MOBI-ROMA

Mobile Observation Methods for Road Maintenance Assessments

D3 Development of Graphical User Interface

Deliverable Nr 3

September 2012

Foreca Ltd
Semcon AB
Klimator AB
Pöyry Infra GmbH

<u>**Design:**</u>

Rapid and Durable Maintenance Methods and Techniques

Cross-border funded Joint Research Programme

by Belgium, Germany, Denmark, Finland, France, Netherlands, Norway, Sweden, Slovenia and United Kingdom

This joint research programme was initiated by **ERA-NET ROAD**.

Project Nr. 832702

Project acronym: MOBI-ROMA

Project title:

Mobile Observation Methods for Road Maintenance Assessments

is a research project of the cross-border funded joint research programme

"ENR2011 DESIGN – Rapid and durable Maintenance Method and Techniques"

"ENR2011 DESIGN – Rapid and durable Maintenance Method and Techniques" is a trans-national joint research programme that was initiated by "ERA-NET ROAD II – Coordination and Implementation of Road Research in Europe" (ENR2), a Coordination Action in the 7th Framework Programme of the EC. The funding partners of this cross-border funded Joint Research Programme are the National Road Administrations (NRA) of Belgium, Germany, Denmark, Finland, France, Netherlands, Norway, Sweden, Slovenia and United Kingdom.

Deliverable Nr 3 – Development of Graphical User Interface

Due date of deliverable: 30.09.2012

Actual submission date: 10.09.2012

Start date of project: 01.09.2011 End date of project: 31.01.2013

Author of this deliverable:

Magnus Andersson, Semcon AB, Sweden

Version 1.0

Executive summary

MOBI-ROMA aims at improved, affordable and moderate-cost road condition and performance assessment techniques for effective monitoring and assessing maintenance needs across Europe. The maintenance tool developed in MOBI-ROMA requires input data from all seasons and from varying traffic conditions. The three main data types are spring thaw detection, winter road conditions and pavement quality estimation. All three types are collected using the floating car data methodology.

In order to demonstrate the applicability of the earlier data collection, analysis and fusion phase, a Graphical User Interface (GUI) has been developed to be used as a maintenance tool. The aim with the tool is that the maintenance personnel will be able to have

- updated knowledge of the road status,
- be able to identify needs for maintenance,
- be able to perform the proper action that is needed,
- geographically map the distribution of the maintenance needs,
- obtain input for cost and time calculation and planning.

This report describes the programming methods and functionality of the maintenance tool. The GUI has three main blocks: Client-side, Server-side and the Database. Data from several sources have been collected, analysed and fused into the maintenance tool database. Using the GUI these data are presented on an interactive map.

The information utilised in Mobi-Roma is of such extent that an extensive GUI had to be developed. Alternative representation methods, such as text, tables and graphs, would not cope with the large amount of data. Since all data used in Mobi-Roma is geographically positioned the logic choice was to use a map as the main method of data presentation. Surrounding functionality was then added to ease interaction with the map. Due to the extensive amount of data several methods that aim to ease the computation load on the map were developed.

The maintenance tool is available online on http://mobiroma.eu. The GUI is found under the link *Map* in the web page or at its direct link http://mobiroma.eu/Map.aspx. An application walkthrough is available within the GUI or at http://mobiroma.eu/Help.aspx. If changes are made in the maintenance tool, the walkthrough will be updated accordingly.

Next steps are to perform a self-evaluation of results, mainly on the maintenance tool GUI and on data fed into Mobi-Roma. Interviews with stakeholders will be organized, first and foremost with national road administrations. Demonstrations of the GUI will be held during evaluation and validation. A cost-benefit assessment of the maintenance tool together with the utilised data will be performed.

List of Figures

D3 Development of Graphical User Interface

Table of content

1 MOBI-ROMA User Interface in brief

The combination of different sources of data gives a novel opportunity for efficient monitoring and detection of variations in surface conditions, both before and after various maintenance works. The approach of using floating car data together with fixed measurement stations from various data sources is the core of Mobi-Roma. The collected data should be available in one coherent maintenance tool. The maintenance tool can be of several forms and shapes, depending on the data types. Since all data in the Mobi-Roma pilot are geographically positioned a Graphical User Interface based on a map application is the logic choice. Using a map as the application base eases the ability for operators to quickly and efficiently estimate pavement quality, load bearing capacity and winter road conditions of a road. A maintenance tool of this type provides objective information that aids decision making of often costly maintenance work.

