**BLINDNESS DETECTION USING DIABETIC RETINOPATHY**

**A**

**MAJOR PROJECT REPORT**

Submitted by

|  |  |  |
| --- | --- | --- |
| **Aayush A. Raina**  **00214802717** | **Aman Thukral**  **00714802717** | **Harkirat Singh**  **02514802717** |

**BACHELOR OF TECHNOLOGY**

IN

**COMPUTER SCIENCE & ENGINEERING**

Under the Guidance

of

|  |  |
| --- | --- |
| **Ms. Garima Gupta**  **Assistant Professor, CSE** | **Dr. Neeraj Garg**  **Associate Professor, CSE** |

****

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING MAHARAJA AGRASEN INSTITUTE OF TECHNOLOGY (AFFILIATED TO GURU GOBIND SINGH INDRAPRASTHA UNIVERSITY, DELHI)**

**(**JUNE 2021**)**

# **MAHARAJA AGRASEN INSTITUTE OF TECHNOLOGY**

# **Department of Computer Science and Engineering**

****

**CERTIFICATE**

This is to Certify that this Major project report “BLINDNESS DETECTION THROUGH DIABETIC RETINOPATHY” is submitted by “Aayush A. Raina(00214802717); Aman Thukral(00714802717); Harkirat Singh(02514802717)” who carried out the project work under my supervision.

I approve this Major project for submission.

# 

|  |  |
| --- | --- |
| Prof Namita Gupta  (HoD, CSE) | Ms. Garima Gupta  (Asst. Prof., CSE) |
| Dr. Neeraj Garg  (Assoc. Prof., CSE) |

# **ABSTRACT**

Imagine being able to detect blindness before it happened.

Millions of people suffer from diabetic retinopathy, the leading cause of blindness among working aged adults. Aravind Eye Hospital in India hopes to detect and prevent this disease among people living in rural areas where medical screening is difficult to conduct. Successful entries in this competition will improve the hospital’s ability to identify potential patients.

Currently, technicians capture images and then rely on highly trained doctors to review the images and provide diagnosis. The goal is to scale their efforts through technology; to gain the ability to automatically screen images for disease and provide information on how severe the condition may be.

We propose a machine learning model built to speed up disease detection. By working with thousands of images collected to help identify diabetic retinopathy automatically. If successful, we will not only help to prevent lifelong blindness, but these models may be used to detect other sorts of diseases in the future, like glaucoma and macular degeneration.

**ACKNOWLEDGEMENT**

It gives us immense pleasure to express our deepest sense of gratitude and sincere thanks to our respected guide Ms. Garima Gupta; Asst. Prof, CSE and co-guide Dr. Neeraj Garg: Assoc. Prof, CSE of MAIT Delhi, for their valuable guidance, encouragement and help for completing this work. Their useful suggestions for this whole work and cooperative behavior are sincerely acknowledged.

We also wish to express our indebtedness to our parents as well as our family members whose blessings and support always helped us to face the challenges ahead.

|  |  |
| --- | --- |
| Place: Delhi | Aayush A. Raina, 00214802717 |
| Date: 09/06/2021 | Aman Thukral, 00714802717 |
| Harkirat Singh, 02514802717 |

# **TABLE OF CONTENTS**

# 

|  |  |
| --- | --- |
| Content | Page No. |
| *List of Figures* | 6 |
| Introduction | 7 |
| Technology Stack | 9 |
| Architecture | 25 |
| Snapshots | 27 |
| Results | 29 |
| Conclusion | 30 |
| Future Scope | 31 |
| References | 32 |
| Proof of participation in Hackathon | 33 |

**LIST OF FIGURES**

|  |  |
| --- | --- |
| Figure | Page No. |
| Figure 1: Diabetic Retinopathy in eye | 7 |
| Figure 2: Use case diagram of the application | 26 |
| Figure 3: Retinal scan upload and patient details for analysing. | 27 |
| Figure 4: Displaying predicted results for the uploaded scan. | 27 |
| Figure 5: Generating consolidated reports of patient’s history. | 28 |
| Figure 6: Certificate of Participation (Aman) | 33 |
| Figure 7: Certificate of Participation (Aayush) | 34 |
| Figure 8: Certificate of Participation (Harkirat) | 34 |

# **INTRODUCTION**

# Diabetic retinopathy is a diabetes complication that affects eyes. It's caused by damage to the blood vessels of the light-sensitive tissue at the back of the eye (retina).

