SMART
PHARMACY

Abstract

A Doctor's medical prescription is a hand-written paper that a doctor writes to prescribe the medicine to the patient according to the injury or sickness that the patient has been experiencing. Misread medicine names in doctors' medical prescriptions are frequently a consequence of either unreadable handwriting or a pharmacist's incapability to identify drug names in medical prescriptions.

This project demonstrates how develop a system that can recognize handwritten English medical prescriptions. Using transfer learning YOLOV5, VGG 16, and Deep Convolution Neural Network to train this supervised system, input images are segmented and processed to:

- recognize characters and classify them into different predefined characters.
- recognize word and classify them into different predefined classes.

Keywords

Transfer Learning, YOLOV5, VGG 16, CNN.

Acknowledgement

Although we faced many difficulties to complete this the project successfully, there was a person who helped and facilitated many complicated matters for us, we would like to extend our thanks, appreciation, and love to Dr. Marwa Abdelfattah, for her support and encouragement and her all-time availability besides guiding us to the best choice and giving us opportunities to think deeply to get the best ideas to improve our graduation project.

Finally, great thanks to our families and our friends and a special thanks to our families for their continuous support without getting bored and encouraging us to do our best until we reach what we are and we thank friends for being by our sides, helping, supporting, and giving us useful opinions.

Thanks all from the deepest of our hearts.

Table of Contents

1 Introduction

1.1 Overview

As we navigate the digital age & The power of technology to address real-world challenges.

Imagine a scenario where visiting a pharmacy is no longer a daunting task filled with uncertainties and potential errors!

Using Modern Smartphones and Artificial Intelligence to detect medicines accurately, by simply scanning a prescription using the app.

Its key features, technical architecture, and the potential impact it can have on the healthcare landscape.

1.2 Problem Statement

- A Doctor's medical prescription is a hand-written paper that a doctor writes to prescribe the medicine to the patient according to the injury or sickness that the patient has been experiencing. Misread medicine names in doctors' medical prescriptions are frequently a consequence of either unreadable handwriting or a pharmacist's incapability to identify drug names in medical prescriptions.

- Normally, a medical prescription, which is written by a doctor who uses common medical terminologies and Latin abbreviations, is usually extremely hard to be read and understood by a person who has no prior medical knowledge or background.

- It cannot be denied that it is very threatening when medicines are wrongly given to patients as it can lead to some major health problems because of the side effects that some medicines have over each other when taken at the same time, not only that but also the wrong medicine is taken over a long time without a need for it.

1.3 Scope and Objectives

The goal of the project is to provide a convenient way for anyone to read a prescription and track their health status.

We did it through :

- simply scan a prescription using the app.
- Health-report of each user contains his chronic diseases, and the app will raise a warning when ordering harmful drugs for your chronic diseases.
- You donot have to keep your prescription, the app will save your history and you can review it anytime.

1.4 Work Plan

we used two Different ways to recognize handwritten words. word recognition and character recognition.

2 Related work

2.1 Validation Phase

Some of papers that we are read not maintain the result that we need to fetch so let's represent one of them, the problem we faced and how we solve it.

2.2 This Paper generated by CNN

abstract—Admittedly, because of how busy doctors are nowadays, they tend to scribble unreadable prescribed medicines which leads to the problem of misinterpreting medicine names. Patients are sometimes curious to know information about their prescribed medicines before purchasing them. Recently, developers have been searching for a method to address this problem efficiently but, no technique leads to full recognition of medicine names due to the bad handwriting of doctors and its variety so that leads us to machine learning where the system will learn various types of handwritings for the same medicine to be able to recognize new handwritings. This paper proposed a system that presents a solution for both the pharmacist and the patient through a mobile application that recognizes handwritten medicine names and returns a readable digital text of the medicine and its dose. The System identifies the medicines' names and the doses for the collected data set with some pre-processing techniques like image subtraction, noise reduction, and image resizing. After that, the pre-processed images will undergo some processing as it will be classified and feature extracted through Convolutional Neural Network and finally Optical Character Recognition technique applied on the medicines with low accuracy in the post-processing phase to identify their names by comparing the result with the dataset containing all the medicines. This will help in diminishing the instances of distortion of medication names, assisting pharmacists in limiting their doubts. The proposed system tested on different real cases, and accuracy has reached 70% using (CNN) model.

2.2.1 Pre-processing Phase

in the pre-processing phase, firstly scanned prescriptions' images enter the system as an input using the mobile camera and the extension type of the image is set to PNG. Secondly, the size of the image is normalized by cropping white spaces and converting the image into black and white. Thirdly, the morphological operation technique is applied on the image to make all the images of the same size based on a comparison of the corresponding pixel in the input image with its neighbors. Finally, the cropping operation is applied to crop the prescription into 3 parts. The first part is from the beginning of the prescription till (R/) symbol which includes the name of the doctor as shown in figure 4. The second part which starts right after the (R/) symbol that includes the prescribed medicines, doses, and instructions that is the most important part as the system will identify the medicine and the dose from this part as shown in figure 5. In this case, the prescribed medicines and the doses will be trained with the medicines in the system based on the doctor's major which is identified from part one. The third part, which is the least important part of the system as it contains the doctor's phone numbers and the hospital or doctor's clinic addresses as shown in figure 6. The block diagram shown in figure 7 shows the main steps of the pre-processing phase.

Fig. 4: Header

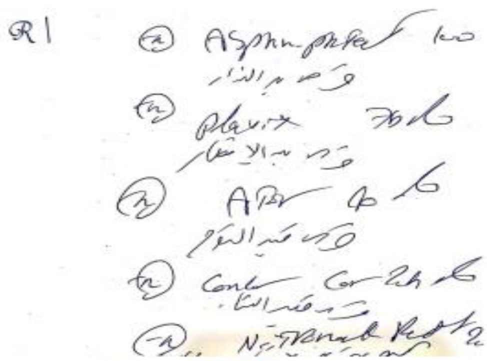


Fig. 5: Middle part

Fig. 6: Footer

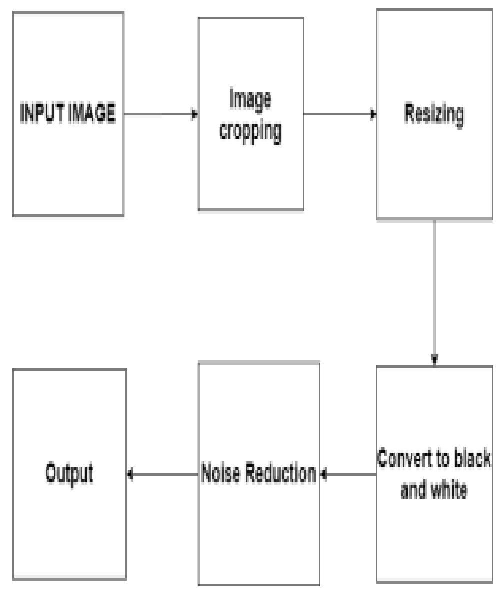


Fig. 7: Pre-Processing Diagram

2.2.2 Processing phase:

After the pre-processing part is done, on the middle that includes the prescribes medicines as shown in figure 9 then will be classified, and the feature extracted by the Convolutional Neural Network (CNN) using backward and forward propagation technique, CNN performs two tasks which are feature extraction and classification to correctly classify images. It consists of multiple layers which are categorized under feature extraction and classification in which the Convolutional layer, Relu layer, and Maxpooling layer are known as feature extraction techniques that are applied on the input image, and then the fully connected layer is applied for classification and output the image as shown in the figure 8.

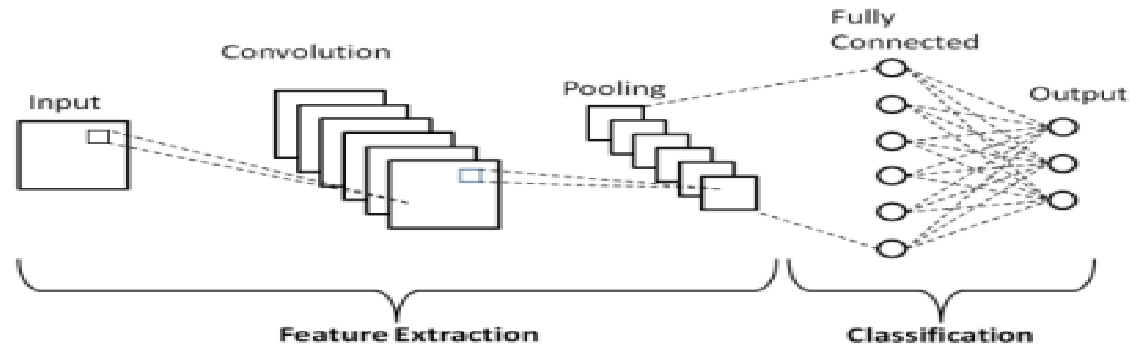


Fig. 8: CNN Layers

using CNN as feature extractor and classifier for the medicines in the prescription. The dataset is collected from multiple doctors and hospitals with varying specializations, our main aim is to collect numerous different prescriptions of each medicine with different handwritings. The dataset has been divided into 70% training and 30% testing to train proposed model. The medical prescription is divided into 3 main parts, the first section until the R/ includes the name and the specialization of the doctor which will help in classifying the medicine according to the doctor's specialization, the second part which starts after the R/ which includes the handwritten prescribed medicines which will be classified and lastly the third part which is the footer includes the addresses and contact numbers of the hospital or clinic will be eliminated. There are various types of data sets as shown in the figures 10.

Firstly, we are going to start the convolution step, which is known as the feature extraction step, it includes the input image, a feature detector, and a feature map, secondly the filter is taken and applied pixel block by pixel block through the multiplication matrix to the preprocessed middle image, so the feature map is filled or completed. Many feature maps are created to get our first Convolutional layer. Secondly, we are going to create an edge detection filter using the Sobel operation. Thirdly, the Rectified Linear Unit (ReLU layer) is another step to the Convolution layer as an activation function is applied to the feature map to increase the nonlinearity in the network. Fourthly, to achieve spatial variance, we use the maxpooling technique to gradually reduce the input representation size as it makes it easier to detect and identify objects wherever they are located inside the image. Not only does pooling aid in reducing the amount of processing and the number of required parameters required but also, it controls the issue of overfitting. Finally, the pooled feature map is flattened into a sequential long vector to allow the information to enter the input layer in the ANN to be furtherly processed.

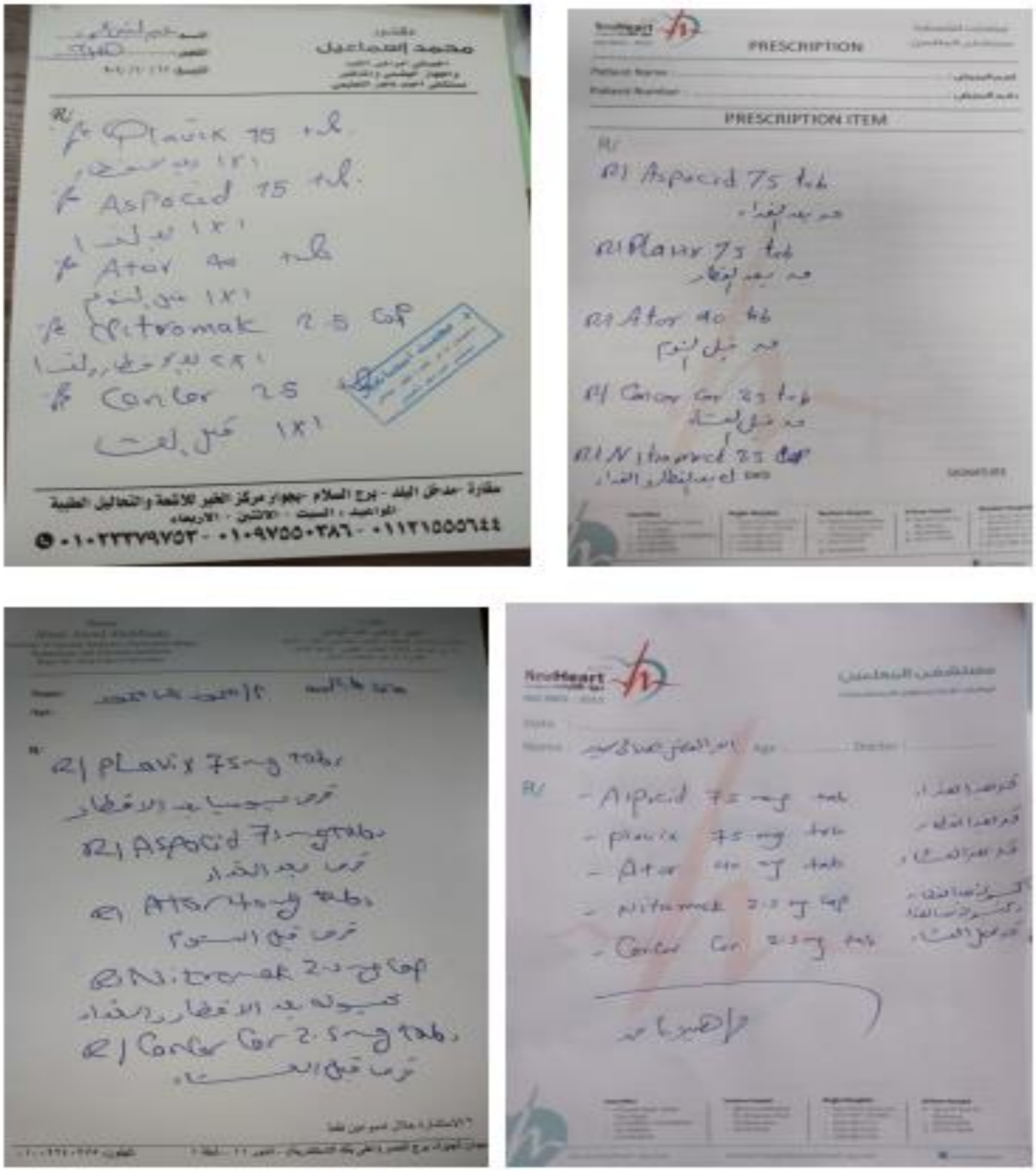


Fig. 10: Prescription templates

2.2.3 Post-processing

The performance and the accuracy of the model, A more handwritten medical prescription will be collected to reach higher accuracy. Also more classification techniques like Optical Character Recognition (OCR) will be applied on the resulted medicines if accuracy is 50% or less to process character by character, comparing the OCR result with a data set contains all the medicine names to recognize which medicine in the dataset nearest to the result.

-The Problem that we faced that we didn't find big dataset to launch this model so we agreed to try on other way to build an model using transfer learning cause that the best solution for the smallest dataset.

- Transfer learning is a machine learning method where a model developed for a task is reused as the starting point for a model on a new problem. It is particularly popular in deep learning because it can train deep neural networks with comparatively little data, making it useful in the data science field. Transfer learning focuses on applying knowledge gained while solving one task to a related task. For example, knowledge gained while learning to recognize cars could be applied when trying to recognize trucks.

3 Pre-Processing (phase)

We used OpenCV to read the image (prescription) and let the maximum width be 1000 because of speed.

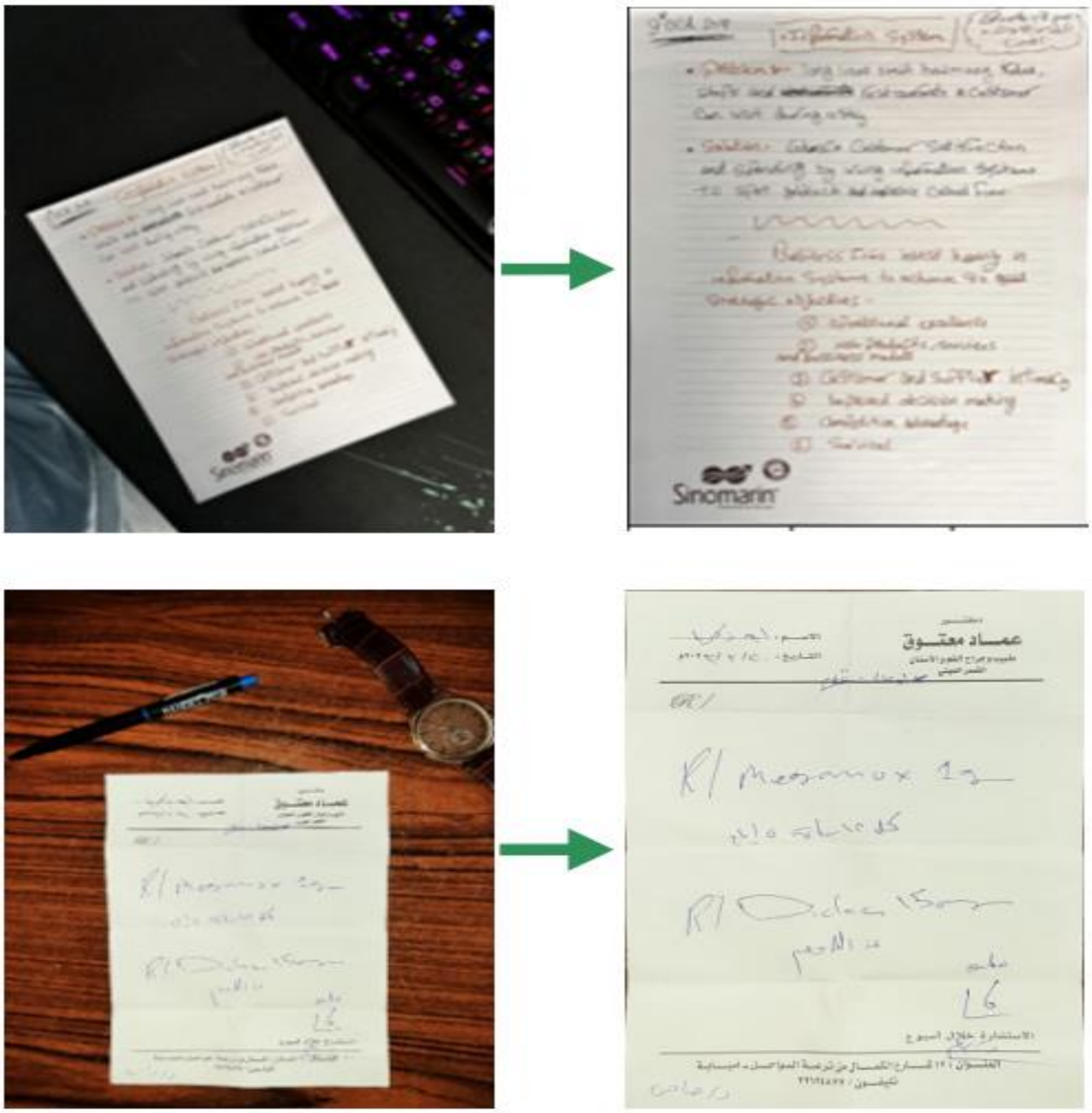
After that, we separated the background from the foreground (text) by threshold function, which is separated by a fixed number from 0 to 255 of a gray color degree.

4 Reading Prescription Problems

4.1 Warp Perspective (objects appear outside the frame of the paper)

A challenge arose, which is that in some pictures, some objects appear outside the frame of the paper, and therefore threshold also happens to them, and since they differ in color from the paper, this affects the separation process, Therefore, they used the canny method to define the contours of the paper by drawing the borders of the shapes in the image, and then defining the corners of the paper by the bigger size of contour and cropping the image on the same area of paper

• Warp Perspective •



(Objects appear outside the frame of the paper)

4.2 Image enhancement (Can’t use fixed threshold only)

After that, a new challenge appeared while using a fixed threshold, which is the change of gray color degree from one picture to another because of the lighting of the photograph and the size and color of the writing line

To solve this problem, we used an adaptive threshold for several reasons:

1. Adaptability to Varying Illumination: One of the primary reasons for the success of adaptive Threshold over fixed thresholding is its ability to handle images with varying illumination conditions. Fixed thresholding applies a single threshold value to the entire image, which can lead to inaccurate results when the lighting conditions change across different parts of the image. Adaptive thresholding, on the other hand, adjusts the threshold value locally based on the pixel intensities in the vicinity, allowing it to handle variations in illumination more effectively.

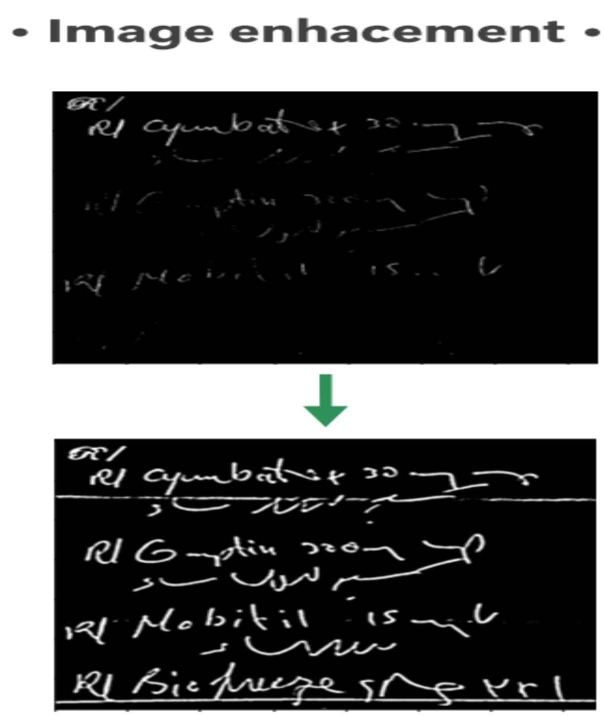
2. Handling Non-Uniform Backgrounds: Images with non-uniform backgrounds pose a challenge for fixed thresholding methods. These methods often struggle to find an appropriate threshold value that can accurately separate the foreground from the background. Adaptive thresholding, by adapting the threshold value locally, can better handle non-uniform backgrounds and achieve more accurate results.

