Winter | 10

FREIE UNIVERSITÄT BOZEN
LIBERA UNIVERSITÀ DI BOLZANO
FREE UNIVERSITY OF BOZEN · BOLZANO

Fakultät für Informatik | Facoltà di Scienze e tecnologie informatiche | Faculty of Computer Science

Mobile Bus Organizer

Internet and Mobile Services Project Report

A Project by Thomas Steingruber (6152) and Jakob Obkircher (5778)

Index

Motivation

Moving from A to B is sometimes not so easy in a city. Using a car is often not so convenient, because there is never a parking space, and looking for a parking garage can take a lot of time. Using the public transportation service can be a good solution, but which bus –line should you choose? When and where does it start?
This is the point where our project comes in use. The Mobile Bus Organizer allows one to view the timetable of a bus line, according to the actual position or address and the destination and a few other practical functionalities.

Functionalities

General description

The Mobile Bus Organizer is a J2ME system, which offers the user to view timetables of the local bus line. In our project we use only bus lines of the SASA organisation, but generally it is usable for all services, having the required data stored in a database.
The System is divided in a client and a server. The client is running on a mobile device and sends the request to a web server, which processes these requests and sends the required data back to the device. Therefore, an Internet connection is required to use the system.
The client can be divided into two general parts:

The Gps Mode

This mode is available when the device of the user supports GPS and the user allows the application to use it. Receiving the coordinates from the GPS module, the user can see him on a map (Google Maps) and has the possibility to insert a destination address. Since we are sending this address to Google maps, to retrieve GPS coordinates, the user has to insert the destination address in the following way: Via Vittorio Veneto Bolzano, to get the correct destination coordinates. According to his position and the destination, the system sends a request to the server, which searches the nearest bus stop, selects a bus line and the according timetable and sends that information back to the mobile device, which displays this information to the user.

The manual Mode

To make this application also usable for people, who do not have a mobile phone with GPS, we created a manual mode. In this mode, the user has to insert the start address and also the destination address like described above and the server sends back the timetable of the selected bus line.

The settings

The application is able to store some settings permanently. These settings are:

- Use of GPS (yes/no)
- Use of Google maps (yes/no)
- Using a bus line as start page (yes/no)
- Preferred bus line (list with all possible bus lines)
- Number of shown start times of a bus line

At the first launch of the application, the user is requested to make these settings and he has to save them (he has no other option). If he sets the selected bus line as start page, then at the future launches, this page would be shown. Otherwise, the application shows the settings as first page.
To store the settings permanently, we used a RecordStore.

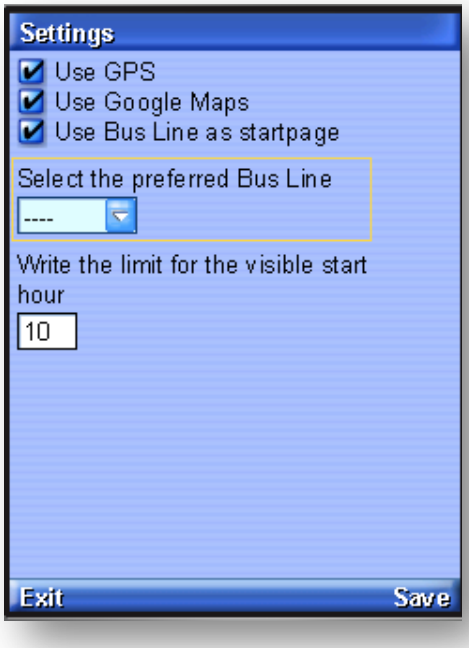
Retrieving the timetable of the selected bus line

This option shows the user the timetable of the bus line, which he has selected in the settings. Furthermore he also has to choose the direction of the bus by selecting the start station from a list box. These stations are stored in a local xml file, which are parsed with an xml parser to retrieve the appropriate information.

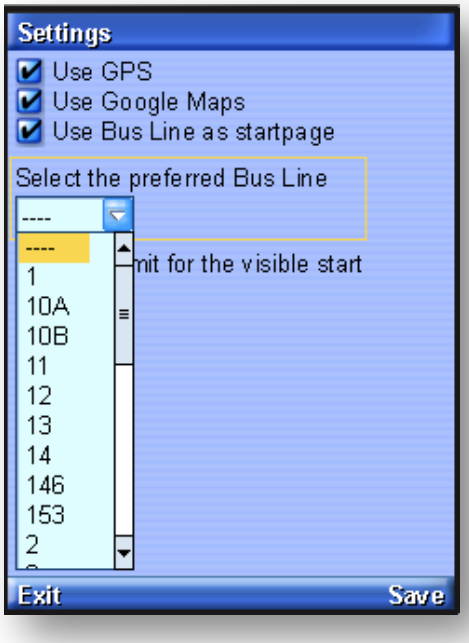
Selecting a bus line and retrieve the timetable

This function is similar to the one above with the only difference that the user has also to select a bus line form a list box (also stored in a xml file).

