# Mobile Software

In this exercise, you will familiarize yourself with mobile software development on a novice level. The main focus is on networking and service integration.

### A Note About Evaluation

We will continue to classify the submitted exercises into barely working ones, proof-of-concept demonstrations and well done solutions. In a perfect world, for example, there would be little need for parallel computing, because every computation would complete in an infinitesimally short time. However, in the real world we need to deal with several constraints, e.g. required level of user experience and network latency. Handle especially the situation where network connectivity is intermittent. If you manage to deal with these issues, you will get the full normal points.

### Deliverables

Please add a new directory called *assignment_3* in the SVN repository. Then put the following items into *assignment_3*. Please also follow the general guidelines.

- README
- documentation
- source code

Please also provide a working server-side code deployment for demonstration purposes.

### Suggested Workload and Grading

An average student is estimated to spend 25 hours for this assignment (in pairs). In addition to the points mentioned with each exercise, **the course staff may give at most 6 extra points** about an extraordinarily good application that has a great idea, spectacular presentation and that uses the techniques in a creative way. The lower limits for each grade are as follows:

1: 8
2: 11
3: 14
4: 17
5: 20

### Setting Up The Environment

We accept the following platforms (preference order of the course staff):

- Android 2.3 (minSdkVersion <= 9)
- iOS 4 (Xcode 4)
- Windows Phone 7.5 (SDK version 7.1)

You can do the work in emulator, and if you want to try on real hardware you can borrow a test phone. Notice that close to deadline there is probably going to be high demand for the phones.

In order to use computers of Aalto IT Services, you will probably be better of by choosing either Android or Windows Phone, since at least their IDEs can be used easily on the machines. We can provide only little support for the platforms themselves, but will help if some problems occur.

## 1. Learning The Basics (6 pts)

In this exercise, you create one application with several proof-of-concept features. The user interface for each sub-exercise should appear in their own tabs. If you insist, you are allowed to use other view-based mechanisms instead of tabs.

### Hello World (1 pt)

Once you have decided what platform to use and set up the environment, please complete the corresponding tutorial for a Hello World project. Complete the project. The tutorial probably told you to just add a simple text element to the center of the screen. For the later exercises, extend the program so that there is a screen-sized tabular container or other means of navigation for the content.

- Android: **http://developer.android.com/guide/tutorials/hello-world.html**
- Xcode 4: **http://www.techotopia.com/index.php/Creating*a*Simple*iOS*4*iPhone*App(*Xcode*4)**
- WP7: **http://create.msdn.com/en-US/education/quickstarts/**

Document your learning experience and justify your decisions.

### Using The Network (2 pt)

Extend your Hello World to use network so that it retrieves the following document and shows it contents and date of modification to the user.

https://playground.cs.hut.fi/t-110.5140_2012/hello.txt

### Using UI Elements (1 pt)

In the third tab, allow the user to enter an URL. Then retrieve a picture from that URL and show it. One example URL is provided below. It is enough that your program supports JPEG.

https://playground.cs.hut.fi/t-110.5140_2012/hello.jpg

### Parsing Simple Data (2 pt)

Yahoo Weather API provides a way to get the weather observations and forecasts from all around the world in XML. You may do an opportunistic lookup for the weather through an URL. The location attribute is specified through a parameter called WOEID (Where on Earth Identifier). In another tab of your program, let the user enter the WOEID parameter of the location for current observations lookup and show the results of the query. You are not required to care about the peculiarities of place matching. The base URL for lookup is:

http://weather.yahooapis.com/forecastrss?w=location

The WOEID for Helsinki is 565346. If you are interested in other locations, you can search for the location of your choice at http://weather.yahoo.com/ and the resulting URL provides you with WOEID for the location. For example, searching Helsinki would result in the following URL.

http://weather.yahoo.com/finland/uusimaa/helsinki-565346/ where 565346 is the WOEID for Helsinki.

Please note that you are not allowed to distribute your program to masses due to terms of service. Yahoo Weather service was just picked for the sake of easiness and no need for registration. You may use any other weather API, provided that the course staff will not need to register with the service to evaluate your program. While we may also let you use any other API that is similar in its technical features, please ask first.