3. Robustness to Noise: Adaptive thresholding algorithms are generally more robust to noise compared to fixed thresholding methods. When applying a fixed threshold to a noisy image, the noise can interfere with the thresholding process and lead to false detections or inaccurate segmentation. Adaptive thresholding techniques incorporate noise reduction mechanisms

or statistical analysis within the local windowing approach, allowing them to mitigate the impact of noise and produce better results.

4. Localized Thresholding: Adaptive Threshold algorithms divide the image into small local regions or windows and compute the threshold value independently for each window. This localized approach enables the algorithm to adapt to the local characteristics of the image, such as texture, intensity variations, or object boundaries. By considering the local context, adaptive thresholding can achieve better segmentation results, preserve finer details, and handle complex image structures more effectively.

5. Increased Flexibility: Adaptive thresholding provides more flexibility in selecting the thresholding method based on the image characteristics or application requirements. There are various adaptive thresholding techniques available, such as mean, median, Gaussian, or Sauvola thresholding, each designed to handle specific image conditions. This flexibility allows users to choose the most appropriate thresholding method for their specific application, resulting in improved performance and accuracy.



- Cannot use fixed thresholds only.
- Thus, we solve the problem of exter
  factors affecting the image.

5 Segmentation of the interested region

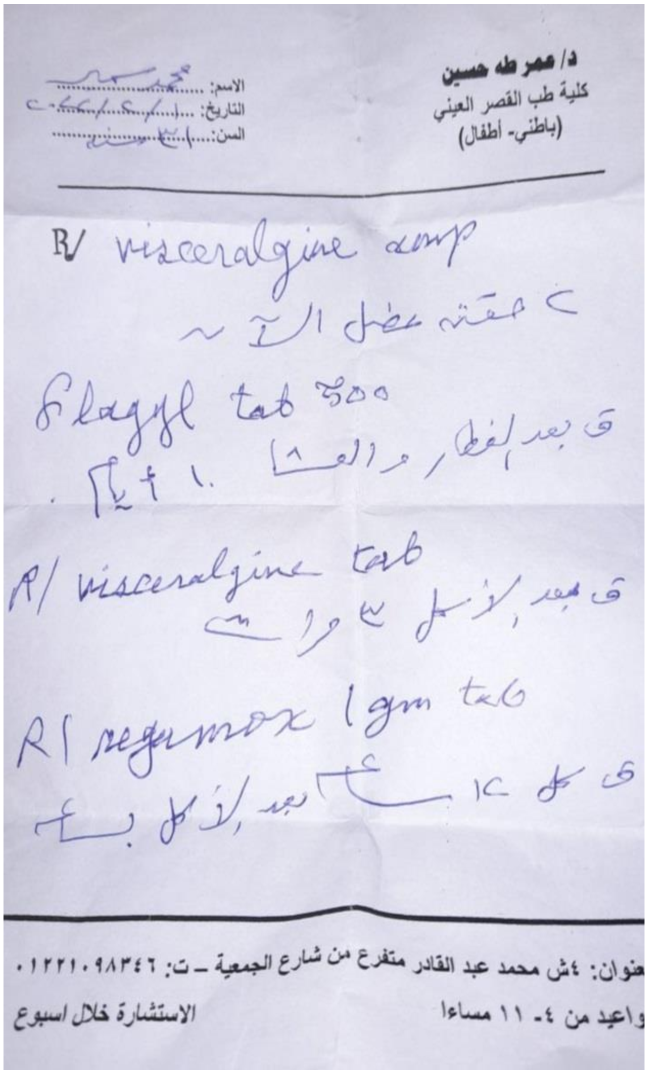
5.1 Remove Header and Footer of Prescription

We had to remove the header and footer from the paper in order to reduce the model's distraction from reading unimportant words.

To remove the header and footer we made dilation with very small Kernel of ones to accentuate features and increase the object area, and we added Gaussian Blur and Threshold to improve the accuracy of text to help with the dilation process, and we used the FindContours function to define the contours of the texts, then put each Contour whose width is greater than 80% of paper's width to the list because we sure that is no word his width is greater than 80% of paper's width, so it is a horizontal line.

Then we cut paper from up to the first line in the list and from down to the last line in the list because of could be lines in the paper another header and a footer.

Remove Header & Footer



د/ عمر طه حسين
كلية طب القصر العيني
(باطني- أطفال)

الاسم:
التاريخ:
السن:

R/ visceralgine amp

Flagyl tab 500

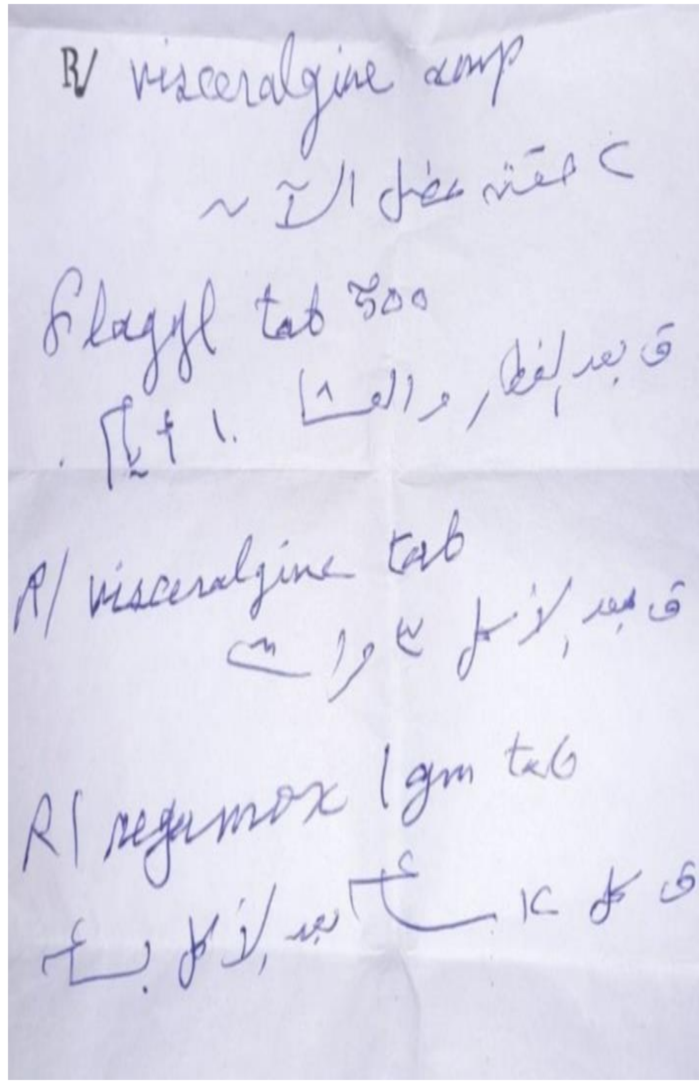
R/ visceralgine tab

R/ regamox 1gm tab

عنوان: ٤ش محمد عبد القادر متفرع من شارع الجمعية – ت: ٠١٢٢١٠٩٨٣٤٦

واعيد من ٤- ١١ مساءا

الاستشارة خلال اسبوع



R/ visceralgine amp

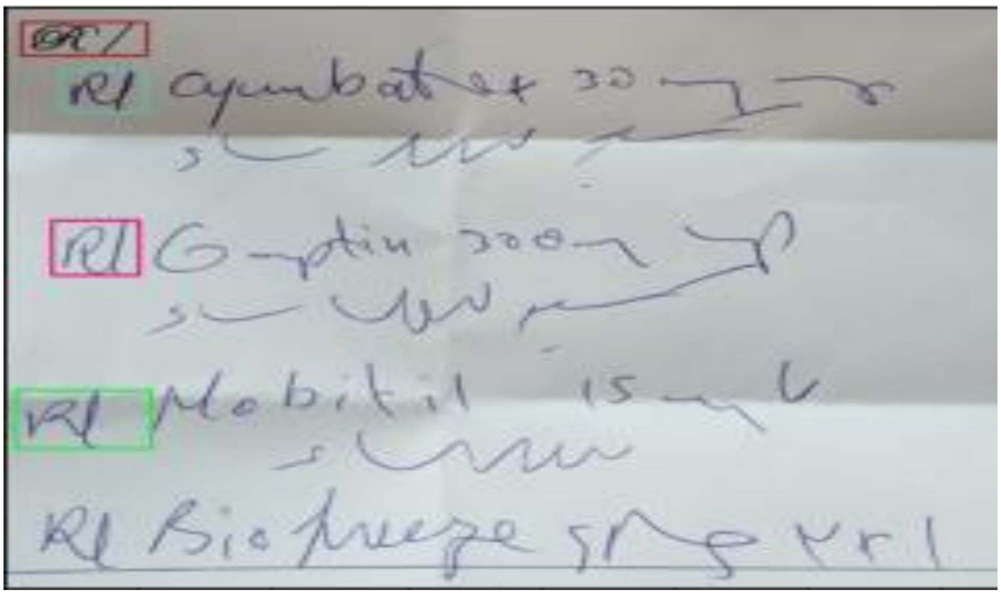
Flagyl tab 500

R/ visceralgine tab

R/ regamox 1gm tab

5.2 Remove “R/” symbol

During the test, we found that in some images, the "R\" symbol was written indicating something, but it was not included in the name of the medicine, so this became the new challenge.



After much thought, we used the idea of Character detection through the CNN to identify the letter "R", so we used a pre-train model on the character recognition, but the idea failed because Often the "R" character is combined with "\" so that it cannot be recognized correctly

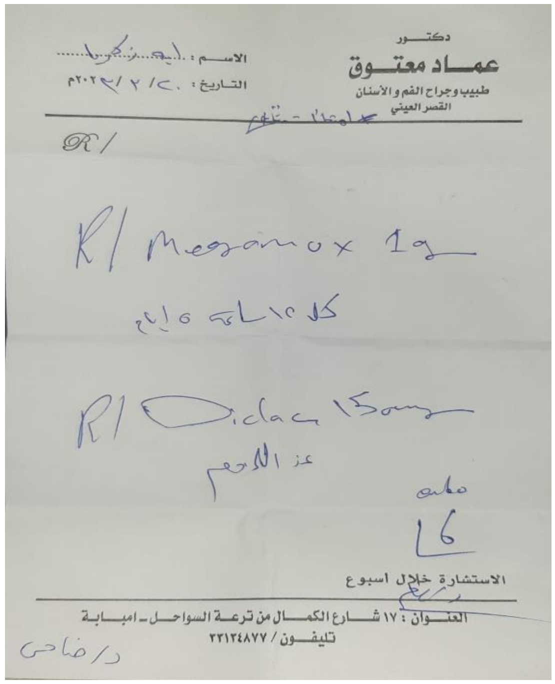
Instead of taking much time and resources to make a specific dataset, we thought that there is no medicine name consisting of two characters or one character, so we made dilation with a very small kernel to recognize small areas, but it will search only in the first third of the image because the "R\" will be at the beginning of the line only

We also found that the image that contains "R\" at the beginning of each line has the "R\" printed at the beginning of the paper, so we took the space of the printed symbol, and then we will take only the "R\" in the first third of the paper, and its area does not exceed twice the area of the printed symbol and the distance between it and the text that followed by within the same limits as the mean distance between words

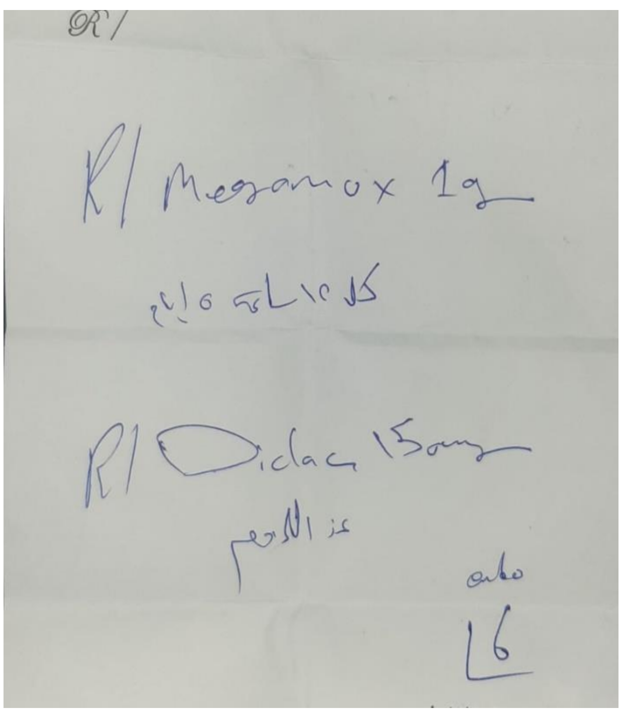
In this way, we have also solved part of the dispersion on the model as well, and we have reduced the problem of merging the words that we will display in the next slide.

In addition, here we have come to a significant part of the pre-processing, we have removed some of the things that could distract the model, and now we have only the important part of the medical prescription, and we are ready to start the part of identifying the names of the drugs, starting from the slide after the next

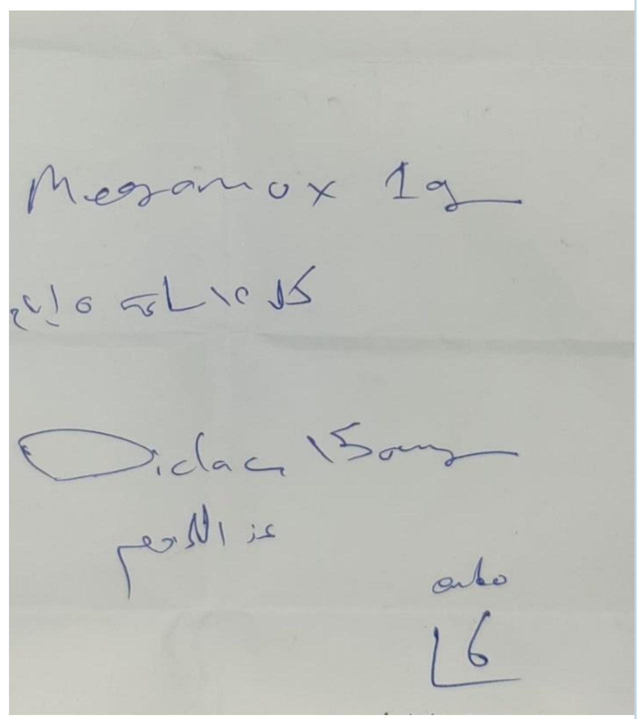
Segmentation of the interested region



a) The original prescription



b) Remove Header and Footer



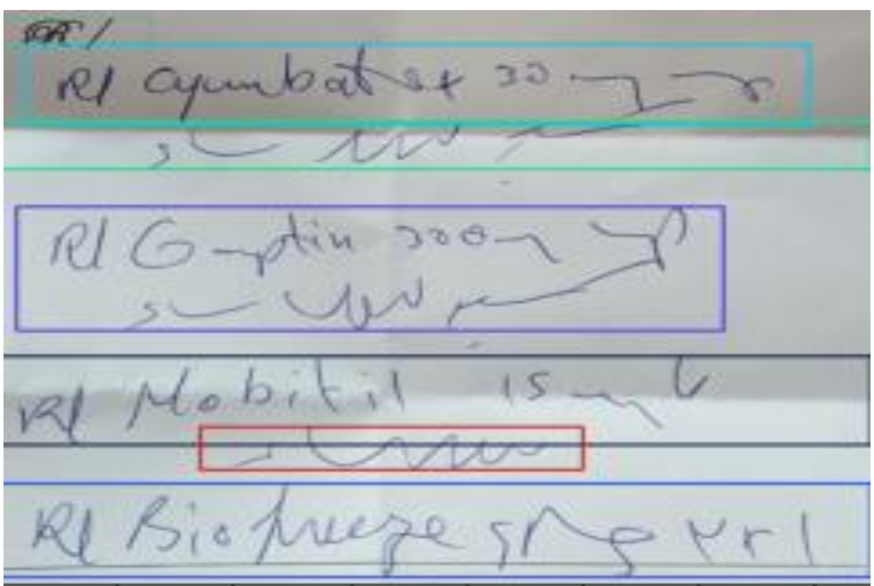
c) Remove "R/" Symbol

6 Segmentation of text

Then we make dilation with Kernel of ones to accentuate features and increase the object area, and we added Gaussian Blur to improve the accuracy of text to help with the dilation process, and we used the FindContours function to define the contours of the texts

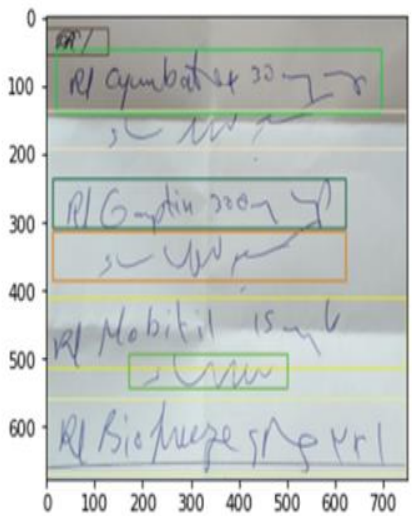
Thus, we have entered the line segmentation stage, so we used a large kernel to define each line in the image and define each line with a frame by the contours

We faced a new challenge, which in some pictures puts borders on two lines as if they were one line, and this is because of the difference in the distance between each line and length of each line

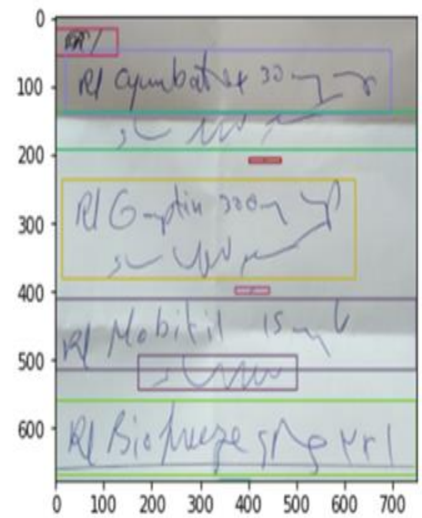


And we tried to solve it with a set of steps, the first of which is the work of function calculating the average line length and function calculating the average spacing between the lines, and we stipulated that each line should not exceed twice the mean length of the lines, and if this happened, then we would divide the line length into two halves, but we found that in some other images, the name of a medicine is written larger than the rest of the medicine in the paper.

In this case, the line length is greater than the Double mean length of the lines, the line is divided into two lines, so we overlooked the idea of dividing the line into two lines so as not to lose some medicine names, as it will also identify the words inside each line through the word segmentation

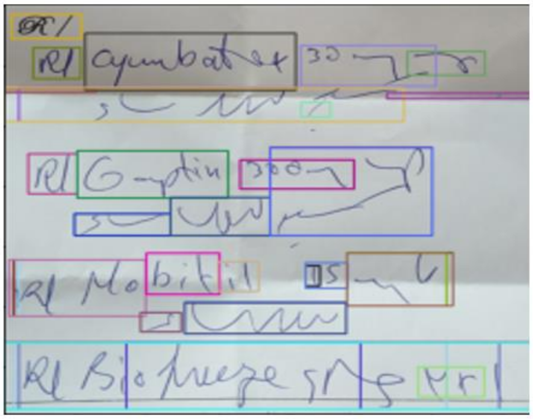


Without merged rows

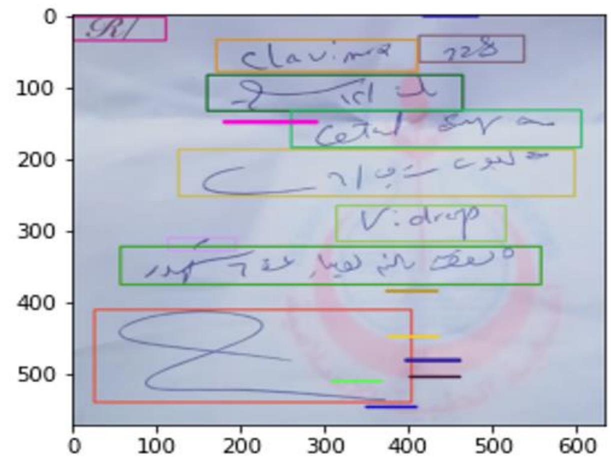


With merged rows

After that, we made dilation by a small kernel inside each line in the image to define each word in the lines to avoid merging of words, and we put borders around each word and we removed noise below the mean word length and cut the paper to images of word

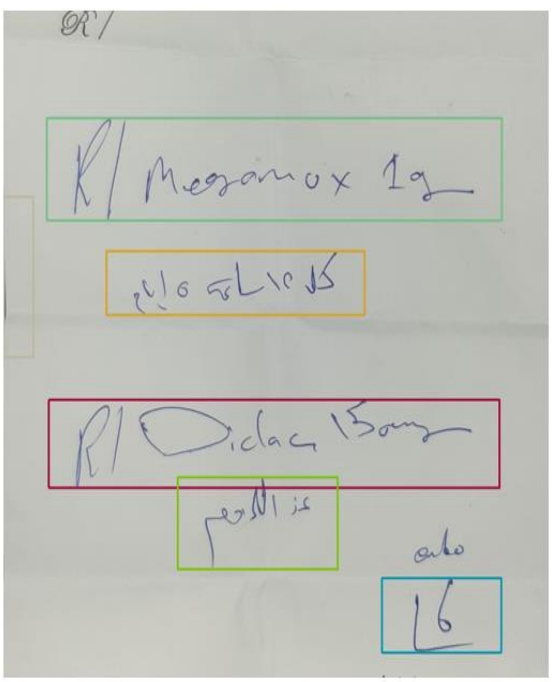


Overlap

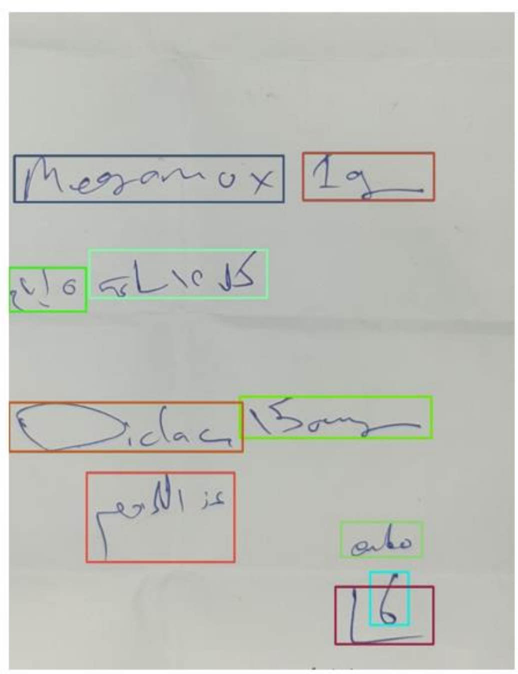


Noise

Moreover, we are almost done with pre-processing's part, we removed the noise, we identified all the words in the picture, we put borders on them, and we are ready to recognize the name of the drug that the patient chooses from the picture



Perfect Line Segmentation



Perfect Word Segmentation

7 Implementation Experimental Setup, & Results

7.1 character recognition

7.1.1 Dataset for character recognition:

We used a dataset named 'Handwritten Medical Term Corpus'

7.1.1.1 Dataset profile:

There are 480 medical words (360 English and 120 Bangla) in the 'Handwritten Medical Term Corpus'. These words are chosen based on the number of appearances in 8324 Bangladeshi prescriptions. The handwritings are collected from 39 medical professionals and doctors of Bangladesh. Due to receiving incomplete data from 12 data providers, 1,289 samples are missing in the dataset. Hence, the dataset has 17,431 handwritten instances of 480 medical related words. All the data were collected by maintaining authenticity, security, and privacy of the data providers, and the experiments were performed in accordance with relevant guidelines and regulations.

