DOCUMENT READING SYSTEM FOR BLIND PEOPLE ("READING EYE")

19-20-J 17

Software Requirements Specification

IT16079328
W.R.P.Fernando

Ms. Suranjini Silva

B.Sc. Special (Honors) Degree in Information Technology

Department of Information Technology

Sri Lanka Institute of Information Technology

Sri Lanka

September 2019

DOCUMENT READING SYSTEM FOR BLIND PEOPLE ("READING EYE")

19-20-J 17

Software Requirements Specification

(SRS documentation submitted in partial fulfilment of the requirement for the Degree of Bachelor of Science Special (honors) In Information Technology)

W.R.P.Fernando

Ms. Suranjini Silva

B.Sc. Special (Honors) Degree in Information Technology

Department of Information Technology

Sri Lanka Institute of Information Technology

Sri Lanka

September 2019

DECLARATION

I declare that this is my own work and this system requirement specification does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any other university or Institute of higher learning and to the best of our knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Name	Student ID	Signature	Date
W.R.P.Fernando	IT16079328		

The above candidates are carrying out research for the undergraduate Dissertation under my supervision.

…….. …..…...…….……

Signature of the supervisor Date

…….. …..…...…….……

Signature of the co-supervisor Date

TABLE OF CONTENT

DECLARATION iii
TABLE OF CONTENT iv
LIST OF TABLES vi
LIST OF FIGURES vii
1 Introduction 1
1.1 Purpose 1
1.2 Scope 1
1.3 Definitions, Acronyms, and Abbreviations 2
1.4 Overview 3
1.4.1 Main goal of the system 3
1.4.2 Specific goals 3
1.4.3 Users 3
1.4.4 Reading Suggestions 4
2 Overall Descriptions 5
2.1 Product Perspective 6
2.1.1 System Interfaces 7
2.1.2 User Interfaces 7
2.1.3 Hardware Interfaces 7
2.1.4 Software Interfaces 8
2.1.5 Communication Interfaces 8
2.1.6 Memory Constraints 8
2.1.7 Operations 8
2.1.8 Site Adaptation Requirements 9
2.2 Product Functions 9
2.3 User Characteristics 15
2.4 Constraints 16
2.5 Assumptions and Dependencies 16
2.6 Apportioning of Requirements 16
3 Specific Requirements 18
3.1 External Interface Requirements 18
3.1.1 User Interfaces 18
3.1.2 Hardware Interfaces 18
3.1.3 Software Interfaces 18

3.1.4 Communication Interfaces 18
3.2 Classes/Objects 19
3.3 Performance Requirements 19
3.4 Design Constraints 20
3.5 Software System Attributes 20
3.5.1 Reliability 20
3.5.2 Availability 21
3.5.3 Security 21
3.5.4 Maintainability 21
3.6 Other Requirements 21
4 Supporting Information 22
4.1 References 22

LIST OF TABLES

Table 1 – Definition for the terms used in this SRS……..2

Table 2 – Glossary of Acronyms……...2

Table 3 – Comparison of the existing products with proposed system…….6

Table 4 - Use Case Scenario of Retrieving files from folders……..12

Table 5 – Use Case of Analyzing Images……12

Table 6 - Use Case of Analyzing Tables…….13

Table 7 - Use Case of Generating Descriptions……..13

Table 8 - Use Case of Generating Audio Files……14

Table 9 – Use Case of User Listening to the Audio Narration……14

LIST OF FIGURES

Figure 1 – High-level architecture diagram for ITR component……9

Figure 2 – Use case diagram for ITR component……11

Figure 3 – Class diagram of ITR component……..19

1 Introduction

1.1 Purpose

The prime purpose of this document is to detail the functional, non-functional requirements, dependencies and all the relevant specifics of the Images and Table Reading (ITR) component of the proposed Reading Eye system. The document will further depict the purpose, key features, product perspective, functionality flow, end-user characteristics, constraints under which the system should perform and various external interface requirements such as system interface, user interface, software interface, hardware interface and communication interface of the proposed component. The intended audience for this document is the supervisor Ms. Suranjini Silva, co-supervisor Mr. Anuradha Jayakody the research team members, any individuals with good computer literacy and all type of stakeholders interested in this research.

1.2 Scope

This document comprises the full depth description of the ITR component which will read images and tabular data in any kind of printed document. Furthermore, the document illustrates all the tools and technologies, applied concepts for implementation, the flow of the system through use case diagrams, use case scenarios and other related Unified Modeling Language diagrams, libraries and external interfaces allied to ITR component.

