Android project proposal

Luca Bedogni Marco Di Felice ({lbedogni,difelice}@cs.unibo.it)

April 12, 2012

Introduction

In this document, we will describe two possible projects for taking the exam of the "Laboratorio di applicazioni mobili" course. One can select a project from this set, or suggest something else based on personal interests or ideas. In that case, please send firstly an e-mail to Dott. Luca Bedogni (lbedogni@cs.unibo.it) and/or Dott. Marco Di Felice (difelice@cs.unibo.it), describing what you want to achieve in that project, to see if it can meet the class requirements.

The following project descriptions should be used as a track for your final work, leading you to some key points that has to be done, but of course you can (and you are encouraged to do so) expand and add new features to your application.

1 RSS client

The first project proposal is about an RSS[1] client for the Android mobile platform. That client should be able to:

- Add new RSS feeds based on user inputs.
- React on Intents launched by external applications.
- Display the RSS feeds.
- Send a post link to social networks
- Configure some parameters

In the following, each feature is further discussed and analyzed in detail.

1.1 Add new RSS feeds based on user inputs

The application should be able to add new RSS feeds directly typed in by the user. More in detail, the user could enter a web URL in which he/she thinks resides an RSS feed. After typing it in the application, a service should check the validity of the URL (e.g. `url not found, url found but not an RSS, RSS found and so on`) and add it to the list of current RSS feeds.

1.2 React on Intents launched by external applications

The application should be able to add new RSS feeds based on Intents launched by other applications. That means that in *AndroidManifest.xml* the application should declare to be able to handle Intents with action `VIEW` and with data corresponding to an internet URL. Of course, every feed added this way should pass the validity check as described in the previous paragraph.

1.3 Display the RSS feeds

The application, in the main Activity, should display the list of all feeds currently registered in the device, providing the number of unread items for each feed too. Clicking on a feed will bring the user to the list of post for that selected feed. By long-clicking on a post, the user should be able to mark the post as unread, if he/she has already read it, or as read, if he/she hasn't already read it, or mark it as favourite.

The application should also provide a combined post list, where each post is displayed regardless of the feed from which it belongs to, while the action depending on the long click are the same as before.

1.4 Send a post link to social networks

The application could provide, via a long click listener or something similar, an action to post a link to the post in social networks such as twitter or facebook. The link should be trimmed with services like bit.ly, and some text could be added by the user if he wants to do so.

---

[1]http://www.rssboard.org/rss-specification

1.5   Configure some parameters

The application must provide a set of parameters to be configured, such as:

- Auto-update frequency

- Notification on new post

In particular, the application should use Android's `SharedPreferences` system, saving user preferences across session and reconfigure application's services based on those preferences.

2 Wireless Hot Spot finder

The second project proposal is about an application that should listen for wireless access points, and save this data in an internal database while displaying them on map. In particular, the application should:

- Find access points in range

- Plot found access points on a map

- Configure some parameters

In the following, each features is further discussed and analyzed in detail.

2.1 Find access points in range

The application should listen to the 802.11 channels to spot access points in range, and save them on a internal database along with:

- Location where the access point has been found

- Signal strength

- Security type (`WEP, WPA, Open` ecc.)

2.2 Plot found access points on a map

The application should be able to plot all found access points on a map, and identify them with different icons and colors. In particular, open access points should be displayed as `green`, secured access points should be displayed as `red`.

In addition to this, the application must color the map, plotting red roads where an open wireless network has been sensed, grey roads where no sensing has been done, red roads where sensing has been done and no open access points were found.

2.3 Configure some parameters

The application must provide a set of parameters to be configured, such as:

- Sensing frequency

- Notification on found access points

In particular, the application should use Android's `SharedPreferences` system, saving user preferences across session and reconfigure application's services based on those preferences.