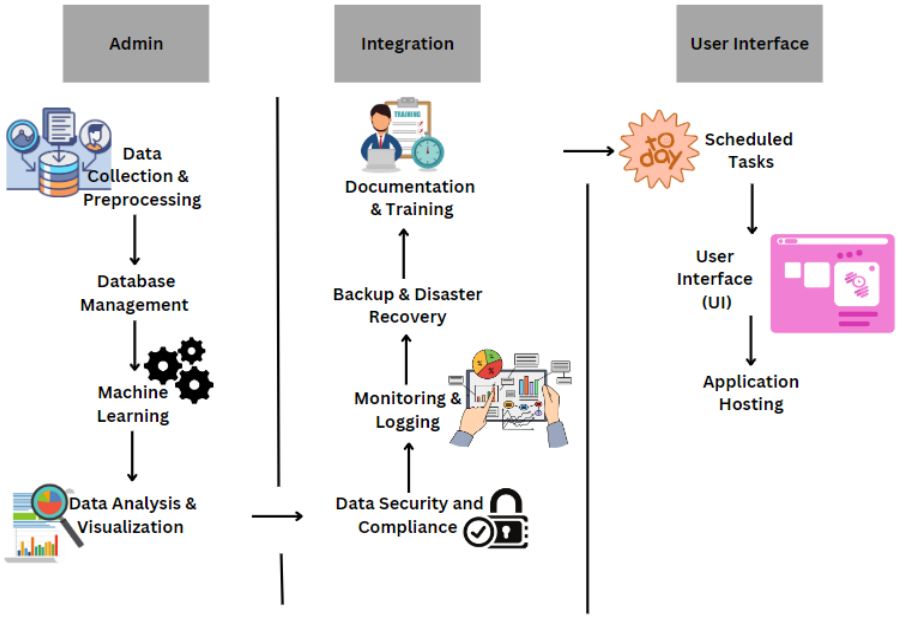
Project Design Phase-II
Technology Stack (Architecture & Stack)

Date	21-10-2023
Team ID	Team-591242
Project Name	India's Agricultural Crop Production Analysis
Minimum marks	4 marks

Technical Architecture:

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



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	User interaction with the web-based application, including data visualization, navigation, and user input.	HTML, CSS, JavaScript, React.js
2.	Data Processing Logic	Logic for data processing, including historical crop data analysis, data cleansing, and transformation.	Python
3.	Machine Learning	Integration of machine learning models for crop yield prediction and anomaly detection.	IBM Watson Machine Learning
4.	Database	Data storage and retrieval for historical crop data, user information, and application configurations.	MySQL
5.	Cloud Database	Cloud-based database service for scalability and data synchronization.	IBM Db2 on IBM Cloud
6.	File Storage	File storage requirements for storing user-uploaded files, data backups, and system logs.	IBM Cloud Object Storage
7.	External APIs	Integration with external APIs for real-time weather data, agricultural market information, and geographic data.	IBM Weather Company Data, Google Maps API
8.	User Authentication	Implementation of user authentication and access control mechanisms.	OAuth 2.0, JWT
9.	Application Hosting	Hosting and deployment of the web application on cloud-based servers.	IBM Cloud Foundry, Docker, Kubernetes

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Utilization of open-source frameworks to streamline development and enhance application capabilities.	Node.js, Express.js
2.	Security Implementations	Implementation of security measures such as data encryption, access controls, and OWASP best practices.	HTTPS, SHA-256, Role-based access control
3.	Scalable Architecture	Adoption of microservices architecture to ensure the application's scalability, resilience, and modularity.	Docker, Kubernetes
4.	Availability	Ensuring high availability through load balancers, auto-scaling, and redundancy measures.	AWS Elastic Load Balancer, Failover Clustering
5.	Performance	Optimization for high performance, including caching, content delivery networks (CDN), and efficient query processing.	Redis, Content Delivery Networks (CDN)