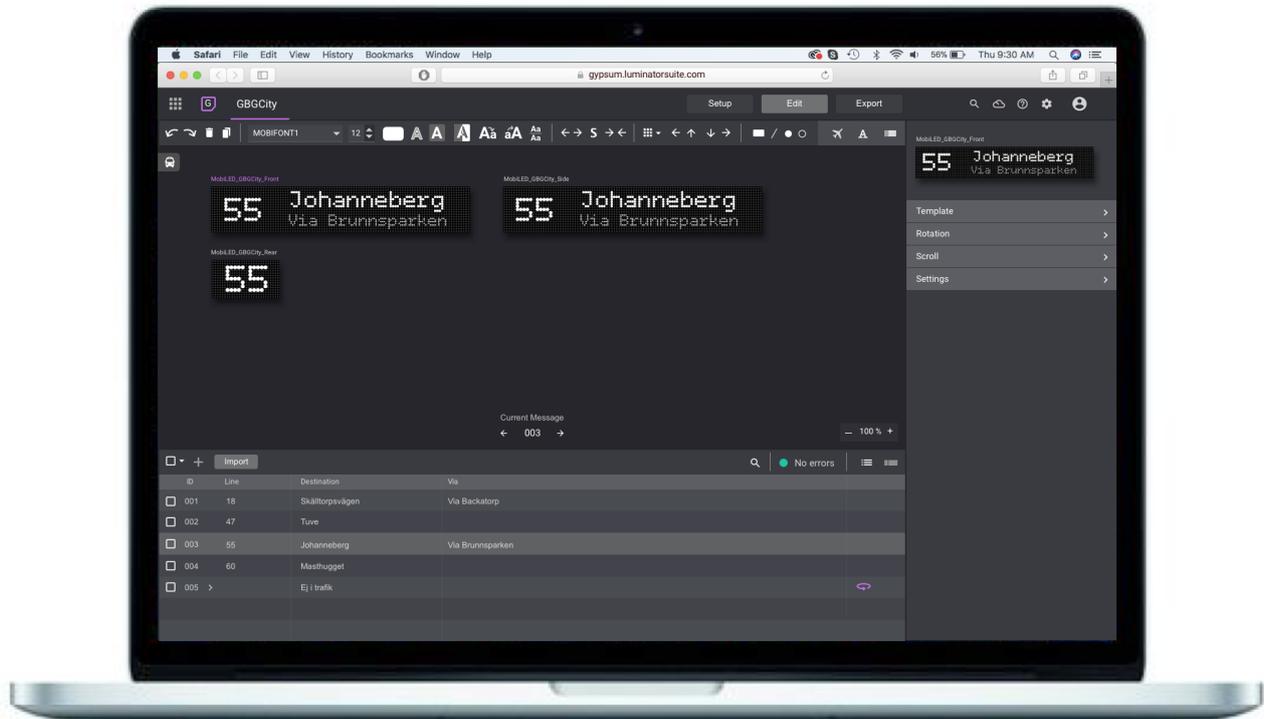




CHALMERS



Behind the scenes of public transportation *Designing an Interface for a Destination Sign Editing Software*

M.Sc. Thesis in Industrial Design Engineering

SOFIA ANDERSSON
LEI HE

CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2018

Department of Product and Production Development
Division of Design & Human Factors

Behind the scenes of public transportation

SOFIA ANDERSSON
LEI HE

SUPERVISOR: LARS-OLA BLIGÅRD
EXAMINER: LARS-OLA BLIGÅRD

Master of Science Thesis PPUX05

Behind the Scenes of Public Transportation

Master of Science Thesis within Industrial Design Engineering

© Sofia Andersson, Lei He

Chalmers University of Technology
SE-412 96 Göteborg, Sweden
Telefon +46(0) 31-772 1000

Cover photo: Sofia Andersson, Lei He
Print: Repro Service Chalmers

ACKNOWLEDGEMENTS

This thesis was carried out during the spring term of 2018, by two students at Industrial Design engineering, Chalmers University of Technology. This project involved over 50 users from 11 countries and we can honestly say without this team effort, we wouldn't have gotten far. We are so thankful for the cultural exchanges and wonderful people we have met during the project. Thank you all external participants who contributed with your knowledge and feedback.

Also, we want to thank LTG for all support, resources and time invested in us. It created good basis for success. Thank you for the teams in LTG Sweden, LTG Rastatt, LTG Switzerland and Luminator for all your contributions to the

project. A special thank you for Gerald Pregger, Christian Brändle and Jonathan Gammell helped us during the abroad field studies. Additional credits for LTG Sweden who hosted our project and provided knowledge and space for us to let loose our skills, creativity (and post-its). The fact that you let us be a part of your team mean a lot to us. Thank you to our supervisors at LTG Sweden, Patrik Nilsson and Mattias Björklund.

Last but definitely not least, we want to thank our supervisor at Chalmers, Lars-Ola Bligård, who was there for us whenever we needed and always keen on reading report parts. Your advices brought structure to a project with many levels of complexity.

Sofia Andersson

SOFIA ANDERSSON

Lei He

LEI HE

TERMINOLOGY LIST

UX - User Experience is what a user feels when interacting with a product, system or service. It is a summary of all the aspects of the product such as usability, accessibility and pleasure.

UI - User Interface is the part of the product, system or service which the user interact with. In the digital world, the UI often refers to the computer or phone interface.

UX DESIGN - User Experience design is the approach to design for enhancing the user experiences. It often involves research to understand the problem and its users.

UI DESIGN - User Interface design is the process when the interface is designed with the research from the UX research as a base.

PTA - Public Transportation Authority who provides transportation services.

PTO - Public Transportation Operator who provides PTAs with vehicles.

LTG - Luminator Technology Group is a group of companies specializing in the field of electronic information systems.

MIE - Destination sign editor developed by LTG Sweden AB.

TEDPlus - Destination sign editor developed by LTG Rastatt GmbH.

ALPHA-NT - Destination sign editor developed by LTG Switzerland GmbH.

IPS - Destination sign editor developed by Luminator.

Elyse - Destination sign editor developed by TwinVision.

ABSTRACT

There are unlimited ways to get where you want to go: walk, bike, car, bus, tram and airplane to name a few. This thesis was situated in the public transportation, a context in which the transportation is a collective activity where passengers put high requirements on the passenger information systems, in order to get where they want to go. Designing for public transportation meant designing for the the big mass of people who are part of a complex system of technology, information and timing.

This thesis focused on improving the user experience of the person behind the scenes who creates information for destination signs in public transportation vehicles, mainly buses. The project was for Luminator Technology Group who specializes in passenger information systems with products all around the world. The scope was to create an interface for a web-based editor for creating destination content. The editor would serve the needs of a global market by collecting inspiration from the five current destination editors within the Group and their users.

The project followed an iterative user centered process. The first part of the project focused on understanding the context of the problem, studying current software, their users and user requirements on the new editor. Many interviews and demonstration sessions were conducted with both internal and external user from different countries. A two week field trip to Germany, Switzerland and US was carried out

to meet and further study users from a global perspective. The requirements from the user studies were used to create a clickable prototype in three iterations. The prototype was developed into more details for each iteration and validated in feedback sessions and usability tests with the help of the contact network created in the research phase.

The user studies shed light on problems with current software, who the users are and how they want to create destination messages in the future. It turned out that the current software are very unmodern even though each software have strengths which could be used in the new software. The current interaction issues consist of a great amount of repetitive work, low automation level, long learning curve, complicated settings and seldom usage. The users differed in software skills where for example external customers who have mechanic background need help by LTG experts very often. These users had very different needs which had to be designed for specially.

This thesis project resulted in a clickable prototype of a destination sign editor concept called Gypsum. It is a web-based application which supports editing of all sign brands within LTG and has the look and feel of LTG digital portfolio. It demonstrates a balance between efficiency, customization, simplicity and powerful functionality to serve the needs of a global market as well as users with various skill level.

CONTENTS

01 INTRODUCTION	1
1.1 Public transportation context	2
1.2 Project scope	3
1.3 Report structure	5
1.4 Final result	6
02 PROJECT CONTEXT	7
2.1 About the company	8
2.2 Public transportation system	11
2.3 System products	12
2.4 Software description	15
2.5 Current and future ways of usage	21
2.6 Market usage	23
03 PROJECT EXECUTION	24
3.1 Method description	26
3.2 Work process	29
04 UNDERSTAND PHASE	37
4.1 Brand analysis	38
4.2 Research questions	41
4.3 Domain knowledge	42
4.4 User types	43
4.5 Interaction	46
4.6 Software evaluation	50
4.7 Conclusion	53
05 PRODUCT VISION	55
5.1 Persona and high level use cases	56
5.2 Improvements	60
5.3 Interaction vision	62
4.4 Conclusion	63
06 CREATE PHASE 1: FRAMEWORK	65
6.1 High level use cases	66
6.2 Software structure	68
6.3 Functionality	70
6.4 Main workflow	71
6.5 Design assumptions	74
6.6 User validation and outcome	75
6.7 Conclusion	77

07 CREATE PHASE 2: INTERACTIVE PROTOTYPE	79
7.1 Interaction framework	80
7.2 Design assumptions	87
7.3 User validation	88
7.4 User validation results	90
7.5 Conclusion	93
08 CREATE PHASE 3: LOOK AND FEEL	95
8.1 Final details	96
8.2 Design assumptions	101
8.3 User validation and outcome	102
8.4 Conclusion	104
09 FINAL DESIGN	105
9.1 Design foundation	106
9.2 Software structure	108
9.3 Software functionality	109
9.4 Design for differences	113
10 EVALUATION	129
11 DISCUSSION	133
11.1 Results	134
11.2 Process and methods	135
11.3 Recommendations for future work	137
12 CONCLUSION	139
REFERENCES	141
APPENDICES	143
Appendix 1: Gantt chart	144
Appendix 2: Interview script for software users	144
Appendix 3: Pre-interview questionnaire for remote software users	146
Appendix 4: Usability test script	146
Appendix 5: Detailed evaluation of each software	166
Appendix 6: Design assumptions in create phase 1	176
Appendix 7: Design assumptions in create phase 2	177
Appendix 8: Design assumptions in create phase 3	179
Appendix 9: Potential improvements gathered from user studies	180
Appendix 10: Potential improvements for prototype in create phase 2	183
Appendix 11: Potential improvements for final prototype	184
Appendix 12: Functional and data elements and their grouping	184
Appendix 13: Low level use cases	188
Appendix 14: Overview of the final design	193
Appendix 15: Usage scenarios of the prototype in create phase 2	198
Appendix 16: Usage scenarios of the final prototype	218

01

INTRODUCTION

This chapter introduces the big perspective and how the thesis is framed. The context in which the project is situated is explained, along with the background and scope. It also explains how the report is structured and shortly present the final outcome.

1.1 Public Transportation Context



Figure 1.1 Public transportation situation in Gothenburg.

Public transportation has shouldered the responsibility of making it possible for people to go from point A to point B. They know their end destination, but what then? There is more to the public transportation than just providing the vehicle itself. With the job description also comes the aspect of guiding the passengers so they know which bus to take, from where and at what time. As shown in figure 1.1, there is a man in hat wearing a backpack on a bus stop in Sweden. He is on his way to work. It is early in the morning and the bus stop is crowded with both

other passengers and buses and trams. Most likely he has checked the departures and knows which line he has to wait for and every time a bus or tram arrives he check the signs to decide if it is the right one or not. Talking about this signage, where does it come from, who decides what should be there and who creates the actual content? This thesis concerns of what goes on behind the scenes of destination signage and how to get “16 Eketrögatan” on the signs.

1.2 Project Scope

This section presents the scope of this project in terms of aim, requirements and delimitation. The aim highlights the purpose and the focused aspects of this project, and the initial requirements describes some requirements on the project outcome from LTG side. Finally, the delimitation sets the boundaries of the project.

Project Background

LTG was relatively recent formed as a Group. Currently, LTG contains several sub-companies including LTG Sweden AB, LTG Rastatt GmbH, LTG Switzerland GmbH and Luminator. From being competitors, now they are supposed to collaborate to gain market shares together. All the brands within the Group has their own technology which are not compatible with each other. This means there are many software which have the same purpose but control different sign systems. The Luminator Suite was an initiative to unite the software by having one software application for each functionality such as infotainment and fleet management. This project involves the design of the module in the Suite which would be used to control destination signs.

Aim

The aim of this master thesis project was to develop an improved version of five destination sign editing software within Luminator Technology Group by looking into the good and bad aspects of each software.

Destination sign editing software refers to the software used to edit text and graphics which will be presented in public transportation exterior signs, “55 Lindholmen Science Park” for example. Figure 1.2 is one example of a destination sign on a Gothenburg electric bus. The main users of them are those who are programming the content of exterior signs. Currently, these software are mostly used in Europe and US, but also used in Asia and Australia. Figure 1.3 shows one example interface of one of these software, MIE.

The project aimed to improve the current destination sign editing software interface according to the user needs of a global market. The interface should become more user friendly while maintaining the current value of existing software. The usage of the software should be more efficient and focuses on intermediate users. Additionally, it should also be easy to understand and learn for both old and novice users. The elements of interface, such as colour scheme and style should be updated and according to the guidelines of LTG Suite (their digital product portfolio).



Figure 1.2 One example of destination sign on Gothenburg electric bus.

Initial Requirements

The project involved looking into the main functions of destination sign editing software within Luminator Technology Group. The new software interface should be user friendly and has the same look and feel according to Luminator Suite. It is also expected to be web-based, and should therefore be platform independent and responsive to different screen sizes. It is also expected to allow multi-usage (computer, iPad and iPhone).

The project was expected to deliver user studies on the software which gives understanding of the experience, needs and pain points of the users and therefore work as a basis for the redesign. For the final concept, a clickable prototype should be developed to demonstrate the interface of the new software and how it works, with regards to different use cases.

Delimitation

Due to the broad scope and time limitation, this project is limited to focusing on developing the destination sign editing software based on current technology and hardware. The focus of the project is on the interface, therefore no coding will be developed. The project also focuses on web application and not mobile application. The project is limited to designing a computer interface, however the design language must be implementable for other platforms as well, such as iPad or iPhone.

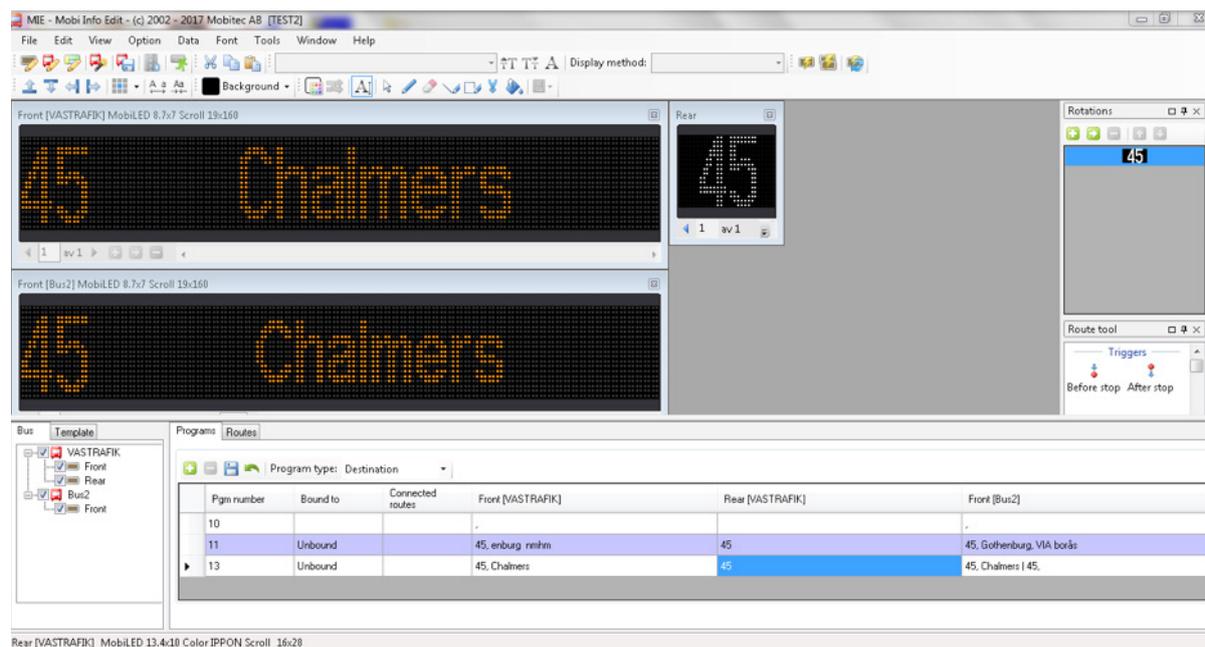


Figure 1.3 One example interface of one of the destination editing software MIE.

1.3 Report Structure

The report is structured in four main sections: Project Basis, Understand, Create and Closing Chapters. Project Basis explains the scope of the project and the context in which it is situated. The company, LTG, and the software relevant for this project are presented, followed by a description of the methods and process used.

Next section, Research, describes all user research and analysis of current software. It also presents the framework of what should be designed in terms of personas, improvements and interaction vision.

The third section in the report describes how the product was developed in three iterations with user feedback, assumptions and design decisions. The final design is presented at the end of this section.

In the fourth and final section, the perspective is widened and the project is wrapped up by evaluating the outcome towards the scope. It is also discussed how the product affects its users and context. Some detailed data is not presented in the report and can be found in the appendices.

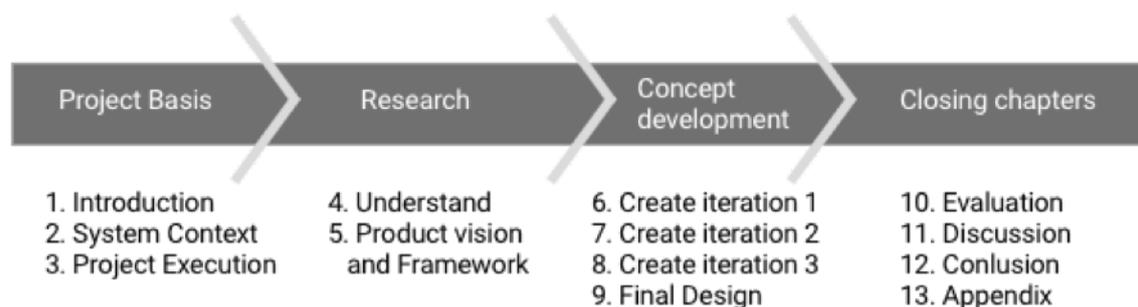


Figure 1.4 Visual representation of report structure.

