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

Date	29 October 2022
Team ID	PNT2022TMID36210
Project Name	Project - Digital Naturalist – AI Enabled tool for Biodiversity Researchers
Maximum Marks	4 Marks

Technical Architecture:

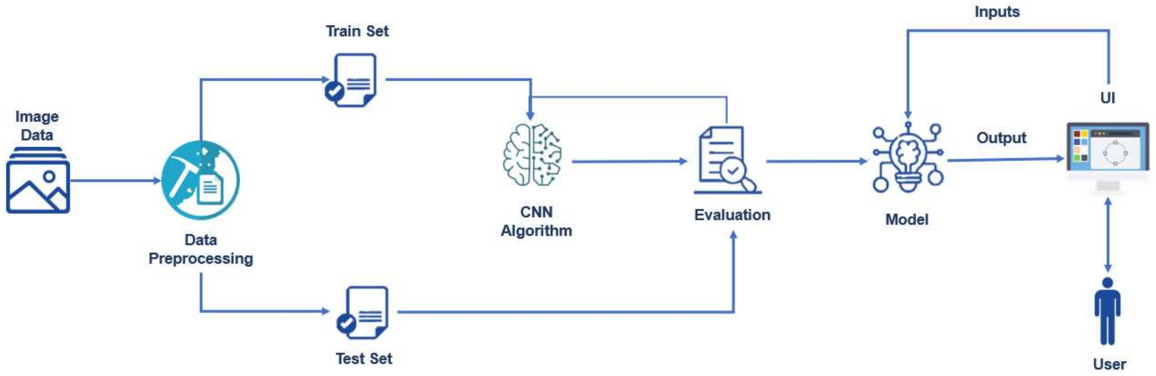

Table-1: Components & Technologies:

S. No	Component	Description	Technology
1.	User Interface	Web UI or Website	HTML, CSS, JavaScript / React JS
2.	Application Logic-1	Model building and then training the model	Python
3.	Application Logic-2	User uploads the image for the prediction	IBM Watson STT service
4.	Application Logic-3	Getting the relevant data from the database and providing to the user	IBM Watson Assistant
5.	Database	Image of all the variety species along with detailed information of each species	MySQL / NoSQL

6.	Cloud Database	Gets the data from database and feed them to model for prediction and also used to retrieve the data required for user.	IBM Cloudant, IBM DB2
7.	File Storage	User Login credentials, Images and their data, code and API keys	IBM Block Storage
8.	External API-1	To get data from the database when user gives the image as the input	IBM Storage API
9.	External API-2	To collect the username and password of the specific user	Secure Authentication API
10	Machine Learning Model	To predict the both flora & fauna through the image which is given as input and also it gives detailed information of the particular species	Image Recognition Model(Detecting the species and identifying the model)
11	Infrastructure (Server / Cloud)	To deploy the Application in Cloud Server	Cloud Foundry

Table-2: Application Characteristics:

S. No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Application is built by using flask	WSGI framework (Web Server Gateway Interface)
2.	Security Implementations	To Authenticate the species data in database as well as User credentials.	SHA-256, Encryptions
3.	Scalable Architecture	To scale our application in server side by supporting clients including desktop browsers, mobile browsers etc..	IBM Auto Scaling
4.	Availability	To make application available both online and offline and also 24/7 service.	IBM Cloud load balancer
5.	Performance	Designing an application which can handle wide range of requests at a time to provide accuracy in prediction as well as without any delay in time.	IBM instance