Internet of Things
UNIT V
IOT Design and System Engineering

Prepared By
Mital Kadu
Assistant Professor
Artificial Intelligence and Data Science
DYPIEMR, Pune

Syllabus

- Discuss IOT Requirements Hardware & Software;
- Study of IOT Sensors;
- Tagging and Tracking;
- Embedded Products;
- IOT Design;
- SIM Card Technology;
- IOT Connectivity and Management;
- IOT Security
- IOT Communication.

IoT Requirement: Hardware and Softw

- **Arduino:**

  - Arduino is a **printed circuit board.**
  - **Open source hardware/software.**
  - Provides an **IDE** based on Processing, can support **C / C++.**
  - The system provide sets of **analog and digital I/O.**
  - Include an **USB** for loading programs.
  - It's **capable of doing everything**.

IoT Requirement: Hardware and Softwar

- Pros:
  - **Low power architectures**
  - **Easy interfacing with sensors and data collection**
  - **Open-source**
  - **Variety for programming: IDE , python, ruby, C,** etc.
- Cons:
  - **Memory limitations and less powerful (compared to Raspberry)**

IoT Requirement: Hardware and Softw

- Raspberry Pi:
  - is a **low cost computer with a credit-card sized.**
  - Uses a **standard keyboard and mouse**.
  - It's **capable of doing everything** you expect a desktop computer to do.
  - It has been used in a wide array of projects: from music machines and detectors to weather stations.

IoT Requirement: Hardware and Softwar

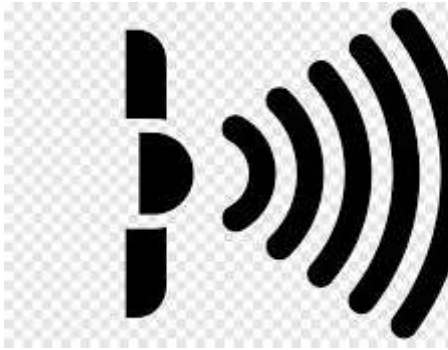
- **Different models**:
  - Raspberry Pi Zero
  - Raspberry Pi 1 Model A+
    Raspberry Pi 1 Model B+
  - Raspberry Pi 2 Model B
  - Raspberry Pi 3 Model B
  - Raspberry Pi Touch Display
  - Raspberry Pi Case

IoT Requirement: Hardware and Softw

- Pros:
  - **Cheap and portable**
  - **Super powerful with lot of memory and processing capabilities**
  - **Endless possibilities of what can be done using it**
- Cons:
  - **You need good knowledge of Linux**
  - **Closed source**
  - **Power hungry**

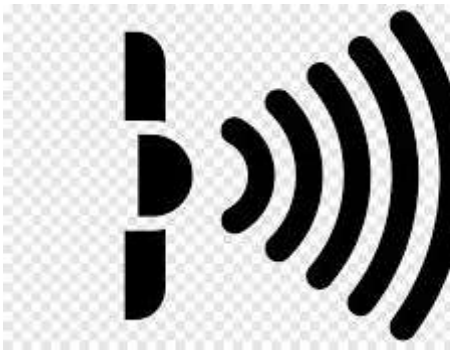
Study of IOT Sensors

- Is an object whose **purpose is to detect events or changes in its environment then provide and output.**
- A sensor is a **type of transducer**.
- Transform **non-electrical signals to electrical signals**.



Study of IOT Sensors

- Specifications of sensor:
  - Accuracy:
    - **Error** between the **result of a measurement** and the **true value being measured.**
  - Resolution:
    - The **smallest increment of measure** that a device can make.
  - Sensitivity:
    - The **ratio** between the **change in the output signal to a small change in input physical signal**. Slope of the input-output fit line.



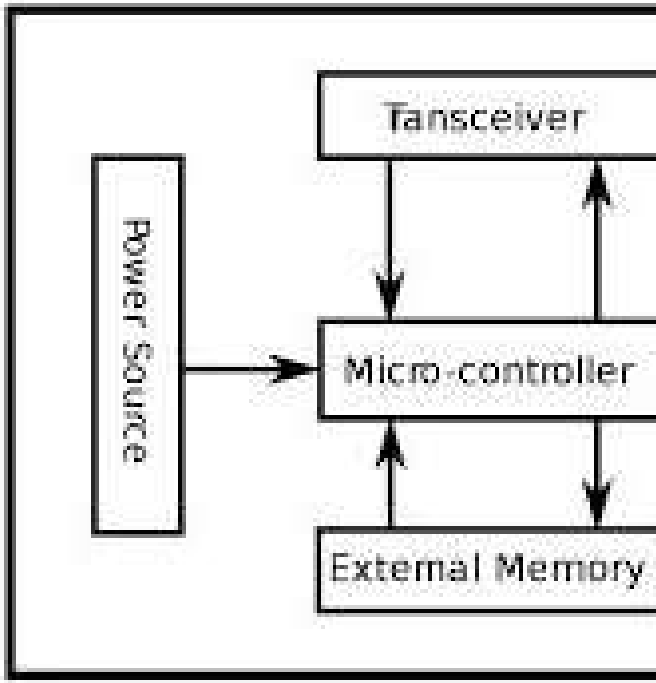
Study of IOT Sensors

- Specifications of sensor:

- Repeatability/Precision:
  - The **ability** of the sensor to **output the same value for the same input over a number of trials.**
- Bandwidth:
  - The **frequency range between the lower and upper cut-off frequencies**, within which the sensor transfer function is constant gain or linear.

Study of IOT Sensors

- Sensor Components:
  - Controller:
    - A controller to process all the relevant data, capable of executing arbitrary code.
  - Memory:
    - Some memory to store programs and intermediate data; usually, different types of memory are used for programs and data.
  - Sensors and actuators:
    - The actual interface to the physical world devices that can observe or control physical parameters of the environment.



Study of IOT Sensors

- Communication:
  - Turning nodes into a network requires a device for sending and information over a wireless channel.
- Power supply:
  - As usually no tethered power supply is available, some form of ba necessary to provide energy. Sometimes, some form of recha obtaining energy from the environment is available as well.
  - For actual communication, both a transmitter and a receiver are req sensor node. The essential task is to convert a bit stream comin microcontroller and convert them to and from radio waves. For purposes, it is usually convenient to use a device that combines t tasks in a single entity. Such combined devices are called transceivers

Study of IOT Sensors

- Sensor Types:
  - Passive, omnidirectional sensors:
    - These sensors can **measure a physical quantity at the point of the s**
      without actually manipulating the environment by active probing. I
      they are passive. Moreover, some of these sensors actually are **self-**
      the sense that they obtain the energy they need from the environme

    - Typical examples for such sensors include **thermometer, light sensor**
      microphones, humidity, mechanical stress or tension in materia
      sensors sensitive for given substances, smoke detectors, air pressur

Study of IOT Sensors

- **Sensor Types:**
- Passive, narrow - beam sensors:
  - These sensors are passive as well, but have a well-defined direction of measurement. A typical example is a **camera**, which **measurements in a given direction, but has to be rotated if nee**
- Active sensors:
  - This last group of sensors **actively probes the environment**, for a sonar or **radar sensor or some types of seismic sensors, which shock waves by small explosions.** These are quite specific, **trig explosion is certainly not a lightly undertaken action and requ special attention.**

