1.Which testing tool is good for your project and why (write full

description and give some example)?

Karma is the best testing tool for my angular project

Because:

- It gives productive testing environment to developers.

-The environment being one where they don't have to set up loads of configurations, but rather a place where developers can just write the code and get instant feedback from their tests. Because getting quick feedback is what makes you productive and creative.

-We can automate tests in different browsers and devices

-We can watch files without manual restart

-Online documentation is good

-We can easily integrate with a continuous integration server

-Easy integration with reporting

Examples:

1.

it(`should have as title 'AngularApp'`, () => {

const fixture = TestBed.createComponent(AppComponent);

const app = fixture.componentInstance;

expect(app.title).toEqual('AngularApp');

});

2.

it('should create the app', () => {

const fixture = TestBed.createComponent(AppComponent);

const app = fixture.componentInstance;

expect(app).toBeTruthy();

});

3:

it('should create', () => {

expect(component).toBeTruthy();

});

2. Write five sample unit test cases in Karma or Jest.

1:

describe('Company.Model', () => {

it('should create an instance', () => {

expect(new Company.Model()).toBeTruthy();

});

});

2:

it(`should have as title ‘UnitTestExample’`, () =>{

const fixture = TestBed.createComponent(AppComponent);

const app = fixture.componentInstance;

expect(app.title).toEqual(‘UnitTestExample’);

});

3:

it(‘should change title to Unit Test App’, async(() => {

const fixture = TestBed.createComponent(AppComponent);

fixture.nativeElement.querySelector(‘button’).click();

fixture.detectChanges();

expect(fixture.nativeElement.querySelector(‘h1’)

.textContent).toEqual('Unit Test App');

}));

4:

it('should have a defined component', () => {

expect(component).toBeDefined();

});

5:

it ('should check incremented value is greater than zero', ()=> {

let counterComponent: CounterComponent = new CounterComponent();const curCounterValue = counterComponent.increaseCounter();

expect(curCounterValue).toBeGreaterThan(0);

});