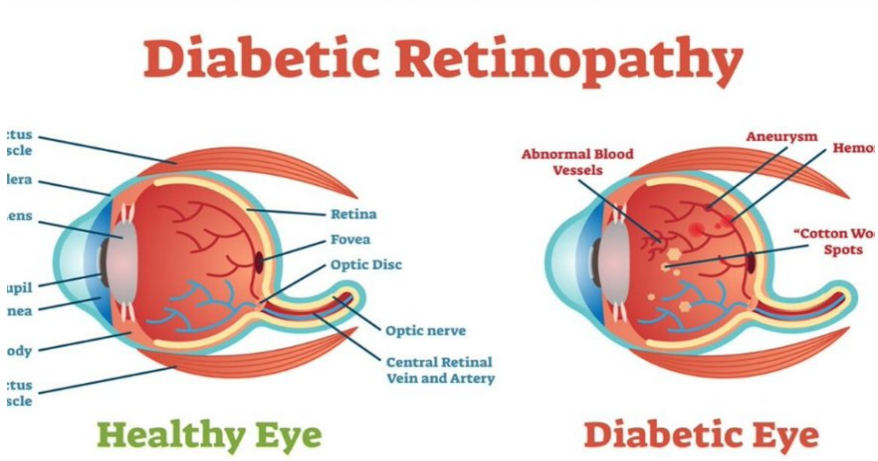
PROJECT DOCUMENTATION

Deep Learning Fundus Image Analysis ForEarlyDetection Of Diabetic Retinopathy



Professional Readiness for Innovation, Employabilityand Entrepreneurship

Team ID : PNT2022TMID36211

110519106034 -Vigneshwaran V

110519106025 - Snega S

110519106027 - Sowmiya R

110519106028 - Sowmiya S

TABLE OF CONTENTS

1. INTRODUCTION :

Project Overview :

Diabetic retinopathy is a diabetes complication that refers to retinal changes that occur in patients with diabetes mellitus. Diabetic retinopathy can develop in anyone who has type 1 or type 2 diabetes. Diabetic retinopathy is caused by damage to the small blood vessels of the light-sensitive tissue at the back of the eye called the retina and can lead to vision loss through several different pathways. It necessitates constant monitoring, and in the event of complications, it may shorten life expectancy. If it is not diagnosed and treated, it can blind you. The medication cannot be cured at this time. Diabetic retinopathy can be stopped or slowed down with treatment. Diabetes management may be used carefully to treat mild cases.

Diabetic Retinopathy {DB) is a complication of diabetes that influences the eyes. Damage to blood vessels in the tissue of the retina, the back layer of the eye, typically causes it. Blurriness, floaters, dark or empty areas in the vision, and difficulty recognizing color blindness are some of the early symptoms. Diabetic retinopathy is one of the most common causes of sight loss among people of working age. You can have diabetic retinopathy and not know it. This is because it often has no symptoms in its early stages.

Purpose :

The most prominent microvascular consequence of diabetes is diabetic retinopathy, and retinal imaging is the most used tool for diagnosing because of its great sensitivity. On the basis of the fundus or retinal images of the patient's eyes, medical professionals today assess the severity and degree of retinopathy associated with a person having diabetes. As the number of diabetic patients rises quickly, more retinal images will be generated by screening programmes, which places a heavy workload on medical professionals and raises the cost of healthcare services. An automation system could help with this both as a mechanism for complete diagnosis or as a complement for the work of scientific professionals.

The application of deep learning models for automated identification of diabetic retinopathy has been examined in two recent publications. Both demonstrate that a deep learning artificial neural network-based automated system can diagnose pertaining diabetic retinopathy, which is classified as moderate or worse eye disease, with significant sensitivity and specificity. Other related eye disorders, such as diabetic macular edema, potential glaucoma, and age-related macular degeneration, have also recently been studied with this method.

An automated system must be able to categorise retinal pictures using severity scales that are clinically applicable, such as the scales for diabetic retinopathy and macular edema that are being suggested internationally and that are being used in Finland. Recent studies for the former example of the diabetic retinopathy scale may be found in the literature, however no studies have been conducted yet to categorise macular alterations with the later scale. The enormous number of annotated photos required for the model to learn is regarded to be a significant impediment to the deeper learning system's wider and more successful application.

The goal in this project is to classify macular edema and diabetic retinopathy using five distinct classification schemes. We provide the most recent findings for the biologically meaningful five categories of classification, as well as the initial four grade macular edema classification, in addition to the prior research of the referable diabetic retinopathy classification system. Additionally, we outline the pre-processing and regularisation processes that must be performed on the data for the deep learning system to perform properly and carefully examine how the scale and quantity of photos used in training affect the system's performance.

2. LITERATURE SURVEY :

EXISITING PROBLEM:

Diabetic Retinopathy (DR) is a degenerative disease that impacts the eyes and is a consequence of Diabetes mellitus, where high blood glucose levels induce lesions on the eye retina. Diabetic Retinopathy is regarded as the leading cause of blindness for diabetic patients, especially the working-age population in developing nations. Treatment involves sustaining the patient's current grade of vision since the disease is irreversible.

Early detection of Diabetic Retinopathy is crucial in order to sustain the patient's vision effectively. The main issue involved with DR detection is that the manual diagnosis process is very time, money, and effort consuming and involves an ophthalmologist's examination of eye retinal fundus images. The latter also proves to be more difficult, particularly in the early stages of the disease when disease features are less prominent in the images.

Machine learning-based medical image analysis has proven competency in assessing retinal fundus images, and the utilization of deep learning algorithms has aided the early diagnosis of Diabetic Retinopathy (DR). This paper reviews and analyzes state-of-the-art deep learning methods in supervised, self-supervised, and Vision Transformer setups, proposing retinal fundus image classification and detection. For instance, referable, nonreferable, and proliferative classifications of Diabetic Retinopathy are reviewed and summarized. Moreover, the paper discusses the available retinal fundus datasets for Diabetic Retinopathy that are used for tasks such as detection, classification, and segmentation. The paper also assesses research gaps in the area of DR detection/classification and addresses various challenges that need further study and investigation.

REFERENCES:

Survey 1 :

AUTHORS: Mohammad Z. Atwany , Abdulwahab H. Sahyoun , And Mohammad Yaqub (March 22).

TITLE: 'Deep Learning Techniques for Diabetic Retinopathy Classification: A Survey.'

METHODS: This paper reviews and analyzes state-of- threat deep learning methods in supervised, self-supervised, and Vision Transformer setups, proposing retinal fundus image classification and detection. For instance, referable, non referable, and proliferative classifications of Diabetic Retinopathy are reviewed and summarized. Moreover, the paper discusses the available retinal fundus datasets for Diabetic Retinopathy that are used for tasks such as detection, classification, and segmentation

Survey 2 :

AUTHORS: Md Mohaimenul Islam, Hsuan-Chia Yang, Tahmina Nasrin Poly, Wen-Shan Jian, Yu-Chuan (2020).

TITLE: 'Deep learning algorithms for detection of diabetic retinopathy in retinal fundus photographs: A systematic review and meta-analysis

METHODS: This project presents a systematic review with a meta-analysis of relevant studies was performed to quantify the performance of DL algorithms to detect DR. The findings of their study showed that DL-algorithms had high sensitivity and specificity for detecting referable DR from retinal fundus photographs.

Survey 3 :

AUTHOR: Akey Sungheetha Kumarasuvamy, Rajesh Sharma Rajendran (2021).

TITLE: 'Design an Early Detection and Classification for Diabetic Retinopathy by Deep Feature Extraction based Convolution Neural Network.'

METHODS: They proposed research work extracts the features by incorporating deep networks through convolution neural networks (CNN). The micro aneurysm may be seen in the early stages of the transformation from normal to sick condition on the images for mild DR. The level of severity of the diabetes condition may be classified by using the confusion matrix detection results.

