A DEEP LEARNING BASED MEDICINE LABEL IDENTIFICATION WITH A SMART CAMERA FOR VISUALLY IMPAIRED

BY

SASICHA BANCHONGAKSORN 62011242

SUPSIN KOTCHA 62011255

A PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF BACHELOR OF ENGINEERING IN BIOMEDICAL ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ACADEMIC YEAR 2022

SCHOOL OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

PROJECT CERTIFICATE

Project Title	A deep learning-based medicine label identification with a smart camera for visually impaired
Student Name	Miss Sasicha Banchongaksorn Student ID. 62011242 Mr. Supsin Kotcha Student ID. 62011255
Degree	Bachelor of Engineering in Biomedical Engineering
Project Advisor	Signed: _____ (Assoc. Prof. Dr.Wibool Piyawattanametha)
Committee	Signed: _____ (Assoc. Prof. Dr. Chuchart Pintavirooj)
Committee	Signed: _____ (Asst. Prof. Dr. Treesukon Treebupachatsakul)
Committee	Signed: _____ (Asst. Prof. Dr. Kasama Srirussamee)
Head of Department	Signed: _____ (Asst. Prof. Dr. Sarinporn Visitsattapongse)

Project Title	A deep learning-based medicine label identification with a smart camera for visually impaired
Student Name	Miss Sasicha Banchongaksorn Mr. Supsin Kotcha
Degree	Bachelor of Engineering in Biomedical Engineering
Project Advisor	Asst. Prof. Dr.Wibool Piyawattanametha, Ph.D.
Academic Years	2022

ABSTRACT

In this research, we developed an iOS application that provides information about drugs with sounds. Our drug label samples are derived from actual drug labels from the Hospital. Our iOS application consists of three main parts, i.e., text recognition module, central database of drug instruction, and text-to-speech module. Figure out the efficiency of detecting text in the Thai language by using Google cloud vision API. Our software that uses Google cloud vision API achieved 100% accuracy, ready from 30 samples with an average processing time for each drug labeling of around 0.56 s per character and a processing time for the play sound of each character of approximately 0.25 s. We were testing our iOS application performance in detecting texts from a photo of drug labels photos which had been taken from various forms. Those commercial drug containers are in various forms packages such as a bottle, a tube, and a pack. An added test was tested with a drug in envelope packaging form. The results showed our iOS application could supply the correct information. Also, we developed a central database to provide other drug information necessary for users. Our database has already contained over 300 drugs sample and their information. This system uses MongoDB as its central platform to classify the data into each characteristic. Due to its unstructured nature, MongoDB has supplied the flexibility to connect the data to users' devices. As of now, we are still developing it into a database server using Visual Studio Code and connecting to Docker and Cloudflare Tunnel.

Keywords: Text to speech, Text recognition, visually impaired, Medicine label, Thai language, Database

ACKNOWLEDGEMENTS

The following thesis has greatly benefited from the invaluable insights and direction provided by several individuals, to whom we express our sincere gratitude. Firstly, This project was achieved with favor from Asst. Prof. Dr.Wibool Piyawattanametha, our affectionate project advisor, for the suggestions and the guidelines to carry out the project.

Furthermore, the project organizers are grateful for the invaluable assistance provided by the members of Dr. Wibool's lab, who helping us oversee and correct the flaws that occurred during the operation throughout the project and educating the machine learning algorithm with more coding part to do this project.

Not least of all, we would like to be thankful to every other person who is not mentioned here for their support, whether for the idea suggestions or their subservience throughout the project period.

Lastly, the project organizers would like to thank the Biomedical Engineering Department, the School of Engineering at King Mongkut's Institute of Technology Ladkrabang and along with National Research Council of Thailand for supporting us to develop and carry out this project.

Sasicha Banchongaksorn

Supsin Kotcha

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

This chapter begins with the background and significance of the research. Subsequently, the research aims are described, followed by the project's hypothesis and scope.

1.1 Background and Significance of the Research

In this section, we explain some of the related works which solve problems in reading text for visual impairment, one of the earlier works was presented by Mauro Avila. They surveyed the Be My Eyes application. This application used volunteers to aid the blind or visually impaired in reading a text in their daily life. However, many users of this application feel insecure and uncomfortable when they must talk to strangers who work as a volunteer in this application because of privacy concerns while they are showing personal information to strangers. But most users still use it when they want to check the end date of food, which is extremely useful to them and means this kind of application or tool idea is particularly useful. It is wanted for this target group of users [3].

There has been considerable research from earlier work trying to solve the privacy concern problem. One of them is using OCR technology to detect text and synthesize sound by the text-to-speech process for the user instead of using another person to read it. Terri Hedgpeth presented this solution. This research developed a device to detect text using a camera and process it on a PDA (personal data assistant-sized device). The biggest challenge of this research is the limitation of camera efficiency and the size of hardware, which is hard to develop a portable device and the design process for navigating text on the book for interacting with users [4], [5].

Similarly, research by Celine Mancas-Thillou develops devices for reading text for the blind and visually impaired. It uses raspberry pi 2 as a processor instead of a PDA and uses a digital camera connected to raspberry pi [6]. This research focuses on developing processes to improve image quality before sending it to OCR processing rather than developing the application OCR feature.

The earlier research example has shown that portable devices have faced the same problem in image resolution problems in each research which occur from the way the user takes

images and the quality of the external camera or cable camera connected to the processor. However, improving the efficiency of cameras that attach to processor devices like mobile phones makes reading assistant devices easier to carry and improves image quality.

In the new era of reading assistant devices developed on smartphones, the research [7] from Huiying Shen and James M. Coughlan focused on developing a system to detect and read aloud text from video images generated on Android mobile phones. This system can help read text signs for blind and visually impaired users. However, it still needs to improve the text detection algorithm to make the system faster and reduce errors in missing some words while processing that can cause wrong reading results to users. Some limitation of this research is that users must hold their mobile phone still during the text recognition process until it can produce a sound; otherwise, the system will read incorrect words. The long duration of the text recognition process can increase human error rates.

To reduce this error, we must primarily increase the speed and accuracy of text detection algorithms and then create fewer false positive readings. Developing systems that focus on reading specific words or information by adding the guessing words processed into the earlier system can reduce errors in false positive reading like this research [8]. The research about “A system helping the blind to get merchandise information” is a good method example to improve reading assistant devices for blind people by developing a guessing word part in the system to find the position of the expired date on the label. It was evaluated by training 193 images as a positive dataset and 32,741 images as a negative dataset to support vector machines (SVM). So, the system can navigate the exact position of the desired text line, reducing the time spent reading and processing other text information in the image.

To help users easily control and communicate with a reading text assistant system by sending feedback to the user by sensing vibrating and sound. In the following research, they developed a process to improve the user experience while using this app by giving sound feedback to users to communicate with them [9]. This application designed an expressible button every time the user touched the button. It will give the information of the button back to the user in sound. So, users will know the location and function of each button on the screen whenever they want to use that function. They can do it by double tapping on the button. Another essential thing in this research that can improve the user experience while using this application is they design applications to stay awake all the time and block or stop applications with a lock screen button.

After, we can improve user experience by the earlier method by the communication tool to communicate with the user to get the correct command from the user and continue the next step, like taking the high-resolution image and then send to text recognition image processing. Nowadays, the most popular OCR engines, such as Microsoft Seeing AI and Google mobile vision API, are commercial OCR engines with high efficiency in text recognition processing with various language choices [10]. But on the other hand, Google Cloud is a free OCR engine with less efficiency in some languages. So, developers who want to use it must add more processes than Microsoft Seeing AI and Google mobile vision API to improve its text recognition accuracy.

1.2 Research Objectives

1.2.1. To develop an assistive application that can help to read drug labels for the visually impaired.

1.2.2. To develop a suitable word correction that supplied the drug information as both text and voice in Thai.

1.2.3. To perform the necessary medicinal information, including precaution and duo-interaction.

1.3 Research Hypothesis

Since visually impaired people have limited access to medication and style and format of a printed label can lead them to misuse drugs. Our assistive application can improve visually impaired life more efficiently with the help of reading on drug labels and can increase their chances of accessing printed information.

1.4 Research Scope

1.4.1. Google autoML and the Roboflow model were used in this study as optical text characterization framework platforms.

1.4.2. The machine learning model for optical text characterization used in this study was the Yolo V7 model.

1.4.3. The machine learning algorithm for text classification used in this study was a Support Vector Machine with a linear kernel function focused on individual words.

1.4.4. The data of the project's central database used in this study was compiled from DrugBank and MedThai websites.

1.4.5. The ratio of data used for training and testing in this study was 7:3.

CHAPTER 2
REVIEW OF A DEEP LEARNING BASED MEDICINE LABEL IDENTIFICATION

2.1 Optical character recognition (OCR)

Optical character recognition is a system that converts text from images or printed text into machine-encoded text from an image. The OCR system will digitize printed texts and supply output that can be stored, displayed, edited, and used in machine processes. The system can produce high recognition accuracy by being trained with various images. The method that is used in OCR is as follows:

2.1.1. Pre-processing

2.1.1.1) De-skew sets the document image's orientation perpendicular in horizontal and vertical directions by adjusting it clockwise or anti-clockwise. The rotation direction depends on the document's orientation before the image is adjusted.

2.1.1.2) De-speckle will eliminate the spots in both positive and negative spots inside the image to smoothen it.

2.1.1.3) Binarization is the method to adjust the image color to be grayscale which will set the image to be black and white. The advantage of black and white images is to easily separate the text inside the image from the rest of the background.

2.1.1.4) Line removal removes non-glyph symbols within the sentence and group of words.

2.1.1.5) Layout analysis or Zoning is the method that uses columns and text boxes to classify paragraphs, groups of words, and sentences inside the image.

2.1.1.6) Line and word detection is used when word recognizing and sentence splitting paragraph of the distorted image. This technique will help to detect and identify the text inside the image, as shown in Figure 1.

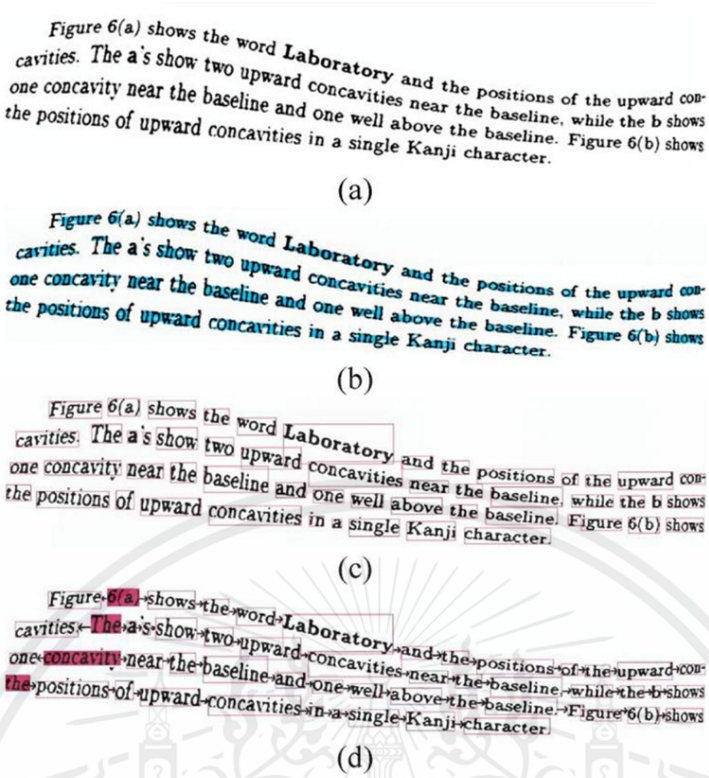


Figure 1: shows how optical character recognition works using text line detection. [11]

2.1.1.7) Script recognition is used to recognize a type of language as an article image can consist of several types of language due to each language having a different arrangement.

2.1.1.8) Character isolation (segmentation) is used to separate letters from two joined words due to the distortion effect of the image. And when the letters of two words are separated, the missing letters of each word are added to complete the meaningful word.

2.1.1.9) Normalizing aspect ratio and scale are the methods to adjust the text size with different font widths to have similar proportions through scaling, as shown in Figure 2.

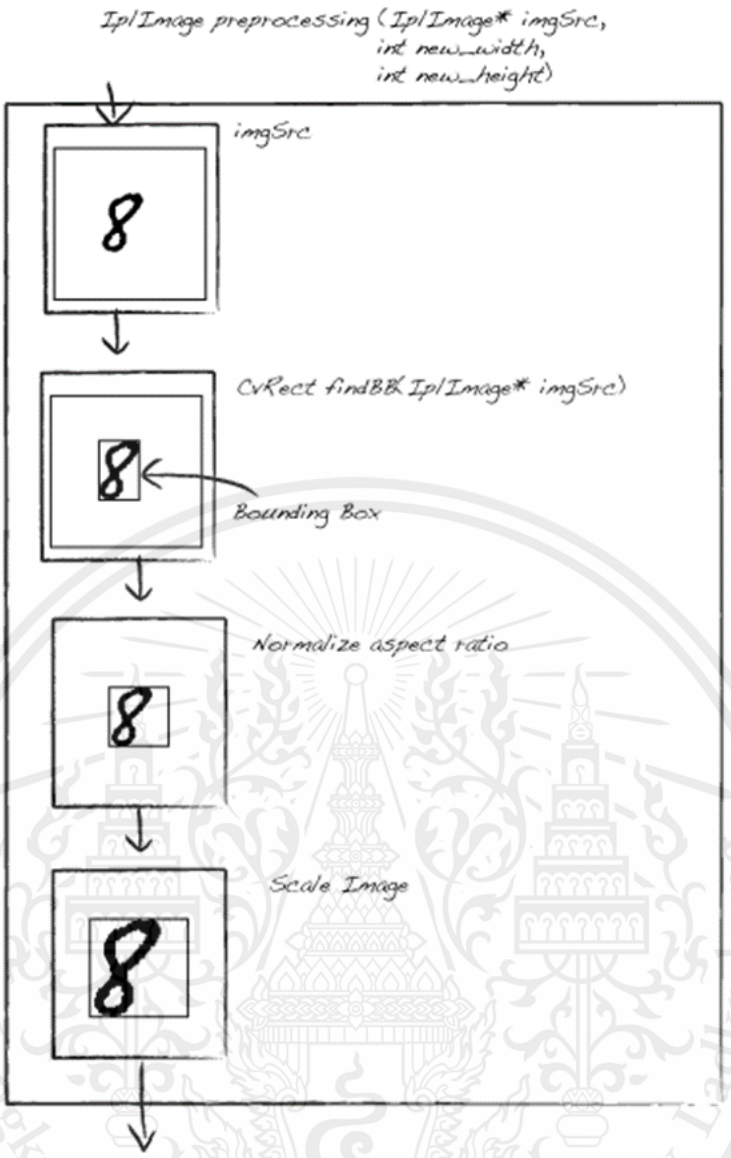


Figure 2: The schematic of image text processing. [12]

2.1.2. Text recognition

There are two main methods used in optical character recognition, which are divided by their capability in text recognition in different types:

2.1.2.1) Pattern recognition or image correlation is the optical text recognition system from the image by comparing text characteristics in an article with a pixel. By comparing font characteristics to the database, the system can detect the similarity of each font pixel and size of an article with the database. The disadvantage of this system is it cannot detect handwritten text.

2.1.2.2) Nearest neighbor classifiers are similar to Pattern recognition. The difference between them is this system compares the font characteristics with the database by using the loop, line direction of each font, space between the lines, whitespace, etc. As such, this system will compare the vector of the font with the database. The system will predict the font using the closest similarity of vector

characteristics from the database. As such, it can recognize both printed and handwritten text.

2.1.2.3) Neural networks are systems trained to memorize the line pattern from the message rather than letter by letter, in which it can recognize both printed and handwritten text.

2.1.3. Post processing

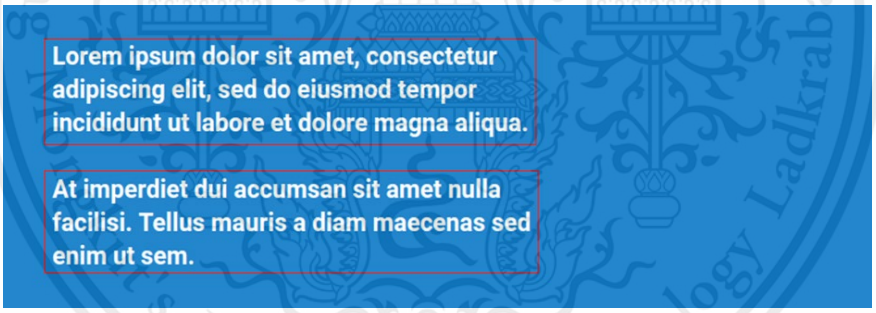
The final method is to use the word in the dictionary to compare with text from the text recognition process, resulting in a precise OCR system.