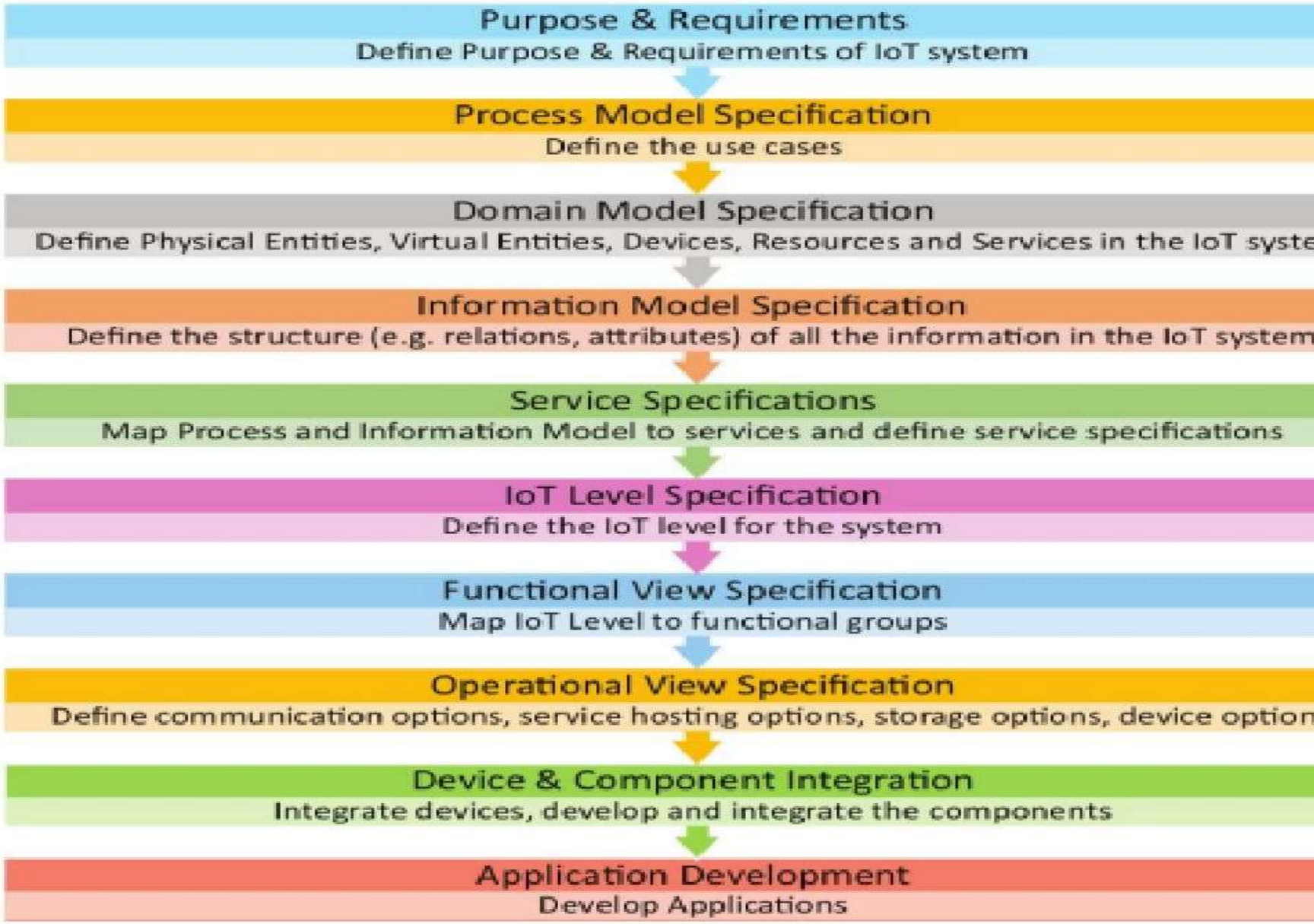
Study of IOT Sensors

- Sensor Types:
  - Active sensors:
    - Require an external source of power (excitation voltage) that p majority of the output power of the signal
  - Passive sensors:
    - The output power is almost entirely provided by the measured signa excitation voltage.
  - Digital sensors:
    - The signal produced or reflected by the sensor is binary.
  - Analog sensors:
    - Produce a continuous output signal or voltage which is generally pro the quantity being measured.

IoT Design

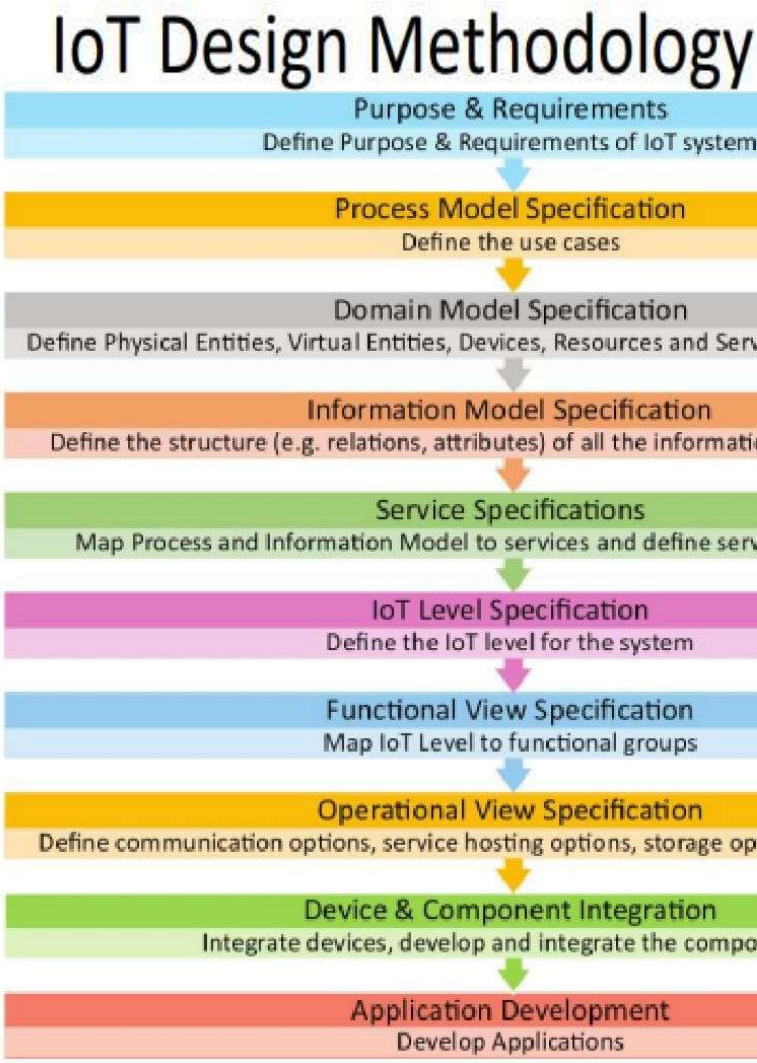
- IoT Design Methodology that includes:
  - Purpose & Requirements Specification
  - Process Specification
  - Domain Model Specification
  - Information Model Specification
  - Service Specifications
  - IoT Level Specification
  - Functional View Specification
  - Operational View Specification
  - Device & Component Integration
  - Application Development

