DEVELOPING MOBILE APPLICATION TO HELP DISABLED PEOPLE WITH MACULAR DEGENERATION

A THESIS SUBMITTED TO THE GRADUATE SCHOOL OF APPLIED SCIENCES OF NEAR EAST UNIVERSITY

By
RAFIA KHALLEEFAH HAMAD MOHAMMED

In Partial Fulfilment of the Requirements for the Degree of Master of Science in Computer Information Systems

NICOSIA, 2018.

RAFIA KHALLEEFAH HAMAD MOHAMMED
DEVELOPING MOBILE APPLICATION TO HELP DISABLED PEOPLE WITH MACULAR DEGENERATION TO SEE
2018
NEU

DEVELOPING MOBILE APPLICATION TO HELP DISABLED PEOPLE WITH MACULAR DEGENERATION

A THESIS SUBMITTED TO THE GRADUATE SCHOOL OF APPLIED SCIENCES OF NEAR EAST UNIVERSITY

By
RAFIA KHALLEEFAH HAMAD MOHAMMED

In Partial Fulfilment of the Requirements for the Degree of Master of Science in Computer Information Systems

NICOSIA, 2018.

Rafia Khalleefah Hamad MOHAMMED: DEVELOPING MOBILE APPLICATION TO HELP DISABLED PEOPLE WITH MACULAR DEGENERATION

Approval of Director of Graduate School of Applied Sciences

Prof. Dr. Nadire CAVUS

We certify this thesis is satisfactory for the award of the degree of Masters of Science in Computer Information Systems

Examining Committee in Charg

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name:RAFIA MOHAMMED

Signature:

Date:3/12/2018

ACKNOWLEDGEMENTS

This thesis would not have been possible without the help, support and patience of my principal supervisor, my deepest gratitude goes toProf.Dr. Dogan Ibrahim, for his constant encouragement and guidance. He has walked me through all the stages of the writing of my thesis. Without his consistent and illuminating instruction, this thesis could not have reached its present from.

I would like to thank Prof.Dr. Nadire Cavus who has been very helpful through the duration of my thesis.

Above all, my unlimited thanks and heartfelt love would be dedicated to my dearest family for their loyalty and their great confidence in me. I would like to thank my mothers for theirun-ending support, encouragement and constant love which sustained me throughout my educational endeavor. I would like to thank my wife for her personal support and great patience at all times. I would also like to thank my son for always making me feel at ease with his trouble whenever I am stressed up with work. I would also want to thank my brothers and sisters for always being there for me.

Finally, I would like to also thank my friends, for they have been supporting me to achieve my goals right from the beginning.

To my parents, wife and son…

ABSTRACT

There exist a lot of different disabilities which tend to make it difficult or almost impossible for the disabled victims to carry out some day to day activities. These disabilities might be in the form of no or reduced vision, hearing or physical. However, many research works have been carried out to make life easier and better for people with disabilities of all kinds. Today, mobile applications are widely used as accessibility tools for the disabled individuals, even though there are many other mechanisms that could be used as accessibility tools. The mobile applications could be regarded as more usable than other accessibility tools due to their portability and low cost.

This research study aims at maximizing the efficiency of information use in the digital environment for people with reduced vision, especially in the immediate educational and business community, and those with a high degree of work intensity by developing a mobile applicatin. The application is specifically aimed at transforming the information into a workable form, facilitating accessibility and removing language barrier so that the information can be understood and as well as enabling information sharing.The application developed in this study uses OCR technology to allow visually disabled individuals to have access to documents by taking snapshot using their phone camera. The text in the captured image is recognized and therefore could be converted to speech using TTS technology. It can further convert the text to different languages using the Google translate. Furthermore, the app present them with the opportunity to edit the recognized text, share the text on their social media accounts and even save the text in PDF format on the device. However, the app is developed only for Android devices.

Keywords: Optical character recognition (OCR); text-to-speech (TTS); accessibility tools; lower case; mobile application; visually impaired

ÖZET

Engelli mağdurların günden güne aktivitelerini gerçekleştirmelerini zorlaştıran veya neredeyse imkansız kılan pek çok farklı engel bulunmaktadır. Bu engeller, görme, işitme veya fiziksel duygularda azalmalar şeklinde olabilir. Bununla birlikte, her türlü engelli insanlar için hayatı kolaylaştırmak ve daha iyi hale getirmek için pek çok araştırma yapılmıştır. Günümüzde, mobil uygulamalar, erişilebilirlik araçları olarak kullanılabilecek başka mekanizmalar olsa bile, engelliler için erişilebilirlik araçları olarak yaygın şekilde kullanılmaktadır. Mobil uygulamalar, taşınabilirlik ve düşük maliyetleri nedeniyle diğer erişilebilirlik araçlarından daha kullanışlı olarak değerlendirilebilir.

Bu araştırma, iş dünyasında ve görmede zorlanan kişiler için geliştirilmiş olan bir mobil uygulamadır ve burada amaç bu insanların görme derecelerini artırarak hayatlarındaki verimi artırmaktır. Uygulama özellikle bilginin uygulanabilir bir forma dönüştürülmesi, erişilebilirliğin kolaylaştırılması ve dil engelinin kaldırılması, böylece bilginin anlaşılabilmesi ve bilgi paylaşımının mümkün kılınması amaçlanmıştır. Bu çalışmada geliştirilen uygulamada, görme engelli bireylerin telefon kameralarını kullanarak fotoğraf çekerek belgelere erişmelerine izin vermek için OCR teknolojisini kullanmaktadır. Çekilen görüntüdeki metin tanınır ve bu nedenle TTS teknolojisi kullanılarak konuşmaya dönüştürülür. Google çevirisini kullanılarak metin farklı dillere dönüştürebilir. Ayrıca, uygulama onlara tanınan metni düzenleme, sosyal medya hesaplarındaki metni paylaşma ve hatta metni cihazdaki PDF formatında kaydetme fırsatını da sunmaktadır. Ancak, uygulama sadece android cihazları için geliştirilmiştir.

***Anahtar Kelimeler*:** Optik karakter tanımı (OCR); teksden konuşma (TTS); ulaşabilme araçları; küçük harf; mobil uygulama; görme engelli

TABLE OF CONTENTS

ACKNOWLEDGEMENTS ii
ABSTRACT iv
ÖZET v
TABLE OF CONTENTS vi
LIST OF TABLES ix
LIST OF FIGURES x
LIST OF ABBREVIATIONS xi

CHAPTER ONE: INTRODUCTION 1
1.1Problem Statement 2
1.2Aim of the Study 2
1.3Importance of the Study 2
1.4Limitations of the Study 3
1.5Overview of the Thesis 4

CHAPTER TWO: RELATED RESEARCH 5

CHAPTER THREE: THEORETICAL FRAMEWORK 13
3.1Optical Character Recognition (OCR) Technology 13
3.1.1OCR technology structure 14
3.2Text-to-Speech (TTS) Technology 17
3.2.1TTS technology structure 18
3.3Cloud Systems and Data Management 19
3.3.1Cloud systems and data management structure 21
3.3.2Advantages and disadvantages of cloud processing and storage systems 23
3.4Mobile Applications Development 23
3.4.1Types of mobile applications 24
3.4.2Operating systems of mobile devices 24
3.4.2.1Android OS 25
3.4.2.1.1Advantages and disadvantages of android OS 25
3.4.2.2IOS operating system 26
3.4.2.2.1Advantages and disadvantages of IOS 26

3.4.2.3Windows IOS 27
3.5Summary 27

CHAPTER FOUR: SYSTEM ANALYSIS AND DESIGN........ 29
4.1System Architecture 29
4.2System Description 30
4.3System Technologies........ 31
4.3.1Mobile applications technology........ 31
4.3.2Server-client applications and cloud computing 31
4.3.3Programming language technology 31
4.4Application Features 32
4.5Unified Modeling Language (UML) and Use-cases 32
4.5.1OCR operation actions........ 32
4.5.2TTS operation actions........ 33
4.5.3Cloud data backup operation actions........ 34
4.5.4Data sharing operation actions 34
4.6System Development Methodology 35
4.7User Interface (UI) Design........ 35
4.7.1Design arguments 36
4.7.2Principle of user interface design 37
4.8Summary 37

CHAPTER FIVE: SYSTEM IMPLEMENTATION 38
5.1Introduction 38
5.2Testing........ 48
5.2.1Testing on ordinary paper........ 48
5.2.2Testing on newspaper 48
5.2.3Testing the TTS conversion........ 49
5.2.4Testing the translation feature 49
5.2.5Testing the PDF generation 49

CHAPTER SIX: CONCLUSION AND RECOMMENDATIONS 50
6.1Conclusion........ 50
6.2Recommendations 50

REFERENCES 51

APPENDICES 56
Appendix 1: OCR Implementation 57
Appendix 2: TTS and Google Translate Implementation 61

LIST OF TABLES

Table 2. 1: Summary of Related Research.. 10
Table 3. 1: Tesseract Performance Comparison Results.. 16

LIST OF FIGURES

Figure 3. 1: Tesseract Process Flow 17
Figure 3. 2: Text To Speech Process 19
Figure 3. 3: Google TTS Restfull Web Service Architecture Diagram. 19
Figure 3. 4: Cloud System Infrastructure.. 21
Figure 3. 5: Cloud System Functionalities.. 22
Figure 4. 1: System Architecture...…………….....…………………………….........30
Figure 4. 2: OCR Processing Diagram 33
Figure 4. 3: TTS Processing Diagram 34
Figure 4. 4: Cloud Processing Diagram.. 34
Figure 5. 1: Home Screen. 38
Figure 5. 2: Release Note.. 39
Figure 5. 3: About/Contact Information of the Creator. 39
Figure 5. 4: Manage Languages.. 40
Figure 5. 5: Sample Snapshot. 40
Figure 5. 6: Warning Message. 41
Figure 5. 7: Crop Snapshot. 41
Figure 5. 8: Column and Language selection. 42
Figure 5. 9: Character Recognition Process.. 42
Figure 5. 10: Output of Recognition Process.. 43
Figure 5. 11: Text Settings.. 43
Figure 5. 12: Table of content.. 44
Figure 5. 13: Edit, Copy and Paste Options.. 44
Figure 5. 14: Document Title.. 45
Figure 5. 15: Output Translated to Arabic. 45
Figure 5. 16: Output Translated to Turkish. 46
Figure 5. 17: Generated PDF file. 46
Figure 5. 18: Share Options. 47
Figure 5. 19: Accessing files from device. 47
Figure 5. 20: Accessing files from Google drive. 48

LIST OF ABBREVIATIONS

OCR: Optical Character Recognition

API: Application Programming Interface

TTS: Text to Speech

OS: Operating System

HMD: Head Mounted Display

CCTV: Closed Circuit Television

SNS: Social Networking Service

AR: Augmented Reality

GPS: Global Positioning System

HTTP: Hypertext Transfer Protocol

SOAP: Simple Object Access Protocol

XML: Extensible Markup Language

ERP: Enterprise Resource Planning

CRM: Customer Relationship Management

IaaS: Infrastructure as a Service

SaaS: Software as a Service

PaaS: Platform as a Service

MNO: Mobile Network Operator

SDK: Software Development Kit

IOS: Internet Operating System

ADT: Abstract Data Type

HTML: Hypertext Markup Language

PC: Personal Computer

CSS: Cascading Style Sheet

PDA: Personal Digital Assistant

APK: Android Package Kit

NFC: Near Field Connection

IDE: Integrated Development Environment

UML: Unified Modeling Language

RAD: Rapid Application Development

UI: User Interface

ARCS: Attention, Relevance, Confidence and Satisfaction

UID: User Interface Design

CHAPTER ONE

INTRODUCTION

Nowadays, it is of great importance that information can be transformed into workable, analyzed format and consequently stored in safe environments. The fact that many written data are in an irreducible format, and the fact that they are stored on hard platforms that are difficult to process are greatly reducing the usability of this information and its speed of use. It is important for us to know that informationis to be stored on platforms that can be accessed quickly in many environments.It has become very difficult for written sources to be processed and analyzed for people who have lost their sight. Taking too much time to process long texts causes more time loss in learning or education and business.

Currently, mobile applications are becoming widely accepted in the learning process in various areas. But unfortunately most of the applications today do not put individuals with visual impairement into consideration(Jaramillo-Alca and Luja N-Mora, 2017). However, a substantial amount of effort has been exerted on the development of different infrastructures worldwide in other to assist visually impaired individuals to interact with their surroundings. It is undoubtedly hard and expensive to enhance every infrastructure at their disposal with a projection that there are 285 million visually impaired persons in the world, of which 39 million are blind(Jiang et al., 2017).

Optical and digital magnifiers as well as assistive software like screen magnifiers and contrast enhancement largely support people with low vision to access information on digital devices in their environments. Theses technological advancements employ vision enhancement techniques to help people to view detailed information. Conversely, the vision aid technologies support low vision individuals to see details, but they are not designed to support them with some other visual activities(Szpiro et al., 2016).

Thanks to the solutions provided by the Optical Character Recognition (OCR) technology, which makes it possible to transform textual data in unprocessed form into workable form. The cloud technology is another important aspect worth considering which makes it possible to convert these manipulable data into sound data and store it in the cloud environment. With high-speed voice conversion services and storage cloud services, it is aimed to reduce the time of usage ratio of user groups who make high use of text resources in the educational and business sectors.

