CONTENT

INTRODUCTION

Project Overview

Lesions on the retina caused by “Diabetic Retinopathy” (DR), a frequent complication of diabetes mellitus, impair vision. Blindness may result if it is not identified in its early stages. Unfortunately, DR cannot be reversed, and treatment only keeps vision intact. The risk of vision loss can be significantly decreased with early detection and treatment of DR. In contrast to computer-aided diagnosis methods, the manual diagnosis process of DR retina fundus images by ophthalmologists is time-, effort-, and cost-consuming as well as prone to error.

One of the most popular methods for enhancing performance across a wide range of domains, particularly in the classification and image analysis of medical images, is “Transfer Learning”. We used Transfer Learning techniques that are more frequently used in medical image analysis, such as Inception V3, Resnet 50, and Xception V3, and they are very successful.

Purpose

The seriousness of the condition raises the necessity of the modern solution. And the biggest problem that lies in the contemporary practices is that, to diagnose the case we need help from a clinical professional to take a fundus image of the patient and analyze manually to detect the presence of the condition. The requirement of a clinical professional to be present in order to diagnose the patient can be nullified using modern technology. And that solution can be achieved through machine learning or deep learning.

LITERATURE SURVEY

Title - Year	**Authors**	**Description**	**Merits**
Preprocessing and Feature Extraction for Early Detection of Diabetc Retinopathy - **2017**	Dilip Singh Sisodia, Shruti Nair, Pooja Khobragade	Preprocessing of retinal images using extraction of green channel, CLAHE, image enhancement, and preprocessing techniques.	The paper's significance lies in extracting the green channel of the image and discusses its importance.
Image Preprocessing in classifying and identification of Diabetic Retinal Eye diseases - **2021**	Rubina Sarki, Khandakar Ahmed, Hua Wang, Yanchun Zhang, Jiangang, Kate Wang	Study and comparison of image preprocessing techniques for Diabetic eye disease classification.	Using the green channel extraction on CLAHE yields better results than a normal AHE
Comparison of different preprocessing methods used for retinal fundus images - **2017**	Anoop Balakrishnan Kadan	Comparison of different preprocessing methods on fundus images	Median Filtering and Adaptive Histogram Equalization are concluded to be the most effective preprocessing techniques
Comparison of different preprocessing methods on fundus images for early diagnosis of Glaucoma - **2022**	S.Rathinam, S.Selvarajan	Comparison and Evaluation of different preprocessing techniques	Median filter and Average Filter are concluded to be the most effective pre-processing techniques.
A convolutional neural network for the screening and staging of diabetic retinopathy - **2020**	Mohamed Shaban,Zeliha Ogur,Ali Mahmoud,Andrew Switala,Ahmed Shalaby,Hadil Abu Khalifeh,Mohammed Ghazal	A prior pre-processing stage was deployed where image resizing and a class-specific data augmentation were used.	The proposed approach is considerably accurate in objectively diagnosing and grading diabetic retinopathy.

Existing problem

- A sophisticated interface between the patient and the clinical professional. Like a web application where both the stakeholders will be able to access necessary resources.
- Accuracy of the existing solution. With the advancement of AI technology, we can make use of state of the art deep learning algorithms to achieve best results.
- Lack of quality preprocessing which would extensively boost the performance of the solution.

References

- Sisodia D. S, Nair S, Khobragade P. Diabetic Retinal Fundus Images: Preprocessing and Feature Extraction for Early Detection of Diabetic Retinopathy. Biomed Pharmacol J 2017;10(2).
- Sarki, R., Ahmed, K., Wang, H. et al. Image Preprocessing in Classification and Identification of Diabetic Eye Diseases. Data Sci. Eng. 6, 455–471 (2021). https://doi.org/10.1007/s41019-021-00167-z
- (Comparison of Image Preprocessing Techniques on Fundus Images - IJSER Journal Publication, 2022)
- Kadan, Anoop. (2017). Comparison of different image preprocessing methods used for retinal fundus images. 10.1109/ICEDSS.2017.8073677.
- V. Aslantas and M. Tunckanat, "Differential Evolution Algorithm For Segmentation Of Wound Images," 2007 IEEE International Symposium on Intelligent Signal Processing, 2007, pp. 1-5, doi: 10.1109/WISP.2007.4447606.

