PROJECT REPORT

Date	18 November 2022
Team ID	PNT2022TMID53189
Project Name	Project - Deep Learning Fundus Image Analysis For Early Detection Of Diabetic Retinopathy
Project Members	Dharsini,Kayanat,Karthika,Kavya
Project Mentors	Industry Mentors-Shivam Shivare, Indra Prakash Faculty Mentors- Dr. Malathy

INDEX

1.INTRODUCTION

1.1Project Overview

Diabetic Retinopathy (DR) 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. Unfortunately, DR 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.

Transfer learning has become one of the most common techniques that has achieved better performance in many areas, especially in medical image analysis and classification. We used Transfer Learning techniques like Inception V3,Resnet50,Xception V3 that are more widely used as a transfer learning method in medical image analysis and they are highly effective.

1.2Purpose

The project has the following purpose:

- It provides the patient a way to detect Diabetic Retinopathy as early as possible because the treatment can reduce the risk of vision loss.
- It provides a way to automate the diagnosis process because the time, effort and cost is significantly reduced.
- It provides a way to have a count on the number of patients having Diabetic Retinopathy because they consider them for further evaluations.

2.LITERATURE SURVEY

2.1Existing problem

Diabetic Retinopathy (DR) 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. Unfortunately, DR 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.

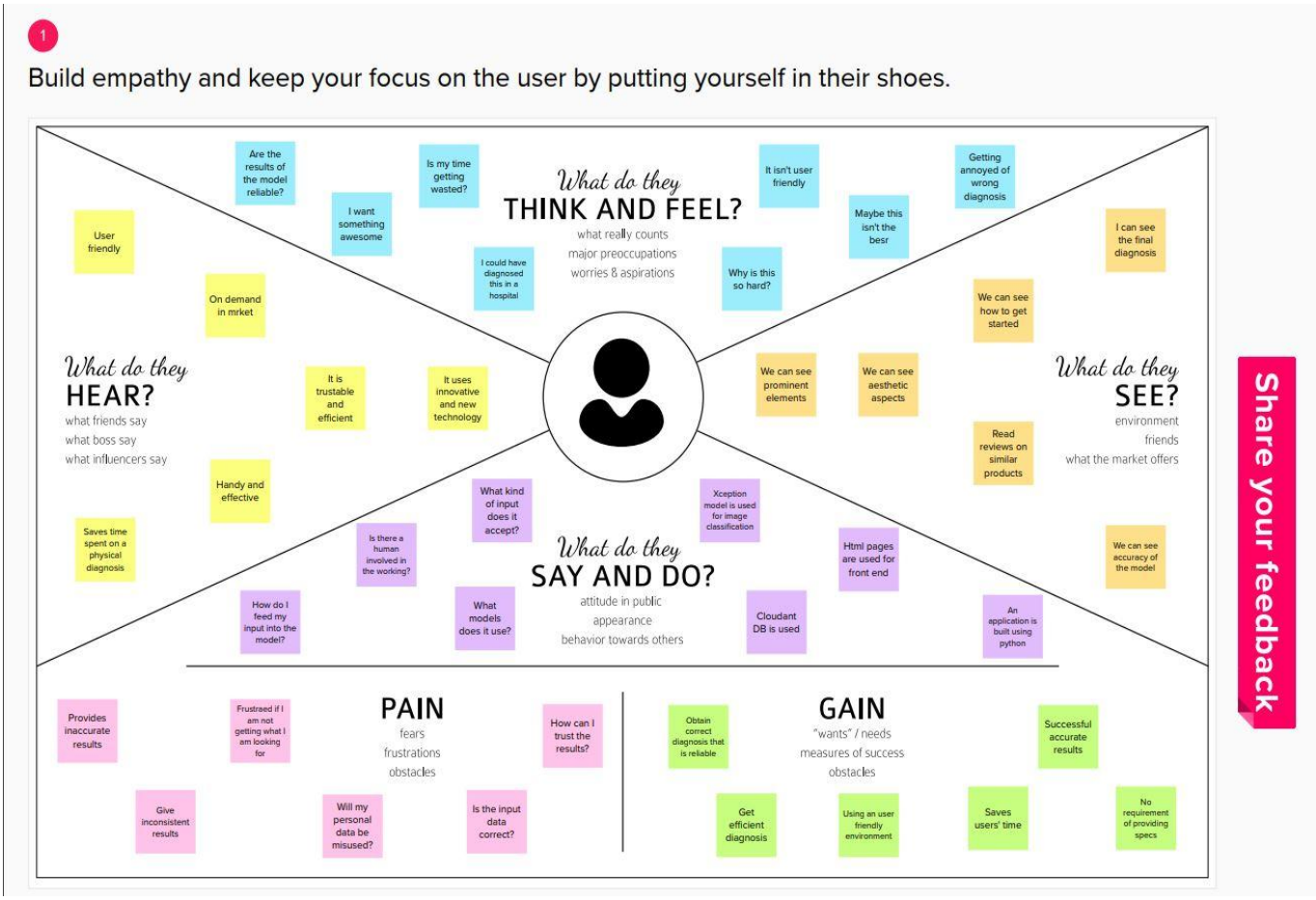
2.2Problem Statement Definition

1. A patient needs a way to detect Diabetic Retinopathy as early as possible because the treatment can reduce the risk of vision loss.
2. An ophthalmologist needs a way to automate the diagnosis process because the time, effort and cost is significantly reduced.
3. A hospital management needs a way to have a count on the number of patients having Diabetic Retinopathy because they consider them for further evaluations.

