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

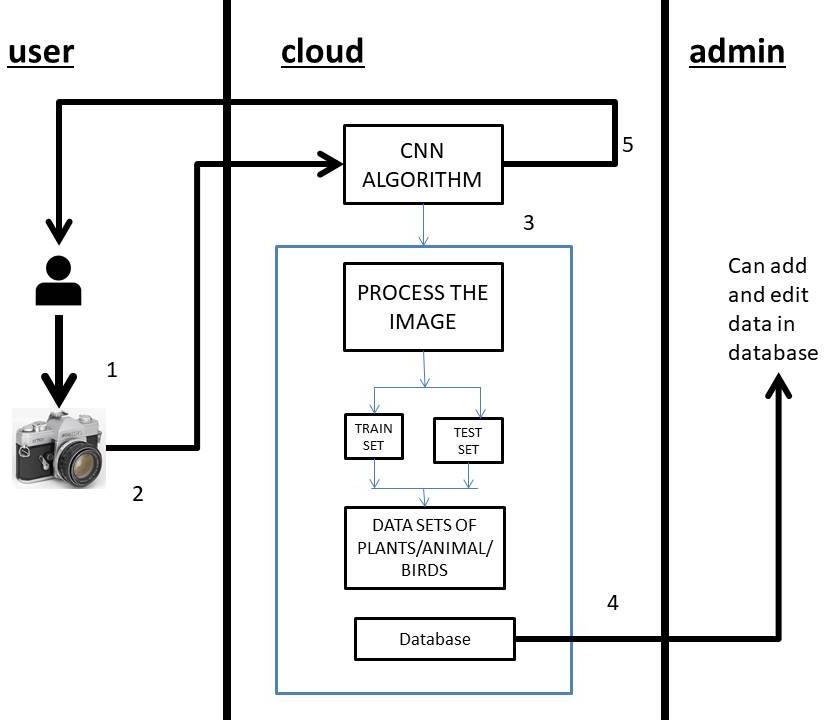
|  |  |
| --- | --- |
| Date | 15 October 2022 |
| Team ID | PNT2022TMID40645 |
| Project Name | **Project -** Digital Naturalist – AI Enabled tool for  Biodiversity Researchers |
| Maximum Marks | 4 Marks |

# Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

# Example: Order processing during pandemics for offline mode

**Reference:** [**https://developer.ibm.com/patterns/ai-powered-backend-system-for-order-processing-during-pandemics/**](https://developer.ibm.com/patterns/ai-powered-backend-system-for-order-processing-during-pandemics/)



Guidelines:

1. Include all the processes (As an application logic / Technology Block)
2. Provide infrastructural demarcation (Local / Cloud)
3. Indicate external interfaces (third party API’s etc.)
4. Indicate Data Storage components / services
5. Indicate interface to machine learning models (if applicable)

# Table-1 : Components & Technologies:

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Component** | **Description** | **Technology** |
| 1. | User Interface | How user interacts with application e.g.  Web UI, Mobile App, Chatbot etc. | HTML, CSS, JavaScript / Angular Js / React Js etc. |
| 2. | Camera | Capture photos for processing | Normal phone camera |
| 3. | CNN algorithm | For processing the photos | - |
| 4. | Database | Data Type, Configurations etc. | MySQL, etc. |
| 5. | Cloud Database | Database Service on Cloud | IBM DB2, IBM Cloudant etc. |
| 6. | File Storage | File storage requirements | IBM Block Storage or Other Storage Service or Local Filesystem |
| 7. | External API-1 | Location service | Location or google api |

**Table-2: Application Characteristics:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Characteristics** | **Description** | **Technology** |
| 1. | Open-Source Frameworks | Datasets,api etc | Sql or csv |
| 2. | Security Implementations | List all the security / access controls implemented, use  of firewalls etc. | e.g. SHA-256, Encryptions, IAM Controls,  OWASP etc. |
| 3. | Scalable Architecture | Justify the scalability of architecture (3 – tier, Micro-  services) | Python, Mysql |
| 4. | Availability | Justify the availability of application (e.g. use of load  balancers, distributed servers etc.) | Python, Mysql |
| 5. | Performance | Design consideration for the performance of the  application (number of requests per sec, use of Cache, use of CDN’s) etc. | - |

References:

<https://c4model.com/>

<https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/> <https://www.ibm.com/cloud/architecture>

<https://aws.amazon.com/architecture>

<https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d>