Project report format

1. INTRODUCTION

1.1 Project overview

1.2 Purpose

2.LITERATURE SURVEY

2.1 Existing Problem

2.2 Reference

2.3 Problem Statement Definition

3.IDEATION & PROPOSED SOLUTION

3.1 Empathy map canvass

3.2 Ideation and brainstorming

3.3 Proposed Solution

3.4 Problem Solution fit

4.REQUIREMENT ANALYSIS

4.1 functional requirements

4.2 non-functional requirements

5.PROJECT DESIGN

5.1 Data flow diagrams

5.2 Solution & Technical Architecture

5.3 User Stories

6.PROJECT PLANNING & SCHEDULING

6.1 Sprint planning and Estimation

6.2 Sprint Delivery Schedule

6.3 Reports from JIRA

7.CODING & SOLUTIONING

7.1 Feature

1.INTRODUCTION:

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, XceptionV3 that are more widely used as a transfer learning method in medical image analysis and they are highly effective. The aim of this project is as follows:

- Gain a broad understanding of image data.
- Know how to pre-process/clean the data using different data preprocessing techniques.
- Know how to build a web application using the Flask framework

  The user interacts with the UI (User Interface) to choose the image.

  The chosen image analyzed by the model which is integrated with flask application.

1.1 Project overview:

1.2 Purpose:

The other common eye conditions reported by the World Health Organization include: myopia (near-sightedness), late detection in poorly integrated eye care services, and diabetic retinopathy (increasing numbers of people are living with diabetes, particularly nearly all people with diabetes will have some form of retinopathy in their lifetimes). For this reason, the early detection of diabetic retinopathy is essential to guarantee the maintenance of the vision. The first signs of diabetic retinopathy can be noticed using fundus photographs acquired by means of a retinal camera.

Diabetes occurs when the pancreas does not secrete enough insulin, or the body is unable to process it properly. Diabetes affects the circulatory system, and therefore to the retina. When

fluid leaks from blood vessels into the retina, this is damaged and this medical condition is called diabetic retinopathy (DR).

2.LITERATURE SURVEY:

2.1 Existing Problem:

1.Early Detection of Diabetic Retinopathy by Using Deep Learning Neural Network

This project presents a method to detect diabetic retinopathy on the fundus images by using deep learning neural network. Alex net Convolution Neural Network (CNN) has been used in the project to ease the process of neural learning. The data set used were retrieved from MESSIDOR database and it contains 1200 pieces of fundus images. The images were filtered based on the project needed. There were 580 pieces of images types. If it has been used after filtered and those pictures were divided into 2, which is Exudates images and Normal images. On the training and testing session, the 580 mixed of exudates and normal fundus images were divided into 2 sets which is training set and testing set. The result of the training and testing set were merged into a confusion matrix. The result for this project shows that the accuracy of the CNN for training and testing set was 99.3% and 88.3% respectively.

ADVANTAGES:

This project successfully detects the diabetes by using deep learning on a fundus image and it can be used as one of method to detect the diabetes on the future.

DISADVANTAGES:

On the testing set, it shows that the accuracy was 88.3% while on the training sets the accuracy was 99.3% which is approximately 100%. This shows a huge difference between the accuracy. It needs some improvement to make the accuracy of the project nearest to 100%.

ALGORITHM: MESSIDOR

2.Tomas, R., Halim, S., Gurudas, S., Sivaprasad, S. & Owens, D. Idf diabetes atlas:

A review of studies utilizing retinal photography on the global prevalence of diabetes related retinopathy between 2015 and 2018. Diabetes Research and Clinical Practice, p. 107840(2019).

The purpose of this study is to assess the prevalence of diabetic retinopathy (DR) world-wide from articles published since 2015 where the assessment of the presence and severity of DR was based on retinal images.

ADVANTAGES:

The global prevalence of DR and DME, for the period 2015 to 2019 were 27.0% for any DR comprising of 25.2%, NPDR, 1.4% PDR and 4.6% DME.

DISADVANTAGES:

This study illustrated difficulties in deriving a meaningful global prevalence rate for DR and DME due to the lack of uniformity in defining the study populations, methodological differences, retinal image capture and grading criteria.

3.Convolutional Neural Networks for Diabetic Retinopathy:

The diagnosis of diabetic retinopathy (DR) through color fundus images requires experienced clinicians to identify the presence and significance of many small features which, along with a complex grading system, makes this a difficult and time-consuming task. In this paper, we propose a CNN approach to diagnosing DR from digital fundus images and accurately classifying its severity.

We develop a network with CNN architecture and data augmentation which can identify the intricate features involved in the classification task such as micro-aneurysms, exudate and hemorrhages on the retina and consequently provide a diagnosis automatically and without user input. We train this network using a high-end graphics processor unit (GPU) on the publicly available Kaggle dataset and demonstrate impressive results, particularly for a high-level classification task. On the data set of 80,000 images used our proposed CNN achieves a sensitivity of 95% and an accuracy of 75% on 5,000 validation images.

ADVANTAGES:

The potential benefit of using this trained CNN is that it can classify thousands of images every minute allowing it to be used in real-time whenever a new image is acquired.

DISADVANTAGES:

The dataset itself is a disadvantage because the number of healthy eyes is in large number which makes the network has no learning issue to detect an image of a healthy eye. Whereas in case of having large number of unhealthy retinal images makes the network inappropriate to learn and detect. In training the learning required to classify the images at the extreme ends of the scale was significantly less. The issues came in making the network to distinguish between the mild, moderate and severe.

2.2 Reference:

1. Marshall, S. M. & Flyvbjerg, A. Prevention and early detection of vascular complications of diabetes. BMJ. 333(7566), 475–480 (2006).

2. Hutchinson, A. et al. Effectiveness of screening and monitoring tests for diabetic retinopathy–a systematic review. Diabet Med. 17(7), 495–506 (2000).

3. Taylor, R. & Batey, D. Handbook of Retinal Screening in Diabetes: Diagnosis and Management. Wiley (2012).

4. Gulshan, V. et al. Development and Validation of a Deep Learning Algorithm for Detection of Diabetic Retinopathy in Retinal Fundus Photographs. JAMA. 316(22), 2402–2410 (2016).

5. Ting, D. S. W. et al. Development and Validation of a Deep Learning System for Diabetic Retinopathy and Related Eye Diseases Using Retinal Images from Multiethnic Populations with Diabetes. JAMA. 318(22), 2211–2223 (2017).

6. Wilkinson, C. P. et al. Proposed international clinical diabetic retinopathy and diabetic macular edema disease severity scales. Ophthalmology. 110(9), 1677–1682 (2003).

7. Summanen, P. et al. Update on Current Care Guideline: Diabetic retinopathy. Duodecim. 131(9), 893–894 (2015).

8. Krause, J. et al. Grader Variability and the Importance of Reference Standards for Evaluating Machine Learning Models for Diabetic Retinopathy. Ophthalmology. 125(8), 1264–1272 (2018).

9. Guan, M. Y., Gulshan, V., Dai, A. M. & Hinton, G. E. Who Said What: Modeling Individual Labelers Improves Classification. arXiv e-prints., https://ui.adsabs.harvard.edu/\#abs/2017arXiv170308774G. (Accessed March 01, 2017).

10.Wang, F., Casalino, L. P. & Khullar, D. Deep Learning in MedicinePromise, Progress, and Challenges. JAMA Intern Med. (2018).

11.MedicalResearchAct, https://www.finlex.fi/en/laki/kaannokset/1999/en19 990488_20100794.pdf (2010).