2.2 Firebase ML Kit

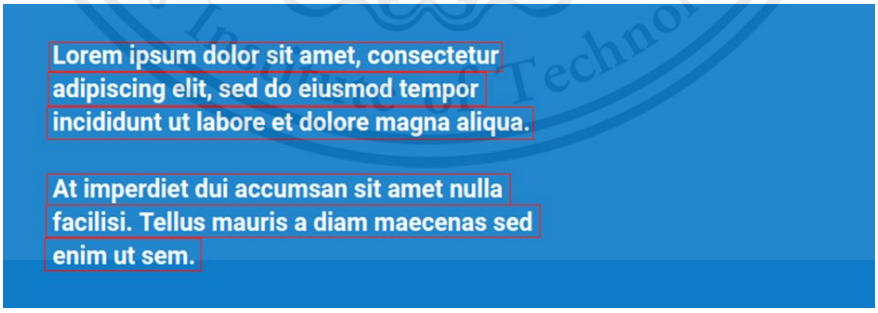
Firebase ML Kit recognition is one of the systems for detecting English characters from an image using the Google Machine Learning application on Android or IOS. The Main features include Text recognition, Image labeling, Barcode scanning, Face detection, and Landmark recognition. All features can be used both on the device and Cloud. Depending on the type of use and accuracy that the user wants.

The operating principle of Firebase ML Kit is that the Text recognizer detects a passage by taking information for evaluation in the form of a Bitmap, a Media Image, a Byte Buffer, and a Byte file. This information will be sent to be evaluated by the Firebase vision image then the ML kit tool will be employed for the last evaluation process before displaying the result, as shown in Figure 3.

a.



b.



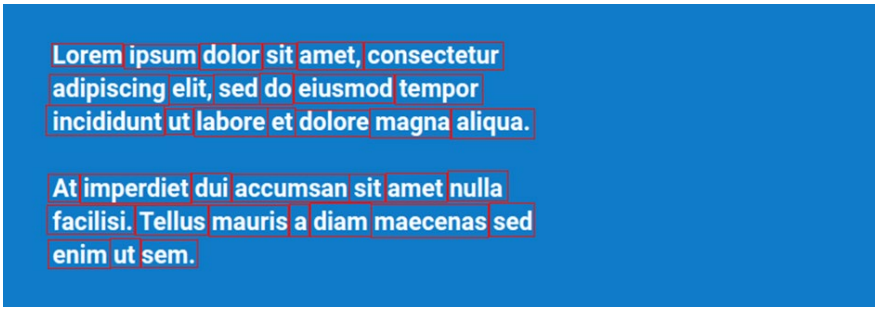
c. 

Figure 3: shows how the text can be characterized in different forms | a. Text characterizes in form of text box| b. Text characterizes in form of sentence| c. Text characterizes in form of word [13]

A convolution neural network comprises a filter layer that selects various images. Due to the different images used to train the filter in each layer, the filters' properties will also be opposed. Therefore, the algorithm will send the result after passing the last filter to process the later filter continuously, as shown in Figure 4.

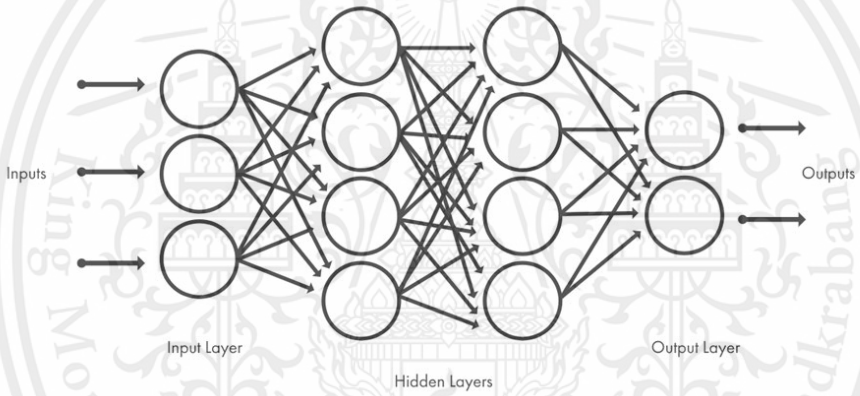


Figure 4: shows how data transmission process in neural network [14]

Examples of 3 types of filter layers that are commonly used:

1. The convolution layer is a layer that uses a convolution filter to detect and recognize the characteristic of each image through a matrix system. Which filter will be a matrix that slowly moves across the image matrix, as shown in Figure 5.

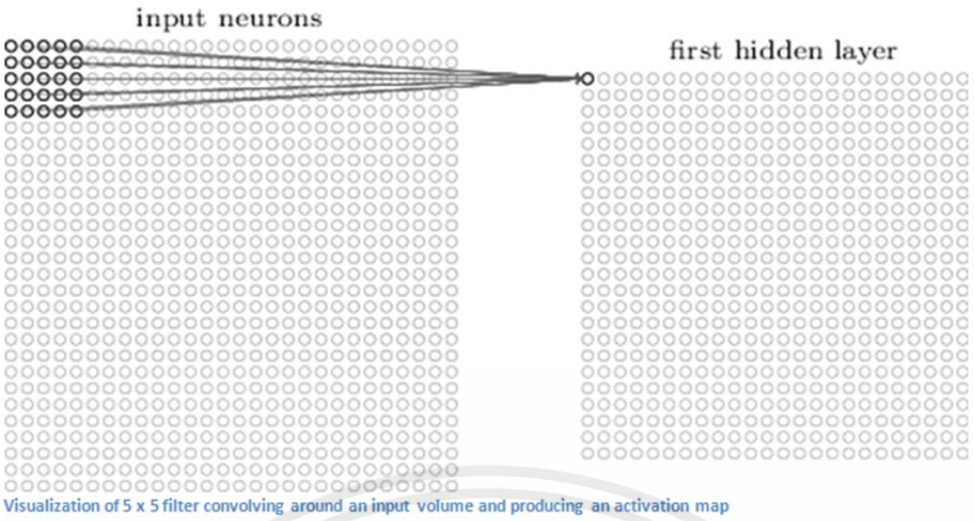


Figure 5: shows how the filter process in the convolution layer [15]

2. The rectified linear unit (ReLU) layer is a layer that is like a tool that helps the system know the non-linear text from incoming images, which has mapping negative and positive values. This mapping will set the negative value in the image to 0 while the positive value will be the same. These values will be sent to the next layer.
3. Pooling is the layer used for flattening the text size and amount through a non-linear down-sampling process, as shown in Figure 6.

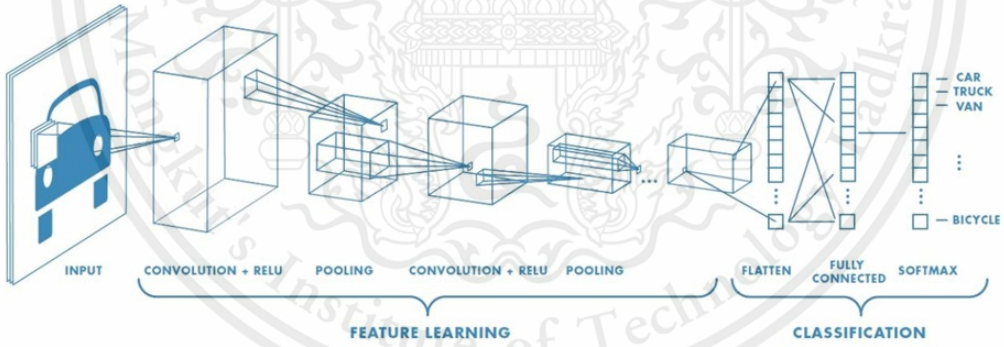


Figure 6: shows the structure of the network comprised of each layer of a different filter in which different layers will train data with different resolutions, and its result will be sent to process the later filter [16]

The principle of Firebase ML Kit Text recognizer is detecting the text that starts from getting information that comes into processing in the form of bitmap media. Image, ByteBuffer, byte []. This information will be sent to process with Firebase Vision Image. After that, the algorithm will use the ML kit tool for the last processing before the text display, as shown in Figure 7.

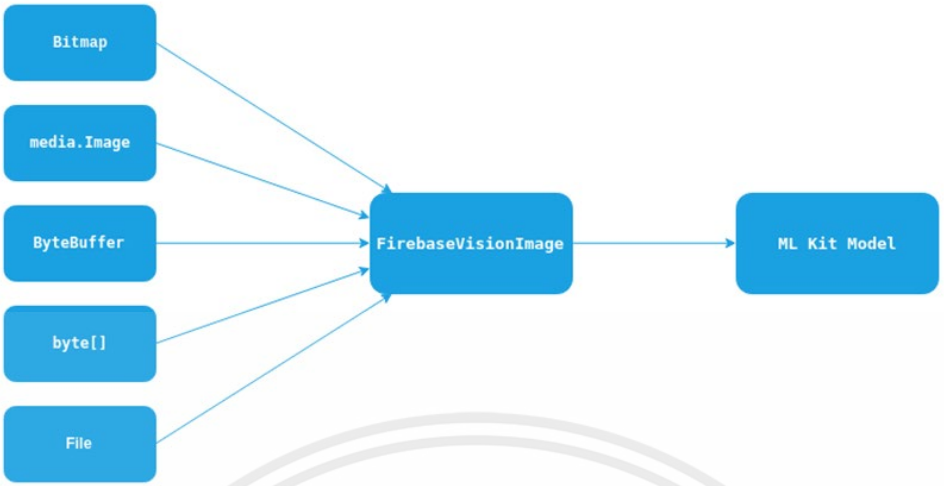


Figure 7: The schematic of Firebase Vision lmage. [17]

2.3 Text-to-speech (TTS)

Text-to-speech (TTS) is the technology to convert text-to-speech messages. The working principle of text-to-speech is divided into two parts.

2.3.1 Front-end NLP part

The first part is Front-end (NLP part), which converts raw text into written-out words and converts each word into phonetic words. This front-end process consists of techniques that include text processing helps to convert specific phonetic and prosodic messages into words into text. Phonetic analysis is the process of analyzing the sound characteristics of each term, and prosodic analysis is used to analyze the tone of voice and the difficulty in pronunciation of each word.

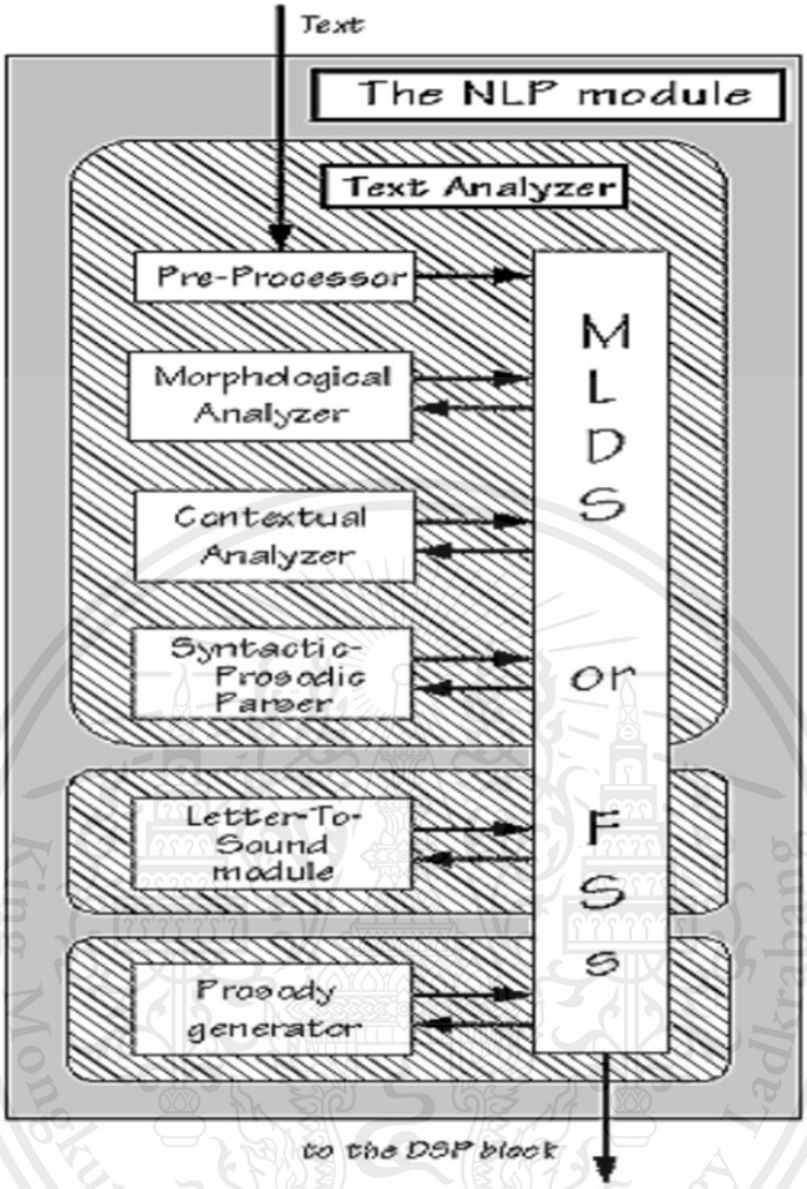


Figure 8: The schematic of Natural language processing of Text-to-speech. [18]

2.3.1.1. Text processing helps to convert specific phonetic and prosodic text messages. The text processing consists of

1) Document structure checks for spaces. Text abbreviation inserting a symbol including various marks.

2) Text normalization is an abbreviation for checking the symbols compared to the readings. For example, Miss. is Miss, St. Is Street, the school is school, etc.

3) Text mark-up interpretation This is a step that helps to check the font style. Shorten to match the correct reading method. Also, check the context of the message to match emotions to get the proper reading tone.

4) Linguistic analysis considers sentence form, emphasis, sentence spacing in reading, and reading speed. The word has different positions in the sentence and can be pronounced differently.

2.3.1.2 Phonetic analysis is the process of analyzing the pronunciation of each word, including

1) Morphological analysis helps the determination of prefixes and suffixes to where the word is derived from.

2) Homograph disambiguation considers the pronunciation of a single word in a sentence.

3) Grapheme-to-phoneme (G2P) conversion uses letter-to-sound (LTS) word analysis to verify language roots. In Thai, there are four lines to form. Thai language must have consonants, vowels, and tones. Some words require a dictionary to analyze how they read.

2.3.1.3. Prosodic analysis is used to analyze the tone of voice. The pronunciation of each word controls its tone by adjusting the amplitude, during, and pitch of the sound. The models used in this process are as follows:

1) Klatt's duration model assumes that each phonetic period has its specific duration.

2) Pierre Humbert's intonation model divides tones into high (H) and low (L).

3) Tilt model has four different tone divisions with Pitch accents, boundary tones, connections, and silences.

4) Fujisaki's intonation model is a model that considers the contour of a log graph consisting of two phases, Phrase command and Accent commands, as shown in Figure 5.

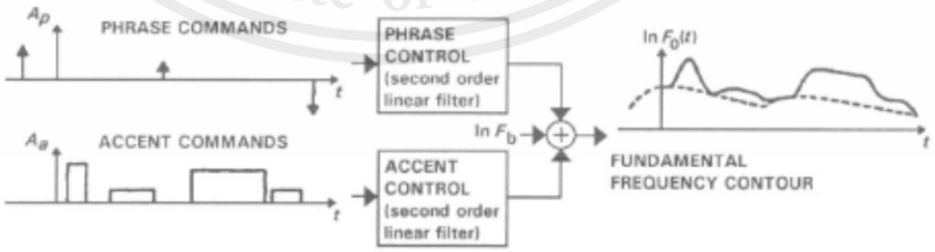


Figure 9: The schematic of Fujisaki's intonation model. [19]

2.3.2 Back-end DSP part

The other part is Back-end (DSP), which converts symbolic linguistics to sound based on the working principle. In this back-end process, the techniques, including formant synthesis, are analyzing sound patterns, analyzing physical characteristics of speech and word pronunciation,

synthesizing sounds from the combination of short samples for the desired sound, and synthesizing sounds to make it more like human speech.