# At first, diabetic retinopathy may cause no symptoms or only mild vision problems. Eventually, it can cause blindness.

# The condition can develop in anyone who has type 1 or type 2 diabetes. The longer you have diabetes and the less controlled your blood sugar is, the more likely you are to develop this eye complication.

# **Illustration showing severe nonproliferative diabetic retinopathy**

*Figure 1.* Diabetic Retinopathy in eye

## **SYMPTOMS**

# You might not have symptoms in the early stages of diabetic retinopathy. As the condition progresses, diabetic retinopathy symptoms may include:

# Spots or dark strings floating in your vision (floaters)

# Blurred vision

# Fluctuating vision

# Impaired color vision

# Dark or empty areas in your vision

# Vision loss

# Diabetic retinopathy usually affects both eyes.

**DETECTION**

# Diabetic retinopathy is diagnosed entirely by recognizing abnormalities on retinal images taken by fundoscopy. Color fundus photography is mainly used for staging the disease. Fluorescein angiography is used to assess the extent of retinopathy that aids in treatment plan development. Optical coherence tomography (OCT) is used to determine the severity of edema and treatment response.

# Because fundoscopic images are the main sources for diagnosis of diabetic retinopathy, manually analyzing those images can be time-consuming and unreliable, as the ability of detecting abnormalities varies by years of experience.

# 

# **PROBLEM STATEMENT**

Blindness detection due to various conditions is a tedious process. Especially, due to the prevalence of diabetes, diabetic retinopathy is a crucial but difficult task for doctors. The prediction of blindness through pictures of the retina; a precise and tough task however busy for a person, is easier for a computer trained to do.

We propose a machine learning pipeline performing the precise task of accurate and quick prediction.

# **SCOPE**

Detection of blindness in diabetic patients

Project proposes a tool for doctors to utilize in the task of reading and understanding the factors of the patient’s eye condition.

It provides an interface that is:

* Easy-to-use
* Quick
* Reliable.
* Scalable
* Supported by multiple platforms

## **BENEFITS TO THE SOCIETY**

* The main benefit of the project is to assist the medical staff in analyzing a retinal eye scan for the possibility of blindness.
* The project will save doctors the tedious task of manual inspection of scans.
* The project will benefit patients in the speedy diagnosis and on-time treatment.

# **FEASIBILITY STUDY**

The proposed project is a cost-effective solution to the problem of probable error encountered during the manual reading of retinal reports by doctors, for diabetic retinopathy.

**ECONOMIC FEASIBILITY**

There is no need to purchase any specific hardware or any licensed software specifically for the application, either during development or post-development for the application to work. This makes the product economically feasible.

**TECHNOLOGICAL FEASIBILITY**

The technologies and datasets used are open-source and thus can be used free of cost.

The project requires the following concepts to be implemented with the highest precision and care:

1. ***Security*:** The project requires the user data to be stored and transmitted between client and server at the safest level possible. For the safety of data, we must encrypt the user data with the safest algorithms and mechanisms present in today's world.

2. ***Scalability*:** This project is intended to be used by multiple people together. So, to handle multiple users at once we must also take into account the factor of scalability. Scalability ensures that the project does not fail when there are multiple users, using the product together.