12.Goodfellow, I., Bengio, Y. & Courville, A. Deep Learning. MIT Press (2016).

2.3 Problem Statement Definition:

Diabetic retinopathy is one of the most threatening complications of diabetes that leads to permanent blindness if left untreated. One of the essential challenges is early detection, which is very important for treatment success. Unfortunately, the exact identification of the diabetic retinopathy stage is notoriously tricky and requires expert human interpretation of fundus images. Simplification of the detection step is crucial and can help millions of people. Convolutional neural networks (CNN) have been successfully applied in many adjacent subjects, and for diagnosis of diabetic retinopathy itself. However, the high cost of big labeled datasets, as well as inconsistency between different doctors, impede the performance of these methods. In this paper, we propose an automatic deep-learning-based method for stage detection of diabetic retinopathy by single photography of the human fundus. Additionally, we propose the multistage approach to transfer learning, which makes use of similar datasets with different labeling. The presented method can be used as a screening method for early detection of diabetic retinopathy

3.IDEATION & PROPOSED SOLUTION:

3.1 Empathy map canvass:

GAINS	PAINS
PATIENT RECOVERY	MISSING PATIENT DATA
NEW TECHNOLOGY EVOLUTION	CONVENCING PATIENTS TO UNDER GO NEW TREMENT
NEXT LEVEL OF MEDICAL RESEARCH	SECURITY FOR PATIENT DATA'S
OBTAINED CURABLE MEDICINE	STRUGGLES/CHALLENGES

3.2 Ideation and brainstorming:

I am	I'm Trying to			
Student	To find exact solution to myinfected blurred vision	I Couldn't find a exact solution For this blurred vision	I Couldn't Find What's a problem in my eyes for a blurred vision	Sad, Scored poor marks

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.

- **15 minutes** to prepare
- **1 ½ hour** to collaborate
- **1 people** recommended

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.

10–15 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 →

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 - 10 minutes

PROBLEM

How can we detect diabetic retinopathy and avoid total loss of eyesight?

