



# Lovos Maps

Logistics Route Planner

<b>Data student:</b>	
Family name , initials:	Kusuma, D.A.
Student number:	2184973
project period: (from – till)	9 February 2015 – 19 June 2015
<b>Data company:</b>	
Name company/institution:	Vos Logistics
Department:	IT Application Management
Address:	Vorstengrafdonk 39, Oss, the Netherlands
<b>Company tutor:</b>	
Family name, initials:	Rouwenhorst, A.
Position:	Application Department Team Lead
<b>University tutor:</b>	
Family name , initials:	Henning, F.L.
<b>Final report:</b>	
Title:	Lovos Maps
Date:	June 19 <sup>th</sup> 2015

Approved and signed by the company tutor:

Date:

Signature:



## *Preface*

*This report represents the activities carried out in the last five months of my graduation internship at Vos Logistics. It was a great opportunity to be able to complete this graduation internship as final part of my study at Fontys Hogeschool.*

*I would like to thank the company tutor Arno Rouwenhorst (Application Team Lead), company software engineer Marco Stolk for their kindness and guidance during this project, all other employees of Vos Logistics for the feedback and support, and last but not least I would like to thank school project tutor Frank Henning for the guidance during this graduation assignment and writing this report.*

*I would also like to thank my family for supporting me, without them I will not be able to finish four years of study in the Netherlands.*

**Dani Ardi Kusuma**

## Contents

Summary .....	i
Glossary.....	ii
1. Introduction .....	1
2. Company .....	2
2.1. Quality .....	2
2.2. History .....	3
2.3. Organization Chart .....	3
3. Assignment Overview .....	4
3.1. Initial Situation .....	4
3.2. Description .....	5
3.3. Project Constraint .....	6
3.4. Method of Approach.....	7
3.4.1. Research methodology .....	7
3.4.2. Development methodology .....	8
4. Process .....	9
4.1. Project Initiation .....	9
4.2. Research.....	9
4.3. Implementation .....	15
4.4. Prototyping and Feedback .....	21
4.5. Requirement Change .....	22
4.6. Testing.....	22
4.7. Final Product .....	22
5. Conclusion & Recommendation .....	23
Evaluation .....	24
Reference .....	25
Attachments.....	26
I. Project Plan .....	26
II. Screenshots.....	41
III. Questionnaire Result Overview .....	47
IV. Test Report.....	52
V. Class Diagram .....	57
VI. Database Diagram.....	59

## Summary

This graduation internship was accomplished at Vos Logistics, an independent European logistics service provider, delivering a wide range of transport and customer-specific logistics services. Vos Logistics developed in house software, a transport management system, called Lovos. Lovos is the core transaction system for Vos Logistics for entering orders up to the creation of invoices. The system further supports route calculation, which is the middle process between ordering and invoicing. The route planning software makes use of an external route planning server called 'PTV'. PTV is a company that offers specialized route calculation server focused on the transport companies. PTV has many varieties of features to be implemented in the route calculation software.

The current route planning software developed approximately 10 years ago was no longer supporting the recent conditions that the users require. The problems are that the user interface is not user friendly, more functionalities are needed to calculate efficient routes, some of the functionalities implemented are not working properly and the program were written in an unstructured way (many multiple codes) in C++ language which is no longer supported by the new version of Microsoft Visual Studio.

These days users are getting more familiar with the user friendliness and modern features of for example Google Maps. It is becoming a standard user requirement. The current map cannot optimally support the user requirements standard. Therefore it is decided to create a new standalone map with responsive display and efficient route calculation possibilities. The student trainee was responsible to do research about the solution for the problems.

The requirements were not set ahead and the application has a lot of user interaction, for that reason the student trainee used the prototype methodology. By using the prototype methodology the application prototype was built, tested, evaluated by the users and then refined as necessary until an acceptable prototype is achieved. The user evaluation was the most important part of this approach.

To be able to achieve the complete application the student trainee did research for the application requirement. Afterwards the student trainee implemented the prototype to be evaluated by the users. After the requirements were achieved the next prototype was built. These processes reiterated until the end of the graduation internship period and application testing finalized the software development.

The application was completed and written in C# language. The user interface shows a lot of improvement compared to the current route planning software. The new map consists of many filters which enable the users to calculate the most efficient route possible. New features were also implemented to optimize the route calculation and user experience. Considering the lack of time the student trainee were not able to integrate the new Lovos Maps to every place where the current route planning software is implemented in Lovos.

The new standalone map application called "Lovos Maps" is delivered to Vos Logistics and available to be used for business activities. In future development the integration of Lovos Maps in Lovos needs to be done and more features like an intermodal journey where the route consists of the combination of a truck with another transportation modality can be added.

## Glossary

### **Lovos - Vos Logistics Fleet Management information System**

Lovos is a transport management system that can process all transport orders from Order Entry, with all necessary information, up to invoicing. (Vos Logistics, 2015)

### **C++ - C Plus Plus**

C++ is a general-purpose programming language. It has imperative, object-oriented and generic programming features, while also providing the facilities for low-level memory manipulation. (Wikipedia, 2015)<sup>[1]</sup>

### **C# - C Sharp**

C# is a multi-paradigm programming language encompassing strong typing, imperative, declarative, functional, generic, object-oriented (class-based), and component-oriented programming disciplines. (Wikipedia, 2015)<sup>[2]</sup>

### **PTV**

The PTV Group provides software and consulting services for transport, logistics and geomarketing. (PTV)<sup>[3]</sup>

### **WPF – Windows Presentation Foundation**

Windows Presentation Foundation (WPF) provides developers with a unified programming model for building rich Windows smart client user experiences that incorporate UI, media, and documents. (Microsoft, 2015)<sup>[4]</sup>

## 1. Introduction

Technology evolves rapidly, software becomes out-of-date while time passes by. Vos logistics is proud of their own established route planning software product in the 'Lovos' transport management system, which is in use for more than 10 years now. With the fast emerging markets and the need of better cost management to survive the market competition, the route planning software needs a major update.

Vos Logistics, established in 1944, is an independent European logistics service provider, delivering a wide range of transport and customer- specific logistics services. The company's head office is located in Oss, the Netherlands. The company has four core activities: transport of cargo (packed goods) and bulk (dry unpacked goods both granulates and powders), logistics services (forwarding, warehousing and value added services and distribution to full supply chain), and fleet service. They have 1200 trucks, 3000 trailers, and 1900 employees servicing around 30 countries in Europe.

The transport management system used in the transport divisions of Vos logistics called 'Lovos', developed from 1994 onwards, is still being used by this company with 88 released versions. Lovos is a wide range software package used for managing shipments, planning, invoicing, and route planning. There are about 800 transport orders handled every day, equaling 200.000 shipments per year (2014). For every shipment, the route planning software is used.

The current route planning software, implemented approximately 10 years ago is no longer up to date to the current standards. More functionality is needed to cope with the strong competition on the market and to execute transport efficiently. The current route planning software does not obtain all the necessary filters and functionalities.

Vos Logistics decided to renew the current route planning software in order that the application meets the user requirements. To resolve the problems occurred, a new standalone route planning software is produced. The development process make use of a prototyping methodology because this software has a lot of user interactions. During the development a lot of feedback sessions need to be held to achieve a user friendly end product which meets the requirements that are described in the initiation phase.

During this graduation internship the student trainee focused on the research to find the best possible adjustments for the new route planning software, developed the application, gave demos of the prototype and tested the application.

The second chapter gives in depth details about the company. The third chapter describes the assignment. The fourth chapter explains about the development process and approach. The fifth chapter shows the result. The sixth chapter describes the conclusion.

In the attachment the project plan, screenshots, questionnaire result overview, test report, class diagram and database diagram can be found.



## 2. Company

(Vos Logistics, 2015)<sup>[5]</sup> Vos Logistics is an independent European logistics service provider, delivering a wide range of transport and customer-specific logistics services. The company has four core activities:



Figure 1: Vos Logistics' Trucks (taken from [www.voslogistics.com](http://www.voslogistics.com))

- **Cargo** – consists of the transport of packed goods. Its operations comprise mainly of FTLs (Full Truck Loads), part loads and intermodal transport. The markets Vos Logistics serves include the automotive, consumer electronics, retail and construction industries.
- **Bulk** – comprises mainly the transport of dry unpacked goods (both granulates and powders). The products include chemicals, foodstuffs, animal feed, starch, minerals, ADR and waste. In the chemical industry. Vos Logistics is a specialist in freight management, steering goods flows from a variety of production locations. In addition, bulk provides handling, storage and packaging services as well as tank cleaning activities.
- **Logistic Services** – offers customer-specific logistics solutions: from forwarding, warehousing and value added services and distribution to full supply chain solutions in which Vos Logistics assumes the management of its customers' goods flows in full or in part.
- **Fleet Services** – offers efficient transport and distribution solutions to companies with their own dedicated vehicle fleets: analysis of transport needs, fleet financing, replacement and sale of used equipment, preventive maintenance, fuel optimization, insurance, management of seasonal influences, recruitment, training and management of drivers and planning tools.

With a network of 25 group-owned locations, Vos Logistics is active throughout Europe and in the bulk and volume transport markets ranks among the largest road haulers in Europe. With just under 1,900 employees, the company operates a modern fleet of 1,200 (mainly Euro 4 and 5) vehicles, 3,000 loading units and 170,000 m<sup>2</sup> of storage space.

The strength of the company lies in its combination of innovative skills in both transport and logistics, a highly developed European network, advanced transport management systems (e.g. Lovos) and a sharp focus on quality and service. These qualities are reflected in the high degree of customer satisfaction, particularly with prompt delivery times, low loss rates and value for money.

### 2.1. Quality

Vos Logistics is ISO (International Organization for Standardization) 9001, ISO 14001, ISO 22000, GMP (Good Manufacturing Practices), SQAS (Safety and Quality Assessment Systems) and ADR (certificate for transport dangerous good) certified. Vos Logistics also has an AEO (Authorized Economic Operator) status



for three locations and is an active participant in the Responsible Care (R) program, an initiative by the ECTA (European Chemical Transport Association). The company trains its employees at its own Vos Logistics Education Centre. The Mobile Quality Team, which monitors transport procedures and can be deployed throughout Europe, is also closely involved in driver training.

## 2.2. History

On 1 June 1944, Harry Vos began a delivery service between Oss and Nijmegen using just one Chevrolet. In the reconstruction following the war, the founder benefited from the strong growth of the transport sector and Harry Vos Transport grew into one of the biggest transport companies in the Netherlands. With 21 vehicles and 37 employees, he also began to provide international transport services. The company's name was changed in 1970 to “Expeditie en Internationaal Transportbedrijf Harry Vos BV”. By the end of that decade, the company had 130 employees and 80 trucks. Under the leadership of Wim Vos, the founder's son, the company expanded quickly in the 1980s through both organic growth and acquisitions at home and abroad. The name of the logistics company that emerged was changed to “Harry Vos Transport Group”. The company has been operating under its current name, “Vos Logistics”, since 1998.

## 2.3. Organization Chart

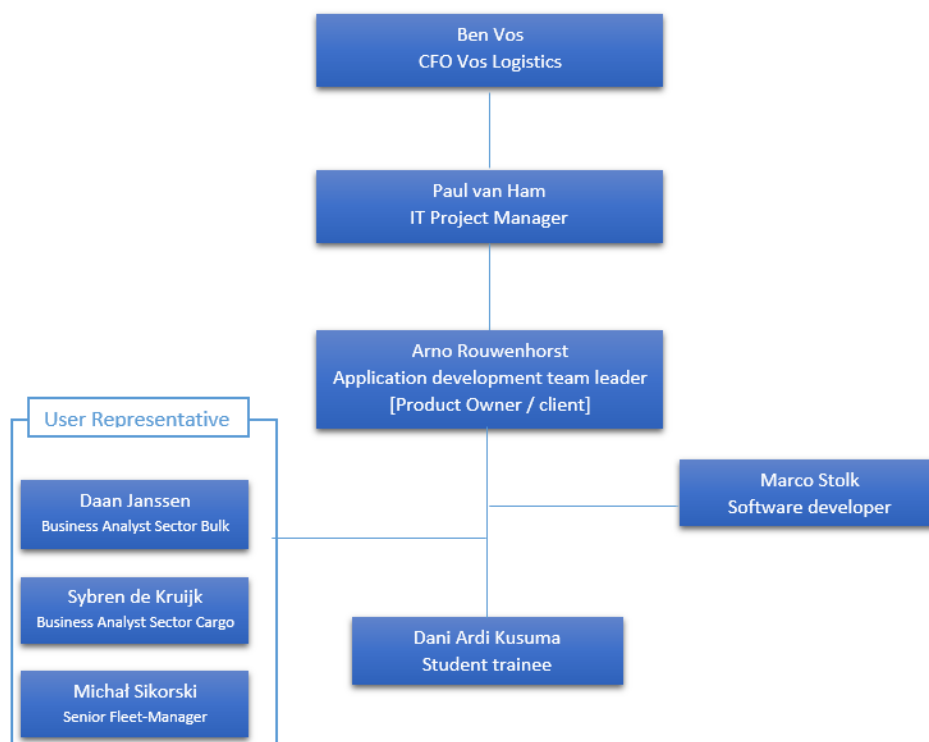


Figure 2 : Organization Chart

As stated in the chart above, the student trainee has a responsibility to develop a new route planning software called “Lovos Maps” for the client Arno Rouwenhorst. The prototype of the software will be tested by the user representative.

### 3. Assignment Overview

#### 3.1. Initial Situation

Lovos, the transport management system of Vos Logistics, is the core transaction system for Vos Logistics for entering orders to the creation of invoices. The system further supports route calculation, which is the middle process between ordering and invoicing. The current route planning software developed approximately 10 years ago is no longer supporting the recent conditions that the users require. Some of the features and filters are not working optimally.

The Lovos program is written in C++ language with an unstructured pattern and has many unused code. Nowadays the software developers of Vos Logistics are busy with revitalizing and restructuring the entire Lovos into C# environment step by step. The unused and faulty code will be deleted from the system in the future.

The current route planning software is a part of Lovos and makes use of an external route planning server called 'PTV'. PTV is a company who offers specialized route calculation server focused on the transport companies. PTV has many varieties of features to be implemented as a route calculation software. PTV supports in house development of route planning for Java and C#. For C# development, they provide a toolbox for the WPF application, demo version and user manual.

There are three types of servers for which Vos Logistics purchased a license from PTV. XMap server provides the map to be displayed in the route planning software. XRoute server provides functionality to calculate a route which will return many information about the route calculated. XLocate server provides functionality to find a location on the maps or by input.

