Automatic Classification Of The Preliminary Diabetic Retinopathy Stages Software Requirement Specification Document

Mahmoud Hazem, Mohamed Alaa, Omar Khaled, Youssef Talaat.
Supervised By: Dr. Alaa Hamdy and Eng. Youmna Ibrahim.

February 2, 2020

Contents

1 Introduction

1.1 Purpose

The purpose of this Software Requirements Specification document is to thoroughly study and examine all the different ideas and concepts that make up our system and vividly demonstrate its requirements, with respect to our potential market and customers. In this document, we shall also figure out how this software will be used so as to attain a better understanding of the whole system, define other approaches and methods that we may develop along the way as the project progresses, and finally archive all the options that are being considered, but may be discarded as the project advances. The system's main purpose is to automatically detect and classify the early stages of Diabetic Retinopathy, before it evolves and leads to more severe complications, such as total blindness and/or Glaucoma. The Project will consist of a Web application, in which doctors can access the website, Log-In if they have already created an account on the system, and start diagnosing the patients, by inserting their retinal fundus images into the system, and receiving the result and class of the disease.

1.2 Scope

The scope of this document is related to and concerns all Diabetic patients, which make up huge numbers around the whole world and specifically in the MENA (Middle East and North Africa) region. The System is a Web application, that can be accessed by many doctors everywhere, and each doctor has got their own account on the system, where all their previous patients statuses and cases have been stored. The system focuses on automatically detecting the early stages of Diabetic Retinopathy, classifying the level of the disease in the patient's body, and telling the patient whether he/she are in need of any serious medical attention or not. This SRS is aimed at specifying and analyzing all the requirements of the software to be developed later on.

1.3 Overview

The main goal of this software is to automatically detect the early stages of Diabetic Retinopathy and classify the level of the disease in the patient's body. Our aim in this project is to help as many Diabetic patients as possible, by preventing Diabetes from affecting their eyesight and progressing to Diabetic Retinopathy. The idea of the system, after thoroughly reading about Diabetes and Eye diseases, was that Diabetic Retinopathy doesn't show any symptoms, until a very late stage in life. The System will take retinal fundus images as input from the diseased patient and apply some data preprocessing techniques. Using a deep learning approach, the system will then detect whether a person suffers from Diabetic Retinopathy or not; based on the answer, the system will then classify the level of the disease and finally propose a solution to the patient.

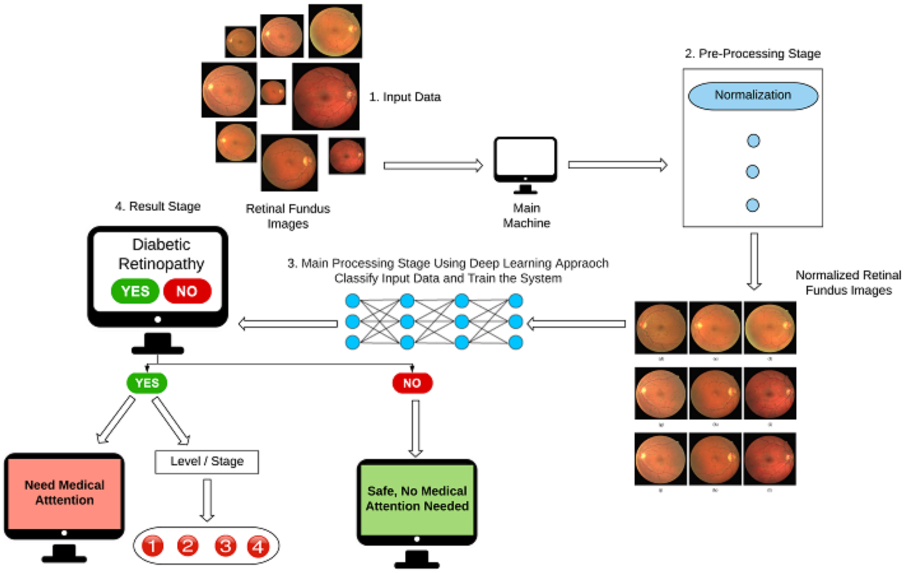


Figure 1: System Overview

- First, The Main Computer/Machine will collect the information from the Input Images, which are unnormalized Retinal Fundus Images with different sizes chosen from our dataset. The System will then apply some data pre-processing algorithms such as Normalization so that all the images are the same size and dimension. This will thoroughly help us in the main processing phase by simply reducing the complexity of the Input images. Then comes the Main Processing stage in which we test and train our system, We use a Convolutional Neural Network in which a group of connected nodes distributed on multiple layers enhance and strengthen each other along with the Tensor-flow library for feature extraction and classification of the Input images. The System will then proceed to the final stage, which is the Result stage; If the result turned out to be (YES), the System will show the Level/Stage of the disease on a scale of 4-stages, and finally propose that the patient needs medical attention right away. Else (NO) the System will propose that there is no need for medical attention.

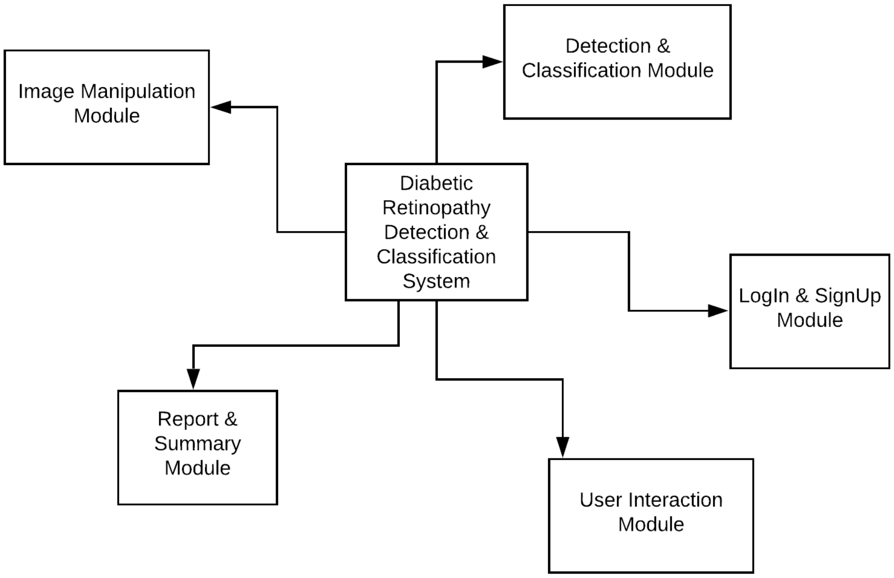


Image Manipulation Module
Detection & Classification Module
Diabetic Retinopathy Detection & Classification System
LogIn & SignUp Module
Report & Summary Module
User Interaction Module