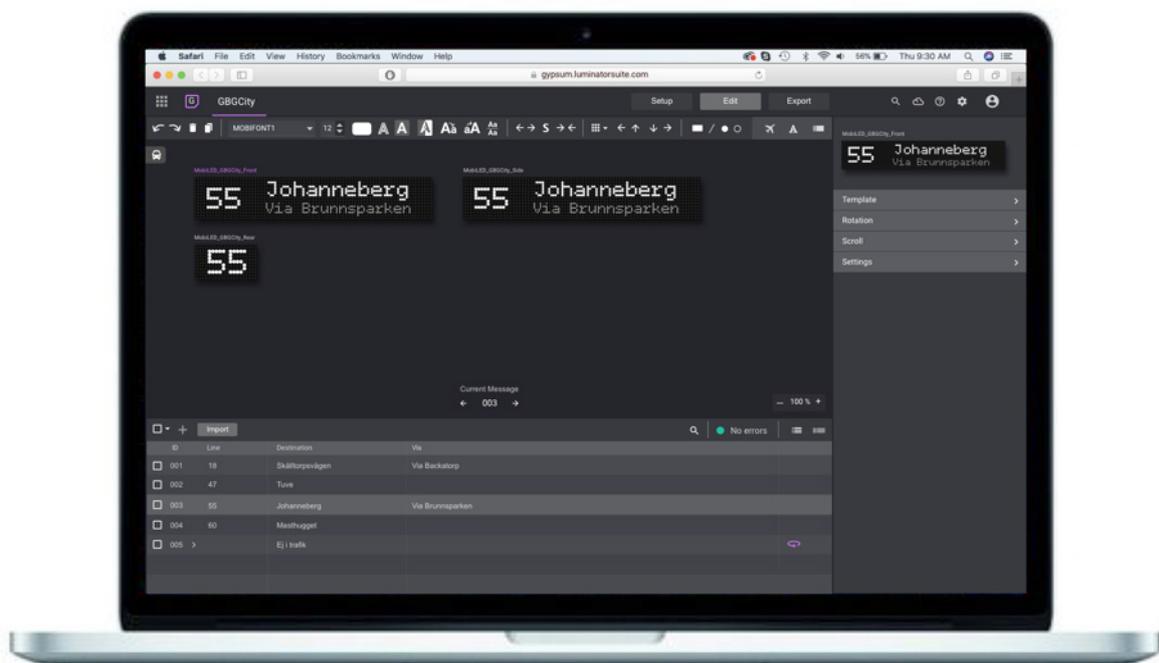
1.4 Final Result

This project resulted in a clickable web-based software prototype Gypsum with the same look and feel as Luminator Suite, see figure 1.5. The software is used to pixel edit the text and graphics for all LTG destination signs, typically the signs showing line number and destination facing outside the buses. With the outcome of the user studies as a basis, Gypsum is designed for three key user types, internal customer support, external administrators and external fixers, as well as satisfying the needs of a global market.

Gypsum offers a good balance between simplicity and powerful functionality, and efficient editing and pixel-by-pixel control. It reduces much of the repetitive work with users were forced to do previously, by offering higher automation level with for example import and auto-formatting functionalities. Gypsum makes it possible to edit destination information for all sign brands within the Group and it combines all current functionally in an intuitive way. At the same time it pushes innovation by for example

collaboration functions and error report system. Graphic, font and template editors are available for customization and all-in-one purposes.

The software is structured into three main sections namely Setup, Edit and Export, which follows the main workflow of destination sign editing. First of all, in the Setup section, vehicles and signs needs to be setup to let the software know what hardware users have. This is followed by content editing for created signs in the Edit section. In order to edit signs in a efficient and smart way, users are expected to enter the destination information in the message listing and the content will be auto-formatted to all sign. To make small adjustments, the toolbar and property section can be used to do individual sign editing. Lastly, when the editing is finished, users can generate outfiles in the Export section. These outfiles can then be loaded into the control unit in the bus to control the content of the signs.



6 Figure 1.5 One example interface from the final prototype of this project.

02

SYSTEM CONTEXT

Many different systems and elements are involved in the destination sign editing process. This chapter describes the ones that are relevant for this project. First of all, the public transportation system model concerning LTG, vehicles and passengers is introduced to explain how the system works from a broad perspective. Existing software within the Group that were redesigned in this project are then explained to present the current design and characteristics of each software. Based on the software description, the similarities and differences between these software are further defined. Today there are two main ways of using the software. These are explored to define requirements for future destination sign editing software.

2.1 About the Company

The thesis project was for LTG Sweden (former Mobitec, <http://www.mobitec.se>) which is a company specializes in the design and manufacturing of system technology for mobile passenger information on bus and rail transportation. The company was established in 1987 in Sweden. Today it owns 90% of the Swedish market and is a world leader serving international markets of 50 countries from three production facilities. Their objective is to provide comfortable and user friendly products or services for passenger, driver, transport manager and maintenance teams (Mobitec, 2018). Their office is located in Herrljunga.

The company was acquired by Luminator Technology Group in 2012, which is a group of companies specializing in the field of electronic information systems. Currently the companies are working on uniting and updating the softwares within the group. There are five similar softwares that were developed by different com-

panies within the Group, namely MIE (developed by LTG Sweden AB), IPS (developed by Luminator), Elyse (developed by Twinvision), TEDPlus (developed by LTG Rastatt GmbH), and ALPHA-NT(developed by LTG Switzerland GmbH). More details about these software will be presented in chapter 2.4.

LTG product portfolio

Today LTG’s products are installed on buses, trains, aircraft and other commuter vessels around the world that guide and inform passengers throughout their journey. Figure 2.1, 2.2 and 2.3 illustrate the main product portfolios LTG offers to different vehicle categories. The products that are involved in this project mainly belong to the software system, which allow users to program the content of destination signs for buses, rails and other vehicles (Luminator Technology Group, 2018).

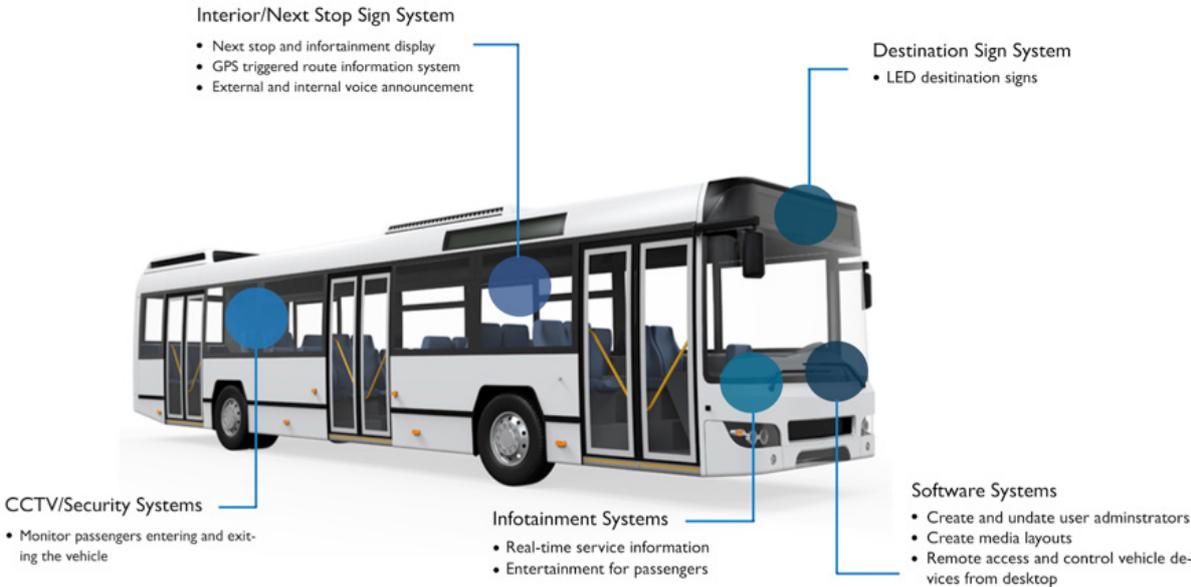


Figure 2.1 Visual representation of LTG products for bus system.

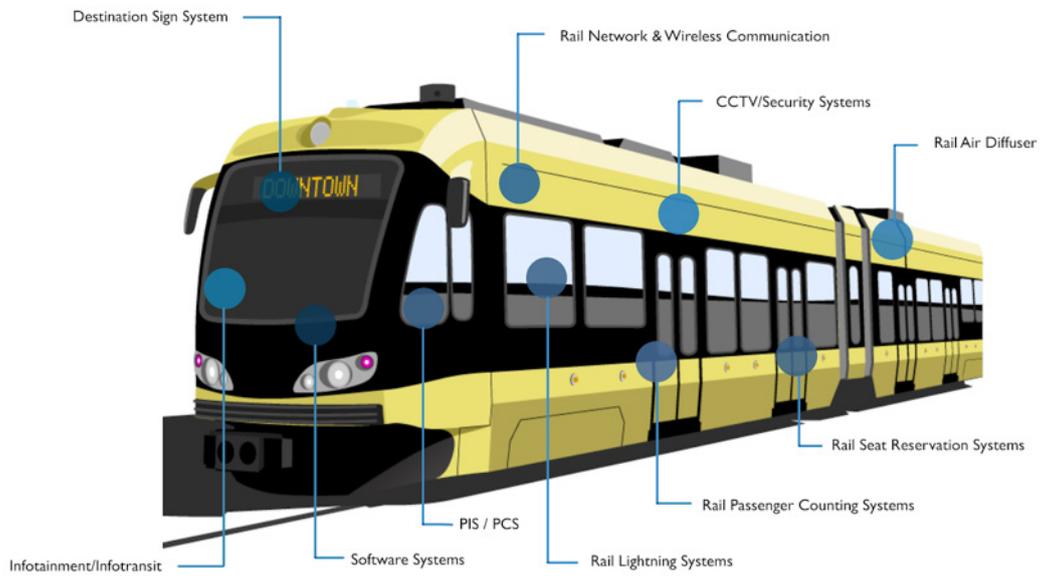


Figure 2.2 Visual representation of LTG products for rail system.

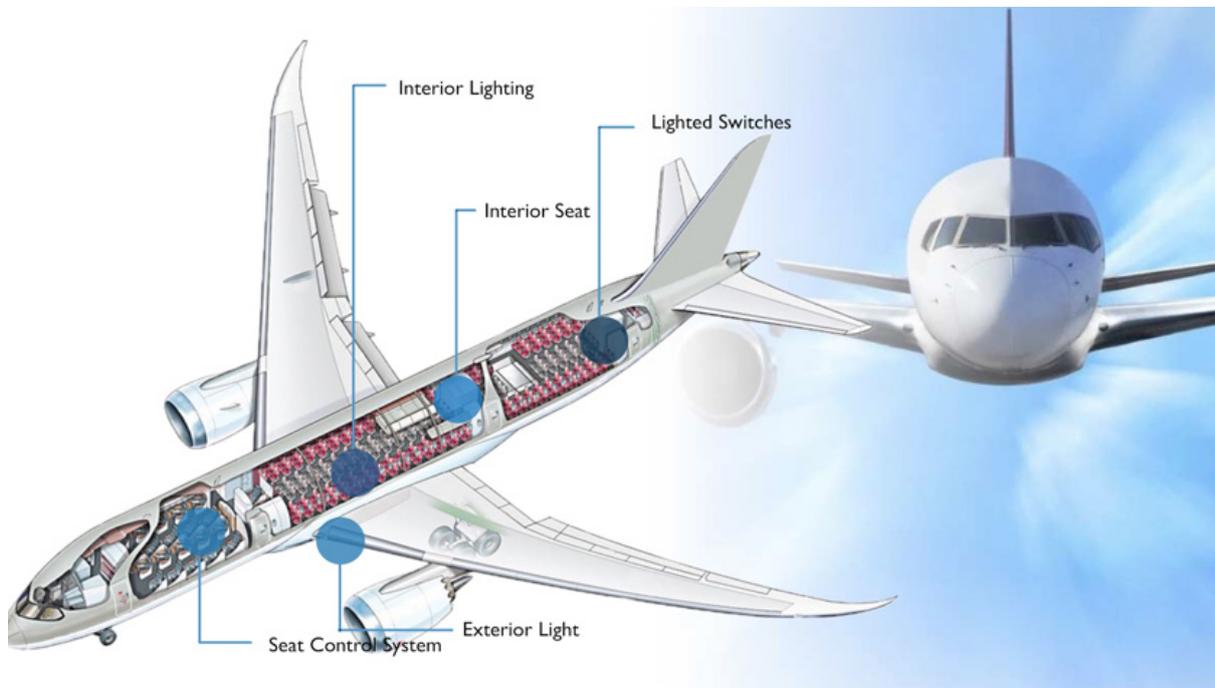


Figure 2.3 Visual representation of LTG products for fixed wing aircraft system.

LTG Suite

Luminator Suite (also called LTG Suite) is a web-based software application suite. It aims to include the main digital software needed to control LTG technology. At the moment LTG suite contains two software, Jasper for infotainment

content management, see figure 2.4, and Cinnabar for fleet management. The result of this project will be included in LTG Suite for destination sign editing purpose, thereby the new editor is expected to have the same look and feel as Jasper and Cinnabar.

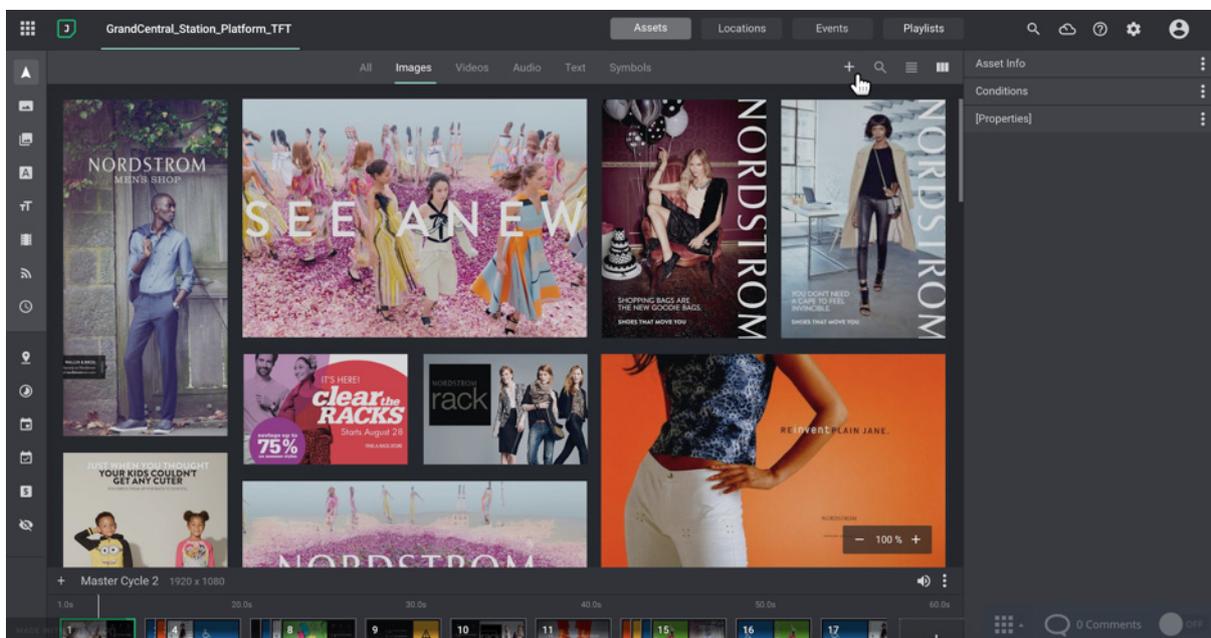


Figure 2.4 One example interface from Jasper in current LTG Suite.

2.2 Public Transportation System

Figure 2.5 presents the system model that describes the relation between LTG, their customers, public vehicles and passengers from a big perspective. The targeted customers of LTG can be grouped as PTAs (Public Transportation Authorities) who are offering the transportation services and PTOs (Public Transportation Operators) who are offering the vehicles. In Swe-

den, examples of PTA could be Västtrafik and SL, and examples of PTO could be Nobina and Keolis. Customers purchase signs, screens and related software systems from LTG. And these products offered by LTG will be installed in public vehicles, namely buses, trains, airplanes and other commuter vessels to guide and inform passengers throughout their journey.

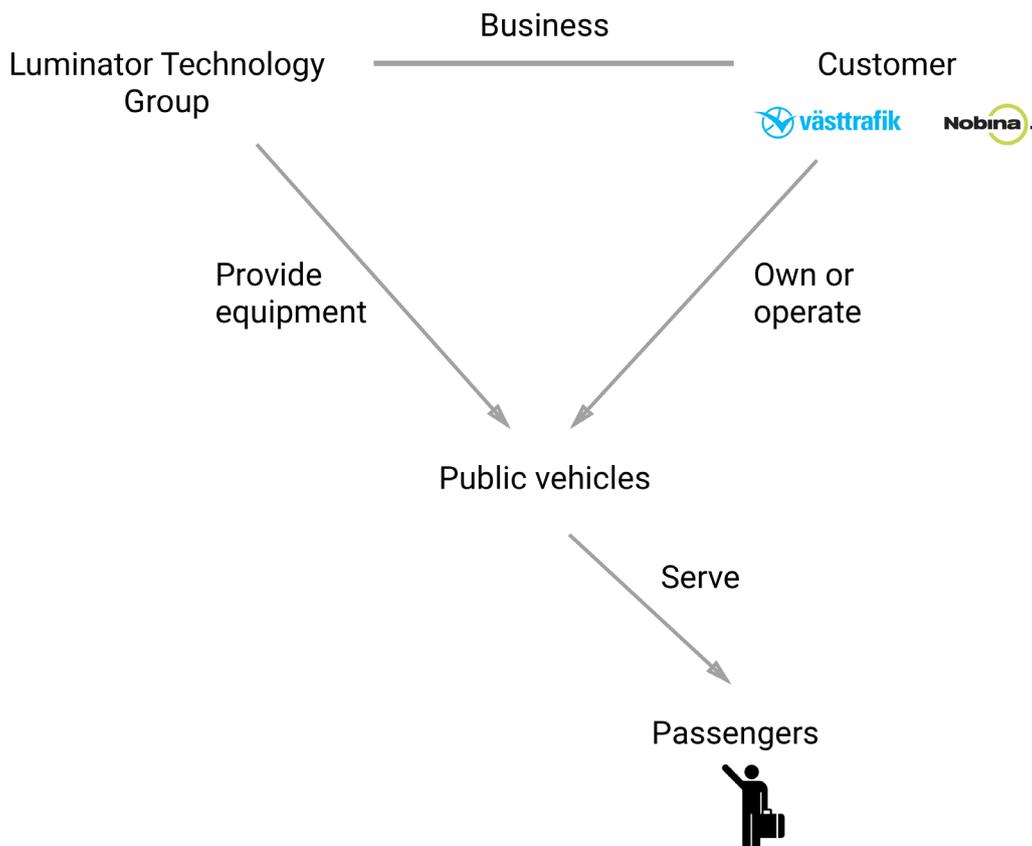


Figure 2.5 Public transportation system from LTG, their customers, public vehicles and passengers perspective.

2.3 System Products

All brands within the Group has their own signs and different software controlling them. They are not compatible with each other. Also, there are many other products that work together with destination signs and it is important to get a basic understanding of these products in order to use destination sign editing software. This section explains the relevant system products for this project, as figure 2.6 shows.