The maintenance tool GUI is developed so that several data types can be fed into the same system; thus, allowing operators to use one tool for different road types and data types. The GUI also makes it possible to show long term effects on the road network; therefore, it is possible to compare current measurements with data collected last year. This allows operators in road maintenance stations to see trends, for example that a specific corner on a dirt road is affected by decreased load bearing capacity every year. In this example it could be an indication that the underlying road bed is weakened and has to be strengthened.

A graphical user interface, GUI, is the space where human-machine interaction occurs. The GUI allows users to manipulate application content, alter the application setup, to decide what information to show and much more. The GUI also displays the effects of a user's actions/manipulations. [1]

The information utilised in Mobi-Roma is of such extent that an extensive GUI had to be developed. Alternative representation methods, such as text, tables and graphs, simply would not cope with the large amount of data. Since all data used in Mobi-Roma is geographically positioned the logic choice was to use a map as the main method of presentation. Surrounding functionality was then added to ease interaction with the map. Due to the extensive amount of data several methods that aim to ease the computation load on the map were developed.

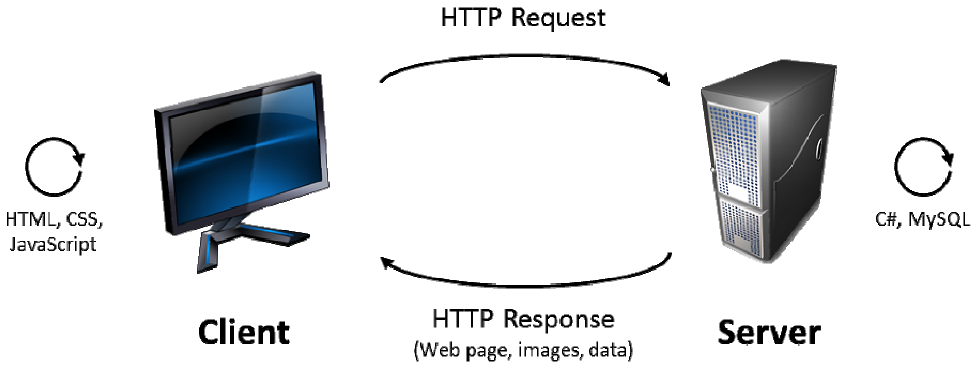

Figure 1 Network layout and functionality used in the Mobi-Roma GUI

The GUI can be divided into three main blocks:

- Client-side
- Server-side
- Database

In computer networking the client-side refers to operations performed at the client. Scripts on the client-side often run on a user's web browser, either on a standard desktop computer or on a smart-phone/tablet. In this case the client-side handles most of the functionality related to presenting graphical information, such as the map, buttons and images. The programming languages used on the client-side are JavaScript[1], HTML[2] and CSS[3]. A couple of external libraries eased development; the two main libraries are jQuery[4] and Google Maps JavaScript API[5].

[1] JavaScript is a prototype-based scripting language that is dynamic, weakly typed and has first-class functions.

[2] HTML is short for HyperText Markup Language and is the main markup language for displaying information in a web browser, such as web pages, images, buttons, etc.

[3] CSS is short for Cascading Style Sheet and describes presentation semantics of a document written in a markup language, such as HTML.

[4] jQuery is a JavaScript library that simplifies HTML document traversing, event handling, animating, etc.

[5] Google Maps JavaScript API allows developers to embed Google Maps on web pages. It also includes methods for adding objects on top of the map.

The server-side, on the other hand, handles operations performed by a server. In this case the server is located in an external web hotel. The server-side holds data processing methods, database logic and a web service used to retrieve information from the vehicle fleet. The server also houses the climate model and the interpreter, which runs every third hour, generating and interpreting new spring thaw forecasts. All server-side functionalities are built upon C#/.NET[6].

The database, located on the server-side, holds all data used in the Mobi-Roma pilot project. It keeps track of independent vehicles and the data collected by each vehicle and at the same time all information regarding load bearing capacity, pavement quality and winter road conditions. Due to privacy regulations no information is stored that connects a specific vehicle directly with individual measurements. The database also holds weather information, used by the climate model, downloaded from external sources. The database is built upon a MySQL[7] database.

