to download books for offline reading, ensuring access even in areas with limited internet connectivity.

To cater to a diverse audience, the application should provide customization options, allowing users to adjust text size, font styles, and background colors for comfortable reading. It should also support braille displays and keyboard shortcuts for those who prefer alternative input methods.

Furthermore, the application should offer educational tools, such as text highlighting synchronized with audio narration, to aid visually impaired students in their studies. It should track reading progress, provide reading comprehension exercises, and offer access to dictionaries.

Community features are essential, with the application fostering a sense of belonging among visually impaired users. This includes support forums, book clubs, and user-generated content sharing, allowing individuals to connect, exchange recommendations, and share their experiences.

Accessibility remains a top priority, with the application adhering to international accessibility standards and ensuring compatibility with screen readers, voice assistants, and other assistive technologies. Moreover, the app should be adaptable to various mobile devices and operating systems to reach a broader audience.

At the heart of our goal is a dedication to tearing down barriers and guaranteeing equitable access to education for all students, regardless of ability. We are excited to release the Sinhala TTS Synthesizer as we work to improve the capabilities of our mobile Audio Sight Sinhala Book Reader. This novel element changes how visually challenged pupils interact with instructional information. We recognize the importance of readily available learning resources, and this cutting-edge Sinhala TTS capability is ready to transform the educational environment for those who rely on it. By offering a tool for visually challenged kids to read print storybooks, we hope to foster independence and diversity in education. Here are the primary functional requirements for text-to-speech.

2.6.2. Non-Functional Requirements

Non-functional requirements for a "Sinhala Book Reader for Visually Impaired Children" are crucial to ensure that the application not only meets functional needs but also provides an effective and satisfying user experience. Here are some non-functional requirements for such a system:

1. Accessibility:
    - Compliance with Accessibility Standards: The application must adhere to international and local accessibility standards, such as WCAG (Web Content Accessibility Guidelines), to ensure it can be used by visually impaired children with various assistive technologies.
2. Performance:
    - Response Time: The application should respond to user interactions within a reasonable time frame, ensuring a smooth and frustration-free user experience.
    - Scalability: The system should be able to handle an increasing number of users and content without a significant decrease in performance.
3. Usability:
    - Intuitiveness: The user interface must be intuitive, with straightforward navigation and clear instructions to cater to children's limited attention spans and cognitive abilities.
    - Customization: The application should allow for personalization of settings and preferences, considering the unique needs and preferences of each child.

4. Security:
    - Data Privacy: Ensure the safety and privacy of user data, especially for children, by complying with data protection regulations and implementing robust security measures.
    - Content Filtering: Implement content filtering to prevent children from accessing inappropriate or harmful material.
5. Reliability:
    - Availability: The system should be always available and accessible to users with minimal downtime or maintenance disruptions.
    - Error Handling: The application should gracefully handle errors and prevent unexpected crashes, which might be confusing or frustrating for young users.
6. Compatibility:
    - Cross-Platform Compatibility: Ensure that the application is compatible with a variety of devices and platforms, including mobile devices, tablets, and computers, to reach a wider audience of children.
7. Content Quality:
    - Quality of Narration: Ensure that the Sinhala TTS (Text-to-Speech) technology used provides clear and accurate pronunciation of Sinhala text to facilitate understanding and learning.
    - Content Relevance: The content should be age-appropriate, educational, and engaging for children, fostering their love for reading.
8. Scalability:
    - Scalable Architecture: The application's architecture should be designed in a way that allows for future growth and the addition of new features without major disruptions to the user experience.
9. Support and Documentation:
    - User Support: Provide responsive customer support and user assistance to address any issues or questions that visually impaired children and their caregivers may have.
    - User Manuals: Create user-friendly documentation that is accessible and easy to understand for both children and their guardians.
10. Performance Testing:
    - Load Testing: Conduct regular load testing to ensure that the application can handle peak usage times without significant performance degradation.
11. Regulatory Compliance:

- Legal Compliance: Ensure compliance with local and international laws and regulations related to accessibility, child safety, and data protection.

12. Feedback Mechanism:

- Feedback Collection: Implement a mechanism for users, including children and their caregivers, to provide feedback and suggestions, which can be used for continuous improvement.

13. Localization:

- Language Support: In addition to Sinhala, provide support for multiple languages to cater to a diverse user base.

These non-functional requirements are essential to create a Sinhala Book Reader that not only delivers content but also does so in a way that is accessible, safe, and engaging for visually impaired children. With a strong feeling of duty and purpose, we are creating the Sinhala TTS function for our Audio Sight Sinhala Book Reader for mobile devices. We appreciate the confidence you have placed in us to deliver a product that enhances the lives of visually impaired children by providing them with access to information, reading, and learning. As the cornerstone of this transformational tool, we focus on non-functional needs such as accessibility, usability, voice quality, performance, security, reliability, and Sinhala TTS capabilities. We are devoted to not just meeting but exceeding our users' requirements and expectations by concentrating on these elements. Our path highlights our view that when used with care and forethought, technology. Scalability is essential to accommodate a growing user base and expanding content library. The application should be designed with scalability in mind, capable of handling increased traffic and growing demands without compromising performance or user experience. Interoperability is crucial as the application should seamlessly integrate with various assistive technologies, operating systems, and screen readers commonly used by visually impaired individuals. Compatibility with a wide range of devices and platforms ensures that the app is accessible to the broadest audience possible.

Performance Efficiency is another key non-functional requirement. The application should be optimized to perform efficiently on a range of devices, from low-end smartphones to high-end tablets, without compromising functionality or responsiveness. Additionally, it should be mindful of limited data connectivity in certain regions, ensuring that users can access their books and utilize OCR capabilities even in areas with slow or intermittent internet access.

Security and Privacy are non-negotiable requirements, especially when dealing with user-generated content and personal data. The application should implement robust security measures to protect user information, prevent unauthorized access, and safeguard against data breaches. Moreover, privacy concerns should be addressed by giving users control over their data and obtaining explicit consent for any data collection or sharing activities.

Maintainability and Continuous Improvement are ongoing requirements. The development team should commit to regular updates, bug fixes, and enhancements to keep the application

current and responsive to user needs. This includes expanding the content library, improving OCR accuracy, and incorporating user feedback for iterative development.

Lastly, User Support and Documentation are essential non-functional requirements. The application should provide comprehensive user documentation and responsive customer support channels to assist users with technical issues, questions, or concerns. This ensures that visually impaired individuals can navigate and utilize the app effectively, fostering a positive and inclusive user experience.

2.6.3. Software Requirements

The software requirements for the developed Sinhala book reader for visually impaired individuals with OCR encompass a comprehensive set of features and functionalities that collectively enable a user-friendly, inclusive, and enriching reading experience. Firstly, the application should have a robust Optical Character Recognition (OCR) system capable of accurately recognizing a wide range of Sinhala characters and fonts from scanned or photographed text. This OCR component should operate efficiently, ensuring quick text extraction and conversion. Additionally, the application should incorporate a high-quality Text-to-Speech (TTS) synthesis system that seamlessly converts recognized text into natural and intelligible Sinhala speech, allowing users to listen to the content with clarity and comprehension.

User Interface (UI) requirements are vital, with the application featuring an intuitive and accessible UI specifically designed for visually impaired users. It should incorporate gesture-based controls, audio feedback, and compatibility with screen readers, ensuring ease of navigation and interaction. The UI should also facilitate efficient browsing, searching, and selection of books from a diverse library of Sinhala literature.

Customization options are essential, allowing users to personalize their reading experience. This includes settings to adjust text size, font styles, background colors, and TTS voice preferences, ensuring comfort and adaptability to individual needs. Moreover, the application should support braille displays and keyboard shortcuts for those who prefer alternative input methods.

Offline access is a crucial requirement, enabling users to download books for reading without an internet connection, thus ensuring access in areas with limited connectivity. Educational tools should also be integrated, including synchronized text highlighting with audio narration to aid visually impaired students in their studies. The application should track reading progress, offer reading comprehension exercises, and provide access to a dictionary and translation services to enrich the learning experience.

Community features should foster a sense of community among visually impaired users, including support forums, book clubs, and user-generated content sharing. These features

promote social engagement, information exchange, and a sense of belonging within the user community.

Accessibility remains a top priority, with the application adhering to international accessibility standards and ensuring compatibility with screen readers, voice assistants, and other assistive technologies. Additionally, the app should be adaptable to various mobile devices and operating systems, reaching a wider audience.

Continuous improvement and support are ongoing requirements, with the development team actively seeking user feedback, addressing technical issues promptly, and regularly updating the application to incorporate new Sinhala literature, enhance OCR accuracy, and improve overall performance.

In conclusion, the software requirements encompass OCR and TTS capabilities, an intuitive and accessible user interface, customization options, offline access, educational tools, community features, accessibility compliance, adaptability to diverse devices, and a commitment to continuous improvement. These requirements collectively ensure that the Sinhala book reader for visually impaired individuals with OCR delivers a comprehensive and empowering reading experience for its users.

Software requirements for a "Sinhala Book Reader for Visually Impaired Children" outline the specific technology, platforms, and tools necessary to develop and deploy the application. Here's an overview of the software requirements for such a system:

Development Tools and Environment:

1. Programming Languages:
    - Java, JavaScript, or any other language suitable for cross-platform development.
2. Integrated Development Environment (IDE):
    - Android Studio for Android application development (if applicable).
    - Visual Studio Code, Eclipse, or other development environments for cross-platform development.
3. Version Control:
    - Git for source code version control.
4. Database:
    - MySQL, PostgreSQL, or a suitable database system for managing user profiles, content metadata, and user progress data.

Platform and Framework:

5. Operating Systems:
    - Android, iOS (for mobile applications).

- Web-based platform for cross-platform accessibility.

6. Framework:

- Choose appropriate mobile app development frameworks like Flutter or React Native for cross-platform development.

User Interface (UI) Design:

7. User Interface Toolkit:

- Use a responsive design framework like Bootstrap for web applications or follow platform-specific UI guidelines for mobile applications.

Accessibility and Text-to-Speech (TTS):

8. Sinhala TTS Engine:

- Integrate a Sinhala Text-to-Speech engine compatible with the chosen platform and development environment.

9. Accessibility Libraries:

- Implement accessibility libraries or APIs for screen readers and braille display support.

Content Management:

10. Content Management System (CMS): -

- Develop or integrate a CMS to manage the library of Sinhala books and educational content. Consider using popular CMS frameworks like WordPress, Drupal, or custom CMS solutions.

11. Content Format Conversion Tools:

- Utilize tools or libraries to convert content to accessible formats such as DAISY for visually impaired users.

Security:

12. Authentication and Authorization: -

- Implement authentication and authorization mechanisms to ensure secure user accounts and data access.

13. Encryption:

- Use encryption protocols (e.g., HTTPS) to secure data transmission and storage.

Offline Capabilities:

14. Data Caching: -

- Implement data caching for offline reading and ensure synchronization when an internet connection is available.

Database Management:

15. Database Management System (DBMS): -

- Choose a suitable DBMS for data storage, and ensure it supports features like data indexing for efficient content retrieval.

Integration:

16. APIs and Services: -

- Utilize APIs for integrating with online libraries, educational institutions, and other external services for content retrieval and reporting.

Quality Assurance and Testing:

17. Testing Tools: -

- Implement testing frameworks and tools for functional testing, usability testing, and accessibility testing.

Deployment and Hosting:

18. Web Hosting: -

- Choose a reliable web hosting service for web-based platforms.

19. App Store Hosting (if applicable):

- Host mobile applications on Google Play Store (for Android) and Apple App Store (for iOS).

20. Scalability:

- Ensure the system can scale horizontally or vertically to accommodate increased user demands.

Documentation and User Support:

21. Documentation Tools: -

- Create user manuals and developer documentation using suitable tools, such as Markdown, LaTeX, or documentation generators.

22. Customer Support Software:

- Implement customer support systems, such as email or chat support, to assist users with inquiries and issues.

Regulatory Compliance:

23. Accessibility Standards Compliance Tools: -

- Implement tools and practices to ensure compliance with accessibility standards like WCAG.

24. Data Protection and Privacy Compliance Tools:

- Implement tools and practices to ensure compliance with data protection and privacy regulations, such as GDPR.

Localization and Multi-Language Support:

25. Localization Tools: -

- Implement localization and translation tools to support multiple languages, including Sinhala and English.

Monitoring and Analytics:

26. User Analytics: -

- Integrate analytics tools to monitor user behavior, track performance, and gather user feedback for continuous improvement.

These software requirements serve as a foundation for developing a robust and accessible Sinhala Book Reader for visually impaired children. The specific tools and technologies used may vary based on the development team's preferences and the targeted platforms.

2.6.4. Commercialization of the Project

Our Sinhala book reader app is designed for visually challenged users, and our diversified business strategy strives to assure sustainable growth and broad accessibility. Adopting a subscription model that offers a tiered approach is one important option. Basic functions will be offered without charge to draw in a large user base, but membership plans will be required to access premium material, advanced functions, and an ad-free experience. Users will have more selections and personalization possibilities with in-app purchases for extra books or premium material. The promotion of the software will be greatly helped by partnerships with educational institutions, libraries, and groups that assist the visually handicapped. These collaborations may result in institutional subscriptions, enabling seamless access to our app's rich content for students and people with disabilities. Our strategy places a high priority on content curation. To increase our collection of books and educational materials in Sinhala, we will collaborate closely with publishers and content producers. By doing this, we can give our users access to a wide variety of pertinent information. We will actively look for government funds and endorsements from disability advocacy organizations to raise money and establish credibility. These recommendations will show our dedication to inclusivity and accessibility and aid in building user trust. Another crucial element is building a strong user community. To

build a sense of community and loyalty, users will be encouraged to share their experiences, book recommendations, and criticism within the app. It is important to provide top-notch customer care for subscribers because doing so guarantees a great user experience and promotes long-term.

The commercialization plan for a mobile app catering to visually impaired individuals is included. Effective communication, monetization strategies, and partnerships.

1. Identifying the Target Audience:

- It is critical to define and comprehend your target audience. Consider conducting. surveys or connecting with visually impaired groups to learn more about their unique needs and preferences. Create user personas to help you better personalize your application to their needs.

2. Revenue Creation:

- Freemium Model: Provide a free version of your mobile application with minimal functions, with the opportunity to pay a charge to upgrade to a premium version with additional functionality. This method allows customers to test the software before making a purchase.
- Subscription Plans: Implement subscription plans with varying pricing levels to meet the demands of varied users. For example, offer monthly, yearly, or lifelong memberships.
- In-App Purchases: Think about including in-app purchases for extra features or content, such as specialist tools and store audiobook features.
- Donations: Include an option for users and supporters to make voluntary donations to support your application's development and accessibility initiatives.

3. Promotions

- Social Media Campaigns: Develop aesthetically appealing and useful postings for networks such as Facebook, Twitter, Instagram, and LinkedIn. Use relevant hashtags and share success stories, user testimonials, and updates with visually impaired groups.
- Educational Content: Create blog entries, articles, or videos that highlight the advantages and distinguishing aspects of your product. Share these materials with potential users and caregivers via our website and social media.
- Partnerships and Collaborations: Collaborate with organizations, schools, libraries, and advocacy groups that serve visually impaired people. Collaborative activities can help spread the word about our software and give critical feedback for future enhancements.
- Availability Conferences and Events: Attend or sponsor accessibility and assistive technology-related events to promote our application and engage with potential users and partners.

- User Communities: Create online forums or communities for users to discuss their experiences, seek help, and provide feedback. Engage actively in these communities to establish a loyal user base.

2.6.5. Requirement Gathering

To make sure that the app satisfies the demands of the users, there are a few procedures that must be taken when gathering requirements for a Sinhala book reader app for blind users. Here are some ideas for how to obtain the specifications for this kind of app: Identify the target user group: We identified the target user group for the app. The target audience in this scenario would be readers who speak Sinhalese but are visually challenged. We visited the Ceylon School for the Deaf and Blind and identified a sample group for blind users. Conduct user interviews: Doing user interviews with Sinhala-speaking visually impaired people will provide us with important insights into what they want from a book reader software. The interview could touch on subjects like their favorite genres of books, the features they would want to see in a book reader app, and how they prefer to read. Gather feedback from advocacy groups: There are advocacy people that concentrate on the requirements of people who are blind. Getting input from these groups will help us understand what features and functions are crucial for the app. So, we conducted interviews for some of the teachers who teach blind children. Conduct competitor analysis: It would be beneficial to examine current book reader apps, particularly those made for users who are visually impaired, to determine the features and functionalities that are already on the market. Finding market gaps that the new app might cover may be made easier with the aid of this investigation.

2.6.6. Requirement Gathering

To make sure that the app satisfies the demands of the users, there are a few procedures that must be taken when gathering requirements for a Sinhala book reader app for blind users. Here are some ideas for how to obtain the specifications for this kind of app: Identify the target user group: We identified the target user group for the app. The target audience in this scenario would be readers who speak Sinhalese but are visually challenged. We visited the Ceylon School for the Deaf and Blind and identified a sample group for blind users. Conduct user interviews: Doing user interviews with Sinhala-speaking visually impaired people will provide us with important insights into what they want from a book reader software. The interview could touch on subjects like their favorite genres of books, the features they would want to see in a book reader app, and how they prefer to read. Gather feedback from advocacy groups: There are advocacy people that concentrate on the requirements of people who are blind. Getting input from these groups will help us understand what features and functions are crucial for the app. So, we conducted interviews for some of the teachers who teach blind children. Conduct competitor analysis: It would be beneficial to examine current book reader apps,

particularly those made for users who are visually impaired, to determine the features and functionalities that are already on the market. Finding market gaps that the new app might cover may be made easier with the aid of this investigation.

Chapter 03

3. Methodology

3.1. Introduction

This chapter gives an overview of the software implementation process of the design and provides a detailed discussion about the technologies and tools used.

3.2. Overview of the Proposal Work

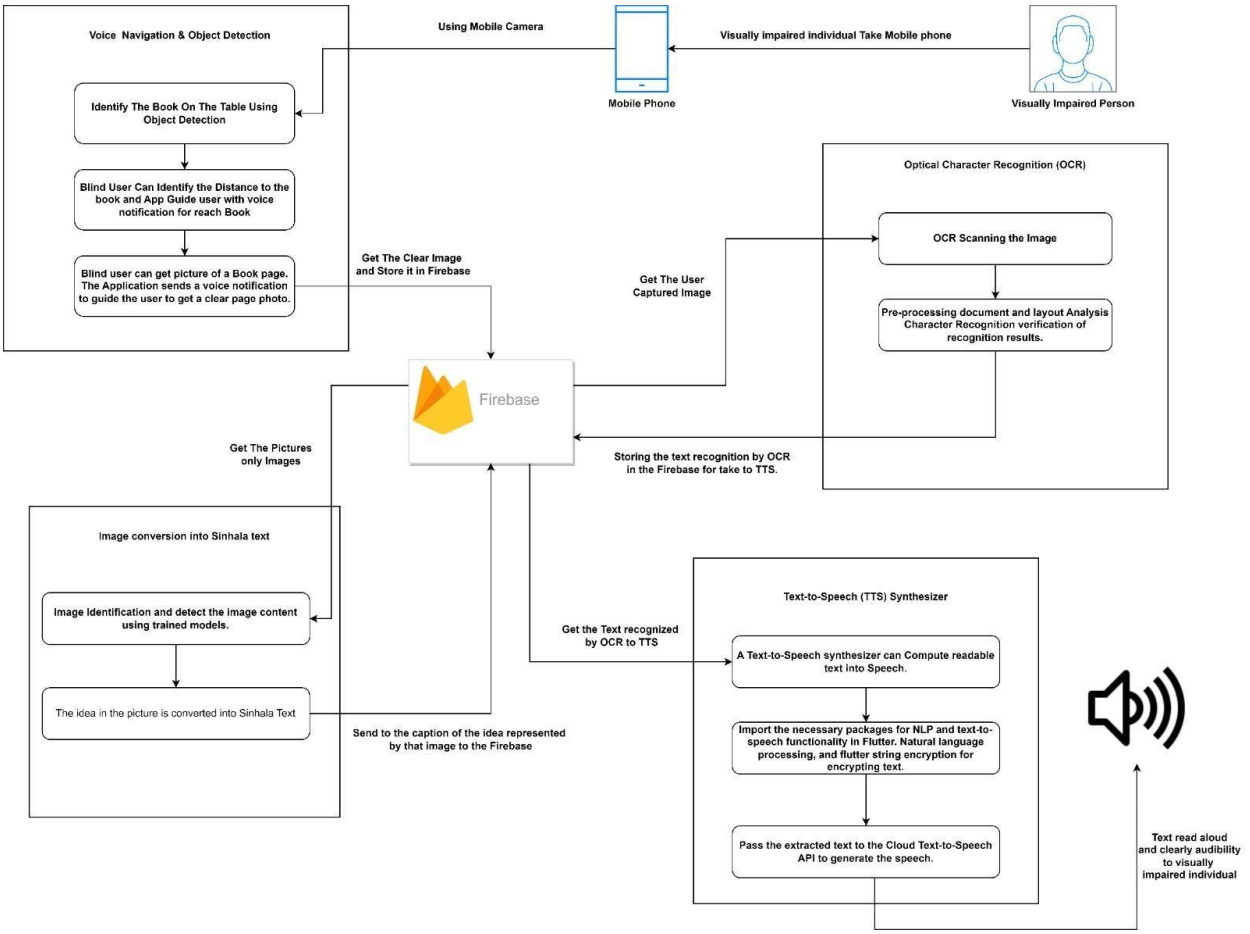


Figure 4:Overall system diagram

Audible guidance helps the user navigate the app through the functions of the app and get a clear idea and guidance whenever the user faces difficulty performing a task. When a user wants to find a book on the table, he opens the camera through the app and points it toward the table or desk and the app identifies the user's hand and navigates to the book. Using real-time Image processing technology, identifying the dangerous object near the blind user and the

distance to the object will notify verbally to the user and identify the probability of an accident occurring. Users will navigate in a pristine environment evading dangers and harmful things.

Sinhala Character Identification and word formation through the engine and translated to the speech then sent to the TTS. When the app is not running, users can read the time using a background process. The program should also be able to start the camera when the user launches it using voice commands. The program should be able to quickly scan the document when the user launches it. On the paper in front of the camera, the app needs to be able to automatically focus. Until the document is within the capture frame, the system should alert the user audibly. When the user wants to capture an image on the book the app alerts and navigates the user to capture the image onto the frame of the phone. The device's storage should be used to store the image that was captured. Before submitting information to the OCR system, the system must detect and correct skew.

A computer vision system that uses cutting-edge methods to recognize and describe objects and scenes in real time is an image detection software for blind students. The program takes pictures of the user's surroundings using the camera on a smartphone or tablet. The objects and their characteristics are then identified from these photos using image processing techniques like edge detection, color analysis, and feature extraction. After locating the things in the image with the use of object detection algorithms, machine learning models are used to identify and categorize the objects. These models use methods like convolutional neural networks (CNNs) to recognize objects in real time and are trained on massive datasets of annotated photos.

Here, the main purpose of using Text-to-speech (TTS) technology is to give a blind person the ability to access the written text of a Sinhala book. This allows them to easily listen to the valuable content of Sinhala books. TTS technology allows the written text in a Sinhala book to be read out loud in a natural-sounding voice, which makes it easier for visually impaired people to understand the content. The technology uses computer algorithms to analyze the Sinhala text and generate an appropriate pronunciation, intonation, and rhythm for each word and sentence.

3.3. Individual Contribution

3.3.1. Optical Character Recognition (OCR)

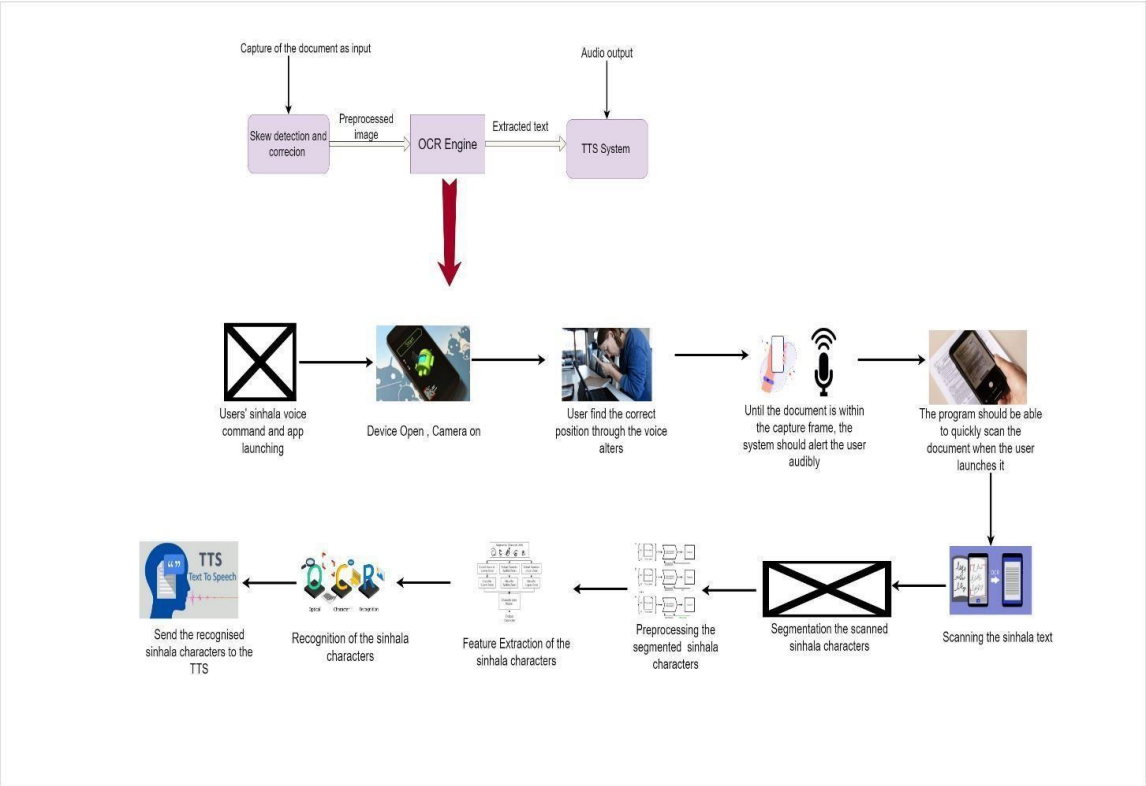


Figure 5:OCR system diagram

Developing a Sinhala OCR (Optical Character Recognition) methodology involves a series of systematic steps: beginning with image acquisition and enhancement to improve image quality, followed by character segmentation to identify individual Sinhala characters within the images. Feature extraction selects critical attributes from the segmented characters, creating feature vectors. Character recognition leverages machine learning algorithms, utilizing a dataset of known Sinhala characters for training, and subsequently assigning character labels based on extracted features. Post-processing steps may include language models and dictionaries to refine recognition results. The output is machine-readable Sinhala text, which can be stored digitally or converted into speech for accessibility. Continuous accuracy evaluation and iterative improvement, often driven by user feedback, refine the OCR system's performance, with the aim of accurately transcribing printed Sinhala text into a digital format for diverse applications, including assisting visually impaired individuals and facilitating information retrieval.

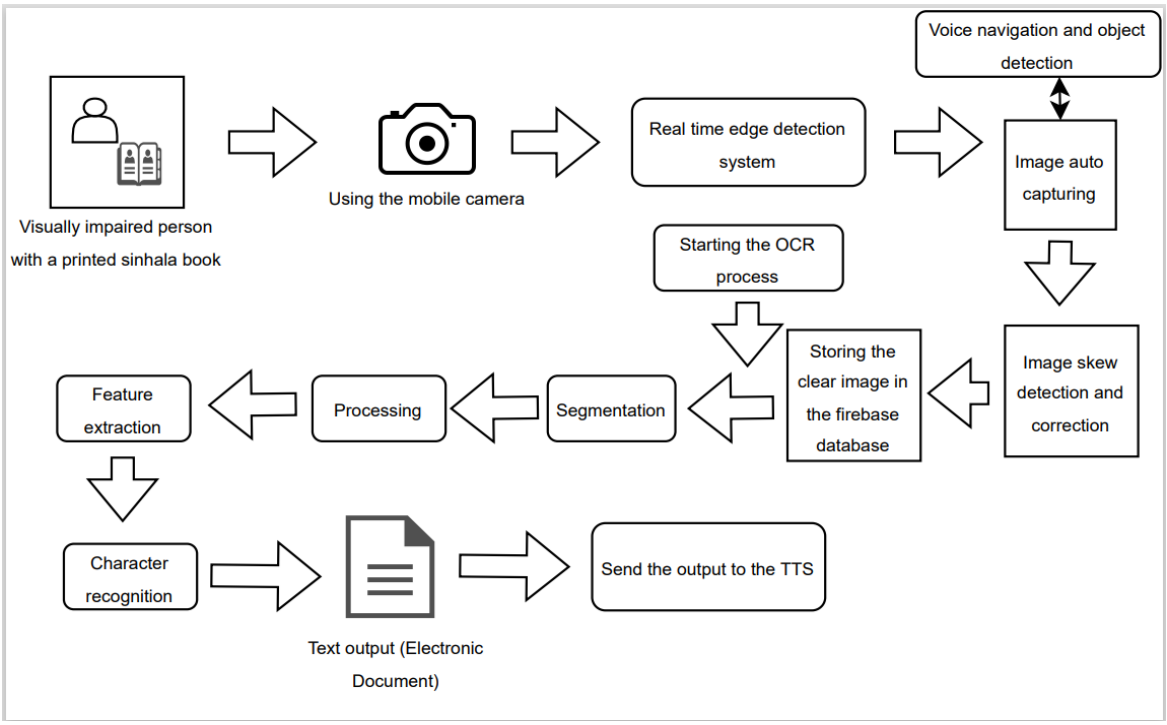


Figure 6:OCR System diagram Structural

First and the most I have collected different types of Sinhala images and made many data set as my dataset. This dataset is about 800 printed images of Sinhala sentences and Sinhala words.

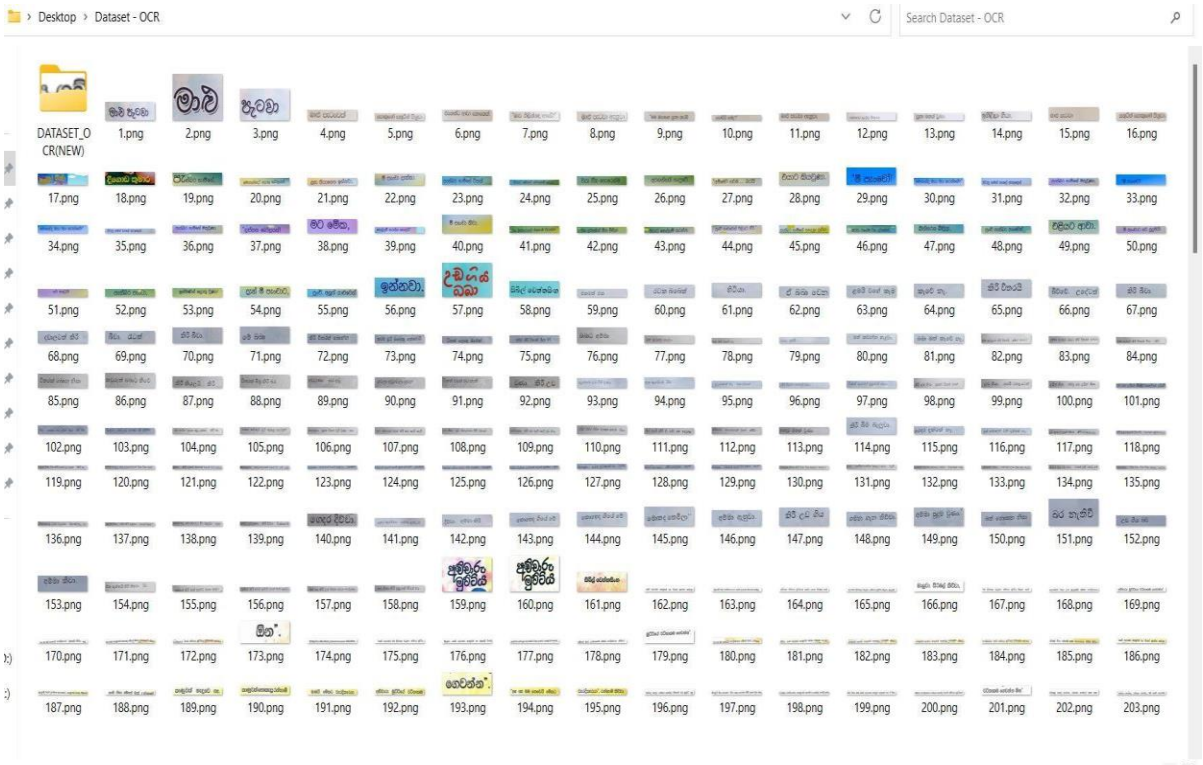


Figure 7:Customized OCR dataset

Then I made another dataset as the labeling of the collected dataset.