Figure 2.6 Sign setup in bus.

- *Destination signs:* Destination signs refer to the signs that are mounted on the front, side or rear of a public transportation vehicle such as bus, tram, boat and train. Typically a destination sign shows the route number and destination text.
- *Flip-dots:* Flip-dot signs are exterior display which are built up with dots which are reflective and yellow on one side and black non-reflective on the other side. The dots are flipped by electromagnetics, therefore the name. Compared to LCD and LED displays, flip-dot is a relatively outdated technology. However, due to the long lifetime of flip-dots signs many bus companies are still using flip-dots signs.



Figure 2.7 Flipdot sign.

- *Sign v.s. Screen:* In this project, signs refer to LED displays, LCD displays and flip dot displays that are assembled facing outside of the bus, indicating bus line number and destination. Screens refers to TFT (Thin-Film-Transistor) displays that have high resolutions and mainly used as interior signs showing stop list in buses. Compared to screens, signs are more robust, has longer lifetime and higher brightness, so are more often used for outdoor display. See figure XX for examples.
- *Control unit:* A unit which is mounted in the vehicle driver cabin. It is connected to all the signs in the vehicle and it controls their content.



Figure 2.8 Control unit ICU 602.

- *Sign resolution:* Sign resolution refers to the total number of pixels by width and height of a sign.

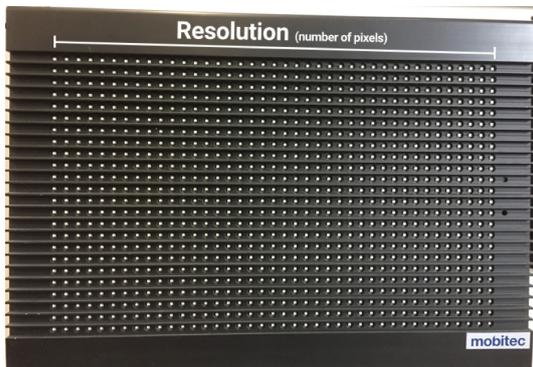


Figure 2.9 The resolution of a rear sign.

- *Pitch:* Pitch describes the density of pixel on LED displays and correlates with resolution. The pixel pitch is the distance in millimeters from the center of a pixel to the center of the adjacent pixel.

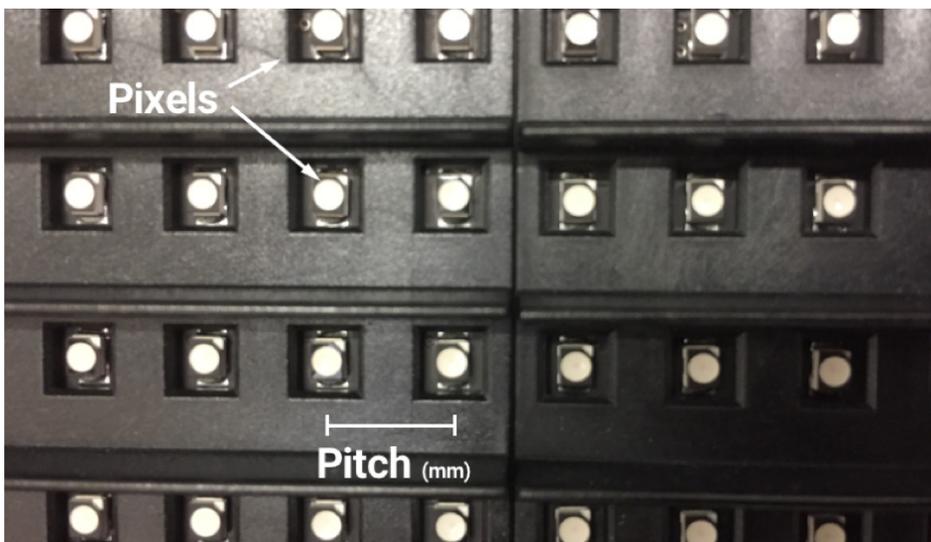


Figure 2.10 Close up of LED sign explaining pitch.

- *Outfile:* In this project, outfile means the exported file from destination sign editing software. All brand systems produce different outfiles containing different data, but all contains at least destination messages. After generating an outfile, it needs to be plugged into the control unit in the vehicle and the control unit will distribute the content of the outfile into each sign.
- *Address:* Address is what allows the control unit to communicate with signs. Each sign gets a specific address in the editing software, in this way, when the control unit sends destination data to all signs in a vehicle, they will know which content to take.

2.4 Software Description

This section presents the existing destination sign editing software in LTG separately. These software were studied in this project in order to find the similarities and differences between them in terms of workflow and main characteristics to pave the way for the design of the new editor.

MIE

MIE (Mobi Info Edit) is a sign editing software developed by LTG Sweden, which provides a dynamic way of editing text and graphics for Mobitec brand signs, see figure 2.11. Currently, MIE is a single user windows installed software, used by bus companies in many countries, including Sweden, Norway, Germany, France, Australia, US, Brazil, Singapore and Dubai. MIE is available in two versions, namely Basic and Advanced version. The main difference is the type of system the data is edited for.

The Basic version is used for editing destination sign systems, where all information is selected manually from the bus driver and does not change along the route. Typically this would be destination text and route number shown on destination signs faces to the outside of the bus.

The main workflow of MIE Basic version could be summarised as setup of buses and signs, edit sign content and generate out file. The out file will be loaded into the control unit of the bus which is connected to all the signs in the bus and send commands to individual sign based on sign address.

The Advanced version is used to setup data for a complete route information system including automatic vehicle localisation via GPS or tachometer signal. The driver has to start the route, all information changes will be initiated automatically based on bus localization and a preprogrammed route system. In UK they have recently introduced “Talking buses” which refers to voice announcements, inner and exterior sign displays controlled by GPS position, technology made possible with MIE Advance.

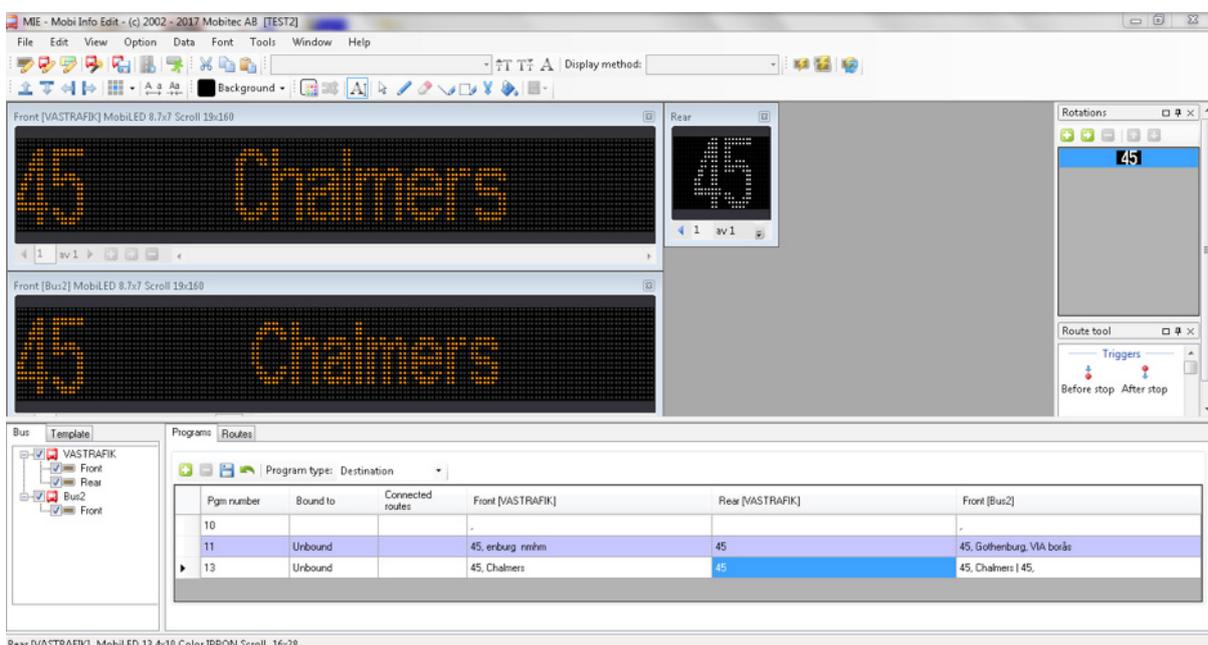


Figure 2.11 One example interface of destination editing software MIE.

MIE has some unique functionalities compared to the rest of the software within the Group. MIE is the only software within the Group which allows the user to input text directly in the preview signs. It is also the only one which is template based and can auto-format the text into the signs based on the template. It is also possible to import destination lists in MIE. High automation level and lower customization possibilities are the keywords for MIE.

TEDPlus

TEDPlus is a sign editing software developed by LTG Rastatt for LAWO brand signs. LTG Rastatt is a sister company of LTG Sweden, which is located in Rastatt, Germany. The main functionality of TEDPlus is similar to the Basic version of MIE, which allows users to edit destination signs, see figure 2.12.

The main workflow of TEDPlus could be concluded as configuration of buses, signs and fonts, edit of sign content and exportation of .mo1 file. The configuration process is a special process in TEDPlus compared to other editors. It is done by editing a custom .cfg file in text-editors where users could set the parameters for

the project, such as set the basic values of bus and sign, define font and define sign separation line. The configuration process is quite complicated and requires good understanding of the custom.cfg file, usually LTG technicians edit the configuration file for customers. Customers then need to import this file into TEDPlus. In this way, most of the customers could not setup their projects by themselves and for LTG technicians, this means they need to setup projects to customer wishes.

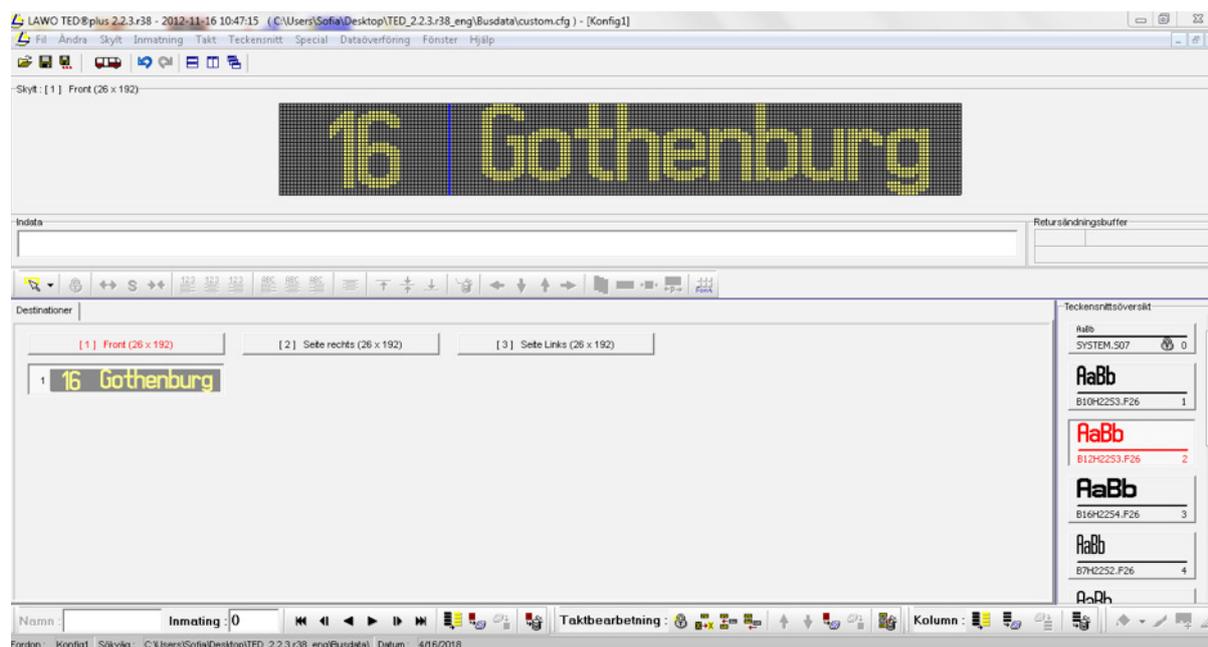


Figure 2.12 One example interface of destination editing software TEDPlus.

ALPHA-NT

ALPHA-NT is another destination sign editing software in the Group, developed by LTG Switzerland for Groba brand signs. LTG Switzerland is based in Oberbüren, Switzerland, see figure 2.13. Different from other editors, ALPHA-NT does not show an overview of the signs and buses, instead, the interface is pop-up window based, the five buttons from the top at the left side indicate the sections users need to operate to edit for destination signs. This layout gives a good guidance for beginners on how to do things in the correct order, however, in this way it is hard to keep track on all the information and ongoing tasks.

Different from MIE, ALPHA-NT has a input bar which allows user to input the text and then check and adjust effect on preview signs. ALPHA-NT allows perview one sign at the time.

The terminology system of ALPHA-NT is very much based on programming language, such as telegram, X/Y/Z offsets, block attributes and so on. This could cause barriers for users who have little programming background.

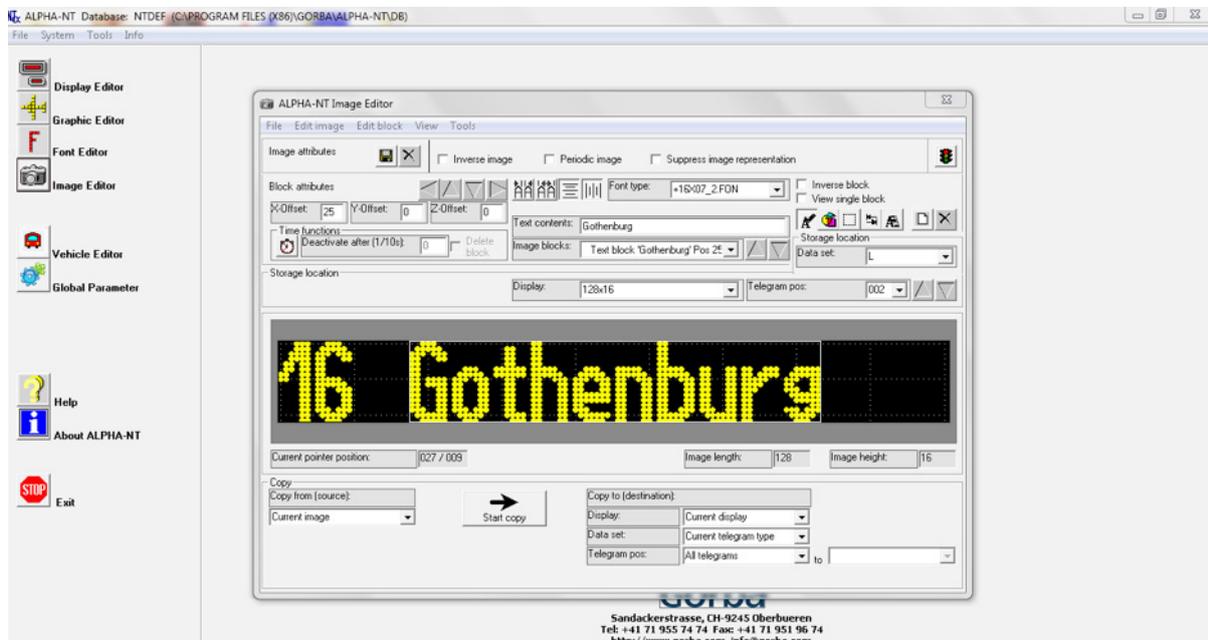


Figure 2.13 One example interface of destination editing software ALPHA-NT.

IPS

IPS is a destination editing software for Lumina-tor brand signs, developed by Luminator which is the headquarter of LTG located in Dallas, US. IPS is widely used by users in US, both internal-ly and externally, see figure 2.14.

IPS has a message listing function which allows user to input sign content in the list and apply it to all the signs. However, this feature is in rela-tively low automation level, in most situations, the content will not display well on preview signs, which means users need to check the ef-fect and adjust on each individual sign. At the same time, no template also gives users freedom to create everything they want to show on their signs. Customizable, flexible and much manual work are the main features of IPS.

Elyse

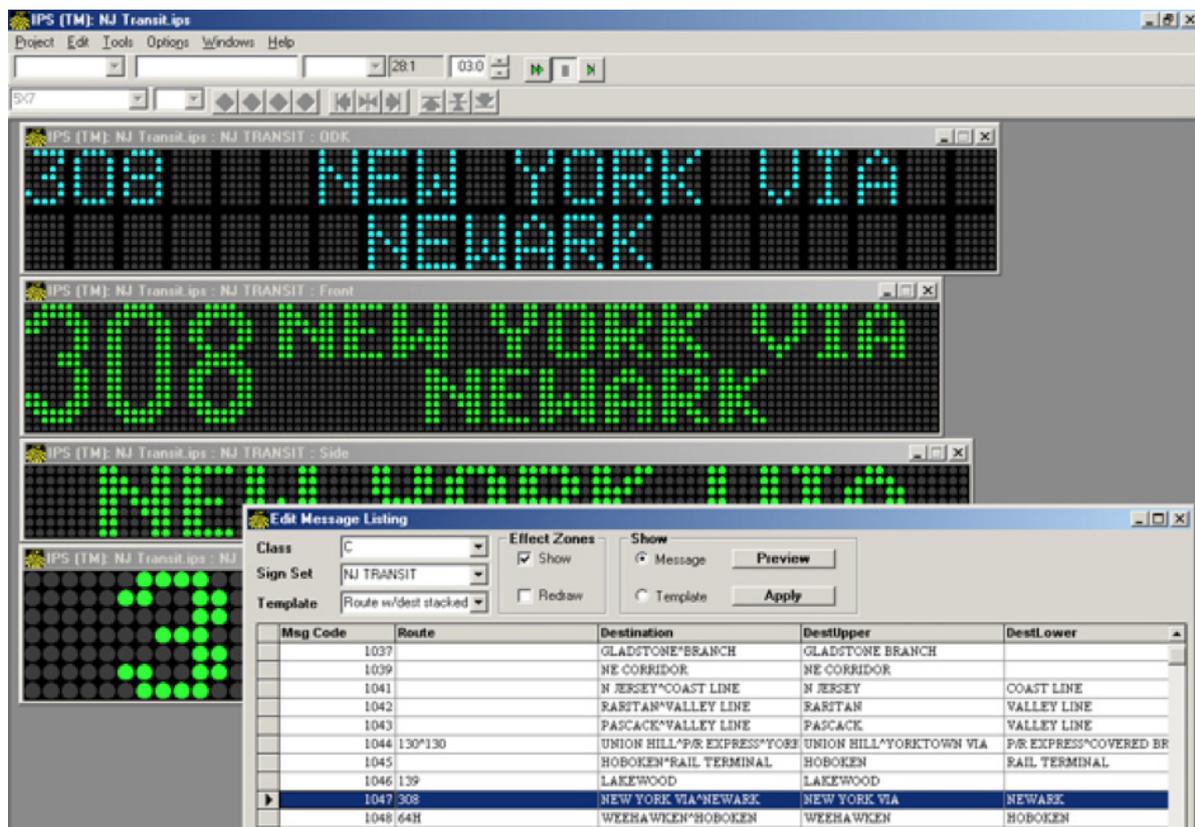
Elyse is destination sign editing software for Twinvision brand signs within the Group. Sim-ilar as IPS, Elyse is mainly used in US, by both internal and external users.

