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

Date	03 October 2022
Team ID	PNT2022TMID20053
Project Name	Project – Personal Expense Tracker
Maximum Marks	4 Marks

Technical Architecture:

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

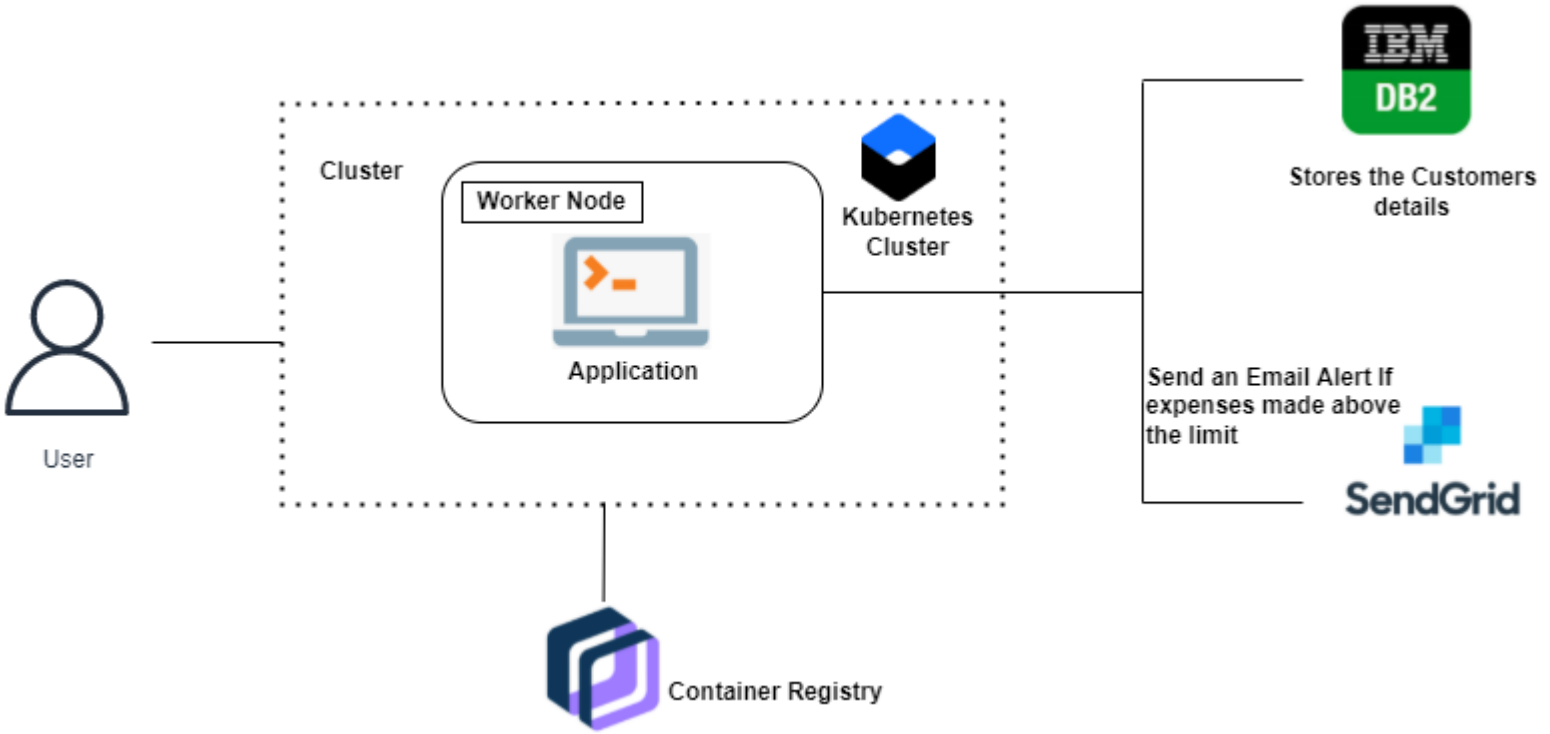

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	How user interacts with application e.g. Web UI, Mobile App, Chatbot etc.	HTML, CSS, JavaScript etc.
2.	Application Logic-1	The application contains the sign in/sign up page where the user will login into the main dashboard	Python
3.	Application Logic-2	Dashboard contains the fields like add expenses , add budget limit.	IBM Watson STT service
4.	Application Logic-3	The user will receive alert notification if budget limits exceeds and gets the expense and income analysis report with better ideas.	IBM Watson Assistant
5.	Database	The database contains expenses, budget limit informations.	MySQL
6.	Cloud Database	User information are stored securely in cloud database.	IBM DB2
7.	File Storage	The user's income and finance data are stored in ibm block storage.	IBM Block Storage or Local Filesystem
8.	External API-1	Purpose of External API used in the application	SendGrid API
9.	External API-2	Purpose of External API used in the application	Economic News API
10.	Container Registry	Container images storage	Local Registry: DockerHub Cloud Registry: Container Registry
11.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: Minikube Cloud Server Configuration : Kubernetes	Local, Cloud Foundry, Kubernetes, etc.

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Flask framework in python is used to implement this application	Python - Flask
2.	Security Implementations	This application provides high security to the user financial data. It can be done using the container registry in IBM cloud	Container Registry, Kubernetes
3.	Scalable Architecture	This application has lifetime access. Large amount of data can be stored in cloud database.	Container Registry, Kubernetes
4.	Availability	This application can accessible by for any number of users and it is available at any time.	Container Registry, Kubernetes cluster
5.	Performance	The Performance of this application is high because there will be no network traffic.	Kubernetes cluster