3. **IDEATION & PROPOSED SOLUTION**

3.1 Empathy Map Canvas

To prepare Empathy Map Canvas to capture the user Pains & Gains, Prepare a list of problem statement.



3.2 Ideation & Brainstorming

To list the ideas by organizing the brainstorming session and prioritize the top 3 ideas based on the feasibility & importance.

Template

Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

- **10 minutes** to prepare
- **1 hour** to collaborate
- **2-8 people** recommended

Share template feedback

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

Team gathering
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

Set the goal
Think about the problem you'll be focusing on solving in the brainstorming session.

Learn how to use the facilitation tools
Use the Facilitation Superpowers to run a happy and productive session.

Open article →

1 Define your problem statement

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

5 minutes

Need some inspiration?

See a finished version of this template to kickstart your work.

Open example →

2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

TIP

You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing!

Dhanish P

Kavya S

Karthika B

Gayanat

3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. In the last 10 minutes, 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

ARTIFICIAL INTELLIGENCE

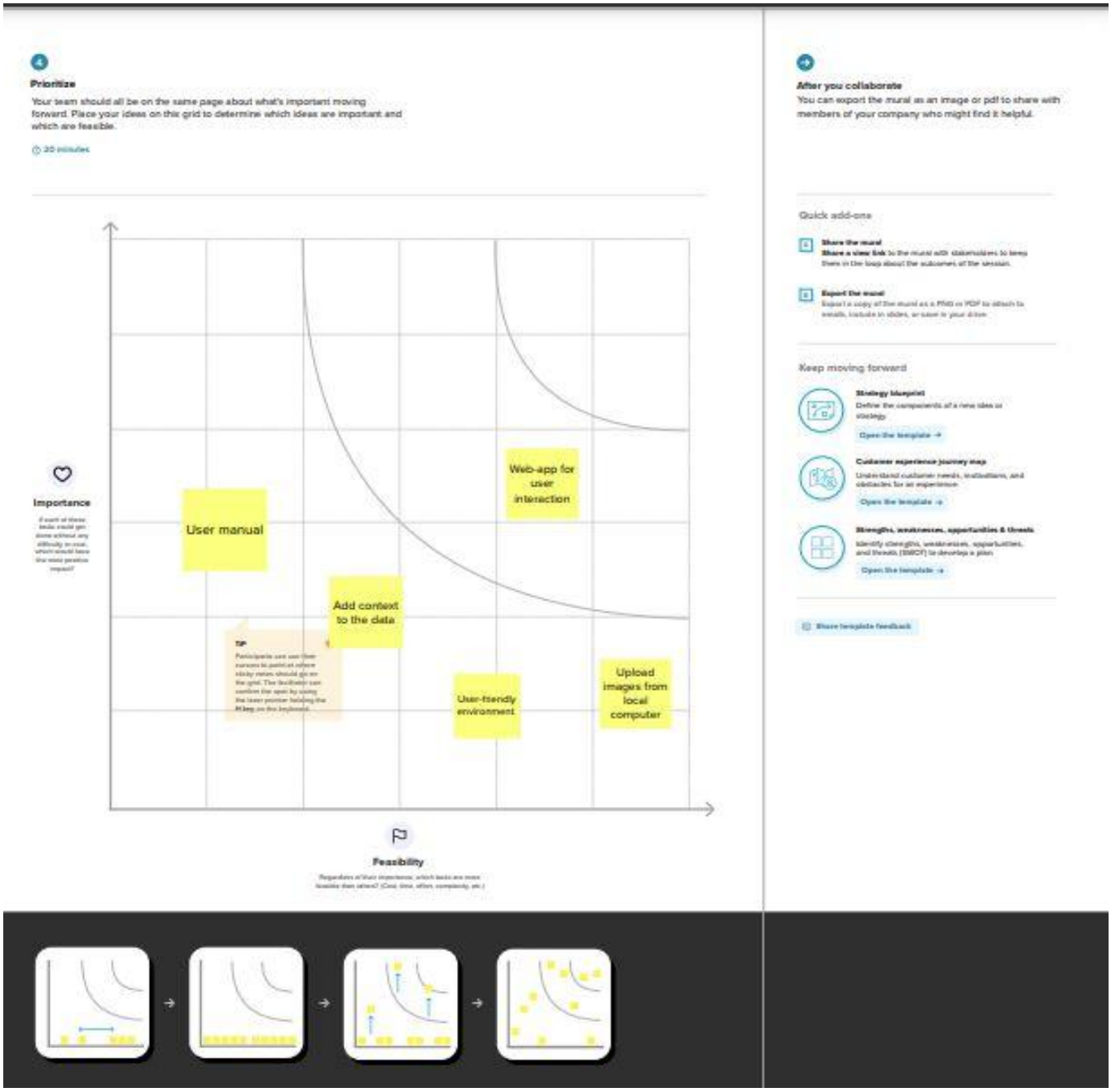
INTELLIGENT DESIGN

RULE BASED SYSTEMS

DOCUMENTATION

TIP

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



3.3 Proposed Solution

To prepare the proposed solution document, which includes the novelty, feasibility of idea, business model, social impact, scalability of solution, etc.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Diabetic Retinopathy (DR) 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. Unfortunately, DR is not a reversible process, and treatment only sustains vision. DR early detection and treatment can significantly reduce the risk of vision loss. An ophthalmologist needs a way to automate the diagnosis process because the time, effort and cost is significantly reduced. A patient needs a way to detect Diabetic Retinopathy as early as possible because the treatment can reduce the risk of vision loss. A hospital management needs a way to have a count on the number of patients having Diabetic Retinopathy because they consider them for further evaluations.
2.	Idea / Solution description	In this project, we intend to build a Deep Learning Fundus Image Analysis For Early Detection Of Diabetic Retinopathy using a convolutional neural network (CNN).We plan on creating a web application where the user interacts with the UI (User Interface) to choose the image. The chosen image is analysed by the model which is integrated with flask application. The Xception Model analyses the image, then the prediction is showcased on the Flask UI.