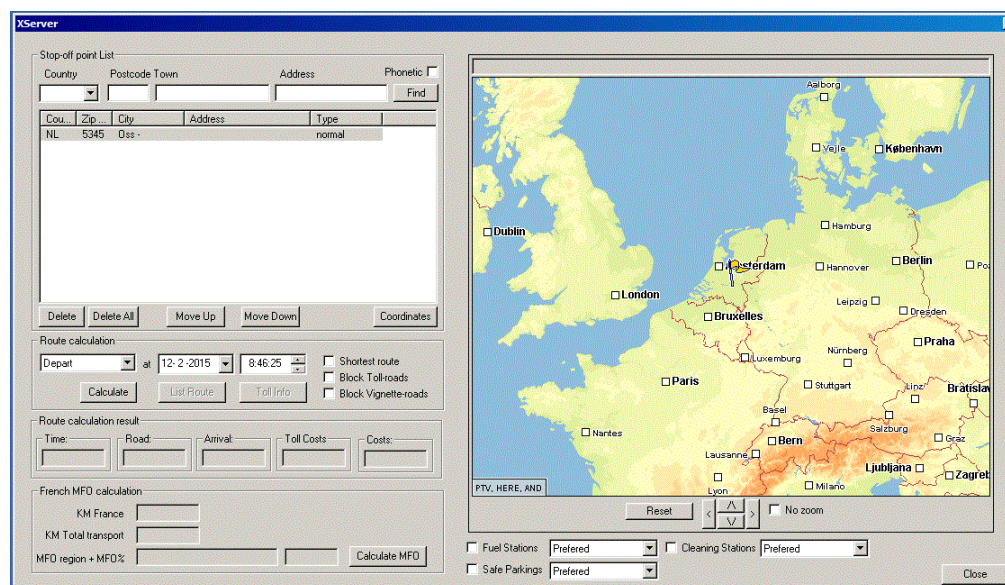


Figure 3: The current route planning software (taken from Vos Logistics Lovos)

The current route planning software (figure 3) cannot optimally support 800 shipments daily made by the planning and fleet management departments. Some of the problems are:

- The user interface is not user friendly (display too small, map cannot be moved by dragging and it has old map style).
- Many new functionalities provided by PTV are not being used such as: route calculation with vehicle dimension, alternative routes, avoid ferry routes, drag to move the map and many more.
- Functionalities implemented are not working properly.

These days user get used to the user friendliness and modern features of online map service like “Google Maps”. It is becoming the standard of user requirement these days. The current route planning software cannot optimally support the user requirements standard, it is inevitable to create a new route planning software with responsive display and efficient route calculation possibilities.

### 3.2. Description

Considering the problems of the current route planning software, the upgrade for the map is crucial for Vos Logistics being able to compete with other logistics companies. The outdated route calculation software needs a thorough update to be capable to calculate the most effective route for truck transportation. Therefore there is a need to recreate the software using WPF in C# which can be used for modern style design and possibility to apply the intelligent features from PTV. Developing the new route planning software with WPF can make sure that the user interface can be designed responsive and modern, development with WPF is easier and faster than windows form and the WPF application is extensible to a web application for future purposes.

The requirements for the new Lovos Maps below are the outcome of the research phase which will be explained in the next chapter.

The main functionalities to be implemented are:

1. Displaying modern style, responsive and interactive (zoom using the scroll wheel, drag move, and bigger display) maps.
2. Searching for places by keyword.
3. Calculating routes between two or more addresses.
4. Estimating travel time.
5. Filtering route calculation with blocking the toll road.
6. Filtering route calculation with blocking the vignette road.
7. Filtering route calculation with blocking the highway road.
8. Filtering route calculation with blocking the ferry road.
9. Finding alternative routes (shortest, optimal, and fastest route).

10. Calculating routes depending on vehicle characteristics, such as height, weight and the EURO norm of the vehicle.
11. Ability to view POI (point of interest) (for example a truck cleaning station, safe parking place or preferred gas station) on the maps then adds them to the route

The extra functionalities that can be added when the main functionalities are implemented, are:

1. Displaying the details of the route.
2. Calculating toll costs on the route.
3. Displaying address details.
4. Integrating with Lovos.
5. Locating with coordinate.
6. Showing toll part of the route with different color.
7. Locating with postal code.
8. Calculating emission depending on vehicle type.
9. Displaying satellite view.
10. Changing routes with drag and drop.
11. Displaying live traffic information.
12. Displaying the use of gas and cost of it.

The new route planning software has to be able to calculate the most efficient route with many filters and it has to be integrated with Lovos. Demo (training) of the application to the users and a user manual will be delivered.

### 3.3. Project Constraint

The constraints during this project are:

1. The product at least with the main functionalities should be ready by June 5<sup>th</sup>, 2015.
2. The tool used to produce the product is Microsoft Visual Studio 2012.
3. The program must be written in C# language.

### 3.4. Method of Approach

#### 3.4.1. Research methodology

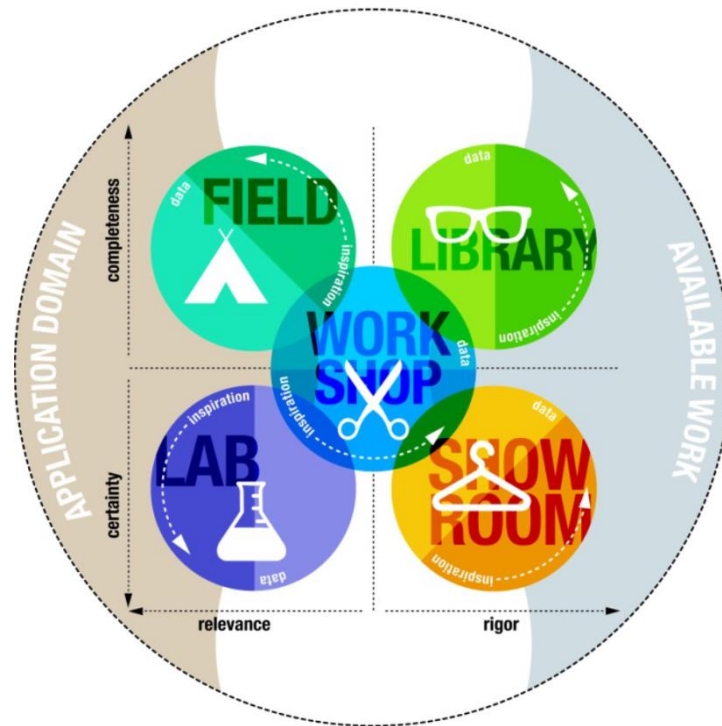


Figure 4: The DOT Framework (Turnhout, K. van et al., 2015)<sup>[9]</sup>

The research methodology being used is the DOT (Development Oriented Triangulation) framework. This framework is used for all research questions that arose during the development. The DOT framework consists of 5 points: Field, Library, Workshop, Lab, and Showroom.

- **Library** → check the internet, books, articles and other source. For example: check on the internet how does a modern route planning software look like.
- **Field** → this strategy is used to position the project in the applied area by looking at, searching and investigating the applied area (inside the company). For example: hold an interview with the expert in the company and create a questionnaire to ask the users what they need.
- **Workshop** → Design and build the program based on the library and field results.
- **Lab** → Test the solution built with the use of the predefined test plan. Use different test approaches: user acceptance, performance and integration tests.
- **Showroom** → demonstrate the program to the client comparing the application with the former application if applicable.

### 3.4.2. Development methodology

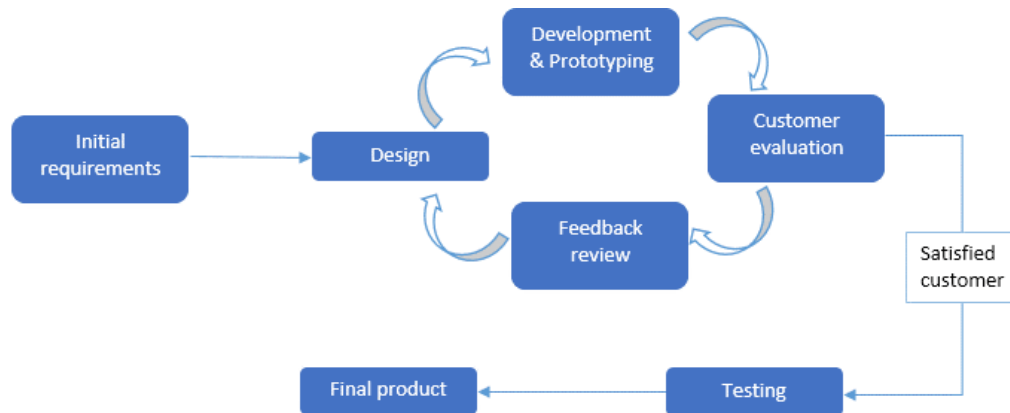


Figure 5: Prototyping Methodology

The prototyping methodology as shown in figure 5 above is a software development methodology in which a prototype (an early approximation of a final system or product) is built, tested, and then refined as necessary until an acceptable prototype is achieved. Therefrom the complete system or product can now be developed. This model works best in scenarios where not all of the project requirements are known in detail ahead of time. It is an iterative, trial-and-error process that takes place between the developer and the users. (Rouse, M.) <sup>[8]</sup>

#### **Advantages of the prototype model:** (istqb exam certification) <sup>[7]</sup>

- Users are actively involved in the development.
- Since in this methodology a working model of the system is provided, the users get a better understanding of the system being developed.
- Errors and bugs can be detected and solved much earlier.
- Quick user feedback is available leading to better solutions.
- Missing functionality can be identified easily.
- Confusing or difficult to understand functions can be identified. Requirements validation, quick implementation of incomplete but functional application.

#### **Disadvantages of the prototype model:** (istqb exam certification) <sup>[7]</sup>

- If the requirements are not yet achieved, then it leads to repairing the built systems.
- The scope of the system can be expanded during the development which may increase the complexity of the system.
- Incomplete application may cause the application not to be used at all.

The prototyping methodology is used for this project, because this software has a lot of user interaction. In order to give the best quality and user experience, feedback sessions with the users are held. The evaluation from the customer is an important part of this project. When the users are involved in a project, the feedback from the users can be used to adjust the user friendliness of the user interface. The phasing of the methodology will be explained in the next chapter.

## 4. Process

In this chapter, the process of completing the project is explained. The explanation will follow the structure of the development method in the chapter 3.

### 4.1. Project Initiation

At the beginning of the graduation internship, the student trainee tried to apply an agile methodology for this project, in which there is working software after each sprint. However, the situation in the company is not enabled to apply an agile methodology, because the company does not apply agile methodology. After consultation with the client and university tutor the student trainee decided to use the prototyping methodology considering this software will have a lot of user interactions. By using prototyping methodology the student trainee gets more input from the end users to improve the application.

For the division of work, the requirements is divided into small parts between two periods of development. In the first period the main requirements are developed. During the second period, the extra requirements of the product are developed depending on the time left.

Every week a meeting with the client was held. In the meeting, feedback about the working software was given by the client (Arno Rouwenhorst). In order to realize the requirements, the research about how to accomplish the requirements is done and why does the student trainee choose the approach. The next subchapter will contains explanations about the research that is done.

### 4.2. Research

To be able to design and implement the application, the student trainee need to do research about several topics.

The research topics are divided into research questions and the questions are answered using the research methodology (DOT framework) described in chapter 3. Every research question will answer what, how and why it is being chosen.

The research questions during the project were:

1. How to build a modern, responsive and user friendly user interface for the user (Vos Logistics' employees)?
  - Library
    - ✓ The student trainee searched and studied how the most popular online map service design their planner (For example: Google Maps, Bing Maps, Tomtom route planner and Here Maps) which result in a list of requirements for the display. Below the reader can find a screenshot of Google Maps, the most popular online map service worldwide.
  - Field



- ✓ The student trainee created a questionnaire (see attachment III for the questionnaire details) for the end users to find out what their preferences are. In total, 27 Vos Logistics colleagues responded (result of the questionnaire: Google Maps layout is the most chosen one).



Figure 6: Google Maps - Online Map Service (taken from Maps.google.com)

➤ Lab

- ✓ The student trainee implemented the requirements from the result of library and field.

➤ Showroom

- ✓ The new prototype built is compared with the current application and it showed a lot of improvements on the layout.

➤ Workshop

- ✓ The student trainee then built and designed the prototype based on the result of showroom.

## 2. WPF or windows form application?

➤ Library

- ✓ The student trainee did the research of the company documentation.
- ✓ The student trainee searched and studied why WPF is better than windows form application from the internet. The student trainee found that WPF can give a better layout for the application because (found on website : [www.wpf-tutorial.com](http://www.wpf-tutorial.com)) [6]:
  - WPF is newer, therefore it can fulfill current layout requirements better.
  - WPF is flexible, users are able to make their own controls.
  - WPF use XAML, therefore it is easier to create and edit the layout.

- WPF user interface can be used for windows and extensible for web applications.
- WPF is used by the Microsoft to create the new application like Visual Studio.

➤ Field

- ✓ The student trainee interviewed the software developers of Vos Logistics to find out what are their system working on. The software developers of Vos Logistics make use of WPF mostly to revitalize the Lovos program.

➤ Lab

- ✓ The student trainee tried to implement in both forms and it results that the WPF form is easier to create and gives a better style for the layout.

➤ Showroom

- ✓ Comparing the WPF and windows forms, the student trainee chose to use WPF because of its advantages.

➤ Workshop

- ✓ The student trainee implemented the new route planning software in WPF.

### 3. How to find the functionalities and features provided by PTV?

➤ Library

- ✓ The student trainee studied the PTV manual.

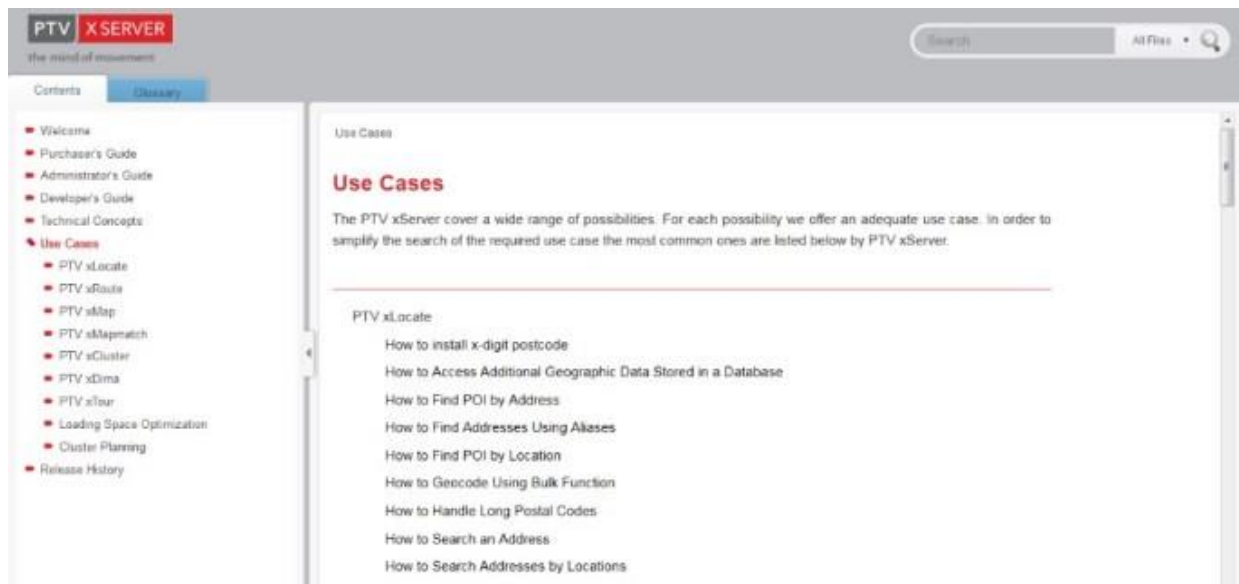


Figure 7: PTV Manual

- ✓ Tried and studied the demo version from PTV
- ✓ Asked the PTV support for help

➤ Field

- ✓ The student trainee asked the software developer what kind of functionalities that PTV provide.
4. How to integrate the new application (Lovos Maps) with the Lovos application? Is it better to merge the program or to call the Lovos Maps application separately?
- Library
    - ✓ The student trainee searched on the internet ways to call an application from C++ code.
    - ✓ The student trainee searched the best way how to integrate, only call or merge the program.
  - Field
    - ✓ The student trainee studied the Lovos system to understand where to implement the call for the new Lovos Maps.
    - ✓ The student trainee asked the software developer of the company about how the Lovos application is integrated.

