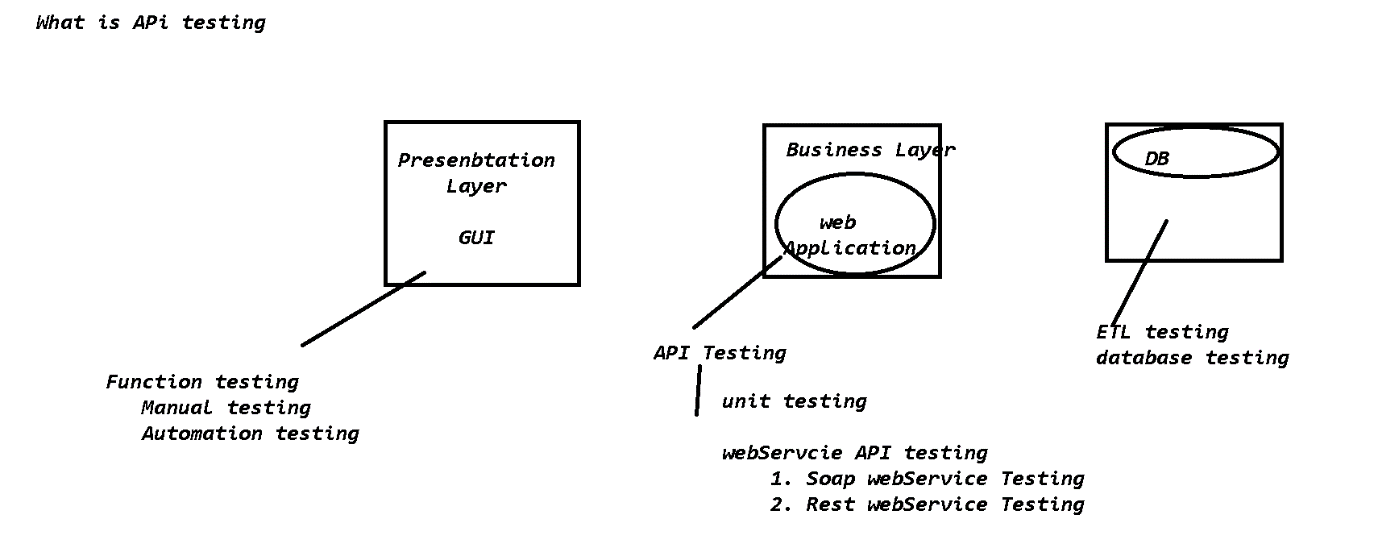
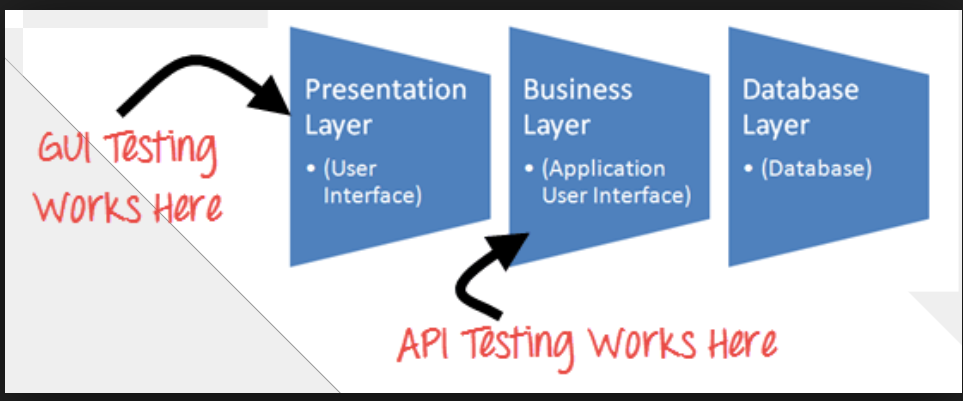
Web Service introduction

API testing

* API testing is a type of software testing where application programming interfaces (APIs) are tested to determine if they meet expectations for functionality, reliability, performance, and security.
* Testing the application in Source code layer (Business layer) is called API testing
* Testing interface between two Application is also called API testing
* Testing is done without browser is called API testing

