3

IoT Design Methodology

3.1 Design Methodology

The process of developing a product needs a proper execution of predefined design methodology. Generally, a product development process includes the following steps:

1. Describe the objective
2. Requirements to achieve the objective
3. Design the system architecture
4. Identify the stages of development
5. Assembling and coding each stage
6. Integrate all stages
7. Testing and troubleshooting
8. Debugging
9. Launch of product

Internet of Things (IoT) system design is the complete design of devices that are interconnected with each other. It combines physical and digital, both components to collect data from remote devices and deliver actionable signals.

It consists of various sensors, secured communication, gateway, cloud-enabled servers, and dashboards. All these components need to be designed with consideration of their interdependency.

The basic design principles are as follows:

Interoperability: It is the ability of a system to exchange the information and make use of it. The basic requirement components are sensor, machine, and device to communicate.

Information transparency: The connected devices over a network share the information. Information transparency is recording of the physical process and storing it virtually.

Technical assistance: Technical assistance is the ability of providing and displaying data of a connected system. It is to solve the issue and make the operational decisions easier, to improve the productability.

Decentralized decision: The decision-making is the principle for the connected system; it is required to execute the process with defined logic.

3.2 Challenges in IoT Design

IoT is a combination of many domains at a single platform, so the designing of an IoT system is quite challenging. The data security of customers is a big challenge for the service providers. If an IoT device suffers with connectivity problems due to a poor network, the purpose of deploying IoT is futile. It will be an even bigger issue when a large number of devices are connected. Heterogeneity of the network has different challenges with respect to security, privacy, and functionality.

A few of the challenges of IoT design are as follows:

1. **Availability:** Availability is the consistency of the network, even in case of attacks. As IoT services need to be real time, so security with availability is the prime concern.
2. **Authenticity:** It is the process where users need to prove the identity to access the services. It is required for the protection of the system. It is required to avoid illegal access of services.
3. **Confidentiality:** For data confidentiality, only an authorized person can access or modify the data.
4. **Integrity:** The data received by the user is undamaged, unmodified, and original as sent by the sender; this assurance is provided by integrity.
5. **Nonrepudiation:** It is the assurance of sending the correct information by end node without denying the sharing of data at any time and the acknowledgment from the receiver for the same.

3.3 IoT System Management

According to a study by International data corporation (IDC), the number of devices connected to the internet is expected to reach 30 billion by 2020. Management is the most important part of any system. The IoT system management includes the device deployment, provision of the device and authentication, configuration and control, monitoring and diagnostic, and then software updates and maintenance. IoT is not only about deploying the sensors and capturing the data to communicate to the server, but once the system is established, there may be a requirement of software updates, and repairing and replacing the faulty devices along with the security of data.

Provisioning and authentication: Authentication is a process of establishing the identity to ensure the security and trust. A cloud-hosted service is required to be implemented to check the authenticity of the software and hardware connected to the network.

Provisioning is a method to provide access of a device to a system with suitable authentication.

Configuration and control: Configuring a system means an arrangement of parts in a specific form, figure, or combination of elements. Configuring an IoT device includes attributes, such as name, location, and specific settings for application.

IoT devices need to be configured and authenticated from user attributes to make it reliable. The control is capability of handling the device and help for configuration changes.

Monitoring and diagnostics: Monitoring is the process of observing the progress of a system over a time period. The IoT system is connected with thousands of remote devices over the internet, and a small mistake in data monitoring may cause loss of trust of the customer. Even small issues need to be addressed with appropriate diagnosis of the problem. For troubleshooting, the developers need to implement good a program and must be capable of updating through cloud analytics.

Software maintenance and update: Software maintenance is another task in IoT that needs to support firmware, which should be free from any bugs. However, updating firmware is another important concern. The developer must have secure updated software, including the boot loaders.

Software maintenance on remote devices is a long-term process. A persistent and reliable connection is required with remote devices for maintenance and updating. This is a complicated process and needs to be performed when there is minimum impact on the business.

3.4 IoT Servers

Many cloud providers are in the market, which provides suitable IoT-based services for specific applications.

3.4.1 KAA

KAA is an IoT middleware platform and open-source framework for building smart connections for end-to-end IoT solutions with Apache License 2.0. It provides services for data analysis, visualization, and cloud service for IoT systems (http://www.kaaproject.org/).

3.4.2 SeeControl IoT

SeeControl is an IoT-enabled cloud platform that is efficient in data analytics and data visualization to maintain proper work flow in monitoring and controlling (http://www.seecontrol.com).

3.4.3 Temboo

Temboo is a cloud-based platform for application code generation. It involves less wiring and coding of hardware and software. It has more than 90 inbuilt libraries named "Choreos" for third-party services, including Yahoo weather, Twilio telephony, eBay product shopping, Flickr photo management, Amazon cloud, Twitter microblogging, Facebook Graph API, Google analytics, PayPal payment, Uber vehicle confirmation, YouTube video streaming, and many more (https://temboo.com).

3.4.4 SensorCloud

SensorCloud is an IoT cloud that provides the Platform as a Service (PasS) to gather, visualize, monitor, and analyze the information coming into sensors connected by wire or wirelessly. It allows the data to be analyzed with complex mathematical algorithms (http://www.sensorcloud.com).

3.4.5 Carriots

Carriots is platform that helps anyone to build quick IoT applications. It saves time, cost, and troubles. The PasS is designed to add features like remote device management and control, rule-based listeners' activity logging, triggering custom alarms, and data export (https://carriots.com).