Here are the full instructions and terms of service of Yahoo Weather API:
http://developer.yahoo.com/weather/

## 2. Own Application (12pts, Mandatory)

In this exercise, you will implement a mobile interface either to your social media application (assignment 2) or to another application that you are asked to implement. Therefore you have three alternatives. Whichever your choice will be, we will evaluate your submission roughly according to the following criteria:

- REST API (or similar) on the server side and its use on the mobile side: ~3pts
- Using phone's context or data in other phone applications (e.g. location, calendar data) as supplemental sources of input: ~3pts
- Adapts the content to the target device's interface: ~3pts
- Provides means for accessing server-side data that requires authentication: ~3pts

Documentation is required "as a side order". In other words, you may get some point reductions if the evaluator does not understand the contents and motivation of your solution.

### Alternative 1: Mobile Interface for Your Social Media Application

Implement a mobile interface to the application that you implemented for the Social Media assignment. Consider *mobile context* as the foundation in designing the feature set that your program covers.

### Alternative 2: Helsinki Region Transport APIs

HRT (http://www.hsl.fi/en/) provides a number of transportation information services to the population in Helsinki area. For instance, Journey Planner (http://www.reittiopas.fi/en/) makes commuting easy and fast. In addition to the mainstream services, many of the services have great public APIs for third-party software developers.

Implement a mobile application that uses at least one of these APIs. You need to implement any needed server-side parts as well. As an example, your program could help the user to find a route to the location where the user is willing to go. As helping features, it could add a reminder to the calendar when the user should leave. Also the current location may be available through APIs.

Note: In this alternative, you might need to provide a small amount of your personal information to HRT. If you disagree but would still like to select this alternative, please inform the course staff.

**Services:**

Journey Planner: **http://www.reittiopas.fi/en/**
Omat lähdöt: **http://www.omatlahdot.fi/omatlahdot/web?command=lang&lang=3**
Traffic Exceptions: **http://www.poikkeusinfo.fi/**

**APIs:**

Reittiopas: **http://developer.reittiopas.fi/pages/en/home.php**
Omat lähdöt: **http://dl.dropbox.com/u/20567085/Omat*Lahdot*v101.pdf** (only in Finnish)
Traffic Exceptions: **http://dl.dropbox.com/u/20567085/Poikkeusinfo*XML*rajapinta*V*22_01.pdf** (only in Finnish)

### Alternative 3: Your own idea?

In addition to the previously mentioned topics, you might want to try out many other things. If you are up to something special, we would be glad to hear from you! Before you start implementing your idea, please write a short non-technical description of your idea (max. 100 words), your first thought of technical implementation (no more than half a page) and send it to us by email for approval. Your description does not affect on the grade, but we reserve the right to reject the idea. We will answer to to you within 2 business days, after which you can start working on the topic.

## 3. Additional Exercises (Max. 6pts, Optional)

The following exercises are completely optional. We evaluate at most two bonus features per student group.

### Twitter Client (2pts)

Implement a simple Twitter client that shows the timeline for the search word given by the user. A simple hello world -alike proof-of-concept is expected. Hence no login features required.

### Image Browser (2pts)

Implement an image browser, which retrieves images on the Internet according to the XML/JSON file that contains the list of the images. The user should be able to navigate to next/previous image by simply clicking a button. Some potential wow effects: zooming, animated transitions.

You may complete this exercise by extending your existing Hello World suite.

https://playground.cs.hut.fi/t-110.5140_2012/images/[?format=json|xml]

### Advertisements (4pts)

There are plenty of choices for a software company that wants to monetize through mobile advertisements. Present and compare some of the services from commercial and technological aspects.

Experiment the concept with one of the services you presented by providing a simple proof-of-concept program.

To give you a starting point, you may want to peek at the following services:

- iAd (iOS)
- AdWhirl, AdMob (iOS, Android, WP7)
- Smaato (iOS, Android, WP7, Maemo)
- Microsoft Advertising (WP7)

## Information Visualizer (3pts)

Find a data source that could be used for visualization purposes. Write an application that retrieves a portion of the available data and then shows it to the user in an intuitive way. Some of the processing can be done on a server-side program, if justified in the documentation.

One suggestion for the data source: **http://playground.cs.hut.fi/t-110.5140_2012/visualizer/**

## Your own idea? (2-6pts)

As in the third alternative of the “Own Program” exercise, you may propose your own idea to us. Please follow the same rules.