Survey 4 :

AUTHOR: Sraddha Das, Krity Kharbanda, Suchetha M, Rajiv Raman, Edwin Dhas D (2021).

TITLE: 'Deep learning architecture based on segmented fundus image features for classification of diabetic retinopathy.'

METHODS: In this thesis, they have used a convolution neural network (CNN) to train the classifier for performing classification. The CNN, constructed for classification, comprises a combination of squeeze and excitation and bottleneck layers, one for each class, and a convolution and pooling layer architecture for classification between the two classes. For the performance evaluation of the proposed algorithm, They use the dataset DIARETDB1, comprised of fundus scans of both affected and normal retinas.

Survey 5 :

AUTHOR: Recep Emre Hacisoftaoglu (Dec 2019).

TITLE: 'Deep Learning Frameworks For Diabetic Retinopathy Detection Using Smartphone-Based Retinal Imaging Systems.'

METHODS: In this thesis, we first investigate the smartphone-based portable ophthalmoscope systems available on the market and compare their Field of View and image quality to determine if they are suitable for Diabetic Retinopathy detection during a general health screening. Then, we propose automatic Diabetic Retinopathy detection algorithms for smartphone-based retinal images using deep learning frameworks, AlexNet and GoogLeNet. To test our proposed methods, we generate smartphone-based synthetic retina images by simulating the different Field of View with masking the original image around the optic disk and cropping it.

Survey 6 :

AUTHORS: Tao Li, Yingqi Gao, Kai Wang, Song Guo, Hanruo Liu, Hong Kang. (2019).

TITLE: 'Diagnostic assessment of deep learning algorithms for diabetic retinopathy screening.'

METHODS: They collected 13,673 fundus images from 9598 patients. These images were divided into six classes by seven graders according to image quality and DR level. Moreover, 757 images with DR were selected to annotate four types of DR-related lesions. Finally, we evaluated state-of-the-art deep learning algorithms on collected images, including image classification, semantic segmentation and object detection.

2.3.PROBLEM STATEMENT DEFINITION:

Diabetic Retinopathy (DR) is common complication of diabetes mellitus, which will cause lesions on the retina that affects vision. If it is not detected early, it can lead to blindness. Unfortunately, DR is not a reversible proves, and the given treatment will only give us a sustain vision. DR early detection and treatment can significantly reduce the risk of vision loss.

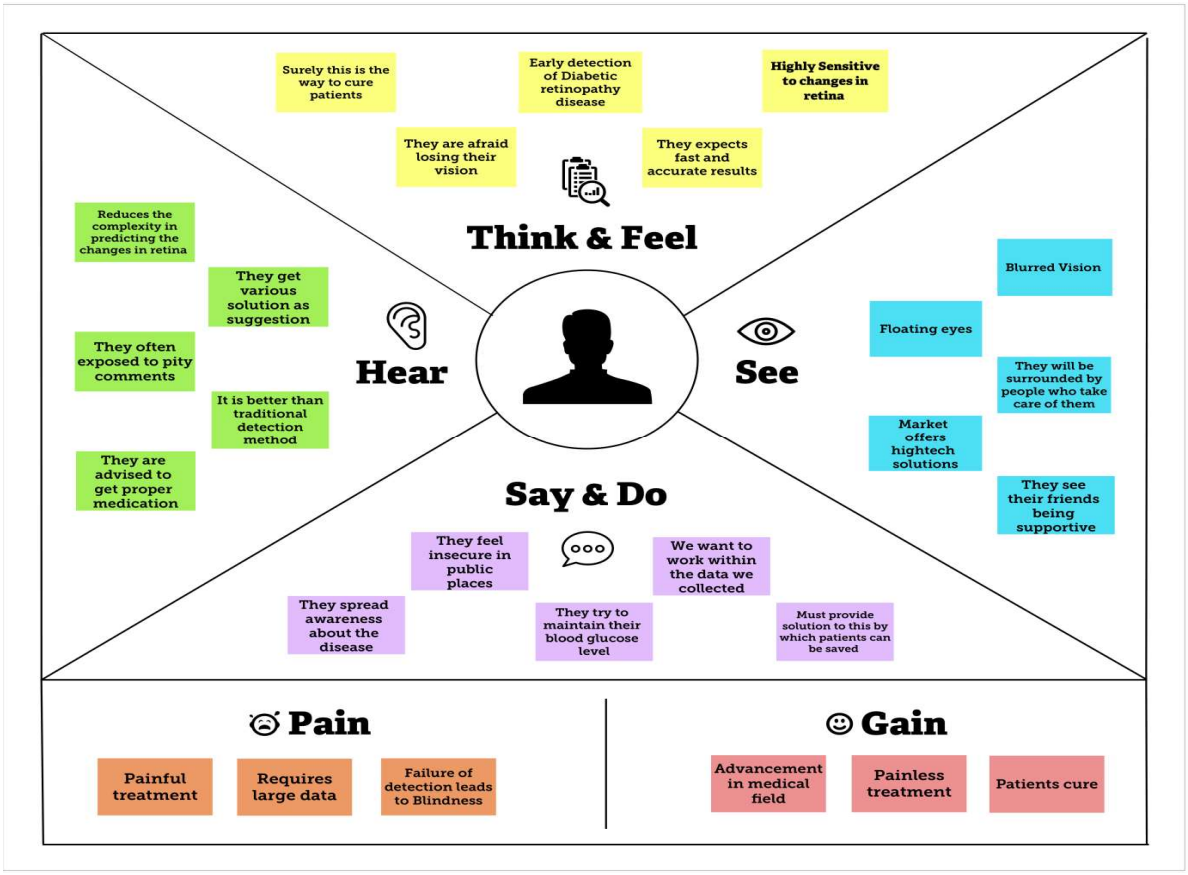
Problem Statement 1:

Problem Statement 2:

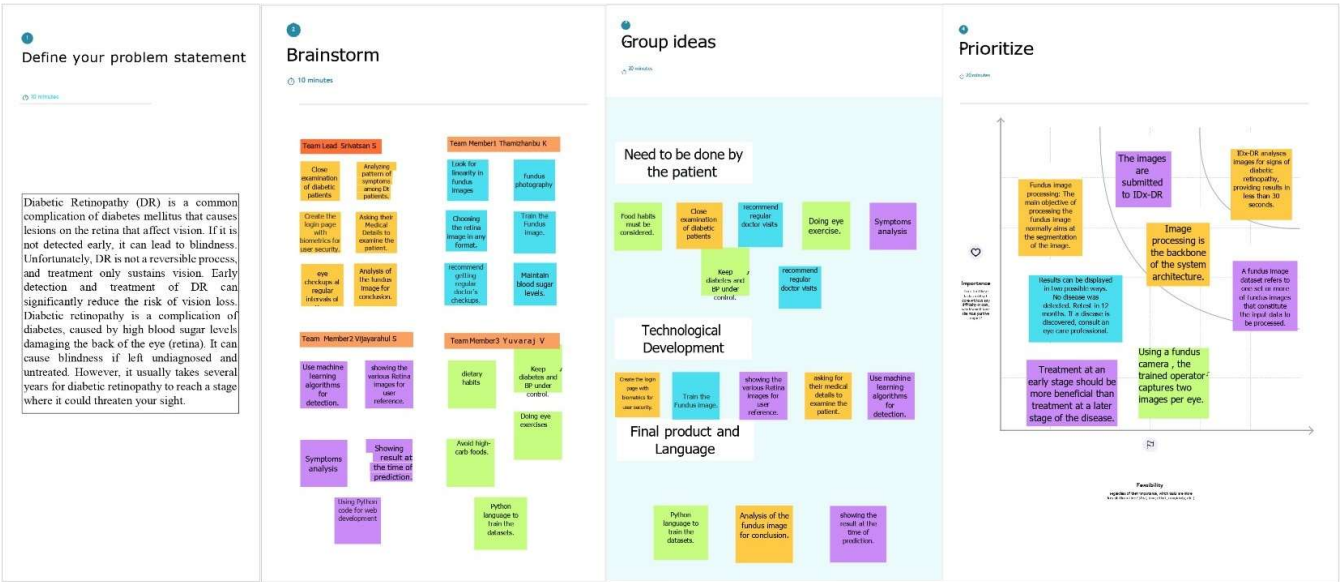
Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me feel
PS 1	Ophthalmalogist	Identify the causes that leads to blurry Vision	Diabetic Retinopathy can be one of the reason	Diabetic Retinopathy can cause permanent loss of vision	Serious
PS 2	a Patient	get screened for Diabetic Retinopathy	I want a better technology which will be pain free	Some techniques cause pain and damage the blood vessels	Anxious

