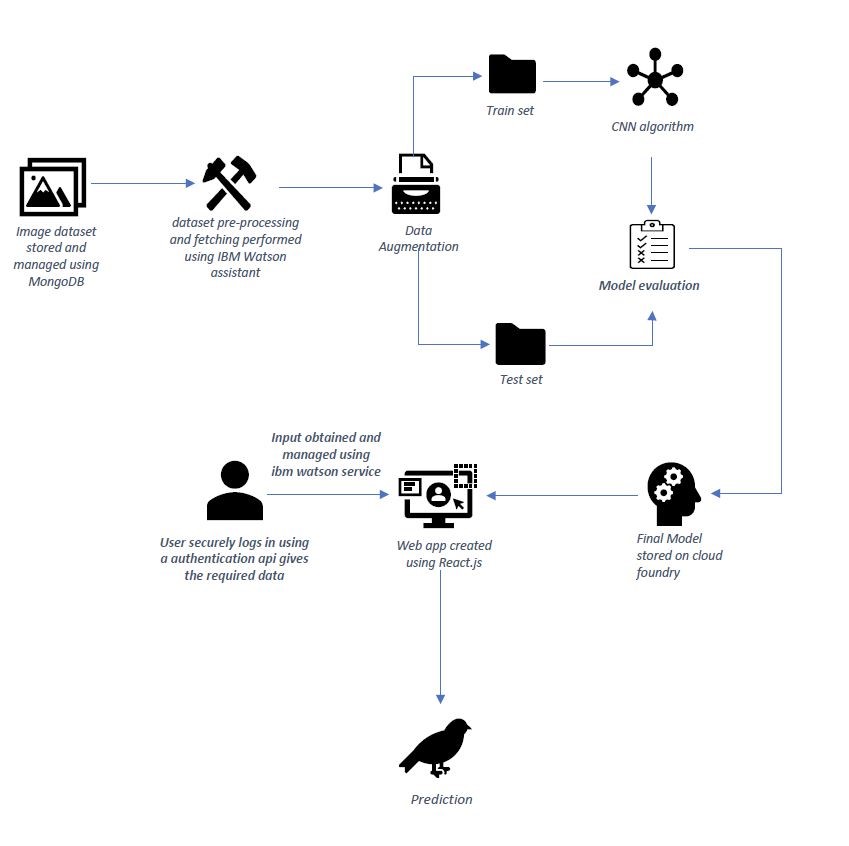
**Project Design Phase-II Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 23 October 2022 |
| Team ID | PNT2022TMID39141 |
| Project Name | 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 or Web app | HTML, CSS, React.js |
| 2. | Application Logic-1 | Model building and training | Python |
| 3. | Application Logic-2 | Getting image or text data from user for prediction | IBM Watson STT service |
| 4. | Application Logic-3 | Fetch the relevant data from the database and project them to user | IBM Watson Assistant |
| 5. | Database | Image and text data of all the species along with detailed view of each species | NoSQL (MongoDB) |
| 6. | Cloud Database | Fetch data from database and feed them to model for prediction and also used to retrieve the data required for user. | IBM Coundant |
| 7. | File Storage | Image data, login credentials, code (backend and frontend) and API keys | IBM Block Storage |
| 8. | External API-1 | To get data from the database when user give the image input | IBM Storage API |
| 9. | External API-2 | To get the username and password of the specific user | Secure Authentication API |
| 10. | Machine Learning Model | To predict the species (flora or fauna) through the image input and also it gives detailed view of the particular species | Species detection and identification model |
| 11. | Infrastructure (Server / Cloud) | To deploy our 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 | For authenticating the user data and protecting the data about species in database | SHA-256 and 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 that can handle wide range of requests at a time without any delay and to provide accuracy in prediction | IBM instance |