Communication between the client-side and the server-side is handled using HTTP[8] and JSON[9].

[6] The .NET framework provides a comprehensive and consistent programming model for building desktop- and web-applications. C# is one of several programming language within the .NET framework.

[7] MySQL is an open source relational database. SQL stands for Structured Query Language and is a programming language designed for managing data in relational database management systems, RDBMS.

[8] HTTP stands for Hypertext Transfer Protocol and is an application protocol used in distributed, collaborative and hypermedia information systems. HTTP is the foundation of data communication on the World Wide Web, WWW.

[9] JSON stands for JavaScript Object Notation and is a data-interchange format that is easy to read/write and parse/generate for both humans and machines.

2 Contents and how to use MOBI-ROMA GUI

The Mobi-Roma maintenance tool GUI currently consists of four main data types, collected from several parallel projects.

- Load bearing capacity generated by combining floating car data with weather data and information collected from fixed measuring stations along the road net. The computations result in about 30.000 4x4 km squares that currently cover Sweden. The outputted result in each square contains a timestamp and can take three colours: green, orange or red. Green means that the risk finding a soft gravel road within that square is very low. An orange square means that roads within the square could be affected by spring thaw and thereby a possible decreased load bearing capacity. Red squares represent high risk of decreased load bearing capacity, several road segments within the square are affected by spring thaw. The squares are interpolated to three square sizes; 20x20, 40x40 or 200x200 km. Thus allowing the opportunity for specific users to see less detailed information.

- Load bearing capacity information generated directly in the sensor equipped vehicles. The data in this category can be used to see exactly in which corner of a road spring thaw is occurring. It is also possible to see the relationship between soft and dry/frozen indications. This data type is presented as geographically positioned indications/markers. In this mode there are two types of indications; red/yellow markers indicate that a soft segment of the road has been identified and a green marker that the road segment is dry or frozen.

- Pavement quality is presented as indications/markers. Vehicles automatically upload new information to the database and it is in real-time available on the map. A scale from zero to ten indicates the pavement quality, ten being good/smooth pavement. Each grade is represented by a color, going from purple (bad) to green (good). This allows maintenance operators to easily detect stretches with a low quality grade.

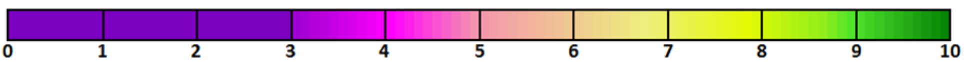

Figure 2 Pavement quality color scale, where each color represents a grade

- Winter road conditions are displayed as geographically positioned markers placed on their corresponding coordinates. A red/yellow marker indicates that a slippery road surface has been detected and blue indications correspond to measurements where no slipperiness has been detected.

- In addition to the four main categories presented above weather information is also available in the GUI. The information is fetched from an external weather provider and is not related to the weather information or fixed measuring stations used to generate outputs in the four main categories. Both load bearing capacity and winter road conditions are highly related to the weather and can therefore be used when interpreting results from the main data types available in the maintenance tool.