Human / Computer interaction



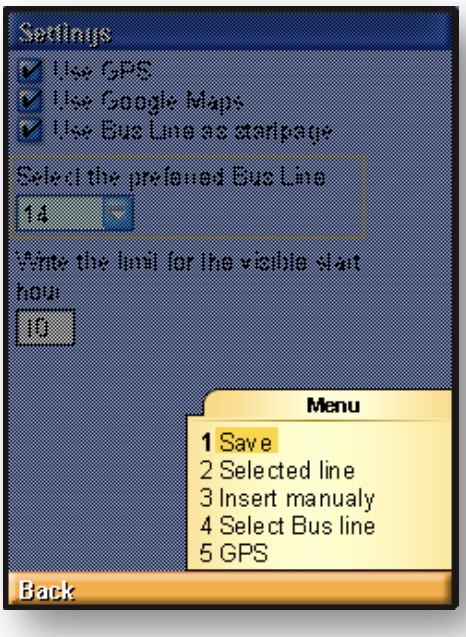
This is the first display shown to the user. Here he can make his settings by checking or un-checking the checkboxes. Clicking on save will save the settings in a RecordStore.



Clicking on the list box will open a drop down list with all possible bus lines available. This operation accesses a local xml file of the following format (see image on the next page)

```
<routes>
  <route>1</route>
  <route>10A</route>
  <route>10B</route>
  <route>11</route>
  <route>12</route>
  <route>13</route>
  <route>14</route>
  <route>146</route>
  <route>153</route>
  <route>2</route>
  <route>3</route>
  <route>4</route>
  <route>5</route>
  <route>6</route>
  <route>7A</route>
```

This is the xml file with all the possible bus lines. When this data is required an xml parser scans the file and reads out all the lines.

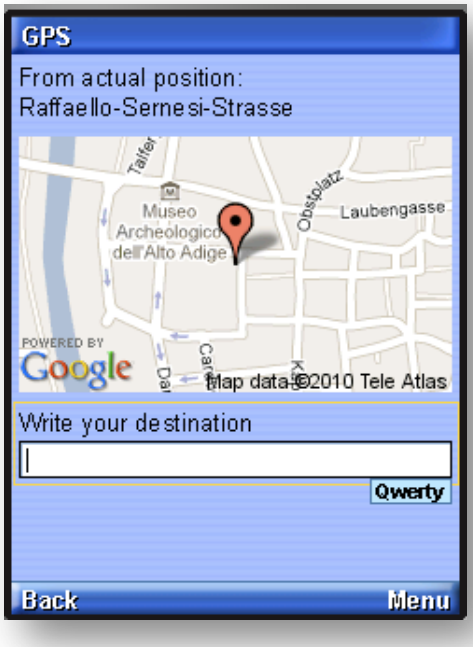


When the settings are saved, the user has the options as shown in the image above. Clicking now on GPS will open the GPS form.

Web service

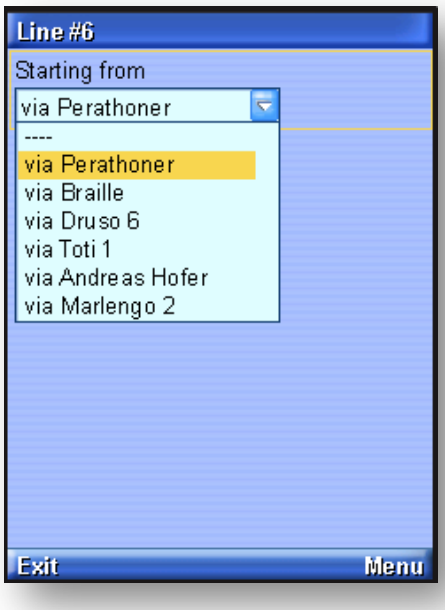
Sending longitude and latitude to the web service

Server sends the address and the Google map image back



This is the GPS form. The user can read the address of his position on the top of the form and see it on the map (red marker).

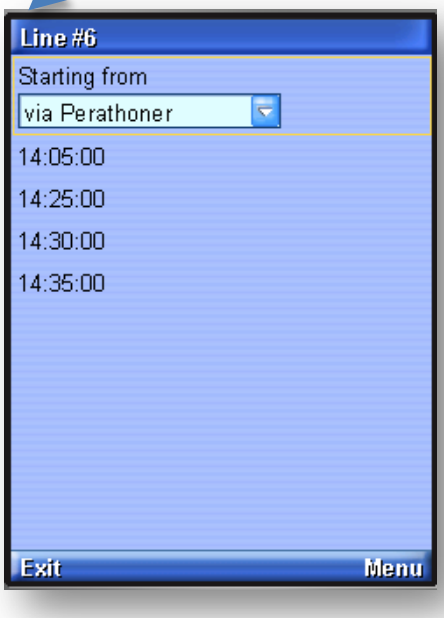
In the text field, the user can insert his destination and clicking on Menu- Get Info will request the server for the timetable and the server sends an xml file back.