2.3.2.1 Concatenative synthesis is a technique for synthesizing sounds by combining short-duration samples into the desired sound. The size of the sound subunits used to be connected has a duration of 10 milliseconds to 1 second. Create a new sound [12]

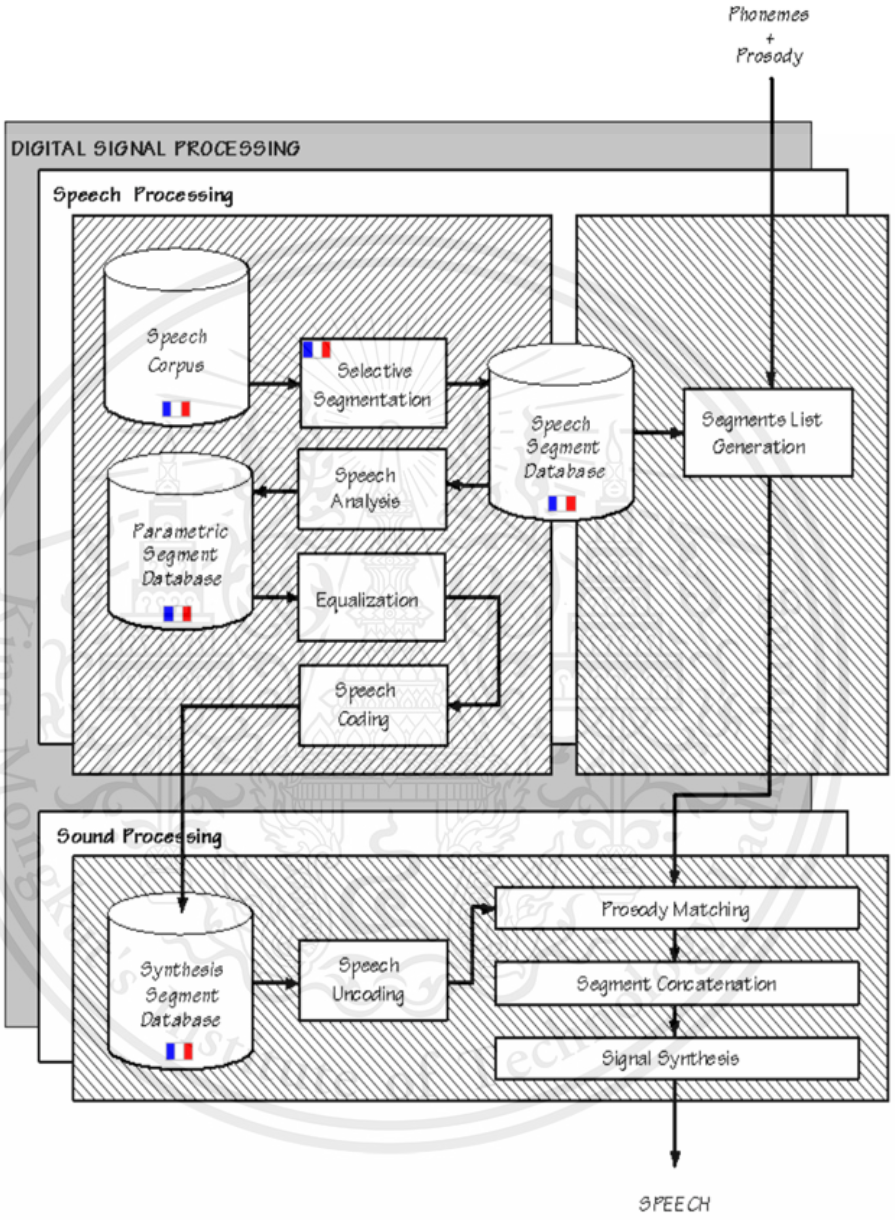


Figure 10: The schematic of DSP in type of Concatenative synthesis [20]

2.3.2.2 HMM-based synthesis allows for the synthesis of sounds that are close to speech.

2.3.2.3 Formant synthesis is to analyze the patterns of sounds as humans speak. Each word is unique in terms of pitch and tone. Each person has a different sound pattern for each speech. At this stage, it will help you analyze the sound pattern. Then, the sound patterns of each word are used to create a model for adjusting words, such as the Bandpass filter (BPF) and Voltage control amplifier (VCA). Tune in through these filters. Subsequently, the terms that have gone through the two filters are

combined to produce a synthetic sound as close to the natural human sound as possible, as shown in Figure 11.

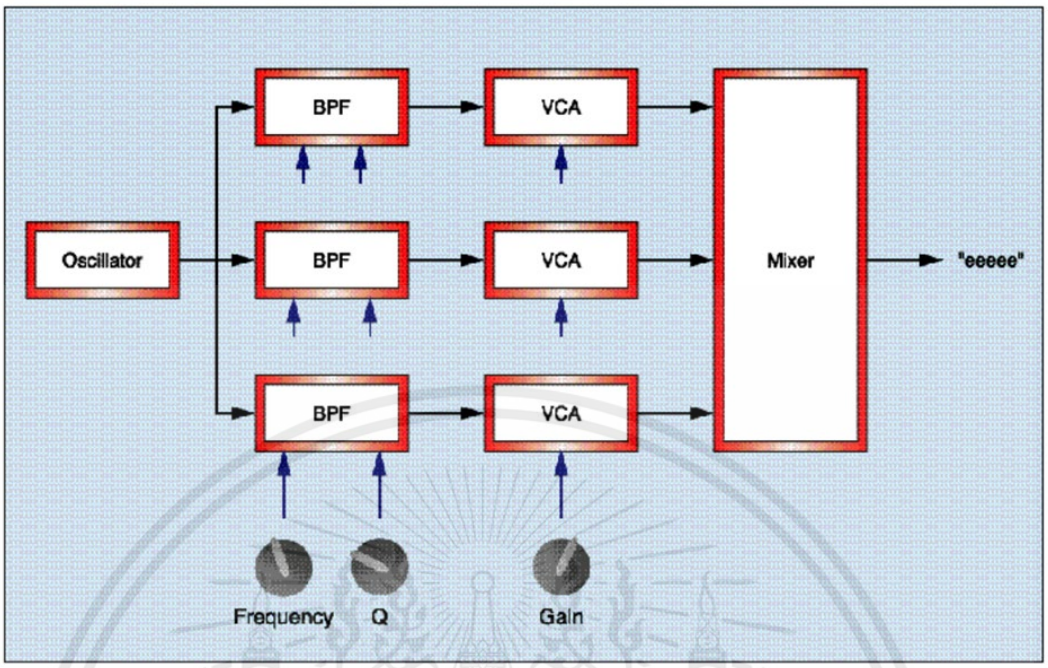


Figure 11: The schematic of Fujisaki's intonation model. [21]

2.3.2.4 Articulatory synthesis is analyzing the physical characteristics of speech and the pronunciation of a word. They investigate the position of the jaw, tongue, and lips. These are used to create a 3D simulation model during speaking, pronunciation, and airflow while speaking.

2.4 Natural language processing (NLP)

Natural language processing is an artificial intelligence tool that allows computers to understand and interpret human language and deal with human messages. To develop natural language processing is necessary to rely on computational linguistics and computer science knowledge. Combined the workflow within the Natural language processing structure consists of

2.4.1. Content categorization is the procedure for separating the category of documents by content, and the desired item checks the words within the document against the words specified in the index.

2.4.2. Topic discovery and modeling is the process of interpreting the document's contents for extraction. Documents according to the nature of the title or subject matter of the document of the desired messages communicates

2.4.3. Contextual extraction analyzes six structures for writing text in a document.

2.4.4. Sentiment analysis helps analyze the tones of the article and intent feeling of the writer in the article.

2.4.5. Speech-to-text and text-to-speech is the process of converting sounds into characters and vice versa text-to-speech that is to convert text into voice

2.4.6. Document summarization: It is the process of summarizing the article's main points.

2.4.7. Machine translation is a tool for translating text from one language to another.

2.4.8. Using NLP after the ORC process improves the OCR data to be more accurate because OCR can show error results if the image has smudged cover text or is deformed. There is an NLP model that is often used to improve OCR performance which is:

2.4.8.1. BERT (Bidirectional Encoder Representations from Transformers) is an NLP model used to predict words, text, and sentences. The BERT model will work if a tool recognizes incorrect words/sentences or text that does not match the database's terms and replaces space with those words with [MASK]. After that, the BERT model will take the letter in [MASK] to compare it with other text from the OXR process. Moreover, the BERT model can help in separating the sentence with vectors such as "Tom went to the store. He bought two gallons of milk." and BERT model will convert it to [0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1].

To improve the text performance from OCR with the BERT model, this can be done using SpellChecker to predict the missing word in [MASK]. [22]

2.4.9. The SQL database application is used to update the information obtained from OCR, which is left with more specific and detailed information or provided additional information from the central database [23].

2.5 Digital signal processing (DSP)

This is the processing of digital data by computers or specialized digital signal processors, where digital signal processing processes continuous data in domains such as time, space, and frequency, where the data may be in non-linear form, and this type of processing commonly used for applying to audio, speech, sonar, radar video coding, and digital image, etc.

2.6 React Native and Expo Frameworks

2.6.1 React Native

React Native is a framework used to develop native apps for Android applications, and iOS application development with React uses JavaScript as a coding language so that the API platform is compatible. React works with native platforms like Xcode and Android Studio, creating a user interface (User Interface, UI) of native Xcode and Android. The studio uses different languages in the development of the user interface of the application. React Native can use JavaScript for running React libraries. Applications on both iOS and Android platforms will have the same format as shown in Figure 12.

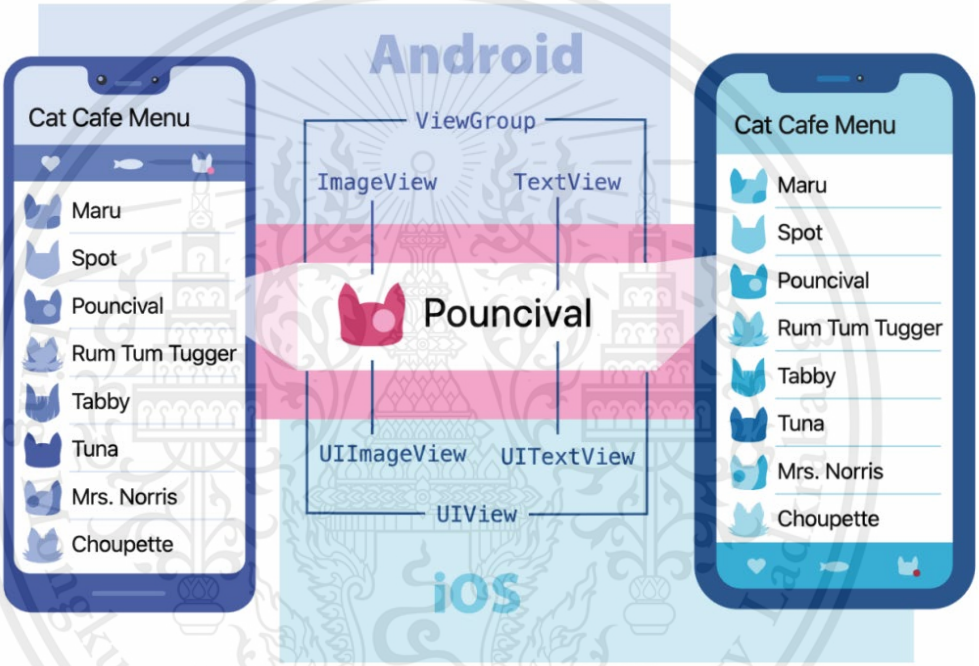


Figure 12 shows how the interface works with users on iOS and Android platforms [24].

2.6.2 Expo

Expo is a framework and platform for React applications and services to support the functionality of the React Native framework to provide users with this framework can develop applications easily and quickly. The advantage of an expo is that it can test a website or program written by scanning a QR code on your mobile phone to get started testing, as shown in Figure 13.

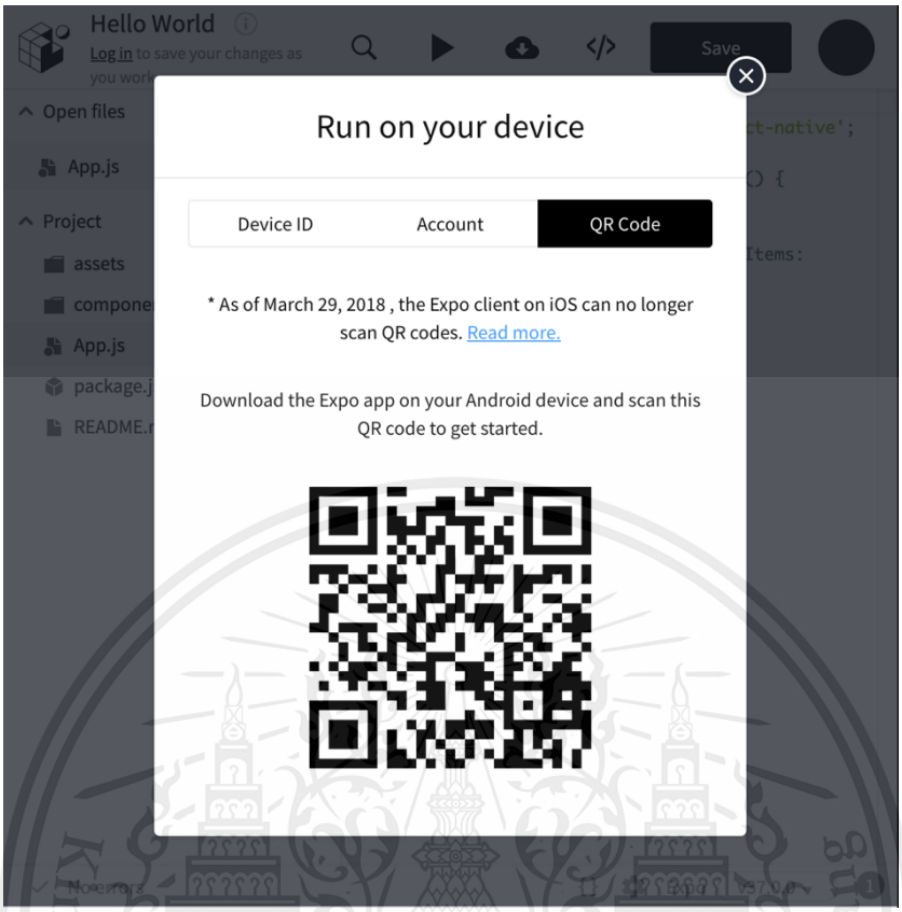


Figure 13 shows the start-up test by scanning the QR code [25].

Libraries related to this research

2.6.1. React-native-firebase/ml-vision It is a library used for OCR processes. Detecting characters from an image. This library is for on-device text recognition and cloud text recognition [26]

2.6.2. Camera from expo-camera

2.6.3. Ionicons from expo/vector-icons

2.6.4. ImageManipulator from expo-image-manipulator

2.6.5. Audio from expo-av

2.6.6. activateKeepAwake from expo-keep-awake

2.6.7. StyleSheet, TouchableOpacity, TouchableHighlight, Modal, Alert, View from react-native

2.6.8. Wave file from wave file

2.6.9. Axios from axios

2.6.10. FileSystem from expo-filesystem

2.6.11. Cio from react-native-cheerio

2.7 Support Vector Machine (SVM)

Support Vector Machine (SVM) is a set of supervised machine learning algorithms for outliers' detection, classification, and regression purposes. This algorithm finds the optimal hyperplane, the decision boundary that differentiates the two classes in SVM to classify between data points. The most suitable hyperplane for SVM is the plane with a maximum margin or distance between data points of both classes, as shown in Figure 14.

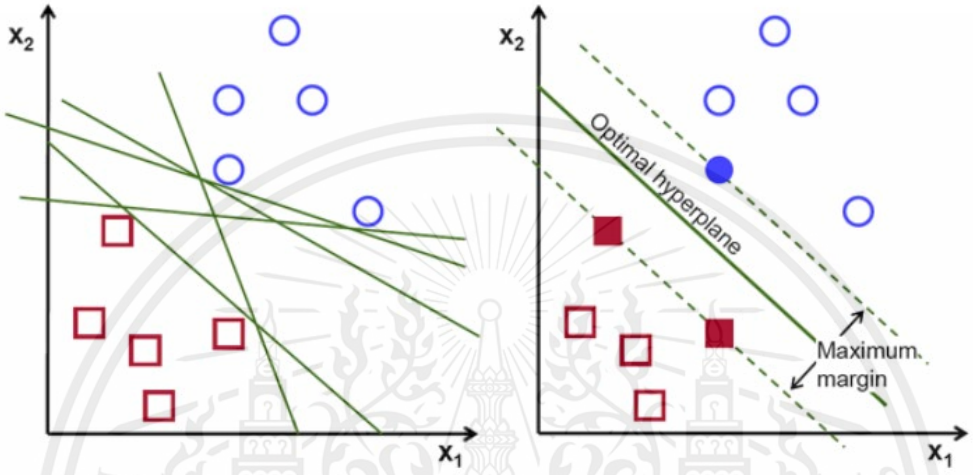


Figure 14: Figure of possible hyperplanes. [26]

Also, hyperplanes can have various dimensions that depend on the number of features. For example, the hyperplane will become a line or two-dimensional plane if the number of input features is 2 and 3, respectively. The position and orientation of the hyperplane can be influenced by data points, also known as support vectors, as shown in Figure 15. By using support vectors, we can change the position of the hyperplane and maximize the margin of classifiers.