IoT Design Methodology - Steps



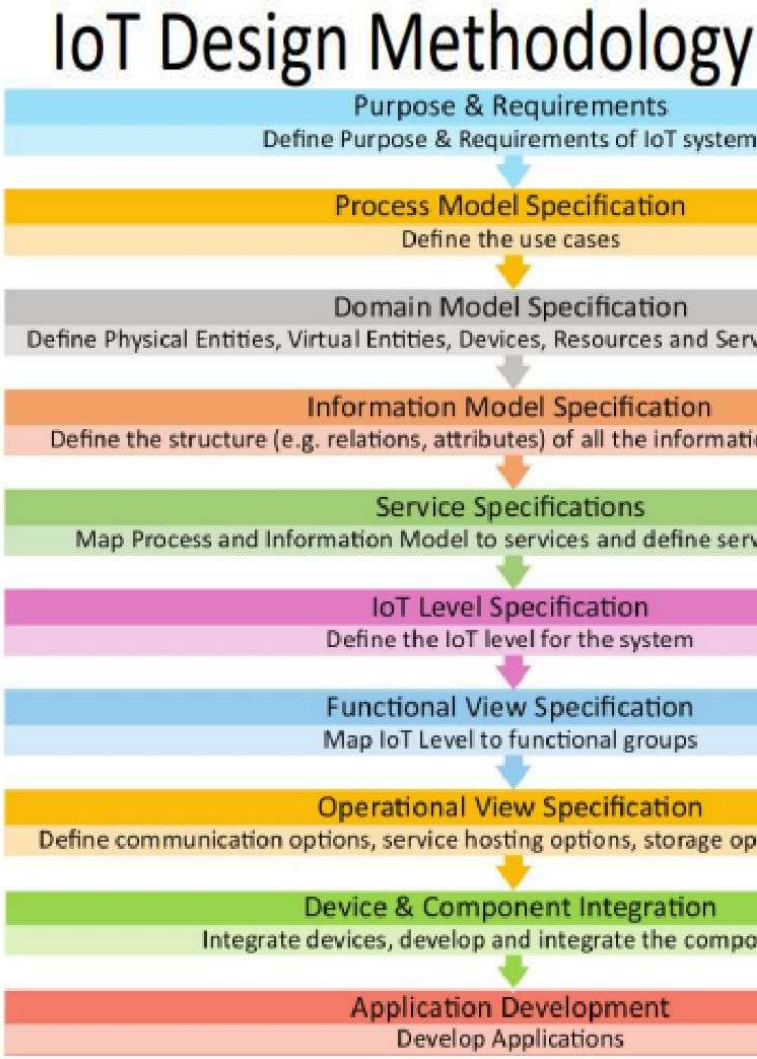
IoT Design

- Step 1: Purpose & Requirements Specification
  - The first step in IoT system design methodology is to define the purpose and requirements of the system. In this step, the system purpose, behavior and requirements (such as **data collection requirements, data analysis requirements, system management requirements, data privacy and security requirements, user interface requirements,** ...) are captured



IoT Design

- Step 2: Process Specification
  - The second step in the IoT design methodology is to define the 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**



IoT Design

- Step 3: Domain Model Specification
  - The third step in the IoT design methodology is to define the D Model.
  - The domain model
    - **describes the main concepts, entities and objects in the domain system to be designed**.
    - **defines the attributes of the objects and relationships be objects**.
    - **provides an abstract representation of the concepts, object entities in the IoT domain, independent of any specific technol platform**.
    - **With it, the IoT system designers can get an understanding of t domain for which the system is to be designed**.

IoT Design

- Step 4: Information Model Specification
  - The fourth step in the IoT design methodology is to define t Information Model.
  - Information Model
    - **defines the structure of all the information in the IoT system, example, attributes of Virtual Entities, relations, etc.**
    - **does not describe the specifics of how the information represented or stored.**
    - **To define the information model, we first list the Virtual Entit defined in the Domain Model.**
    - **adds more details to the Virtual Entities by defining th attributes and relations**.

IoT Design

- Step 5: Service Specifications
  - The fifth step in the IoT design methodology is to define the service specifications.
  - Service specifications
    - define the services in the IoT system, service types, service inputs/output, service endpoints, service schedules, service preconditions and service effects.
- Step 6: IoT Level Specification
  - The sixth step in the IoT design methodology is to define the **IoT level for the system**.

IoT Design

- Step 7: Functional View Specification

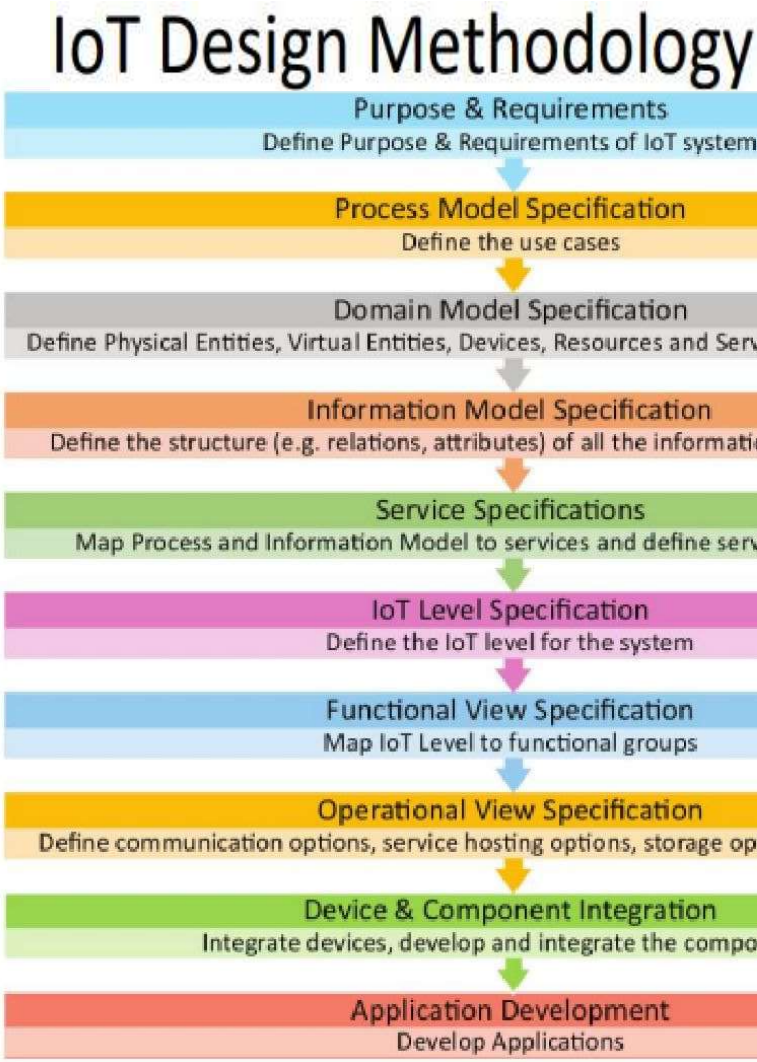
  - The seventh step in the IoT design methodology is to define the Funct View.
  - The Functional View (FV)
    - **defines the functions of the IoT systems grouped into various Funct Groups (FGs).**
    - **Each Functional Group either provides functionalities for intera with instances of concepts defined in the Domain Model or pro information related to these concepts.**