The student trainee decided to call Lovos Maps from Lovos application because of 2 reasons:

- Considering the time that will be used for merging the program will take too much time.
  - By calling Lovos Maps from the Lovos application it will create a new process therefore it can make the work faster.
5. Which functionalities does the Lovos Maps need to have (to support the work of Vos Logistics' employees)?
- Library
    - ✓ The student trainee studied the popular online route planning software, which functionalities they have.
    - ✓ The student trainee studied the demo version from PTV and tried to find out which functionalities can be used for the application.
  - Field
    - ✓ The student trainee created a questionnaire and collected the requirements.
    - ✓ Asked the user representatives during the feedback sessions and collect the requirements.
    - ✓ Asked to the client and collect the requirements.
    - ✓ Studied the current application.
  - Workshop
    - ✓ The student trainee implemented the requirements collected in library and field.

## 6. How to make the most efficient route calculation for international transport?

- Library
  - ✓ Read and studied the manual of PTV to find the best solution for creating the most efficient route possible.
- Field
  - ✓ Asked the client about the experience with the current route planning software for the efficiency of the route
    - The client explained about the existing problem with the route efficiency. The route planning software need to have better route filters (routing with truck dimensions, accuracy of the map locations, alternative routes, and block various types of roads).
- Lab
  - ✓ The student trainee created a prototype which implement the filters and functionalities to create the most efficient route possible.
- Showroom
  - ✓ The student trainee compared the prototype with the current route planning software. The result is that the new route planning software can calculate more efficient routes for the heavy vehicles with more filtering functions. For example: by avoiding the toll road the truck has to drive a little bit further, on the other hand they don't have to pay for the high toll cost, which results in less total cost.
- Workshop
  - ✓ The student trainee implemented the functionalities from the prototype in the new application.

## 7. POI (Point of Interest) data

- Field
  - ✓ Asked the software developer where to find the data of POI's shown in the current route planning software. These are special POI's for Vos Logistics' trucks, the data is saved in the database, and therefore the study of the application database is needed.
- Lab

Remarks: after reviewing the database structure the student trainee could make a conclusion that the database is not relational and there are many of multiple same data in the database regarding the POI. When the student trainee tried to get the data from the database, millions of data are returned containing many of multiple same data. The student trainee made a new design of the database for POI with a relational structure, but the software developer suggested not to change the database structure.

- ✓ The student trainee made a prototype and took the data from the database by writing a query.

- Showroom
  - ✓ The student trainee checked if the results of the query did show the same POI data with the POI data from the current route planning software. The result is correct.
- Workshop
  - ✓ The student trainee implemented the POI as shown in the prototype.

8. What is the best coordinate format to be used for the maps?

- Library
  - ✓ The student trainee searched and studied many different types of coordinates from the internet.
  - ✓ The student trainee searched on the internet what coordinates format is mostly used by online map services.
  - ✓ The student trainee searched and studied from PTV manual, which coordinate types they support. PTV supports 3 types of coordinates that can be used (Mercator, Geominsec, and Geodecimal).
- Field
  - ✓ The student trainee asked the software developers what type of coordinate is being used. The current route planning software makes use of Geominsec type.
- Lab
  - ✓ After reading the resources the student trainee chose a coordinate type Mercator to be used in the new application. The reasons for using this type were:
    1. The Mercator format is being used by most of the online map service like Google Maps, Yahoo Maps, Openstreetmap, Bing Maps, and many more.
    2. The Mercator format has a perfect accuracy for location near the equator line.
    3. Universal Transverse Mercator, a coordinate format that is being used by the military is based on the Mercator format.
- Workshop
  - ✓ The student trainee implemented the new Lovos Maps using the Mercator format.

9. Which IDE is better to use, Visual Studio 2010 or 2012?

- Library
  - ✓ The student trainee searched and studied the differences between Visual Studio 2010 and 2012.
- Field
  - ✓ The software developers of the Vos Logistics make use of Visual Studio 2010.
- Lab

- ✓ The student trainee gained experience with both of the IDE during the study period and the student trainee chose to use Visual Studio 2012. The reason why the student trainee chose Visual Studio 2012 are:
  1. The IntelliSense (code completion feature which can speed up the coding) in Visual Studio 2012 is much faster and better which can help the developer to develop a program faster.
  2. Visual Studio 2012 can build and run the program faster than 2010.
  3. The new version of .Net framework (4.5) is being used in Visual Studio 2012.
  4. Visual Studio 2012 has integrated NuGet packet support which makes it easier for the developer to install third-party library.
  5. Visual Studio 2012 has a more user friendly interface than Visual Studio 2010.
  6. A program that is created in Visual Studio 2012 can also being opened and edited in Visual Studio 2010.

➤ Workshop

- ✓ As a result, the student trainee chose Visual Studio 2012 because it has more advance features to be used to create a modern program.

### 4.3. Implementation

After the research was accomplished, the student trainee implemented the software based on the research result.

The main functionalities implemented are:

1. Displaying a modern style, responsive and interactive maps.
  - After the student trainee has done the research for the layout, the student trainee made a list with all the requirements for the layout. The requirements on the list are the combination of the layout requirements from the client, feedback sessions and research result.
  - The requirements for the layout are listed as below:
    - ✓ The user should be able to zoom using the mouse scroll wheel or using double click on the map.
    - ✓ The user should be able to move the map by dragging the map screen.
    - ✓ The size of the map should be bigger (Google Maps standard).

- ✓ The user should be able to use shortcut keys to show help (F1), show menu (F12) and set full screen (F11).
  - ✓ The search for location field should be simple (single field).
  - ✓ There should not be too many buttons on the layout.
  - ✓ The menu can be hidden and expanded.
  - ✓ The user should be able to locate a place on the map using a single mouse click.
  - The student trainee has been successfully implemented the requirement for the layout.
2. Searching for places by keyword.
- The idea of this requirement is to search for addresses when the user only uses the three first characters of the address as input. For example: if the user type 'eind' then a list of suggestions will be shown, like 'Eindhoven', 'Eind', etc., then the user can choose the location by selecting the name in the list.
  - The student trainee searched for solutions available on the internet how to implement a field in WPF which can give a suggestion when the user searches for places.
  - The student trainee decided to use auto complete box by installing an extension third party library for Visual Studio. Then data binding is used to show the suggestion.
  - The searching places by keyword is implemented successfully.
3. Calculating routes between two or more addresses.
- The student trainee has studied the PTV manual and PTV demo version during the research phase. Therefrom the student trainee has knowledge about how to make a route calculation between two addresses.
  - After the route calculation between two addresses was completed, the student trainee implemented route calculation for multiple addresses.
4. Estimating travel time.
- The student trainee has studied the PTV manual during the research phase. Therefore the student trainee has knowledge about what kind of information is retrieved from the PTV server when a route is calculated. The information retrieved from the PTV server contains the details of the route calculated including the travel time.



- The client wants to see the arrival time according to the selected departure time. The student trainee implemented a date and time picker which enables the user to change the departure date and time easily.
  - Afterwards, showed the details of the route calculated including the travel time and arrival time in the display according to the date and time selected by the user.
5. Filtering route calculation with blocking toll roads.
- After the student trainee studied the PTV manual, student trainee found the way to set the filter before the route is being calculated.
  - In the display a checkbox is added to enable the user to activate the filter or not.
  - The student trainee implemented the filter successfully.
6. Filtering route calculation with blocking the vignette road.
- This filter has the similar case as the blocking toll road filter. The student trainee has implemented the filter at the beginning of the development.
- Note: in some countries the vignette road tax is not applicable for heavy truck vehicles with a maximum gross weight more than 3.5 tons. For heavy vehicle with maximum gross weight more than 3.5 tons toll is applied and the price depends on the mileage.
7. Filtering route calculation with blocking highways.
- This filter has the similar case as the blocking toll road filter and the student trainee has implemented the filter successfully.
8. Filtering route calculation with blocking the ferry routes.
- This filter has the similar case as the blocking toll road filter and the student trainee has implemented the filter successfully.
9. Finding alternative routes (shortest, optimal, and fastest route).
- After studying the PTV manual, student trainee found the way to set the optimization during the route is being calculated. The optimization can be set as the shortest route has the least distance, the fastest route has the least time, and the optimal route is the optimum between distance and time.

- The student trainee added radio buttons to let the user select which route optimization they want, by default it is the optimal route.
10. Calculating routes depending on vehicle characteristics, such as height, weight and the EURO norm of the vehicle.
- The idea of this requirement is that the user can avoid obstacles like tunnels in the route if the dimension of the vehicle not fulfilled the tunnel's specification.
  - The student trainee gathered the knowledge by studying the manual, asking the PTV support, and asking the software developers of the company about how to calculate a route with vehicle characteristics.
  - Because there are a lot of vehicle types that Vos Logistics owns, the student trainee set the default value for the vehicle dimension. But the user can also customize the dimension of the vehicle by filling in the dimension.
11. Ability to view POI (point of interest) (for example a truck cleaning station, safe parking place or preferred gas station) on the maps, then add them to the route
- The student trainee has done research before and based on the research result the student trainee implemented the functionality for showing the POI.
  - The user can show the POI by ticking the checkbox in the menu, then by right click on the POI the user can add the POI to the route as via point.

The extra functionalities implemented are:

1. Displaying the details of the route.
  - The idea is that the user can see the direction guide / description.
  - After studying the PTV manual the student trainee found the way to retrieve the route details and show them in the map display.
2. Calculating toll costs on the route.
  - After the student trainee studied the PTV manual, the student trainee found the way to calculate the cost of the toll.
  - The student trainee then implemented the toll cost calculation and display the cost in the route details. Vos Logistics received updates for toll tariff every six months. The user can also calculate toll cost in the future, if the new tariff is available.

3. Displaying the address details.
  - The idea is that the user can see the details of the specific address such as the coordinates of the address using different types of coordinates.
  - This requirement is completed successfully, the user can see the coordinates of the address in the format Mercator, geominsec, and geodecimal.
4. Integrating with Lovos.
  - The current route planning software is integrated in too many places in the Lovos system. One of them is on the homepage of Lovos, a button where the current route planning software is called. Because the lack of time the student trainee couldn't integrate the new Lovos Maps into every place.
  - The student trainee succeeded to implement the new Lovos Maps on the homepage button where the current route planning software is called.
5. Locating with coordinates.
  - The idea is when the user (Vos Logistics' employee) get only the coordinates of the address from the customer, they don't have to convert it first using an online map service. The user can input the coordinate (geominsec format) to the search field to locate the address.
  - This requirement is part of the feedback and completed successfully by the student trainee.
6. Showing toll part(s) on the route with another color.
  - As part of the feedback, it is nice if the user can see where there is toll in the route calculated.
  - Because this is not a standard feature from PTV, the student trainee has to ask the PTV support about the toll information returned from the PTV server. The PTV support answered that it is possible by combining some information returned. The student trainee tried to find a way to combine the information to get the toll segments.
  - After some trials the student trainee succeeded to show a different color where there is toll in the route.
7. Locating with postal codes.
  - As part of the feedback this requirement arose. This functionality need to be implemented because some departments in Vos Logistics make a route calculation only by using postal codes.

- This requirement is implemented successfully.

The extra functionalities which not implemented are:

1. Calculating emission depending on vehicle type.
  - Vos logistics has many types of trucks with many different brands and different vehicle specifications for each brand. Every type of vehicle can have different euro norms, dimensions and fuel usage which can affect the emission calculation.
  - There is no total list of vehicles available in the company, therefore it is impossible to use the specific vehicle characteristics. If the calculation is using the average vehicle characteristics then it will not be accurate, therefore there is no necessity to do it.
  - To use the emission information from PTV, a license need to be purchased with additional costs (exceeding the development budget). For this reason it is impossible for the student trainee to implement this functionality.
2. Displaying satellite view.
  - Satellite view is not a primary function needed. There is a need to purchase a license to make use of it which not provided by the company. Therefore the student is not able to implement it.
3. Displaying live traffic information.
  - Live traffic is not a primary function needed. There is a need to purchase a license to make use of it which is not provided by the company. Therefore the student trainee is not able to implement this functionality.
4. Changing routes with drag and drop.
  - To implement the drag and drop functionality, there is a need to change a lot of codes, but there is not enough time left. That is why it is not implemented.
5. Displaying the use of gas and cost of it.
  - This case is similar to emission calculation problem, to be able to calculate the accurate use of gas the specification of the vehicle should be collected. But there are too many types of vehicles that Vos Logistics owns, therefore it is impossible.

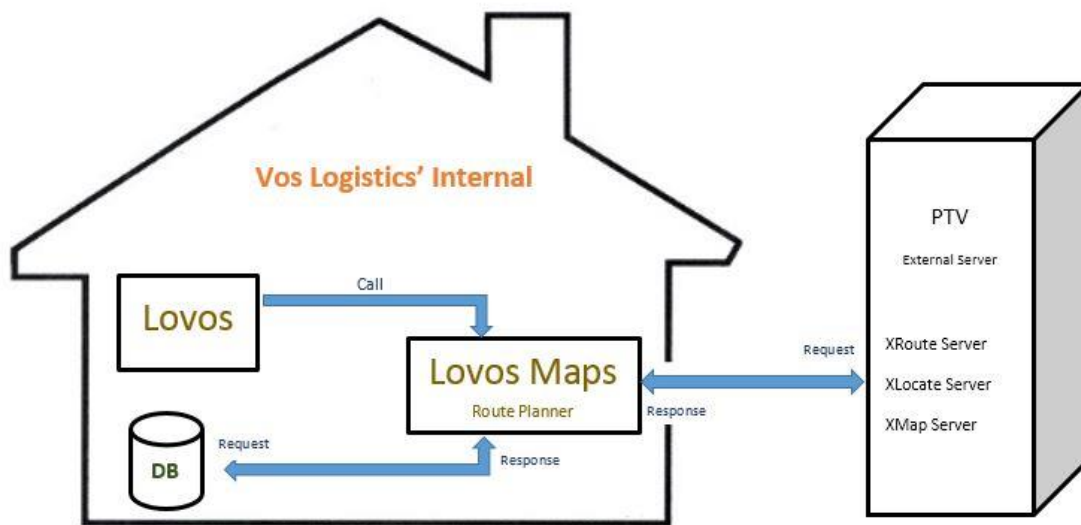


Figure 8: Lovos Maps Diagram

The diagram above explains how the new Lovos Maps and Lovos system are integrated, how Lovos Maps get data from the PTV (external server) and connection to the Lovos database. The class diagram can be found in the attachment V and database diagram in the attachment VI.