- we filter dataset and remove Bangla words.

7.1.1.2 Prepare dataset:

- Roboflow:
  - The Roboflow web application helps you to build better computer vision models, faster.
  - Available functionality includes : creating projects, dataset upload, labeling workflow management, exporting datasets.

- We upload 1003 images from our dataset to roboflow and select every letter carefully and then select the label.

7.1.2 Character recognition using YOLOV5

7.1.2.1 1st Trail:

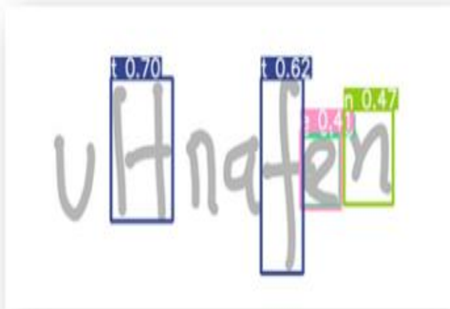
Preprocessing: Grayscale Applied

Augmentation:

1. Output per training example: 3
2. Grayscale: Apply to 67% of images

	Training Set	**Validation Set**	**Testing Set**
Images	**732**	**116**	**155**
%	**73%**	**12%**	**15%**

The next section represents random images with the description.



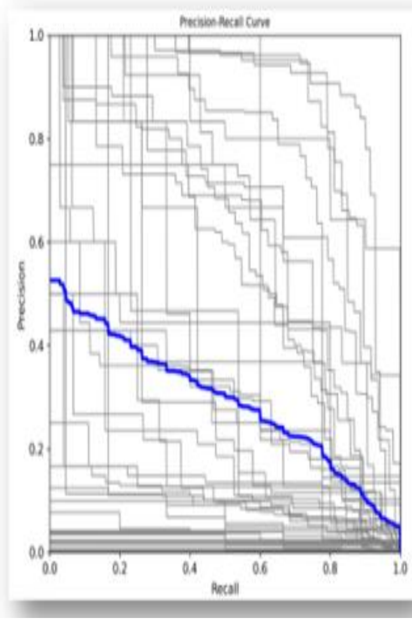
• Duplicated Letters Detection •

The model detects two litters together (lt) as (t).

• Noraml Letters Detection •

The model detects all the letters rightly with no problem found

Accuracy



This image represents the PR-Curve considered by 29.4% Accuracy.

7.1.2.2 2nd Trail

Preprocessing:

Auto-Orient : Applied

Resize: Stretch to 640x640

Grayscale: Applied

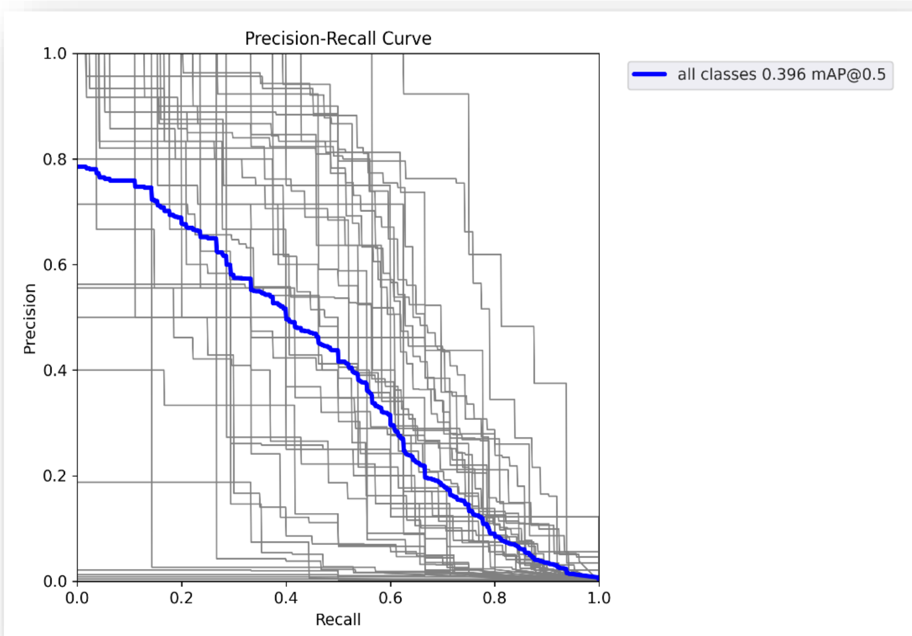
Tile: 2 rows x 2 columns

Modify Classes: 0 remapped, 11 dropped

Augmentation:

No augmentations were applied.

Same Problems that represented before but the Accuracy Increased.



represents the PR-Curve considered by 39.6% Accuracy.

Same Problems that represented before but the Accuracy Increased.

	Training Set	Validation Set	Testing Set
Images	**2800**	**412**	**800**
%	**70%**	**10%**	**20%**

7.1.2.3 3rd Trail:

Preprocessing:

Auto-Orient : Applied

Resize: Stretch to 640x640

Grayscale: Applied

Tile: 2 rows x 2 columns

Modify Classes: 0 remapped, 11 dropped

Augmentation:

Outputs per training example: 3

Grayscale: Apply to 67% of images

Hue: Between -69° and +69°

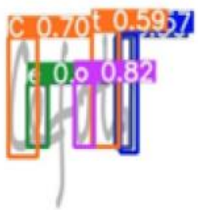
Saturation: Between -25% and +25%

Exposure: Between -25% and +25%

Bounding Box: Rotation: Between -7° and +7°

Bounding Box: Brightness: Between -31% and +31%

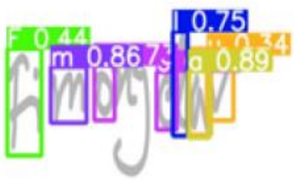
	Training Set	**Validation Set**	**Testing Set**
Images	**2500**	**84**	**69**
%	**94%**	**3%**	**3%**



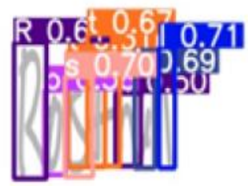
• Missing Letters Detection •

There are missing detect but not like at previous trails

• Overlapping Letters Detection •



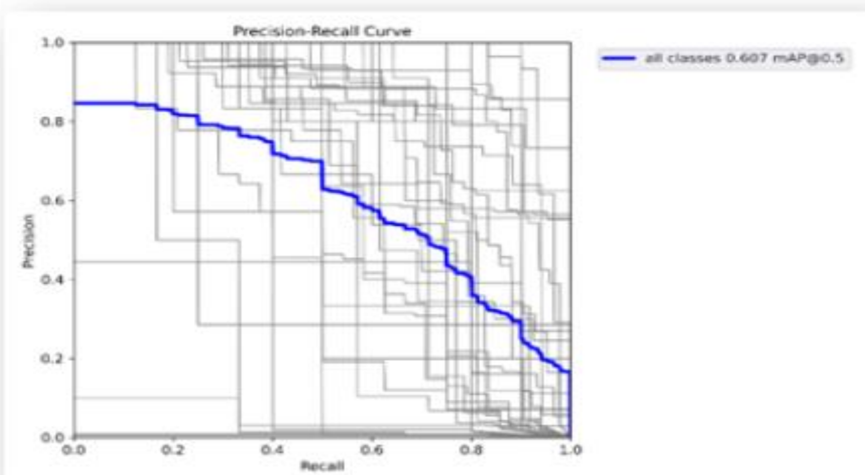
• Duplicated Letters Detection •



• Normal Letters Detection •

hard detection as the letters so closely and detected

Accuracy



This image represents the PR-Curve considered by 60.7% Accuracy.

7.2 Word Recognition

7.2.1 Dataset for word recognition

- In word recognition, we write the data set manually because the data set used for character recognition, the frequency of a single word is very small (12 to 17)
- No other data set was found to contain a doctor's prescription.
- so we decide write our dataset.

7.2.1.1 Dataset profile:

- **First** we write five medicines (Panadol, Ketolac, Cataflam, Catafast, Brufen) written by only three people, The total number of images was 1043 images.
- **Second** our friends helped us write data with us and increase the number of data.
  Therefore, the number of images became 2130, distributed over 5 classes, which means about 430 images for each drug,
- **Third** we both increased the number of classes and wrote 2 additional classes to become 7 classes ('Cataflam', 'Ketolac', 'Brufen', 'Panadol', 'Actos', 'Diclac', 'Insulin'), this time taking into account the standards for writing dataset.

  Therefore, the total data became 2861, or 510 images for each drug, as in the previous scenario, written by the same nine people and the same structure.

7.2.2 word recognition using model CNN

We built the CNN model from scratch, but it was not the right solution because of the small of dataset, which is 1043, and the number of layers was small, because the model consumed large resources during learning,

so it took about 6 hours to learn, and because of the lack of data, overfitting happened, so the accuracy reached 100%, and so on.

One of the reasons for our thinking in the direction of transfer learning

7.2.3 word recognition using VGG 16

7.2.3.1 Transfer Learning

Why use transfer learning?

Transfer learning is a machine learning technique where a pre-trained model, which has been trained on a large dataset, is used as a starting point for solving a related but different task. Instead of training a model from scratch, transfer learning allows us to leverage the knowledge and features learned by the pre-trained model on a different task and apply it to our specific problem.

VGG16 (Visual Geometry Group 16) is a popular pre-trained convolutional neural network (CNN) architecture for image classification. The Visual Geometry Group at the University of Oxford developed it. VGG16 consists of 16 layers, including convolutional layers, max-pooling layers, and fully connected layers.

The advantages of using VGG16 in transfer learning include:

Powerful feature extraction: VGG16 has shown excellent performance in extracting meaningful features from images. The network's deep architecture allows it to learn and represent complex patterns and structures present in images effectively.

Generalization capabilities: Due to its architecture, VGG16 has been trained on a large-scale dataset, enabling it to learn a rich set of features that can generalize well to various visual recognition tasks. This makes it a suitable choice for transfer learning across different image classification problems.

Availability of pre-trained weights: Pre-trained weights for VGG16 are readily available in popular deep learning frameworks like Keras and PyTorch. These pre-trained weights save significant training time and computational resources, as you can directly use them as a starting point for your task.

Ease of fine-tuning: VGG16's modular architecture makes it easier to modify and fine-tune for specific tasks. By freezing certain layers and only training the last few layers or adding additional layers on top, you can adapt the pre-trained VGG16 model to your specific problem with relatively fewer training samples.

Wide adoption and community support: VGG16 has been widely used and studied in the deep learning community, resulting in extensive documentation, code examples, and online resources. This availability of knowledge and support makes it easier to implement and troubleshoot when using VGG16 for transfer learning.

7.2.3.1 VGG-16 Model

The previous challenges that we mentioned were not the last challenges, but while working on this transfer-learning model, many challenges appeared, so I will present these challenges in the form of scenarios, but first we will talk about the common things in all scenarios

Architecture: VGG16 is known for its simplicity and deep structure, consisting of 16 layers, including convolutional layers, max-pooling layers, and fully connected layers. Here is a summary of the VGG16 architecture:

Input layer: The input to the network is an image of fixed size, typically 224x224 pixels.

Convolutional layers: VGG16 starts with a stack of convolutional layers, each using small 3x3 filters with a stride of 1. The number of filters increases as we go deeper into the network, starting from 64 filters and doubling after each max-pooling layer. These layers are responsible for learning hierarchical features from the input image.

Max-pooling layers: After each stack of convolutional layers, max-pooling layers with 2x2 filters and stride 2 are applied to reduce the spatial dimensions and extract the most prominent features.

Fully connected layers: The last few layers of VGG16 are fully connected layers. These layers take the flattened features from the previous layers and map them to the desired number of output classes. The fully connected layers are followed by a softmax activation to produce class probabilities.

Activation function: Throughout the network, rectified linear units (ReLU) are used as the activation function, which helps introduce non-linearity and allows the model to learn complex relationships in the data.

Dropout: VGG16 also incorporates dropout regularization, specifically after the fully connected layers, to prevent overfitting and improve generalization performance. Dropout randomly sets a fraction of the input units to 0 during training.

The VGG16 architecture is characterized by its deep structure, with multiple stacked convolutional layers and relatively small filter sizes. This design allows the network to learn more complex and abstract features from images. However, the large number of parameters in VGG16 makes it more computationally expensive to train compared to shallower networks.

VGG16 has achieved impressive performance on various image classification benchmarks and has become a popular base model for transfer learning due to its strong feature extraction capabilities and generalization performance.

First scenario

In the beginning, we started training the model with five classes

(Panadol, Ketolac, Cataflam, Catafast, Brufen) written by only three people. The total number of images was 1043 images, divided into approximately 230 images for each drug, taking into account the balance of data and its distribution equally among all classes.

As for the structure, we gave another layer just to be trained on our data.

Then we started training the model, after 500 epochs, The Training accuracy became 99%, training loss = 4.6%, test accuracy = 95.9% and test loss = 9.3%

The challenge in the first scenario is overfitting on the data, and although it is one of the advantages of the VGG16 model to handling the problems of overfitting, it happened and this reason is due to the lack of the data.

Second scenario

In this scenario, we tried to solve the problem of lack of data, so our friends helped us write data with us and increase the number of data.

Therefore, the number of images became 2130, distributed over 5 classes, which means about 430 images for each drug, twice the number of images in the previous scenario, but they were written by 9 people, and we did not change anything in the structure of the model.

<u>But</u> the new challenge in this scenario was the opposite of the challenge in the previous scenario, so under fitting was our problem at this time. Despite the increase in the amount of data, the data accuracy was 79.3%, and testing accuracy was 77%

The last two Classes were always predicted wrong, no matter what the name of the last two Classes was, and no matter how much we tried to replace them

So we started thinking about what might be the cause of this problem, so we started to change the transfer learning model from VGG16 to MobileNet, but there was no significant difference, so the old model was not the obstacle.

Third scenario

On the next one, we thought the hitch was in the architecture, so we changed part of it and added 5 hidden layers, but that change was not right, and instead of increasing efficiency, the training accuracy became 77.8%, and testing accuracy 73.5%

Fourth scenario

Therefore, we returned the structure as it was and made sure that the error would be in the dataset

Nevertheless, in order to verify this, we reduced the number of classes to three classes, so the total number of data became 1276 images written by nine people, they were according to our correct expectation, and the training accuracy became 95.5%, training loss 16.1%, testing accuracy 96.8% and testing loss 14%

In addition, the new challenge became the small number of classes

Fifth scenario

So after we made sure that the obstacle was in the dataset, it was necessary to determine where the error was first in order to correct it, so we divided the data of each of the nine people separately in order to test each of them separately and know which one was wrong

After testing the data, we found the error in one person, and we were not thinking about the criteria for writing the dataset, the error was that one of the colleagues had written the medications with a pencil, and when photographing, the flash was lit, so the pen line was close to the color of white, but it can be distinguished by the human eye, but in the model when converting The image to Gray could not be recognized by the model, and therefore the training was weak

Models	Train Acc	Train Loss	Test Acc	Test Loss	Notes
Model 8	95.9%	17.6%	91.4%	32.5%	best
Model 4	99.4%	2.6%	88.7%	54.1%	OverFit
Model 5	97.8%	8.0%	88.2%	51.5%	
Model 7	97.8%	14.5%	87.3%	56.4%	
Model 1	92.2%	30.8%	86.9%	53.8%	
Model 2	99.0%	4.3%	82.4%	77.7%	OverFit
Model 3	99.7%	1.7%	74.7%	97.0%	OverFit
Model 6	99.7%	2.2%	63.0%	161.8%	OverFit

Data test results

Sixth scenario

After correcting the corrupted data, we have about 2,550 images, out of about 510 images for each drug, distributed unevenly over five chapters and written by nine people. The final output was as follows:

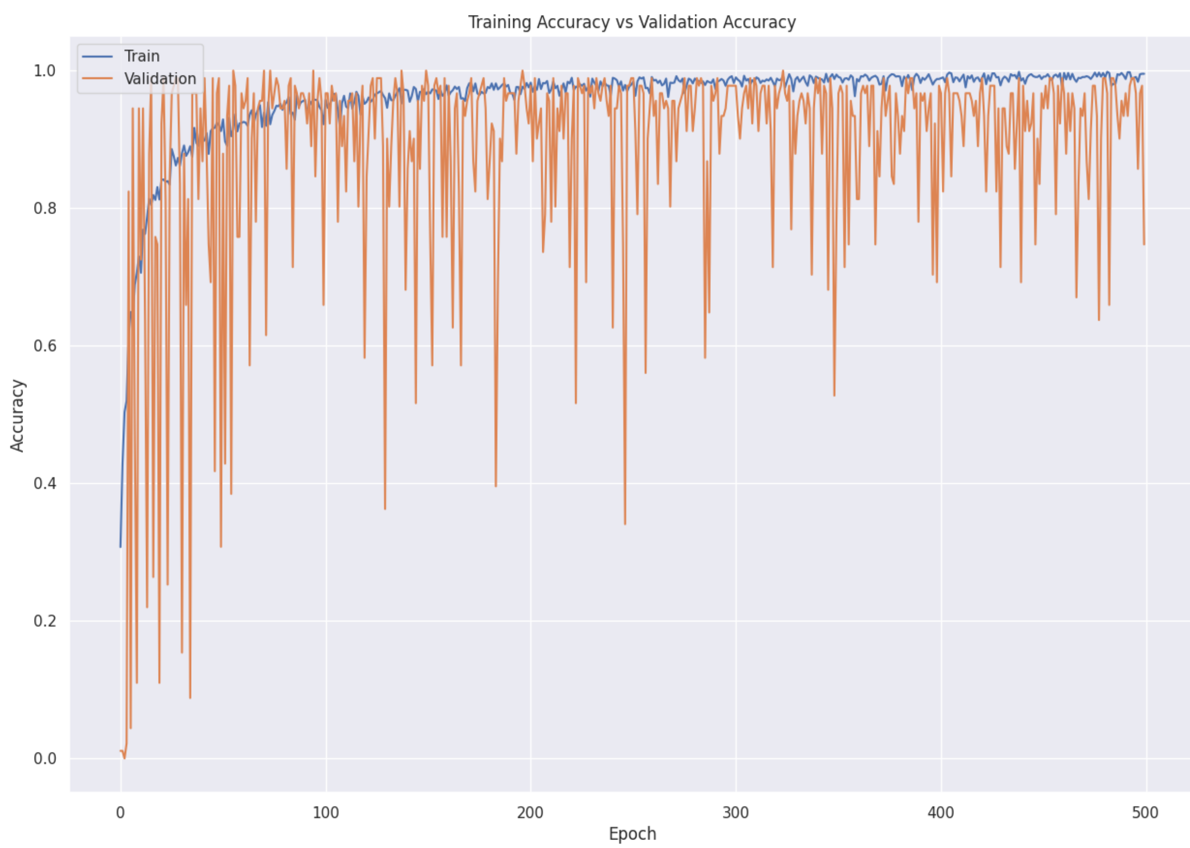
Training Accuracy = 98.6%

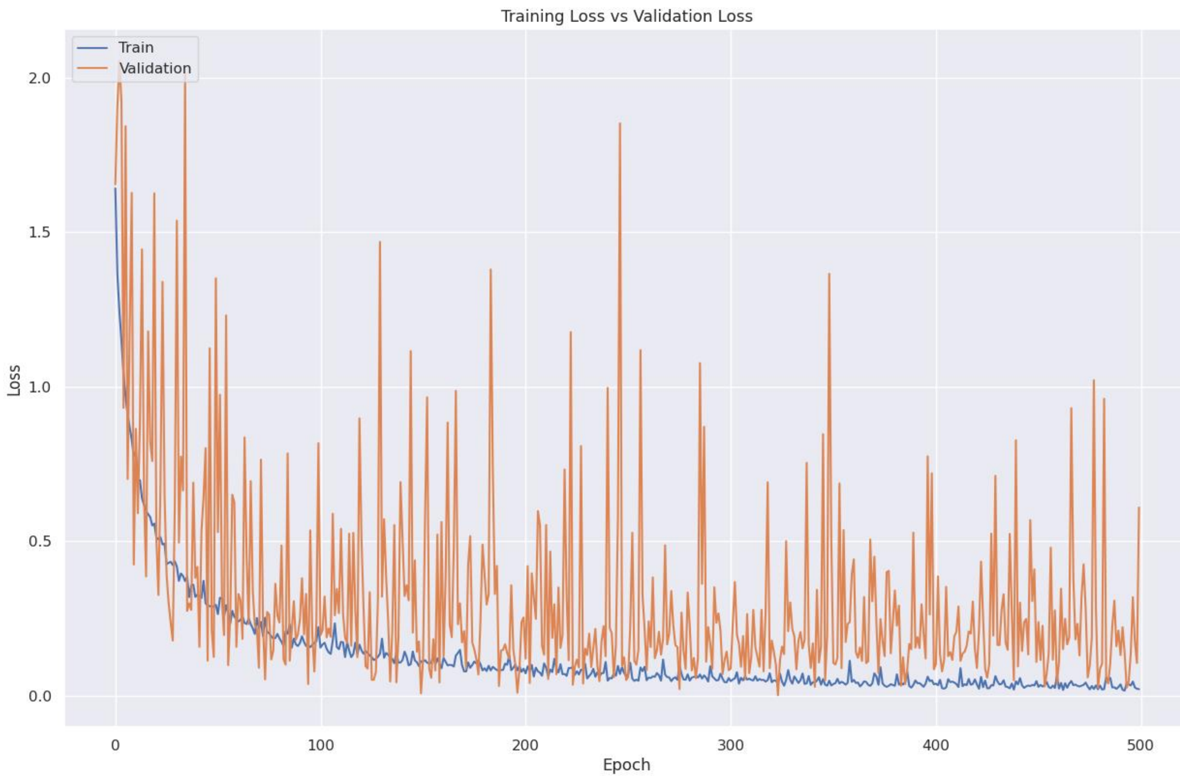
Training loss = 4.8%

Testing Accuracy = 96%

Testing loss = 19%

Noting that about twenty images written in the hands of people who are not among the previous nine people were tested, and the model predicted all of them correctly.





Seven scenario

After we succeeded in predicting the five classes successfully, we both increased the number of classes and wrote 2 additional classes to become 7 classes ('Cataflam', 'Ketolac', 'Brufen', 'Panadol', 'Actos', 'Diclac', 'Insulin'), this time taking into account the standards for writing dataset.

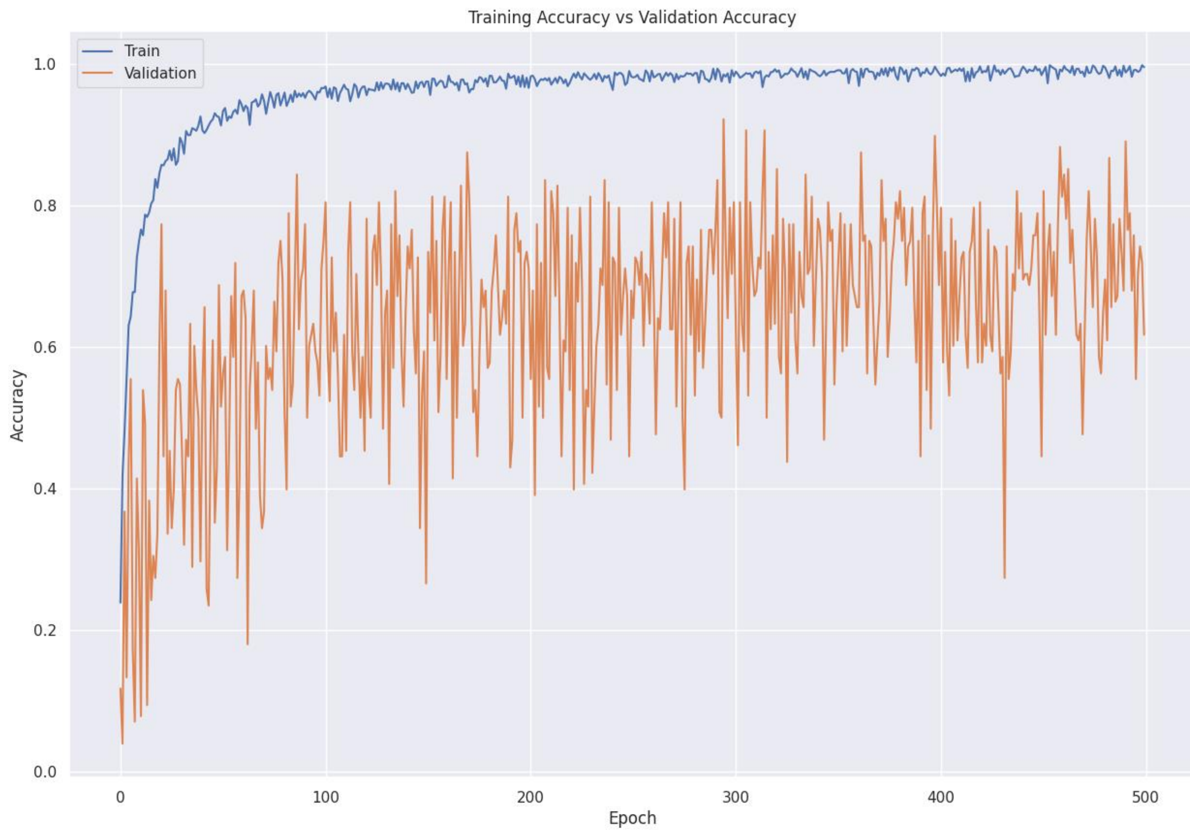
Therefore, the total data became 2861, or 510 images for each drug, as in the previous scenario, written by the same nine people and the same structure, and its efficiency was as follows:

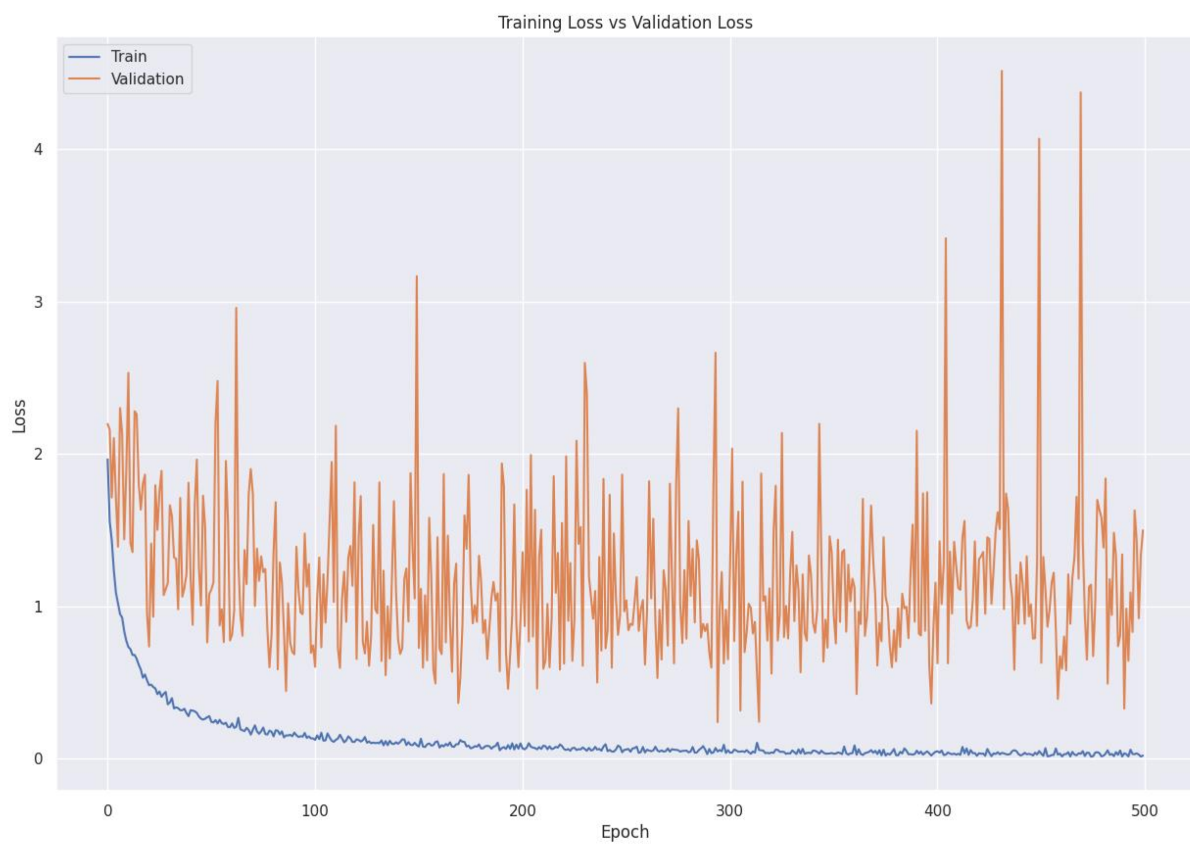
Training accuracy = 97.2%

Training loss = 10.3%

Testing accuracy = 94.7%

Testing loss = 18.5%


Training Accuracy vs Validation Accuracy
Train
Validation
Accuracy
Epoch
1.0
0.8
0.6
0.4
0.2
0.0
0
100
200
300
400
500


Training Loss vs Validation Loss
Train
Validation
Loss
Epoch
4
3
2
1
0
0
100
200
300
400
500

8 Application

8.1 Abstract for application

The advancements in technology and artificial intelligence (AI) have revolutionized various industries, including healthcare. In this paper, we present a smart pharmacy application that utilizes AI to scan prescriptions through a camera and extract relevant information regarding prescribed medicines. By leveraging computer vision algorithms and natural language processing techniques, our application aims to improve the efficiency and accuracy of medication management in pharmacies.

To ensure the highest level of accuracy, our application employs robust AI algorithms that continually learn and improve from a diverse set of prescription data. By leveraging the power of AI, the application becomes adept at recognizing different handwriting styles, adapting to various prescription formats, and handling potential ambiguities in prescription information. This adaptability enables the application to cater to a wide range of prescriptions, making it applicable in different regions and healthcare systems.

Furthermore, the smart pharmacy application integrates with existing pharmacy management systems, allowing seamless transfer of prescription data. Pharmacists can efficiently access and update patient records, track medication history, and facilitate medication adherence. The application also provides real-time alerts for potential drug interactions, allergies, and dosage discrepancies, enhancing patient safety and care.

Overall, our smart pharmacy application offers a novel and efficient solution for prescription management in pharmacies. By leveraging AI and computer vision technologies, it simplifies the prescription reading process, improves accuracy, and enhances patient safety. The integration of this application into existing pharmacy systems has the potential to transform medication management, enabling pharmacists to deliver quality care while minimizing errors and improving overall healthcare outcomes.

8.2 Introduction for application

8.2.1 Overview for application

Reading the prescription to extract the information it contains, finding multiple ways to order the medication, and making multiple calls to reach it are all major problems. It can be challenging to read the medication's description and any warnings and precautions after buying the medication.

8.2.2 Objectives for application

- Reading medications, and baying them from the prescription, and determining the quickest route to find them.
- Saving the prescriptions for you and your family collectively for quick access to them and their reads and purchasing the medications inside prescription again.
- Reading the descriptions of all currently available medications,
- Searching for medications to thoroughly understand the warnings, precautions, and their effects.

8.2.3 Purpose

Assistance in obtaining medications, reading them from the prescription, and determining the quickest route to them, purchasing, and saving the prescriptions for you and your family collectively for quick access to them and their reads again, purchasing the medications inside them again, reading the descriptions of all currently available medications, and searching for them to thoroughly understand the warnings, precautions, and their effects.

8.2.4 Scope

The scope of the smart pharmacy application that scans prescriptions from a camera to extract medicine information using AI encompasses several key aspects, including technology, functionality, and implementation considerations. The following outlines the main components within the scope of the application:

8.2.4.1 Functionality

- Prescription Scanning: The application allows users to capture prescription images using a camera or upload existing images for analysis.

- Text Extraction: The application employs optical character recognition (OCR) techniques to extract textual information from prescription images, including medicine names, dosages, and instructions.
    - Medication Identification: By utilizing AI algorithms and a comprehensive pharmaceutical database, the application identifies and matches the extracted medicine information with accurate drug records.

8.2.4.2 Implementation Considerations:

- User Interface: The application should have an intuitive and user-friendly interface, allowing pharmacists to easily capture and process prescription images.

- Security and Privacy: Robust security measures should be implemented to ensure the confidentiality and integrity of patient data throughout the prescription scanning and processing stages.

- Data Accuracy and Validation: The application should continuously validate and refine the AI models to improve accuracy in prescription interpretation and medicine identification.

- Adaptability: The application should be designed to handle variations in prescription formats, handwriting styles, and regional specificities to cater to a wide range of prescriptions across different healthcare systems.

It is important to note that the scope of the smart pharmacy application may vary based on the specific requirements, resources, and regulations of the implementing organization or healthcare setting.

8.3 Project Planning and Analysis

8.3.1 Functional/System Requirements

Requirement ID: SP001

Requirement Title: **Scan Prescription**.

Requirement Rationale: User scan Prescription to read medicines from them.

Requirement Description: User scan Prescription to read medicines from them.

To order by medicines name.

Save Prescription in user account.

Requirement ID: SP002

Requirement Title: **Search Medicine**.

Requirement Rationale: User Search Medicine by name.

Requirement Description: search medicine by name to read description.

Understand the warnings and precautions.

Requirement ID: SP003

Requirement Title: **Get the Nearest Pharmacy**.

Requirement Rationale: User Search Pharmacy by his/her location.

Requirement Description: User Search Nearest Pharmacy by his/her location.

And offers available.

Requirement ID: SP004

Requirement Title: **Search Her/His Family**.

Requirement Rationale: User Search his/her Family by username.

Requirement Description: User Search his/her Family to join the group.

And view his/her family members.

Requirement ID: SP005

Requirement Title: **Join Her/His Family**.

Requirement Rationale: User can join her/his family.

Requirement Description: User can join her/his family.

To view his/her family members.

Requirement ID: SP006

Requirement Title: **Delete/View Prescription**.

Requirement Rationale: User can Delete/View Prescription.

Requirement Description: User can Delete/View Prescription.

And his/her family.

Requirement ID: SP007

Requirement Title: **View Family History**.

Requirement Rationale: User can view her/his family History.

Requirement Description: User can view her/his family History.

Order/Save/Join/Leave.

Requirement ID: SP008

Requirement Title: **View Family members**.

Requirement Rationale: User can view her/his family members.

Requirement Description: User can view her/his family members.

And can view all profiles family.

Requirement ID: SP009

Requirement Title: **View Medicine**.

Requirement Rationale: User can view Medicine.

Requirement Description: User can view Medicine when he/she purchase an Order, Or view its description.

Requirement ID: SP010

Requirement Title: **View/Add/Delete Cart**.

Requirement Rationale: User can View/Add/Delete Her/His Cart.

Requirement Description: User can view Cart to purchase an order.

User can Delete/Add any medicine in cart.

Requirement ID: SP011

Requirement Title: **Display Warning Message**.

Requirement Rationale: System warns users when ordering any medicine conflict any Chronic disease.

Requirement Description: System warns users when ordering any medicine conflict any Chronic disease.

To Warning users.

Requirement ID: SP012

Requirement Title: **Display Warning Message**.

Requirement Rationale: System warns users when ordering any medicine conflict any Chronic disease

Requirement Description: System warns users when ordering any medicine conflict any Chronic disease

To Warning users.

8.3.2 Non-Functional Requirements

1) Usability

System must provide a convenient User Interface and user experience for the intended user, and full documentation must be supplied for the user too.

2) Reliability

System Failure rate must be 0% and do not exceed 5% for better Software experience and Business performance

3) Performance

Systems response time should not exceed 5 seconds.

System overall size should be max 2 GB.

4) Supportability

System must support all different platforms and all different Web Browsers.

All test cases must be maintained and covered before publishing the product in the market. Feature maintenance must me tracked after the application goes into production

5) Implementation

This Android/ISO application must be implemented using

- Flutter for front-end
- Dart for back-end
- SQL Server For DB

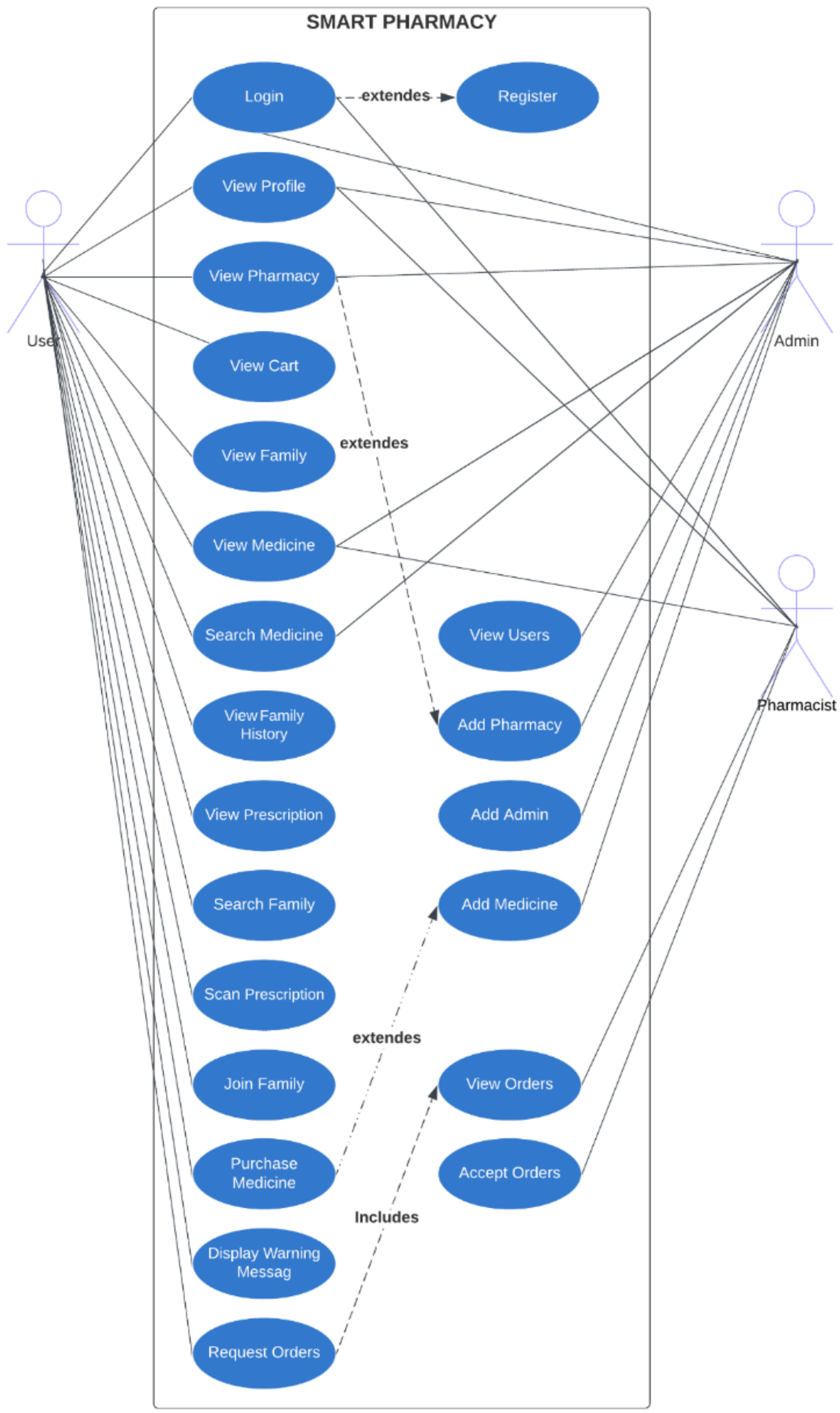
6) External Interface Requirements

- Communication Interface

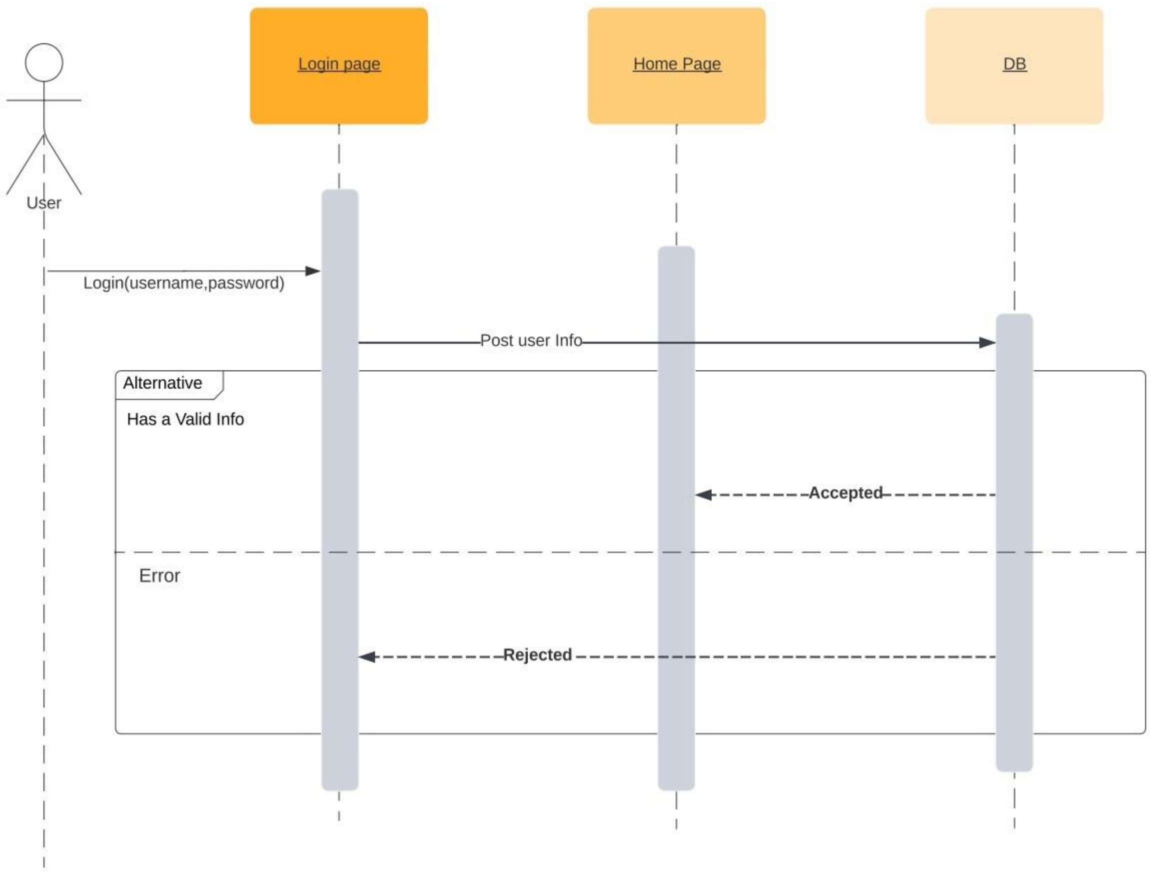
TCP/IP communication that is necessary for the operation of this system.