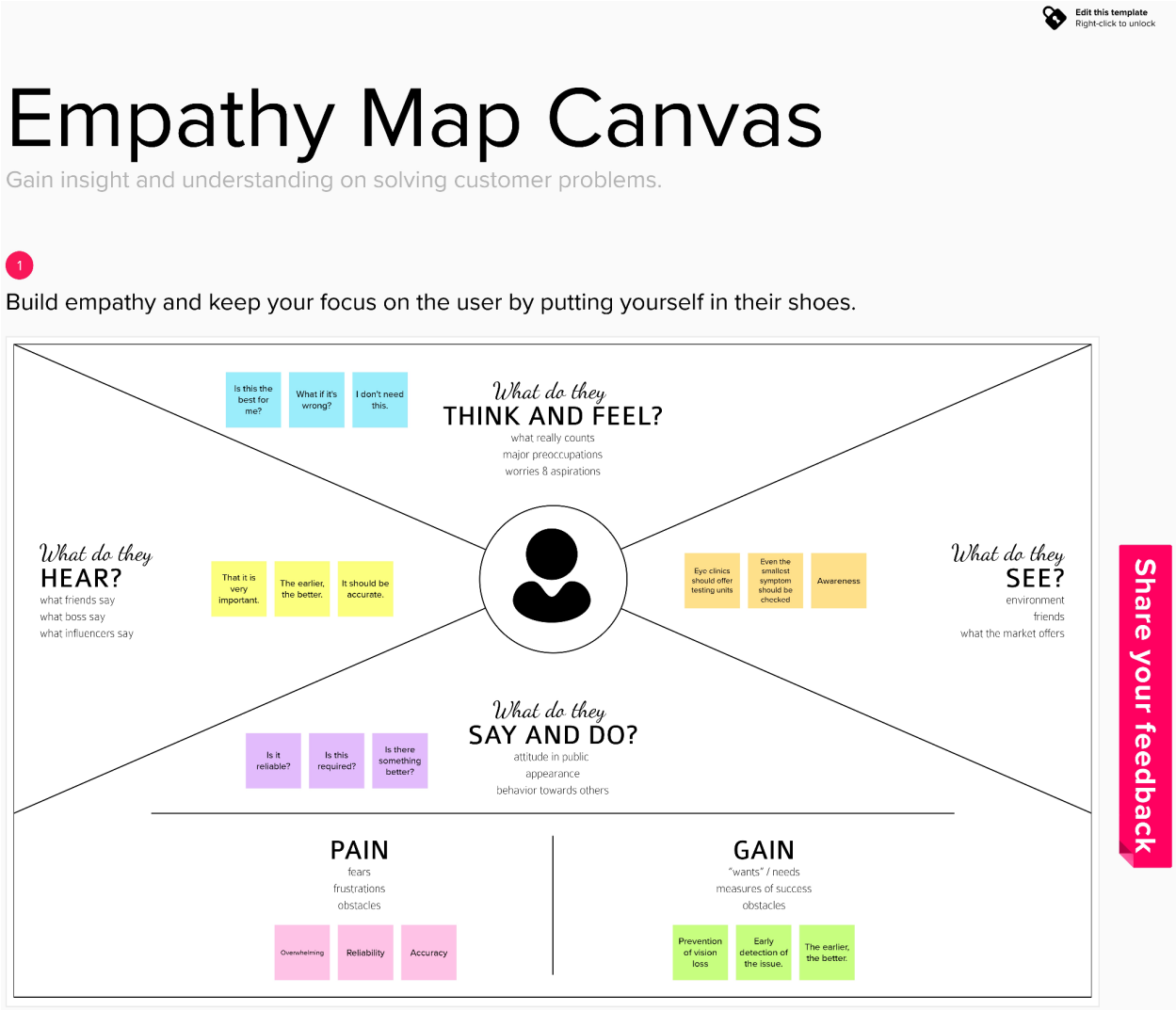
Problem Statement Definition

Diabetic Retinopathy(DR) is a condition that can develop in your eye if you are diagnosed with type 1 or type 2 diabetes. It damages a part of one’s eye called the retina, which is the tissue that lines the back of one’s eye. Signals transmitted from the retina to the brain allow one to see. High blood sugar levels due to diabetes cause

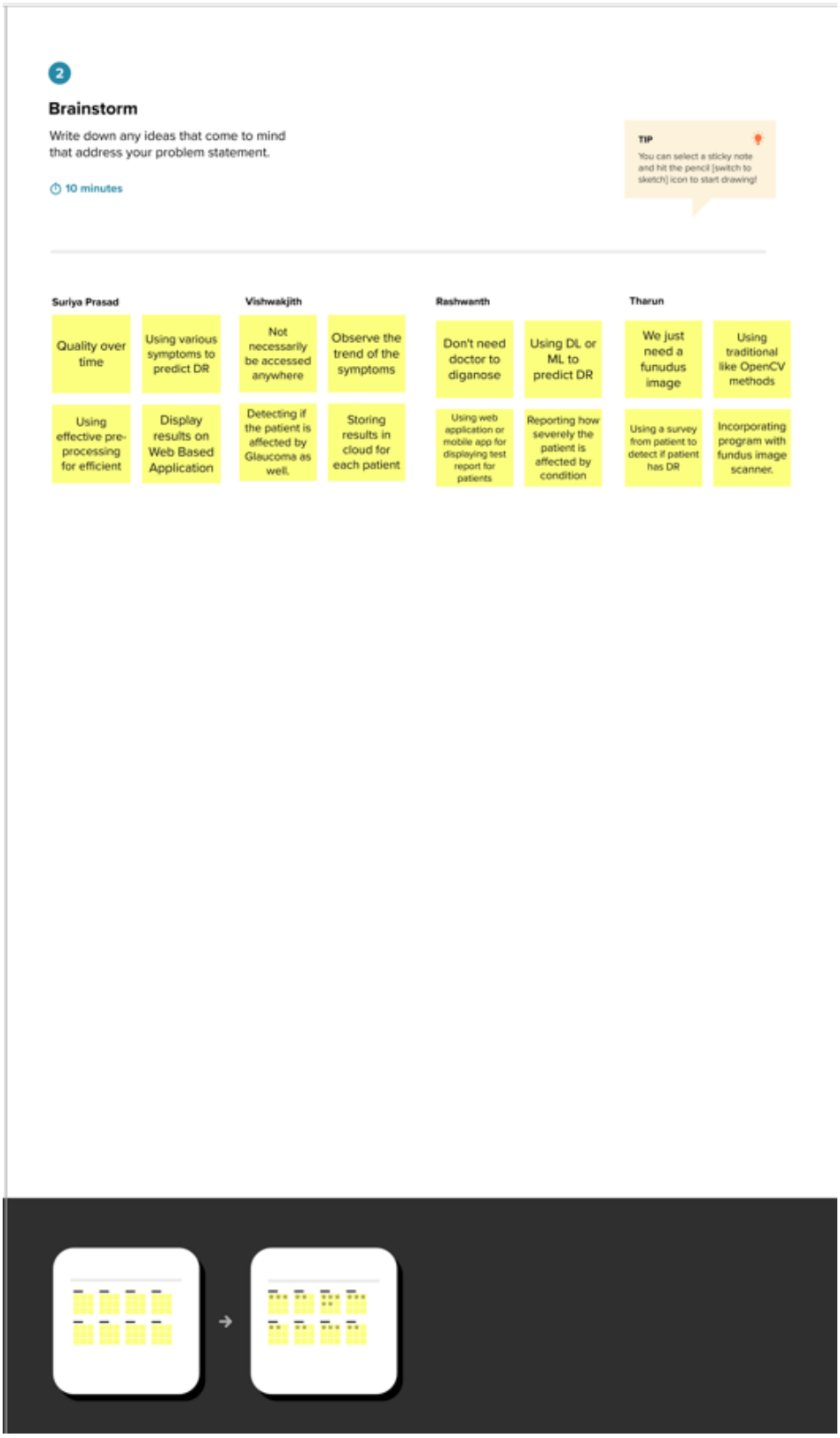
damage to its blood vessels. This initially induces partial vision loss and over time one can lose their eyesight completely. The good news is that, when the symptoms of the patient are discovered in the early stages, with proper treatment one can prevent further vision loss. So it is very important for us to detect diabetic retinopathy in its early stages and prevent the patient from further damaging their eyesight. Hence, we propose a solution wherein if the fundus image, an image which displays the interior surface of the eye, is available, this problem can be easily detected by running a machine learning or deep learning model which can predict the presence of Diabtetic Retinopathy in the early stages.

IDEATION & PROPOSED SOLUTION

Empathy Map Canvas



Ideation & Brainstorming



3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

TIP

Add customizable tags to sticky notes to make it easier to find, browse, organize, and categorize important ideas as themes within your mural.

