Project Design Phase-II
Technology Stack (Architecture & Stack)

Date	16 October 2022
Team ID	PNT2022TMID04135
Project Name	Deep Learning Fundus Image Analysis For Early Detection Of Diabetic Retinopathy

Technical Architecture:

Reference :https://careereducation.smartinternz.com/saas-guided-project/3/deep-learning-fundus-image-analysis-for-early-detection-of-diabetic-retinopathy

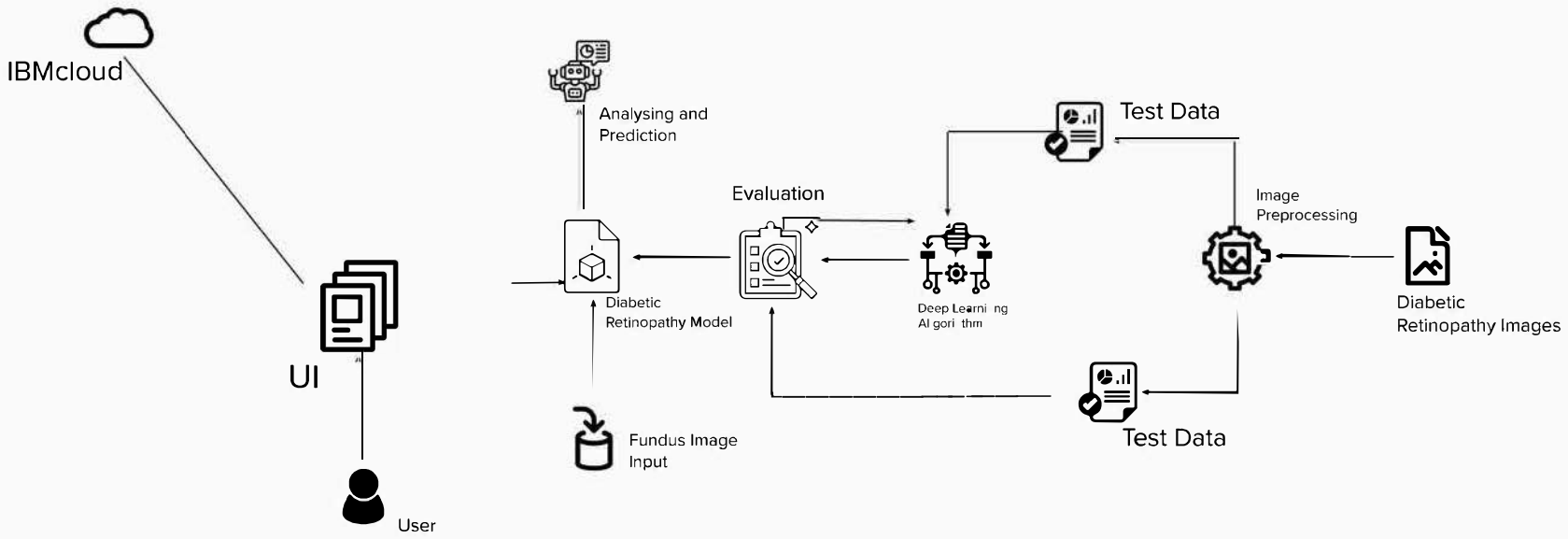
IBMcloud
UI
User
Analysing and
Prediction
Diabetic
Retinopathy Model
Fundus Image
Input
Evaluation
Deep Learni ng
Al gori thm
Test Data
Test Data
Image
Preprocessing
Diabetic
Retinopathy Images

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	Web UI	HTML, CSS, JavaScript,Python
2.	Application Logic-1	Data Preprocessing	Keras,Tensorflow,Numpy(Importing Essential Libraries)
3.	Application Logic-2	CNN Model Creating	Keras,Tensorflow,Numpy-(Importing Essential Libraries)
4.	Application Logic-3	Web Application(UI)	Flask
5.	Database	Images(Jpeg,PNG,Jpg,etc..,)	Upload Folder
6.	File Storage	File Storage requirements	IBM Cloud Storage.
7.	External API	Keras	Image processing API.
8.	Deep L earning Model	Inception v3Architecture	Pretrained convolutional neural network model that is 18 layers deep.
9.	Infrastructure (Server / Cloud)	Application Deployment on Webserver	Flask- A python WSGI HTTP server

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Flask	Werkzeug,Jinja2,Sinatra Rubyframework
2.	Security Implementations	CSRF protection,secure flag for cookies	Flask-WTF, SESSION_COOKIE_SECURE
3.	Scalable Architecture	Micro Services	Micro web application framework by Flask
4.	Availability	Development server and fast debugger Support for unit testing RESTful request Dispatching Jinja2 template Unicode	Werkzeug,Jinja2,Sinatra Rubyframework
5.	Performance	ORM-agnostic,web framework,WSGI 1.0 complaint,HTTP request handling functionality high flexibility	SQLAlchemy,extensions,Werkzeug,Jinja2,Sinatra Rubyframework

References:

https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/