******

SOA [Service Oriented Architecture]



***Difference between Frontend & Backend Testing***

|  |  |
| --- | --- |
| *Front End testing* | *Back End testing* |
| *Front-end is always performed on GUI* | *Back-end testing is done on Business layer and engineer should have knowledge on Data base and backend* |
|
| *Testing becomes slower because it have to wait for browser rendering time* | *It is faster and saves time.* |
| *Performance testing Is not easy in front-end* | *Performance testing is easy and faster* |
| *Adoc, usability and compatibility testing is possible* | *Adoc, usability and compatibility testing is not possible* |
| *Manual & selenium automation will be done at front end(BBT)* | *white Box , Webservice testing will be done in the backend(WBT. GBT)* |
| *End to end testing is essay* | *End to end testing is difficult* |
|  |  |
| *Tools: selenium, QTP, test complete etc* | *SOAP UI, POSTMAN, RESTCLIENT, RESTASSURED, testNG , JUNIT* |

***Difference between Unit & API Testing***

|  |  |
| --- | --- |
| UNIT testing [WBT] | API testing [Webservice testing] |
| Done by development team | Done by manual testing , API testing team or SDET team |
| Should have knowledge on source code & database | Should have knowledge on expectations and agreements of API [API –Swagger Doc] |
| Test each and every source code of the application | Test the data flow between 2 application |
| It is a white box testing | It is a kind of grey box testing |
| Tools: eclipse+testNG | Tools: SOAP UI, POSTMAN, RESTCLIENT, RESTASSURED |