Sends the bus line and the station name to the web service

Web service

Sends xml file with all times included

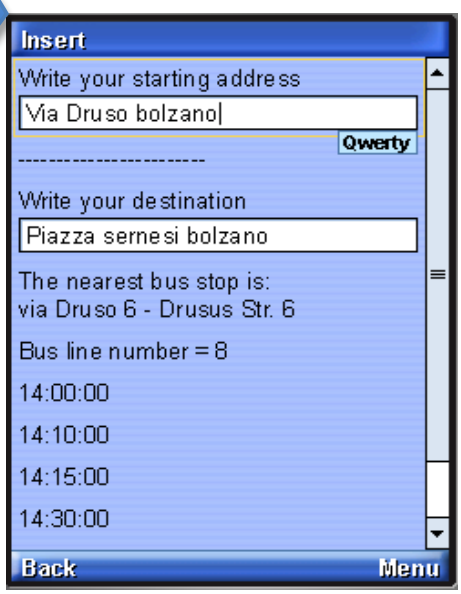


This is the form of the menu Selected Line, which shows the timetable of the bus line saved in the settings. The user has to choose the start station in order to indicate in which direction he wants to go; on the right image you can see the timetable.

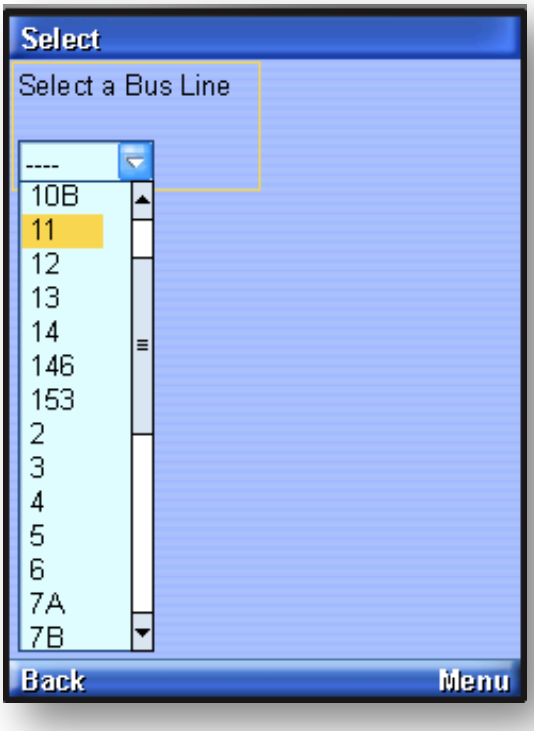
Web service

Sends both addresses to the web service

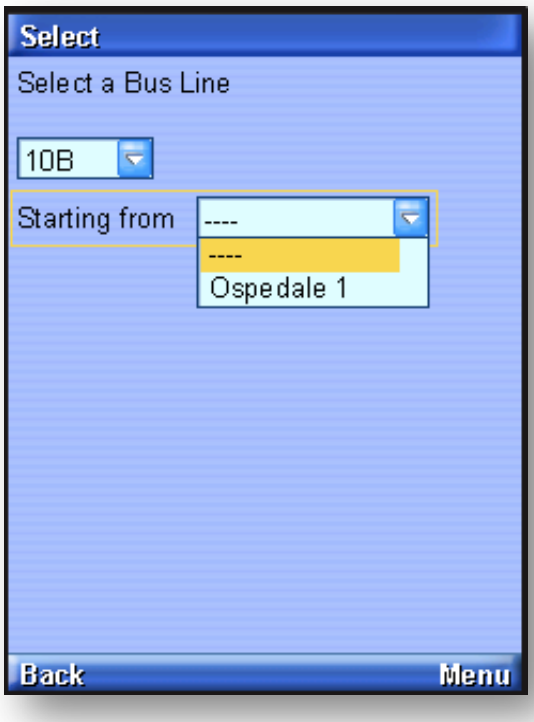
Web service sends an xml file to the client, which parses it and displays the timetable



This is the manual insertion form. The user can insert the start address and the destination. Clicking on Menu- Retrieve Time table will send the request to the web service, which process the received data and sends an xml file back to the mobile application.