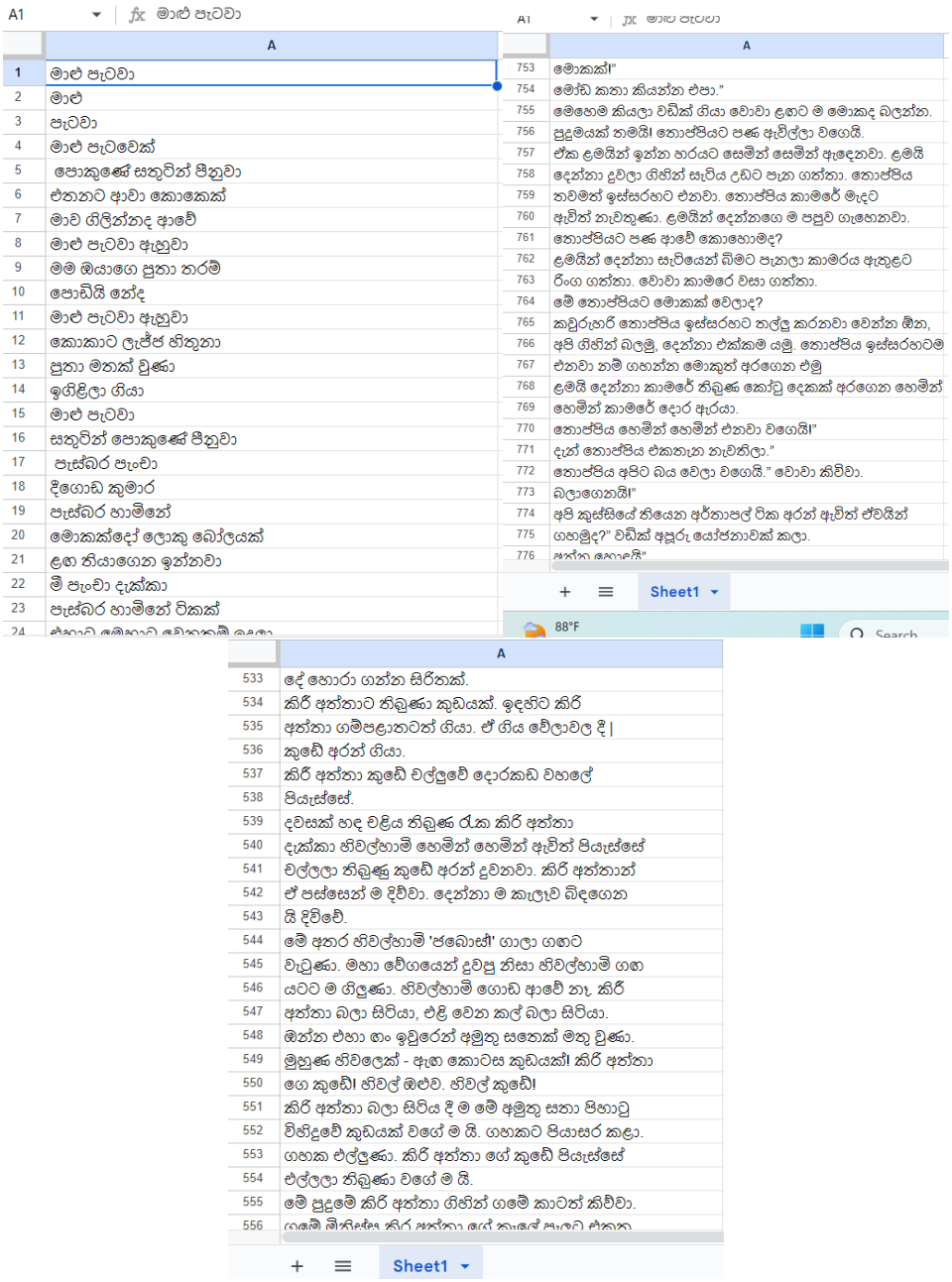


Figure 8:Customized Labeled Dataset

Here the dataset processing and training part: -

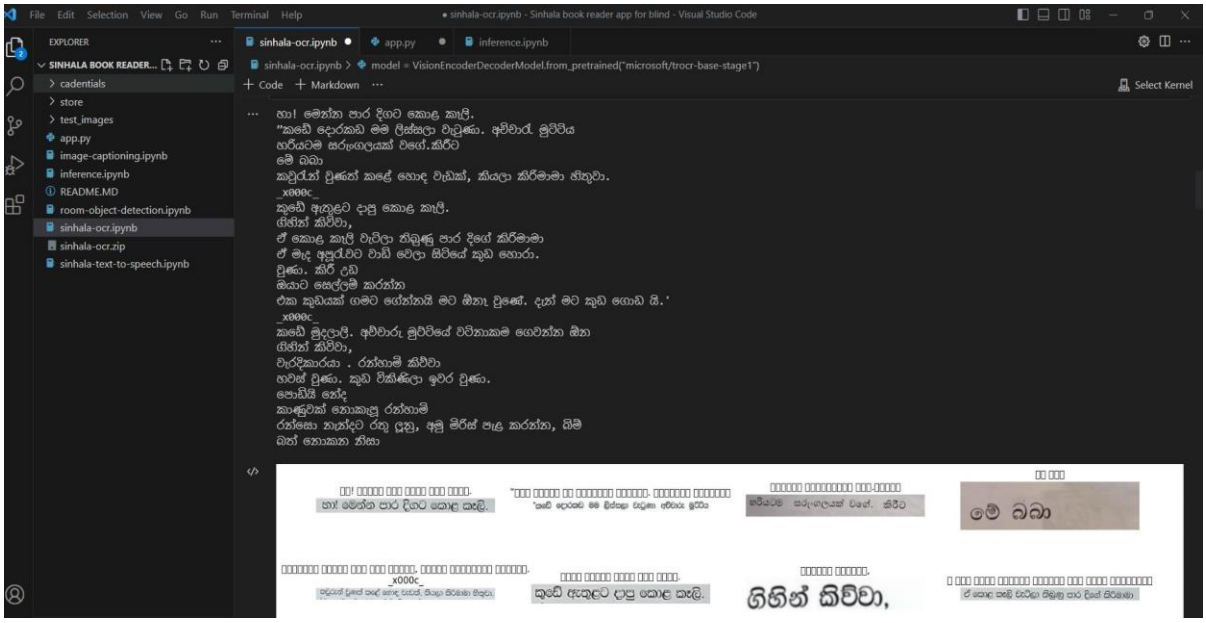


Figure 9:Dataset processing

Creating a customized dataset for OCR (Optical Character Recognition) involves several key steps to ensure the dataset is well-structured, diverse, and representative of the data you want to recognize. Here's a step-by-step guide on how to create a custom OCR dataset:

1. Define Dataset's Purpose:
    - Determine the specific objectives of the OCR system. What type of text or characters want to be recognized? Are you focusing on printed text, handwriting, or a combination of both? Knowing the dataset's purpose is essential for gathering the right data.
2. Data Collection:
    - Gather a diverse set of data that represents the variability you expect your OCR system to encounter in real- world scenarios. This may include: Printed text samples from books, magazines, and newspapers, Text samples in different fonts, sizes, and styles. Text samples with varying backgrounds and noise levels.
3. Data Preprocessing:
    - Prepare and preprocess the collected data to ensure consistency and quality: Resize and standardize image dimensions. Normalize image contrast and brightness. Convert images to a common format (e.g., JPEG or PNG). Remove unnecessary noise or artifacts from the images.
4. Annotation and Labeling:
    - Annotate each image to indicate the ground truth. This involves specifying the text content present in each image. Annotations should be accurate and complete. Store the

annotations in a structured format, such as XML, JSON, or CSV, linking each image to its associated text.

5. Data Augmentation (Optional):
    - To increase dataset size and improve model robustness, consider applying data augmentation techniques such as rotation, scaling, flipping, and adding noise to create variations of your existing data.
6. Splitting the Dataset:
    - Divide your dataset into distinct subsets for training, validation, and testing. Common splits include 70% for training, 15% for validation, and 15% for testing. Ensure that each subset maintains the same distribution of data characteristics.
7. Dataset Balance:
    - Ensure that your dataset is balanced, meaning it has an even representation of different classes or characteristics you want to recognize. This helps prevent bias in the OCR system.
8. Quality Control:
    - Review the dataset for any labeling errors, inconsistencies, or outliers. Manually verify a random sample of images to ensure that annotations match the image content accurately.
9. Dataset Size:
    - The size of your dataset depends on the complexity of the recognition task. For deep learning-based OCR models, larger datasets are often beneficial. Aim for a dataset size that provides sufficient diversity and generalization.
10. Documentation:
    - Maintain clear documentation of your dataset, including details about data sources, image preprocessing steps, and annotation guidelines. Proper documentation ensures reproducibility and helps others understand your dataset.
11. Ethical Considerations:
    - Be mindful of privacy and copyright issues when collecting and using data. Ensure you have the necessary permissions to use any copyrighted or sensitive material.
12. Sharing and Collaboration (Optional):
    - If appropriate, consider sharing your dataset with the research community to encourage collaboration and further advancements in OCR technology. Creating a custom OCR dataset requires careful planning and attention to detail. The quality and diversity of your dataset play a crucial role in the success of your OCR system. Regularly update

and refine your dataset as needed to improve model performance and adapt to changing recognition requirements.

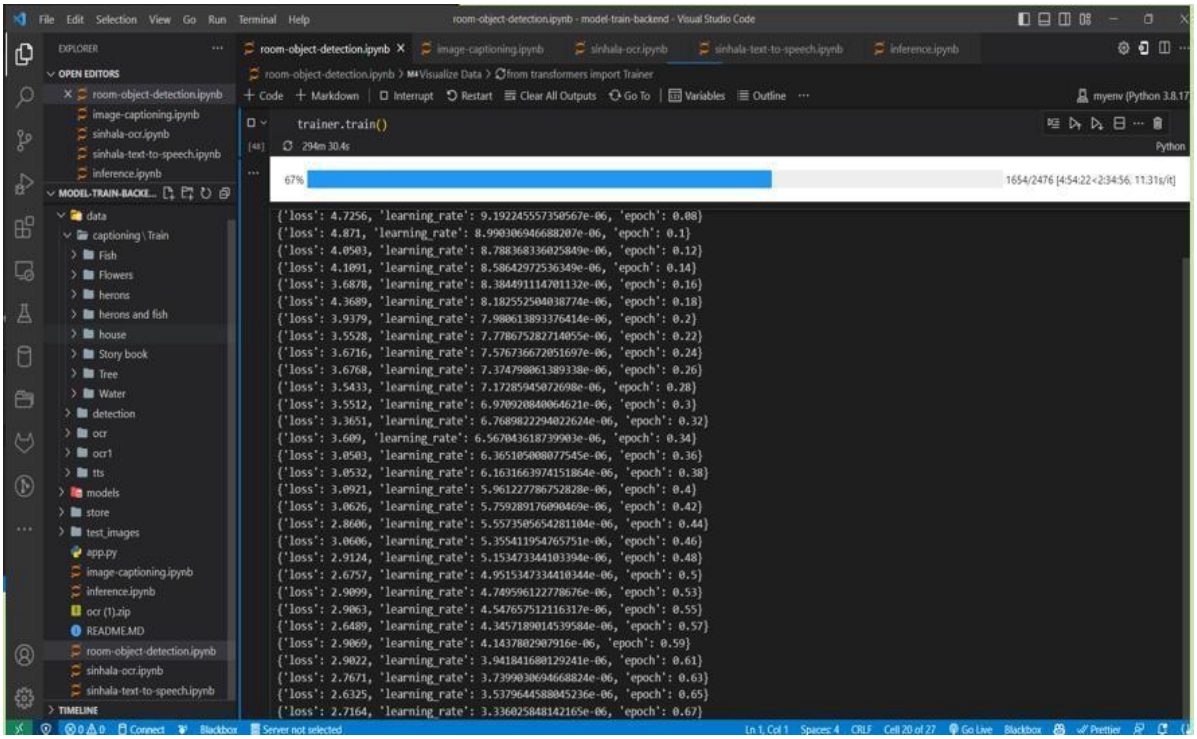


Figure 10:Data Training

```
subject='OCR'
acc=model.evaluate( test_gen, verbose=1, steps=test_steps, return_dict=False)[1]*100
msg=f'accuracy on the test set is {acc:5.2f} %'
print_in_color(msg, (0,255,0),(55,65,80))
generator=train_gen
scale = 1
model_save_loc, csv_save_loc=saver(working_dir, model, model_name, subject, acc, img_size, scale,  generator)
```

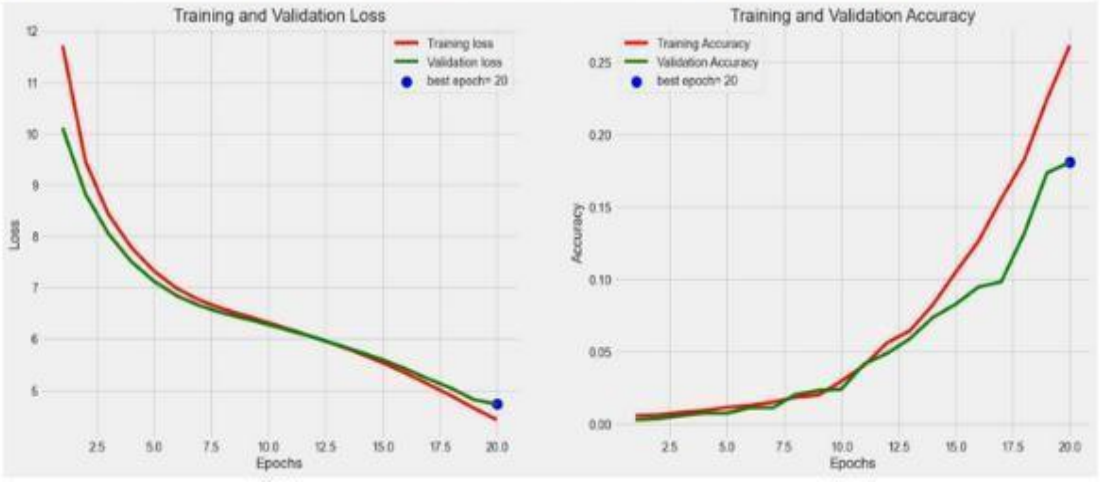


Figure 11:OCR Accuracy Chart

Figure 12:Used Models

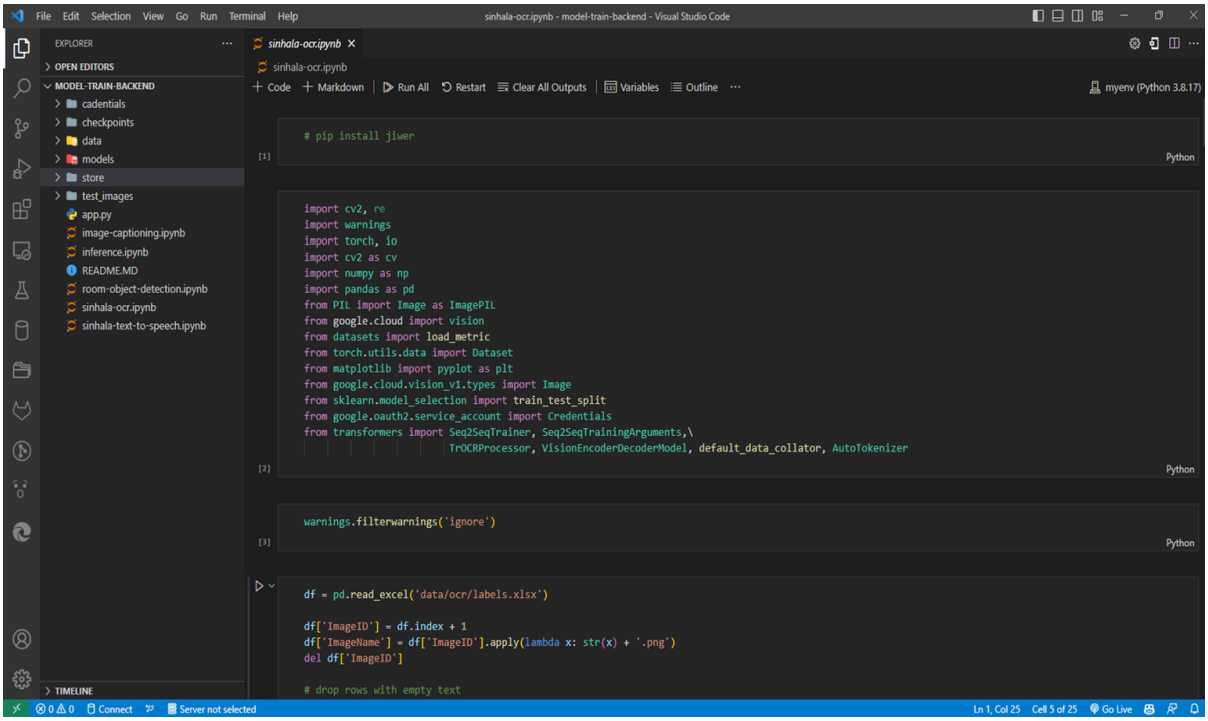


Figure 13:Imported libraries

Explanation of the libraries, calling APIs'

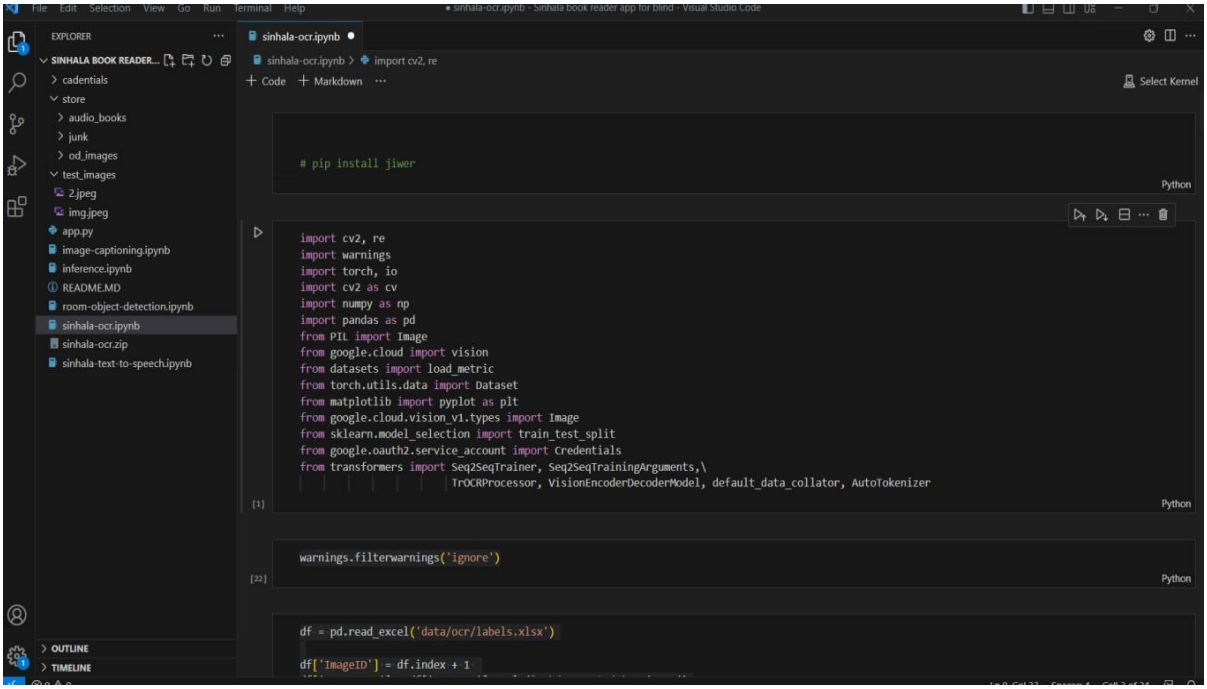


Figure 14:Libraries

Here several libraries are being imported and utilized for various tasks related to computer vision, natural language processing, and machine learning. The code begins by importing standard libraries like cv2 (OpenCV), warnings, torch (PyTorch), io, numpy, pandas, and re (regular expressions) to handle image processing, data manipulation, and text operations. Next, the code imports the Image module from both PIL (Python Imaging Library) and google.cloud.vision for image handling and processing.

For dataset manipulation and metric evaluation, it uses datasets to load metrics and defines a custom dataset class using torch.utils.data.Dataset. To visualize images, the code relies on matplotlib.pyplot.

For integration with Google Cloud Vision API and authentication, it imports google.oauth2.service_account.Credentials. Finally, the code utilizes the Hugging Face Transformers library,importing Seq2SeqTrainer, Seq2SeqTrainingArguments, TrOCRProcessor, VisionEncoderDecoderModel, default_data_collator, and AutoTokenizer. These components are essential for training and fine-tuning sequence-to-sequence models, particularly for Optical Character Recognition (OCR) tasks, using pre-trained models and custom data processing.

Overall, these libraries and modules come together to create a comprehensive environment for working with image-based text recognition, combining computer vision and natural language processing techniques within the context of machine learning.

3.3.1.1. Tools and technologies

Developing an OCR (Optical Character Recognition) system involves a combination of tools and technologies that enable the recognition and conversion of printed or handwritten text into machine-readable text. Here are the key tools and technologies commonly used in OCR development:

- OpenCV (Open-Source Computer Vision Library): OpenCV is used for image acquisition, preprocessing, and basic image manipulation in OCR applications. It can help with tasks such as image enhancement, noise reduction, and image binarization.
- TrOCRProcessor : This processor is specifically tailored to handle OCR-related data, making it easier for developers to prepare their data for training, fine-tuning, or inference with OCR models.
- Tesseract OCR: Tesseract OCR is often integrated into OCR systems to perform character recognition on preprocessed images. It provides an API for easy integration into various programming languages.
- Google Cloud Vision API: Google Cloud Vision API offers cloud-based image analysis and OCR capabilities with high accuracy.
- Jupyter Notebook: This interactive development environment for running Python code, making them ideal for prototyping and experimenting with OCR models. Use these environments to write, test, and refine OCR algorithms, as well as visualize results.
- AutoML and Hyperparameter Tuning Tools: Automated Machine Learning (AutoML) tools and hyperparameter tuning platforms can automate the process of model selection and optimization, saving time and effort. These tools can be used to find the best OCR model architecture and hyperparameters for a specific task.
- Version Control Systems (e.g., Git): Version control systems help manage code repositories, track changes, and collaborate with team members. It is essential for maintaining codebase integrity and collaborating on OCR project development.

3.3.2. Text-to-Speech (TTS)

We've collected a large dataset of roughly 3,300 voice WAV files to help us improve the development of our Sinhala text-to-speech (TTS) technology within our mobile-based Sinhala book reader for visually impaired people. This dataset, obtained from a reliable GitHub repository, is a critical asset in our model training efforts. These voice recordings include a wide range of Sinhala language variances and subtleties, which are critical for developing a TTS system that is truly appealing to our target audience.

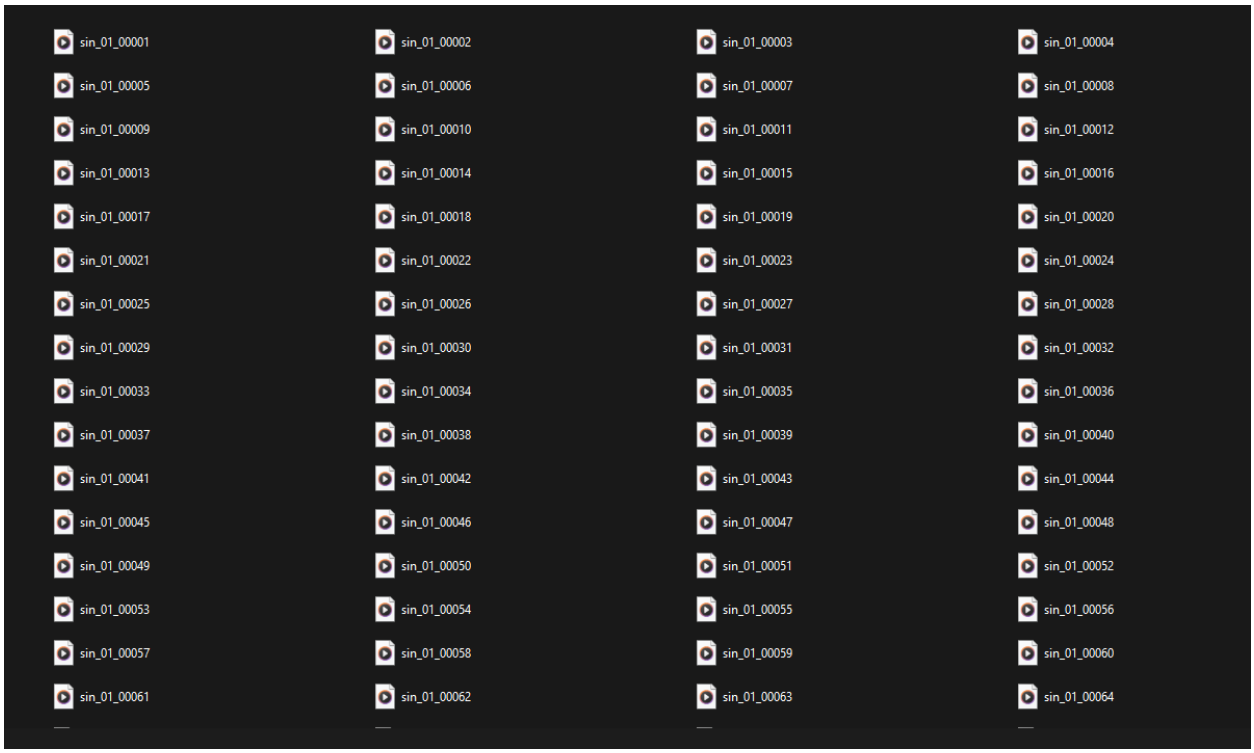


Figure 15:Text to Speech Dataset Collection

The use of this large dataset allows us to begin complete model training, allowing our TTS system to understand the nuances of Sinhala pronunciation, intonation, and rhythm. With this wealth of data at our disposal, we are well-positioned to create a sophisticated TTS model that not only improves the accessibility of our mobile-based book reader but also caters especially to the linguistic demands of visually impaired Sinhala-speaking persons. We think that including these voice WAV files will considerably improve the quality and naturalness of the synthetic speech, thereby improving the user experience and fostering more inclusion for our visually impaired users.

Speech is the principal mode of human communication, employing a complex system that mixes words and names from large vocabularies in a syntactic framework. Each spoken word is made up of a small number of phonetically related vowel and consonant speech sound components [13]. Tens of thousands of different and mutually unintelligible linguistic systems originate from the global variety of human languages.

The choice of interface is critical to user success in the field of human-machine interaction. Voice user interfaces, powered by voice recognition and synthesizing technologies, have grown in popularity. Voice communication is the most basic means of human contact, involving spoken verbal expression. It is the most natural and effective way for people to exchange their expertise.

Voice processing is a discipline that studies the analysis and manipulation of spoken signals. It is widely used as a front-end component in a variety of language processing systems. The scope of speech processing includes a wide range of issues, such as:

- Speaker Identification: This involves recognizing and distinguishing individual speakers based on their unique vocal characteristics. It has applications in security and authentication systems.
- Speech Identification: Speech recognition systems are designed to recognize and transcribe spoken words into written form. This technology lies at the heart of voice assistants and transcribing services.
- Speech Coding: Speech coding is concerned with efficiently compressing and encoding voice signals for storage and transmission, which is useful in telecommunications and multimedia applications.
- Speech Enrichment: Speech enrichment refers to techniques for improving the quality and intelligibility of speech signals, which benefit applications such as hearing aids and audio restoration.
- Speech Compression: This pertains to the reduction of data size in speech signals while maintaining perceptual quality. It's crucial for efficient data transmission and storage.
- Speech Synthesis: Speech synthesis involves generating artificial speech from text or symbolic representations, enabling applications like text-to-speech (TTS) systems and voice assistants.

In essence, speech processing is a multidisciplinary discipline that bridges the gap between human communication and technology, improving interactions and enabling a diverse range of applications in a variety of domains.

3.3.2.1. Speech Synthesis

Text-to-speech systems must first convert input text into lexical or phonological interpretations before generating the matching sounds associated with these representations. Given that the input is often in plain text format, linguistic models must be enhanced with insights into elements such as pitch and tempo to guarantee that the synthesized speech reflects natural patterns. A Natural Language Processing (NLP) module, which is a fundamental component of most speech processors, manages this fine-tuning of prosody and linguistic subtleties.

The NLP module functions as a text analysis behemoth, comprehending the complexities of the supplied text [6]. It considers the semantic and syntactic context, as well as an understanding of tone, rhythm, and language structures. The NLP module, armed with this extensive information, enables the synthesis process to sound more realistic and human-like.

Following that, an Artificial Sound Generation phase takes center stage, which is powered by a Digital Signal Processing (DSP) module. This module makes use of the NLP modules precisely obtained prosodic and linguistic data. It converts this data into artificial sound by painstakingly constructing speech waveforms that closely resemble the intricacies of human speech. Text-to-speech systems bring to life the translation of plain text into expressive and understandable synthetic speech by integrating these crucial components, a feat accomplished through the harmonic interplay of language analysis and powerful signal processing.