#### 4.4. Prototyping and Feedback

During the first period a working prototype of the software was made almost every week. The client gave feedback and discussed the prototype with the student trainee whether the requirements were correctly implemented or not.

By the end of the first period, a prototype and demo session was held with the Vos Logistics' employees. This demo session for the first period was also being held during the company bimonthly IT meeting. The feedback was held with the client (Arno Rouwenhorst), software developers (Wim Overmars and Marco Stolk), information analyst (Eliza), Lovos application manager (Mark Uittenboger) and user representatives (Daan Janssen, Sybren de Kruijk and Michal Sikorski).

The feedback that the student trainee received was positive. The layout has improved a lot compared to the current route planning software and the style is modern. There are some bugs and errors found during the sessions, but already fixed by the student trainee.

New functionalities or requirements from the feedback sessions are discussed with the client to decide whether it will be added as an extra requirement or not as explained in the implementation before.

#### 4.5. Requirement Change

Vos Logistics has a Lovos change board, during the meeting they will decide whether the change will be made to Lovos application or not.

After the feedback and demo session are held, the Lovos change board decided not to implement the routing with vehicle characteristics because of different opinions. The functionality routing with vehicle dimensions is deleted from the application.

#### 4.6. Testing

There were four types of tests that were done by the student trainee. These are performance, user acceptance, interface, and integration (unit) test.

1. Integration test is to test the integration between functions combined with several elements work correctly or not.
2. User interface test is done by assigning the user a certain task to see if the interface is clear enough and the software operates as supposed.
3. Performance test of the software by examining the time while performing certain tasks.
4. User acceptance test is done by checking if the requirements from the customer are met and accepted by the customer.

All the tests ran successfully and have positive results.

The test result report (see attachment IV) can be found in the attachments.

#### 4.7. Final Product

By the end of the graduation internship period, the product is delivered by the student trainee to the company including the user manual. A presentation for the software developer is given by the student trainee.

The screenshots of the product with the explanation can be found in the attachment II.

## 5. Conclusion & Recommendation

Lovos Maps is an important piece of software for Vos Logistics. When an order is created a route should be calculated to be able to produce the invoice. The current route planning software, which was created around 10 years ago, cannot optimally support 800 shipments daily made by the planning and fleet management departments of Vos Logistics. The problems of the current route planning software are that the user interface is not user friendly, many new functionalities provided by PTV are not being used and functionalities implemented are not working properly.

The goal of this graduation project is to create a new route planning software which can solve the problems described above. The development of the project makes use of prototyping methodology. The initiation phase was the start of the project, where the student trainee gathered the problems and requirements of the application. The next step was the research, where the student trainee gathered the knowledge about the possible solutions of the problems. After the research, the student then implement the software based on the research results.

Afterwards, the student trainee gave the demo of the prototype and held a few feedback sessions with the client, user representatives, software developers and other Lovos Maps users. The project is finalized with the testing phase. During the feedback sessions, a positive result was given to the student trainee. The application testing ran successfully and smoothly.

The aforementioned problems are solved through this new Lovos Maps application. The user interface got a modern and new look, which is more user friendly regarding the user interface testing. More features and new functionalities are added such as:

- Responsive and interactive map display.
- Searching for places by keyword feature.
- Route of multiple destination can be calculated easily.
- New filters (block highways and block ferry).
- Alternative routes (shortest, optimal and fastest).
- Easily view and add POI to the route.
- Locate a place using coordinates.
- Toll part of the route shown in another color.

All the main requirements are successfully completed by the student trainee. 60% of the extra requirements are completed successfully. The other 40% of the extra requirements could not be completed by the student trainee because of lacking components (more licenses need to be purchased).

For the future development and research, some recommendations can be:

- Apply the integration in other pages in the Lovos system.
- Add intermodal journey features.
- If more licenses will be purchased the other functionalities can be implemented, such as: live traffic information, satellite view and emission details based on the vehicle characteristics.



## Evaluation

In my opinion, by doing an internship, I can understand and get more experiences about how work is done in the real world. By applying the theory in practice I can develop the knowledge better than only studying the theory. In fact, it is difficult to apply a theory in practice.

At the beginning of this graduation internship I was confused which development methodology I had to use. At school I learnt that scrum agile is the best method to be used, but in practice scrum agile is very difficult to be applied correctly. Therefore the university tutor advised me to use the prototype methodology. Then I did some research on the methodology and I found out that it is a good methodology for my project. Next to that, the prototype methodology is a good methodology to be practiced too. The necessary requirements can be added along the development to improve the application. By using prototype methodology there is a working prototype where the user can try the application for real. Therefore the user can give the feel of using the application and the feedback about the application.

During this graduation internship I learnt how to work with new methodologies other than methodologies that I learnt at school, not only for development methodologies (prototype methodology) but also for research methodologies (DOT framework). In this graduation internship I worked more independently than in my 3<sup>rd</sup> year internship, where I was told what I had to make, this time I had to find what I had to do by myself. I think it is interesting to search what kind of functionalities that can be added to the application to make it more advanced.

To collect the requirements from the users I made a questionnaire, because Vos Logistics offices are spread all over Europe. It is the most efficient way to collect the requirements from the users. The problem is that not every user respond to the questionnaire immediately, therefore the requirements could not be collected completely at the beginning.

This project went quite smoothly as the project plan was followed in order and research was done carefully before implementing a feature. The only difficulty is only when I need to collect feedback from the user or employees it took a lot of time and not everyone wanted to participate. In a next project, I will have to do the documentation better. In this project I did make a log only for what kind of features I implemented, but not what I researched.

Four years of study in the Netherlands are over, time passed by really fast. Now it is time for me to jump into the real world as a software developer. In fact, what I learnt in four years is just the start for me to learn more and dive deep into the IT world. Nowadays IT world is really big, there are so many branches and languages that you can learn. It makes you feel like you just get started on the first page of a dictionary.

## Reference

1. Wikipedia. (2015, April 17). Definition of C++. Retrieved from <http://en.wikipedia.org/wiki/C%2B%2B>
2. Wikipedia. (2015, June 09). C Sharp (programming language). Retrieved from [http://en.wikipedia.org/wiki/C\\_Sharp\\_%28programming\\_language%29](http://en.wikipedia.org/wiki/C_Sharp_%28programming_language%29)
3. PTV. (n.d.). About company. Retrieved from <http://company.ptvgroup.com/en/ptv-group/>
4. Microsoft. (2015). WPF. Retrieved April 20, 2015, from <https://msdn.microsoft.com/en-us/library/ms754130%28v=vs.110%29.aspx>
5. Vos Logistics. (2015). About Vos Logistics. Retrieved from <http://www.voslogistics.com/about-vos-logistics/profile>
6. WPF vs. WinForms. (n.d.). Retrieved from <http://www.wpf-tutorial.com/about-wpf/wpf-vs-winforms>
7. What is Prototype model- advantages, disadvantages and when to use it? (n.d.). Retrieved from <http://istqbexamcertification.com/what-is-prototype-model-advantages-disadvantages-and-when-to-use-it>
8. Rouse, M. (n.d.). Prototyping Model definition. Retrieved from <http://searchcio.techtarget.com/definition/Prototyping-Model>
9. Turnhout, K. van, Zwart, J. P., Craenmehr, S., Holwerda, R., Bakker, R. R., & Menijn, M. (2014, October 21). DOT-Framework Poster. Retrieved from [http://www.researchgate.net/publication/267153777\\_DOT-Framework\\_Poster](http://www.researchgate.net/publication/267153777_DOT-Framework_Poster)

## Attachments

- I. Project Plan



# Project Plan

---

VERSION 1.3

Dani Ardi Kusuma

VOS LOGISTICS | VORSTENGRAFDonk 39, OSS



University of Applied Sciences

PROJECT PLAN


FONTYS UNIVERSITY OF APPLIED SCIENCES

HBO-ICT: English Stream

<b>Data student:</b>	
Family name , initials:	Kusuma, D.A.
Student number:	2184973
project period: (from – till)	February 9 <sup>th</sup> 2015– June 19 <sup>th</sup> 2015
<b>Data company:</b>	
Name company/institution:	Vos Logistics
Department:	IT Applications Management
Address:	Vorstengrafdonk 39, Oss, the Netherlands
<b>Company tutor:</b>	
Family name, initials:	Arno Rouwenhorst
Position:	Application Department Team lead
<b>University tutor:</b>	
Family name , initials:	Henning, Frank
<b>Final report:</b>	
Title:	Lovos Maps & Route Calculation
Date:	June 19 <sup>th</sup> 2015


Approved and signed by the company tutor:

Date:  
Signature:

March 13, 2015  



Approved and signed by the university tutor:

Date:  
Signature:

13-3-2015  


Agreed and signed by the student:

Date:  
Signature:

March 13, 2015  


## Contents

1. Introduction .....	26
2. Project Statement .....	30
2.1. Formal Client .....	30
2.2. Current Situation .....	30
2.3. Project Justification .....	31
2.4. Project Product .....	31
2.5. Project Deliverables and Non-deliverables .....	32
2.6. Project Constraints .....	32
2.7. Project Risks .....	33
2.8. Research Component .....	33
2.9. Research Question .....	33
3. Methodology & Phasing .....	34
3.1. Software Development Methodology & Phasing .....	34
3.1.1. Initiation and Planning .....	34
3.1.2. Design & Development .....	34
3.1.2.1. First period .....	34
3.1.2.2. Second period .....	35
3.1.3. Testing .....	35
3.1.4. Closure .....	35
3.2. Research Methodology & Phasing .....	35
4. Management plan .....	37
4.1. Money .....	37
4.2. Skills .....	37
4.3. Quality .....	38
4.4. Time .....	38
4.5. Organization .....	39

## 1. Introduction

Imagine route calculation with 'Google Maps' it's nice, isn't it? Interactive maps and efficient route calculation with large varieties of filters are ones of the important features for a logistics company. These maps allows the user to make a calculation of the most efficient and safe route for the driver.

One of the logistics companies in the Netherlands that have a transport management system with route calculation implemented is Vos logistics, with their head office located in Oss, the Netherlands. They have 1200 trucks, 3000 trailers, and 1900 employees servicing around 30 countries in Europe.



*Figure 9: Vos Logistics Trucks*

Vos Logistics, established in 1944, is an independent European logistics service provider, delivering a wide range of transport and customer- specific logistics services. The company has four core activities: transport of cargo (packed goods) and bulk (dry unpacked goods both granulates and powders), logistics services (forwarding, warehousing and value added services and distribution to full supply chain), and fleet service.

The transport management system used in the transport divisions of Vos logistics called 'Lovos', developed from 1994 onwards, is still being used by this company with 88 released versions. Lovos is a wide range software package used for managing shipments, planning, invoicing, and route calculation. There are about 770 transport orders handled every day, equaling 200.000 shipments per year (2014). For every shipment, the route calculation and map software is used.

The objective of this graduation internship is to design and built a new application of interactive maps with more functionality and filters for truck vehicle route calculation. The software will be integrated in the existing Lovos transport management system. Another requirement is to modernize the user interface, which will be implemented after extensive research. The basic functionalities are the ability of calculating routes and the view of route from different points.

The following chapters describe the project details, methodology and phasing, as well as the management plan.

## 2. Project Statement

### 2.1. Formal Client

The formal client of this project is Paul van Ham, the IT manager at Vos Logistics. The company tutor and sponsor for this graduation assignment is Arno Rouwenhorst, the Application Department Team Lead at Vos Logistics.

### 2.2. Current Situation

In a logistics company, the availability of a good map for efficient route calculation is inevitable. The current (existing) software, implemented approximately 10 years ago is not supporting the recent conditions, like alternative routes, route calculation per vehicle type (height, weight), drag and drop route, satellite view, zoom ability and many more. The employees of Vos Logistics cannot make use of available online maps, like 'Google maps', because it is not purposed for decent truck route calculation.

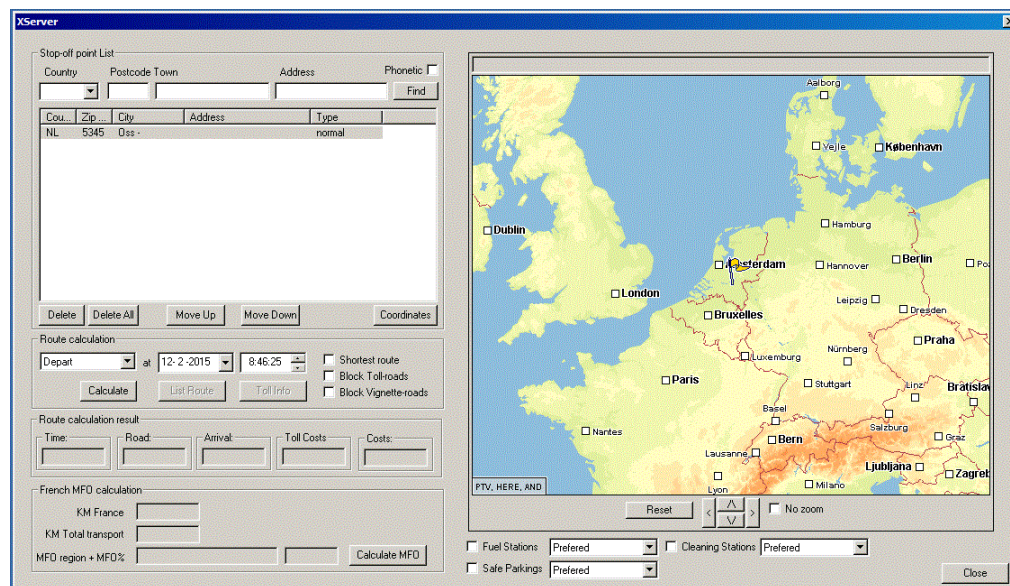


Figure 10: Current Lovos Maps

The current maps software is written in C++ which gives a less opportunities to use all the features offered by the maps supplier called 'PTV'. PTV supports in house development maps for Java and C# for their products. They also provide a toolbox for WPF application, a demo version for .NET application and a user manual. The current Lovos Maps program is written in an unstructured pattern and therefore it is difficult to read and restructure. As a result, it is decided to create a new Lovos Maps application that will be integrated in the Lovos system.



The 'look and feel' is very old (see Figure 2) and not user friendly (for example, users cannot drag the maps which is the standard of the available online maps nowadays), via routes are not optimal and some functionalities and filters are not working. The current outdated map software, with a lot of unused functionalities, does not support the planning and fleet management departments in managing hundreds of transport orders every day.

These days the software developers from Vos Logistics are busy with revitalizing the entire Lovos transport management system to C# environment step by step. In the future the Lovos program will be easier to maintain and develop. All old code, unused code or faulty code will be removed from the system and after the migration to C# it will be deleted from the system.

