IBM Cloud Services

IBM Watson IoT Platform:

This service is the hub for IBM Watson IoT and lets you communicate with and consume data from connected devices and gateways. Utilize the built-in web console dashboards to track and analyse your IoT data in real time. Then, by creating and integrating your own apps via messaging and REST APIs, you may improve and customise your IBM Watson IoT Platform experience.

FEATURES:

Connect:

Register and connect your devices and gateways quickly and securely. In our recipes section, you may discover straightforward step-by-step instructions for connecting widely used gadgets, sensors, and gateways.

Information Control:

You may decide what happens to the data that your linked gadgets send you. Manage data storage, set up data transformations, and connect to other platforms for data services and devices.

Real-time analysis:

Utilize dashboards, analytics, and rules to keep track of your real-time device data. In order to respond rapidly to important changes, define rules to monitor circumstances and launch automatic actions like as alarms, emails, IFTTT, Node-RED flows, and other services.

Security and risk management:

Through secure connectivity and application and user access management, our secure-by-design control features safeguard the integrity of your IoT system. To visualise significant threats and automate operational reactions with policy-driven mitigation measures, extend the base security with threat intelligence for IoT.

Node-RED Service:

Input Node: The IBM Watson Internet of Things Platform sends commands to the input node, which controls the devices. The node can establish a connection as a Device or a Gateway:

• Device: The node can be set up to either accept all commands for the Device or to only accept a certain sort of command.

• Gateway: The node can be set up to only receive commands for a certain subset of the devices connected through the gateway, or for all of them.

Result Node:

To the IBM Watson Internet of Things Platform, send device events. In registered mode or by utilising the Quickstart service, the node can establish a connection as either a Device or a Gateway. The connection will utilise the device type node-red-ibmwiotp and a randomly generated device id when utilising the Quickstart service, which can be specified in the node.

IBM Cloudant DB

Providing autonomous serverless scaling of the throughput capacity and storage that has been allocated, Cloudant is a fully managed JSON document database. For online, mobile, and IoT applications, Cloudant is compatible with Apache CouchDB and accessible via an easy-to-use HTTPS API.

Features:

• 99.99% SLA

• Secure: All data is encrypted both in transit and when stored. BYOK is optional, and Dedicated Hardware environments with private service endpoints. HA/DR: For in-region HA/DR, every Cloudant JSON document is kept in three copies. In places like Dallas, where availability zones are supported, Documents are stored in three different availability zones in the following locations: Washington, D.C., London, Frankfurt, Tokyo, and Sydney.

• Conformity: PCI, SOC2, GDPR, and ISO 27001 compliant. HIPAA compliance requires opting into a HIPAA-supported option and is only supported on Dedicated Hardware setups. Any Cloudant instance that is launched from the Frankfurt region will be in an environment that is supported by the EU.

• A 1MB maximum document size for JSON.

TTS Service:

The IBM Watson Text to Speech service offers APIs that make use of IBM's speech synthesis technology to convert text into realistic-sounding speech in a range of tongues, accents, and voices. For each language, the service offers at least one male or female voice, and occasionally both. With little lag, the audio is transmitted back to the client.

The service offers both a WebSocket and synchronous HTTP Representational State Transfer (REST) interface for speech synthesis. Both interfaces accept input in plain text and SSML. For speech-synthesis applications, SSML, an XML-based markup language, allows text annotation. Additionally supported by the WebSocket interface are the SSML <mark> element and word timings.