Open source mobile development at DBC

Status for the current work here: in March I (Rasmus Erik Voel Jensen) have been allocated to develop mobile app prototype for "Spørgetjenesten" - a danish ask anything service. It will most likely be licensed under GPL or AGPL due to company policy. Development will happen in our internal svn repositories, but it will also be published via version control on github. Expect running prototypes in end of March.

Notes on frameworks

The app should work on different devices, which makes phonegap the obvious choice, for creating the app bundle.

Apparently the two major open source mobile javascript frameworks are jqm(JQuery Mobile) and sencha(Sencha Touch). Jqm is targeting a broader range of platforms, and more traditionally html-application-like, whereas sencha is more app-oriented and more javascript than html. Jqm is still in alpha, but seems to have a larger community, stronger backing and will probably be strongest in the long term, where sencha is slightly more mature at the moment. Beside mobile frameworks, a third approach is just to write the app directly on the browser or use smaller simpler libraries such as xui and not rely on the larger frameworks.

Approach and abstraction layer

The approach here will probably be to make a simple abstraction layer between the ui and the actual application logic. And this layer will then probably built upon jqm or plain browser for presentation.

Using this abstraction layer, and keeping the application logic in a subset of JavaScript, it may also be possible to port the apps to run on lower-end devices with a custom JavaScript-like implementation on J2ME. The abstraction layer should also make it possible to transform the app into a mobile web site.

The ui abstraction layer works by showing a series of pages, - similar in programming model to traditional CGI scripts, but capable of running directly in the client using javascript. And instead of html, we use a mobile gui description markup language, where each page is an xml document encoded as jsonml, and which is then transformed into mobile html with different capabilities depending on the device.

Platforms

Apps and websites build with HTML5 + JavaScript. When targeting multiple smartphones (Android, iPhone, Symbian, Blackberry, etc.), html5+javascript is the way to make crossplatform applications, possibly wrapped as an App in PhoneGAP to gain access to lower level apis. Application logic in JavaScript will be able to run directly on the device.

Mobile web - WAP2 / XHTML Mobile Profile. To target the broadest variety of platforms, XHTML Mobile profile, with gracefull degradation, is the option, as it works both on featurephones and smartphones. The downside is limitations on the user interface and capabilities, and poor interactivity. Application logic in JavaScript would need to run on the server, which generates the pages.

Apps with Java Micro Edition. Most low-end phones supports apps written in Java Micro Edition. There are no central distribution network (can instead be pushed via sms, and there are distribution channels via mobile operators and independent like getjars). If application logic is kept in a subset of JavaScript, it is still possible to run directly on the device, using one of the downshrunk mobile JavaScript interpreters such as lightscript or minijoe.

Native applications, - Objective-C, Android-java, etc. - gives better access to the access to the device, but requires separate development for each platform.

At first the focus will be on an abstraction layer that works both with HTML5 and WAP2, and then in the long run also add support for jme-apps.