API testing Types

Types of API testing

1. Whitebox testing/ Unit Testing

2. WebService Testing

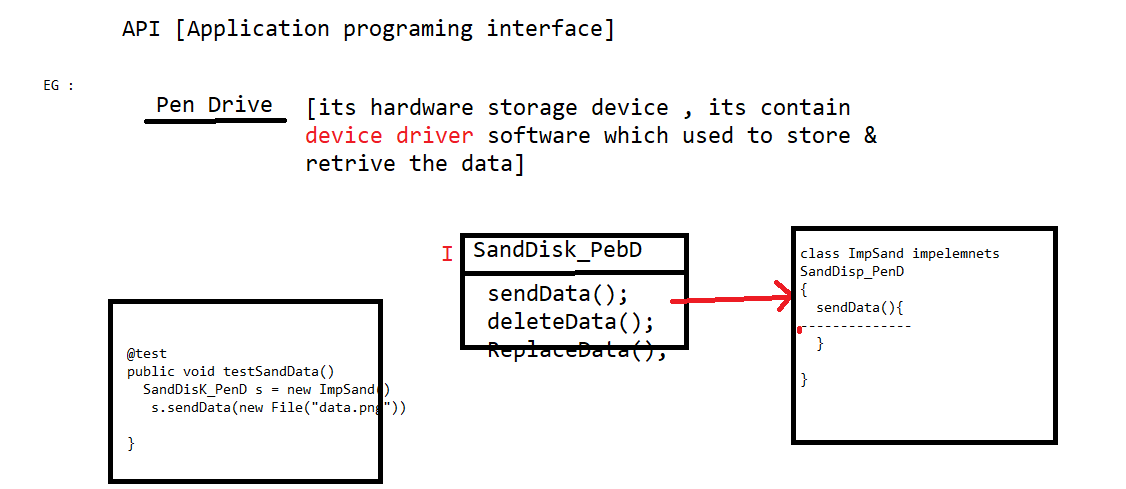
2.1 Soap WebService API-testing

2.2 RESTFull WebService API-testing

**Unit Testing [White Box testing or Unit API testing ]**

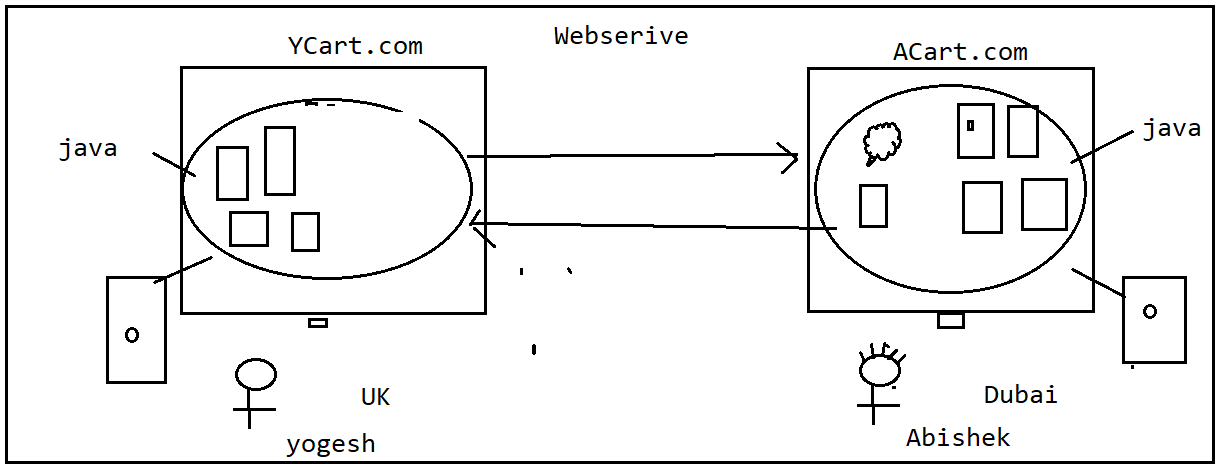
* Testing the business logic of the source code using another program is called unit testing
* White Box testing is also called as unit testing
* In order to automate unit test case, we have to go for Unit testing tools like Junit / TestNG
* Few java developers say Unit testing is also part of API Unit test

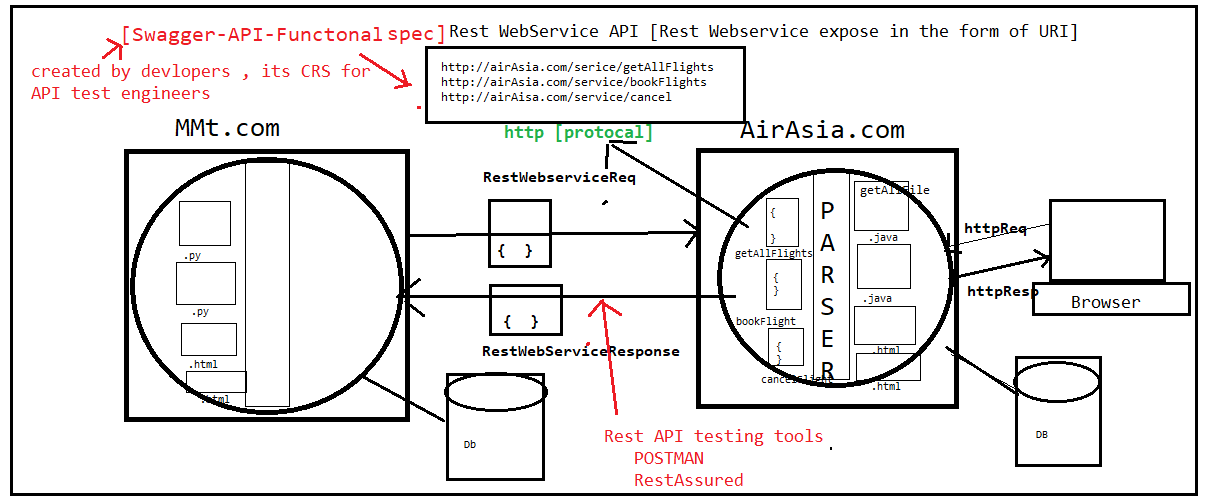
EG : Antivirus application , device drivers , JDBC, WebDriver

