Unit II

Embedded IoT Platform Design Methodology
Outline

• Purpose and Requirement Specification
• Process Specification
• Domain model specification
• Information Model Specification
• Service specifications
• IoT Level Specifications
• Functional view specification
• Operational View Specification
• Device and Component integration
• Application development
Steps involved in IoT System Design Methodology

1. Purpose & Requirements
   Define Purpose & Requirements of IoT System

2. Process Model Specification
   Define the use cases

3. Domain Model Specification
   Define Physical Entities, Virtual entities, Devices, Resources and Services in IoT System

4. Information Model Specification
   Define structure (e.g. relations, attributes) of all the information in the IoT System

5. Service Specification
   Map Process and information model to services and define service specifications

6. IoT level Specification
   Define IoT Level for the system

7. Functional view Specification
   Map IoT Level to functional groups

8. Operational view Specification
   Define communication options, service hosting options, storage and device options

9. Device & Component Integration
   Integrate devices, develop and integrate the components

10. Application Development
    Develop applications
Advantages of Using Design methodology

• Reducing the design, testing and maintenance time
• Provide better interoperability
• Reduce the complexity
Step 1 : Purpose and Requirement Specification

Defines

• System purpose
• behavior and
• Requirements (such as data collection requirements, data analysis requirement, system management requirements, data privacy and security requirements, User interfaces requirements)
Step 1 : Purpose and Requirement Specification

• **Purpose** : An automated irrigation mechanism which turns the pumping motor ON and OFF on detecting the moisture content of the earth without the intervention of human

• **Behavior** : System should monitor the amount of soil moisture content in soil. In case the soil moisture of the soil deviates from the specified range, the watering system is turned ON/OFF. In case of dry soil, it will activate the irrigation system, pumping water for watering the plants.

• **System Management Requirements** : system should remotely provide monitoring and control functions
Step 1 : Purpose and Requirement Specification

• **Data Analysis Requirements**: system should perform local analysis of data

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

• **Security**: Authentication to Use the system must be available
Step 2 : Process Specification

• Define the process with the help of use cases
• The use cases are formally described based on Purpose & requirement specification
• In this use case :
  – Circle denotes a state or an attribute
Step 2 : Process Specification

- Draw the use case

```
  Threshold
  / 
//
Exceed Not Exceed

  Signal
  
  Low High
  
  Motor State: On Motor State: Off

  Motor State
  
  State: On State: Off
  
  Motor On Motor Off
```
Step 3 : Domain Model Specification

• Describes the main concepts, entities and objects in the domain of IoT system to be designed

• Entities, Objects and Concepts include the following: Physical entity, Virtual entity, Device, Resource, Service
Step 3 : Domain Model Specification

• **Physical Entity:**
  – Discreet identifiable entity in physical environment
  – For eg. Pump, motor, LCD
  – The IoT System provides the information about the physical entity (using sensors) or performs actuation upon the Physical entity (like switching a motor on etc.)
  – In smart irrigation example, there are three Physical entities involved:
    • Soil (whose moisture content is to be monitored)
    • Motor (to be controlled)
    • Pump (To be controlled)

• **Virtual Entity:**
  – Representation of physical entity in digital world
  – For each physical entity there is a virtual entity
Step 3 : Domain Model Specification

- **Device:**
  - Medium for interactions between Physical and Virtual Entities.
  - Devices (Sensors) are used to gather information from the physical entities.
  - Devices are used to identify Physical entities (Using Tags).
  - In Smart Irrigation System, device is soil moisture sensor and buzzer as well as the actuator (relay switch) attached to it.
Step 3 : Domain Model Specification

- In smart irrigation system there are three services:
  - A service that sets the signal to low/high depending upon the threshold value
  - A service that sets the motor state on/off
  - A controller service that runs and monitors the threshold value of the moisture and switches the state of motor on/off depending upon it.

  When threshold value is not crossed the controller retrieves the motor status from database and switches the motor on/off.
Step 4 : Information Model Specification

- Defines the structure of all the information in the IoT system (such as attributes, relations etc.)
- It does not describe the specifics of how the information is represented or stored.
- This adds more information to the Virtual entities by defining their attributes and relations
- I: e, Draw Class diagram
Step 4 : Information Model Specification

Virtual Entity: Soil
  Entity Type: Soil
  ID: Soil1

Virtual Entity: Pump
  Entity Type: Pump
  ID: Pump1

Virtual Entity: Motor
  Entity Type: Motor
  ID: Motor1

Attribute: State
  Attribute Name: motorState
  Attribute Type: state
  Is in State

