3.1 INTRODUCTION

- In chapter 2 we have studied about IoT, physical design of IoT, logical design of IoT, different application and deployment of levels of IoT.
- So the Designing IoT systems can be a complex and challenging task. IoT systems involve interactions between various components such as IoT devices, network resources, web services, analytics components, application and database servers.
- The role of IoT designer is to design IoT system to keeping specific products/services in mind.
- The proposed methodology have reduced design, testing and maintenance time, better interoperability and reduced complexity of IoT System.

3.2 IOT DESIGN METHODOLOGY

UQ. Explain the various steps in IoT design methodology? **(SPPU – Mar 18, May 18, 4 Marks)**

IoT Design Methodology that includes :

1. Purpose and Requirements Specification
2. Process Specification
3. Domain Model Specification
4. Information Model Specification
5. Service Specifications
6. IoT Level Specification
7. Functional View Specification
8. Operational View Specification
9. Device and Component Integration
10. Application Development

To explain every step of IoT design methodology we will consider one example, **IoT based Weather Monitoring System**.

3.2.1 Purpose and Requirements Specification

UQ. Explain purpose and requirements specifications step of IoT system design methodology, consider smart IoT - based home automation system as an example. **(SPPU – Mar 18, May 19, 5 Marks)**

The first step in IoT system design methodology. In this step it defines the purpose, behavior and requirements of the system. This step also defines the

(1) Data collection requirements
(2) Data analysis requirement
(3) System management requirements
(4) Data privacy and security requirements
(5) User interfaces requirements

Consider our example **IoT based Weather Monitoring System**, the purpose and requirement of system are :

(1) **Purpose :** Purpose of this system is to collect data on environmental conditions such as temperature, pressure, humidity and light in an area using multiple end nodes.

(2) **Behaviour :** Weather alert can be sent to the subscribed users with the help of such application.

(3) **Data collection requirements :** The end nodes send the data to the cloud where the data is aggregated and analyzed.

6. **Application :** This FC includes an application that provides an interface to the users. It is also used to control and moniter various aspects of the IoT systems.

The Fig. 3.2.6 shows the Mapping deployment level to functional groups for the weather monitoring system.

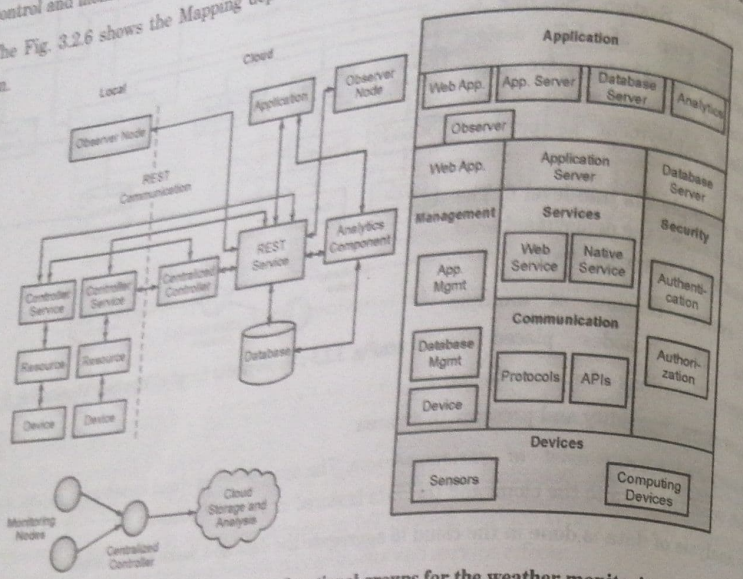


Fig. 3.2.6 : Mapping deployment level to functional groups for the weather monitoring system

In above diagram

- IoT device maps to the Device FG and management FG (Device Mgmt.)
- Database maps to the Database FG (Database Mgmt.) and Security FG.
- Resources maps to the Device FG and Communication FG.
- Controller Service and Web service maps to the Service FG.
- Analytics component maps to the Application FG.
- Observer node maps to the Application FG.
- Application maps to the Application FG, Management FG and Security FG.

3.2.8 Operational View Specification

GQ. Explain in detail operational view specification.

In this step, it defines the various options suitable to the IoT system deployment and operation. These can be a service hosting options, storage options, device options, application hosting options, etc.

Operational view Weather Monitoring System is as follows :

. Computing Device : Raspberry Pi, Temperature, Pressure, Light and Humidity Sensor.

Communication APIs : REST service API

3. Communication Protocol : Link Layer - 802.11, Network Layer - IPv4/IPv6, Transport TCP, Application – HTTP.
4. Native Service : controller service
5. Web Application : Django Web Application, Application Server - Django App Server, Database Server – Xively Cloud Storage, Analytics: Hadoop, Observer- Cloud App, Mobile App.
6. Security : Authentication: Web App, Database, Authorization: Web App, Database
7. Management : Django App Management Database Management - MySQL DB Management, Device Management - Raspberry Pi device Manage

3.2.9 Device and Component Integration

GQ. What is device and component integration?

- The ninth step in the IoT design methodology is the integration of the devices and components.
- The schematic diagram of the weather monitoring system is shown below.

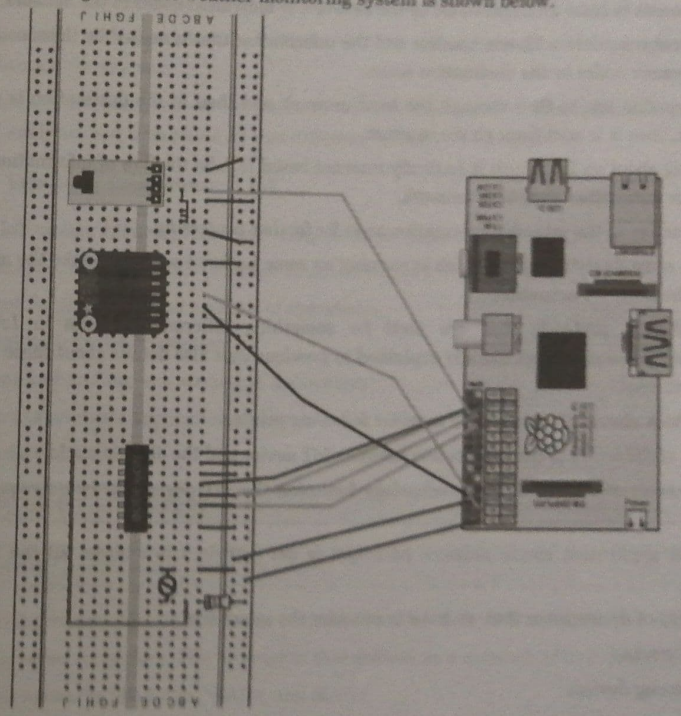


Fig. 3.2.7 : Device and Component Integration of Weather Monitoring System

- The devices and components used in this example are Raspberry Pi minicomputer, temperature sensor, humidity sensor, pressure sensor and LDR sensor.