3. ***Synchronicity*:** The project is going to be used by multiple people together. There are going to be multiple database reads and writes at the same instant. We aim to make this process fault-proof. Keeping things in sync will be our utmost priority since here synchronicity also contributes to the security of the product

## 

## **TECHNOLOGY STACK**

## **SOFTWARE REQUIREMENTS:**

* Operating System: Windows 10, Linux, Unix
* Frontend: HTML/CSS,
* Backend: NodeJS, Python
* Browser: Chrome, Mozilla, Safari
* Tools: Visual Studio Code, Postman

**TECHNOLOGIES USED**

**HTML (Hypertext Mark-up Language)**



HTML stands for Hyper Text Markup Language, which is the most widely used language on the Web to develop web pages. HTML was created by Berners-Lee in late 1991 but “HTML 2.0” was the first standard HTML specification which was published in 1995. HTML 4.01 was a major version of HTML and it was published in late 1999. Though the HTML 4.01 version is widely used, currently we are having the HTML-5 version which is an extension to HTML 4.01, and this version was published in 2012.

Why to Learn HTML?

Originally, HTML was developed with the intent of defining the structure of documents like headings, paragraphs, lists, and so forth to facilitate the sharing of scientific information between researchers. Now, HTML is being widely used to format web pages with the help of different tags available in HTML language.HTML is a MUST for students and working professionals to become a great Software Engineer specially when they are working in Web Development Domain.

Some of the key advantages of learning HTML:

**Create Web site** – You can create a website or customize an existing web template if you know HTML well.

**Become a web designer** – If you want to start a career as a professional web designer, HTML and CSS designing is a must skill.

**Understand web** – If you want to optimize your website, to boost its speed and performance, it is good to know HTML to yield best results.

**Learn other languages** – Once you understand the basics of HTML then other related technologies like javascript, php, or angular become easier to understand.

**Applications of HTML**

As mentioned before, HTML is one of the most widely used languages over the web. I’m going to list few of them here:

**Web pages development** – HTML is used to create pages which are rendered over the web. Almost every page of the web has html tags in it to render its details in the browser.

**Internet Navigation** – HTML provides tags which are used to navigate from one page to another and is heavily used in internet navigation.

**Responsive UI** – HTML pages now-a-days works well on all platforms, mobile, tabs, desktop or laptops owing to responsive design strategy.

**Offline support** HTML pages once loaded can be made available offline on the machine without any need of the internet.

**Game development**- HTML5 has native support for rich experience and is now useful in the gaming development arena as well.

**CSS (Cascading Style Sheets) **

CSS is used to control the style of a web document in a simple and easy way.

CSS is the acronym for "Cascading Style Sheet". This tutorial covers both the versions CSS1,CSS2 and CSS3, and gives a complete understanding of CSS, starting from its basics to advanced concepts.

**Why to Learn CSS?**

Cascading Style Sheets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable.CSS is a MUST for students and working professionals to become a great Software Engineer specially when they are working in Web Development Domain.

Some of the key advantages of learning CSS:

**Create Stunning Website** - CSS handles the look and feel part of a web page. Using CSS, you can control the color of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, layout designs, variations in display for different devices and screen sizes as well as a variety of other effects.

**Become a web designer** - If you want to start a career as a professional web designer, HTML and CSS designing is a must skill.

**Control web** - CSS is easy to learn and understand but it provides powerful control over the presentation of an HTML document. Most commonly, CSS is combined with the markup languages HTML or XHTML.

**Learn other languages** - Once you understand the basics of HTML and CSS then other related technologies like javascript, php, or angular become easier to understand.

**Applications of CSS**

**CSS saves time** - You can write CSS once and then reuse the same sheet in multiple HTML pages. You can define a style for each HTML element and apply it to as many Web pages as you want.

**Pages load faster** - If you are using CSS, you do not need to write HTML tag attributes every time. Just write one CSS rule of a tag and apply it to all the occurrences of that tag. So less code means faster download times.

**Easy maintenance** - To make a global change, simply change the style, and all elements in all the web pages will be updated automatically.

**Superior styles to HTML** - CSS has a much wider array of attributes than HTML, so you can give a far better look to your HTML page in comparison to HTML attributes.

**Multiple Device Compatibility** - Style sheets allow content to be optimized for more than one type of device. By using the same HTML document, different versions of a website can be presented for handheld devices such as PDAs and cell phones or for printing.

**Global web standards** - Now HTML attributes are being deprecated and it is being recommended to use CSS. So it's a good idea to start using CSS in all the HTML pages to make them compatible with future browsers.