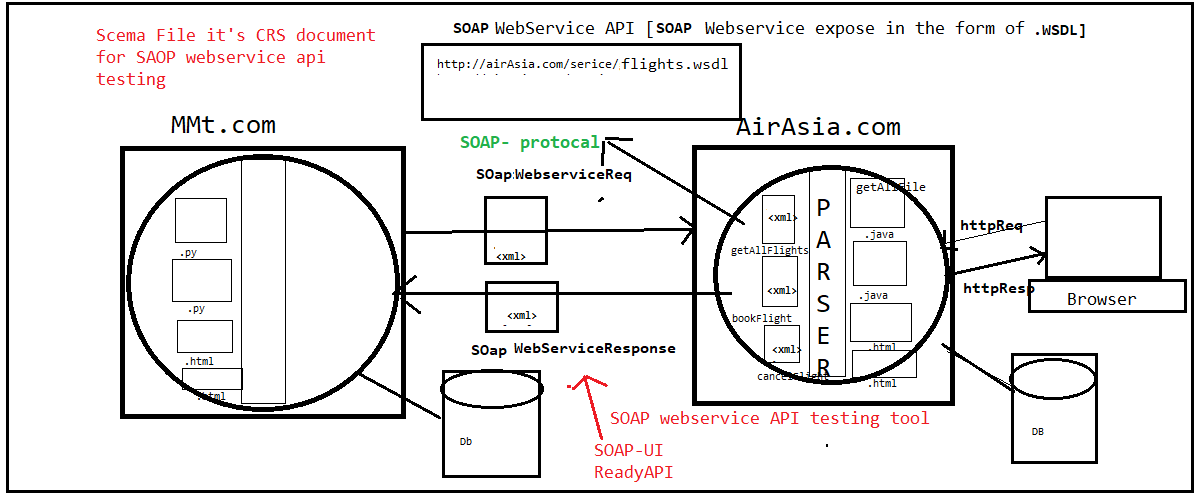
****

**Web service**

* Web Services is the mechanism or the medium of communication through which two applications / machines will exchange the data irrespective of their underline technology
* Web services help 2 application exchange information with each other when application running in same or different platform.
* Any service available on web is called webservice
* All Web service are API , but API are not webservice
* WebService help us to share the functionality of one application to any other application without sharing the source code & database data, even though both application running in same or different platform







Why WebService Testing

* The purpose of *Web Service Testing* is to verify that all of the APIs exposed by your application working as expected or not? WRT functionality & performance, security
* All web service is exposed via API
* Testing request and response of the API is called webservice testing
* Webservice provider as to test all the API’s , because to make sure all the functionality which is exposed via web service is working or not ?

What is SOAP

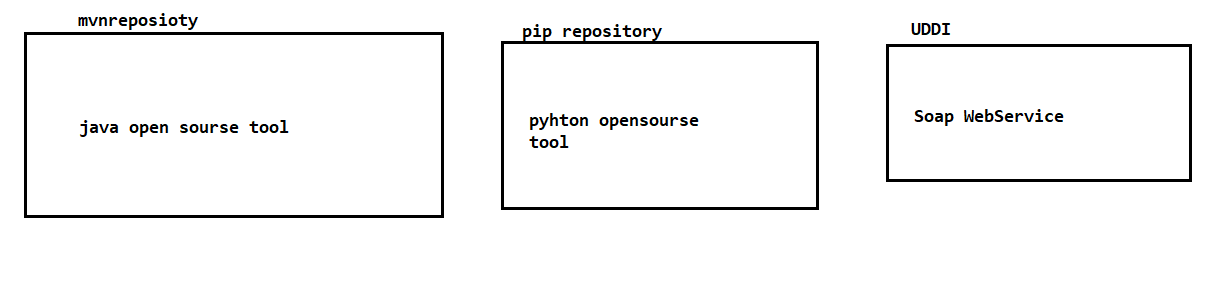
* SOAP is a simple XML-based protocol to let applications exchange information over HTTP.
* SOAP it uses XML to exchange information between applications.
* SOAP provides a way to communicate between applications running on different operating systems, with different technologies and programming languages.
* SOAP stands for Simple Object Access Protocol
* SOAP is a communication protocol
* SOAP is for communication between applications
* SOAP communicates via Internet
* SOAP is platform independent
* SOAP is language independent
* SOAP is based on XML
* SOAP is simple and extensible