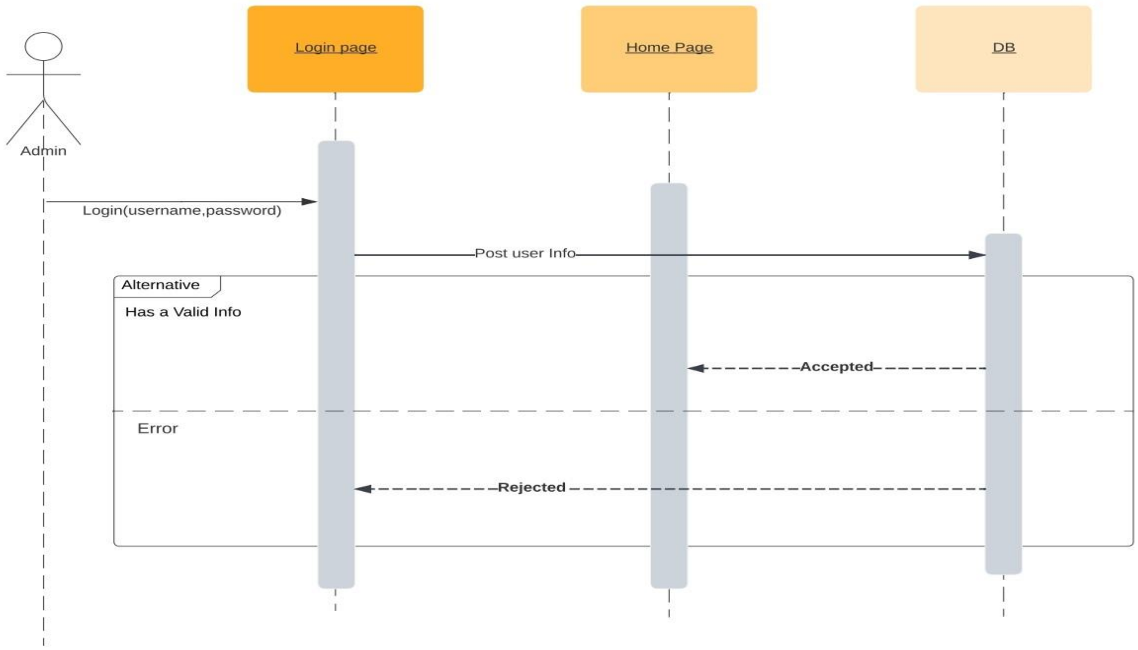
8.4 Software Design

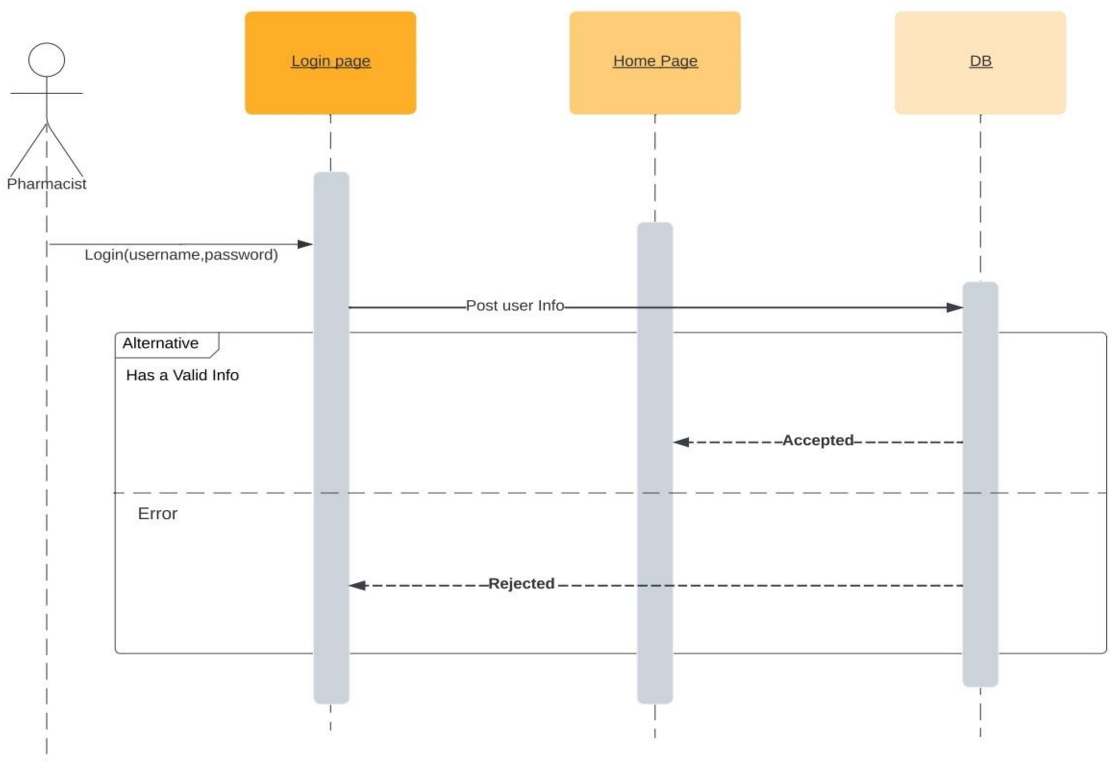
1. Use Case Diagram

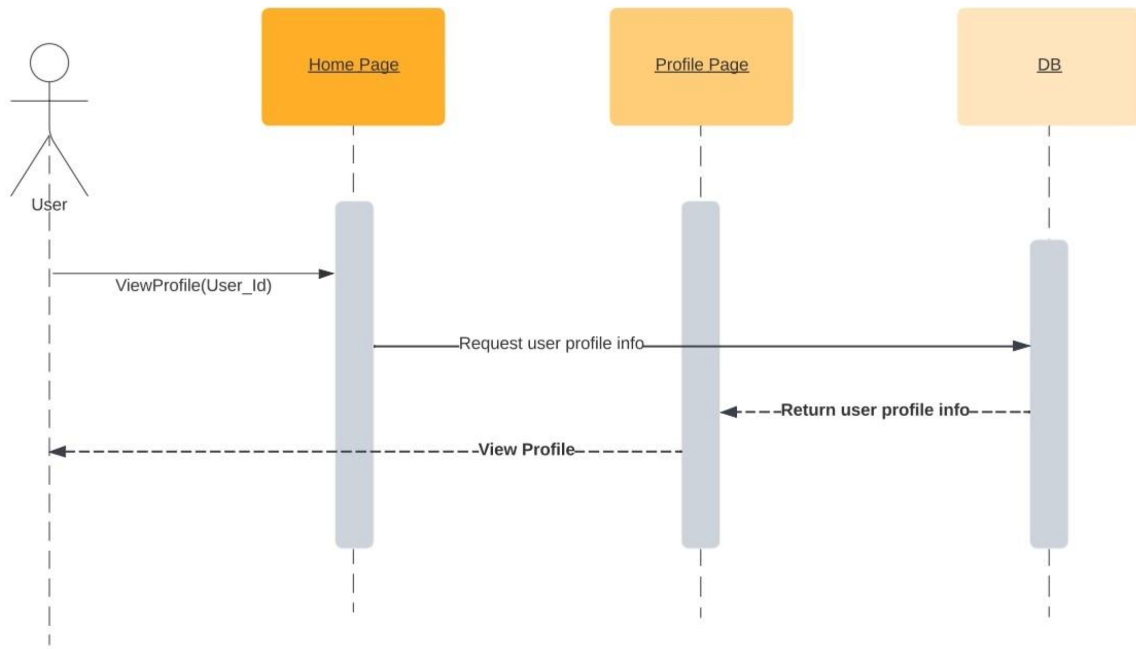


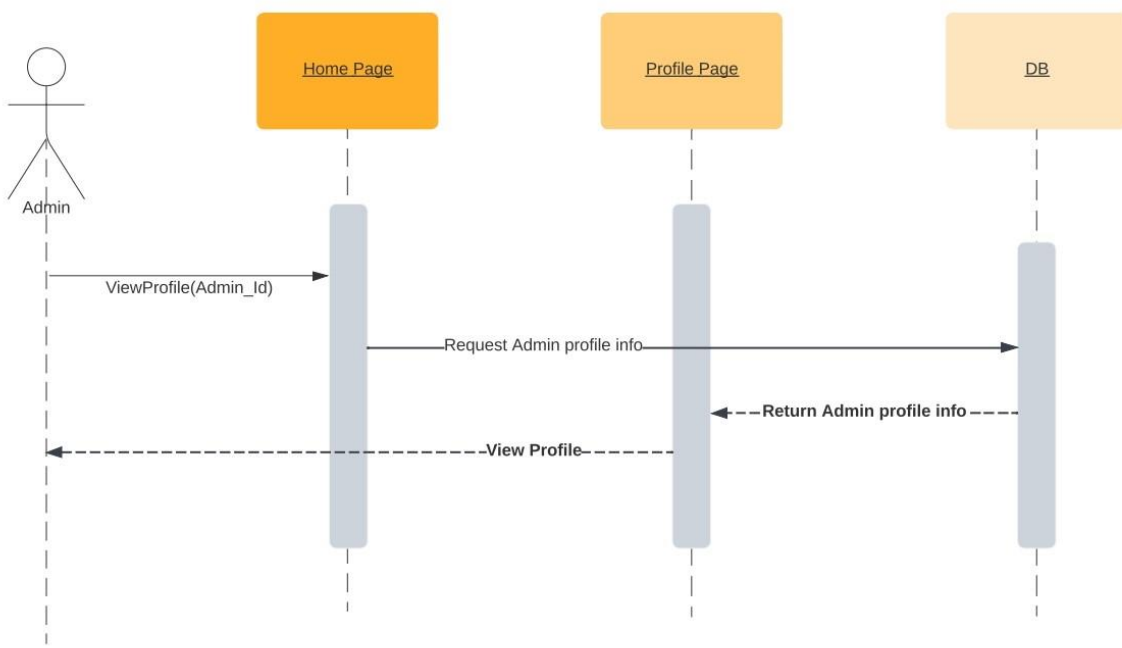
2. Sequence Diagram

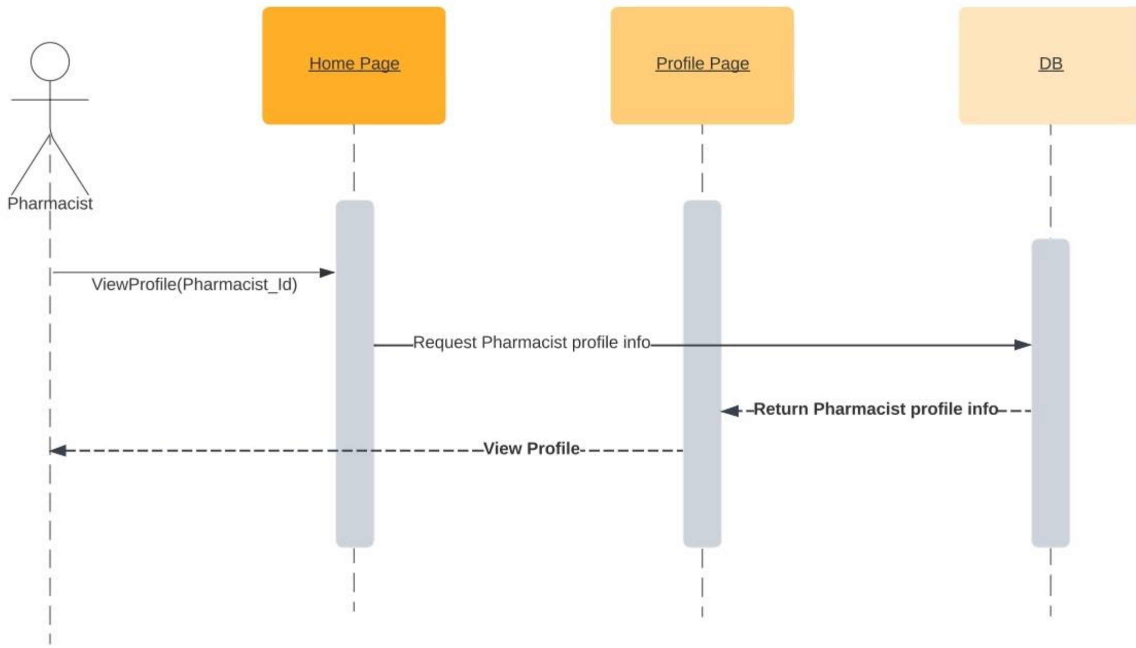


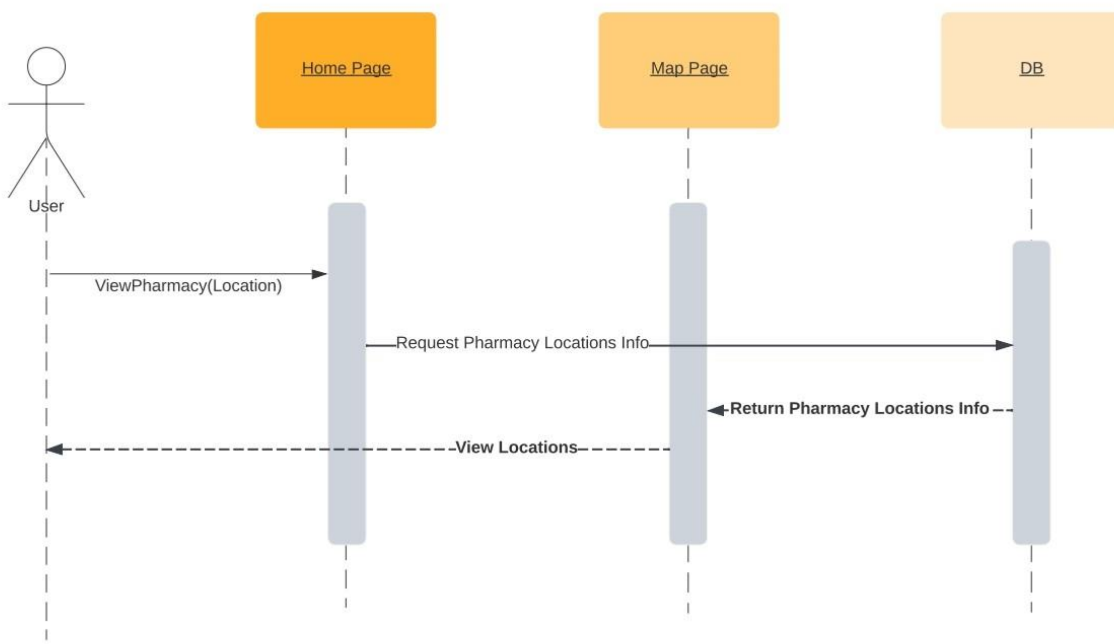

Login page
Home Page
DB
Admin
Login(username,password)
Post user Info
Alternative
Has a Valid Info
Accepted
Error
Rejected

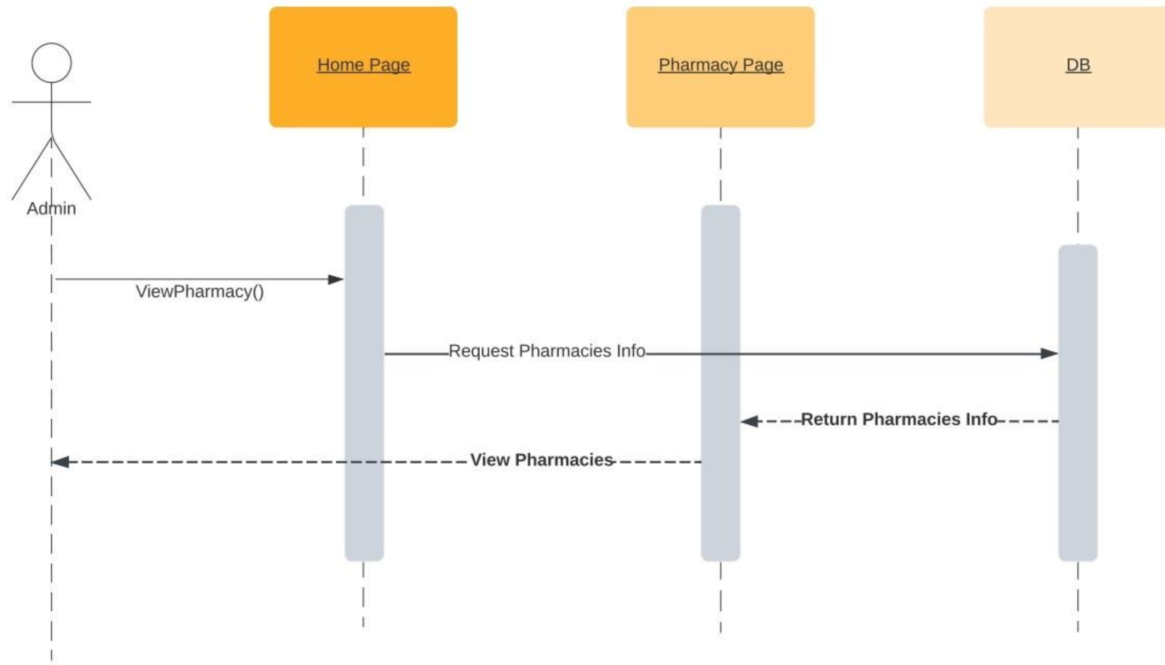

Login page
Home Page
DB
Pharmacist
Login(username,password)
Post user Info
Alternative
Has a Valid Info
Accepted
Error
Rejected