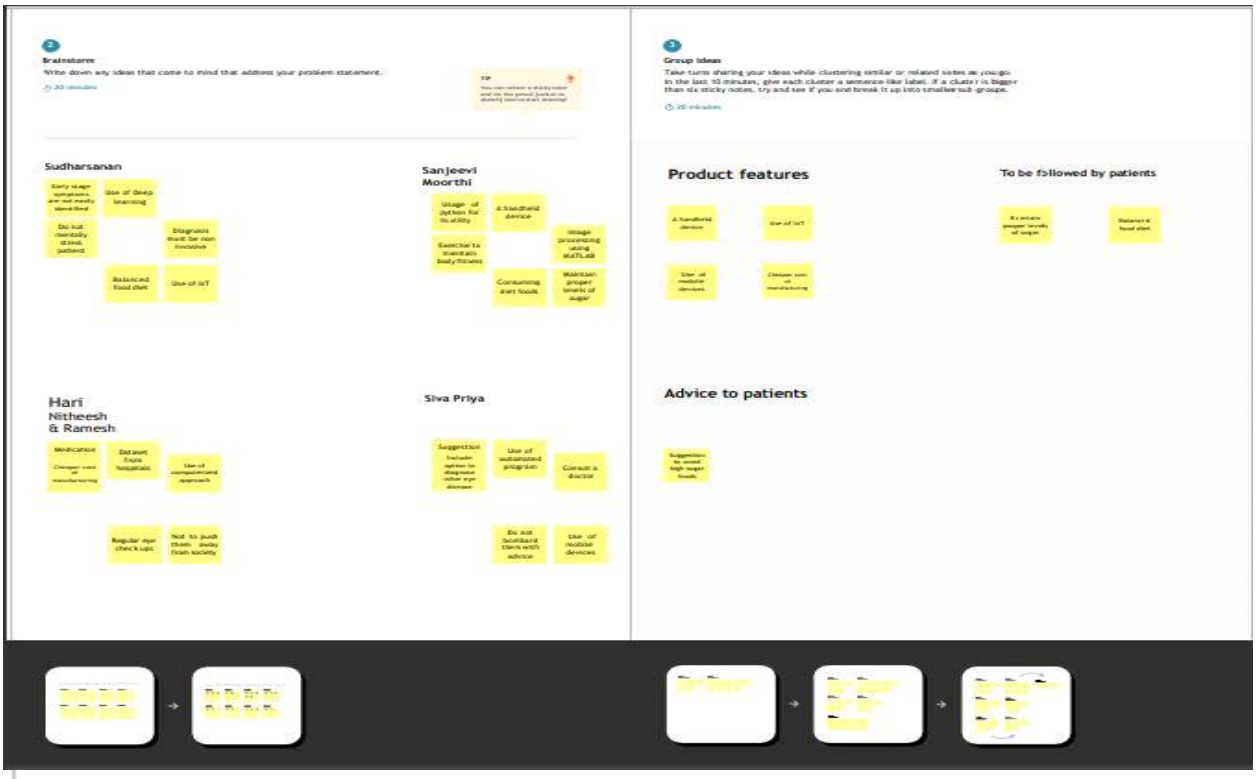
Key rules of brainstorming

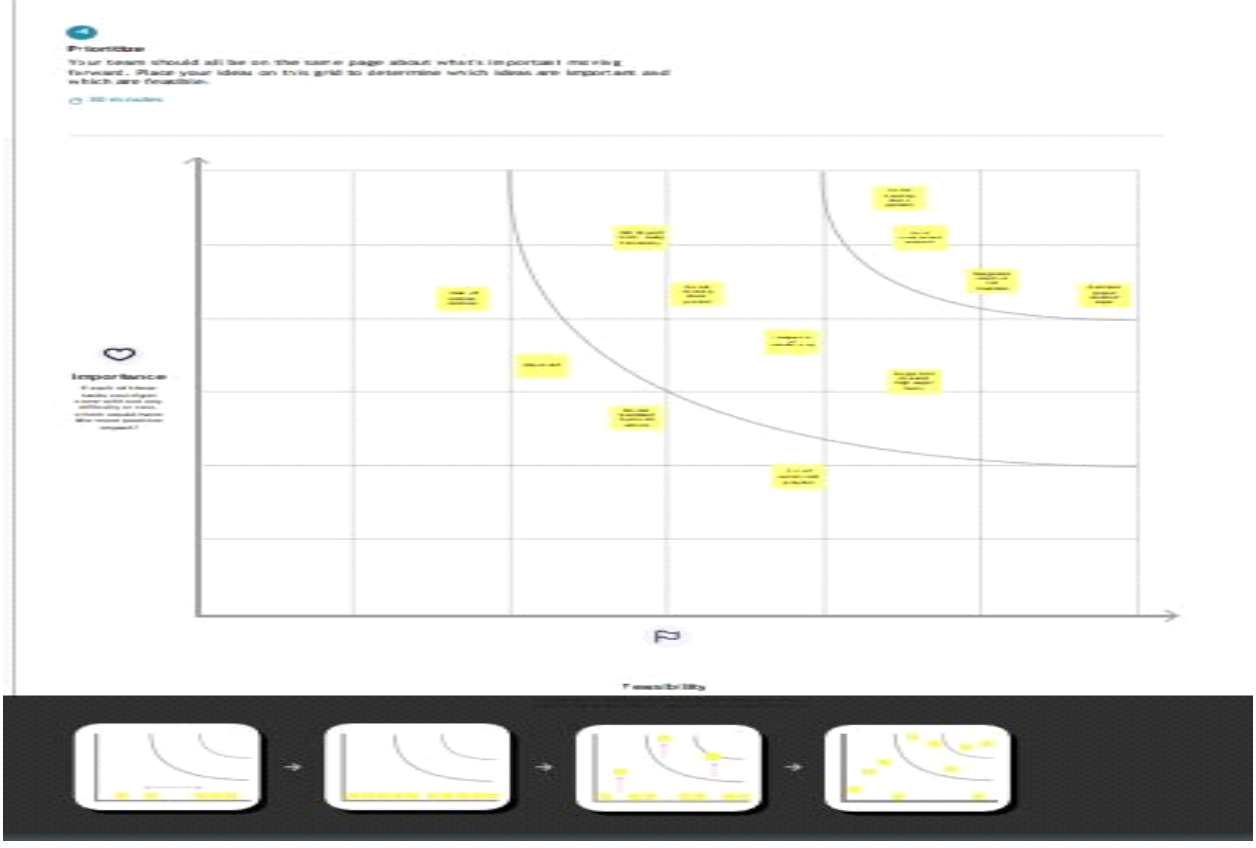
To run an smooth and productive session

- Stay in topic.
- Encourage wild ideas.
- Defer judgment.
- Listen to others.
- Go for volume.
- If possible, be visual.

Need some inspiration?

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



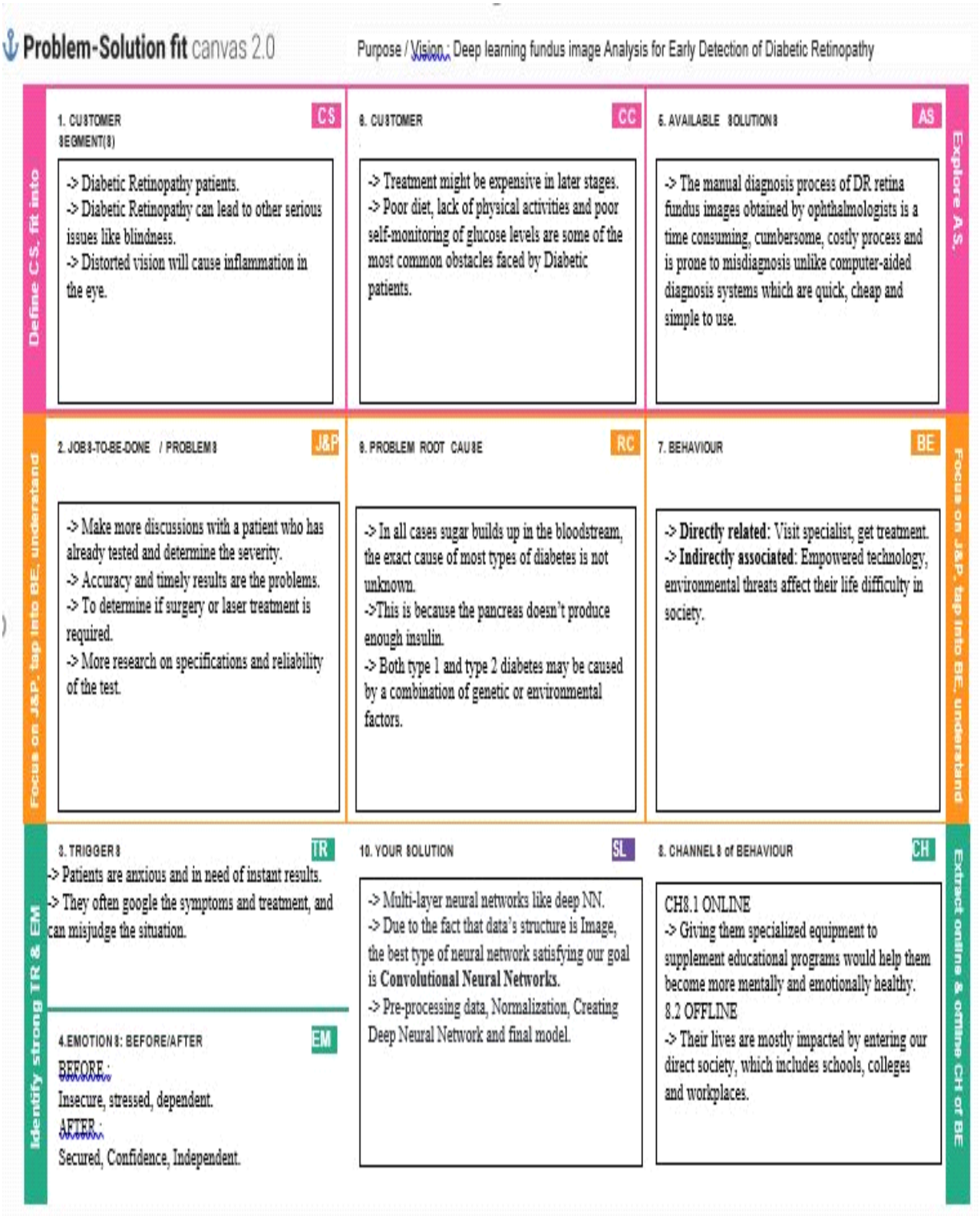


3.3 Proposed Solution:

The main aim of this project is to create an appropriate machine learning model to detect Diabetic Retinopathy as early as possible.

S.No.	Parameter	Description
•	Problem Statement (Problem to be solved)	1)To find the presence of lesions in the eye. 2) To find HbA1c level 3) Early detection of illness
•	Idea / Solution description	1)Prediction is done at a faster rate. 2)Accuracy of prediction. 3)laser treatment can stop or slow the leakage of blood and fluid in the eye.
•	Novelty / Uniqueness	1) Use of powerful deep neural network. 2) It provides robust and trusted support. 3)Maintaining database which contains details of the disease.
•	Social Impact / Customer Satisfaction	1) Reduction of Diabetic Retinopathy risk. 2) Provides Digital Assistance. 3) Very helpful in making decisions faster. 4) Can be used 24x7.
•	Business Model (Revenue Model)	1) This can be implemented as an essential diagnosis method in every hospital. 2) Accurate detection and analysis can encourage the increase in financial benefit.
•	Scalability of the Solution	1) Accurate predictions and extensive use. 2) Based on the times of the correct diagnosis. 3) Availability.

3.4 Problem Solution fit:



4.REQUIREMENT ANALYSIS:

4.1Functional requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Uploading Image	Uploading image by taking selfie or by accessing the camera album.
FR-2	Get Verification	Get verification code from mail.
FR-3	Collect Data	Collect data from various sources for training model , by accessing the data.
FR-4	Create Samples	Creating samples from the data collected and train the module according to the information from the user.
FR-5	Conclusion Report	After analysis a conclusion report is generated to the user.