3. IDEATION PHASE & PROPOSED SOLUTION :

Empathy Map Canvas :



IDEATION AND BRAINSTORMING:



PROPOSED SOLUTION:

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Diabetic Retinopathy is a common complication of diabetes mellitus, which causes lesions on the retina that affect vision. If it is not detected early, it can lead to blindness. This project focuses on detecting such underlying lesions that could potentially result in blindness.
2.	Idea / Solution description	Diabetic retinopathy is not a reversible process, and treatment only sustains vision. DR early detection and treatment can significantly reduce the risk of vision loss. The manual diagnosis process of DR retina fundus images by ophthalmologists is time, effort and cost-consuming and prone to misdiagnosis unlike computer-aided diagnosis systems. So, deep learning techniques can be used for early detection of diabetic retinopathy that can prevent blindness and other eye related diseases.
3.	Novelty / Uniqueness	The deep learning system identifies referable diabetic retinopathy comparably or better than presented in the previous studies. Hence, this model provides the patient with the result whether they have serious condition or normal condition. The prediction comes with different levels of illness helps to diagnose properly.
4.	Social Impact / Customer Satisfaction	Since diabetic retinopathy cannot be reversed, early detection prevents many people from losing their vision and developing other serious illnesses. Because the manual screening is more expensive than this model, it is more practical for the people to undergo this screening without any difficulties.

5.	Business Model (Revenue Model)	We can collaborate with the health care centres and diabetic diagnosis centres for regular screening of diabetic retinopathy whenever the diabetic patient comes to check their diabetic level. We can raise awareness because many individuals are unaware of the implications of diabetic retinopathy, which could lead to an increase in screening tests in the future. Integrating this model in hospitals will increase the rate of patients which will automatically increase the organization's revenue. Detecting the problem early for a patient increases the trust and the connection with hospital which leads to a good bond between patients and the hospital.
6.	Scalability of the Solution	The system offers a better solution for diabetic retinopathy and can be detected at an early stage. The model developed using deep learning technology can be implemented on many clinical examinations. It is more adaptable to new images and datasets because it was built with a versatile dataset. It gives higher performance than manual examination. Thus, this system can be used to detect diabetic retinopathy early in real-time for new variations.

PROPOSED SOLUTION FIT

Project Title: Deep Learning Fundus Image Analysis for Detection of Diabetic Retinopathy

Project Design Phase-I - Solution Fit

Team ID: PNT2022TMID36211

Define CS, fit into CC

1. CUSTOMER SEGMENT(S) CS

Who is your customer?
i.e. working parents of 0-5 y.o. kids

Diabetic patients between the age group of 51 to 45

6. CUSTOMER CONSTRAINTS CC

There might be possibility of not getting completely accurate results due to overfitting and underfitting in some special and rare cases further examination may be required

5. AVAILABLE SOLUTIONS AS

Explore AS, differentiate

Diabetic retinopathy is best diagnosed with a comprehensive dilated eye exam For this exam,drops placed in your eyes widen (dilate) your pupils to allow your doctor a better view inside your eyes. The drops can cause your close vision to blur until they wear off After this, Fluorescein Angiography and Optical Coherence Tomography tests are performed for diagnosing Diabetic Retinopathy

2. JOBS-TO-BE-DONE / PROBLEMS J&P

Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides.

The given input image of the patient's retina is analysed and classified using which the class of diabetic retinopathy in the patient's eye is identified

9. PROBLEM ROOT CAUSE RC

What is the real reason that this problem exists?
What is the back story behind the need to do this job?
i.e. customers have to do it because of the change in regulations.

More than 5.2 million people lose eyesight due to diabetic retinopathy when it is diagnosed in its later stages If this condition is diagnosed way earlier than before, it can prevent more patients from going visually impaired

7. BEHAVIOUR BE

What does your customer do to address the problem and get the job done?
i.e. Directly related: find the right solar panel installer, calculate usage and benefits; Indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace)

Directly related: the stage of diabetic retinopathy in a patient can be diagnosed earlier

Indirectly related: volunteers can help in training the model

Identify strong TR & EM

3. TRIGGERS TR

Seeing other medical organization easily identifying their patient's problem and providing the appropriate treatment

4. EMOTIONS: BEFORE / AFTER EM

stressed, anxious >calm, confident
stressed and anxious due the time consumed in the diagnosis

10. YOUR SOLUTION SL

The collected images of the patients retina is analysed and classified using the image processing and deep learning model throught which the type of diabetic retinopathy can be easily identified

8. CHANNELS of BEHAVIOUR CH

Patient tries to look up online for what condition they are suffering from

Patient consults a doctor through an appointment ottime

Identify strong TR & EM

4.REQUIREMENT ANALYSIS:

Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story/ Sub-Task)
FR-1	User Registration	Using a phone number to register signing up with Gmail
FR-2	User Confirmation	Reassurance via OTP mail confirmation
FR-3	Describe what the product does	Before you notice any changes in your vision, our projectcan identify early retinal changes.
FR-4	Focus on user requirements	Reduce the chance of blindness and vision loss in diabetes patients who have retinal complications.
FR-5	Usually defined by the user	A patient's fundus image was obtained.
FR-6	Define product features	A cutting-edge technique for eye screening that allowsfor the early detection of diseases related to the eyes.

Non-functional Requirements:

Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Confirming that a piece of software can successfully carry out one or more specific tasks.
NFR-2	**Security**	Only the system administrator may grant permission.
NFR-3	**Reliability**	Even though the system has the ability to roll back to its original state if a system update fails or thereare bugs in the code.
NFR-4	**Performance**	The loading of an image just takes twoseconds. The model's performance is intended to provide patients with quick results.
NFR-5	**Availability**	The gadget facilitates access, cost, and quality of healthcare.
NFR-6	**Scalability**	Even when several users are utilising the product simultaneously, it must remain reliable.

5. PROJECT DESIGN:

DATA FLOW DIAGRAM:

The classic visual representation of how information moves through a system is a data flow diagram (DFD). The ideal amount of the system needs can be graphically represented by a tidy and understandable DFD. It demonstrates how information enters and exits the system, what modifies the data, and where information is kept.

- Diabetic retinopathy disease is frequently detected and examined using retinal fundus Pre-processing of raw retinal fundus images isperformed using extraction of the green channel, histogram equalization, image enhancement, and resizing techniques.
- One of the main tasks in retinal image processing is the segmentation of the retinal vasculature from images of the eye fundus.

- By omitting the optic disc (OD) region of the retina, the computer-assisted automatic recognition and segmentation of blood vessels.
- Mathematical binary morphological techniques are used to identify the retinal blood vessels.
- The term "feature extraction from the fundus images for the diagnosis of Diabetic Retinopathy" refers to a sophisticated eye screeningtechnique that allows for the early detection of eye-related disorders.

