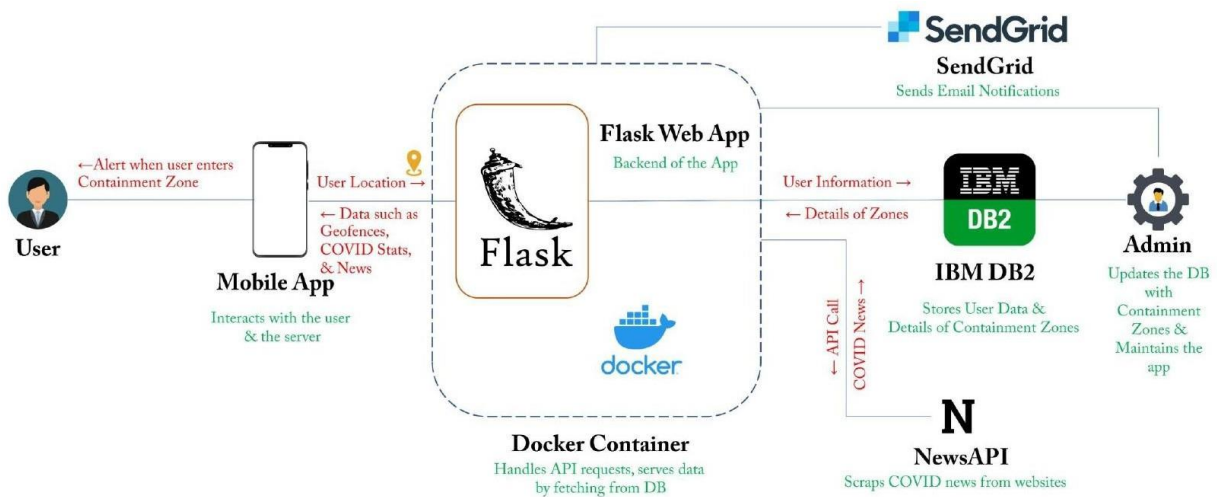


Project Design Phase-II Technology Architecture

Date	Oct 15, 2022
Team ID	PNT2022TMID07010
Project Name	Containment Zone Alerting Application



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.	UI (User Interface)	App user can view about the recent updated zones.	HTML, CSS, Bootstrap.
2.	Application Logic-1	The initial step of the application to access the location from the user device	Python
3.	Application Logic-2	By the access of the location user can see the nearby containment zone by geofencing.	IBM Watson STT service
4.	Application Logic-3	While entering into the containment zone the notification will push to their registered ema id	IBM Watson Assistant
5.	Database	Admin update the frequent changes of alerted Zones, for their purpose to show aware of the user.	MySQL, NoSQL, etc.
6.	Cloud Database	All the datum are push in the IBM cloud.	IBM DB2, IBM Cloudant etc
7.	File Storage	The files are stored in the Binary format.	IBM Block Storage or Other Storage Service or Local Filesystem
8.	External API-1	They can show of their Weather conditions from their imported dependencies.	IBM Weather API, etc.
9.	External API-2	For the purpose of User registration and their convenience for better interaction to get a better visual.	Aadhar API, etc.
10.	Machine Learning Model	For the easy segregation of data in the entered datum, model can plot to their respective columns	Object Recognition Model, etc.
11.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: shown the nearby zones Cloud Server Configuration: updated the newly alerted zones	Local, Cloud Foundry, Kubernetes, etc.

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	We used a Flask framework the development of python.	Flask
2.	Security Implementations	Session based authentication. Role management. Password hashing. Basic HTTP authentication. Token based authentication. Token based account activation (optional) Token based password recovery / resetting (optional) User registration (optional)	Flask Security.
3.	Scalable Architecture	This micro-framework modularizes the entire code and let developers work on independent chunks and use them as the code base grows.	Flask - python
4.	Availability	it does is precompute the output of an operation that usually takes a lot of time otherwise. Once this precomputed output is stored somewhere, the next user request doesn't involve rerunning all the subprocesses of that operation but instead just serving the precomputed output (instead of rerunning the same piece of expensive code	Flask Reliability
5.	Performance	Flask is the most popular micro-framework for web programming in Python. Known for its lightweight build and flexibility, it is a fan favorite amongst beginners because of how easy it is to get started with, especially for building prototypes and smallscale projects	Frame Work