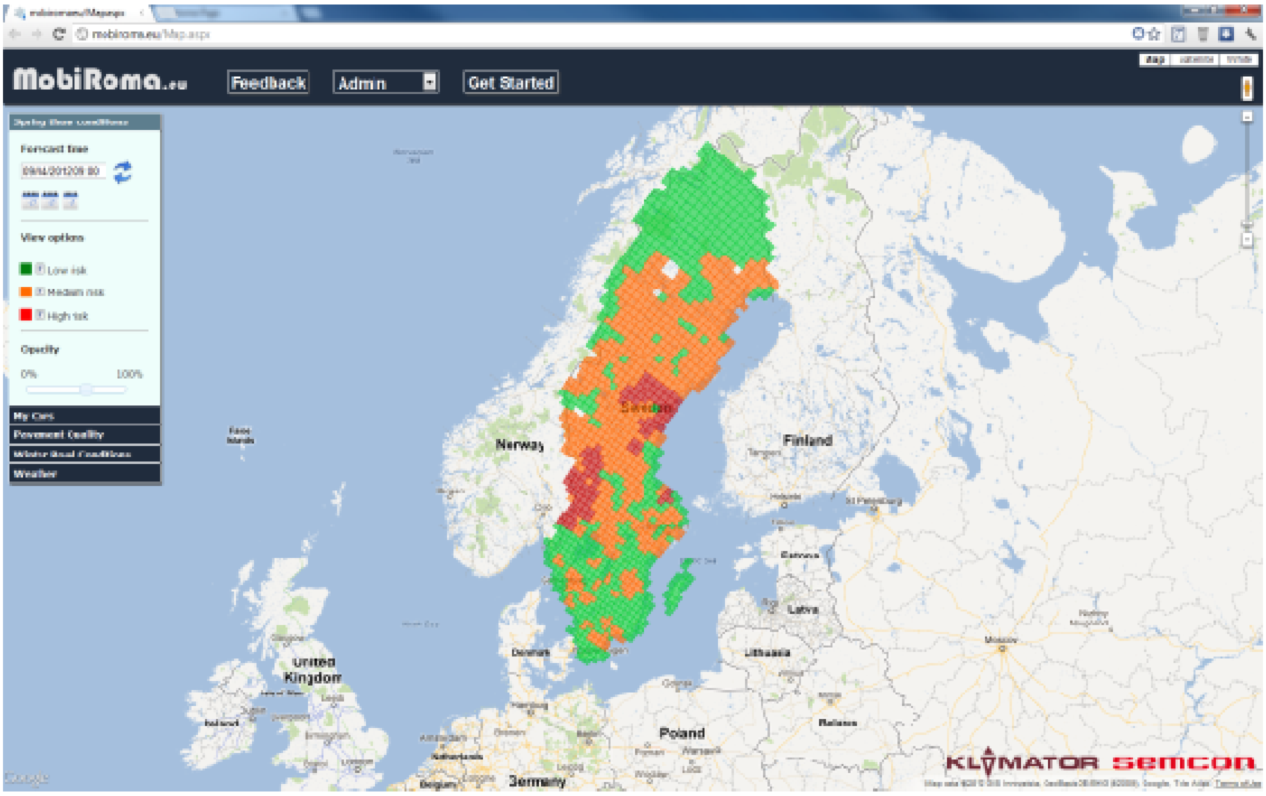

Figure 3 Mobi-Roma maintenance tool GUI

Each data type has its own section with corresponding settings and customization possibilities attached to it. When a new section/tab is selected the map is cleared and information for that specific section/tab is shown on the map. The new information is shown with suitable default settings. As seen in Figure 4 the *Pavement Quality* tab is selected and data regarding other tabs is hidden while pavement quality data is displayed. The graphics of the underlying map can be changed and currently there are three map modes: normal map, satellite map or a white map. To change the map type use the buttons in the top right corner of the GUI.