IoT Design

- Step 8: Operational View Specification
  - The eighth step in the IoT design methodology is to define the Operat View Specifications.
  - In this step, **various options pertaining to the IoT system deployment operation are defined, such as, service hosting options, storage opt device options, application hosting options, etc**
- Step 9: Device & Component Integration
  - The ninth step in the IoT design methodology is the **integration of devices and components**

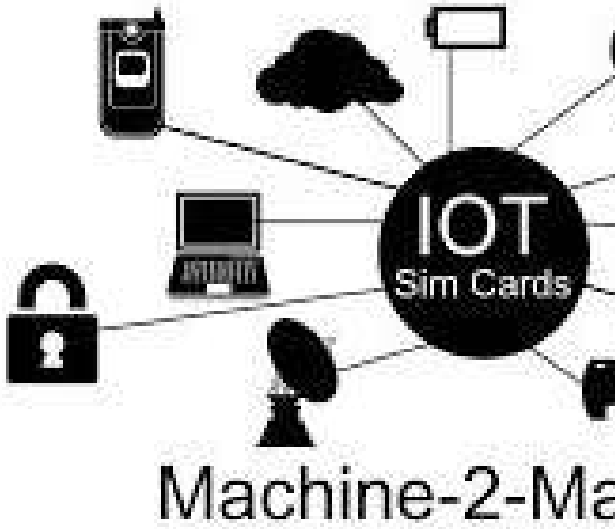
IoT Design

- Step 10: Application Development
  - The final step in the IoT design methodology is to **develop the IoT application.**



SIM(Subscriber Identity Module) Card Techn

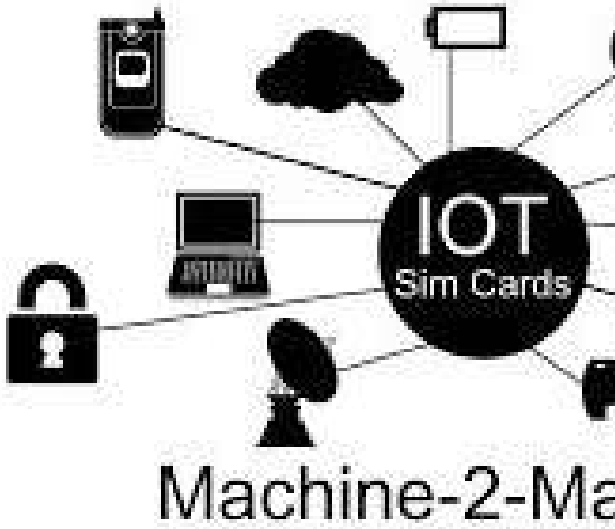
- IoT sim cards:
  - also called **M2M card**
  - are **designed for smart devices**
  - allow **devices** to be **connected** and able to **communicate** with each other.
  - are designed to connect between devices and between two machines **northbound apps**.
  - **Applications:**
    - automotive,
    - industrial,
    - logistics and telematics and
    - medical industries



Machine-2-Ma

SIM Card Technology

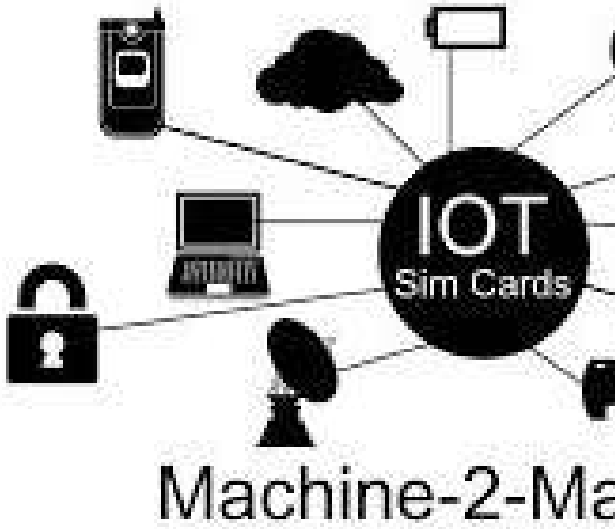
- **Traditional SIM cards** are mostly applications consumer devices such as **smartphones and tablets.**
- Devices with **M2M SIM cards can send and receive data across cellular networks.**
- In IoT devices, the M2M SIM may **share data directly with other devices** and/or the **software that manages the platform.**



Machine-2-Ma

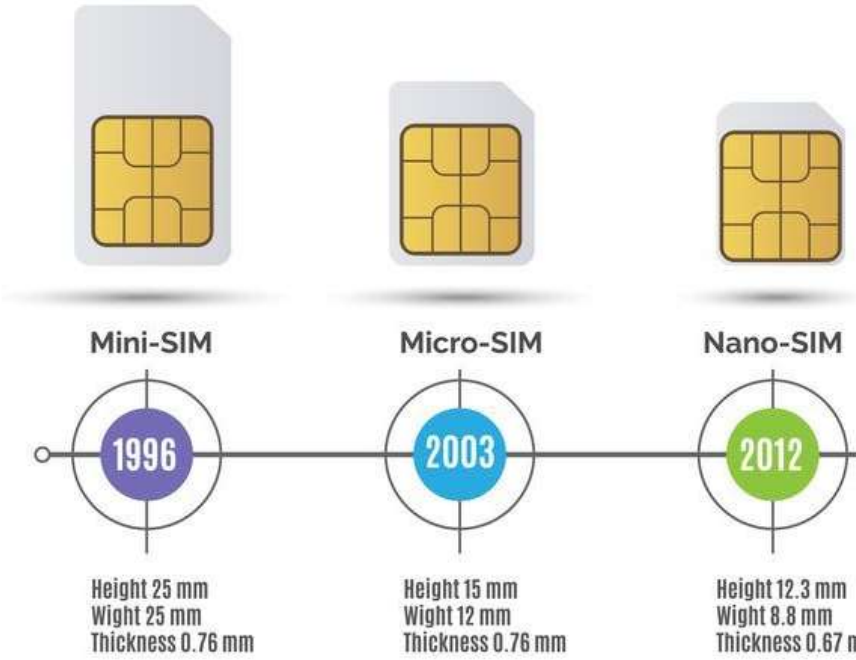
SIM Card Technology

- IoT SIM cards are
  - **available with 2G, 3G, 4G/LTE, 5G, NB-IoT, LTE-M (Cat M1) network options**
  - **compatible with** thousands of **wireless IoT devices**.
  - **Features:**
    - M2M SIMs are **more durable**.
    - M2M SIMS are **remotely manageable**.
    - M2M SIMS **support data plan aggregations**.
    - M2M SIMs **may have fixed IP addresses**