Unit III End Sem.

3.2.10 Application Development

This is the last step of IoT design methodology. In this we develop the IoT system.

3.3 BASICS OF IOT NETWORKING

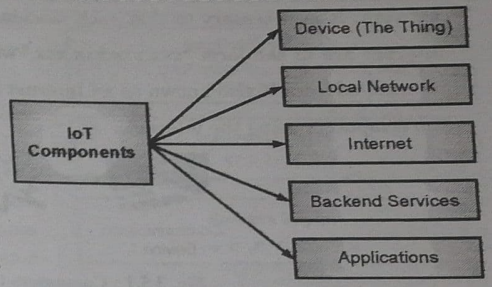
Q. What is Network?

- We know that that IoT has evolved a lot. An IoT as a very complex system involving sensors, actuato networks, local area, wide area internet and different servers, different algorithms, machine learni and so on.
- All these executing together to make the system function as one single entity.
- In IoT based system consist of physical objects and that are fitted with different sensors. These sens basically sense different physical event or information that is occurring around them.
- These sensor are fitted things, sensors actuators and different other devices. These are one compon of the IoT.
- These components become different nodes in the network or individual nodes in the network.
- These nodes communicate with one another and the information that is sensed by these sensors is se to the other sensor nodes or the destination nodes.
- So this information has to flow through the local network and then, if the destination is outside t local network, then it is sent through the internet.
- We are talking about an IoT which is basically internet based IoT. So the flow of information is throug the internet or some other wide area network.
- Finally, it is arrive at the intended destination node for further processing.
- There can be some analytic engine which is running on some backend server to make the analytics an decision can be made for actuation.
- So there different protocols that are used for something different purposes in IoT. The Io communication protocol concept already explained in previous unit also details about these protocols w will see in next unit.
- So, when we talk about IoT network we consider following points.
  1. Network architecture is communication between IoT device and the outside world.
  2. Correct choice of communication technology indicates the IoT device hardware requirement an costs.
  3. IoT based application single network paradigm is not sufficient to address all the needs of Io device.
  4. Complexity of networks in that we have to consider the issues like
- Growth of networks
- Interfacing among devices
- Network Management
- Heterogeneity in network
- Protocol standardization with network.

3.4 NETWORKING COMPONENTS

GQ. Explain Networking components which used for IoT system.

- In previous section we discussed about basics of IoT networking. Now in this section for IoT system, we will discuss networking components.
- The following Fig. 3.4.1 shows the IoT based Networking component.
- We have different things. These things can be different physical objects which are fitted with different sensors. For example things like telephones, lightning systems, cameras, different other scanner, sensors like the temperature sensor and so on.



(ICa)Fig. 3.4.1 : Basic IoT Components

- These things are able to communicate with one another with the help of wireless technologies like Zigbee, Bluetooth, Wi-Fi and so on.
- This information from these devices, will pass through a local network and from a local network, they will go through the internet.
- These data are basically sent to the backend services concerning different server's processors to run different analytics and then based on that different devices can be actuated.

3.4.1 Functional Components of IoT

For building IoT system following functional component are important.

1. Component for interaction and communication with other IoT devices
2. Component for processing and analysis of operations.
3. Component for internet interaction
4. Component for handling web services of application.
5. Component to integrate application service.
6. User interface to access IoT.

3.5 INTERNET STRUCTURE

GQ. What is Internet? Explain structure of Internet.

Internet is interconnection of worldwide computers in the form of a network. The user can access the knowledge from other computers. The internet is also defined as a network of networks. There are different variety of public networks like LANs, WANs, and MANs.

Structure of the internet

The structure of internet is divided into three parts.

1. Internet Address

- Computers connected to the internet means that the systems are connected to computers' worldwide network. It is necessary to that each machine/device has its own or unique address. Addresses of the internet are in the form "xxx.xxx.xxx.xxx," where each "xxx" ranges from 0-256. The structure of the internet address is also known as an Internet Protocol address. Fig. 3.5.1 shows the connection between two computers using the internet

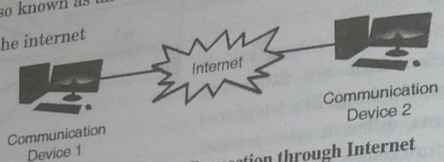


Fig. 3.5.1 : Computers Connection through Internet

- Using Internet Service Provider (ISP) a client connects the computer with the internet. The client's system is allocated a temporary internet protocol address till the communication. However, if someone with internet using a local area network (LAN), the client is probably assigned to a permanent internet protocol. At the time of connection, the system will have a unique internet protocol address. Through command "Ping" to ensure the internet connection on the system. This facility is available on all the Microsoft Windows operating systems and sometimes on a flavor of Unix OS.

2. Protocol Stack and Packets

- As the device is connected to the internet with a unique address. The next thing, what is the procedure to communicate the device with the system at another end? For understanding purpose, we are considering an example. As we discussed in Fig. 3.5.1 one system holds an IP address, i.e., 173.196.95.98, and the second system contains an IP address, i.e., 162.194.60.98. Suppose client want to send a message "Hello XYZ" to another. In this example medium of communication is wire that connects "Your computer" to the internet.
- If you are using ISP facilities, then the message will be communicated via phone line of ISP. In that case, the first message will be encrypted in digital form. All the alphanumeric characters will be converted into an electronic signal. The electronic signal will be delivered to the other computer and then again decrypted into the original form as received on the second IP system.

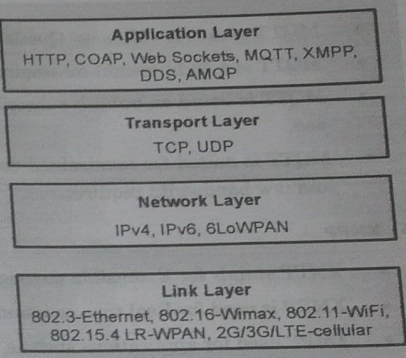
3. Complete Infrastructure

- The framework of the internet consists of multiple interconnected large networks. That large networks called as Network Service Providers (NSPs). The internet structure is built up of packets and routers. Addresses are embedded in the headers of packets.

3.6 CONNECTIVITY TECHNOLOGIES

GQ. What are different connectivity technologies used for IoT system.

- In previous section we have understood different basic concepts that are involved in the networking aspects of IoT.
- In this section we will discuss different connectivity technologies used in IoT system. As we know that IoT device communicates with each other in a network. For this communication we need some protocols. There are various protocols available for Iot system. These protocols are used to establish communication between a node device and server over the internet.
- The following fig shows the structure of different protocols used in various layer. We will discuss the protocols in each layer short.



(IC9)**Fig. 3.6.1 : IoT Protocols**

(a) Application Layer

GQ. Explain different Application Layer protocols.