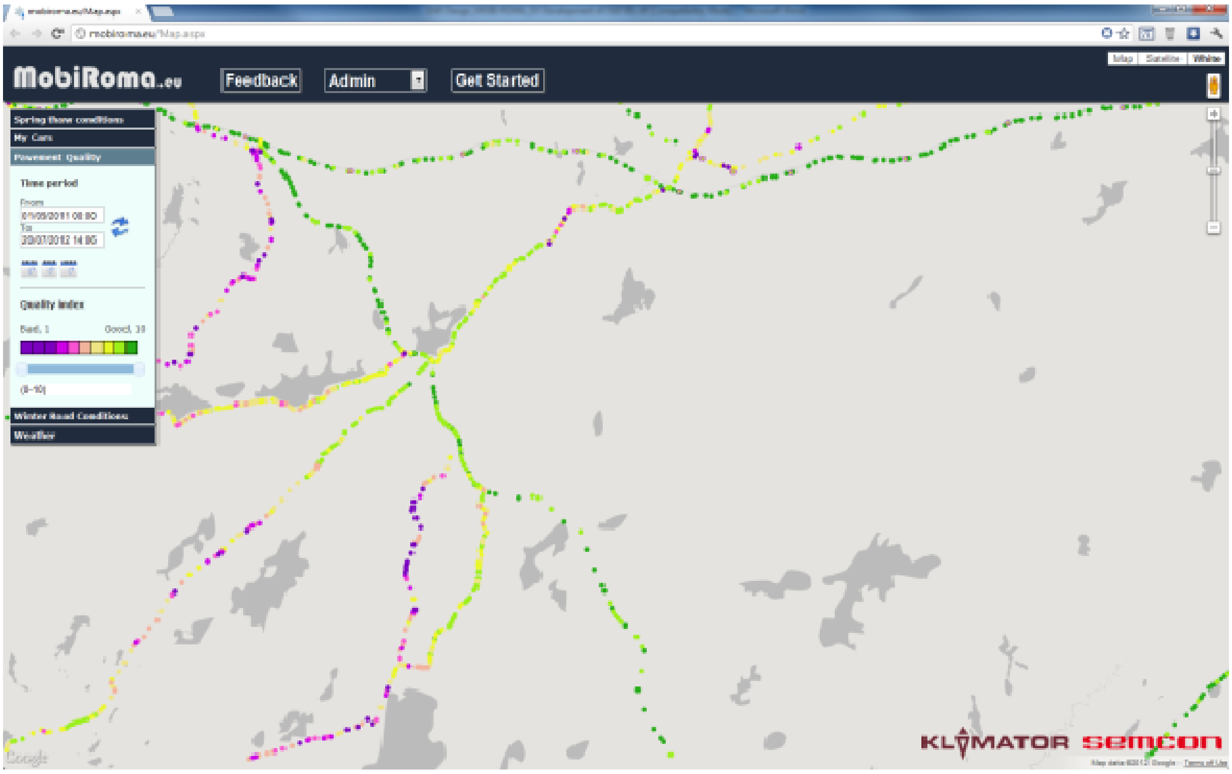

Figure 4 Overview of GUI with the Pavement Quality tab selected

In Figure 5 the settings window for the computed load bearing capacity is shown, in the GUI this is currently named *Spring thaw conditions*. The window allows the user to choose a timestamp at which data should be fetched. Under the timestamp three pre-set example timestamps are available. It is also possible to change view options and opacity. By clicking the view option buttons the user decides what risk levels to show. The opacity level sets the transparency of all squares, used to locate roads, cities, etc. initially hidden behind the squares. When the user is satisfied with the settings the map is updated when the button showing two blue arrows is pressed.

Figure 6 shows the settings for *My Cars* which corresponds to the information fetched directly from the vehicle fleet. The user sets a time span within which data should be fetched. Three example time spans are available directly underneath the input fields. View options function in the same way as in the *Spring thaw conditions* window.

The settings for pavement quality seen in Figure 7 include the same time span settings as in *My Cars*. In addition to this it is possible to set the quality range. The span slider sets the maximum and minimum pavement quality grade.

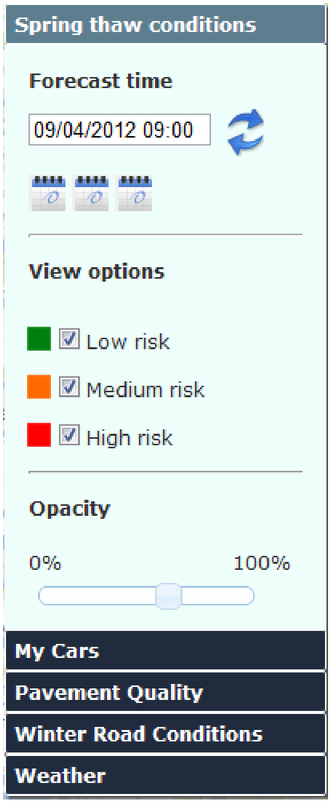

Figure 5 Spring thaw conditions settings

Figure 6 My Cars settings

Figure 7 Pavement quality settings

In Figure 8 the user can set display options for *Winter road conditions*. The timestamp has the resolution of one day, and all data between 00:00 and 24:00 on the chosen day is fetched. In addition to setting a date the user can set whether markers indicating slippery road surface should be shown or not. The same option is available for the non-slippery indications.

The standard weather information is always shown, but the user can show/hide an overlaying cloud cover. As seen in the settings window shown in Figure 9.

Figure 8 Winter road conditions settings

Figure 9 Weather settings

Almost all indications contain information windows which are displayed when a marker is clicked. The information window contains additional measurement information such as timestamp, coordinates and sensor imei id. The imei id is used to connect the sensor units with its corresponding measurements stored in the central database.