Compared to other software within the Group,

the most unique feature of Elyse is its two ed-iting modes, standard and freestyle. Standard editing mode is highly automated like in MIE, it allows user to input message and the software could assign the content into signs automatic-ly, in most situations the effect looks good and what users need to do is only minor adjustments. This helps users save much time and effort during editing. Freestyle mode allows user to do more pixel by pixel editing. At the same time, freestyle mode is not as smart as standard mode, similar to IPS, it requires much manual work.

icenter

The icenter software suite is a web-based mod-ular application system that manages all aspects of the passenger information system developed by LTG Switzerland. Destination sign editing is one part of functionality of icenter, but it can also handle infotainment screen editing and sta-tionary sign editing. icenter is relatively new among all the software in the Group, it has been developed for several years and the destination editing module is still under development cur-rently. Therefore, no user study was conducted with icenter users in this project.



18 Figure 2.14 One example interface of destination editing software IPS.

Software Comparison

Even though these software were developed by different sister companies originally, they were still designed for achieving the same purpose, editing destination signs. After learning all software and analysing them at a hierarchical task based level, see figure 2.15, it was found that all software have similar functionality and main workflow.

The main workflow of all software could be concluded as Setup, Edit, Export and Import, as illustrated in figure 2.16.

In Setup phase, users first select the database or the project they want to edit. Then, depending on the software, in MIE, ALPHA-NT and TEDPlus, users need to create vehicles and then assign signs to them by inputting sign size

and address manually. In software like IPS and Elyse, users create signs directly without the concept 'vehicle'. However, this way requires all signs in the same project to make sure every sign has different address. Signs are created by entering sign size manually in IPS and Elyse as well. Usually the setup of vehicle signs requires users to have good domain knowledge of bus signs and know what signs their vehicle fleet is using, e.g. LED signs, LCD signs or flip-dot signs, as well as understanding of terms like X-pixel, Y-pixel and pitch. For people who have little knowledge of signs, current setup stage could cause confusion and don't know where to find the sign information the software is asking for.



Figure 2.15 HTA process.

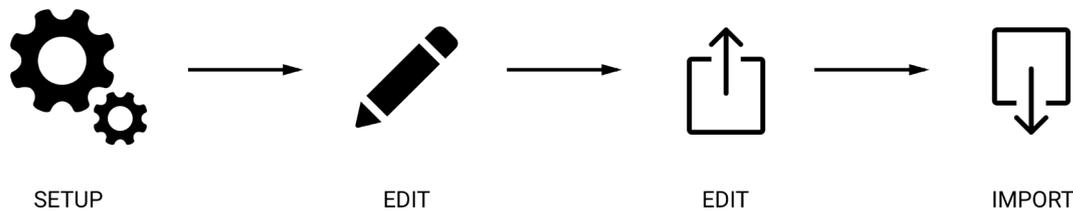


Figure 2.16 The main workflow of destination sign editing software.

In Edit stage, the workflow and functionality of editing are similar in all software, however, different software offer different ways for users to achieve the same task. Considering text input as an example, MIE allows users to input directly on the preview signs, while in ALPHA-NT and TEDPlus, users input in a text input area and then check effect and make position adjustments on preview signs. In addition, even though users could realize the same result with all software, the automation level of different software are different and lead to different time consumption. For example, MIE has unique features like fastinput and auto-formatting, which help users save much time and effort during editing, and make it smarter than other editors.

After editing, users are supposed to export an outfile that contains the information they just created in the software. In the export phase, users select which vehicle type they want to export as well as what control unit they are using in that vehicle, then save the exported file in USB and load it into the control unit.

As discussed above, different software have different key features and different ways of achieving the same things. Beside functionality, the terminology systems among the software are also very different. To give an example on this, the name of one text display, MIE uses the

term ‘program’, ALPHA-NT uses ‘telegram’, TEDPlus uses ‘entry’, while IPS and Elyse uses ‘message’. These terminologies are not coherent with each other, and require some time to learn and understand for the users who are using several software. The file formats of these software also differ from each other. Each software has their own format for database, configuration file and exported file. Finally, these differences made each software develop their own characteristics, considering MIE and IPS as examples, MIE has high automation level but not flexible enough when it comes to different customer needs, while IPS requires much manual work but allows high customization level.

Understanding the background and main characteristics of existing destination sign editing software in LTG is an essential step to evaluate them as well as generate relevant topics and questions to interview current software users to identify their needs for the new editor. Moreover, by comparing the software in terms of workflow, the differences and similarities between them gradually appeared which set a good skeleton for the design of the new editor.

2.5 Current and future ways of usage

There are two ways of loading the exported outfile from the destination sign editing software into the control unit in the bus, today's USB mode and future dynamic mode. The initial focus of the project was on the current USB mode, however, the stakeholders in LTG wished to expand the scope to involve future dynamic mode during the project. Therefore, future dynamic mode was studied as well to decide if it should be involved or not. The following section describes the two modes separately and discusses why future dynamic mode was decided not to be involved in this project eventually.

Current USB mode

The destination editing software are used to create an outfile that controls the messages displayed in destination signs. USB mode refers to the way of exporting an outfile on a USB which is commonly used today and capable with current sign hardware. The USB is plugged into the control unit in the bus and then the bus driver selects in the control unit which message to display on the signs. The passengers can then read the message on the sign. Below figure 2.17 illustrates how USB mode works in reality.

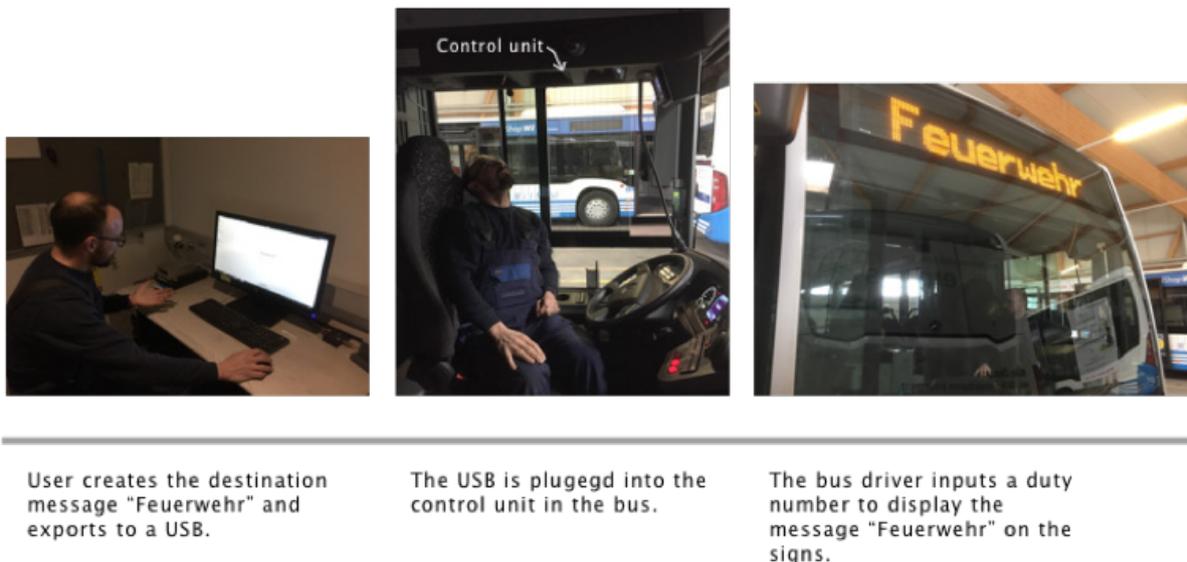


Figure 2.17 Detailed procedures for exporting an outfile in USB mode.

Future dynamic mode

Based on the development of sign technology, one future trend of destination sign is IP sign which means all signs will be connected with Internet in the future. In this way, destination signs will not require manual editing in editors, instead, signs could fetch real-time information continuously via Internet and GPS positions. When it comes to future editors, pixel by pixel editing and USB transferring will no longer needed, instead, the software should be able to support dynamic template editing where users only need to define the template and conditions

of signs.

However, IP signs are not commonly used today and still requires many years of development and implementation. Incorporating future mode in the scope also requires more research and studies on the users and market which is not feasible within the available time frame. Therefore, the focus of this project was decided to be the USB mode which requires users to generate an oufile and load into control unit manually. In this case, future dynamic mode will not be designed in the final prototype.

2.6 Market Usage

Different markets are using different software and are also using them in different ways. Market and culture has been seen to affect the user in their behaviors. Below are some pictures of destination signs around the world to demonstrate the variety.



Figure 2.18 Bus sign in Australia.



Figure 2.19 Bus sign in US.



Figure 2.20 Bus sign in China.



Figure 2.21 Bus sign in UK.



Figure 2.22 Bus sign in Brazil.

Market differences can be seen from:

- *Sign layout*
- *Effects*
- *Regulations*
- *Language*
- *Equipment*
- *US vs. Europe*

What can be seen from the images is that there is a large variety in for example layout and color and language. But what is actually market difference and what is usage difference? Layout is one thing that has a clear difference depending on where the users are located. In Australia and UK, they mostly configure their signs with line number on the right and destination on the left since they have left-hand traffic, see figure 2.18 and 2.21. The images from China displays the line number in the middle and on the left as well, maybe that is a usage difference because of no consistency can be seen?

The usage of effects, like rotation, scrolling and graphics, is often a result of both equipment (e.g. if the bus has a Mobitec ICU402 control unit you can't use scrolling on your signs) and regulations (see section below). However, it can also be market differences. From the user study, see figure 2., it was noted that for example Brazilian users like to use scrolling and Americans tend to be creative and often use graphics.

Another big difference is the language. All countries use their individual languages and the biggest difference lies in the Arabic and many Asian languages which are using non-latin characters. Other languages like French and German are using other special letter as á, ú and ü. Some countries display several languages on the signs, in Israel they using rotations to show the messages in Hebrew, Arabic and English.

The messages on the signs have certain regulations depending on the country. In US they follow ADA (American Disability Act) which for example put limitations on text size. One of the customers who was visited in the US, explained because they have a smaller sign size and they are not allowed to have two line messages, because then the text size would be too small. In Sweden it not allowed to have certain colors like red and blue on the signs and no scrolling and rotation when the vehicle is moving.

When it comes to differences between Europe and US, there are some different needs. In US for example, it is more common with dash signs (a small sign in the lower left side of the front of the bus for bus driver) and they also need to be able to have hexadecimals in the entry number. Different markets also use different equipment. In US they use Luminator, TwinVision, Mobitec and also competitor Hanover signs. In Europe there are LAWO, Gorba, Mobitec and Hanover.

03

PROJECT EXECUTION

The following chapter explains the execution of the project. First, the methods that were applied throughout this project are presented. Following this, the overall work process and method implementation are discussed with regards to the three main phases of this project: Understand, Create and Deliver.

3.1 Method Description

The methods that were used in this project are described in this section. These methods provided a theoretical and empirical foundation of this project. Depending on the purpose of each method, the procedures were adapted to the situation at hand. More implementation details and outcomes will be presented in work process section and appendix chapter.

Brand Map

The brand map is a tool to analyse the personality of a company or a brand. The analysis can be helpful for understanding the values and market position of a company. In a product development process, the brand map can be used to make sure the project outcome contributes and also pushes the brand into the right direction. It helps the designer to make the company’s branding promises to come true. The map is divided into two personality directions (extrovert and introvert) and five personality types (dominance, freedom, belonging, cautiousness and expertise) (Leidenkrantz, 2017).

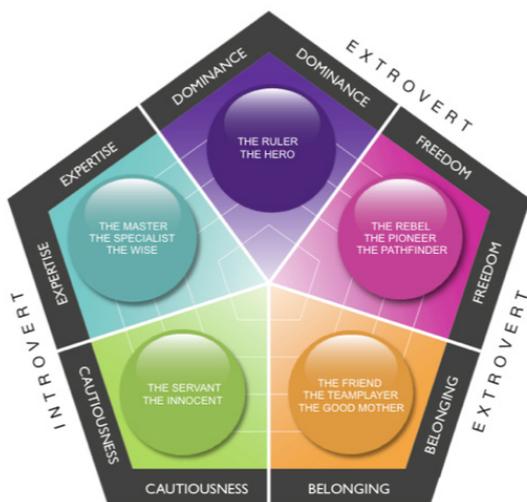


Figure 3.1 Brand mapping platform.

Hierarchical Task Analysis

Hierarchical task analysis, also called HTA, is a widely used task analysis method where a high-level task is decomposed into a set of low-level subtasks (Annett, 2003). In its most basic form, a hierarchical task analysis provides an understanding of the tasks users need to perform to achieve certain objectives. Hierarchical task analysis is often applied in human machine system design because it brings an overview and allows designer to explore various possible approaches to completing the same task as well as optimizing particular interactions.

Research Questions

In a research based project, research questions are the fundamental core of what the researcher wants to know or the questions they want to answer (Draper, 2004). It guides the decisions made during user studies to make sure the designer have the right knowledge and understanding of the problem in order to design an appropriate solution for it.

Interview

In qualitative researches, interview is a popular approach to collect in-depth information pertaining to participants’ experiences and viewpoints of a particular topic. It usually refers to a conversation where prepared questions are

asked to elicit information from interviewees. Depending on the format of the interview, interviews could be categorized into three main categories, informal conversational interview, general interview guide approach and standardized open-ended interview (Gall, Gall, Borg, 2003). Informal conversational interviews refer to spontaneous generation of questions in a natural interaction. The general interview guide approach is more structured compared to informal conversational interview. The questions asked during the interview are predetermined and worded, therefore, one obvious disadvantage with this approach is lack of consistency. Lastly, the standardized open-ended interview is extremely structured in terms of the wording of the questions. All interviewees are always asked the same questions, however, the questions are formed in the way that allows them to be open-minded. In order to collect effective and in-depth data, choosing the appropriate format of interview as well as good preparation is the core foundation.

KJ Method

KJ method is used to organize large amounts of data into groups based on their natural relationships. KJ method is often applied by designer to organize and make sense of data from user studies. This method is mainly conducted in 5 steps, namely gather information, display information, sort information into groups and create headers (Raymond, 1997). Following these steps helps to define the problems and develop potential ideas for solutions.

Persona

In user-centered design, a persona is a fictional character created to represent a user type that might use the product or service in a similar way (William, Kristina, Jill, 2010). Creating perso-

nas helps designers to understand users better in terms of their needs, experiences, behaviors and goals. The descriptions of personas are created based on the real data collected from multiple individuals, which is also beneficial in identifying patterns in the research, different user types and their expectations. Personas are especially valuable for communicating user study results in big teams or with stakeholders.

Use Cases

Understanding the needs of users is the core foundation to achieve user-centered design. Using use case is a good approach to define the needs of users and then lead the design process. A use case defines how users achieve a goal using the software. The documentation of use cases represents the needs of users, and how users are expected to interact with the system or sub-system to meet their needs (Usabilitygeek, 2016).

Prototype

A prototype is defined as an experimental model of an object built to test a design. Now it is widely used to perfect items and processes before implementing them on a large scale (Blackwell, Manar, 2015). It can provide designers with insight into the functionality of the design and reveal potential areas of weakness for further improvements. Generally, a series of prototypes will be made iteratively during the product development process, each with additional changes or adjustments, until the last prototype satisfies the requirements and expectations. In interaction design projects, prototypes work as mediating objects of the idea to let users interact with and provide feedback with regards to the experience.

Usability Test

Usability tests refer to evaluating the design by testing it with representative users. Typically, usability tests are carried out to let development teams identify potential problems before implementing the design. The earlier the issues are identified and fixed, the less expensive the fixes

will be in terms of the effort and cost consumption. During a test, participants will try to complete certain tasks according to given instructions. In order to run an effective usability test, a solid test plan including participant recruitment, scenarios, test tasks and instructions should be developed (Usabilitygeek, 2016).



Figure 3.2 Usability test from create phase 2.

3.2 Work Process

In order to establish a steady foundation for the process, the project was initiated by creating a process model. Together with the process model, a Gantt Chart was created which offered an overview of different activities within different phases, when they are supposed to start and end, and what deliverables should be produced at the end of each phase. See appendix 1 for more information about the Gantt Chart.

In general, all projects are unique and therefore require their own process, the same goes for this project. The process was developed for this specific project, however much inspiration was taken from other UX processes. As illustrated in figure 3.3, the execution of the the project was divided into 3 major phases, Understand, Create and Deliver. This specific process model was chosen because it contains a large user study to, in short, understand what needs to be designed and an iterative concept development phase with user feedback validating the progress.

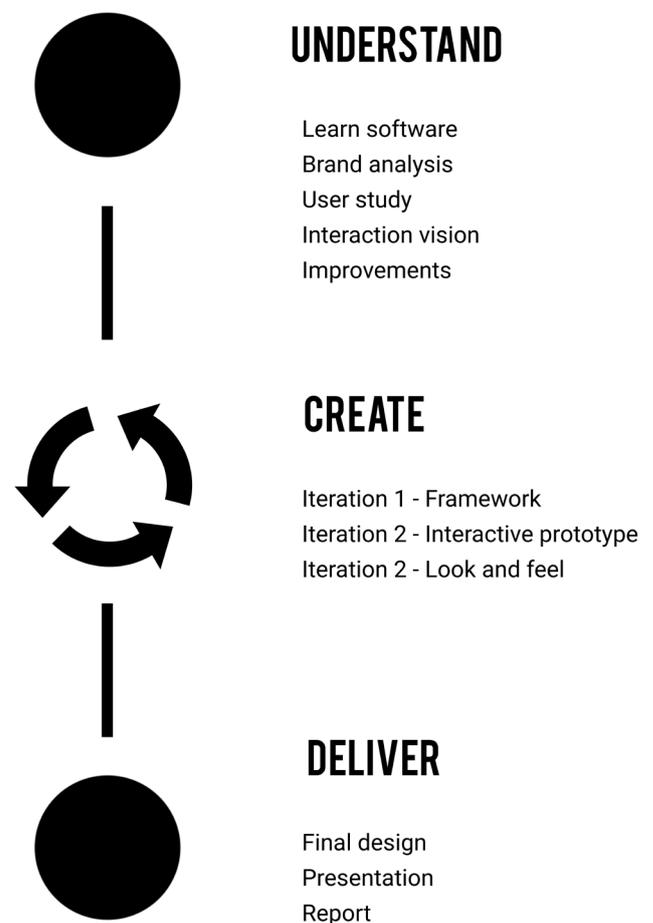


Figure 3.3 Visual representation of the work process.