  State: On
  State: Off

Has signal level
  Signal: High
  Signal: Low

Has signal level
  Is in State
Step 5 : Service Specification

- Define the services in IoT System, service types, service inputs/outputs, service endpoints, service schedules, service preconditions and service effects
- Services can be controller service, Threshold service, state service, for smart irrigation system
- These services either change the state/attribute values or retrieve the current values.
- For eg.
  - Threshold service sets signal to high or low depending upon the soil moisture value.
  - State service sets the motor state: on or off
  - Controller service monitors the threshold value as well as the motor state and switches the motor on/off and updates the status in the database
Step 5: Service Specification

Threshold Service

Motor State

State: On

Motor

State: Off

Threshold

Exceed

Not Exceed

Signal

Low

High

Motor State

State: On

Motor

State: Off

Virtual Entity: Soil

Entity Type: Soil
ID: Soil1

Attribute: Signal
Attribute Name: signal
Attribute Type: signal

Signal: High

Controller Service

Virtual Entity: Pump

Entity Type: Pump
ID: Pump1

Attribute: Signal
Attribute Name: signal
Attribute Type: signal

Signal: Low

Virtual Entity: Motor

Entity Type: Motor
ID: Motor1

Attribute: State
Attribute Name: motorState
Attribute Type: state

State: On

State: Off
Step 5: Service Specification
..>Controller Service

Input
Threshold: High/Low

Has input

Has schedule

Schedule
Interval: Every 5 Sec

Has Output

Output
Motor State: On/Off

Service
Name: Controller
Type: Native
Step 5: Service Specification

..>Threshold Service

- **Input**
  - Threshold: High/Low

- **Output**
  - Threshold: Low/High

- **Service**
  - Name: Threshold
  - Type: REST

- **Endpoint**
  - Endpoint:/home/threshold
  - Protocol: HTTP
Step 5: Service Specification ..>State Service

- **Input**: State : On/Off
- **Service Name**: State
- **Type**: REST
- **Endpoint**: /home/state
- **Protocol**: HTTP
- **Output**: Motor : On/Off
- **Has input**: Has endpoint
- **Has Output**: Has endpoint
Step 6: IoT Level Specification

- Decide the deployment level of IoT System. Here I am using Deployment Level 1.
Step 7 : Functional View Specification

• Define the functions of IoT System grouped into various functional groups.

• These functional groups provide functionalities for interacting with the concepts defined in Domain model specification.
Step 7: Functional View Specification...Drawing Functional Groups
Step 7: Functional View Specification

...Deployment level to Functional Group Mapping

![Functional View Diagram](image)

<table>
<thead>
<tr>
<th>Application</th>
<th>Web App</th>
<th>Application Server</th>
<th>Database Server</th>
</tr>
</thead>
<tbody>
<tr>
<td>Management</td>
<td>Services</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Application Mgmt</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Web Service</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Native Service</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Database Mgmt</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Protocols</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>APIs</td>
<td></td>
</tr>
<tr>
<td>Devices</td>
<td></td>
<td>Sensors</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Actuators</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Computing Devices</td>
<td></td>
</tr>
</tbody>
</table>
Step 8 : Operational View Specification

• Define the Operations/options related to IoT System development

• Such as Device options, Storage options, Application hosting option
Step 8: Operational View Specification of automated irrigation system

- **Application**
  - Web App: PhP WebApp
  - Application Server: Google App engine
  - Database Server: MySQL

- **Services**
  - Native: Controller Service
  - Web: REST

- **Communication**
  - Communication APIs: REST APIs
  - Communication Protocol:
    - Link Layer: 802.11
    - N/w: IPV6
    - Transport: TCP
    - Application: HTTP
Step 8: Operational View Specification of automated irrigation system

- **Management**
  - Device Management: Arduino device management
  - Application Management: PHP App Management
  - Database Management: MySQL Db Mgmt

- **Security**
  - Login Management
Step 9: Device and Component Integration

- Integrates the devices and components and draw a schematic diagram showing the same.
Device and Component Integration: Alternate Diagram

- Probes to Soil
- Sensor Circuit
- AT895S2 Microcontroller
- LCD Screen
- Motor Driver Circuit
- DC Motor
- Power Supply
Step 10: Application development

- GUI / Screenshot of IoT Application
Purpose of the Project: The Health Monitoring System is basically used to monitor the Patient Body Temperature and Pulse by respective sensors. This information is captured and stored, so that the authorized personnel can view and analyze the Data remotely at any time.
**IoT Health Monitoring Application (HMA)**

**Purpose and Requirement Specification**

- **Behavior:** The Monitoring is done in real time to identify the state of the patient. This information can be used to analyze the state of a patient or to get sensitive data in order to be sequentially captured for medical diagnosis. The system Alarms in Emergency Situation and notifies the staff and relatives of the patient.
IoT Health Monitoring Application (HMA) Purpose and Requirement Specification

- **System Management Requirement**: The system provides remote monitoring and control functions.

- **Data Analysis Requirement**: System allows the analysis of data and can be visualized in graphical format.

