# Practical Exam

## Overview

The exam is designed to test your ability to cope with new technologies and ability to learn and integrate with external frameworks. There is no algorithmic complexity or computation challenge. It is completely trivial in this respect.

Please note however, that besides the correctness of the implementation and the external result, the elegancy of the code, usage of the correct language idioms and patterns and the coding conventions and practices applied will also play a major factor when evaluating the solution. It should behave well both externally and internally.

## Development Environment

Your implementation should be based on Google’s angularjs JavaScript library which is designed for creating single page applications for the web and for touch. The library is open source and can be freely downloaded from <https://angularjs.org/>, where you can also find lots of documentation, tutorials and examples.

You can use the IDE of your liking to develop this part of the solution. If you have no JavaScript IDE you like, we recommend WebStrom by JetBrains, which is available for a 30 days trial at <http://www.jetbrains.com>.

To execute and debug your application we recommend using Google Chrome that has a ‘Develop’ menu with an error console and a javascript debugger. You can also ensure that your app displays well on devices, using the emulation feature in this development tool.

You should create the sample application, which is a mobile single page app, using a single html file that loads one or more javascript, templates and css files, including the angular ones.

Once finished with the JavaScript application, please deploy the application as Hybrid application to a smart phone (iOS or Android) using either Apache Cordova (previously called PhoneGap). This step should be easy enough and requires only ’hello world tutorial’ level knowledge for using cordova command line tool.

## Sample Application Description

The sample application is called myLocations and it allows the user to maintain a list of categorized name locations.

The domain model contains two main entities, a Category and a Location. A Category has a single property: Name. A Location has the following properties: Name, Address, Coordinates, and Category.

All data is saved to the locale storage of the browser (an HTML5 feature) for simplicity.

The application should use the new angular ui-router and ngTouch module.

## Use Cases

The user can manage (view, add, remove and edit) the list of Categories.

The user can manage (view, add, remove and edit) the list of Locations.

The user must fill all properties when saving an item.

The user must choose a category from a list of existing categories when defining a Location.

Each screen has a top toolbar with title and action buttons. The user executes an operation on a list item by clicking the appropriate button in the top toolbar.

The application screen has a bottom bar with two iconic buttons: Categories and Locations. The user moves between Categories and Location management by clicking on their respective icons on the bottom button bar.

The user can view all Locations sorted by alphabetical order, grouped or ungrouped by category.

The user can view only the locations assigned a specific category she chooses.

When clicking a location in the list, the user can choose to see the definition of the item or view it in on an actual map (using google maps or similar service, native or not).

When the user clicks on a location, the device will vibrate (via native bridge support).

The application exposes a share button which allows sending an email with the list of locations and categories in json format as an attachment (via native bridge support).

Bonus: allow selecting the coordinates from the map service and not entering by hand.

Bonus: allow relating multiple categories to a single item, define and enhance the use cases and ui related to this.

Good Luck