In this layer, protocols define how the data can be sent over the network with the lower layer protocols using the application interface. The protocols HTTP, COAP, Web Socket, XMPP, MQTT, DDS, and AMQP are used in application layer. In this section we will discuss each protocol.

(1) HTTP

- HTTP stands for Hypertext Transfer Protocol(HTTP)
- Forms foundation of World Wide Web(WWW)
- HTTP includes commands for communication such as GET,PUT, POST, HEAD, OPTIONS, TRACE..etc
- HTTP Follows a request-response model
- HTTP Uses Universal Resource Identifiers(URIs) to identify HTTP resources

(2) COAP

- COAP stands for Constrained Application Protocol (COAP)
- COAP is used for Machine to machine (M2M) applications.
- COAP is used as web transfer protocol for IoT and uses request-response model
- COAP uses client –server architecture
- COAP also provides methods for communication such as GET, POST, PUT and DELETE.

(3) Web Socket

- Web socket is a TCP type of communication protocols.

(1) Ethernet

- Ethernet is the most popular physical layer LAN technology.
- 802.3 - Ethernet is the traditional technology for connecting devices in a wired local area network (LAN) or wide area network (WAN).
- A standard Ethernet network can transmit data at a rate up to 10 Megabits per second (10 Mbps).

(2) WiMAX

- WiMAX stands for Worldwide Inter-operability for Microwave Access.
- 802.16-Wimax wireless broadband connection medium.
- Data rate is 1.5 Mb/S to 1 Gb/S

(3) Wi-Fi

- 802.11-WiFi most popular protocol for Wireless Local Area Network.
- Data rate of 1Mb/S to 6.75 Gb/S can be achieved using these standard.

(4) LR-WPAN

- 802.15.4 LR-WPAN stands for Low Rate- Wireless Personal Area Network
- Data rate of 40Kb/S to 250 Kb/S can be achieved using WPAN standards.
- LR-WPAN is low power communication protocol.
- The example of such protocol is Zigbee - to enable low-cost, low-power wireless machine-to-machine (M2M) communication.

(5) 2G/3G/LTE-cellular

- These standards provide communication over cellular network.
- 2G/3G/4G standards used for mobile communication.
- Data rates of 9.6 Kb/S for 2G devices upto 100 Mb/S for 4G devices can be achieved using these standards.

3.7 IOT COMMUNICATION MODELS

GQ. What are IoT communication models?

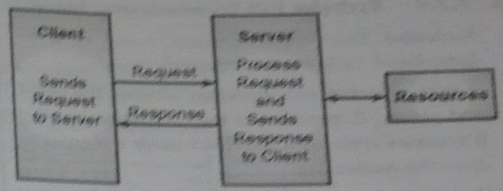
UQ. List various IoT communication models. **(SPPU – Mar 18, 2 Marks)**

In IoT communication can be carried in different ways. So following section describes different communication model in detail.

3.7.1 Request- Response Communication Model

- As per name Request-Response communication model, the client sends requests to the server and the server responds to the requests.
- When the server receives a request, it processes the request, retrieves resource representations, prepares the response, and then sends the response to the client.

- Request-Response model is a stateless communication model so each request-response pair is independent of others.
- The Fig. 3.7.1 shows the structure of Request-Response communication model

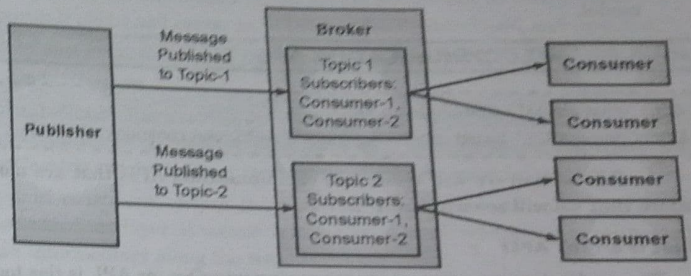


(IC10)**Fig. 3.7.1 : Request- Response communication model**

3.7.2 Publish–Subscribe Communication Model

UQ. What is Publish-Subscribe Communication Model **(SPPU – May 19, 5 Marks)**

- Publish-Subscribe communication model has three elements publishers, brokers and consumers.
- Publishers are the source of data in communication. Publishers send the data to the topics. Broker is managed that data.
- Broker has record of all subscribed consumer as per the topic.
- So once the data is received from publisher, broker sends the data to all subscribed consumer.
- In this model publisher are not aware about the consumer.

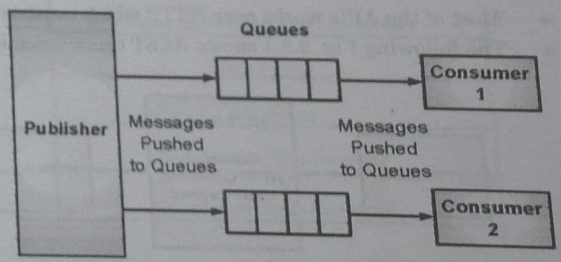


(IC11)**Fig. 3.7.2 : Publish–Subscribe Communication Model**

- The Fig. 3.7.2 shows the structure of Publish- Subscribe communication model

3.7.3 Push–Pull Communication Model

- In Push-Pull communication model, in which the data producers push the data to queues and the consumers pull the data from the queues.
- In this communication model queues are used to separate out the single producer consumer communication.
- Like previous model producers do not need to be aware of the consumers.
- In this queues also act as a buffer.
- Queues used in such situations when there is a mismatch between the rate at which the producers push data and the rate at which the consumers pull data.

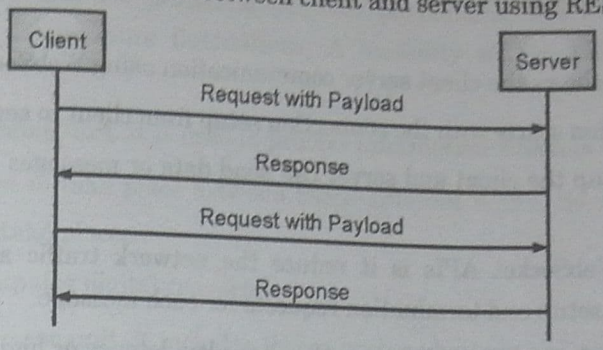


(IC12)**Fig. 3.7.3 : Push Pull Communication Model**

- The Fig. 3.7.3 shows the structure of Push-Pull communication model

Unit III End Se

- The Fig. 3.8.2 shows the communication between client and server using REST.



(1C15)**Fig. 3.8.2 : Client Server communication using REST**

REST based APIs follows the following characteristics or constraints.

