PROJECT DESIGN PHASE II

TECHNOLOGY STACK (ARCHITECTURE AND STACK)

DATE	16 OCT 2022
TEAM ID	PNT2022TMID52556
PROJECT NAME	Inventory Management System for Retailers
MAXIMUM MARKS	4 MARKS

TECHNICAL ARCHITECTURE:

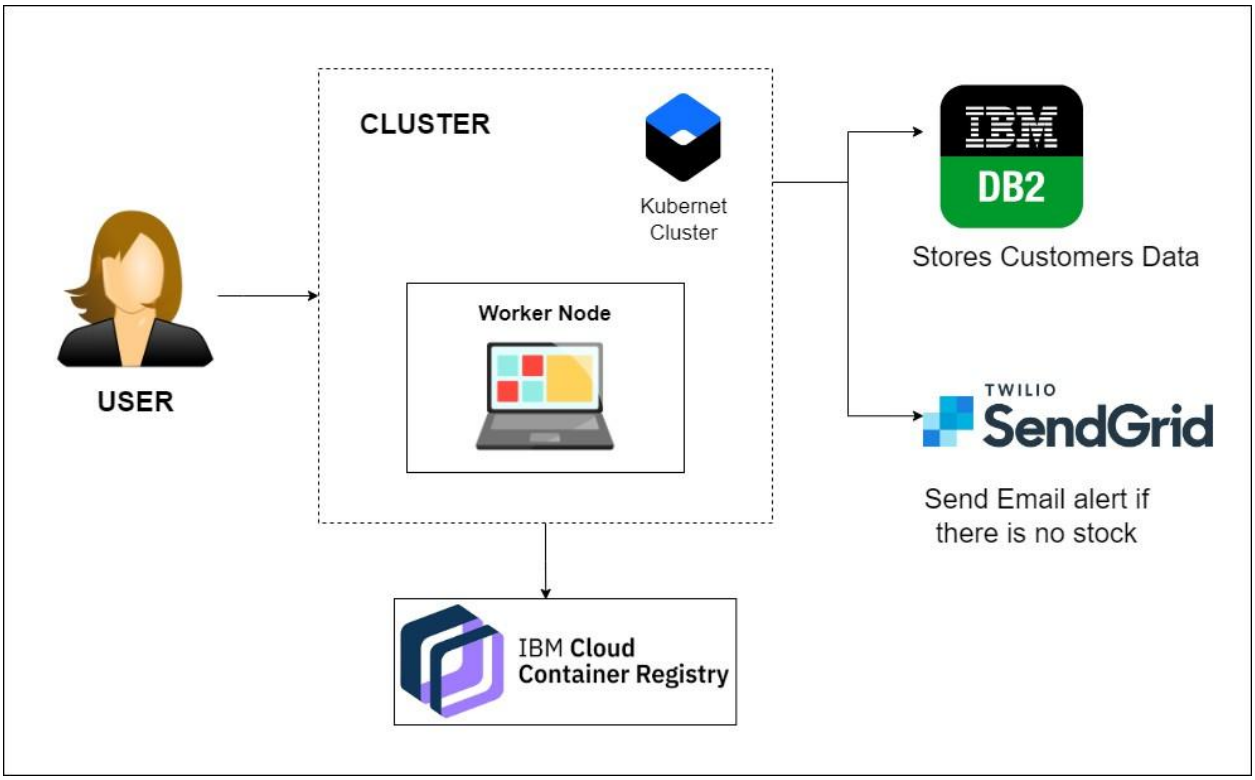

Table 1: Components & Technologies:

S.NO	COMPONENT	DESCRIPTION	TECHNOLOGY
1.	User Interface	Information processed will be sent to the user as a mail through web application	HTML, CSS, JQuery, JS, Python, etc.
2.	Application Logic-1	User registration through form and confirmation will be sent to the user via email.	Flask, SendGrid
3.	Application Logic-2	Dashboard is used to track the sales of product and inventory levels.	Flask
4.	Application Logic-3	User will get notified about the real time stock status	Flask
5.	Database	The data can be stored in a database and the user can retrieve or manipulate the data anytime	IBM DB2.
6.	Cloud Database	Information of the stocks will be stored and hosted on the cloud	IBM DB2.
7.	File Storage	Requirements to store files.	IBM Block Storage or Other Storage Serviceor Local File system
8.	External API-1	SendGrid which is used in application will sendthe email alert if there is less number or no stock to the user in real time	SendGrid
9.	External API-2	IBM container Registry enables you to store and distribute Docker images in a managed private registry	IBM container registry
10.	Infrastructure (Server/Cloud)	Application Deployment on Local System / CloudLocal Server Configuration:localhost:5001(Flask) Cloud Server Configuration : Kubernetes	Local, Cloud Foundry, Kubernetes, etc.

Table-2: Application Characteristics:

S.NO	CHARACTERISTICS	DESCRIPTION	TECHNOLOGY
1.	Open-Source Frameworks	SendGrid will send email alerts, if there is less stock to the user, Kubernetes for manipulating Kubernetes API objects, IBM DB2 is used for storing and retrieving the data efficiently.	Flask, SendGrid, IBMDB2, Kubernetes
2.	Security Implementations	We use login for the user and the information will be hashed so that it will be very secure to use.	IBM container registry
3.	Scalable Architecture	It is scalable that we are going to use data in kb so that the quiet amount of storage is satisfied.	Flask
4.	Availability	Prediction will be available for every user but only for premium user news, database and price alert will be alert	Flask
5.	Performance	It will perform fast and secure even at the lower bandwidth	Flask, IBM container registry, IBM DB2.

REFERENCES:

https://c4model.com/

https://www.ibm.com/cloud/architecture

https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/

https://aws.amazon.com/architecture