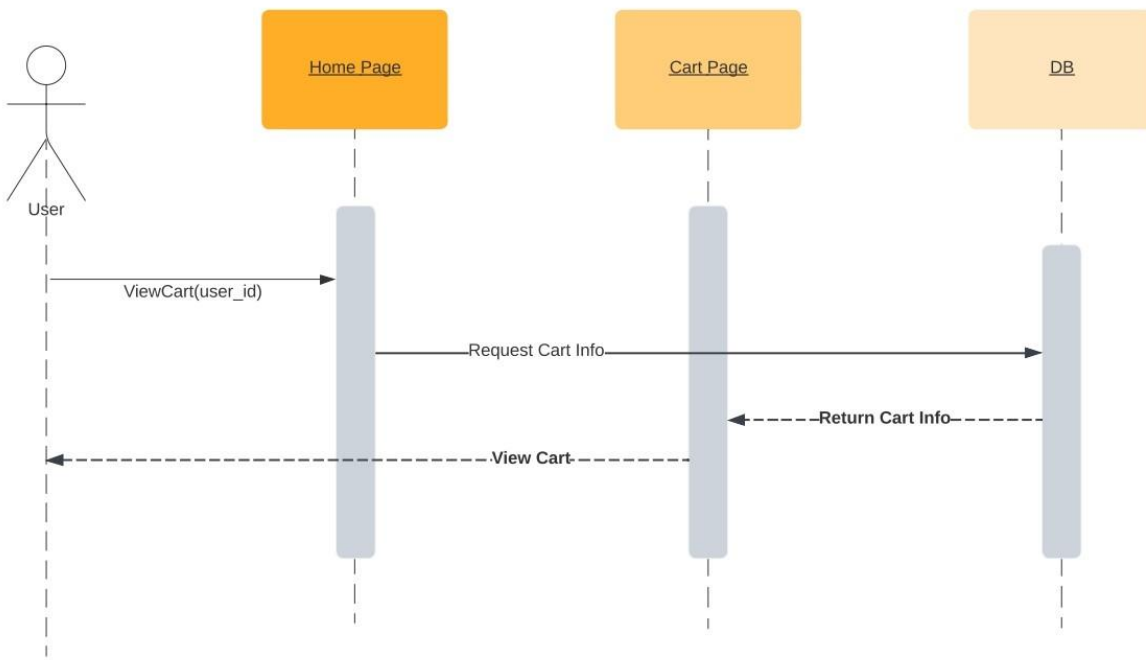

Home Page
Profile Page
DB
User
ViewProfile(User_Id)
Request user profile info
Return user profile info
View Profile

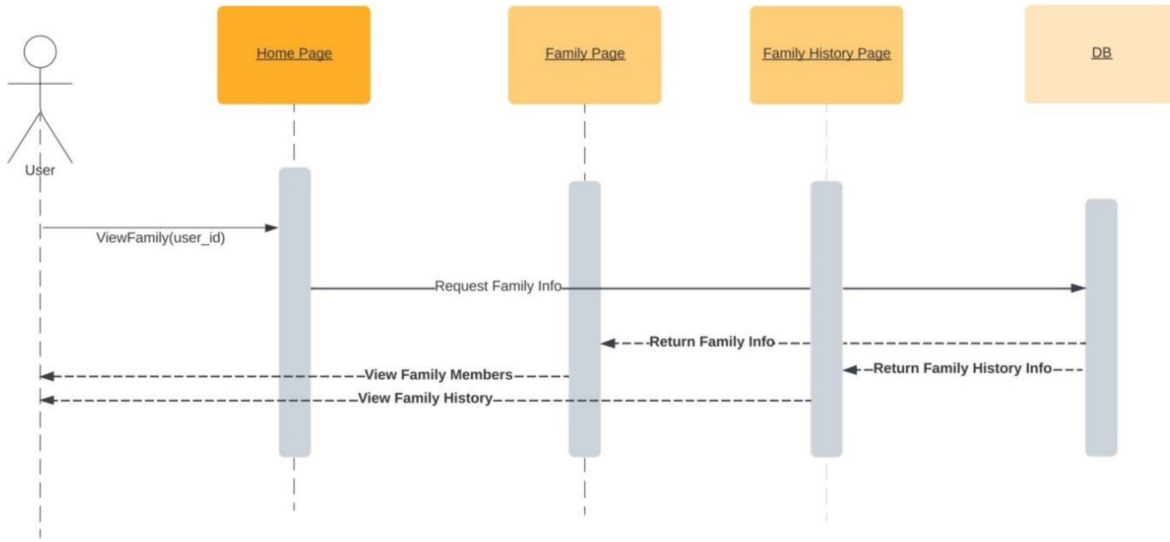

Home Page
Profile Page
DB
Admin
ViewProfile(Admin_Id)
Request Admin profile info
Return Admin profile info
View Profile

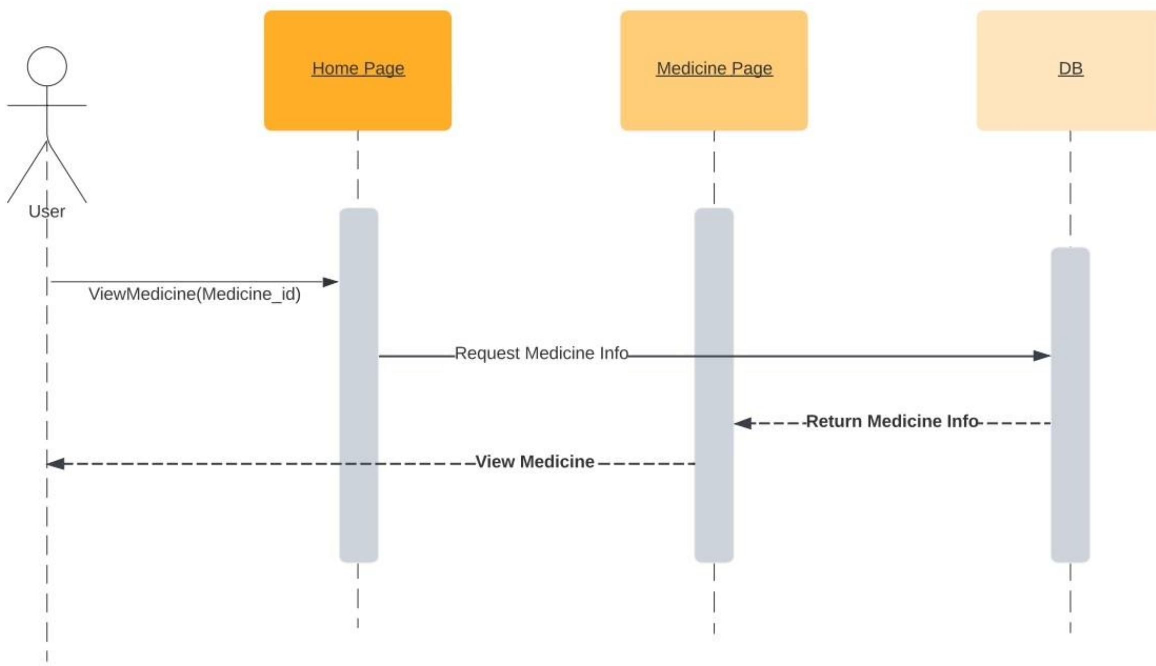

Home Page
Profile Page
DB
Pharmacist
ViewProfile(Pharmacist_Id)
Request Pharmacist profile info
Return Pharmacist profile info
View Profile

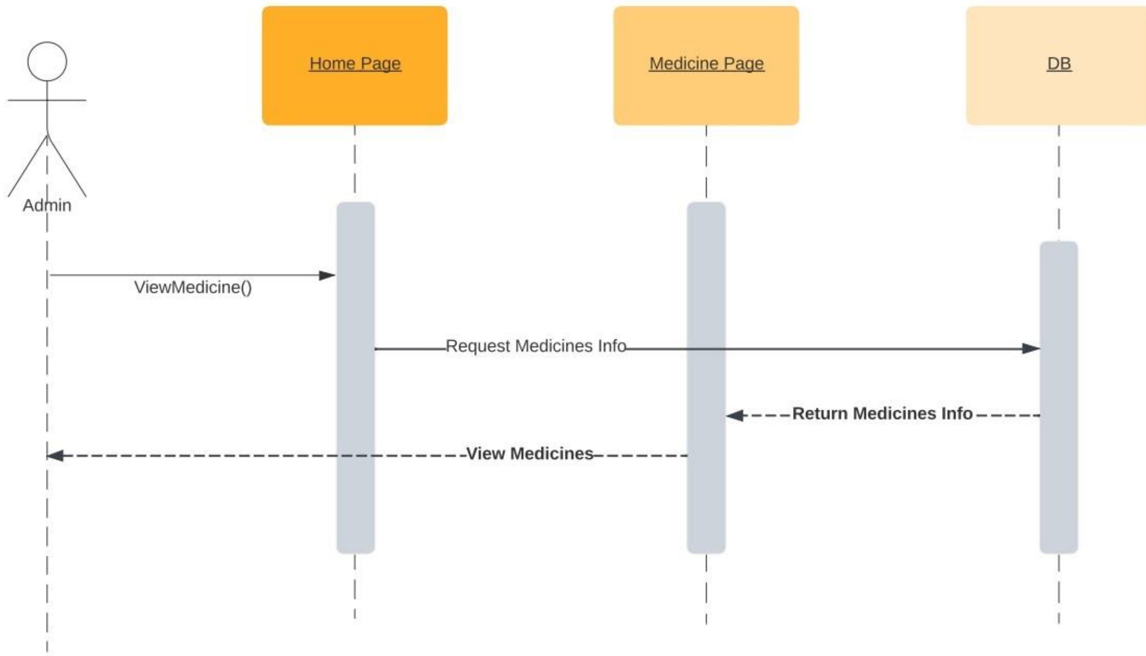

Home Page
Map Page
DB
User
ViewPharmacy(Location)
Request Pharmacy Locations Info
Return Pharmacy Locations Info
View Locations

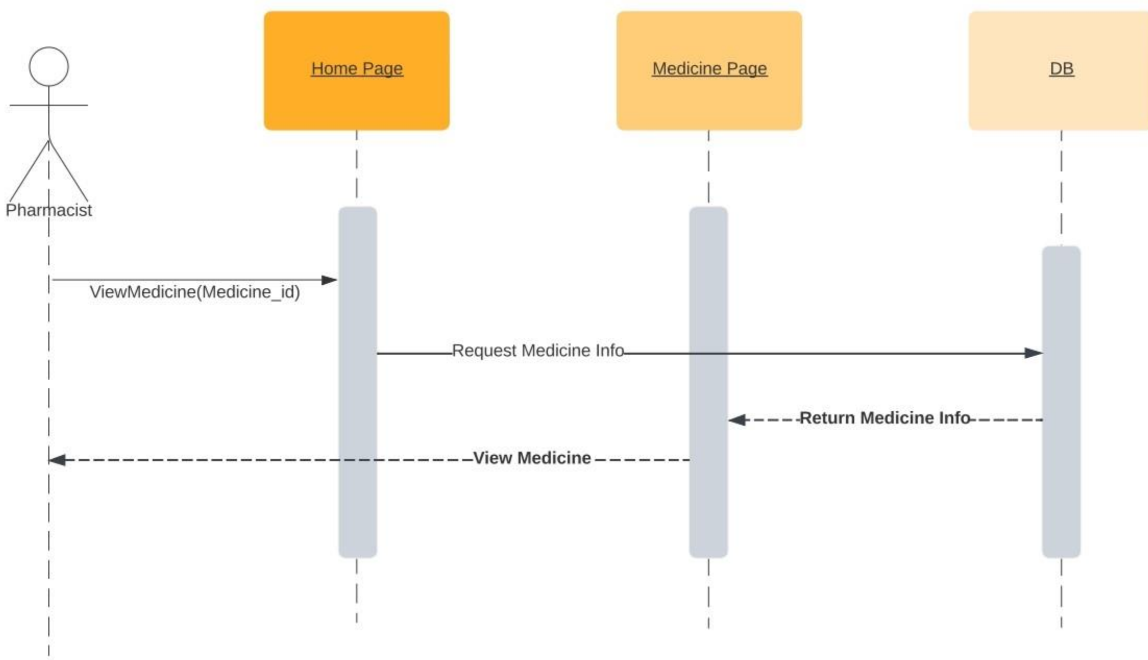

Home Page
Pharmacy Page
DB
Admin
ViewPharmacy()
Request Pharmacies Info
Return Pharmacies Info
View Pharmacies

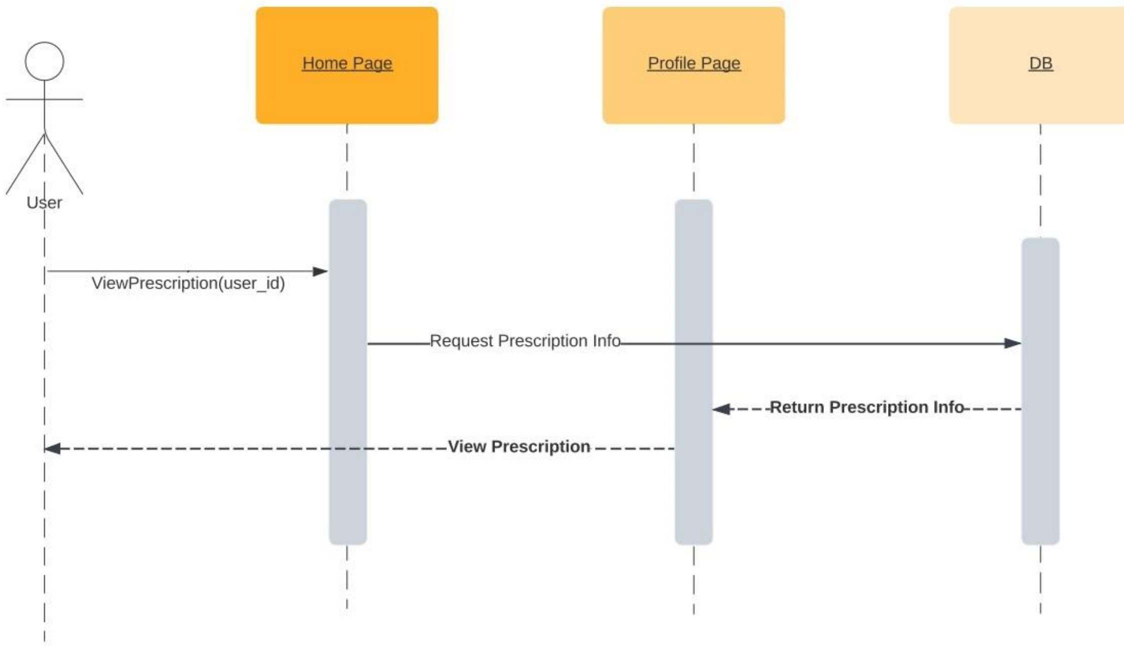

Home Page
Cart Page
DB
User
ViewCart(user_id)
Request Cart Info
Return Cart Info
View Cart

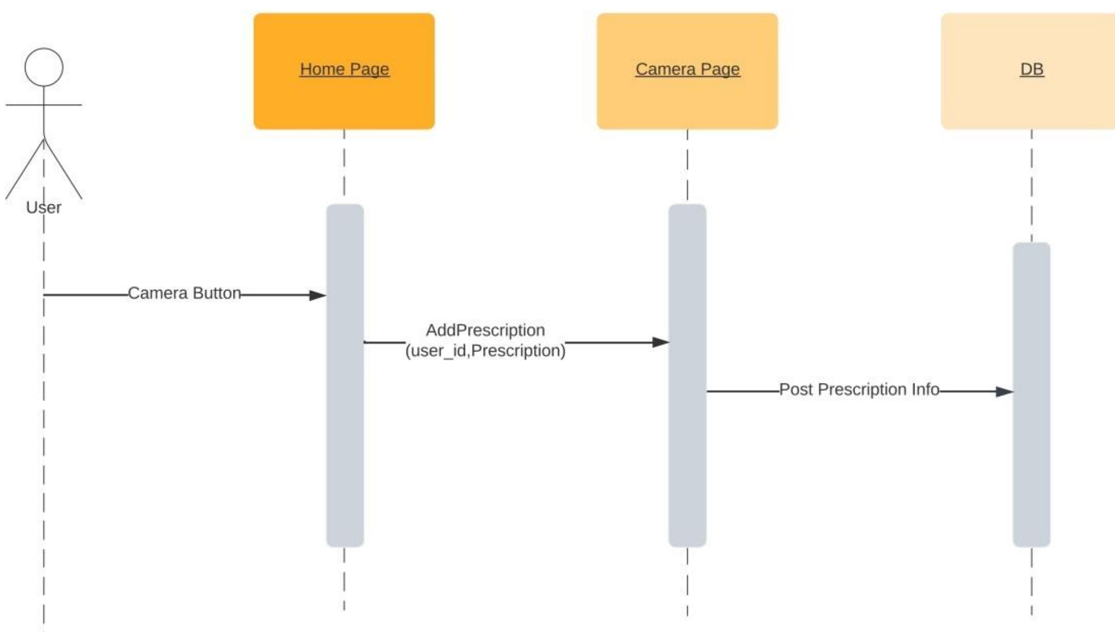

Home Page
Family Page
Family History Page
DB
User
ViewFamily(user_id)
Request Family Info
Return Family Info
Return Family History Info
View Family Members
View Family History

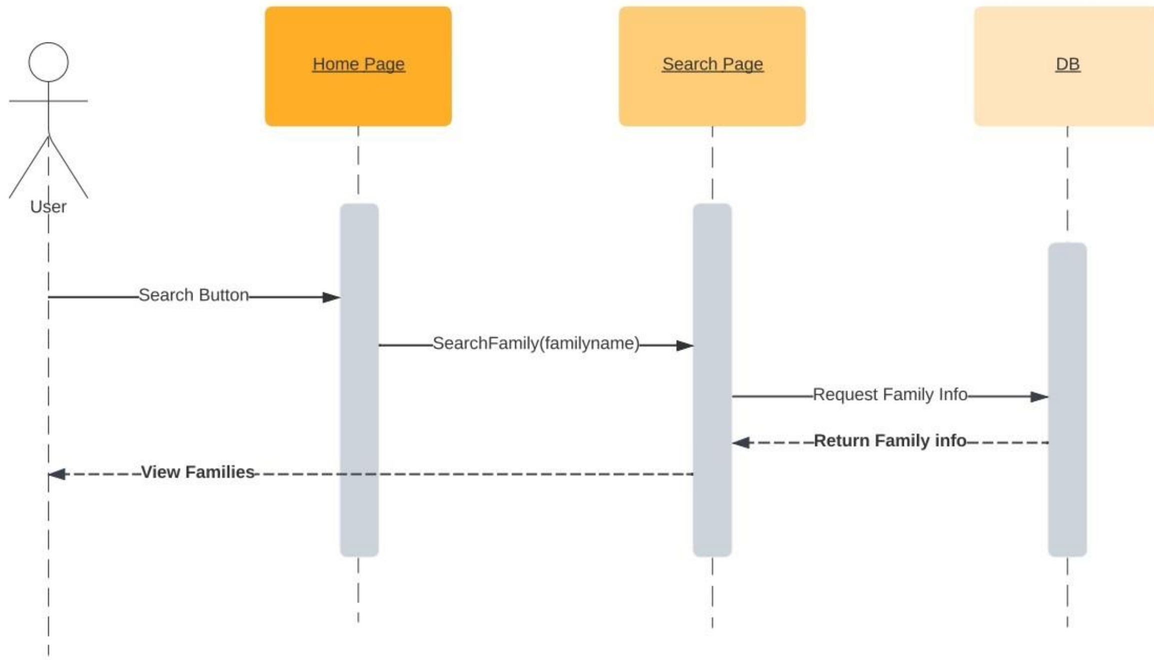

Home Page
Medicine Page
DB
User
ViewMedicine(Medicine_id)
Request Medicine Info
Return Medicine Info
View Medicine

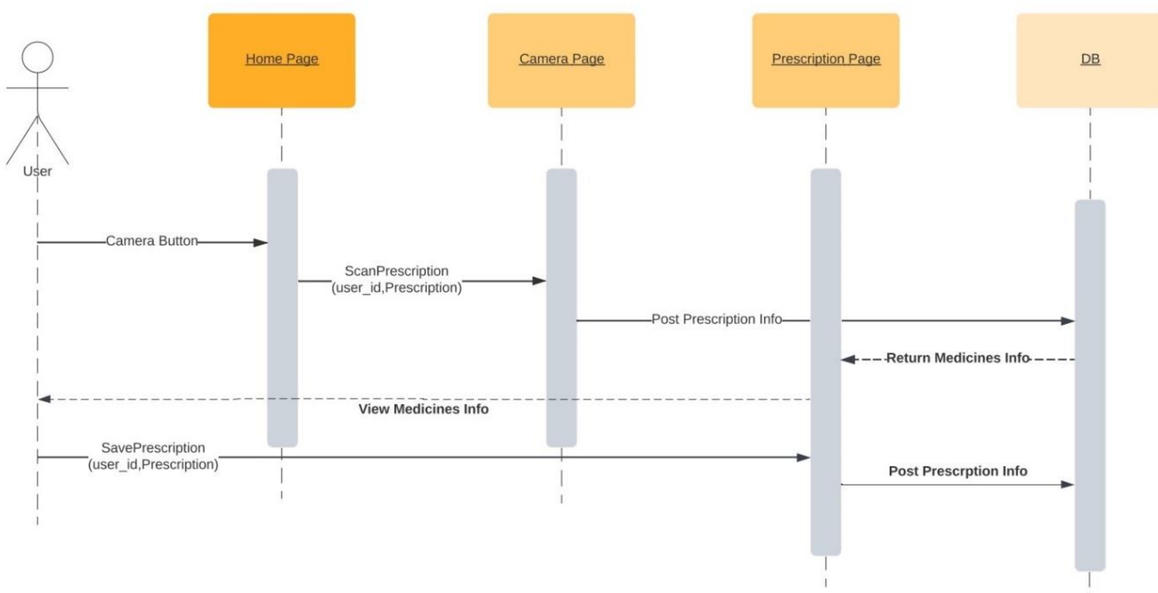

Home Page
Medicine Page
DB
Admin
ViewMedicine()
Request Medicines Info
Return Medicines Info
View Medicines