Figure 2: Block Diagram

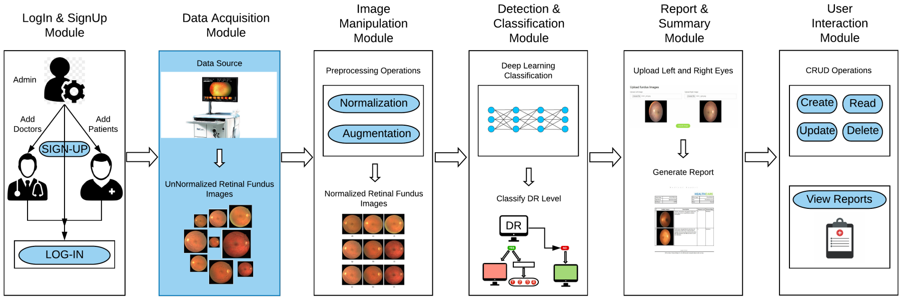


Figure 3: Extended Block Diagram

1. **Log-In & Sign-Up Module :** Module responsible for the Log-In and Sign-Up operations, conducted by the Admin, Doctor, and Patient.

2. **User Interaction Module :** Module responsible for all the operations conducted by the Doctors, Patients, and the admin of the system, such as Add, Edit, Search, Delete and View/List.

3. **Image Manipulation Module :** Module responsible for handling and manipulating all the Retinal Fundus Images that are uploaded to the sys-

tem by the Doctors, such as Image Normalization.

4. **Report & Summary Module :** Module responsible for generating and handling the final full status report by the Doctor.

5. **Detection & Classification Module :** Module responsible for detecting the presence of Diabetic Retinopathy in the patient's body, and classifying the Retinal Fundus Images on a scale of 4 stages (No DR), (PDR), (NPDR) which is further divided into (Mild, Moderate and Severe).

1.4 Business Context

Diabetes Mellitus is one of the world's most famous and dominant diseases that strike many people at a very early stage in life. This disease later on leads to another completely unforeseen chronic eye disease called Diabetic Retinopathy (DR). Diabetic Retinopathy is a retinal disease that usually leads to or can naturally progress to irreversible vision loss (blindness) and/or Glaucoma. Around 147 million people worldwide are expected to be diagnosed with (DR). In collaboration with professor Dina Hossam (Assistant Professor Of Ophthalmology in Cairo University), she has agreed to assist us in this project and to provide us with all the sufficient information and data needed to complete this project. She considered Diabetic Retinopathy to be one of the most prevalent, preventable eye diseases in Egypt and the Middle East, and that succeeding in setting a comprehensive screening and management program to detect (DR), would have an extremely positive impact on the rates of blindness among the Egyptian population, and other countries as well. Our aim in this project is to be able to bring this visualization to life, by creating an automatic system that detects and classifies the early stages of Diabetic Retinopathy and proposes a solution to the patient. Thus, decreasing the rates of blindness and other eye diseases among the Egyptian Population as well as other countries.

2 General Description

2.1 Product Functions

- Log-In / Sign-Up Module :
  1- Admin can create a new account for both the Doctor and the Patient on the system, by providing a username and a password.
  2- Admin / Doctor / Patient can Log into the system by using his/her username and password.
  3- The Post Log-In page will display different features and pages based on the user type.

- User Interaction Module (Doctor) :
  1- Doctor can Search for a specific patient record within the system.

- User Interaction Module (Admin) :
  1- Admin can Add new Doctors and Patients to the system.
  2- Admin can Edit the Doctor's and Patient's personal information.
  3- Admin can Remove Doctors and Patients from the system's Database.
  4- Admin can search/view Doctors and Patients in the system's Database.

- User Interaction Module (Patient) :
  1- Patient can view his/her previous reports.

- Image Manipulation Module:
  1- Doctor can Upload multiple Retinal Fundus Images of the patient to the system.

- Detection and Classification Module :
  1- Doctor can Detect the presence of Diabetic Retinopathy in the patient's body.
  2- Doctor can classify the level of the disease on a scale of 4 stages.

- Report and Summary Module :
  1- Doctor can add his/her feedback and diagnosis of the case based on the classification result.
  2- Doctor can generate a final report, including the patient's status (Class) and the doctor's feedback on the case, and store it in the system's Database.

2.2 Similar System Information

Carson et al. [3] had Similar System, The project's main goal and purpose is to implement an automatic DR grading system capable of classifying retinal fundus images, using pre-trained CNN models i.e. AlexNet and Inception (GoogleNet). The System performs a data Pre-Processing stage, such as input images normalization and uses both the CNN and the N-ary tree classifying algorithms. The System uses two datasets for training, the Messidor and EyePacs datasets and has 4 stages. However, The System has very low accuracy levels, which will be solved in our project. However, there are some common grounds between both systems. Both projects use a deep learning approach for classification of different Diabetic Retinopathy levels. And both projects have worked with the same two datasets.

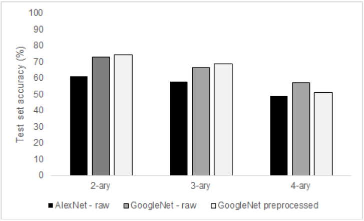
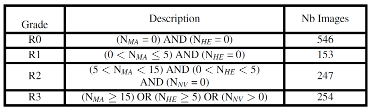
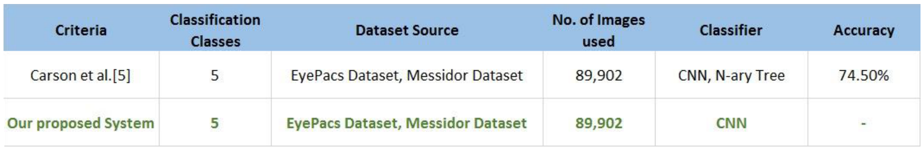


Figure 4: Test set Accuracies using different Learning Models.



Grade	Description	Nb Images
R0	$(N_{MA} = 0)$ AND $(N_{HE} = 0)$	546
R1	$(0 < N_{MA} \leq 5)$ AND $(N_{HE} = 0)$	153
R2	$(5 < N_{MA} < 15)$ AND $(0 < N_{HE} < 5)$ AND $(N_{NV} = 0)$	247
R3	$(N_{MA} \geq 15)$ OR $(N_{HE} \geq 5)$ OR $(N_{NV} > 0)$	254

Figure 5: Number Of each Class using the MESSIDOR Dataset.