SIM Card Technology

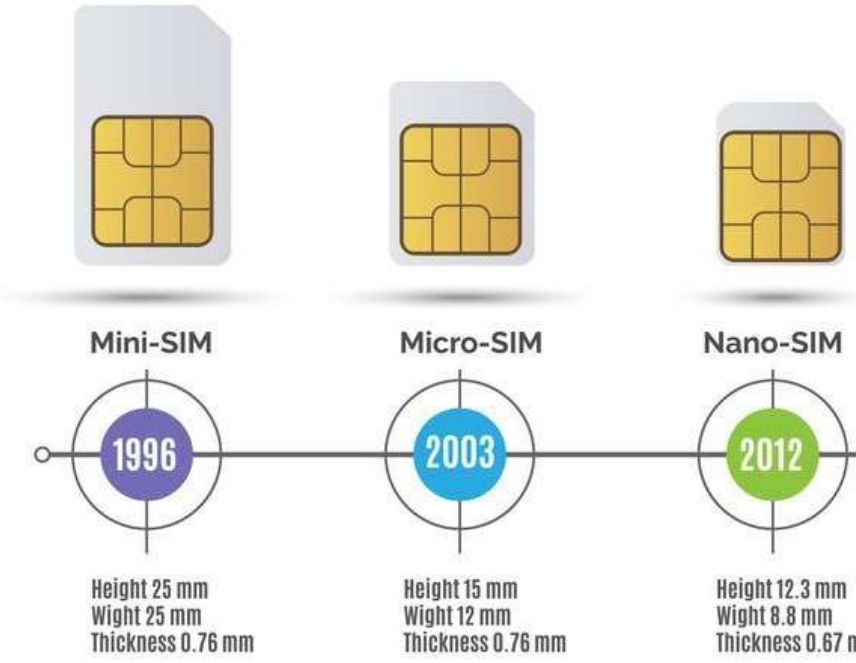
- Types IoT SIM cards:
  - Full Size (1FF-Form Factor):
    - **largest M2M SIM card**
    - **size of a credit card**
    - **phased out by smaller, modern SIMs.**
  - Mini-SIM (2FF):
    - **industry standard SIM card**
    - **Size: 25 mm x 15 mm x 0.76 mm.**
    - **used in devices like vehicles, vending machines and payment points**
  - Micro-SIM (3FF):
    - **half the size of the mini card**
    - **Size: 15 mm x 12 mm x 0.76 mm.**
    - **used in portable devices like tablets, GPS, mHealth and other mobile IoT devices.**



SIM Card Technology

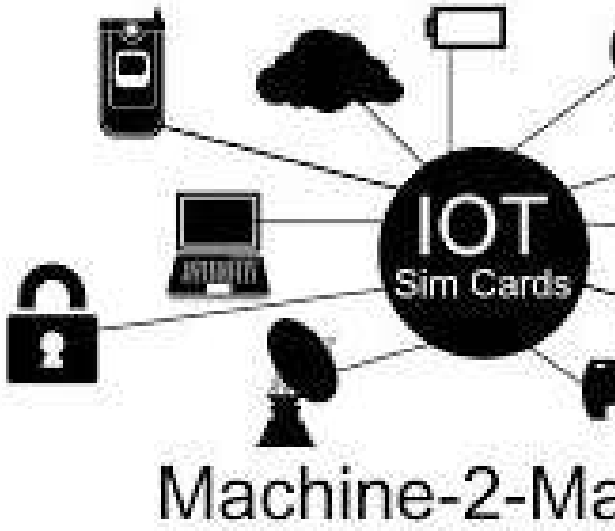
- Nano SIM (4FF):
  - **40 % smaller than the micro**
  - **Size: 12.3 mm x 8.3 mm x 0.67 mm.**
  - **variation, making it great for small IoT devices**
  - **have relatively little protection so they're not recommended for harsh environments.**
- eSIM (MFF2):
  - **Size:6 mm x 5 mm x 0.67 mm**
  - **most popular IoT SIM because of their convenient size and durability**
  - **is not removable or interchangeable.**

- SIM cards typically store a set of authentication credentials which help keep their data secure.



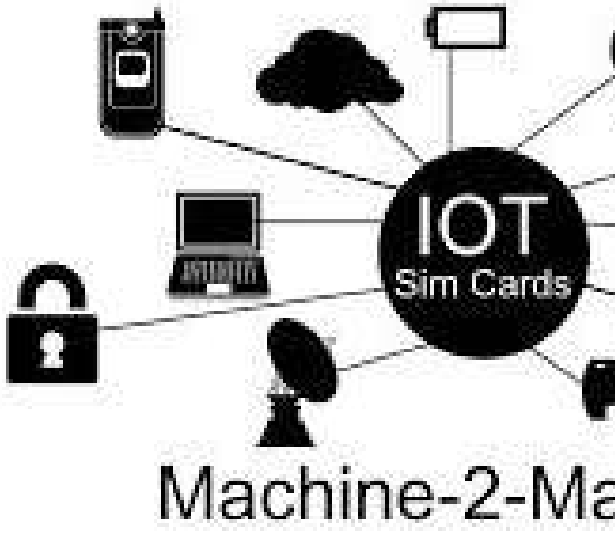
SIM Card Technology

- IoT SIM cards
  - Applications:
    - Wearables:
    - Home automation devices:
    - Agricultural sensors:
    - Healthcare monitors:
    - Logistics and fleet management sensors:



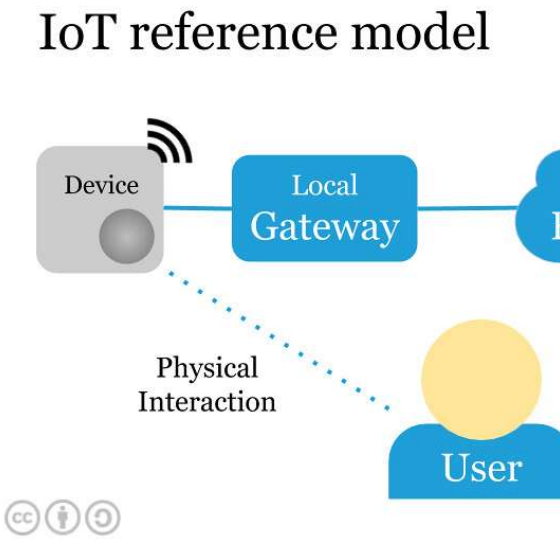
SIM Card Technology

- IoT Connectivity and Management:
  - An IoT connectivity management **platform connects and streamlines the management of IoT devices**,
  - such as M2M SIM cards.
  - Sometimes **referred to as a "connected devices platform**,“
  - the **software promotes**
    - **scalability of IoT device controls**
    - **reduces time-consuming onboarding processes.**
  - Connectivity Management Platform (**CMP**) is an application that **allows user to monitor, analyze, provision and modify our cellular Internet of Things and Machine - to - Machine (M2M) deployments.**