TECHNOLOGY ARCHITECTURE:

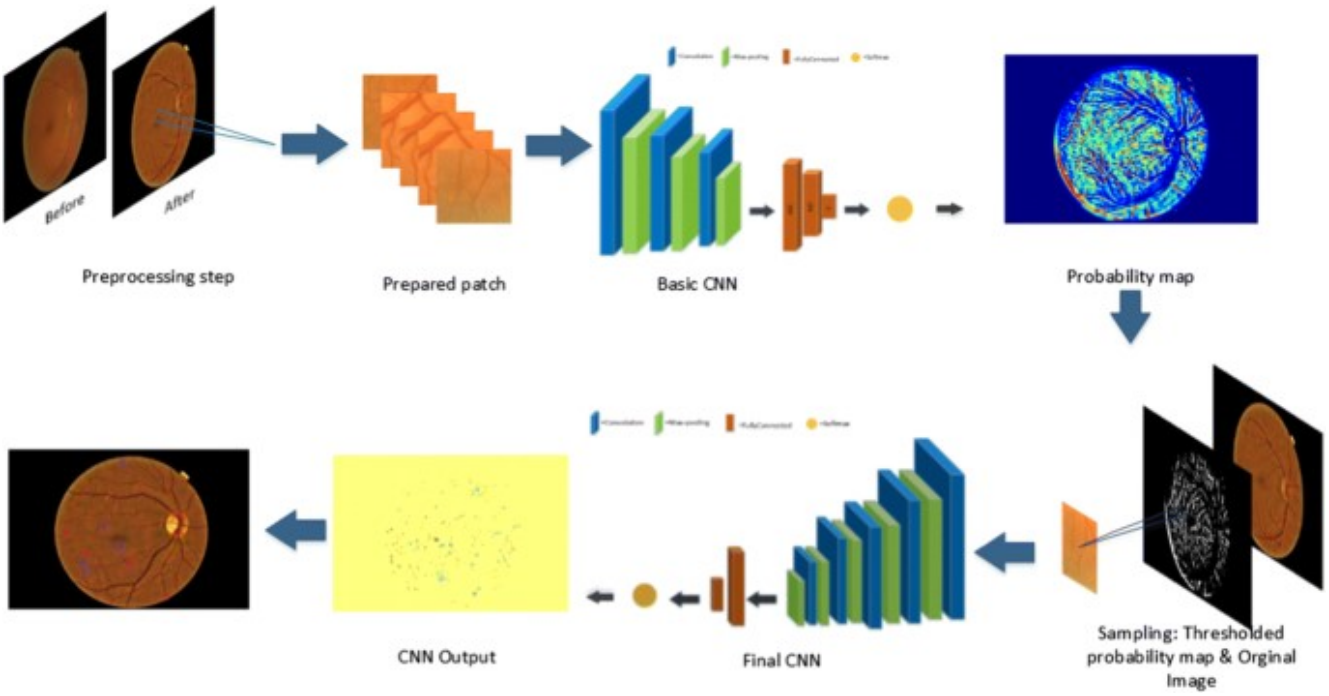


Table-1: Components& Technologies:

1.	User Interface	Web UI	HTML, CSS, JavaScript, Python
2.	Application logic-1	Image Pre-processing	Keras, Tensorflow, Numpy
3.	Application logic-2	CNN Model	Keras, Tensorflow, Numpy
4.	Application logic-3	Web UI Application	Flask
5.	Database	DR Images (Jpeg,Png,Jpg,Etc.,)	Uploads Folder
6.	File storage	File Storage Requirements (Only If Necessary)	IBM Block Storage, Google Drive
7.	External API	Keras	Image Processing API
8.	Deep Learning Model	Inception V3 Architecture	Pre-Trained Convolution Neural Network Model
9.	Infrastructure (Server)	Application Deployment on Webserver	Flask-A Python WSGI HTTP Server.

Table-2: Application characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Flask	Flask Frameworks
2.	Security Implementations	CSRF Protection, Secure Flag For Cookies	Flask-WTF, Session CookieSecure
3.	Scalable Architecture	Micro-Services	Micro Web Application FrameworkBy Flask

USER STORIES:

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Patient (Web user)	Registration	USN-1	I can register as a user on the website with either an email address or a phone number and password.	I can create my account.	High	Sprint-3
	Login	USN-2	With the provided Login credentials, I canaccess the website as a user.	I can log in and access my account.	High	Sprint-3
	Upload image	USN-3	I can post my data as a user in formats likepdf and doc.	I can upload my data.	Medium	Sprint-3
Administration (Web developer)	Admin Login	USN-4	I can log in to the website as the admin and analyse the user information.	I can log in and analyze the user data.	High	Sprint-3
	Data collection	USN-5	I can gather the dataset for the DR from thesource as an admin.	I can collect the dataset.	Low	Sprint-1
	Create model	USN-6	I can build the model and train it using the dataset as an administrator to make predictions.	I can create and train the model.	High	Sprint-1
	Test the model	USN-7	I can evaluate the model's predictive abilities as an admin.	I can test the model.	High	Sprint-2
Patient (Web user)	Diagnosis	USN-8	I can access the application's diagnosis results as a user and continue with treatments.	He/she can get the results and continue the treatment.	High	Sprint-2

6. PROJECT PLANNING AND SCHEDULING:

SPRINT PLANNING AND ESTIMATION:

Sprint	Functional Requirement (Epic)	User Story Number	User Story/ Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email or phone number and password, and confirming my password.	10	High	4
Sprint-1	Dashboard	USN-2	As a user, I will Redirect to the dashboard after registration which shows the importance of DR.	10	Medium	4
Sprint-2	Login	USN-3	As a user, I can log into the application by entering Login credentials.	5	High	3
Sprint-2	Upload Images	USN-4	As a user, I should be able to upload the image of eye Retina.	10	High	2
Sprint-2	Dashboard	USN-5	As a user, based on my requirement I can navigate through the dashboard.	5	Medium	2
Sprint-3	Train the model	Task 1	As a developer, the dataset will be uploaded and trained by developed algorithm.	20	High	4
Sprint-4	Testing & Evaluation	Task 2	As a developer, we tested the trained model using the provided dataset and model will beevaluated for accurate results.	10	High	3
Sprint-4	Display predictedresult	USN-6	As a user, I can view the predicted result in the dashboard.	10	High	4

Sprint	Total story point	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

Velocity:

Imagine we have a 10-daysprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV)per iteration unit (story points per day).

$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$

AV=20/6=3.33points per day.

7. CODING AND SOLUTIONING:-

Feature 1:-

We have devloped a website which authenticates users and help them upload and check the seriousness of the diabetics.

Feature 2:-

We have devloped a multilayer deep convolutional nueral network that classifies the user image of a eye to which extense has the disease diabetics has been affected.The model will classify the images into 5 categories of diabetics and report them on asking for prediction. We have also devloped a messaging service for recieiving message for the type of diabetics.

8. TESTING:-

TEST CASES:-

USER ACCEPTANCE TESTING:-

1. Purpose of Document:-

This document serves as a quick reference for the Deep Learning Fundus Image Analysis for Early Detection of Diabetic Retinopathy project's test coverage and open issues as of the project's release for user acceptance testing.

2. Defect Analysis:-

This shows how many bugs were fixed or closed at each severity level and how they were fixed.

Resolution	Severity 1	Severity 2	Severity 3	Severity4	Subtotal
By Design	5	4	2	3	14
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	9	2	4	15	30
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won'tFix	0	5	2	1	8
Totals	17	14	13	21	65

3. Test-Case Analysis

This report shows the number of test cases that have passed, failed,and untested.