Criteria	Classification Classes	Dataset Source	No. of Images used	Classifier	Accuracy
Carson et al.[5]	5	EyePacs Dataset, Messidor Dataset	89,902	CNN, N-ary Tree	74.50%
Our proposed System	5	EyePacs Dataset, Messidor Dataset	89,902	CNN	-

Figure 6: Comparison Table.

2.3 User Characteristics

- Doctor :
  1- Must have the basic knowledge of dealing with a computer (PC/Laptop).
- Admin :
  1- Must have the basic knowledge of dealing with a computer (PC/Laptop).
  2- Must understand how to deal with and manage a database system.
- Patient :
  1- Must have the basic knowledge of dealing with a computer (PC/Laptop).

2.4 User Problem Statement

Automatically Detecting the presence of Diabetic Retinopathy in the patient's body, classifying the level of the disease on a scale of 4 stages, and proposing a solution to the user/patient.

2.5 User Objectives

Using our automatic system, diabetic patients along with their doctors, will be able to know if the patient has any signs of Diabetic Retinopathy, and what stage is the disease currently in. Retinal Fundus Images detection accuracy will have to be very high, so as to be able to correctly classify the disease level, and propose the most suitable solution to the patient. The system's detection speed will also have to be increased, for it to be usable by as many patients and doctors as possible.

2.6 General Constraints

1- The System is mainly designed for doctors to use in detecting and classifying Diabetic Retinopathy disease only.

2- Doctor's Desktop / Laptop must be connected to the internet.

3 Functional Requirements

SRS/3.1	
Function ID	FR1.
Name	Login.
Description	Doctors and Admin must Login into the system to start using it's main functions, using their usernames and passwords.
Pre-Condition	No access to the system's Functions and Data.
Input	Username, Password.
Output	Login Successful, Proceed to Post-Login page. / Login Failed, Try Again.
Post-Condition	Login Successful, Proceed to Post-Login page and grant access to the system's Functions and Data.
Action	Function takes the Username and Password, and checks whether the account is valid or not.
Dependency	AddDoctor (FR2), AddPatient (FR6).

SRS/3.2	
Function ID	FR2.
Name	AddUser.
Description	Admin can add new Users to the system, by creating new accounts for them.
Pre-Condition	No User account, User can't access the system.
Input	User's Personal and Business Information.
Output	User's Information added to the system.
Post-Condition	User's account created successfully.
Action	Function takes all the needed information about the User, and creates a new User account in the system.
Dependency	Login (FR1).

SRS/3.3	
Function ID	FR3.
Name	SearchUser.
Description	Admin can search for existing Users' records within the system.
Pre-Condition	No specific User account or data searched for.
Input	User's Name or ID.
Output	User's Information/Record.
Post-Condition	Retrieve and Display all the User's Information.
Action	Function takes the User's Name or ID stored within the system, searches through all the User' records and finally returns the specific record to the admin.
Dependency	Login (FR1).

SRS/3.4	
Function ID	FR4.
Name	ListAllUsers.
Description	Admin can get a whole list of all the Users available within the system.
Pre-Condition	No List of all User shown.
Input	No Input needed.
Output	A List of all the Users within the system.
Post-Condition	Retrieve and Display all the Users available.
Action	Function retrieves all the Users' records available within the system, and displays all of them to the Admin.
Dependency	Login (FR1).

SRS/3.5	
Function ID	FR5.
Name	EditUserInfo.
Description	Admin can edit a specific User's record, after searching for the User using his Name or ID.
Pre-Condition	User's record with it's Old data.
Input	User's Record.
Output	Updated User's Record.
Post-Condition	User's Record has been successfully updated with the New data.
Action	Function takes the ID of the User's record and updates the database with the New changed data of the record.
Dependency	SearchUser (FR3).

SRS/3.6	
Function ID	FR6.
Name	DeleteUser.
Description	Admin can remove/delete a specific User's record permanently from the system, after searching for the User using his/her Name or ID .
Pre-Condition	User's record found and running in the database.
Input	User's Record.
Output	Deleted User's Record.
Post-Condition	User's record has been successfully removed/deleted from the system's database, and can no longer be found or searched for.
Action	Function takes the ID of the User's record and removes/deletes the record permanently from the database.
Dependency	SearchDoctor (FR3).

SRS/3.7	
Function ID	FR7.
Name	UploadImage.
Description	Doctor can upload Retinal Fundus Images of the patient to the system, before proceeding to the detection and classification stage.
Pre-Condition	No Image available for detection and classification.
Input	Image ID and Path.
Output	Uploaded Retinal Fundus Image.
Post-Condition	Patient's Retinal Fundus Images are uploaded to the system's database and can now be used in the next stage, which is the detection and classification stage.
Action	Function takes the ID and Path of the Patient's Retinal Fundus Image and stores it into the system's database for further detection and classification procedures.
Dependency	AddPatient (FR7).

SRS/3.8	
Function ID	FR8.
Name	NormalizeImage.
Description	Retinal Fundus Images are normalized before they proceed to the detection and classification stage.
Pre-Condition	Unnormalized Retinal Fundus Images of different sizes and dimensions.
Input	Image ID and Path.
Output	Normalized Retinal Fundus Images of same sizes and dimensions.
Post-Condition	Patient's Retinal Fundus Images are now normalized, and they all have the same dimensions of 224p x 224p.
Action	Function takes the ID and Path of the Patient's Retinal Fundus Images and normalizes the Images into 224 pixels by 224 pixels, and finally produces the output images for the detection and classification stage.
Dependency	UploadImage (FR12).