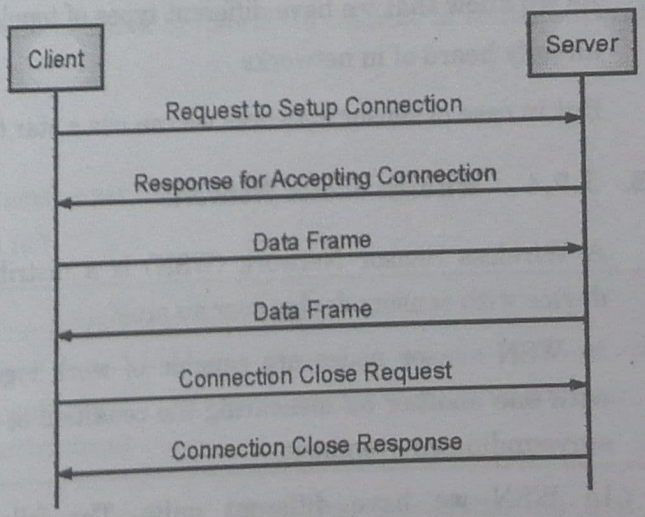
1. **Client-Server :** The principle behind the client-server constraint is the separation of concerns. Clients should not be worry about storage of data in server. Similarly, the server should not be worry about the user interface of the client. So separately client and server to be independently developed and updated.
2. **Stateless :** Each request from client side must contain all the information necessary to understand the request. So as per request server sends response
3. **Cache-able :** Cache constraint can be labeled as cache-able or non-cache-able. If a response is cache-able, then a client cache is given the right to reuse that response data for later, equivalent requests. To improve efficiency and scalability Caching can partially or completely eliminate some interactions.
4. **Layered System :** Layered system constraint, define the behavior of components such that each component cannot see beyond the immediate layer. it means a client cannot tell whether it is connected directly to the end server, or to an intermediary along the way.
5. **Uniform Interface :** The communication between a client and a server must be uniform this constraint is achieved by using Uniform Interface constraint. Resources are identified in the requests by URIs in web based system and which is uniform.
6. **Code on demand :** This constraint provides provide executable code or scripts for clients to execute in their context. This constraint is the only one that is optional.

Unit III End Sem.

3.8.2 WebSocket-based Communication API

UQ. With the help of appropriate diagram explain WebSocket-based communication APIs **(SPPU – Dec. 19, 3 Marks)**

- WebSocket APIs is bi-directional, full duplex communication between clients and servers.
- WebSocket APIs follows the exclusive pair communication model.
- Unlike Request - Response APIs such as REST, the WebSocket APIs do not require a new connection to be setup for each message to be sent.



(1C16)**Fig. 3.8.3 : WebSocket-based Communication API**

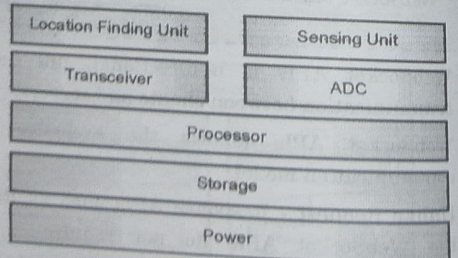
- Client to the server communication through the WebSocket communication begins with a connection setup request.
- The following Fig. 3.8.3 shows the client server communication using WebSocket.
- WebSocket communication starts with the connection setup from client to server.
- After the connection setup the client and server can send data or messages to each other in full-duplex mode.
- Advantages of using WebSocket APIs is it reduce the network traffic and latency as there is no overhead for connection setup and termination requests for each message.
- WebSocket APIs is suitable for IoT applications that have low latency or high throughput requirements.

3.9 SENSOR NETWORK

- Sensor network is a very important technology and one of the most important enablers of IoT that is used for building IoT.
- Sensors, transducers, actuators these are all very important things for understanding of IoT systems.
- Sensors connect with one another; we can obtain important information continuously, in real time remotely, from a larger environment. This is the benefit of sensor network.
- In sensor networks we have individual sensors, which are embedded in something known as sensor devices or sensor nodes, or sometimes also known as sensor modes.
- So, these modes or nodes or devices they have one of their components which is the sensor, and they have other components as well.
- So, these components taken together they comprise that particular node or the device which can help them to communicate.
- One device communicates with another device, that device communicates with another device, the third device with a fourth, fourth with the first and so on.
- As we know that we have different types of topologies. We can have all sorts of topologies that we have already heard of in networks
- But in case of sensor networks we can use a star topology.

3.9.1 Wireless Sensor Network

- A Wireless Sensor Network (WSN) is a distributed device with sensors deploy over an area.
- In WSN sensor nodes are capable of work together with one another ad measuring the condition of their surrounding environment.
- In WSN we have different units. The following Fig. 3.9.1 shows component of WSN.
- In WSN, the first thing required is sensing unit. The



(IC17)**Fig. 3.9.1 : Basic components of Wireless Sensor Node**

sensing unit basically senses the particular physical phenomena. For example temperature sensor would be sensing the temperature fluctuations; A humidity sensor will be sensing the humidity fluctuations.

- Then next unit is processing unit. It is used to process information which is sensed by sensor.
- Third is communication to take place between these different nodes. So with the help of transceiver devices communication take place.
- Then a next unit is the analog digital converter.
- At last the unit is the power unit, it includes things such as battery and so on, which is going to power these devices.
- So, these are the different units and we have other optional units such as the location finding systems for example, GPS etcetera.

3.10 FOUR PILLARS OF IOT

UQ. Draw and explain the four pillars of IoT paradigms (SPPU – Mar 18, 4 Marks)

- The four pillars of IoT are M2M (Machine to Machine), RFID (Radio Frequency Identification), WSN (Wireless Sensor Network) and SCADA (Supervisory Control and Data Acquisition).
- In next section we will see details about each pillar.

3.10.1 M2M

UQ. Explain M2M (the internet of device) pillar of IoT. (SPPU – Dec. 18, 4 Marks)

- M2M literally means 'Machine to Machine' communication.
- M2M enables flow of data between machines which monitors data by means of sensors.
- At other end gathered data is extracts the information and processes it.
- M2M mostly uses cellular wireless networks, sometimes wired or hybrid, to connect to central server (software program).
- M2M also called as a Subset of IoT.
- The following Table 3.10.1 shows the M2M applications as per Industry.

Table 3.10.1

Sr. No.	Industry / Vertical	M2M applications
1.	Automotive	Passenger vehicle anti theft / recovery, monitoring /maintenance, safety / control, entertainment
2.	Transportation	Fleet management, asset tracking, telematics, manufacturing and logistic

Sr. No.	Industry / Vertical	M2M applications
3.	Utilities/ Energy	Smart metering, smart grid, Electric line monitoring, gas /oil / water pipeline monitoring.
4.	Security	Commercial and home security monitoring, Surveillance applications, Fire alarm, Police / medical alert
5.	Financial	Point of sale (POS), ATM, Kiosk, Vending machines, digital signage and handheld terminals
6.	Healthcare	Remote monitoring of patient after surgery (e-health), remote diagnostics, medication reminders, Tele-medicine
7.	Public Safety	Highway, bridge, traffic management, homeland security, police, fire and emergency services.