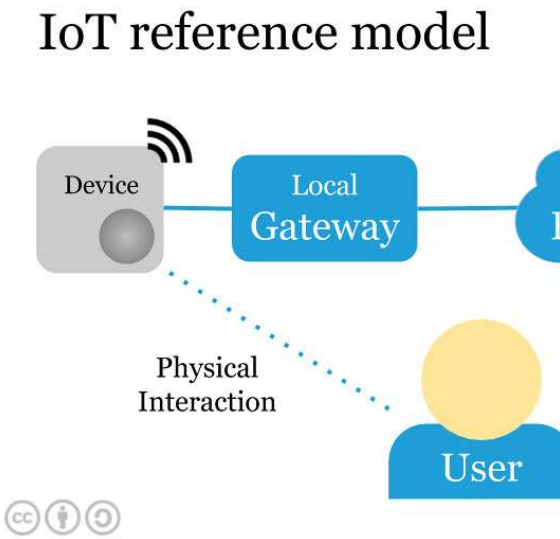
IoT Security

- In a network, **endpoints** are the devices that are **connected to the internet**, and this **includes IoT devices**.
- **Endpoint provides a potential point of entry for a bad actor to expose the network to outside risks.**
- So, as with other **endpoints, IoT devices are vulnerable.**
- **IoT security covers both physical device security and network security**, and **impacts the processes, technologies**, and **measures necessary to protect IoT devices and networks.**



IoT Security

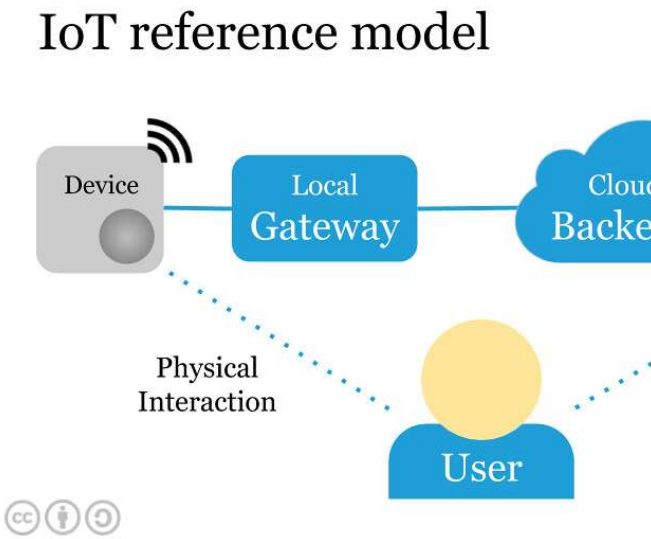
- Vulnerabilities of IoT:
  - **Limited computational abilities and hardware limitations**
  - **Heterogeneous transmission technology**
  - **Components of the device are vulnerable**
  - **Users lacking security awareness: Lack of user security awareness could expose smart devices to vulnerabilities and attack openings**



IoT Security

IoT **vulnerabilities due to** participation of the **number of layers, hardware sublayers and software in applications and services**

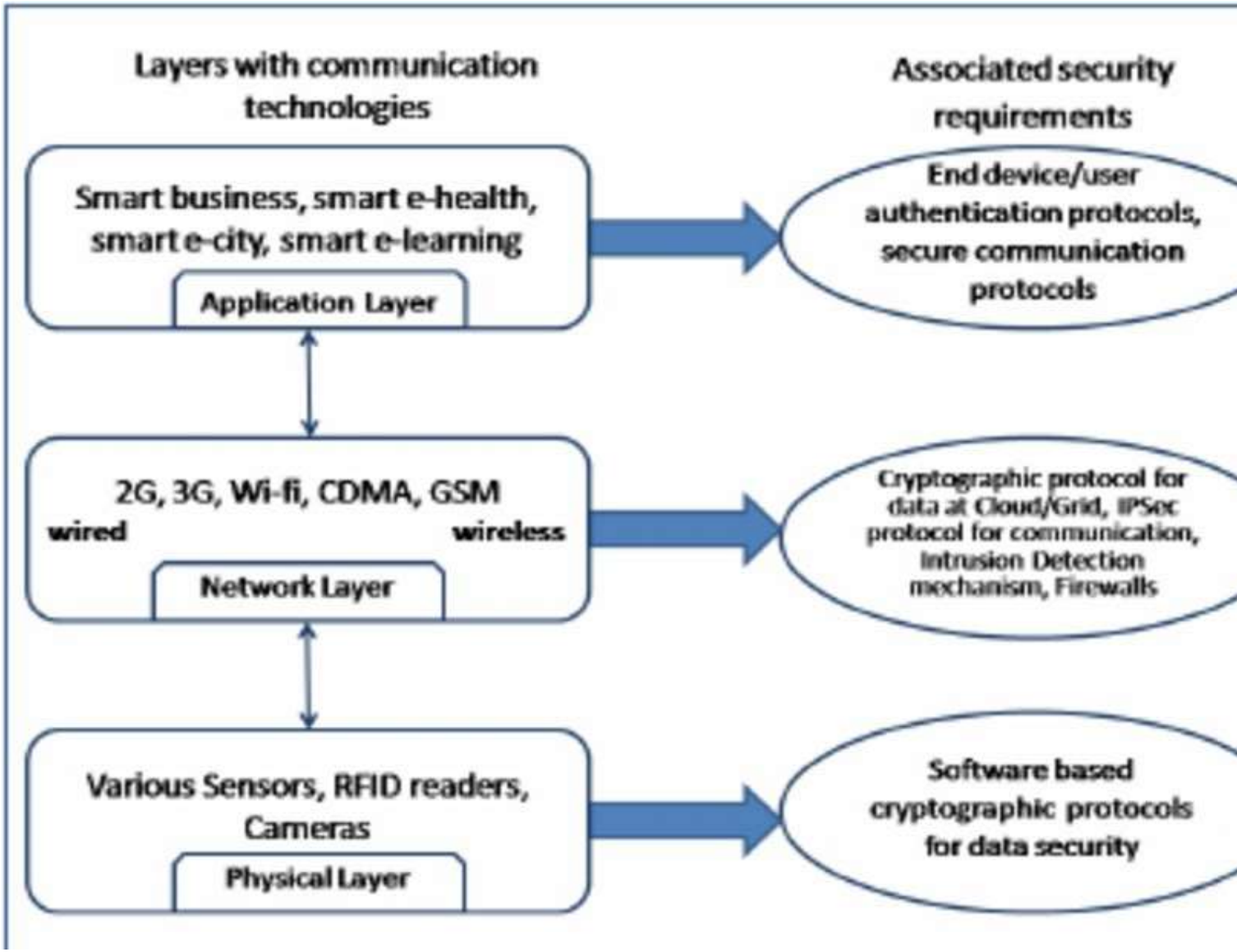
The nature of IoT Vulnerabilities varies, for example, **sensors, machines, automobiles, wearables, and so on. Each faces different kind of vulnerabilities and has complex security and privacy issues.**



