

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

Date	03 October 2022
Team ID	PNT2022TMID01159
Project Name	Project – Hazardous area monitoring for industrial plant powered by IoT.
Maximum Marks	4 Marks

IoT architecture

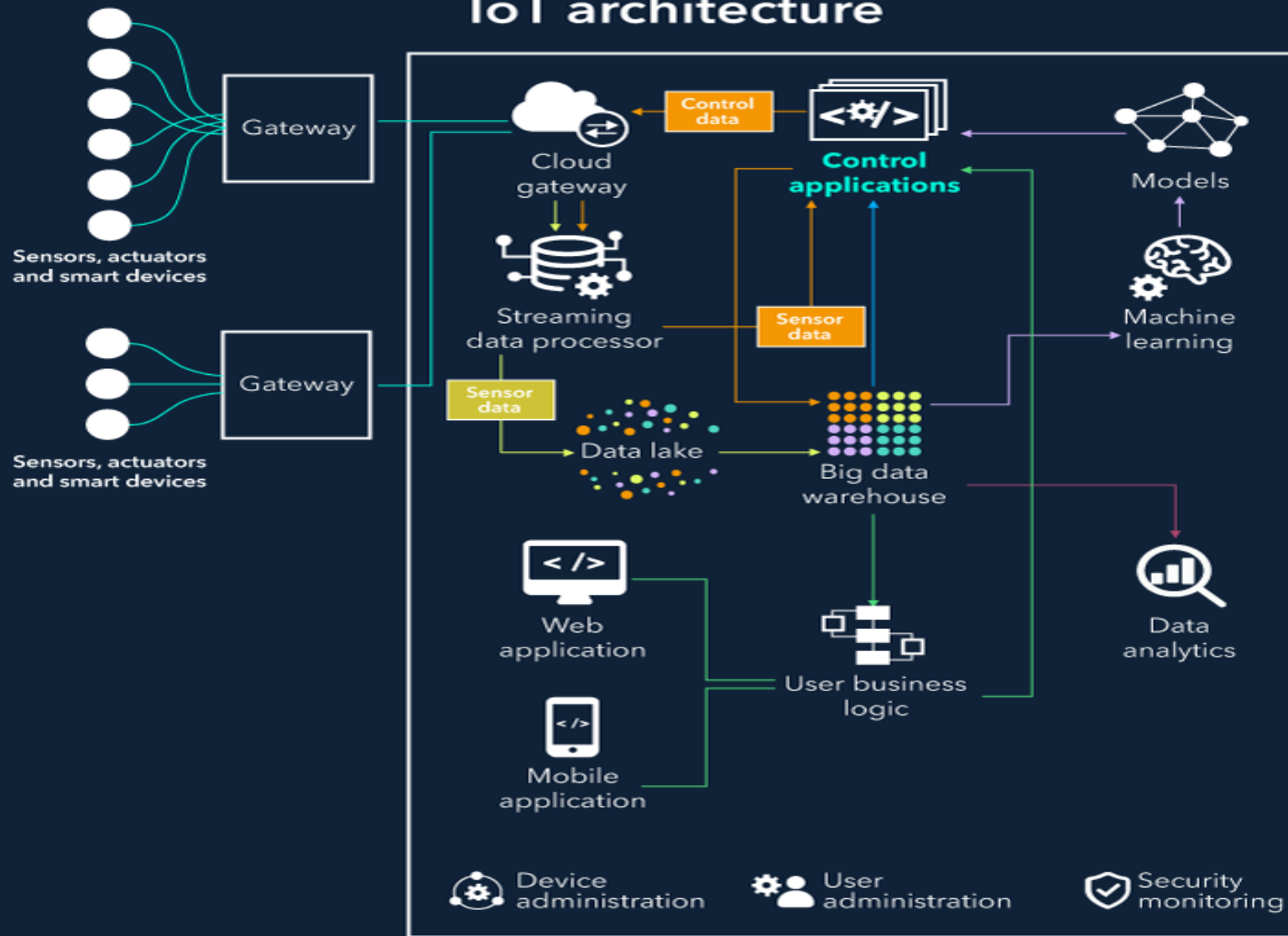


Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	Web UI, Mobile App, Chatbot , MIT inventor etc.	HTML, CSS, JavaScript / Angular Js / React Js etc.
2.	Application Logic-1	<p>IoT applications use machine learning algorithms to analyze massive amounts of connected sensor data in the cloud.</p> <p>Using real-time IoT dashboards and alerts, you gain visibility into key performance indicators, statistics for mean time between failures, and other information.</p>	Java / Python
3.	Application Logic-2	<p>The internet of things, or IoT, is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people that are provided with unique identifiers (UIDs) and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction.</p>	IBM Watson STT service

4.	Application Logic-3	<p>Watson Assistant lets you build conversational interfaces into any application, device, or channel.</p> <p>Add a natural language interface to your application to automate interactions with your end users.</p> <p>Common applications include virtual agents and chat bots that can integrate and communicate on any channel or device.</p>	IBM Watson Assistant
5.	Database	<p>IoT data comes in three different types, based on the device generating it and the use case.</p> <p>Status data: Status data is basic, raw data that communicates the status of a device or system.</p> <p>Automation data: This type of data is created by automated devices and systems such as smart thermostats and automated lighting.</p> <p>Location data: Location data communicates the geographical location</p>	MySQL, NoSQL, etc.