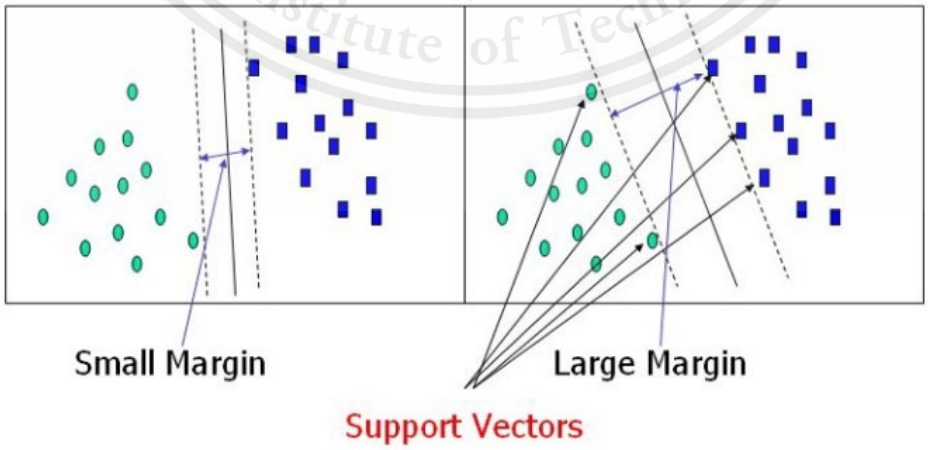


Figure 15: Figure of support vectors. [26]

A set of mathematical functions is used in the SVM, defined as the kernel. The function of the kernel is to map non-linear separable data into a high dimensional space by taking data as input and transforming it into the required form. As such, the data can be linearly separable, as shown in Figure 16. There are many types of kernel functions; some examples are linear, non-linear, polynomial, radial basis function (RBF), and sigmoid. Moreover, the influence of the kernel can be specified by using an additional parameter called gamma.

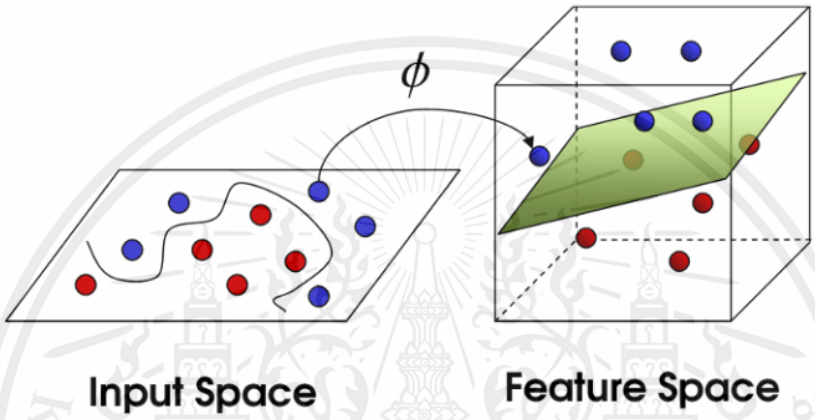


Figure 16: Figure of Kernel function. [27]

It is usually suggested to use either linear or RBF kernels. While RBF kernels have localized and finite responses along the entire x-axis, linear kernels should be used when the number of features is larger than observations in the dataset.

2.8 MongoDB

MongoDB is a cross-platform document-oriented No-SQL database program. As a No-SQL database, MongoDB has a unique mechanism for storage and retrieval, which helps in storing vast volumes of data and retrieving within a fraction of a second, as shown in Figure 18. Furthermore, MongoDB is often perfected for simple retrieval and appending operations in exchange for higher performance. Due to the above reasons, MongoDB supplies high performance, wide availability, and scalability. This is useful for web applications having a large volume of read-only requests. In addition, the data can be distributed and load-balanced easily compared to other databases.

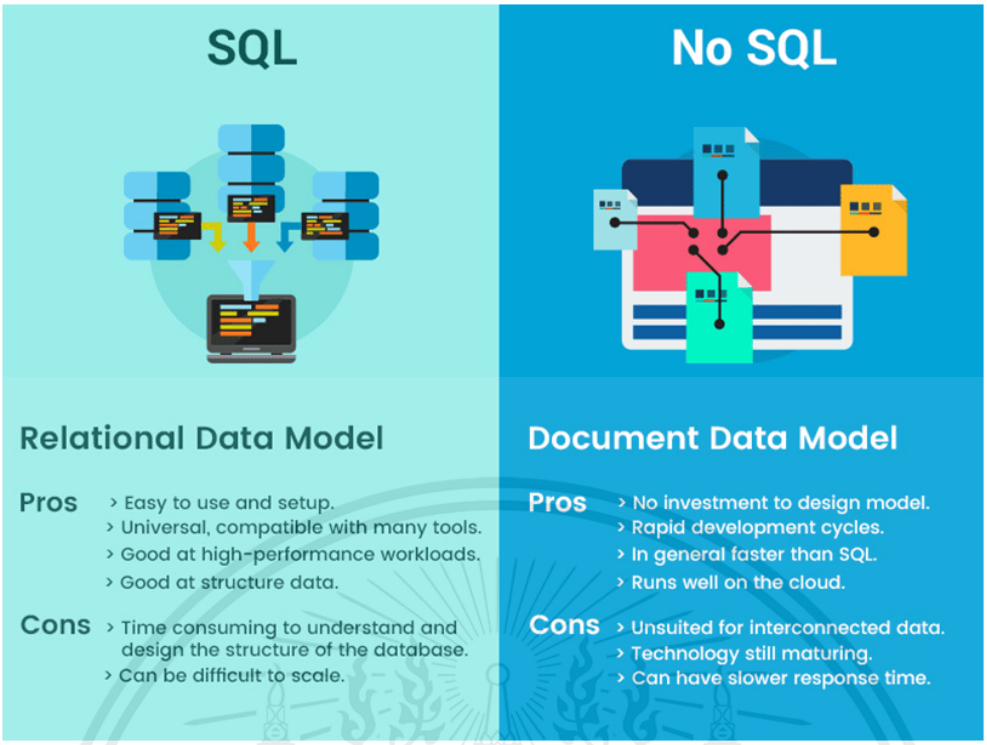


Figure 17: Comparison between SQL and No-SQL model. [28]

2.9 DrugBank

DrugBank is an online database website that supplies information on drugs and drug targets. This website combines detailed drug data with comprehensive drug targets provided and kept by the University of Alberta and Metabolomics Innovation Centre. The advantage of DrugBank is that the website holds over two-thousand FDA-approved drugs and six-thousand experimental drugs. In addition, the drug information that DrugBank supplied has detailed bioinformatics and cheminformatics that our project desired. Moreover, DrugBank has been updating drug information continuously.

As such, most of the project's database data will be used from DrugBank by taking drug samples of each setting category, translating and putting essential data into a central database, and supplying extra information to users.

2.10 Home remedies

Home remedies (First Aid Kits) is a drug that the Ministry of Public Health has prescribed as a drug that should be at home to treat, relieve or prevent injuries or illnesses such as headache, stomachache, diarrhea, constipation, vomiting, runny nose, motion sickness, seasickness, etc., with the primary drug and can be used to treat the initial symptoms. The research team is used in this research with the following names and details.

2.10.1 Relieve stomach pain, flatulence, flatulence, indigestion

2.10.1.1. Alumina-magnesia antacid tablets to relieve heartburn symptoms. Reduce stomach acid, flatulence, flatulence, distension, indigestion, and acid reflux. Take half an hour before meals or 1 hour after meals.

2.10.1.2. Liquid alumina-magnesia antacids help relieve heartburn symptoms and reduce stomach acid, flatulence, flatulence, distension, indigestion, and acid reflux. Take half an hour before or 1 hour after meals and shake the bottle each time before using the drug.

2.10.1.3. Sodamint tablets, for flatulence and indigestion, help reduce stomach acid. Relieve heartburn, heartburn, and indigestion, and eat 1 hour after meals or when symptoms occur.

2.10.1.4. Liquid carminative helps relieve flatulence and indigestion by expelling gas in the stomach. Take 3-4 times daily and shake the bottle each time before using medication.

2.10.1.5. Stomachic mixture helps relieve flatulence and indigestion and expels gas in the stomach. Take three times a day before meals and shake the bottle each time before using the drug.

2.10.1.6. Asafetida tincture, medicine for flatulence and indigestion, helps relieve flatulence and indigestion in children.

2.10.1.7. Diarrhea medication

2.10.2 Laxatives

2.10.2.1. Glycerin laxative rectal suppositories help relieve constipation and stimulate excretion. It can be used by both children and adults

2.10.2.2. Magnesia liquid laxative helps relieve constipation and stimulate excretion. Use it before going to bed or after waking up in the morning.

2.10.2.3. Senna laxative, tablet type, helps relieve constipation and stimulate excretion. Use it before going to bed or after waking up in the morning.

2.10.2.4. Sodium chloride laxative, rectal type, helps relieve constipation. This will stimulate excretion by enema and holding back until one cannot bear it and goes to the bathroom.

2.10.3 Deworming drugs

2.10.3.1. Roundworm anthelmintic, containing Bendazole, can help deworm many types of parasites, such as threadworms, pinworms, hookworms, roundworms, and whipworms.

2.10.4 Pain relievers, fever reducers

2.10.4.1. Aspirin relieves pain and reduces fever by taking it immediately after meals or every 4-6 hours and drinking plenty of water when symptoms are present. Please consult your doctor before using the drug in children under 18 as it may cause side effects such as Reye's Syndrome. This can lead to liver and brain swelling, vomiting, fatigue, seizures, and loss of consciousness.

2.10.4.2. Paracetamol relieves pain and reduces fever by taking it every 4-6 hours when symptoms are present.

2.10.4.3. Pain relief plaster helps relieve muscle pain by thoroughly cleaning the skin and drying it before applying it to the affected area. It should be changed 1-2 times a day.

2.10.5 Antihistamines, reduce mucus

2.10.5.1. Antihistamines reduce nasal mucus and chlorpheniramine. It helps relieve symptoms from colds, sneezing, runny nose, itching, hives, and allergies by taking every 4-6 hours when symptoms occur.

2.10.6 Cough medicine, Expectorant

2.10.6.1. Cough medicine for children helps relieve cough and expectorant. Take 3-4 times a day and shake the bottle every time before using the drug.

2.10.6.2. Brown mixture or M. tussis helps relieve cough, expectorant. Take 3-4 times a day and shake the bottle every time before using the drug.

2.10.7 Inhalers or Ointments for dizziness, fainting, stuffy nose

2.10.7.1. Nasal inhaler ammonia helps relieve dizziness, and lightheadedness, like fainting. Also, apply it to the skin to relieve symptoms from poisonous plant or insect bites. Use a cotton pad moistened and inhaled or applied.

2.10.7.2. An inhaler for dizziness relieves nasal congestion and helps reduce dizziness, stuffy nose, and suffocation.

2.10.7.3. Ointment for relief of nasal congestion in form of ointments helps relieve cold and stuffy nose symptoms

2.10.8 Drugs for motion sickness, seasickness

2.10.8.1. Dimenhydrinate tablets, a seasickness medicine, help relieve and prevent dizziness, nausea, vomiting, or medical conditions that may arise from travel, such as motion sickness or seasickness.

2.10.9 Medicines for eye diseases

2.10.9.1. Sulfacetamide eye drops help treat red eyes and inflammation.

2.10.9.2 Eyewash helps to relieve stinging and eye irritation from dust, dust, smoke, or dirt entering the eyes.

2.10.10 Medicines for mouth and throat diseases

2.10.10.1. Throat suppository helps relieve inflammation and sore throat in adults.

2.10.10.2. Yantian Violet, medicine for the treatment of blemished tongue, helps to treat fungal skin infections such as ringworm, foot bites, oral fungi, vaginal fungi, abscesses, and white melasma in the mouth and are used to prevent skin infections in minor wounds.

2.10.10.3. Toothache pain reliever helps relieve toothache.

2.10.10.4. Throat lozenges for sore irritation help relieve throat irritation and moisten the throat.

2.10.10.5. Throat lozenges for sore throat help relieve sore throat.

2.10.11 Wound dressing and wound washing

2.10.11.1. Iodine tincture wound dressing helps heal fresh wounds.

2.10.11.2. Thimerosal tincture wound dressing helps heal fresh wounds.

2.10.11.3. Povidone-iodine wound dressing helps heal fresh wounds.

2.10.11.4. Isopropyl alcohol helps clean the wound.

2.10.11.5. Ethyl alcohol helps clean the wound.

2.10.11.6. Normal saline solution (NSS) helps clean the wound.

2.10.12 Medicine to relieve muscle pain, insect bites

2.10.12.1. Beeswax balm helps relieve muscle pain or pain, swelling, and inflammation from insect bites.

2.10.13 Medicines for skin diseases

2.10.13.1. Benzyl benzoate, scabies and head lice medicine, can help treat skin diseases such as scabies, head lice, and pubic lice.

2.10.13.2. Sulfur ointment, a remedy for scabies, helps cure scabies.

2.10.13.3. Medicine for dermatophytosis, athlete's foot helps to treat dermatophytosis, athlete's foot.

2.10.13.4. Drugs for chronic skin diseases help treat certain chronic skin diseases, such as eczema or rashes.

2.10.13.5. Calamine rash ointment relieves minor skin irritations such as itching, pain, discomfort, rash, and allergic reactions to plants, urticaria, chemicals, and cosmetics. Burning skin after sun exposure, insect bites

2.10.13.6. Sodium thiosulfate, a medicine for tinea versicolor, helps to treat tinea versicolor.

2.10.14 Body tonic

2.10.14.1. The vitamin B complex helps nourish and strengthen the functions of various systems in the body and prevents vitamin B deficiency by taking it after meals once a day.

2.10.14.2. Vitamin C helps nourish and repair various body parts, such as bones, teeth, and skin and prevents vitamin C deficiency. By eating once a day.

2.10.14.3. Ferrous sulfate, a blood tonic drug, helps treat anemia due to adult iron deficiency.

2.10.14.4. Multivitamins help prevent vitamin deficiency in adults.

2.10.14.5. Cod liver oil helps prevent vitamin A and vitamin deficiency.

2.11 Cloudflare Tunnel

Cloudflare Tunnel allows us to securely expose a locally hosted web server to the internet without exposing it directly to the public internet through Cloudflare's network. Using Cloudflare Tunnel, a locally hosted web server will access through a secure, encrypted tunnel with encrypted traffic end-to-end to protect it from interception. A tunnel will be established between your local machine and a Cloudflare edge server, as shown in Figure 18.

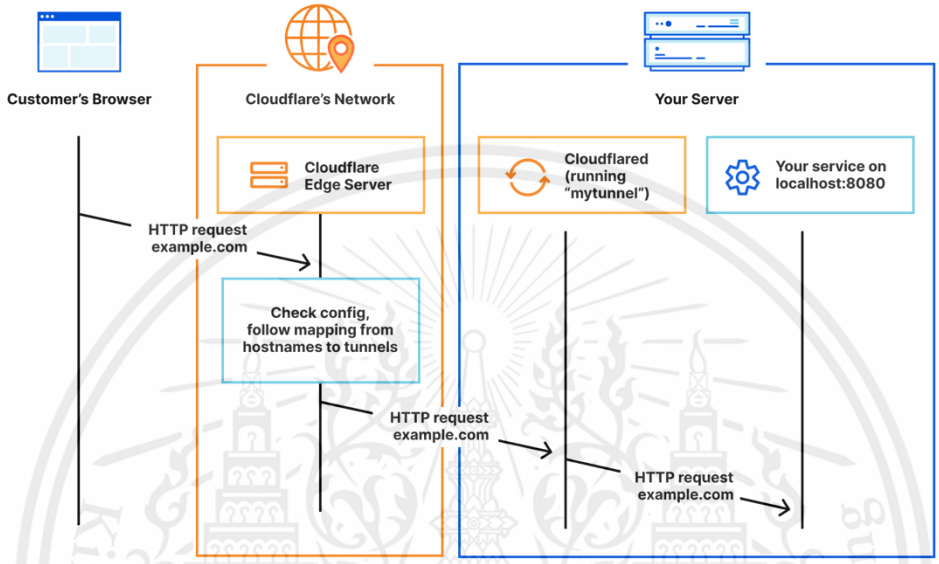


Figure 18: The schematic of Cloudflare Tunnel. [29]

This allows us to access our own created server without exposing it to potential security threats on the public internet. As such, Cloudflare Tunnel supports a variety of cases, such as exposing local development environments and enabling remote access to IoT devices. Due to the above reasons, Cloudflare Tunnel is helpful for our central drug database by showing our data to the users with security that prevents interception.

2.12 Docker Desktop

Docker is a software platform that allows developers to package their applications and other dependencies into containers that can be easily deployed across different environments. Docker containers can be run on any machine that supports Docker, regardless of operating system or hardware. Then, Docker uses a client-server architecture by using the Docker daemon to manage and create containers. Docker images can be built locally or pulled from a registry like Docker Hub. This can be explained by following Figure 19.