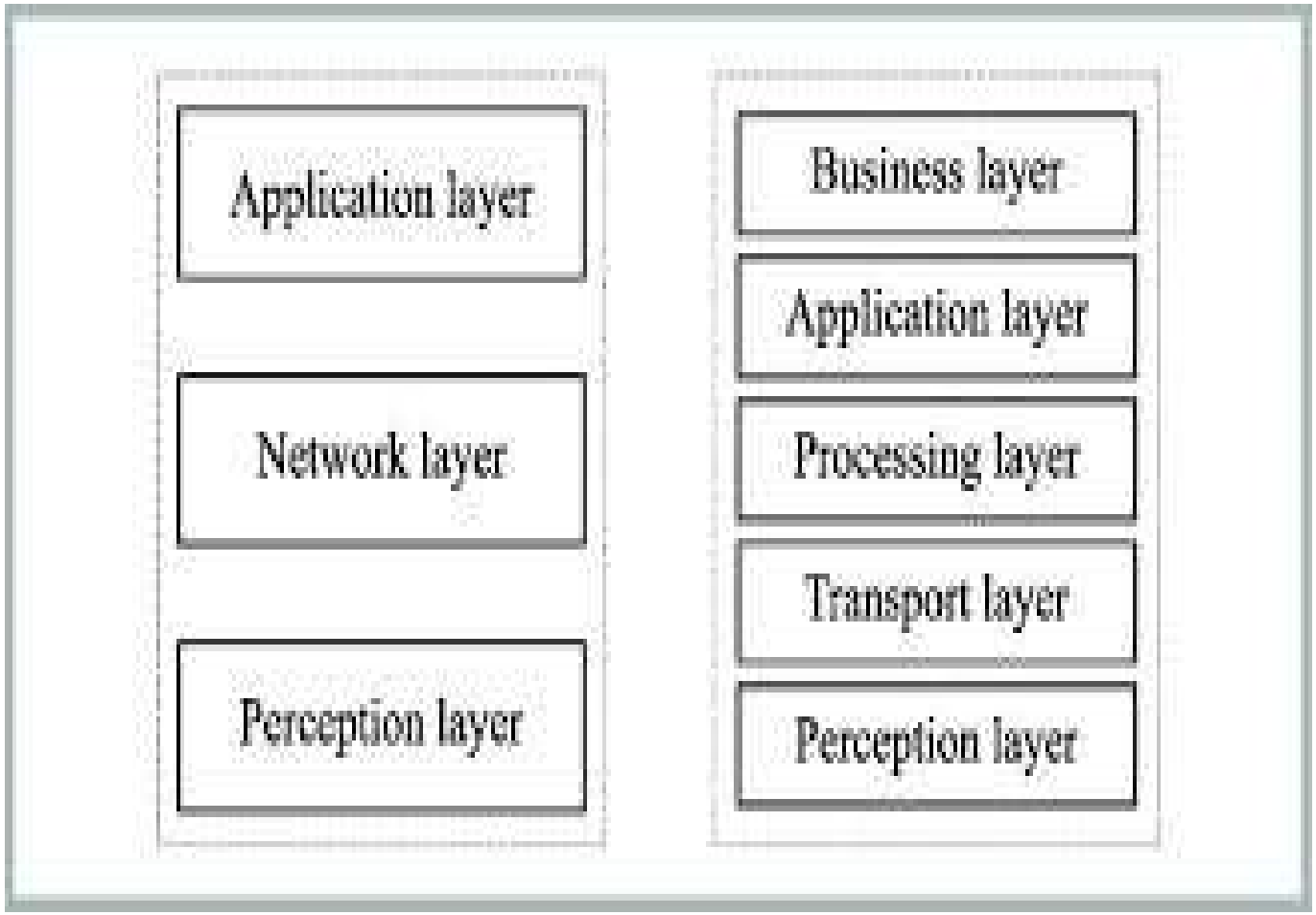
IoT Security

- Attacks:
  - Various possible attacks in different layers of IoT.
  - There are various attack surfaces available for attackers, protection considered at three different layers:
    - **Edge protection:** Ensures device, mobile app and web app integrit devices from becoming attack entry points
    - **Network protection :** Secures communication channels t man-in-the-middle attacks.
    - **Cloud protection :** Assures data privacy and prevents data leakage.

IoT Security-Security Architecture:



IoT Security-Security Architecture:



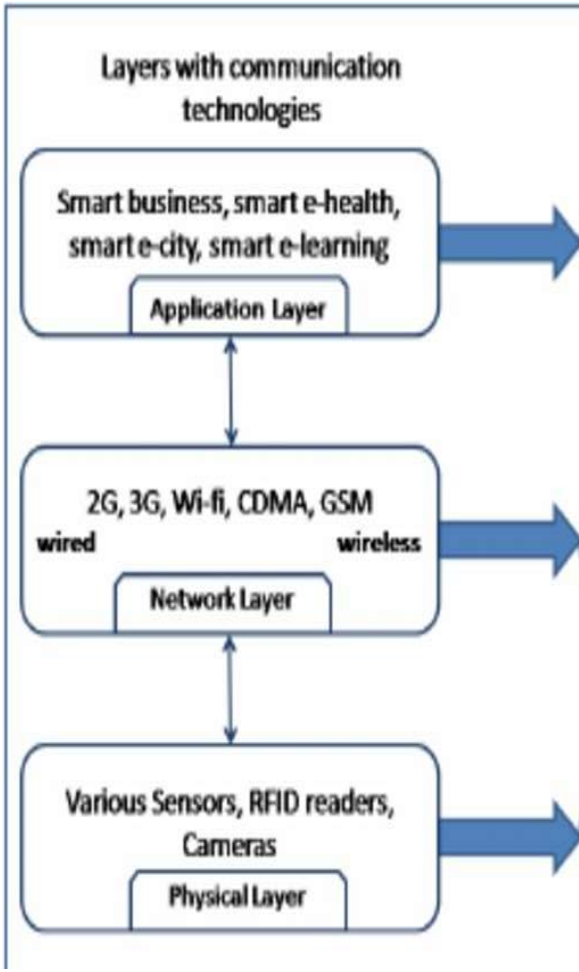
IoT Security-Security Architecture:

- Security Architecture:
  - IoT systems are often **highly complex**, requiring end-to-end security that span **cloud and connectivity layers**, and support resource-co IoT devices that often aren't powerful enough to support traditiona solutions.