Frontend Development

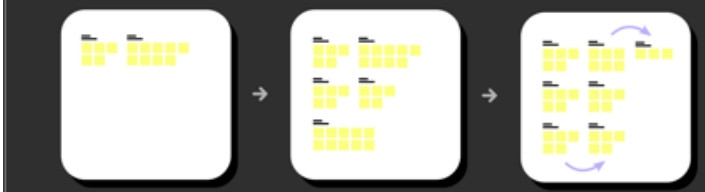
- Display results on Web Based Application
- Storing results in cloud for each patient
- Using web application or mobile app for displaying test report for patients

Scalability

- Incorporating program with fundus image scanner.
- Reporting how severely the patient is affected by condition
- Detecting if the patient is affected by Glaucoma as well.

Prediciton Technique

- Using various symptoms to predict DR
- Observe the trend of the symptoms
- Using DL or ML to predict DR
- Using traditional like OpenCV methods
- Using effective pre-processing for efficient
- Using a survey from patient to detect if patient has DR

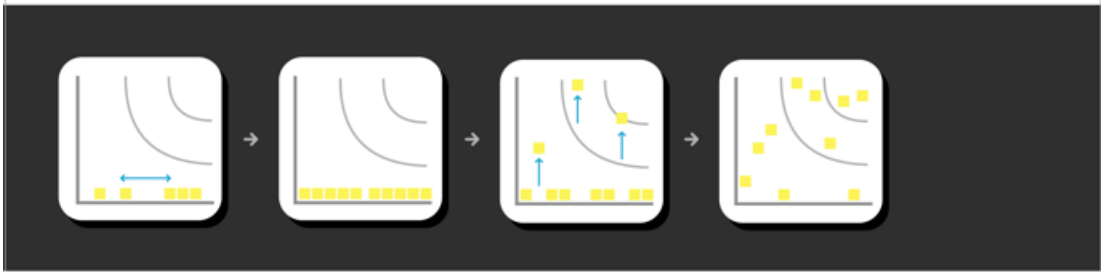


4

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

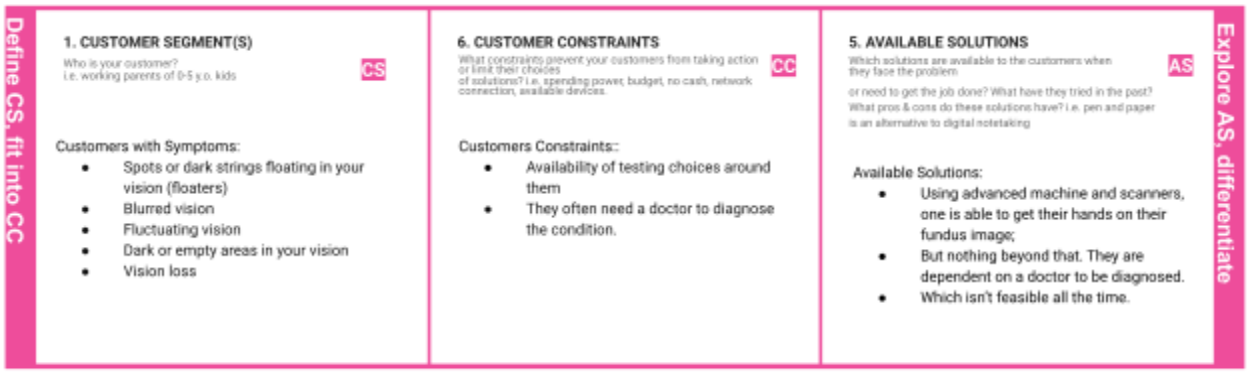
20 minutes

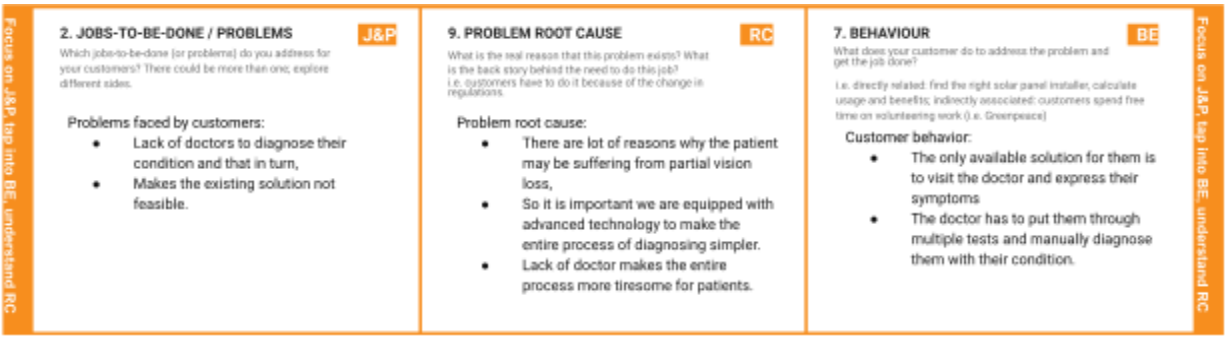


Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Diabetic Retinopathy(DR) is a condition that can develop in your eye if you are diagnosed with type 1 or type 2 diabetes. It damages a part of one's eye called the retina, which is the tissue that lines the back of one's eye. When DR is detected at early stages, we can prevent the patient from completely losing their eyesight.
2.	Idea / Solution description	We propose a solution wherein if the fundus image, an image which displays the interior surface of the eye, is available, this problem can be easily detected by running a machine learning or deep learning model which can predict the presence of DR in the early stages.
3.	Novelty / Uniqueness	● Displaying test results or the report on a web based portal for each and every patient. ● Try a different Computer Intelligence technique and compare the performance with contemporary solutions.
4.	Social Impact / Customer Satisfaction	● Patients who are affected by DR at early stages can prevent further loss of vision. ● Without the need of a doctor, the program would be able to diagnose/predict whether the patient is affected.
5.	Business Model (Revenue Model)	● A program to be incorporated with the system that pictures the fundus image. ● Hospitals would make use of this program. ● Ad revenue from the portal through which we would share the test results.
6.	Scalability of the Solution	● Can also detect if the patient is affected by Glaucoma or not, since both the tests make use of the same fundus image.

Problem Solution fit





3. TRIGGERS

TR

What triggers customers to act? i.e. seeing their neighbor installing solar panels, reading about a more efficient solution in the news.

Partial loss of vision which can be blurred or fluctuating. These symptoms trigger the patient to visit the doctor to be diagnosed.

10. YOUR SOLUTION

SL

If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality.
If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.

The solution is to use a computer intelligence technique like deep learning to diagnose the existence of DR using a fundus image. The early detection of DR and that too without the help of a doctor could play a significant role in the feasibility of the entire process of testing for a patient. And getting their results on their diagnoses immediately also plays a crucial role for the patient to decide the next steps accordingly.

8. CHANNELS of BEHAVIOUR

CH

8.1 ONLINE

What kind of actions do customers take online? Extract online channels from #7

8.2 OFFLINE

What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development.