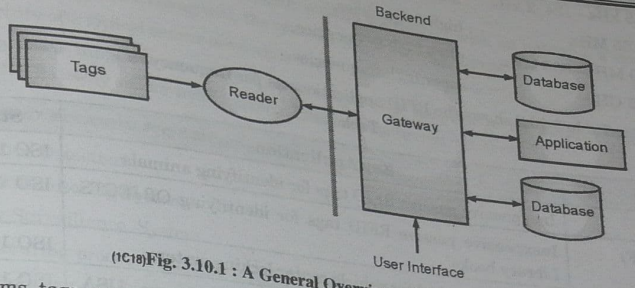
- For example, if your train is cancelled due to poor weather, a smart alarm clock would determine the extra time you'll need to take a different route, and wake you up early enough so that you're not late for work.
- Another example M2M is Smart Home, a connected thermostat can automatically switch the heating on when room temperature falls below a certain point. You might also have a remote-locking system enabling you to open the door to a visitor via your smart phone if you're not at home.
- Key features of M2M communication system are given below:

(a) Low Mobility
(b) Time Controlled
(c) Time Tolerant
(d) Packet Switched
(e) Online small Data Transmissions
(f) Monitoring
(g) Low Power Consumption
(h) Location Specific Trigger

3.10.2 RFID

- Second pillar of IoT is RFID that is Radio Frequency Identification.
- This RFID tag is a simplified, low-cost, disposable contactless smartcard.
- RFID tags include a chip that stores a static number (ID) and attributes of the tagged object. It also include antenna that enables the chip to transmit the store number to a reader.
- An RFID system involves hardware known as readers and tags, as well as RFID software or RFID middleware.
- Uses radio frequency to read and capture information stored on a tag attached to an object.
- A tag can be read from up to several feet away and does not need to be within direct line-of-sight of the reader to be tracked.
- It uses NFC (Next Field Communication protocol), IC (Integrated Circuit) Cards, and Radio Waves.
- RFID mostly used in many industries for tasks such as personal tracking, access control, supply chain management and so on.

- The following Fig. 3.10.1 shows the architecture of RFID.



(1C18)**Fig. 3.10.1 : A General Overview of RFID Architecture**

- Firstly items tags are scanned by reader. Secondly in backend transmitted data coming through antenna are being recognized by RFID based system PC.
- Gateway acts as middleware communication between items, reader a system database. At the end it filters out and store data in RFID database. Also it checks the data fault and relevant operation.

RFID applications

Following Table 3.10.2 shows RFID applications used in different domain.

Table 3.10.2

Sr. No.	Domain Name	Example
1.	Manufacturing and Processing	Inventory and production process monitoring , Warehouse order fulfilment
2.	Supply Chain Management	Inventory tracking systems, Logistics management
3.	Retail	Inventory control and customer insight, Auto checkout with reverse logistics
4.	Security	Access control , Theft control/prevention
5.	Location Tracking	Traffic movement control and parking management , Wildlife/Livestock monitoring and tracking

The working example of RFID application

- A RFID base Attendance System: RFID system is used to marinating and checking database for members of an institution.
- The basic idea involves each person of the institution having ID card. So when this card is swiped against the reader, the person's information is matched with the existing system in the database and his/ her attendance is marked.
- RFID is considered as a non specific short range device. It can use frequency bands without a license. The categories of frequency range are as follows.

LF : 125 kHz - 134, 2 kHz : low frequencies,
HF : 13.56 MHz : high frequencies,
UHF : 860 MHz - 960 MHz : ultra high frequencies,
SHF : 2.45 GHz : super high frequencies

- The following Table 3.10.3 shows the RFID applications as per frequency range.

Table 3.10.3

RFID	Key Applications	Standards
125 kHz (LF)	Inexpensive passive RFID tags for identifying animals	ISO 18000-2
13.56 MHz (HF)	Inexpensive passive RFID tags for identifying OBJECTS, Library books	ISO 14443
400 MHz (UHF)	For remote control for vehicle center locking systems	ISO 18000-7
868 MHz , 915 MHz, 922 MHz (UHF)	For active & passive RFID for logistics in Europe, USA, Australia	ISO 18000-2
2.45 GHz (MW)	ISM band used for active & passive RFID tags, Temp Sensors , GPS	ISO 18000-2
5.8 GHz (MW)	Used for long-reading Range passive and active RFID tags for vehicle identification, Highway toll collection	ISO 18000-2

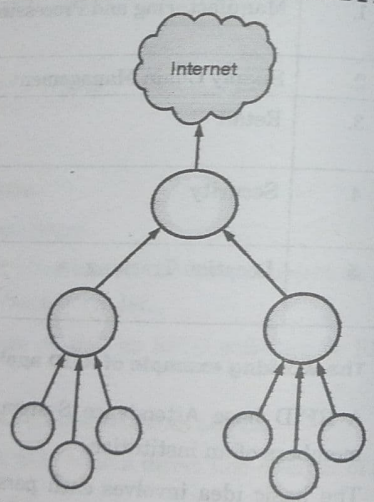
3.10.3 WSN

UQ. Explain WSN (the internet of transducers) pillar of IoT **(SPPU – Dec. 19, 3 Marks)**

- Wireless Sensor N/w t senses and gathers data using sensors which are spatially distributed.
- WSN collect this data into a centralized location with the help of wired / wireless connection
- Recently with the development of WSN, it led to distributed wireless sensor and actuator networks (WSANs) that are capable observing the physical world.
- Based on observation it makes the decision and performed the appropriate action.
- Wireless Sensor network consist of three elements, Sensors, Wireless communication modules and Open Source API.

The development of WSNs was motivated by military applications for example battlefield surveillance.

The WSN is consist of nodes from a few to several hundred or even thousand.



(1C19)**Fig. 3.10.2 : Sensor network architecture**

- Each node connected to one or several sensors.
- Sensor network node consists of several parts: a radio transceiver with an antenna, a microcontroller, an electronic circuit for interfacing with the sensors, and an energy source.
- The architecture of a typical sensor network is shown in Fig. 3.10.2
- The topology used in WSNs can vary from a simple star network to an advanced multihop mesh network with a gateway sensor node connected with a remote central server.

Wireless Sensor Networks Applications

1. Forest Fire Detection
2. Weather Monitoring System
3. Military Surveillance System
4. Manufacturing process control in large industries.
5. Smart Building system to control indoor climate.