IoT Security

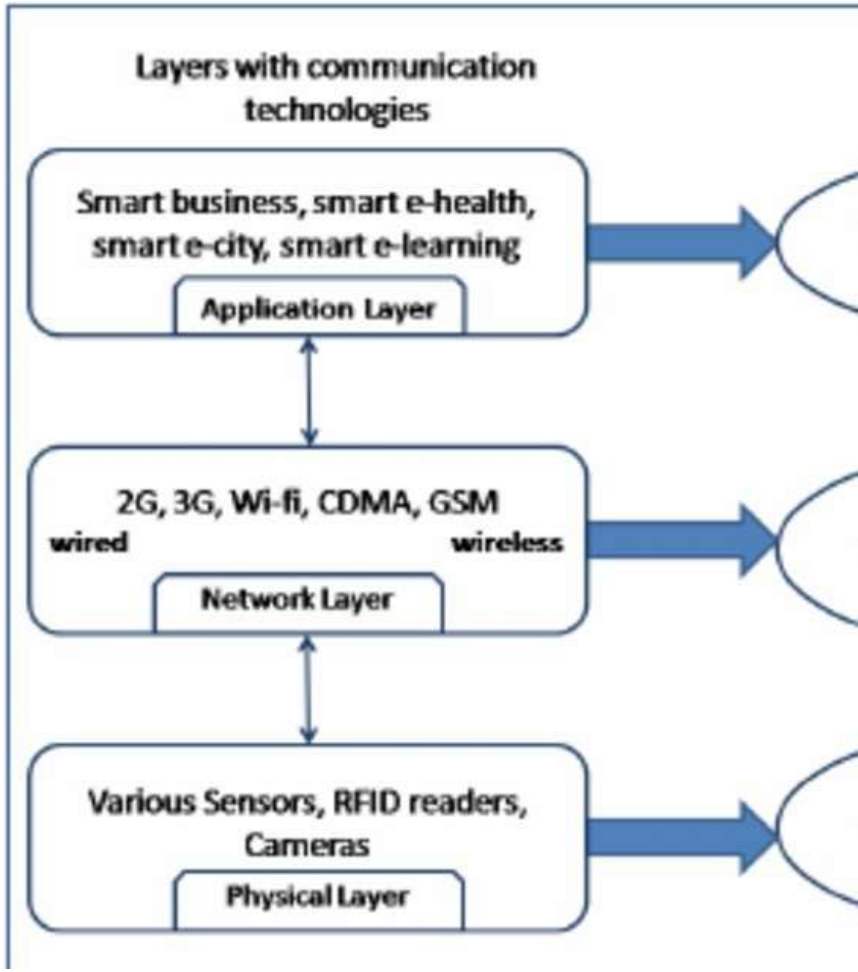
- Application layer **support user services**. This layer helps **users access IoT through the interface using PC, mobile equipment** etc. This layer also **support secure communication protocol and authentication protocols.**

- Network layer support **wired and wireless communication protocol and technology**. This layer is **responsible for dependable broadcast of data and information from the below layer.**



IoT Security

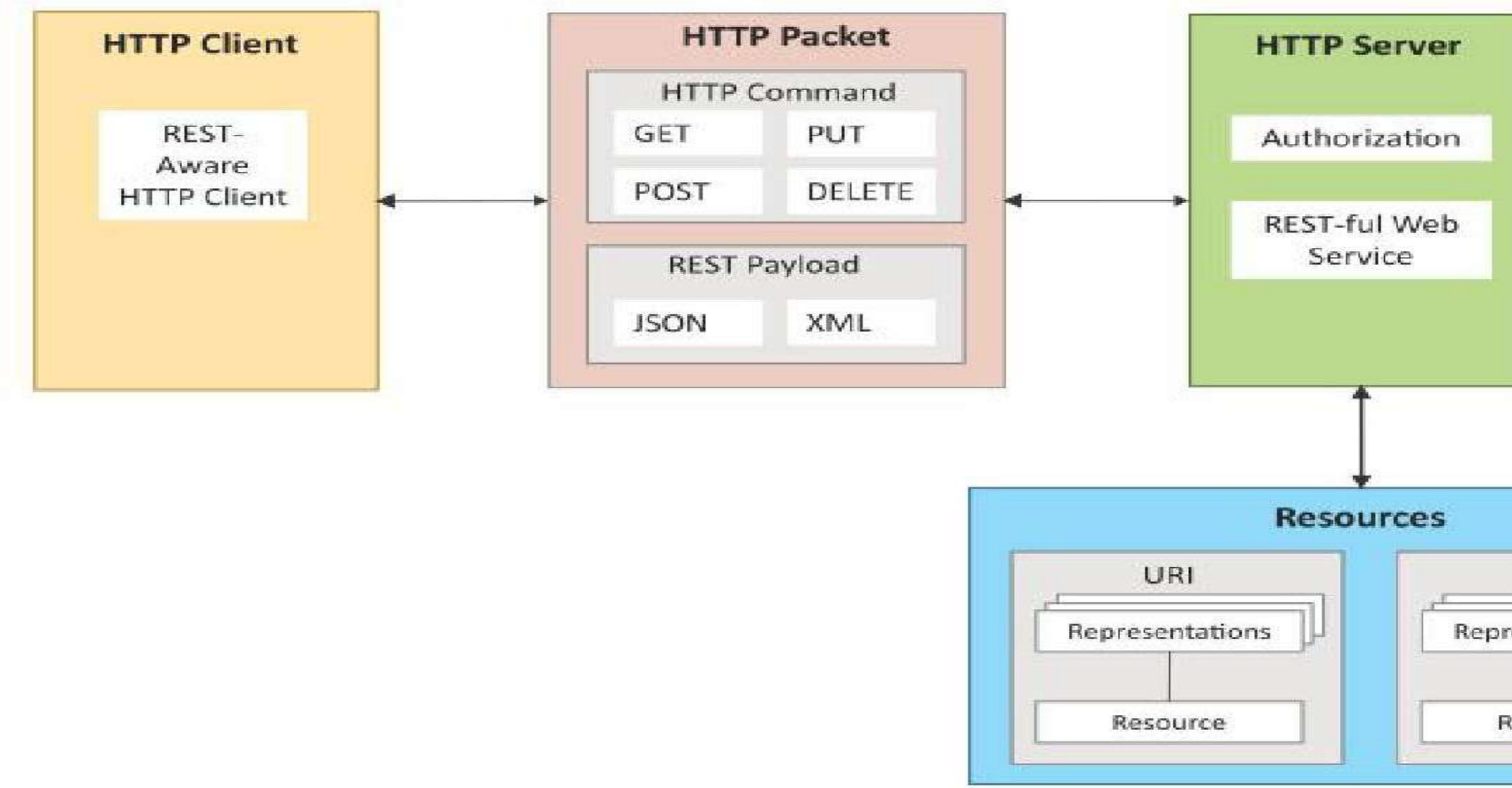
- Security Model for IoT:
  - Application layer:
    - IOT Application
    - Application support layer (Middleware technology security)
  - Transportation Layer:
    - Local Area Network
    - Core Network (Internet Security)
    - Access Network (Ad hoc security, GPRS security, WiFi security)
  - Perception Layer:
    - Perception Network (RFID security) (WSN Security)
    - Perception Node (RSN Security)



IoT Security

- Security challenges:
  - **Devices are not reachable**: Most of the time a device is not co
  - **Devices can be lost and stolen:** Makes security difficult when is not connected.
  - **Devices are not crypto-engines:** Strong security difficult processing power.
  - **Devices have finite life:** Credentials need to be tied to lifetime
  - **Devices are transportable :** Will cross borders.

IoT Communication APIs



IoT Communication APIs

- REST-based Communication APIs:
  - Representational State Transfer (**REST**) **is a set of archit principles** by which you **can design web services and web AP focus on a system's resources and how resource stat addressed and transferred.**
  - REST APIs **follow the request- response communication mode**
  - The REST **architectural constraints apply to the compo connectors, and data elements, within a distributed hype system.**

IoT Communication APIs

- WebSocket-based Communication APIs:
  - WebSocket APIs **allow bidirectional, full duplex communication between clients and servers.**
  - WebSocket APIs **follow the exclusive pair communication model**