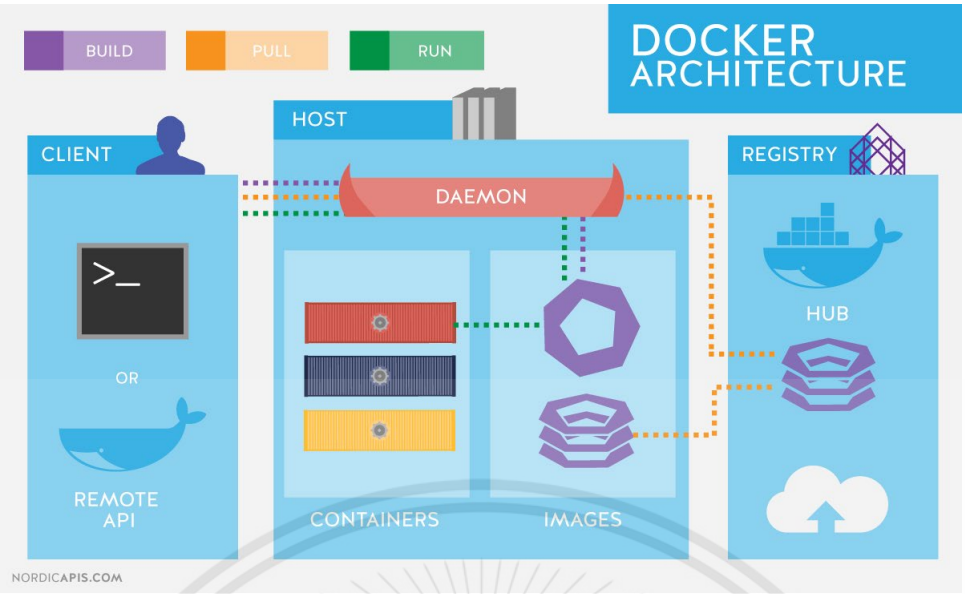


Figure 19: The schematic of Docker architecture. [30]

Due to the reasons above, Docker is applicable for developing a central database, as it enables developers to easily create and deploy applications across different environments while providing isolation, security, and portability benefits.

2.13 Visually impaired person

2.13.1 Visually impaired or blind person meaning a person has a visibility level less than 3/60 or 20/400 (for the specification of visual field requirements in a population sample is hard to find. As such, this is often not used in surveys).

2.13.2 Low vision person means patients with a vision that is less than 6/18 or 20/70 to 3/60 or 20/400 (moderate visual impairment).

The most critical risk factor for visually impaired and blind people is in a group of people that are 50 years old or above. Significant causes of vision impairment are uncorrected refractive errors and cataracts. Because of that, the researchers have given this group of people the utmost importance.

2.14 Target Group

Due to the reasons above, this project will be used on a sample group of visually impaired people, especially older people between 55 and 80. Researchers have made a satisfaction list for the users of the application, as shown below:

No	Assessment list	Satisfaction level				
		5	4	3	2	1
1	The application can display and read the information correctly.					
2	The application can read the information clearly in Thai.					
3	The application can provide additional drug information.					
4	The application can provide caution to prevent drug duo-interaction.					
5	The application is convenient to use and easy to understand.					
6	Overall satisfaction with service quality					

Table 1: Figure of satisfaction assessment form

This satisfaction assessment will have a sample group in number of 10 - 30 people at this location below:

1. Krabi Hospital
2. Vachira Phuket Hospital
3. Bangkok Hospital Phuket

CHAPTER 3
METHODOLOGY

3.1 Introduction

This chapter will describe the design and structure of application, along with centralized database. First, the chapter describes the project's design methodology, and the system requirements. A proposed solution and discussed, followed by a process of solution.

3.2 Design Methodologies

3.2.1 Application Development

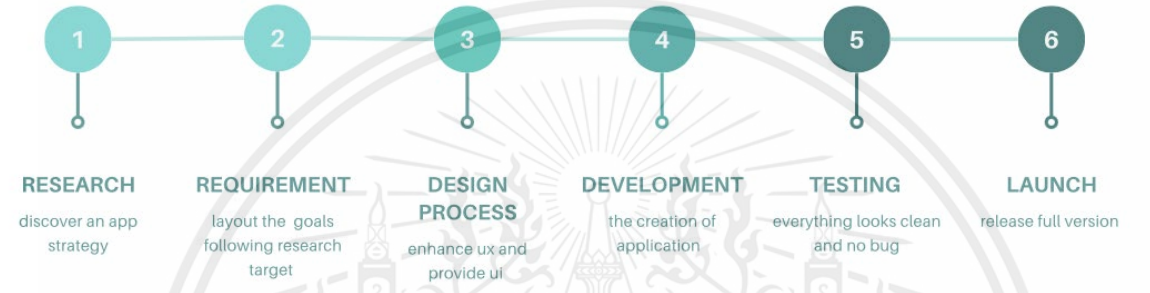


Figure 20: Figure of Application Development.

The explanation of each step in the application development process:

3.2.1.1 Research: This is the first step in application development, where you identify the problem and conduct research to understand the market and user needs. This step involves gathering information about the project, identifying stakeholders, and determining the scope of the project.

3.2.1.2 Requirements: After conducting research, you move on to the requirements gathering phase. In this step, you identify and document the functional and non-functional requirements of the application. This includes defining the features, functionalities, and user flows that the application should have.

3.2.1.3 Design: Once the requirements are defined, you move on to the design phase. In this step, you create the visual and interactive design of the application. This includes designing the user interface, user experience, and creating wireframes and prototypes.

3.2.1.4 Development: Once the design is finalized, the actual development of the application begins. This step involves writing code, integrating APIs, and creating the backend infrastructure. It is the most time-consuming phase of application development.

3.2.1.5 Testing: After the development phase, the application undergoes rigorous testing to ensure it meets all the requirements and is free from bugs and errors. This includes functional, integration, and user acceptance testing.

3.2.6 Launch: The final step in the application development process is the launch phase. This involves deploying the application to production and making it available for users. This step also involves post-launch monitoring, bug fixing, and adding new features based on user feedback.

3.2.2 Application Design

When designing an application, it is necessary to write a schematic diagram of the application, including defining software and APIs used in application development. The working flow of the application is shown in Figure 21. The application operation starts with the user taking a picture of the drug that wants to read the information on how to use it. Then send it for processing with Google cloud vision to detect words on drug labels (OCR). The given drug name will be sent to you for information on drug usage from the central database. The obtained drug administration information is transmitted to text-to-speech with Bot Noi API software and displayed in Thai-speech text and English for users.

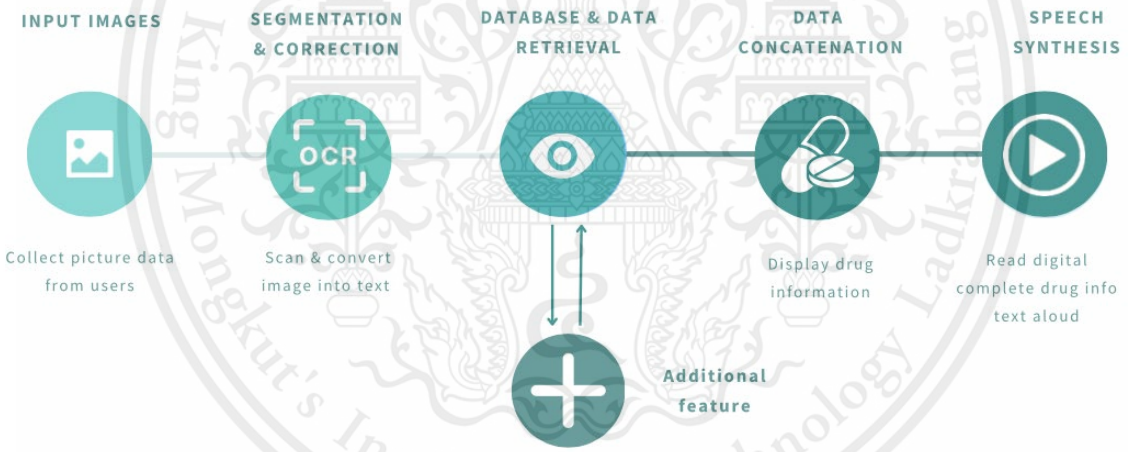


Figure 21: Figure of design of Application Design.

3.2.2 Central Database Design

A central database is a platform that provides supplementary information on user-selected drugs for promoting the proper use of medicine and healthcare. This platform will identify the detected names of prescription and non-prescription drugs to search and provide drug information because this information is of utmost importance to the users. So, the researchers have concluded to create our specific central database for this project especially. This database platform will contain 200 - 300 samples of drug data that emphasize non-prescription to cover the most drug that patients and users will use. Most of the information

that is contained in the central database comes from Drugbank and MedThai. Moreover, the researcher will provide the information that users should know before use, which includes:

3.2.2.1 Drug usage (in general): Drug usage will provide information on how to use the drug in the most efficient way possible. The database will provide information on drug amounts per day, usage directions, medication time, medical advice, and drug maintenance.

3.2.2.2 Adverse effect: This information will warn users how a drug can cause an adverse effect that can be common or need a consultant. In the latter case, users must suspend the usage of the medication and contact medical professionals for directions.

3.2.2.3 Caution from the duo - interaction: This information will provide the situation that should use the drug to reach the most efficacy and prevent the complications that could happen by the cause of drug duo – interaction when the user misuses the drug.

3.2.3 Application structure

React Native is a JavaScript framework that can develop applications on both android and iOS platforms. To start application development, the platform must be prepared. and install the necessary libraries through the following steps

- prepare tools and frameworks before starting to code applications
- Install XCode
- Install node and watchman with brew via terminal command
  >> brew install node
  >> brew install watchman
- Install Cocoa Pods watchman with brew via terminal command
  >> Sudo gem install Cocoa pods
- Create a new application with the command >> npx react-native init “application name.”
- To install the library for using the iPhone camera, you need to install the library first from the expo via the terminal command
  >> expo install expo-camera then enter information for permission to use the camera in the iOS system in the Info.plist file and install the pod for the camera library.

The application structure can be divided into three main parts: App.js, BotnoiAPI.js, and Search.js, each of which has specific function as shown in Figure 22.

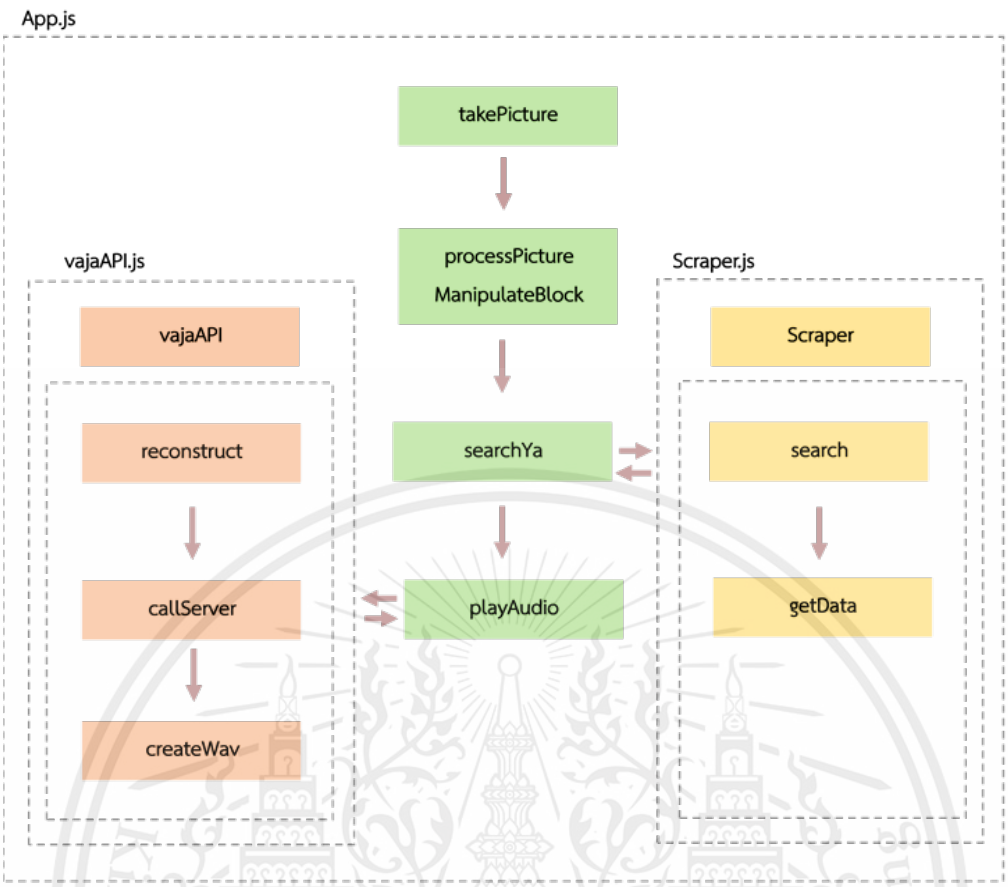


Figure 22: Figure of design of application structure.

3.2.3.1 App.js

App.js is the main file for call function from BotNoiAPI.js and Search.js. This main file includes

3.2.3.1.1 Take Picture: Function for collecting picture data. This will be used in the library in expo-camera to create the command.

3.2.3.1.2. Process Puncture Manipulate: Function for detecting text from picture that come from Take Picture function. Then send the data to cental database function Which use Text Recognizer Process Image from react-native-firebase/ml-vision library to process.

3.2.3.1.3 Search Ya: Function for process data that come from Process Picture Manipulate function by using Scraper function in Scraper.js. Then send back the medical data.

3.2.3.1.4 Play Audio: Function for transforming text data from central database into sound. Text data will be sent to BotNoiAPI function in VajaAPI.js file to process and then turn back in sound form.

3.2.3.2 BotNoiAPI.js

BotNoiAPI.js is a file that contains 3 main functions for converting text data into sound.

3.2.3.2.1 Reconstruct: Function for arrange the medical data from Search Ya Function. To divide into set of syllables which no more than 100 syllables to prepare the data before sending to Call Server.

3.2.3.2.2 Call server: Function for sending text data to BotNoiAPI software to convert into sound.

3.2.3.2.3 Create wav: Function for taking sound files from BotNoiAPI software and convert to .wav file.

3.2.3.3 Meddb.js

Meddb.js is a file that contains a function for decision tree searching in created central database. In this file, it has three sub-functions, Match, Check and Get Data.

3.2.2.3.1 Match: Function for arrange and schematize the detected medical data into categorized dataset.

3.2.2.3.2 Check: Function for scanning detected dataset and check if there's information in each set.

3.2.2.3.3 Get Data: Function for connect and taking the information which is missing in dataset from centralized database through API connection.

3.2.4 Information Architecture

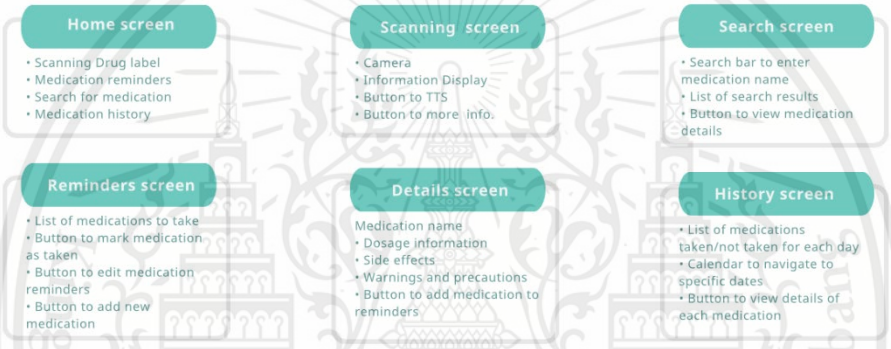


Figure 23: Figure of design of Information Architecture.

Information architecture is the process of organizing, structuring, and labeling content in a way that makes it easy to find, understand, and navigate. It involves designing the overall structure of an information system, such as a website, app, or database, by defining how different pieces of content are related and how users can interact with them. Information architecture considers the user experience, as well as technical and business requirements, to create a logical and intuitive system that meets the needs of its intended audience. Key elements of information architecture include navigation systems, labeling, and categorization.

3.2.5 Algorithm

OCR process algorithm:

- Receive the medication label image.
- Preprocess the image by applying filters, resizing, and normalization to improve the image quality.
- Apply OCR engine to the preprocessed image to extract the text from the image.
- Perform text cleaning, segmentation, and recognition to extract the relevant medication information such as name, dosage, and instructions.
- Store the extracted medication information in the database.

TTS algorithm:

- Receive the medication information from the database.
- Convert the medication information into an audio format using a TTS engine.
- Save the audio file in the server or cloud storage.
- Send the audio file to the frontend for playback or download by the user.

Search algorithm:

- Receive user input for medication search query.
- Query the medication database for the matching medication(s) based on the user input.
- Retrieve medication information such as dosage, side effects, and precautions from the database.
- Return the medication information to the frontend for display to the user.

Notification and delivery scheduling algorithm:

- Receive user input for medication name, dosage, and schedule.
- Save the medication information in the user's profile or database.
- Create a notification schedule based on the medication schedule and save it in the notification system.
- Send a confirmation message to the user that the medication reminder has been set.
- Check the notification system for scheduled medication reminders.
- Send push notifications or SMS messages to the user at the scheduled time.
- If the user does not respond to the notification, send additional reminders at predetermined intervals.
- Log the delivery of each notification in the user's activity history.