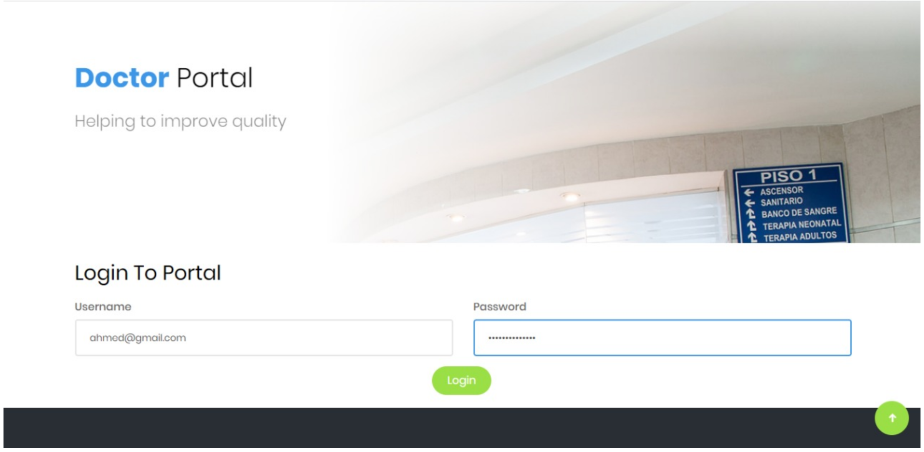
SRS/3.9	
Function ID	FR9.
Name	ClassifyDisease.
Description	Retinal Fundus Images proceed to the next process, where detection and classification happens.
Pre-Condition	Disease undetected, Disease level unclassified.
Input	Retinal Fundus Images.
Output	Detection result, Disease Level on a scale of 4 stages.
Post-Condition	Patient's Result has been calculated, and the Level of the Disease has been classified.
Action	Function takes the Patient's Retinal Fundus Images, and detects whether the person suffers from (DR) or not, and then classifies the Level of (DR) on a scale of 4 stages.
Dependency	UploadImage (FR12).

SRS/3.10	
Function ID	FR10.
Name	GenerateReport.
Description	Doctor Generates a Final Full-status report about the patient.
Pre-Condition	Patient's Case/Status is not stored for later use.
Input	Doctor's Feedback, Patient's Classification Result, Patient's Retinal Fundus Images.
Output	Full Status Report about the Patient.
Post-Condition	Doctor owns a Full Status report about the patient's case.
Action	Function takes the Patient's Classification Result, Retinal Fundus Images, and the Doctor's Feedback and generates a Final Full Status report about the patient's case. The Report can be printed out as a PDF file for the patient and for later use.
Dependency	ClassifyDisease (FR14).

4 Interface Requirements

4.1 User Interfaces

4.1.1 Doctor / Patient - Log-In Portal



Doctor Portal

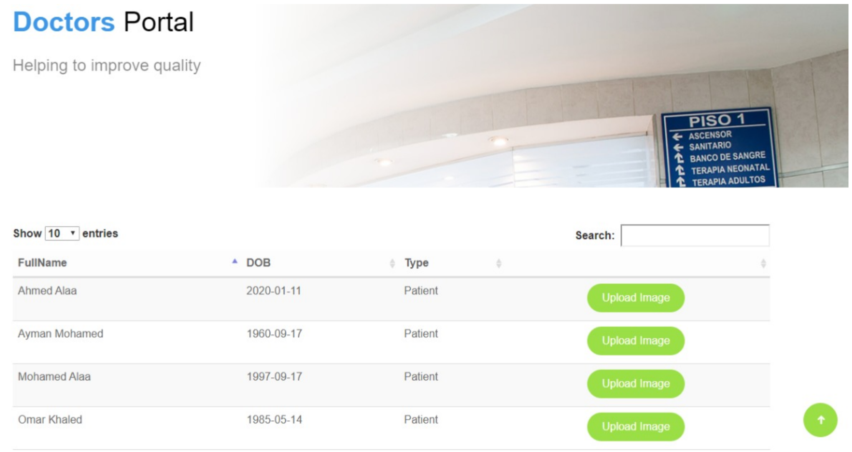
Helping to improve quality

Login To Portal

Username: ahmed@gmail.com

Password: ··············

4.1.2 Doctor - Main Homepage



Doctors Portal

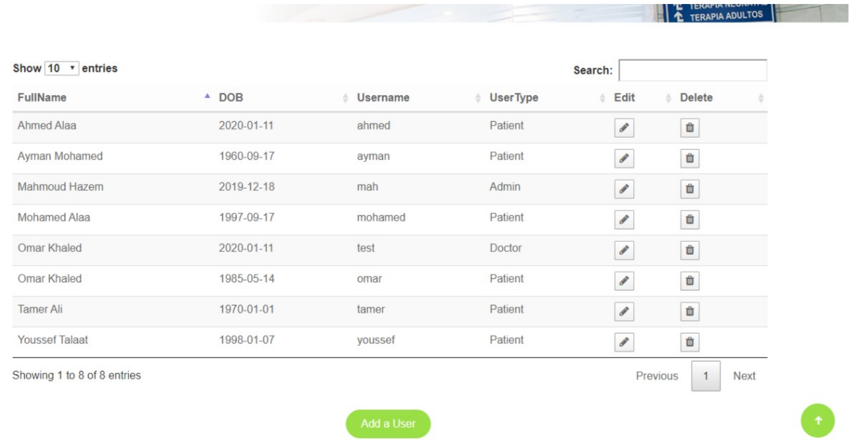
Helping to improve quality

Show 10 entries

Search:

FullName	DOB	Type	
Ahmed Alaa	2020-01-11	Patient	Upload Image
Ayman Mohamed	1960-09-17	Patient	Upload Image
Mohamed Alaa	1997-09-17	Patient	Upload Image
Omar Khaled	1985-05-14	Patient	Upload Image

4.1.3 Admin - Main Homepage



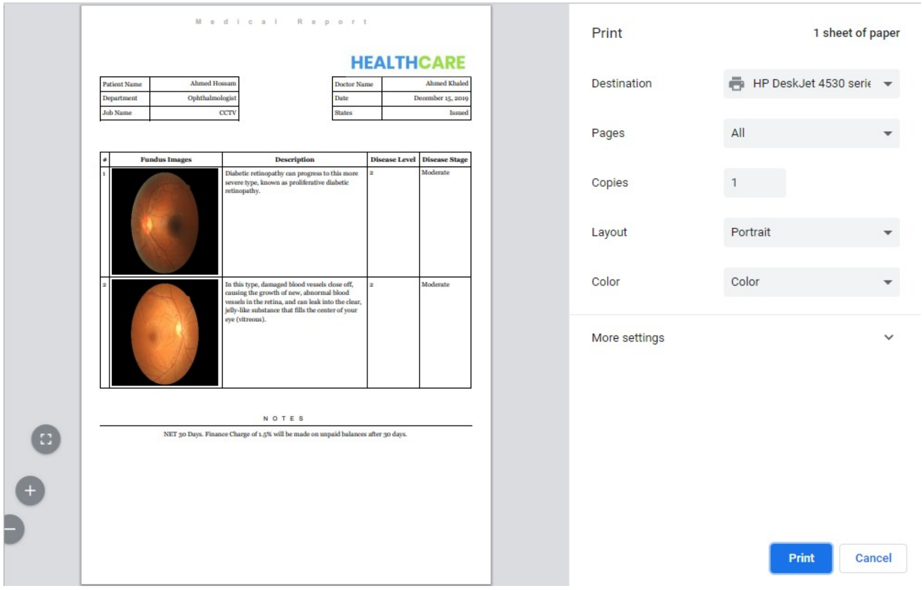
Show 10 entries

Search:

FullName	DOB	Username	UserType	Edit	Delete
Ahmed Alaa	2020-01-11	ahmed	Patient	✎	🗑
Ayman Mohamed	1960-09-17	ayman	Patient	✎	🗑
Mahmoud Hazem	2019-12-18	mah	Admin	✎	🗑
Mohamed Alaa	1997-09-17	mohamed	Patient	✎	🗑
Omar Khaled	2020-01-11	test	Doctor	✎	🗑
Omar Khaled	1985-05-14	omar	Patient	✎	🗑
Tamer Ali	1970-01-01	tamer	Patient	✎	🗑
Youssef Talaat	1998-01-07	youssef	Patient	✎	🗑

Showing 1 to 8 of 8 entries

4.1.4 Doctor - Generating Report



Medical Report

Patient Name	Ahmed Hossam
Department	Ophthalmologist
Job Name	CCTV

Doctor Name	Ahmed Khaled
Date	December 15, 2019
States	Issued

#	Fundus Images	Description	Disease Level	Disease Stage
1		Diabetic retinopathy can progress to this more severe type, known as proliferative diabetic retinopathy.	2	Moderate
2		In this type, damaged blood vessels close off, causing the growth of new, abnormal blood vessels in the retina, and can leak into the clear, jelly-like substance that fills the center of your eye (vitreous).	2	Moderate

NOTES

NET 30 Days. Finance Charge of 1.5% will be made on unpaid balances after 30 days.

Print 1 sheet of paper

Destination HP DeskJet 4530 serie

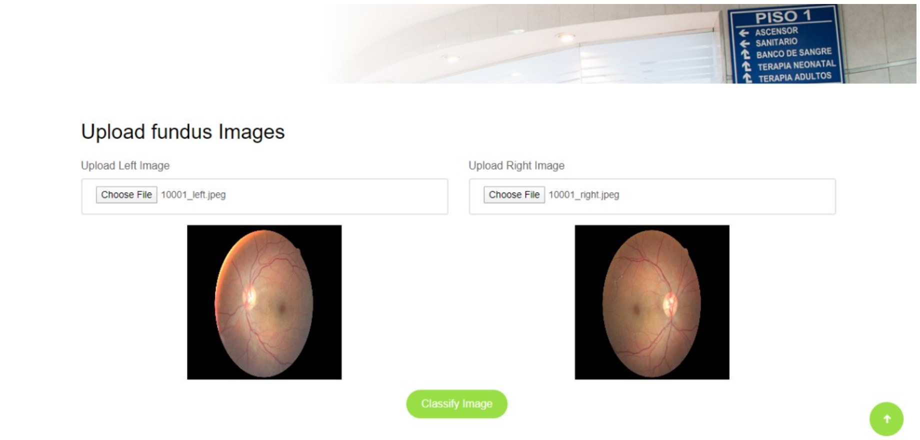
Pages All

Copies 1

Layout Portrait

Color Color

4.1.5 Doctor - Uploading Retinal Fundus Images (L/R)



Upload fundus Images

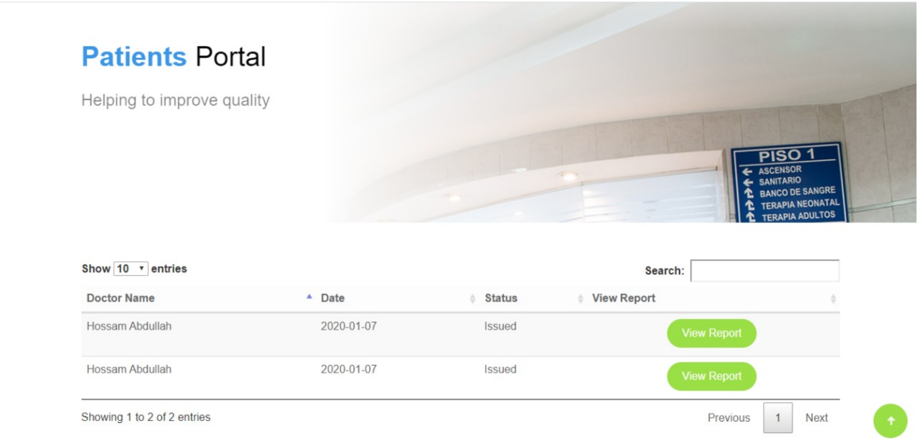
Upload Left Image

Choose File 10001_left.jpeg

Upload Right Image

Choose File 10001_right.jpeg

4.1.6 Patient - Main Homepage



Patients Portal

Helping to improve quality

Show 10 entries

Search:

Doctor Name	Date	Status	View Report
Hossam Abdullah	2020-01-07	Issued	View Report
Hossam Abdullah	2020-01-07	Issued	View Report

Showing 1 to 2 of 2 entries

5 Performance Requirements

1- The Results of diagnosing the Retinal Fundus Images would have to appear to the doctor in a very short amount of time.

2- The Results of processing/reading the Retinal Fundus Images would have to be very accurate and error-free, to further aid in the classification process.

6 Design Constraints

The System must be convenient, Intuitive and easy to use in order for the users to be able to achieve their tasks quickly and easily. Doctors must have a clear and an uncomplicated design that displays all the system's functions and features in a user-friendly way, to ease the process of achieving the required output/results.

6.1 Hardware Limitations

This system needed a powerful computer hardware, for us to be able to train the model on a huge dataset. However, since the system is a Web Application, and the only processing required will be to test images on the pre-trained model, there will be no need for high-end computer hardware devices.

7 Non-Functional Requirements

7.1 Security

1. The Patient's Data and Diagnosis must be protected and preserved since they are sensitive and privileged information, and should not be accessed by anyone other than the assigned Doctor.

2. The Back-end database should never display any sensitive information such as the Doctors' passwords, therefore all passwords must be Encrypted or Hashed using functions such as (Sha1 or Md5).

3. Doctors can't have immediate access to the system just by creating an account on the system. Any new Doctor's Account must first be approved by the system's Admin before gaining access to the system's data and features.

7.2 Reliability

The System must be very accurate and error-free since the result provided by the system determines whether a Patient suffers from Diabetic Retinopathy or not. So it is very important and pivotal that doctors can rely and trust the system in diagnosing the patient, and be assured that the output result is precise. To achieve this requirement, the system will be trained and tested on a very big Dataset, which will increase the accuracy of the system's classification.