Section	TotalCases	Not Tested	Fail	Pass
PrintEngine	9	0	0	9
ClientApplication	45	0	0	45
Security	2	0	0	2
Out-sourceShipping	3	0	0	3
ExceptionReporting	9	0	0	9
FinalReportOutput	4	0	0	4
VersionControl	2	0	0	2

9.RESULTS:-

Performance Metrics:-

Model Performance Testing:

S. NO	Parameter	Values	Screenshot
1.	Model Summary	**Total params: 21,885,485** **Trainable params: 1,024,005** **Non-trainable params: 20,861,480**	
2.	Accuracy	Training Accuracy – **0.7917** Validation Accuracy – **loss 3.2610**	
3.	Confidence Score(Only Yolo Projects)	Class Detected - Confidence Score -	- - - - - - - -

Project team shall fill the following information in model performance testing template.

10. ADVANTAGES AND DISADVANTAGES:-

ADVANTAGES:-

There are several advantages of using deep learning for fundus image analysis for early detection of diabetic retinopathy.

First, deep learning is well-suited for image analysis tasks. This is because deep learning algorithms can automatically learn features from images, which is essential for accurate image analysis.

Second, deep learning is efficient at handling large amounts of data. This is important for medical image analysis, as medical images are often very large.

Third, deep learning is scalable. This means that it can be used to train models on very large datasets, which is important for medical image analysis tasks where data is often limited.

Fourth, deep learning is able to learn from data with little supervision. This is important for medical image analysis, as often there is limited labeled data available.

Finally, deep learning is robust. This means that it is less likely to overfit to the data, which is important for medical image analysis where data is often limited.

DISADVANTAGES:-

There are several disadvantages of deep learning for early detection of diabetic retinopathy. One disadvantage is that deep learning requires a large amount of data to train the models. This can be a challenge for researchers who do not have access to a large dataset. Another challenge is that deep learning models can be very complex, which can make them difficult to interpret. Finally, deep learning models can be computationally intensive, which can make them difficult to deploy in resource-limited settings.

11. CONCLUSION:-

Diabetic retinopathy (DR) is a leading cause of blindness in the United States. Early detection and treatment of DR is critical to preventing vision loss. However, DR is often asymptomatic in its early stages, making it difficult to detect.

Deep learning (DL) is a type of artificial intelligence that can be used to automatically detect patterns in data. DL has been shown to be effective for detecting DR in images of the retina.

In this study, a DL algorithm was used to automatically detect DR in fundus images. The algorithm was able to accurately detect DR in early stages, before it is symptomatic. This could potentially lead to earlier diagnosis and treatment of DR, which could help to prevent vision loss.

12. FUTURE SCOPE:-

There is a great potential for deep learning in fundus image analysis for early detection of diabetic retinopathy. However, there are a few challenges that need to be addressed. First, the current data sets are small and lack diversity. Second, the images are often low quality and need to be pre-processed before they can be used for deep learning. Third, the ground truth labels for the images are often not available. Finally, the current deep learning models are not able to generalize well to real-world data.

13. APPENDIX:-

app.py:-

```
import numpy as np
import os
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
from tensorflow.keras.applications.inception_v3 import preprocess_input
from flask import Flask, request,flash, render_template, redirect,url_for
from cloudant.client import Cloudant
from twilio.rest import Client
model = load_model(r"Updated-xception-diabetic-retinopathy.h5")
app = Flask(_name_)
app.secret_key="abc"
app.config['UPLOAD_FOLDER'] = "User_Images"
# Authenticate using an IAM API key

client = Cloudant.iam('08bcbaf0-260b-48e0-abdb-08db348afcf2-bluemix',
                      'yhZfUubpS3vS1vEKZSS37teD6IAUi8oLynOCQLIwnQsa', connect=True)
# Create a database using an initialized client
my_database = client.create_database('my_database')
if my_database.exists():
   print("Database '{0}' successfully created.".format('my_db'))
# default home page or route

user = ""

@app.route('/')
def index():
   return render_template('index.html', pred="Login", vis ="visible")

@ app.route('/index')
def home():
   return render_template("index.html", pred="Login", vis ="visible")


# registration page
@ app.route('/register',methods=["GET","POST"])
def register():
   if request.method == "POST":
      name = request.form.get("name")
      mail = request.form.get("emailid")
      mobile = request.form.get("num")
      pswd = request.form.get("pass")
      data = {
         'name': name,
         'mail': mail,
         'mobile': mobile,
```

```
            'psw': pswd
        }
        print(data)
        query = {'mail': {'$eq': data['mail']}}
        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 Successful , please login using your
details ")
        else:
            return render_template('register.html', pred=" You are already a member , please login using your
details ")
    else:
        return render_template('register.html')


@ app.route('/login', methods=['GET','POST'])
def login():
    if request.method == "GET":
        user = request.args.get('mail')
        passw = request.args.get('pass')
        print(user, passw)
        query = {'mail': {'$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="")
        else:
            if ((user == docs[0][0]['mail'] and passw == docs[0][0]['psw'])):
                flash("Logged in as " + str(user))
                return render_template('index.html', pred="Logged in as "+str(user), vis ="hidden",
vis2="visible")
            else:
                return render_template('login.html', pred="The password is wrong.")
    else:
        return render_template('login.html')


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

@app.route("/predict",methods=["GET", "POST"])
def predict():
    if request.method == "POST":
        f = request.files['file']
        # getting the current path 1.e where app.py is present
        basepath = os.path.dirname(_file_)
```

```
        #print ( " current path " , basepath )
        # from anywhere in the system we can give image but we want that
        filepath = os.path.join(str(basepath), 'User_Images', str(f.filename))
        #print ( " upload folder is " , filepath )
        f.save(filepath)
        img = image.load_img(filepath, target_size=(299, 299))
        x = image.img_to_array(img)  # ing to array
        x = np.expand_dims(x, axis=0)  # used for adding one more dimension
        #print ( x )
        img_data = preprocess_input(x)
        prediction = np.argmax(model.predict(img_data), axis=1)
        index = [' No Diabetic Retinopathy ', ' Mild NPDR ',
                ' Moderate NPDR ', ' Severe NPDR ', ' Proliferative DR ']
        result = str(index[prediction[0]])
        print(result)
        account_sid = 'AC8e0f2f5263d71c8f630a6486779cf08b'
        auth_token = '30b489873afb3c47340070eabd6bfb15'

        client = Client(account_sid, auth_token)

        ''' Change the value of 'from' with the number
        received from Twilio and the value of 'to'
        with the number in which you want to send message.'''
        message = client.messages.create(
                                from_='+16075363206',
                                body ='Results: '+ result,
                                to ='+919445979800'
                            )
        return render_template('prediction.html', prediction=result, fname = filepath)
    else:
        return render_template("prediction.html")

if__name__  == "__main__":
    app.debug = True
    app.run()
```

cloud.ipynb:-

```
from cloudant.client import Cloudant
client=Cloudant.iam('655489f8-18d0-4a44-a701-5de60570a973-
bluemix','Jc4eF6CXk72w0wGCsM_KUuXKVjsCcT4a54UKBXckK5Bv',connect=True)
my_database=client.create_database('my-database')
```

index.html:-

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <!-- CSS only -->
    <link
      href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css"
      rel="stylesheet"
      integrity="sha384-
iYQeCzEYFbKjA/T2uDLTpkwGzCiq6soy8tYaI1GyVh/UjpbCx/TYkiZhlZB6+fzT"
      crossorigin="anonymous"
    />
    <!-- JavaScript Bundle with Popper -->
    <script
      src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.js"
      integrity="sha384-
u1OknCvxWvY5kfmNBILK2hRnQC3Pr17a+RTT6rIHI7NnikvbZlHgTPOOmMi466C8"
      crossorigin="anonymous"
    ></script>
    <style>
       #navbarRight {
          margin-left: auto;
          padding-right:10px;
       }
       .navbar-brand{
          padding-left:15px;
       }
    </style>
    <title>DR Predcition</title>
  </head>
  <body>
    <nav class="navbar navbar-expand-lg navbar-light bg-dark">
       <div>
       <a class="navbar-brand" href="#" style="color:aliceblue">Diabetic Retinopathy Classification</a>
       </div>
       {{msg}}
       <div class="navbar-collapse collapse w-100 order-3 dual-collapse2" id="navbarNav">
         <ul class="navbar-nav mr-auto text-center" id="navbarRight">
           <li class="nav-item active">
             <a class="nav-link" href="index" style="color: aliceblue;">Home </a>
           </li>
           <li class="nav-item" style="visibility:{{ vis2 }}">
             <a class="nav-link" href="predict" style="color: aliceblue;">Prediction</a>
           </li>
           <li class="nav-item">
             <a class="nav-link" href="login" style="color: aliceblue;">{{pred}}</a>
```