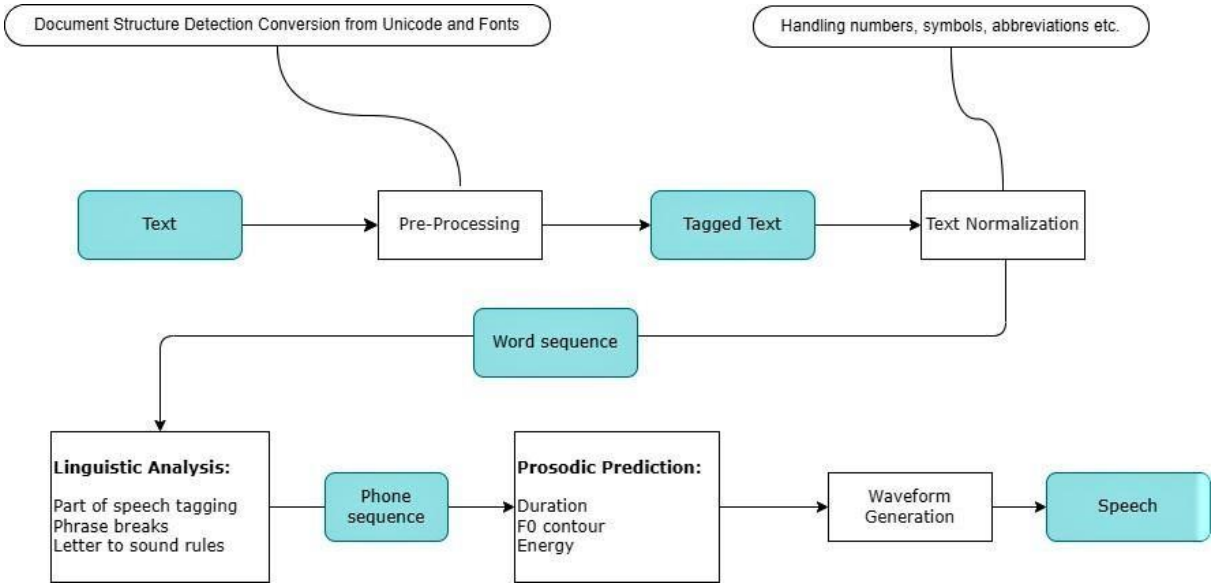


Figure 16:Text-to-Speech Synthesis System Architecture

There are two types of text-to-speech (TTS) systems: restricted domain TTS and generic TTS.

- Restricted Domain TTS: The first type, Restricted Domain TTS, is intended for a specific, predetermined purpose. As a result, the value of this sort of TTS is restricted to its intended purpose. It works by utilizing a preset collection of words and phrases designed specifically for voice synthesis. Restricted Domain TTS can be found in applications such as talking dictionaries, travel information systems, talking clocks, and other similar situations. In many cases, the TTS system is fine-tuned to meet the unique demands and vocabulary of the targeted application.

- Generic TTS: The second category, Generic TTS, is designed to accommodate a wider range of information. This adaptable TTS system can read a broad range of text formats, including internet news, emails, articles, and more. It can produce synthetic voices for any words or phrases, making it a good choice for general-purpose applications. Generic TTS systems enable the conversion of a wide range of written information into speech, meeting a wide range of user demands and preferences.

Restricted Domain TTS excels in specialized areas, but Generic TTS offers adaptability for a wide variety of applications and content kinds.

3.3.2.2. Individual System Diagram

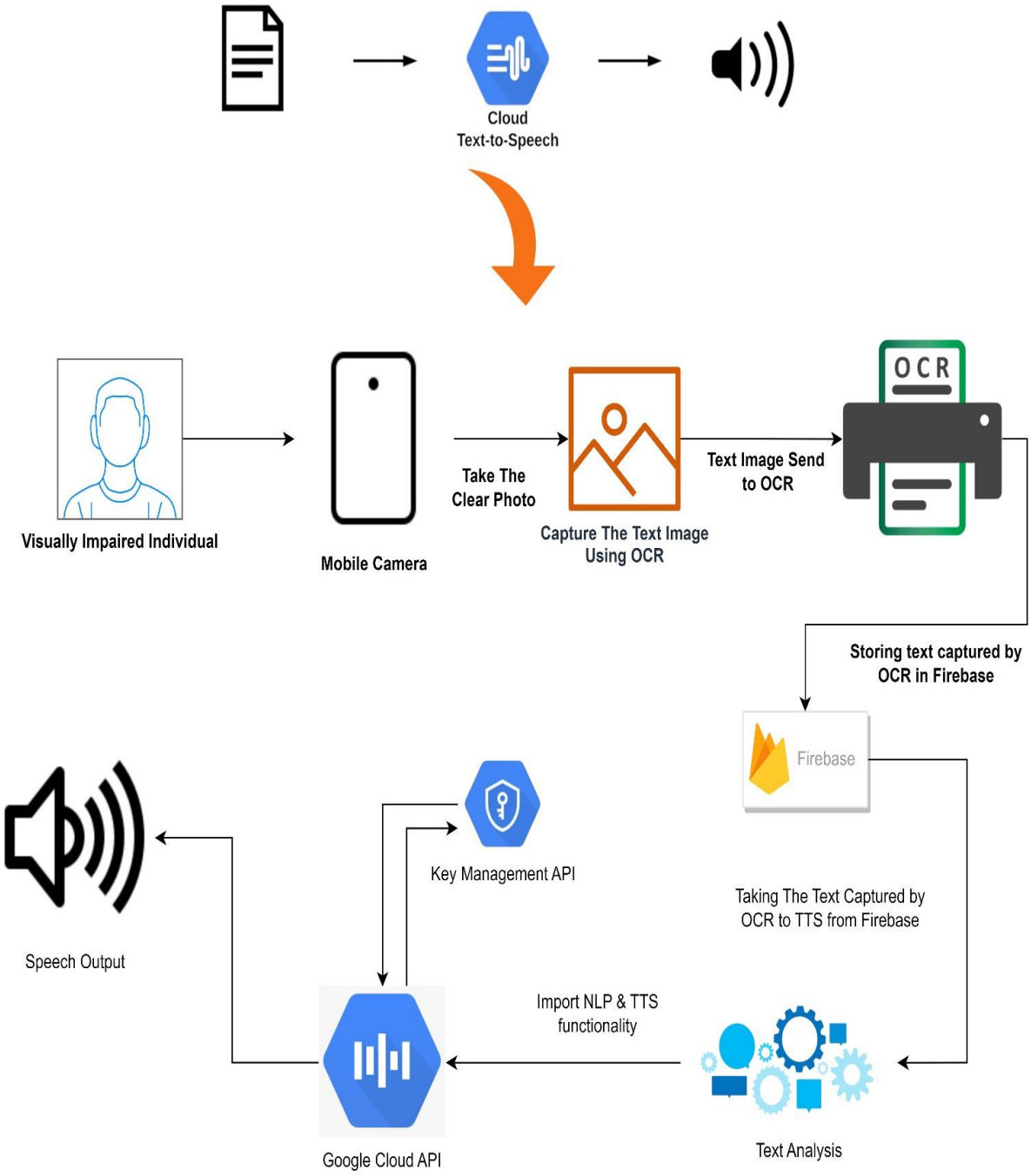


Figure 17:Individual System Diagram

A computer-based system known as a text-to-speech synthesizer is critical in converting computer-readable text into audible speech. This procedure includes multiple critical stages that may be divided into three basic phases: text analysis, linguistic analysis, and wave-form generation.

The procedure is as follows:

- A visually impaired individual is instantly told vocally upon completion of OCR, ensuring they are aware of the task's completion.
- At this point, the system passes the extracted and recognized Sinhala characters from the scanned text onto the TTS system for processing.
- The TTS system then takes center stage, utilizing its ability to vocally recreate the Sinhala text from the camera picture, making it accessible to blind folks.

In summary, the goal of incorporating TTS into the Sinhala language book reader for the visually impaired is to offer them a way to access written information and fully understand the content of books while overcoming the hurdles imposed by their visual disability. This technology enables people to successfully engage with books and information, fostering diversity and accessibility.

3.3.3. Requirement Analysis

A critical stage in our development process is requirement analysis for our mobile-based Sinhala text-to-speech (TTS) system within the Sinhala book reader for visually impaired users. The first component of our requirement analysis is a thorough study of the subtleties and nuances of the Sinhala language to achieve proper pronunciation and natural-sounding speech synthesis.

In addition, to improve the user experience for those with visual impairments, we prioritize accessibility elements including intuitive user interfaces and interoperability with multiple mobile devices. We also realize the significance of incorporating excellent conversation management technologies to enable user interactions, allowing our application to serve as both a reader and an intelligent helper for programming-related activities. Our in-depth requirement research seeks to link our development efforts with the specific demands of our target audience, thereby creating inclusion and accessibility in the field of Sinhala literature for the visually impaired.

3.3.4. System Analysis

System analysis is critical in the development of our mobile-based Sinhala book reader for visually impaired people, which includes Sinhala text-to-speech (TTS) capabilities. During this phase, we thoroughly examine our application’s requirements, functionality, and user demands. Understanding the specific issues that visually impaired users confront is critical because it influences the design and execution of user-friendly features. To provide a seamless and inclusive reading experience, we investigate user interfaces, navigation techniques, and accessibility standards.

Furthermore, system analysis comprises a thorough examination of Sinhala TTS technologies, with a focus on their accuracy, naturalness, and adaptation to the linguistic subtleties of our target audience. We establish the groundwork for a robust and user-centric Sinhala TTS book

reader by doing extensive system analysis, enhancing the lives of visually impaired folks via accessible literature and technology.

Techniques Used for Speech Synthesis

Speech synthesis, the fascinating discipline of reproducing human-like speech using technology, has evolved significantly with substantial technical advances. Voice synthesizers, which are among the most outstanding technologies in this sector, use computers' computational capacity to meticulously recreate the intricacies of spoken language [14]. Text-to-speech (TTS) is the most common and effective solution in this arena, beautifully converting written text into expressive speech that nearly mimics the richness of human vocalization. Furthermore, various methodologies within this discipline are investigating novel ways to describe symbolic linguistics, pushing the limits of what is possible in the world of artificial speech creation. These improvements are not only improving accessibility for people with speech difficulties, but they are also finding useful uses in areas such as entertainment, education, and customer service. The future of speech synthesis promises great prospects for more human-like and emotionally expressive voices as technology advances.

Among these strategies, concatenation stands out as a trailblazing strategy for producing artificial speech. This is accomplished by skillfully combining pre-recorded speech fragments drawn from large databases, resulting in cohesive and lifelike synthetic speech. This approach works in tandem with another effective strategy: the conversion of textual input into voice using Text-to-Speech (TTS) technology. The panorama of voice synthesis, on the other hand, stretches even farther, incorporating systems that cleverly transform symbolic language representations into captivating vocal renditions, pushing the frontiers of what is feasible in artificial speech production.

Concatenation, a popular approach for producing artificial speech, entails combining pre-recorded speech fragments from a database. Text-to-speech (TTS) is another famous technique that turns written text into speech. Aside from this, several systems investigate novel approaches for converting symbolic language representations into spoken speech [15]. Two fundamental technologies dominate the realm of speech synthesis: formant synthesis and concatenative synthesis. They both attempt to develop synthetic speech that closely mimics genuine human speech, demonstrating the many ways in which the area is evolving and improving.

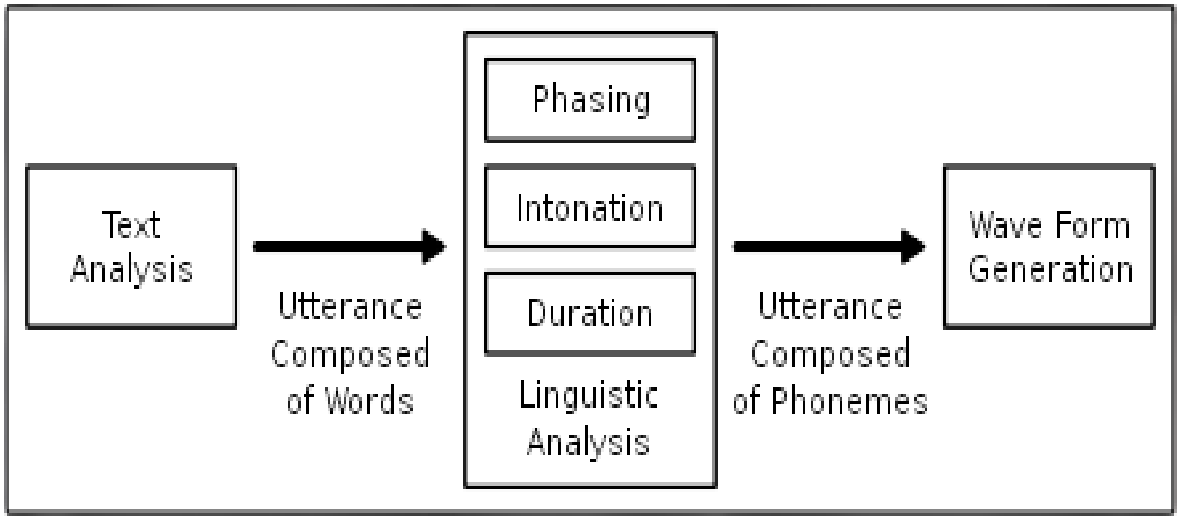


Figure 18:Text-to-Speech Wave Generation

TTS technology is a multidimensional system made up of two important components: the front end and the back end. The front end is critical because it performs two key tasks: text normalization and pre-processing. During the pre-processing step, the system recognizes individual words in the incoming text and turns them into phonetic transcriptions. Meanwhile, the text normalization step ensures that the incoming text follows a specified structure, allowing for more accurate and smoother voice synthesis. These two critical front-end procedures jointly establish the groundwork for TTS technology's effective functioning, allowing it to convert written text into natural sounding spoken language.

Text-to-speech (TTS) technology refines the input text further by segmenting it into prosodic units, which include phrases, sentences, and clauses [16], before transforming them into phonemes. This technique also entails identifying these prosodic units with pertinent prosody information, which captures details such as pitch, rhythm, and stress. The front end creates a symbolic language representation of the input text using this augmented data. This representation combines prosody information with phonetic transcriptions, which improves the TTS system's capacity to create expressive and lifelike synthetic speech that mimics the complexities of actual human conversation.

The front end precisely created symbolic language representation serves as the foundation for the back end, where actual sound synthesis occurs. This translation of the text's symbolic representation into audible sound is accomplished using a process suitably dubbed "synthesis." Synthesis is the critical process that transforms the symbolic representation into a cohesive and natural-sounding output, bringing artificial speech to life. The confluence of complicated algorithms and acoustic models joins here, at the back end, to generate synthetic speech that closely matches the rhythm, tone, and articulation of real voice.

- **Sinhala Consonant**

Sinhala, Sri Lanka's native language, has a distinct aural identity due to its 26 consonants, which are an important component of the language's phonetic inventory [16]. Figure 10 in the chart presents a thorough list of these sounds, emphasizing their importance in the linguistic composition of Sinhala. The retroflex sounds stand out among these consonants because they are spoken by bending the tongue backward, providing a distinct phonetic feature. The language also has hissing alveolar fricatives, which distinguishes it from others. Notably, Sinhala has a sequence of pre-nasalized voiced stops, which contributes to its phonetic profile and makes it a linguistically intriguing language.

		Labial	Dental	Alveolar	Retroflex	Palatal	Velar	glottal
Stops	Voiceless	p	t		ʈ		k	
	Voiced	b	d		ɖ		g	
Affricates	Voiceless					c		
	Voiced					ɟ		
Pre-nasalized voiced stops		ᵐb	ⁿd		ᶯɖ		ᵑg	
Nasals		m		n		ɲ	ŋ	
Trill				r				
Lateral				l				
Spirants		f	s			ś		h
Semivowels		v				y		

Figure 19:Spoken Sinhala Consonant Classification

The sound patterns inherent in Sinhala consonants are not only important to the language's identity, but also to its cultural past. These different phonetic traits have played an important part in the language's growth, allowing it to adapt to the changing terrain affected by the increased usage of English loanwords. As Sinhala incorporates these loanwords, its consonants continue to play an important role in molding the language's growth and integrating it with modern society. This dynamic combination between tradition and modernity demonstrates Sinhala's endurance and flexibility, demonstrating how it has developed to stay relevant in today's linguistic and cultural setting.

- Sinhala Vowel

Sinhala’s 14 vowels play an important role in defining the language’s distinct aural character [13]. These vowels, as seen in Figure 11, are an important part of Sinhala’s phonetic inventory. What distinguishes Sinhala vowels from vowels in other languages is their distinguishing qualities. Sinhala has a great degree of phonemic length difference between its vowels, which gives depth and subtlety to its spoken sound. The language also utilizes a sophisticated system of vowel harmony, which further enriches its phonetic landscape. These outstanding characteristics of Sinhala vowels contribute greatly to the language’s rich phonological heritage and undeniable aural beauty.

	Front		Central		Back	
	Short	long	Short	long	short	long
High	**i**	**i:**			**u**	**u:**
Mid	**e**	**e:**	**ə**	**ə:**	**o**	**o:**
Low	**æ**	**æ:**	**a**	**a:**		

Figure 20:Spoken Sinhala Vowel Classification

The vowels and consonants that make up the phonetic tapestry of the Sinhala language are representative of the language's rich cultural background. These linguistic characteristics not only show the language's historical roots but also its adaptation to modern conditions. They are important pillars in creating Sinhala's identity and heritage, demonstrating the language's ability to adapt while keeping its own characteristics. The rising assimilation of English loanwords is a noteworthy example of this linguistic development, demonstrating how Sinhala adopts new sounds and vocabulary while retaining its core identity.

Consonants and vowels work together to form the foundation of Sinhala's acoustic and linguistic uniqueness. They continue to play an important role in the language's evolution, acting as conduits for its cultural heritage. These phonetic components continue to play an important role in Sinhala, strengthening its linguistic legacy. As Sinhala adopts new sounds and vocabulary, such as the increasing use of English loanwords, it demonstrates its adaptability and resilience, balancing tradition, and modernity in an ever-changing linguistic context.

- Sinhala Character Set

Sinhala, Sri Lanka's official language, predominantly employs the Sinhala character set, which has a diverse linguistic palette. There are 40 consonants and 20 vowels in this set, each with unique features that contribute to the language's peculiar aural quality [10]. The usage of diacritics, which are symbols used to change the sounds generated by both consonants and vowels, further demonstrates the adaptability of these letters. It is possible to control and refine the composition of consonants and vowels by expertly adding or deleting diacritics, allowing for a subtle and expressive portrayal of the phonetic nuances of the Sinhala language.

A total of 18 diacritics are used in the Sinhala letter set to shape the language's phonetic subtleties. 17 of these diacritics are specially created as vowel modifiers, making them vital tools for differentiating Sinhala's varied array of vowels. They allow for the difference of nasalized vowels as well as long and short vowels, both of which contribute to the language's phonetic richness. The last diacritical mark has a specific function: it indicates the unmodified consonant form, completing the set and guaranteeing clarity in the depiction of Sinhala consonants and vowels.

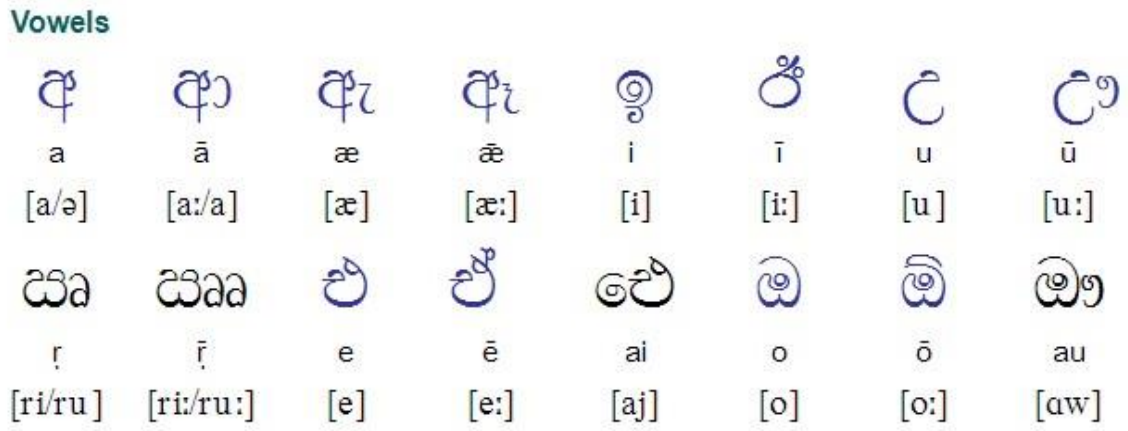


Figure 21:Sinhala Language 20 Vowels Table

Vowels are extremely important in the Sinhala language, playing a critical role in defining its distinct aural character. Each of the 20 vowels in the Sinhala letter set has unique features that add to the richness and complexity of the language. Sinhala vowels are classified into three types: short, long, and nasalized [16]. These many vowel kinds not only add to the distinctive sound of the language, but also give a nuanced and expressive framework for communication, reflecting the complexities of Sinhala's linguistic legacy.

To express distinct sorts of vowels in Sinhala, a systematic technique is used. Short vowels are represented by plain vowel symbols, but long vowels are represented with diacritical markings known as "pure." In addition, nasalized vowels are denoted by the niggahita tilde diacritic. The appropriate use of these diacritics is required to correctly discern and portray the subtleties of distinct vowels within the Sinhala language. This thorough use of diacritics improves the precision and clarity of written Sinhala, allowing readers and speakers to properly navigate its vast phonetic terrain.

The Sinhala letter set consists of 40 consonants, each with its own distinct qualities and sounds, making them essential components of the Sinhala language. These consonants are divided into two categories: pure consonants and compound consonants. These consonants' specific characteristics and phonetic properties add greatly to the depth and complexity of the Sinhala language, highlighting their critical significance in both written and spoken communication.

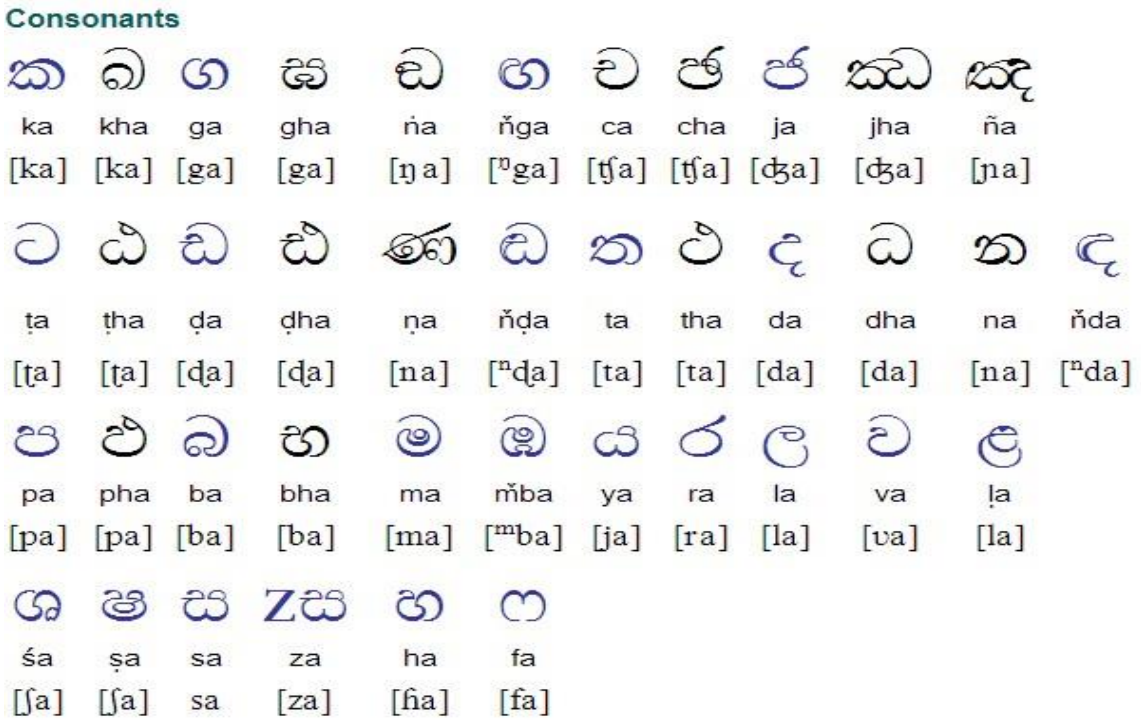


Figure 22:Sinhala Language Consonants Table

Pure consonants, which lack intrinsic vowels, constitute simple and distinct phonetic sounds in Sinhala. Compound consonants, which are generated by mixing consonants and vowels, serve as linguistic carriers for more complex sounds. The complicated interaction of pure and compound consonants, which is frequently adjusted by the addition or removal of diacritics, enables exact identification of consonant and vowel compositions [13]. This dynamic use of consonants and diacritics not only allows for the articulation of a broad range of phonetic subtleties, but it also emphasizes the intricate character of the Sinhala script, where consonants and vowels merge to produce a harmonious language system.

Vowels and consonants work together to produce the aural character of the Sinhala language. The Sinhala letter's set of 40 consonants spans a broad range of both fundamental and nuanced sounds, while the 20 vowels are essential for differentiating between the language's various phonetic variants. The correct use of diacritics is critical in identifying the composition of consonants and vowels, bringing depth and complexity to the Sinhala script [16].

Furthermore, the availability of text-to-speech alternatives has substantially improved the accessibility and inclusivity of learning and communication in the Sinhala language, particularly for those with visual impairments. These technological advances have created new opportunities for making Sinhala more accessible and user-friendly, guaranteeing that the beauty and depth of the language may be enjoyed and embraced by a larger audience.

3.3.5. Utilizing SpeechT5 Models

The quality of a Text-to-Speech (TTS) system depends on its ability to accurately imitate human speech and convey meaning effectively. However, if the TTS output lacks authentic expressions, the application's usefulness can be severely affected. This highlights a critical issue in developing TTS technology that can generate synthetic speech that accurately reflects human speech from text.

The primary goal of TTS technology is to mimic the full range of human speech, including different speech patterns, subtleties, and intonations while minimizing any mechanical or robotic aspects in the voice output.

The Sinhala Text to Speech (TTS) system was developed using the Transformer Model. It includes 3300 Sinhala words and their corresponding waves, as shown in Figure 23.

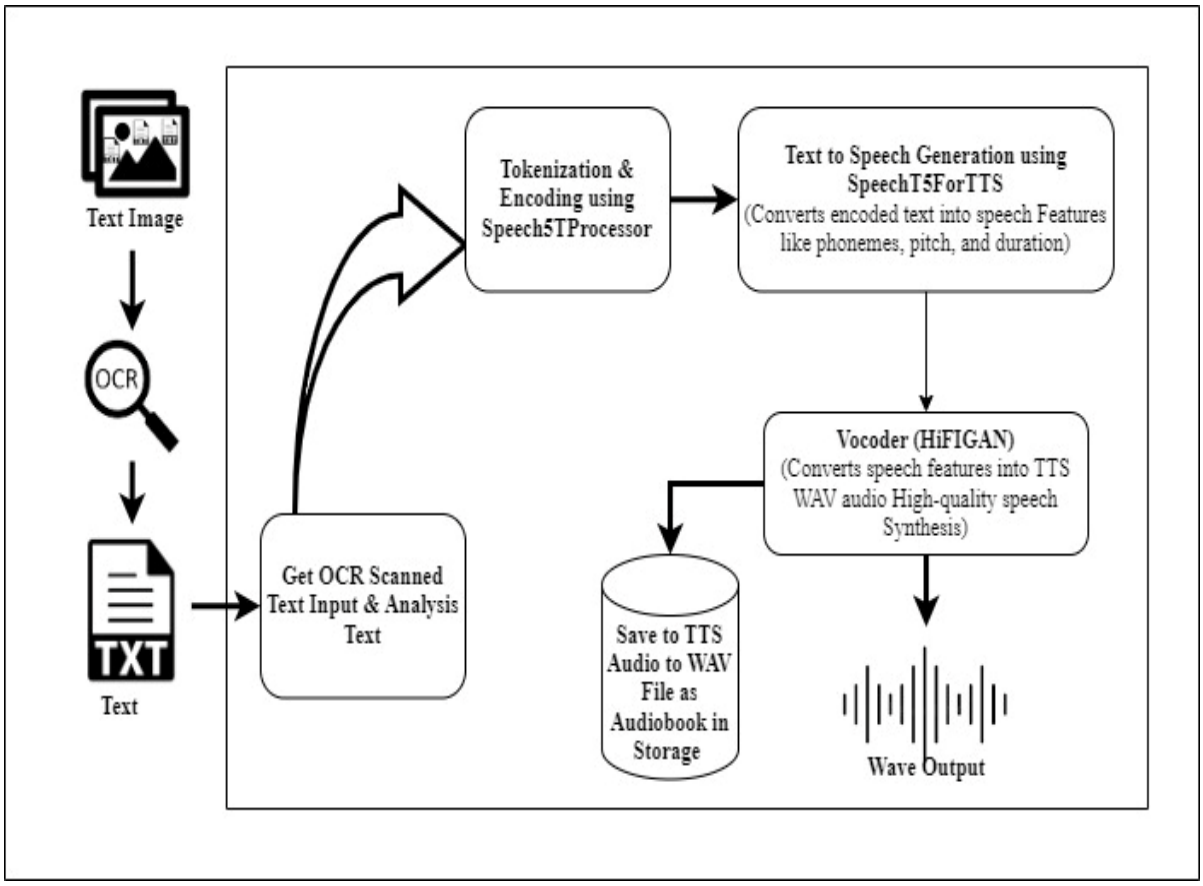


Figure 23:Utilizing SpeechT5 Models and the HiFIGAN Vocoder, produce Sinhala TTS Wavs

3.3.6. Design and Implementation

3.3.6.1. Data Preprocessing (Text and Audio)

To improve the text-to-speech experience, raw text with correct space and punctuation is required. During the preprocessing step, our major aim was to smoothly integrate the raw text input into the function of our system. By correcting mispronunciations, omissions, and redundancies, we were able to solve the issue of confusing and non-fluid spoken information. Our approach was developed by creating a preprocessing model that normalizes incoming text and removes impediments.

The developed preprocessing method included numerous critical transformations. To begin, we transformed all the input text to uppercase to ensure uniformity. Following that, we deleted intermediate punctuation marks one by one, simplifying the text for more effective linguistic analysis. To guarantee genuine prosody, each statement was then suitably punctuated with a question mark or a period.

We developed specific separator characters to distinguish between words, words with relaxed pronunciation, words with brief pauses, and spaces to address the issue of spaces introduced during speaking [17]. This flawless alignment of text and audio significantly minimized long silences and permitted exact synchronization, considerably improving auditory performance.

Our research has significantly improved text preprocessing, leading to more enriched text-to-speech outcomes. The proper format sliced audios are taken from public domains listed in TABLE 1.

TABLE 1:Preparing Audios Datasets

Functionalities	Sinhala
Name	“pathnirvana”
Audio Resource from	Kaggle & GitHub
File Format	.wav file
Audio Count	3300
Speaker Type	Single Speaker
Single audio Duration	1seconds to 10seconds
Text Resource from	Converted Sinhala font

3.3.6.2. Tokenization and Encoding

Figure 23 illustrates the process of converting OCR-scanned text into Text-to-Speech (TTS) WAV audio using SpeechT5 models and HiFIGAN vocoder. The input is OCR-scanned text, which can be obtained from physical documents or images. The text undergoes two steps, Tokenization and Encoding, to make it suitable for modeling. Tokenization involves segmenting the text into smaller units called tokens while encoding transforms these tokens into a numerical format that the models can understand.

3.3.6.3. Text-to-Speech Generation

Once the initial preprocessing stage is completed, the Text-to-Speech Generation process commences with the support of the SpeechT5ForTextToSpeech model, as illustrated in Figure 23. The model accepts the encoded text as input and produces speech features, including phonemes, pitch, and duration, that precisely capture the nuances of human speech.

3.3.6.4. VoCoder (HiFIGAN)

We use the Vocoder, specifically HiFIGAN, to convert the characteristics shown in Figure 23 into high-quality speech. The Vocoder interprets these qualities and produces a TTS WAV audio output that sounds natural and brings the text to life in spoken form.

