

Proposal for Assignment for HTML5 Mobile web app course

Aalto University

2012-03-19

Team 10

[Roland Prinz](#), [Antony Meyn](#) and [Aleksi Rossi](#)

Mobile application

The mobile application to be developed as the assignment is front-end for the Open Ministry (<http://avoiministerio.fi>). The main functionality is similar to the Financial Times app: work as a caching standalone app to display content of the site, and allow basic interaction.

The main functionality is:

- Home page, perhaps split differently in mobile context:
 - Sign-in for existing users
 - through email + password, or
 - through Facebook
 - Sign-up for new users
 - email + password, or
 - through Facebook
 - Welcoming message
 - List of current or “hot” Bills (ie. Proposals of new law to be supported)
 - List of current or hot Law Drafts, being debated
 - List of current or hot ideas
 - List of late articles (blog posts) or Expert statements
 - Later on the home page can be a personal stream of events of interest, like Facebook “home page”. This functionality has not yet been implemented in Avoiministeriö, but might get ready during the course
- List of ideas

- to be sorted by variety of criterias: newest / oldest, most/less viewed, most/less commented, most/less voted, most/least supported, most divided votes
- later on,
 - these might be highlighted by new ideas or recent comments by people this person follows
 - the list might be organised by tags (currently being developed)
 - the list might be organised by a person, or by ideas commented by a person
- An idea
 - main content
 - statements by experts
 - commentary
 - if the idea was added by you, editing the idea
 - adding a comment
 - editing a comment
 - voting pro or against (switching the vote)
 - later on, editing article tags, or proposing new tags
- Articles
 - viewing basically blog articles, textual content, maybe with embedded videos
- Later on
 - a user can follow ideas or persons
 - clicking a person opens a page by the person:
 - that is a page displaying ideas and comments made by the person
 - and a page displaying what the person is following
 - there will probably be other voting, ie. Thumbs up for a comment or for the person

For required mobile device technology we will:

- take a photograph of a person and update that to the website for a new profile pic (currently there's no Open Ministry functionality for that but will come during the spring)
- or associate GPS location for comments, ideas or votes

Changing profile pic is actually useful, and associating GPS locations is a curiosity but nevertheless very interesting.

Initially the application will be used over the net, but as time allows it will be packaged as a separate app, and perhaps even delivered over the app stores.

The application will be minimal but very useful, and hopefully will be used by real users during the course of the course.

Used technologies

Here's the outline of the initial stack.

Mobile Frontend:

- JQuery or Zepto as needed
- We will probably start with Ember (<http://emberjs.com/>), but if it doesn't work out we will fallback on Backbone (<http://documentcloud.github.com/backbone/>), maybe with Model bindings (<https://github.com/derickbailey/backbone.modelbinding>)
- for templating on Javascript, we will start with Handlebars with Ember, but might fall back on some other templating like Mustache
- Local storage, through the model framework, like Ember Data (<https://github.com/emberjs/data>)
- CSS generation perhaps for Less or SCSS through Compass on Rails app
- If there will be server side rendering, naturally appropriate Rails stack will be used, Haml etc.

Backend:

- Open Ministry provides the content but does not have an API at the moment. Shortly there will be either REST or JSON API for the needs of our development. If not we will add it during the project as needed. Open Ministry is free software under MIT license , and we can setup personal development servers on Heroku if needed.
 - Open ministry API will be serving JSON through Rails app, or it might be wrapped on Grape (<https://github.com/intridea/grape>) and throttled by Rack throttler (<https://github.com/datagraph/rack-throttle>)

The app will be packaged with Phone Gap. Or we might try new HTML5 mobile app packages, like AppGyver.

Contribution of the each individual

(mixing with people outside of the course is possible but each individual contribution must be stated upfront)

Whole team will participate in developing the mobile application, so there will be commits by everyone.

We might need help from Open Ministry developer team for setting up the backend API, or doing changes for it. Aleksis is part of OM core team, so the communication with them is easy to arrange. OM team will not take part of the assignment mobile app side, but of course occasional pull requests from them will not be despised. There will most probably be help for setting up the API from other from OM developers.

Roland and Antony would be able to contribute for the HTML5/Javascript design on making this web application feel like a native mobile app within the browser, by using the local storage API, and designing the application to be functional [offline](#).