1.1 Problem Statement

Today, numerous solutions to visual impairement have been presented by many researchers; ranging from traditional glasses, head mounted devices, finger-based readers, mobile applications and so on. Due to the rise in the possession of mobile devices, the use of mobile applications for supporting visually impaired persons become very important and worth studying. Moreover, most of the existing applications for helping visually impaired individuals possess a lot of vital features like magnification of text to enable them read, identification of signs around them to allow for easy navigation within their environments, and even reading text out for them to make life easier for them. However, these assistive tools are yet to provide all that is required to help the visually impaired because, majority of them only support reading. Considering mobile applications, the missing features in the existing applications includes offline access, ability to translate recognized text to different languages, storing vital information in trusted and secure environments, sharing of information pieces on social media networks when the need arises, generating files in portable formats, recognizing hand written text and many other features that will allow people with visual impairement not only to read but to engage in various activities.

The MyReader application developed in this study helps to offer some of the features to cover the knowledge gap. It is going to offer offline access, translation; both online and offline, generation of pdf files and sharing of information in text format on social networks.

1.2 Aim of the Study

This study is aimed at maximizing the efficiency of information use in the digital environment for people with a reduced vision, especially in the immediate educational and business community, and those with a high degree of work intensity by developing a mobile applicatin- MyReader. The application is specifically aimed at transforming the information into a workable form, facilitating accessibility and removing language barrier so that the information can be understood and as well as enabling information sharing.

1.3 Importance of the Study

One of the prominent importance of this study is developing the MyReader application which has the ability to process data without a connection to the internet and making it possible to increase the usability of the information in many environments. In addition to this, the conversion of the written data to the loudspeaker in accordance with the linguistic option of the user makes it possible to increase the performance of the information usage to a great extent. In a simpler manner, this study is regarded very important because it presents an application that can enable individuals with visual impairement to perform some tasks that could be very difficult without such a useful tool.

1.4 Limitations of the Study

Functional and infrastructural limits in the study;

1. The software can only run on the Android OS platform.
2. OCR technology does not have handwriting capture capability. Because OCR technology only supports font characters created in the digital environment, human-readable data can not be translated into correct form.
3. The OCR process requires that the source data display be vertical
4. Maximum data source can be 32767x32767 pixels in size.
5. The conclusion of operational functions on the mobile platform is that software performance depends on mobile device performance.
6. The cloud data storage medium is dependent on the data multiplication capacity of the Google Drive environment.
7. The operation of converting audio data after OCR operation depends on the performance of Google Translate services.

The clarity of the image data, the pixel depth depends on the resolution quality of the camera of the user’s mobile device.

1.5 Overview of the Thesis

This section briefly explains the components of the thesis report.

- **Chapter one:** introduce the study and present the problem statement as well as the aim and objectives and the importance of the study in a comprehensive manner. The chapter also presents the obstacles and difficulties that face the work in this study.
- **Chapter two:** presents related research to this study and also tries to compare between the reviewed works and this study with respect to the kind of features offered by the subjects.
- **Chapter three:** presents the theoritical framework of the study including the concept of Optical Character Recognition (OCR), Text-to-Speech, Google Translate, cloud computing, mobile application and mobile cloud computing.
- **Chapter four:** presents the methodology used in implementing the system.
- Chapter five: holds the result of the study including screenshots of the various features of the mobile application and how they work.
- **Chapter six:** presents the discussion, conclusion and future recommendation for the study.

CHAPTER TWO

RELATED RESEARCH

This chapter presents a literature review in the field of mobile application development as well as other assistive tools for visually impaired individuals. The research review in this chapter is presented with regards to the year of publication from the oldest to the most recent. At the end of the review, a summary table is presented to make clearer and easier the understanding of what each research work presents and also try to compare the features of the proposed work to that of the reviewed works.

Haddad, Chen and Krahe(2016) in their research work presented a proposed method that could provide a fast and simple solution to the issue of visual impairment by offering a tool attempts to automatically find the primary information that is portrayed by an image and then communicate it to a visually impaired persons. They reported that it takes the system little time to find a relief image which significantly simplifies the task of the creator and that their method makes it possible for any person in the same environment with a blind person to easily create a relief image for him or her. With regards to the recent improvements in scientific research in the fields of pattern recognition and image processing, they put forward a solution to the problem through text detection, recognition and transcription in Braille and then segmenting the different image areas and textures affiliation. They carried out an experimental study with eight blind people and eight pedagogical images to see whether blind people can understand the content. The different relief images were presented to the participants, and the participants were allowed enough time to be experiment and understand the content of the image in relief and then given a questionnaire to fill afterwards. However, they suggested that the work could be lengthened to put forward a tool for scanned or downloaded numeric and web graphics. They also suggested that a relief image system that is tablet compatible having voice synthesizer or a feedback-electro-vibration touch screen.

Sandnes (2016) reported in their study that the recent development in affordable wearable devices creates new prospects for ground-breaking visual aids. They aim to identify the functionalities needed by visually impaired individuals in different contexts to reduce barriers. A semi-structured guide was employed to gather information from three visually impaired academic individuals. Their research shows that the main challenge or problem for low-vision individuals is recognizing peoples’ faces. The second most significant challenge for them is recognizing text on buildings or structures and moving vehicles. An interesting finding by the interview was revealed questioning the use of smart glasses. They however, suggested that future studies should focus on development systems for facial and text recognition and how to test them in different contexts.

Stearns et al. (2016) carried out a controlled laboratory study consisting of 19 blind individuals to deeply measure the efficiency of finger-based sensing and feedback used for reading printed text. They made a comparison on an iPad-based test bed between audio and haptic directional finger guidance. To complement their study, they requested four of the participants to give feedback on a prototype called HandSight. Their findings shows that the performance between haptic and audio directional guidance is equal despite the fact that audio may have an advantage of accuracy for tracing lines of texts. The ease of use and the level of required concentration was questioned even though many participants valued the direct access to information delivered by the finger-based study. Moreover, they suggested that future study on finger-based reading should try to examine the possibility of putting text-heavy materials capability into consideration for the benefit of users with low vision in the case of finger-based readers.

Szpiro et al. (2016) in their study carried out a contextual inquiry in the form of an interview over the phone to find out whether the participants were actually low vision by asking if they are using or have used aids that enhanced their vision. They examined 11 low vision individuals with their mobile phones, tablets, and computers by carrying out some tasks like reading an email. Their research shows that many individuals preferred visual access to information than screen readers and that the tools could not provide with the right and sufficient assistance. They also found that for a participant to view a content comfortably, they have to perform multiple gestures. The challenges found made the individuals unproductive. Other challenges revealed were that low vision software utilities were difficult to use and the participants mostly did not use some tools because they find it difficult to disclose their disability.

Torres-Carazo, Rodriguez-Fortiz and Hurtado (2016) in their research work examined 94 applications that were precisely developed for visually impaired individuals. They tried to analyse if the applications could be considered as serious games and at the same time suitable for use by the visually impaired persons based on their characteristics. They however reported that the objective of their study is to improve the perceived inappropriate classification of such applications, thereby also improving there searchability. They added that this will help them deeply in making recommendations to individuals with visual impairment.

Voykinska et al. (2016) carried out a research with the use of Social Networking Services (SNSs) to discover enthusiasm, difficulties, activities and familiarity of people with visual impairment with regards to the visual content. 11 people participated in an interview and 60 people participated in a survey carried out by the researchers. The selected sample included individuals with little to no vision. It was found that the blind individuals faced accessibility difficulties. To efficiently access the SNS features, they came up with variety of strategies which later failed. Then, they turned to asking for help from trusted individuals or simply shunned some features. Their study claim to create better understanding of the usage of SNS by blind persons. However, the perception of trust when there is need for interaction partners was raised. Finally, the researchers suggested that the designers of SNSs should consider designs that will bring advancement in social networking for users; be them able or disabled.

Zhao et al. (2016) presented an augmented reality application called CueSee which runs on a head-mounted display (HMD) that could help make product search. The system uses visual cues to draw the mind of the user to a product after automatically recognizing it. They designed five visual cues within the application. To evaluate the visual cues, they engaged 12 participants with visual impairment. To find out whether the participant fits their study, they conducted a screening over the phone in the form of an interview. Volunteers who have used assistive tools like magnifiers or CCTVs were chosen as fit for the study over those who only made use of screen readers. They reported that the individual volunteers were found to have different vision conditions. Moreover, their study revealed that the participants chose CueSee over regular assistive tools for product searching in stores. They also found that their application performs even better than the corrected visions of the participants in terms of efficiency and correctness. They suggested that in the future they will consider designing a more suitable interaction method for the application users to target products and also generating the best visual cues for different groups of users. They finally propose to conduct the evaluation of the application in real sense, for example as a grocery shop to see how feasible CueSee is.

Gonnot, Mikuta and Saniie (2017) presented a study in which they came up with an algorithm that could be used in helping people with impaired vision to be able to recognize their surrounding environment through the manipulation of captured pictures from a camera into controlled frequency composed together to single melody which is played back to the user. The images could be from the camera of a smartphone or implanted into eyeglasses. They reported that the system might be uneasy for a normal user to comprehend all information presented through this approach, but that the positive thing is that training the users with impaired vision will make it easy for them to interpret the data. They further argue that developing a device for people with visual problem’s objective is to make it as less complicated as possible, and so their proposed algorithm was made extremely simple which could possibly be run on small devices. Their algorithm was executed and tested with some images in MATLAB. A spectrum analyser called Spectrum Lab was fed with the audio which demonstrates a waterfall illustration of the audio. Moreover, the initial results shows that images with adequate resolution could be transformed to identify shapes, traffic signs or deepness for collision prevention. They suggested that in future, the algorithm should be optimized and its deployment on mobile platform should also be looked into. They also added that the algorithm could be executed on a hardware directly.

Jaramillo-Alca and Luja N-Mora (2017) reported that inability of usage of serious games affects people with disabilities from having access to knowledge on an equal grounds with those without disabilities. They however carried out a study with the aim of supporting people with visual impairment who have difficulties in accessing video games due to their condition, more specifically serious games. Their work mainly presented a collation and exploration of guidelines for accessibility with regards to video games development for the need of persons that are visually impaired. Putting in consideration, the approach for their study, they chose to use the Serious Games CEOE which happens to be the only mobile application that falls to the educational category. They downloaded the app from the Google play store. They reported that it includes five different serious games for promoting daily healthy life. However, they suggested that the experiment could be carried out with people suffering from visual impairment so as to measure the efficiency of the features of the serious games pointed out for this study. Additionally, they suggested putting into consideration people with different disabilities from visual impairment.

Jiang et al. (2017) reported that the advancement in new technologies has boosted the invention of systems with the intention of providing information for people with visual impairment regarding their immediate environment. They carried out a project by developing an application with the use of current technologies like the Optical Character Recognition

(OCR) and Text-to-Speech (TTS) for the Android platform. These technologies are employed to detect and identify signs and texts within the surrounding environment of a visually impaired person and help guide them to navigate. They reported that the system works with computer vision and internet connectivity to also restructure sentences and then change them to sound. The system uses a smartphone camera to find the various sources of information in the environment and then inform the user about their location using Text-to-Speech techniques. OCR is also used by the system to read about variety of sources and relate their content to the visually impaired person. To carry out a usability test of their system, the application was used on an android device to take pictures, and then carried out an OCR and sign detection. The text recognized by the application is shown over the image, and when the sign is touched on the screen, it reads out the text to the user. They concluded that their experiment shows that the concept of the system is feasible on Android smartphones. They added by suggesting that it could be extended in the future to have real-time implementation instead of the still images.

Pundlik et al. (2017) postulated that viewport control using head motion can be natural and assist in having access to magnified displays. They employed Google Glass to execute the idea which shows the magnified screenshots that are received via Bluetooth in real time. Users can see different screen locations by moving their head and greatly interacting with the smartphone, rather than using touch gestures on the magnified mobile phone display to navigate. Two different applications forms the screen share application; a host application on the mobile phone and a client application on the Google Glass. To carry out an evaluation of their approach, 8 normally sighted and 4 visually impaired participants were assigned tasks using a calculator and music player applications. The result of their evaluation shows that the Glass is more efficient than the phone's screen zoom in the calculation task given. The performance measurement was carried out based on the time to complete the task. However, they suggested that in the future the implementation could allow for more gestures on the Glass for better interaction with the mobile device. And also that, the navigation based on head motion should be compared with other generally used vocal based mobile convenience features.