**JavaScript**

JavaScript is a lightweight, interpreted programming language. It is designed for creating network-centric applications. It is complementary to and integrated with Java. JavaScript is very easy to implement because it is integrated with HTML. It is open and cross-platform.

**Why to Learn Javascript?**

Javascript is a MUST for students and working professionals to become a great Software Engineer specially when they are working in Web Development Domain.

Some of the key advantages of learning Javascript:

Javascript is the most popular programming language in the world and that makes it a programmer’s great choice. Once you learn Javascript, it helps you develop great front-end as well as back-end softwares using different Javascript based frameworks like jQuery, Node.JS etc.

Javascript is everywhere, it comes installed on every modern web browser and so to learn Javascript you really do not need any special environment setup. For example Chrome, Mozilla Firefox , Safari and every browser you know as of today, supports Javascript.

Javascript helps you create really beautiful and crazy fast websites. You can develop your website with a console-like look and feel and give your users the best Graphical User Experience.

JavaScript usage has now extended to mobile app development, desktop app development, and game development. This opens many opportunities for you as a Javascript Programmer.

Due to high demand, there is tons of job growth and high pay for those who know JavaScript. You can navigate over to different job sites to see what having JavaScript skills looks like in the job market.

Great thing about Javascript is that you will find tons of frameworks and Libraries already developed which can be used directly in your software development to reduce your time to market.

There could be 1000s of good reasons to learn Javascript Programming. But one thing for sure, to learn any programming language, not only Javascript, you just need to code, and code and finally code until you become an expert.

There are many useful Javascript frameworks and libraries available

· Angular

· React

· jQuery

· Vue.js

· Ext.js

· Ember.js

· Meteor

· Mithril

· Node.js

· Polymer

· Aurelia

· Backbone.js

It is really impossible to give a complete list of all the available Javascript frameworks and libraries. The Javascript world is just too large and too much new is happening.

**Applications of Javascript Programming**

As mentioned before, Javascript is one of the most widely used programming languages (Front-end as well as Back-end). It has its presence in almost every area of software development. I'm going to list few of them here:

**Client side validation** - This is really important to verify any user input before submitting it to the server and Javascript plays an important role in validating those inputs at the front-end itself.

**Manipulating HTML Pages** - Javascript helps in manipulating HTML pages on the fly. This helps in adding and deleting any HTML tag very easily using javascript and modifying your HTML to change its look and feel based on different devices and requirements.

**User Notifications** - You can use Javascript to raise dynamic pop-ups on the webpages to give different types of notifications to your website visitors.

**Back-end Data Loading** - Javascript provides an Ajax library which helps in loading back-end data while you are doing some other processing. This really gives an amazing experience to your website visitors.

**Presentations** - JavaScript also provides the facility of creating presentations which gives the website look and feel. JavaScript provides RevealJS and BespokeJS libraries to build web-based slide presentations.

**Server Applications** - Node JS is built on Chrome's Javascript runtime for building fast and scalable network applications. This is an event based library which helps in developing very sophisticated server applications including Web Servers.

This list goes on, there are various areas where millions of software developers are happily using Javascript to develop great websites and other softwares.

**Node.Js**

Node.js is a platform built on Chrome's JavaScript runtime for easily building fast and scalable network applications. Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient, perfect for data-intensive real-time applications that run across distributed devices.

Node.js is an open source, cross-platform runtime environment for developing server-side and networking applications. Node.js applications are written in JavaScript, and can be run within the Node.js runtime on OS X, Microsoft Windows, and Linux.Node.js also provides a rich library of various JavaScript modules which simplifies the development of web applications using Node.js to a great extent.

Node.js = Runtime Environment + JavaScript Library

**Features of Node.js**

Following are some of the important features that make Node.js the first choice of software architects.

**Asynchronous and Event Driven** − All APIs of Node.js library are asynchronous, that is, non-blocking. It essentially means a Node.js based server never waits for an API to return data. The server moves to the next API after calling it and a notification mechanism of Events of Node.js helps the server to get a response from the previous API call.

