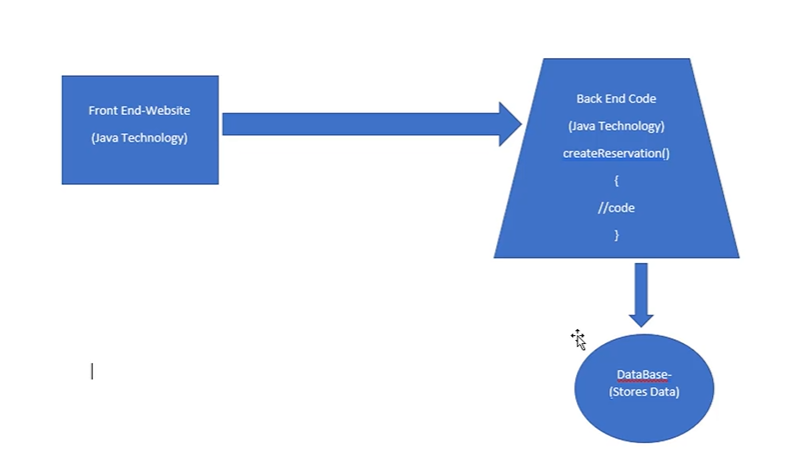
RestAssured is written in java, it is a java library, For selenium Automation we use selenium library with java similar Fashion for for doing API automation now we are using RestAssured library with java

First what is API?

Application program interface

Below code was sitting in one repository, lets suppose say Mariotte hotel application



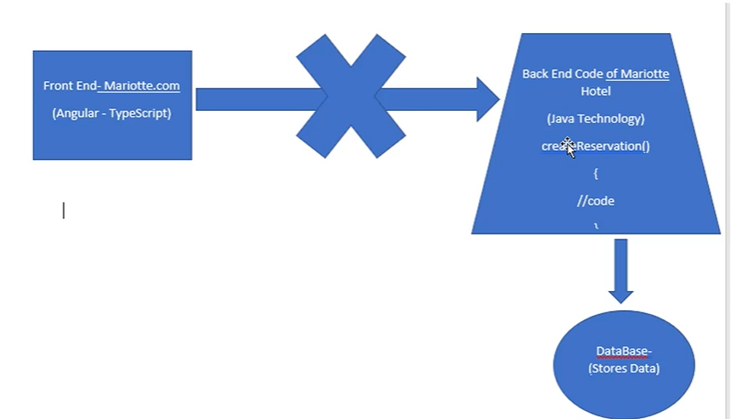
But this Mariotte hotel details are available at hotels.com also, then how this Hotels.com is able to get the info about Mariotte? Hotels.com has a separate code repository and Mariotte has its own repository. Then how they are communicating? Are they directly accessing or interchanging data? If they access, there will be a security breach.

A diagram of a diagram

Description automatically generated

Second Issue: front end is written in type script and back end was written using java, then how the communication will take place between front end and back end? Now this will become design problem.

Angular Type script was very popular for front end and java is very popular for backend.



To solve these problems API came into picture, This API is independent of all languages. In API we use Http protocol to receive/send data, and this protocol is independent of languages.

For sending/Receiving the data there should be one format, the format we use here is JSON/XML, which are independent of all languages, does not belongs/specific to any specific language. All languages uses JSON/XML format. JSON/XML is representation of data.

To interact with any application or to perform any action on website, first we need to check with developers like are there any API’s to do so? This API will process the request from frontend and respond to the requested data. These API’s are exposed by the backend application to interact with them, Now the front end application/outside application’s will interact with API’s provided and not directly interacting with back end applications.

A diagram of a software process

Description automatically generated

Now lets see what is API?



Most commonly used API’s are rest API, Rest poses some Rules and regulations for creating the API, so similarly we have SOAP API, we also call as SOAP webservices, SOAP services have different standards. Most of the legacy services are relied on SOAP, Rest is the latest one in the market. SOAP consists of so many conditions and authorizations and authentications to communicate, where rest is very simple in terms of communication, Rest is a light weight protocol now most of the applications are developed/moved to rest. Rest is easy to build and maintain.

Now we will test this Integration part, i.e RestAPI testing using Rest assured.

Rest Architecture has specific reules, to Interact with API you should use EndPoint/BaseURI. To perform some action you should always use methods like(get,put,patch,post,delete), generally we call them as CRUD operations(create, retrieve , update, Delete).

EndPoint/Base URI is place where API is available, if you want to communicate with API first you should know the address where it is hosted.



A screenshot of a computer program

Description automatically generated

A black text on a white background

Description automatically generated

PUT is used to update resources.

Once after reaching Google API’s place with help of EndPoint URL or server URL, we should know which resource or API we want to access, Example after reaching Google server then we should know which resource we want to access like (maps,search,images), here maps, search, images are API names or resource names.

These API’s sometimes expect some parameters to be passed as part of our request.

There are 2 types of parameters.

1. Path Parameters
2. Query Parameters

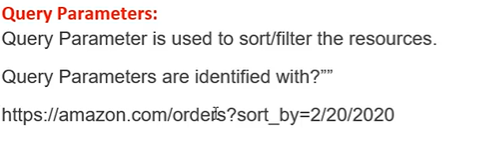
To point to specific resources within collection then we use Path Parameters.

In the below example Images is resource and 1123343 is sub resource.

A screenshot of a computer error

Description automatically generated

Query parameters does not have any sub resources, here we do filtering of existing resource only. For query parameters we should use ? before writing parameter name and it has parameter value.



<https://www.google.com/search?q=hyderabd&rlz=1C1GCEU_enIN1015IN1015&oq=hyderabd>

If we have more than 2 query parameters then we use & to join them.

EndPoint URL= BaseURL/resource/(Query/Path)Parameters

A close up of a sign

Description automatically generated

A black text on a white background

Description automatically generated

Header represent the type of data which we are sending like JSON/XML/Text. Cookie is additional details to API like Authentication(username/pass)/Authorization or we can send login cookie.

A screenshot of a computer

Description automatically generated

The below are the three libraries which are required for testing restAssured, hamcrest is a library used for input and output validations

A screenshot of a computer

Description automatically generated

We have 3 methods Given() When() Then(), these methods are principles of API

In given() method we provide all inputs to API

when method is used to Submit the details of API

Then method is used to validate the response code of API

Given(), when(), then() are static methods from static package, we need to import the restAssured static package manually

PUT

A screenshot of a computer program

Description automatically generated

GET

A screenshot of a computer code

Description automatically generated

Pacth :partial update

A computer screen shot of a program

Description automatically generated

Delete

A screen shot of a computer code

Description automatically generated

JsonPath is a class which takes string as input and converts into JSON, for parsing JSON

Or

you can Directly Store in Response response object and use response.path(JsonPath); to get the value from Json output as String

Here in the given above examples query parameter is the default way to Authenticate API’s provided.

For Get http method no need to provide body but for remaining methods we need to send boday, but we need to provide query parameter for getting resources or directly provide path parameter