The proposed component utilizes the Cloud Computing technologies along with Deep Learning (DL) techniques to process data and uses cross mobile platform to present the outcome to the end-user. Moreover, the Optical context of character recognition (OCR) is used with natural language processing (NLP) to identify characters in a digital image and form cohesive descriptions. The final outcome of will be an audio file of the generated descriptions which will be created using a text-to-speech API.

Therefore, this document will guide through the proposed methodology to accomplish the target objectives of the proposed component.

1.3 Definitions, Acronyms, and Abbreviations

Term	Definition
Reading Eye	The name of the proposed system.
TensorFlow	Open-source machine learning framework
Keras	High-level neural networks API which runs on top of TensorFlow.
Natural Language Processing	Artificial Intelligence-based component which makes computers to read the text, hear speech, interpret it, measure sentiment and determine which parts are important.

Table 1-Definition for the terms used in this SRS.

Acronym/Abbreviation	Definition
ITR	Images and Tables based contents Reading and Detection.
DL	Deep Learning
NLP	Natural Language Processing
API	Application Programming Interface

Table 2-Glossary of Acronyms.

1.4 Overview

ITR is a component of the proposed Reading Eye system. This will read images and tabular data in any kind of printed document. In order to achieve this goal, the system will be implemented using Deep Learning methodologies supported by mobile and cloud platforms. The component will use photos of printed documents which will be captured by mobile camera. These images will then be uploaded to the cloud to be used as the input data for the component.

The highlighting part of this component is the accuracy level of identifying objects in an image and data in a table. Objects in an image and data in a table can be varied vastly. Therefore, they should be analyzed deeply to identify the correct object and data. The output will be presented in a voice format to the end-user to listen and understand.

1.4.1 Main goal of the system

The main goal of the proposed system is to read text, images, charts, tables and equations in any kind of printed document accurately making it easier for a blind person to read documents without any hassle.

1.4.2 Specific goals

The specific goals for ITR component are as follows.

- Identify objects like animals, flowers, vehicles and background areas like sky, snow in a document image using available datasets and create a detailed description based on the identified data.
- Identify the table region and its contained data separately according to columns, rows and create a detailed description to explaining the tabular data to the user.
- To increase the accuracy of identifying each objects and data in both images and tables and reduce the time it takes to identify and analyze contents of printed documents.

1.4.3 Users

Reading Eye system will be developed for blind and low vision people in mind. Therefore, system features will be developed to make vision impaired person's reading experience easier and more comfortable.

Although it is mainly for vision impaired people, with its high quality features and functions it will be suitable for any person who prefers to read documents by hearing. Hence, any person

who wishes to read printed documents by listening can be considered to be the user of the proposed product.

1.4.4 Reading Suggestions

The flow of the document starts with Introduction, the first part of the SRS, which gives a brief knowledge about the entire software product. This includes the purpose of the document, scope, intended audience and finally the overview of the product which include the users, objectives and the reading suggestion as well. The Overall description of the document presents a complete idea about the product, which is listed as the second part of the document. Details about user interfaces, hardware interfaces, software interfaces and communication interfaces can be found under this section. The problem and the solution of identified for the domain are also discussed under this section along with product functions, constraints, user characteristics and assumptions made for this system.

The proposed system's non-functional requirements such as reliability, availability, security and maintainability quality attributes and the detailed description of external interfaces for developers' understanding are illustrated under chapter three, Specific requirements. This chapter includes other requirements which will be considered in future versions of the system.

All the sections in the document are numbered accordingly and framed into partitions of headings and contents for reading conventions. The document is written in Times New Roman font where headings are in bold and font size is set to 14, which gives higher priority and makes map reading easy. The content under each heading is in font size 12 with 1.5 line spacing and aligned justified. The headings under each section are inherited by detailed contents. The document contains diagrams and tables to provide more information with clarity. Additional diagrams can be found in appendixes which have been referred in the main content area.

2 Overall Descriptions

Vision impaired people use Braille codes when reading and writing. Finding materials which are written in Braille format is quite hard because majority of the books and other documents typically are not written in Braille. Even if one finds a book written in Braille format that will also only text-based content. The situation is even worse when it comes to mathematical and scientific books as they contain lot of charts, equations and tables. Designing a science, mathematics or engineering book is a tedious task because of this reason. Besides, those books are too expensive and take up lot of space. Since mathematics or scientific books are usually around 500-1000 pages or more, Braille format of the same book will get larger than that size. Using that kind of large book is not an easy and practical thing especially for blind user. Therefore, in such a situation, anyone will prefer to have a proper document reading system that acts promptly.