**Very Fast** − Being built on Google Chrome's V8 JavaScript Engine, Node.js library is very fast in code execution.

**Single Threaded but Highly Scalable** − Node.js uses a single threaded model with event looping. Event mechanism helps the server to respond in a non-blocking way and makes the server highly scalable as opposed to traditional servers which create limited threads to handle requests. Node.js uses a single threaded program and the same program can provide service to a much larger number of requests than traditional servers like Apache HTTP Server.

**No Buffering** − Node.js applications never buffer any data. These applications simply output the data in chunks.

**License** − Node.js is released under the MIT license.

**Who Uses Node.js?**

Company list includes eBay, General Electric, GoDaddy, Microsoft, PayPal, Uber, Wikipins, Yahoo!, and Yammer to name a few.

Projects, Applications, and Companies Using Node

**Concepts**

The following diagram depicts some important parts of Node.js which we will discuss in detail in the subsequent chapters.

**Where to Use Node.js?**

Following are the areas where Node.js is proving itself as a perfect technology partner.

· I/O bound Applications

· Data Streaming Applications

· Data Intensive Real-time Applications (DIRT)

· JSON APIs based Applications

· Single Page Applications

**Where Not to Use Node.js?**

It is not advisable to use Node.js for CPU intensive applications.

**ExpressJs**

****

**What is Express.js?**

Express is a fast, assertive, essential and moderate web framework of Node.js. You can assume express as a layer built on the top of the Node.js that helps manage a server and routes. It provides a robust set of features to develop web and mobile applications.

Let's see some of the core features of Express framework:

· It can be used to design single-page, multi-page and hybrid web applications.

· It allows setting up middlewares to respond to HTTP Requests.

· It defines a routing table which is used to perform different actions based on HTTP method and URL.

· It allows dynamically rendering HTML Pages based on passing arguments to templates.

**Why use Express?**

· Ultra fast I/O

· Asynchronous and single threaded

· MVC like structure

· Robust API makes routing easy

**Python**

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. It was created by Guido van Rossum during 1985- 1990. Like Perl, Python source code is also available under the GNU General Public License (GPL). This tutorial gives enough understanding on Python programming language.

## Why to Learn Python?

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

Python is a MUST for students and working professionals to become a great Software Engineer specially when they are working in Web Development Domain. I will list down some of the key advantages of learning Python:

* Python is Interpreted − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
* Python is Interactive − You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
* Python is Object-Oriented − Python supports Object-Oriented style or technique of programming that encapsulates code within objects.
* Python is a Beginner's Language − Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

## **Characteristics of Python**

Following are important characteristics of Python Programming −

* It supports functional and structured programming methods as well as OOP.
* It can be used as a scripting language or can be compiled to byte-code for building large applications.
* It provides very high-level dynamic data types and supports dynamic type checking.
* It supports automatic garbage collection.
* It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

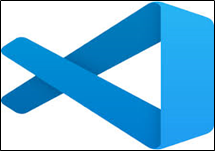
## **Applications of Python**

As mentioned before, Python is one of the most widely used language over the web. I'm going to list few of them here:

* Easy-to-learn − Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
* Easy-to-read − Python code is more clearly defined and visible to the eyes.
* Easy-to-maintain − Python's source code is fairly easy-to-maintain.
* A broad standard library − Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
* Interactive Mode − Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
* Portable − Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
* Extendable − You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
* Databases − Python provides interfaces to all major commercial databases.
* GUI Programming − Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
* Scalable − Python provides a better structure and support for large programs than shell scripting.

**IDE used**:

**Visual Studio Code:**



What is Visual Studio Code?

Visual Studio Code is a lightweight but powerful source code editor which runs on your desktop and is available for Windows, macOS and Linux. It comes with built-in support for JavaScript, TypeScript and Node.js and has a rich ecosystem of extensions for other languages (such as C++, C#, Java, Python, PHP, Go) and runtimes (such as .NET and Unity).

Why use Visual Studio Code?

Visual Studio Code combines the simplicity of a source code editor with powerful developer tooling, like IntelliSense code completion and debugging.