Table 2. 1: Summary of Related Research

Author	Device/App Used	Technologies	Description	Evaluation approach
Proposed Application	Android app developed	Optical Character Recognition, Text-to-Speech and Google Translate	- Uses camera to capture. - It can also use images from local folder. - Allow for magnification (zooming). - Allow sharing on social networks. - It can save file as PDF. - Allow for offline access	Interview and Questionnaire
Haddad, Chen and Krahe (2016)	Relief images in Braille	Pattern recognition and image processing	- They put forward a solution to the problem through text detection, recognition and transcription in Braille. - They segment the different image areas and textures affiliation.	Experimental study with 8 blind people and 8 pedagogical images
Sandnes (2016)	___	Text recognition and wearable visual devices	- They tried to identify the functionalities needed by visually impaired individuals in different contexts to reduce barriers	Interview was conducted to gather information from three visually impaired academic individuals
Stearns et al. (2016)	Finger-based sensing app-HandSight	Text recognition	- They made a comparison between audio and haptic directional finger guidance	Controlled laboratory study consisting of 19 blind individuals

Szpiro et al. (2016)	Phones, tablets and computers	Visual tools	- They try to find out how people with low vision access computing devices.	Interview and Questionnaire
Torres-Carazo, Rodriguez-Fortiz and Hurtado (2016)	Mobile applications for visually impaired.	Serious games	- They tried to analyse applications developed for visually impaired persons to see if they could be considered as serious games. - They tried to see if the applications are suitable for use by the users	Examined 94 applications that were developed for visually impaired individuals
Voykinska et al. (2016)	Mobile phones and computers	Social Networking Services (SNSs)	- They tried to discover enthusiasm, difficulties, activities and familiarity of people with visual impairment with regards to the SNS visual content	Interview and Survey
Zhao et al. (2016)	AR app called CueSee was developed	Augmented Reality (AR) and head-mounted display (HMD)	- The app is for searching products. - The system uses visual cues to draw the mind of the user to a product after automatically recognizing it.	Interview was conducted for volunteer selection and 12 participants were engaged in the testing
Gonnot, Mikuta and Saniie (2017)	Algorithm was developed	Mobile application, Head mounted devices and simulation	- They implemented an algorithm that could be used in helping people with impaired vision to be able to recognize their environment.	MATLAB and Spectrum Lab were used for the testing
Jaramillo-Alca and Luja N-Mora (2017)	CEOE- serious games was adopted	Mobile video games and serious games	- They aim to support people with visual impairment who have difficulties in accessing video games.	collation and exploration of guidelines for accessibility with

			• Also, presented a collation of guidelines for accessibility with regards to video games development for the visually impaired	regards to video games development
Jiang et al. (2017)	Android app was developed	OCR, TTS, GPS and computer vision	• Application is developed to detect and identify signs and texts within the surrounding environment of a visually impaired person and help guide them to navigate.	The application was used on an android device to take pictures, and then carried out an OCR and sign detection
Pundlik et al. (2017)	Mobile app and Google glass	A host application on the mobile phone and a client application on the Google Glass.	• They used Google Glass to show the magnified screenshots that are received via Bluetooth in real time. • Users can see different screen locations by moving their head. • Users can also interacting with the smartphone to navigate.	8 normally sighted and 4 visually impaired participants were assigned tasks

CHAPTER THREE

THEORETICAL FRAMEWORK

This chapter presents the basic concepts of data transformation systems from image-like data sources to sound-like data and the overall structure of the data conversion systems. In addition it presents some of the basic concepts of cloud computing expressing it's characteristics, cloud computing service layers and mobile cloud computing architecture and it's advantages. The chapter also presents a clear description of the concept of mobile application development and mobile devices.

3.1 Optical Character Recognition (OCR) Technology

Optical character recognition systems, which are developed on the basis of interpretation of digital characters which are one step ahead of the conventional optical technology in the optical environment, enable character-based data to be transmitted digitally and to enable people with such disorders to more easily perceive the processed data in this environment. With this improvement, character-based data, which is difficult to read, could be processed and translated into more readable form.

OCR technology can benefit a lot of individuals wih visual impairement by transforming texts and signs that are difficult to be seen and understood by them into clearer and readable state. The OCR technology in most cases could be supported by TTS technology to give a better solution to the problems of the visually impaired people. The TTS technology would be discussed in detail in this chapter. Some researchers believed that people who have lost their sense of sight have almost zero use of written sources in the digital environment and in order to solve this problem, OCR technology is in the first place (Johnson et al., 2010). It provides real-time solutions by facilitating the education of people who have visual disabilities. (Wong M et al., 2012). It is seen as a great advantage that the books can be moved to the digital medium and visually understood.

The implementation of OCR technology driven applications methodology in libraries where digital data is combined will increase both training performance and productivity. (Hakim et al., 2017). In everyday life, OCR, which is the solution of environmental perception problems, enables complex data to be understood. This technology can also be used in order to better analyze and identify the people's perception of their lost sight. (Guo et al., 2018).

3.1.1 OCR Technology Structure

Optical character recognition systems take image data as input and process the image data to recognize the characters in the image and output character set data. The first step in this process is to convert the image data to a greyscale image. The reason for the gray scale conversion is that the shapes in the picture can be analyzed more clearly and the data can be categorized. The action after this step can be defined as the separation of the part to be analyzed in the obtained data. This process is performed by cutting out the black-and-white tones from the specific coordinates which will increase the analysis performance and efficiency (Mennillo et al., 2015). The data that is separated from the image pixel coordinates surrounding the car- keret cluster continues with this pixel-analysis operation after this phase. Here, the word classification data is analyzed through the data classification algorithms and then the character data is output (Smith et al., 2007).

The OCR process can be performed in 6 steps. These steps are as follows;

Image Scanning: Image scanning can be done in many different ways. Some of them are using a scanner to convert the written document into digital picture format or other digital formats and transmit it to the digital medium. Another way can be done with today's technology and mobile cameras with high resolution ratios. The MyReader mobile application allows users to digitally print using their mobile camera without using a scanner. In addition, the pictorial data source supports data source entry from cloud data storage systems via local disk. In this platform design, where the user is not restricted from the source data, many data sources can be used for OCR operation. This feature is a great advantage for users. Since the user can have low data storage capacity in the local disk environment, cloud-supported data storage media has been added as a data source. One of the most important elements in pictorial data entry is the high resolution of the source data, the pollution intensities on the side, the density of the character data in the data, and the position angle of the character data in the pictorial data.

Image Resolution: Increasing the resolution of the image makes it possible to analyze the character data in the image data more clearly. In this respect, it is important that the image data is obtained with a quality device. With the increase in image resolution, the number of pixels to be analyzed is increased, which will increase the time spent in the process. But as a result, the quality of the image data is important so that the character data can be transferred correctly. The high image resolution due to the focus of providing the correct data will have positive results.

Image Noise: A lot of the pixel data in the image data is filtered during the OCR process. The increased data contamination directly affects the performance of the OCR process directly in a negative way. Filtering is important for pattern scanning performance. Failure to clear the contamination results in an incorrect information being generated as a result of the pattern scan. As a result, the OCR operation will produce data with incorrect characters. For this reason, data sources with data pollution must be cleaned prior to character recognition using various filtering techniques during the cleaning process. (Mennillo et al., 2015).

Image Binarization: At this stage, the digital image is transferred to the binary data type and prepared for the analysis phase. Data pollution contrast ratio plays an important role in image data during this process. The thresholding method and dynamic window methods used in the Tesseract OCR infrastructure ensure that this process is completed. The success rate of the procedure was approximately 85.1% (Patel et al., 2012).

Connected Component Analysis: At this stage, various image formats on the image data are determined. This is an expensive operation in terms of calculation time. The image data in black and white format is stored in the blob data through rounding. Different words will be divided at this stage according to the character range within the textual. Proportional text is revealed by determining certain gaps (Smith et al., 2007).

Finding text Lines And Words: At this stage, the alignment of the resultant words applied in the previous step on the image data is calculated. The presence of data alignment along with other words will be facilitated. After this step a two-stage filtering process will be applied to find the words.

Recognizing Words Phase 1 And Phase 2: Each phase of image processing data words are extracted words. In Phase 2, the entire page is scanned again. The data obtained after the entire page scan provides the presence of words not found after the first scan. The words in the whole image data can be found on this page. After completion of the process, all the words on the image data will be found and the OCR operation will be completed.

This study employed the Tesseract OCR and other Frameworks to achieve its aim. However, the Tesseract will be compared to other OCR in this section. At the beginning as discussed in the optical character recognition systems, it is ensured that the data can be correctly examined and interpreted. It is very important to perform these operational procedures with the correct methodology. One of the operations performed prior to the data review phase is to use a technique known as grayscale or image binarization of the image-based data. Removing the background data from the backplane after this operation will significantly affect performance. These steps are performed sequentially in the structure we used. In terms of performance comparison.

Table 3. 1: Tesseract Performance Comparison Results (Mennillo et al. 2015)

Performance metric	HANWANG OCR		ABBYY Finereader		Tesseract	
	Original image	Processed image	Original image	Processed image	Original image	Processed image
Basic	0.657	0.866	0.849	0.927	0.889	0.911
Recall	0.806	0.895	0.887	0.942	0.901	0.928
Precision	0.779	0.890	0.879	0.937	0.907	0.929
Hybrid	0.684	0.815	0.802	0.893	0.840	0.868

Table 3.1 shows the performance comparison of four different OCR platforms. As a result of this comparison, we can see that the Tesseract sub-structure has a significant advantage over other infrastructure. The measurement here is given as the ratio of the number of correct character sets to the number of incorrect character sets. The flow diagram of the Tesserract OCR structure is shown in Figure 3.1.

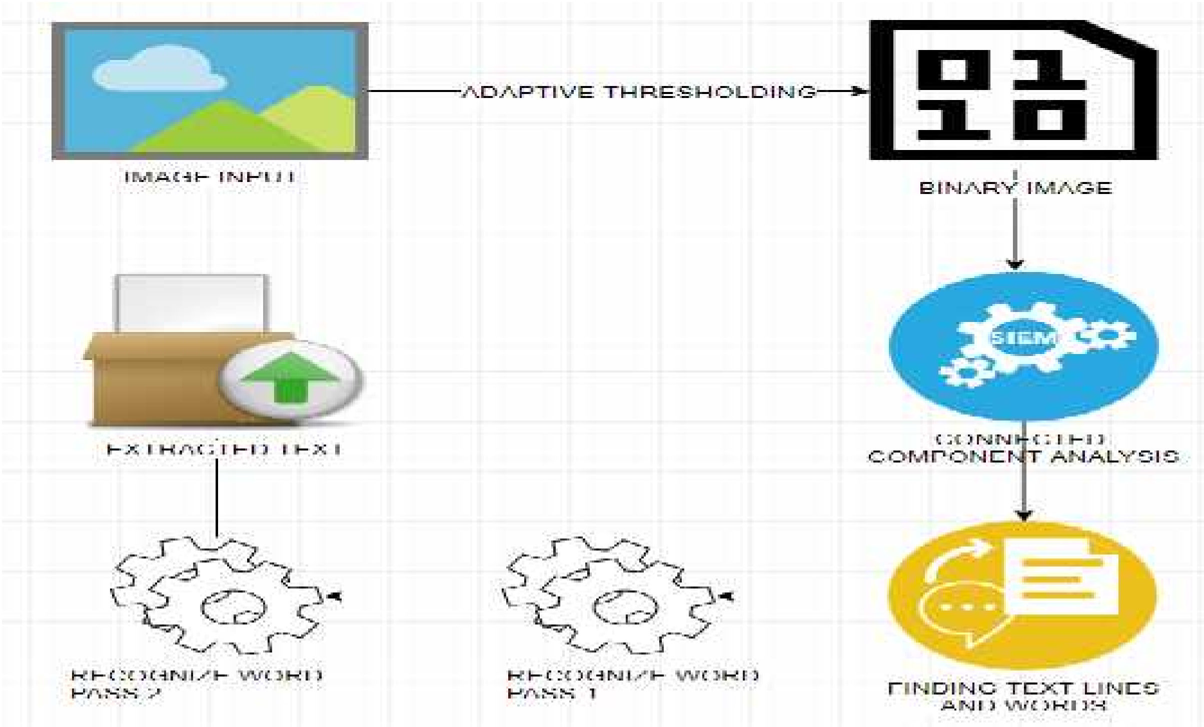

Figure 3. 1: Tesseract Process Flow

3.2 Text-to-Speech (TTS) Technology

The TTS architecture, which provides character-based conversion to the voice-based data, is the second major step of this project. The conversion of written text to audio data works with the server-client architecture. The infrastructural architecture used is Google TTS services. Textual data on this architecture is sent to Google's servers via Google TTS web services. Subsequently, the data processed by the server is transmitted to the client side as voice data. The continuously evolving Google TTS services offer a multitude of languages around the world. For this reason, application architecture will be able to serve many users. There are also disadvantages with the flexibility that Google TTS services provide. This is because the TTS infrastructure needs internet connection. The user will be deprived of audio conversion functionality because it is not possible to access server side services when the user does not have an internet connection.

On the other hand, processor and storage resources will be wasted as extra to enable the sound conversion architecture to be performed locally. The end result is a huge performance loss. Many locally used TTS infrastructures are not continuously developed. In addition, the language support for this conversion architecture is also limited. As a result, when we compare the advantages and disadvantages of the chosen architecture with the advantages and disadvantages, it seems clear that the architecture has more advantages.

3.2.1 TTS technology structure

We can examine TTS technology, which provides the voice conversion of character based data, in two main categories architecturally. These are server based client service applications that can run online and the other category is applications that are embedded in the local system. The basic functionalities in both categories are almost the same and algorithmically separated from each other. Functionally, TTS operations can be completed in 9 different steps. The flow chart of the process is shown in the figure below. The character data from the user is controlled by a dictionary based on word-based analysis in the word separator processor. As a rule, the analysis commences after this phase, and regular analysis of the data is obtained after analysis of the cues. Preliminary data preparation will be finished. Sentence accents are prepared in voice data conversion operations. These sentence accents are processed according to the chain of rules of the voice data generator processor. After the generated audio data passes through the various audio editing filters, the audio data is taken as output. After all these operations are

completed in the local (mobile, pc, etc.) environment, the output of the audio data is presented to the user by the local media center.

In the other category, all of the above mentioned processes are prepared in the server environment and transmitted to the user as sound output. The MyReader application developed in this study adopts the second category.