Understand Phase

Understand is the phase where a thorough understanding of the software, brand, target user group, as well as their needs was formed. This phase was initiated by studying the context of the editing software (public transportation) and learning the software within the group. Tutorial sessions with expert users in LTG Sweden and sister companies were conducted to gain general knowledge of destination sign system and the usage of the software. In total, ten demo sessions were conducted where three of them were performed face to face and seven of them were carried out remotely. During each session, the user was requested to demonstrate the main functionality of the software and the most common tasks he/she does with the software. This study provided fundamental understanding regarding both the software and hardware of destination sign system and how they work together with each other, as well as the history and current usage of each editing software.

This was followed by a hierarchy task analysis of each software to understand what different tasks, procedures and in what order need to be performed in the software to achieve destination sign editing. The analysis was done on a general level because looking into the detail was not relevant when the project was meant to design a new interface with inspiration of the other software, not improve one interface. This analysis provided insights with regards to understanding existing software structures, identify users' primary goals and the steps users need to perform to achieve their goals. Even though the destination sign software involved in this project looked very different in terms of interface, however, by applying HTA to them the similarities among high-level subtasks appeared. The HTAs were therefore also used to compare the subtasks between different software as well as to identify the main workflow that commonly exist in all of them.

The main task of Understand phase was to conduct user studies and then based on the result, define their needs and generate a series of design specifications that could be used to guide the following design process. In the beginning of user research, research questions were proposed to guide the later investigation process. They were created by identifying the a range of questions that need to be answered in order to design a successful destination sign editing software. Four featuring different categories were then listed, namely domain knowledge, user types, interaction, and software evaluation. Within each category one or several research questions were generated to specify the focus points of that category, more details about research questions will be presented in chapter 4. Research questions were also used as foundation for generating following interview and questionnaire questions.

With research questions as foundation, empirical studies were conducted to map out the good and pain points of each software as well as the needs of different users. Because the software involved in this project are used worldwide, it was vital to gather information from geographically different user areas in the user studies. In order to gather feedback from both internal and external users from different markets, user studies was divided into three tracks; at company, remote session and abroad face-to-face visit, see figure 3.4. And the expected delivery of user studies were detailed software evaluation, answers to research questions and product vision for this project.

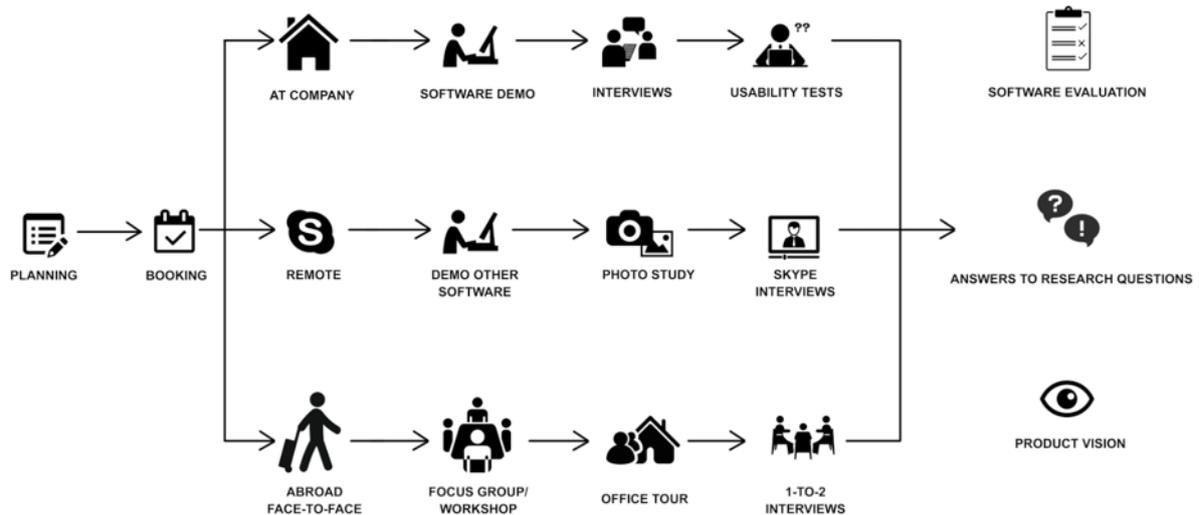


Figure 3.4 Visual representation of the user study plan.

At company

The track at company was carried out at LTG Sweden office in Herrljunga with mainly internal MIE users. Demo sessions were held with experienced users to gain a good understanding of the main functionality of the software MIE in a short time. This was followed by interviews with customer support and assisting editors who have deep knowledge of MIE and have been dealing with complicated use cases from customers. This track gave good insights about how the software MIE is used in LTG Sweden, how the software is evaluated by internal users and what feedback internal users received from external users.

Remote sessions

Besides MIE (LTG Sweden software), other software involved in this project were developed by other sister companies in LTG that are located in different countries. To learn these software remote sessions were conducted with internal users, mainly customer support, in each sister company (the input from external customers was covered in the abroad user study). These sessions also gave good understanding of their current experiences as expert users of

the software. In order to demo the software, users were asked to share their screen via remote communication tools. Following this, remote interviews were carried out with external customers to understand how the software is used in each company and how they evaluate their current experience.

Face-to-face visit

External users is an important user group that received much focus in this project. Therefore, it was essential to talk with them, see their working environment and know their opinion on the software they are currently using. Field trips were carried out in Germany, Switzerland and US for two weeks in order to meet with external users face-to-face, understand their needs to be able to design for them. This visit contributed good insights about who the users are in customer companies, the difference between internal and external users, how the software is used and special needs of different markets. Each of the trips started by experiencing local public transportation and sign usage, visiting sister companies, following by visits to 2 to 3 customers where they showed their business, how they used the software while answering interview questions.

Among all three tracks, 30 interviews were conducted with 12 internal users and 18 external users totally. The time set for each interview is 1 hour. The questions that were asked in the interviews are based on previous research questions, involving various topics of the participant and the usage of the software, general data and usage goal, first impression, teamwork, domain knowledge, domain documents, software function, local laws and preferences and software evaluation, see appendix 2 for full interview script. During the interviews, participants were also asked to open their software and perform a quick demo with one of their projects.

Among 30 interviews, 23 interviews were held face to face while 7 interviews were conducted remotely. Prior to remote interviews, each participant was required to fill in a questionnaire which includes general questions about the participants and the software they are using. Also, they are asked to send pictures of their working environment and screenshot of their recent projects. The pre-interview questionnaire, see appendix 3, contributed with valuable information about the participants and it was used to revise interview questions to better adapt to each participant. By doing this, participants are better prepared and knew what questions to expect in later interview session. This type of interview exacted valuable information that were needed to answer research questions in a structured way, and also allowed participants to express their opinions of current software and expectations on future software freely. The photos of participants' working environment and recent

projects were used as mediating objects, giving good clues about user types and their interaction with the software.

At the end of Understand phase when the user studies were finished, KJ method was used to group the data into different topic categories according to the research questions raised in the initial stage of understand, see figure 3.5. After sorting the notes into different groups, each category was discussed, keywords and key sentences were summarised in order to find clear answers to the research questions. In this process, KJ method and research questions were used as a combination, which made the KJ method application faster.

After applying KJ method, personas were used as deliverables of the user studies to acquire communicate better understanding of the different types of destination sign editing software users. Based on the results user studies, software users were grouped as three user types, internal customer support, external administrator and external mechanics. Personas were then created to based on different job responsibilities, personality traits and needs that are influencing the interaction between them and the software. For each user type, the representing persona covers general user information, does, wants, pain points, values and skills. It was a good way to illustrate the differences between different user types and also made the explanation of user types more tangible. The personas, how they differ from each other and how they will influence design process will be presented in chapter 5.

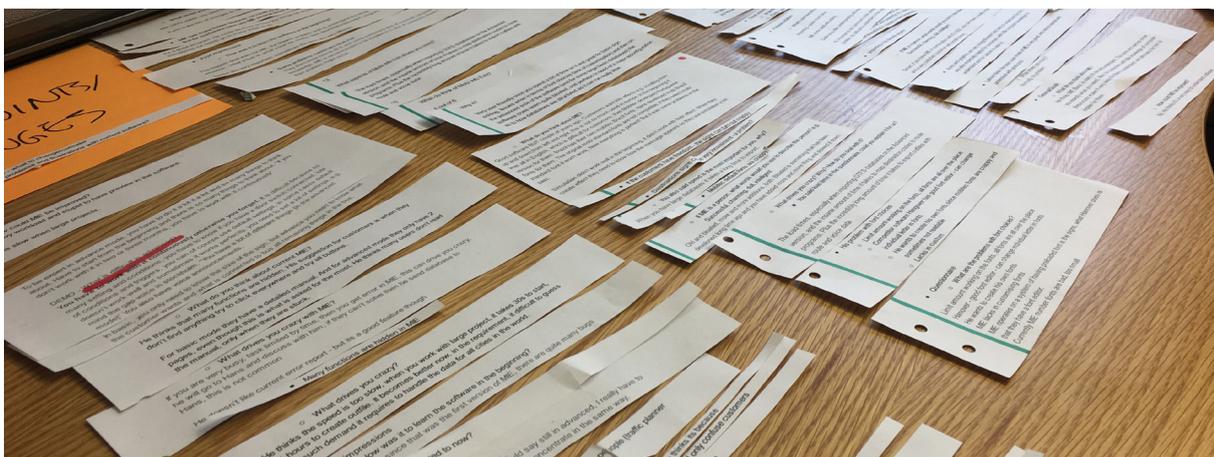


Figure 3.5 Sorting data from user studies with KJ method.

Create Phase

With the outcome of Understand phase as foundation, in the Create phase, the insights and knowledge iteratively generated a destination editing software concept. This phase was divided into three prototyping iterations with different focus. In each iteration the concept was developed further by adding more details and then tested with users to gather feedback on the changes. The continuous user input was conducted through interviews and usability tests both remotely (with users abroad) as well as in person (with user in Sweden mainly).

In each iteration of Create phase, design assumptions were evaluated to determine what worked well, what did not and how the software should behave based on user study results. Each assumption was composed in following structure: 'IF...' (design decision), 'THEN...' (certain effect), and ended up with a 'BECAUSE...' (motivation). All of the assumptions were stated in a subjective and falsifiable way. Further changes and refinements were made in the next iteration if some assumptions were proved to be wrong. Design assumptions and how they were validat-

ed will be presented in chapter 6,7 and 8.

Prototyping is a way to 'sketch with interactions' to create a rough way to test usability. In the Create phase, prototypes were used constantly to visualise different interface concepts, simulate user experience as well as to collect feedback from users and internal teams. Prototypes were produced as deliverables at the end of each iteration, three versions of prototypes emphasising different aspects of the software design were generated in total. For this project, creating prototypes was direct and natural since as it was a quick, cheap and iterative approach to visualize interface design ideas before implementation. Furthermore, interactive prototypes demonstrated the expected user experience to test users without much explanation, which contributed much in the usability test and evaluation stage.

Iteration 1 - Initial framework

The first iteration was focused on developing the initial wireframe prototype of the software which was composed of the main workflow, functionality and rough layout. The design was created based on high level use cases and design assumptions from the user study. The first version of the prototypes was low-fidelity paper prototype, which illustrated the initial architecture of the software with focus on functionality, main workflow and software structure. This iteration lasted for two weeks and at the end of this phase, the prototype was presented to users and teams in sister companies for validation.

Iteration 2 - Interactive prototype

In the second iteration, the focus lied on designing the interaction framework of the software on the basis of the prototype in iteration 1 and input from users. Literature study was performed in this iteration to support a good approach. The book *About face* provided insights on guidelines to follow in interface design process. It also gave clear instructions regarding how interaction wireframe should be developed step by step. By following these guidelines and instructions, an interactive prototype was developed by identifying posture, the software functional and data elements and determining functional groups and hierarchy. This iteration lasted for two weeks. Based on the previous prototype and the feedback gathered in iteration 1, a more detailed level prototype was produced to communicate the interaction wireframe of the software design. This prototype was created in the interface design software Sketch and Adobe XD. The prototype was also used in the usability test in this iteration, which allowed participants to click and perform certain tasks.

The usability tests were carried out based on

this prototype to let users try the interfaces and get a better feeling of the design. A sample of 5 participants were asked to perform a range of tasks with the prototype according to given instructions. Among them, 2 tests were conducted face to face and 3 tests were performed remotely by screen sharing (the prototype link was sent to the participants who shared their screen to show when they performed the tasks). All participants had experience with destination sign editing software because it was important to compare how the new concept stood against the current software. The purpose of this test was to evaluate the prototype in terms of several aspects, terminology, organization, discoverability, effectiveness, workflow and qualities. Participants were asked to perform 8 tasks in a determined sequence, and each task was composed of a scenario and task description that were constructed to suit their job responsibility, see appendix 4. A customer support would for example receive a scenario where he or she is supposed to help a customer. Furthermore, they were asked to think loud while performing these tasks to make it clear what they were thinking. A test documentation sheet was used during each test session which recorded the task completion status, participant comments, as well as errors, help requests and confusion actions participants made during the test.

Iteration 3 - Look and feel

The last iteration was emphasized on the look and feel of the software as well as refinements based on the input gathered in iteration 2. A detailed prototype with LTG Suite look and feel that covers the main functionality and usage scenarios of destination sign editor was produced. The iteration was concluded in two remote sessions where the final prototype was demonstrated to further validate the design.

Deliver Phase

The last phase, Deliver, of the project was spent on wrapping everything together with report and presentation materials. The documentation was done continuously during the project and was finalized in this phase.

04

UNDERSTAND PHASE

The following chapter presents the results of the initial stage of this project, Understand. In this stage, the understanding of LTG and destination sign systems was formed through brand analysis and learning of the software. Based on this, user studies were conducted in three tracks, in company, remote sessions and face-to-face visit with users of all 5 software to evaluate current user experience. During this stage, over 50 participants from 11 countries were involved to provide a global perspective of the project. This chapter summarizes the results of the three tracks according to four main categories, domain knowledge, user types, interaction and software evaluation.

4.1 Brand Analysis

A brand analysis LTG was conducted early in the project. The reason was to get to know the company, understand what values they have and what market position they want, By looking at their branding, this makes it easier to know how to design the project outcome to fit their product portfolio as well as how to improve their market position and brand personality. Below are branding from LTG (LTG, 2018).

Vision

Our vision is to become the preferred global provider of transportation information management, safety and security solutions.

Mission

Our mission is to provide the most innovative and reliable transportation solutions to protect clients assets, and promote efficiency and safety for public services.

Strategy

Our strategy aims to earn the trust of current and future clients through continuous improvement in our service. Support and solutions - driven by innovation, quality and ease of use.

Guiding Principles and Core Beliefs

- **Innovation:** We believe that constant improvement, continuity and forward thinking is vital to our success. We believe in evolving our focus and strategy when needed to achieve our goals.
- **Excellence:** We believe in providing service and solutions that exceed the expectations of our clients.
- **Value:** We believe our success lies in providing customer value that exceeds a measurable dollar amount through seamless technology advancements.
- **Customer focus:** We strive towards developing relationships where we understand and anticipate our customers need, both

locally and globally. our priority is serving customers directly or suppose those who do.

- **Diversity:** We embrace and open welcoming workplace for professionals of all backgrounds.
- **Ingenuity:** We believe honesty respect and professionalism.
- **Entrepreneurship:** We nurture as a sustainable organisation that fosters mutual trust innovation initiative and creativity
- **Engagement:** We depend upon participation from all, to create a culture where ideas that may benefit our customers and colleagues are welcomed.
- **Teamwork:** We leverage the experiences, expertise and developments of the various LTG member companies to increase value for our customers.

Brand map LTG

As illustrated in figure 4.1, the brand map of current LTG has DNA mainly based in Expertise and Belonging area in the brand map but also contain Cautiousness and Belonging. The size and position of the dots communicate its weight compared to the other dots. LTG identifies foremost with expertise because there is a great focus on delivering qualitative solutions which exceeds the expectations of the customers by innovation and ingenuity (therefore the Freedom).

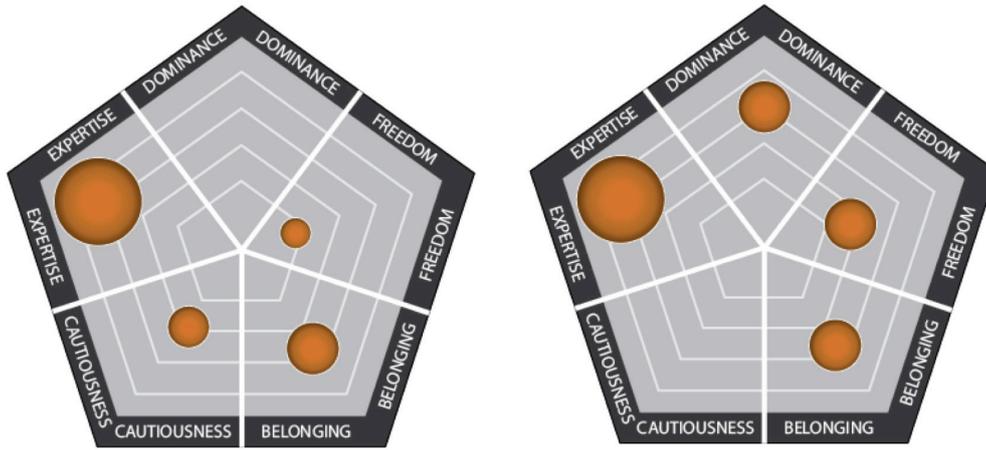


Figure 4.1 Current versus Desired Company DNA.

LTG also has great customer focus and is very service minded (Belonging). The position in cautiousness is due to the company's tendency to try new ideas but in secret. Based on LTG's branding material they envision a slightly different position in the brand map. LTG also has a strong ambition of being the preferred provider of leading global transit solutions driven by latest technology and entrepreneurship, and can therefore also be argued to belong in the Dominance and Freedom area.

Brand persona Mark

The brand persona Mark is created to communicate LTG's branding personality, see figure 4.2. The personality of Mark reflects the values of the company.



Figure 4.2 Visual representation of brand persona Mark.

Mark is 55 years old and a CEO at a large technology company. He works long hours but loves his job because it challenges him and keeps him on his toes. He lives in New York City with his wife and three children. On his spare time he values spending time with his family, fishing and playing golf.

Personality of the brand persona Mark

The expertise part of LTG personality can be described as competent and experienced. Mark has a lot of experience and likes to work focused with adapting solutions for specific customer needs. He is a specialist in his field of work. He is also dominant which makes him a self-confident and powerful leader who prefers to be the winner and in the center of attention. He also has a belonging personality with friendliness, honesty and consideration. He is someone to depend on.

Marks rational needs

As the expertise part of Mark's identity suggests, Mark requires to get the job done effectively with high quality. As a dominant personality, Mark has to be the best and deliver premium results in order to be ahead of competition. He wants to be in control. For Mark, it is also important, from a belonging perspective, to deliver user friendly and accessible solutions.