First and foremost, it is an editor that gets out of your way. The delightfully frictionless edit-build-debug cycle means less time fiddling with your environment, and more time executing on your ideas.

Available for macOS, Linux, and Windows

Visual Studio Code supports macOS, Linux, and Windows - so you can hit the ground running, no matter the platform.

Visual Studio Code runs on macOS, Linux and Windows

Edit, build, and debug with ease

At its heart, Visual Studio Code features a lightning fast source code editor, perfect for day-to-day use. With support for hundreds of languages, VS Code helps you be instantly productive with syntax highlighting, bracket-matching, auto-indentation, box-selection, snippets, and more. Intuitive keyboard shortcuts, easy customization and community-contributed keyboard shortcut mappings let you navigate your code with ease.

For serious coding, you'll often benefit from tools with more code understanding than just blocks of text. Visual Studio Code includes built-in support for IntelliSense code completion, rich semantic code understanding and navigation, and code refactoring.

And when the coding gets tough, the tough get debugging. Debugging is often the one feature that developers miss most in a leaner coding experience, so we made it happen. Visual Studio Code includes an interactive debugger, so you can step through source code, inspect variables, view call stacks, and execute commands in the console.

VS Code also integrates with build and scripting tools to perform common tasks making everyday workflows faster. VS Code has support for Git so you can work with source control without leaving the editor including viewing pending changes diffs.

Make it your own

Customize every feature to your liking and install any number of third-party extensions. While most scenarios work "out of the box" with no configuration, VS Code also grows with you, and we encourage you to optimize your experience to suit your unique needs. VS Code is an open-source project so you can also contribute to the growing and vibrant community on GitHub.

VS Code includes enriched built-in support for Node.js development with JavaScript and TypeScript, powered by the same underlying technologies that drive Visual Studio. VS Code also includes great tooling for web technologies such as JSX/React, HTML, CSS, SCSS, Less, and JSON.

Robust and extensible architecture

Architecturally, Visual Studio Code combines the best of web, native, and language-specific technologies. Using Electron, VS Code combines web technologies such as JavaScript and Node.js with the speed and flexibility of native apps. VS Code uses a newer, faster version of the same industrial-strength HTML-based editor that has powered the “Monaco” cloud editor, Internet Explorer's F12 Tools, and other projects. Additionally, VS Code uses a tools service architecture that enables it to integrate with many of the same technologies that power Visual Studio, including Roslyn for .NET, TypeScript, the Visual Studio debugging engine, and more.

Visual Studio Code includes a public extensibility model that lets developers build and use extensions, and richly customize their edit-build-debug experience.

**ARCHITECTURE**

The model will be running on the server side where it will fetch the fundus image uploaded by the user, and process it for detecting the possibility of blindness in the patient’s eye.

The server is set up in the *NodeJS* environment written in *Express*.

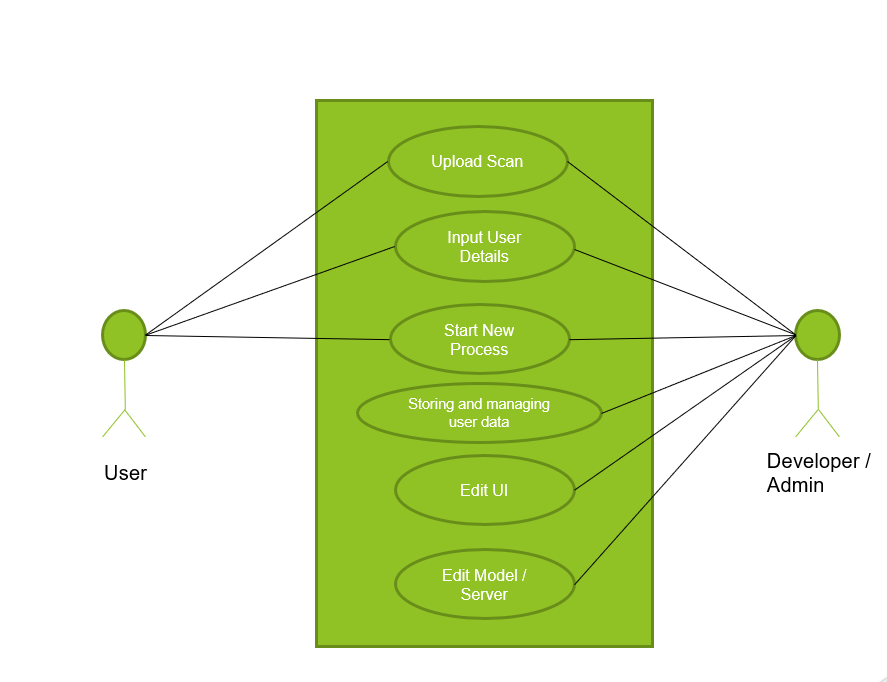
The input image gets uploaded to the destination directory with the help of an NPM package called *multer* which is a node.js middleware for handling multipart/form-data, primarily used for uploading files and storing them locally.

