Testing

General part

- Describe testing in general and the different kind of testing use in software development.
- Explain test principles like mocks, stubs, suites, etc.
- Describe the difference between test first and test last and their pros and cons.

Practical part

Testing is an important part of software development. Angular is created with testing as a first class citizens, i.e. the structure of angular is built with testability in mind, e.g. the use of dependency injection. On the client side it is important to be able to test not only the validity of the logic within the program but also the scalability to the large set of different browsers.

The problem to solve here is to provide tests of a small web application that maintain a list of persons. Start project can be found in Q4_Testing.zip.

Backend testing

1) Write mocha/should tests that verify the restful API. All CRUD methods must be tested.

Frontend testing - The frontend is made with angular. Write the following tests using karma/jasmine:

2) Test that the `AddPersonController` sets the title to "add Person".

3) Test that the `editPerson` function in the `PersonsController` make a call to the `$location.path` with "/edit-person/5" as argument if called with `id = 5`;

4) Test that when you set `location.path("/")` you actually go to this location

5) Test that, when you set `location.path("/UnknownPath")` you go to the `location.path("/")`

6) Test that the `PersonsController` sets the $scope.persons property to an array of persons fetched via the call to `/api/persons`.

7) Test that the `EditPersonController` fetches a person when called with the parameter 5.

Note: this part is NOT a part of the exercise, it's meant as FYI.

Hints for the Angular Part:

Use http://slides-spring2015.azurewebsites.net/testingKarmaJasmine/test.html#24 as background for 3-5

Use the $httpBackend to fake responses for 6+7

Test:

You have been introduced to tests several times during your three semesters with us.

As pre-preparation for test questions, you could find some of your old test examples. You will not have time to do this during your 80 minutes preparation, so do this during your general preparation before the exam.

You should have examples from second semester, including your second semester project, but also first semester introduced unit testing.

This semester, we have worked a lot with testing so do not forget to revisit these examples too.

General advices:

Spend you 80 minutes well.

In this question you get some code to start with. Get the code up and running. You don't need to run `npm install` in this example so it ready to roll.

Read the question carefully identify the parts you know how to get started with and begin there. If you get stock continue to the next part. You can always return later if you have the time.

Links and references:

- http://en.wikipedia.org/wiki/Software_testing
- http://en.wikipedia.org/wiki/Unit_testing
- https://docs.angularjs.org/guide/unit-testing
- http://mochajs.org/
- http://en.wikipedia.org/wiki/JUnit
- http://en.wikipedia.org/wiki/Immediately-invoked_function_expression