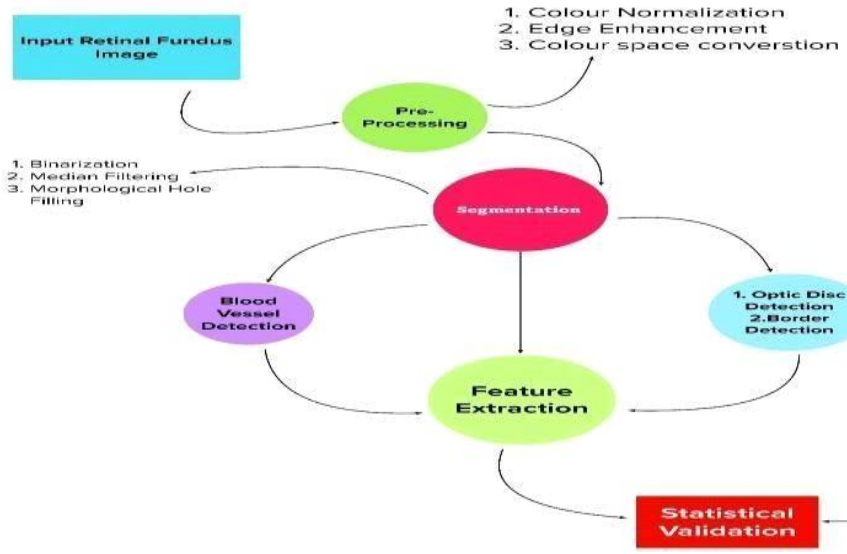
4.2 Non-functional requirements:

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

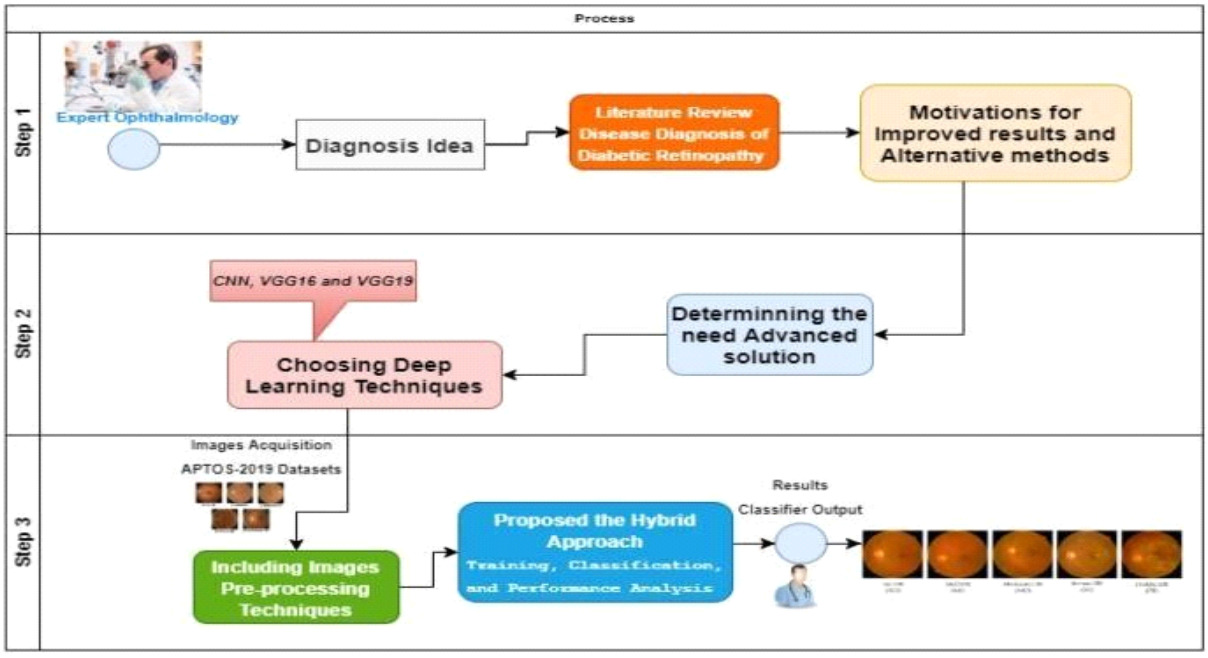
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The specific product should be easily usable to each and every user all around the world.
NFR-2	**Security**	Security of the users plays a very important role and the specific product should satisfy it.
NFR-3	**Reliability**	The specific product must be totally trusted by the user .
NFR-4	**Performance**	The performance of the product plays a very important role.It determines the satisfaction of the user.
NFR-5	**Availability**	The product should be easily available in the market.
NFR-6	**Scalability**	The product should be small and compact in size,in that case it can be easily carried by the user.

5.PROJECT DESIGN:

5.1 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.



- The retinal fundus images are commonly used for detection and analysis of diabetic retinopathy disease in clinics.
- Pre-processing of raw retinal fundus images are performed using extraction of green channel, histogram equalization, image enhancement and resizing techniques.

- The segmentation of retinal vasculature from eye fundus images is a fundamental task in retinal image analysis.
- The computer aided automatic detection and segmentation of blood vessels through the elimination of optic disc (OD) region in retina.
- The retinal blood vessels are detected using mathematical binary morphological operations.
- Feature extraction from the fundus images for the diagnosis of Diabetic Retinopathy refers to an advanced eye screening technology by which eye related diseases can be detected at an early stage.

5.2 Solution & Technical Architecture:

Solution Architecture:

The main objective is to detect the Diabetic Retinopathy in early stages by processing the Retinal fundus images. 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. Deep Learning, Machine learning, Neural Networks and knowledge in Python will play a significant role in the development of our Project.

Since the data we use are Images, the best type of neural network satisfying our goal is Convolutional Neural Networks. Before any classification, pre-processing techniques will be implemented. For this, mentioned techniques has been used to find and bold the intensity of the abnormal areas and pieces for decreasing the effect of outliers. Some of images have abnormal structures. For instance, optic disk and vessels are abnormal, using gray-scaled images reduces its effects.

Consequently, better accuracy will be achieved by pre-processed data. After doing pre-processing and normalizing, appropriate features will be extracted for the neural network to be trained on. Over this the deep NN will be trained and results will be evaluated by varying the hyper parameters.

Technical Architecture:

Project will fulfill the following information in this technology architecture.

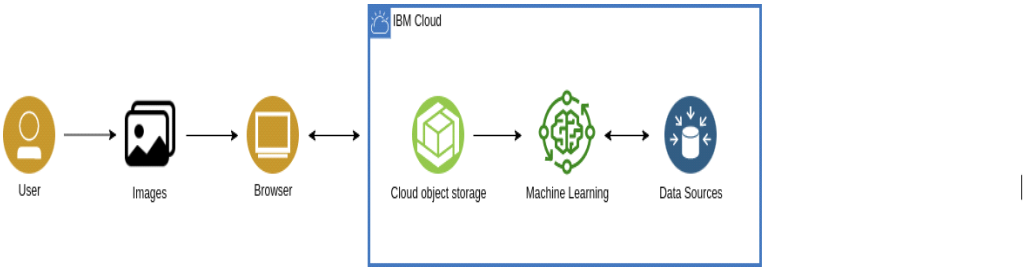


Table-1 : Components & Technologies:

S.N o	Component	Description	Technology
1.	User Interface	How user interacts with application e.g. Web UI, Mobile App, Chatbot etc.	HTML, CSS, JavaScript etc.
2.	Application Logic-1	Logic for a process in the application	Python, Flask
3.	Database	Data Type, Configurations etc.	MySQL, NoSQL, etc.
4.	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant etc.
5.	Machine Learning Model	Purpose of Machine Learning Model	Diabetic Retinopathy detection
6.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: Cloud Server Configuration :	Cloud.

Table-2: Application Characteristics:

S.N o	Characteristics	Description	Technology
1.	Open-Source Frameworks	List the open-source frameworks used	Flask, TensorFlow. Keras. Numpy, Pandas
2.	Security Implementations	List all the security / access controls implemented, use of firewalls etc.	Built-in protection.
3.	Scalable Architecture	Justify the scalability of architecture (3 – tier, Micro-services)	3-tiers.
4.	Availability	Justify the availability of applications (e.g. use of load balancers, distributed servers etc.)	Load balancer.
5.	Performance	Design consideration for the performance of the application (number of requests per sec, use of	It depends upon the input images.

		Cache, use of CDN's) etc.	