```
          </li>
          <li class="nav-item" style="visibility:{{ vis }}">
            <a class="nav-link" href="register" style="color: aliceblue;">Register</a>
          </li>
        </ul>
      </div>
    </nav>
    <br><br>
      <div class="jumbotron container">
        <h1 class="display-4">Diabetic Retinopathy</h1>
        <p class="lead">Diabetic retinopathy is a diabetes complication that affects eyes. It's caused by
damage to the blood vessels of the light-sensitive tissue at the back of the eye (retina).
          At first, diabetic retinopathy might cause no symptoms or only mild vision problems. But it can
lead to blindness.
          The condition can develop in anyone who has type 1 or type 2 diabetes. The longer you have
diabetes and the less controlled your blood sugar is, the more likely you are to develop this eye
complication.</p>
        <hr class="my-4">
        <div class="d-flex justify-content-center">
          <img style="width:70vw;" src="static/diabetic-retinopathy-home.jpg">
          </div>
      </div>
  </body>
</html>
```

login.html:-

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <!-- CSS only -->
    <link
      href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css"
      rel="stylesheet"
      integrity="sha384-
iYQeCzEYFbKjA/T2uDLTpkwGzCiq6soy8tYaI1GyVh/UjpbCx/TYkiZhlZB6+fzT"
      crossorigin="anonymous"
    />
    <!-- JavaScript Bundle with Popper -->
    <script
      src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.js"
      integrity="sha384-
u1OknCvxWvY5kfmNBILK2hRnQC3Pr17a+RTT6rIHI7NnikvbZlHgTPOOmMi466C8"
      crossorigin="anonymous"
    ></script>
    <style>
```

```
      #navbarRight {
         margin-left: auto;
         padding-right:10px;

      }
      .navbar-brand{
         padding-left:15px;
      }
   </style>
   <title>DR Predcition</title>
  </head>
  <form action="",method='POST'>
   <nav class="navbar navbar-expand-lg navbar-light bg-dark">
      <div>
      <a class="navbar-brand" href="#" style="color:aliceblue">User Login</a>
      </div>
      <div class="navbar-collapse collapse w-100 order-3 dual-collapse2" id="navbarNav">
       <ul class="navbar-nav mr-auto text-center" id="navbarRight">
         <li class="nav-item active">
           <a class="nav-link" href="index" style="color: aliceblue;">Home </a>
         </li>
         <li class="nav-item">
           <a class="nav-link" href="login" style="color: aliceblue;">Login</a>
         </li>
         <li class="nav-item">
           <a class="nav-link" href="register"style="color: aliceblue;">Register</a>
         </li>
       </ul>
      </div>
    </nav>
    <br><br>
    <form class="form-inline" action="/login" method="GET">
    <div class="container" style="width: 600px; height: 600px;">
     <div class="mb-3 d-flex justify-content-center"><script
src="https://cdn.lordicon.com/xdjxvujz.js"></script>
         <lord-icon
            src="https://cdn.lordicon.com/elkhjhci.json"
            trigger="hover"
            style="width:200px;height:200px">
         </lord-icon></div>
         <div class="mb-3">
            <input type="email" class="form-control" id="exampleInputEmail1" name="mail" aria-
describedby="emailHelp" placeholder="Enter Registered Mail ID">
          </div>
          <div class="mb-3">
            <input type="password" class="form-control" id="exampleInputPassword1" name="pass"
placeholder="Enter Password">
          </div>
          <div class="mb-3">
          <button type="submit form-control" class="btn btn-dark btn-primary" style="width:100%;"
type="submit">Login</button>
```

```
          </div>
          {{pred}}
      </div>
    </form>
  </body>
</html>
```

logout.html:-

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <!-- CSS only -->
    <link
      href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css"
      rel="stylesheet"
      integrity="sha384-
iYQeCzEYFbKjA/T2uDLTpkwGzCiq6soy8tYaI1GyVh/UjpbCx/TYkiZhlZB6+fzT"
      crossorigin="anonymous"
    />
    <!-- JavaScript Bundle with Popper -->
    <script
      src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.js"
      integrity="sha384-
u1OknCvxWvY5kfmNBILK2hRnQC3Pr17a+RTT6rIHI7NnikvbZlHgTPOOmMi466C8"
      crossorigin="anonymous"
    ></script>
    <style>
       #navbarRight {
          margin-left: auto;
          padding-right:10px;
       }
       .navbar-brand{
          padding-left:15px;
       }
    </style>
    <title>DR Predcition</title>
  </head>
  <body>
    <nav class="navbar navbar-expand-lg navbar-light bg-dark">
       <div>
       <a class="navbar-brand" href="#" style="color:aliceblue">Diabetic Retinopathy</a>
       </div>
       <div class="navbar-collapse collapse w-100 order-3 dual-collapse2" id="navbarNav">
         <ul class="navbar-nav mr-auto text-center" id="navbarRight">
           <li class="nav-item active">
             <a class="nav-link" href="index" style="color: aliceblue;">Home </a>
```

```
          </li>
          <li class="nav-item">
            <a class="nav-link" href="login" style="color: aliceblue;">Login</a>
          </li>
          <li class="nav-item">
            <a class="nav-link" href="register"style="color: aliceblue;">Register</a>
          </li>
        </ul>
      </div>
    </nav>
    <br><br>
    <div class="d-flex justify-content-center">
      <div class="row d-flex display-3 justify-content-center">
        Successfully Logged Out!
        <br><br>
        <a href="login" class="btn btn-lg btn-dark">Login for more Information</a>
          </div>
      </div>
  </body>
</html>
```

prediction.html:-

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8" />
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <!-- CSS only -->
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css" rel="stylesheet"
    integrity="sha384-
iYQeCzEYFbKjA/T2uDLTpkwGzCiq6soy8tYaI1GyVh/UjpbCx/TYkiZhlZB6+fzT"
crossorigin="anonymous" />
  <!-- JavaScript Bundle with Popper -->
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.js"
    integrity="sha384-
u1OknCvxWvY5kfmNBILK2hRnQC3Pr17a+RTT6rIHI7NnikvbZlHgTPOOmMi466C8"
    crossorigin="anonymous"></script>
  <style>
    #navbarRight {
    margin-left: auto;
      padding-right: 10px;
    }

    .navbar-brand {
      padding-left: 15px;
    }
```

```
    .row {
      width: 90%;
    }
  </style>
  <title>DR Predcition</title>
</head>

<body>
  <nav class="navbar navbar-expand-lg navbar-light bg-dark">
    <div>
      <a class="navbar-brand" href="#" style="color:aliceblue">Diabetic Retinopathy Classification</a>
    </div>
    <div class="navbar-collapse collapse w-100 order-3 dual-collapse2" id="navbarNav">
      <ul class="navbar-nav mr-auto text-center" id="navbarRight">
        <li class="nav-item active">
          <a class="nav-link" href="index" style="color: aliceblue;">Home </a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="logout" style="color: aliceblue;">Logout</a>
        </li>
      </ul>
    </div>
  </nav>
  <br><br>
  <div class="container justify-content-center" style="width:700px">
    <form action = "/predict" method = "POST" enctype="multipart/form-data">
    <label for="formFileLg" class="form-label">Upload Image</label>
    <input class="form-control form-control-lg" name ="file" type="file" />
    <br>
    <button class="btn btn-lg btn-dark" type = "submit">Predict</button>
    </form>
    <br>
    <h1>{{prediction}}</h1>
  </div>
  <br><br><br>
  <div class="d-flex justify-content-center" >
      <img src="static/level.png" style="width: 90%">
  </div>
</body>
</html>
```

