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

Date	03 October 2022
Team ID	PNT2022TMID14648
Project Name	Project - Personal Expense Tracker Application
Maximum Marks	4 Marks

Technical Architecture:

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

Personal Expense Tracker Application

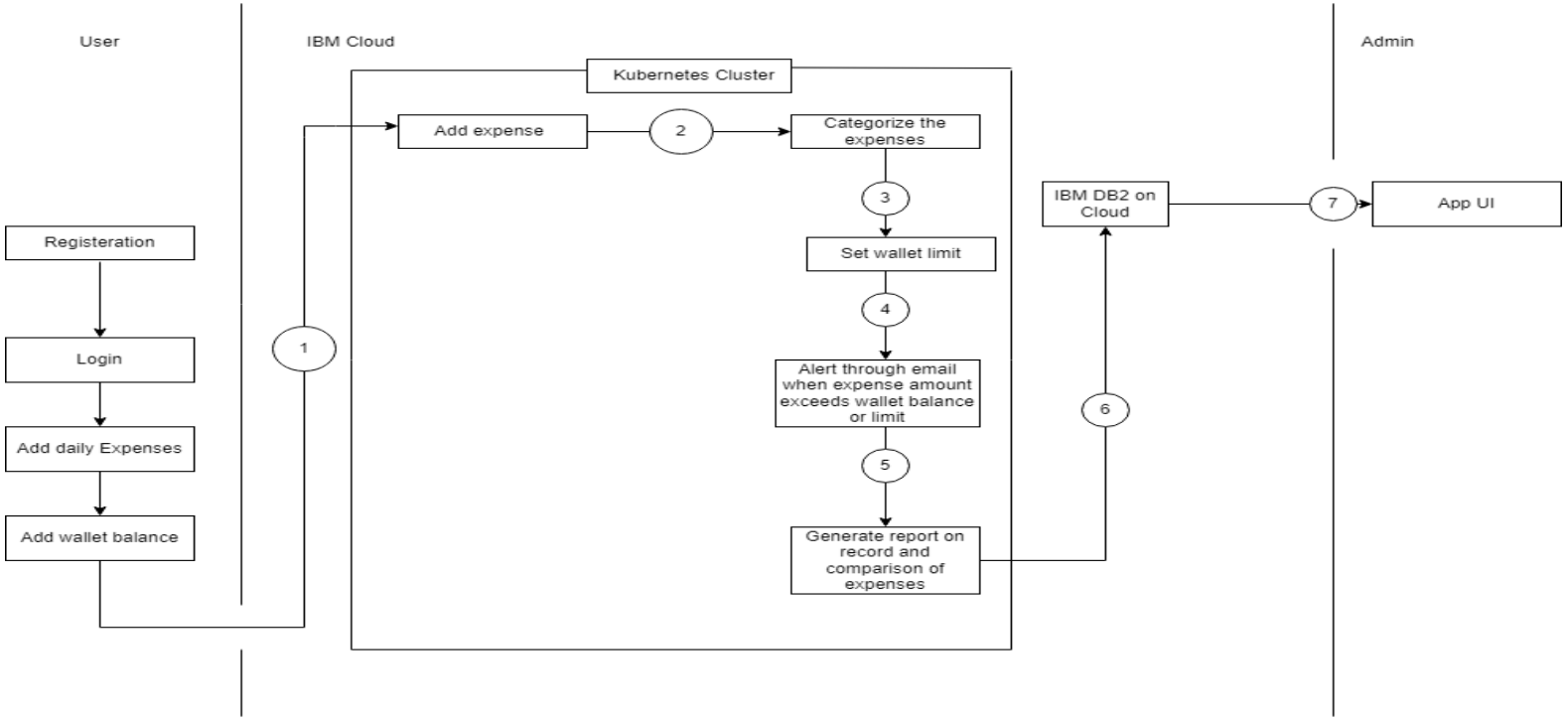

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	User interacts with application through Web UI, Mobile App.	HTML, CSS, Python
2.	Registration and Login	To access the application through email	Python, Docker
3.	Wallet Dashboard	Allow user to enter wallet limit amount and wallet balance	Python
4.	Tracking of Expenses.	IBM Container Registry enables to store and distribute Docker images in a managed, private registry	IBM Cloud Container Registry
5.	Report	The user will get the expense report in the graph form and also get alerts if the expense limit exceeds	IBM Watson Assistant, Sendgrid
6.	Database	The Income and Expense data of user are stored in the MySQL database	MySQL
7.	Cloud Database	With use of Database Service on Cloud, the User data are stored in a well secured Manner	IBM DB2.
8.	File Storage	IBM Block Storage used to store the expenses data of the user	IBM Block Storage or Other Storage Service or Local File system
9.	External API-1	To send email alerts when the expenses are exceed the wallet limit.	App UI
10.	External API-2	Allow user to fill the feedback form	App UI
11.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Cloud Server Configuration :Kubernetes services,Sendgrid	Local, Cloud Foundry, Kubernetes

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Flask is a application framework written in Python used to implement this Application	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	Container Registry, Kubernetes Cluster
3.	Scalable Architecture	Three-tier architecture- user server, application server and cloud server.	Python, IBM Cloud Services
4.	Availability	This application will be available to the user at any part of time	Container Registry, Kubernetes Cluster
5.	Performance	The probability to handle the increasing demand and number of user is high	IBM Container Registry.