Medical History algorithm:

- Receive the medication information from the database.
- Store the medication information in a separate table in the database, with fields for the medication name, dosage, date and time of the medication taken, and a flag indicating whether the medication was taken or missed.
- Whenever the user takes their medication, update the corresponding row in the medication history table to indicate that the medication was taken.
- If the user misses their medication, update the corresponding row in the medication history table to indicate that the medication was missed.
- Provide a way for the user to view their medication history, either by displaying it directly in the app or by exporting it to a file.
- Allow the user to filter and sort their medication history based on various criteria, such as date range, medication name, and dosage.

- Send reminders to the user based on their medication history, such as reminding them to refill their medication or to schedule a doctor's appointment.

3.3 Interesting Problem

This is due to the improved application's performance still having errors in both word recognition and segmentation.

3.3.1. Read in certain specific formats: KinYa apps is only readable information on drug labels from Siriraj Piyamaharajkarun Hospital and Siriraj Hospital drug labeling samples

3.3.2 No auto-correction functions

3.3.3 Insufficient drug database

3.4 Proposed Solution

3.4.1. Readable all drug labelling formats

3.4.2 Create word segmentation and auto-correction functions

3.4.3 Create centralized database

Gantt Chart

Activity	Month											
	Jan	**Feb**	**Mar**	**Apr**	**May**	**Jun**	**Jul**	**Aug**	**Sep**	**Oct**	**Nov**	**Dec**
1. Research and design the overall system												
2. Collect drug label in order to improve optical character recognition												
3. Collect drug information in order to improve central database												
4. Create central database												
5. Create optical character recognition from printed label												
6. Create automatic word corrections from printed label												
7. Create automatic word arranging and chopping by												

using detected word												
8. Create drug information searching function from central database												
9. Improve application and database to have suitable format												
10. Testing optical character recognition's performance using in label detection												
11. Using the test result to improve the system												

Table 2: Figure of Gantt Chart

Collect drug printed label in order to trained optical character recognition (OCR)

From the research plan, researchers need to have input data in order to train OCR system to have maximum performance that can be achieve. First, researchers have collected drug label samples from three hospitals which includes: Krabi Hospital, Vachira Phuket Hospital, Bangkok Hospital Phuket. In which, we have drug label over 300 samples. The researchers have concluded that when OCR system train with these drug labels, the system's performance improves. Due to the reason, we have agreed on expanding the drug label sample by creating crowd sourcing medicine label website. Moreover, this website can also collect different drug label types from various hospitals.

3.5 Summary

The researcher has studied the OCR system and reconstructed KinYa Apps. We require several drug labeling formats by creating KINYA website crowdsourcing to obtain the pattern and style for training word segmentation and Autocorrect function. The centralized databases allow existing knowledge user APIs to build a system that the researcher has pre-configured to generate the requested drug information, such as shelf life, precaution, and duo interaction.

CHAPTER 4

EXPERIMENTAL RESULT

4.1 Introduction

Chapters 3 and 4 of our thesis described the design and implementation of the KINYA Application, a mobile application that allows visually impaired individuals to access and understand medication label information easily. In this chapter, we present a testing method and its results that show the effectiveness of the KINYA app in fulfilling its objectives.

The chapter is organized as follows:

Section 4.1: Introducing the testing methodology used in the evaluation of the KINYA app. We chose to conduct usability testing as it is an effective way to assess the effectiveness of user interfaces. This section describes the participants, tasks, and metrics used in the testing.

Section 4.2: We present the results of the usability testing. The testing showed that MedLabel effectively enables visually impaired users to access and understand medication information. Participants could complete tasks quickly and with a high level of accuracy, and they found the app to be easy to use and navigate.

Section 4.3: We present a summary of the testing results, which demonstrates that KINYA is a highly effective tool for visually impaired individuals to manage their medication safely and effectively.

4.2 Result

The primary objective of this study was to collect a comprehensive dataset of drug labels from various hospitals for the purpose of training artificial intelligence (AI) algorithms in the healthcare domain. A total of 300 drug labels were collected from a diverse range of hospitals, including Vachira Phuket Hospital, Krabi Hospital, and Bangkok Phuket Hospital. These hospitals were selected based on their representation of different healthcare settings and patient populations. The drug labels were obtained through a collaborative effort involving direct communication with the hospital administration and obtaining the necessary permissions and authorizations. The data collection process adhered to strict privacy regulations and ethical guidelines to ensure the anonymity and confidentiality of patient information.

The collected dataset consists of approximately 300 drug labels, which encompassed a wide range of medications, including [list specific types or categories]. Each drug label contained essential information such as drug name, dosage, administration instructions, contraindications, and potential side effects. The obtained dataset represents a valuable resource for training AI algorithms in the healthcare domain. The inclusion of diverse drug labels from different hospitals enhances the generalizability and applicability of the AI models developed.

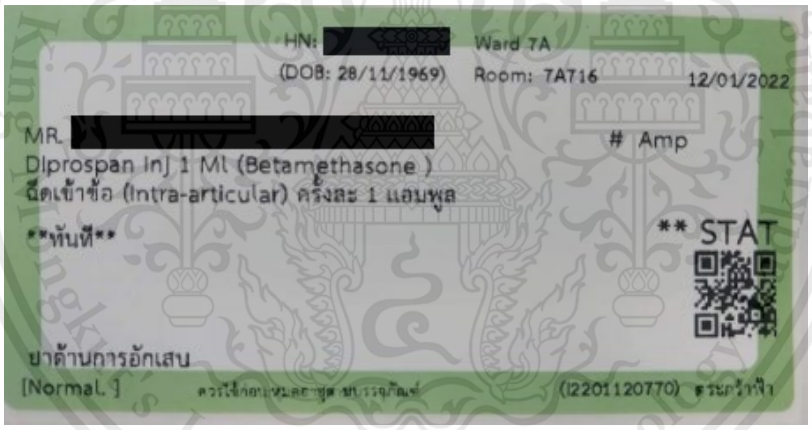


Figure 24: Examples of collected medicine label from Bangkok Phuket Hospital

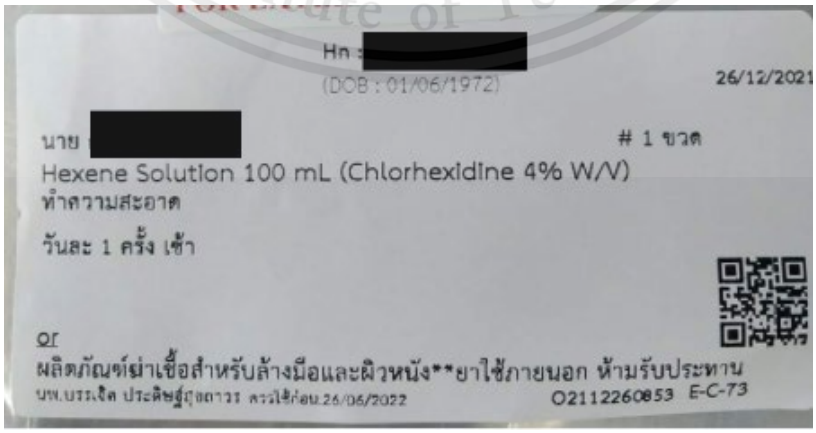


Figure 25: Examples of collected medicine label from Krabi Hospital

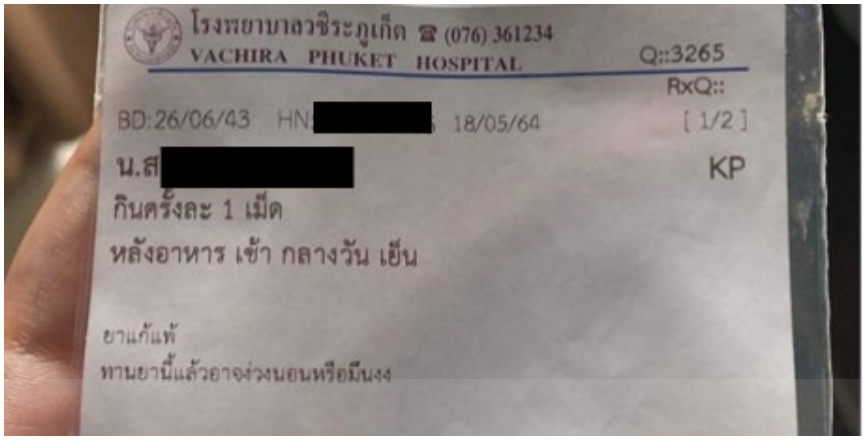
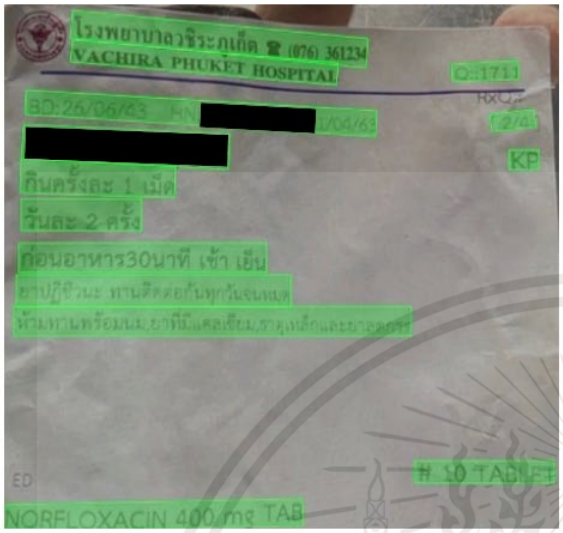


Figure 26: Examples of collected medicine label from Vachira Phuket Hospital

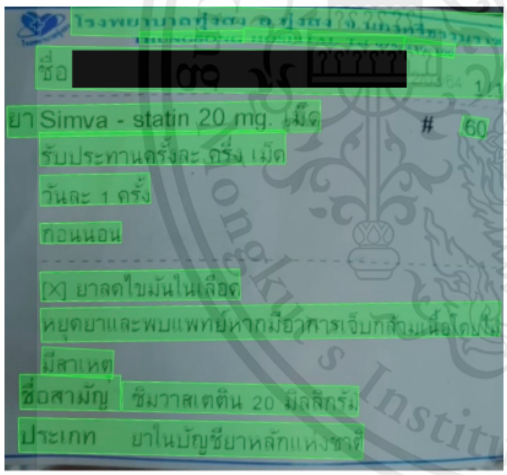
Figure 27: Examples of 300 collected medicine labels in Google Cloud

Following the collection of drug labels, optical character recognition (OCR) was performed to convert the physical labels into a machine-readable format. The OCR process aimed to extract structured data from the labels, facilitating further analysis and utilization in training artificial intelligence (AI) models.




```
(Sat Jun 7 2022 22:39:46.919 LOG
Start filter text Array:
[*boundingPoly";("vertices*;
(Array]), "description":โรงพยาบาล
วชิระภูเก็ต 2 (076) 361234
VACHIRA PHUKET HOSPITAL
BD:26/06/43 HN: [redacted] 01/04/63
[redacted]
กินครั้งละ 1 เม็ด
วันละ 2 ครั้ง
ก่อนอาหาร30นาที เช้า เย็น
ยาปฏิชีวนะ ทานติดต่อกันทุกวันจนหมด
ห้ามทานพร้อมนม,ยาที่มีแคลเซียมธาตุเหล็กและยา
ลดกรด
Q::1711
[2/41
KP
#10 TABLET
NORFLOXACIN 400 mg TAB
```

Figure 28: The results after performing OCR on the collected drug labels of Vachira Phuket Hospital




```
(Wed Nov 9 2022 23:22:45.459 LOG
Start filter text Array:
[*boundingPoly";("vertices*; (Array]),
"description":โรงพยาบาลกุ่งสง อ.กุ่งสง จ.นครศรีธรรมรา
HOSPTTAL Tel 675-
HUNGSONG
[redacted] 20364 1/1
ยา Simva - statin 20 mg. เม็ด
รับประทานครั้งละ ครึ่ง เม็ด
วันละ 1 ครั้ง
ก่อนนอน
[x] ยาลดไขมันในเลือด
หยุดยาและพบแพทย์หากมีอาการเจ็บกล้ามเนื้อโดยไม่
มีสาเหตุ
ชื่อสามัญ ซิมวาสเตติน 20 มิลลิกรัม
ประเภท ยาในบัญชียาหลักแห่งชาติ
416-160
60
```

Figure 29: The results after performing OCR on the collected drug labels of Thungsong Hospital

To extract the text and relevant information from the drug labels, OCR techniques were applied using advanced optical character recognition software in this case we use Cloud Vision API. The software utilized machine learning algorithms trained on a vast amount of text data to accurately interpret the characters and layout of the labels. The resulting output from the OCR process was a JSON (JavaScript Object Notation) file. JSON provides a structured and standardized format for representing the extracted data, enabling seamless integration with AI algorithms and facilitating data processing and analysis.

The JSON file contained a hierarchical structure that organized the extracted data into key-value pairs. Each drug label was represented as a separate object within the JSON file, with the key-value pairs capturing specific information such as drug name, dosage, administration instructions, contraindications, and potential side effects. The structured data obtained from the OCR process offers several advantages. It enables efficient data organization and retrieval, simplifies the feature extraction process for AI models, and facilitates data-driven decision-making.

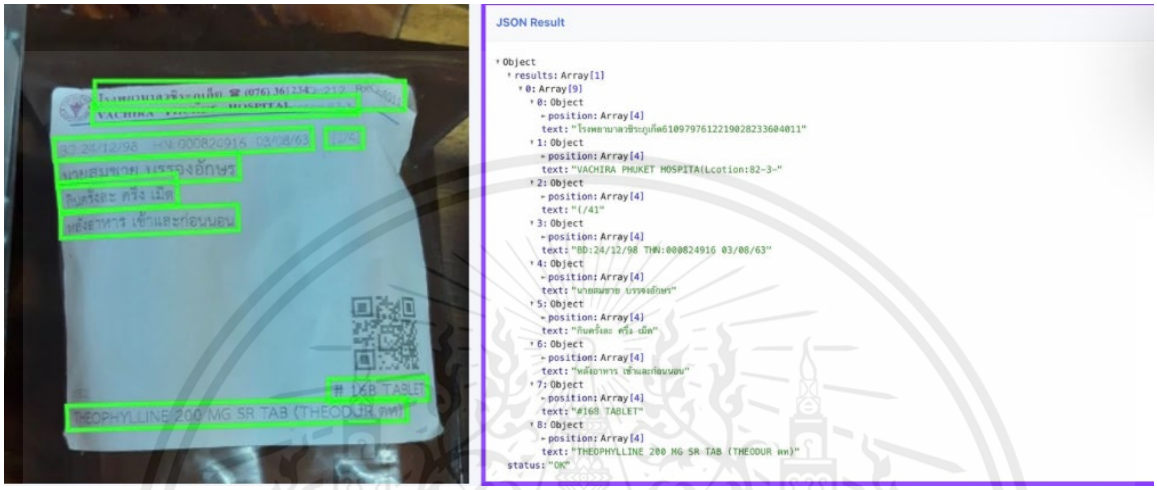


Figure 30: The results after performing OCR on the collected drug labels, the resulting JSON file contains structured data

The application could be divided into front-end and back-end components:

Front-end: The app will be developed using React Native framework, a popular platform for developing mobile applications. The app will be compatible with both iOS and Android devices. In this project, we have only focused on the IOS platform.

User interface design

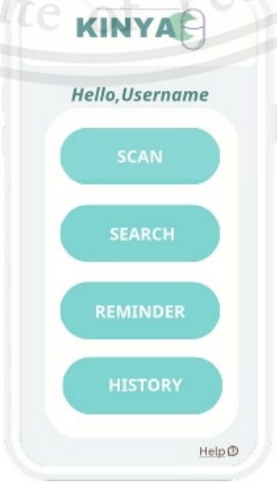


Figure 31: Figure of user interface design

- Medication label scanning feature:

Using the smartphone's camera, the app uses a deep learning algorithm to identify medication label information. It also provides text-to-speech functionality, allowing users to hear the medication label information in a clear voice.

Figure 32: Figure of medication label scanning feature

- Medication reminder

The reminder feature allows users to set up reminders for taking their medication at the appropriate times. It displays all the medications the user needs to take and provides a clear overview of what medications the user needs to take and when. This would help users stay on track with their medication schedule and avoid missing doses, which will be connected to their medical history.

Figure 33: Figure of medication reminder feature

- Medication history

This feature provides users with a history of their medication usage, including which medications they have taken. It allows users to track which medications they have taken or not. A dot will represent each day. If the user takes their medication, the dot will be filled in.

If they miss their medication, the dot will be empty. This information could be helpful for users to share with their healthcare providers and could help prevent medication errors or interactions. It could also help users track their medication adherence and identify potential issues.

Figure 34: Figure of medication history

- Drug information search

This feature allows users to search for specific medications and access information about their purpose, dosage, side effects, and warnings or precautions. It would be beneficial for individuals who are prescribed multiple medications, as they may need to be more familiar with all of them. It would also benefit individuals who may need to remember the details of their medication regimen. By providing easily accessible drug information, the search function would help users make informed decisions about their medication use and better understand their treatment plans.

