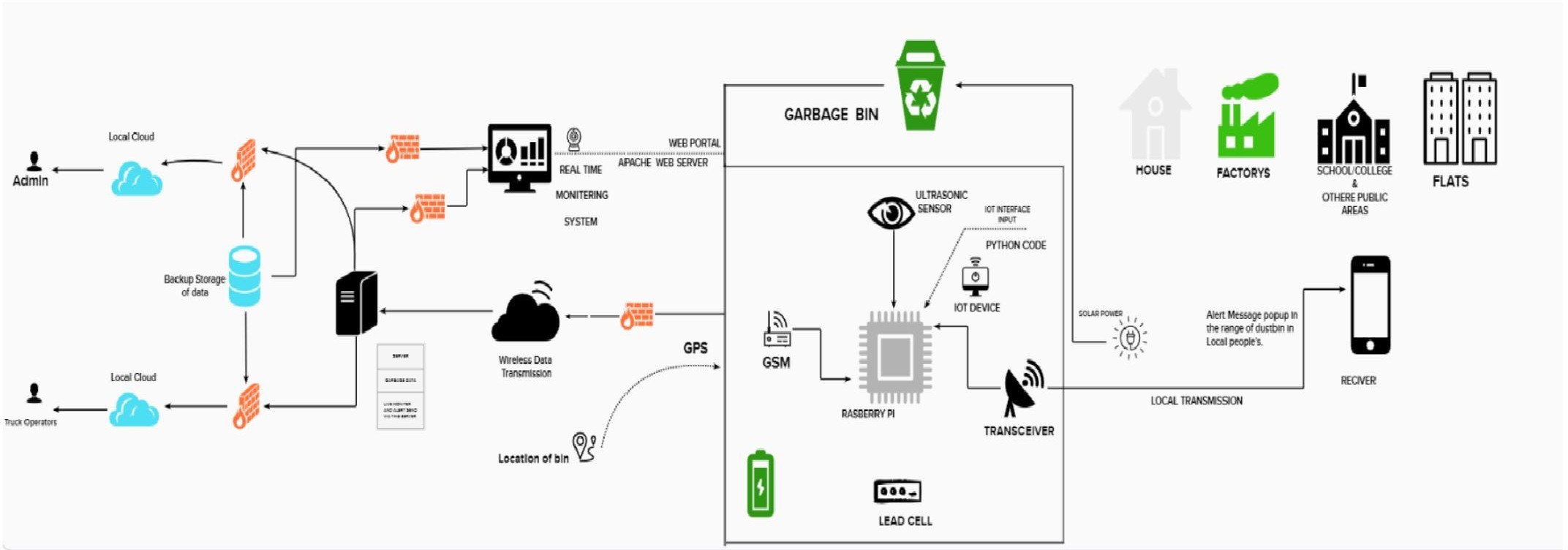
**Project Design Phase-II**

**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 28 October 2022 |
| Team ID | PNT2022TMID19657 |
| Project Name | Project - Smart Waste Management System For Metropolitan Cities |
| Maximum Marks | 4 Marks |

**Technical Architecture:**



**Table-1 : Components & Technologies:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Component** | **Description** | **Technology** |
|  | User Interface | How user interacts with application e.g.  Web UI, Mobile App, Chatbot etc. | HTML, CSS, JavaScript / Angular Js / React Js etc. |
|  | Application Logic-1 | Logic for a process in the application | Python |
|  | Application Logic-2 | Logic for a process in the application | IBM Watson service |
|  | Application Logic-3 | Logic for a process in the application | IBM Watson Cloud service |
|  | Database | Data Type, Configurations etc. | MySQL, NoSQL. |
|  | Cloud Database | Database Service on Cloud | IBM Cloudant. |
|  | File Storage | File storage requirements | Storage Area Network (SAN). |
|  | External API-1 | Purpose of External API used in the application | Location Tracking. |
|  | External API-2 | Purpose of External API used in the application | Waste Monitoring. |
|  | Machine Learning Model | Purpose of Machine Learning Model | Object Recognition Model. |
|  | Infrastructure (Server / Cloud) | Application Deployment on Local System / Cloud  Local Server Configuration:  Cloud Server Configuration : | Local, Cloud Foundry, Kubernetes, etc. |

**Table-2: Application Characteristics:**

| **S.No** | **Characteristics** | **Description** | **Technology** |
| --- | --- | --- | --- |
|  | Open-Source Frameworks | List the open-source frameworks used | Monitor and Clean. |
|  | Security Implementations | List all the security / access controls implemented, use of firewalls etc. | Encryptions. |
|  | Scalable Architecture | Justify the scalability of architecture (3 – tier, Micro-services) | To prevent the environment clean. |
|  | Availability | Justify the availability of application (e.g. use of load balancers, distributed servers etc.) | Available at any time. |
|  | Performance | Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN’s) etc. | Monitor and dispose the waste. |