Due to the major development of modern technology, there are many existing document reading apps are available for visually impaired people. However, the majority of software applications developed to detect and analyze text contents but most of them cannot detect image-based contents and table-based content. Therefore, vision impaired people cannot use these applications for reading tasks.

The proposed system is our solution to the mentioned problem. As it can read any type of content in a printed document. Proposed system is divided based on the content type of the document.

The ITR sub system is responsible for reading both images and tabular data in a document. It will help the user to understand what an image is about and what kind of data a table consists of in a given document.

When implementing the ITR submodule, convolutional neural networks will be used. CNN will be trained with predefined datasets to detect objects in an image individually. This is known as instance segmentation. Area identification in an image is known as semantic segmentation. While instance segmentation identifies each object or a person individually, semantic segmentation will detect the areas like sky, snow covered places, grounds likewise. However, both features will be developed using the same type of neural network. That is Mask R-CNN. Mask R-CNN is efficient for detecting objects in a given image. It is a derivation of many R-CNN models that were built earlier to identify objects. When training a neural network

model, providing datasets is a must. Hence, popular image datasets like ImageNet will be used to train and test the models. CNN can learn when it is fed with data (dataset must be large) and then it can identify objects easily without any human intervention. It is like, it will work out on its own without our help. That is the beauty of the neural networks. There is no need to explicitly instruct the model the ways to identify objects.

The method that have to be used for table reading is not similar to the image reading. Because tables contain numbers and characters. In this feature, it is required to identify those numbers and characters in the table using Optical Character Recognition methods. Firstly, the table region and then its columns, rows should be identified. After that using OCR techniques we can read the data inside the table. For these tasks also a Mask R-CNN will be used as mentioned before because it is the most suitable CNN type for region identification. After detection of both images and tables, detailed descriptions will be created using Natural Language processing methods. NLP also based on neural networks. Therefore, CNNs will be trained to identify human languages and text generation tasks when implementing this module.

2.1 Product Perspective

Reading Eye is a complex app which reads various types of contents in a printed document. Most of the available solutions in the market targets only on reading text as big percentage of a document will be text. The proposed component is not only a solution for identifying images and other content types but also for filling the possible range of gap in the existing solutions in the market.

Below is the identified comparison of the proposed system features and the existing systems' features.

Existing Products / **Features**	Amazon Kindle	BARD Mobile	Capti Voice	KNFB Reader	Schmoozer	**Reading Eye**
Image based contents identification and reading	✘	✘	✘	✘	✓	✓

Table based contents identification and reading	✘	✘	✘	✘	✘	✓
File size reduction for effective communication	✘	✘	✘	✘	✘	✓
Audio narration	✓	✘	✓	✓	✓	✓

Table 3 – Comparison of the existing products with proposed system.

2.1.1 System Interfaces

Since this sub module has to detect objects in an image, we need to gather image datasets to train the models properly. To gather required datasets, special image databases like ImageNet will be used.

2.1.2 User Interfaces

The sub system is doing the server side processing in a cloud server. It is a process which does not require any kind of user intervention. Therefore, for ITR sub module user interfaces will not be required.

2.1.3 Hardware Interfaces

The Reading Eye product uses the smart mobile phone to present the final output to the user. Therefore, there is no need for the end-users to have special dedicated devices to run this component. Having a smart phone of Android or iOS platform with a camera will be sufficient to run this app successfully.

2.1.4 Software Interfaces

Following mentioned are the software tools that are used in implementation of this mobile application.

- React Native

  React Native apps may target iOS 9.0 and Android 4.1(API 16) or newer.
- Visual Studio Code
- Spyder IDE
- Anaconda python distribution
- TensorFlow
- Keras
- MS Azure

2.1.5 Communication Interfaces

The Internet is required to get connected to the cloud services. Therefore, mobile data or portable Wi-Fi should be used for this purpose. Usage of a high connection bandwidth may ease the usage of application without waiting.

2.1.6 Memory Constraints

Since the ITR component process the information in cloud beforehand, there is no need for the mobile phone to allocate much of memory for processing or storing. Approximately 2GB memory is required to run the app with the rest of the components.

2.1.7 Operations

- The proposed system will be delivered to its end-users in a form of an android/iOS mobile app. Therefore, the user should download the app from the store and install it in an Android/iOS mobile device.
- The user should always enable Wi-Fi or mobile data to make the internet connection continuously.
- The mobile phone should place clearly to capture the photo from the camera to get a clear photograph. However, the captured photo is not clearly identified it will notify to end-user through the voice because all the alerts and notifications will be notified through voice using a text to speech API.