3.3.6.5. Text-to-Speech Wave (Audio) Outputs

The outcome of the processing is the TTS WAV audio, which can be utilized in different ways as shown in Figure 23. For easy storage and sharing of the generated speech, it can be saved as an audio file. Otherwise, it can be instantly played through an audio player, enabling users to hear the TTS output in real time. This approach showcases the smooth transformation from written text to spoken language and the exceptional capabilities of the SpeechT5 models and HiFIGAN vocoder.

3.3.7. Encoder

The system primarily handles text input and uses an encoder to translate it into an internal representation. This representation records learning results through fully convolutional layers. The encoder is built to handle two types [18] of input: phoneme embeddings and phoneme stress embeddings. It converts phonemes or letters into vector representations, which can be further enhanced through training. The embedding percentage gradually becomes a fully linked structure after passing through a few convolutional blocks. Attention key vectors are then generated by referencing the embedding proportions. These key vectors are used within each attention block to determine attention weights. Finally, the context vector is calculated by merging the value vectors in a weighted fashion using a standard procedure. According to Figure 14, the following diagram shows how to encode texts in Sinhala TTS.

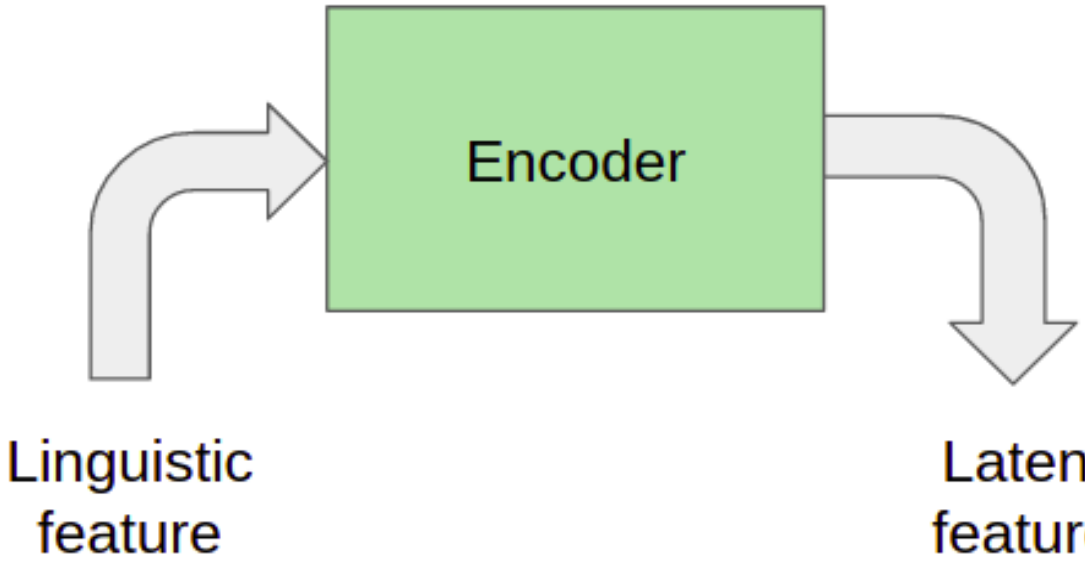


Figure 24:Encoder the Preprocessor Text in Sinhala TTS

The encoding procedure begins with embedding the input data and progresses through a chain of convolutional blocks from embeddings to a fully connected representation. Following that, the attention key vectors are constructed by looking back to the initial embedding proportions. These key vectors are used to calculate attention weights inside each attention block. The final context vector is calculated using a typical manner as a weighted aggregate of the value vectors.

The Encoder design is distinguished by a convolutional block with excellent connection to the input data. It also includes sequential gated components, which, curiously, do not rely on rigid sequence-based dependencies but rather feature certain scaling characteristics. To accommodate various sequence lengths, the input is supplemented with timestamps, allowing the Encoder to handle inputs with diverse temporal scopes successfully. Because of its adaptability, the Encoder is a reliable component for a broad range of applications, particularly those needing the extraction of significant characteristics and representations from textual or sequential input.

3.3.8. Decoder

The decoder's primary goal is to make predictions in an auto-regressive manner, successfully decoding the learned outcomes. The decoder is specifically developed to estimate future audio sequences based on previous audio files. The decoder is built as a fully convolutional neural network, with a strong emphasis on totally causal convolutions [18]. This design decision is critical in ensuring that predictions are created in a sequential and causal fashion, which means that each prediction is exclusively based on previous information, matching the natural flow of audio data.

This decoding procedure is based on transferring the attention mechanism of multi-hop convolutional networks into an audio representation of low-dimensional Mel-band spectrograms [11]. This modification enables the model to focus on essential audio elements when making predictions, improving its ability to grasp complex patterns and structures in audio data. Furthermore, the decoder operates on audio sequences that have been grouped together, which has a substantial influence on performance. Decoding numerous audio collections at the same time has been found to be more efficient than processing individual audio files separately. This method makes use of the contextual information contained in linked audio sequences, resulting in more accurate and coherent predictions. According to Figure 14, the following diagram shows how to decoder Audio and Texts in Sinhala TTS.

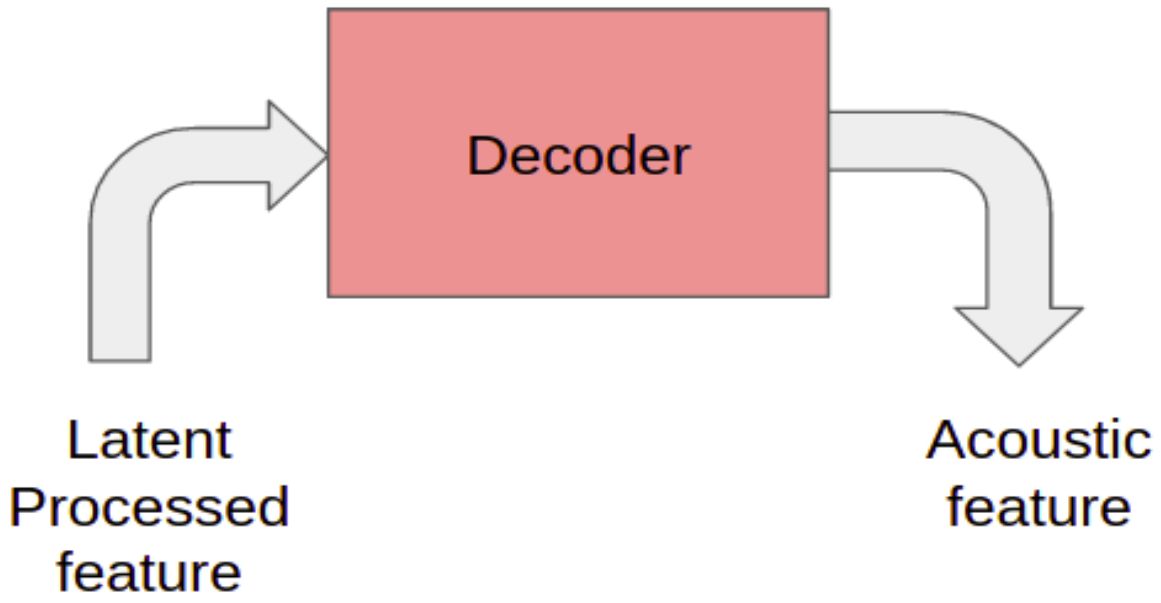


Figure 25:Decoder the Audios and Text in Sinhala TTS

Rectified Linear Units (ReLU), attention blocks, and output layers are all important components of the decoder network. The ReLU activation function covers nonlinearities in several fully linked layers, allowing the model to capture complicated data connections [11]. The attention blocks are arranged in a certain order to assist the decoder in focusing on relevant information from the encoder's output. The output layers are completely linked layers that forecast the next batch of audio frames. A binary wrapping procedure is used to the last frame to assure continuity. Dropout is selectively applied to all fully connected layers before the attention blocks, save the first layer, to reduce overfitting and increase model generalization. Using the output spectrograms, the model creates predictions for the missing data functionality, and its performance is assessed using the cross-entropy loss function, with a focus on the "done" prediction.

The decoder's hidden state is critical for its operation, especially when utilizing the dot-product attention technique. This technique computes the output vector, which is a weighted average, using the encoder's vectors at each time step. It's worth noting that the model only receives

audio data from a single speaker during training. The encoding position rate is set to one to maintain alignment with the decoder and remains constant about the encoder, allowing the model to successfully train and output coherent audio sequences.

The architecture of the decoder network, which includes ReLU activations, attention blocks, and output layers, is intended to reliably forecast future audio frames. Dropout techniques are used to improve model resilience, and the dot-product attention mechanism makes precise context-based predictions using the encoder's per-timestep vectors. This method works especially well when trained with data from a single speaker, resulting in coherent and high-quality audio creation.

3.3.9. Converter

The converter's principal job is to forecast the final output characteristics required by the post-processing network. This prediction is strongly reliant on the mechanism used to produce the waveform from the decoder's hidden states. The converter network is built as a fully convolutional architecture, with a strong emphasis on gathering context information relevant to the job at hand.

The converter network's input activations are obtained from the decoder's final hidden layer outputs [18]. The converter uses its non-causal and non-autoregressive convolution blocks to anticipate important vocoder parameters, based on the decoder's upcoming context. This contextual data is critical for making accurate predictions and producing high-quality audio outputs that accurately replicate the intended audio waveforms.

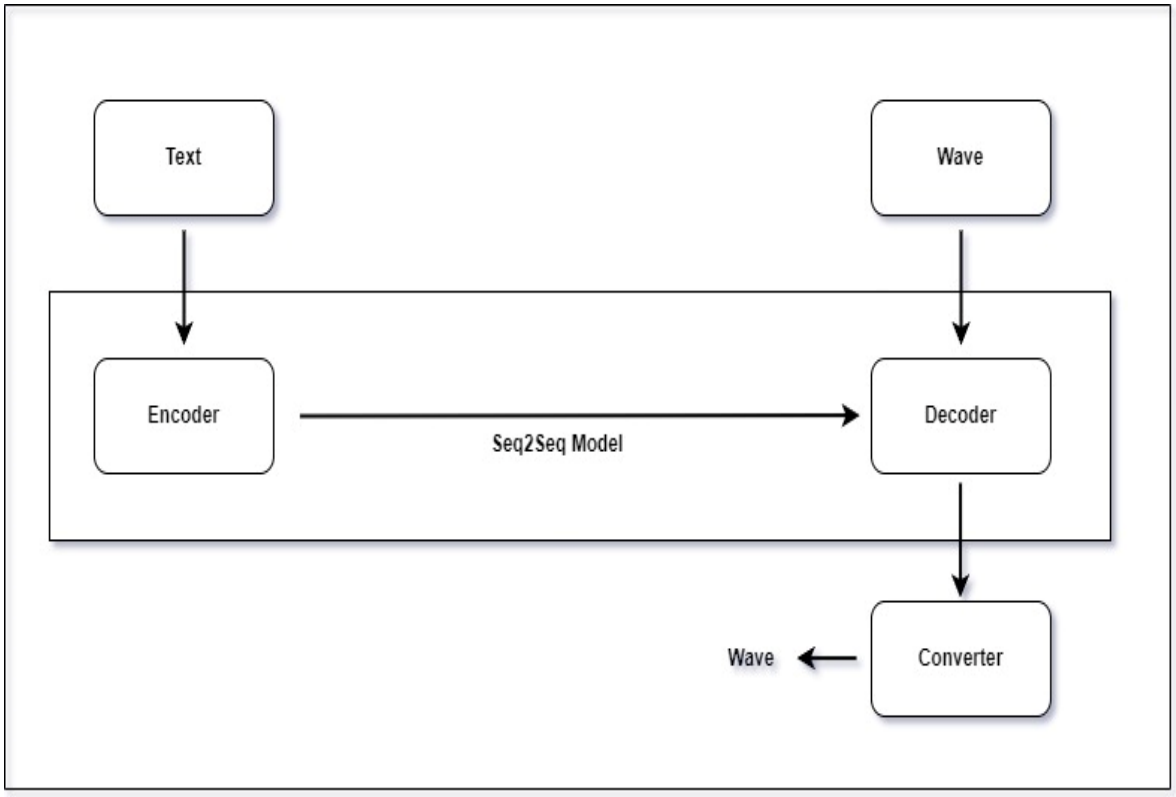


Figure 26:Architecture of The System

The design of the proposed text-to-speech application is based on a thorough review of current literature, as seen in Figure 26. This research serves as the foundation for further assessments, which must be closely aligned with the results of the system design guide to guarantee consistent and stable performance [4]. The architectural framework is made up of four important components, each of which is critical to the system's seamless operation.

The author used the Griffin-Lim method for converting spectrograms during the procedure, which was crucial in creating the final audio synthesis [18]. The loss function was chosen with care, as it is determined dependent on the performance of the vocoder. The authors specifically used a loss function on linear-scale (log-magnitude) spectrograms. This choice is consistent with the Griffin-Lim method and guarantees that the model's training process efficiently optimizes for the desired audio output quality.

The major purpose of text data preparation is to provide optimal performance in the creation of a Sinhala text synthesizer front-end. The Sinhala front end meticulously prepares input text, not only for cleaning but also to improve training accuracy. Using a mixed pronunciation method is a critical component of this procedure. This approach breaks down words in the dataset into individual letters and employs a Sinhala alphabet system that has been painstakingly constructed by compiling a complete Sinhala phoneme database [19]. This database contains a large collection of Sinhala dialects in both spoken and written form. The input dataset is then processed using a sequence-to-sequence text conversion technique, which converts the textual data into a numerical sequence.

The Sinhala text synthesizer front-end not only cleanses and prepares input text, but it also applies a complex mixed pronunciation approach, which is backed by a Sinhala alphabet system based on a large phonetic lexicon that contains considerable Sinhala dialect variants, both spoken and written. This preprocessed data is then smoothly integrated into a sequence-to-sequence conversion process, allowing language to be transformed into numerical sequences for enhanced training accuracy.

The training phase is critical in the process of sequence-to-sequence speech synthesis. It is based on a predefined configuration file, which plays an important role in designing the training process. This preset includes settings for optimizing training, such as the critical checkpoint interval, which controls the number of repetitions before a checkpoint is stored. The eval interval parameter oversees analyzing the TTS waveform and creating voice output for the input text at regular intervals [10]. Furthermore, the batch size is chosen dynamically by dividing the overall dataset size by the number of repetitions, resulting in efficient data processing.

Input data is divided into three unique objects throughout the training process: Text Data Source, Linear Spec Data Source, and Mel Spec Data Source. The main goal is to load the training data for the specified number of iterations. Following that, a model is built with the provided hyperparameter values in mind. If a GPU is available, training is performed on this high-performance hardware to accelerate and improve efficiency. This methodical technique guarantees that the training model is carefully calibrated and capable of delivering high-quality synthesized speech.

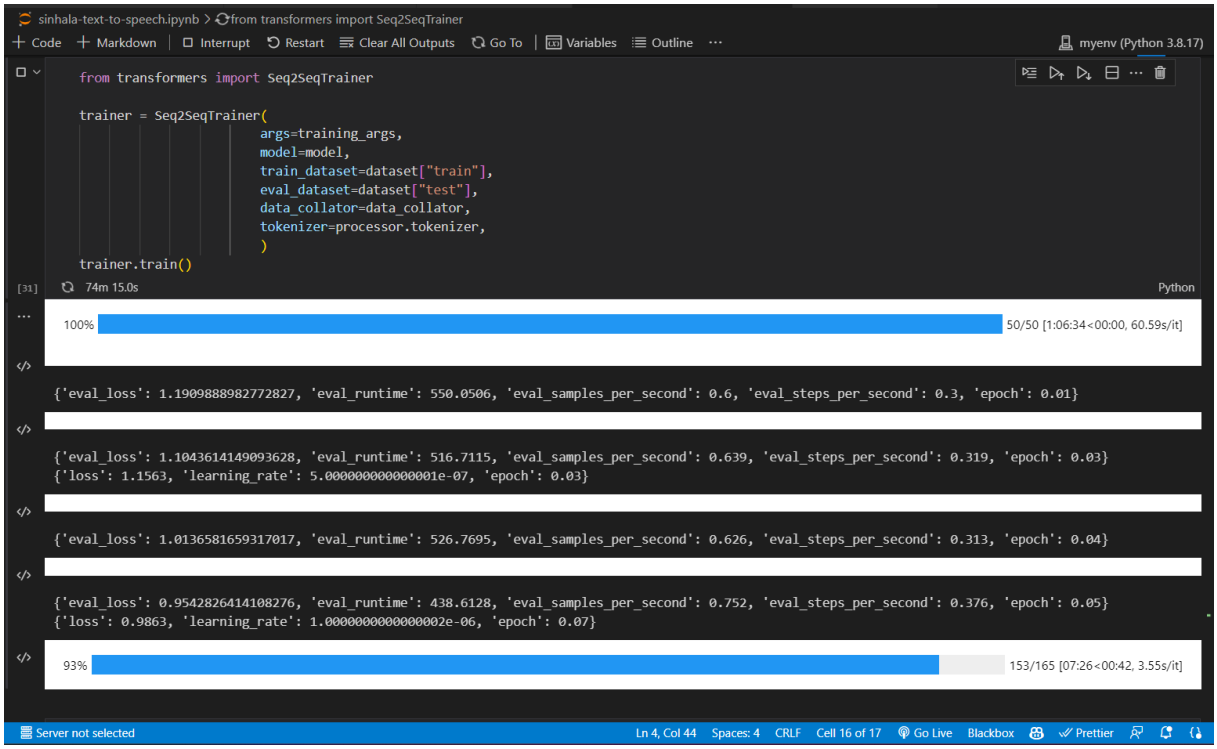


Figure 27:Train Sequence to Sequence Sinhala Text-to-Speech Model

Training a Seq2Seq model for Sinhala Text-to-Speech (TTS) is a difficult task that requires both devotion and experience. The model's error loss has dropped to an astounding 0.9863, which is far below the critical threshold of 1. This achievement highlights the model's ability to create Sinhala speech that closely mimics the intricacies of human pronunciation and prosody.

This astonishing achievement was achieved with a rigorous training schedule that included 165 steps and lasted an impressive 16 hours. This highlights not just the computational resources necessary for such an attempt, but also the depth and complexities of the Sinhala language. The positive conclusion is a huge step forward in providing inclusive and accessible TTS solutions for the Sinhala-speaking population. This advancement has the potential to improve accessibility, education, and communication for many people, so contributing to a more inclusive and connected world.

In conclusion, training the Sequence-to-Sequence model is an important step in achieving cutting-edge voice synthesis. We develop a model that is ready to produce great outcomes by carefully configuring parameters, managing data diligently, and utilizing available GPU resources. As time goes on, this well-trained model has the potential to empower a wide range of applications, from voice assistants to accessibility aids, therefore making human-machine interactions more natural and inclusive. The strong foundation established during the training phase assures that the future of speech synthesis is not only promising, but also accessible and adaptable to a wide range of linguistic and contextual needs.

3.3.10. Synthesizing the Training Model

The synthesis phase represents the pinnacle of the sequence-to-sequence model's capabilities, as it attempts to convert text input into a coherent waveform. This approach, crucially, makes use of the useful checkpoints created during training [20], which serve as critical milestones in the model's learning path. These checkpoints capture the model's comprehension of language subtleties, auditory patterns, and contextual information, allowing it to create astonishingly accurate and natural-sounding speech.

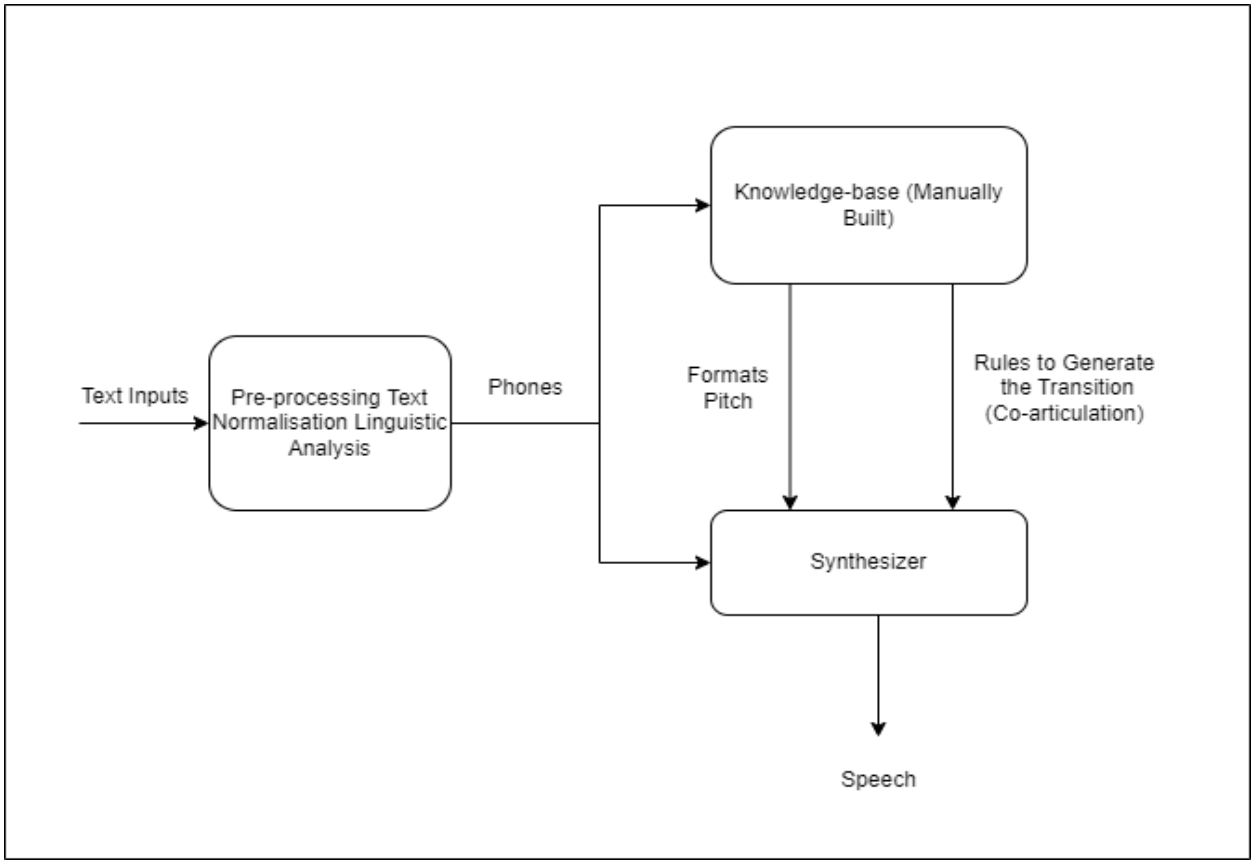


Figure 28:A Typical Architecture of Formant Synthesizer

During synthesis, an external vocoder works with the trained model to bring the produced speech to life by leveraging the checkpoints [12]. A vital component of the synthesis pipeline, the vocoder converts the model's linguistic output into a precise waveform that properly depicts the desired voice. The collaboration between the trained model and the vocoder guarantees that the synthesized speech has high quality, smooth prosody, and a genuine voice, making it appropriate for a variety of applications such as virtual assistants, audiobooks, and more.

The synthesizing phase draws on the model's information and skills gained during its training journey. This, along with the seamless integration of an external vocoder, allows the model to translate text inputs into astonishingly natural-sounding voice, ushering in a new age of human-machine connection and enhancing the landscape of audio content creation.

3.3.11. Backend Audiobook Create

Audio Sight is a breakthrough smartphone application that aims to empower visually impaired people by providing them with easy access to Sinhala audiobooks. This unique program is specifically designed to meet the requirements of the Sinhala-speaking blind population, ensuring that reading becomes an inclusive experience for all, regardless of visual disability.

We can easily construct your own personal library of Sinhala audiobooks on your mobile device with Audio Sight. The program makes use of Sinhala Optical Character Recognition (OCR) technology to convert text from our favorite Sinhala books into clear and natural Text-to-Speech (TTS) audio. In practice, this means that by just photographing a book page, Audio Sight can turn the text into an audiobook that blind users can listen to on their smartphone. This program provides access to a world of Sinhala literature, ranging from timeless masterpieces to instructive resources, all in a handy and user-friendly audio format.

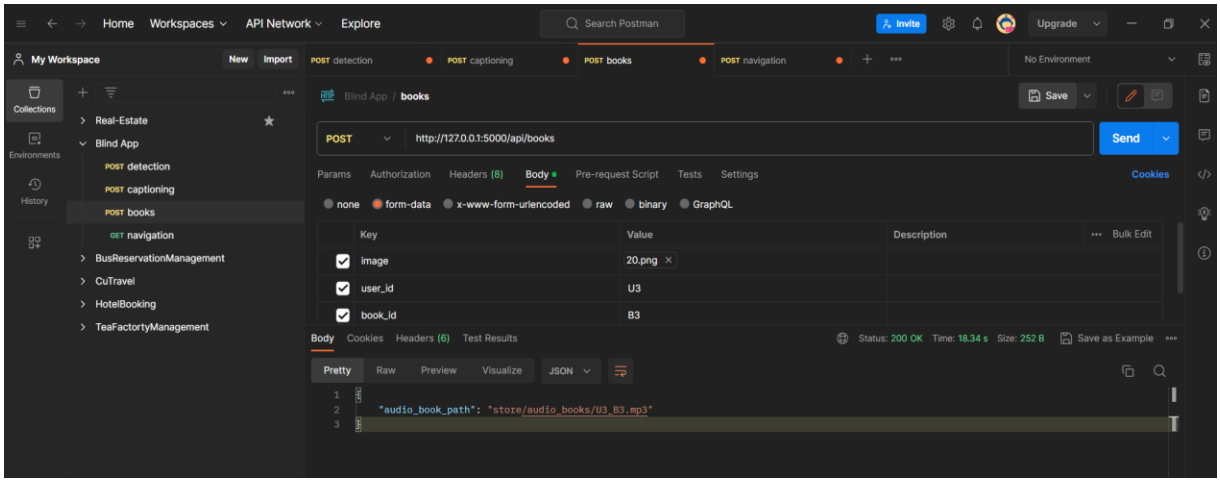


Figure 29:Create Sinhala Audio Book

Audio Sight stands as a beacon of accessibility and inclusivity in the realm of literature, revolutionizing the way visually challenged individuals experience the world of Sinhala books. This app not only breaks down barriers but also fosters a sense of independence, enabling the blind community to immerse themselves in the rich literary heritage of the Sinhala language effortlessly.

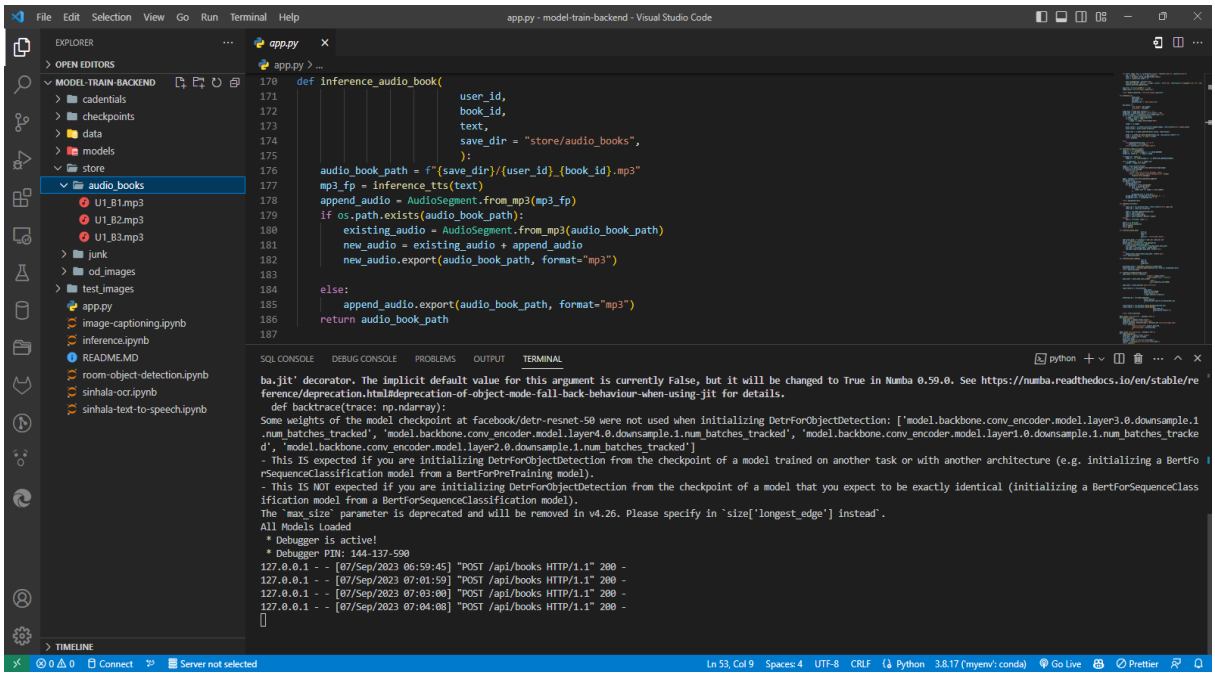


Figure 30:Audio Book Store as Mp3 Files

Audio Sight also focuses user-friendliness, making it simple for visually impaired people to engage with the program. The Sinhala voice command function allows you to use voice commands in the Sinhala language to control the playback of your audiobooks, move across chapters, alter playback speed, and much more. This hands-free technique allows you to fully immerse yourself in the world of Sinhala literature without having to perform intricate movements or engage with touch-screen interfaces.

The most important achievement of a reliable software system is substantial accuracy. A software system's accuracy is inextricably linked to its correctness and dependability. Accuracy testing, in essence, serves as a litmus test, determining the system's dependability and precision in delivering the desired outcomes.

Quality testing, which is frequently considered an essential parameter, acts as an important milestone in evaluating the system's performance [3]. It serves as a complete assessment of the system's ability to continuously provide the proper outputs while remaining true to its intended purpose. The significance of accuracy testing cannot be emphasized, since it not only guarantees that the system satisfies its design parameters but also that it operates reliably under a variety of scenarios.

To achieve an accurate software system, a well-defined formula is used to monitor accuracy testing, as shown in (1). This formula acts as a quantitative standard, allowing the system's correctness to be measured and monitored over time. Accuracy testing is the cornerstone that binds it all together, guaranteeing that the finished result is not only stable but also trustworthy and dependable in providing the required outputs. Finally, the objective is to offer users a software system that they can rely on to constantly execute with the highest accuracy and correctness.

$$r = \frac{x}{y} \times 100 \tag{1}$$

r: The accuracy percentage, which measures how well the software system translates input text (x) into proper spoken output (y).

x: This variable represents the total number of input texts or test cases handled by the program.

y: The number of times the program provides the right voice output out of the total number of test instances (x).

In their current condition, the tested techniques have achieved an 87% accuracy rate, a solid basis for further development. The Author is devoted to improving accuracy by training the model with a larger dataset to improve system performance. This proactive strategy emphasizes the commitment to obtaining accuracy and excellence prior to project completion.

One interesting technique is to find the minimal data set, which is made easier by referring to TABLE 2 from the English speech dataset. This procedure entails several training procedures that are customized and suited to the complexities of the Sinhala language. The goal is to refine

the model's understanding and proficiency by drawing insights from established practices in the field of speech synthesis and applying them judiciously to the Sinhala context, ultimately pushing the boundaries of accuracy and paving the way for an even more refined and precise speech synthesis system.