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Patient (Web user)	Registration	USN-1	As a user, I can register through website either email id or phone number with password.	I can create my account.	High	Sprint-3
	Login	USN-2	As a user, I can login to the site by the given Login credentials.	I can login and access my account.	High	Sprint-3
	Upload image	USN-3	As a user, I can upload my data in the form of pdf, doc etc.	I can upload my data's.	Medium	Sprint-3
Administration (Web developer)	Admin login	USN-4	As an Admin I can login to the site and analyze the user data.	I can login and analyze the user data.	High	Sprint-3
	Data collection	USN-5	As an admin, I can collect the dataset related to the DR from source.	I can collect the dataset.	Low	Sprint-1
	Create model	USN-6	As an admin, I can create the model and train the model from the dataset for prediction.	I can create and train the model.	High	Sprint-1
	Test the model	USN-7	As an admin, I can test the model for prediction.	I can test the model.	High	Sprint-2
Patient (Web user)	Diagnosis	USN-8	As a user I can get the diagnosis result on the application and follow up with treatments.	He/she can get the results and continue the treatment.	High	Sprint-2

5.3 User Stories:

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Patient (Webuser)	Registration	USN-1	I can register as a user on the website with eitheran email address or a phone number and password.	I can createmy account.	High	Sprint-3
	Login	USN-2	With theprovided Login credentials, I canaccessthe website as a user.	I can log in andaccess myaccount	High	Sprint-3
	Upload image	USN-3	I can post my data as a userin formats likepdf and doc.	I can uploadmy data.	Medium	Sprint-3
Administration (Web developer)	Admin Login	USN-4	I can log in to the website as theadmin and analyze the user information	I can log in and analyze the user data.	High	Sprint-3
	Data collection	USN-5	I can gatherthe dataset forthe DR fromthe source as anadmin.	I can collect the dataset.	Low	Sprint-1
	Create model	USN-6	I can buildthe model andtrain it using the dataset as an administrator to makepredictions.	I can create andtrain the model.	High	Sprint-1
	Test the model	USN-7	I canevaluate the model's predictive abilities as an admin.	I can testthe model.	High	Sprint-2
Patient (Web user)	Diagnosis	USN-8	I can access the application's diagnosisresults as a userand continue with treatments..	He/she can get the resultsand continue the treatment.	High	Sprint-2

6.PROJECT PLANNING & SCHEDULING

6.1 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 should be able to register myself with username, password, mobile number, email-id, location	5	High	Sudharsanan Hari Nitheesh Siva priya
Sprint-2	Login	USN-2	As a user, I should be able to register myself and should have forget password for recovery	5	High	Sudharsanan Hari Nitheesh Siva priya
Sprint-1		USN-3	As a user, I can login into my application using my username and password	5	High	Sudharsanan Hari Nitheesh Siva priya
Sprint-2		USN-4	As a user, I should be able to post my queries in the application	7	High	Sanjeevimoorthi Ramesh

Sprint-4	Dashboard	USN-5	As a user, I should	5	Medium	Ramesh
			be able to modify the credentials given by me like my location to get correct suggestions of hospitals nearby			Sanjeevimoorthi
Sprint-4	Database	USN-6	As an administrator I should be able to update the contact details and addresses of hospitals	5	Medium	Sanjeevimoorthi Ramesh
Sprint-3		USN-7	As an administrator I should be able to read and respond to all the user queries from comment section	5	Medium	Sanjeevimoorthi Ramesh
Sprint-3	User Interface (Detection)	USN-8	As a user, I should be able to upload the image of my retina and should get accurate results of the diagnosis	9	High	Sanjeevimoorthi Ramesh

6.2 Sprint Delivery Schedule:

Sprint-1

Task	Hours	Day-1	Day-2	Day-3	Day-4	Day-5	Day-6	Total
USN-1	15	3	2	3	2	2	3	15
USN-3	15	4	3	2	2	2	2	15

Sprint-2

Task	Hours	Day-1	Day-2	Day-3	Day-4	Day-5	Day-6	Total
USN-2	12	2	2	2	2	2	2	12
USN-4	24	4	0	5	5	5	5	24

Sprint-3

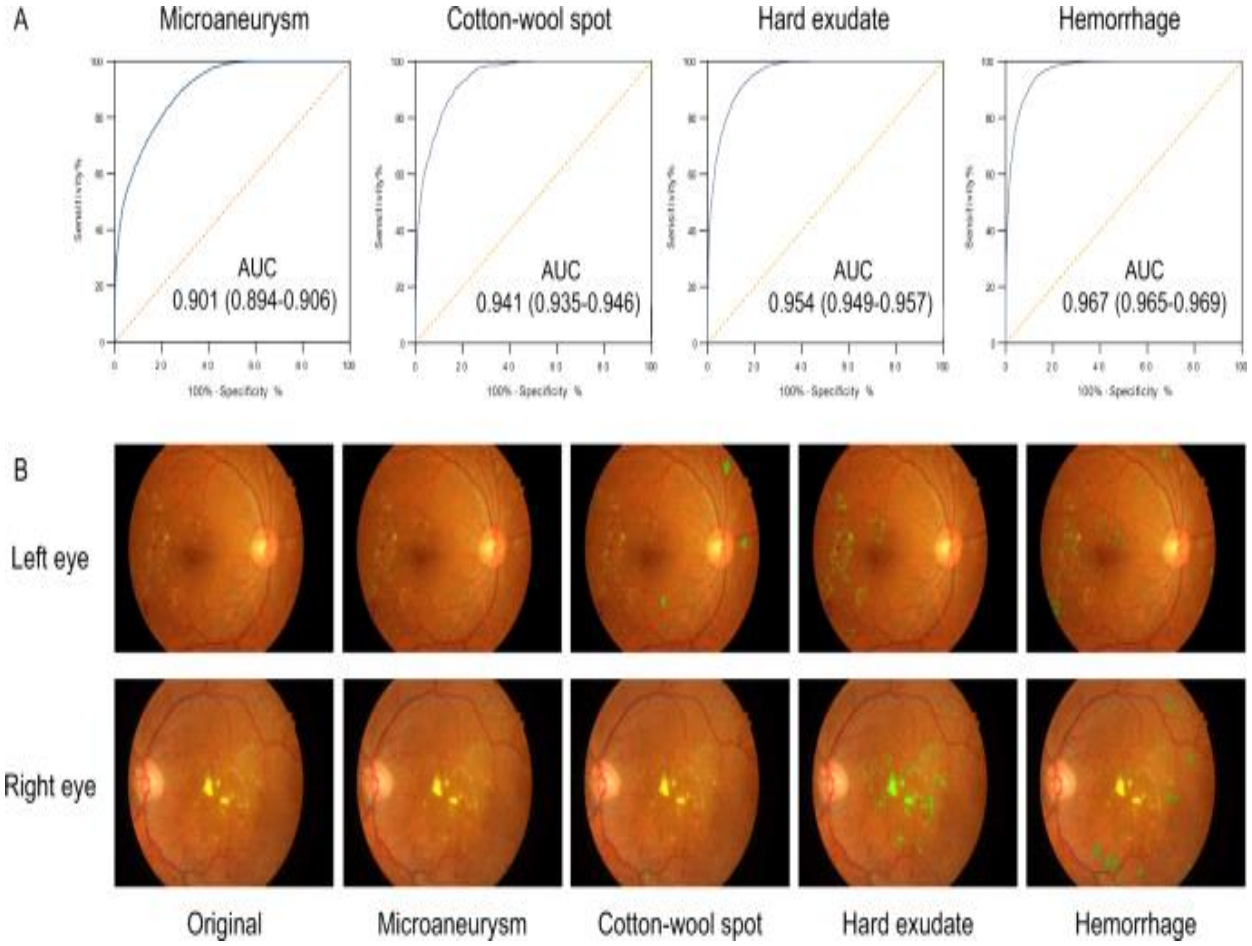
Task	Hours	Day-1	Day-2	Day-3	Day-4	Day-5	Day-6	Total
USN-7	15	2	2	4	3	2	2	15
USN-8	27	5	5	5	4	4	4	27