2.1.8 Site Adaptation Requirements

- This application will be used by blind and low vision people. Hence the application should be simple, understandable and user friendly along with accessible features.
- Mobile phone should be able to provide a proper and constant internet connection.

2.2 Product Functions

ITR is a component of the proposed Reading Eye. The user can use the app in his/her mobile phone/tablet to read a printed document. The mobile camera will take clear photos of the document. Then the photos will be uploaded automatically to the cloud for analyzing purposes. If the uploaded photos of documents contained images, they will be analyzed by the model. After that, the model will detect objects, background areas in the image. Then system will form cohesive sentences for each object explaining what the object is and its special characteristics if there are any. Then a detail description will be formed using the generated sentences. As the final output, the system will generate an audio file of that description in order to play for the user to listen.

2.2.1 Diagram for the ITR component

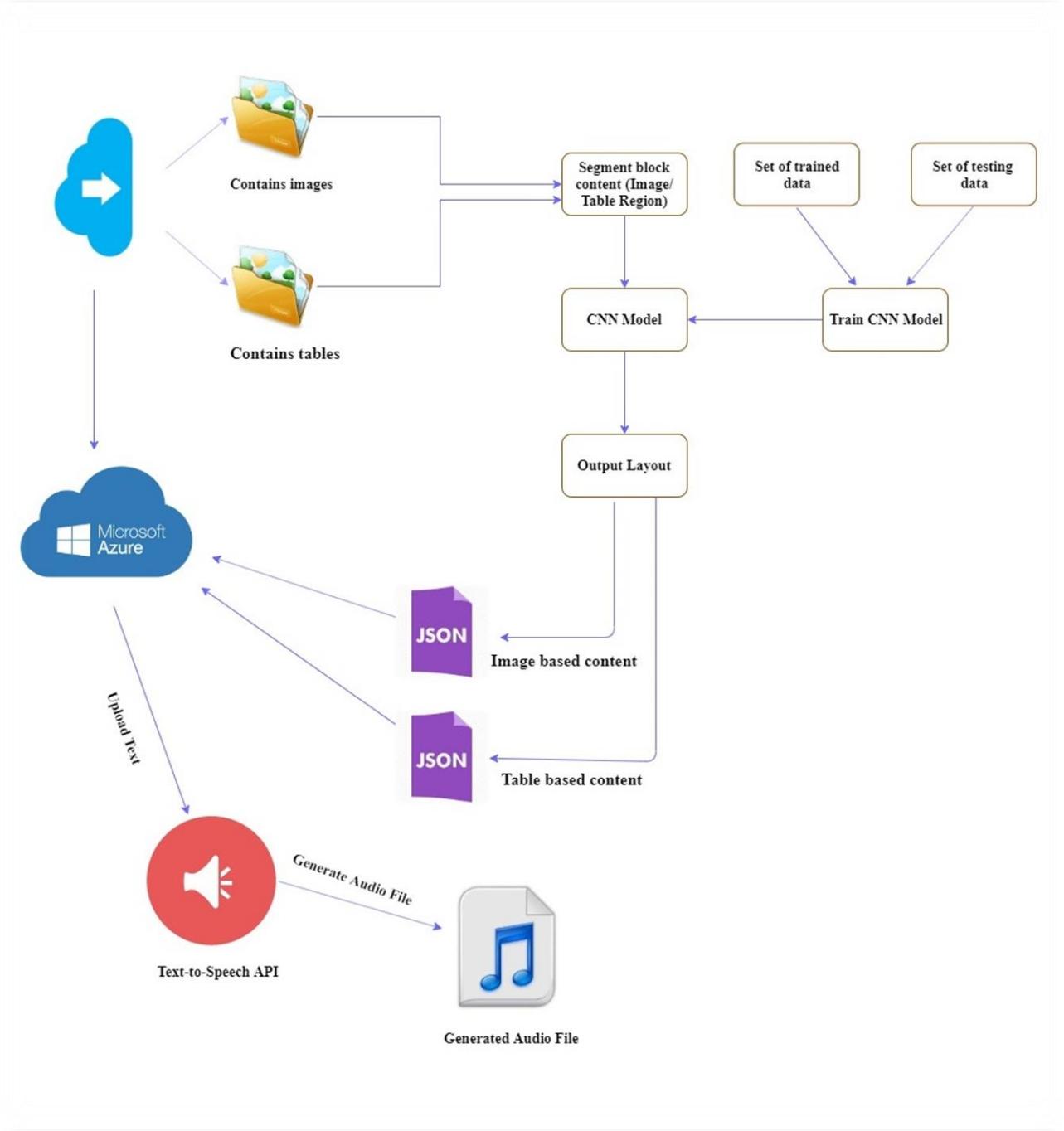

Figure 1 – High-level architecture diagram for ITR component.