What is WSDL

* An WSDL document describes a soap web service. It specifies the location of the service, and the methods of the service, using these major elements:

|  |  |
| --- | --- |
| * Element | * Description |
| * <types> | * Defines the (XML Schema) data types used by the web service |
| * <message> | * Defines the data elements for each operation |
| * <portType> | * Describes the operations that can be performed and the messages involved. |
| * <binding> | * Defines the protocol and data format for each port type |

UDDI

* UDDI is an XML-based standard for describing, publishing, and finding web services.
* UDDI stands for **Universal Description, Discovery, and Integration.**
* UDDI is a specification for a distributed registry of web services.
* 

Soap WebService API Testing

* When two application exchange information via **soap** protocol, which is based on xml based , testing those request and response is called Soap webservice testing
* SOAP Webserive exposed via .WSDL file
* EG : SOAP API

<http://www.dneonline.com/calculator.asmx?WSDL>

<http://map.google/api/soap/service.wsdl>

What is Rest

* REST stands for REpresentational State Transfer.
* It means when a RESTful API is called, the server will transfer to the client a representation of the state of the request
* REST is an architectural style for developing web services. REST is popular due to its simplicity and the fact that it builds upon existing systems and features of the internet's http resource.
* In order for an API to be RESTful, it has to adhere to 6 constraints:

Why JSON is Popular in RestFull Webservice

* Java Script Object Notation
* It’s a Programing language to exchange information between 2 application
* The JSON format is syntactically identical to the code for creating JavaScript objects.

Because of this similarity, a JavaScript program can easily convert JSON data into native JavaScript objects.

* JSON is a lightweight format for storing and transporting data
* All Browser & Mobile UI can easily consume json language
* NO SQL databases can directly store the data in the from of JSON (MangoDb, Casendra )
* Platform independent

RestFull WebSerive Api Testing

* When two application exchange information via http protocol , which is based json/xml/text/html , testing those request and response is called Rest webservcie testing
* Rest API Webserice are exposed via URI
* EG : Rest API

[http://map.google/api/getlocation](http://map.google/api.getlocation)

<http://map.google/api/gettraffice?src=‘value’&> dest=value

<http://map.google/api/gettraffic>

Difference between SOAP/ Rest

|  |  |
| --- | --- |
| SOAP Web Service | Rest Web Service |
| SOAP is a protocol | REST is Architectural style |
| Simple Object Access Protocol | Representational State Transfer |
| SOAP can't use REST Web services, it is a protocol | REST can use soap as well since it is concept |
| SOAP expose the services (Business Logic) via .WSDL file | REST expose the service (Business Logic) via URI |
| SOAP designed using too much standards | REST does not define too much standards |
| SOAP permits XML data format only | REST permits different data formats- Plain text, html, JSON, XML ,JS |
| SOAP requires more bandwidth and resources | REST requires less bandwidth and resource than SOAP |
| SOAP defines its own security | Inherits security measures from underlined transport Protocol (ouath-1.0 outh-2.0 , Bearer token) |
| Less preferred than REST | REST more preferred than SOAP |

Advantages of Web Service

**1. Web Services Interoperability (WS-I)**

* Web Services are "Application, Platform and Technology Independent"
* Ex: Uber / OLA and Google Maps shares the data among each other

**2. Loosely Coupled**

* Each application is independent of one another. Hence changes done to one application will not impact the "unrelated areas"

**3. No need of re-inventing the wheel**

* Web Services reduces the software development time
* This helps the other business partners to quickly develop application and start doing business
* This helps business to save time and money by cutting development time
* Ex: Uber / OLA can make use of Google Maps

**4. Business Opportunity**

* Web Services will open the door for new business opportunities by making it easy to connect with partners
* Ex: Dominos can get the order from Food Panda / Swiggy along with getting orders from its own site

**5. Service Reuse**

* Web Services takes code reuse a step further
* Ex: An organization can have a "Single Payment Gateway service" which helps other webapplications of that organization to interact