register.html:-

```
<!-- <!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
```

```
    <!-- CSS only -->
    <link
      href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css"
      rel="stylesheet"
      integrity="sha384-
iYQeCzEYFbKjA/T2uDLTpkwGzCiq6soy8tYaI1GyVh/UjpbCx/TYkiZhlZB6+fzT"
      crossorigin="anonymous"
    />
    <!-- JavaScript Bundle with Popper -->
    <script
      src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.js"
      integrity="sha384-
u1OknCvxWvY5kfmNBILK2hRnQC3Pr17a+RTT6rIHI7NnikvbZlHgTPOOmMi466C8"
      crossorigin="anonymous"
    ></script>
    <style>
       #navbarRight {
          margin-left: auto;
          padding-right:10px;

       }
       .navbar-brand{
          padding-left:15px;
       }
    </style>
    <title>DR Predcition</title>
  </head>
  <form action="{{url_for('register')}}" method="post" >
    <nav class="navbar navbar-expand-lg navbar-light bg-dark">
       <div>
       <a class="navbar-brand" href="#" style="color:aliceblue">Registration</a>
       </div>
       <div class="navbar-collapse collapse w-100 order-3 dual-collapse2" id="navbarNav">
         <ul class="navbar-nav mr-auto text-center" id="navbarRight">
           <li class="nav-item active">
             <a class="nav-link" href="index" style="color: aliceblue;">Home </a>
           </li>
           <li class="nav-item">
             <a class="nav-link" href="login" style="color: aliceblue;">Login</a>
           </li>
           <li class="nav-item">
             <a class="nav-link" href="register"style="color: aliceblue;">Register</a>
           </li>
         </ul>
       </div>
     </nav>
     <br><br>
     <form class="form-inline" method ="POST">
     <div class="container" style="width: 600px; height: 600px;">
       <div class="mb-3 d-flex justify-content-center"><script
src="https://cdn.lordicon.com/xdjxvujz.js"></script>
```

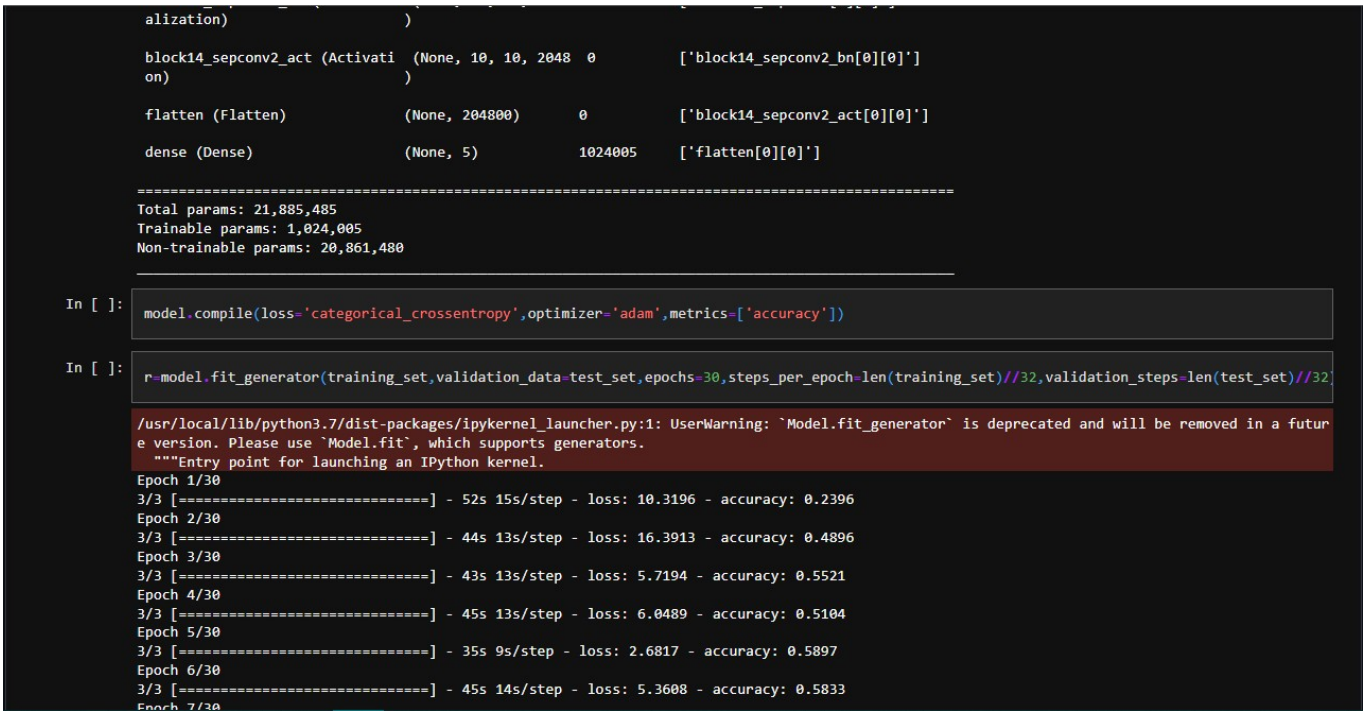
```
        <lord-icon
           src="https://cdn.lordicon.com/elkhjhci.json"
           trigger="hover"
           style="width:200px;height:200px">
        </lord-icon></div>
                            <div class="mb-3">
           <input type="text" class="form-control" id="exampleInputName" name = "name" aria-
describedby="nameHelp" placeholder="Enter Name">
          </div>
          <div class="mb-3">
           <input type="email" class="form-control" id="exampleInputEmail1" name="emailid" aria-
describedby="emailHelp" placeholder="Enter Mail ID">
          </div>
          <div class="mb-3">
           <input type="number" class="form-control" id="exampleInputNumber1" name="num" aria-
describedby="numberHelp" placeholder="Enter Mobile number">
          </div>
          <div class="mb-3">
           <input type="password" class="form-control" id="exampleInputPassword1" name="pass"
placeholder="Enter Password">
          </div>
          <div class="mb-3">
          <button type="submit form-control" class="btn btn-dark btn-primary"
style="width:100%;">Register</button>
        </div>
                            <div class="mb-3 d-flex justify-content-center">
                            <a href="login" class="nav-link"> Already Registered: Login Here</a>
    </div>
    {{pred}}
    </div>
   </form>
  </body>
</html> -->
```