### 2.3. Project Justification

Vos logistics is standing at a point where it is crucial for them to update their route calculation (existing route planning software) feature in Lovos, their transport management system. The update is needed in order to be able to compete with the fast changes of markets and demands. Since the opening of the borders with for instance Poland and other old Eastern European countries, Vos Logistics needs to find new technique to grow their business and stay stable in the market nowadays.

In the Eastern European countries many small family companies can offer better prices for most of Vos Logistics' transport orders. New innovation is needed for larger companies like Vos Logistics. One of the innovations that can be added to Vos Logistics is an effective route calculation.

The outdated route calculation software and user interface needs a thorough update so it is capable for calculating the cost of the most effective route for transportation. Every extra euro earned can make the difference for the future. Other route calculators are getting more and more intelligent with not only route information, but also with prices of toll roads, roadblocks, heights of tunnels, actual traffic information and so on. These new intelligent feature can be used for calculating the cost of the most effective transport route where lots of unnecessary cost can be saved each trip.

Therefore, there is a need to recreate the program using WPF in C# which can be used to design a modern style of user interface and possibility to explore the advanced (intelligent) features provided by PTV.

### 2.4. Project Product

The final product of this project is represented by a new maps application consisting of a modern style and an interactive user interface with many different filter possibilities.

The main (must have in MOSCOW) functionalities to be implemented are:

1. Displaying modern style and interactive maps.
2. Searching for places by keyword.
3. Calculating routes between two or more addresses.
4. Estimating travel time.
5. Filtering route calculation with blocking the toll road or vignette road.
6. Finding alternative routes.
7. Calculating routes depending on vehicle characteristics, such as height, weight and EURO norm of the vehicle.
8. Ability to view so called VIA points (for example a washing point, safe parking spot or preferred gas station) on the maps then add them to the route

The extra (could have in MOSCOW) functionality that can be added when the main functionalities are implemented, are:

1. Changing routes with drag and drop.
2. Importing coordinates or addresses from excel files and calculate routes out of it.
3. Displaying the details of the route.
4. Displaying the use of gas and cost of it.
5. Displaying live traffic information.
6. Calculating toll costs on the route.
7. Converting addresses to coordinates and vice versa.
8. Calculating emission depending on vehicle type.
9. Displaying satellite view.

More features can be added by the client during the process.

## 2.5. Project Deliverables and Non-deliverables

Deliverables	Non-Deliverables
Application with main functionalities	Training
User requirements (User stories)	
User manual	Maintenance
Developer manual	
Project plan	
Project final report	

## 2.6. Project Constraints

1. The application should run on Windows 7 Operating System and above.
2. The application should be a WPF version 3.0 form.
3. The application should be programmed in C# version 5.0 released with .NET framework 4.5.

4. The application should run in a Citrix environment (client software that provides employees with universal access to virtual apps).

## 2.7. Project Risks

Risk	Solution
<b>Miscommunication</b>	<b>Weekly meetings are scheduled with the company mentor (Mr. Rouwenhorst) and a biweekly meeting with the software developer (Mr. Stolk)</b>
<b>Lack of technical knowledge (student trainee)</b>	<b>Research technical solutions. Ask the project guide, the developers in Vos Logistics or teacher at school</b>
<b>Lack of feedback from users (input)</b>	<b>Use the survey and provide a demo version to get feedback from users</b>
<b>Lack of support from PTV</b>	<b>Communicate and ask for support from PTV support center</b>
<b>Time planning for implementation</b>	<b>Use estimation based on the experience of the student trainee</b>
<b>The program has errors or bugs</b>	<b>Use the prototype to test the program to user representative and test the program after the first period</b>

## 2.8. Research Component

The research components are the information analysis regarding the new functionalities from PTV that can be implemented in Lovos Maps and the information analysis regarding the adjustment in the technical interface of Lovos Maps.

The result of the information analysis might be a list of requested functionalities. The functionalities that fit into the internship might be chosen to be implemented.

## 2.9. Research Question

1. What is the target audience of this application?
2. How to find the functionalities and features provided by PTV?
3. How to build a modern and user friendly user interface for Vos' employees?
4. How to integrate the new application (Lovos Maps) with the old application (Lovos system)?
5. Which functionalities that Lovos maps has to have (to support the work of Vos' employees)?
6. How to use the functionalities or feature provided by PTV?
7. How to run the application in Citrix environment?
8. How to make the most efficient route calculation for logistics?
9. How to solve the current problem with routing for big truck (e.g. the truck is too high to pass the tunnel)?

### 3. Methodology & Phasing

#### 3.1. Software Development Methodology & Phasing

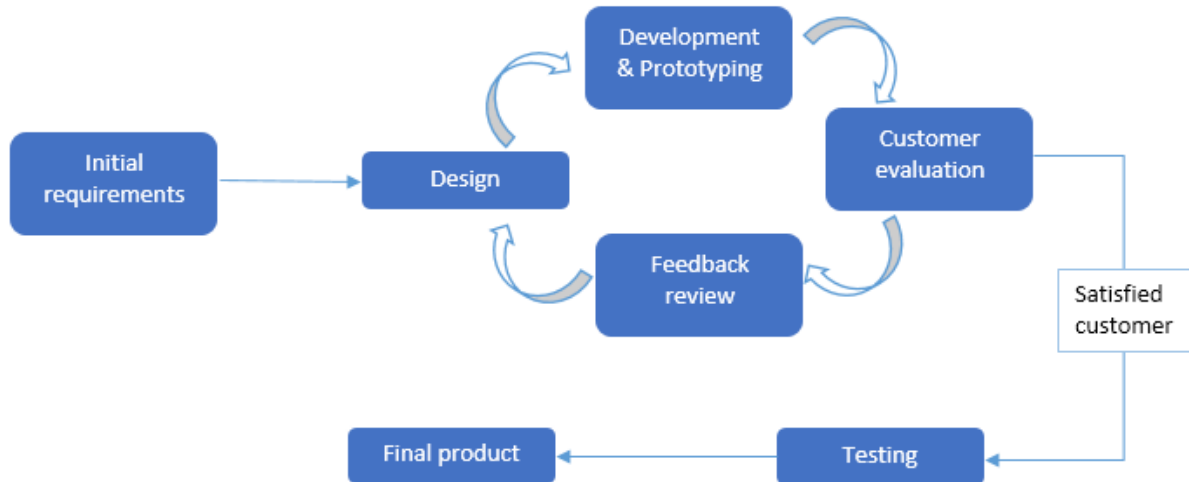


Figure 11: Prototype Model

In this project a prototyping methodology is used because this software has a lot of interaction with the users. The evaluation from the customer is important part of this project. When the users are involved in a project the feedback from the users can be used to adjust the user friendliness of the user interface.

##### 3.1.1. Initiation and Planning

In this phase, the kick-off of the project takes place. Therefore, the goals and scope of the project are described and divided into two levels (must have and could have as in Moscow). In order to accomplish that, research and analysis of the current information about the project is needed. In the initiation phase it is decided that there will be 50 working days in the first period and 25 working days for second period.

##### 3.1.2. Design & Development

###### 3.1.2.1. First period

During this period the main feature of the product will be designed and implemented. After this period is finished, the prototype or demo version will be tested by the users. If necessary, the user friendliness of the maps can be adjusted, the bugs and error from the prototype test will be resolved.

### 3.1.2.2. Second period

In the second period. The extra features will be designed and implemented depending on the remaining time. The prototype will be made and reviewed by the user. Then if necessary changes will be made according to the review from the users.

### 3.1.3. Testing

The software will be tested by performance, user acceptance and a unit test. The testing will be done both for first period product and second period product. The result of this phase is the final product.

### 3.1.4. Closure

In this phase the final project report will be completed and a presentation of the results will be given.

## 3.2. Research Methodology & Phasing

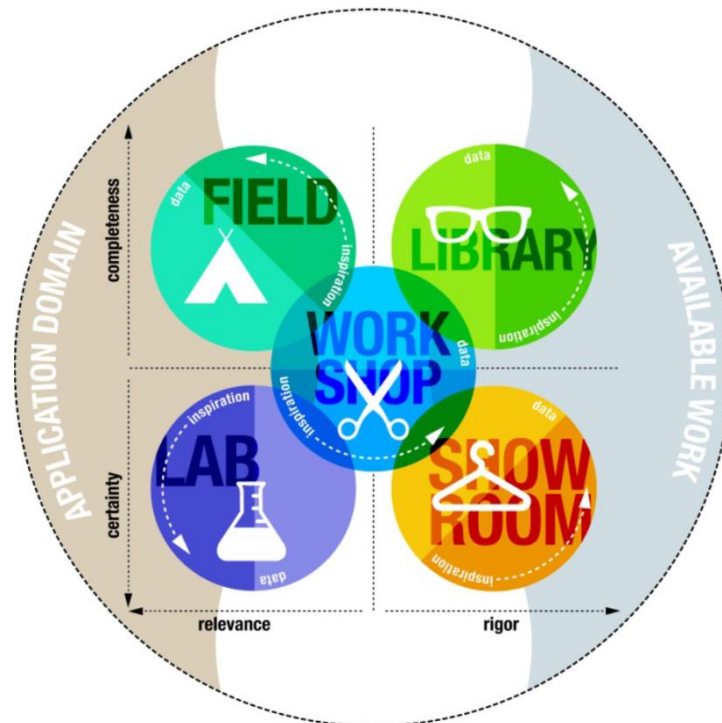


Figure 12: Research Framework

In this project, a new research framework called 'dot' framework will be used. The five strategies which will be repeated during each period as explain before consist of field, library, workshop, lab, and showroom.

- Library → check the internet, books, articles and other source, how other logistics companies use map software and which functionalities. Check the products from PTV. What functionality would be useful and needed for immediate and in the future.
- Field → Reflect the library search items with your end-users. Use interviews and survey to find out what do Vos Logistics' employees need. After wrapping up the requirements, discuss them with the product owner, if they are satisfied with the results for implementation.
- Workshop → Design and build the map software using the prototyping methodology as explained before. In this area, software will be designed, code will be written, and a test plan will be made.
- Lab → Test your solution build with the use of the predefined test plan. Use different test approached: user acceptance, performance, and integration test.
- Showroom → demonstrate the program to all end users and management.

## 4. Management plan

### 4.3. Money

Description	Amount
<b>Student trainee salary</b>	+/- €1100 (5 months x €225)
<b>Student travel expenses allowance</b>	+/- €900 (90 days x €10 /day)
<b>IT team introduction day</b>	+/- €1200 (3 days x 8 hours x €50 /hour)
<b>Company mentor (weekly meeting)</b>	+/- €900 (18 days x 1 hour x €50 /hour)
<b>Company software developer (once every 2 weeks meeting)</b>	+/- €450 (9 days x 1 hour x €50 /hour)
<b>IT team implementation internal server</b>	+/- €2000 (5 days x 8 hours x €50 /hour)
<b>IDE (Visual Studio)</b>	€0
<b>PTV maps server</b>	(provided by the company)
<b>Total</b>	<b>+/- €6550</b>

### 4.4. Skills

#### 1. Analytical

- The student trainee should be able to analyze the requirements for the project to determine the client's needs. When problems occurs, the student trainee should be able to analyze them to find the root cause and to come up with a solution.

#### 2. Design

- The student trainee needs to design a user-friendly interface and a sustainable code (robust and maintainable).

#### 3. Planning

- The student trainee should be able to plan clearly and ahead of time in order to accomplish all the goals set for the project, either final or intermediate, and in order to handle or avoid any problems that can be foreseen.

#### 4. Programming

- In order to build the application, a set of programming skills is needed: C#, WPF and maps service server.

#### 5. Reporting

- At the end of the project, the student trainee should be able to write a clear final report on everything that has been done to accomplish the goal.

#### 6. Communication

- Throughout the project, clear communication is needed between the student trainee and the client. The progress of the project should be clearly communicated to the company mentor.

#### 7. Project management

- Project management skills are required to ensure that the project will stay on track.

### 4.5. Quality

The key performance indicators (KPI) are:

1. On time delivery.
2. Meeting requirements given by the client and the result of the research.
3. Performance of the application (see definition below).

Before each of the period, a thorough planning is made, based on the requirements from the client. After the period, an evaluation session takes place to check the project progress.

Careful testing will also be held after each period. This will be executed according to the test plan made upfront. Performance of the application (for example: how many clients can Lovos Maps handle at the same time), the user acceptance and the system (unit test) will be tested here. The final testing is held at the end of the project, in order to ensure the quality of the application.

### 4.6. Time

Activities	Start Date	End Date	Number of days
Initiation, planning, gathering general requirements	9-Feb-2015	11-Feb-2015	3
First period	12-Feb-2015	26-Apr-2015	50
Second period	27-Apr -2015	5-Jun -2015	25
Finalizing the project	6-Jun-2015	19-Jun-2015	12



## 4.7. Organization

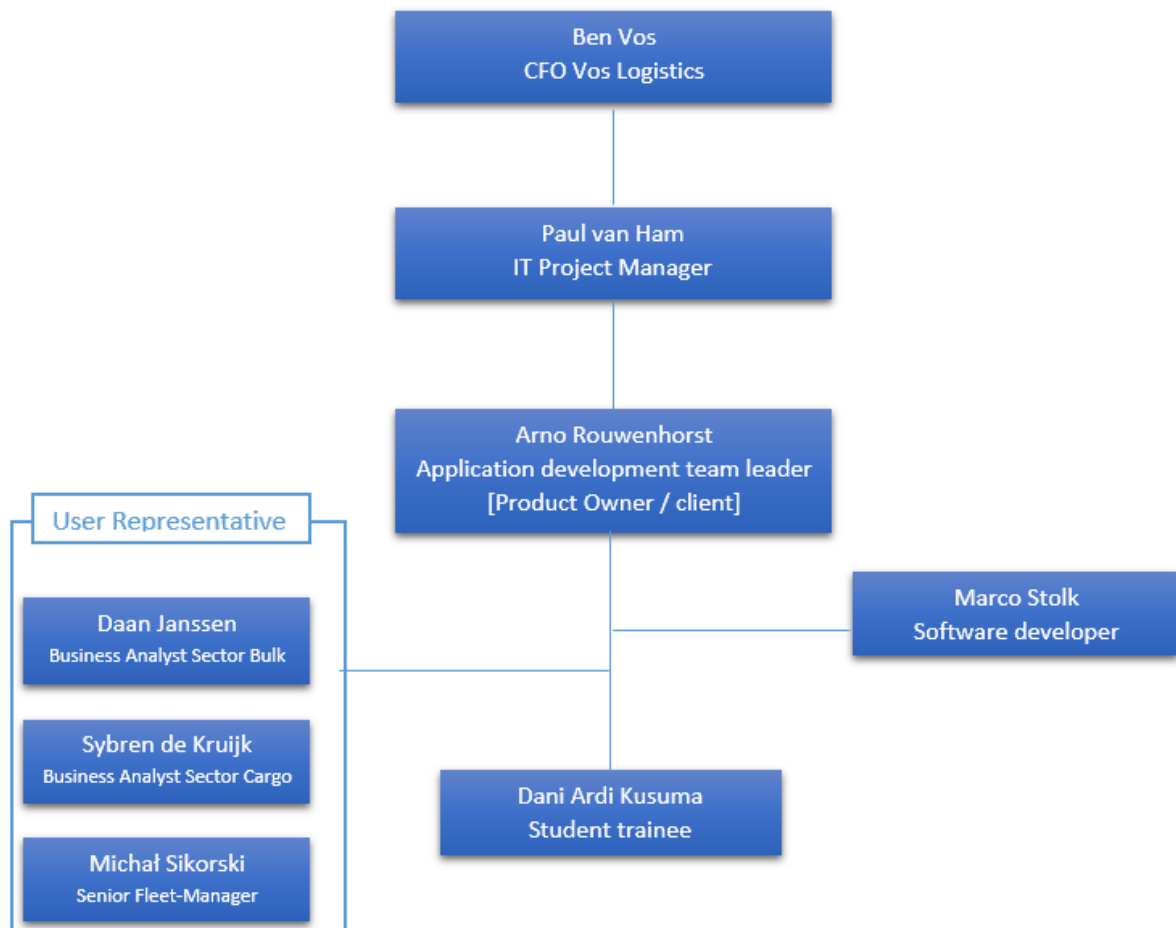


Figure 13: Organization structure

During this project a weekly meeting with the product owner will be held and a biweekly meeting with the software developer (see figure 5). After the first period a product prototype can be tested by the user representatives (group of people which often make use of Lovos maps).