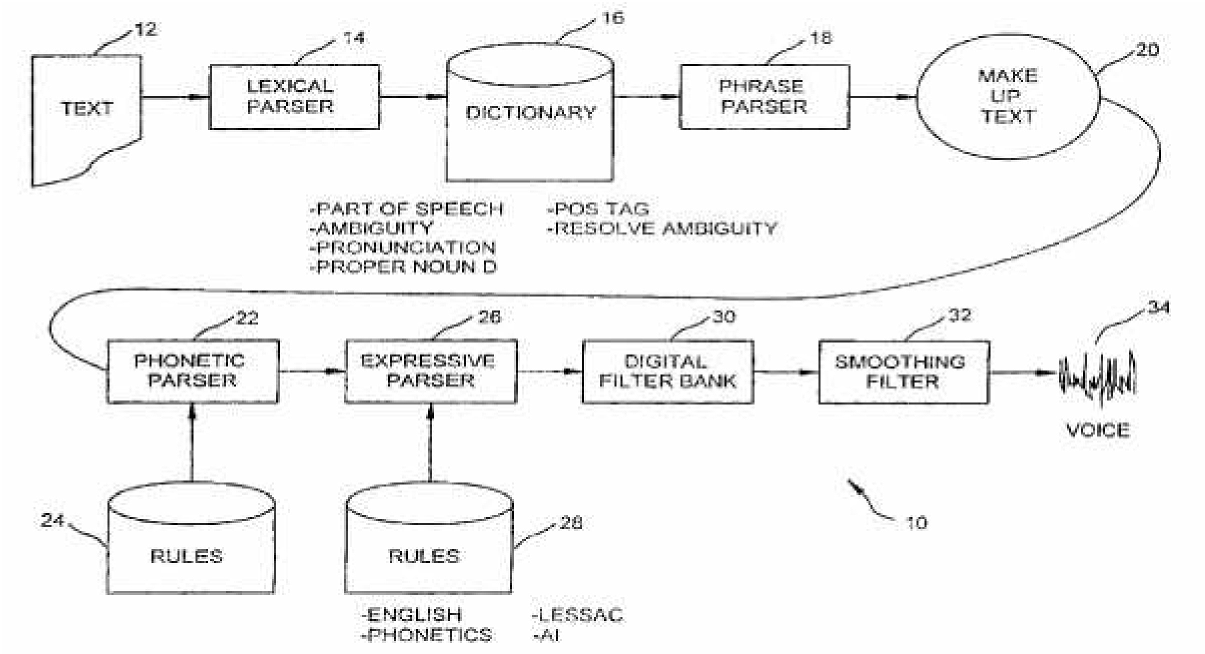

Figure 3. 2: Text To Speech Process Flow (Addison,2005)

The output from the user as textual data after the OCR operation is transmitted to the Google TTS services via HTTP "Post" method. At this stage, after the implementation of the process, the voice data is transmitted securely to the mobile client side. The flow diagram of the process is shown in Figure 3.3.

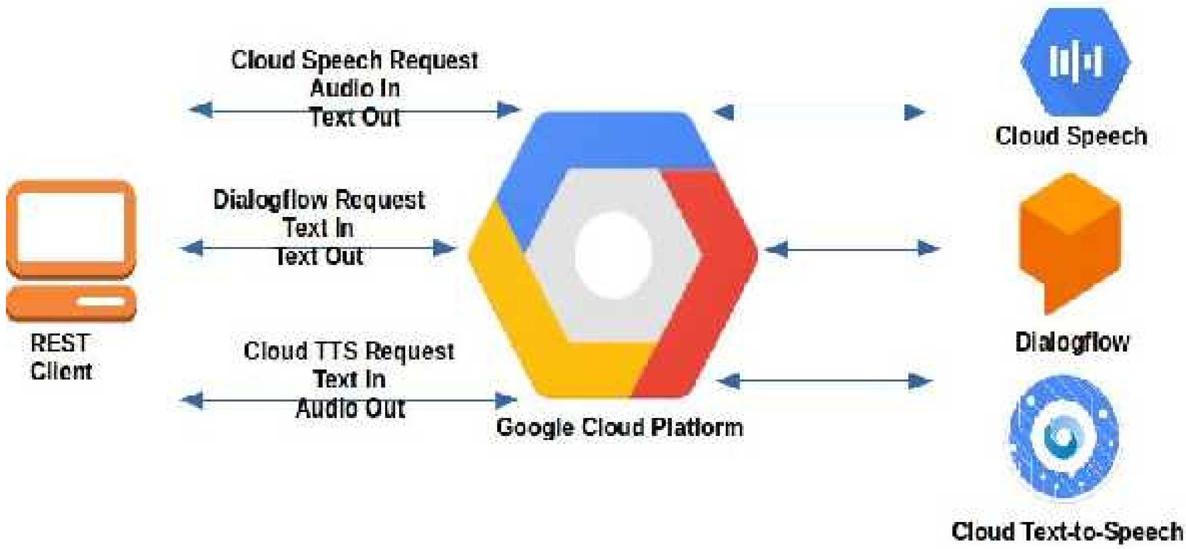

Figure 3. 3: Google TTS Restfull Web Service Architecture Diagram (Whelan, 2018)

3.3 Cloud Systems and Data Management

The foundation of cloud computing is the distributed computing that occurs in data centers located in physically separate geographical locations where data are held. Due to the fact that cloud computing history is a technology based on close ties, this model is made possible through the use of past web services, virtualization and grid computing technologies.

Web services are platform independent software that can be accessed from the Internet. Web services can be used on different platforms thanks to the advantages of being open source and can be developed by independent users and show a rapid development graph. Web services use standard protocols such as SOAP and XML in the internet environment as interfaces and they are low cost and provide faster development opportunities for software developers. In this way, software developers can integrate their programs with other services on the internet in order to create lower cost programs

Virtualization is a technology that provides server efficiency optimization by dividing logical parts into a desired physical ratio. With virtualization technology, as the number of physical computers decreases, the number of virtual computers increases and the available hardware capacity can be used optimally. At this rate, labor cost efficiency and flexibility are increasing while cost is decreasing.

The ever-increasing needs and expectations require today's organizations to have a large number of hardware units in their businesses. Institutions need a large number of servers and storage units to accommodate many business applications such as web server, database server, enterprise resource planning (ERP) systems and customer relationship management (CRM) systems. In order to reduce these hardware costs, businesses prefer virtualization technologies that can have different operating systems. It is possible to create more than one virtual server on a physical computer, which saves on maintenance and capital investments as well as significant reductions in the costs of the enterprises, resulting in a more environmentally friendly system by reducing energy consumption. Maintenance and management of these systems is crucial for businesses, as systems are essential for uninterrupted service businesses, and virtualization technologies provide important advantages in this important process. Grid computing technology is the sharing of computer resources, which are physically located in separate places, by means of high-speed networks. Resources such as calculations, storage and memory on computers are evaluated and idle capacities are evaluated to reach higher capacities and productivity is increased. Grid computing also allows computers located in different physical environments to combine the computing power, as well as to run programs in parallel and distributed manner with the sharing of multiple computers.

The key features of cloud computing are distributed architecture, scalability, low cost, security, media independence, multi-use, maintenance, reliability, performance monitoring, continuity and business process improvement. We can categorize the activities of cloud technology in the field of data management as shown in Figure 3.4.

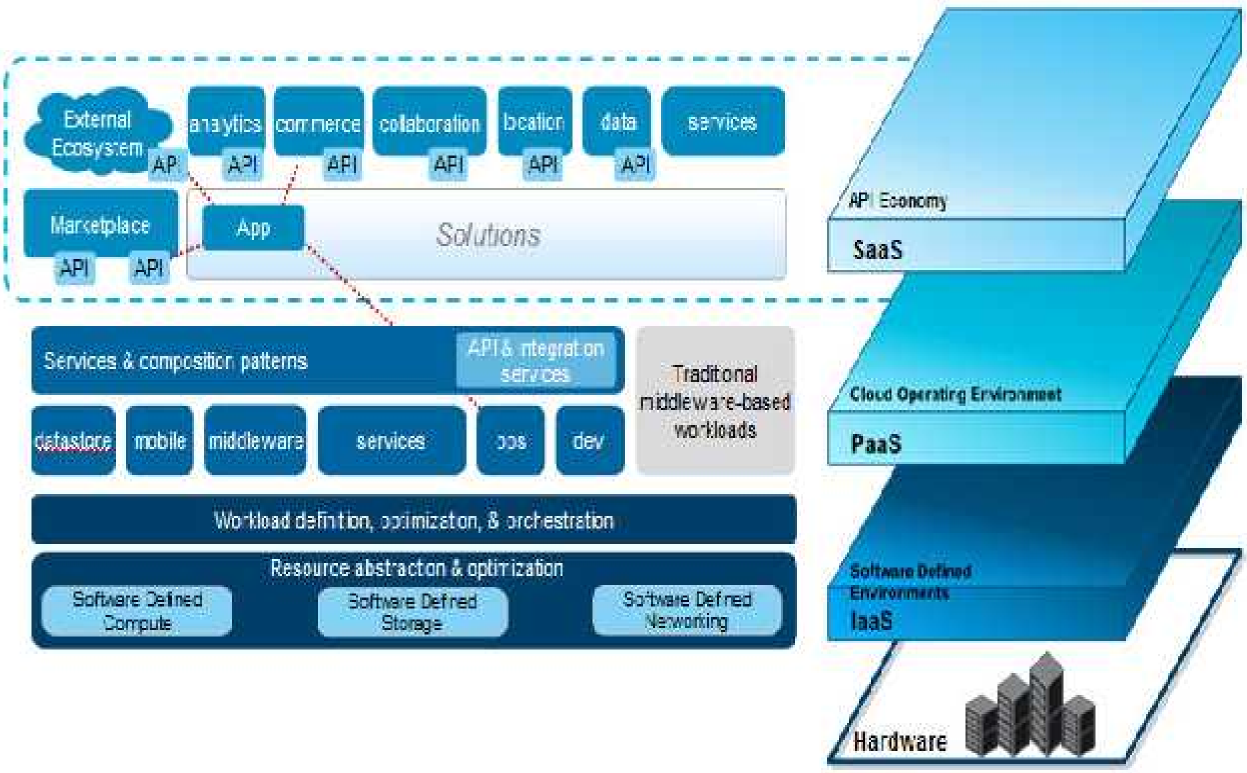

Figure 3. 4: Cloud System Infrastructure (Riousset, 2013)

3.3.1 Cloud systems and data management structure

The infrastructure of cloud technologies can be generally examined in three subcategories. These categories are as follows; Software as a Service (SaaS), Platform as a Service (PaaS), Infrastructure as a Service (IaaS). Each cloud technology layer provides solutions in different areas. Layers responsible for the development, control and distribution of various web-based services using the virtualization infrastructure have made service tree structures more flexible.

Software as a Service (SaaS): Users can access their applications systems without having to install any software through internet browsers to access applications on cloud computing systems. Clients do not control or control components such as network, server, operating system, and storage devices in the infrastructure. They can only make application-specific adjustments that they use.

Platform as a service (PaaS): The service provider provides a platform for the customer to develop and run their own application. This platform includes complementary services and the necessary technological infrastructure, along with the environment in which the application will be developed. Apart from the user's own application, there is no control and management over the components that make up the platform infrastructure.

Infrastructure as a Service (IaaS): In the model of serving infrastructure as a cloud service, the customer can configure the necessary processor, storage, network resource and other basic information resources and implement the operating system and applications on them. Although the client has no management and full control over the network structure. Figure 3.6 shows the management features and functional infrastructure structure characteristics in the categories of cloud technologies.

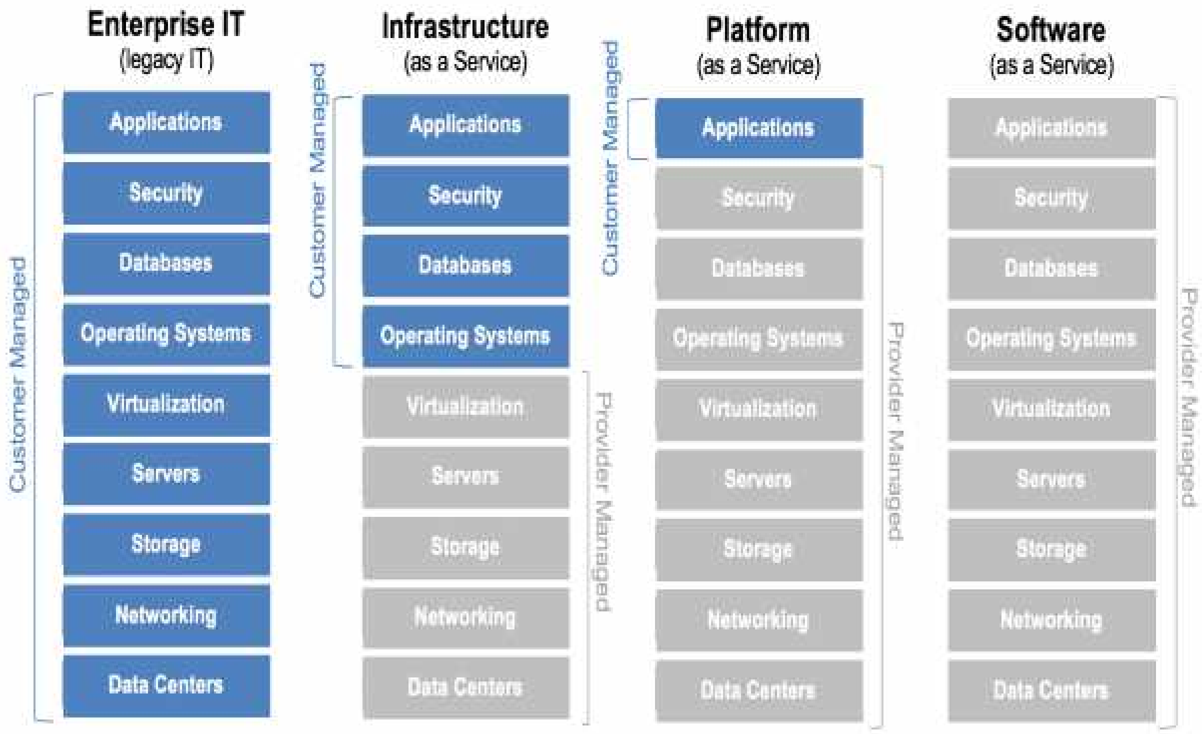

Figure 3. 5: Cloud System Functionalities (Bond, 2013)

3.3.2 Advantages and disadvantages of cloud processing and storage systems

The solutions and advantages offered by cloud technology can be listed as low hardware cost, low software cost, current system structure, almost unlimited storage capacity, high level data security. Among these advantages, it can be said that time cost is low. Among the most important features is the reduction of data loss to the minimum, the data backup can be done in a short time and the data sharing can be done very quickly. With cloud technology, textual data obtained through MyReader application developed for this study can be stored in media such as Google Drive, Microsoft One Drive, and shared easily.