7.3 Performance

The System must be very interactive and the delays involved between the processes must be as minimal as possible; So for every Action-Per-Response of the system there shouldn't be any immediate delays. To achieve this requirement, the system should process query calls in less than 1000ms (1 second) when retrieving any information from the database [2].

7.4 Maintainability

1. The System should be easily and smoothly extended and enhanced in the future, whether by adding new features or modifying the existing system functions. To achieve this requirement, the system's code must be written in a way that is adaptable and adjustable; This will be achieved by implementing and applying the (MVC) design strategy, and other design patterns as well, such as Singleton and Facade Design Patterns.

2. The System will be able to modify and upgrade itself by re-training itself on a specific set of Retinal Fundus Images chosen by the doctor, after the Relevance Feedback process is completed.

7.5 Usability

1. The System should have a User-friendly view; the Interface of the System should be easily used and learned by the Doctors, and should be easy to remember.

2. Doctors should not be required to consult a manual or an instruction guide every time they need to use the system.

3. The System should accomplish its goals easily with very few or no user errors.

4. The System must be intuitive and appear easy to use, rather than intimidating and confusing [1].

8 Preliminary Object-Oriented Domain Analysis

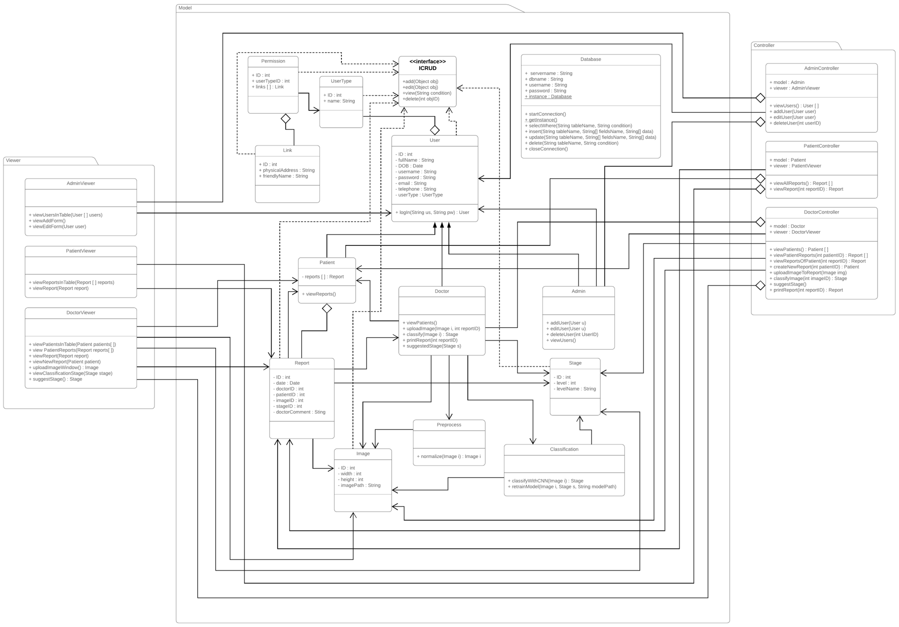
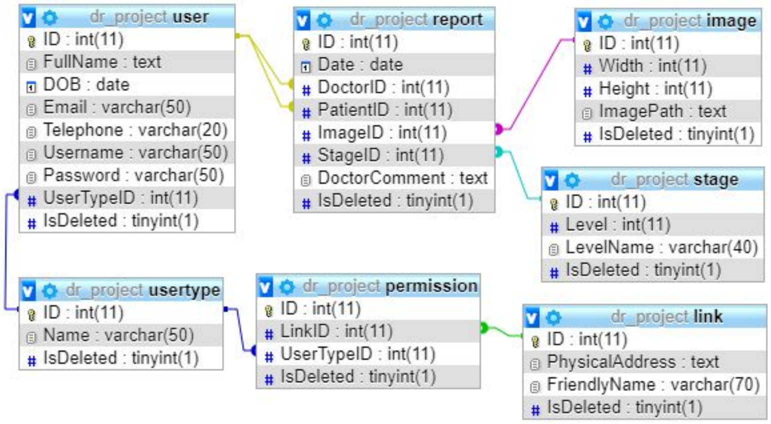


Figure 7: Class Diagram


dr_project user
ID : int(11)
FullName : text
DOB : date
Email : varchar(50)
Telephone : varchar(20)
Username : varchar(50)
Password : varchar(50)
UserTypeID : int(11)
IsDeleted : tinyint(1)

dr_project report
ID : int(11)
Date : date
DoctorID : int(11)
PatientID : int(11)
ImageID : int(11)
StageID : int(11)
DoctorComment : text
IsDeleted : tinyint(1)

dr_project image
ID : int(11)
Width : int(11)
Height : int(11)
ImagePath : text
IsDeleted : tinyint(1)

dr_project stage
ID : int(11)
Level : int(11)
LevelName : varchar(40)
IsDeleted : tinyint(1)

dr_project usertype
ID : int(11)
Name : varchar(50)
IsDeleted : tinyint(1)

dr_project permission
ID : int(11)
LinkID : int(11)
UserTypeID : int(11)
IsDeleted : tinyint(1)

dr_project link
ID : int(11)
PhysicalAddress : text
FriendlyName : varchar(70)
IsDeleted : tinyint(1)

Figure 8: Database Schema (ERD).

8.1 Inheritance Relationships

- This Sections represents the Inheritance Relationships between the most important components that make up the system.

1- **User Class Inheritance**:

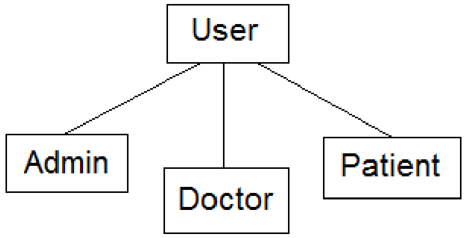


Figure 9: User Inheritance Relationship.