3.4.6 Xively

Xively is a gravity cloud technology-based IoT cloud service. It helps companies manage their products by addressing various features like scalability, reliability, and secure. It is easy to integrate with devices but has minimum notification services (https://xively.com).

3.4.7 Etherios

Etherios supports comprehensive products and services for connected enterprises. Its cloud is designed on the PaaS model to enable users for connecting product and gain real-time visibility into their assets. It is a specialized cloud, but developers are restricted with limited devices (http://www.etherios.com).

3.4.8 thethings.io

thethings.io is a platform that gives a complete solution for the back-end developer with easy and flexible application program interfaces (APIs). thethings.io is a hardware agnostic that allows the connection of any device that is capable of using MQTT, CoAP protocols, HTTP, or WebSockets (https://thethings.io).

3.4.9 Ayla's IoT Cloud Fabric

Ayla IoT fabric is a PaaS-modeled enterprise class. Ayla Networks provide a firmware intermediator embedded in both devices and mobile device applications for end-to-end support. It provides easy mobile application development platform but is not suitable for small-scale developers (https://www.aylanetworks.com).

3.4.10 Exosite

Exosite is modular, enterprise-grade IoT software platform that helps to bring connected products to the market. It has a cloud platform based on IoT Software as a Service (SaaS), which provides real-time data visualization and analytics support to the users. The system development is easy with it, but it lacks in big data provisions (https://exosite.com).

3.4.11 OpenRemote

OpenRemote is an open source the IoT middleware solution, which allows users to integrate any device—protocol—design using available resources like iOS, Android, or web browsers. It supports open cloud services but has a high cost (http://www.openremote.com).

3.4.12 Arrayent Connect TM

Arrayent is an IoT platform that enables heterogeneous brands like Whirlpool, Maytag, and First Alert to connect users' products to smart handheld devices and web applications. Arrayent Connect Cloud is an IoT operating system that is based on the SaaS model (http://www.arrayent.com).

3.4.13 Arkessa

Arkessa provides services to companies to empower them with maximum revenue and to enhance customer satisfaction. It helps companies to develop IoT devices to enhance connectivity, monitoring, and controlling with enterprise. It has enterprise-enabled design facets, but its visualization apps are not proper (http://www.arkessa.com).

3.4.14 Oracle IoT Cloud

It is comprised four crucial parameters. It performs operations on received data including analysis, acquisition, and integration. It supports the database but lacks in connectivity of open-source devices (https://cloud.oracle.com/iot).

3.4.15 ThingWorx

ThingWorx is a data-driven decision-making cloud. It provides M2M and IoT services based on SQUEAL. Zero coding facility is available (https://thingworx.com).

3.4.16 Nimbits

Nimbits is a cloud server that provides solutions to edge-computing IoT-related services. It performs operations such as noise filtering and sends data on the cloud. It is easy to adopt but lacking in the real-time processing of query (http://www.nimbits.com).

3.4.17 InfoBright

InfoBright is an IoT-based analytical database platform that connects business to store and acts on machine-generated data for a complete ecosystem (https://www.infobright.com/index.php/internet-of-things).

3.4.18 Jasper Control Center

Jasper Control Center is a platform based on Jasper control. Control center is designed to automate the connected devices and help to analyze real-time behavior patterns. The main advantage is its rule-based behavior pattern (https://www.jasper.com).

3.4.19 AerCloud

AerCloud platform collects, manages, and analyzes sensory data for IoT and M2M applications. It is scalable to M2M services but not suitable for developers (http://www.aeris.com).

3.4.20 Echelon

Echelon is an IoT-based platform for the cloud with resources like microphones, hardware devices, and other applications. It is good for the industrial prospective but lacks in basics for beginners (http://www.iiot.echelon.com).

3.4.21 ThingSpeak

It is an open-source public cloud platform specially developed for IoT-based applications. It has open API that receives real-time data. It has data storage, monitoring, and visualization facilities (https://thingspeak.com).

3.4.22 Plotly

Plotly is a data visualization cloud service provider for the public. It provides data storage, analysis, and visualization services. Python, R, MATLAB, and Julia-based APIs are implemented in Plotly (https://plot.ly).

3.4.23 GroveStreams

GroveStreams is a public cloud for data visualization. It supports various data types. It enables seamless monitoring but lacks in statistical services (https://thingworx.com).

3.4.24 IBM IoT

IBM IoT is an organized architecture cloud platform. It supports complex industry solutions. It can enable a device's identity but the application prototyping is difficult (https://internetofthings.ibmcloud.com).

3.4.25 Microsoft Research Lab of Things

Lab of Things is an IoT platform design developed by Microsoft. It is used to analyze experimental research evidences in academic institutions (http://www.lab-of-things.com).

3.4.26 Blynk

It is an open-source platform with iOS and Android apps, which allows the control of Raspberry Pi and Arduino over the internet. It supports a graphical interface to build projects just by dragging the widgets. It supports many IoT modules.

3.4.27 Cayenne APP

Cayenne is an app for smartphones and computers that controls Raspberry Pi and Arduino through the use of a graphical interface. It has customizable dashboards with drag-and-drop widgets for connection devices. It supports quick and easy setup.

3.4.28 Virtuino APP

Virtuino platform creates amazing virtual screens on smartphones or tablets to control the automation system created with Arduino or similar boards. It supports Arduino and can be connected with the HC-05 Bluetooth, Ethernet Shield, and ESP8266 modules. It supports monitoring sensor values from the IoT server ThingSpeak.