- **Application Deployment Requirement**: Application will be deployed locally on the device, but can be monitored remotely.

- **System Requirements**: Only Authorized users can access and control the Application.
IoT Health Monitoring Application (HMA) Process Specification

• The Use Case Diagram describes the use case’s of the system and the actors involved.
• The Process diagram shows the steps involved in the process.

   The sensors read the information from the Human Body and stores it in Database, when the values go beyond the threshold limit it sends alerts.
IoT Health Monitoring Application (HMA) Process Specification
IoT Health Monitoring Application (HMA) Domain Specification

- **Physical Entity**: In HMA, human body is the physical entity where body temperature and pulse are monitored using respective sensors.

- **Virtual Entity**: It’s a representation of Physical Entity in Digital World. For each physical entity there exist one virtual entity in Domain Model.

- **Device**: Provides medium for interactions between Physical and Virtual Entities. In HMA, device is a single board (Arduino) which has temperature and pulse sensor attached to it.
IoT Health Monitoring Application (HMA)
Domain Specification

- **Resource**: Can be Software Components which can be either “on-device” or “network-resources”. On-Device resources are hosted on device and include software components that provide information about physical entity. Network Resources are software components on network such as Database.

- **Service**: In HMA, the services will be, service that retrieve current information, native service that runs on the device.
IoT Health Monitoring Application (HMA) Information Model Specification

Virtual Entity:
Human Body

Entity Type:
Human Body

Attribute:
State
Attribute Name: Body Temperature
Temperature: val

Attribute:
State
Attribute Name: Pulse
Pulse: val
IoT Health Monitoring Application (HMA) Service Specification

Service

Name: Controller
Type: Native

Output

Body Temperature
Pulse

Schedule

Interval: Every 30 min
IoT Health Monitoring Application (HMA)  
IoT Level Specification

- The system has a single device that performs sensing, stores data, performs analysis, and hosts the Application. Thus, IoT Level 1
IoT Health Monitoring Application (HMA) Functional Specification
IoT Health Monitoring Application (HMA) Functional Specification
Step 8 : Operational View
Specification of HMA

• Application
  – Web App : PhP WebApp
  – Application Server : Google App engine
  – Database Server : MySQL

• Services
  • Native : Controller Service
  • Web : REST

• Communication
  – Communication APIs : REST APIs
  – Communication Protocol :
    • Link Layer: 802.11
    • N/w : IPV6
    • Transport : TCP
    • Application : HTTP
Step 8: Operational View
Specification of HMA

- **Management**
  - Device Management: Arduino device management
  - Application Management: PHP App Management
  - Database Management: MySQL Db Mgmt

- **Security**
  - Login Management
Step 9: Device and Component Integration
Step 10: Application Specification
Home Intrusion Detection System
• **Purpose of the Project**: The purpose of home intrusion detection system is to detect intrusions using sensors and raise alerts, if necessary.
HID
Purpose and Requirement Specification

• **Behavior:** In case of any intrusion, I intend to capture a picture of the intruder, mail the image to the respective end users and alert them. I would like to use an alarm which goes on in case of an intrusion.
HID Purpose and Requirement Specification

• **Data Analysis Requirement:** The light dependent resistor data is analyzed locally. If the sensor reading falls below the threshold value, an image is captured and send as an alert to the user.

• **Application Deployment Requirement:** The application is deployed locally on the device and can be accessible from anywhere via node-red.

• **System Requirements:** Only Authorized users can access and control the Application.
The Use Case Diagram describes the use case’s of the system and the actors involved.

The Process diagram shows the steps involved in the process.

With the help of Light dependent resistor and PIR motion sensor, I am going to detect the motions in the room.

If a motion is detected, I intend to capture the image with the help of a webCam and store locally.

Now the alerts are sent to the user with the captured image.

Also I am using the buzzer which turns on in case of any intrusion.
HID Process Specification

1. Read the motion sensor values.
2. [Checks if the motion is detected]
   - Condition: no
   - Condition: yes
3. If Condition is yes:
   - Capture the image
   - Trigger Alarm
   - Alert the user
From the above process diagram, we can see that the process starts at the circle. The PIR motion sensor which is shown in the rectangular box in the above figure, is placed in a room. It detects the motion in the room. If there is a motion then an image is captured and an alert will be sent to the user. And also it turns on the buzzer. This decision (yes/no) to raise an alert is shown in the diamond box in the above figure.
• The domain model describes the main concepts, entities and objects in the domain. The domain includes physical entities for rooms. The devices in my system are nodemcu to which PIR and light dependent resistor is attached and raspberry pi to which camera and buzzer are attached to send alerts.
IoT level Specification

• implement my IOT system that is “Home intrusion detection system” in level 5. The IOT level 5 system has multiple nodes and one coordinator node. Coordinator node collects the data from the end nodes and sends to the cloud.