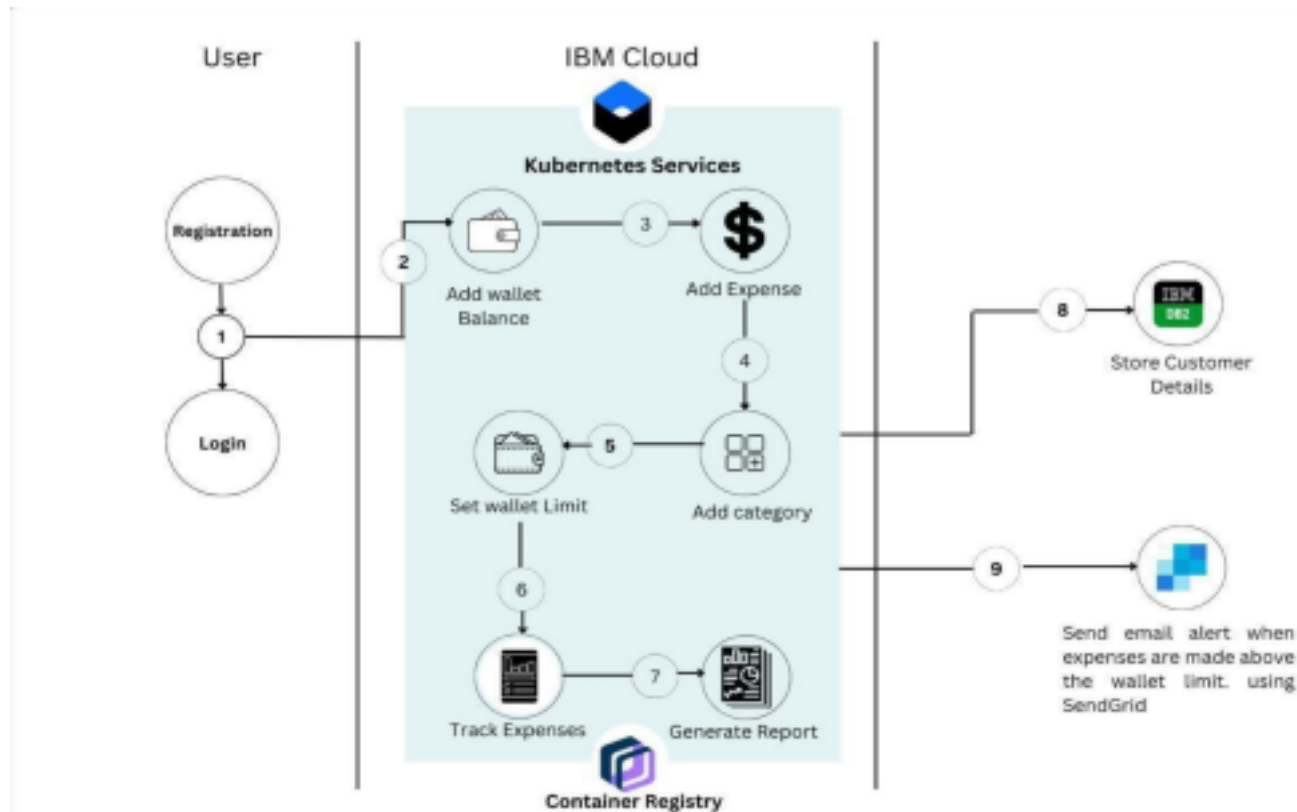


**Project Design Phase-II**  
**Technology Architecture**

Date	20 October 2022
Team ID	PNT2022TMID23410
Project Name	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:**

	Component	Description	Technology
1.	User Interface	The user can Interact with the application with use of Chatbot	HTML, CSS, JavaScript
2.	Registration and Login	The application contains the sign in/sign up where the user will	Python , Docker

		login into the main dashboard	
3.	Wallet Dashboard	Dashboard contains the fields like Add income, Add Expenses, Save Money	IBM Cloud Kubernetes Services
4.	Tracking of expenses	The user will get the expense report in the graph form and also get alerts if the expense limit exceeds.	IBM Watson Assistant , SendGrid
5.	Database	The Income and Expense data are stored in the MySQL database.	MySQL
6.	Cloud Database	With use of Database Service on Cloud, the User data is stored in a well secured Manner.	IBM DB2
7.	File Storage	IBM Block Storage used to store the Financial data of the user	IBM Block Storage or Other Storage Service or Local Filesystem

**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 by using the Container Registry in IBM cloud	SHA-256 Hashing Algorithm , 256-bit AES Encryption
3.	Scalable Architecture	Expense Tracker is a lifetime access application. It's demand will increase when the user's income are high	Container Registry, Kubernetes Cluster
4.	Availability	This application will be available to the user at any part of time	Docker , Kubernetes Cluster
5.	Performance	The performance will be high because there will be no network traffics in the application	IBM Container Registry