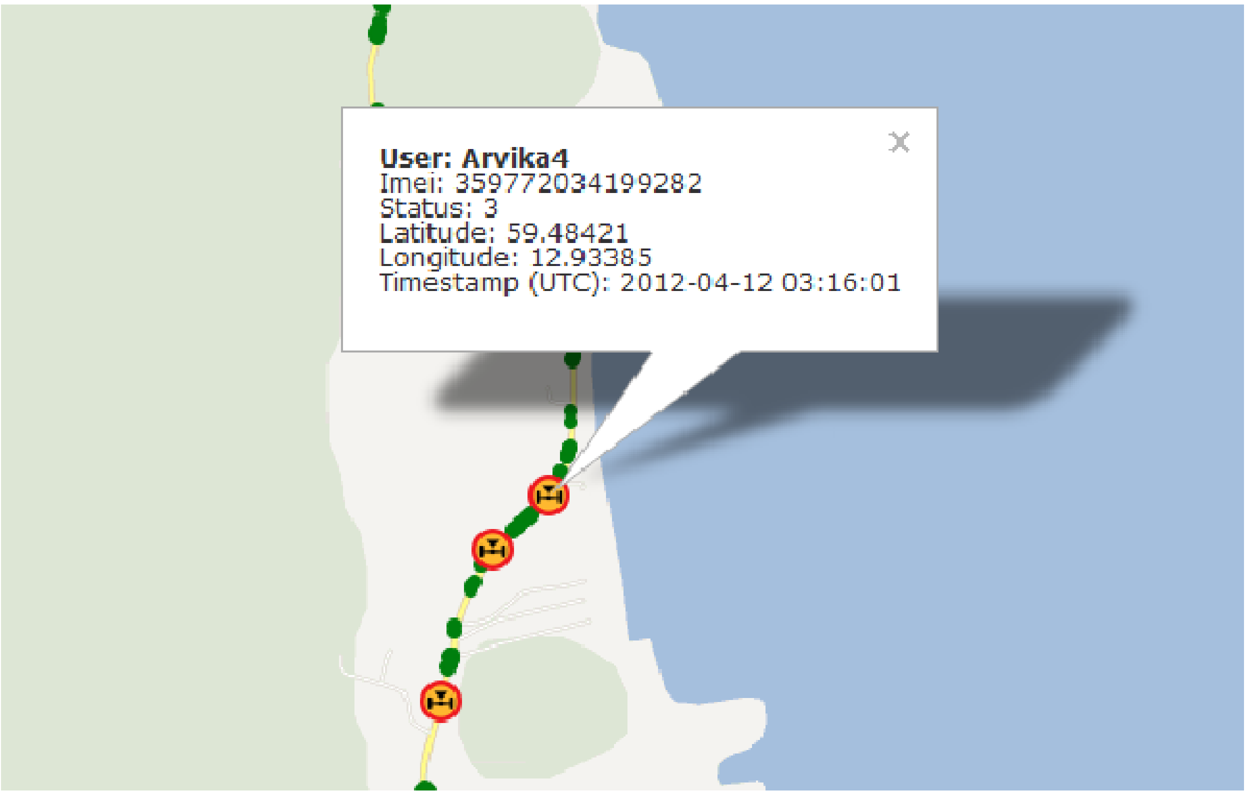

Figure 10 Information window

The maintenance tool is available online on http://mobiroma.eu. The GUI is found under the link *Map* in the web page or at its direct link http://mobiroma.eu/Map.aspx. An application walkthrough is available within the GUI, seen in Figure 11, or at http://mobiroma.eu/Help.aspx. If changes are made in the maintenance tool, the walkthrough will be updated accordingly.

