

What is the Internet of Things?

The Internet of Things, or IoT, refers to the billions of physical devices around the world that are now connected to the internet, all collecting and sharing data. The Internet of Things (IoT) unites physical objects with the virtual world. Intelligent devices and machines are connected to each other and the Internet. They capture relevant information about their direct environment, then analyze and link it. The devices perform specific tasks on that basis. A sensor, for example, measures the temperature outside and the smart device it is installed in responds by turning up the heating. All of that is done automatically, without users taking any actions themselves. Users can still control the IoT devices remotely if they wish, for example, using an app on their smartphone.

What is an example of an Internet of Things device?

Pretty much any physical object can be transformed into an IoT device if it can be connected to the internet to be controlled or communicate information.

A lightbulb that can be switched on using a smartphone app is an IoT device, as is a motion sensor or a smart thermostat in your office or a connected streetlight. An IoT device could be as fluffy as a child's toy or as serious as a driverless truck.

IOT Components

An IOT system comprises three basic Components.

1. The Things -sensors actuators etc
2. The Network and protocols
3. The Platforms, Apps and services

1. The Things – Sensors and Devices

In contrast to computers and tablets which are the main devices currently connected to the Internet.

Internet of things devices will mainly be:

- Low Power- Power usage and computational Power.
- Low cost
- Wireless

Examples are Simple sensors – temperature, pressure etc

To turn an everyday object like a house or a car into a smart house or car or a “**thing**” will require that the object has:

- A unique address – IPv6 address
- A way to connect to a network – Wireless

2. IOT Networks And Protocols

The Internet of things will utilize the existing networking infrastructure, technologies and protocols currently used in homes/offices and on the Internet, and will introduce many more.

Protocols are designed to operate at a particular level in the networking stack.

TCP/IP uses a 4 level model and we will discuss IOT networking using this model.

However because of the requirement for **low powered end devices** there will be major developments in the Wireless connectivity protocols.

Wi-Fi and **Bluetooth** are being actively developed for low powered applications and there are new connection technologies like LPWAN, ZigBee, 6LoWpan and Thread.

At the networking level **IPv6** is set to become the standard, but in the intermediate time frame IPv4 will also be used. See IPV6 Basics and IPv4 addressing basics

At the **application level** there are a host of new protocols. Some have been available for a long time like **HTTP** and **MQTT**, whereas others have been developed especially for the IOT e.g. **COAP**.

3. IOT Platforms, Apps and Services

An IOT platform combines several IOT functions in one.

It can collect and distribute data, convert data between protocols, store and analyse data.

They are available as **cloud based** and standalone platforms and are available from many companies both large and small.

Examples

- Amazon Web services (AWS)
- IBM Watson Bluemix
- Microsoft Azure

Basics of Blockchain

What is blockchain technology?

Blockchain is the digital and decentralized ledger that records all transactions. Every time someone buys digital coins on a decentralized exchange, sells coins, transfers coins, or buys a good or service with virtual coins, a ledger records that transaction, often in an encrypted fashion, to protect it from cybercriminals. These transactions are also recorded and processed *without* a third-party provider, which is usually a bank.

Main features of blockchain:

1. **Immutability:** Immutability refers to the feature that any new block added to the blockchain is referred to as the previous block. This makes a permanent chain of blocks linked with each other. Also, no information present in the blocks can be modified once it is deployed on the blockchain. This maintains the integrity of the blocks and makes it immutable.
2. **Verifiability:** As the ledger is decentralized, the blockchain has a copy of itself on all the nodes participating in the blockchain network. This makes it possible to verify the transactions easily and securely as there is no single point of failure.
3. **Distributed Consensus:** Distributed consensus is a protocol that decides which participant of the node will add the new transaction on the blockchain. As it is a distributed network, if most participants agree, only the proposed transaction is added on the blockchain.

Terms Related to Blockchain

There are a lot of terms that are generally used in blockchain technology. It is essential to know these terms to understand the **blockchain basics**.

1. Block

Block is the basic structure in a blockchain that holds information and other primary details such as timestamp, the hash of the previous block, the hash of the current block, and further details.

2. Timestamp

The timestamp is the details of the exact time when the block is created in the blockchain.

3. Transactions

The transactions refer to the transactional data that has been collected and stored inside the block.

4. Hash

Hash is the cryptographic hash of the block generated by applying an appropriate hash function, chosen for the blockchain, on the data stored in the block.

5. Previous Hash

Previous hash is the hash of the last block, which is used to link the block from the last block present on the blockchain.

Different Types of Blockchain

There are presently three types of blockchains: public blockchains, private blockchains, and consortium blockchains.

1. Public Blockchains

These blockchains store data on a public ledger, which can be viewed by every user on the internet. Users can modify and verify the blocks and then add these blocks on the blockchain.

2. Private Blockchain

These blockchains are managed by an authority and only verified, and permitted users can access the data on the blockchains.

3. Consortium Blockchains

A group of organizations manages these blockchains, and they can only modify the blockchains.

How are blockchain transactions validated?

Processing transactions on blockchain also comes with the issue of ensuring that the same cryptocurrency coin isn't being spent twice. That's where transaction validation comes into play.

There are two primary ways that transactions on blockchain are validated: proof-of-work (PoW) and proof-of-stake (PoS).

Bitcoin runs on the PoW model. What happens with PoW is that cryptocurrency miners (a fancy term for people with really high-powered computers) compete against one another to solve complex mathematical equations that are a result of the encryption protecting transactions on a blockchain network. The first miner to solve these equations, and in the process validate a block of transactions, receives what's known as a "block reward." For bitcoin, a block reward is paid as a fraction of digital bitcoin.