3.	Novelty / Uniqueness	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. Many models were developed but extracting the input and predicting the accurate output is till now not that accurate. Our model focuses particularly on this aspect where we not only detect Diabetic Retinopathy but also try to do so with higher accuracy to avoid any false positives to a greater extent that may lead to dangerous results in treatment. This is done by passing the input to the trained model which will compare the fundus image with the training sample and calculate the probability of detection of Diabetic Retinopathy and it will display the results. This model is tested with a large number of samples and it predicts accurately compared to all the previous models that have even been built till now.

4.	Social Impact / Customer Satisfaction	This deep learning model provides the healthcare industry with the ability to analyse data at exceptional speeds without compromising on accuracy. One of the biggest advantages of using this deep learning approach is its ability to execute feature engineering by itself. In this approach, an algorithm scans the data to identify features which correlate and then combine them to promote faster learning without being told to do so explicitly. It provides useful and precise information and guidance for Diabetic Retinopathy prediction, clinical diagnosis, and medical services.
5.	Business Model (Revenue Model)	We intend this project to be a not-for-profit , one that is not driven by profit but by dedication to a given cause. To recover server and hosting charges we intend on monetizing the website using Google AdSense to yield a monthly revenue.
6.	Scalability of the Solution	Initially, this project focuses on a small number of users. Once there is an increase in the number of the users, larger number of samples from users can also act as the input to the training model. Cloud can also be made use for large storage and better performance. With google ads we can add user subscription for a year/month which can generate revenue for the project maintenance.

3.4 Problem Solution fit

To prepare a problem - solution fit document.

Define CS fit, intro CL	CUSTOMER SEGMENT(S) CS Patient come under the category of individual users. A group of medical professionals come under the category of business users	CUSTOMER LIMITATIONS CL Patients and the ophthalmologist can use the application using their smartphones, laptops, and iPads as well, because our application is a web application that can be used on any device. The application must be device-friendly.	AVAILABLE SOLUTIONS AS In this project, we intend to build a Deep Learning Fundus Image Analysis For Early Detection Of Diabetic Retinopathy using a convolutional neural network (CNN).We plan on creating a web application where the user interacts with the UI (User Interface) to choose the image. The chosen image is analysed by the model which is integrated with flask application. The Xception Model analyses the image, then the prediction is showcased on the Flask UI.
Focus on PR, tap into BE, understand RC	PROBLEMS/PAINS PR A **patient** needs a way to detect Diabetic Retinopathy as early as possible because the treatment can reduce the risk of vision loss. An **ophthalmologist** needs a way to automate the diagnosis process because the time, effort and cost is significantly reduced. A **hospital** management needs a way to have a count on the number of patients having Diabetic Retinopathy because they consider them for further evaluations.	**PROBLEM ROOT/CAUSE** Users are reluctant to do the tedious and trivial calculations Diabetic Retinopathy (DR) is a common complication of diabetes mellitus, which causes lesions on the retina that affect vision. Unfortunately, DR is not a reversible process, and treatment only sustains vision. 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.	BEHAVIOR BE Users have the option of uploading photographs from their local computer or their drive. The outcome will be presented using the Xception learning model along with a graphical depiction of the diagnostic. Thus the user can determine the severity of the illness.
Identify strong TR & EM	TRIGGERS TO ACT TR This programme allows users to obtain findings when they have symptoms that are connected to a particular illness.	YOUR SOLUTION S In this project, we are building a web application that allows users to upload photographs. The image is analysed by the Xception Model, and the prediction is subsequently shown on the display.	CHANNELS of BEHAVIOR CH Offline This application is not available in offline mode.
	EMOTIONS EM Before: User is not aware of that the diagnosis may lead to blindness. The early stage of identification is very important to cure the blindness. After: The user can get an idea of the severity of the symptoms of the disease and take precautions at an early stage to avoid blindness.		Online The application will be marketed through the usage of various social media platforms. As users begin to use the application, ratings in Google, resulting in a huge influx of customers.

4.REQUIREMENT ANALYSIS

4.1Functional requirement

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User Login	Separate login for doctor and patients along with username and password
FR-3	Contact details	The contact details of the health care specialists are displayed.
FR-4	Input	User will be able to upload the image from there personal system into the website
FR-5	Output	The accuracy of the situation and the relevant information is displayed according to the obtained result from the prediction
FR-6	Training	The model should be able to be trained with new datasets to increase the accuracy.
FR-7	Image processing accuracy	The prediction accuracy should be correct and there should not be any discrepancy.
FR-9	Feedback input	The feedback from the user of the system is required to make the system more efficient.

Non-Functional requirements

Functional Requirements:

FR No.	Non-Functional Requirement	Description

NFR-1	**Usability**	The website must built with simple English vocabulary so that the users can understand The input dialog box should mention the type of image and maximum size of the image which is permitted to be uploaded
NFR-2	**Security**	Only the admin should have the permission to access the whole system and have the privilege to update the model with datasets.