However, the disadvantages of cloud technology include the need for continuous internet connection, the possibility of service interruption at low connection speeds, the possibility of service interruption due to various updates.

3.4 Mobile Applications Development

Over the last years, mobile systems and mobile applications have drawn lots of attention of different researchers and institutions. Mobile devices market seems to be constantly growing as time passes (König-Ries, 2009).

Today, mobile devices are able to run robust standalone applications and even distributed client-server applications that could use the web gateway to access information better than ever before. These and so many features of the mobile devices have created an open avenue for future development of mobile applications and services. Some years earlier, mobile services development was largely handled and managed by phone producing companies, mobile network operators (MNO) and some of the big mobile application and content providers. In recent times, things have taken a different dimension with the arrival of new mobile phone technologies and platforms like the Android and iPhone. Today, independent and freelance developers has a lot of interest in the area of mobile applications development.

Sofware Development Kit (SDK) is a very vital part of every mobile application development platform. This is because it gives the third party developers the opportunity to deliver different applications running on the specific development platform. The kit includes utility features like debuggers, libraries, emulators and so on. To address the issue of sharing the SDKs with developers, different existing platforms adopts different approaches. Amongst the platforms, some decided to have high access restrictions while others have chosen to make the entire source code of their SDK and OS openly disclosed and free (Holzer and Ondrus, 2009).

3.4.1 Types of mobile applications

Most of the mobile devices of nowadays are seen to be running on iOS, Android and Windows operating systems. The operating systems are also called operating platforms in most cases (Mallıkarjun, 2017). On the grounds of the technologies and platforms used in the development of mobile applications, they could be classified into three namely Native , Hybrid and Mobile web applications (Sharma, 2016).

Native applications: these apps are developed for specific operating system. A native app developed for a specific platform can not run on another different platform. For example, an app built for Android can not work on iOS or Windows operating system running devices and vice-versa. Applications developed here will always remain dependent on their platform. If such app is required for another platform, then it has to be developed again specifically for the new platform of interest. Software platforms and languages that support the development of native applications would generally be the likes of Swift or Objective-C for iOS, Java and ADT for OS and .NET(C#) for Windows operating system.

Mobile web applications: The applications that render web pages on a browser running on a mobile device are referred to as mobile web apps. These applications work on the different operating systems because they target browsers and not the mobile device platform. Mobile web apps are easily viewed on android, iOS or windows devices. Moreover, they could even be viewed on a PC's web browser. Development languages for building mobile web apps includes Hyper-Text Markup Language (HTML), Cascading Style Sheet (CSS), JavaScript, jQuery and so on.

Hybrid applications: these are a combination of both the native and the mobile web applications. They are best known for their cross-platform compartibilty. A closer look at these type of apps will make you know that they are similar to mobile web applications in structure since they are also built using technologies like Hyper-Text Markup Language (HTML), Cascading Style Sheet (CSS), JavaScript, jQuery, Mobile Javascript Frameworks, Cordova/PhoneGap, Ionic and so many others.

3.4.2 Operating systems of mobile devices

A mobile operating system (OS) runs on a smart phone, tablet, PDA or other mobile devices. The smart phones combine some features of a personal computer with cellular technology such as wireless networking, Bluetooth, GPS navigation, touch screen, music player, camera, and other features. The mobile OS controls all these features and provides to the users the accessing ways and interacting with them (Ballagas et al. 2006). The major mobile OSs platforms are Android, IOS, Windows Phone and BlackBerry, based on the market shares. The share of Android is 79.3%, IOS is 13.2%, Windows Phone is 3.7%, BlackBerry OS is 2.9% and other platforms 1.0% (ABI, 2013).

In this section we are going to discuss about the three main operating systems that are most commonly used in this current time.

3.4.2.1 Android OS

Android is a comprehensive open source platform designed by Google and owned by Open Handset Alliance. This alliance aims to accelerate innovation in mobile computing and offer consumers a richer, less expensive, and better mobile experience. Android is a Linux- based operating system, mainly used for running mobile devices such as smartphones and tablets (Butler, 2011). September 23rd, 2008 was the initial release date of Android. Android's source code is made fully available to the manufacturers. The copyright holders gives the right to learn, manipulate and share the software to anybody and for variety of purposes. Java, C and C++ are the programming languages used to code Android operating system.

The Google Play Store has released over 3,000,000 Android applications by June 2017. As at last year 2017, over 80 billion android apps have been downloaded by individuals. The Google I / O company has also found that they had over 2 billion active users every month. This shows an increase from their previous year's number of active users which was estimated to be 1.5 billion active users per month. From the birth of Android to the present time, android had came up with multiple number of advancements in the form of upgrading their operating system through addition of features and fixture of erros found in the previously developed versions. Each new version developed was given a name after a desert in an alphabetic arrangement; Cupcake 1.5; Donut 1.6; Eclair 2.0; Froyo 2.2; Gingerbread 2.3; Honeycomb 3.0; Ice Cream Sandwich 4.0; Jelly Bean 4.1; KitKat 4.4; Lollipop 5.0; Marshmallow 6.0; Nougat 7.0 and Oreo 8.0 versions (Lazareska and Jakimoski, 2017).

3.4.2.1.1 Advantages and disadvantages of Android OS

One of the greatest advantages of Android is that it's code is made open (open source), which gives room for development of millions of applications that could run across many devices such as smartphones, tablets, watches, head monted devices and so on. Android also gives a smart device owner the freedom to download and install applicatons of their desire from the Google Play store or as an executable (APK) file. Another advantage is that android is a very intelligent multi-tasking system that can manage many simultaneously running applications. An important advantage of android that grossly increases user experience is it's complete customisability. Android has many advantages few of which are mentioned here.

However, it also has a number of disadvantages. Android allows numerous open apps which causes a very serious battery consumption. Also, due to the openness of the platform, the operating system is mostly prone to a lot of vulnerabilities as well as cyber crime activities. Other disadvantages are that Android is difficult to manage because it is not very intuitive and that the system does not always close all applications even though we do not want them open some times (Lorecentral, 2017).

3.4.2.2 IOS operating system

Apple Inc. developed the iOS operating system predominantly for smartphones and tablets. iOS was firstly released on June 29, 2007. iOS is closed source, which mean that its source code is not made available to manufacturers or developers to build their own version of OS based on the iOS. For this reasons it is regarded as a proprietary operating system. The iOS is written in C, C++ and Objective C programming languages (Padhya Desai and Pawade, 2016).

Apple's App Store comprises over 2.2 million iOS applications as at January 2017. In general, the whole applications were downloaded over 130 billion times. Still in 2017, it was recorded that the iOS had 710 million active devices in use. In the newest data gotten on the basis of the reports established in the third quarter of 2017, Apple proclaimed that it has sold over 1.2 billion devices (Lazareska and Jakimoski, 2017).

3.4.2.2.1 Advantages and disadvantages of IOS

The iOS is highly advantageous as described by many of its characteristics and features. Some of its advantages are listed below:

- It has very high performance.
- Less heat is generated when compaired to Android.

- It has the excellent business and gaming experience.
- Excellent security, user interface and support multiple languages.
- It allows for multi-tasking.
- Wearable are getting launched.

However, the iOS platform is also associated with some drawbacks which are listed below:

- It is not flexible because it only supports iOS devices (The iOS is not Open Source).
- Devices and numerous applications are costly to purchase.
- Not highly customisable when compaired to Android.
- It has poor battery performance when on 4G.

No Near Field Connection (NFC) or Radio support.

3.4.2.3 Windows IOS

On November 8, 2010, Microsoft Corporation initially released the Windows mobile operating system which was designed and developed for smartphone and tablets (specifically touch screen devices). Windows have released also different versions of their operating system. The Windows OS is a closed source operating system just like the iOS. Meaning, it does not share it source codes with manufacturers or developers to enable them to create other versions of the OS. It is programmed in C, C++ programming languages.

Like any other operating system or platform, windows also has its advantages and disadvantages. One advantage of windows is that in the recent version, side loading is allowed. It is easy to roll out updates in a timely manner since Microsoft has greater control over their applications. They also posses high security feature. They seem to be the hardest devices to crack as of recent times. Windows OS offers accurate and the best results when it comes to queries regarding location (Padhya Desai and Pawade, 2016).

However, Windows also has some disadvantages. Some of the disadvantages of windows are briefly discussed within this section. Among the disadvantages of Windows OS is the number of available applications and their quality. Currently, we can say that major problem encountered by window phone users is lack of availability of apps when compared to other platforms. Also, the Microsoft Store is noted to be behind in terms of usability and aesthetics. Another issue is that of customization amongst many others (Murali, 2016).

3.5 Summary

In this section, detailed explanations of technologies used in the study have been made and the infrastructural characteristics of these technologies have been mentioned in general. İt was discussed that the use of Tesseract libraries in the infrastructure of the OCR system will ensure that the application runs 24/7 without interruption. We also discussed the concept of TTS technology where we mentioned that the most important reasons for the use of the Google TTS and Google Translate infrastructure, which use cloud technology infrastructure among TTS technologies, is that they can be serviced at high speed and support almost all languages in the world. It is important to offer high data retention capacity and to be able to share data within a very short period of time. So, we also metioned the Google Drive infrastructure, which also provides cloud technology solutions for data backup and sharing.

Moreover, mobile applications development, the types of mobile applications and the Operating Systems (OS) that support different mobile devices were discussed in the chapter.

CHAPTER FOUR

SYSTEM ANALYSIS AND DESIGN

4.1 System Architecture

The Tesseract framework is used in system architecture to enable conversion of image-based data to word-based data conversion. The Java infrastructure, which allows Tesseract libraries to be used by many operating systems, is a great advantage. Image-based data obtained with the help of the handset's camcorder is converted into word-phrase data with the help of the Tesseract system. Depending on the needs of the user, the converted data is integrated with the Google Translate services, which makes it possible to translate it into its own language, and the conversion is done in this way. Depending on the user's requirements, word-based data can be converted to voice data in the native language with Google TTS services. In Figure 4.1, all operations are shown in order.

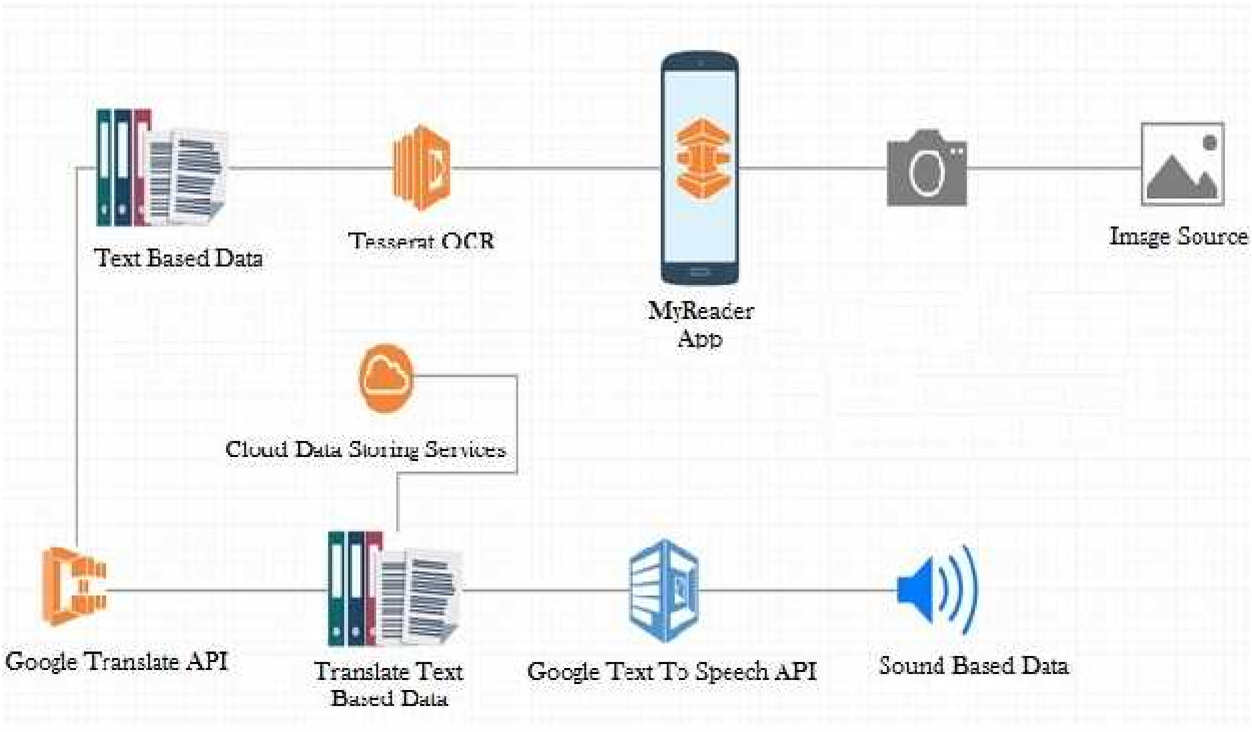

Figure 4. 1: System Architecture

4.2 System Description

The general structure of the system consist of five sub-systems working together synchronously. These sub systems are depicted in the system architecture. The mentioned sub-system structures in the architecture works as follows:

- Optical Character Recognition (OCR): The OCR system allows image-based data to be translated into word-based data.
- Language Conversion: Allows users to translate word-based data into the language of their choice.
- Text To Speech (TTS): Provides voice-based conversion of word-based data.
- Data Backup: It allows the system-generated data to be stored and backed up in the cloud environment.
- Data Sharing: It provides sharing in various data sharing environments created by the system.