This is the form where the user can select a bus line from the list box and then he has to click on Menu – get Start stations to specify the direction.

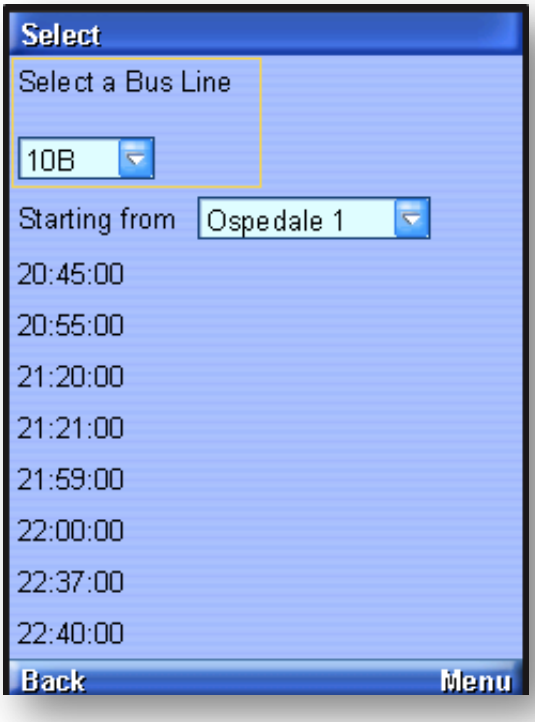


Then another list box appears, where the user can select a start station. This list is parsed out of an xml file stored on the mobile device. Once the start station is selected, the user has the possibility to retrieve the timetable from the webservice.

The device sends the bus line and the start station to the web service

Web service

The web service sends the appropriate xml file, which contains all start times, back to the mobile phone.



Code structure

Server side

Class name	Description
ConnectionManager	This class is used to create a connection with the MySql database.
DataRetriever	This class contains all the methods used to retrieve the requested data. The method “nearestStop” finds the 5 nearest stops in the range of 0.005 degree from the requested position. The method “findBusLine” finds the bus line that connects the starting address with the destination. “getTimetable” retrieves the timetable of a specified bus line. “getTimetableStartingStation” retrieves the timetable of a selected bus line starting from the selected station
GetTransitData	This class is used to create the connection between the client and the server. After a control on the parameters used in the URL a specific method form “DataRetriever” will be called

Client side

Class name	Description
Coordinates	This class is used to retrieve the GPS coordinates with the GPS device if present in the mobile phone
GoogleMaps	The class “GoogleMaps” is used to retrieve the map with a marker on the actual position. This will work only with the Gps form and only if the “Use google maps” checkbox in the Settings form is selected
GpsForm	This class can be used only if the “Use GPS” checkbox in the Settings form is selected. It will retrieve the GPS coordinates using the “Coordinates” class and show the map retrieved from the “GoggoleMaps” class
InsertForm	This class is used to show a form where the user can insert a starting address and a destination to retrieve the nearest stop and the bus line that connect the two addresses

LineForm	The “LineForm” class is used to show the selected line timetable at the start up if the user has selected a preferred bus line in the Settings form. Selecting a starting bus stop the user retrieve the timetable for the selected bus line.
MainMidlet	This is the main class. When the midlet starts up, the “MainMidlet” reads the Properties and chooses if the first form to show will be the Settings form or the Line form. This class will also create a connection with the GPS via the “Coordinates” class
Properties	This class is used to store the settings in a record store. It is used by all the other classes to retrieve the settings that will be used.
SelectForm	This class permits the user to select a bus line and a starting bus stop and retrieve the timetable
SettingsForm	The class “SettingsForm” is used to show the user the possible settings for the midlet. The Settings form is shown at the first start up. For the following start up it will be shown only if the user has not selected a preferred bus line. In any case the Settings form can be accessed via the menu
XmlParser	This class is used from all the other classes to parse the information retrieved from the server or from the local XML files.

Technical problems and solutions

Data management

The main problem we encountered during the development was the data storing. The first solution we tried to develop was the creation of some XML files with all the necessary data, but they were too big (33Mb) and were impossible to store all of it in the mobile phone. The final solution we decided to use to solve this problem was to combine 3 different possibilities. As explained before, we have used the RecordStore for the storing of the settings, 2 XML files to store some of the data about the bus lines and then we have created a webservice to store all the data about the timetable, the location of the bus stops and some other useful data.

User interface

Another big problem we had to solve was the design of the user interface. Despite our graphical solution is not very elaborated, we had to decide how to help the user to use our

program with less menus and choices as possible. Therefore we have decided to add the selection of a preferred bus line to show at start up, so the user can immediately have an overview to the usually used bus line timetable.
Also the disposition of the menu items had taken some time. They are studied to help the user to have all the useful option at the first position.