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

Date	06 May 2023
Team ID	NM2023TMID10705
Project Name	**Project - The Issue Tracker: A Reliable Complaint Management System For Improved Customer Service.**

Technical Architecture details:

User Interface: HTML, CSS, JavaScript, and Bootstrap for creating an interactive and visually appealing user interface.

Flask Web Framework: Integrating the web application with the Flask framework to handle the back-end logic and connect with front-end services.

IBM DB2: Connecting the application to IBM DB2 for storing and retrieving complaint data.

Cloud Object Storage: Utilizing cloud object storage (e.g., IBM Cloud Object Storage) to store multimedia data associated with complaints.

Docker: Containerizing the application into a Docker image for easy deployment and scalability.

Kubernetes: Managing the deployed application using Kubernetes for efficient container orchestration, scaling, and load balancing.

Cloud Deployment: Deploying the containerized application in the cloud, providing availability, accessibility, and scalability to users.

Technical Architecture:

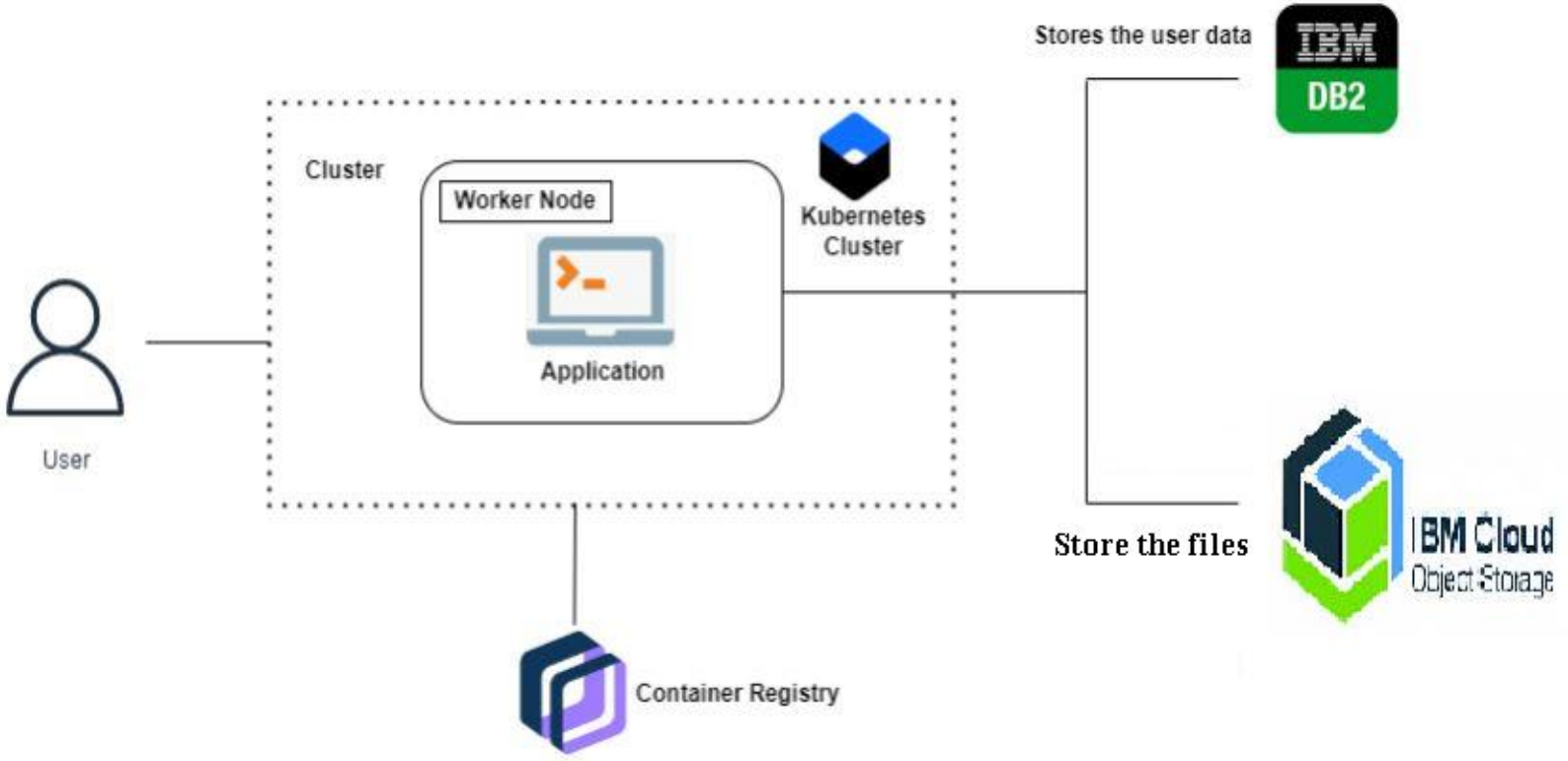

Table-1 : Components & Technologies:

S.No	Component	Description	
1.	User Interface	The user interface i web application to t Technology	
2.	Flask Application	The Flask applicati that handles the bac front-end user inter services	
3.	Database	The database stores user complaints, ag status.	
4.	Cloud Object Storage	The cloud object storage is used to store multimedia data such as images related to user complaints	Cloud Object Storage (e.g., IBM Cloud Object Storage)
5.	Containerization	The application is containerized as a Docker image, which allows for easy deployment and scalability.	Docker
6.	Container Orchestration	Kubernetes manages and orchestrates the deployment of the containerized application, ensuring scalability, availability, and load balancing.	Kubernetes

Guidelines:

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

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	User Interface:	The user interface provides the visual and interactive elements for users to interact with the web application.	HTML, CSS, JavaScript, Bootstrap
2.	Database Connection:	Establishes a connection between the web application and the database for storing and retrieving data.	Python, IBM DB2
3.	Schema and Tables:	Creation of necessary database schemas and tables to organize and store complaint-related data.	IBM DB2
4.	Cloud Object Storage:	Utilizes cloud object storage to store multimedia data associated with user complaints, such as images.	Cloud Object Storage (e.g., IBM Cloud Object Storage)
5.	Flask Application:	Develops the Flask web application to handle the back-end logic, process user requests, and interact with the database and object storage.	Python,Flask
6.	Containerization	Containerizes the application into a Docker image, providing a portable and consistent environment for deployment.	Docker
7.	Container Management :	Utilizes Kubernetes to manage the deployment, scaling, and availability of the containerized application in the cloud.	Kubernetes