There is nothing a customer can do online regarding this issue. They are completely reliant on professionals, clinics and hospitals. So they have to deal with their situation offline. Initially a test has to be taken, a fundus image of their eye which would be taken to a professional, a doctor, who would diagnose the presence of the condition of DR. And none of this process can be done by the customer themselves, nor use any online modes to alternate.

4. EMOTIONS: BEFORE / AFTER EM

How do customers feel when they face a problem or a job and afterwards?
i.e. lost, insecure > confident, in control - use it in your communication strategy & design.

Before: The patient is unsure about their problem and that might make them panic.

After: Diagnosing the patient with their condition will give them clarity to decide on their treatment, especially something like DR which can't be fixed and only be prevented from getting worse.

REQUIREMENT ANALYSIS

Functional requirement

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	User registration via form using email ID and Password
FR-2	User Confirmation	Confirmation via Email
FR-3	User Login	Using the registered email ID and Password as login credentials
FR-4	Profile Dashboard	Viewing the profile, Changing Password and Previous Records
FR-5	Checking For Diabetic Retinopathy	Uploading the FUNDUS Image of the eye to predict the disease
FR-6	User Tracking	Maintaining the Record of how far the eye has been affected
FR-7	Feedback & Support	Collection of feedback whether the results were accurate enough

Non-Functional requirements

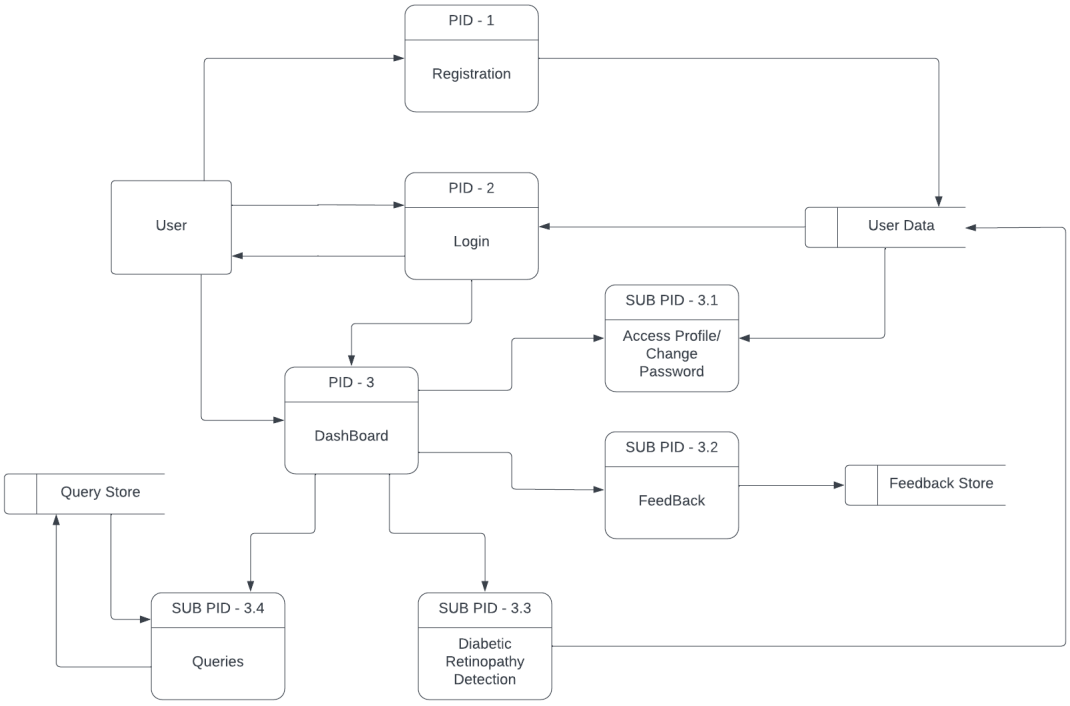
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	● The system should have a quality attribute that must be able to assess the ease of the usage of the UI. ● The system should not expect any technical prerequisites from the user's side.
NFR-2	**Security**	● User details and login credentials should be safe and secure. ● The confirmation of a valid user is required for authentication.
NFR-3	**Reliability**	● Portable and cross-platform independent. ● The application should be subjected to an experiment, test, or measuring procedure that yields the same results on repeated trials. ● Easy to use and flexible.
NFR-4	**Performance**	● The system should handle the traffic efficiently and service requests while consuming less bandwidth. ● The accuracy of the result of a measurement, calculation, or specification should be dependent on the datasets. ● The page should not take a lot of time to load the contents and display them.
NFR-5	**Availability**	● The version of the application should be available even at the time of maintenance and updating. ● The system should run 24 hours a day, 7 days a week [24/7 available].
NFR-6	**Scalability**	● The application should be in the way of adding new functionalities or modules without affecting the existing functionalities. ● The system should be able to manage numerous users at a time and be less prone to errors.