2.2.2 Use case diagram

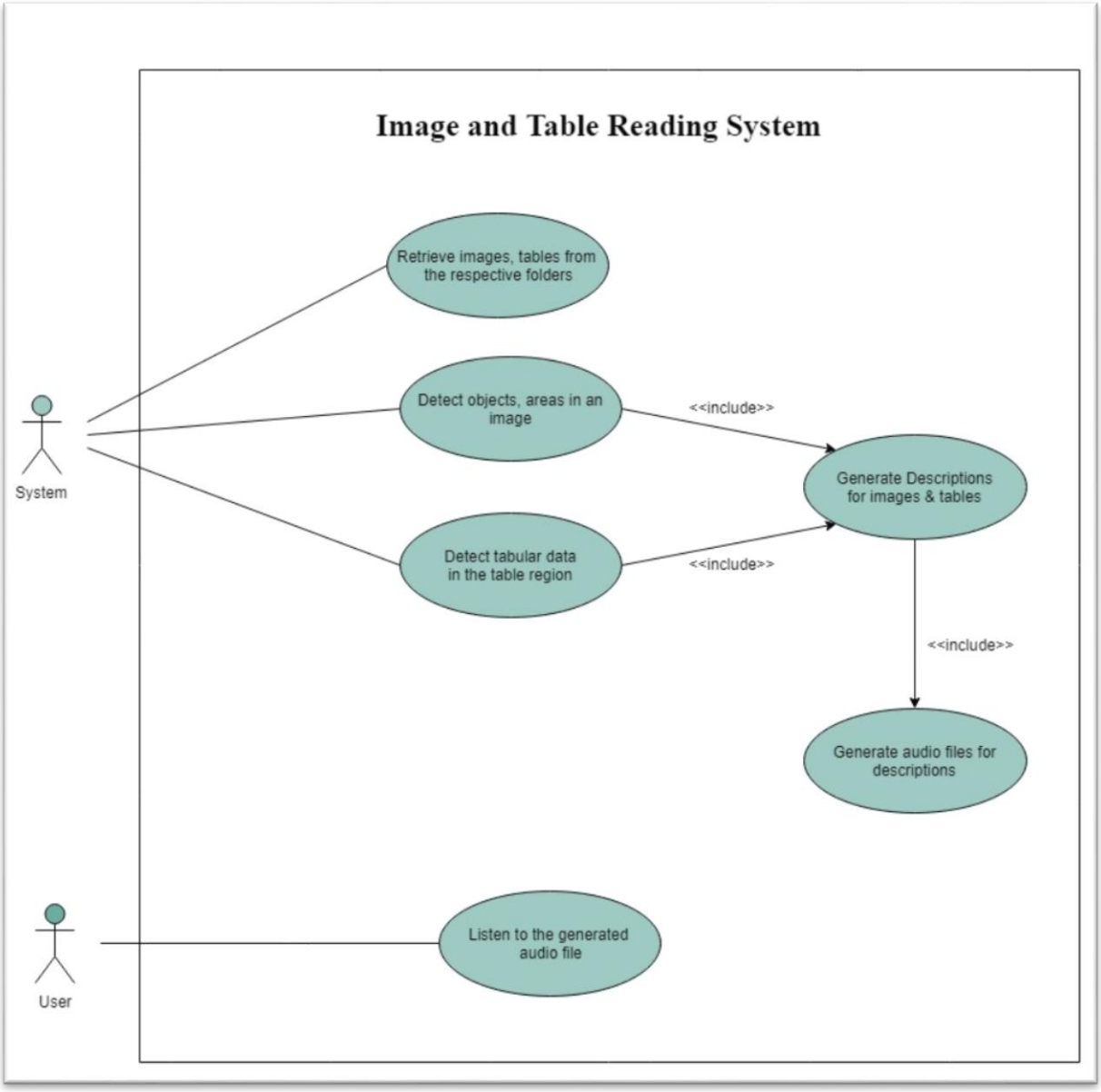

Figure 2 – Use case diagram of ITR component.