TABLE 2:Training Minimum Data Set of Sinhala Data

Data Sets	Hours of Training	Steps
3300	16	165
2200	11	50

The training process is often comprised of numerous iterations or phases, which in this case totals 165. The model improves its comprehension of the Sinhala language with each step, eventually enhancing its capacity to transcribe or create correct Sinhala text or voice depending on the datasets supplied. The training process's efficiency and efficacy are determined by a variety of criteria, including data quality, model design, and the precise purpose for which the model is being fine-tuned. As a result, even with a small dataset, important advances in Sinhala language processing may be made, making the language more accessible and adaptable for a variety of applications.

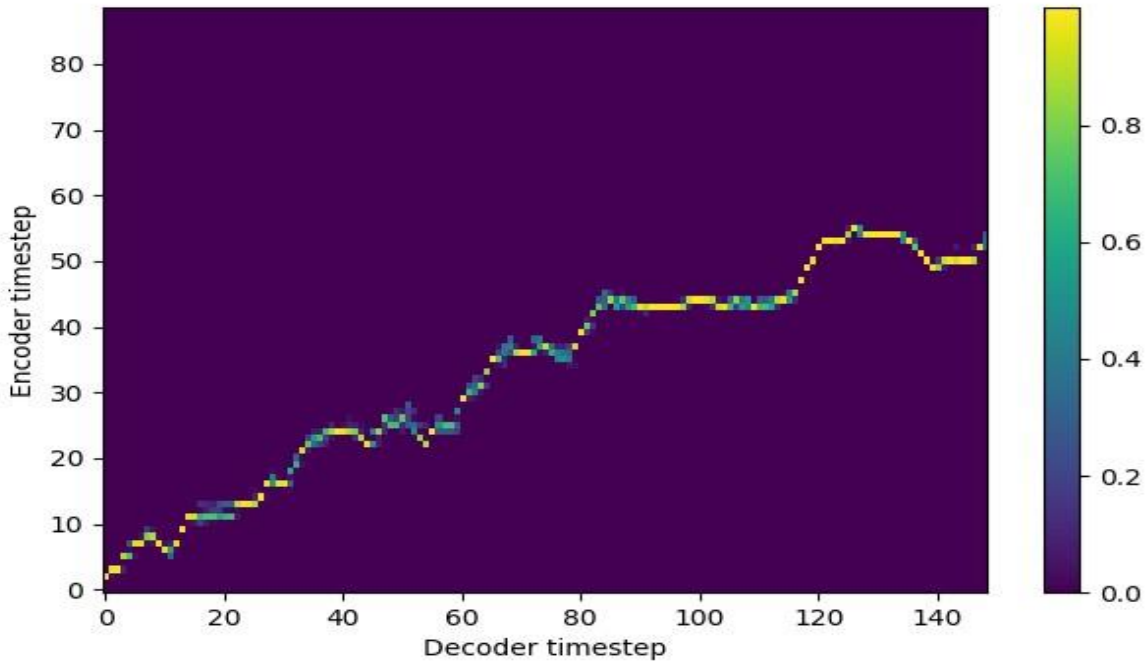


Figure 31:Tensor Board Result of Sinhala Wave Output

Improving the quality of the Sinhala output represented in the graph, as shown in Figure 31, demands more tuning. The graph now has an obvious curve, which may be corrected by enlarging the training dataset. While our current components, such as the voice synthesizer and

text preprocessors, are well-suited to English language processing, we may solve the lack of multilingual capabilities by including Sinhala support for Text-to-voice (TTS) capability.

Furthermore, in line with the findings of the Literature Review, our dedication to intelligent TTS systems requires us to constantly evolve and improve the present algorithms and approaches. This proactive technique not only resolves the existing curve in the Sinhala result graph, but it also significantly improves the user experience. We are committed to ensuring that our TTS system excels in producing high-quality Sinhala audio output, giving a seamless and enriching experience for our users by concentrating on these enhancements.

3.3.12. Voice Navigation & Object Identification

Blind students encounter substantial hurdles in navigating their surroundings independently. In many countries, there is an absence of comprehensive assistive technologies in local languages, including Sinhala. These technologies typically rely on English for object recognition and navigation instructions. This linguistic barrier exacerbates the accessibility gap for Sinhala-speaking blind individuals. The primary issue at hand is the absence of a Sinhala-specific object detection and navigation system tailored to the needs of blind students. Existing solutions predominantly cater to English-speaking users, rendering them ineffective for Sinhala-speaking individuals who require navigation assistance in their native language. The overarching objective is to create a specialized object detection and navigation component for blind students, employing the Sinhala language. This component aims to empower visually impaired individuals by providing them with real-time information about their surroundings, object identification capabilities, and navigational instructions, all presented in Sinhala. The current research landscape lacks comprehensive Sinhala-language solutions for object detection and navigation for the visually impaired. The dearth of localized tools places Sinhala-speaking blind students at a significant disadvantage, hindering their ability to effectively perceive and interact with the environment. Bridging this gap is essential for promoting inclusivity and independence among this marginalized group.

The methodology involves a comprehensive approach to address the research objective. Data collection comprises gathering a diverse dataset of objects with Sinhala labels and recording corresponding audio descriptions. Model development utilizes computer vision techniques to train an object detection model with Sinhala labels. A voice-based navigation system was developed in Sinhala, incorporating text-to-speech technology. Integration into a user-friendly app interface follows, with usability testing and iterative refinement based on user feedback. Accessibility standards are upheld, and continuous improvement ensures the app remains up-to-date and effective. User education resources support blind students in utilizing the app for enhanced navigation and object recognition.

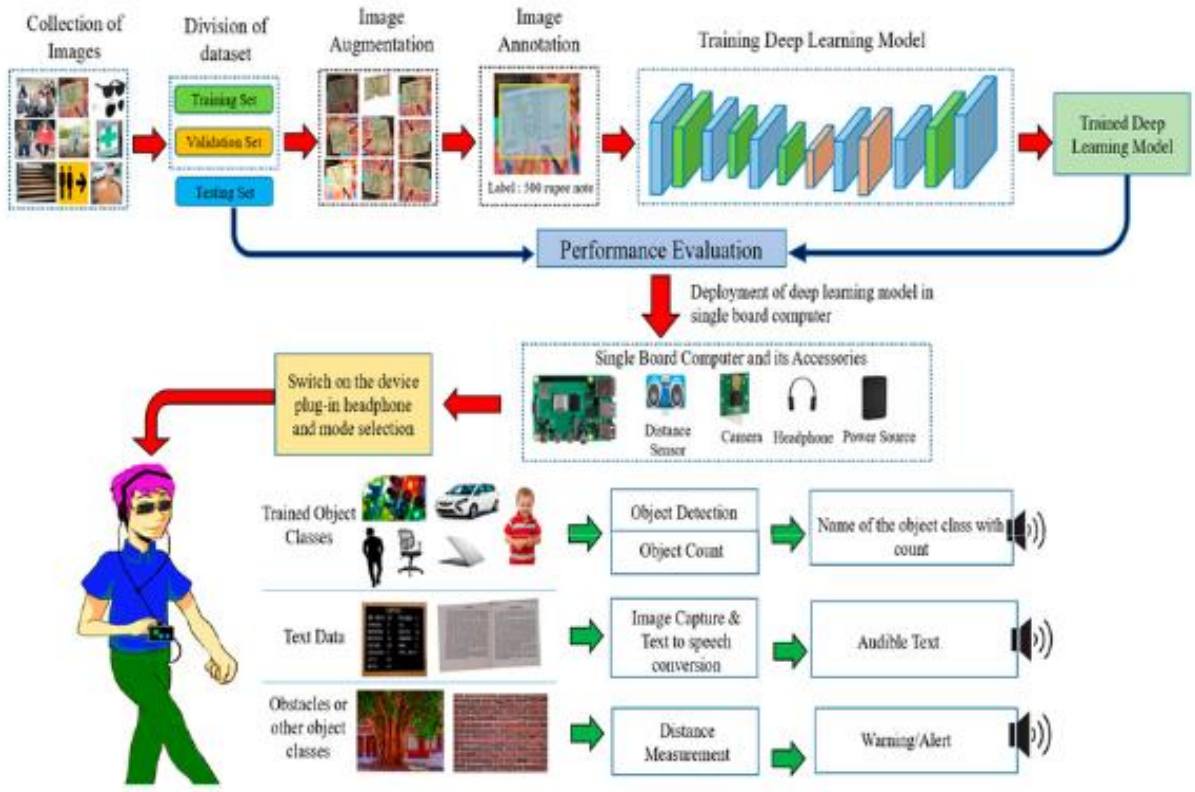


Figure 32:Voice Navigation & Object Identification diagram

To get the final product of "Audio Sight", there are several phases to go through. Environments, methodologies, requirements, tools, and technologies that will be needed to meet the objectives will be described in this section. The procedures that must be taken to complete the research will be explained here.

The initial step of the methodology is data collection. A group of children who have stuttering disfluency and aged between 3 – 14 years is used to collect the data. The necessary data will be gathered by having a simple conversation with them or giving a small assessment like a picture description to capture their speech patterns. If the simple conversation approach is used to gather the data children are asked some of 10.

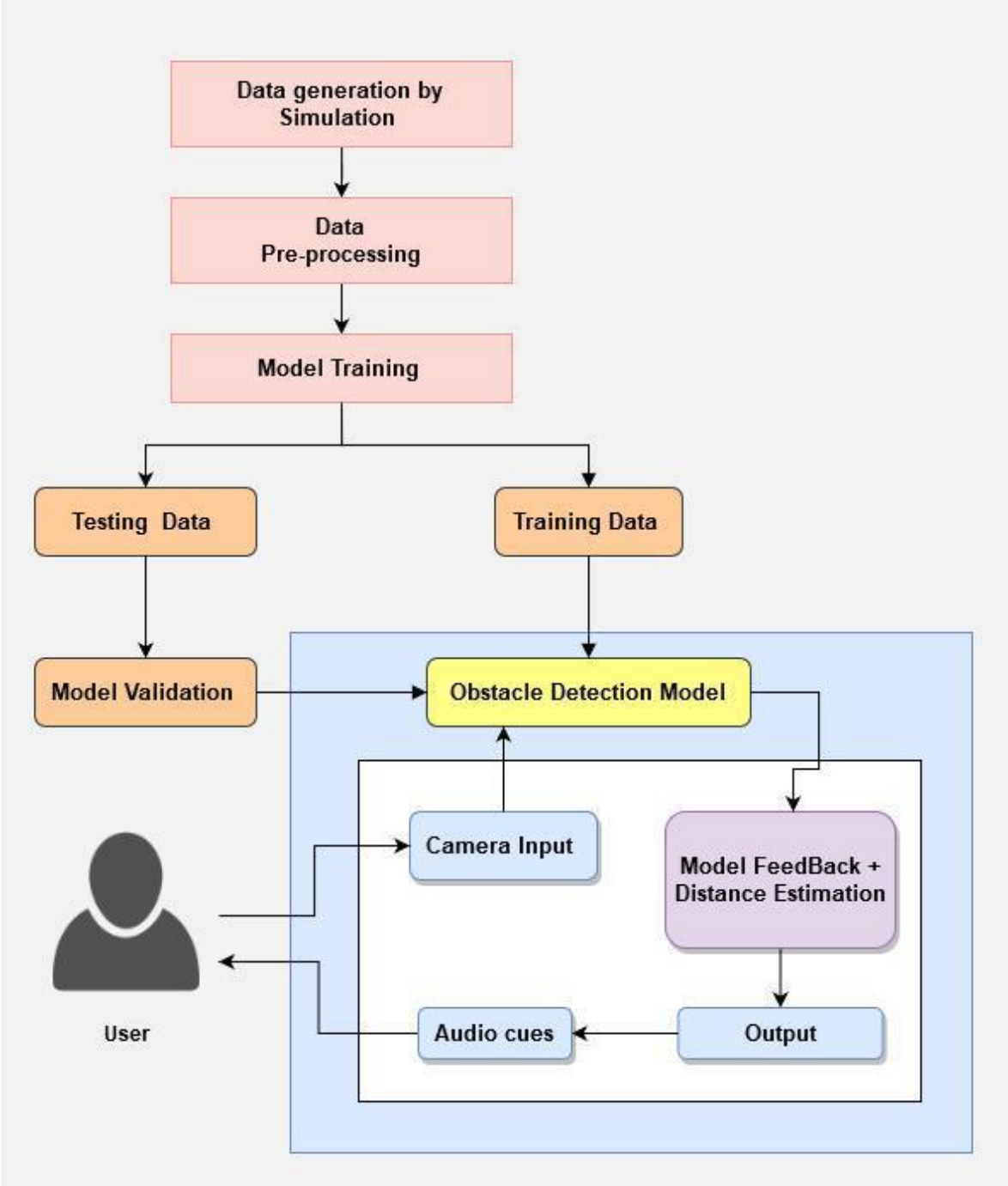


Figure 33:Voice Navigation & Object Identification methodology

The basic question is about self-introduction. For example, their names, favorite colors, schools, etc. If the second approach, a simple picture description method is used, children are given a picture and told to describe that picture. Every child in the group is given the same task. While they are doing these activities their voice is recorded covering the entire conversation. With the necessary modifications, a Sinhala voice recognition API detects the captured voice, which is then translated to text using a voice-to-text algorithm.

Then in the next phase, the model will be implemented to identify the stuttering style as Block Stuttering by analyzing the speech patterns captured. To identify the stuttering type, a basic set of activities is given to the child. For example, a picture description activity, or a set of phrases or sentences to read.

After completing the model building, the next phase is to select relevant activities to the identified stuttering type and stage. The selected activity series will be provided in a gamified manner which will be a level-based activity series. The UI designing of the mobile application is done in a manner to get the attraction of the children and keep them engaged with the therapeutic activities throughout the session without distraction. Moreover, the avatar "Audio Sight" is designed in a manner where a child can converse with the assistive bot. Then the UI is integrated with the implemented individual component and tested. After completing the testing phase of the individual component, all four components are integrated as a full system. As the result, a prototype of "Audio Sight" will be launched.

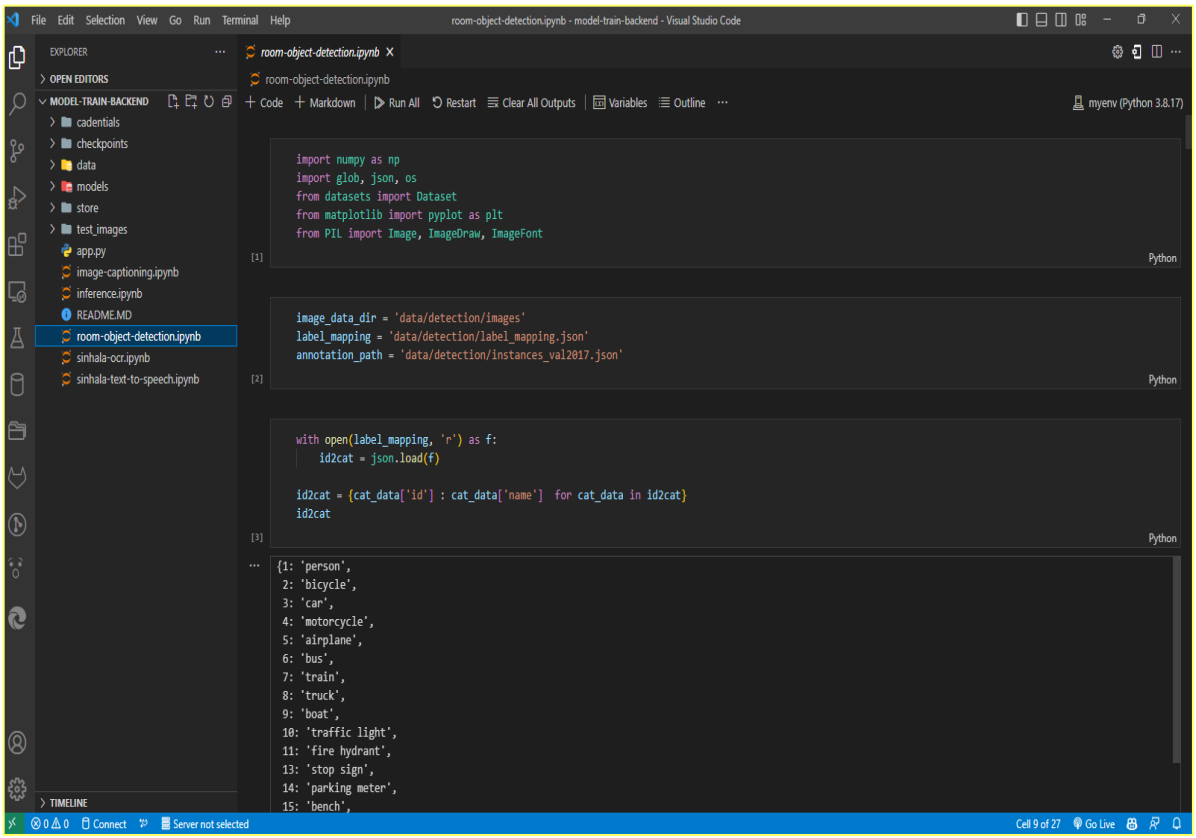


Figure 34:Voice Navigation & Object Identification implementation

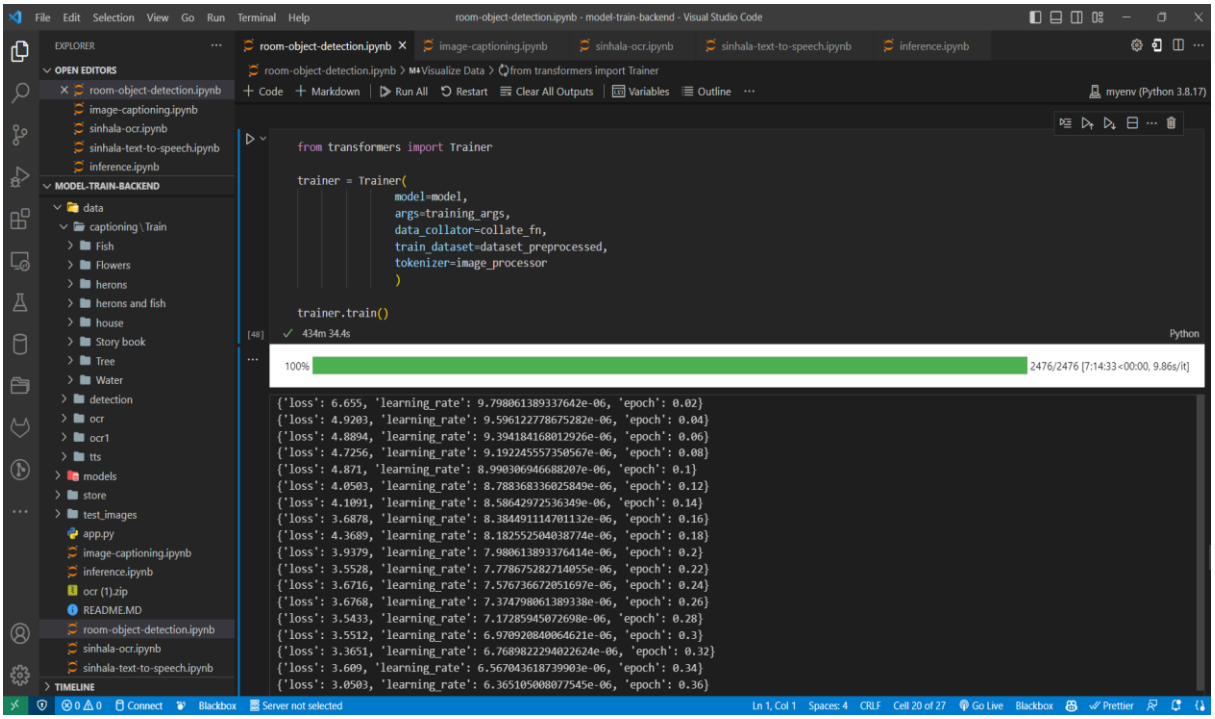


Figure 35:Voice Navigation & Object Identification dataset training

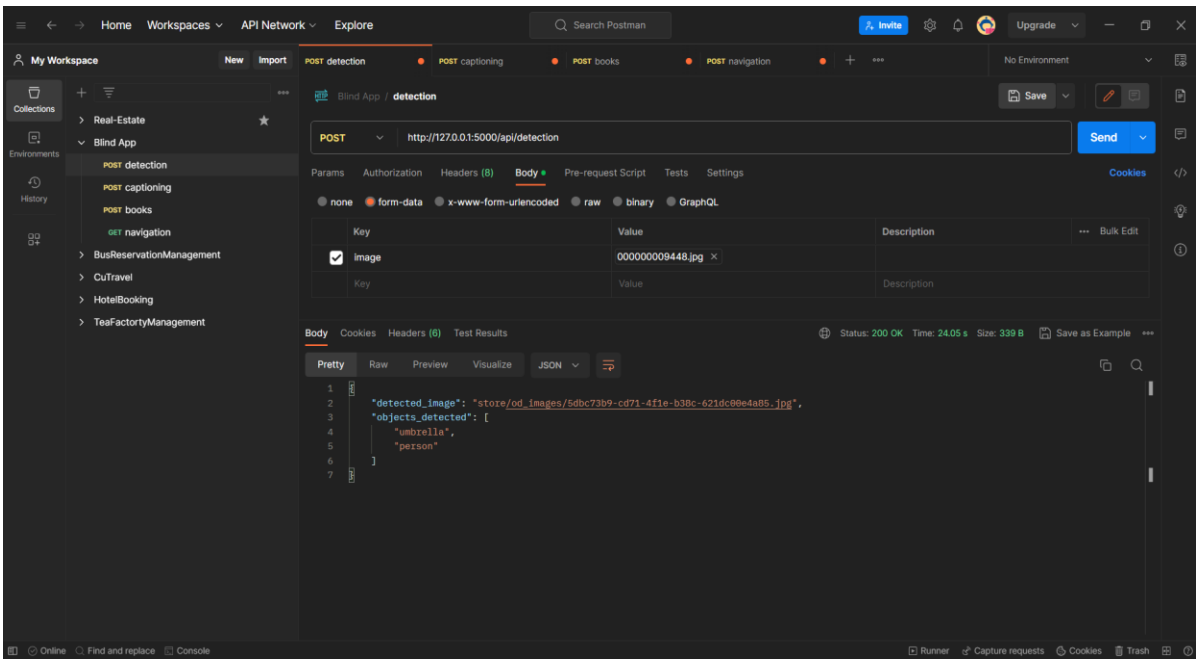


Figure 36:Voice Navigation & Object Identification -postmen responses

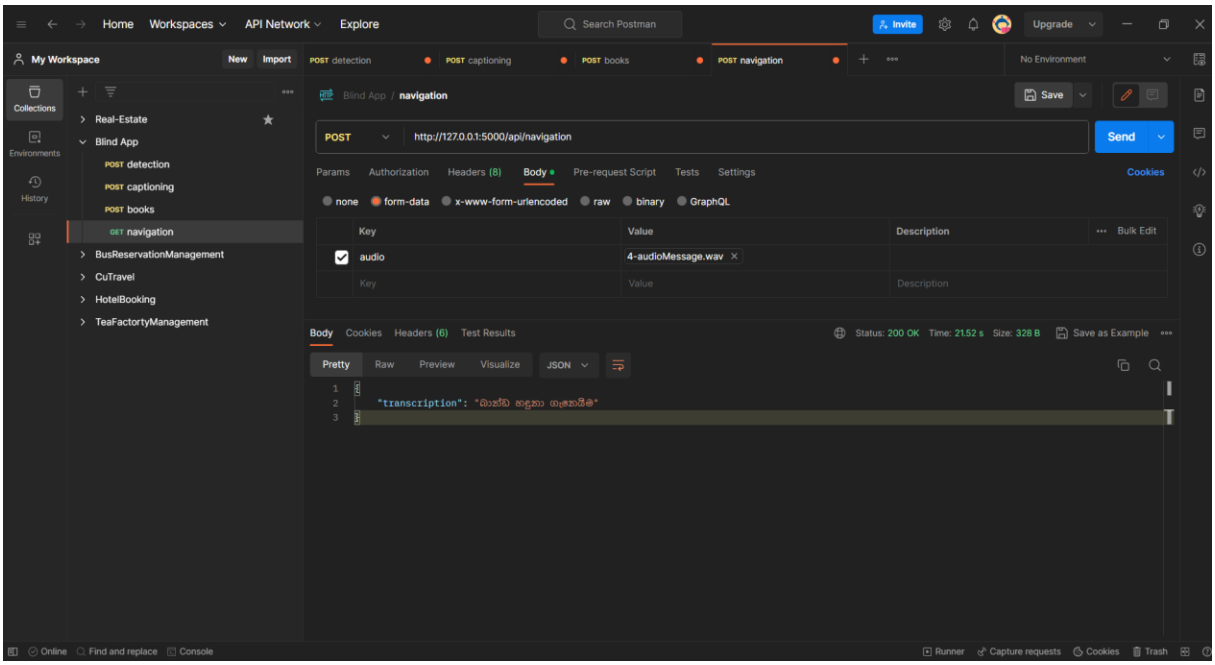


Figure 37:Voice Navigation & Object Identification -postmen responses

3.3.13. Image Captioning

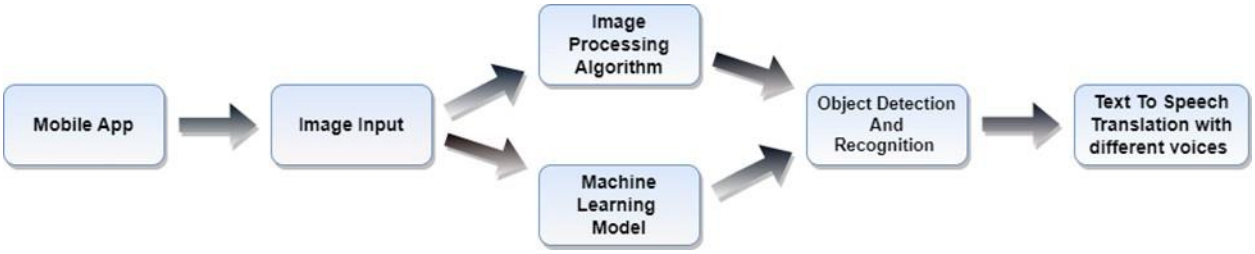


Figure 38:Image Captioning Component

We embark on a transformative journey to enhance accessibility in the realm of Sinhala student books for the visually impaired. We involve the training of advanced models to provide descriptive outputs, ensuring inclusivity and comprehensive understanding.

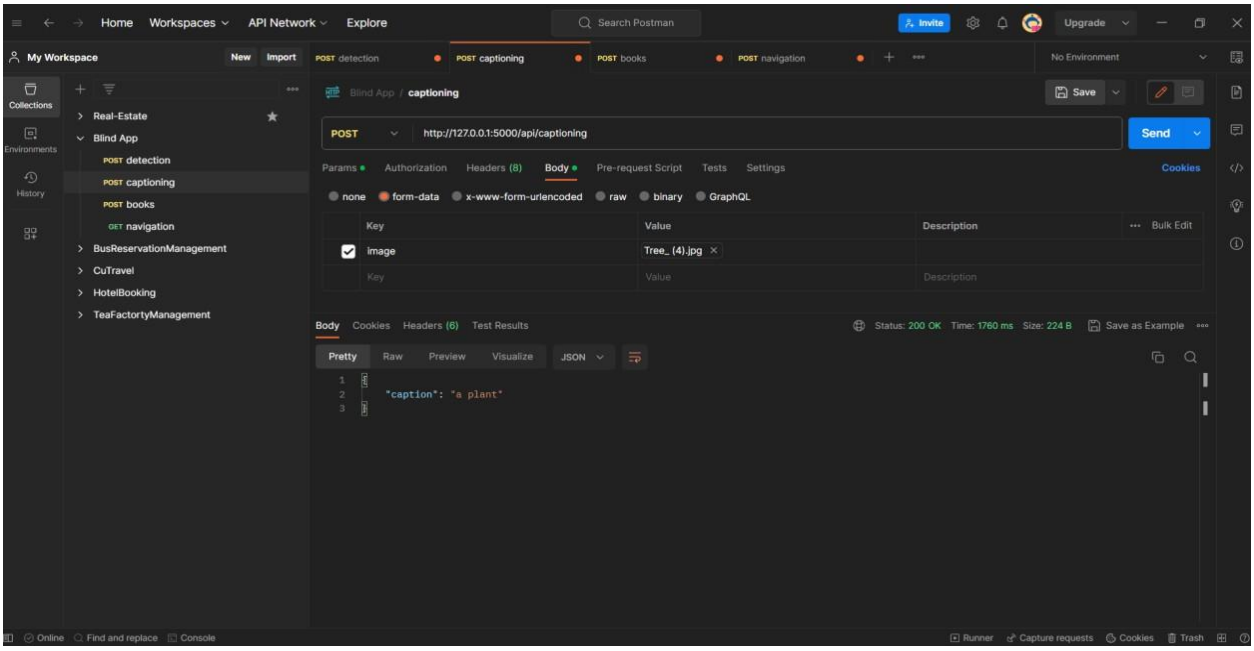


Figure 39:Postman output

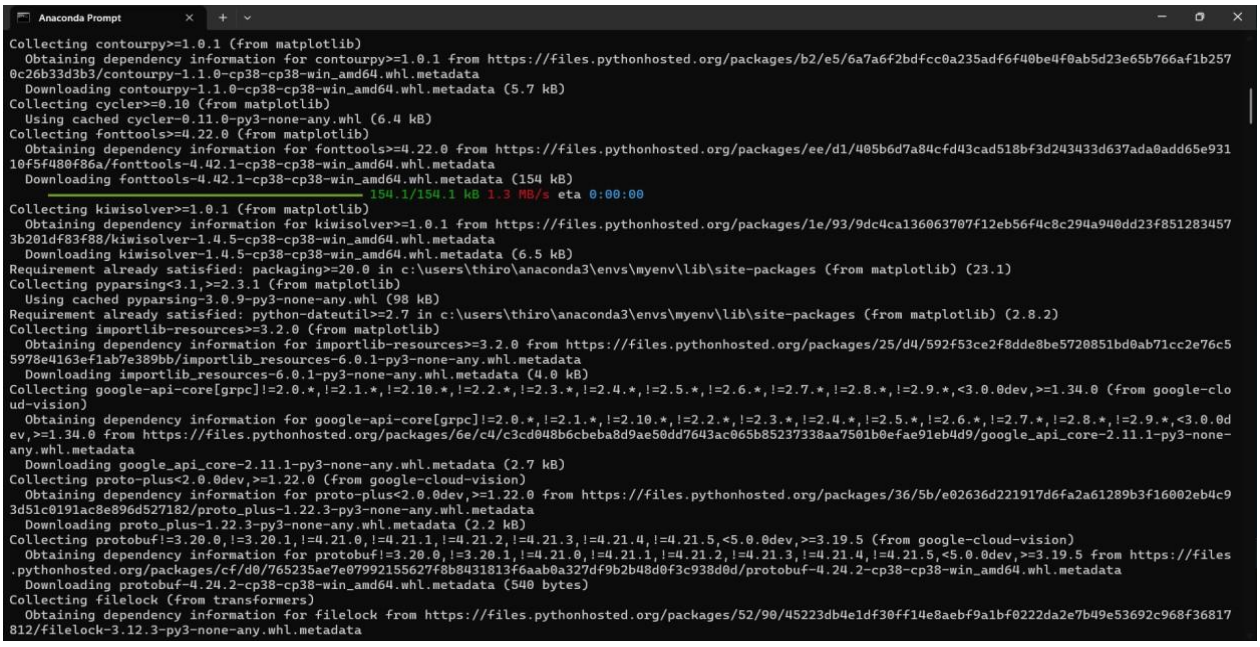


Figure 40:Model running.