		of the device or system. It's frequently used in logistics, warehousing, and manufacturing.	
6.	Cloud Database	Cloudant handles software and hardware provisioning, management and scaling, and support.	IBM DB2, IBM Cloudant etc.
7.	File Storage	<p>IBM Cloud® Block Storage is persistent, high-performance iSCSI storage that is provisioned and managed independently of compute instances.</p> <p>iSCSI-based Block Storage LUNs are connected to authorized devices through redundant multi-path I/O (MPIO) connections.</p>	IBM Block Storage or Other Storage Service or Local Filesystem
8.	External API-1	<p>Runtime APIs</p> <ul style="list-style-type: none"> • Admin HTTP API <p>This HTTP-based API can be used to remotely administer the runtime. It is used by the Node-RED Editor and command-line</p>	IBM Weather API, etc.

		<p>admin tool.</p> <ul style="list-style-type: none">• <u>Hooks</u> <p>The Hooks API provides a way to insert custom code into certain key points of the runtime operation.</p> <ul style="list-style-type: none">• <u>Storage</u> <p>This API provides a pluggable way to configure where the Node-RED runtime stores data.</p> <ul style="list-style-type: none">• <u>Logging</u> <p>A custom logger can be used to send log events to alternative locations, such as a database.</p> <ul style="list-style-type: none">• <u>Context Store</u> <p>This API provides a pluggable way to store context data outside of the runtime.</p>	
--	--	---	--

		<ul style="list-style-type: none"> • <u>Library Store</u> <p>This API provides a pluggable way to add additional libraries to the Node-RED Import/Export dialog.</p>	
9.	External API-2	<p>Editor APIs</p> <p>The APIs available in the editor for nodes and plugins to use. This includes a set set of standard UI widgets that can be used within a node's edit template.</p> <p>Module APIs</p> <p>The APIs provided by npm modules that Node-RED is built from. These can be used to embed Node-RED into existing Node.js applications.</p>	Aadhar API, etc.
10.	Machine Learning Model	OpenCV (Open Source Computer Vision Library) is an open source	<ul style="list-style-type: none"> • Opencv

		<p>computer vision and machine learning software library.</p> <p>OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products.</p>	<ul style="list-style-type: none"> • Object Recognition Model, etc.
11.	Infrastructure (Server / Cloud)	<p>Application Deployment on Local System / Cloud</p> <p>Local Server Configuration:</p> <p>Cloud Server Configuration :</p>	Local, Cloud Foundry, Kubernetes, etc.

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	<p>It is an open-source IoT framework. The main purpose of the framework is data collection and device management.</p> <p>Further, it uses IoT protocols like HTTP, MQTT, and CoAP for device connectivity.</p> <p>It is also highly scalable as every type of device easily integrated.</p>	Thingsboard.
2.	Security Implementations	<p>Safety</p> <p>The IoT platform should never do something it isn't supposed to do. The principal game changer regarding software in the domain of IoT is safety coupled with accountability and responsibility. Any applied automation through an IoT solution means that we have faith in the system and trust that it will never do harm in the environment.</p>	e.g. SHA-256, Encryptions, IAM Controls, OWASP etc.

S.No	Characteristics	Description	Technology
		Security The IoT platform must ensure proper device management (via authentication and authorization mechanisms), data privacy, integrity, and confidentiality via secure communication and encryption of data. Security is especially crucial for an IoT platform, as it will rely more on automated security.	
3.	Scalable Architecture	Portability The IoT platform must be portable if it is destined to heterogeneous nodes. This may be achieved by leveraging virtualization technologies (for example, by using the Java Virtual Machine), or packing the deliverable into host operating system oblivious form (like the Docker image). Adaptability To support an extensive list of devices, and provide more service APIs for integration purposes, it is mandatory to have an adaptable IoT platform. The possible usage scenarios are vast, and	<ul style="list-style-type: none"> • Automated Bootstrapping • Better Control Over IoT Data Pipeline • Three-Axis Approach for Scaling • Reliable Microservices Architecture

S.No	Characteristics	Description	Technology
		<p>cannot be predetermined in advance.</p> <p>Usability To reduce the deployment hassle, and quickly get users up and running with an IoT platform, it must be in a user-friendly form in multiple aspects. This includes the management, supervision, and reporting facilities.</p> <p>Efficiency The IoT platform should ideally have a small footprint, employ advanced data storage technologies, and require adequate hardware resources to be usable in both real time and regular contexts. To move the computation near devices it should run on less capable hardware (for example, inside a smart meter or smartphone).</p>	

S.No	Characteristics	Description	Technology
4.	Availability	Load balancing is a core networking solution used to distribute traffic across multiple servers in a server farm. Load balancers improve application availability and responsiveness and prevent server overload.	<ul style="list-style-type: none"> • Load balancing • Distributors
5.	Performance	<p>Fog Computing is a new paradigm and an extension of Cloud Computing. This better performance results justifies the suitability of IoT applications using Fog-Based Cloud Network approach.</p> <p>Imperva and other CDNs can be used to reduce your website's latency, improving overall site performance and UX.</p> <p>Among other methods, this is done through: Content caching – CDNs cache and compress mirror versions of your web pages, which are then stored in strategically placed data centers.</p>	Fog Computing