PROJECT DESIGN

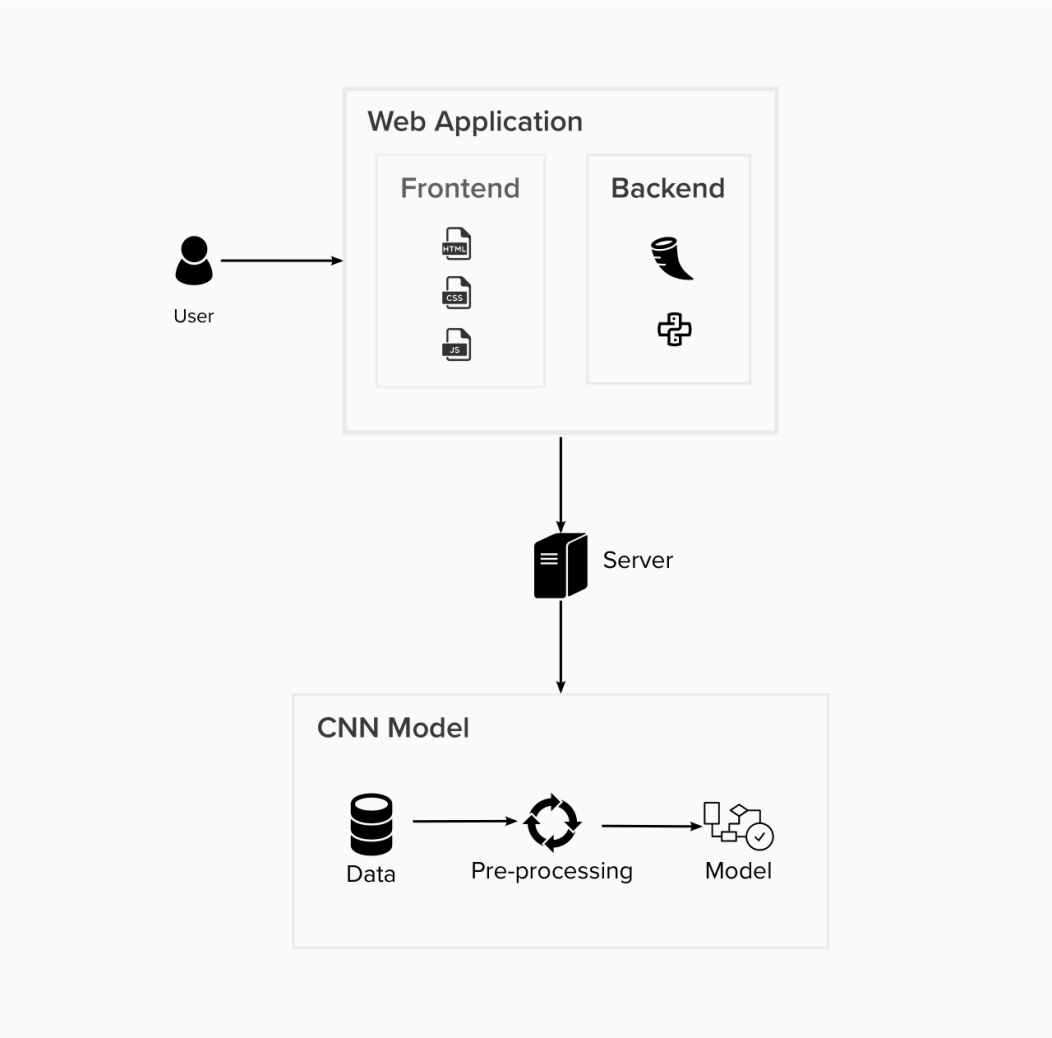
Data Flow Diagrams

0 - Level Dataflow Diagram

1 - Level Dataflow Diagram



Solution & Technical Architecture



User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	MUSN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		MUSN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1

		MUSN-3	Validation of the user can be done directly using email or OTP.	Account validated and got access to profile dashboard.	Medium	Sprint-1
	Login	MUSN-4	Enter the username and password to login to the application.	Right account credentials should be entered.	High	Sprint-1
Customer (Web user)	Dashboard	WUSN-1	Users can access their information like age, gender, previous history etc..	Monitoring Diabetes level	High	Sprint-1
		WUSN-2	Users can upload the FUNDUS image and obtain the results.	Prediction of DR	High	Sprint-2
Customer Care Executive	Overview	CCE-1	As a customer, I can view the data in the graphical representation.	The level of diabetes as well as the DR are analysed	High	Sprint-2
	Database Analysis	CCE-2	As an operator, I should be provided with the image.	Manual detection of DR	High	Sprint-2
Support Team	Support	ST-1	Responds to user queries via telephone or email.	Queries can be raised in certain situations.	Medium	Sprint-3
Administrator	User Control and help	AN-1	Helping users in case of any technical issues faced by them like forgetting the password, collecting feedback and improving the UI.	Maintaining the application's functionality and business	Medium	Sprint-4

PROJECT PLANNING & SCHEDULING

Sprint Planning & Estimation

Release	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Data collection	Task-1	To build a Deep learning Model which begins with the process of splitting data into training and testing set	7	High	Tharun, Suriya Prasad
Sprint-1	Data preprocessing	Task-2	We import the required libraries for preprocessing. We instantiate the ImageDataGenerator class to configure and augment different types of image data	4	High	Vishwakjith, Raswanth
Sprint-1	Data preprocessing	Task-3	Application of the ImgaeDataGenerator to the Train and Test Set.	6	Medium	Tharun
Sprint-1	Building Homepage	USN-1	As a user, she will be given a brief description in the homepage.	3	High	Suriya Prasad, Raswanth
Sprint-2	Feature Extraction	Task-4	Build a CNN Model and only use it as a feature extraction by freezing the convolution blocks.	8	High	Tharun, Raswanth
Sprint-2	Building the layers	Task-5	Adding of dense layers with the aid of Keras. Addition of Optimizer, choosing loss function and the Metrics.	6	High	Vishwakjith, Tharun
Sprint-2	Train, Save, Test	Task-6	To train the model with the configured neural network and save the model. Test the built model against the testing dataset.	5	High	Suriya Prasad, Vishwakjith
Sprint-2	Building Registration Page	USN-2	As a user, she will be able to register for the application	7	High	Suriya Prasad
Sprint-3	Create Service Instance	Task-7	Configure the location of resources, such as web server,	4	Medium	Tharun, Vishwakjith