Sprint-4

Task	Hours	Day-1	Day-2	Day-3	Day-4	Day-5	Day-6	Total
USN-5	12	2	2	2	2	2	2	12

USN-6	12	2	2	2	2	2	2	12

6.3 Reports from JIRA:



7.CODING & SOLUTIONING:

7.1 Feature:

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

7.2 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:

8.1 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

3.Test-CaseAnalysis:

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:

9.1 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 -	-- -- -- --

10.ADVANTAGES & DISADVANTAGES:

ADVANTAGES :

Advantages of implementing deep learning–based algorithms in DR screening include reduction in manpower, cost of screening, and issues relating to intragrader and intergrader variability. However, limitations that may hinder such an implementation particularly revolve around ethical concerns regarding lack of trust in the diagnostic accuracy of computers. Considering both strengths and limitations, as well as the high performance of deep learning–based algorithms, automated DR classification using deep learning could be feasible in a real-world screening scenario.

DISADVANTAGES:

The abnormal blood vessels associated with diabetic retinopathy stimulate the growth of scar tissue, which can pull the retina away from the back of the eye. This can cause spots floating in your vision, flashes of light or severe vision loss.

11.CONCLUSION:

In this paper, we proposed the multistage transfer learning approach and an automatic method for detection of the stage of diabetic retinopathy by single photography of the human fundus. We have used an ensemble of 3 CNN architectures (EfficientNet-B4, EfficientNet-B5, SE-ResNeXt50) and made transfer learning for our final solution. The experimental results show that the proposed method achieves high and stable results even with unstable metric. The main advantage of this method is that it increases generalization and reduces variance by using an ensemble of the networks, pretrained on a large dataset, and finetuned on the target dataset. The future work can extend this method with the calculation of SHAP for the whole ensemble, not only for a particular network, and with more accurate hyperparameter optimization. Besides, we can do experiments using pretrained encoders on other connected to eye ailments tasks. Also, it is possible to investigate meta-learning (Nichol et al., 2018) with these models, but realized that it requires the separate in-depth research.

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:

Index.html

```
<!DOCTYPE html>
```

```
<html>
    <head>
        <title>PNT2022TMID43448 Deep learning fundus image analysis for early detection of Diabetic Retinopathy</title>
        <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Roboto">
    </head>
    <style>

        html,body{
        margin:0;
        width:100%;
        height:100%;
        font-family:Roboto;
        display: flex;
        flex-direction: column;
    background-image: url(back.webp);
        }
        .header{
        position: fixed;
        top: 0px;
        width:100%;
        height:100px;
        background-color:rgb(74, 246, 255);
        display: flex;
        flex-direction: row;
        align-items: center;
        justify-content: space-between;
        }
        #heading{
```

```
padding:10px;
color:white;
font-size:25px;
font-style: italic
}
#option1{
margin-right:2px;
padding:10px;
color:white;
font-size:20px;
}
.options{
display: flex;
}
a{
text-decoration:none;
color:white;
}
#option1:hover{
border:1px solid white;
border-radius:10px;
background-color:black;
}
.container{
display:flex;
justify-content:center;
align-items: center
flex-direction:row;
```

```
        }
        #imgs{
        margin-top:100px;
        height: 100px;
        width: 60vw;
        }
        #subhead{
        font-size: 30px;
        color: rgb(247, 247, 247);
        text-align: center;
        text-decoration-line: underline;
        }
        th, td {
        width:40vw;
        padding: 15px;x
        }
        th{
        color:rgb(5, 24, 231);
        text-decoration-line: underline;
        }
        td{
        color:rgb(250, 247, 247);
        text-align:justify;
        }

        </style>
<style>
   html,body{
```

```
margin:0;
width:100%;
height:100%;
font-family:Roboto;
display: flex;
flex-direction: column;
}
.header{
position: fixed;
top: 0px;
width:100%;
height:100px;
background-color:rgb(4, 10, 15);
display: flex;
flex-direction: row;
align-items: center;
justify-content: space-between;
}
#heading{
padding:10px;
color:white;
font-size:25px;
font-style: italic
}
#option1{
margin-right:2px;
padding:10px;
color:white;
```

```
font-size:20px;
}
.options{
display: flex;
}
a{
text-decoration:none;
color:white;
}
#option1:hover{
border:1px solid white;
border-radius:10px;
background-color:black;
}
.container{
margin-top: 100px;
display: flex;
flex-direction: row;
align-items: center;
justify-content: center;
border:2px solid rgb(255, 255, 255);
}
.retinaimg{
margin-top:25px;
height:100px;
width: 100px;
border-radius:50%;
padding:20px;
```

```
        }
        .stages{
        height:500px;
        width:600px
        }
        hr {
        width:100%;
        border:1px solid black;
        background:black;
        }
        </style>
    <body>
            <div class="header">
                <p id="heading">Deep learning fundus image analysis for early detection of diabetic retinopathy</p>
                <div class="options">
                    <p id="option1"><a href="Index.html">Home</a></p>
                    <p id="option1"><a href="register.html">Register</a></p>
                    <p id="option1"><a href="login.html">Login</a></p>
        <p id="option1"><a href="Pridiction.html">Pridiction</a></p>
                </div>
            </div>
            <div class="container">
    </div>
            <table><br>
                <pre>ABOUT PROJECT</pre>
                <tr><th>Problem</th>
                    <th>Solution</th>
        <img src="drimage.jpg" alt="drimage">
```