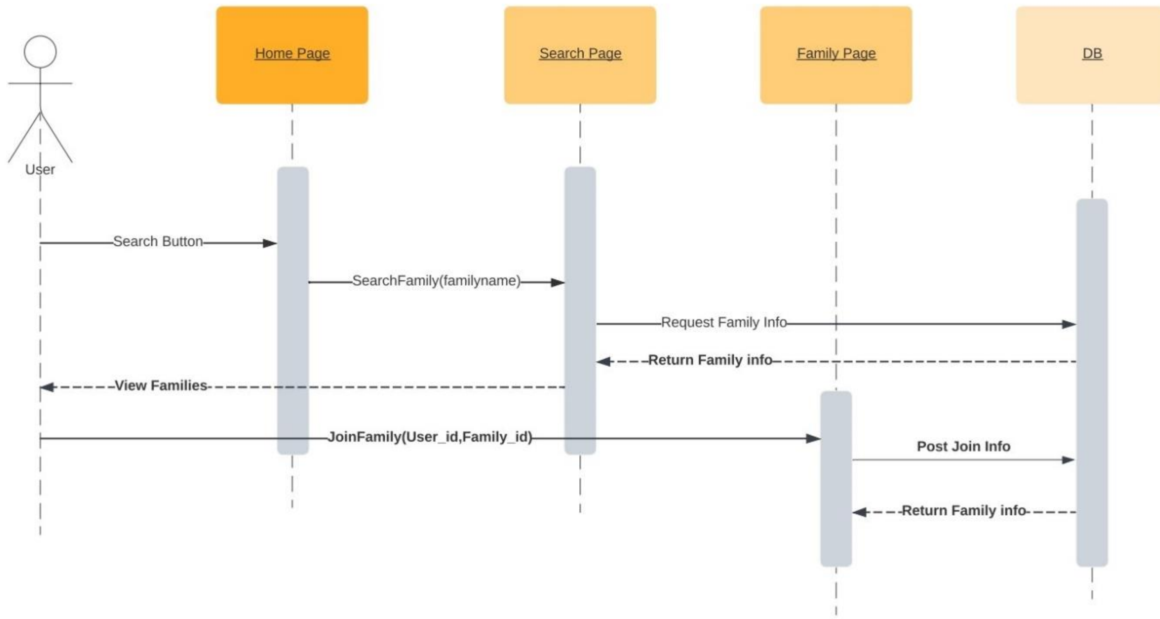

Home Page
Medicine Page
DB
Pharmacist
ViewMedicine(Medicine_id)
Request Medicine Info
Return Medicine Info
View Medicine

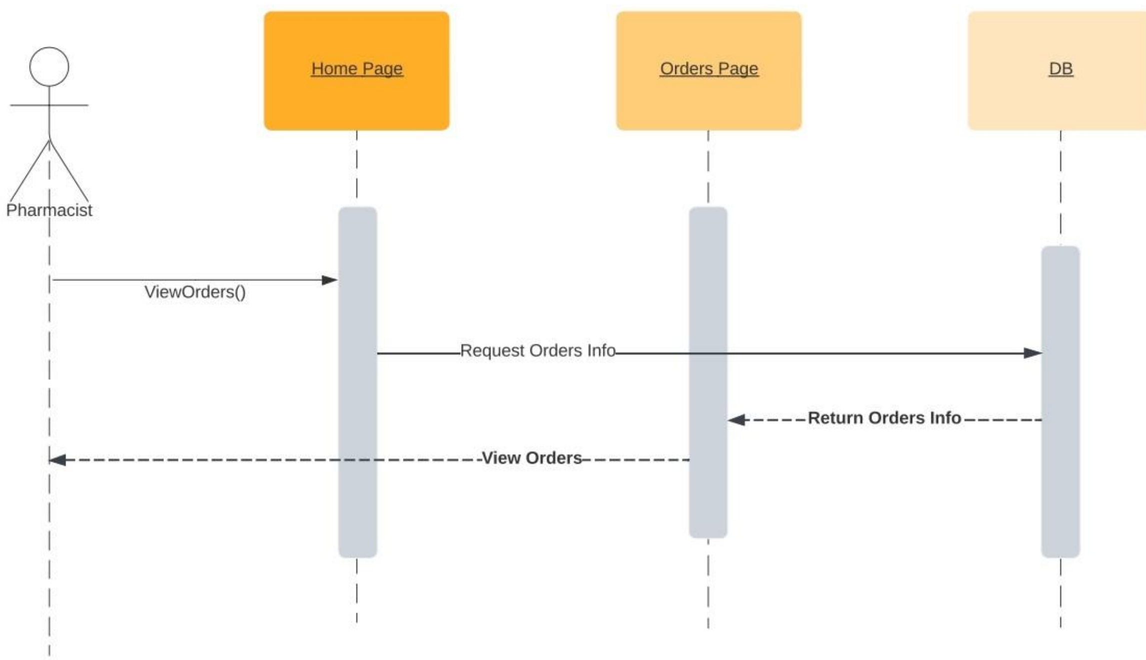

Home Page
Profile Page
DB
User
ViewPrescription(user_id)
Request Prescription Info
Return Prescription Info
View Prescription

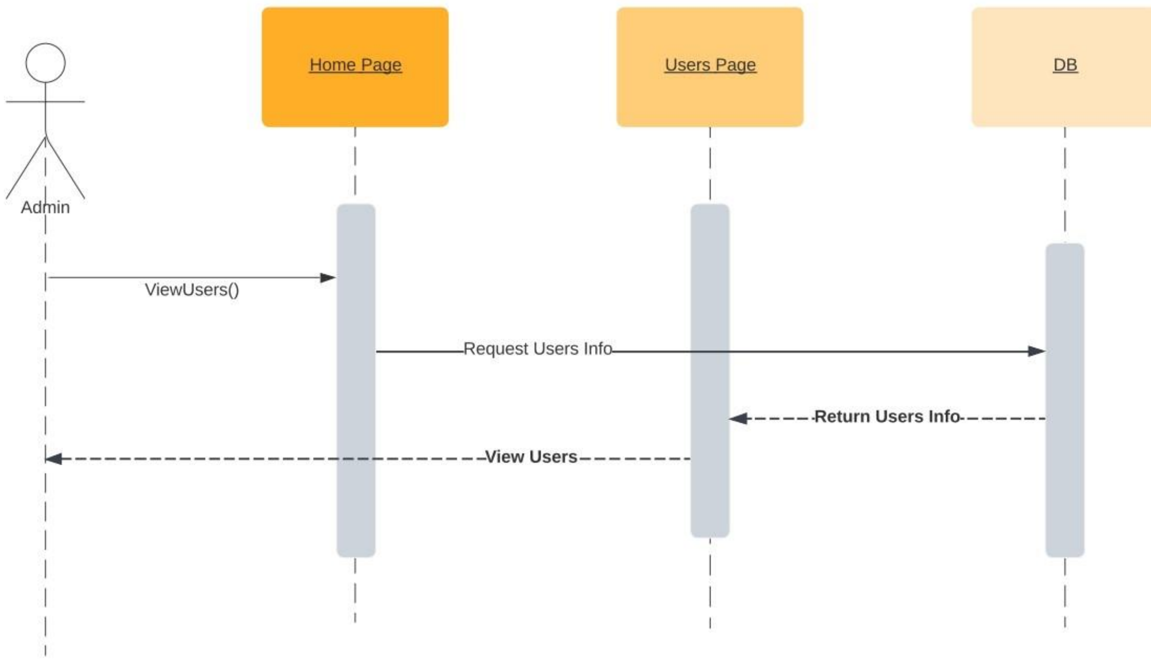

Home Page
Camera Page
DB
User
Camera Button
AddPrescription
(user_id,Prescription)
Post Prescription Info

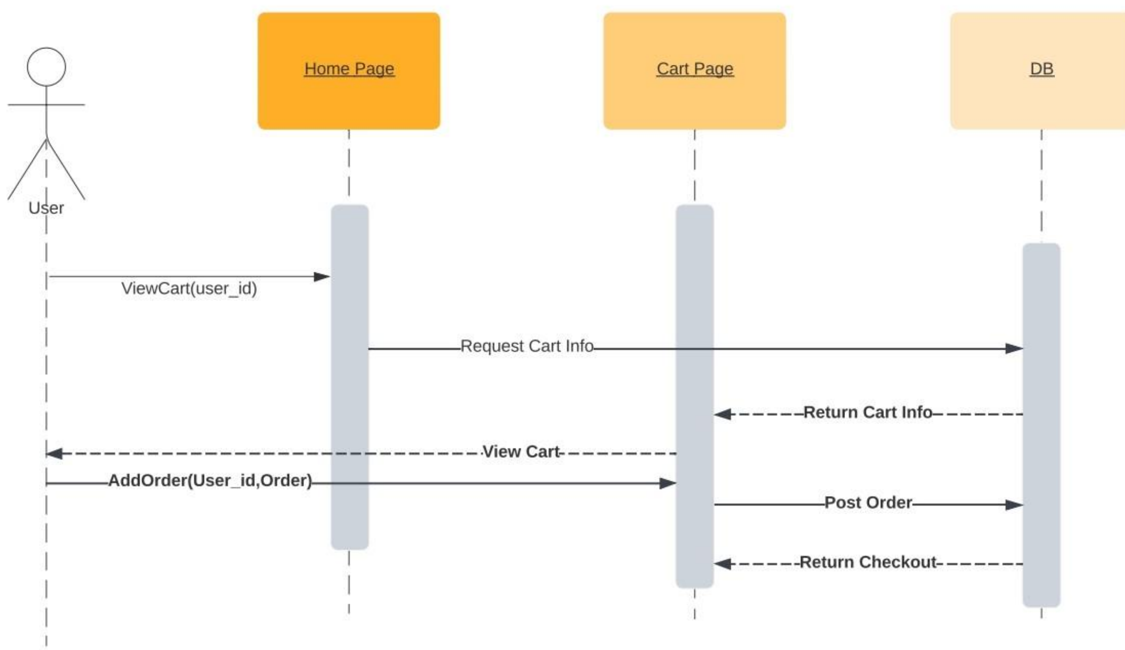

Home Page
Search Page
DB
User
Search Button
SearchFamily(familyname)
Request Family Info
Return Family info
View Families

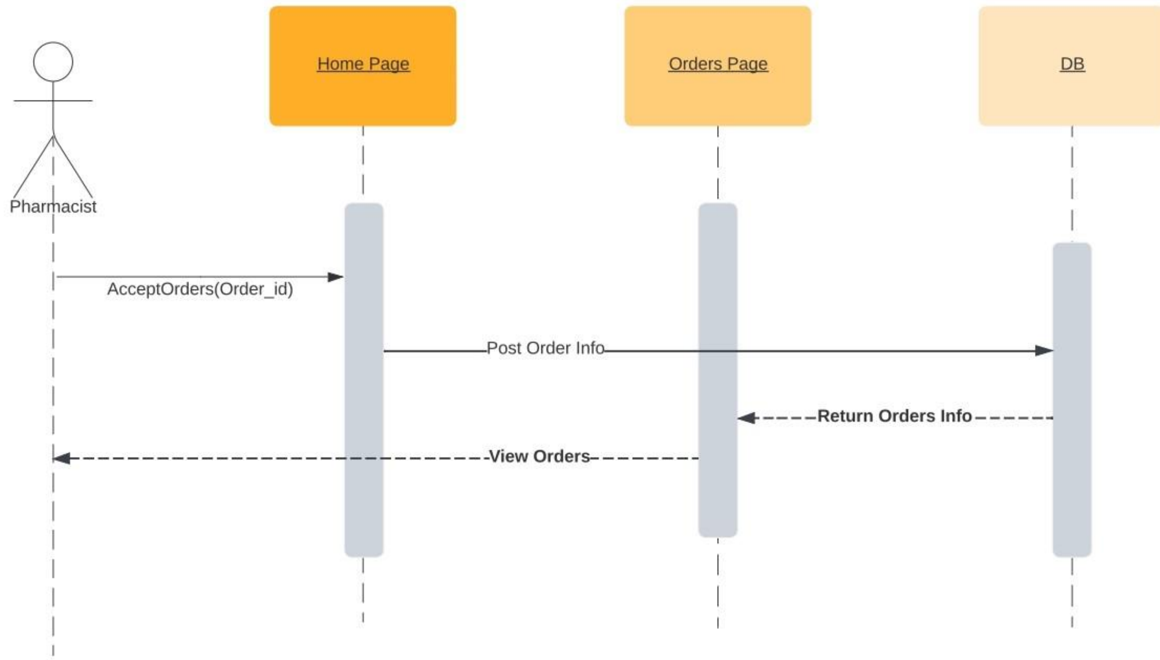

Home Page
Camera Page
Prescription Page
DB
User
Camera Button
ScanPrescription
(user_id,Prescription)
Post Prescription Info
Return Medicines Info
View Medicines Info
SavePrescription
(user_id,Prescription)
Post Prescrption Info

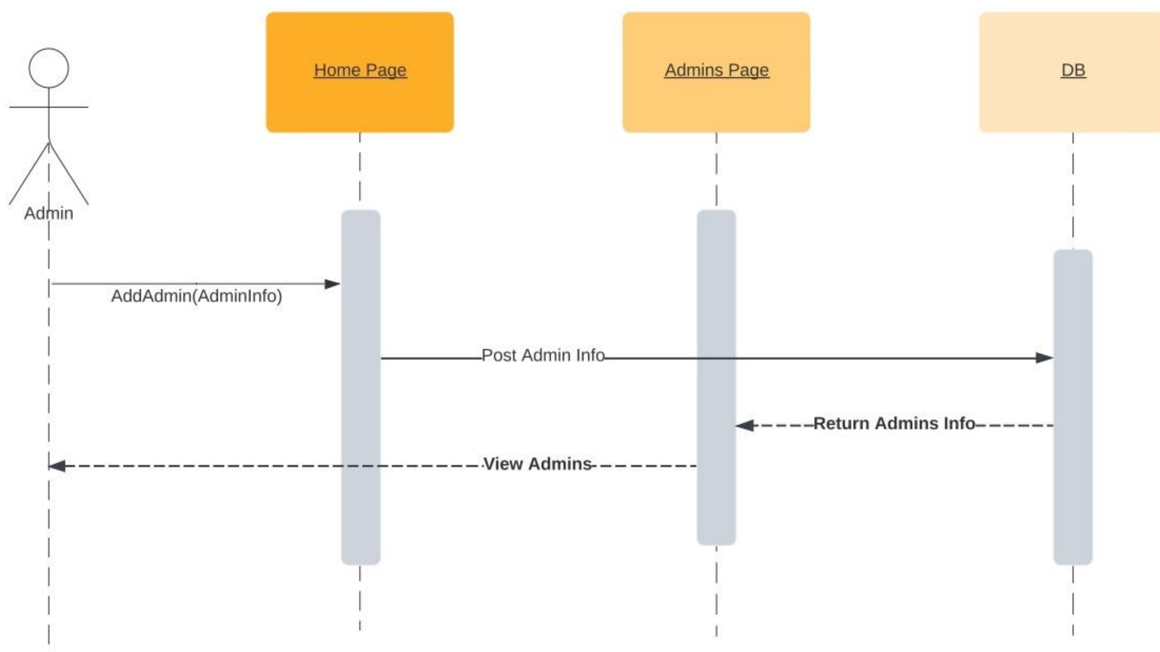

Home Page
Search Page
Family Page
DB
User
Search Button
SearchFamily(familyname)
Request Family Info
Return Family info
View Families
JoinFamily(User_id,Family_id)
Post Join Info
Return Family info

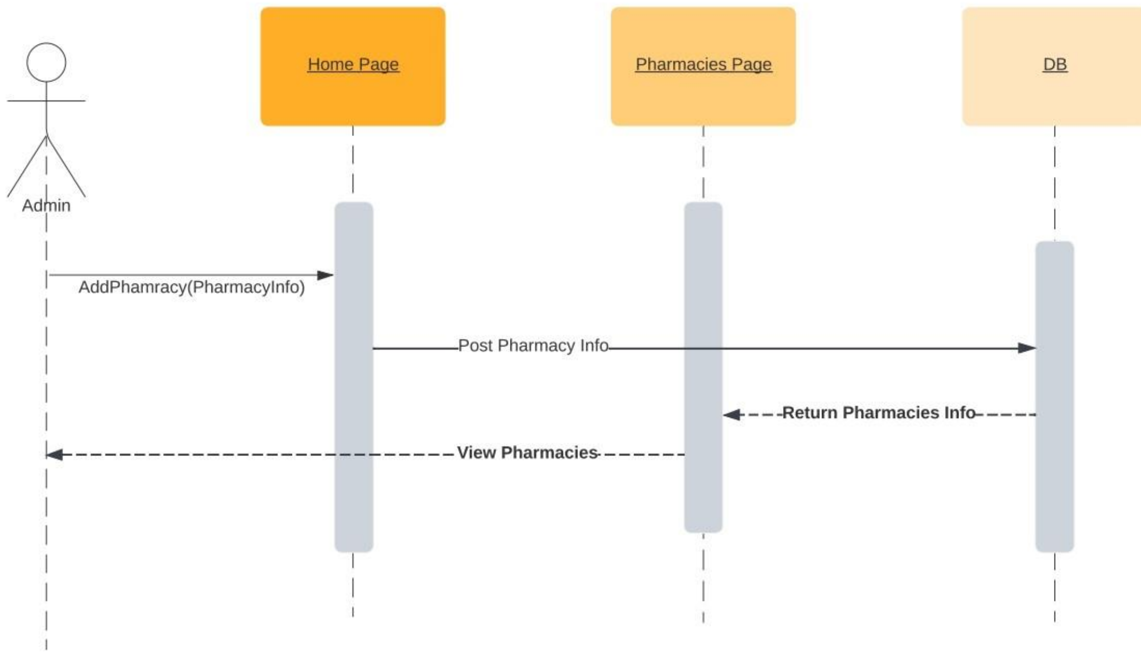

Home Page
Orders Page
DB
Pharmacist
ViewOrders()
Request Orders Info
Return Orders Info
View Orders


Home Page
Users Page
DB
Admin
ViewUsers()
Request Users Info
Return Users Info
View Users

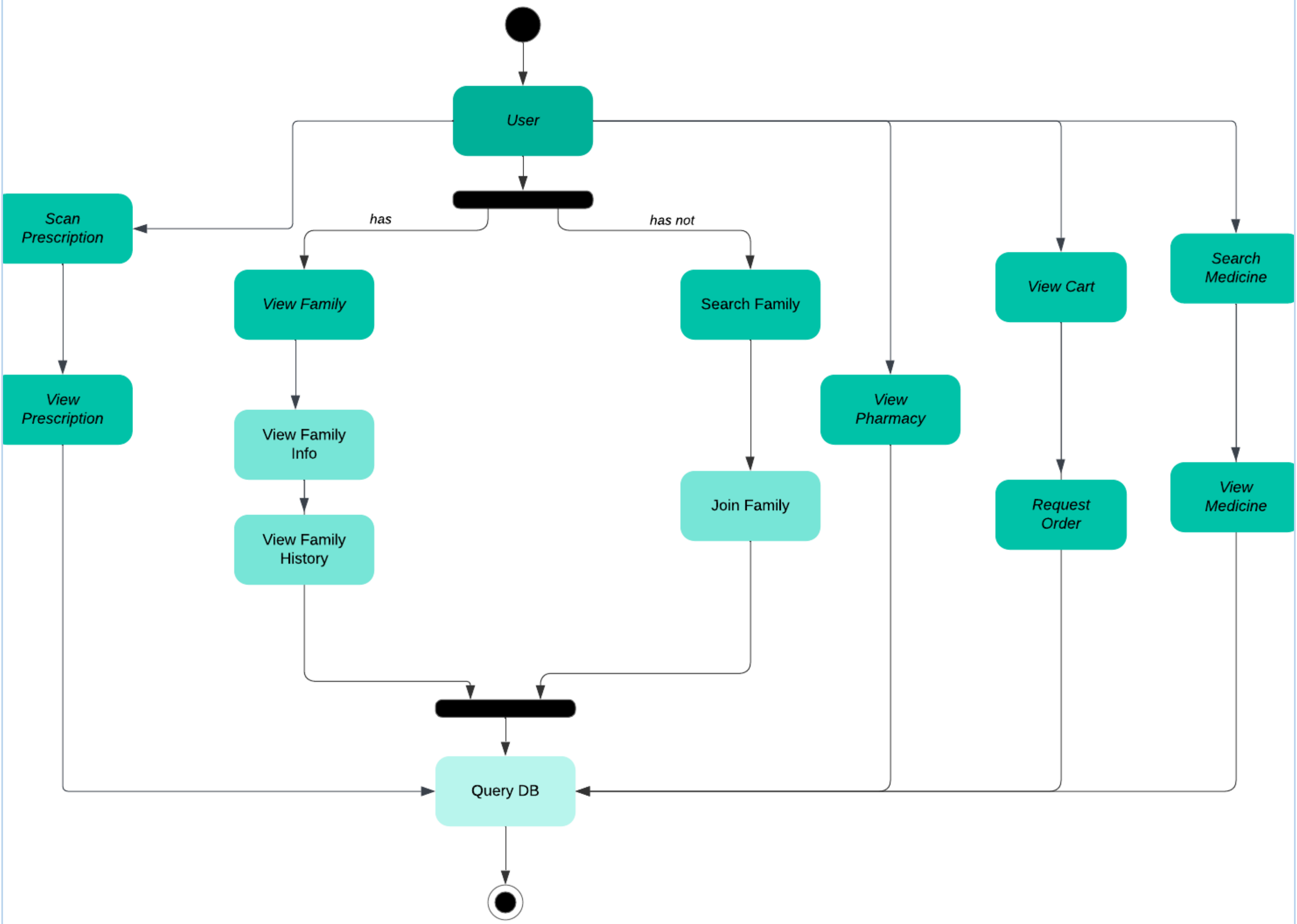

Home Page
Cart Page
DB
User
ViewCart(user_id)
Request Cart Info
Return Cart Info
View Cart
AddOrder(User_id,Order)
Post Order
Return Checkout