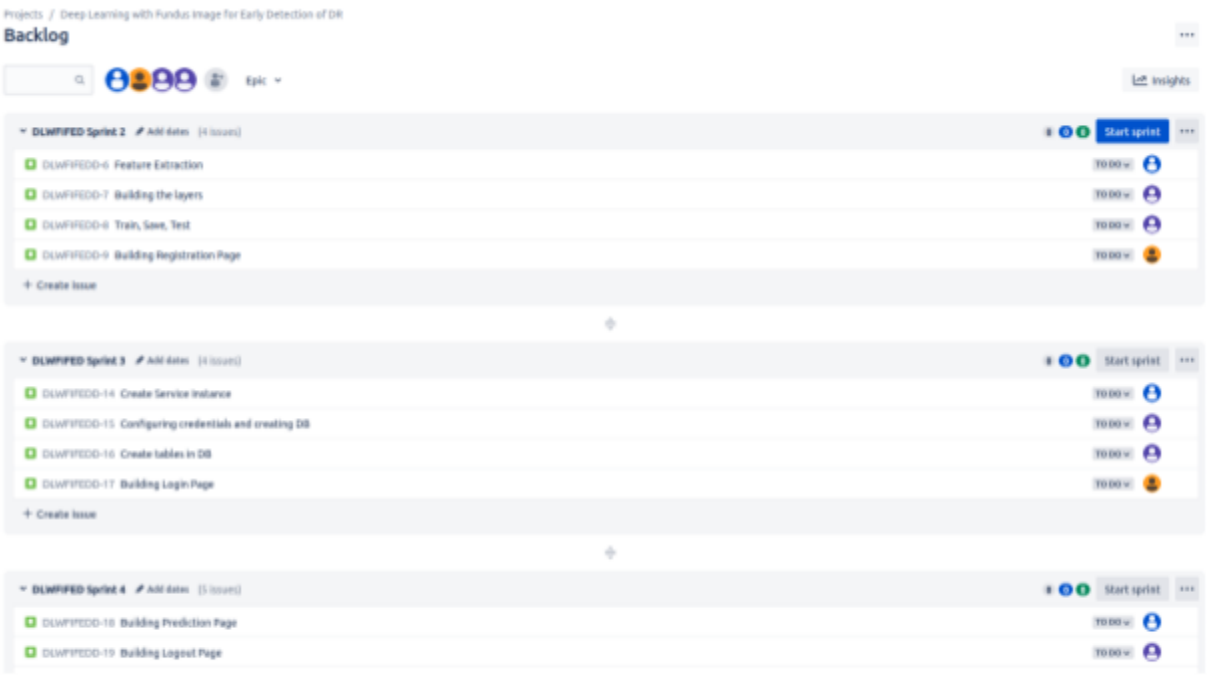
			and Cloud Storage for an application.			
Sprint-3	Configuring credentials and creating DB	Task-8	Define the credentials that are required to access the services offered by IBM Cloudant and add users to access the DB.	4	High	Vishwakjith, Raswanth
Sprint-3	Create tables in DB	Task-9	structure the required tables with necessary attributes in Cloudant DB.	5	Medium	Tharun
Sprint-3	Building Login Page	USN-3	As a user, she will be able to login using her credentials	6	High	Suriya Prasad, Raswanth
Sprint-4	Building Prediction Page	USN-4	As a user, she will be able to receive the diagnosis on her diabetic retinopathy.	4	High	Tharun, Raswanth
Sprint-4	Building Logout Page	USN-5	As a user, she will be able to logout of her account on this page.	5	High	Vishwakjith, Tharun
Sprint-4	Build Python Code	Task-9	Import the libraries and Initialise the necessary modules	7	High	Suriya Prasad, Vishwakjith
Sprint-4		Task-10	Use the database using initiated client and rendering HTML pages	8	Medium	Suriya Prasad, Vishwakjith
Sprint-4		Task-11	Configuring the registration, login pages and validating the credentials	5	High	Tharun, Suriya Prasad
Sprint-4		Task-12	Showcasing the model's prediction on UI.	4	High	Vishwakjith, Raswanth
Sprint-4	Run the application	Task-13	Run the application in the anaconda prompt to check the application	5	Medium	Tharun
Sprint-4		Task-14	In the homepage, after logging on using credentials, upload the image to predict the diagnosis on diabetic retinopathy.	4	High	Suriya Prasad, Raswanth

Sprint-4	Train Moden on IBM	Task-15	train the model on IBM and integrate it with the flask Application.	3	High	Suriya Prasad, Vishwakjit h

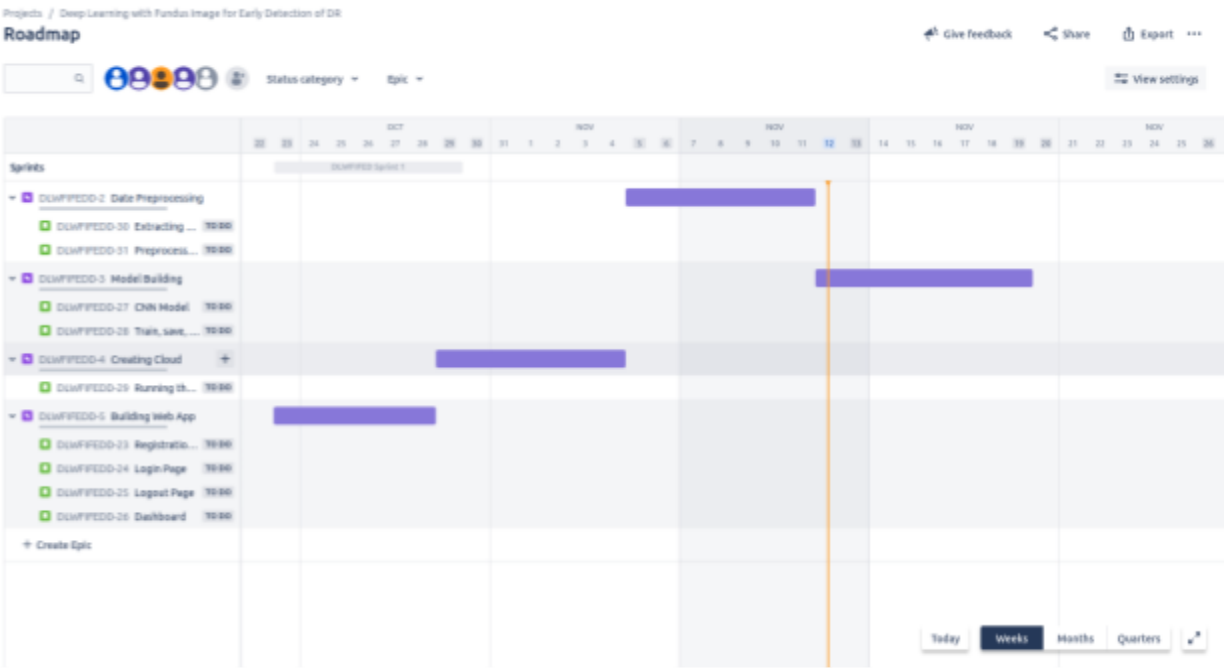
Sprint Delivery Schedule

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

Reports from JIRA



Backlog in JIRA



Roadmap in JIRA

CODING & SOLUTION

Severity Detection

The patient/user can login into the website and upload a fundus image to detect the condition of Diabetic Retinopathy. There are also additional benefits of knowing how severe the condition is. With this feature they can further understand their situation and act accordingly. And diagnosing the severity of the case is classified under five different groups: not affected, mild, moderate, severe and proliferative.

Efficiency (Time taken to provide results)

The patient/user can login into the website and upload a fundus image to detect the condition of Diabetic Retinopathy. It is important for the design of the system to provide results for the user instantaneously. So the model is designed to be efficient and fast in fetching back the necessary information for the user/patient. The results are displayed on the user/patient's screen within seconds, and it only depends on the internet speed.