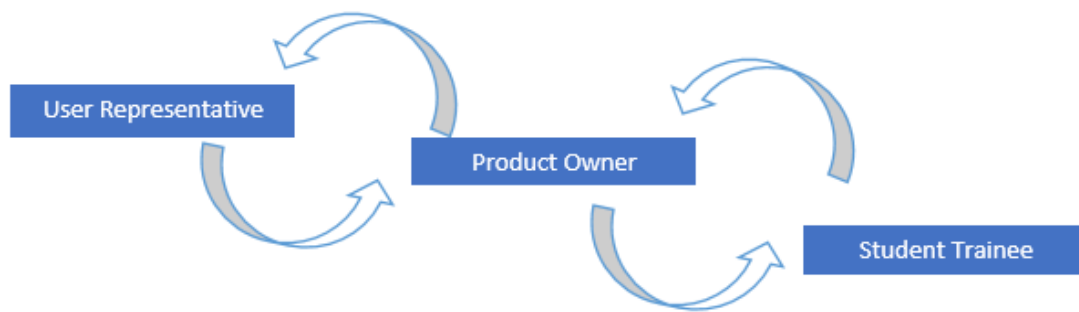
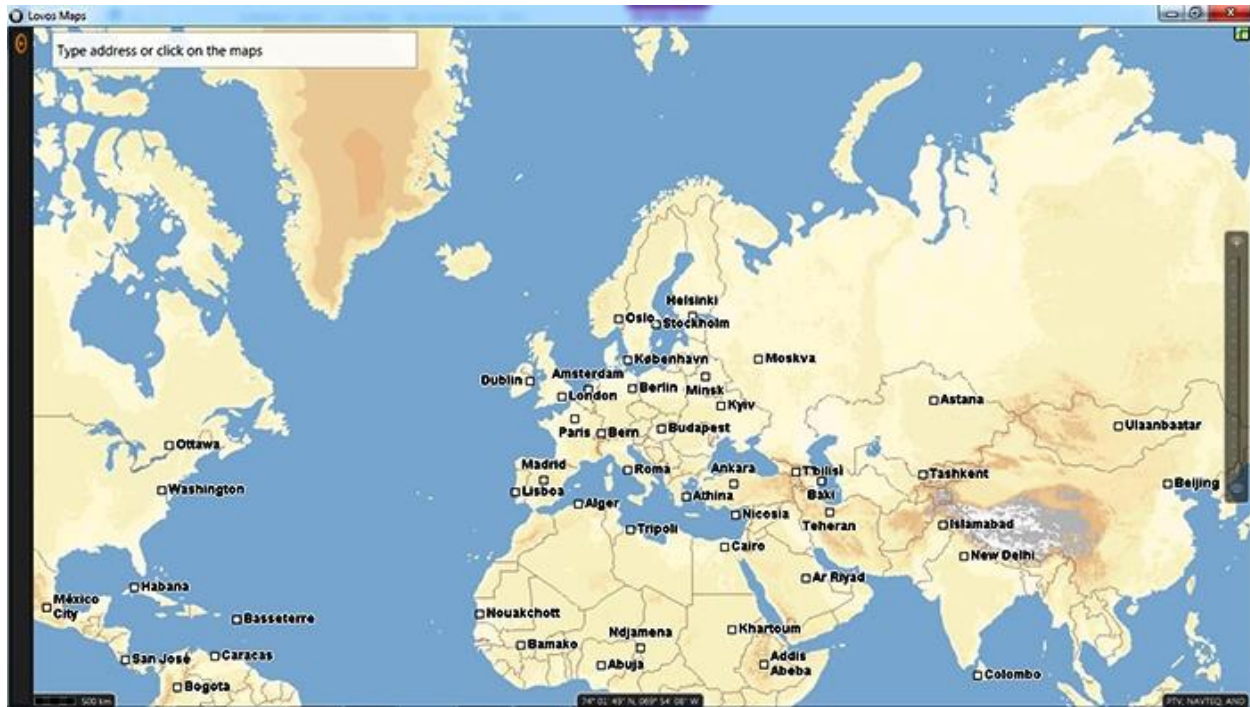


Figure 14: Change Procedure

Every change request from the user representatives will be validated and analyzed by the product owner. Student trainee and product owner will make an impact calculation of the time, costs and effects on the software. Based on this calculation, the product owner and Mr. Rouwenhorst will decide to implement the change or not. If it is decided to implement the change, the student trainee will respond on the request by executing the change and discuss the change with product owner. The product owner will close the request and update the change made with the user representatives (see figure 6).

## II. Screenshots

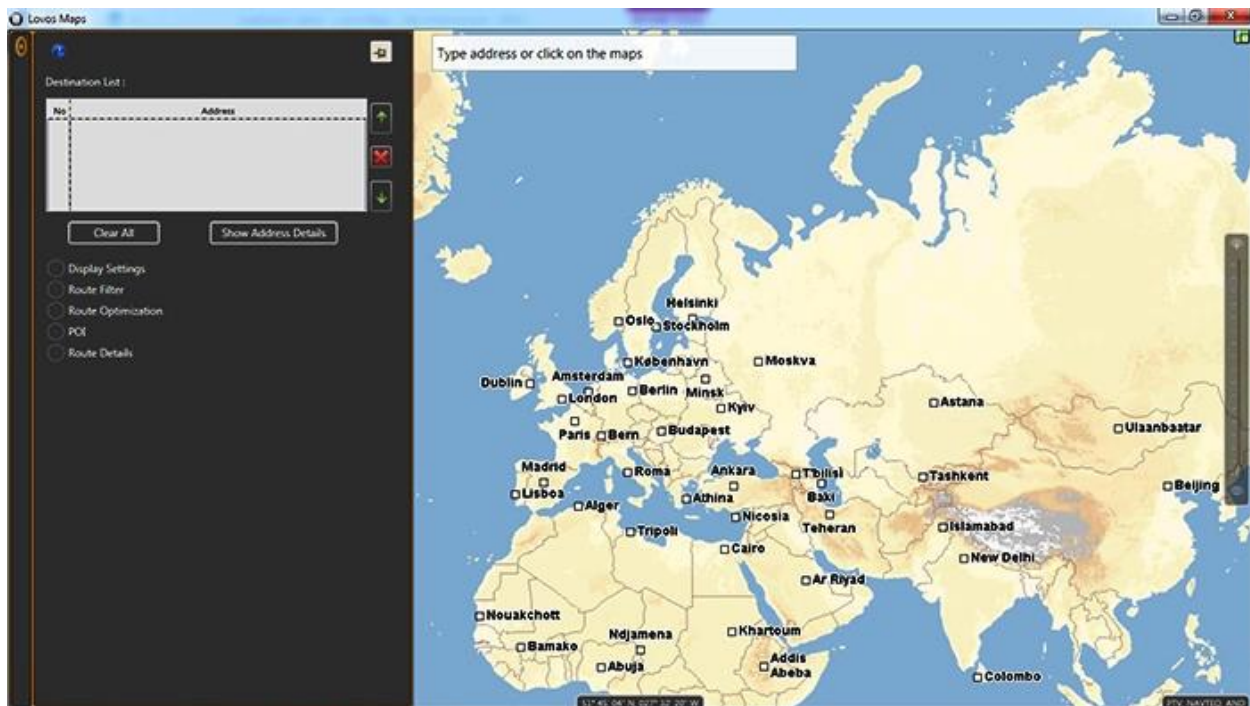
In the picture below, it shows the screenshot of the home screen layout. It is being shown when the application is started. The user can zoom the map using mouse wheel scroll up and down or by using the zoom bar on the right side of the layout. The user can search an address by typing the address in the text box on the home screen or by clicking on the text box and then clicking a location on the map. There is also information about the coordinates where the location of the mouse pointer is. On the bottom left the user can see what the current zoom level is. On the left part there is an expander-menu where the user can adjust the settings of the map.



*Lovos Maps Home Screen*

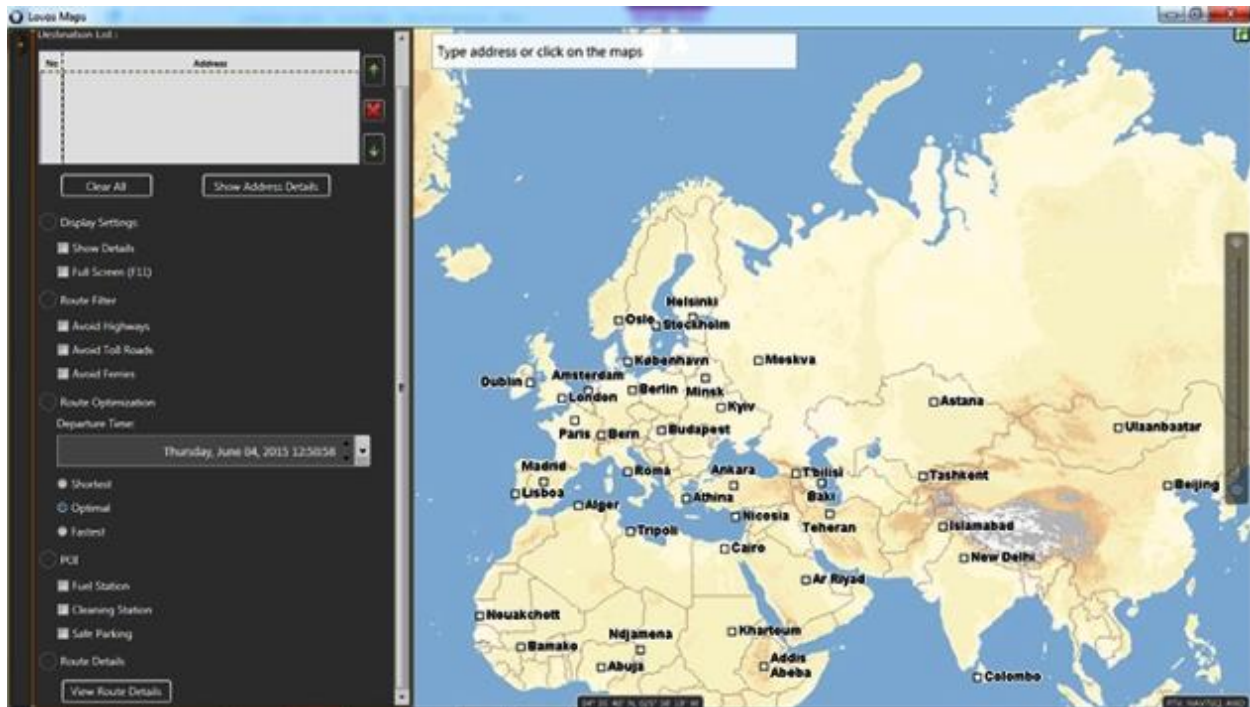
Picture below, the expander menu is shown. On the left top corner there is a button for help where the user can read the manual of the application. On the right top corner of the menu there is a pin button that can be used to keep the menu open while the user makes use of the map. After that there is a list box which can contain the list of the destination addresses. On the right side of the destination list, there are two buttons to move addresses up or down and a button to remove the destination address in the list.

Under the list box there are two buttons shown, the left button is to clear all the addresses added to the route, on the right side there is an address details button in which the user can show the address details selected from the list box.



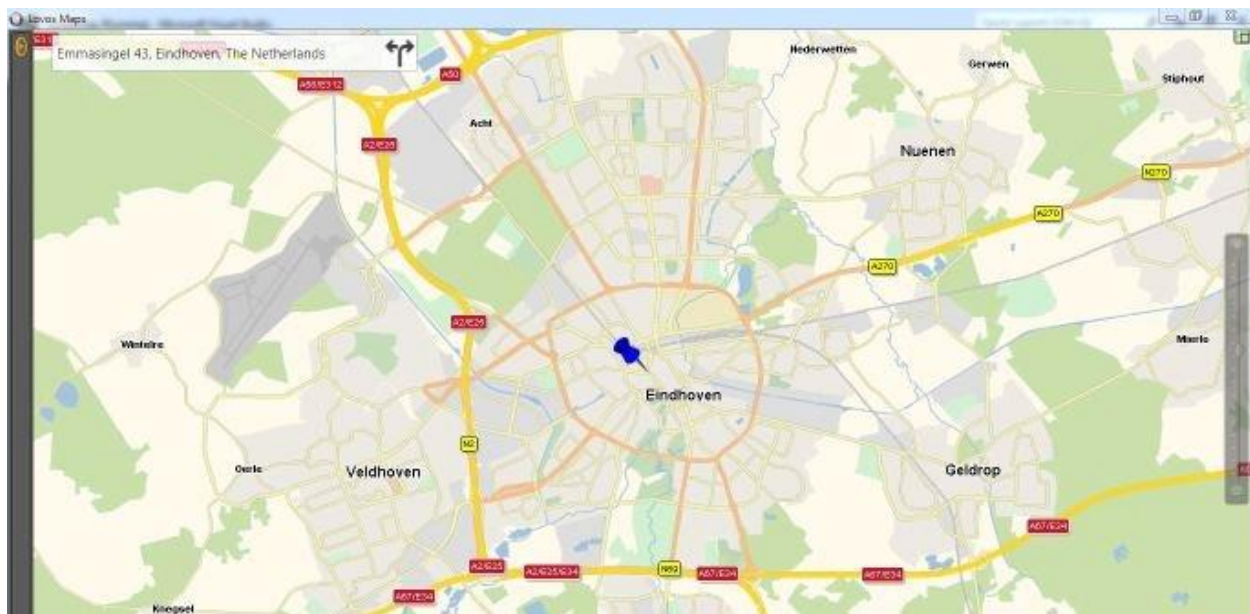
*Menu Expanded*

In the picture below, the submenus are expanded. The first submenu is the display settings. In this submenu the user can hide or expand the route details and show the map in full screen or not. The second submenu allows the user to filter the route by avoiding toll roads, highways and ferries.