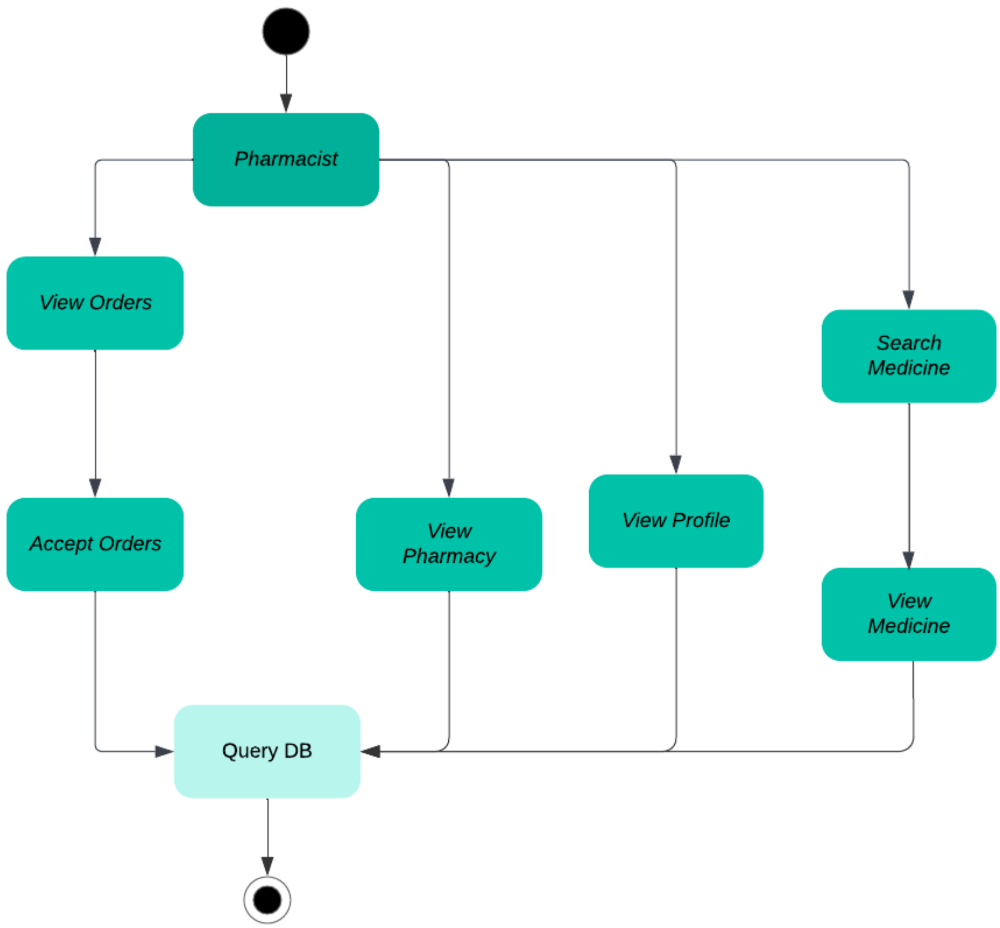

Home Page
Orders Page
DB
Pharmacist
AcceptOrders(Order_id)
Post Order Info
Return Orders Info
View Orders


Home Page
Admins Page
DB
Admin
AddAdmin(AdminInfo)
Post Admin Info
Return Admins Info
View Admins


Home Page
Pharmacies Page
DB
Admin
AddPhamracy(PharmacyInfo)
Post Pharmacy Info
Return Pharmacies Info
View Pharmacies

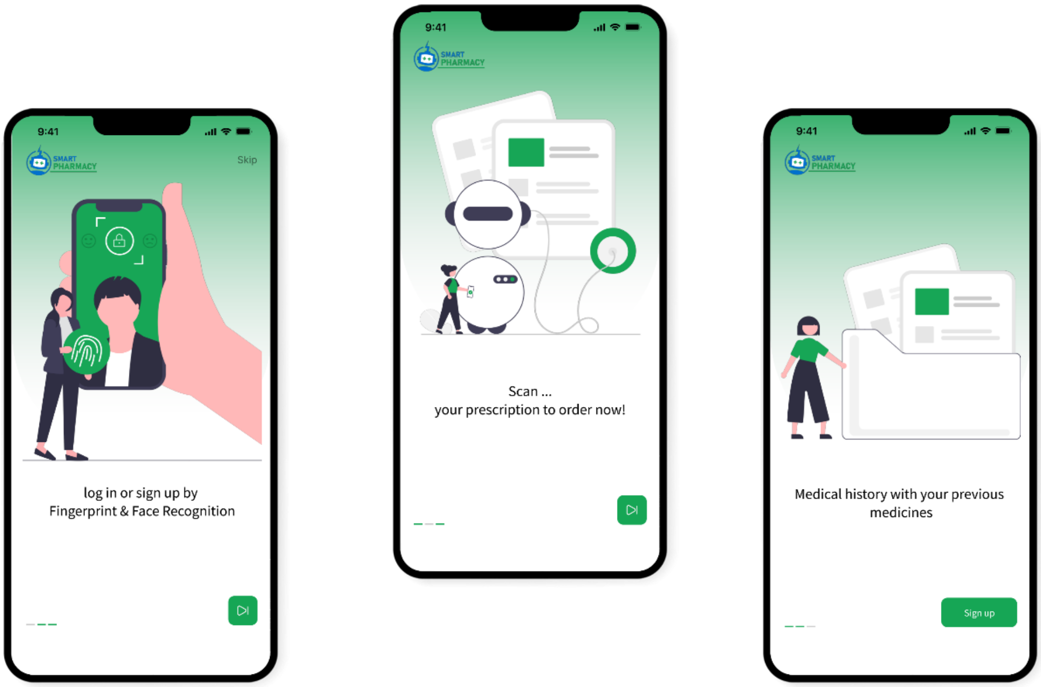
3. Activity Diagram



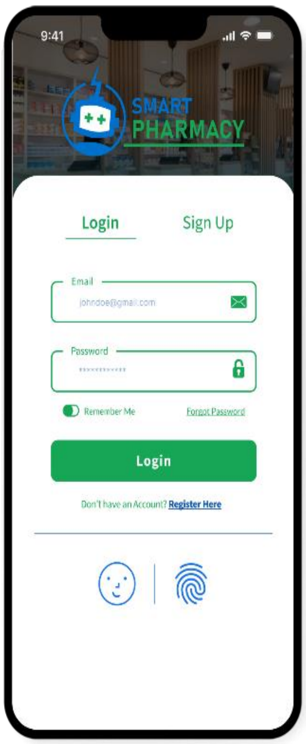


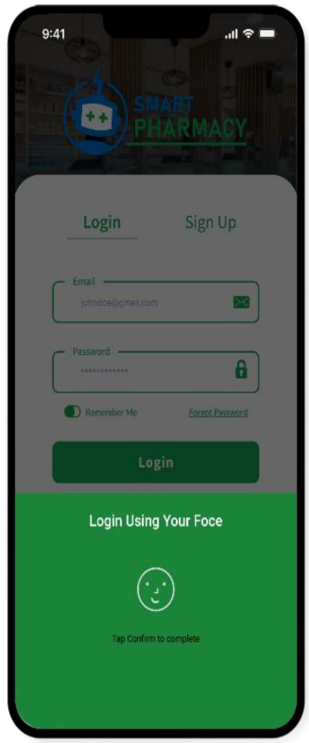
Pharmacist
View Orders
Search Medicine
Accept Orders
View Pharmacy
View Profile
View Medicine
Query DB

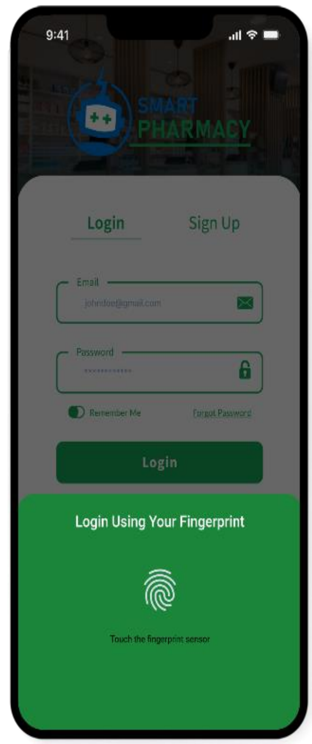
8.5 User interface (User)



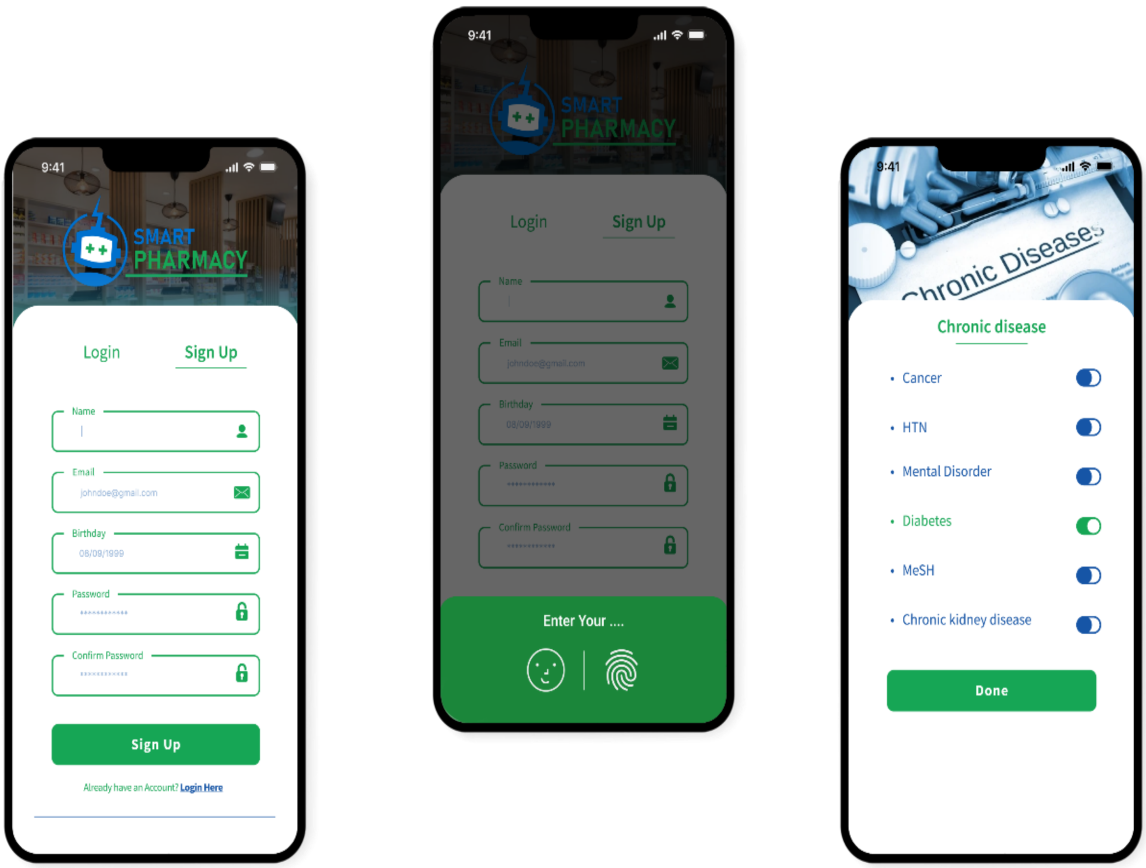
Login


9:41
SMART
PHARMACY
Login
Sign Up
Email
johndoe@gmail.com
Password
Remember Me
Forgot Password
Login
Don't have an Account? Register Here

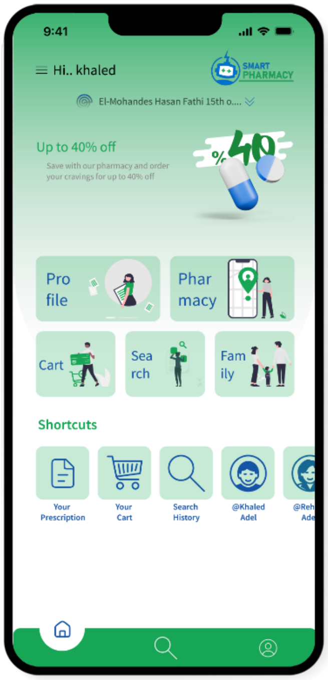

9:41
SMART
PHARMACY
Login
Sign Up
Email
Password
Login
Login Using Your Face
Tap Confirm to complete


9:41
SMART
PHARMACY
Login
Sign Up
Email
Password
Login
Login Using Your Fingerprint
Touch the fingerprint sensor

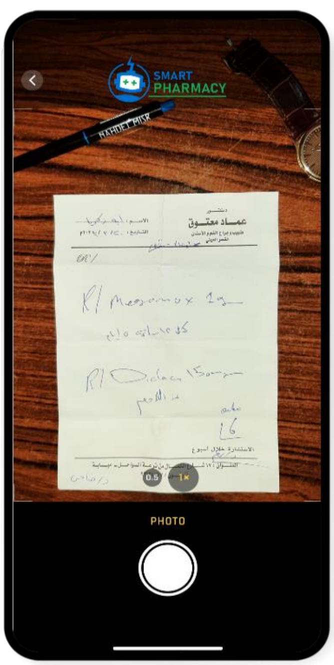
Sign up

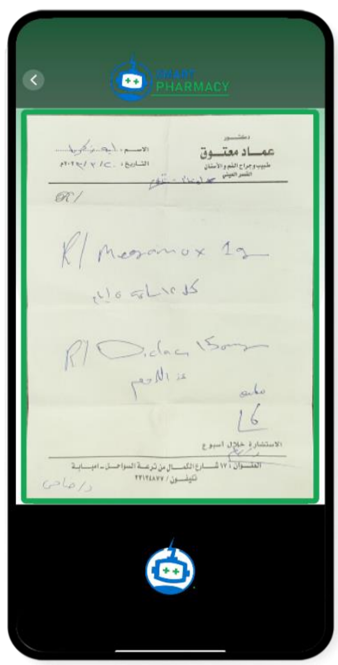

9:41
SMART
PHARMACY
Login
Sign Up
Name
Email
johndoe@gmail.com
Birthday
08/09/1999
Password
Confirm Password
Sign Up
Already have an Account? Login Here
9:41
SMART
PHARMACY
Login
Sign Up
Name
Email
Birthday
Password
Confirm Password
Enter Your ....
9:41
Chronic Diseases
Chronic disease
Cancer
HTN
Mental Disorder
Diabetes
MeSH
Chronic kidney disease
Done

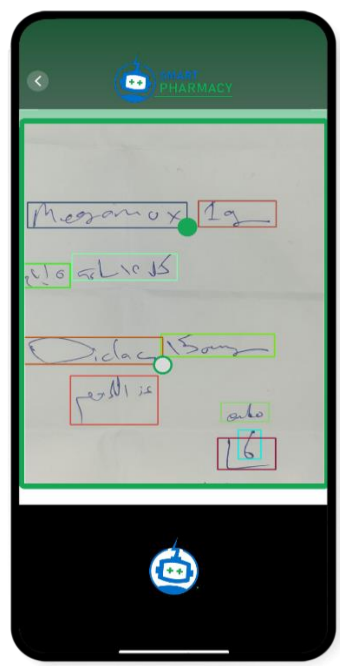
Home,   Profile,   Family



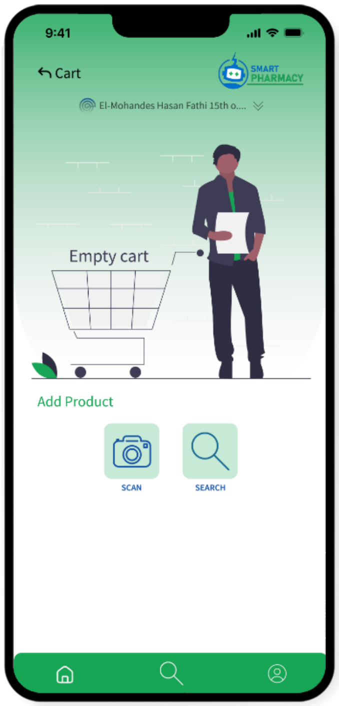
Scan Prescription



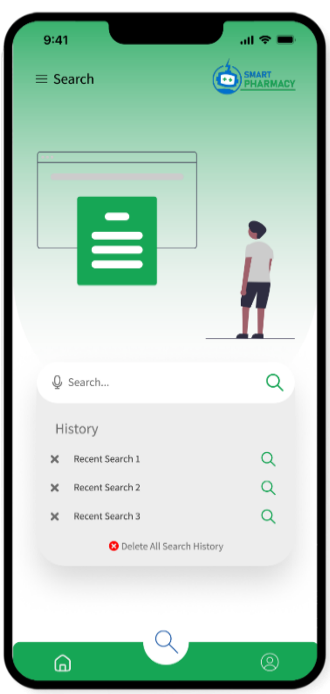


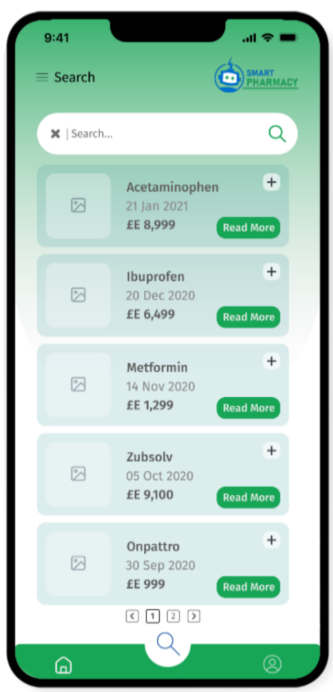


Cart



Search Medicine





9 Summary & Conclusion

The aim of this project was to identify doctors' handwriting. To achieve this goal, this project contributes in three steps:

1. Image before processing (prescription):

- We show some problems reading prescriptions:
  - warp perspective (objects appear outside the frame of the paper)
  - Image enhancement (Can't use fixed threshold only)

- Segmentation of the interested region:
  - Remove Header and Footer of Prescription.
  - Remove "R/" symbol .

- Segmentation of the interested region.
- Segmentation of text.

2. Introducing a unique data augmentation technique:

- Dataset named 'Handwritten Medical Term Corpus', dataset has 17,431 handwritten instances of 480 medical related words.
- Unique Dataset has 2,861 handwritten instances of 7 classes ('Cataflam', 'Ketolac', 'Brufen', 'Panadol', 'Actos', 'Diclac', 'Insulin') 510 images for each drug.
  this dataset written by the nine people , (our team and our friends)

3. Using some techniques of approach for final recognition. It also compares recognition accuracy in different stages of enhancement.

- we used YOLOV5 to recognize character and achieved accuracy 60.7%.
- we built CNN model to recognize word and achieved accuracy 100% (overfitting)
- we used VGG 16 to recognize word and achieved train accuracy 97.2%, test accuracy 94.7%

10 Future Work

Certainly, there will be future plans to improve the project, including increasing the number of classes that the model is trained on, and also making the application read the entire medical prescription with all its contents, whether the details of the medicine or the number of doses written by the doctor, and reading the patient's data to facilitate the patient's instead of writing it m

11 References

1] An online cursive handwritten medical words recognition system for busy doctors in developing countries for ensuring efficient healthcare service delivery | Scientific Reports (nature.com).

2] http://socialtech.gramweb.net/media-archive/codes.

3] (PDF) Doctor's Cursive Handwriting Recognition System Using Deep Learning (researchgate.net)

4] https://paperswithcode.com/sota/handwritten-text-recognition-on-bentham

5] https://paperswithcode.com/dataset/iam

6] https://www.analyticsvidhya.com/blog/2022/07/training-cnn-from-scratch-using-the-custom-dataset/

7] https://prince-canuma.medium.com/image-pre-processing-c1aec0be3edf

8] https://learnopencv.com/implementing-cnn-tensorflow-keras/

9] https://docs.opencv.org/4.x/d7/d4d/tutorial_py_thresholding.html

10] https://www.tensorflow.org/tutorials/images/transfer_learning

11] https://analyticsindiamag.com/a-comparison-of-4-popular-transfer-learning-models/

12] https://link.springer.com/chapter/10.1007/978-3-030-86337-1_27?fbclid=IwAR2mOrKVlvjd-v-Zh_2JJAm00zrjQaXPPC0v8_52RC1YEUu6OJKzRTGvtug

13] https://github.com/GoodNotes/GNHK-dataset

14] https://link.springer.com/content/pdf/10.1007/978-3-030-86337-1_27.pdf

15] https://machinelearningmastery.com/how-to-use-transfer-learning-when-developing-convolutional-neural-network-models/

16] https://github.com/vicjicaman/repoflow-tool

17] MNIST extended: a dataset for semantic segmentation and object detection - Away with ideas

18] GitHub - ultralytics/yolov5: YOLOv5 🚀 in PyTorch > ONNX > CoreML > TFLite

19] Roboflow: Give your software the power to see objects in images and video

Thanks