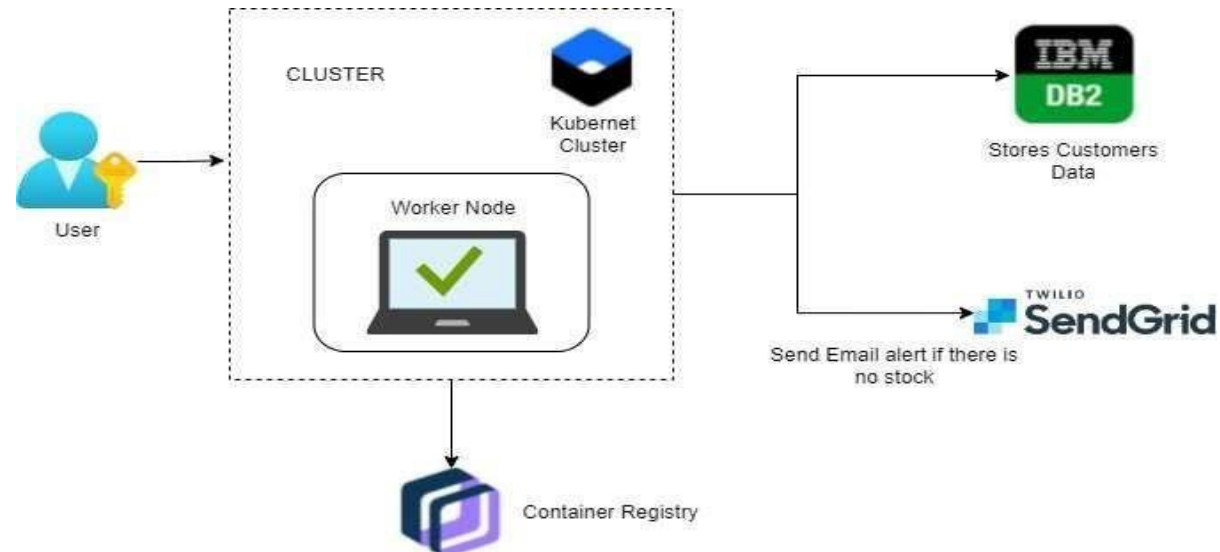


**Project Design Phase-II**  
**Technology Stack (Architecture & Stack)**

Date	16 October 2022
Team ID	PNT2022TMID24681
Project Name	Project – Inventory Management System for Retailers
Maximum Marks	4 Marks

**Technical Architecture:**



**Table-1 : Components & Technologies:**

S.No	Component	Description	Technology
1.	User Interface	web application with chatbot.	HTML, CSS, node js, JavaScript, python, etc.
2.	Application Logic-1	User registration through form and confirmation will be sent to the user via email. Inventory stock details send to registered mail id.	Flask, SendGrid
3.	Application Logic-2	Dashboard is used by which the system will Maintaining inventory levels.	Flask
4.	Application Logic-3	User will get notified about the stock status via email.	Flask
5.	Database	The data can be stored in database and user can retrieve the data whenever required. If enter the product name automatically shows the inventory stock count.	IBM DB2.
6.	Cloud Database	Information of the stocks will be stored and hosted on the IBM cloud.	IBM DB2.
7.	File Storage	Requirements to store files.if the product count zero,it will automatically notify.	IBM Block Storage or Other Storage Service or Local File system
8.	External API-1	SendGrid used in application will send the email alert if there is less number or no stock to the user	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 / Cloud Local 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 alert, if there is less number of stock to user, Kubernetes for manipulating Kubernetes API objects, IBM DB2 is used for storing and retrieving the data efficiently.so it is used for all retailers.	Flask, SendGrid, IBMDB2, Kubernetes
2.	Security Implementations	We use login for the user and the information will Be stored 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 kubernetes so that the quite 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.