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

Date	03 October 2022
Team ID	PNT2022TMID21522
Project Name	Project – Customer Care Registry
Maximum Marks	4 Marks

Technical Architecture:

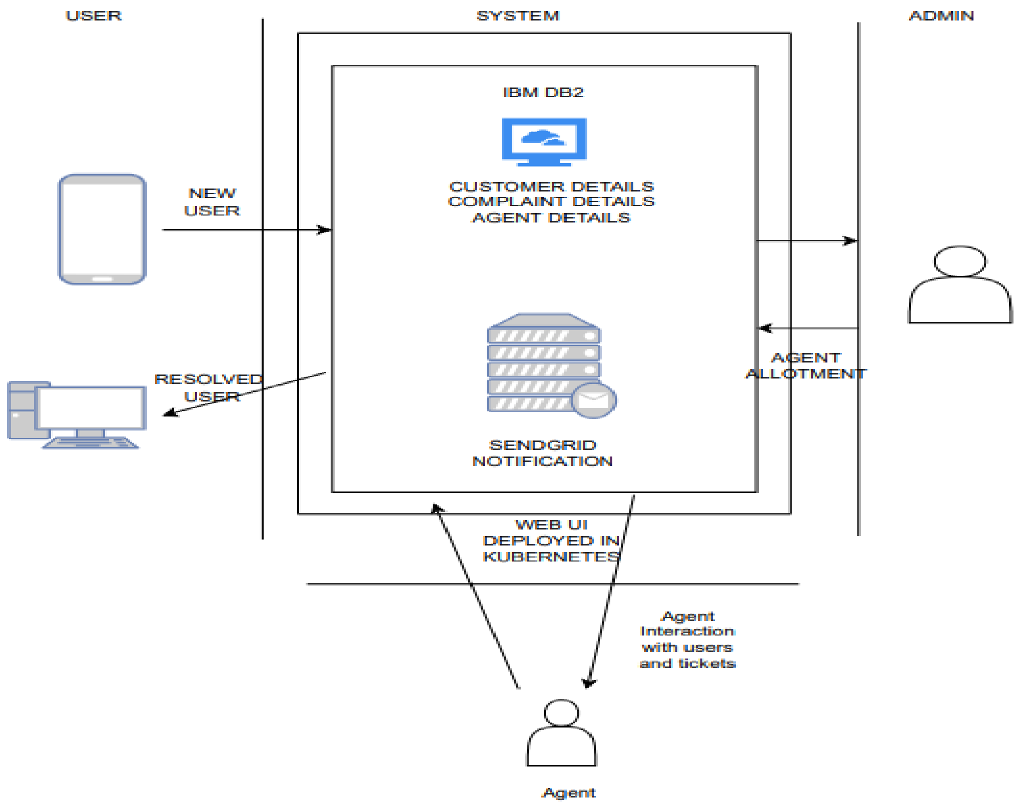

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	The user interacts with the web application	HTML, CSS, JavaScript, Bootstrap, Jinja python etc.
2.	Application Logic-1	Logic for a login in the application	Python
3.	Application Logic-2	Logic for a registration in the application	Python
4.	Application Logic-3	Logic for a process in the application	IBM Watson Assistant
5.	Database	Data Type, Configurations etc.	MySQL
6.	Cloud Database	Database Service on Cloud	IBM DB2
7.	File Storage	File storage requirements	IBM Object Storage
8.	Infrastructure (Server / Cloud)	Application Deployment on Cloud Server Configuration	Kubernetes, Cloud Foundry

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Flask	Python flask, Bootstrap
2.	Security Implementations	Basic HTTP authentication, Session based authentication, User Registration, Login Tracking	Flask
3.	Scalable Architecture	Kubernetes allows users to horizontally scale the total containers used based on the application requirements, which may change over time. It’s easy to change the number via the command line.	Kubernetes

S.No	Characteristics	Description	Technology
		we can also use the Horizontal Pod Auto scaler to do this.	
4.	Availability	Kubernetes High-Availability is about setting up Kubernetes, along with its supporting components in a way that there is no single point of failure. A single master cluster can easily fail, while a multi-master cluster uses multiple master nodes, each of which has access to same worker nodes.	Kubernetes
5.	Performance	Kubernetes and containers allow for much better resource utilization than hypervisors and VMs do. Because containers are so lightweight, they require less CPU and memory resources to run.	Kubernetes

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