Figure 35: Figure of drug information search

Back-end:

MongoDB Compass is being used as the database for user data storage and retrieval in the application. As said before, MongoDB has a unique mechanism for storage and retrieval, which helps in storing vast volumes of data and retrieving it within a fraction of a second. This made MongoDB to be our choice for the platform for our central drug database. As of now, the central database has 300 documents of information drugs which comprises of adverse reactions, duo interaction, maintenance, properties, dosage type and drug usage. Moreover, the central database has a sample picture for each medicine for users to see, as shown in Figure 36.

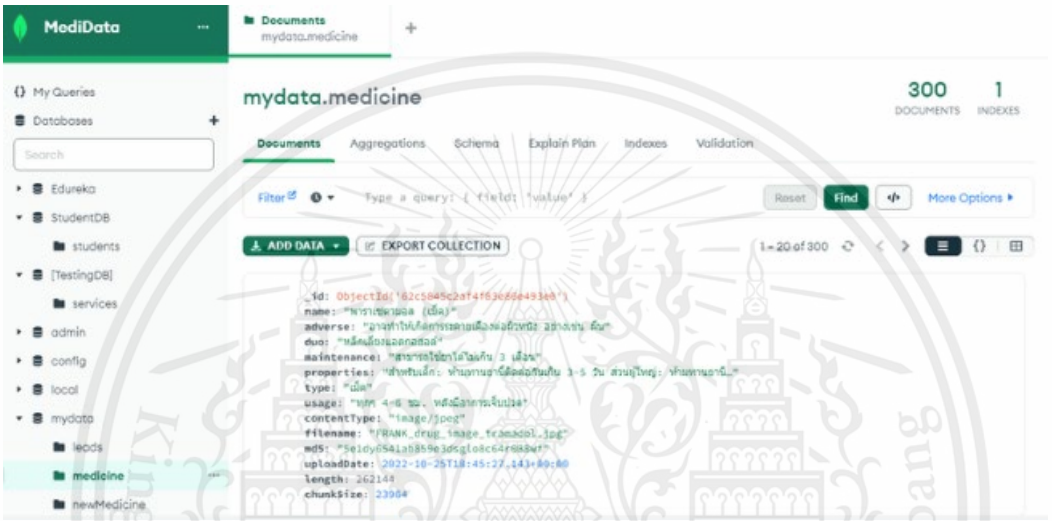


Figure 36: Central Database in MongoDB Compass

- User data storage and retrieval

Database design: It cover the structure of the database, such as the drug collections and fields used to store user data. This section will explain any decisions made during the design process.

In the central database, there are two main categories of drugs: household and prescription drugs, as household drugs are safe and effective if one follows the directions on the label. On the other hand, prescription drugs will only be available for users after receiving authorization from a healthcare practitioner as they have side effects and risks.
As such, the type of medicine is essential for users that we need to separate each other even though they intend to treat the same medical conditions.

After that, in each type of medicine, various therapeutic classes will be contained as a system of organizing medications based on the medical conditions they are designed to treat. Then, each therapeutic class has many dosage forms or physical forms that users can take with the option of tablet, syrup, gel, etc. With all the mentioned parameters, there is a group of medicine that has the same properties. This set of medications will have information on their properties and precautions. The information can be mainly categorized as drug usage, adverse reactions, duo interaction, and shelf life, as shown in Figure 37.

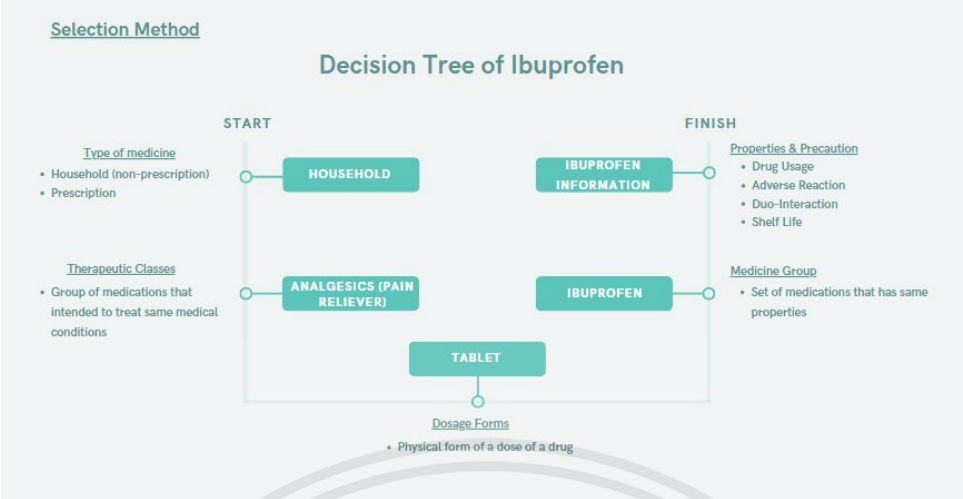


Figure 37: Figure of Selection Method

- User data storage and retrieval

Moreover, in the central database, there is one sample photo of household and prescription drugs to show the users what the shape and form of the drug could look like. In order to do this, we implemented GridFS for storing and retrieving files, such as photos, that exceed the size limit of 16 MB into our code in VSCode as shown in Figure 38.

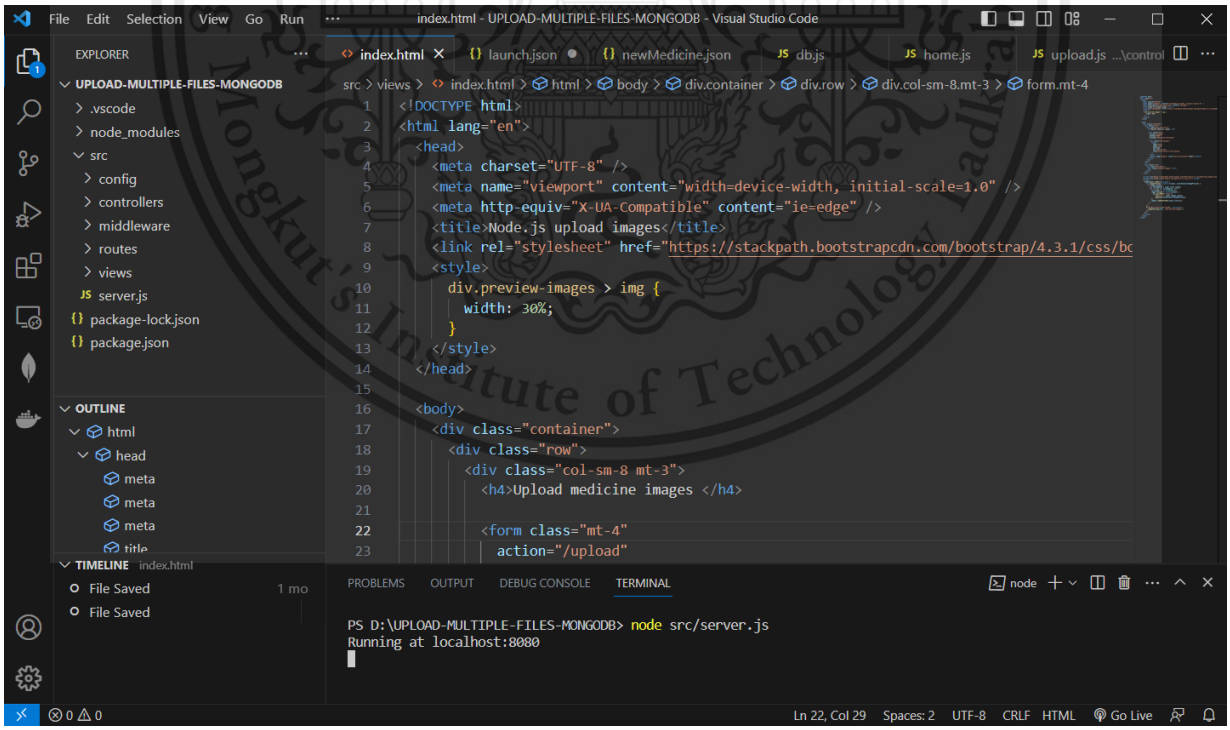


Figure 38: Figure of GridFS Coding

Once we enter “node src/server.js” in the terminal, we will get the page for uploading medicine photos to our central database. On this page, you can upload the maximum number of photos in 10 photos at once, as shown in Figure 39. After uploading, we will match the photo with its relevant information inside the database.

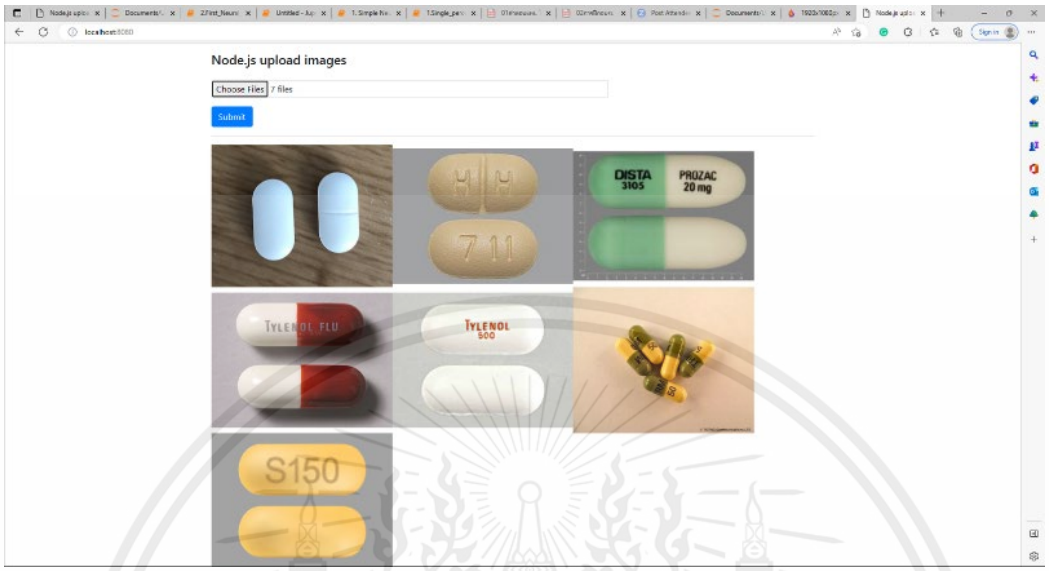


Figure 39: Figure of image uploading page

- Data export from the central database in MongoDB Compass

In MongoDB Compass, there is the option to both export and import the data in two files: JSON and CSV file types. In order to export the file, we must select the output field first as information is divided into the subsection of fields. Once you selected the wanted field, you can choose to export in JSON or CSV file type as shown in Figure 40.

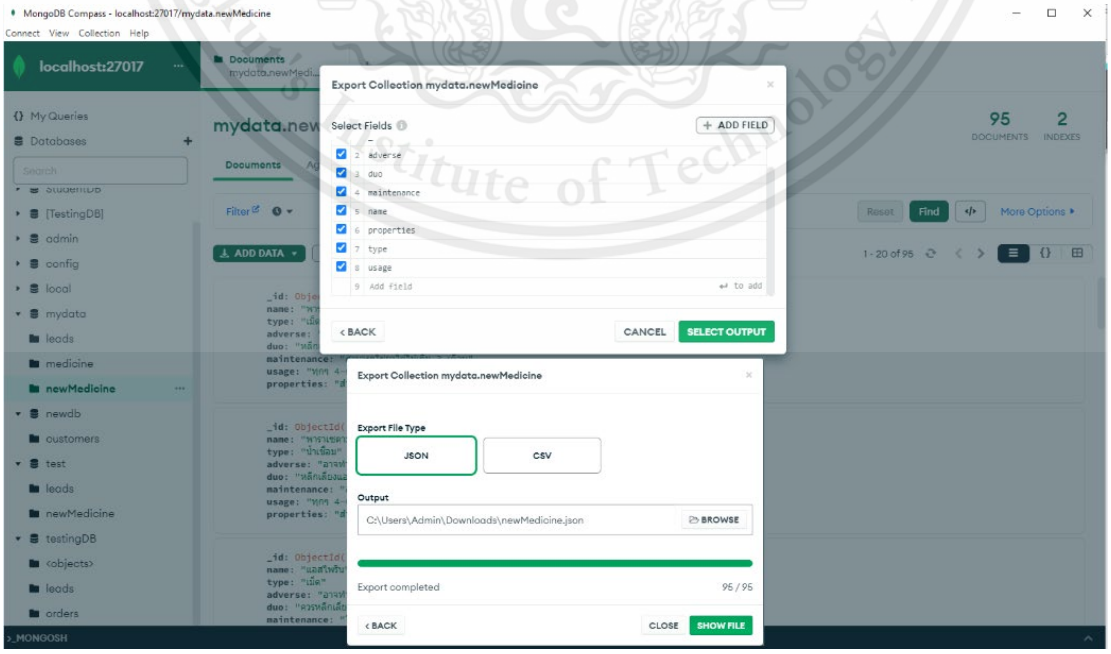


Figure 40: Figure of Data export from MongoDB Compass

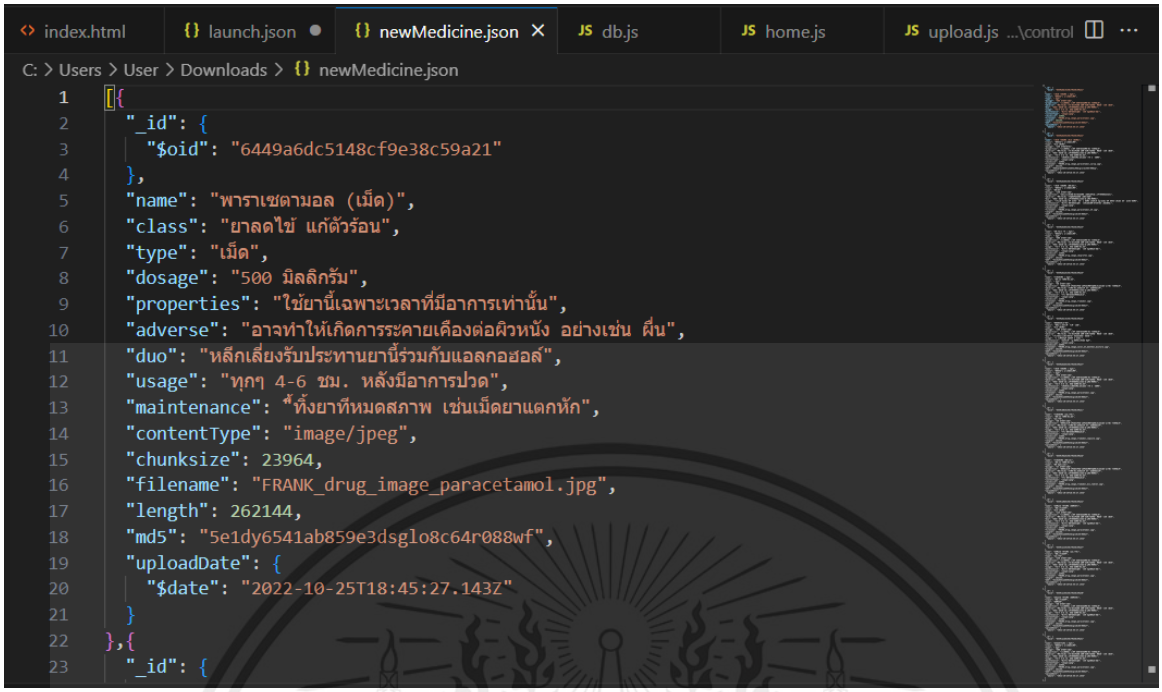


Figure 41: Example of exported data from MongoDB Compass

- API integration for medication information database

Upon entering a search query for a specific medication, the API will retrieve relevant information from the database and display it to the user. This information may include the medication's name, purpose, dosage, potential side effects, and warnings or precautions. The API may also provide additional functionality, allowing users to filter search results based on specific criteria (e.g., dosage strength, brand vs. generic) or offering suggestions for alternative medications based on the user's search history.