Marks emotional needs

When Mark works he has to feel like he knows what to do while feeling competent and in control of the outcome. Feeling successful and respected is important for Mark, which goes in line with his dominant personality. He also values feeling important and superior, he wants to feel like he is invincible and can't fail. Due to the belonging part of Marks DNA, he wants to be a part of a community and values collaboration and the relationships to his customers.

This brand persona Mark is a realistic representation of LTG attitude and values. By involving the brand analysis in this project, it helped designers to understand this brand better but also to make sure the later design process fits the big direction that LTG is heading towards.

4.2 Research Questions

In order to form a structured way to conduct the user study, a list of research questions were formulated in the beginning of user study. They were created as questions from the big perspective that support and direct the planning of the user study. Four main areas, domain knowledge, user types, interaction, and software evaluation were identified as the fundamental topics that were the most relevant to the design of a new destination sign editor. Within each topic, one or several research questions were proposed and were expected to be answered with the information gathered from the user study. Following chapters provides answers to the research questions based on the user studies.

Domain knowledge

What do the users need to know to use the software?

User types

Who are the users?

What are the goals and motivation for using the software?

What are the differences between user types?

Interaction

When and how the software is or will be used?

What basic tasks does the user do to accomplish tasks?

How is the current interaction between the user and the software?

Software evaluation

What are the problems and frustrations with current software?

What are the good points with current software?

How can the software be improved?

4.3 Domain Knowledge

In order to use the editing software, first of all, users need to have basic computer skills and good understanding of how bus system works in the city. Besides this, knowledge of destination signs such as sign brand, sign size and how signs work together with control unit and bus driver is also required. During the setup phase, if customers setup the signs by themselves, then some basic sign knowledge is also required, like

resolution, pitch, address and colour. Figure 4.3 is a control unit developed by LTG Sweden. More information about system products and terms could be found in chapter 2.

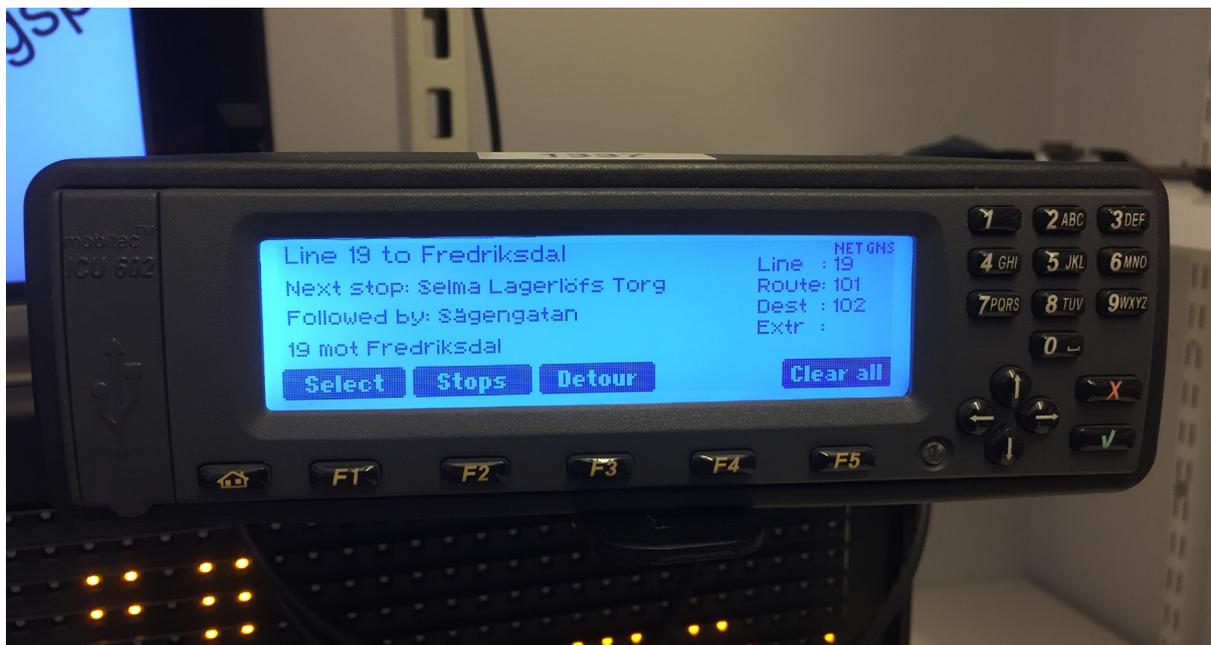


Figure 4.3 Control unit 602 developed by LTG Sweden.

4.4 User Types

Five user types can be seen among the users of LTG software and they can be divided into two groups: external and internal users. External users are those who work for companies that use LTG equipment, often public transport authorities having LTG signs on their buses. The external users can be divided due to their responsibilities and background: Administrator and Mechanics. Internal users are those who work for LTG: Customer Support and Developers. There is also a fifth user type called Assisting Programmer who can be both internal and external user. The primary user types are the Administrators and the Mechanics. The secondary users are the Customer support and the Assisting Programmer because they only have a supporting role of the external users. The Developer is a tertiary user.

The user types differ from many perspectives. For example, they differ by background, personal motivation and values. First of all, all users speak different languages and the signage comes in for example European, Arabic and Asian languages. All language groups are using different characters. There are other trends which could be seen as well when it comes to motivation. The external customers are motivated by producing accessible signage which the passengers appreciate and is profitable for their company. There is a lot of varieties in the level of ambition among users. The users with little

computer skills sometimes only care about “if it works” compared to some users who values great customization and creativity possibilities. The customer support in Texas describes her users: “*Control is very important for the super users who are very capable, often big companies*”. The customer supports on the other hand, are motivated by happy customers and solving daily challenges when assisting the customers whenever they have issues. Customer support are working with several cases in parallel, customer support in Sweden said: “*I’ve never experienced no open issues*”, which is the likely reason why they are motivated by completing tasks. They have very high expectations on the software and value efficiency, precision and functionality in order to assist the customers. They tend to be social and service minded.

External users

Administrator

Administrator is the user type who works for external companies with destination sign programming and administrative work. This user group tends to work for larger transit authorities with 50 buses and over. They use the software a few times per year, typically when there is a schedule change or when they get a new bus type to their fleet. During these times they use the software more frequent, the rest of the time they spend on administrative tasks. Typical tasks the Administrator does with the software is therefore making changes in the message listing, removing or adding destinations and adding vehicle types or vehicle equipment. This user type often has good computer knowledge and editing skills (depending on experience) and less technical skills. They often work in office based environment, as shown in figure 4.4.

Mechanic

Mechanics are those who are responsible for both editing of destinations and maintenance of the bus fleet. They often work for smaller transit authorities. They interact with the editing software, like the Administrators, when there is a schedule change or new buses and equipment. Since they often work for smaller authorities, they gain new buses less frequent than big authorities, they might interact with the software less frequent than the Administrators. The Mechanics often have low computer skills but very high domain and system knowledge and they could use the system to test the signs as well. Because of their low computer knowledge they sometimes struggle with the editing work and can have a low ambition level of the result. They often do their editing work in a office in close relation to their workshop, see figure 4.5.

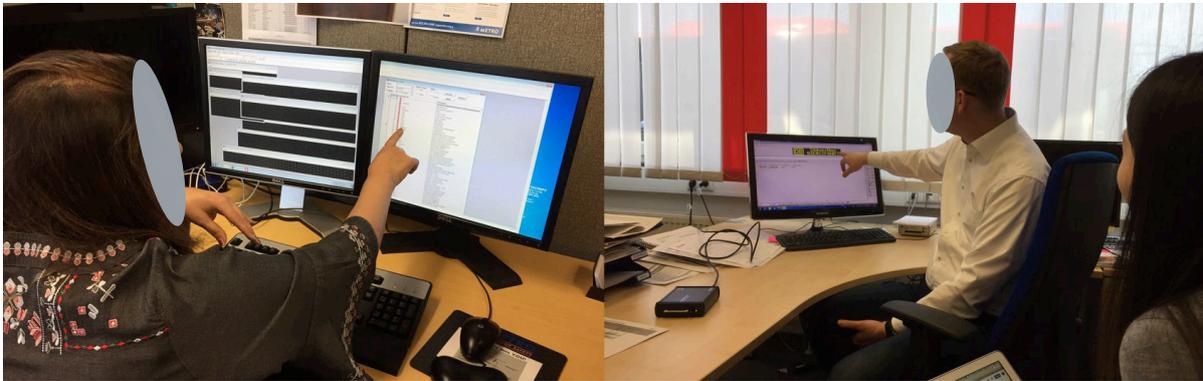


Figure 4.4 Examples of external administrator users and their working environment.



Figure 4.5 Examples of external mechanic users and their working environment.

Internal users

Customer support

Customer Support is responsible for educating and assisting the customers whenever they have issues with LTG destination editing software. The position can be combined with after sales technical support (also repairing equipment for example), but not always. The Customer Support has expertise knowledge of the software from using it several times a week and helping customers with wide variety of use cases. They also have good domain knowledge. Common tasks for the Customer Support is setting up new customers who bought LTG signs and receive calls for customers who have various problems. A common problem for the customers is exporting their outfiles. The help can be done both remotely via email, phone or remote controlling programs like JoinMe and TeamViewer and also in field. They can also do programming work for customers. The Customer Support puts high demand on functionality (because they have to handle all use cases in order to help customers) and efficiency (because of the amount of work they perform with the software). Depending on what equipment their customers are using, they often have to handle several software. Customer Support in US must support MIE, IPS and Elyse for example. They can have various back-

grounds, but those involved in this project have backgrounds such as maintenance and engineering.

Developer

The developers of the software are based in LTG offices. They are responsible for developing and making updates to the software and sometimes also supporting work. They are the super users to which for example the customer support turns if they have problems they can't solve by themselves. They generally handle the trickiest use cases. The developers have very good computer skills and software skills.

Assisting programmer

Finally, the last user type are those people who support the external users by doing editing work for them. They can assist both small and big authorities. They can be both internal and external users. They have high software skill and can also be considered to be the super users. Figure 4.6 shows some examples of internal users and their working environment.

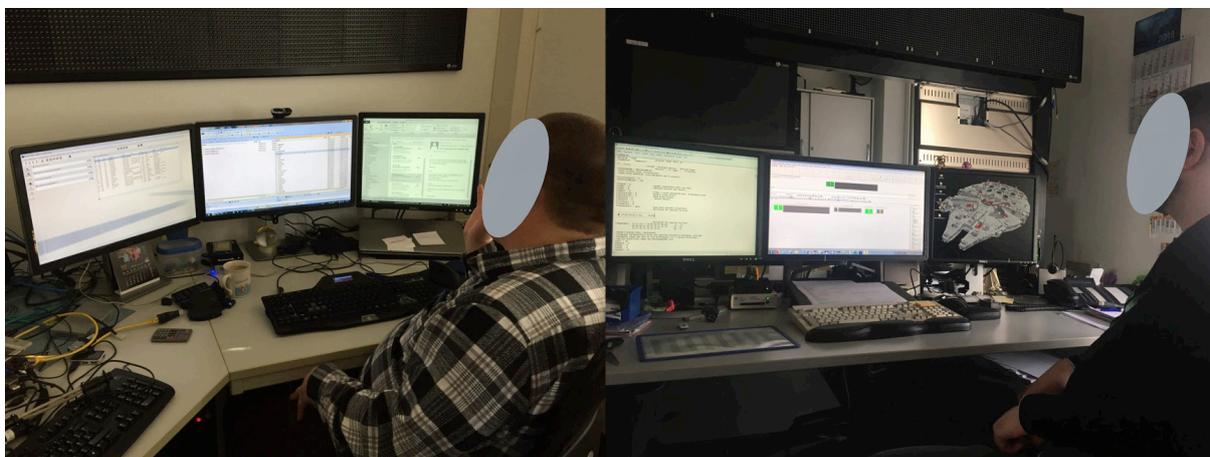


Figure 4.6 Examples of internal users and their working environment.

4.5 Interaction

Following section describes how different user types are currently using the software, based on the output of the user studies.

Collaboration and documents

Before editing signs in the software, the users have to know what equipment their vehicle fleet has. They also need to know what destination information should be on the signs. This information is produced by a traffic planner and the document normally includes bus line, destination and route information. In some cases, the bus companies are big enough to make their own planning, in other cases, there is a third party traffic planning institution doing planning work. The document can come in many different formats, but is often in Excel. Some more unstructured companies even give paper notes to customer supports so they can program the destinations for them.

Doing editing work also involves collaboration between different parties. As external users in bus companies, if there is any question with signs or editing software, they could always seek help from LTG customer support team. Feedback from bus drivers is another important way for users to check and improve their work in the software continuously. Bus drivers have very close contact with passengers and can gather their feedback regarding the content, readability and layout of the signs, and then pass on their opinions to editing people.

Customer support usually gets emails or phone calls from customers describing their problems. If customers need guide on the usage of the software, then customer support uses remote control software like Teamviewer to give remote education to customers. For customers who want customer supports to set up new projects for them, then LTG technicians ask for their bus and sign information, such as bus type, sign serial numbers, destination list, GPS coordinates, fonts requests and time schedules. This information can come in different formats, such as Excel, paper notes, text files and emails, largely depends on the preferences of customers, see figure 4.7. Customer support have very good knowledge when dealing with most use cases, however, when it comes to very complicated use cases, they also need to seek help or collaborate with internal expert users or developers to help customers. Handling the special needs of Brazilian market could be considered as one example. When MIE was first introduced to Brazil, many features didn't work well or didn't suit Brazilian customer preferences in the beginning. Then internal expert users and developers in LTG Sweden helped them and developed a special version of MIE containing special fonts and special effects for Brazilian customers to suit their needs.

For internal users, their main workflow regarding the sign editing could be summarised as, first of all, get requests from customers via several ways such as phone calls or emails, understand their needs, and figure out if it is hardware or software problem. Then, if customers' problem is about the usage of the software, ask for their sign types, bus types, sign resolution or serial number and use remote control software to control their computers to check problems or ask customers to send their databases to LTG and check for them. When it comes to usage habits, internal users usually don't have any obvious preferences. Their main purpose is to help customers solve their problems, therefore, all of their settings in the software are based on customers' requirements or preferences.

External users

For external users, programming destination signs in the software usually is only a small part of their job responsibility. Generally, their interaction frequency with the software is very low. Most of them only interact with the software a couple times a year when the bus company purchases a new bus or there is any route adjustments. They deal with only one or a few databases and setup new project, new signs or new buses very seldomly. Their main tasks in the software is to maintain or update existing database if there is any changes, add, remove or change bus signs and types.

Compared to customer support or internal expert users, external users can have shallow or basic knowledge of the software. Many of them prefer to do their tasks step by step following the demo LTG technician gave to them in the beginning, and not willing to spend extra time on exploring the software by themselves. However, the skill level varies a lot between users. Some reasons are age of the user, experience and interaction frequency.

Due to the long lifetime of LED signs, many bus companies are not only using signs from the same brand, but several sign systems from different brands, especially in US market. For external users, this means they need to use several editing software to program for different sign systems at the same time. For instance, one bus company which was visited in US are using Hanover signs, TwinVision signs and MIE signs together in the same bus fleet, so users in that company have to edit the same content in three software individually, which leads to much repetitive work for them.

For external users, their main workflow in the software is: firstly, get education session from LTG technicians, this session could last from one to several hours, could be either remotely or in field. Secondly, depending on which LTG company and which software they are using, some customers get already set up project files from LTG, some customers get blank project files, then they start to edit signs by themselves. After editing, then they need to export out files, save files in USB and push USB into each control unit in the vehicles individually. However, MIE and MAW software users have one more option, they could also use MAW to distribute out files into each control unit remotely. Finally, after this step, customers make continuous updates of their databases.

Their usage habits of the software also differ a lot. The main influencing factors are market differences, personal preferences and what software they are using. Market differences include languages, law and regulations as well as local conventions, these could influence what and how the information will be displayed on the signs, see market usage in chapter 2. Regarding law and regulations, one example could be in Nordic countries, the layouts of bus signs are relatively fixed to only showing line number, final destination and via station, as shown in figure 4.8 below. Usually, effects like scrolling or rotations are sparsely used. In Sweden are buses not allowed to have scrolling texts on moving

buses. However, this situation is very different in Brazil market for example, users there like to use scrolling, rotation effects and can be very creative as they don't have any regulations regarding this. In US they have ADA (American Disability Act) which limits their usage habit to certain text sizes for example. Besides laws and regulations, the usage of effects, fonts, sizes, sign layouts and hotkeys are largely depend on the user's personal preferences.

As mentioned before, depending on the sign set in the buses (from which company within LTG the signs in the buses are equipped with), the user has to edit them with different software. Some companies, especially in the US market, companies tend to have several signs sets and have to interact with several software. A common case with the customers visited in USA was that the same company had between 2 to 4 different sign sets, for example IPS, Elyse, MIE and competitive software Helen.



Figure 4.8 One example of destination sign usage in Nordic countries.

4.6 Software Evaluation

After analysing data from the user study trip, individual theoretical study of the software and the usability tests, the evaluation of the software can be divided into good and pain points and potential improvements. Figure 4.9 illustrates the most significant good and pain points of individual software, see appendix 5 for more detailed information about evaluation of individual software.

Good points

After visiting the users and interviewing them in person and remotely about their opinion of the software, it was clear there are a lot of both positive and negative points. Many of the users have used the software for a long time and learned how to use it. Those users like the software because they can get their work done with it and appreciate the simpleness of many of the software once you have learnt it. Other users are new to the software and have problems understanding the concepts. All the software have different functionality and combined there is a lot of valuable functionality within the group. For example MIE is very efficient with functions like import of data, fast input, template and auto-formatting. We also have the precision and allowance of creativity in IPS and a good overview of the workspace in TEDPlus and MIE.

Pain points

When it comes to pain points of the software, many of the users struggles with following points:

- Old software
- Repetitive work
- Low automation level
- Seldom use
- Unused functionality
- Complicated settings
- Long learning curve
- Low computer knowledge
- Acceptance to new editor

Especially users on the west coast of US expressed concerns with the software in terms of its age. With a 20 years old software, how can they know for sure that it will be compatible with their new signs for ten years ahead? Some of the software have even stopped developments like Elyse and IPS. Old software can therefore lead to customers choosing competitors. While some customers have seen that they are in need for a new editing software, some customers are more conservative and prefer to stick with a software they know and learned to appreciate. Therefore acceptance for a new editor can potentially be a problem if users don't see the new value it can contribute with.