A more practical description of the systems is, when a user of the application; most likely low vision individual starts it, three options are given. The options are whether to snap a document containing text to be read by the individual, to access a document which is intended to be read from the phone storage or to access a document from cloud storage. The application then applies the OCR technology to extract the text from the image. Then, it gives the user an option to convert the recognized text to the language of their preference using the Google Translate API. Moreover, the system allows for rendering or converting the original recognized text and the translated text to speech or audio format. All activities carried out in the system are also backup for reference purpose. Finally, the application can even allow the processed data to be shared on various environments.

4.3 System Technologies

In this section, a brief discussion about the system technologies is presented.

4.3.1 Mobile applications technology

The developed system is a mobile application targeting smartphone users, because of the mobile device advantageous properties like the device mobility with user, its portability, allowing access to internet, easy to use, etc. The author used set of mobile development technologies and tools to develop application for the system. The most important tools that have been used are:

- **Android Studio:** Integrated development environment (IDE) tools was used to help the developer to develop application. The Android Studio 5.0 version was used by developer in carrying out this project.
- **Android SDK:** The different libraries in the Android system depend on the versions. (SDK) libraries are development tools necessary to build and test applications for Android, The application that was developed in Android 5.0 API level 14 or above.

4.3.2 Server-client applications and cloud computing

Some of the cloud technologies used in the developed system are cloud service integrations of the Saas type. The main cloud services integrations used are Google Drive cloud services, Microsoft One Drive, Google Translate cloud service, and finally Google Speech cloud service.

4.3.3 Programming language technology

Mobile applications can be written in several different programming languages and it depends on the mobile phone platform, to write mobile applications program we can use Java in the case of Android native apps, objective C for iOS, C# for Windows Mobile, Blackberry etc. Therefore, in this application Java programing language is used because of its advantages as mentioned in the previous chapter. The greatest advantage of Android is that it is open source.

4.4 Application Features

- **Image Optical Character Recognition:** This feature allows the user to extract the word data contained in the image files and translate it into an editable form. In this case, the analytical data, which cannot be analyzed, is transformed into analytical form. The readability of easy-to-read cursive data is increasing.
- **Document to Voice Conversion (Text To Speech):** This feature allows the user to convert voice data into voice data. Documents that are difficult to read and long-lasting can be analyzed within minutes.
- **Document language Translation:** This feature allows the user to translate the data in the document into his or her own language. This makes it easier to understand documents.
- **Document Cloud Sharing:** Use this feature ensures that documents can be backed up in the cloud environment. Use this to reduce data loss to a minimum level.

4.5 Unified Modeling Language (UML) and Use-cases

The analysis of use-case, illustrates the way how system behaves, how the sequence of actions for each request is applied, and relations between the user and other subsystems. Here how the user can get the benefits of the system through some of potential usage scenarios is explained.

4.5.1 OCR operation actions

- **Choosing Image Data Source:** At this stage, the source of the pictorial data to be subjected to the OCR operation can be selected from 3 different points. First, the user can use the camera function as a picture source. With the camera function of the mobile phone, the user is directed to the picture editing screen by providing the image data. As a second data source, the user can select an image file manager that is already in the file system. The selection is then directed to the image editing screen. Finally, the user can switch to the image editing stage by selecting any image from the cloud data sources.

- **Choosing Image Part for OCR Process:** At this stage, the image editing step is performed to allow the user to select a specific area on the picture for OCR operation. In large size images, the entire image field may not contain word data. For this reason, the user can select the data field required for this step.
- **OCR Processing:** At this stage, the image area that the user has specified is filtered by the software and word processing is performed to finalize the user.

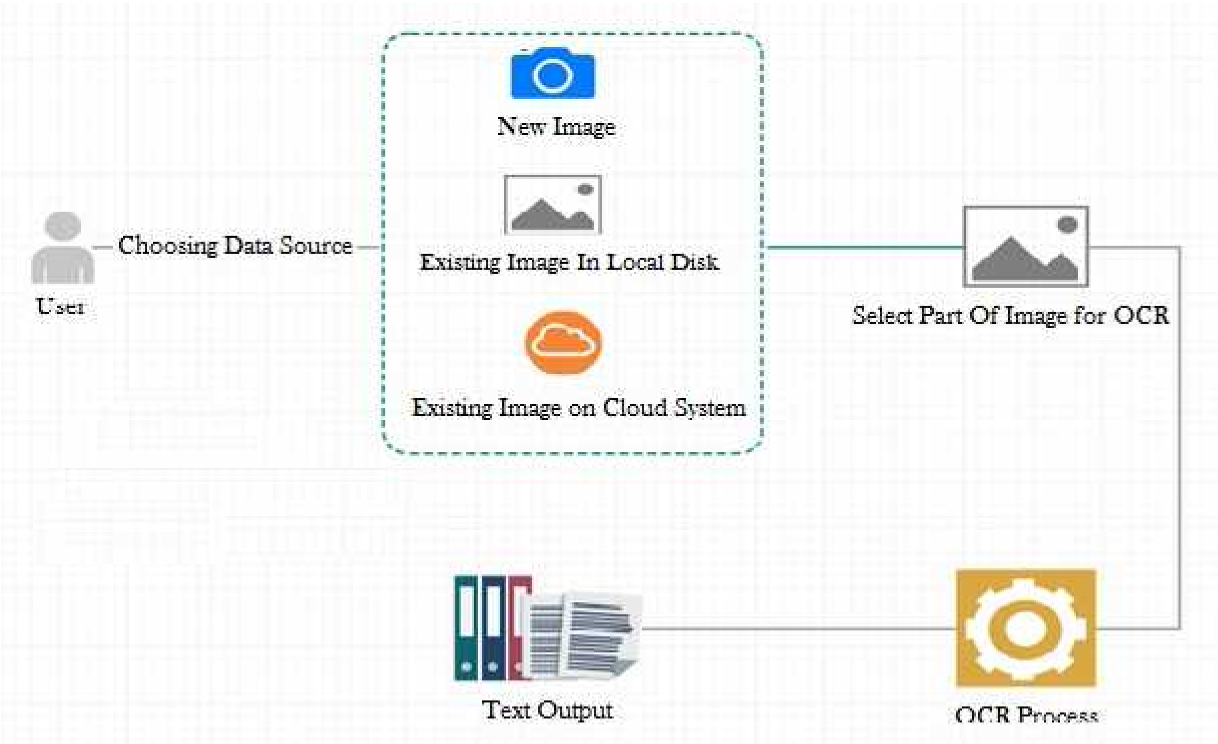

Figure 4. 2: OCR Processing Diagram

4.5.2 TTS operation actions

In this phase, word-based data is sent by the user to the Google TTS services and the user is presented with the audio data returned via the service.

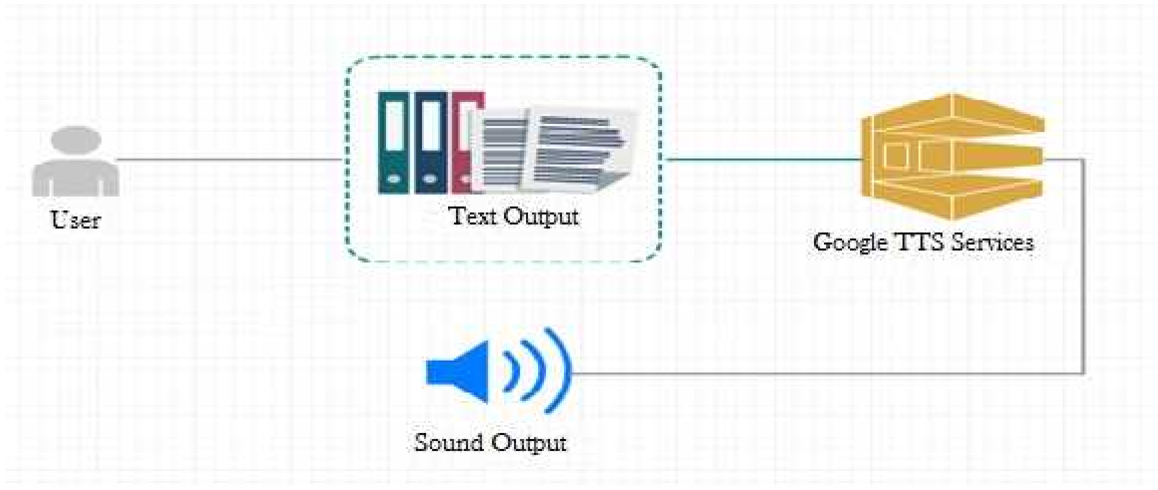

Figure 4. 3:TTS Processing Diagram

4.5.3 Cloud data backup operation actions

In this process, the user can transfer the textual data obtained by the user to the cloud services where the account is located.

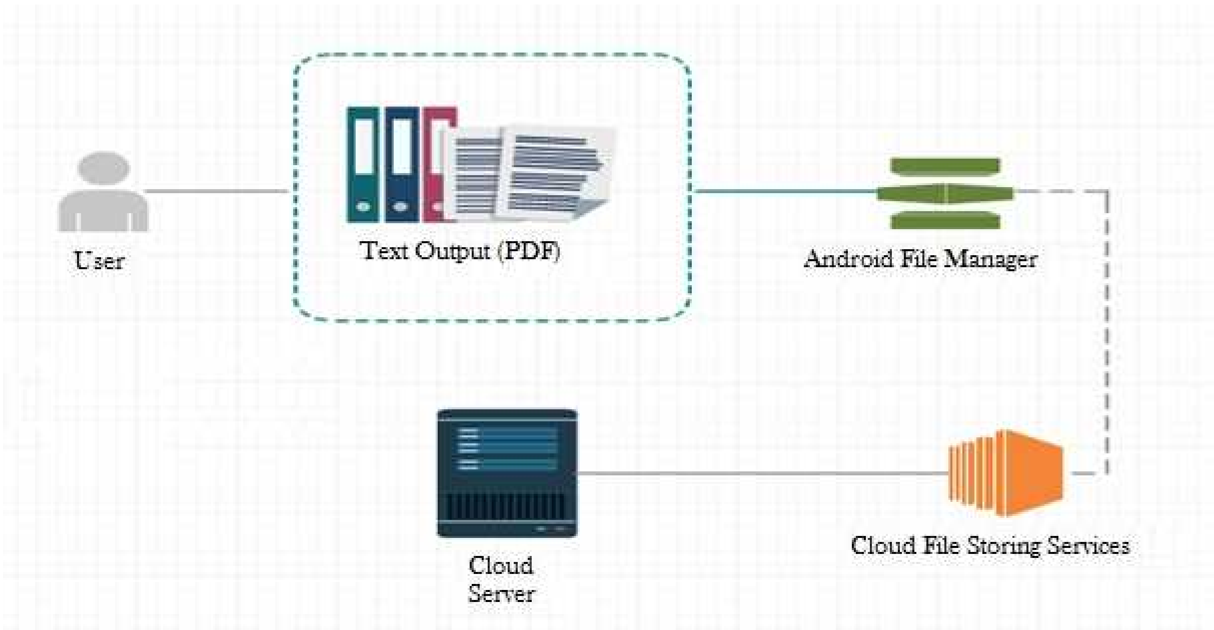

Figure 4. 4: Cloud Processing Diagram

4.5.4 Data sharing operation actions

Choosing Data Sharing Platform: In this phase, the user chooses the platform he wants to share the data, then the data is transferred to this platform. The example, a processed data could be shared via email, or any social media platform or a blog.

4.6 System Development Methodology

A large application development project should include the adoption of systems analysis and design methodology. In general, developing an application for mobile devices is very similar to other software development; some of the issues include combining with device hardware, as well as come other conventionalproblems of performance, safety, reliability, and data saving limitations. Moreover, mobile applications present some additional requirements that are less common with traditional software applications, such as power consumption, complexity of testing, user screens, possible interaction with other applications, and basic and combined applications (Wasserman, 2010).

There are various software development methodologies or models. The most commonly adopted are as follows:

- Waterfall
- Rapid Application Development (RAD)
- Prototyping
- Incremental
- Spiral.