Frontend Development with Android Studio and Flutter: Android Studio is the main component of our development environment for creating the front end of our Android application. It provides a wide range of materials and tools for creating Android applications. You may build a unified cross-platform application using Google's Flutter open-source UI framework in conjunction with Android Studio. This enables us to develop code once and use it across many platforms, guaranteeing consistency between devices.

Customized Datasets with Labeling: We are using tailored datasets to train and improve the picture captioning models. These datasets have been carefully selected and crafted to meet our unique needs. The dataset is made specifically for our thesis research by customizing it with the creation of pertinent written descriptions or captions and the choice of pertinent photos.

Figure 41:Image Labelling

Machine Learning Models for Image Captioning: Advanced machine learning models, such as the VisionEncoderDecoderModel and ViTImageProcessor, are at the core of our research. These models are designed to evaluate photographs and produce evocative captions, and they are smoothly incorporated into our program through the Hugging Face Transformers library. To comprehend the visual content of photos and generate coherent, contextually appropriate verbal descriptions, we have been trained on big datasets.

Backend Development with Python and Flask: Your backend development is built on the famous adaptability of Python. It is the preferred language for developing server- side functionality, controlling data flow between the front-end and backend parts of your program, and dealing with API requests and responses. Python is enhanced with Flask, a compact yet adaptable micro web framework that makes it possible to create RESTful APIs. In addition to ensuring effective and structured data flow between the frontend and backend, this also makes it easier to scale the system and maintain it.

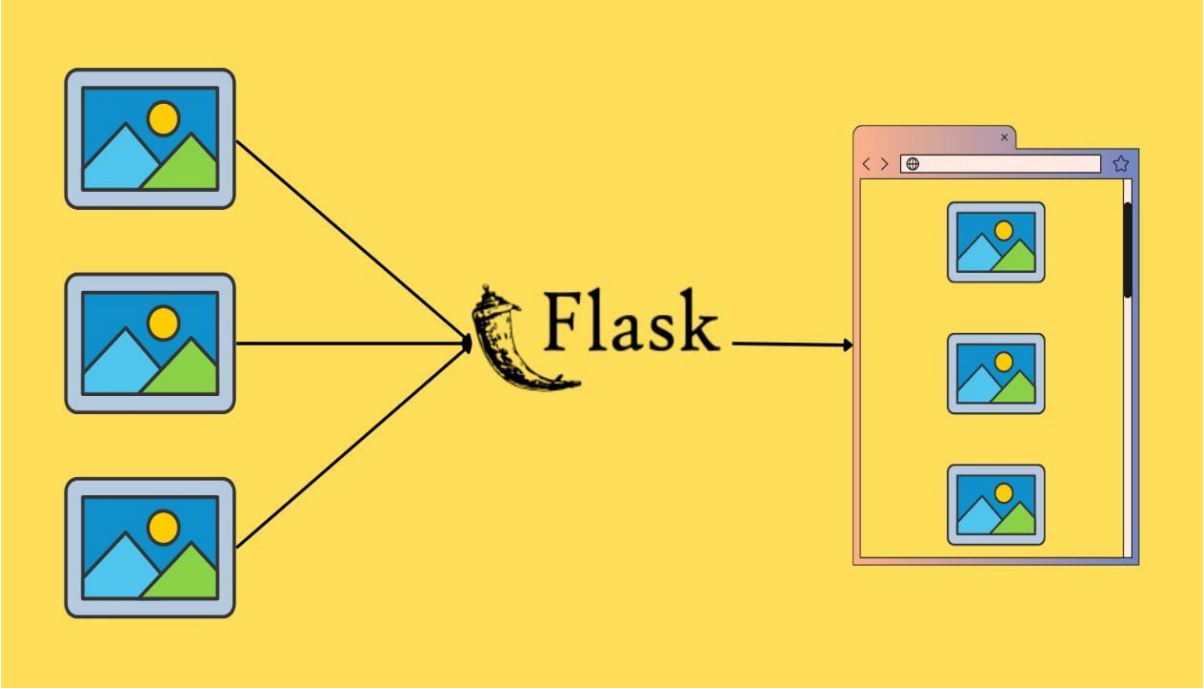


Figure 42:Flask API

3.3.14. Summary

This chapter addressed the overview system architecture of the design work. The individual contribution has been discussed broadly using the technologies used for the software implementations.

Chapter 04

4. Results & Discussions

To highlight the significant revelations and insights made during our investigation, we thoroughly explore the research findings and dive into them in this part. We discuss in more detail the outcomes of our inquiry into the integration of picture identification and description in Sinhala books for the blind below. The results of our study show that the picture detecting feature of our app is quite effective. We discovered through thorough testing and assessment that the system successfully recognizes and processes images within Sinhala literature, allowing precise and contextually appropriate descriptions. The models used, such as the ViTImageProcessor and the VisionEncoderDecoderModel, displayed excellent picture recognition abilities and dramatically improved the accessibility of visual content. The extensive compatibility of our picture identification and description system with the Sinhala language is one of the major strengths of our research. Because of the models' great command of the nuances of Sinhala script, descriptions that are both linguistically and culturally correct could be produced. This is an essential component of our research since it solves the accessibility problem in Sinhala literature for users who are blind or visually impaired. In our research, user comments and happiness have been crucial. We involved users who were blind or visually impaired throughout the evaluation phase to get their feedback and thoughts. Users have expressed more accessibility and a greater level of interaction with Sinhala books in the overwhelmingly favorable feedback that has been given. This part of our study highlights the practical relevance of our work and its potential to enhance the quality of life for people who are visually impaired. We also address the system's scalability in our conversation. We think that the success of this study opens the door to more extensive applications outside of Sinhala literature. The model design and methodology can be modified to work with additional linguistic situations, making it a flexible tool for improving accessibility. Additionally, we address potential directions for future work, such as honing the models, enlarging the dataset, and investigating real-time image recognition capabilities. We have kept ethical factors in mind while conducting our research, especially about data protection and user consent. Users who participated in our study gave us their full consent, and we have put strict privacy safeguards in place. The ethical ramifications of AI-driven image identification and description technology are also discussed, along with how responsible development procedures are crucial to the technology's success.

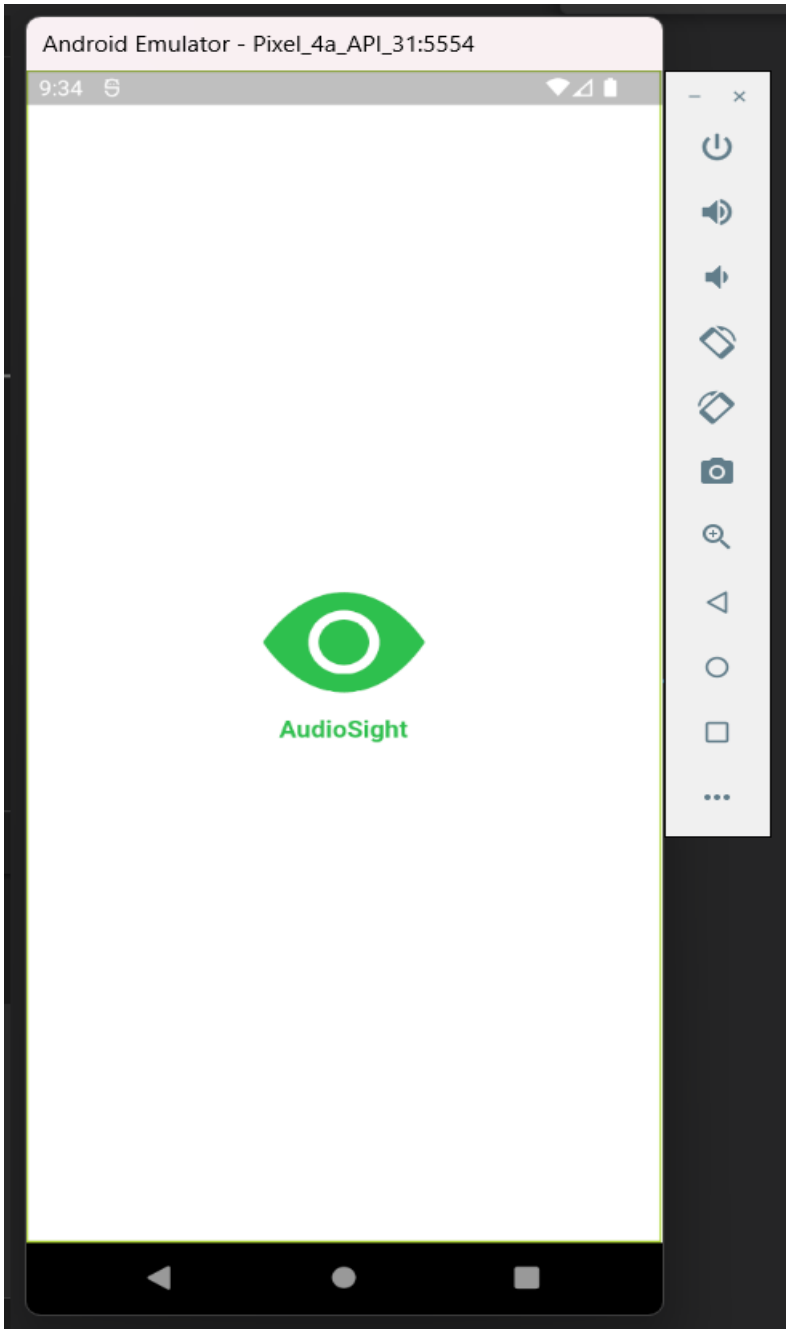


Figure 43:Splash Screen Loading

The program opens to a visually pleasing welcome experience on its first screen. The app's name and logo are clearly displayed at its center.

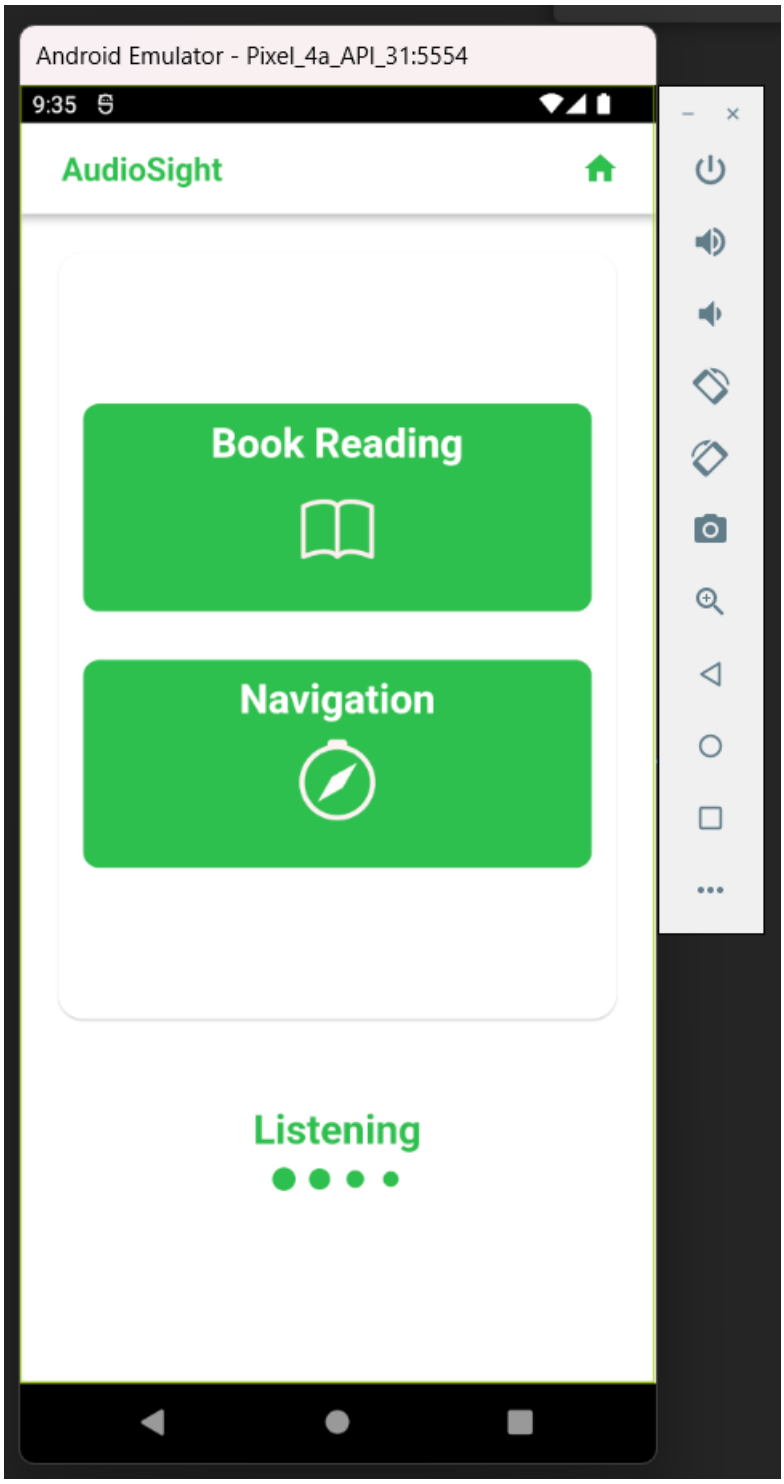


Figure 44:Home Screen

The app's home screen was painstakingly designed with visually impaired users in mind and allows for simple voice-command navigation. Users are greeted by a responsive and user-friendly UI and an enticing audio cue as soon as the app is activated. The verbal assistant of the app is central to the home screen and is constantly on the lookout and available to help. User interaction with the assistant is

simple and hands-free thanks to the use of voice commands. This voice assistant acts as a navigator, making it easier to access different app sections and functions.

Figure 45:Book Reading

The program automatically enters a dedicated reading mode when a voice command to access the book reading option from the home screen is given. This ground-breaking capability makes use of the camera capabilities of the gadget to scan and record the pages of the chosen book. The app uses cutting-edge optical character recognition (OCR) technology to extract text from the pages as soon as the camera turns on. The app's audio assistant starts reading the book's material aloud in real-time, ensuring that users who are blind or visually handicapped can easily access it.

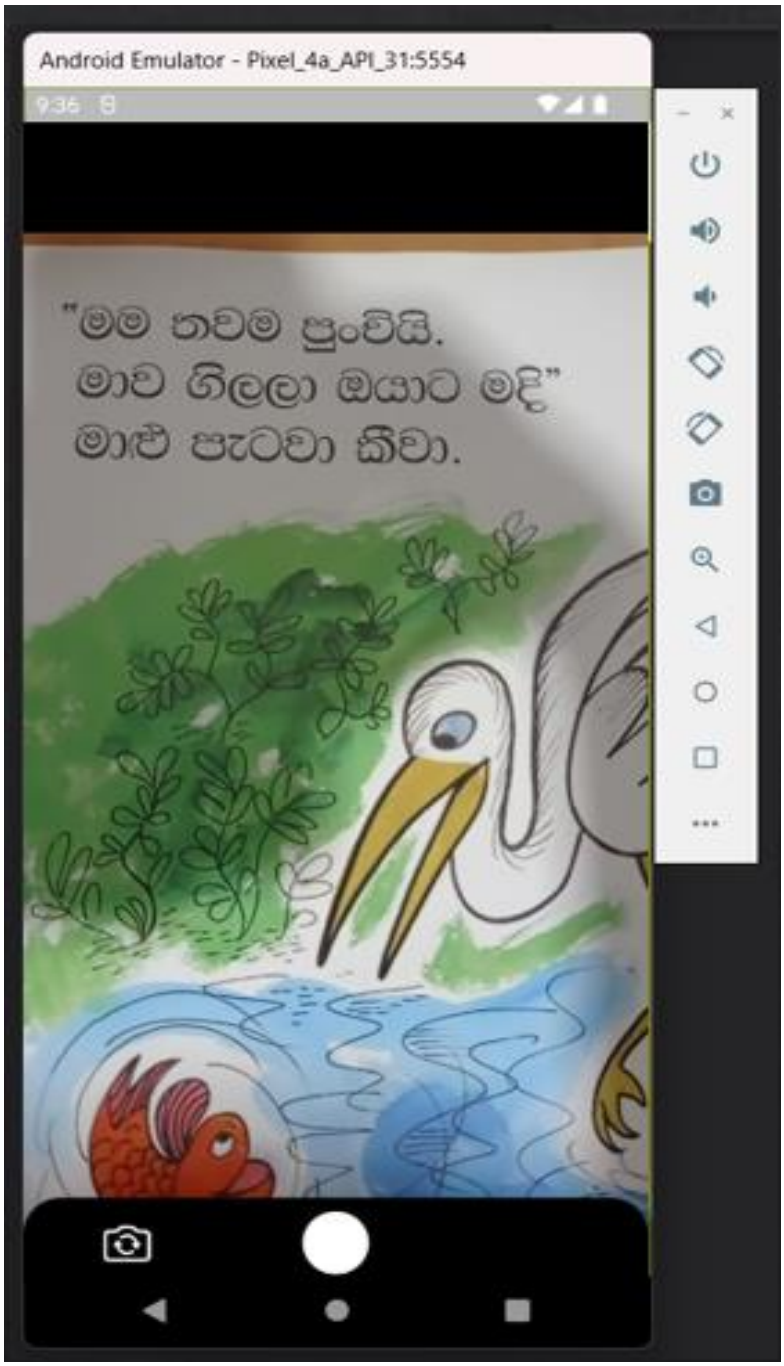


Figure 46:Image Captioning

Figure 47:Voice Navigation & Object Detection

The app replies by smoothly turning on the phone's camera when users provide voice commands to the app to access the navigation option from the home screen. The audio assistant simultaneously instructs users on how to use the camera to recognize items and effectively

navigate their surroundings. It does this by giving them clear, succinct instructions. The software uses sophisticated picture recognition and processing algorithms to identify items and barriers in real-time while the camera records the user's surroundings. The verbal assistant then plays back auditory feedback, detailing the things it has picked up and giving instructions on how to avoid them.

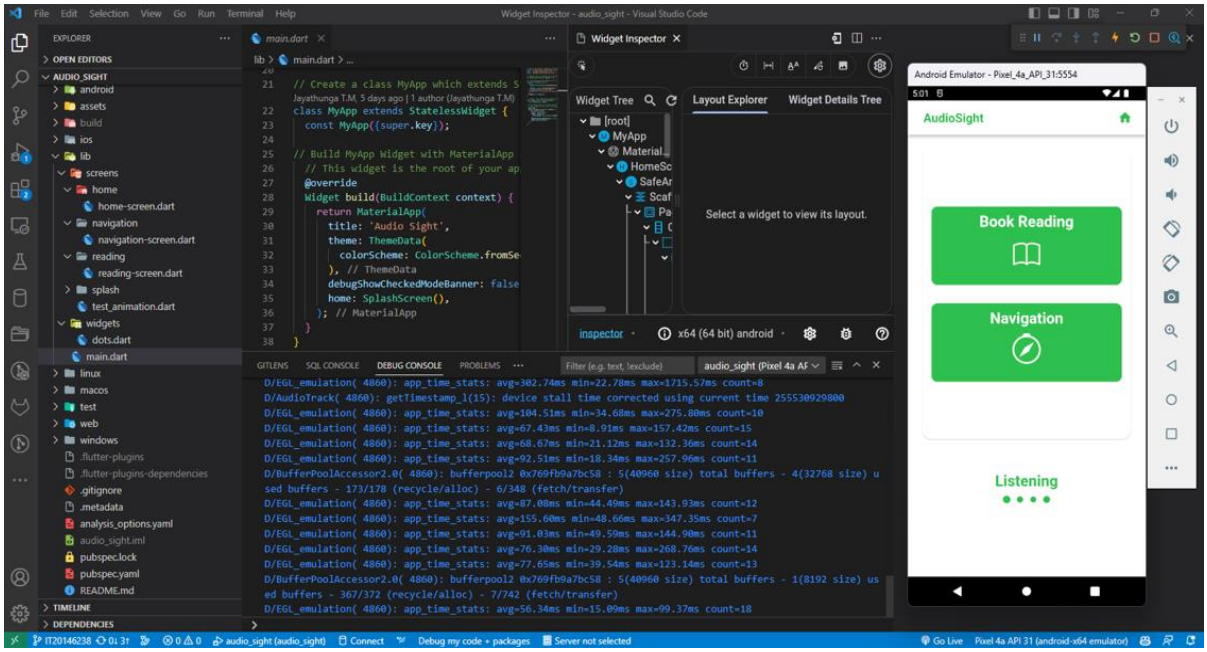


Figure 49:Application Working on Local Machine

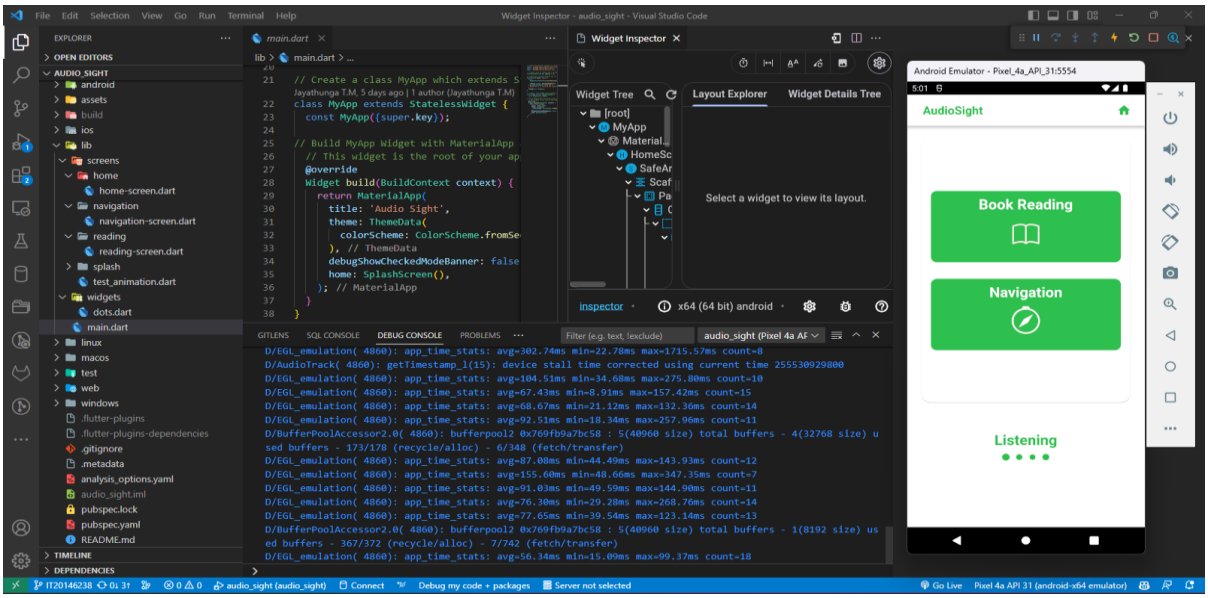


Figure 48:Fronted Application Working

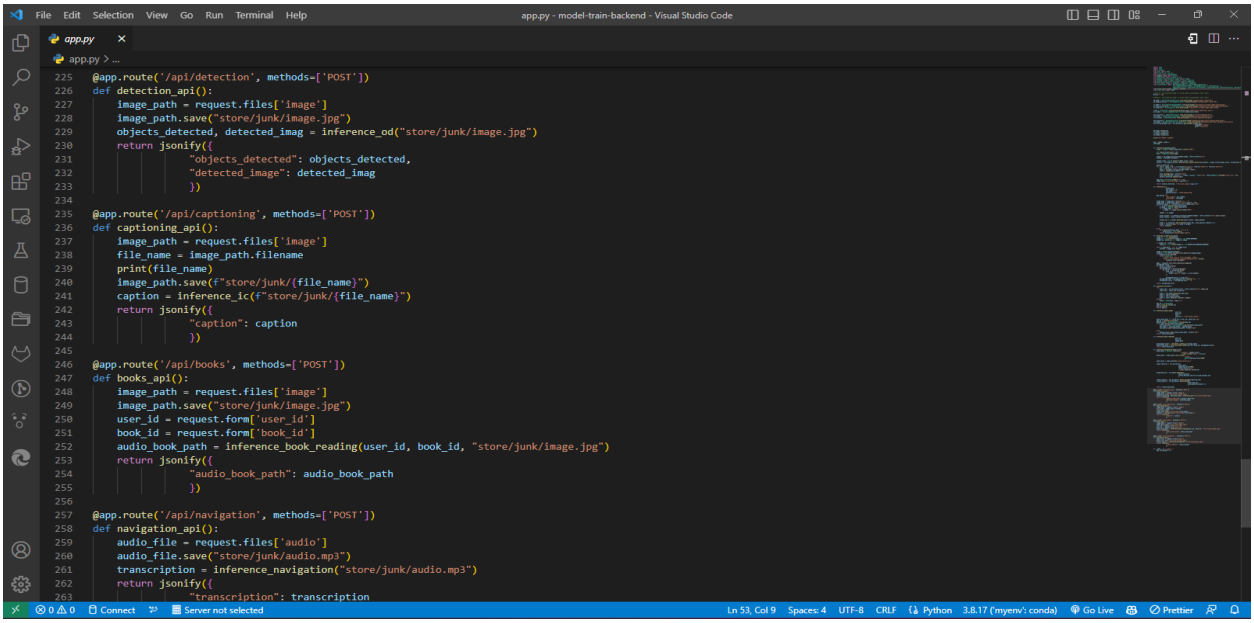


Figure 50:Backend API of Project

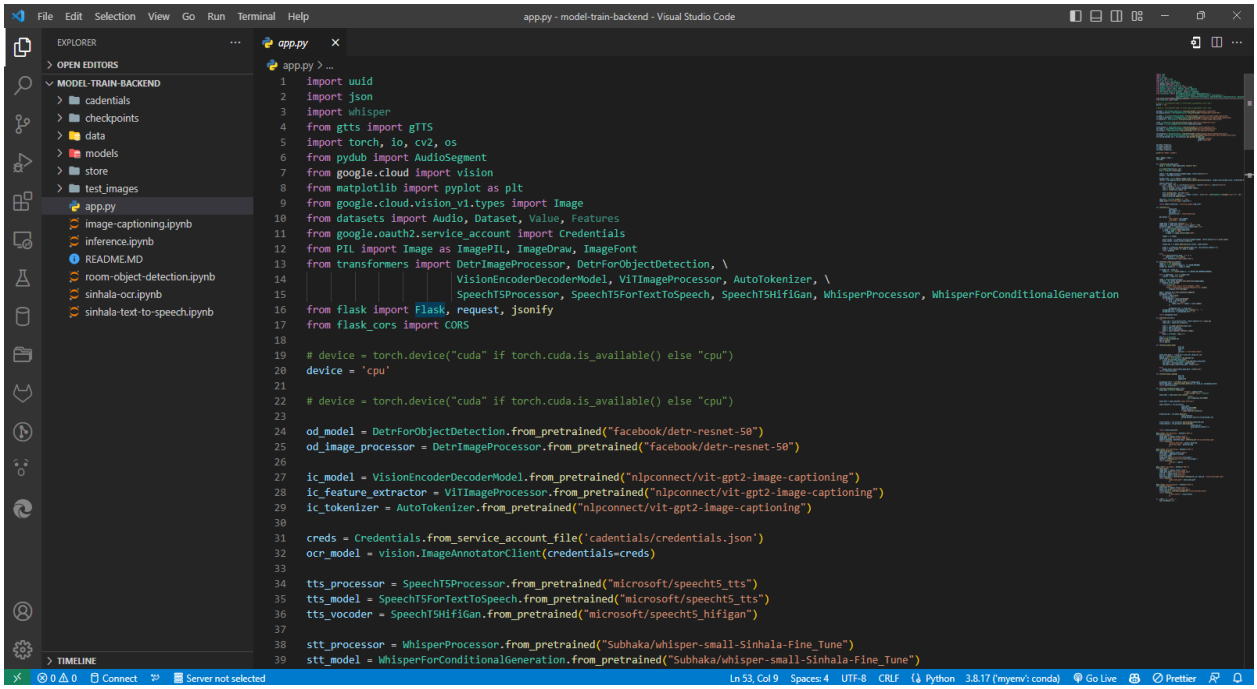


Figure 51:Model Integrated

Chapter 05

5. Testing

TABLE 3:Testcase 01

Test Case ID	**TC_01**
Test Priority Level **(Low/Medium/High)**	High
Description	Verifying that the OCR is capturing Sinhala characters correctly or not
Pre-Condition(s)	Users select the appropriate conditions
Test Procedure	Step1: Open “Audio Sight” app Step2: Go to the home page and give a Sinhala command. Step3: Capture the frame whatever you want. Step4: Listen to the output
Test Input	Sinhala words containing image
Expected Output	1. OCR should identify Sinhala words correctly
Actual Output	1. OCR identifies the Sinhala words correctly
Test Status	Pass

TABLE 4:Testcase 02

Test Case ID	**TC_02**
Test Priority Level **(Low/Medium/High)**	High
Description	Verify that the user can correctly capturing the image for the OCR.
Pre-Condition(s)	Users select the appropriate conditions
Test Procedure	Step1: Open “Audio Sight” app Step2: Go to the Home page. Step3: Capture the image
Test Input	Sinhala words containing image
Expected Output	The user should be able to capture the Sinhala letters in the image
Actual Output	They can capture the Sinhala letters in the image
Test Status	Pass

TABLE 5:Test Case 03

Test Case ID	**TC_03**
Test Priority Level **(Low/Medium/High)**	High
Description	Verify the accuracy of OCR
Pre-Condition(s)	Users select the appropriate conditions
Test Procedure	Step1: Open “Audio Sight” app Step2: Go to the Home page. Step3: Capture the image. Step 4: Compare the audio output with the input Sinhala words
Test Input	Sinhala words containing image
Expected Output	The pronunciation of the words should be similar to the words contained in the image.
Actual Output	The accurate level is good
Test Status	Pass

TABLE 6:Testcase 04

Test Case ID	**TC_04**
Test Priority Level **(Low/Medium/High)**	High
Description	Verify that the OCR send the output of captured Sinhala characters to the TTS on time
Pre-Condition(s)	Users select the appropriate conditions
Test Procedure	Step1: Open “Audio Sight” app Step2: Go to the Home page. Step3: Capture the image. Step 4: Hear to the voice output
Test Input	Sinhala words containing image
Expected Output	The output voice should hear within a small time after capturing the image
Actual Output	The output voice is coming within a small time after capturing the image
Test Status	Pass

TABLE 7:Testcase 05

Test Case ID	**TC_05**
Test Priority Level **(Low/Medium/High)**	High
Description	Verify that the TTS function is working properly
Pre-Condition(s)	Users select the appropriate conditions
Test Procedure	Step1: Open “Audio Sight” app Step2: Go to the Home page. Step3: Capture the image. Step 4: Hear to the voice output
Test Input	Sinhala words containing image
Expected Output	The output voice should hear correctly
Actual Output	The output voice is coming correctly
Test Status	Pass

TABLE 8:Testcase 06

Test Case ID	**TC_06**
Test Priority Level **(Low/Medium/High)**	High
Description	Verify that the Object Detection & Voice Navigation function is working properly.
Pre-Condition(s)	Users select the appropriate conditions
Test Procedure	Step1: Open “Audio Sight” app Step2: Go to the Home page. Step3: Capture the image and give Sihala voice commands.
Test Input	Sinhala words containing image
Expected Output	The user should be able to navigate through the app using voice commands
Actual Output	The user can navigate through the app using voice commands
Test Status	Pass

TABLE 9:Testcase 07

Test Case ID	**TC_06**
Test Priority Level **(Low/Medium/High)**	High
Description	Verify that the user can direct the camera to an image. Users focus the camera on an image in thebook.
Pre-Condition(s)	Users select the appropriate conditions
Test Procedure	Step1: Open “Audio Sight” app Step2: Go to the Home page. Step3: Hold the device up to the object or scene that the user wants to identify
Test Input	Sinhala words containing image
Expected Output	Camera should be gone on through a voicecommand
Actual Output	Camera can go on through a voicecommand
Test Status	Pass

Chapter 05

6. Conclusion

In conclusion, the development of the "Sinhala Book Reader for Visually Impaired Children" signifies a crucial milestone in promoting accessibility, education, and inclusivity. This project underscores the paramount importance of a user-centric approach, resulting in an application that is not only technically robust but, more importantly, deeply intuitive, and customizable to cater to the specific needs of visually impaired children. By seamlessly integrating with online libraries and providing content in accessible formats, it has significantly expanded the repository of Sinhala literature and educational materials available to this audience, facilitating their access to diverse, age-appropriate, and engaging content.

