**Building Blocks of IoT Devices?**

The internet of things denotes the connection of devices, machines and sensors to the internet. The IoT system comprises of **four basic building blocks**: sensors, processors, gateways and applications.

This are given Below:

* **Sensors** convert a non-electrical input to an electrical signal. Sensors are classified into two types: active and passive sensors. Whereas active sensors use and emit their own energy to collect real-time data, whereas passive sensor use external sources.
* **Processors** are the brain of the IoT system. They process the raw data captured by the sensors and extract valuable information. Examples of processors are microcontroller and microcomputers.
* **Gateways** are the combination of hardware and software used to connect one network to another. Gateways are responsible for bridging sensor nodes with external internet or World Wide Web.
* **Applications** provides a user interface and effective utilization of data collected. Below is his figure:



**Raspberry Pi?**

The Raspberry Pi is a low cost, **credit-card sized computer** that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. It is a capable little device that enables people of all ages to explore computing, and to learn how to program in languages. It’s capable of doing everything you’d expect a desktop computer to do, from browsing the internet and playing high-definition video, to making spreadsheets, word-processing, and playing games.

What’s more, the Raspberry Pi has the ability to interact with the outside world, and has been used in a wide area of digital maker projects, from music machines and weather stations and tweeting birdhouses with infra-red cameras. We want to see the Raspberry Pi being used by kids all over the world to learn to program and understand how computers work.

**Interfaces**

You can link devices and components to your Raspberry Pi using a lot of different types of connections. The Interfaces tab is where you turn these different connections on or off, so that your Raspberry Pi recognises that you’ve linked something to it via a particular type of connection.



* Camera — enable the Raspberry Pi Camera Module
* SSH — allow remote access to your Raspberry Pi from another computer using SSH
* VNC — allow remote access to the Raspberry Pi Desktop from another computer using VNC
* SPI — enable the SPI GPIO pins
* I2C — enable the I2C GPIO pins
* Serial — enable the Serial (Rx, Tx) GPIO pins
* 1-Wire — enable the 1-Wire GPIO pin
* Remote GPIO — allow access to your Raspberry Pi’s GPIO pins from another computer

**Other IoT devices?**

IoT devices are the nonstandard computing devices that connect wirelessly to a network and have the ability to transmit data, such as the many devices on the internet of things (IoT). Below is given some IoT devices:

1. Google Home Voice Controller

Google Home voice controller is one of the most popular IoT devices out there today. It provides voice-enabled services like alarms, lights, thermostats, volume control and lots more.

2. Amazon Echo Plus Voice Controller

Amazon Echo Plus voice controller is another popular and reliable IoT device on the market. It provides voice-enabled services like answering phone calls, setting timers and alarms, checking the weather, and lots more.

3. August Doorbell Cam

August Doorbell Cam is an IoT device that allows you to answer your door from any remote location. It constantly captures motion changes and suspicious activity in your doorstep.

4. August Smart Lock

August Smart Lock is a proven and reliable security IoT device that helps users to manage their doors from any remote location. It helps keep thieves away and provides an extra layer of security for your home.

5. Foobot

Foobot is an IoT device that can accurately measure indoor pollution. It helps to improve the air quality in houses, cafes, workplaces, and other indoor public spaces.

**What is WAMP?**

WAMP stands for Web Application messaging protocol. WAMP is an open standard WebSocket and TCP/IP subprotocol that provides two application messaging patterns in one unified protocol: Remote Procedure Calls + Publish & Subscribe.

Combining these two patterns into a single protocol allows it to be used for the entire messaging requirements of a distributed system, thus reducing technology stack complexity, as well as networking overheads.WAMP provides two messaging patterns: Publish & Subscribe (PubSub) and Routed Remote Procedure Calls (RPCs).

**Publish & Subscribe (PubSub)**

Publish & Subscribe (PubSub) is an established messaging pattern where a component, the Subscriber, informs the router that it wants to receive information on a topic (i.e., it subscribes to a topic). Another component, a Publisher, can then publish to this topic, and the router distributes events to all Subscribers.

**Routed Remote Procedure Calls (RPCs)**

Routed Remote Procedure Calls (RPCs) rely on the same sort of decoupling that is used by the Publish & Subscribe pattern. A component, the Callee, announces to the router that it provides a certain procedure, identified by a procedure name. Other components, Callers, can then call the procedure, with the router invoking the procedure on the Callee, receiving the procedure's result, and then forwarding this result back to the Caller. Routed RPCs differ from traditional client-server RPCs in that the router serves as an intermediary between the Caller and the Callee.

<https://mediacomem.github.io/comem-archioweb/2021-2022/subjects/wamp/?home=MediaComem%2Fcomem-archioweb%23readme#6>

**What is Django?**

Django is a high-level Python web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of web development, so you can focus on writing your app without needing to reinvent the wheel.

It’s free and open source. In other words, if you were creating a website from scratch you would need to develop these components yourself. By using a framework instead, these components are already built, you just need to configure them properly to match your site.

Django offers a big collection of modules which you can use in your own projects. Primarily, frameworks exist to save developers a lot of wasted time and headaches and Django is no different.

You might also be interested in learning that Django was created with front-end developers in mind. “Django’s template language is designed to feel comfortable and easy-to-learn to those used to working with HTML, like designers and front-end developers. But it is also flexible and highly extensible, allowing developers to augment the template language as needed.”

**Portable**

Django is written in Python, which runs on many platforms. That means that you are not tied to any particular server platform, and can run your applications on many flavours of Linux, Windows, and Mac OS X. Furthermore, Django is well-supported by many web hosting providers, who often provide specific infrastructure and documentation for hosting Django sites.

**What is Skynet?**

<https://support.skynetlabs.com/#why-does-skynet-matter>