One of the biggest pain points today is the repetitive and inefficient work. There are several reasons behind this, for one, many of the users are using several brand sets. Currently you can't edit signs from different brands in the same software which means that you have to learn several software and do the exact same editing several times. Depending on the software you have to edit the same listing several times within the same software as well. To give an example on this, MIE is structured with having one message listing per bus type, which means if you have two bus types on the same route (and therefore the same listing) you still have to create that listing twice. Depending on the software there is also much inefficient work because of low level of automation in all the software except for MIE and standard mode in Elyse. In the rest of the software the user has to input and position text as well as choose font and size manually for each message. It requires time and tedious work. Another problem with the software is that they aren't always intuitive to use. Some examples of this are the issues of unused functionality because of hidden features, high support from Customer support when setting up the program and long learning curve. During one of the customer visits in US, it was discovered that the customer was very new to MIE and suggested as an improvement in MIE that more editing could be done by itself. The user was informed that MIE has auto-formatting and import features and the comment was "That blew my mind!". After that he couldn't come up with more suggestions for improvement. That was one example of how the software aren't intuitive enough to make it easy to find and use valuable functions. Another example would be ALPHA-NT, which initially looks quite complicated and use unfamiliar terminology. When asked about the setting up process of projects for customers, the customer support in LTG Switzerland told about that some customers ask if they can get access

to the software and learn by themselves. They usually respond that they can try but it always result in them getting back quite soon asking for guidance. This is one example of many of the setup processes of the software rely of the customer support rather than the end user. Another software with a complicated setup process is TEDPlus, where the setup process consists of making changes to a configuration file which is written in code, practically impossible for customers to setup and make changes to.

Many of the users use the software seldomly, some use it as seldom as once a year. This puts high demands on the reusability of the software, especially because of many of the users have lower computer skills. Some customers say that to make it easier to get in to the software when it's time for the yearly schedule change, they practise a little throughout the year.

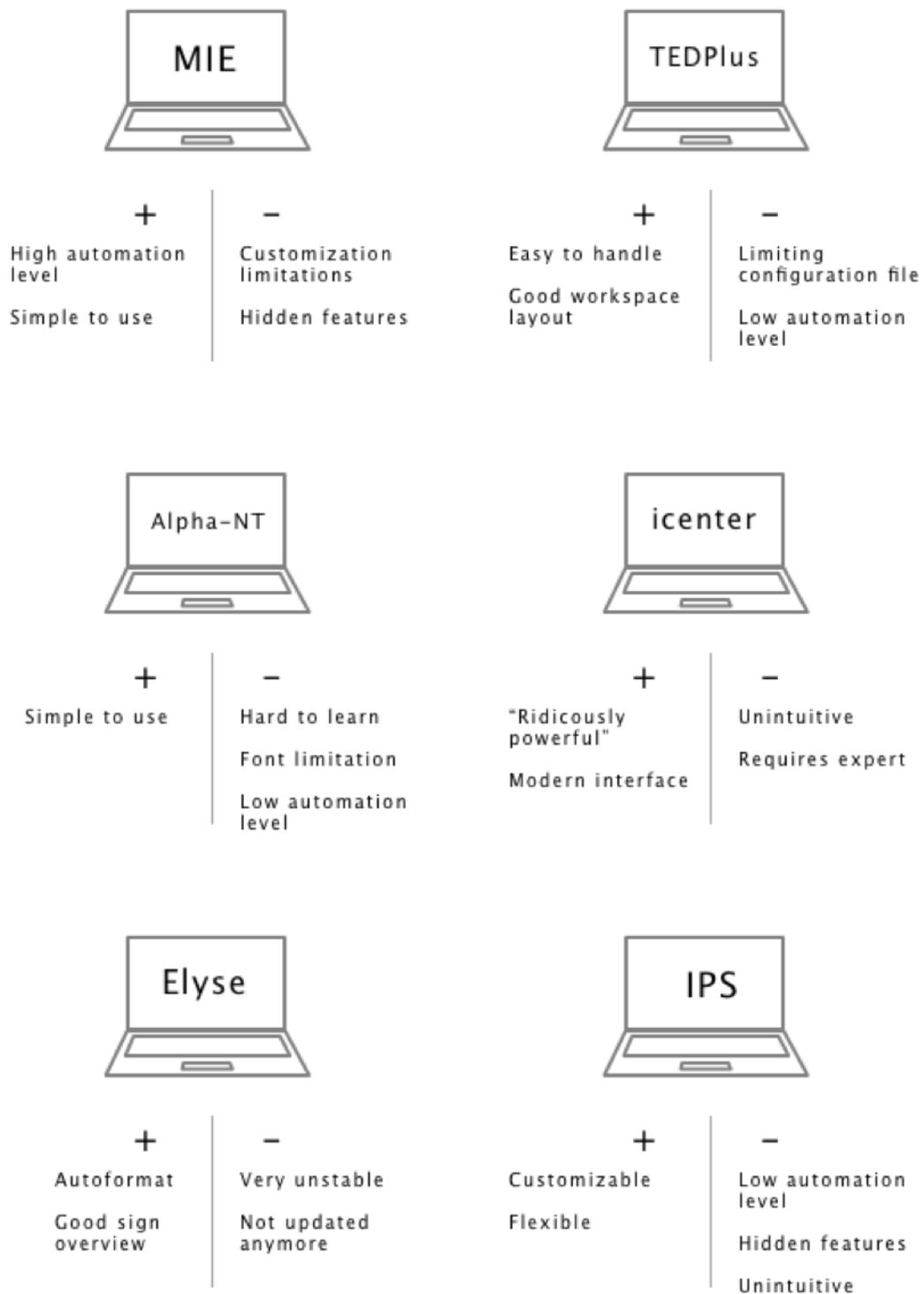


Figure 4.9 Visual representation of the evaluation for each software.

4.7 Conclusion

The Understand phase gave understanding of the current editing software within the group and their users. Following list concludes the research questions for the Understand phase:

Domain knowledge

To be able to use a destination sign editor software one should have basic computer skills, have understanding of sign technology systems and terminology.

User types

Users of LTG destination editors are spread globally by PTAs and LTG technicians. They can be divided into three main types: Mechanics, Administrators and Customer supports. They all want to create destination messages to guide passengers. They differ by software skills, experience and ambition level, functionalities requirements and use frequency. The three user types represents the entire spectra of low skill to

expert users.

Interaction

The main reason why external users are using the software is a few times a year when the bus schedules are updated. The necessary interaction steps are setting up the vehicle fleet, create destination messages and export the content into outfiles.

Software evaluation

Currently the software systems within LTG aren't compatible with each other which means some users have to learn and work with several software. The current software are very unmodern but have valuable functionality like auto-formatting and pixel-by-pixel editing. However, user are in need for a new software with higher level of automation which reduces the repetitive work in many of the current software.

05

PRODUCT VISION

Following chapter describes the guidelines which were produced based on the user studies. The guidelines were meant to support the Create phase when developing the software prototype. They contain three personas representing the users of current software, the high level use cases the software should be designed for, improvements to solve pain points from the user studies and an interaction vision. This chapter describes what to design in order to meet user needs and solve current interaction pain points.

5.1 Personas and high level use cases

Three personas of the primary users were developed to summarize and make it easier to design for the real users during the development process. The three personas are: Rob the Fixer, Admin Mary and Support Carl. They can be considered as beginner, intermediate and expert user. They are described below along with representative examples of how these personas would use the software in real world. These examples are collected during the user studies and make up the most common things which the users would do with an editing software on a high level. The high level use cases are:

- *Buy a new bus with LTG signs for the first time*
- *Update message listing due to schedule changes*
- *Make changes to bus fleet and update message listing*

These high level use cases describes how the personas would interact with the new editing software. They tell a story of who is using the software and why. The high level use cases are not unique for the persona it is presented with. In the section below, each persona is presented with a unique high level use case, in reality can the personas perform all of these high level use cases.

Buy a new bus with LTG signs for the first time

Robert Müller (60, married, three children) lives in Oldenburg, Germany. At his work at Olden-Bus GmbH his nickname is “Rob the Fixer” because he is responsible for maintaining their fleet of 25 buses. If there is any problem - he will fix it. Rob has worked there for the last 30 year. When growing up he he spent most of his spare time in his father’s car repair shop, developing mechanical skills and a genuine interest for classic cars. Today he is a proud owner of a 60’s Porsche 911. His persona profile is presented in the figure 5.1 below.

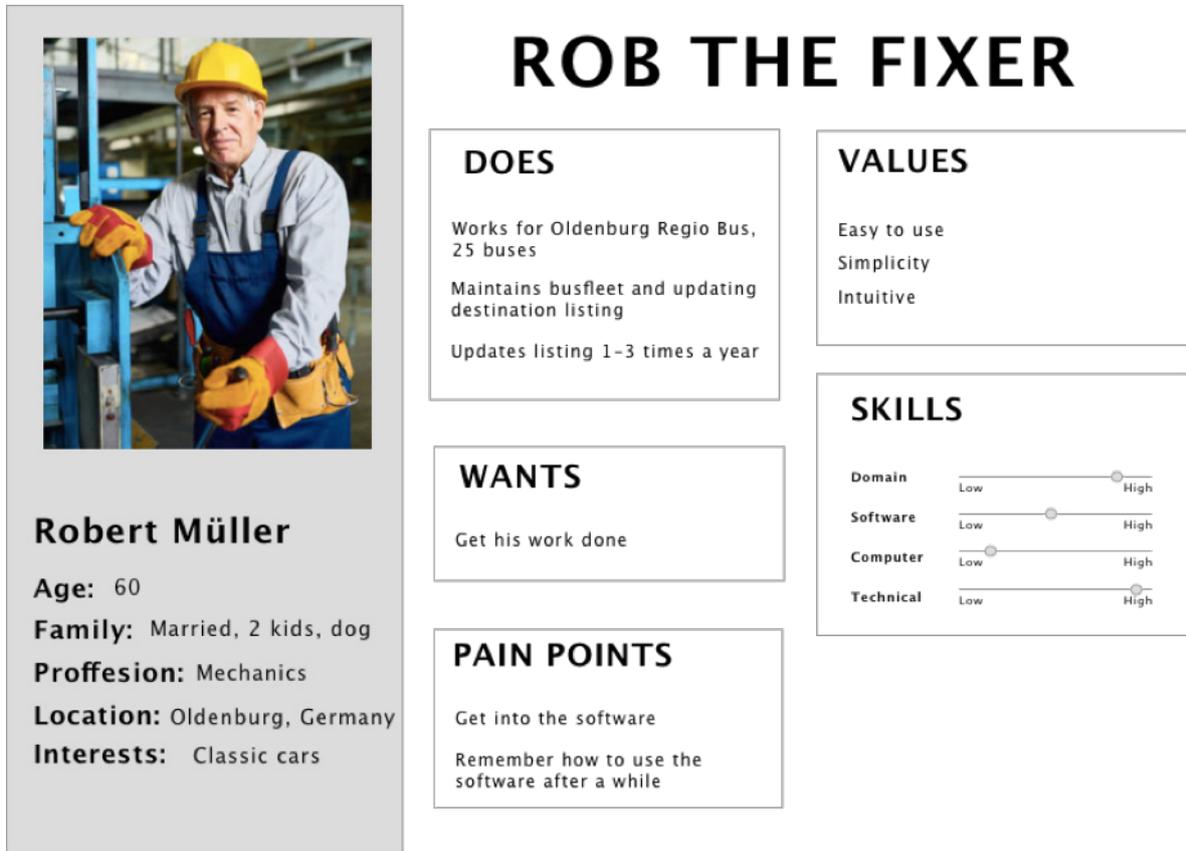


Figure 5.1 Persona profile: Rob the fixer.

Description: The digital revolution has taken him by storm and he struggles to keep up, especially now when he is responsible for programming the destination signs since three years back. He thinks it is difficult to work the computers and try to minimize the interaction with the software. He admits that's sometimes he neglects the task of correcting errors he have done. Last week they bought a new bus type with LTG signs. He just learnt that he has to setup the new bus in a new software and enter their list of destination messages because the new bus will run the same routes as the rest of their fleet. He also wants to produce an outfile to plug into the bus with USB. He feels reluctant for the task because he knows it took him long time the get used to and appreciate the old one. He also expects having to enter all the messages manually because that's what he is used to in his current software.

Update message listing due to schedule changes

Mary Kellher (40, married, 2 children) lives in Dallas, Texas. She works at DTA (Dallas Transport Authority) since five year back. She is responsible for whatever administrative work her

colleagues dump on her desk every morning and also create destination lists for their 450 buses. She doesn't really mind the work but because she is at her desk most of her days, she enjoys her spare time which she likes spending baking cakes with her kids or choir lessons. Her persona profile is presented in the figure below.



Figure 5.2 Persona profile: Admin Mary.

Description: Four times a year they updates their destination lists. She receives an email from their traffic planner with an excel sheet of the new list, telling her that it's time again. Mary has been using the LTG software for over three years and know the drill by now. Normally when she is asked to update the destinations she looks through the list and check how many changes there are. If there are many, she usually import the entire list and delete the old one. In this case there were few changes so she spends a few minutes to go through the changes on individual signs.

Make changes to bus fleet and update message listing

Carl Rodrigues (35, married, son) lives in Caxias do Sol in Brazil. He works at Mobitec Brazil Ltd as customer support. He spends his days educating and solving problems for customers. Work can be quite overwhelming form time to

time with helping many customers at the same time with issues concerning everything between heaven and earth. Good that he has several years of experience so he learnt the most common problems and solutions and how to work the editing software. His persona profile is presented in the figure below.

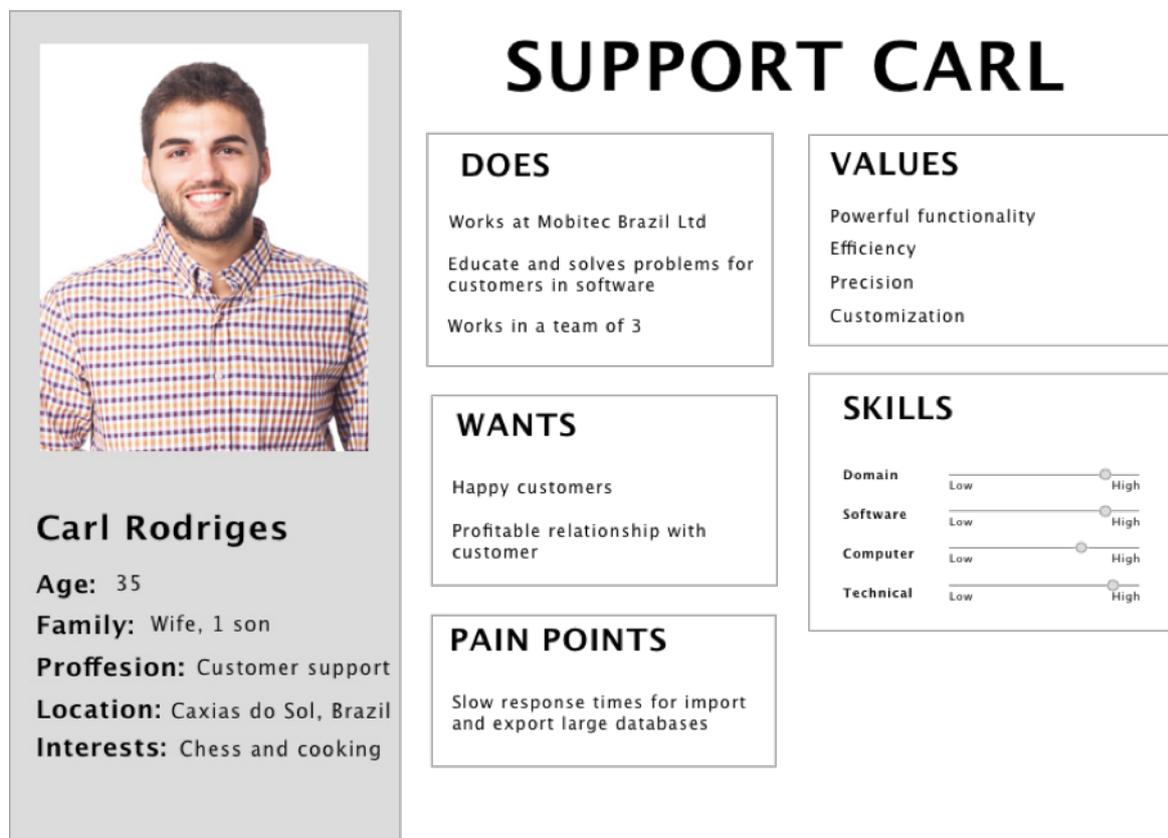


Figure 5.3 Persona profile: Support Carl.

Description: One of their smaller customers contacted him to get help with adding a new bus into their system. For some reason they weren't able to add it by themselves and asked Carl to do it for them. Carl consider himself as service minded person who values happy customers, but right now he just feels stressed when adding this task to his mental lists of open issues in need of solving. He receives editing permission to join the customer project in the editing software and adds the new bustype with signs according to the customers specification. To check with the customer that his work was what they wanted, he sends a print of the vehicle setup to the customer.

5.2 Improvements

Below are the main improvement points for the design of the new software. They are suggestions on how to solve biggest pain points which were observed in the user studies. Moreover, these improvements are the most relevant aspects to the project scope in terms of user experience and interface design rather than technical issues. Full list of potential improvements can be found in appendix 9.

Compatible with all software and sign sets

Today, many of the customers have bus fleets which contains equipment from several brands. Every brand is controlled by separate software which means many of the customers, especially in the US, are editing their destinations in several softwares. Making it possible to transfer old databases, edit all messages in the same software and being able to export to different formats would probably make a big difference and reduce the amount of work.

More automatic and less repetitive work

One of the biggest issues of today's software is that they have low automation level and everything has to be done manually when it comes to the editing messages. Adding functions like import of destination lists, default auto-formatting according to templates, copy and paste would drastically reduce the repetitive work.

More flexible, customizable and less limitations

While some of the software are very manual, some software are more automatic and they can

be very limiting and don't fit the users diverse needs. For example, users of Alpha-NT have big troubles because of a ten font limitation and users of MIE have problems with pixel by pixel editing their messages and position limitations. The new editor should present a good balance between automation and customization. It should allow the users to create their own templates, no font or position limitation and pixel by pixel control.

Better guidance and support

During the user study many users asked for better manuals and it also became clear that there is a lot of pressure on customer support. By improving the support, users can solve their own problems and make it possible for customer support to focus on bigger issues. Step by step manuals, video tutorials, tooltips, wizards, FAQ and help button would make a great difference for the user flows for especially beginner and intermediate users. To make the interaction between customers and support even easier, a collaboration function could be added. The collaboration function would allow several users in the same project with different roles - read or editing permission for example.

Better prioritization of functions

During the user study visit in US, there was one customer who was frustrated that the process of inputting destination messages was too manual in MIE. After explaining that MIE have an auto-formatting function, he expressed: "That blew my mind!" This is an example of one issue that occurred several times. Many of the software have valuable functionality that is not utilized. In the new editor it is crucial that the interface is better prioritized, to make it natural to use the efficient way and make it more com-

plicated to use the less efficient and "dangerous" options.

Web-based and modern interface

Lastly, the new editor should have a flat style and web-based interface with the look and feel coherent with Luminator Suite. The current software, with exceptions of icenter, have less modern graphic design. The user experience would probably improve a great deal by just creating a modern interface.

5.3 Interaction Vision

The interaction vision explains the desired interaction between the user and the software, having in mind that the user experience is the all the interactions with the software combined, not only when the user sits in front of the screen. The qualities are created based on the improvements in previous section.

