Phase 4

Testing and deploying the application.

Phase 4

Testing tools.

Jasmine and Karma

Mocha and Chai

Superset testing tool

JEST

Graph QL : Self learning

Docker

Jenkin

Cloud computing

AWS Cloud : EC2, S3 EBS Volume Etc

**Day 1**

Input

Process

Output

MVC

App.js Router, controller, model database

If we want to test controller functionality

Controller must be call from router and router must be call from App.js. We can test the application using some testing tool which verify the application running properly or not.

Testing is use to find the error or defect or bugs in the application or software.

function chckUser(name,pass) {

if(name!=”Raj” && pass !=”123”) {

return “success”;

}else {

return “success”;

}

}

Testing: Testing mainly divided into two types

Black box testing

Input ---🡪 Process -----🡪 Output

A =10 display value

B=20 B is 20

Actual

Expect the output : 30

White box testing

Input---------🡪Process-------------🡪Output

A=10

B=20 coding output put 30

Actual

Expected

Both must be match

Manual testing : we run the testing and manually write the result in testing template.

Automation testing : tool generate the reports. Less interaction from programmer or tester for the loop.

Unit testing: unit testing is testing which help to do the testing for smallest piece for code that can logically verified. Unit also known as function/ method/ class/ modules etc.

Unit is testing is a type of white box testing.

Unit testing is a kind of software testing method is which each individual and independent part of the source code is tested to determine that it is good enough for the user.

When we perform unit testing on the front end (client side) of the software. It is called or known as front end unit testing.

Jasmine : Jasmine is a open source testing framework which help to do the unit testing on front end side technologies.

Using jasmine we can do Client side as well as server side(node JS) testing.

Jasmine provide pre-defined functions which help to do the testing.

1. **Test suite:** Test suite is technically contains more than one test case as well as another test suite. To write test suite testing framework provide pre-pre-definedfunction ie describe.

**Syntax**

describe(“suiteName”,callbackfucntion)

describe(“message”,()=> {

})

1. **Test spec :** Test spec is known as test case which actually help to do the testing. Test spec contains more than one expect which help to check actual and expected output.

Syntax

it(“spectName”,callback)

it(“addition”,()=> {

})

1. **expect :** this construct aids in testing if the expectation from the software are fulfilled or not.

**Testing hook or life cycle functions.**

**beforeEach() : it call again and again before every it functions.**

**afterEach() : it call again and again after every it function.**

**beforeAll() : it call only once before all it functions.**

**afterAll() : it call only once after all it function.**