2.2.3 Use case scenarios

Use Case Element	**Description**
Use Case ID	ITR_UC1
Use Case Name	Retrieve images, tables from the respective
Use Case Description	System has to retrieve images, tables from the respective folders.
Primary Actor(s)	System
Main Flow	1. System checks the repository where files are saved. 2. System checks image and table folders. 3. System returns the relevant file for analyzing. 4. The use case ends.

Table 4 - Use Case Scenario of Retrieving files from folders.

Use Case Element	**Description**
Use Case ID	ITR_UC2
Use Case Name	Detect objects, areas from the image.
Use Case Description	System detects objects and areas in the image.
Primary Actor(s)	System
Main Flow	1. System analyzes the image. 2. System compares the image with datasets for object detection. 3. System detects objects (e.g. animals) and areas (e.g. sky, snow) separately.

	4. Then it saves the identified objects and areas temporarily as string. 5. The use case ends.
Post-conditions	Output will be saved as a .txt file for text generation.

Table 5 – Use Case of Analyzing Images.

Use Case Element	**Description**
Use Case ID	ITR_UC3
Use Case Name	Detect data from the table.
Use Case Description	System detects data from the table.
Primary Actor(s)	System
Main Flow	1. System analyzes the table. 2. System identifies the column length along with their headings. 3. System identifies the rows in the table separately. 4. Then it reads row data individually and according to the order. 5. Then saves the read data temporarily as string. 6. The use case ends.
Post-conditions	Output will be saved as a .txt file for text generation.

Table 6 – Use Case of Analyzing Tables.

Use Case Element	**Description**
Use Case ID	ITR_UC4
Use Case Name	Generate descriptions.

Use Case Description	System generates descriptions for images and tables.
Primary Actor(s)	System
Main Flow	1. System retrieves the temporarily text files that were saved earlier. 2. Using NLP techniques a description will be generated taking text files as input data. 3. A description files will be saved in JSON format. 4. The use case ends.

Table 7 – Use Case of Generating Descriptions.

Use Case Element	**Description**
Use Case ID	ITR_UC5
Use Case Name	Generate audio files.
Use Case Description	System generates audio files using descriptions.
Primary Actor(s)	System
Main Flow	1. System retrieves the JSON files that were saved earlier. 2. All the data from all the relevant JSON files will be concatenated. 3. Using MS Azure text-to-speech API, system will generate an audio file. 4. The use case ends.

Table 8 – Use Case of Generating Audio Files.

Use Case Element	**Description**
Use Case ID	ITR_UC6

Use Case Name	Listen to the audio files.
Use Case Description	User will listen to the audio narration.
Primary Actor(s)	User, System
Main Flow	1. The use case starts with taking photos of the document. 2. After processing system will return the generated audio file. 3. Then the user can listen to the audio narration. 4. The use case ends.

Table 9 – Use Case of User Listening to the Audio Narration.

2.3 User Characteristics

Reading Eye system will be developed for blind and low vision people in mind. System features will be developed to make vision impaired person's reading experience easier and more comfortable.

Although it is mainly for vision impaired people, with its high quality features and functions it will be suitable for any person who prefers to read documents by hearing. Hence, any person who wishes to read printed documents by listening can be considered to be the user of the proposed product.

Some essential pre-requisites to use this product are as follows:

a. A basic knowledge of using a smart mobile phone with the base of Android/iOS platform.
b. Having a sufficient knowledge of English language as all the instructions of the app will only be in English.

2.4 Constraints

- For this whole application, internet connection is a constraint, since the backend processing will be handled in a cloud server and fetching data from APIs using internet is also required.
- Instructions and displays will be only in English. Audio narration will also be only in English.
- Training a model will take time to produce an accurate result. Therefore, it will take some time for the system to achieve a minimum accurate result at the very beginning. So have to limit the number of objects to be detected at the beginning stage.
- Models need to be trained with more datasets in order to get accurate output. Therefore, the required time for training will be very high.

2.5 Assumptions and Dependencies

For the implementation purpose, we have to consider some assumptions.

- Users are capable of operating a mobile phone or an application at least in minimum level.
- Blind and low vision people will be interested in listening to an audio narration and will use the application for their readings.
- User's English literacy is good enough to use the app and to understand the document content.

  For dependencies, having a better network connection is a must. Wi-Fi connection will be suitable.

2.6 Apportioning of Requirements

Since the proposed product is an outcome of a research project, the path to reach the objectives might differ. The major components and their outcomes mentioned in this document will not change in future. Nevertheless, the methodologies and the technologies mentioned to achieve those outcomes might change in order to make the results more accurate, reliable and efficient.