Furthermore, the commitment to ongoing development and user feedback underscores the adaptability of the system, ensuring that it evolves to meet the changing needs of its young users. The real power of this initiative lies in its role as an enabler for visually impaired children to embark on a literary journey. It breaks down the physical barriers that have traditionally hindered their access to books, technology, and knowledge, thereby enriching their educational prospects. The Sinhala TTS technology, accessibility features, and adherence to accessibility standards collectively create an environment where reading becomes not just an educational activity but a pleasurable and transformative experience.

As the "Sinhala Book Reader for Visually Impaired Children" transitions from development into the hands of children and their caregivers, it holds the promise of making a profound and lasting impact on the lives of its users. This project serves as an inspiration and a model for similar initiatives worldwide, with the potential to improve the lives of visually impaired individuals and to promote accessibility and inclusivity in both education and technology. In this ongoing journey, the collaboration and support of educators, caregivers, the visually impaired children themselves, and the wider community are indispensable. Together, we can ensure that the magic of reading transcends physical limitations and that every child, regardless of visual impairment, can embark on an enchanting and educational literary adventure.

7. Reference

[1] A. S. a. B. Kubendran, "Optical Character Recognition of Printed Tamil Characters," 2000.

[2] R. Smith, " An overview of the Tesseract OCR Engine," pp. pp629-633, Sep 2007.

[3] K. B. K. A. K. a. E. R. Muhammad Farid Zamir, "Smart Reader for Visually Impaired People Based on Optical Character Recognition," Department of Telecommunication Engineering, UCET, The Islamia University of Bahawalpur, Bahawalpur 63100, Pakistan.

[4] R. N. a. N. Fonseca, "Camera Reading For Blind People," Polytechnic Institute of Leiria, Leiria 2411-901 Leiria, PORTUGAL.

[5] V. B. a. R. Sinha, "A Complete OCR for Printed Hindi Text in Devnagari Script," pp. Page(s): 800-804, Sixth International Conference on Document Analysis and Recognition, IEEE Publication, Seatle USA, 2001.

[6] Velmurugan, "A Smart reader for visually impaired people using Raspberry piIJESC," https://doi.org/10.4010/2016.699. ISSN 2321 3361 ©2016.

[7] C. S. Y. ,. P. V. ,. V. Y. Raghuraj Singh1, "Optical Character Recognition (OCR) for Printed Devnagari Script Using Artificial Neural Network," pp. pp. 91-95, January-June 2010.

[8] S. M. e. al, "Historical Review of OCR Research and Development," pp. pp. 1029-1058, July 1992..

[9] "World Health Organization, “Blindness and vision impairment," 05-Mar-2021.

[10] G. N. T. N. S.V. Rice, "Optical Character Recognition: An Illustrated Guide to the Frontier," pp. pp. 57-60, Kluwer Academic Publishers, USA 1999.

[11]S. Zhou, "Open Source OCR Framework Using Mobile Devices, Multimedia on Mobile Devices 2008," 2008.

[12] A. R. C. ,. M. I. Ranjan Jana, "Optical Character Recognition from Text Image".

[13] C. G. a. X. Apostolidis, "Text Detection and Segmentation in Complex Color Images," pp. pp. 2326- 2330, 2000.

[14] M. Laine and O. S. Nevalainen, "A standalone OCR system for mobile camera-phones," pp. pp.1-5, Sept. 2006.

[15] A. W. D. H. a. V. W. Ruvan Weerasinghe, "NLP Applications of Sinhala: TTS & OCR".

[16]"Optical Character Recognition. Retrieved from: http://en.wikipedia.org/wiki/Optical_character_recog nition," 2007.

[17] A. M. O. Azham Hussain(*), "Usability Evaluation Model for Mobile Visually Impaired Applications".

[18] L. N. S. D. S. a. S. D. Jayasinghe, "Optical Character Recognition for Sinhala Language using Deep Learning Techniques"".

[19] A. P. Gerratt, M. Tellers and S. Bergbreiter, "Soft polymer MEMS," 2011 IEEE 24th International Conference on Micro Electro Mechanical Systems, Cancun, Mexico, 2011, pp. 332-335, doi: 10.1109/MEMSYS.2011.5734429.

[20] T. -T. -H. Nguyen, A. Jatowt, M. Coustaty, N. -V. Nguyen and A. Doucet, "Deep Statistical Analysis of OCR Errors for Effective Post-OCR Processing," 2019 ACM/IEEE Joint Conference on Digital Libraries (JCDL), Champaign, IL, USA, 2019, pp. 29-38, doi: 10.1109/JCDL.2019.00015.

[21] A. Kokawa, L. S. P. Busagala, W. Ohyama, T. Wakabayashi and F. Kimura, "An Impact of OCR Errors on Automated Classification of OCR Japanese Texts with Parts-of-Speech Analysis," 2011 International Conference on Document Analysis and Recognition, Beijing,China, 2011, pp. 543-547, doi: 10.1109/ICDAR.2011.115.

[22] A. Beg, F. Ahmed and P. Campbell, "Hybrid OCR Techniques for Cursive Script Languages – A Review and Applications," 2010 2nd International Conference on Computational Intelligence, Communication Systems and Networks, Liverpool, UK, 2010, pp. 101-105, doi: 10.1109/CICSyN.2010.36.

[23] V. Mohane and C. Gode, "Object recognition for blind people using portable camera," 2016 World Conference on Futuristic Trends in Research and Innovation for Social Welfare (Startup Conclave), Coimbatore, India, 2016, pp. 1-4, doi: 10.1109/STARTUP.2016.7583932.

[24] S. Deshpande and R. Shriram, "Real time text detection and recognition on hand held objects to assist blind people," 2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT), Pune, India, 2016, pp. 1020-1024,doi : 10.1109/ICACDOT.2016.7877741.

[25] R. Prabha, M. Razmah, G. Saritha, R. Asha, S. G. A and R. Gayathiri, "Vivoice - Reading Assistant for the Blind using OCR and TTS," 2022 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 2022, pp. 01-07, doi: 10.1109/ICCCI54379.2022.9740877.

[26] WHO, World report on vision, vol. 214, no. 14. 2019. [Online]. Available: https://www.who.int/publications-detail/world-report-on-vision.

[27] D. S. S. De Zoysa, J. M. Sampath, E. M. P. De Seram, D. M. I. D. Dissanayake, L. Wijerathna, and S. Thelijjagoda, “Project Bhashitha - Mobile based optical character

recognition and text-to-speech system,” 13th Int. Conf. Comput. Sci. Educ. ICCSE 2018, no. Iccse, pp. 623–628, 2018, doi: 10.1109/ICCSE.2018.8468858.

[28] M. Awad, J. El Haddad, E. Khneisser, T. Mahmoud, E. Yaacoub, and M. Malli, “Intelligent eye: A mobile application for assisting blind people,” 2018 IEEE Middle East North Africa Commun. Conf. MENACOMM 2018, no. September, pp. 1–6, 2018, doi: 10.1109/MENACOMM.2018.8371005.

[29] P. Jayawardhana, A. Aponso, N. Krishnarajah, and A. Rathnayake, “An Intelligent Approach of Text-To-Speech Synthesizers for English and Sinhala Languages,” 2019

IEEE 2nd Int. Conf. Inf. Comput. Technol. ICICT 2019, no. May, pp. 229–234, 2019, doi: 10.1109/INFOCT.2019.8711051.

[30] A. Mishangi, “Android based sinhala document reader for visually impaired people,” 2021.

[31] S. Gallege, “Analysis of Sinhala Using Natural Language Processing Techniques,” 2010, [Online]. Available: www.defence.lk.

[32] S. Y. Senanayake, K. T. P. M. Kariyawasam, and P. S. Haddela, “Enhanced Tokenizer for Sinhala Language,” 2019 Natl. Inf. Technol. Conf. NITC 2019, pp. 8–10, 2019, doi:10.1109/NITC48475.2019.9114420.

[33] R. Weerasinghe, A. Wasala, D. Herath, and V. Welgama, “NLP applications of Sinhala: TTS & OCR,” IJCNLP 2008 - 3rd Int. Jt. Conf. Nat. Lang. Process. Proc. Conf., vol. 2, pp. 963–966, 2008.

[34] A. Joy* and D. R. Saranya, “A Pilot Research on Android Based Voice Recognition Application,” Int. J. Recent Technol. Eng., vol. 8, no. 4, pp. 7272–7277, 2019, doi:10.35940/ijrte.d5284.118419.

[35] G. Na, G. N. Surname, G. N. Surname, G. N. Surname, G. N. Surname, and G. N. Surname, “Mobile Base Sinhala Book Reader For The Visually Impaired Individuals,” 2007.

[36] A. A. Kumar, B. Senthilvasudevan, and H. U. Farhan, “Translation of Multilingual Text into Speech for Visually Impaired Person,” 7th Int. Conf. Commun. Electron. Syst. ICCES 2022 - Proc., no. Icces, pp. 60–64, 2022, doi: 10.1109/ICCES54183.2022.9835819.

[37] D. H. Klatt, “Review of text-to-speech conversion for English,” J. Acoust. Soc. Am., vol. 82, no. 3, pp. 737–793, 1987, doi: 10.1121/1.395275.

[38] World Health Organization. (2019). Visual impairment and blindness. Retrieved from https://www.who.int/news-room/fact-sheets/detail/blindness-and-visual-impairment.

[39] Sri Lanka National Federation of the Visually Handicapped. (2021). About us. Retrieved from https://www.visuallyhandicapped.lk/about_us.html.

[40] Wijayasinghe, I., Ranathunga, I., Kumara, H., & Wijekoon, A. (2016). A survey on assistive technologies for the visually impaired. Journal of Engineering Research, 4(1), 87-100.

[41] Perera, T., & Rupasinghe, T. (2020). Accessibility of digital resources for visually impaired individuals in Sri Lanka. In Proceedings of the 3rd International Conference on Advancements in Computing (pp. 203-208). IEEE.

[42] Herath, D., & Wijesinghe, P. (2019). An accessibility framework for the visually impaired people to use mobile applications. In Proceedings of the 4th International Conference on Advances in Computing and Technology (pp. 1-7). IEEE.

[43] Hassan, M., Ahmad, M., Ali, S., & Qureshi, R. (2019). An assistive tool for blind people to read text from books using image processing. In Proceedings of the 3rd International Conference on Intelligent Computing and Control Systems (pp. 266-270). IEEE.

[44] Iqbal, M., Hussain, A., & Baig, I. (2020). A comparative study of assistive technologies for visually impaired individuals. In Proceedings of the 2nd International Conference on Computer Science, Engineering and Applications (pp. 1-6). IEEE

[45] Silva, N., Ranasinghe, K., & Gamage, D. (2019). A mobile-based system for Sinhala text recognition and speech synthesis. In Proceedings of the 15th IEEE International Conference on Industrial and Information Systems (pp. 188-193). IEEE.

[46] Jayasundara, C., Arachchi, H., & Fernando, A. (2020). Sinhala text-to-speech system for visually impaired individuals. In Proceedings of the 4th International Conference on Intelligent Computing and Control Systems (pp. 296-299). IEEE.

[47] Munasinghe, T., Weerasena, J., & Kankanamge, U. (2021). Braille transcribing tool for Sinhala text. In Proceedings of the 7th International Conference on Advances in Computing and Technology (pp. 120-125). IEEE.

[48] N. D. K.N.Withanage, "Towards ICT based Solution for Stuttering," 08 2017. [Online]. Available: https://www.researchgate.net/publication/319058777_Towards_ICT_based_Solution_for_Stuttering. [Accessed 21 01 2022].

[49] f. e. staff, "Stuttering," 02 2021. [Online]. Available: https://familydoctor.org/condition/stuttering/. [Accessed 21 01 2022].

[50] G. Williamson, "The Nature of Stuttering," 30 01 2014. [Online]. Available: https://www.sltinfo.com/the-nature-of-stuttering/. [Accessed 22 01 2022].

[51] "Stuttering," [Online]. Available: https://en.wikipedia.org/wiki/Stuttering. [Accessed 21 01 2022].

[52]"Stuttering Modification Therapy," [Online]. Available: https://speecheasy.com/stuttering-modification-therapy/#:~:text=Stuttering%20modification%20therapy%20is%20a,be%20removed%20from%20speaking%20situations.[Accessed 22 01 2022].

[53] M. M. J. H. S. Hector R. Perez, "Stuttering," 06 2016. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4907555/. [Accessed 22 01 2022].

[54] "Technologies for Stuttering," [Online]. Available: https://casafuturatech.com/daf-devices/. [Accessed 31 01 2022].

[55] S. A. Ling Qiu, "Voice Assistants for Speech Therapy," [Online]. Available: https://ubicomp-mental-health.github.io/papers/2021/ubicomp21g-sub1960-cam-i7.pdf. [Accessed 01 02 2022].

[56] D. E. a. K. Dalton, "SPEECHEASY," [Online]. Available: https://www.mnsu.edu/comdis/kuster/journal/hallen/speecheasy.html. [Accessed 01 02 2022].

Appendix

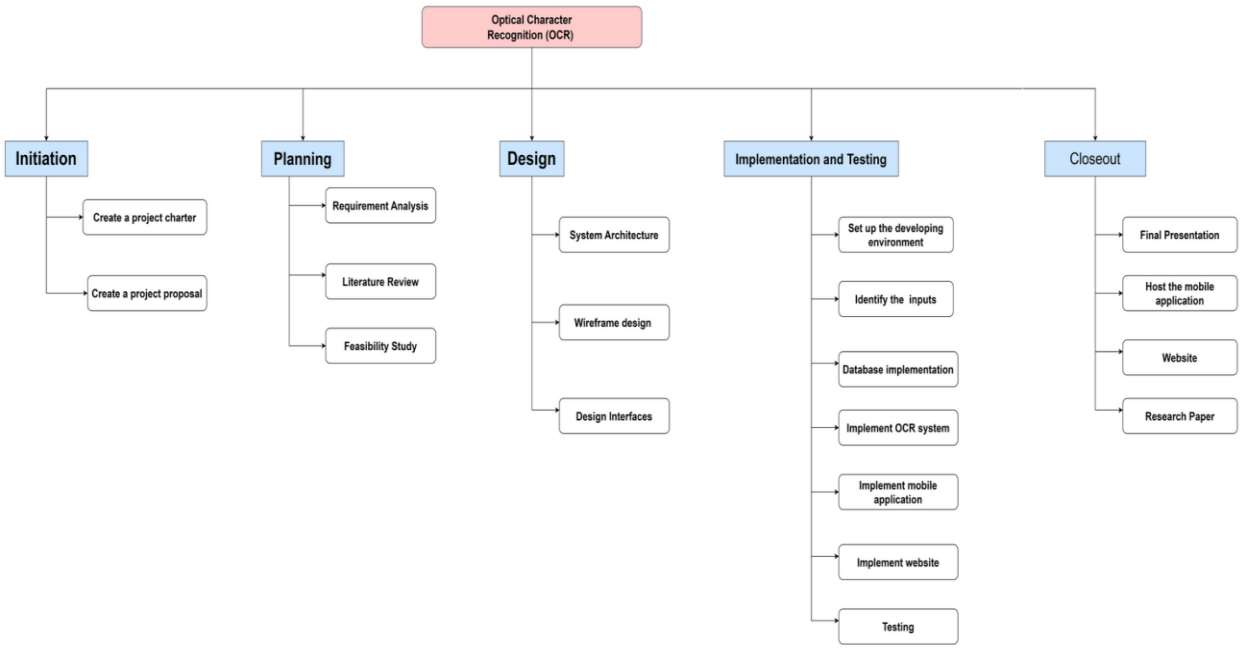


Figure 52:Work breakdown chart OCR

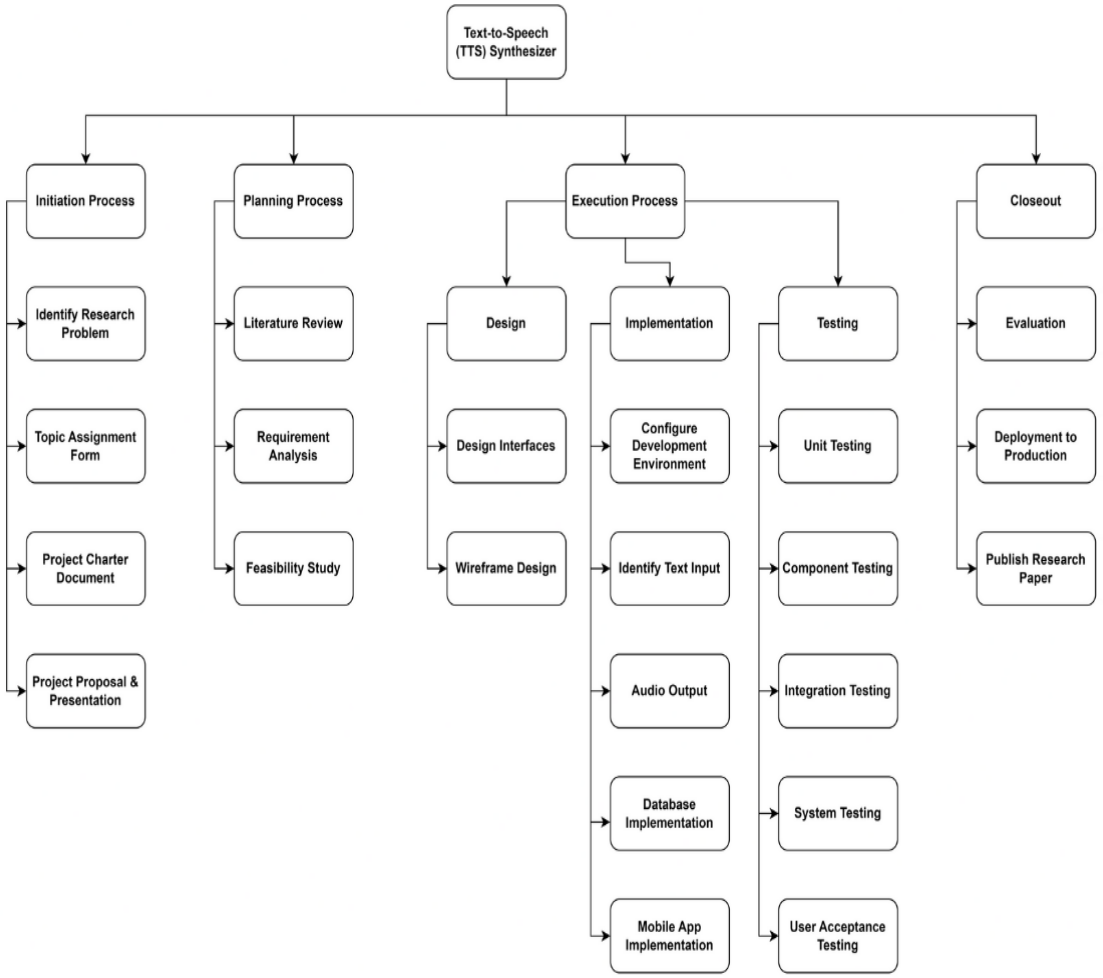


Figure 53:Work breakdown chart TTS

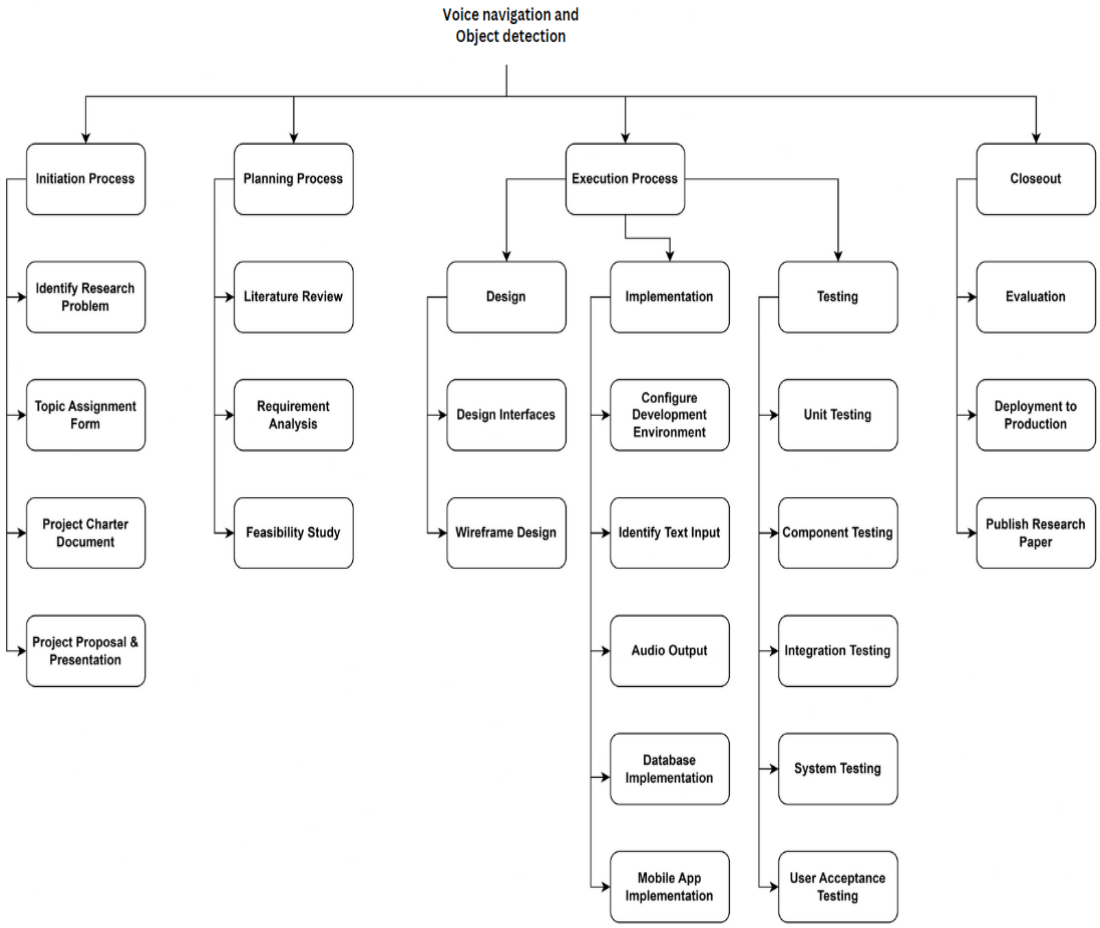


Figure 54:Work breakdown chart voice navigation and object detection

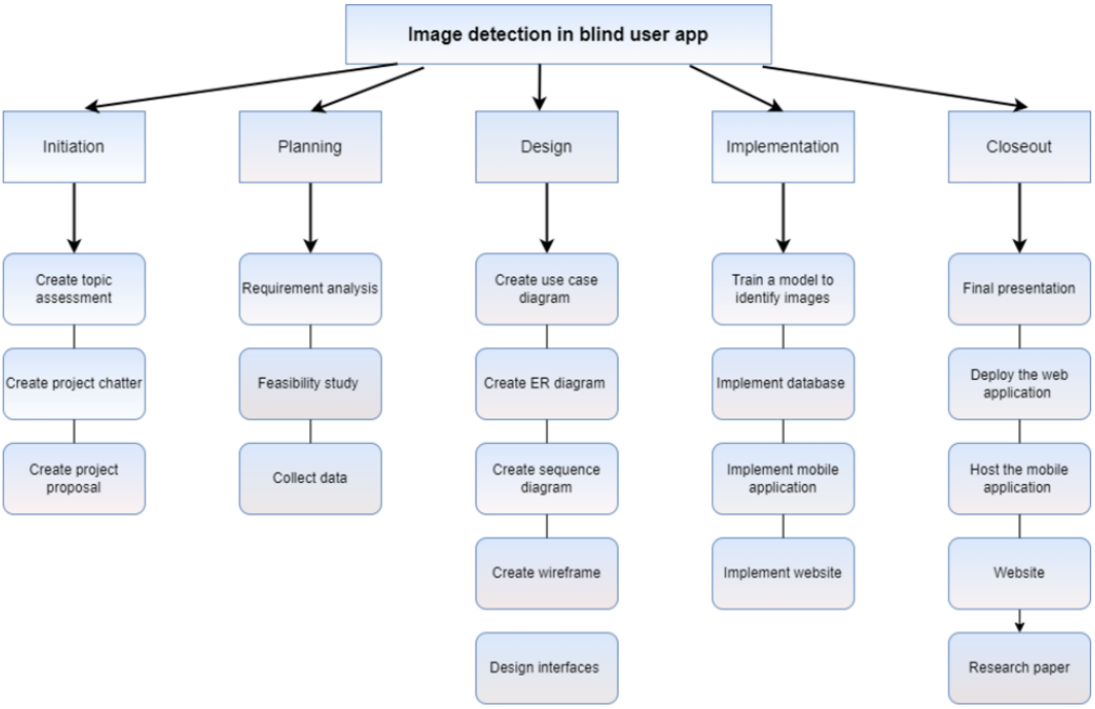


Figure 55:Work breakdown chart image captioning

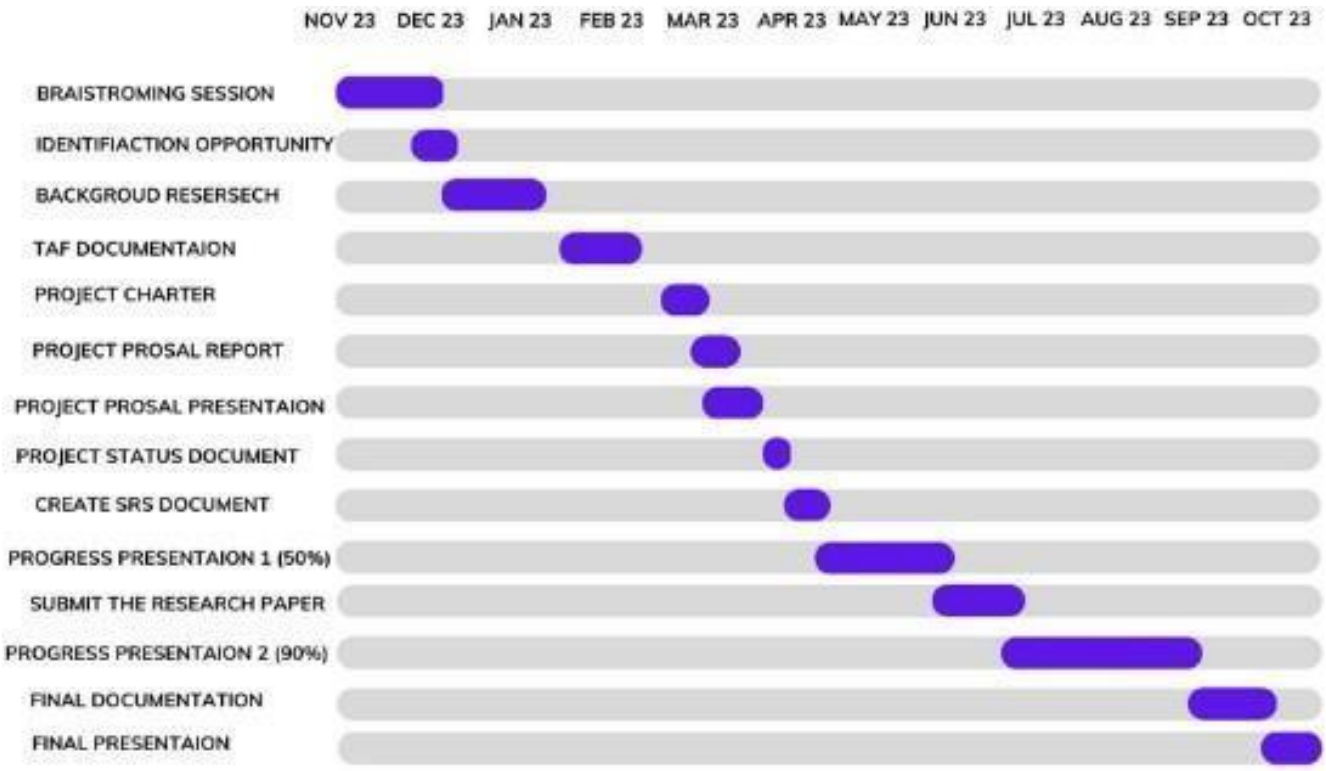


Figure 56:Grant chart

```
In [2]: import tensorflow as tf
        from tqdm import tqdm
        import pandas as pd
        import numpy as np
        import os, shutil, math, cv2, json, random, re, sys, datetime, glob
        import matplotlib.pyplot as plt
        import seaborn as sns
        from sklearn.model_selection import train_test_split
        from PIL import Image
        from tensorflow.keras.initializers import LecunNormal, Zeros, RandomNormal, GlorotNormal, GlorotUniform
        ##from keras.initializers.initializers_v2 import LecunNormal, Zeros, RandomNormal, GlorotNormal, GlorotUniform
        from keras.layers import *
        from sklearn.metrics import accuracy_score, precision_recall_fscore_support, f1_score, classification_report, confusion_matrix
```

Training accuracy: 28%

```
with open('classes_indices.json', 'w') as json_file:
    json_file.write(json_str)

every_class_num = []
supported = ['.jpg', '.png', '.jpeg', '.PNG', '.JPG', '.JPEG']

for klass in classes:
    classpath = os.path.join(root, klass)
    images = [os.path.join(root, klass, i) for i in os.listdir(classpath) if os.path.splitext(i)[-1] in supported]
    every_class_num.append(len(images))
    flist = sorted(os.listdir(classpath))[0: 1000]
    desc = f'{klass:23s}'
    for f in tqdm(flist, ncols=110, desc=desc, unit='file', colour='blue'):
        fpath = os.path.join(classpath, f)
        f1 = f.lower()
        index = f1.rfind('.')
        ext = f1[index:]
        if ext in supported:
            try:
                img = cv2.imread(fpath)
                filepaths.append(fpath)
                labels.append(klass)
            except:
                bad_images.append(fpath)
                print('defective image file: ', fpath)
        else:
```

```
class VisionTransformer(tf.keras.Model):
    def __init__(self, img_size=224, patch_size=16, embed_dim=768, depth=12, num_heads=12, qkv_bias=True,
                 qk_scale=None, drop_ratio=0, attn_drop_ratio=0, drop_path_ratio=0, representation_size=None,
                 num_class=1000, name='ViT-B/16'):
        super(VisionTransformer, self).__init__(name=name)
        self.num_classes = num_class
        self.embed_dim = embed_dim
        self.depth = depth
        self.qkv_bias = qkv_bias

        self.patch_embed = PatchEmbed(img_size=img_size, patch_size=patch_size, embed_dim=embed_dim)
        num_patches = self.patch_embed.num_patches
        self.cls_token_pos_embed = ConcatClassTokenAddPosEmbed(embed_dim=embed_dim,
                                                               num_patches=num_patches,
                                                               name='cls_pos')
        self.pos_drop = Dropout(drop_ratio)

        dpr = np.linspace(0, drop_path_ratio, depth)
        self.blocks = [Block(dim=embed_dim, num_heads=num_heads, qkv_bias=qkv_bias, qk_scale=qk_scale,
                             drop_ratio=drop_ratio, attn_drop_ratio=attn_drop_ratio, drop_path_ratio=dpr[i],
                             name='encoderblock_{}'.format(i)) for i in range(depth)]
```