In this project, RAD methodology was used because it is the best fit for this project especially there is one developer and reduced development time in addition to increase reusability of components. The main reason why RAD is the best fit for this project is that this project is highly interactive with two clearly defined user groups: visually impaired persons and students. The RAD methodology is also of a great fit for the way the developer prefers to work on a project in the sense that it is broken up into manageable iterations.

4.7 User Interface (UI) Design

Using UI principles and guidelines is very important to develop and design mobile application, because the UI design can be useful toprogrammers to increase their performance and understand application. A basic user design of screens will mainly support the application popularity. Providing an intuitive and user-friendly screen is a challenging task. This challenge is even harder when it comes to mobile phones such as devices that have limited interaction possibilities and smaller display size. Therefore, the designer needs a strategicunderstanding of the design to be included in the design of the mobile application. The designer chooses a motivational design strategy that depends on Attention, Relevance, Confidence and Satisfaction (ARCS) becauseof its suitability for the design strategies of mobile applications (Seraj and Wong, 2012).

4.7.1 Design arguments

In general two parameterscan be considered while developing a mobile application. Mainly, users and the concepts of usability of the developed application are the two major points studied by the researchers (Grasso and Roselli, 2005; Seong, 2006).

Perhaps user is one of the most important parameter that needs to be studied and also evaluated by the systems developer while developing a mobile application. Therefore, it is good idea that the system developers should examine users' details such as age, gender, and the highest level of education and also their knowledgeof the mobile devices. Suchstudies can usually help developers as well as to designers to be aware of the user's responss and their expectations while using a mobile device. Evaluating the users' responses before carrying out any further development is also importantas it helps users to use the developd application correctly and easily. If this is not done, there may be serious difficulties for the users while using the developed application.

Usability of the mobile application that was developed is an important factor, especially attention should be paid to the screen design as this is how the users interact with the application. Usability can be measured by the quality of users' experience during the interactions with the screen of an application (Seraj and Wong, 2012). The reasons to choose usability are (Fetaji, 2008):

- Decreasing the cost as well as the time of reading, understanding and analysing
- Increasing the performance of learning and also the satisfaction of the users
- Decreasing or removing totally any errors that the users may come across while using the system
- Making the screens more readable and of higher quality

Therefore, a number of design principles are recommended to be followed to provide a satisfactory mobile application as far as the usability is concerned. Moreover, the user screen size limitations of a mobile application, for example the small size screens, low resolution, limited data storage, and low processing power should also be considered.It can therefore be said that the screen design should be simple and should not include complex figures or text and there should not be need for very fast processing power.

4.7.2 Principle of user interface design

The principles of User Interface Design (UID) according to ISO 9126-4 (2004) and Motiwalla (2007) which was employed in developing the mobile application is as follows:

1. Navigation should be simple and clear from a page to any particular section. In short, navigation should be consistent throughout all pages in an application.
2. Minimize scrolling.
3. Developed applications should be as user friendly as possible and allow the users to learn how to use the system easily and with no difficulty.
4. Similar buttons and keys should be located is same positions to make the application easier to use.
5. Finally, the screen design should be as flexible as possible as this is an important property.

4.8 Summary

In this section, the technical infrastructure design of the thesis project, architectural and functional design of the technologies used will be explained in the architecture. The technologies used to ensure the maximum level of productivity of the software are described in Chapter 3. The systematic design of these technologies will work on the Android OS on the operating system side. The program language is the Java language used in the system subdirectory. Server-side operations of the mobile application have been done with service integration. Service integration schemes and internal process design will be explained in the system architecture section. In general, the architecture in the software is shown in Figure 4.1.

CHAPTER FIVE

SYSTEM IMPLEMENTATION

5.1 Introduction

In this section, the functional features and user interfaces of the MyReader mobile application developed is discussed. Each screen from the start of the application is screen-grabbed and explained. Also, the screenshots of how the application is used and how it carries out its functions are fully discussed.

Figure 5. 1: Home Screen

The figure above shows how the first interface of the MyReader application looks like. It contains the menu option, language setting and the file access option which could be directly from the camera or from the device or from Google drive.

Figure 5. 2: Release Note

The figure above displays all the features of the application. It shows a tick beside all the services it provides.

Figure 5. 3: About/Contact Information of the Creator

This page gives a brief description of the application and also give a user the opportunity to give a feedback to the developer via email.

Figure 5. 4: Manage Languages

The above interface displays all the languages which could be added to the application. A user can download and delete languages whenever they want.

Figure 5. 5: Sample Snapshot

This interface shows an example shot of a text document which was taken from the camera of a phone through the MyReader application installed on the smartphone. The text document was a printed A4 paper. Moreover, the language of the text is English.

Figure 5. 6: Warning Message

The interface above shows a warning message that was displayed after snapping a low quality picture of a document. The application tells the user that the picture is not clear whenever a low quality image is snapped to avoid too much errors in the character recognition. To solve this issue, the app allows the user to retake another image or continue with the blurry one as they wish.

Figure 5. 7: Crop Snapshot

After taking a clear shot of the document, the user is allowed to crop a portion of the document that he wants to make use of. If the whole document is what he desire to be recognized, then he draws the crop canvas to cover the whole text of the document. The interface above shows an example of a cropped area of a scanned document.

Figure 5. 8: Column and Language selection

In the screenshot above from the application, the options for choosing the number of columns in which you want the recognized text to be arranged and the language in which the target text to be scanned are displayed for the user to select to enable the system to function properly.

Figure 5. 9: Character Recognition Process

After scanning and cropping the required section of a document, the application then tries to run a character recognition process as shown in the figure above. As it recognize the characters, it shows the percentage covered from top to the bottom of the scanned area.

Figure 5. 10: Output of Recognition Process

The interface above shows the result of the character recognition as carried out on the scanned document. It also displays the options provided by the application. The actions in the option can be to share the text, copy it to clipboard, translate to a different language, convert the text to speech, generate a pdf file or switch between the image and the recognized text to allow for zooming.

Figure 5. 11: Text Settings

The interface above shows the options associated with the text setting of the application. It allows a user to choose the line spacing and the text alignment.

Figure 5. 12: Table of content

The table of content interface of the application is where the tasks carried out on the application are saved based on selected file activity. All files saved on the same day are kept and displayed in one row indicating the number of files. This is just like an alternative to the tile display of saved files at the home page of the application.

Figure 5. 13: Edit, Copy and Paste Options

When the recognition is done, the app presents the text to the user in an editable format. A user can further copy or cut from the text, or paste other text into the one presented by the app. Sharing could be done through this interface also.

Figure 5. 14: Document Title

On the tiles display of files, a user can click on the pen icon above a saved scan to give it a title as shown in the above interface.

Figure 5. 15: Output Translated to Arabic

The above interface shows how the application translates the recognized text to Arabic language. It also provide a text to speech feature for the Arabic text.

Figure 5. 16: Output Translated to Turkish

The above interface shows how the application translates the recognized text to Turkish language. It also provide a text to speech feature for the Turkish text.

Figure 5. 17: Generated PDF file

Above interface shows one of the generated pdf files from the app. The whole files scanned in one of the days is converted into one pdf file. You can see that we have two languages within the file; this is just for example purpose to show the difference in the files.

Figure 5. 18: Share Options

Sharing of captured and recognized text is allowed by the application via many other applications; like WhatsApp, Facebook, Email, direct message and many more. It also allows for sharing via Bluetooth and Wi-Fi. Some of the platforms that allow sharing of the text are shown in the interface above.

Figure 5. 19: Accessing files from device

One of the important features of the application is accessing files already stored on the device. Here, the user does not need to take a picture of a document but instead, the user choose from one of the saved files on the device and then run the character recognition process.

Figure 5. 20: Accessing files from Google drive

Another important feature of the application is accessing files already stored on Google Drive. In this case, files saved in the Google drive are accessed by the app and the character recognition is done just like any other file.

5.2 Testing

Here, the developed application was tested in order to verify that the application meets the necessary functional requirements. The main functionalities verified are optical character recognition (OCR), Text-to-Speech (TTS) conversion, the translation feature and also the PDF generation. To check for the usability of the application, the MyReader app was examined in two case as described in the subtopics below.

5.2.1 Testing on Ordinary Paper

To check for the apps usability, 100 shots were taken and processed by the app from an ordinary printed A4 paper. It was discovered that only 3 out of the 100 shots gave a poor result.

5.2.2 Testing on Newspaper

The second case of testing the MyReader app was by taking another 100 shots from a newspaper. After carrying out the examination process, it was found that the application gave 9 poor results.

The results of the testing indicate that the application has high usability and accuracy. It was found that the shots taken from ordinary paper gives better results than newspaper. However, it was found that the errors or poor results might be because of the low quality of the document or the image taken from the document, it can also be as a result of the level of light the document is exposed to. The distance at which the image is taken might also be a reason for bad results.

Moreover, newspapers are generally printed with darker backgrounds which also might be another reason for erroneous results. This statement could be backed up by comparing the number of poor results in the first test case to the second one.

5.2.3 Testing the TTS Conversion

To test for the TTS conversion ability of the developed application, all the 200 shots from both the ordinary paper and the newspaper was converted to speech. All the recognized texts from the images were successfully converted to speech. However, the error in the converted speech is converted to speech also, since the TTS API converts exactly what it is provided.

5.2.4 Testing the Translation Feature

Just like the case of testing for TTS feature, the translation feature was also tested in all the 200 shots. The MyReader application was able to translate all into Arabic and Turkish languages.

5.2.5 Testing the PDF Generation

To test for the PDF generation functionality of the application, 20 of the generated texts from the ordinary paper and 20 from that of the newspaper were converted to PDF successfully.

CHAPTER SIX

CONCLUSION AND RECOMMENDATIONS

6.1 Conclusion

The MyReader application is designed to make people who have problems in visual sense to benefit from written or printed sources more quickly and comfortably. The application, developed within the scope of this study, has provided a variety of OCR systems to translate documents into an audible and more readable form. With the use of Cloud technology, system cost has been reduced and maximum performance has been achieved. However, integrations that offer flexible development will provide a high degree of convenience for future versions of the project.

With this application, users can convert their documents into voice document form whenever they want, translate these documents into editable form, and also save documents in cloud form that can be edited. These documents stored in the cloud environment can easily be shared with other people and platforms. This application, which enables easy use of information, will be made more automated and available to users in the future.

The System features can work offline and online and therefore providing great advantages. For this reason, functions can be performed in all kinds of environments. The main offline feature of the system; OCR system allows documents to be converted into editable form and converted into more readable form. This feature is the most emphasized feature for the general purpose of the application. Other features from the functional standpoint are online.

6.2 Recommendations

Moreover, by means of the system design of the application and the programming language used, the previously mentioned features of the application has been achieved. This study came up with a system that could be adoptable to other operating systems due to its cross platform workable infrastructure. Therefore, it is recommended that the system is adopted in the future for IOS and Windows mobile devices. Another important feature that could be implemented in future study is allowing a user to save an audio of any text converted to speech when needed. Also, the issue of file format that could be accessed by the application from the device should be increased. This is because at this stage the MyReader application can only support access to few file formats like JPEG, PNG and BMP.

REFERENCES

ABI (2013). IDC Spins a Tale of Android vs. iOS Stats in Smartphones Only - Patently Apple. Retrieved September 9, 2018 from https://www.patentlyapple.com/patently-apple/2013/10/idc-spins-a-tale-of-android-vs-ios-stats-in-smartphones-only.html

Addison, E. R., Wilson, H. D., Marple, G., Handal, A. H., & Krebs, N. (2005). *U.S. Patent No. 6,865,533*. Washington, DC: U.S. Patent and Trademark Office.

Ballagas, R., Borchers, J., Rohs, M., & Sheridan, J. G. (2006). The smart phone: a ubiquitous input device. *IEEE Pervasive Computing,1*(1), 70-77

Bond James M. (2013). Who Manages Cloud IaaS, PaaS, and SaaS Services – The Enterprise Cloud Blog. Retrieved September 5, 2018 from https://mycloudblog7.wordpress.com/2013/06/19/who-manages-cloud-iaas-paas-and-saas-services/

Butler, M. (2011). Android: Changing the mobile landscape. *IEEE Pervasive Computing*, *10*(1), 4-7.

Fetaji, M. (2008, June). Devising a strategy for usability testing of M-Learning applications. *In Proceeding of the EdMedia: World Conference on Educational Media and Technology* (pp. 1393-1398).Vienna, Austria: Association for the Advancement of Computing in Education.

Gonnot, T., Mikuta, M., & Saniie, J. (2017). Image to Audio Frequencies Modulation for Visually Impaired People. *In Proceeding of the IEEE International Conference on Electro Information Technology*.USA-Lincoln: IEEE.

Grasso, A., & Roselli, T. (2005, November). Guidelines for designing and developing contents for mobile learning. *In Proceeding of the Wireless and Mobile Technologies* *in Education. IEEE International Workshop on* (pp. 123-127).Japan-Tokushima: IEEE.

Guo, A., & Bigham, J. P. (2018). Making Everyday Interfaces Accessible: Tactile Overlays by and for Blind People. *IEEE Pervasive Computing*, *17*(2), 66-70.

Haddad, Z., Chen, Y., & Krahe, J. L. (2016, July). Image Processing and Pattern Recognition Tools for the Automatic Image Transcription. *In Proceeding of the International Conference on Computers Helping People with Special Needs* (pp. 197-203). Austria –Linz: ICCHP.