*Submenu Expanded*

After the user has adjusted the route filter, the user can also optimize the calculated route by changing the departure date and time and choose the type of route to be used (shortest, optimal and fastest). Under the route optimization the user can show the point of interest (fuel station, safe parking spot and truck cleaning station) on the map and add them to the route. When a route is calculated, the user can also show the route description (for example, "Turn left after 200 meter") on the map.



*Locate a Place*



In the picture above, a place is located on the map by using a single mouse click. After that the user can create the current location as a start location of the route by clicking the *get direction* button in the text box.

In the picture below, a route is calculated between the Netherlands and France. A line of the route is drawn between two points and the toll part of the route is shown with a red color. In the details the user can see the distance, travel time, the arrival time and the toll cost details. To cancel the route the user can click on the cross button in the text box. If the user zooms in deep enough, the user can see the route direction which is shown by a moving line.



*Route Calculated*

In the picture below, a route with multiple destinations is calculated. By right click on the map the user can set another destination, re-calculate the route, cancel the route and manage the destination points on the route.



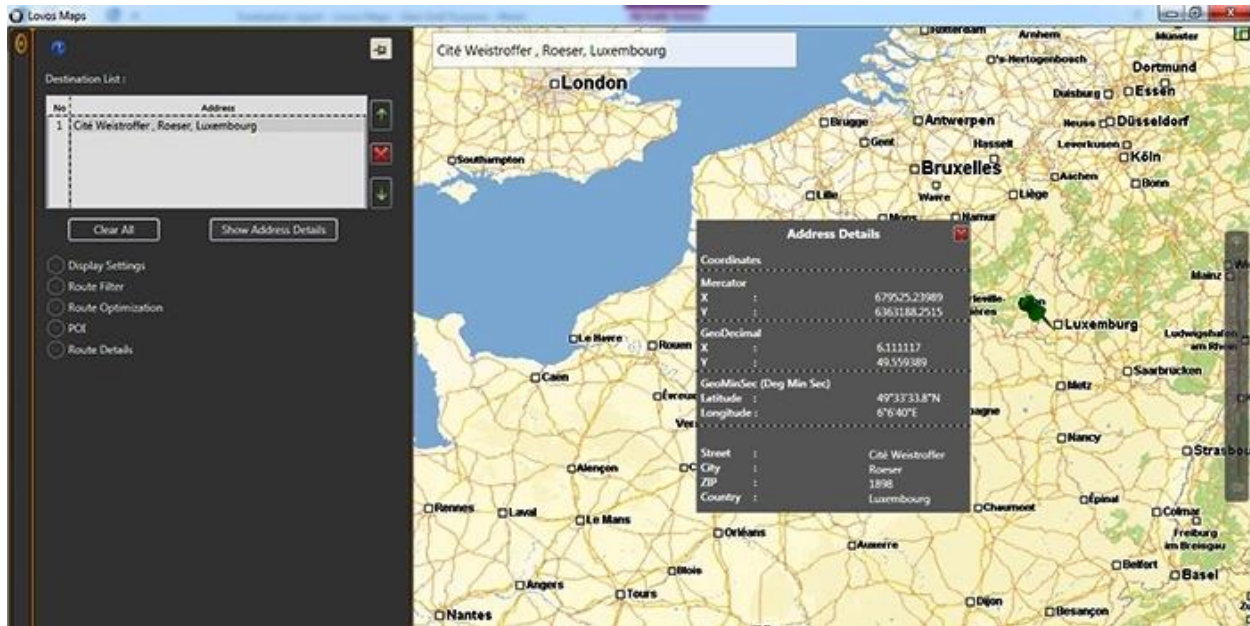
Multiple Destination Route



Locate With Postal Code

In the picture above the user can locate using postal codes, by using multi search fields. The user can choose the country code by using the dropdown list. After 2 digits of the postal code are inputted, the user will get a list of suggestions.





Showing Address Details

In the picture above, after a place is located a user is able to view the address details by clicking the *show address details* button. In the address details window, a user able to see three types of coordinates of the address and the details of the address selected.

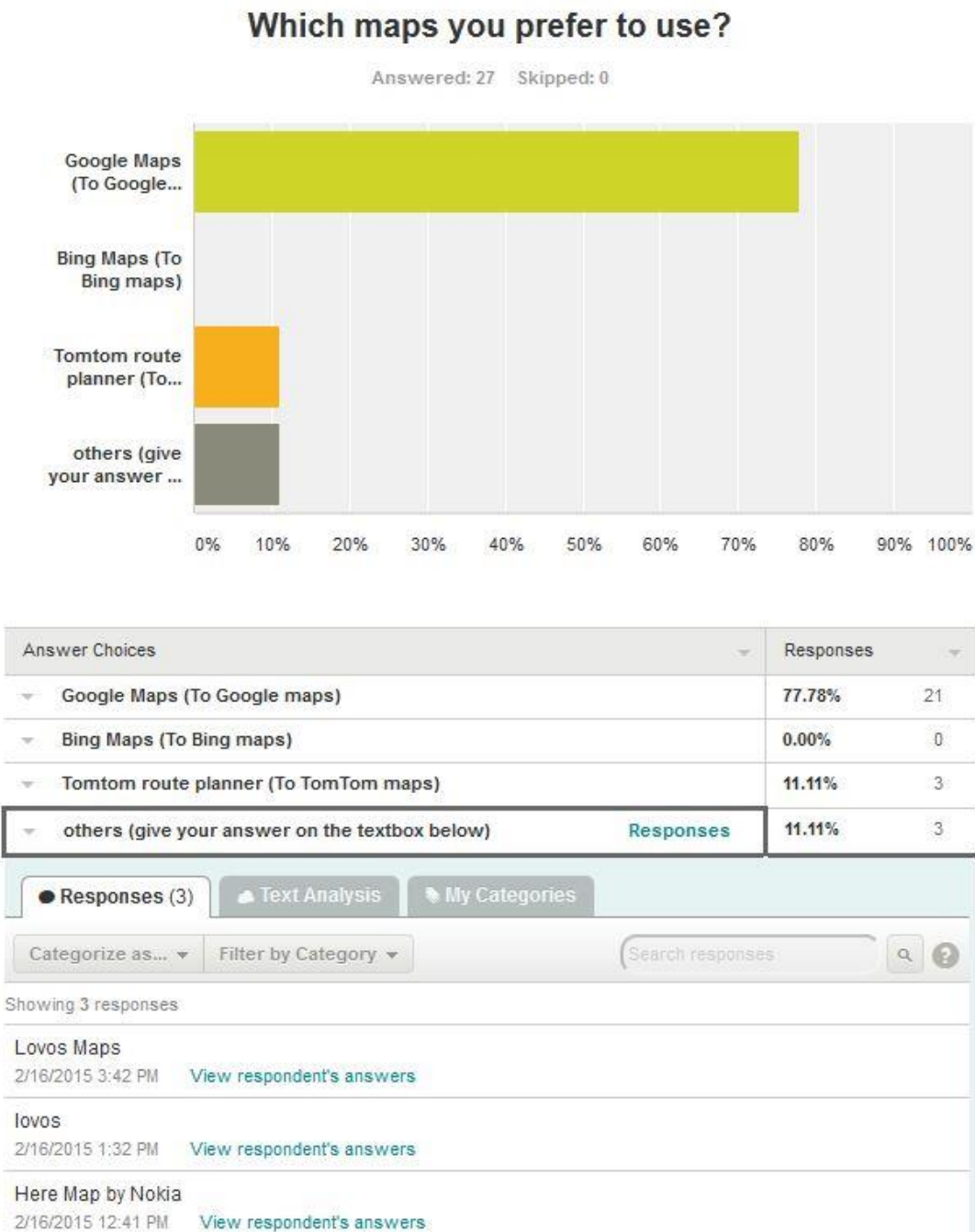
In the picture below, the user is able to show the point of interest on the map. When a POI is right clicked the user is able to set the POI as a start or via point.



Showing POI (Point of Interest)

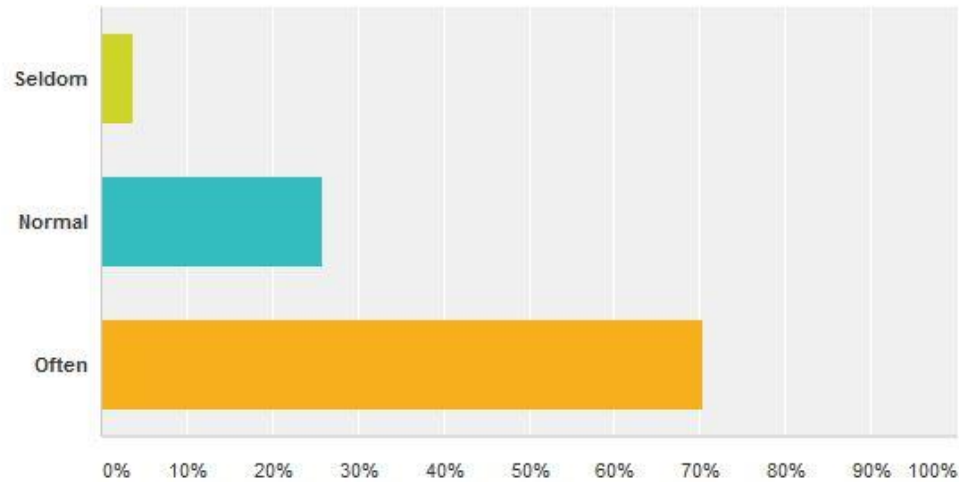


### III. Questionnaire Result Overview



## How often do you use Lovos Map (route calculation)?

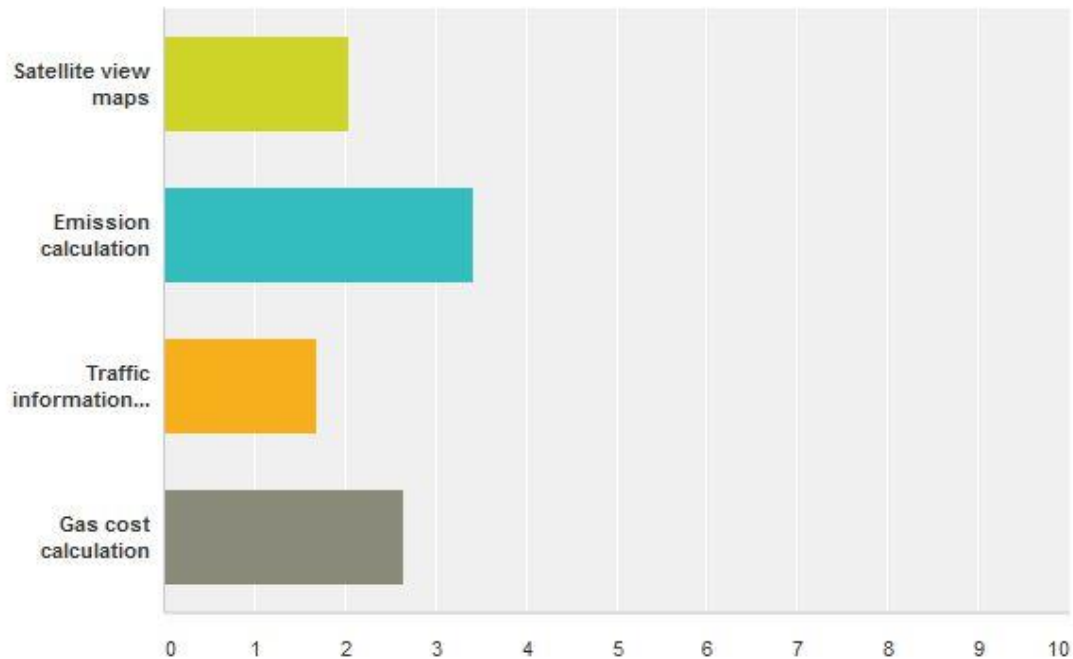
Answered: 27 Skipped: 0



Answer Choices	Responses	
Seldom	3.70%	1
Normal	25.93%	7
Often	70.37%	19
Total		27

Here below you can find some extra features besides the default feature that can be implemented in the new Lovos Map. Please put numbers from 1-4 on the field below, 1 is most important and 4 is least important.

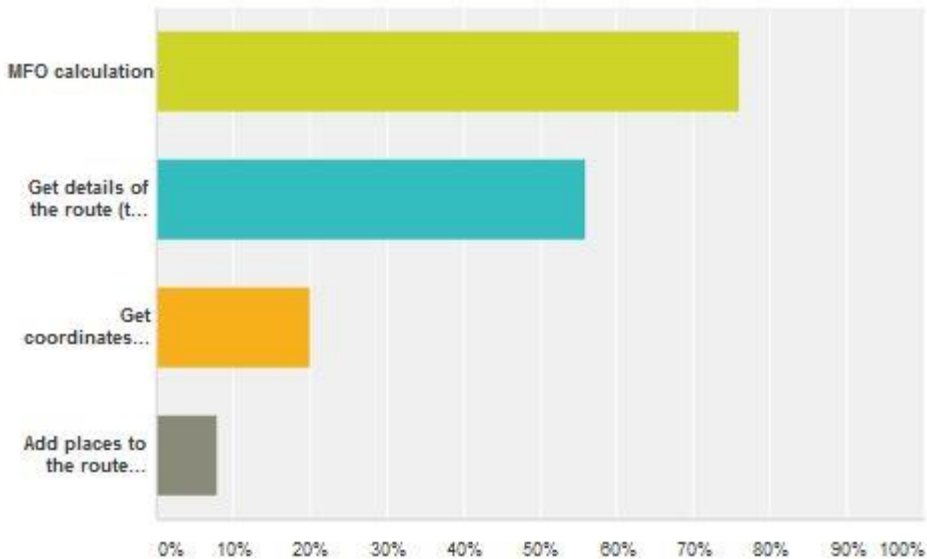
Answered: 27 Skipped: 0



Answer Choices		Average Number	Total Number	Responses
Satellite view maps	<a href="#">Responses</a>	2	55	27
Emission calculation	<a href="#">Responses</a>	3	89	26
Traffic information (e.g. traffic jam information)	<a href="#">Responses</a>	2	44	26
Gas cost calculation	<a href="#">Responses</a>	3	69	26

## Which functionality in the recent maps that you never use?

Answered: 25 Skipped: 2



Answer Choices	Responses
MFO calculation	76.00% 19
Get details of the route (turn left after 200m on wallstreet)	56.00% 14
Get coordinates from the address	20.00% 5
Add places to the route calculation (safe parking spot, fuel stations, cleaning stations)	8.00% 2
Total Respondents: 25	

Comments (1)

● Responses (1)

▲ Text Analysis

📁 My Categories

Categorize as...

Filter by Category

Search responses

🔍

?

Showing 1 response

Ilodk Vignette-roads. Does it even work?

/16/2015 12:07 PM

[View respondent's answers](#)

## If you are familiar with Lovos Route calculation, what more functionalities do you want to add or change to make your job easier?

Answered: 15 Skipped: 12

● Responses (15)

▲ Text Analysis

● My Categories

Categorize as...