Figure 57:voice navigation training process accuracy

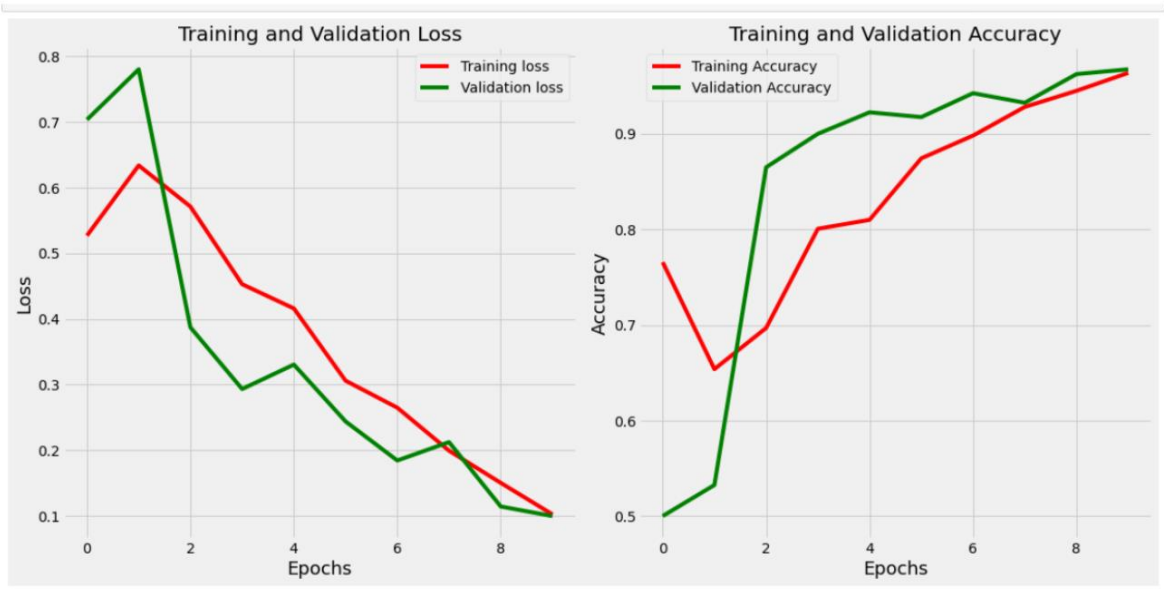


Figure 58:voice navigation training process accuracy

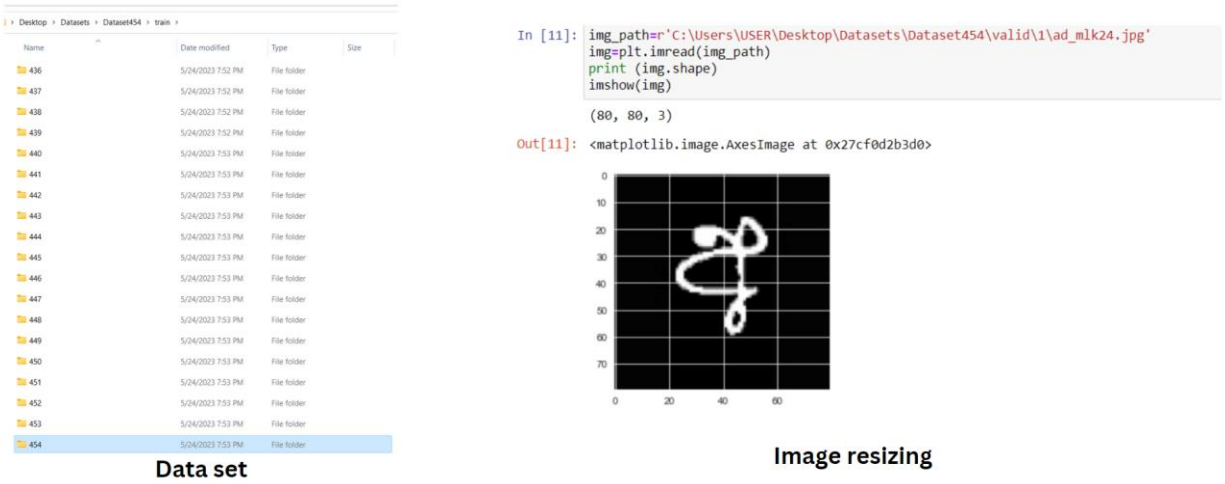


Figure 59:OCR training process

```
In [14]: sdir=r'C:\Users\USER\Desktop\Datasets\Dataset454\dataset'
         trsplit=.9
         train_df, test_df, valid_df= preprocess(sdir,  trsplit)

train_df length:  9806   test_df length:  10896   valid_df length:  1090
[22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 2
2, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22,
22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22,
22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22,
22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22,
22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22,
22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22,
22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22,
22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 22, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21,
21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21,
21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21,
21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21,
21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21,
21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21,
21, 21, 21, 21, 21, 21, 21]
```

Figure 60:OCR training process

```
import tensorflow as tf
from keras.preprocessing.image import ImageDataGenerator
from keras.models import Sequential
from keras.layers import Conv2D, MaxPooling2D, Flatten, Dropou
from keras.callbacks import EarlyStopping, ModelCheckpoint
```

```
# Set up data generators
train_datagen = ImageDataGenerator(
    rescale=1./255,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True
)
```

```
# Set up data generators
train_datagen = ImageDataGenerator(
    rescale=1./255,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True
)

test_datagen = ImageDataGenerator(rescale=1./255)

training_set = train_datagen.flow_from_directory(
    'C:\\Users\\Azend\\Story_book\\Train',
    target_size=(48, 48),
    batch_size=32,
    color_mode="grayscale",
    class_mode='categorical'
)

test_set = test_datagen.flow_from_directory(
    'C:\\Users\\Azend\\Story_book\\Test',
    target_size=(48, 48),
    batch_size=32,
    color_mode="grayscale",
    class_mode='categorical'
```

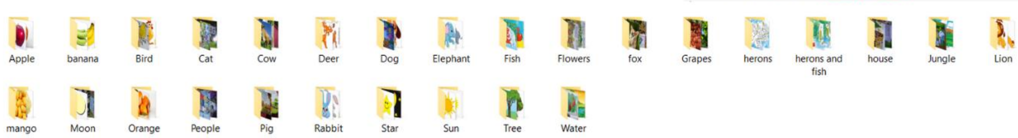
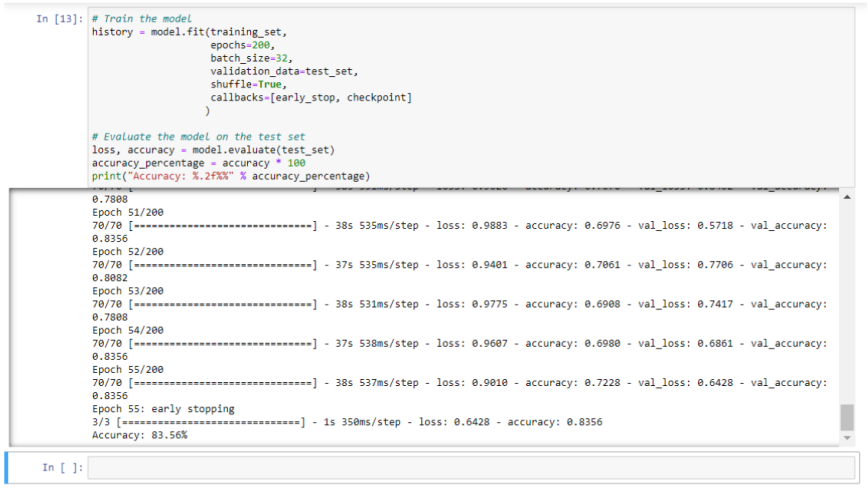


Figure 61:Data Collection, Data Augmentation, and Data Preprocess for image captioning



Training accuracy: 83.56%

Figure 62:Training accuracy for image captioning

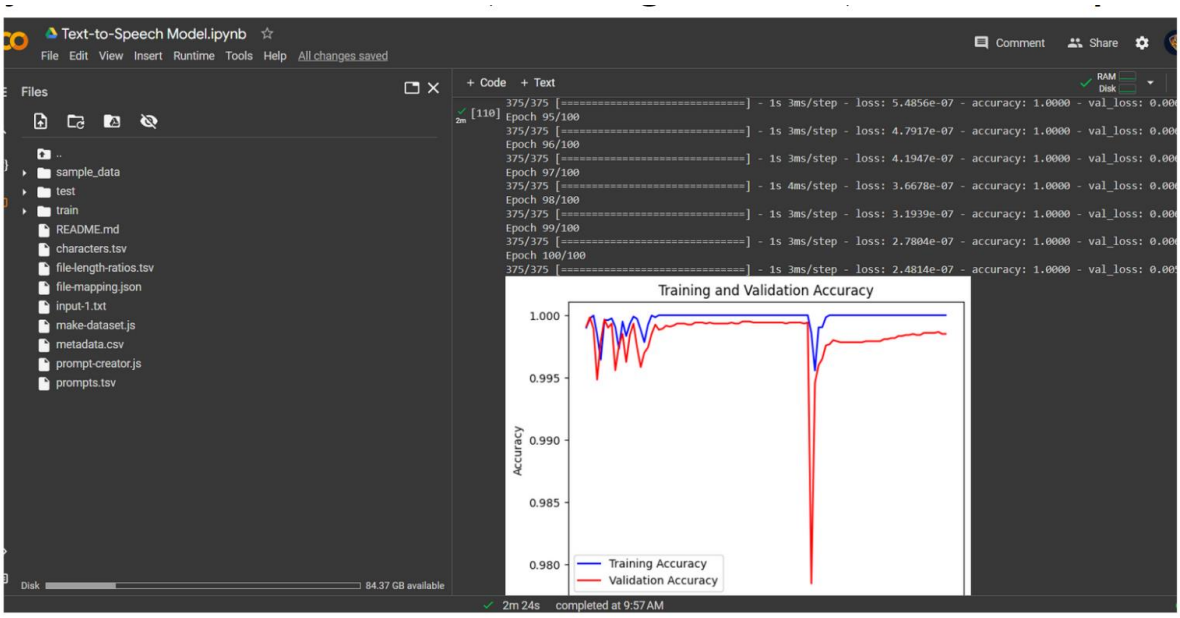


Figure 63:TTS Training and validation accuracy

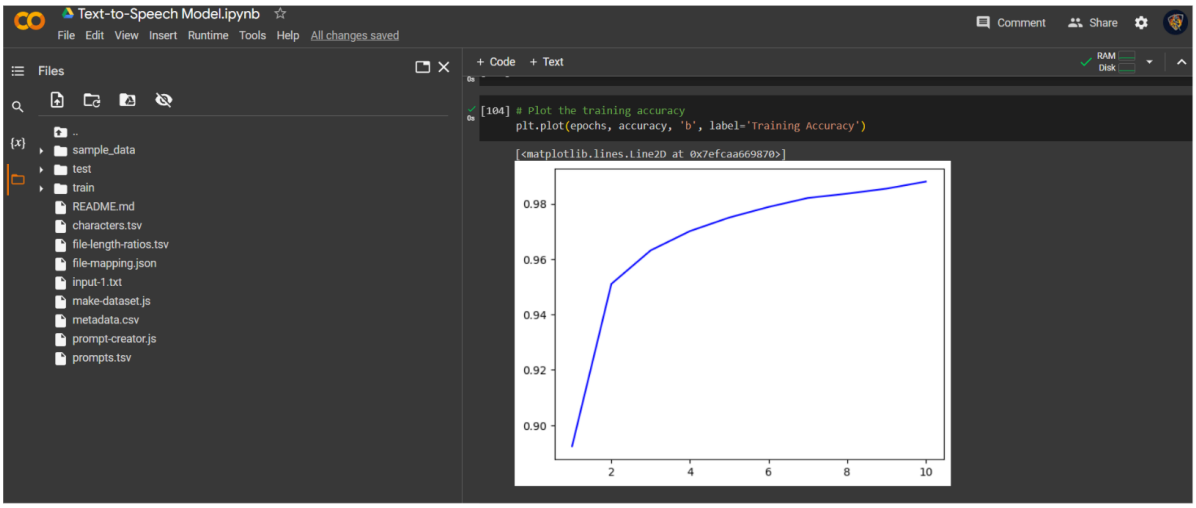


Figure 64:TTS Data Collection, Data Augmentation, and Data Preprocess

SRI LANKA
INSTITUTE OF INFORMATION TECHNOLOGY

16th Floor, BoC Merchant Tower, No. 28, St. Michael's Road, Colombo 03

Date:28/04/2023 Your Ref : My Ref : 2023-198

The Ceylon School for the Deaf &
Blind RDC Donations office 521,
Galle Road,
Ratmalana
Sri Lanka

Dear Sir / Madam,

Certifying the project titled "Mobile Base Sinhala Book Reader for Visually Impaired Individuals" is conducting as a BSc in IT final year research project.

The Sri Lanka Institute of Information Technology (SLIIT) is the largest Degree Awarding Institute in the field of information Technology recognized by the University Grants Commission under the Universities Act. It was established in the year 1999 to educate and train Information Technology (IT) Professionals required by the fast-growing IT Industry in Sri Lanka.

This letter is to certify that the following students.
IT20146238 - Jayathunga T.M.
IT20129712 - Godakanda P.G.S.
IT20241346 - Semini J.P.D.L.
IT20254520 - Bhagya H.D.M.

They are final year undergraduate students who conduct research entitled "Mobile Base Sinhala Book Reader for Visually Impaired Individuals" as partial fulfillment of the B.Sc. in Information Technology degree at Sri Lanka Institute of Information Technology (SLIIT). The students are conducting the research under the supervision of Prof. Koliya Pulasinghe

I kindly request your assistance in enabling these students to collect data from your organization to build their dataset for the research project. If you have any questions or require further clarification about the project, please do not hesitate to contact me.

Thank you for your cooperation

Dr. Jayantha Amararachchi
Assistant Professor/
Research Project Coordinator,
jayantha.a@sliit.lk
+94 11 754 4103

Tel: +94(0)11 2301904 - 5 Fax: +94(0)11 2301906 E-mail: info@sliit.lk URL: www.sliit.lk

Figure 65:Field visit approval form from SLIIT

GitLab Link: http://gitlab.sliit.lk/23-198/mobile-base-sinhala-book-reader-for-visually-impaired-individuals

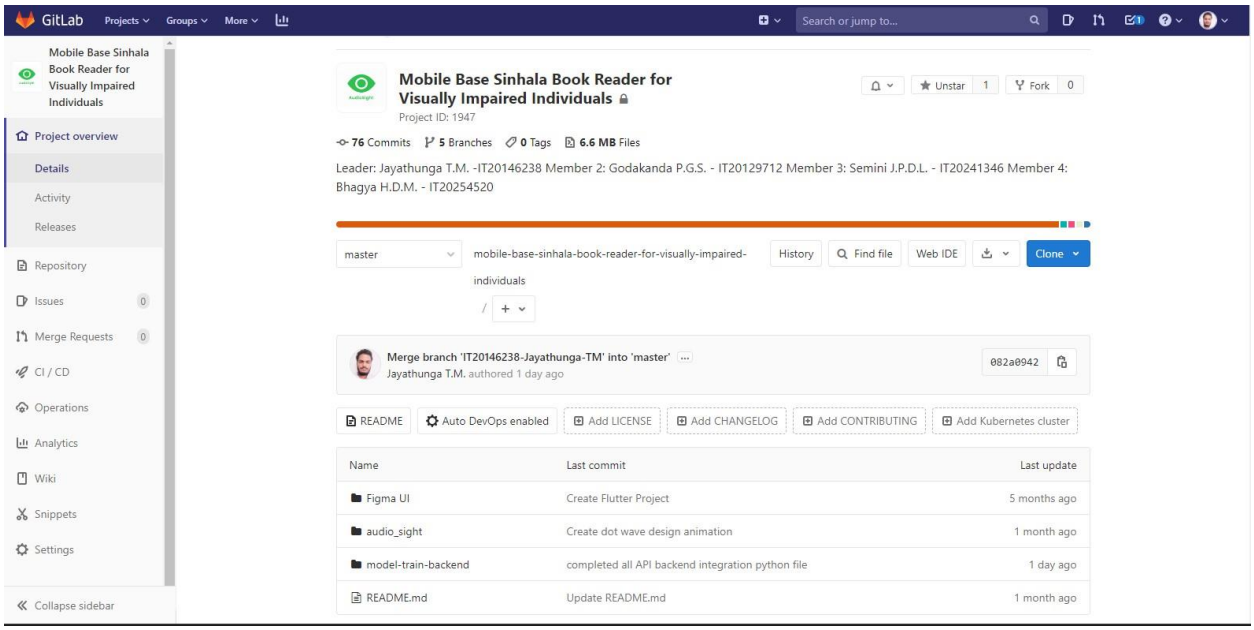


Figure 66:GitLab Project Overview

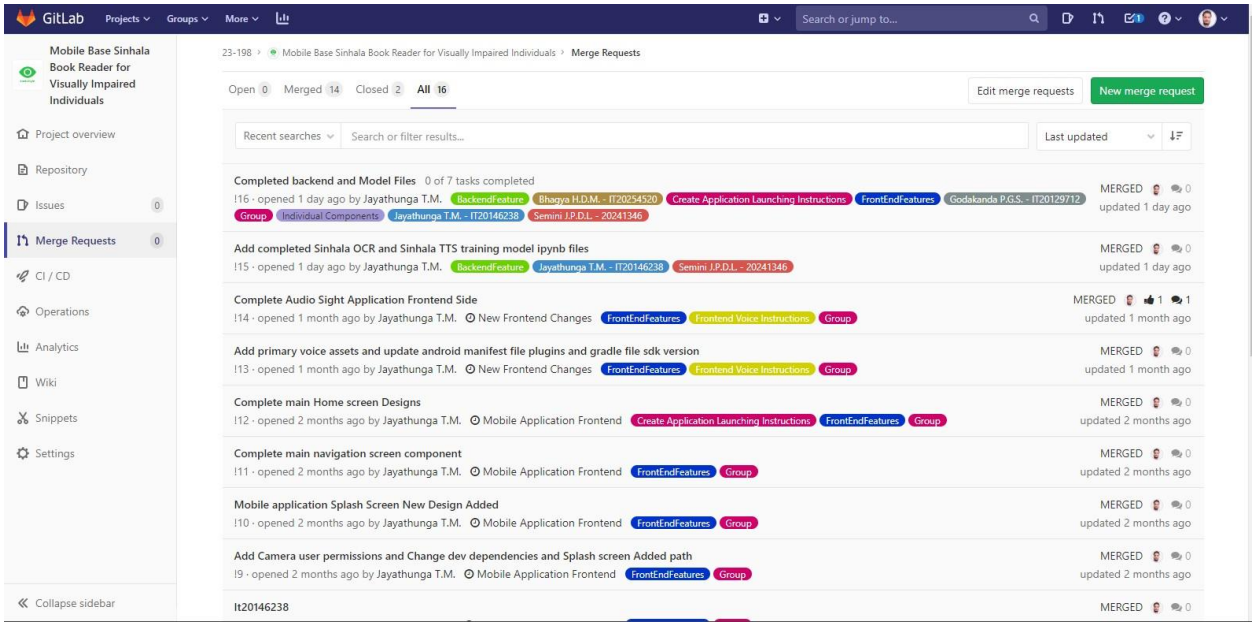


Figure 67:GitLab Merge Requests

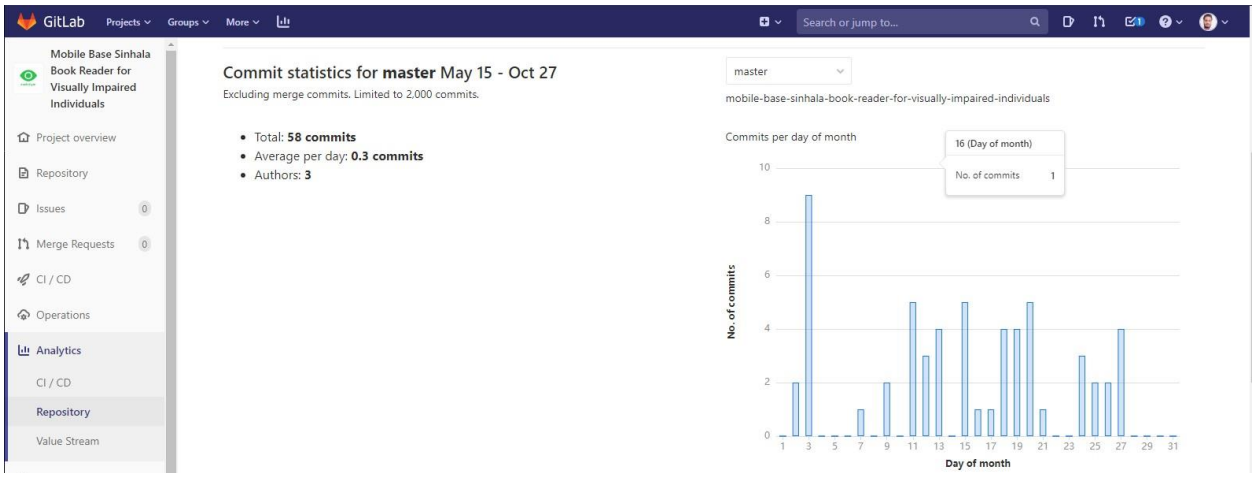


Figure 68:GitLab Monthly Commits

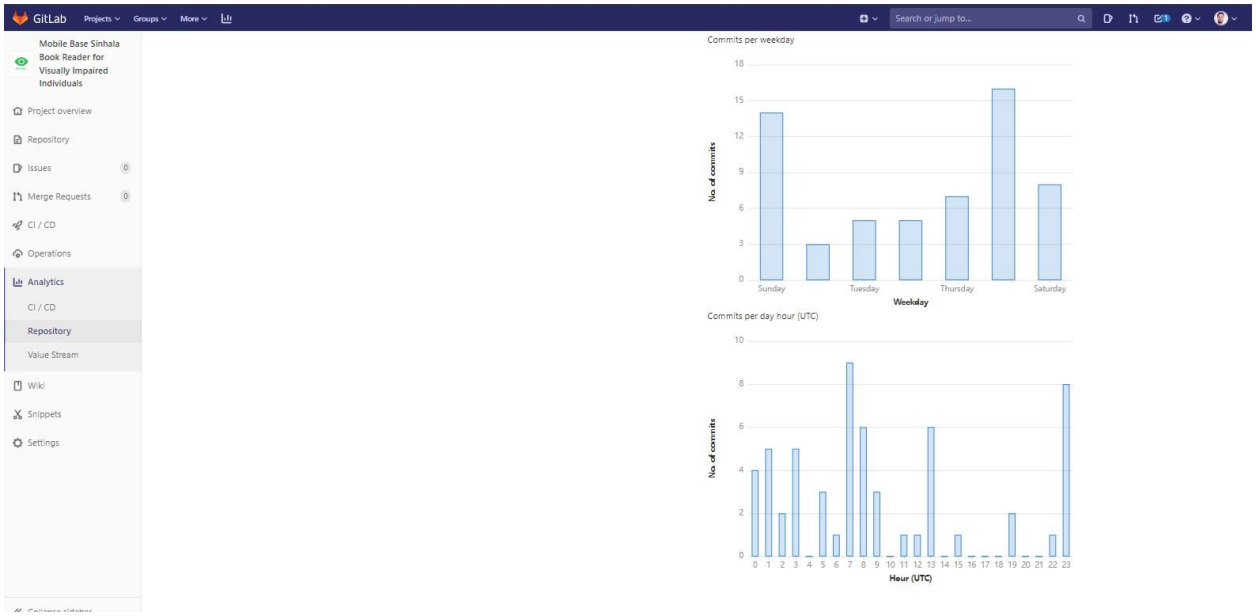


Figure 69:GitLab Commits Weekly and Hourly Analysis Charts

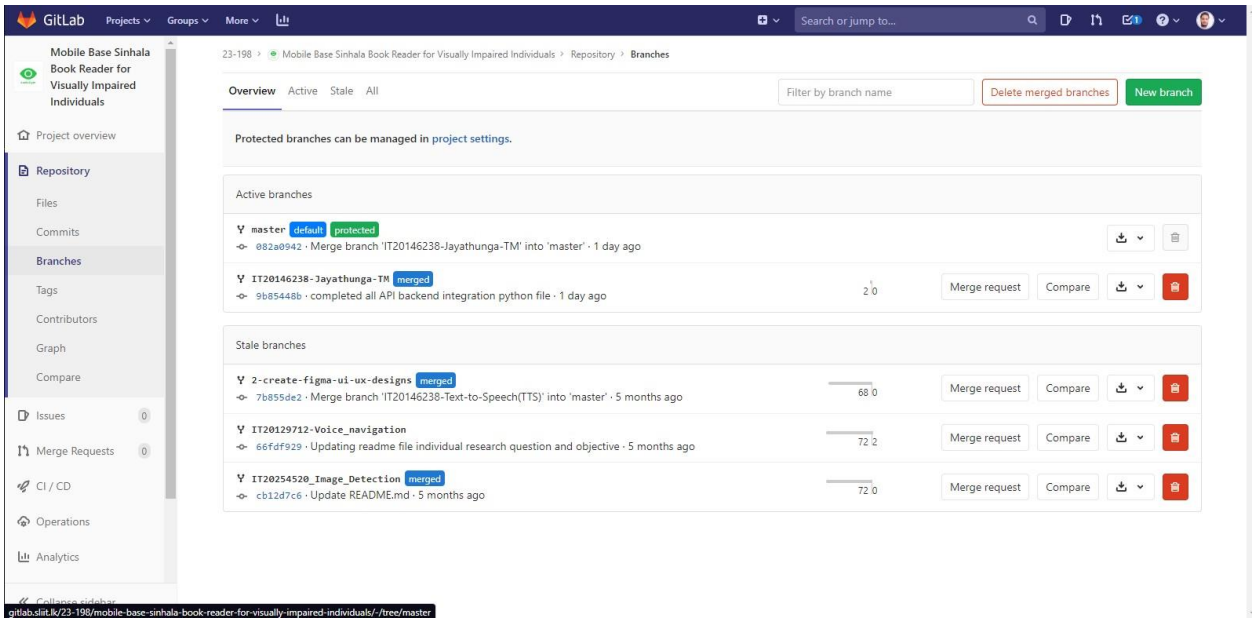


Figure 70:GitLab Branches

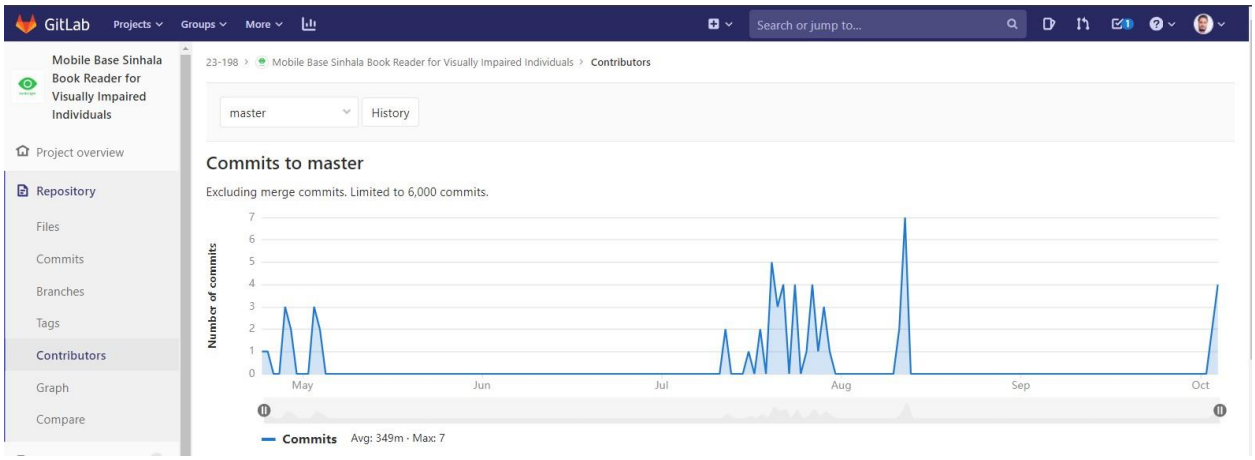


Figure 71:GitLab All Members Contribution Charts

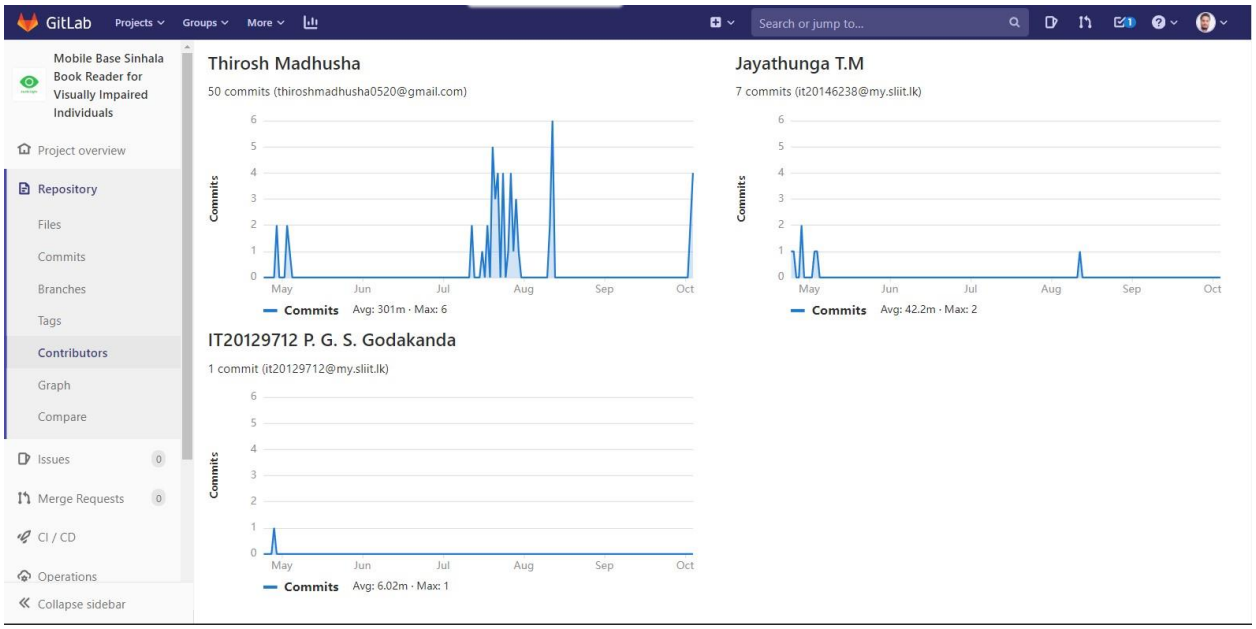


Figure 72:GitLab Contributions Charts