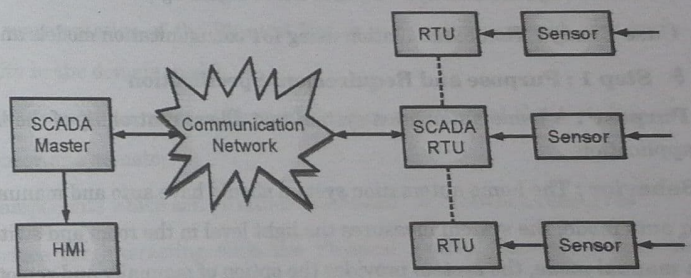
- For example consider Natural Environment Protection System
- Objectives - Detect and prevent forest fires. Detect flames, heat and gases that help to identify the molecules of chemical compounds generated during combustion (CO and CO2). With GPS, allow the exact geo location of the nodes.
- Prevention - After installing the WSN, the network can also acquire the daily values for temperature and relative humidity in order to determine the likelihood of a fire in each zone under surveillance.
- Alarm - Send an alarm indicating the status of the fire or the probability level and the area.

Unit III End

3.10.4 SCADA

UQ. What is SCADA ? What are the different blocks of SCADA **(SPPU – May 18, 5 Marks)**

- SCADA is Supervisory Control and Data Acquisition.
- SCADA is used to connect, monitor and control equipment's using short range network inside a building or an industrial plant.
- SCADA is software used to control the hardware. For example PLC, drives, servers, sensors and also obtain the data which is stored on the personal computer or Human Machine Interface (HMI).
- SCADA uses BacNet (communication protocol), CanBus and Wired Field Buses (Industrial Computer Network Protocols).



(1C20)**Fig. 3.10.3 : SCADA Architecture**

- The architecture of SCADA System is shown below.

As shown in Fig. 3.10.3 SCADA system consist of following key elements

1. **Sensors :** There are two types of sensors analog and digital. Different sensors are use like temperature, humidity, current, motion, and water applications. For data acquision sensors are attached with RTUs to take measurements.
2. **RTU- Remote Terminal Unit :** RTU connects to sensor in the process as well as SCADA master using communication network. They deliver various parameters to central station (SCADA master) to be managed by them.
3. **HMI- Human Machine Interface :** HMI is interaction on human operators and machines. HMI is tools that presents process data to a human operator and through this the human operator monitors and controls the process.
4. **SCADA Master :** SCADA master consist of programmable controls, multiprotocol support and provides human interface. It takes inputs from sensors through RTUs and controls various applications. SCADA master provides various display formats like graphs, tabular and other forms. It also provides email/paging based on certain conditions.
5. **Communication medium/network :** It is work as interfaces to connect SCADA master with SCADA RTUs.

☞ **SCADA Applications**

SCADA has following major applications of use.

1. **Electric Utilities** - Manage Current, voltage, circuit breaker, power grid.
2. **Water and Sewage** - Monitor and control water level, water flow and water pipe pressure.
3. **Building** - Control heating, ventilation, air conditioning, visualization, lighting and building access systems.
4. **Mass Transit** - regulation of electricity, track and locate buses, trains
5. **Railways/Roadways** - Control traffic signal lights

Case Study : Home Automation using IoT communication models and IoT communication APIs

Step 1 : Purpose and Requirement Specification

Purpose : A home automation system that allows controlling of the lights in a home remotely using a web application.

Behavior : The home automation system should have auto and manual modes.

In auto mode, the system measures the light level in the room and switches on the light when it gets dark.

In manual mode, the system provides the option of manually and remotely switching on/off the light.

System Management Requirement : The system should provide remote monitoring and control functions.

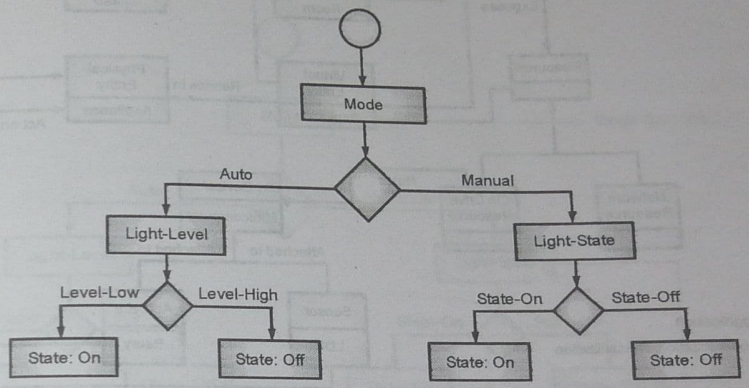
Data Analysis Requirements : System should perform local analysis of data.

Application Deployment Requirement : Deployed locally on device, but acts remotely without manual intervention.

Security Requirement : Authentication to Use the system must be available

Step 2 : Process Specification

In this step, the use cases of the IoT system are formally described based on and derived from the purpose and requirement specifications. So for our example Home Automation system use case diagram is shown below.



(1C21)**Fig. 3.10.4 : Use Case of Home Automation System**

Step 3 : Domain Model Specification

Domain model specification elements for Home automation System described as follows.

Physical Entity : a room, a light, an appliance

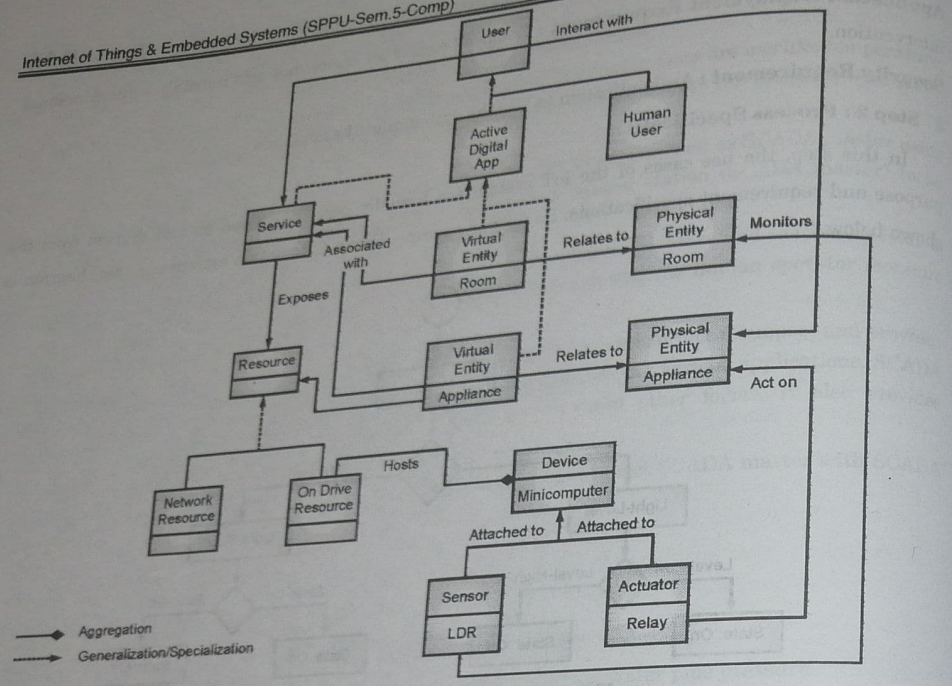
Virtual Entity : Virtual Entity is a representation of the Physical Entity in the digital world. For each Physical Entity, there is a Virtual Entity in the domain model.

