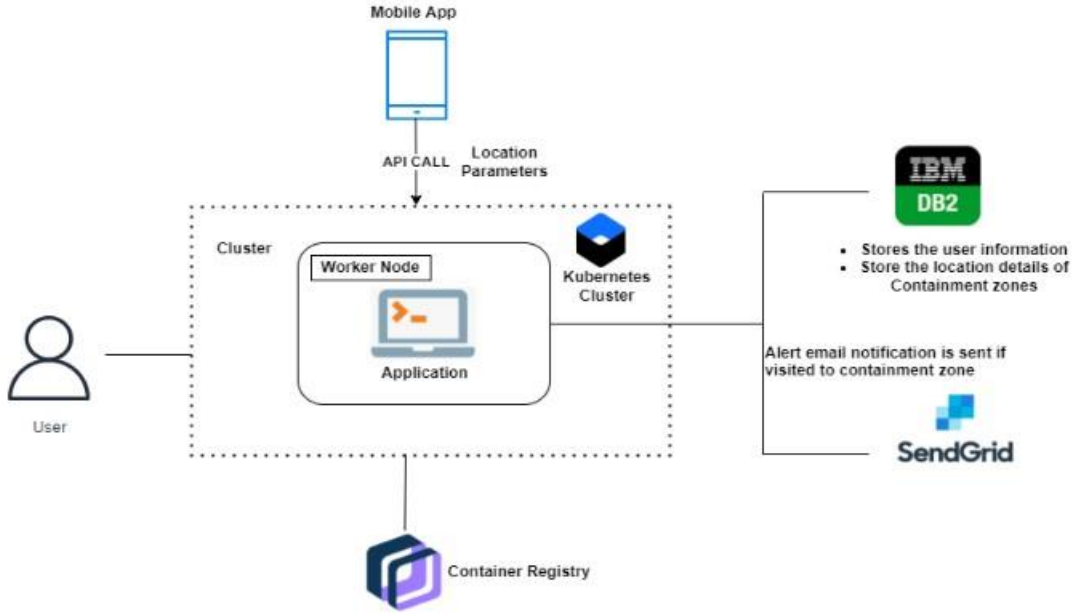
Project Design Phase-II

Technology Stack (Architecture & Stack)

Date	20 October 2022
TeamID	PNT2022TMID48287
ProjectName	Containment Zone Alerting Application
MaximumMarks	4 Marks

TechnicalArchitecture:

TheDeliverableshallincludethearchitecturaldiagramasbelowandtheinformationas perthetable1&table2



Guidelines:

1. Include all the processes(As an application logic/Technology Block)
2. Provide infrastructural demarcation(Local /Cloud)
3. Indicate external interfaces (third party API’s etc.)
4. Indicate Data Storage components/services
5. Indicate interface to machine learning models(if applicable)

Table-1:Components&Technologies:

S. No	Component	Description	Technology
1.	UserInterface	How user interacts with application e.g. Web UI, Mobile App, Chat bot etc.	HTML, CSS, Javascript & Bootsrap.
2.	Routing	Connect backend to the user interface	Python Flask
3.	Application Logic- 1	Access the user location and updating the new infected zone by admin	Python , Javascript
4.	Application Logic- 2	Access the user by sending the notification while enter into the marked zone using geo fencing.	IBM Watson Assistant, Python
5.	Database	Storing the coordinates the geo fencing	SQL
6.	CloudDatabase	To push the objects, files into the storage and use in the application	IBM DB2, IBM Storage object
7.	FileStorage	File storage requirements and stores in a binary data	IBM Block storage
8.	Google Map API	To integrate the Google map in our website	Java Script API, Places API, Geo coding API etc
9.	Infrastructure(Server/Cloud)	Application Deployment on Local System / Cloud Local Server configuration: Show the nearby infected zone by geo fencing Cloud Server Configuration: update the newly affected zone	Local, cloud Foundry, kubernetes etc.

Table-2:Application Characteristics:

S. No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Flask	Python Flask
2.	Security Implementations	Access permission for login and encrypt the user's password	Python Encryption algorithm - bycrpt
3.	Scalable Architecture	The application respond much more quickly	Python Flask
4.	Availability	The system should be handle many number of user's account	Python Flask
5.	Performance	Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN's) etc.	Python Flask framework