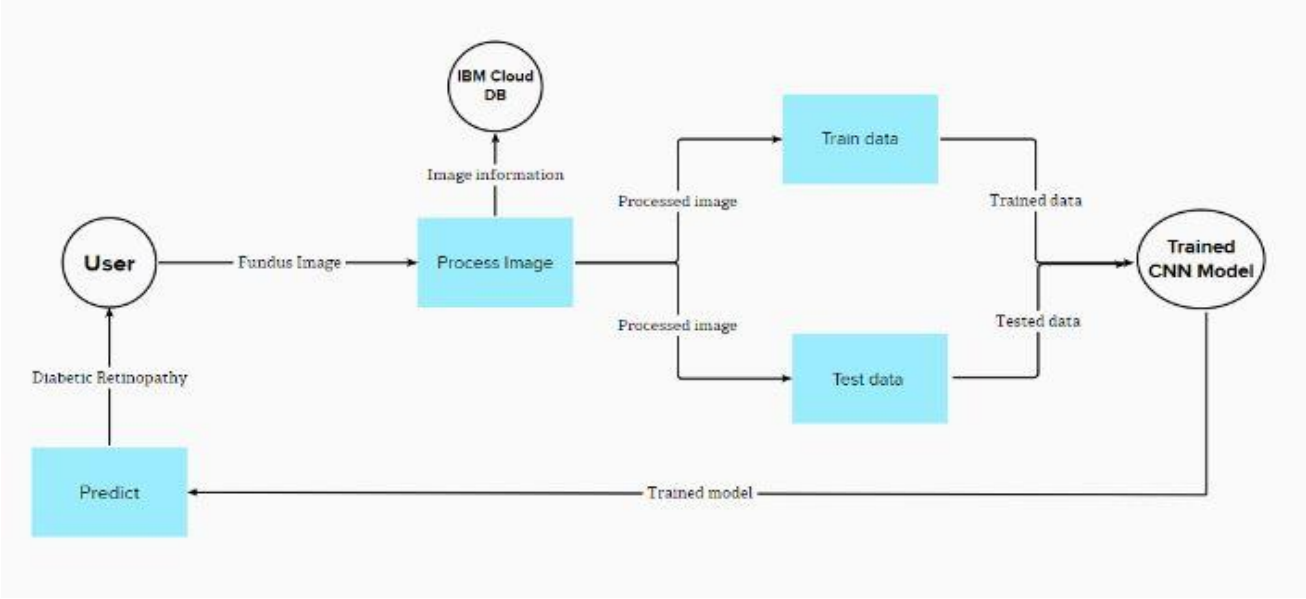
NFR-3	**Reliability**	All the data should be securely stored in the cloud for backup Should be available even during update or rollback phases.
NFR-4	**Performance**	The loading time for each page should be less than 2 sec to provide the user a better experience The prediction time should be less and the be able to achieve better accuracy.
NFR-5	**Availability**	New module deployment mustn’t impact website pages availability and mustn't take longer than one hour to be live. The pages that may experience problems must display a notification with a timer showing when the system is going to be up again.
NFR-6	**Scalability**	The database size should be able to be increased without affecting the existing records. The website should be able to handle up to 5000 users at a time

5. PROJECT DESIGN

5.1 Data Flow Diagrams

To prepare the Data Flow Diagrams.

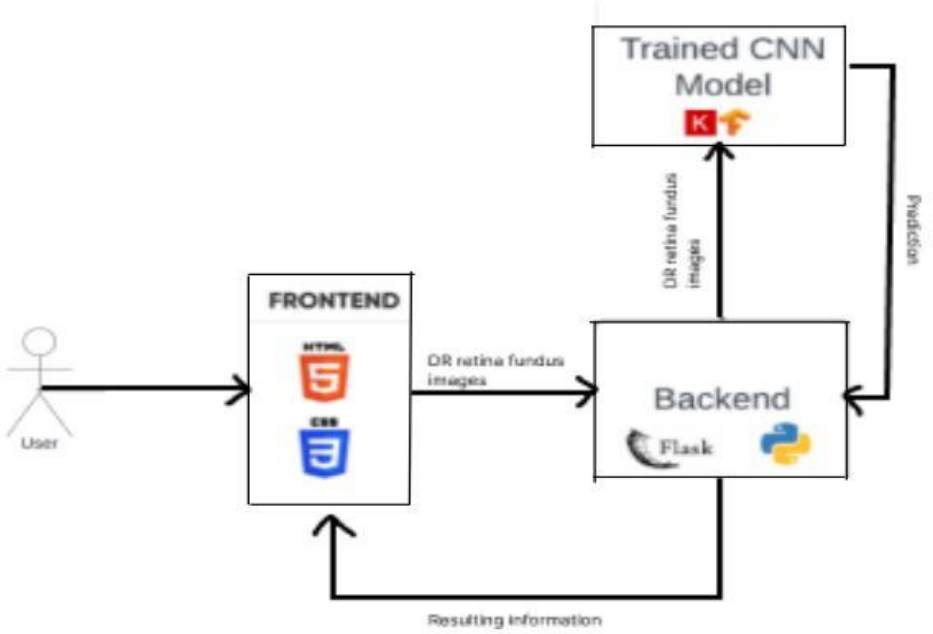
A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



5.2 Solution & Technical Architecture

To prepare problem - Solution and Technical Architecture.