Device : Device provides a medium for interactions between Physical Entities and Virtual Entities. For our example devices are minicomputer, sensor and actuator

Resource : Resources are software components which can be either "on-device" or "network-resources".

Service : Services provide an interface for interacting with the Physical Entity. Services access the resources hosted on the device or the network resources to obtain information about the Physical Entity or perform actuation upon the Physical Entity.

The Domain Model specification for Home Automation System shown in Fig.3.10.5.

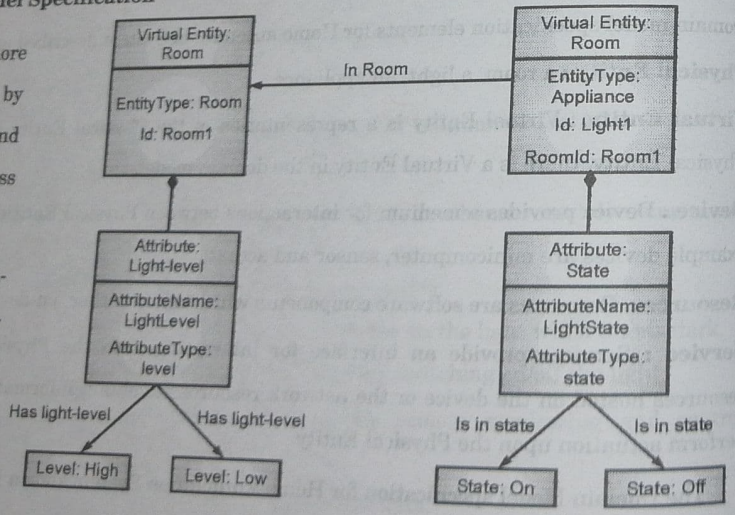


(1C22)**Fig. 3.10.5 : Domain Model Specification of Home Automation System**

Step 4 : Information Model Specification

Information model adds more details to the Virtual Entities by defining their attributes and relations with the help of class diagram.

Home automation System- There are two virtual entities- Light Appliance (light state), Room (light level)



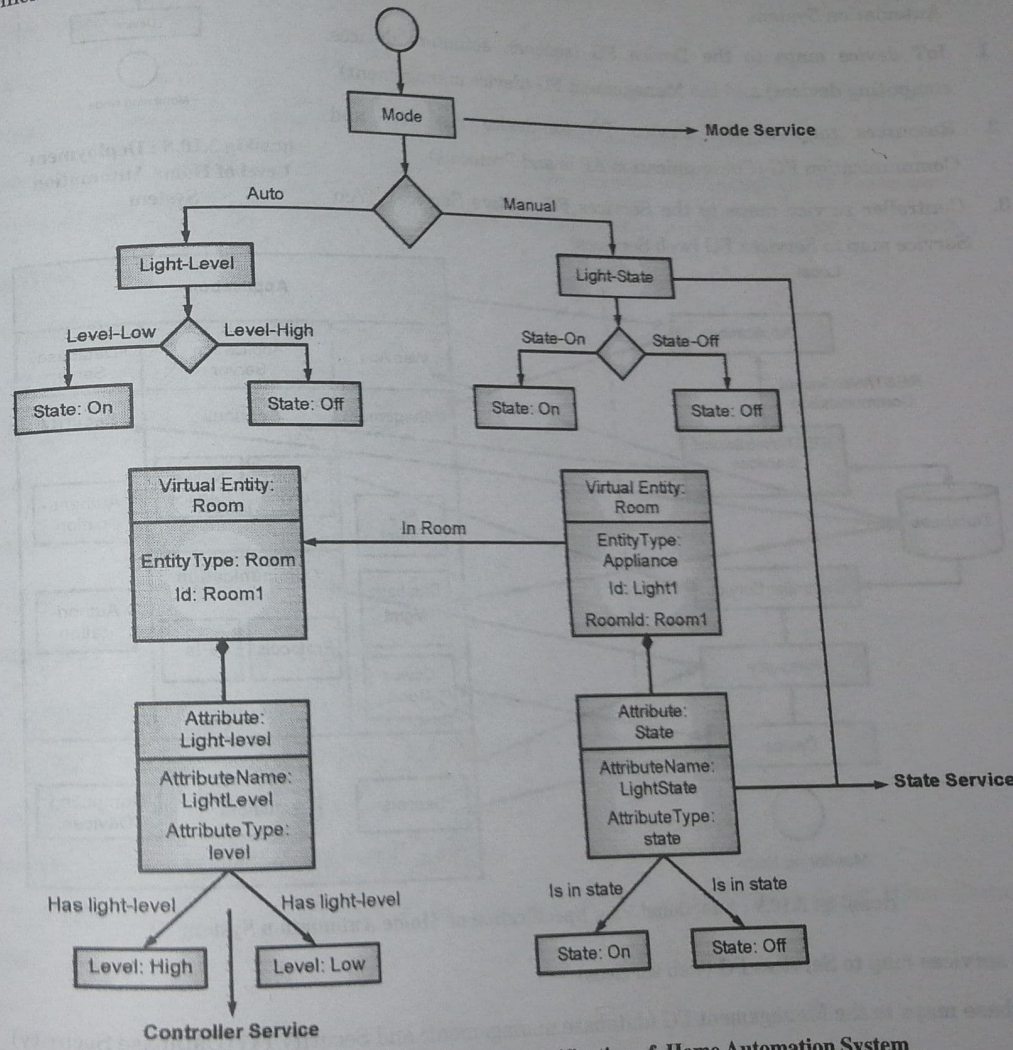
(1C23)**Fig. 3.10.6 : Class diagram of Home Automation System**

Step 5 : Service Specification

There are three types of services used in Home Automation System.

1. **Mode Service :** It sets the mode auto or manual or retrieves the current mode.
2. **State Service** : Sets the light appliance state to on/off or retrieves the current light state.
3. **Controller Service :** In auto mode, the controller service monitors the light level and switches the light on/off and updates the status in database. In manual mode, the controller service, retrieves the current state from the database and switches the light on/off

Service specification is shown below with help of process specification and information model specification.



(1C24)**Fig. 3.10.7 : Information Model Specification of Home Automation System**

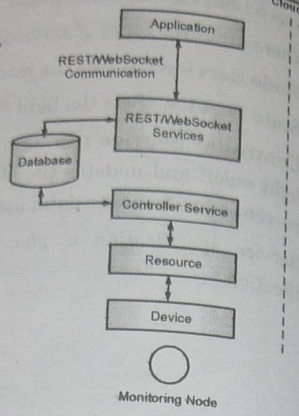
Step 6 : IoT Level Specification

Decide the deployment level of IoT System. The deployment level of the home automation IoT system is level-1.