The requirements mentioned in chapter 1 and 2 in this document are the primary requirements and is very unlikely to be changed in the future. The requirements mentioned in chapter 3 are desirable requirements which will be taken into consideration in the current version's latter stages of implementation, probably in upcoming versions of the proposed app.

3 Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

ITR sub module will not have user interfaces as the users will not directly interact with this component and for its processing tasks user input will not be required.

3.1.2 Hardware Interfaces

Data gathering of ITR component requires a camera to capture clear and quality photos for the analyzing purposes.

3.1.3 Software Interfaces

- React Native
  React Native apps may target iOS 9.0 and Android 4.1(API 16) or newer.
- Visual Studio Code
- Anaconda python distribution
- Spyder IDE
- TensorFlow
- Keras
- MS Azure

3.1.4 Communication Interfaces

The Internet is required to get connected to the cloud services. Therefore, mobile data or portable Wi-Fi should be used for this purpose. Usage of a high connection bandwidth may ease the usage of application without waiting.

3.2 Classes/Objects

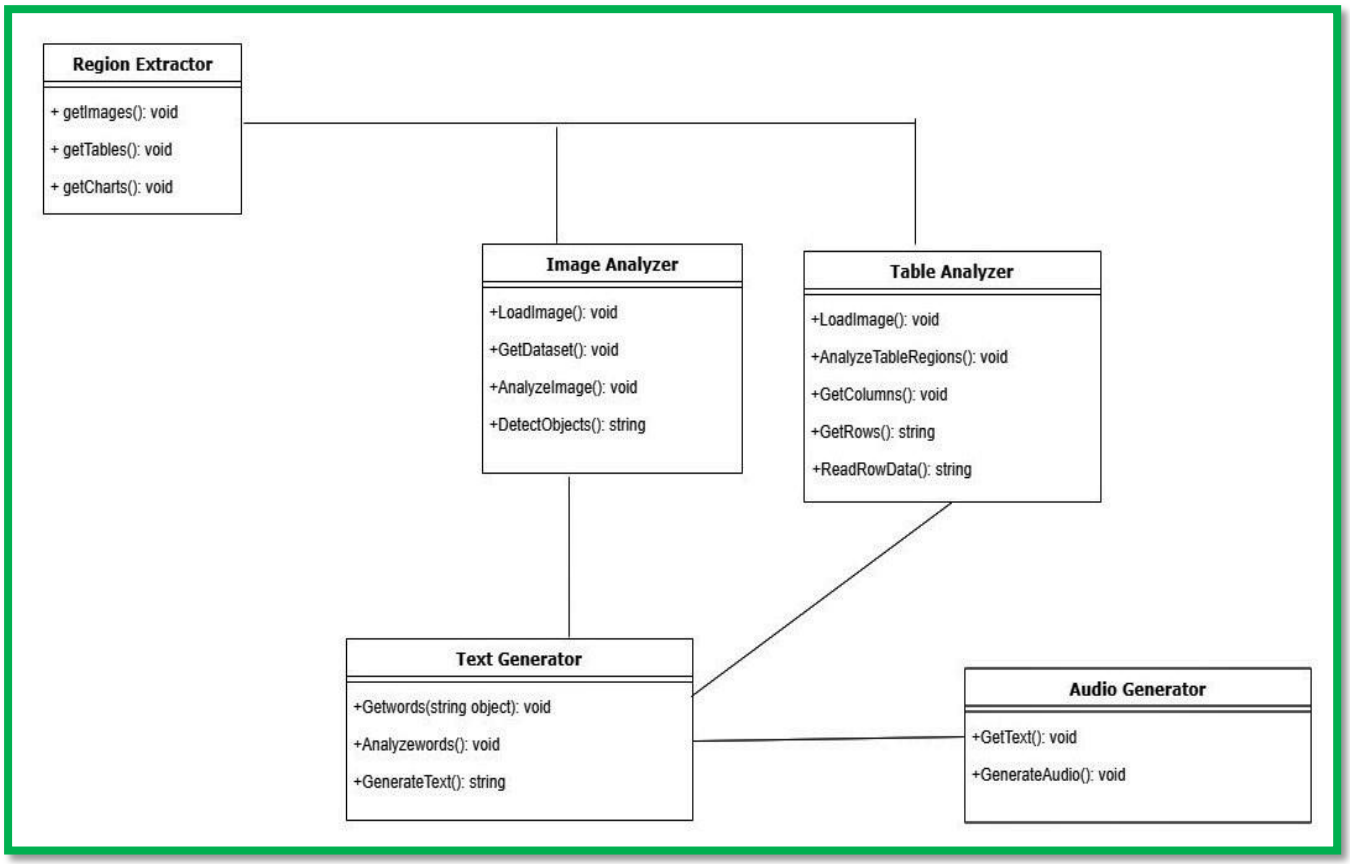

Figure 3 – Class diagram of ITR component.