SRS/8.1.1	
Class Name	User
Super Class	None
Sub Class	Admin, Doctor, Patient.
Purpose	The class that stores the common attributes and operations between all users.
Collaborations	This class aggregates UserType class and associates Admin, AdminController and AdminViewer classes, and implements ICRUD interface.
Attributes	ID, Fullname, DOB, username, password, email, telephone and UserType object
Operations	logIn(username, password)

SRS/8.1.2	
Class Name	UserType
Super Class	None
Sub Class	None
Purpose	This class represents the type of the user that will affects which module the user will be allowed to interact with.
Collaborations	This class associates Permission class and aggregated by User class, and implements ICRUD interface.
Attributes	ID, Name
Operations	None

SRS/8.1.3	
Class Name	Link
Super Class	None
Sub Class	None
Purpose	This class represents each page in the web application.
Collaborations	This class is aggregated by Permission class, and implements CRUD interface.
Attributes	ID, physicalAddress and friendlyName
Operations	None

SRS/8.1.4	
Class Name	Permission
Super Class	None
Sub Class	None
Purpose	This class is responsible for allowing each user type to access the allowed links only on the web application.
Collaborations	This class is assisted by UserType class and aggregates Link class, and implements ICRUD interface.
Attributes	ID, userTypeID, ArrayList of Link object.
Operations	None

SRS/8.1.5	
Class Name	Admin
Super Class	User
Sub Class	None
Purpose	This class is responsible for managing all users in the system.
Collaborations	This class assisted by the User class and is aggregated by AdminController class.
Attributes	None
Operations	ICRUD Users.

SRS/8.1.6	
Class Name	Doctor
Super Class	User
Sub Class	None
Purpose	This class represents the Expert that is responsible for carrying out all the diagnosis processes.
Collaborations	This class is assisted by Patient, Image, Preprocess, Classification and Stage classes, and is aggregated by DoctorController class.
Attributes	None
Operations	viewPatients(), uploadImage(Image i, int reportID), classify(Image i), printReport(int reportID), suggestStage(Stage s)

SRS/8.1.7	
Class Name	Patient
Super Class	User
Sub Class	None
Purpose	This class represents each patient in the system.
Collaborations	This class aggregates Report class and associates Doctor, Report, DoctorController and DoctorViewer classes, and is aggregated by PatientController class.
Attributes	ArrayList of Report object
Operations	viewReports()

SRS/8.1.8	
Class Name	Image
Super Class	None
Sub Class	None
Purpose	This class represents the fundus image that will be classified.
Collaborations	This class associates Doctor, Preprocess, Classification, Report, DoctorController and DoctorViewer classes, and implements ICRUD interface.
Attributes	ID, width, hieght, imagePath.
Operations	None

SRS/8.1.9	
Class Name	Preprocess
Super Class	None
Sub Class	None
Purpose	This class contains the operations that will be applied on the fundus images before classification process.
Collaborations	This class is assisted by Image class and associates Doctor class.
Attributes	None
Operations	normalize(Image i)

SRS/8.1.10	
Class Name	Classification
Super Class	None
Sub Class	None
Purpose	This class contains the main classification operation.
Collaborations	This class is assisted by Image, Stage classes and associates Doctor class.
Attributes	None
Operations	classifyWithCNN(Image i), retrainModel(Image i, Stage s, String modelPath)

SRS/8.1.11	
Class Name	Stage
Super Class	None
Sub Class	None
Purpose	This class represents the level and the name of diabetic retinopathy that the image is classified with.
Collaborations	This class is assisted by Doctor, Report, DoctorController and DoctorViewer, and implements ICRUD interface.
Attributes	ID, level, name
Operations	None

SRS/8.1.12	
Class Name	Report
Super Class	None
Sub Class	None
Purpose	This class represents the output of the classification operation.
Collaborations	This class is assisted by Doctor, Patient, Image and Stage classes, and associates DoctorController, PatientController, DoctorViewer and PatientViewer classes, and implements ICRUD interface.
Attributes	ID, date, doctorID, patientID, imageID, stageID, doctorComment
Operations	None

SRS/8.1.13	
Class Name	Database
Super Class	None
Sub Class	None
Purpose	This class facilitates the interaction between the system and the database.
Collaborations	all classes can use its operations since they are all static.
Attributes	servername, dbname, username, password
Operations	startConnection(), getInstance(), selectWhere(String tableName, String condition), insert(String tableName, String[] fieldsName, String[] data), update(String tableName, String[] fieldsName, String[] data), delete(String tableName, String condition) ,closeConnection()

SRS/8.1.14	
Class Name	AdminController
Super Class	None
Sub Class	None
Purpose	This class manages the interaction between the model class (Admin) and the viewer class (AdminViewer).
Collaborations	This class is assisted by User class, and aggregates Admin and AdminViewer classes.
Attributes	None
Operations	viewUsers(), addUser(User user), editUser(User user), deleteUser(int userID)

SRS/8.1.15	
Class Name	DoctorController
Super Class	None
Sub Class	None
Purpose	This class manages the interaction between the model class (Doctor) and the viewer class (DoctorViewer)
Collaborations	This class is assisted by Patient, Report, Image, Stage classes, and aggregates Doctor and DoctorViewer classes.
Attributes	None
Operations	viewPatients() : Patient [ ], viewPatientReports(int patientID) : Report [ ], viewReportsOfPatient(int reportID) : Report, createNewReport(int patientID) : Patient, uploadImageToReport(Image img), classifyImage(int imageID) : Stage, suggestStage(), printReport(int reportID) : Report

SRS/8.1.16	
Class Name	PatientController
Super Class	None
Sub Class	None
Purpose	This class manages the interaction between the model class (Patient) and the viewer class (PatientViewer)
Collaborations	This class is assisted by Report class, and aggregates Patient and PatientViewer classes.
Attributes	None
Operations	viewAllReports() : Report [ ], viewReport(int reportID) : Report