Database Schema (if Applicable)

TESTING

Test Cases

Test case ID	Feature Type	Component	Test Scenario	Steps To Execute	Expected Result	Status
LoginPage_TC_OO1	Functional	Login Page	Verify users are able to see the Login/Registration popup when the user enters the site.	1.Enter URL and click go 2.Verify login/Singup popup displayed or not	Login/Signup popup should display	Pass
LoginPage_TC_OO2	UI	Login Page	Verify the UI elements in Login/Signup popup	1.Enter URL and click go 2.Verify login/Singup popup with below UI elements: a.name text box b.email text box c.password text box d.Login button e.New customer? Registration link	Application should show below UI elements: a.email text box b.password text box c.Login button with orange colour d.New customer? Create account link e.Last password? Recovery password link	Pass
LoginPage_TC_OO3	Functional	Login page	Verify user is able to log into application with Valid credentials	1.Enter URL and click go 2.Enter Valid username/email in Email text box 3.Enter valid password in password text box 4.Click on login button	User should be able to navigate to user account homepage.	Pass
LoginPage_TC_OO4	Functional	Login page	Verify user is able to log into application with InValid credentials	1.Enter URL and click go 2.Enter InValid username/email in Email text box 3.Enter valid password in password text box 4.Click on login button	Application should show 'Incorrect email or password ' validation message.	Pass
LoginPage_TC_OO5	Functional	Login page	Verify user is able to log into application with InValid credentials	1.Enter URL and click go 2.Enter Valid username/email in Email text box 4.Enter Invalid password in password text box 5.Click on login button	Application should show 'Incorrect email or password ' validation message.	Pass

HomePage_TC_OO1	UI	Home page	Verify user is able to navigate to the Prediction page	1.Enter URL and click go 2.Enter Home Page using valid credentials 3.Click on the Prediction button to navigate to Prediction page.	Prediction Page should be displayed	Pass
HomePage_TC_OO2	UI	Home page	Verify user is able to navigate to the Logout Page	1.Enter URL and click go 2.Enter Home Page using valid credentials 3.Click on the Logout button	Logout Page should be displayed	Pass
PredictionPage_TC_OO1	UI	Predicito n Page	Verify user is able to navigate to the Logout Page	1.Enter URL and click go 2.Enter Home Page using valid credentials 3.Enter on the Prediction Page 4.Click on the Logout button	Logout Page should be displayed	Pass
PredictionPage_TC_OO2	UI	Predicito n Page	Verify user is able to navigate to the Home Page	1.Enter URL and click go 2.Enter Home Page using valid credentials 3.Enter on the Prediction Page 4.Click on the Home button	Logout Page should be displayed	Pass
PredictionPage_TC_003	Funciton	Predicito n Page	Verify user is able to upload an image	1.Enter URL and click go 2.Enter Home Page using valid credentials 3.Enter on the Prediction Page 4.Upload an image in the given input box 5.Click Submit	Image is uploaded	Pass
PredictionPage_TC_004	Funciton	Predicito n Page	Verify user is not able to upload any other image formats	1.Enter URL and click go 2.Enter Home Page using valid credentials 3.Enter on the Prediction Page 4.Upload an image in the given input box 5.Click Submit	Image is uploaded	Fail
LogoutPage_TC_001	UI	Logout Page	Verify User is able to navigate to Login Page	1.Enter URL and click go 2.Enter Home Page using valid credentials 3.Enter on the Logout Page 4.Click on the Login button	Login Page should be displayed	Pass

User Acceptance Testing

Defect Analysis

Resolution	Severity1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	10	4	2	2	18
Fixed	6	2	2	2	12
Skipped	0	0	0	1	1
Won't Fix	0	0	0	1	1
Totals	16	6	4	6	32

Test Case Analysis

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	8	0	0	8
Client Application	17	0	0	17
Security	2	0	0	2
Exception Reporting	2	0	0	2
Final Report Output	5	0	0	5
Version Control	1	0	0	1

RESULTS

Performance Metrics

S.No.	Parameter	Values	Screenshot
1.	Model Summary	Total Parameters:21,885,485 Trainable Parameters:1,024,005 Non-trainable Parameters:20,861,480	Attached Below
2.	Accuracy	Training Accuracy:0.7188 Validation Accuracy:0.7452	Attached Below
3.	Confidence Score	Class Detected:N/A Confidence Score: N/A	N/A

Screenshots:

```
x=Flatten()(xception.output)
pred=Dense(5,activation='softmax')(x)
model=Model(inputs=xception.input,outputs=pred)
model.summary()
```

```
Output exceeds the size limit. Open the full output data in a text editor
Model: "model"
_________________________________________________________________________________________________
 Layer (type)                Output Shape              Param #     Connected to
=================================================================================================
 input_1 (InputLayer)        [(None, 299, 299, 3       0           []
                             )]

 block1_conv1 (Conv2D)       (None, 149, 149, 32       864         ['input_1[0][0]']
                             )

 block1_conv1_bn (BatchNormaliz  (None, 149, 149, 32   128         ['block1_conv1[0][0]']
 ation)                      )

 block1_conv1_act (Activation)  (None, 149, 149, 32    0           ['block1_conv1_bn[0][0]']
                             )

 block1_conv2 (Conv2D)       (None, 147, 147, 64       18432       ['block1_conv1_act[0][0]']
                             )

 block1_conv2_bn (BatchNormaliz  (None, 147, 147, 64   256         ['block1_conv2[0][0]']
 ation)                      )

 block1_conv2_act (Activation)  (None, 147, 147, 64    0           ['block1_conv2_bn[0][0]']
                             )

...
Total params: 21,885,485
Trainable params: 1,024,005
Non-trainable params: 20,861,480
```

Model Training Accuracy

```
model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
r=model.fit(training_set,validation_data=testing_set,epochs=40,steps_per_epoch=len(training_set)//32,validation_steps=len(testing_set)//32)
```

```
Output exceeds the size limit. Open the full output data in a text editor
Epoch 1/40
3/3 [==============================] - 52s 15s/step - loss: 12.2394 - accuracy: 0.3750
Epoch 2/40
3/3 [==============================] - 43s 13s/step - loss: 8.6126 - accuracy: 0.5417
Epoch 3/40
3/3 [==============================] - 45s 14s/step - loss: 9.1320 - accuracy: 0.4375
Epoch 4/40
3/3 [==============================] - 44s 12s/step - loss: 7.3251 - accuracy: 0.4896
Epoch 5/40
3/3 [==============================] - 45s 14s/step - loss: 6.6415 - accuracy: 0.5938
Epoch 6/40
3/3 [==============================] - 44s 13s/step - loss: 5.1138 - accuracy: 0.5938
Epoch 7/40
3/3 [==============================] - 42s 13s/step - loss: 2.6672 - accuracy: 0.7083
Epoch 8/40
3/3 [==============================] - 44s 12s/step - loss: 3.9675 - accuracy: 0.6354
Epoch 9/40
3/3 [==============================] - 45s 14s/step - loss: 3.7436 - accuracy: 0.6146
Epoch 10/40
3/3 [==============================] - 44s 13s/step - loss: 2.9330 - accuracy: 0.6771
Epoch 11/40
3/3 [==============================] - 47s 14s/step - loss: 3.3618 - accuracy: 0.6562
Epoch 12/40
3/3 [==============================] - 46s 13s/step - loss: 3.5601 - accuracy: 0.6354
Epoch 13/40
...
Epoch 39/40
3/3 [==============================] - 51s 16s/step - loss: 3.9316 - accuracy: 0.5833
Epoch 40/40
3/3 [==============================] - 43s 12s/step - loss: 1.9106 - accuracy: 0.7188
```