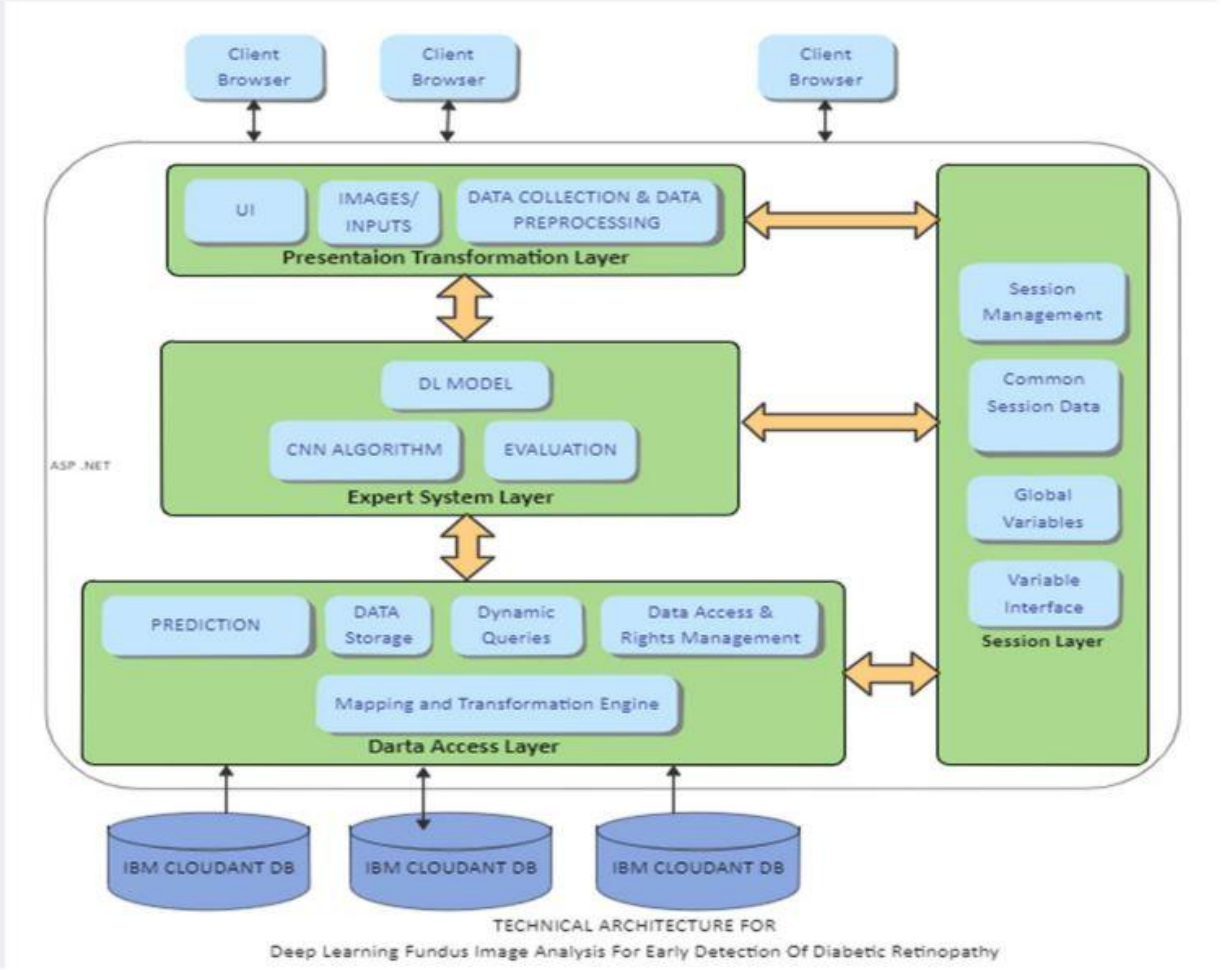
Solution Architecture



Technologies needed for Minimum Viable Product deployment
Upon research it was found that we need require the following software technologies for the systematic development and deployment of the project:

- HTML/CSS/JavaScript/Bootstrap – Front End Development
- Python
- TensorFlow
- Image Processing Basics
- Flask – Backend Development
- Git & GitHub – Project Management
- IBM Cloud – Hosting
- IBM Watson – Training the Deep Learning Mode

Technology Architecture



The Technical Architecture has the following blocks:

- Data Collection.
  - Create a Train and Test path.

- Data Pre-processing.

  - Import the required library
  - Configure ImageDataGenerator class
  - Apply ImageDataGenerator functionality to Train Set and Testset
- Model Building

- Pre-trained CNN model as a Feature Extractor

- Adding Dense Layer

- Configure the Learning Process

- Train the model

- Save the Model
- Test the model
    - Cloudant DB
- Register & Login to IBM Cloud
- Create Service Instance
- Creating Service Credentials
- Launch Cloudant DB
- Create Database
    - Application Building
- Create an HTML file
- Build Python Code

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	The user interacts with the application through Web UI.	HTML, CSS, JavaScript, Flask, Python
2.	Application Logic-Creating an account	The user interacts with the Web UI to create an account. The account details are stored in a secured manner in IBM CLOUDANT DB	Flask, IBM CLOUDANT DB
3.	Application Logic-Logging in	The user logs into the application through the Web UI. The user details are verified by cross-checking with data available in IBM CLOUDANT DB	Flask, IBM CLOUDANT DB
4.	Application Logic-Getting input from the user	The user interacts with the Web UI to get input from the user.	Flask, IBM CLOUDANT DB

5.	Application Logic-Data pre-processing	The input data collected from the user is pre-processed..	Flask, IBM CLOUDANT DB
6.	Application Logic-Run the model on the data	The input data collected from the user is sent to the model and prediction is made.	Flask, IBM CLOUDANT DB
7.	Application Logic-Storage of Data	The result is stored in the IBM CLOUDANT DB .	Flask, IBM CLOUDANT DB
8.	Application Logic-Display the result	The result is retrieved from the IBM CLOUDANT DB and displayed to the user through the Web UI.	Flask, IBM CLOUDANT DB
9.	Database	The data types will be user dependent as the application is made to be customizable and is also a cloud based database.	IBM CLOUDANT DB
10.	Cloud Database	The database mentioned above is a cloud based database.	IBM CLOUDANT DB
11.	File Storage	File storage requirements	IBM Block Storage or Other Storage Service or Local Filesystem
12.	External API	It will be used to alert users of various notifications etc as defined by the user.	SendGrid Service
13.	Machine Learning Model	The pre-trained model Xception is used which is one of the convolution neural net (CNN) architectures which is considered as a very good model for Image classification.	Xception(CNN)
14.	Deployment	: Application Deployment on Local System / Cloud	Local, Cloud Foundry, Kubernetes

		Local Server Configuration: The application will run on the local server/client side to allow user to interact with Web UI. Cloud Server Configuration: The application will be hosted on the cloud for the user to user. This is done through containerization of the application stored in the container registry.	