Filter by Category

Search responses

Showing 15 responses

a better option to put extra stops in the route	3/1/2015 1:50 PM	<a href="#">View respondent's answers</a>
Diferants between 1 or 2 routcalculations	2/26/2015 3:23 PM	<a href="#">View respondent's answers</a>
clean up of out of date addresses	2/26/2015 11:58 AM	<a href="#">View respondent's answers</a>
calculate difference between 2 / 3 routes	2/26/2015 11:52 AM	<a href="#">View respondent's answers</a>
Options to set 'route characteristics' for route calculation: cheapest, quickest, shortest, with or without Traffic information etc.	2/26/2015 8:57 AM	<a href="#">View respondent's answers</a>
- Alternative routes (for example to choose from shortest, fastest, cheapest) - Toll calculation - Fuel consumption information - Adjust vehicle details (weight, EURO norm, height, length)	2/20/2015 5:20 PM	<a href="#">View respondent's answers</a>
search places without being obliged to select a country	2/18/2015 12:20 PM	<a href="#">View respondent's answers</a>
if possible, adding tunnels from the routes in lovos	2/17/2015 10:25 AM	<a href="#">View respondent's answers</a>
nothing	2/16/2015 5:24 PM	<a href="#">View respondent's answers</a>
Work more with GPS coordinates or / and adresses. Add ferry's / trains + ferry choices and schedule's	2/16/2015 2:29 PM	<a href="#">View respondent's answers</a>
-Streetview to correctly validate an adress. -Multiple ferry-routes	2/16/2015 2:25 PM	<a href="#">View respondent's answers</a>
More in detail position of adresses	2/16/2015 12:41 PM	<a href="#">View respondent's answers</a>
putting new via points, by clicking on the map	2/16/2015 12:33 PM	<a href="#">View respondent's answers</a>
more parkingplaces visible on the map, to find good location for couplings.	2/16/2015 12:15 PM	<a href="#">View respondent's answers</a>
Possibility + easy access alternative route. Possibility to view and block Y/N tunnel, cross over heights.	2/16/2015 12:07 PM	<a href="#">View respondent's answers</a>

IV. Test Report



---

# TEST REPORT

---

Lovos Maps



11<sup>TH</sup> MAY, 2015  
VOS LOGISTICS

## TABLE OF CONTENTS

1. Introduction .....	54
2. Scope.....	54
➤ Test items.....	54
➤ Approaches .....	54
➤ Integration Testing.....	54
3. Test Result.....	55
➤ Integration Testing.....	55
➤ Performance Testing.....	55
➤ Interface Testing .....	55
➤ Acceptance Testing .....	56
4. Conclusion.....	56

## 1. Introduction

One of the most important phase during the software development cycle is testing. Software testing is done to ensure the quality of the product delivered to the customer. By run the test the developer can prove if the product deliver the promised requirements.

The software that is being tested is Lovos Maps, a route planning software for logistics company 'Vos Logistics'. This software is done as part of graduation internship project.

## 2. Scope

### ➤ Test items

As described in the test plan, there are features that is being tested and not being tested. Below are the details:

#### ○ Feature being tested

- Routing between two points with default setting
- Calculating Travel time and distance with shortest route optimization
- Calculating Travel time and distance with fastest route optimization
- Calculating Travel time and distance with optimal route optimization
- Filtering route calculation with blocking the toll road
- Filtering route calculation with blocking the vignette road
- Filtering route calculation with blocking the ferries
- Filtering route calculation with blocking the highways road
- Showing VIA points (POI Parking, fuel, clean station)
- Searching for place by address / keywords
- Showing the route details
- Showing the address details
- Calculating toll cost
- Shortcut key features (F1, F11, F12)
- Changing departure time of the route

#### ○ Feature not being tested

- Integrating with Lovos

### ➤ Approaches

As described in the test plan, the features is tested using four approaches:

- Integration Testing
- Interface Testing
- Acceptance Testing
- Performance Testing



### 3. Test Result

The result of the test per approach:

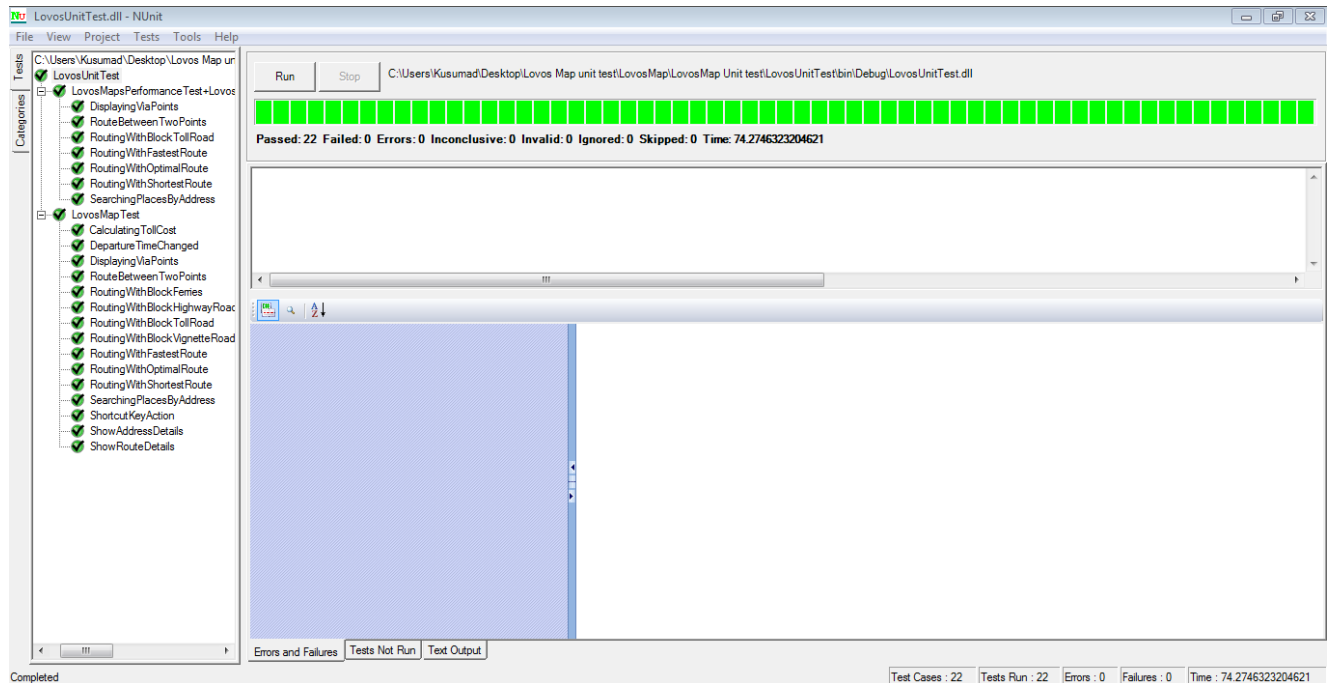


Figure 15: Unit Test & Performance Test Lovos Maps

#### ➤ Integration Testing

As described in the test plan, during the integration test or so called unit test, the functionality is tested with a certain input and check if the output is correct. The tools used is Nunit. The test is done in Citrix environment and locally. The test result can be seen in the Figure 1 above.

#### ➤ Performance Testing

The performance of the application is tested by using time counter, the time of all the features to be tested is counted while doing the task. The tools used is Nunit. The test is done in Citrix environment and locally. The test is successful completed because every features tested is not exceeding 5 seconds of run time. The test result can be seen in the Figure 1 above.

#### ➤ Interface Testing

The user interface test is done during the feedback session or demo session by asking the user to do some tasks without giving any instruction beforehand. This test is done by 9 users, 8 of 9 users are Vos Logistics' employees and 1 other user is an ICT student at Fontys Hogeschool.

The users were able to manage to do the task without big problem/difficulty. Some of the bugs found during the session is fixed by the student trainee. Considering that there is no big difficulty during the session, the interface's logic is user friendly enough and the test pass the criteria.

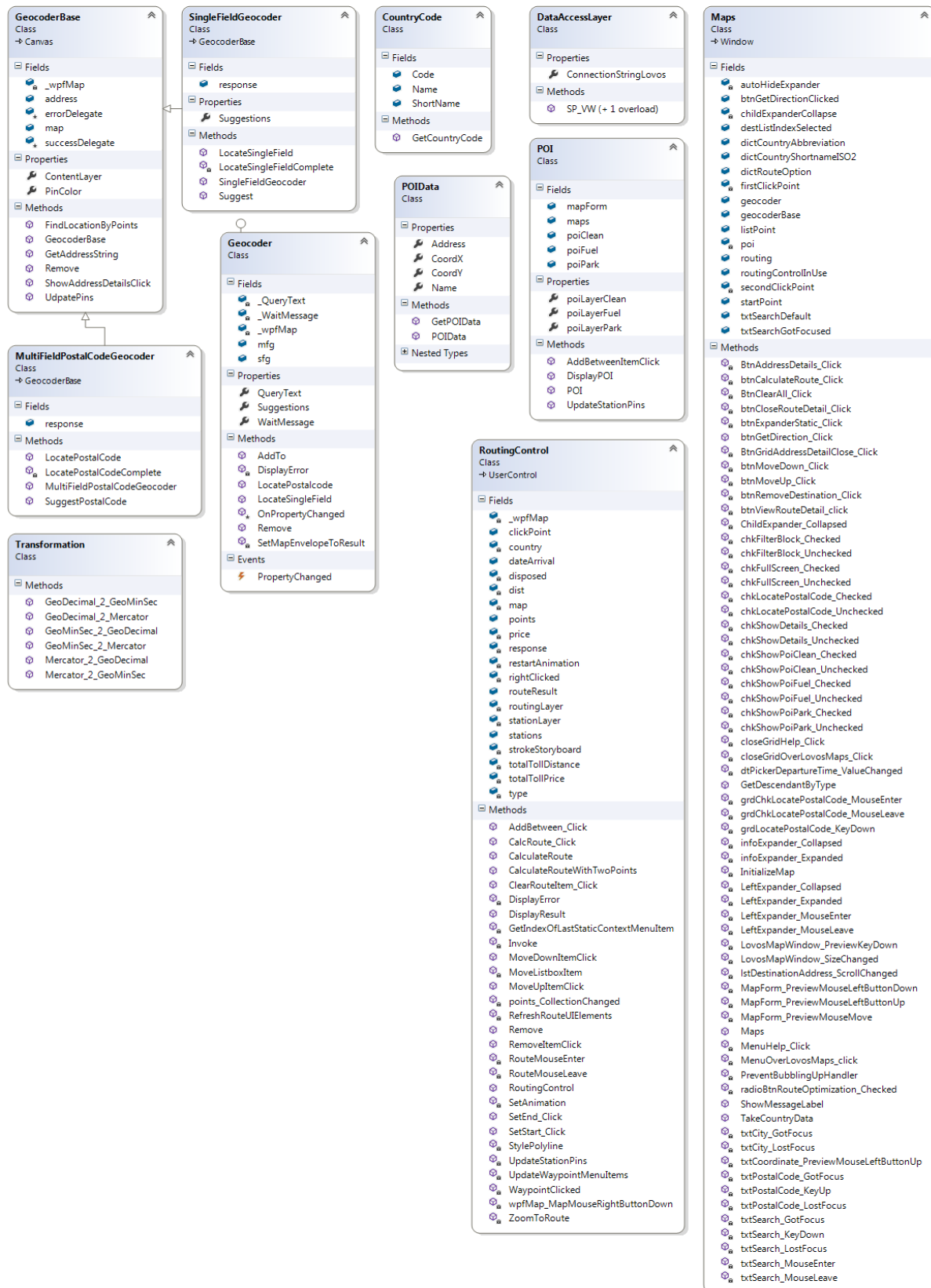
#### ➤ Acceptance Testing

During the feedback and demo session the student trainee also test the acceptance of the user requirements. 100% of the main requirements and 40% of the extra requirements are fulfilled by the application. Other 60% of the extra requirements cannot be completed due to lack of licenses (40%) and time (20%).

## 4. Conclusion


From the results above can be conclude that the test ran successfully. Some of the bugs found are fixed. By using many types of test the developer can ensure the quality of the application as promised to the customer.

## V. Class Diagram





- Maps → class which contains all the event handler (interaction logic) for the layout (maps.Xaml). This class is the startup point of the application.
- RoutingControl → class where the routing instance is formed. It contains the event handler for right-click menu, draw the route on the map and draw the routing pins on the map.
- Geocoder → contains the handler for location suggestion, create instances of singleFieldGeocoder and MultiFieldPostalCodeGeocoder and make the geocoding result visible on the map.
- GeocoderBase → base class for singleFieldGeocoder and MultiFieldPostalCodeGeocoder. It has base functionality, like update the pins on the map for both extension classes.
- SingleFieldGeocoder → extension class of GeocoderBase to locate an address using single field input.
- MultiFieldPostalCodeGeocoder → extension class of geocoderBase to locate postal code using multi field input.
- POI → class where the data of the POI is requested, display the POI on the map and event handler for the POI.
- DataAccessLayer → class where the connection to the database is created.
- POIData → class that represent the POI object.
- CountryCode → class which represent the CountryCode object.
- Transformation → class which contains the transformation of the coordinate formats.

## VI. Database Diagram

tblCountryCodes	
	nCountryNo
	szCountryName
	nCIngInkLedger
	nCIngInkLedgerVAT
	szShortName
	nCityNumber
	szAvoidCountryFlag
	szEuroCountry
	szIso2Code
	szIso3Code
	nRegionCode

tblPOI	
	nId
	nX
	nY
	bType
	nCountry
	szFlyOver
	szCompanyRating

tblPOIStdAddress	
	nId
	nPOI
	nStdAddress
	nCompany
	nRSAddress

tblStdAddress	
	nId
	nCompany
	nCountry
	bDepot
	bRegion
	nPlanArea
	szName
	szAddress
	szZipcode
	nPhoneCountry
	szPhoneNumber
	nFaxCountry
	szFaxNumber
	bPerson
	szPerson
	szRoute
	szBitmap
	bReorg
	tmUsed
	bMutcode
	nNorm
	bWash
	nWashCost
	bWashCurrency
	szExternalReference
	szExternalReference2
	nNormLoad
	nNormUnload
	szEmailAddress
	bTerminal
	nRSId
	szCoords
	szCity
	bCorrect
	szExtra
	bOutOfUse
	dtOutOfUse
	szUseUser
	bLoadAddress
	bCoupleAddress
	bSafeAddress
	bTransferAddress
	bFuelAddress
	szExtra2
	nNewPlanArea
	bValidFromMap
	nGMPFax
	szEmail

- tblPOI → contains all the POI Id, coordinates and types.
- tblStdAddress → contains all the standard address that is often used in Lovos.
- tblPOIStdAddress → contains the combination between ID from tblPOI and tblStdAddress.

Two types of data being taken from the database are POI (Point of Interest) data and Country abbreviation data.

The POI data is taken from 3 different tables. The tables in the database are not relational. The first table being used is "tblPOI". From this table the coordinates of the POI and the type of POI is taken. The second table is "tblPOIStdAddress", this table is being used to get the combination between "tblPOI" and "tblStdAddress". The third table is "tblStdAddress", from this table the name and address of the POI are taken.

The country abbreviation data is taken from table "tblCountryCodes".