Model Validation Accuracy

```
[17]  model.evaluate(testing_set)
```

```
...  23/23 [==============================] - 287s 12s/step - loss: 2.6429 - accuracy: 0.7452

     [2.642930507659912, 0.7452316284179688]
```

ADVANTAGES & DISADVANTAGES

Merits

- With the developed application, anyone in the world with an internet connection will be able to access it and upload the fundus image.
- And on the other hand with the assistance of the Deep Learning techniques, we can quickly diagnose whether a patient has Diabetic Retinopathy.
- And the diagnosis will not only contain the detection of Diabetic Retinopathy by also diagnosing the severity of the case under five different classification: not affected, mild, moderate, severe and proliferative

Demerits

- The accuracy of the classification plays an important role in establishing the model. Since it deals with a very crucial idea, it is necessary to have an accurate working model.
- To deal with this situation we are again forced to rely on the help of the clinical professional.
- The steps to further take after diagnosing the condition are still questionable. Which should be the important concern in the future scope.

CONCLUSION

One of the world's important causes of vision loss is due to Diabetic Retinopathy. And the advancement in technology poses the necessity of developing a solution to address the issue.

So bringing in the most advanced technology like the Deep Learning model to design a solution marks a great beginning for the taken action. Now a user from any part of the world with an internet connection would be able to get clinical level attention using this web application. With the diagnoses he or she should be able to take further steps on how to treat the situation.

Now the future scope of this project will lie on two aspects. Mainly the performance accuracy of the deep learning model and secondly, when diagnosed the provision for the users to take the necessary steps. This should allow the web application to build a complete infrastructure around this problem.

FUTURE SCOPE

The future scope of the project can be proceeded in two aspects: the performance of the deep learning model and the provisions or features to include for the users to proceed with once they are diagnosed with the Diabetic Retinopathy. With this we will be able to create a complete infrastructure around the diabetic retinopathy to provide for the users who cannot access clinical professionals help.

The performance of the deep learning model is crucial as it deals with a very crucial medical matter. If the system detects the patient's case as not affected when he or she actually is affected can topple down the entire purpose of the system. So the accuracy of the system is of the utmost importance for the growth of the project.

Building a complete infrastructure around a web application should include features to also detect Glaucoma and other conditions that can be detected using a fundus image. The other aspect of the infrastructure should enable the users to proceed further with what kind of treatment they can take in order to deal with the condition. This can include provisions to get connected with a clinical professional who can guide the patient with what he or she can do in order to deal with the prevention of the condition getting worse.

APPENDIX

Source Code

```
# import numpy as np
import os
import requests
from flask import Flask, request, render_template, redirect, url_for
from cloudant.client import Cloudant

import numpy as np
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
from tensorflow.keras.applications.inception_v3 import preprocess_input


model = load_model(r"model\Updated-xception-diabetic-retinopathy.h5")

app = Flask(__name__)


#Authenticate using an IAM API key
client =
Cloudant.iam('55a4f815-9a4a-4711-b663-d2733b89f3f9-bluemix','Ga7SGlD639xERt-F6
egdft3j2dNntgT5CelqppKEgSLp', connect=True)


#create a database using an initialized client
my_database = client.create_database('ibm-deeplearning')


# @app.route('/')
# def index():
#         return render_template('index.html')


@app.route('/index')
def home():
        return render_template('index.html')
```

```
@app.route('/')
def index():
	return render_template('login.html')

#registration page
@app.route('/register')
def register():
	return render_template('register.html')


@app.route('/afterreg', methods=['POST'])
def afterreg():

	name = request.form.get('name')
	email = request.form.get('emailid')
	password = request.form.get('pass')

	print(name,email,password)

	data = {
	'_id':email,
	'name':name,
	'psw':password,
	}

	print(data)

	query = {'_id': {'$eq': data['_id']}}

	docs = my_database.get_query_result(query)

	print(docs)

	print(len(docs.all()))

	if(len(docs.all())==0):
	url = my_database.create_document(data)
	return render_template('register.html', pred="Registration successfull, Please login using your details")
	else:
```

```
        return render_template('register.html', pred="You are already a member, Please login using your details")


#login page
@app.route('/login')
def login():
        return render_template('login.html')


@app.route('/afterlogin', methods=['POST'])
def afterlogin():

        user = request.form.get('emailid')
        passw = request.form.get('pass')
        print(user,passw)

        query = {'_id': {'$eq': user}}

        docs = my_database.get_query_result(query)
        print(docs)

        print(len(docs.all()))
        if(len(docs.all())==0):
        return render_template('login.html', pred="The username is not found, please Register")
        else:
        if((user==docs[0][0]['_id'] and passw==docs[0][0]['psw'])):
        return render_template('index.html')
        else:
        print('Invalid User')


@app.route('/logout')
def logout():
        return render_template('logout.html')


#prediction
@app.route('/prediction')
```

```
def prediction():
        return render_template('prediction.html',pred=None)


@app.route('/predict', methods=['POST'])
def predict():

        f = request.files['image']
        basepath = os.path.dirname(__file__) #getting the current path i.e. where app.py is present
        filepath=os.path.join(basepath,'uploads',f.filename)
        f.save(filepath)

        img = image.load_img(filepath,target_size=(299,299))
        x = image.img_to_array(img)
        x=np.expand_dims(x,axis=0)#used for adding one more dimension
        img_data=preprocess_input(x)
        prediction=np.argmax(model.predict(img_data),axis=1)

        print("prediction is", prediction)

        index=['No Diabetic Retinopathy', 'Mild Diabetic Retinopathy', 'Moderate Diabetic Retinopathy', 'Severe Diabetic Retinopathy', 'Proliferative Diabetic Retinopathy']

        res = str(index[prediction[0]])

        color = ''
        if res == 'No Diabetic Retinopathy':
        color = 'style=color:#86c881'
        elif res == 'Mild Diabetic Retinopathy':
        color = 'style=color:yellow'
        elif res == 'Moderate Diabetic Retinopathy':
        color = 'style=color:orange'
        elif res == 'Severe Diabetic Retinopathy':
        color = 'style=color:#ff5500'
        else:
        color = 'style=color:#a00000'

        return render_template('prediction.html',pred=res,color = color)
```

```
if __name__ == "__main__":
        app.run(debug=False)
```

GitHub & Project Demo Link

GitHub Repository: https://github.com/IBM-EPBL/IBM-Project-13087-1659510254

Demonstration Video:
https://github.com/IBM-EPBL/IBM-Project-13087-1659510254/blob/main/Final%20Deliverables/Demonstration%20video.webm