The other primary validation method is PoS. Rather than using a ton of electricity in a competition to solve equations, the PoS method awards the owners of virtual coins the opportunity to validate transactions in a deterministic fashion. In even plainer terms, the more coins you own of a virtual currency operating on the PoS model, the more likely you are to be chosen to validate blocks and add to the blockchain.

What is Augmented Reality?

Augmented Reality (AR) is a perfect blend of the digital world and the physical elements to create an artificial environment. Apps which are developed using AR technology for mobile or desktop to blend digital components into the real world. The full form of AR is Augment Reality.

Example: AR technology helps to display score overlays on telecasted sports games and pop out 3D photos, text messages, and emails.

What is Virtual Reality?

Virtual Reality (VR) is a computer-generated simulation of an alternate world or reality. It is used in 3D movies and video games. It helps to create simulations similar to the real world and "immerse" the viewer using computers and sensory devices like headsets and gloves.

Apart from games and entertainment, virtual reality is also used for training, education, and science. The full form of VR is Virtual reality

KEY DIFFERENCES:

- AR augments the real-world scene whereas VR creates completely immersive virtual environments.
- AR is 25% virtual and 75% real while VR is 75% virtual and 25% real.
- In AR no headset is needed on the other hand in VR, you need headset device.
- With AR, end-users are still in touch with the real world while interacting with virtual objects nearer to them, but by using VR technology, VR user is isolated from the real world and immerses himself in a completely fictional world.

AR Vs VR

AR

The system augments the real-world scene

In AR User always have a sense of presence in the real world

AR is 25% virtual and 75% real

This technology partially immerses the user into the action

AR requires upwards of 100 Mbps bandwidth

No AR headset is needed.

With AR, end-users are still in touch with the real world while interacting with virtual objects nearer to them.

It is used to enhance both real and virtual worlds.

VR

Completely immersive virtual environment

In VR, visual senses are under control of the system

VR is 75% virtual and 25% real

This technology fully immerses the user into the action

VR requires at least a 50 Mbps connection

Some VR headset device is needed.

By using VR technology, VR user is isolated from the real world and immerses himself in a completely fictional world.

It is used to enhance fictional reality for the gaming world.

Single Page Application

Single page application (SPA) is a web application that fits on a single page. All your code (JS, HTML, CSS) is retrieved with a single page load. And navigation between pages performed without refreshing the whole page.

Using Angular routes in a single-page application

In a Single Page Application (SPA), all of your application's functions exist in a single HTML page. As users access your application's features, the browser needs to render only the parts that matter to the user, instead of loading a new page. This pattern can significantly improve your application's user experience.

To define how users navigate through your application, you use routes. You can add routes to define how users navigate from one part of your application to another. You can also configure routes to guard against unexpected or unauthorized behaviour.

Create a single page application step by step:

1. Need a html template.
 - a. I use a bootstrap html template:
2. Define route.
 - a. Home, About, services, portfolio, team, clients, contact
 - b. Add Attributes href="#!yourRouteName"
3. Divided page content based on route.
 - a. Example: About route content
 - i. About.html
 - ii. About page content display page content.
4. The ng-app directive define the application, the ng-controller directive defines the controller.
5. Write AngularJS code inside a js file.
6. Create a js file app.js
 - a. Inside this file you define your app.
 - b. `var app = angular.module("myApp", ["ngRoute"]);`
 - c. Inside this file you define route.
7. Create another js file appCtrl.js
 - a. Inside appCtrl file you declare all controller. You can create multiple file based on controller.
 - b. Create a controller and create a object:
8. `app.controller("aboutCtrl", function ($scope) {`
9. `$scope.aboutInfos=[{dateTime:'March 2011', title:'An Agency is`
10. `Born',message:'Lorem ipsum dolor sit amet, consectetur adipisicing`
11. `elit. Sunt ut voluptatum eius sapiente, totam reiciendis temporibus qui`
12. `quibusdam, recusandae sit vero unde, sed, incidunt et ea quo dolore`
13. `laudantium`
14. `consectetur!',imageUrl:'about/1.jpg'},{dateTime:'2009-2011',title:'Our`
15. `Humble Beginnings',message:'Lorem ipsum dolor sit amet, consectetur`
16. `adipisicing elit. Sunt ut voluptatum eius sapiente, totam reiciendis`
17. `temporibus qui quibusdam, recusandae sit vero unde, sed, incidunt et`
18. `ea quo dolore laudantium consectetur!',imageUrl:'about/2.jpg'}];`
19. `});`
19. Add attribute index.html page Inside html tag ng-app="myApp"
20. You show every page data base on controller.
 - a. Example: about.html
 - b. About page Controller: aboutCtrl

21. You already know we have a about.html page. I use aboutCtrl inside about.html page. The controller attribute use the ng-controller directive and use it to top of the page.

Display data:

Display data using about.html page. I used ng-repeat directive.

```
<ul class="timeline">
  <li ng-repeat="aboutInfo in aboutInfos" ng-class-even="'timeline-inverted'">
    <div class="timeline-image">
      
    </div>
    <div class="timeline-panel">
      <div class="timeline-heading">
        <h4>{{aboutInfo.dateTime}}</h4>
        <h4 class="subheading">{{title}}</h4>
      </div>
      <div class="timeline-body">
        <p class="text-muted">{{aboutInfo.message}}</p>
      </div>
    </div>
  </li>
  <li class="timeline-inverted">
    <div class="timeline-image">
      <h4>Be Part
        <br>Of Our
        <br>Story!</h4>
    </div>
  </li>
</ul>
```