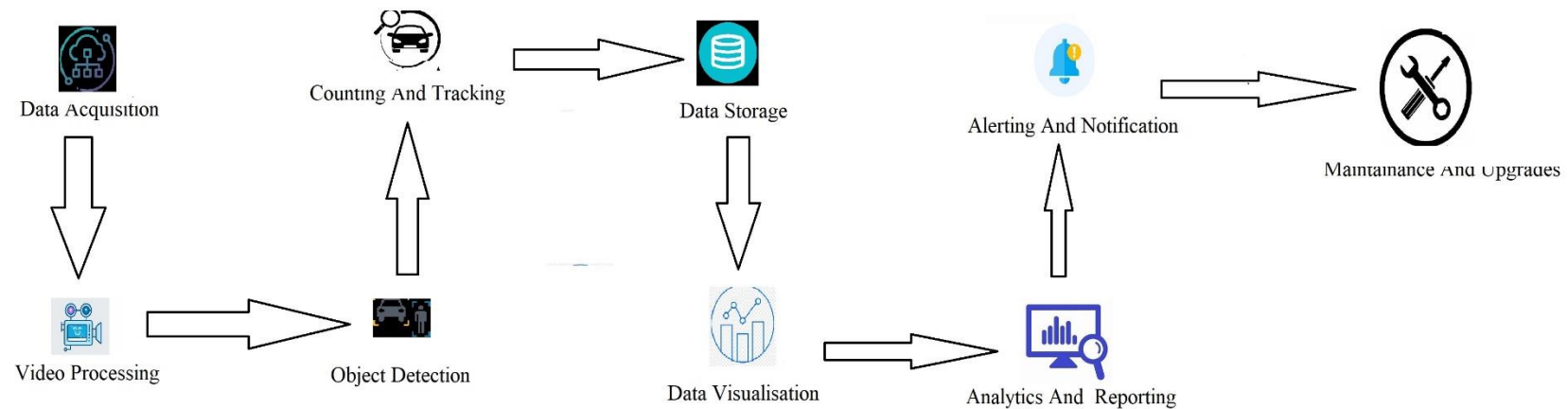


**Project Design Phase-II**  
**Technology Stack (Architecture & Stack)**

Date	18 May 2023
Team ID	NM2023TMID15643
Project Name	Intelligent people and vehicle counting system for secretariat.

**Technical Architecture:**

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



**Table-1 : Components & Technologies**

S.No	Component	Description	Technology
1.	User Interface	How user interacts with application e.g.Web UI, Mobile App, Chatbot etc.	Web UI using React.js
2.	Application Logic-1	Logic for a process in the application	Python
3.	Application Logic-2	Logic for a process in the application	IBM Watson STT service
4.	Application Logic-3	Logic for a process in the application	IBM Watson Assistant API
5.	Database	Data Type, Configurations etc.	MySQL
6.	Cloud Database	Database Service on Cloud	IBM Cloudant
7.	File Storage	File storage requirements	Local Filesystem
8.	External API-1	Purpose of External API used in the application	IBM Weather API
9.	External API-2	Purpose of External API used in the application	Aadhar API
10.	Machine Learning Model	Purpose of Machine Learning Model	Object Recognition Model, etc.
11.	Infrastructure (Server / Cloud)	Application Deployment on Local System / CloudLocal Server Configuration: Cloud Server Configuration:	Local Server Configuration: Apache Tomcat Cloud Server Configuration: AWS (Amazon Web Services)

**Table-2: Application Characteristics:**

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	List the open-source frameworks used	React.js
2.	Security Implementations	List all the security / access controls implemented,use of firewalls etc.	SHA-256 encryption, user authentication, role-based access control (RBAC), OWASP security practices.
3.	Scalable Architecture	Justify the scalability of architecture (3 – tier, Micro-services)	Microservices architecture using Docker and Kubernetes for containerization and horizontal scaling.
4.	Availability	Justify the availability of application (e.g., use ofload balancers, distributed servers etc.)	Load balancers, distributed servers, and redundant infrastructure.
5.	Performance	Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN's) etc.	Caching mechanisms, Content Delivery Network (CDN) for static assets, load testing, and query optimization.