<tr><td>
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.</td>

<td>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. </td>

</table>

</body>

</html>

Python Code:


```
{
 "nbformat": 4,
 "nbformat_minor": 0,
 "metadata": {
  "colab": {
   "provenance": []
  },
```

```
  "kernelspec": {
    "name": "python3",
    "display_name": "Python 3"
  },
  "language_info": {
    "name": "python"
  }
},
"cells": [
  {
    "cell_type": "code",
    "execution_count": null,
    "metadata": {
      "id": "dYnBvBOw6SKE"
    },
    "outputs": [],
    "source": [
      "import numpy as np\n",
      "import os\n",
      "from tensorflow.keras.models import load_model\n",
      "from tensorflow.keras.preprocessing import image\n",
      "from tensorflow.keras.applications.inception_v3 import preprocess_input\n",
      "from cloudant.client import  Cloudant\n",
      "from werkzeug.utils import secure_filename\n",
      "from flask import Flask, request, render_template, redirect, url_for,session"
    ]
  },
  {
```

```
"cell_type": "code",
"source": [
 "app=Flask(__name__)\n",
 "\n",
 "client=Cloudant.iam('f17a994d-760b-40dc-baff-83fc3b995d7b-bluemix','69W0Zt5eGFE1LX4qtysIXCC4Xm-qG1l8ZyfpkpiS10oF',connect=True)\n",
 "my_database = client['db']\n",
 "app.secret_key=\"SECRET_KEY\"\n",
 "model=load_model(r\"inception-diabetic.h5\")\n",
 "image_folder=os.path.join('static','images')\n",
 "app.config['UPLOAD_FOLDER'] = image_folder\n",
 "\n",
 "\n",
 "@app.route('/')\n",
 "def index():\n",
 "    full_filename = os.path.join(app.config['UPLOAD_FOLDER'], 'drimage.jpg')\n",
 "    return render_template('index.html',image=full_filename)\n",
 "\n",
 "@app.route('/index')\n",
 "def home():\n",
 "    full_filename = os.path.join(app.config['UPLOAD_FOLDER'], 'drimage.jpg')\n",
 "    return render_template('index.html',image=full_filename)\n",
 "\n",
 "@app.route('/register')\n",
 "def register():\n",
 "    full_filename = os.path.join(app.config['UPLOAD_FOLDER'], 'registerimg.jpg')\n",
 "    return render_template('register.html',image=full_filename)\n",
 "\n",
 "@app.route('/afterreg',methods=['POST','GET'])\n",
```

```
"def afterreg():\n",
"    x=[x for x in request.form.values()]\n",
"    data={\n",
"        '_id':x[2],\n",
"        'name':x[0],\n",
"        'pwd':x[4],\n",
"        'email':x[1],\n",
"        'location':x[3],\n",
"        'securityquestion':x[5],\n",
"        'logintype':x[6]\n",
"    }\n",
"    query={'_id':{'$eq':data['_id']}}\n",
"    docs=my_database.get_query_result(query)\n",
"    if(len(docs.all())==0):\n",
"        url=my_database.create_document(data)\n",
"        full_filename = os.path.join(app.config['UPLOAD_FOLDER'], 'loginimg.jpg')\n",
"        return render_template('login.html',predict=\"Registration successfull please login using your credentials\",image=full_filename)\n",
"    else:\n",
"        full_filename = os.path.join(app.config['UPLOAD_FOLDER'], 'registerimg.jpg')\n",
"        return render_template('register.html',pred=\"You are already a member login using your credentials\",image=full_filename)\n",
"\n",
"@app.route('/login')\n",
"def login():\n",
"    full_filename = os.path.join(app.config['UPLOAD_FOLDER'], 'loginimg.jpg')\n",
"    return render_template('login.html',image=full_filename)\n",
"\n",
"@app.route('/afterlogin', methods=['POST','GET'])\n",
```

```
"def afterlogin():\n",
"    user=request.form['phoneno']\n",
"    session['pn']=user\n",
"    passw=request.form['pwd']\n",
"    lgnas=request.form['loginas']\n",
"    \n",
"    query={'_id':{'$eq':user}}\n",
"    docs=my_database.get_query_result(query) \n",
"    \n",
"    if(len(docs.all())==0):\n",
"        full_filename = os.path.join(app.config['UPLOAD_FOLDER'], 'loginimg.jpg')\n",
"        return render_template('login.html',predict=\"Phone number/id not found\",image=full_filename)\n",
"    else:\n",
"        if((user==docs[0][0]['_id'] and passw==docs[0][0]['pwd'] and lgnas==docs[0][0]['logintype'] )):\n",
"            if(docs[0][0]['logintype']=='user'):\n",
"                full_filename = os.path.join(app.config['UPLOAD_FOLDER'], 'retina.jpg')\n",
"                full_filename1 = os.path.join(app.config['UPLOAD_FOLDER'], 'image6.png')\n",
"                return render_template('prediction.html',image=full_filename,image2=full_filename1)\n",
"            if(docs[0][0]['logintype']=='admin'):\n",
"                full_filename2 = os.path.join(app.config['UPLOAD_FOLDER'], 'adminimg.jpg')\n",
"                return render_template('admin.html',image=full_filename2)\n",
"        if(lgnas!=docs[0][0]['logintype']):\n",
"            full_filename = os.path.join(app.config['UPLOAD_FOLDER'], 'loginimg.jpg')\n",
"            return render_template('login.html',image=full_filename,predict=\"Incorrect Logintype\")\n",
```

```
"        if(passw!=docs[0][0]['pwd']):\n",
"            full_filename = os.path.join(app.config['UPLOAD_FOLDER'], 'loginimg.jpg')\n",
"            return render_template('login.html',image=full_filename,predict=\"Incorrect password\")\n",
"\n",
"        \n",
"@app.route('/respond')\n",
"def respond():\n",
"    my_database_query = client['my_database_query']\n",
"    \n",
"    dt=[]\n",
"    for document in my_database_query:\n",
"        dt.append(document['who'])\n",
"        dt.append(document['phoneno'])\n",
"        dt.append(document['query'])\n",
"    return render_template('respond.html',data=dt)\n",
"\n",
"@app.route('/afterrespond', methods=['POST','GET'])\n",
"def afterrespond():  \n",
"    my_database_query = client['my_database_query']\n",
"    x=[x for x in request.form.values()]\n",
"    data1={\n",
"        '_id':x[0],\n",
"        'who':x[1],\n",
"        'phoneno':x[2],\n",
"        'query':x[3]\n",
"    }\n",
"    query={'_id':{'$eq':data1['_id']}}\n",
"    docs=my_database_query.get_query_result(query)\n",
```

```
"    if(len(docs.all())==0):\n",
"        url=my_database_query.create_document(data1)\n",
"        my_database_query = client['my_database_query']\n",
"        dt=[]\n",
"        for document in my_database_query:\n",
"            dt.append(document['who'])\n",
"            dt.append(document['phoneno'])\n",
"            dt.append(document['query'])\n",
"        return render_template('respond.html',predict=\"Response posted Successfully\",data=dt)\n",
"    else:\n",
"        url=my_database_query.create_document(data1)\n",
"        my_database_query = client['my_database_query']\n",
"        dt=[]\n",
"        for document in my_database_query:\n",
"            dt.append(document['who'])\n",
"            dt.append(document['phoneno'])\n",
"            dt.append(document['query'])\n",
"        return render_template('respond.html',predict=\"Response posted Successfully\",data=dt)\n",
"\n",
"\n",
"\n",
"@app.route('/fp')\n",
"def fp():\n",
"    full_filename = os.path.join(app.config['UPLOAD_FOLDER'],'forgotpw.png')\n",
"    return render_template('fp.html',image=full_filename)\n",
"\n",
"@app.route('/afterfp', methods=['POST','GET'])\n",
```

```
"def afterfp():  \n",
"    pn=request.form['phoneno']\n",
"    securityques=request.form['secques']\n",
"    npassw=request.form['npwd']\n",
"    cpassw=request.form['cpwd']\n",
"    \n",
"    \n",
"    docs=my_database[pn]\n",
"    \n",
"    \n",
"    if(npassw==cpassw and securityques==docs['securityquestion']):\n",
"        full_filename = os.path.join(app.config['UPLOAD_FOLDER'],'loginimg.jpg')\n",
"        docs['pwd'] = cpassw\n",
"        docs.save() \n",