Table-2: Application Characteristics:

S.No	**Characteristics**	**Description**	**Technology**
1.	Open-Source Frameworks	Anaconda is an open source framework used in the project.	Anaconda
2.	Security Implementations	Some encryption methodology can be used to protect the application from security attacks.	SHA-256, Encryptions.
3.	Scalable Architecture	The architecture can be scaled to include the micro services and a detailed description of the implementation of the application logic.	Micro services

4.	Performance	We can use Automatic Verification Datasets ,Manual Verification Datasets, Manual k-Fold Cross Validation to evaluate the performance metrics	Nil

5.3 User Stories
Follow this link:-
https://github.com/IBM-EPBL/IBM-Project-19074-1659692878/blob/main/Project%20Design%20%26%20Planning/Project%20Design%20Phase%20II/Data%20Flow%20Diagrams%20.pdf

6.PROJECT PLANNING & SCHEDULING

6.1Sprint Planning & 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, password, and confirming my password.	3	High	Dharsini
Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the application	1	Low	Karthika
Sprint-2		USN-3	As a user, I can register for the application through Facebook	5	High	Kayanat
Sprint-1		USN-4	As a user, I can register for the application through Gmail	3	High	Kavya
Sprint-1	Login	USN-5	As a user, I can log into the application by entering email & password	3	High	Karthika
Sprint-2	Dashboard	USN-6	As a user, I can navigate through various sections of the application	5	Medium	Kavya,Karthika
Sprint-2	About Us Page	USN-7	As a user,I can know about the application and about the objectives and outcomes of predictions of the images.	6	Medium	Kavya,Dharsini
Sprint-2	Main Page	USN-8	As a user,I can upload the images for prediction and know more about the process.	10	High	Dharsini,Kayanat,Karthika
Sprint-3	Report Page	USN-9	As a user,I can access the results of my image prediction and use it for further use.	9	High	Kayanat,Karthika
Sprint-4	Contact Us Page	USN-10	As a user,I can contact the administrator via phone,email or post.	5	Medium	Karthika,Kavya

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-4	Hosting	USN-11	As a user,I can access the website both from mobile and PC.	8	High	Dharsini,Kayanat

6.2Sprint 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	10	6 Days	24 Oct 2022	29 Oct 2022	10	29 Oct 2022
Sprint-2	26	6 Days	31 Oct 2022	05 Nov 2022	26	31 Oct 2022
Sprint-3	9	6 Days	07 Nov 2022	12 Nov 2022	9	07 Nov 2022
Sprint-4	13	6 Days	14 Nov 2022	19 Nov 2022	13	14 Nov 2022

7.CODING & SOLUTIONING (Explain the features added in the project along with code)

7.1Login & Logout

This feature allows the user to login and logout into the web application .Thus it gives the user a personalized prediction

App.py;

```
@ 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'])):
                session['user'] = user
                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():
    session.pop('user',None)
    return render_template('logout.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/TYkiZh
lZB6+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+RTT6rIHI7NnikvbZlHgTPOO
mMi466C8"
      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>
```

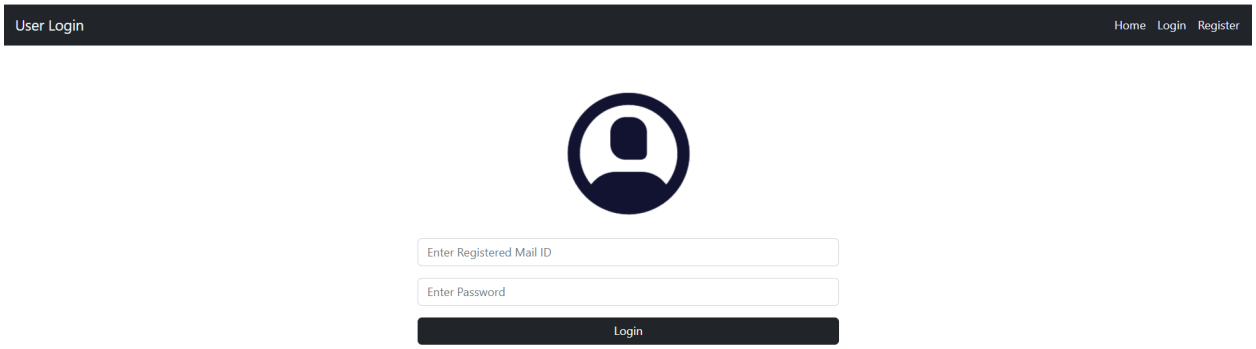
Logout.html:

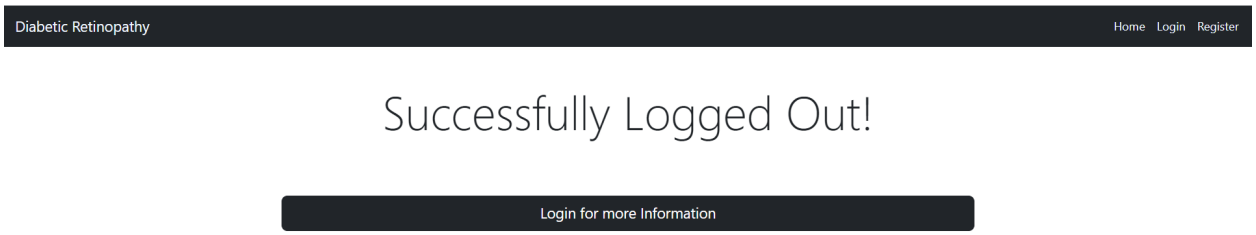
```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
```

```
    <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/TYkiZh
lZB6+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+RTT6rIHI7NnikvbZlHgTPOO
mMi466C8"
      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>
```

OUTPUT:





7.2Register:

This feature allows the user to register and create an account.

App.py:

```
# 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')
```

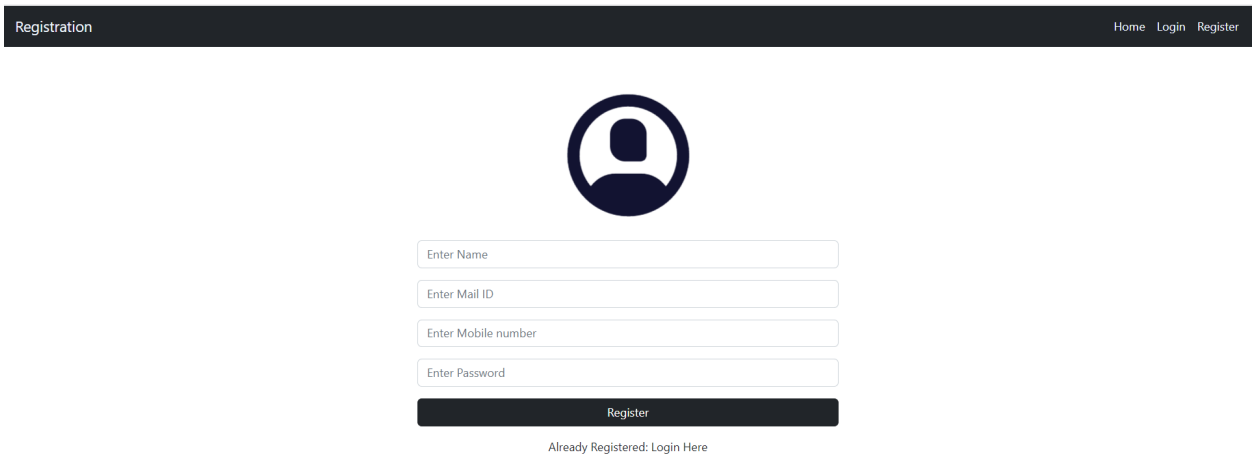
Register.html:

```
<!-- <!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/TYkiZh
lZB6+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+RTT6rIHI7NnikvbZlHgTPOO
mMi466C8"
      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> -->
```

OUTPUT:



7.3Prediction:

This feature allows the user to upload the picture of the scan and get a prediction made on the disease.

App.y

```
@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)
```

```
        return render_template('prediction.html', prediction=result, fname
= filepath)
    else:
        return render_template("prediction.html")
```

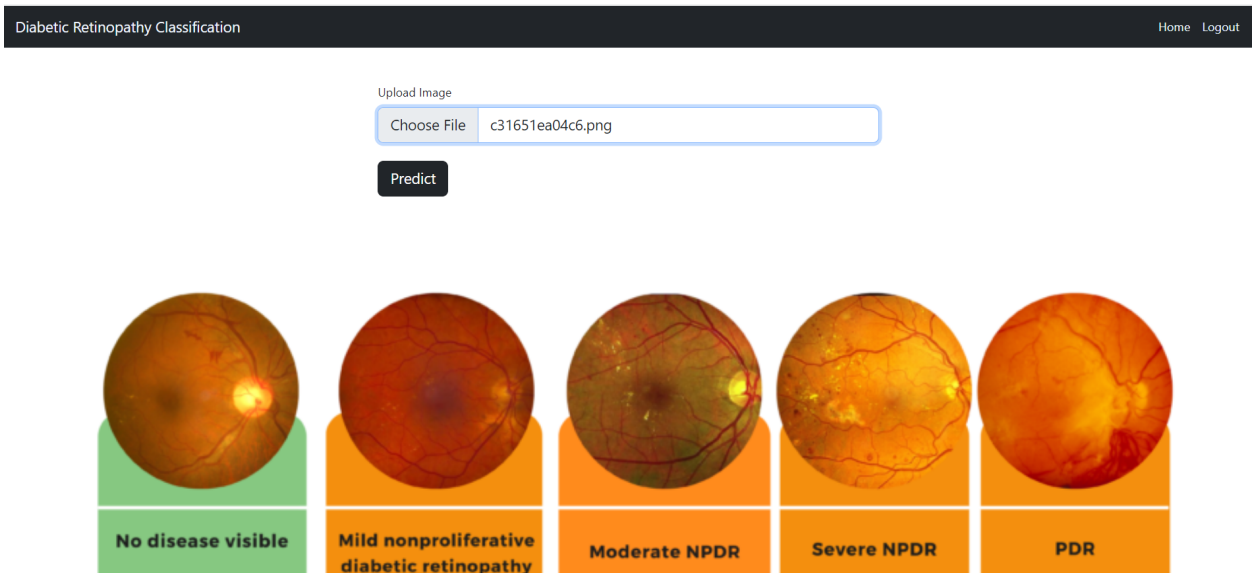
Prediction.html:

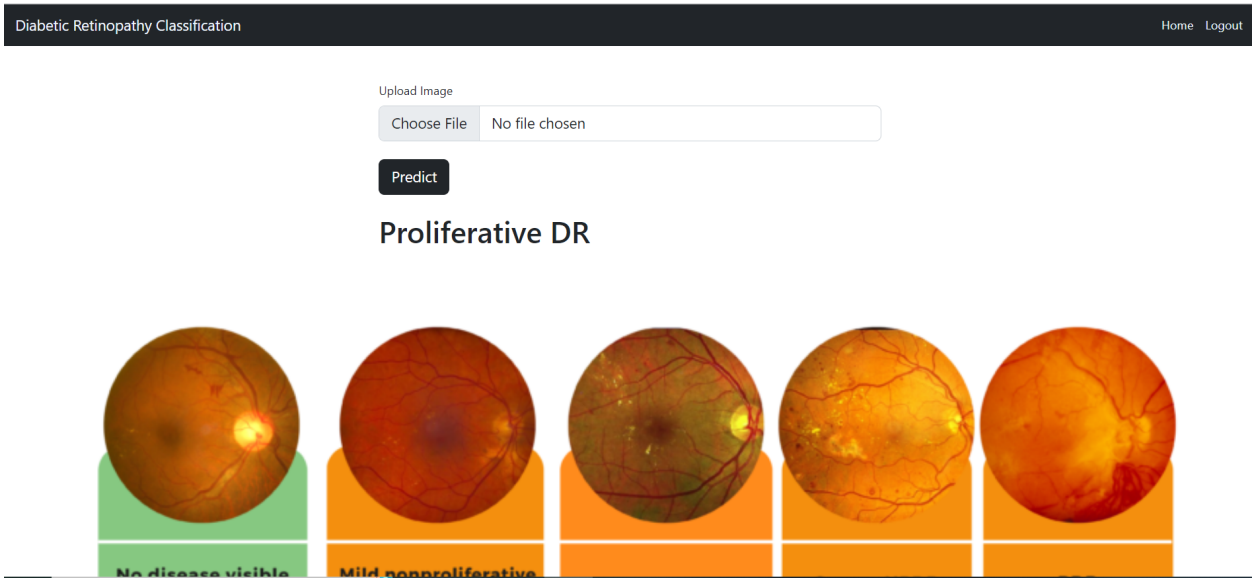
```
<!DOCTYPE html>
<html lang="en">

<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/TYkiZh
lZB6+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+RTT6rIHI7NnikvbZlHgTPOO
mMi466C8"
    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>
```

OUTPUT:





8.TESTING

8.1Test Cases

TEST CASE NO.	TEST CASE	EXPECTED RESULT	ACTUAL RESULT
1.	Incorrect mail id given for registration	Not registered	Not registered
2,	Password not provided	Details not stored	Details not stored
3.	Login to the application	Successful Login	Successful Login
4.	Login to the application with incorrect password	Login failed	Login failed
5.	Database creation	Database created	Database created
6.	Show current user login	Current user displayed	Current user displayed
7.	Logout of the current account	Successful Logout	Successful Logout

8.2User Acceptance Testing

TEST CASE NO.	TEST CASE	EXPECTED RESULT	ACTUAL RESULT
1.	Incorrect mail id given for registration	Not registered	Not registered
2,	Password not provided	Details not stored	Details not stored
3.	Login to the application	Successful Login	Successful Login
4.	Login to the application with incorrect password	Login failed	Login failed
5.	Logout of the current account	Successful Logout	Successful Logout

9.RESULTS

9.1Performance Metrics

Accuracy-The accuracy metric is one of the simplest Classification metrics to implement, and it can be determined as the number of correct predictions to the total number of predictions.

- Accuracy of the model is around-0.703
- Loss of the model is around-3.43

10.ADVANTAGES & DISADVANTAGES

ADVANTAGES:

1. Cross platform compatibility
2. More manageable
3. Highly deployable
4. Secure live data
5. Reduced costs

DISADVANTAGES

1.Less Integration with Device Functions

2.No Offline Availability

3.Slower Speeds

4.Not fully optimized

5.Slower performance

11. Conclusion

A deep learning fundus image analysis for early detection of diabetic retinopathy is successfully built and predicts the results. The effort taken by ophthalmologists to process DR retina fundus images is reduced as the deployed model can do it. The work of ophthalmologists is reduced as further processing and prediction is done by the model. Patients are at ease as the complex process of getting diabetic retinopathy detected is simplified and is available easily in a computer-aided diagnosis system. This system is easily accessible to patients and is portable. The model goes through training and testing of datasets and finally predicting results with greater accuracy.

12. Future Scope

Diabetic Retinopathy (DR) 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. Unfortunately, DR 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. Improvements in diabetes and diabetic retinopathy treatments have resulted from better understanding of pathophysiology and clinical trials that show the benefits of aggressive approaches. Future improvements will build on these successes to further reduce the risk of vision loss and will lead to early diagnosis and less invasive treatments.

13.APPENDIX

13.1Source Code

link- https://github.com/IBM-EPBL/IBM-Project-19074-1659692878/tree/main/Final%20Deliverables

13.2GitHub & Project Demo Link

Github link-https://github.com/IBM-EPBL/IBM-Project-19074-1659692878

Project Demolink-

https://github.com/IBM-EPBL/IBM-Project-19074-1659692878/blob/main/Final%20Deliverables/video_demo.mp4