The front-end is scripted in HTML and styled using CSS.

The architecture of the model will consist of Convolutional Neural Networks(CNN), implementing transfer learning from a pretrained VGG net of 19 layers with batch normalization.

Result is stored in a unique JSON file(for a particular patient) to keep a track on the patient's history for future diagnosis.

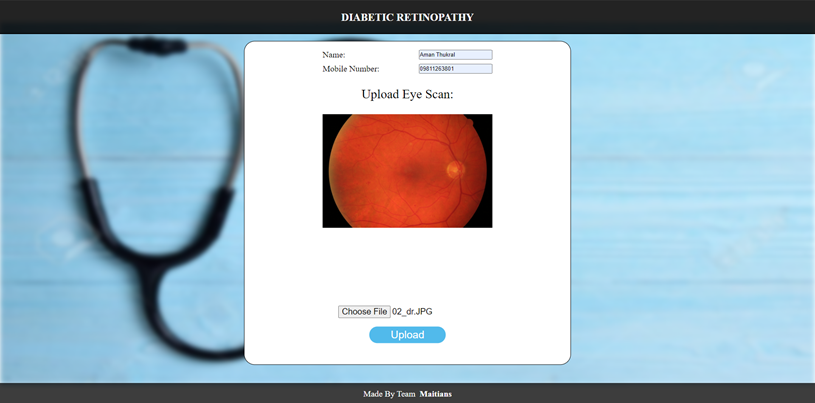
*Figure 2:* Use Case Diagram of the application



