Lesson 1
Key Terms- Application Framework, Stack, OSGi, Thread, …

Framework

- An abstraction which provides reusable software environment and software with generic functionalities

Application framework

- A framework that supports a fundamental structure and provides a skeleton which leads to development of applications, and that provides reusable development environment with generic functionalities
- The framework to help to selectively change and add new functionalities and build the application.

Application Programming Interface (API)

- Software that enables access to an app, application
- or service
- Software that runs on a local or remote computing platform, that sends messages from client (user) end for the server (application) end, and receives server-messages for the client

Device Interfaces

- Device interface for communication over the network and communication circuit/port(s) that also includes a middleware

Middleware

- Creates IPv4, IPv6, 6LowPAN, MQTT, COAP, LWM2M, REST and other communication protocol stacks

Web-Application Framework

- A framework that supports the fundamental structure and skeleton, which leads to the development of web-applications including, web APIs and resources
- Provides reusable web-application development-environment with generic functionalities
- To help the selectively change and add new functionalities and build web-application or web-service and APIs

Community

- A software development initiative or model, which uses collective efforts for software development for an open source project
- An approved open-source independent foundation for distribution, For example, OSGi
- May also hold copyright and commit to follow distribution practices, contributor agreements and licensing, for example, Apache foundation, for the Apache server

Component

- Component is an abstraction for a core set of frameworks, services or software functions that can be reused after reconfiguring them for providing solutions

OSGi (Open Service Gateway Initiative)

- A development framework for Java
- Used for deploying modular software programs, libraries and bundles

OSGi bundle

- Refers to a tightly coupled collection of classes, jars, and configuration files which are dynamically loadable
- The external dependencies explicitly declared, when existing

Software stack

- A set of frameworks and services that is minimum need for an intended complete solution
- Examples: Eclipse IoT Stack for IoT and Berkeley Data Analytics Stack (BDAS) for analytics solutions
- A stack may be community supported. For example, OSGi supports Eclipse IoT Stack. Stack provides a set of software functions, frameworks, packages, modules or subsystems

Why use Software stack?

- Creates a complete platform to support the applications or services when installed and configured on individual systems or added to the templates for automatic installation

Sandbox Server

- A server with data from the source code distributions, and other collections of contents, data or sets of the codes, proprietary or public, and that is protected from changes intentional or unintentional
- Functioning as a working directory, test server or development server

Client usages of the Sandbox Server

- Sandbox client uses the server's minimal functionalities, same environment variables or access to an identical database of the server for the development of specific functionalities and application(s)
- For testing a developed application

Script

- A piece of code in text form, which runs using an Interpreter, which interprets them at the runtime
- Scripting languages are JavaScript, JSON, Perl and PHP, ……..

Exception

- A signal or object which represents occurrence of an event, exceptional condition or conditions on which another set of codes needs to run
- That set of codes may be at a callback() or catch() function

Throwing an Exception

- Refer to exception condition thrown, sent to the system on the occurrence during execution of a function or set of statements

Message

- Message is a general term, which refers to communicating data, request, query or response during communication from one-end, say a client, server, script, function, method or API to another end.

Graphic User Interfaces (GUIs)

- Refer to computer screen display tools, such as status bar, task bar, buttons, check-boxes, menus and dialog boxes which enable easy interactions of APIs between the user and App, Application or Service software

Python

- A high-level language in which codes are interpreted at runtime, expresses code in fewer lines compared to Java or C++, and supports big libraries and open source codes on a computing platform or computer.
- A version is Cpython, which is a community, Python software foundation manages

Event model

- Usages of the functions, methods, scripts or APIs which run on eventing
- The scripts can run in multitasking mode on the multiple events
- Dour events occur then for each event a script or function runs and four tasks, each executing multiple scripts with run on eventing.

Integrated Development Environment (IDE)

- Means a set of software components and modules which provide the software environment for developing and prototyping

Operating System

- A system software which facilitates the running of processes
- Allocation of memory
- System calls to the Ios
- Facilitates use of network subsystems

Operating System

- Manages devices
- Priority allocations of processes and threads
- Enables multitasking and running of number of threads
- Enables many system functions, such as display using the given computing-device hardware

Real-time Operating System (RTOS)

- An OS that enables real-time execution of processes on computing and communication hardware
- Processes uses prioritization and run as per assigned priorities
- Priority assignment enables the execution of processes in real time

Multitasking or Multithreading Software

- Means execution of multiple tasks or threads, according to some plan
- An OS or RTOS supervises the running of the tasks and threads

Multitasking or multithreading Time Slices or Priority Allocations

- May run when active or unblocked and run according to a sequence or
- One after one in allotted time slices or
- According to the priorities assigned for each or run first called task first

Device Interfaces

- A connectivity interface consists of communication APIs, device interfaces and processing units
- Software commands the action on the message or information received followed by hardware port outputs for the actuators

Summary

We learnt key terms such as

- Web Application Framework
- Web API
- Stack
- Sandbox Server
- IDE

Summary

We learnt key terms such as

- OSGi
- GUI
- Exception
- RTOS

End of Lesson 1 on
Key Terms- Application Framework, Stack, OSGi, Thread, …