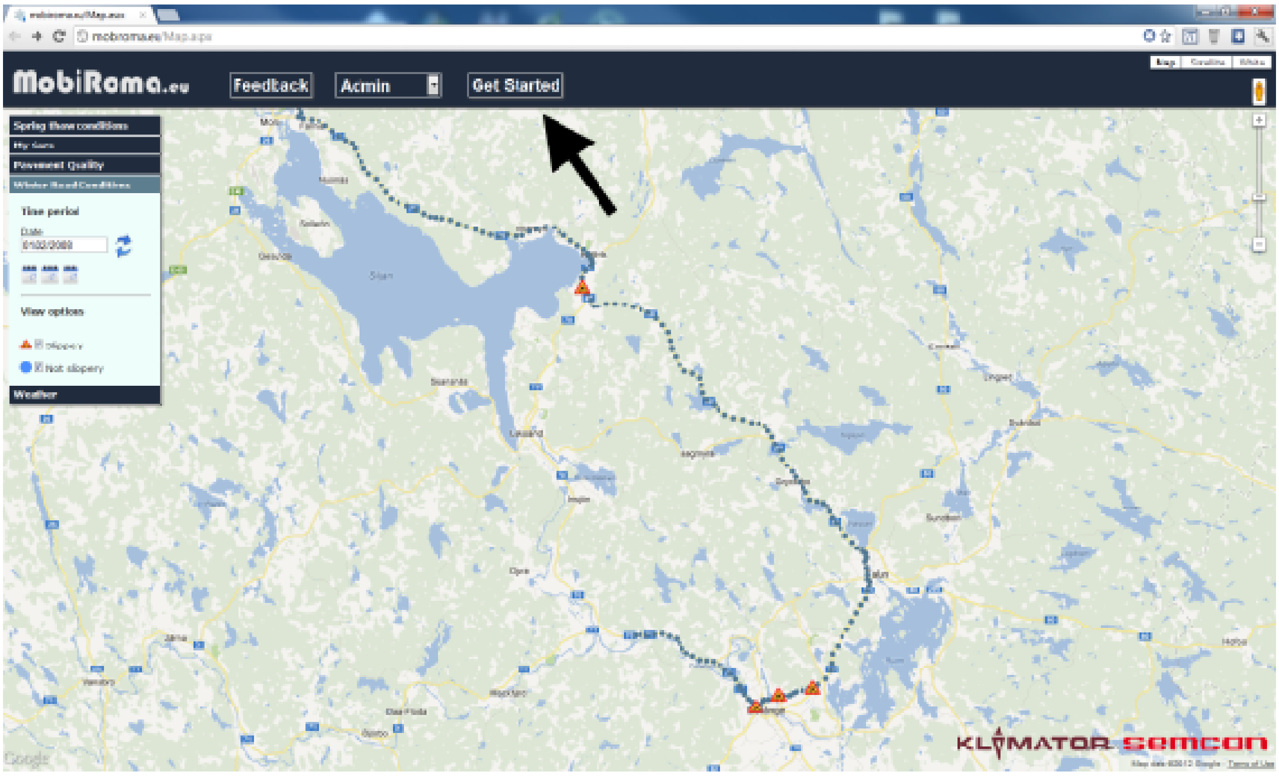

Figure 11 Link to application walkthrough

3 Summary and next steps

The Mobi-Roma GUI fulfills the expectations of a pilot. Data from several sources are fed into the maintenance tool and presented on an interactive map. The GUI shows how data from both, a vehicle fleet and fixed measuring stations, can be presented in a way that allow operators to optimize maintenance work. It also shows how different types of data can be available in one coherent tool. Current data can easily be compared with last season's data. Differences between various times of a year and traffic conditions can be isolated and analyzed. The GUI shows that geographical positioned measurements, displayed on a map, could aid operators and un-trained users to set up maintenance plans.

When data from several sources are combined in one tool the computation load on the GUI and the underlying application is high. Several methods have successfully been developed to cope with the computation load. The core problem is directly related to the demand of real-time data transfer from the central database to the map. When the vehicle fleet is increased the computation load will increase dramatically. The Mobi-Roma pilot GUI will not be able to handle this smoothly, which means that loading times will increase and the usability of the tool will decrease. Work has already begun with modifications that allow for more data to flow through the system, without affecting the real-time map experience that the user has today. The main update is to develop a method that uses road segments onto which measurements are added. These segments are then updated separately without affecting the real-time experience. Using segments will lead to increased usability and for data to be presented in more dynamic ways. For example the mean load bearing capacity of a specific segment from this year can be compared to the same segment mean from last year.

To increase the usability even further purpose-built applications for smartphones and tablets could be developed. This would provide operators with a mobile version of the maintenance tool which could be used during field trips and at sites with on-going maintenance work.

Next steps are to perform a self-evaluation of results, mainly on the maintenance tool GUI and on data fed into Mobi-Roma: load bearing capacity, pavement quality and winter road conditions. Interviews with stakeholders will be organized, first and foremost with national road administrations. Demonstrations of the GUI will be held during evaluation and validation. A cost-benefit assessment of the maintenance tool together with the utilised data will be performed.

4 References

1. Professional Windows GUI programming using C#, Choudhury, Wahid, ISBN: 1861007663