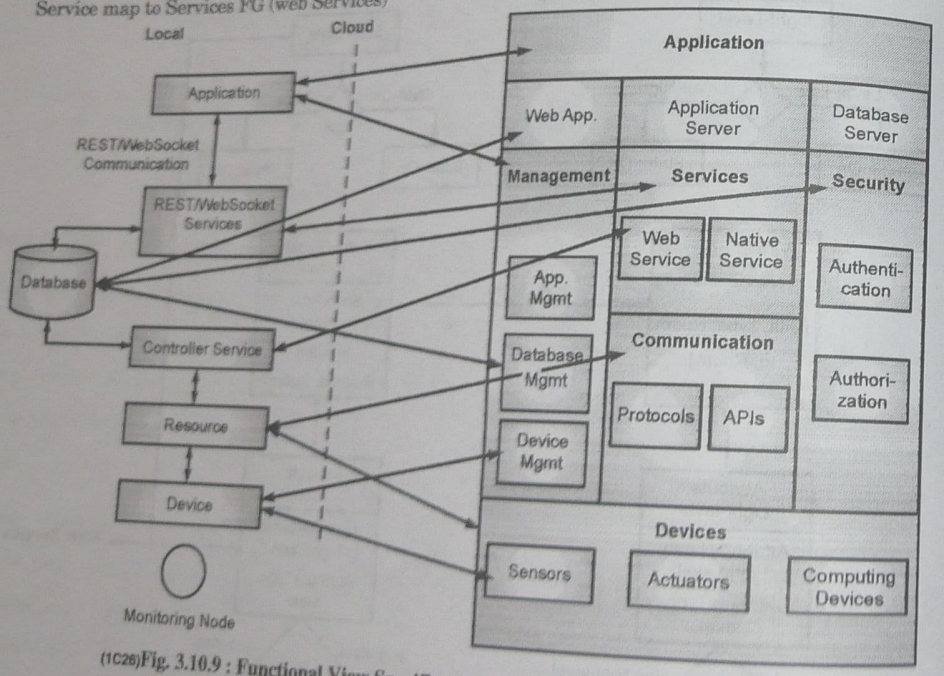
Step 7 : Functional View Specification

- The Functional View (FV) defines the functions of the IoT systems grouped into various Functional Groups (FGs). The following Fig. 3.10.9 shows Functional view specification of Home Automation System.

1. IoT device maps to the Device FG (sensors, actuators devices, computing devices) and the Management FG (device management)
2. Resources map to the Device FG (on-device resource) and Communication FG (Communication AP is and Protocols)
3. Controller service maps to the Services FG (native Service). Web Service map to Services FG (web Services)



(1C25)**Fig.3.10.8 : Deployment Level of Home Automation System**



(1C26)**Fig. 3.10.9 : Functional View Specification of Home Automation System**

4. Web services map to Services FG (web services)
5. Database maps to the Management FG (database management) and Security FG (Database Security)

6. Application maps to the Application FG (web application, application and database servers). Management FG (App management) and Security FG (app Security)

Step 8 : Operational View Specification

Operational View specifications for the home automation example are:

- **Devices :** Computing device (Raspberry Pi), light dependent resistor (sensor), relay switch (actuator).
- Communication APIs: REST APIs
- **Communication Protocols :** Link Layer - 802.11, Network Layer - IPv4/IPv6,Transport TCP, Application – HTTP.

Services

(1) Controller Service - Hosted on device, implemented in Python and run as a native service.

(2) Mode service - REST-ful web service, hosted on device, implemented with Django-REST Framework.

(3) State service - REST-ful web service, hosted on device, implemented with Django-REST Framework.

Application

(1) Web Application - Django Web Application,

(2) Application Server - Django App Server,

(3) Database Server – MySQL

Security

(1) Authentication: Web App, Database

(2) Authorization: Web App, Database

Management

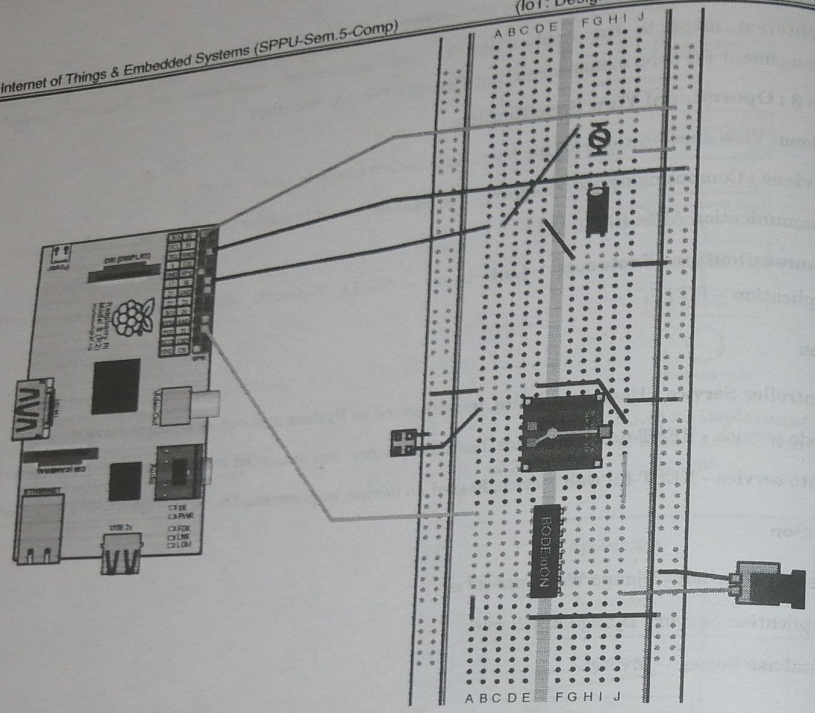
(1) Application Management - Django App Management

(2) Database Management - MySQL DB Management

(3) Device Management - Raspberry Pi device Manage

Step 9 : Device and Component Integration

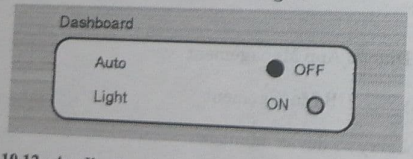
The following Fig. 3.10.11 shows device and component interaction diagram for Home Autor System.



(1C27)**Fig.3.10.11 : Component and Device Interaction of Home Automation System**

Step 10 : Application Development

- The final step in the IoT design methodology is to develop the IoT application.
- Home Automation application has controls for mode and light.



(1C28)**Fig. 3.10.12 : Application Development of Home Automation System**

Chapter Ends...

❑❑❑

UNIT IV

CHAPTER 4 IoT Protocols

Syllabus

Protocol Standardization for IoT, M2M and WSN Protocols, RFID Protocol, Modbus Protocol, Zigbee Architecture. IP based Protocols: MQTT (Secure), 6LoWPAN, LoRa.

Case Studies : LoRa based Smart Irrigation System