IOT Design Methodology

Submitted To,

Ms.Kochumol Abhraham

Assistant Professor

PG Department of Computer Applications

Submitted By,

Ambika M.K

23PMC108

What Is the Internet Of Things (IoT)?

“The Internet of Things is the physical Internet. It is a network of physical objects such as motor vehicles, appliances, and other physical objects linked to the Internet and has sensors, software, and network connectivity installed to gather and exchange data.”

IoT products come in a plethora of shapes and sizes. Some examples include smart microwaves that cook food automatically in the appropriate amount of time. Driverless or self-driving cars use precise sensors to identify objects in their path, and wearable fitness trackers that monitor your heart rate and daily step count.

Internet Of Things IoT Design Methodology

Since IoT systems require interactions between multiple components, designing them can be a difficult and complex undertaking. There are numerous options available for every component. IoT designers frequently tend to create systems with certain goods in mind. The following the IoT design methodology steps are below:-

Purpose And Requirement

Defining the system’s requirements and purpose is the first stage. This step captures the requirements, behavior, and purpose of the system. requisites may include:

- requirements for data collecting
- Requirements for data analysis
- prerequisites for system management
- standards for security
- Requirements for user interface

Procedure Detail

The process definition is defined in the second step of the loT design approach. The goal and requirement specifications are the source of the formal description of the IoT system's use cases in this step.

Domain Model Specification

This is the third step of the IoT platform design methodology. The domain model represents the primary concepts, entities, and objects in the domain of the IoT system that will be created. The domain model specifies the properties of the objects and their connections. The domain model is not dependent on any one platform or technology.

System designers can comprehend the IoT domain for which the system is intended by using domain models

.

Information Model Specification

In the fourth step of loT's design process, the information model is defined. The information model defines the full information architecture of loT systems, including relationships, virtual entity properties, and other structures. Information models don't go into great into regarding the particulars of how data is stored or displayed.

Service Specification

The services in the Internet of Things system, including service types, inputs/outputs, endpoints, schedules, preconditions, and effects, are defined in this step by the service specifications.

These services either retrieve the current values or modify the state or attribute values. The Mode service is a RESTful web service that may be used to retrieve the current mode (GET request) or set the mode to auto or manual (PUT request). The mode is updated to/retrieved from the database.

The State service is a RESTful web service that allows users to receive the current light state (GET request) or set the light appliance status to on or off (PUT request).

The status is either updated in the status database or retrieved from it. On the device, the Controller service functions as a native service.

Functional View Specification

The IoT systems' functions are categorized into different functional groups according to the functional view. Every functional group offers information about the ideas as well as functionalities for interacting with them in the domain model.

A functional view comprises the following functional groups: Application, Security, Management, Device, Communication, and Services.

Operational View Specification

This stage defines the operational view specifications. The deployment and operation of IoT systems are outlined, including options for service hosting, storage, device hosting, application hosting, and so on.

Device And Component Integration

This step involves integrating the devices and component designs, including the relay switch actuator, LDR sensor, and minicomputer.

Application Development

This is the last phase in the IoT design methodology. It is to create an IoT application. The application has controls for the mode (auto-on or auto-off) and the light (on or off).

What is Edge Computing?

Edge computing refers to the practice of processing data near the edge of the network, where the data is generated, rather than relying on a centralized data-processing warehouse. It involves deploying devices and resources at or near the sources of data to process and analyze information in real-time or near real-time.

Key Components

1. **Edge Devices**:
   - These are the hardware components at the edge of the network, such as sensors, IoT devices, gateways, and routers.
   - They generate data and often have some capability to process or pre-process data locally.
2. **Edge Servers**:
   - Local servers or micro data centers that provide more processing power and storage closer to the data source.
   - They can handle more complex processing tasks that edge devices might not be capable of.
3. **Edge Gateways**:
   - Devices that bridge the communication between edge devices and the cloud or central data centers.
   - They can perform protocol conversion, data aggregation, and local data processing.

Benefits of Edge Computing

1. **Reduced Latency**:
   - By processing data locally, edge computing significantly reduces the time it takes for data to travel to a central data center and back.
   - This is crucial for applications requiring real-time responses, such as autonomous vehicles, industrial automation, and augmented reality.

2. **Bandwidth Efficiency**:
    - Edge computing reduces the amount of data that needs to be transmitted to central data centers by filtering, aggregating, and pre-processing data locally.
    - This helps save bandwidth and reduces costs associated with data transmission.
3. **Improved Reliability**:
    - Local processing ensures that applications can continue to function even if connectivity to the central data center is lost.
    - This increases the reliability and resilience of the system.
4. **Enhanced Security and Privacy**:
    - Sensitive data can be processed locally, reducing the risk of exposure during transmission to a central data center.
    - This is particularly important for applications involving personal or sensitive information, such as healthcare and financial services.
5. **Cost Savings**:
    - By reducing the need for large-scale data transmission and central processing, edge computing can lower operational costs.
    - It also allows for more efficient use of network resource