1. אמור שלוש דרכים איך לתקשר בין Modules של האפלקציה שלך בעזרת הפונקציונליות של אנגולר:

\*דרך services

\*דרך events

\*דרך חיבור models ל rootScope

\*ישירות בין controllers בעזרת $parent , $$ChildHead, $$nextSibling וכו'

\*ישירות בין קונטרולירים using controllerAs או דרכים אחרות של ירושה

1. מה מהדרכים הוא testable easliy

\*שמוש ב services זו הדרך הכי נקיה וטובה אפשר לעשות להם טסטים בקלות ובטסט אפשר להשתמש בסרביס אמיתי או בMOCK

\*Events can be tested. In unit testing controllers, they usually are instantiated. For testing events on $rootScope, it must be injected into the test.

Testing $rootScope against the existence of some arbitrary models is testable, but sharing data through $rootScope is not considered a good practice.

For testing direct communication between controllers, the expected results should probably be mocked. Otherwise, controllers would need to be manually instantiated to have the right context.

3.מהם הצעדים הבסיסיים כדי לעשות unit test לangularJS filter

\*הזרקת המודול module שמכיל את הfilter

\*סיפוק הmocks שהfilter תלוי בהם

\*יצירת instance של הfilter באמצעות : $filter('yourFilterName').

\*ביצוע assert לפי צפי התוצאה

4. מה ההבדל בין dercit

5.

The most popular e2e testing tool for AngularJS is

Protractor. There are also others which rely on similar mechanisms. Describe how e2e testing of AngularJS applications work.

The e2e tests are executed against a running app, that is a fully initialized system. They most often spawn a browser instance and involve the actual input of commands through the user interface. The written code is evaluated by an automation program, such as a Selenium server (webdriver). That program sends commands to a browser instance, then evaluates the visible results and reports back to the user.

The assertions are handled by another library, for Protractor the default is Jasmine. Before Protractor, there was a module called Angular Scenarios, which usually was executed through Karma, and is now deprecated. Should you want to e2e test hybrid apps, you could use another Selenium server, called Appium.

Testing can be handled manually, or it can be delegated to continuous integration servers, either custom or ones provided by Travis, SauceLabs, and Codeship.

6. לפניך unit test שנכתב לprotractor

it('should find an element by text input model', function() {

browser.get('/some-url');

var login = element(by.model('username'));

login.clear();

login.sendKeys('Jane Doe');

var name = element(by.binding('username'));

expect(name.getText()).toEqual('Jane Doe');

// Point A

});

הסבר האם האם הקוד סינכרוני או אסנכרוני ואיך הוא עובד:

הקוד הוא אסינכרוני למרות שהוא נכתב בצורה סינכורנית מה שקורה מאחורי הקלעים כל הפונקציות האלה מחזירות promises on the control flow יש אפילו דרך ישירה לגשת אליהם באמצעות “protractor.promise.controlFlow()” ושני המטודות של האובייקט המוחזר הן “.execute()” ו “.await()”.

Other webdriver libraries, such as wd https://github.com/admc/wd, require the direct use of callbacks or promise chains

7. כשSCOPE נגמר ישנם שני EVENTS של "Destory" שמורמים , למה הם משמשים ולמה יש שניים:

הראשון הוא של אנגולר angularJS שמורם והשני הוא של jqLite/jQuery.

בראשון משתמשים הסקופים של אנגולר כשהם accesible לדוג' בcontrollers או link functions

Consider the two below happening in a directive’s postLink function. The AngularJS event:

scope.$on(‘$destroy’, function () {

// handle the destroy, i.e. clean up.

});

And

element.on(‘$destroy’, function () {

// respectful jQuery plugins already have this handler.

// angular.element(document.body).off(‘someCustomEvent’);

})

הdestroy event של jqlite\jQuery מורם כל פעם שNODE מוסר, וזה יכול לקרות בלי מצב שSCOPE נגמר\נהרס.

8.כיצד מנקים\מבטלים $timeout וכיצד מבטלים $watch:

המפתח לשניהם זה לשמור את הערך המוחזר למשתנה

כדי לנקות timeout פשוט לעשות לו cancel

var customTimeout = $timeout(function () {

// arbitrary code

}, 55);

$timeout.cancel(customTimeout);

אותו דבר לגבי $interval()

כדי לבטל $watch()

// .$watch() returns a deregistration function that we store to a variable

var deregisterWatchFn = $rootScope.$watch(‘someGloballyAvailableProperty’, function (newVal) {

if (newVal) {

// we invoke that deregistration function, to disable the watch

deregisterWatchFn();

...

}

});

9.מזה $rootScope ואיך הוא קשור ל$scope

rootSCope הוא האבא של כל ה$scope שקיימים בweb page