3.3 Performance Requirements

Normally, the processing power of a mobile phone is lower than a desktop computer or even a laptop. Since, Reading Eye is implemented as a mobile application; it should be developed to achieve maximum efficiency with minimum resource utilization.

One of the main goals in performance vice is to implement this application to run without interruptions and efficiently. Also, user should have the facility gain quick and accurate outputs, which improves the reliability and the user experience of this application.

This application is supposed to reduce the time it takes a Braille user to read a document. Therefore, it should give accurate results in a short amount of time giving the user a nice reading experience.

3.4 Design Constraints

When developing a system, occasionally there will be limitations on the conditions or the requirements. These are called as design constraints. It's an essential factor to be identified during the designing the system, to obtain all the requirements that are given, since these constraints are the periphery of this process. Following mentioned are few design constraints that we have identified in this application.

- Time

  This research is done to obtain the learning outcomes of modules CDAP-I and CDAP-II under degree program conducted by SLIIT. Since, this is developed with the guidance of the mentioned institute; it should follow the project artefacts to meet the provided schedule. This makes time limitation major concernment in this project.
- Human Resource

  This project will be done by a group under a supervisor's guidance. Project is divided into sub modules so that all the members have a participation in the project. Sometimes, it makes difficult to gain one conclusion as there are different ideas from all members. Also, with the busy schedule everyone is following, it's limiting the chances of having appointments with the supervisors as well.

3.5 Software System Attributes

3.5.1 Reliability

The reliability of the proposed software product heavily depends on the accuracy level of the results on appropriate time. This is applicable for image/table reading component as well. In order to achieve maximum accuracy, the training model should be trained for enough number of datasets and should process the result in a short period of time. The accuracy of results is proportionate to the size of data set. In an image, the model should be able to differentiate one object from another. For example, model should be able to differentiate a dog from a bird. Providing result on appropriate time is equally important as providing an accurate result. No matter how accurate the result is if it is not delivered at appropriate time. Through these strategies, the reliability of the system can be achieved.

3.5.2 Availability

Availability means the probability that a system works in the way it should work when it's required to be operated during a given time period. To ensure that the application has availability, user should have a better internet connection with an android device.

3.5.3 Security

Since this application is mostly based on online, the security is a key attribute to be considered when developing the application. Therefore all the files that will be uploaded to the cloud will be compressed and encrypted using a special algorithm which will be developed as a separate module of this research. Using that algorithm we plan to enhance the security of the application.

3.5.4 Maintainability

Maintainability of the proposed software product is the success rate of the system's performance after a repair within a short period of time. It also incorporates the ease of adding new features, improving the performance of the system without having major changes the system's infrastructure. The system is designed accordingly by considering this in mind. This also prevents additional charges for maintainability.

3.6 Other Requirements

There are no other requirements identified or provided as far as the phase is considered.

4 Supporting Information

4.1 References

[1]"Braille Facts", Royalblind.org, 2019. [Online]. Available:

https://www.royalblind.org/national-braille-week/about-braille/braille-facts. [Accessed: 07- Aug- 2019].

[2] A. Gilani, S. Qasim, I. Malik and F. Shafait, "Table Detection using Deep Learning." [Accessed 10 August 2019].

[3] "Image Detection, Recognition, and Classification with Machine Learning", Azati.ai, 2019. [Online]. Available: https://azati.ai/image-detection-recognition-and-classification-with-machine-learning/. [Accessed: 11- Aug- 2019].

[4] "Accessible Mobile Apps | American Foundation for the Blind", Afb.org, 2019. [Online]. Available: https://www.afb.org/blindness-and-low-vision/using-technology/[Accessed: 20- Aug- 2019].

[5] "An In-depth Evaluation of the BARD Mobile App from the National Library Service for the Blind and Physically Handicapped | AccessWorld | American Foundation for the Blind", Afb.org. [Online]. Available: https://www.afb.org/aw/15/2/15607. [Accessed: 20- Aug- 2019].

[6] Play.google.com. [Online]. Available: https://play.google.com/store.