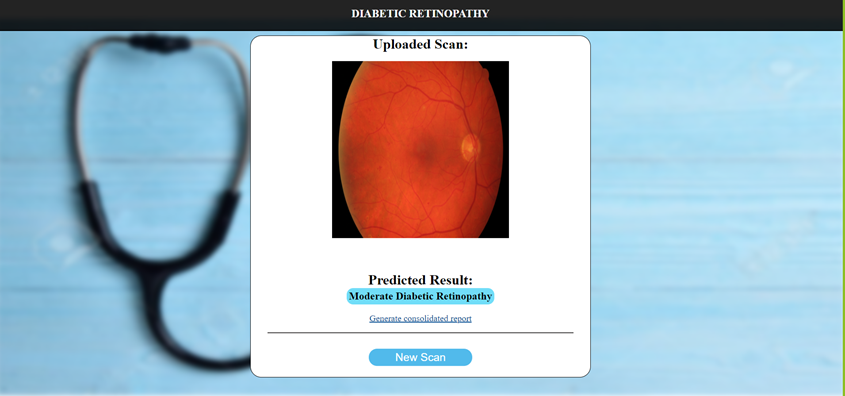
# 

# **SNAPSHOTS**

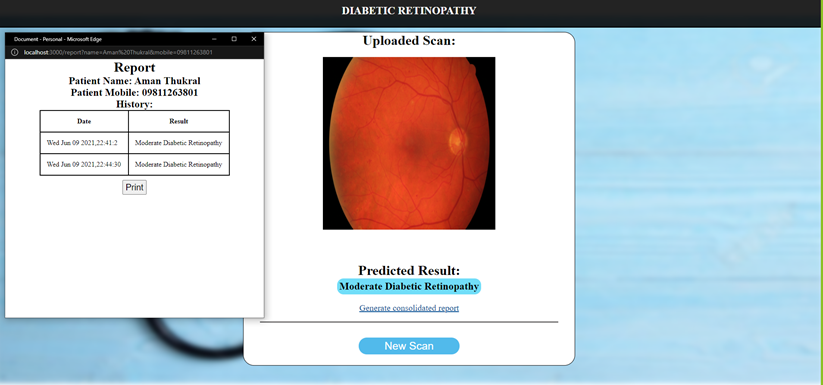
1. *Figure 3:* Retinal scan upload and patient details for analysing.



1. *Figure 4:* Displaying predicted results for the uploaded scan.



1. *Figure 5*: Generating consolidated reports of patient’s history.



# **RESULTS**

This application eases out the tedious task of analyzing the fundus images of retina and provides an efficient solution to manual detection of diabetic retinopathy.

The application provides the functionality to upload, check for, and process a fundus image of the eye and help with diagnosing the extent of diabetic retinopathy in the eye. It also allows storing and retrieving the history of the diagnosis.

# **CONCLUSION**

* Developing this project has helped us with a good understanding of diabetic retinopathy and the problems faced by the doctors during its manual detection. It has also given us an opportunity to apply knowledge of existing technologies to augment the automation of this process.
* The main benefit of the project is to assist the medical staff in analyzing a retinal eye scan for the possibility of blindness.
* The project will save doctors the tedious task of manual inspection of scans.
* The project will benefit patients in the speedy diagnosis and on-time treatment.

**FUTURE SCOPE**

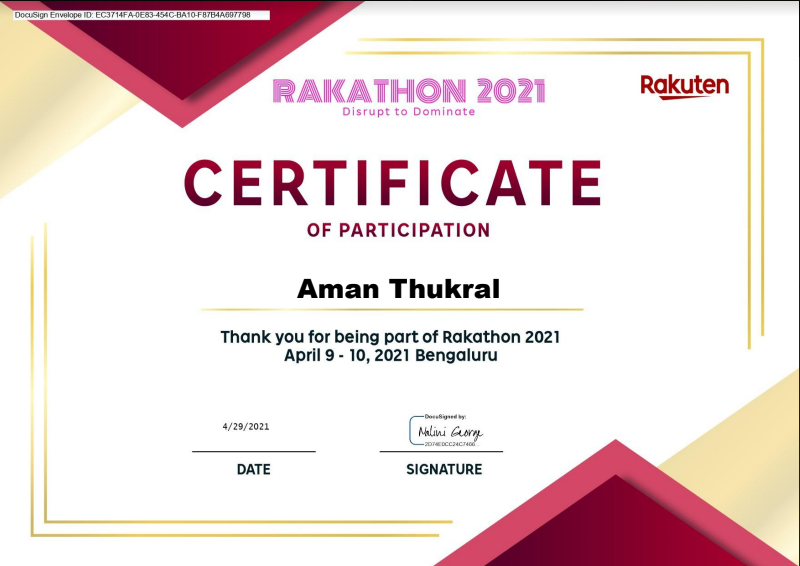
* Future Scope of the project is to improve the accuracy of the model.
* Scale the application if required.
* Making the application available on different platforms(Android, IOS, etc).

**REFERENCES**

* [Data Gov](https://data.gov.in)
* [Diabetic Retinopathy | National Eye Institute](https://nei.nih.gov/health/diabetic/retinopathy)
* [How to Run a Python script from Node.js | by Petros Koulianos | The Startup](https://medium.com/swlh/run-python-script-from-node-js-and-send-data-to-browser-15677fcf199f)
* [How to read and write a JSON object to a file in Node.js](https://attacomsian.com/blog/nodejs-write-json-object-to-file)

**PROOF OF PARTICIPATION IN HACKATHON:**

1. *Figure 6*:Certificate Aman Thukral.



1. *Figure 7*: Certificate Aayush A. Raina.



1. *Figure 8*: Certificate Harkirat Singh.