Python Notebook screenshots:-

```
In [ ]: pip install -q kaggle

In [ ]: mkdir ~/.kaggle

mkdir: cannot create directory ‘/root/.kaggle’: File exists

In [ ]: cp kaggle.json ~/.kaggle/

In [ ]: !chmod 600 ~/.kaggle/kaggle.json

In [ ]: !kaggle datasets download -d arbethi/diabetic-retinopathy-level-detection

Downloading diabetic-retinopathy-level-detection.zip to /content
100% 9.65G/9.66G [01:17<00:00, 186MB/s]
100% 9.66G/9.66G [01:17<00:00, 133MB/s]

In [ ]: !unzip diabetic-retinopathy-level-detection.zip

Archive:  diabetic-retinopathy-level-detection.zip
  inflating: inception-diabetic.h5
  inflating: preprocessed dataset/preprocessed dataset/testing/0/cfb17a7cc8d4.png
  inflating: preprocessed dataset/preprocessed dataset/testing/0/cfdbaef73a8b.png
  inflating: preprocessed dataset/preprocessed dataset/testing/0/cfed7c1172ec.png
  inflating: preprocessed dataset/preprocessed dataset/testing/0/cff262ed8f4c.png
  inflating: preprocessed dataset/preprocessed dataset/testing/0/cffc50047828.png
  inflating: preprocessed dataset/preprocessed dataset/testing/0/d02b79fc3200.png
  inflating: preprocessed dataset/preprocessed dataset/testing/0/d0926ed2c8e5.png
  inflating: preprocessed dataset/preprocessed dataset/testing/0/d160ebef4117.png
  inflating: preprocessed dataset/preprocessed dataset/testing/0/d16e39b9d6f0.png
  inflating: preprocessed dataset/preprocessed dataset/training/4/ebe0175e530c.png
  inflating: preprocessed dataset/preprocessed dataset/training/4/ed246ae1ed08.png
  inflating: preprocessed dataset/preprocessed dataset/training/4/ed3a0fc5b546.png
  inflating: preprocessed dataset/preprocessed dataset/training/4/ee1ec90b980f.png
  inflating: preprocessed dataset/preprocessed dataset/training/4/ef26625121b3.png
  inflating: preprocessed dataset/preprocessed dataset/training/4/f0098e9d4aee.png
  inflating: preprocessed dataset/preprocessed dataset/training/4/f025f33b2c9b.png
  inflating: preprocessed dataset/preprocessed dataset/training/4/f03d3c4ce7fb.png
  inflating: preprocessed dataset/preprocessed dataset/training/4/f0f89314e860.png
  inflating: preprocessed dataset/preprocessed dataset/training/4/f1dc26c4bfa3.png
  inflating: preprocessed dataset/preprocessed dataset/training/4/f2d2a0c92034.png
  inflating: preprocessed dataset/preprocessed dataset/training/4/f549294e12e1.png
  inflating: preprocessed dataset/preprocessed dataset/training/4/f58d37d48e42.png
  inflating: preprocessed dataset/preprocessed dataset/training/4/f5e6226bd2e0.png
  inflating: preprocessed dataset/preprocessed dataset/training/4/f69835dc7c50.png
  inflating: preprocessed dataset/preprocessed dataset/training/4/f6f3ea0d2693.png
  inflating: preprocessed dataset/preprocessed dataset/training/4/f72adcac5638.png
  inflating: preprocessed dataset/preprocessed dataset/training/4/f850cb51fdba.png
  inflating: preprocessed dataset/preprocessed dataset/training/4/f8cf7ed8ef00.png
  inflating: preprocessed dataset/preprocessed dataset/training/4/fa59221cf464.png
  inflating: preprocessed dataset/preprocessed dataset/training/4/fb696a8e055a.png
  inflating: preprocessed dataset/preprocessed dataset/training/4/fce93caa4758.png
  inflating: preprocessed dataset/preprocessed dataset/training/4/fdd534271f3d.png
  inflating: preprocessed dataset/preprocessed dataset/training/4/ff8a0b45c789.png

In [ ]: from tensorflow.keras.layers import Dense,Flatten,Input

In [ ]: from tensorflow.keras.models import Model

In [ ]: from tensorflow.keras.preprocessing import image

In [ ]: from tensorflow.keras.preprocessing.image import ImageDataGenerator,load_img
```

```
In [ ]: from glob import glob

In [ ]: import numpy as np

In [ ]: import matplotlib.pyplot as plt

In [ ]: imageSize=[299,299]

In [ ]: trainPath=r"/content/preprocessed dataset/preprocessed dataset/training"

In [ ]: testPath=r"/content/preprocessed dataset/preproccessed dataset/testing"

In [ ]: train_datagen=ImageDataGenerator(rescale=1./255,shear_range=0.2,zoom_range=0.2,horizontal_flip=True)

In [ ]: test_datagen=ImageDataGenerator(rescale=1./255)

In [ ]: training_set=train_datagen.flow_from_directory('/content/preprocessed dataset/preprocessed dataset/training',target_size=(299,299),batch_size=

Found 3662 images belonging to 5 classes.

In [ ]: test_set=test_datagen.flow_from_directory('/content/preprocessed dataset/preprocessed dataset/testing',target_size=(299,299),batch_size=32,clas

Found 3662 images belonging to 5 classes.

In [ ]: test_set=test_datagen.flow_from_directory('/content/preprocessed dataset/preprocessed dataset/testing',target_size=(299,299),batch_size=32,clas

Found 734 images belonging to 5 classes.

In [ ]: xception=Xception(input_shape=imageSize+[3],weights='imagenet',include_top=False)

Downloading data from https://storage.googleapis.com/tensorflow/keras-applications/xception/xception_weights_tf_dim_ordering_tf_kernels_notop.h
5
83683744/83683744 [==============================] - 0s 0us/step

In [ ]: for layer in xception.layers:
          layer.trainable=False

In [ ]: x=Flatten()(xception.output)

In [ ]: prediction=Dense(5,activation='softmax')(x)

In [ ]: model=Model(inputs=xception.input,outputs=prediction)

In [ ]: model.summary()

Model: "model"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   Connected to
=================================================================
 input_1 (InputLayer)        [(None, 299, 299, 3       0         []
                             )]
```

```
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]']
                             )

 block2_sepconv1 (SeparableConv  (None, 147, 147, 12   8768      ['block1_conv2_act[0][0]']
 2D)                         8)

 block2_sepconv1_bn (BatchNorma  (None, 147, 147, 12   512       ['block2_sepconv1[0][0]']
 lization)                   8)

 block2_sepconv2_act (Activatio  (None, 147, 147, 12   0         ['block2_sepconv1_bn[0][0]']
 n)                          8)

 block2_sepconv2 (SeparableConv  (None, 147, 147, 12   17536     ['block2_sepconv2_act[0][0]']
 2D)                         8)

Epoch 21/30
3/3 [==============================] - 43s 13s/step - loss: 3.4297 - accuracy: 0.6771
Epoch 22/30
3/3 [==============================] - 43s 13s/step - loss: 5.0327 - accuracy: 0.6979
Epoch 23/30
3/3 [==============================] - 37s 14s/step - loss: 5.6452 - accuracy: 0.6026
Epoch 24/30
3/3 [==============================] - 44s 14s/step - loss: 5.8190 - accuracy: 0.6562
Epoch 25/30
3/3 [==============================] - 43s 13s/step - loss: 3.5427 - accuracy: 0.6979
Epoch 26/30
3/3 [==============================] - 43s 13s/step - loss: 3.7831 - accuracy: 0.7083
Epoch 27/30
3/3 [==============================] - 50s 16s/step - loss: 3.7079 - accuracy: 0.6250
Epoch 28/30
3/3 [==============================] - 42s 13s/step - loss: 2.3158 - accuracy: 0.7292
Epoch 29/30
3/3 [==============================] - 46s 13s/step - loss: 5.2872 - accuracy: 0.6979
Epoch 30/30
3/3 [==============================] - 43s 13s/step - loss: 3.2610 - accuracy: 0.7917

In [ ]: model.save('Updated-Xception-diabetic-retinopathy.h5')
```



GITHUB LINK:-

https://github.com/IBM-EPBL/IBM-Project-39696-1660489932.git

DEMO LINK:-

https://github.com/IBM-EPBL/IBM-Project-39696-1660489932.git