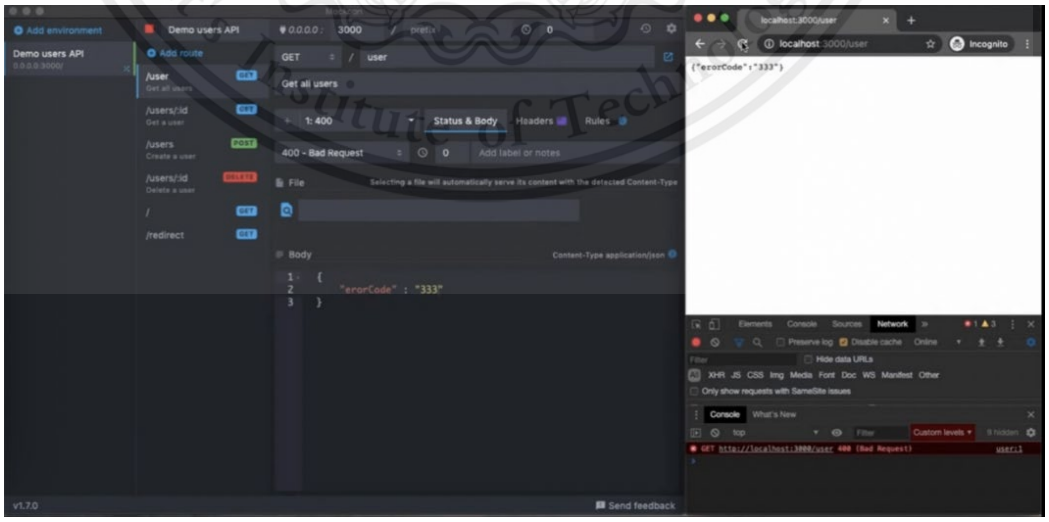


Figure 42: Figure of API integration for database

To prevent data loss, we will connect our database through Docker and Cloudflare Tunnel and also allow you to edit our database as it can connect with any computer with Docker Desktop install. For that, we must both install Docker Desktop on the computer and sign up for the Cloudflare account. Once succeed, start creating your own tunnel inside the Cloudflare account as shown in Figure 43.

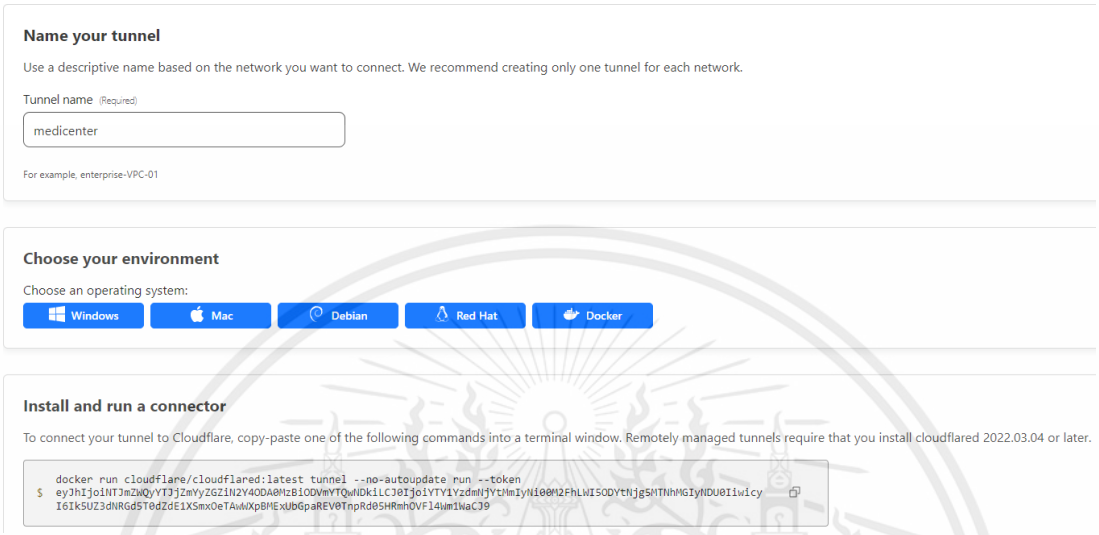


Figure 43: Figure shows how to create Cloudflare Tunnel

After that, run the connector to the command prompt in order to connect with Docker Desktop and check if the tunnel is connected to the Docker Desktop or not, as shown in Figure 44. If there is an error when you trying to open Docker Desktop, please check on your computer if BIOS Visualization or not.

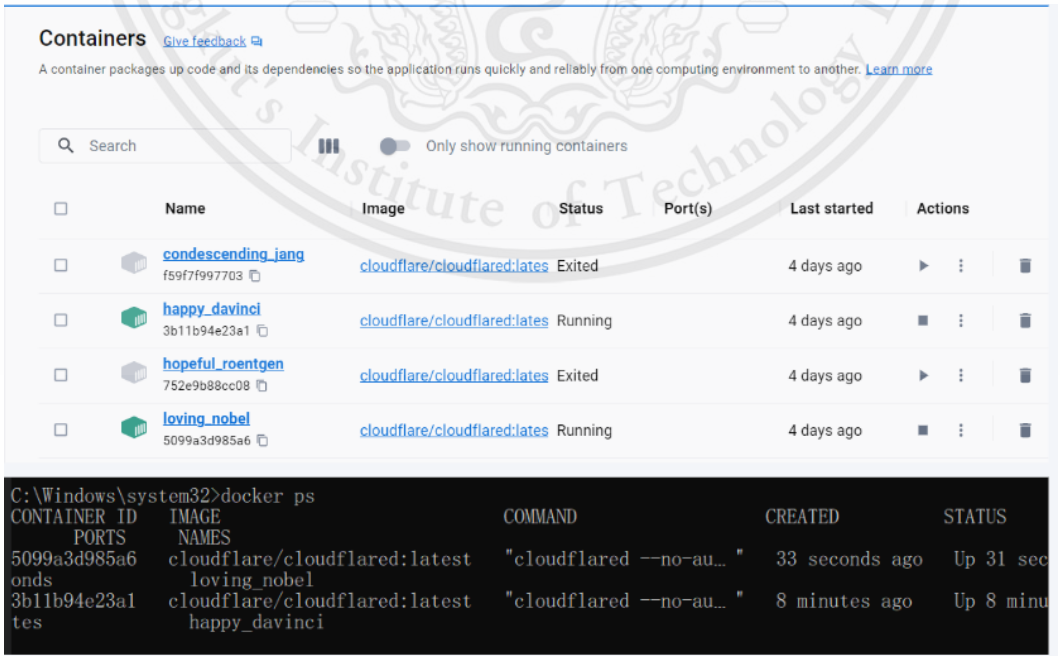


Figure 44: Figure shows Docker Desktop connected with Cloudflare Tunnel

Finally, create your own site and connect with the Cloudflare Tunnel to complete the process, as shown in Figure 45.

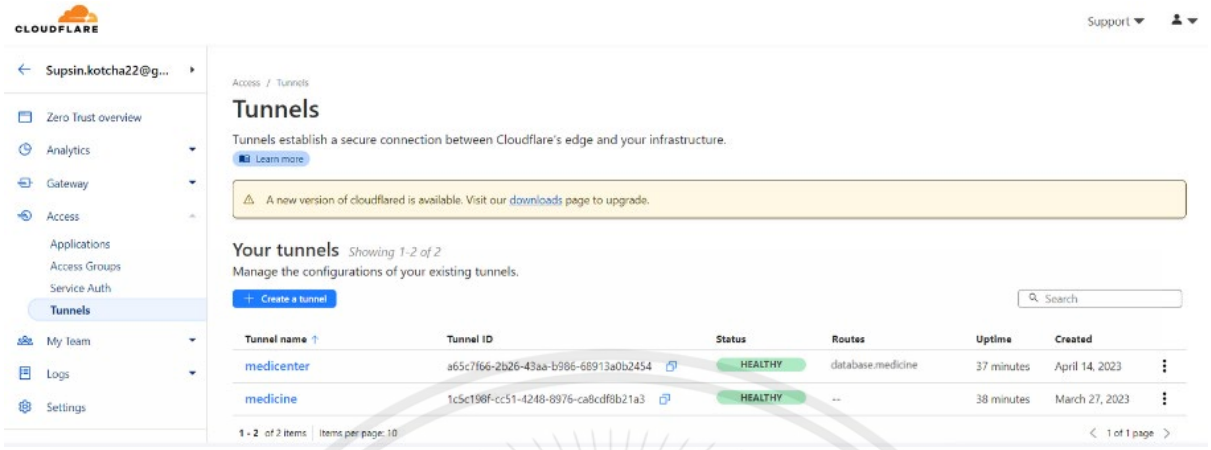


Figure 45: Figure shows our central database connected with Cloudflare Tunnel

4.3 Summary

In summary, this chapter presents the results of usability testing that demonstrates the effectiveness of KINYA app in fulfilling its objectives. The testing methodology and results show that the app is highly effective in enabling visually impaired individuals to manage their medication safely and effectively. This chapter provides valuable insights into the development and implementation of mobile applications for visually impaired individuals and the potential benefits that they can provide.

CHAPTER 5
CONCLUSION

5.1 Conclusions

The aim of this project was to develop an assistive application that can help visually impaired individuals read and access drug label information, including medicinal information, precautions, and potential drug interactions. Our focus was on developing a suitable word correction that could supply drug information in both text and voice formats in Thai.

We designed and implemented a system that combined OCR technology, deep learning models, and a medication database to accurately scan and identify medication labels, provide users with detailed drug information, and set up medication reminders.

Our hypothesis was that our assistive application could significantly improve the lives of visually impaired individuals by providing them with better access to medication information and reducing the risk of medication misuse. We tested this hypothesis by conducting user studies and analyzing the results to evaluate the effectiveness and usability of our application.

Overall, our application proved to be a promising solution to help visually impaired individuals manage their medication safely and effectively. We believe that our application has the potential to make a significant impact in improving the quality of life of visually impaired individuals.

5.2 Suggestion

1. Integration of databases: The app relies on several databases to provide accurate medication information. However, integrating these databases can be a challenge, as they may have different formats or be incompatible with each other.
2. Security: As the app stores personal medication information, security is a critical concern. The app needs to ensure that user data is protected against unauthorized access, data breaches, and other security threats.
3. Integration of voice command: The app aims to provide voice command functionality for users who have difficulty navigating the app. However, integrating voice command can be a complex process, as it requires the app to recognize and interpret spoken commands accurately.
4. Technical limitations: The app relies on several technologies, including OCR, machine learning, and voice recognition, which can be complex and resource intensive. Technical limitations, such as processing power and storage space, could pose a challenge during development.

5.3 Discussion

The development of a medication management application with scanning drug label functionality has presented a significant contribution to the field of healthcare technology. The application aims to provide visually impaired individuals with an easily accessible platform for managing their medications safely and effectively.

The front-end of the application has been almost completed, with features such as medication reminders, manual setup of drug reminders, medication history, drug information search, and OCR for scanning drug labels. The OCR algorithm allows for the identification of drug labels by capturing an image of the label and converting it into text for further processing.

However, the back end of the application requires further development to include voice command, help features, and security protocols. The voice command feature would enable users to access the application hands-free, providing added convenience for visually impaired individuals. The help feature would provide users with access to a support system, in case they require assistance with navigating the application or managing their medications.

In addition, security protocols are crucial for ensuring the privacy and confidentiality of user data. Integration of security features such as data encryption, authentication, and authorization are essential to protect sensitive user information from cyber-attacks.

Future for the development of the application include integrating all the front-end and back-end functionalities, testing, and refining the application based on user feedback. The testing phase will involve evaluating the application's performance in terms of accuracy, efficiency, and usability. The refinement phase will involve incorporating user feedback to improve the application's functionality and user experience.

In conclusion, the medication management application with scanning drug label functionality is a valuable contribution to the healthcare technology field. The application's innovative features and easy accessibility make it a potential solution for visually impaired individuals in managing their medications safely and effectively. Further development and refinement of the application will enhance its functionality and usability, making it a valuable tool in improving patient outcomes.

REFERENCE

[1] Thanyaporn. 2020. "Department for Empowerment of Persons with Disabilities" [Online]. Available: http://dep.go.th/Content/View/6076/1

[2] Wikipedia. 2020. "Be my eyes" [Online]. Available: https://www.bemyeyes.com/.

[3] Mauro Avila 2016 "Remote assistance for blind users in daily life: A survey about Be My Eyes" Available: https://dl.acm.org/doi/10.1145/2910674.2935839

[4] Terri Hedgpeth. 2003. “iCare-Reader – A truly portable reading device for the blind” Higher Education, Boulder, CO.

[5] Celine Mancas-Thillou. 2007. “A multifunctional reading assistance for visually impaired” EURASIP Journal on Image and Video Processing Volume 2007, Article ID 64295.

[6] Ajay Roy. 2017. “Text recognition and face detection aid for visually impaired person using Raspberry PI” International Conference on circuits Power and Computing Technologies [ICCPCT]

[7] Huiying Shen and James M. Coiughlan. 2012. “Toward a real-time system for finding and reading signs for visually impaired users” The Smith-Kettlewell Eye Research Institute, San Francisco, CA.

[8] Nobuhito Tanaka. 2012. “A system helping the blind get merchandise information” Available: ytake@daidi-it.ac.jp.

[9] Tepelea Laviniu. 2014. “OCR application on smartphone for visually impaired people” Journal of Electrical and Electronics Engineering.

[10] Leo Neat. 2019. “Scene text access: A comparison of mobile OCR modalities for blind users” Available: https://dl.acm.org/doi/10.1145/3301275.3302271

[11] Hitanshu, "Medium," Accessed: Oct. 15, 2018. [Online]. Available: https://medium.com/androidiots/firebase-ml-kit-101-text-recognition-3c88f468bff6.

[12] "MathWorks," [Online]. Available: https://www.mathworks.com/solutions/deep-learning/convolutional-neural-network.html.

[13] I. Isewon, "Design and Implementation of Text To SpeechConversion for Visually Impaired People," International Journal of Applied Information Systems (IJAIS) , vol. 7, p. 27, 2014.

[14] Smart sop’s surveillance system using deep neural network - researchgate, https://www.researchgate.net/publication/359727980_Smart_SOP’s_Surveillance_System_using_Deep_Neural_Network (accessed Jun. 8, 2021).

[15] M. A. Nielsen, Neural networks and deep learning, http://neuralnetworksanddeeplearning.com/chap6.html (accessed April. 4, 2021).
[16] Typical CNN architecture [14]. | download scientific diagram - researchgate, https://www.researchgate.net/figure/Typical-CNN-architecture-14_fig1_325803364 (accessed Aug. 18, 2022).
[17] H. Dhawan, “Firebase ML Kit 101 : Barcode Scanning,” Medium, https://medium.com/androidiots/firebase-ml-kit-101-barcode-scanning-9207e9613d3b (accessed Aug. 20, 2022).
[18] The DSP component of a general concatenation- based synthesizer . [17 ..., https://www.researchgate.net/figure/The-DSP-component-of-a-general-concatenation-based-synthesizer-17_fig3_276195975 (accessed Nov. 5, 2022).
[19] Fujisaki’s command-response contour model [8] - researchgate, https://www.researchgate.net/figure/Fujisakis-command-response-contour-model-8_fig1_221482653 (accessed Nov. 5, 2022).
[20] The DSP component of a general concatenation- based synthesizer . [17 ..., https://www.researchgate.net/figure/The-DSP-component-of-a-general-concatenation-based-synthesizer-17_fig3_276195975 (accessed Jan. 5, 2023).
[21] “Simulation acceleration using MATLAB Coder and parallel computing toolbox,” Simulation Acceleration Using MATLAB Coder and Parallel Computing Toolbox - MATLAB & Simulink - MathWorks España, https://es.mathworks.com/help/comm/ug/simulation-acceleration-using-matlab-coder-and-parallel-computing-toolbox.html (accessed Mar. 5, 2021)
[22] R. Ilango, “Using NLP (BERT) to improve OCR accuracy,” Medium, https://medium.com/doma/using-nlp-bert-to-improve-ocr-accuracy-385c98ae174c (accessed Oct. 5, 2022)
[23] R. E. Drinkwater, R. W. N. Cubey, and E. M. Haston, “The use of Optical character recognition (OCR) in the digitisation of herbarium specimen labels,” PhytoKeys, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4086207/ (accessed Oct. 2, 2022)
[24] “Core components and native components · REACT native,” React Native RSS, https://reactnative.dev/docs/0.65/intro-react-native-components (accessed Oct. 1, 2022).
[25] D. Abballe, “React native - first project using Expo Snack,” Zone of Development, https://www.zoneofdevelopment.com/2020/11/25/react-native-first-project-using-expo-snack/ (accessed Oct. 20, 2022).

[26] R. Gandhi, “Support Vector Machine - introduction to machine learning algorithms,” Medium, https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47 (accessed Feb. 6, 2023).
[27] D. Wilimitis, “The kernel trick,” Medium, https://towardsdatascience.com/the-kernel-trick-c98cdbcaeb3f (accessed Feb. 6, 2023).
[28] P. Saeed Mohajeryami, “Navigating the world of nosql: Choosing the right database,” Medium, https://levelup.gitconnected.com/navigating-the-world-of-nosql-choosing-the-right-database-8fc49f87d64b (accessed Jan. 24, 2023).
[29] “Cloudflare Tunnel · Cloudflare Zero Trust Docs,” · Cloudflare Zero Trust docs, https://developers.cloudflare.com/cloudflare-one/connections/connect-apps/ (accessed Jan. 4, 2023).
[30] V. van der Mersch, “API-driven devops: Spotlight on Docker: Nordic apis |,” Nordic APIs, https://nordicapis.com/api-driven-devops-spotlight-on-docker/ (accessed Jan. 5, 2023).