Hakim, H. A. B., Hapsari, P. D., & Putro, A. S. (2017). APPLICATION OF ASSISTIVE TECHNOLOGY IN LIBRARY SERVICE FOR PEOPLE WITH DISABILITIES. *THE FACETS OF KNOWLEDGE, TOWARDS SYNERGY OF MULTIDISCIPLINARY STUDIES*, 127.

Hersh, M., & Johnson, M. A. (2010). *Assistive technology for visually impaired and blind people*. Springer Science & Business Media.

Holzer, A., & Ondrus, J. (2009, April). Trends in mobile application development. *In Proceeding of the International Conference on Mobile Wireless Middleware, Operating Systems, and Applications* (pp. 55-64). Berlin, Germany:Springer.

Jaramillo-Alcázar, A., & Luján-Mora, S. (2017, October). Mobile serious games: An accessibility assessment for people with visual impairments. *In Proceeding of the 5th International Conference on Technological Ecosystems for Enhancing Multiculturality* (p. 66). Spain-Cádiz: ACM.

Jiang, H., Gonnot, T., Yi, W. J., & Saniie, J. (2017). Computer Vision and Text Recognition for Assisting Visually Impaired People using Android Smartphone. *In Proceeding of the IEEE International Conference on Electro Information Technology* (pp. 350-353).Lincoln, USA: IEEE

König-Ries, B. (2009). Challenges in mobile application development. *It-Information Technology Methoden Und Innovative Anwendungen Der Informatik Und Informationstechnik*, *51*(2), 69-71.

Lazareska, L., & Jakimoski, K. (2017). Analysis of the Advantages and Disadvantages of Android and iOS Systems and Converting Applications from Android to iOS Platform and Vice Versa. *American Journal of Software Engineering and Applications*, *6*(5), 116.

Lorecentral. (2017). Advantages and disadvantages of android - LORECENTRAL. Retrieved September 9, 2018 from https://www.lorecentral.org/2017/12/advantages-disadvantages-android.html.

Mallikarjun M K. (2017). Different Types of Mobile Applications - Native, Hybrid and Web Apps | Trigent Software. Retrieved September 5, 2018 from https://blog.trigent.com/different-types-of-mobile-applications-native-hybrid-and-web-apps/

Mennillo, L., Cousty, J., & Najman, L. (2015, May). A comparison of some morphological filters for improving OCR performance. In International Symposium on Mathematical Morphology and Its Applications to Signal and Image Processing (pp. 134-145). Springer, Cham.

Motiwalla, L. F. (2007). Mobile learning: A framework and evaluation. *Computers & education*, *49*(3), 581-596.

Murali Manu. (2016). Windows Phone 10 : Limitations against Android and iOS. Retrieved September 9, 2018 from https://www.techulator.com/resources/15263-Windows-Phone-10-Limitations-against-Android-and-iOS.aspx

Padhya, B., Desai, P., & Pawade, D. (2016). Comparison of Mobile Operating Systems. *International Journal of Innovative Research in Computer and Communication Engineering* *4*(1)15281-15285

Patel, C., Patel, A., & Patel, D. (2012). Optical character recognition by open source OCR tool tesseract: A case study. *International Journal of Computer Applications*, *55*(10).

Pundlik, S., Yi, H., Liu, R., Peli, E., & Luo, G. (2017). Magnifying smartphone screen using google glass for low-vision users. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *25*(1), 52-61.

Riousset Claude. (2013). Towards an Open cloud architecture. Retrieved September 5, 2018 from https://www.slideshare.net/criousset/from-openstack-towards-openpen-stack-to-opencloud

Sandnes, F. E. (2016). What do low-vision users really want from smart glasses? Faces, text and perhaps no glasses at all. *In Proceeding of theInternational Conference on Computers Helping People with Special Needs* (pp. 187-194). Linz, Austria: ICCHP

Seraj, M., & Wong, C. Y. (2012, June). A study of user interface design principles and requirements for developing a mobile learning prototype. *In Proceeding of the* Computer*& Information Science (ICCIS), 2012 International Conference on* (Vol. 2, pp. 1014-1019).Kuala Lumpeu, Malaysia: IEEE.

Sharma. (2016). What are the types of mobile apps? - Quora. Retrieved September 5, 2018 from https://www.quora.com/What-are-the-types-of-mobile-apps

Smith, R. (2007, September). An overview of the Tesseract OCR engine. *In Proceeding of theDocument Analysis and Recognition, 2007. ICDAR 2007. Ninth International Conference on* (Vol. 2, pp. 629-633). Parana, Brazil: IEEE.

Stearns, L., Du, R., Oh, U., Jou, C., Findlater, L., Ross, D. A., & Froehlich, J. E. (2016). Evaluating haptic and auditory directional guidance to assist blind people in reading printed text using finger-mounted cameras. *ACM Transactions on Accessible Computing*, *9*(1), 1.

Szpiro, S. F. A., Hashash, S., Zhao, Y., & Azenkot, S. (2016, October). How people with low vision access computing devices: Understanding challenges and opportunities. In *Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility* (pp. 171-180). Nevada, USA: ACM.

Torres-Carazo, M. I., Rodríguez-Fórtiz, M. J., & Hurtado, M. V. (2016, May). Analysis and review of apps and serious games on mobile devices intended for people with visual impairment. *In Proceeding of the Serious Games and Applications for Health (SeGAH), 2016 IEEE International Conference on* (pp. 1-8). Orlando, USA: IEEE.

Voykinska, V., Azenkot, S., Wu, S., & Leshed, G. (2016, February). How blind people interact with visual content on social networking services. *In Proceedings of the 19th*

ACM Conference on Computer-Supported Cooperative Work & Social Computing (pp. 1584-1595). California, USA:ACM.

Wasserman, A. I. (2010). *Software Engineering Issues for Mobile Application Development*. Retrieved September 7, 2018from http://www-public.imtbs-tsp.eu/~gibson/Teaching/Teaching-ReadingMaterial/Wasserman10.pdf

Whelan. (2018). Voice Interaction Flow - Google Dialogflow. Retrieved September 7, 2018 from http://joeywhelan.blogspot.com/2018/04/

Wong, M. E., & Tan, S. S. (2012). Teaching the benefits of smart phone technology to blind consumers: Exploring the potential of the iPhone. *Journal of Visual Impairment & Blindness, 106*(10), 646.

Zhao, Y., Szpiro, S., Knighten, J., & Azenkot, S. (2016, September). CueSee: exploring visual cues for people with low vision to facilitate a visual search task. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (pp. 73-84). Heidelberg, Germany: ACM.

APPENDICES

APPENDIX 1

OCR IMPLEMENTATION

OCRActivity.kt -> onLayoutElements method, onLayoutChosen method

//When OCR process activity starts, these lines of codes runs.

```
mButtonStartOCR.setOnClickListener { view ->
        val selectedTexts = mImageView.selectedTextIndexes
        val selectedImages = mImageView.selectedImageIndexes
        if (selectedTexts.isNotEmpty() || selectedImages.isNotEmpty()) {
            mAnalytics.sendScreenView("Ocr")
            mImageView.clearAllProgressInfo()

                //Starts OCR for Complex Layout. User selects custom range of the image for
OCR process.
                //startOCRForComplexLayout method goes to OCR.kt
            mOCR.startOCRForComplexLayout(
                this@OCRActivity,
                mOcrLanguage!!,
                it.columns,
                it.images,
                selectedTexts,
                selectedImages
            )
```

```
            mButtonStartOCR.visibility = View.GONE
        } else {
            Toast.makeText(
                    applicationContext,
                    R.string.please_tap_on_column,
                    Toast.LENGTH_LONG
            ).show()
        }
    }
```

OCR.kt startOCRForComplexLayout method Thread implementation as below;

```
  /**
   * native code takes care of both Pixa, do not use them after calling this
   * function
   *
   * @param pixaText   must contain the binary text parts
   * @param pixaImages pixaImages must contain the image parts
   */

fun startOCRForComplexLayout(context: Context, lang: String, pixaText: Pixa, pixaImages: Pixa, selectedTexts: IntArray, selectedImages: IntArray) {
        mExecutorService.execute(Runnable {
            mCrashLogger.logMessage("startOCRForComplexLayout")
            var pixOcr: Pix? = null
            var boxa: Boxa? = null
            try {
                logMemory(context)

                val columnData = mNativeBinding.combinePixa(pixaText.nativePixa, pixaImages.nativePixa, selectedTexts, selectedImages)
                pixaText.recycle()
                pixaImages.recycle()
```

```
val pixOrgPointer = columnData[0]
pixOcr = Pix(columnData[1])
boxa = Boxa(columnData[2])

sendPreview(pixOrgPointer)
ocrProgress.postValue(OcrProgress.Message(R.string.progress_ocr))
if (!initTessApi(
            languages = determineOcrLanguage(lang),
            ocrMode = TessBaseAPI.OEM_TESSERACT_ONLY
    )
) {
    return@Runnable
}

mTess.setPageSegMode(PageSegMode.PSM_SINGLE_BLOCK)
mTess.setImage(pixOcr)

mOriginalHeight = pixOcr.height
mOriginalWidth = pixOcr.width

if (mStopped.get()) {
    return@Runnable
}
var accuracy = 0f
val geometry = IntArray(4)
val hocrText = StringBuilder()
val htmlText = StringBuilder()
for (i in 0 until boxa.count) {
    if (!boxa.getGeometry(i, geometry)) {
        continue
    }
    mTess.setRectangle(geometry[0], geometry[1], geometry[2], geometry[3])
```

```
                hocrText.append(mTess.getHOCRText(0)) // Returns OCR Result.
                htmlText.append(mTess.htmlText)
                accuracy += mTess.meanConfidence().toFloat()
                if (mStopped.get()) {
                    return@Runnable
                }
            }
            val totalAccuracy = Math.round(accuracy / boxa.count)
            ocrProgress.postValue(OcrProgress.Result(pixOrgPointer, htmlText.toString(),
hocrText.toString(), totalAccuracy))
        } finally {
            mNativeBinding.destroy()
            pix.recycle()
            pixOcr?.recycle()
            boxa?.recycle()
            mTess.end()
            mCompleted.set(true)
            mCrashLogger.logMessage("startOCRForComplexLayout finished")
        }
    })
}
```

APPENDIX 2

TTS AND GOOGLE TRANSLATE IMPLEMENTATION

```
//Translator.java shows google translate intent for text to speech and language translation.
/**
 * Triggers translation of a text by the Google Translate app.
 */
class Translator {


    private static final String TRANSLATE_PACKAGE_NAME =
"com.google.android.apps.translate";
    private static final String TRANSLATE_ACTIVITY_NAME =
"com.google.android.apps.translate.TranslateActivity";
    private static final String TRANSLATE_POP_OVER_ACTIVITY_NAME =
"com.google.android.apps.translate.copydrop.CopyDropActivity";
    private static final String EXTRA_TO_LANGUAGE = "to";


    void startTranslation(Activity activity, String text) {
        if (!isGoogleTranslateInstalled(activity)) {
            openPlayStore(activity);
        } else {
            translateWithGoogleTranslate(activity, text);
        }
    }
    private void translateWithGoogleTranslate(Activity activity, String text) {
        if (supportsPopOver(activity)) {
            translateInPopOver(activity, text);
        } else {
            openGoogleTranslateApp(activity, text);
        }
```

```
    }

    private boolean supportsPopOver(Activity activity) {
        final Intent intent = createPopOverIntent();
        return !activity.getPackageManager().queryIntentActivities(intent, 0).isEmpty();
    }
    private Intent createPopOverIntent() {
        return new Intent()
                .setAction("android.intent.action.PROCESS_TEXT")
                .setType("text/plain")
                .setComponent(new ComponentName(TRANSLATE_PACKAGE_NAME,
TRANSLATE_POP_OVER_ACTIVITY_NAME));
    }

    private void translateInPopOver(Activity activity, String text) {
        final Intent popOverIntent = createPopOverIntent();
        Locale current = Locale.getDefault();
        popOverIntent.putExtra("key_text_to_be_translated", text);
        popOverIntent.putExtra(EXTRA_TO_LANGUAGE, current.getLanguage());
        activity.startActivity(popOverIntent);
    }

    private boolean isGoogleTranslateInstalled(Activity activity) {
        PackageManager packageManager = activity.getPackageManager();
        return !packageManager.queryIntentActivities(createGoogleTranslateIntent(),
0).isEmpty();
    }

    private void openPlayStore(Activity activity) {
        try {
            activity.startActivity(new Intent(Intent.ACTION_VIEW, parse("market://details?id="
+ TRANSLATE_PACKAGE_NAME)));
        } catch (android.content.ActivityNotFoundException anfe) {
```

```
            activity.startActivity(new Intent(Intent.ACTION_VIEW,
parse("https://play.google.com/store/apps/details?id=" +
TRANSLATE_PACKAGE_NAME)));
        }
    }

    private void openGoogleTranslateApp(Activity activity, String text) {
        Locale current = Locale.getDefault();
        final Intent intent = createGoogleTranslateIntent().
                putExtra(EXTRA_TO_LANGUAGE, current.getLanguage()).
                putExtra(Intent.EXTRA_TEXT, text);

        activity.startActivity(intent);
    }

    private Intent createGoogleTranslateIntent() {
        return new Intent()
                .setAction(Intent.ACTION_SEND)
                .setComponent(new ComponentName(TRANSLATE_PACKAGE_NAME,
TRANSLATE_ACTIVITY_NAME));
    }

}
```