"        return render_template('login.html',predict=\"Successfully updated\",image=full_filename)\n",
"    if(securityques!=docs['securityquestion']):\n",
"        full_filename = os.path.join(app.config['UPLOAD_FOLDER'],'forgotpw.png')\n",
"        return render_template('fp.html',predict=\"Incorrect answer to security question\",image=full_filename)\n",
"    if(npassw!=cpassw):\n",
"        full_filename = os.path.join(app.config['UPLOAD_FOLDER'],'forgotpw.png')\n",
"        return render_template('fp.html',predict=\"New and confirm password does not match\",image=full_filename)\n",
"    \n",
"    \n",
"\n",
"@app.route('/prediction')\n",
"def prediction():\n",
```

```
"    full_filename = os.path.join(app.config['UPLOAD_FOLDER'], 'retina.jpg')\n",
"    full_filename1 = os.path.join(app.config['UPLOAD_FOLDER'], 'image6.png')\n",
"    return render_template('prediction.html',image=full_filename,image2=full_filename1)\n",
"\n",
"@app.route('/afterpred',methods=[\"GET\",\"POST\"])\n",
"def aftepred():\n",
"    if request.method==\"POST\":\n",
"        full_filename2 = os.path.join(app.config['UPLOAD_FOLDER'], 'retina.jpg')\n",
"        full_filename1 = os.path.join(app.config['UPLOAD_FOLDER'], 'image6.png')\n",
"        f=request.files['pfile']\n",
"        print(\"yes\")\n",
"        filepath=os.path.join('static','uploads',f.filename)\n",
"        f.save(filepath)\n",
"        print(\"saved\")\n",
"        img=image.load_img(filepath,target_size=(224,224))\n",
"        x=image.img_to_array(img)\n",
"        x=np.expand_dims(x,axis=0)\n",
"        img_data=preprocess_input(x)\n",
"        prediction=np.argmax(model.predict(img_data),axis=1)\n",
"        index=[\"no dr\",\"mild dr\",\"moderate dr\",\"severe dr\",\"proliferate\"]\n",
"        result=str(index[prediction[0]])\n",
"        print(result)\n",
"        return render_template('prediction.html',prediction=result,image=full_filename2,image2=full_filename1)\n",
"    else:\n",
"        full_filename = os.path.join(app.config['UPLOAD_FOLDER'], 'loginimg.jpg')\n",
"        return render_template('login.html',pred=\"Please login using your credentials\",image=full_filename)\n",
```

```
"\n",
"\n",
"@app.route('/query')\n",
"def query():\n",
"    my_database_query = client['my_database_query']\n",
"    \n",
"    dt=[]\n",
"    for document in my_database_query:\n",
"        dt.append(document['who'])\n",
"        dt.append(document['phoneno'])\n",
"        dt.append(document['query'])\n",
"    return render_template('query1.html',data=dt)\n",
"\n",
"@app.route('/afterquery', methods=['POST','GET'])\n",
"def afterquery():  \n",
"    my_database_query = client['my_database_query']\n",
"    x=[x for x in request.form.values()]\n",
"    data1={\n",
"        '_id':x[0],\n",
"        'who':x[1],\n",
"        'phoneno':x[2],\n",
"        'query':x[3],\n",
"    }\n",
"    query={'_id':{'$eq':data1['_id']}}\n",
"    docs=my_database_query.get_query_result(query)\n",
"    if(len(docs.all())==0):\n",
"        url=my_database_query.create_document(data1)\n",
"        my_database_query = client['my_database_query']\n",
```

```
"       dt=[]\n",
"       for document in my_database_query:\n",
"           dt.append(document['who'])\n",
"           dt.append(document['phoneno'])\n",
"           dt.append(document['query'])\n",
"       return render_template('query1.html',predict=\"Query submitted Successfully\",data=dt)\n",
"   else:\n",
"       url=my_database_query.create_document(data1)\n",
"       my_database_query = client['my_database_query']\n",
"       dt=[]\n",
"       for document in my_database_query:\n",
"           dt.append(document['who'])\n",
"           dt.append(document['phoneno'])\n",
"           dt.append(document['query'])\n",
"       return render_template('query1.html',predict=\"Query submitted Successfully\",data=dt)\n",
"\n",
"@app.route('/admin')\n",
"def admin():\n",
"   full_filename2 = os.path.join(app.config['UPLOAD_FOLDER'], 'adminimg.png')\n",
"   return render_template('admin.html',image=full_filename2)\n",
"\n",
"@app.route('/locsugges')\n",
"def locsugess():\n",
"   print(\"location\")\n",
"   dbl=client['db1']\n",
"   dbu=client['db']\n",
"   print(\"good\")\n",
```

```
"    pn=session['pn']\n",
"    doc1=dbu[pn]\n",
"    location=doc1[\"location\"]\n",
"    print(\"new\")\n",
"    docs=dbl[location.upper()]\n",
"    hospital=[]\n",
"    location=[]\n",
"    for h in range(len(docs[\"hospitals\"])):\n",
"        hospital.append(docs['hospitals'][h])\n",
"    for l in range(len(docs[\"locations\"])):\n",
"        location.append(docs['locations'][l])\n",
"    full_filename = os.path.join(app.config['UPLOAD_FOLDER'])\n",
"    return render_template('locsugges.html',hospital=hospital,location=location)\n",
"@app.route('/uploc')\n",
"def uploc():\n",
"    full_filename = os.path.join(app.config['UPLOAD_FOLDER'])\n",
"    return render_template('uploc.html')\n",
"\n",
"@app.route('/afteruploc',methods=[\"GET\",\"POST\"])\n",
"def afteruploc():\n",
"    loc=request.form['loch']\n",
"    hname=request.form['hname']\n",
"    lname=request.form['lname']\n",
"    contact=request.form['contact']\n",
"    dbl=client['db1']\n",
"    docs=dbl[loc.upper()]\n",
"    hn=hname.strip().upper()\n",
"    count=0\n",
```

```
"    for i in range(len(docs['hospitals'])):\n",
"      if docs['hospitals'][i].strip().upper()==hn:\n",
"        docs['contacts'][i]=contact\n",
"        docs['locations'][i]=lname\n",
"        count=1\n",
"        docs.save()\n",
"        break\n",
"    if(count==0):\n",
"      docs['hospitals'].append(hn)\n",
"      docs['contacts'].append(contact)\n",
"      docs.save()\n",
"    full_filename = os.path.join(app.config['UPLOAD_FOLDER'])\n",
"    return render_template('uploc.html',predict=\"Updated or added successfully\")\n",
"\n",
"\n",
"@app.route('/modify',methods=[\"GET\",\"POST\"])\n",
"def modify():\n",
"    full_filename = os.path.join(app.config['UPLOAD_FOLDER'])\n",
"    return render_template('modify.html',predict=\"Please enter your new location\")\n",
"\n",
"@app.route('/aftermodify',methods=[\"GET\",\"POST\"])\n",
"def aftermodify():\n",
"    if request.method==\"POST\":\n",
"      user=session['pn']\n",
"      pwd=request.form['pwd']\n",
"      nloc=request.form['location']\n",
"      docs=my_database[user]\n",
"      if (pwd==docs['pwd']):\n",
```

```
        "            full_filename = os.path.join(app.config['UPLOAD_FOLDER'],'pimg.jpg')\n",
        "            docs['location'] = nloc\n",
        "            docs.save()\n",
        "            return render_template('modify.html',predict=\"Successfully updated\")\n",
        "        else:\n",
        "           full_filename = os.path.join(app.config['UPLOAD_FOLDER'], 'retina.jpg')\n",
        "           full_filename1 = os.path.join(app.config['UPLOAD_FOLDER'], 'image6.png')\n",
        "           return
render_template('prediction.html',image=full_filename,image2=full_filename1)\n",
        "\n",
        "\n",
        "\n",
        "@app.route('/logout')\n",
        "def logout():\n",
        "    session.pop('pn', None)\n",
        "    return render_template('logout.html',)\n",
        "\n",
        "if __name__==\"__main__\":\n",
        "    app.run(debug=False)"
      ],
      "metadata": {
        "id": "-2iqkQSuFsbd"
      },
      "execution_count": null,
      "outputs": []
    }
  ]
}
```

Team leader :
Sudharsanan K

Members :

1. Hari Nitheesh
2. Siva priya
3. Sanjeevy moorthi
4. Ramesh