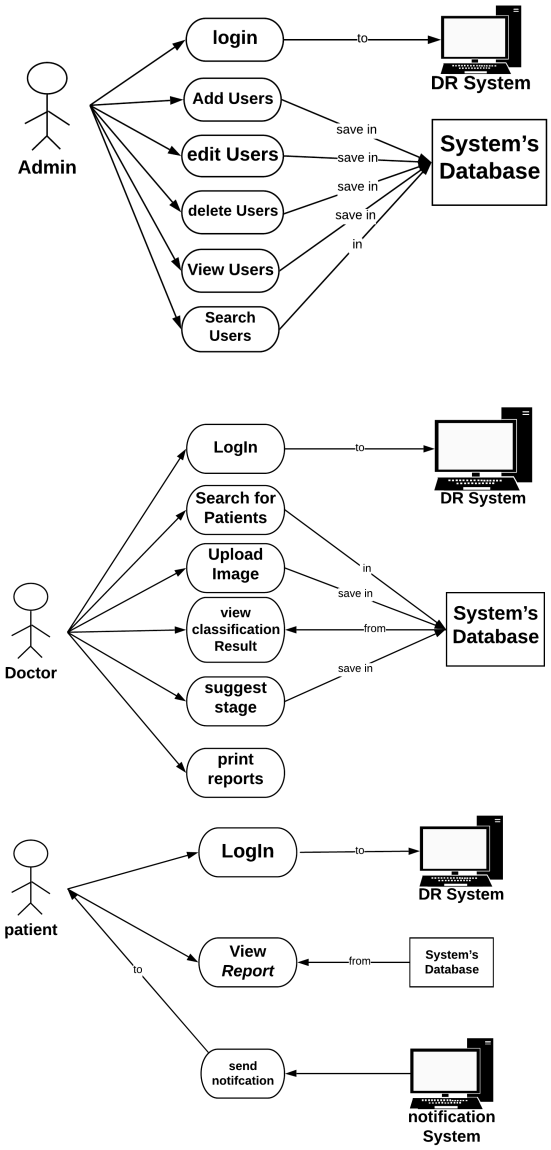
SRS/8.1.17	
Class Name	AdminViewer
Super Class	None
Sub Class	None
Purpose	This class is responsible for displaying views that the Admin needs to view in his module on the web application.
Collaborations	This class is assisted by User class and is aggregated by AdminController class.
Attributes	None
Operations	viewUsersInTable(User [ ] users), viewAddForm(), viewEditForm(User user)

SRS/8.1.18	
Class Name	DoctorViewer
Super Class	None
Sub Class	None
Purpose	This class is responsible for displaying views that the Doctor needs to view in his module on the web application.
Collaborations	This class is assisted by Patient, Report, Image and Stage classes, and is aggregated by DoctorController class.
Attributes	None
Operations	viewPatientsInTable(Patient patients[ ]), view PatientReports(Report reports[ ]), viewReport(Report report), viewNewReport(Patient patient), uploadImageWindow() : Image, viewClassificationStage(Stage stage), suggestStage() : Stage

SRS/8.1.19	
Class Name	PatientViewer
Super Class	None
Sub Class	None
Purpose	This class is responsible for displaying views that the Patient needs to view in his module on the web application.
Collaborations	This class is assisted by Report class and is aggregated by PatientController class.
Attributes	None
Operations	viewReportsInTable(Report [ ] reports), viewReport(Report report)

SRS/8.1.20	
Class Name	ICRUD
Type	Interface
Super Class	None
Sub Class	None
Purpose	This interface allows the classes to perform the main operations(add, edit, view, delete).
Collaborations	User, UserType, Link, Permission, Image, Stage and Report classes implements this class
Attributes	None
Operations	add(Object obj), edit(Object obj), view(String condition), delete(int objID)

9 Operational Scenarios



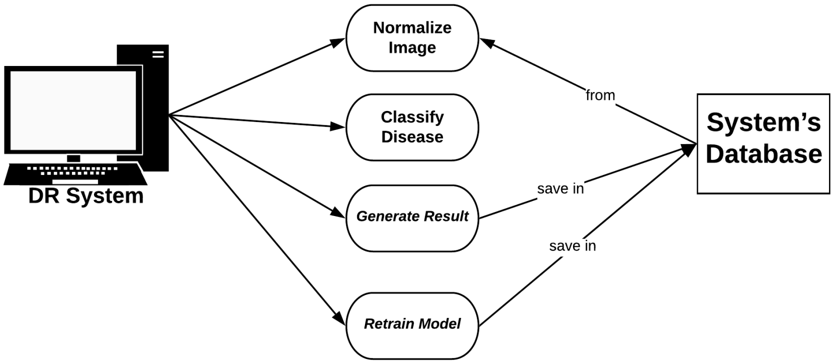


Figure 10: DR System Usecase diagram

9.1 Scenarios

9.1.1 Scenario (1): Admin

The Admin controls all the Users within the system as by :
- Users (Doctors / Patients)

1. Adding Users.
2. Editing Users.
3. Deleting Users.
4. Searching/Viewing Users.

9.1.2 Scenario (2): Doctor

The Doctor starts by logging in into the system, then he/she is supposed to choose a patient to upload their Retinal Fundus Images to be diagnosed. The system then classifies the case and shows the output results to the Doctor. Moreover, the Doctor can finally provide his/her feedback on the output results, which will then be used to retrain the model.

9.1.3 Scenario (3): Patient

The Patient starts by logging in into the system, then he/she would have the option of viewing all the previous reports associated with them.

9.1.4 Scenario (4): Diabetic Retinopathy Detection and Classification

After uploading the Retinal Fundus Images, the system will perform some procedures and methods on these images. First, the system will Normalize the images, to make sure that all the images have the same size and dimensions. Afterwards, the system will detect and classify the Diabetic Retinopathy stage according to the pre-trained model. Finally, the system will generate the results of this classification, and store them in the system's database.

10 Preliminary Schedule Adjusted

Task	Date	Deadline
Proposal Ideas	1 July 2019	15 July 2019
Announce Proposal For Students	16 July 2019	22 July 2019
Proposal Evaluation	First Week Of October	9 October 2019
SRS Evaluation	Second Week Of January	16 January 2020
SDD Evaluation	Third Week Of February 2020	-
Prototype Evaluation	3 Days After Midterm Exam	-
Deliver Contribution Paper	Second Semester	-
Technical Evaluation	First Week Of May 2020	-
Final Thesis	Last 10 Days in June 2020	-
Ceremony	24 June 2020	-

.

11 Preliminary Budget Adjusted

According to SherWeb the average monthly cost of a cloud server is $313.90. Which will be essential for us to host our Web Application.

12 Appendices

12.1 Definitions, Acronyms, Abbreviations

1- MENA: Middle East and North Africa.

2- DR: Diabetic Retinopathy.

3- SRS: Software Requirements Specification.

3- CNN: Convolutional Neural Network.

4- FR: Functional Requirement.

5- PDR: Proliferative Diabetic Retinopathy.

6- NPDR: Non-Proliferative Diabetic Retinopathy.

7- MVC: Model - View - Controller.

13 References

References

[1] "Usability first - about usability - requirements specification — usability first," 2015. [Online]. Available: http://www.usabilityfirst.com/about-usability/requirements-specification/

[2] "Non-functional requirements," 2018. [Online]. Available: https://nhsconnect.github.io/gpconnect/development$_{a}pi_{n}on_{f}unctional_{r}equirements.html$

[3] D. Y. Carson Lam, M. Guo, and T. Lindsey, "Automated detection of diabetic retinopathy using deep learning," *AMIA Summits on Translational Science Proceedings*, vol. 2018, p. 147, 2018.