Simple

The new editor should be simple to use and learn, allowing even the user with little computer knowledge or domain knowledge to operate the software with ease.

Efficient

While being simple the software must also be very efficient, to minimize the effort and time spent with the software.

Customizable

The editor must meet the needs of a global market. It means that it has to allow a broad set of layouts, effects, languages and pixel by pixel editing.

Powerful

An editor with pretty interface but the wrong functionality would bring no value to its users. The editor must be equipped with the functionality which makes it possible for the users to perform their tasks. The functionality of the software should be a combination of the good from the old software and new innovation which solves current issues.

5.4 Conclusion

The research from Understand phase was converted into design requirements in terms of personas, high level use cases, improvements and interaction vision. Three personas were developed to make the users more tangible and easier to design for. Rob the Fixer represents the user types with lower software skills who tend to work as mechanics at smaller PTAs. He uses the editing software seldomly and have relatively low ambition level. Admin Mary represents the intermediate user use values efficiency and precision when editing. Her user type tend to work at bigger PTAs with both destination editing and administrative tasks. Support Carl represents the expert customer supports who helps Rob and Mary when they have problems. This user type sets very high requirements on efficiency, functionality and customization.

Three main high level use cases were set to represent the most common situations why users are using the software: Buy a new bus with LTG signs for the first time, Update message listing due to schedule changes and Make changes to bus fleet and update message listing. Designing for these use cases would create a main work-

flow in the new editor that would match the mental model of the users.

Many improvement possibilities were raised during the user studies, including: compatible with all sign sets, higher automation level, less repetitive work, more flexible and customizable software.

An interaction vision was designed to guide the prototype design process. The interaction should have a balance of simplicity and powerful functionality as well as provide efficiency and customization possibilities.

Above interaction vision covers the pain points between users and existing software gathered from user studies but also set goals and requirements for the design of the new software. In the next stage of the the project, create phase, interaction vision helped guide the focus and direction during the design process, and also guaranteed that the final result of this project is targeting the pain points.

06

CREATE PHASE 1: FRAMEWORK

The expected deliverable of the project is an interactive prototype of the new editor. A top to bottom approach was used to develop the concept, starting with the overall concept and then moving into more and more details. This chapter describes the first iteration of the prototype building, where focus lied on figuring out the main workflow and functionality by designing for the main use cases in which the users would use the software. It was important to validate the software on a conceptual level before moving on to the details. The design assumptions were validated in user sessions, in order to ensure the product corresponds to their mental model of use and needs.

6.1 High level use cases

Before developing the first version of the software, it was important to design a main workflow that made sense for the users. The main workflow was adapted to match high level use cases the users do with the application, previously also described on a more general level in personas and high level use cases. These high level use cases are the main when and why they are even touching the software.

*High level use case 1:
Buy a new bus with LTG signs for the first time*

The first high level use case is when the user

interacts with the software for the first time. The reason can be that the transit authority buy a new vehicle with LTG signs or when the software is first launched. The goal with the use case is for the user to be able to transfer and combine all the old data from previous software and get to know the new software. The data consists of both fleet setup (with bus types and signs) and message listing. This use case puts high demands on learnability (in order for the user to quickly learn how to work the software) and compatibility with other software (in order to be able to transfer old data). Below figure 6.1 explains the general flow of the use case:



Figure 6.1 General flow of high level use case 1.

*High level use case 2:
Update message listing due to schedule changes*

The interaction with the software is most of the time connected with the changes done to the public transportation schedule. The schedule is updated several times during the year and this means the users have to make updates to the destinations in the message listings and update

the outfile on the buses. The second high level use case focuses on the editing and exporting part of the software where the efficiency and precision is important. It is also important to get an overview of the signs to make sure that the destinations in the message listing works in the new signs. Below figure 6.2 explains the general flow of the use case:

Log in → Open project → Add bustype/equipment → Check messages in signs → Create and transfer outfile to bus

Figure 6.2 General flow of high level use case 2.

*High level use case 3:
Make changes to bus fleet and update message listing*

The last high level use case is when the transit authority make changes to the vehicle fleet. Normally this mean that the customer buys a new bustype equipped with LTG signs. Then the user wants to be able to add the bustype in the

fleet with the correct sign setup, check that the message listing works with the new signs and export and transfer the outfile to the new bus. Below figure 6.3 explains the general flow of the use case:

Log in → Open project → Add bustype/equipment → Check messages in signs → Create and transfer outfile to bus

Figure 6.3 General flow of high level use case 3.

6.2 Software Structure

The ambition of the new editor being compatible with all new software is a great challenge because all are using different system technology. For example, it is not possible to plug in a LTG Sweden outfile into a LTG Rastatt system, it will not only work but it could also damage the technology. In reality, see figure 6.4 below, the new editor would have to have migration

and exporting modules built in. The migration module makes it possible to import the vehicle and sign setup and message listings into the new editor so that the users can edit all sign brands in the same software. The exporting module would then export the outfile from the new editor into file formats which are compatible with the different brands.

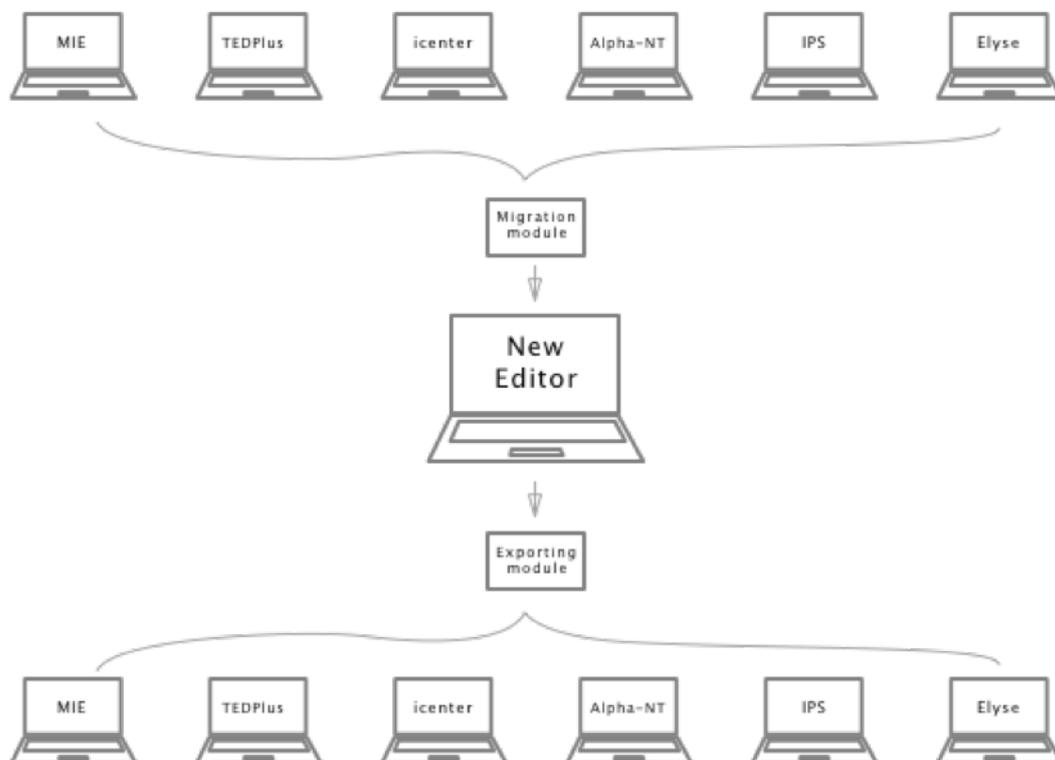


Figure 6.4 How migration module and exporting module work with the new editor.

The software structure was analysed from a user perspective. Just by making it possible to edit all sign brands within the same software, it would reduce a great amount of the current repetitive work in the setup and editing phase. The exporting phase was also questioned to reduce the amount of work for the user. In figure 6.5 seen below, one example of how the current structure and two new concept looks like are illustrated. The first one is structured by bus type with one message listing per bustype. It also uses one outfile per bustype. This creates repetitive work when editing the destination messages, two different bus types can sometimes run the same lines and therefore share the same listing. In concept 1, several bus types can therefore share the same listing and if the user wants to have different message listings then the user can create several projects. It reduces the amount of work when creating the message list.

The exporting procedure was also questioned. One customer support who was interviewed in

this project explained that often when he helps the customer updating the outfile in their buses, he has 10 USB sticks with different outfiles dangling from his arm. This shows a problem of repetitive work and a potential cause of use error if the wrong outfile was plugged into the wrong bustype. In concept 2, the software has the possibility of exporting an universal outfile which could be inserted into any brand or type of vehicle. This concept builds on the different systems ability to recognize what part of the outfile which is relevant for that system: “only eat the good apples”. The universal outfile would make a great difference from a user perspective, but in reality it is difficult to realise because of old legacy systems. It would be no problem with the new wireless technology. It became a question whether to support legacy equipment and being very restricted to keeping the structure as it is, or set a date from which the software would support the equipment (not support older versions of signs) and get more space for innovation.

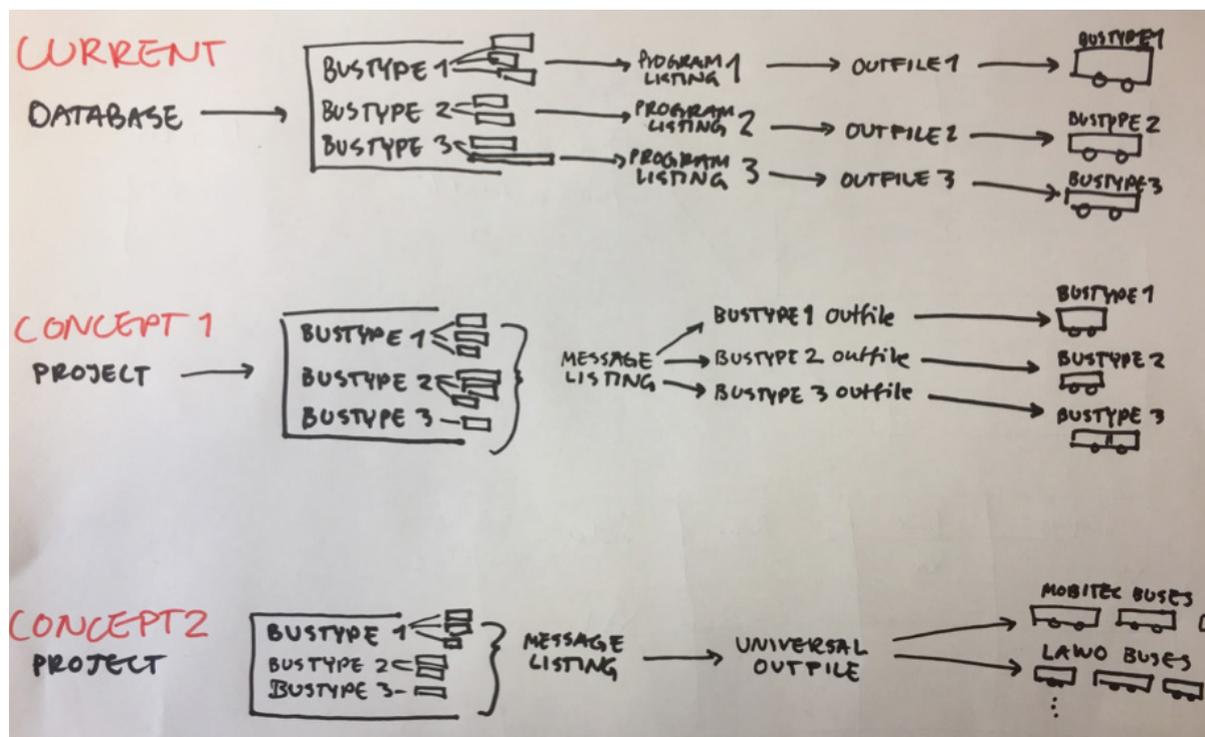


Figure 6.5 Three concepts for the software structure.

6.3 Functionality

Based on high level use cases and previous user studies, the main functionality of the application at this stage is presented in figure 6.6 below. The application is web based and the user would have their own account with username and password. In the software the user can set up their vehicle fleet with assigned signs. The vehicles and signs settings should be reusable with copy and paste functions to speed up the process next time.

The editing functionality is high priority in the application because this is what the user spends most time with. One big difference from current

system is that all sign from all brands can be edited at the same time in the same software. This saves a lot of time and effort for the user. In the new editor it should also be possible to import excel sheets or enter destination messages manually which auto-formats directly into the signs. The user should then be able to quickly check if the signs has the correct content and add effects. After all editing is finished, the user should be able to export the settings into an outfile in the right format for the intended brand system. If you have a bustype with Luminator signs for example, then the export format should support their system.

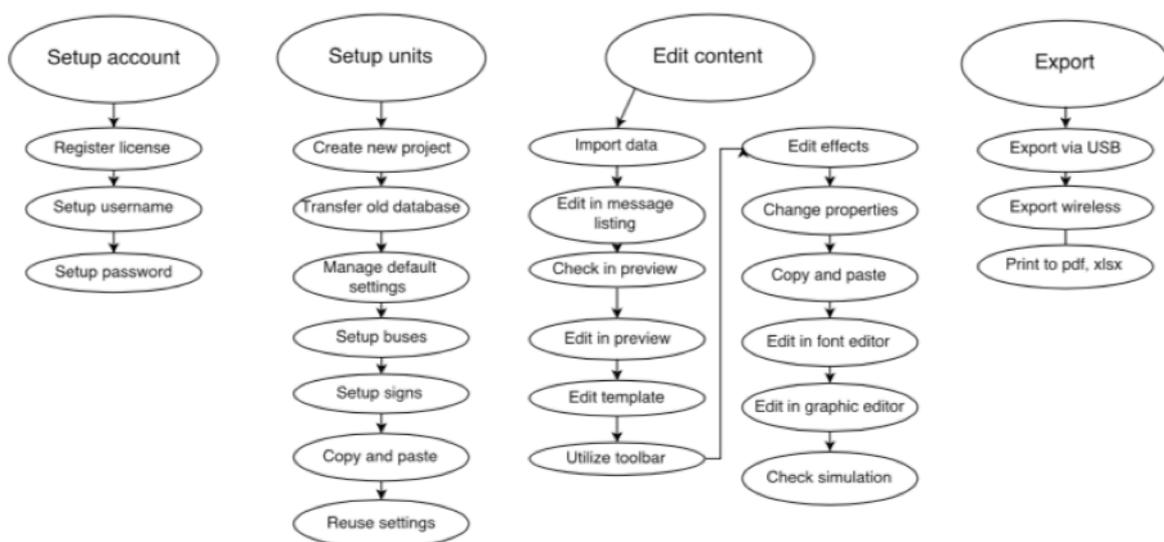


Figure 6.6 Software main functionality developed based on high level use cases and user studies.

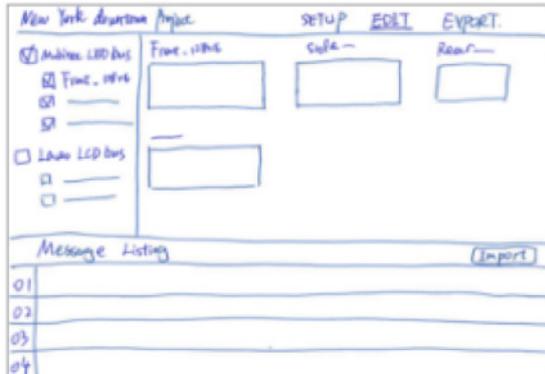
6.4 Main Workflow

The main workflow of the application was created to realise the high level use cases and the functionality. The main workflow is divided into Setup, Edit and Export. Below the first prototype is presented according to these three phases, see figure 6.7, 6.8 and 6.9.



Figure 6.7 Setup workflow in the first prototype.

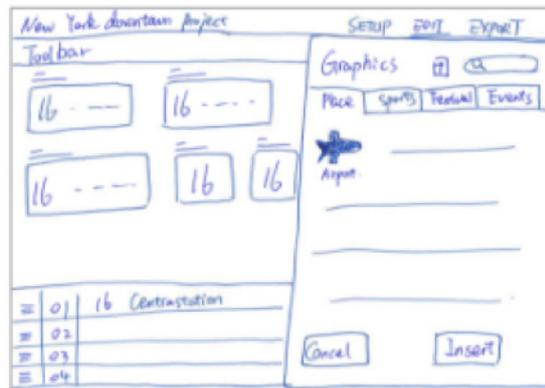
Edit



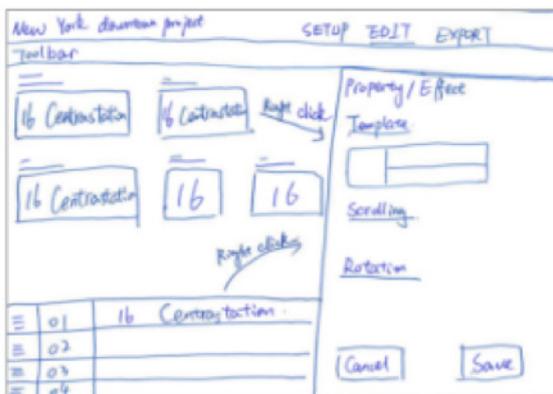
Main frame of the application with setup, edit and export tabs, a workspace with the signs, a fleet list and message list.



To create a destination message, the user can enter it in the table and it autoformats into the signs.



In the graphic editor the user can choose from a default list, create a new or import.



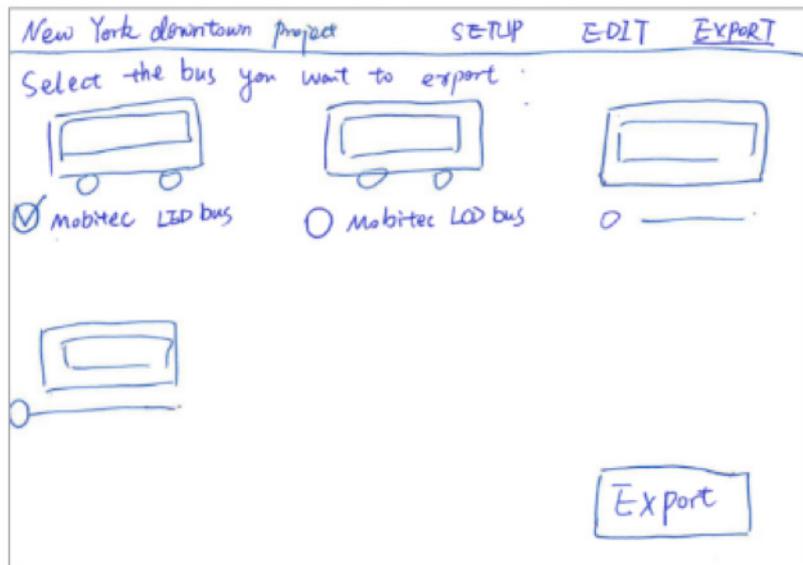
Effects are added in the properties/effect section.



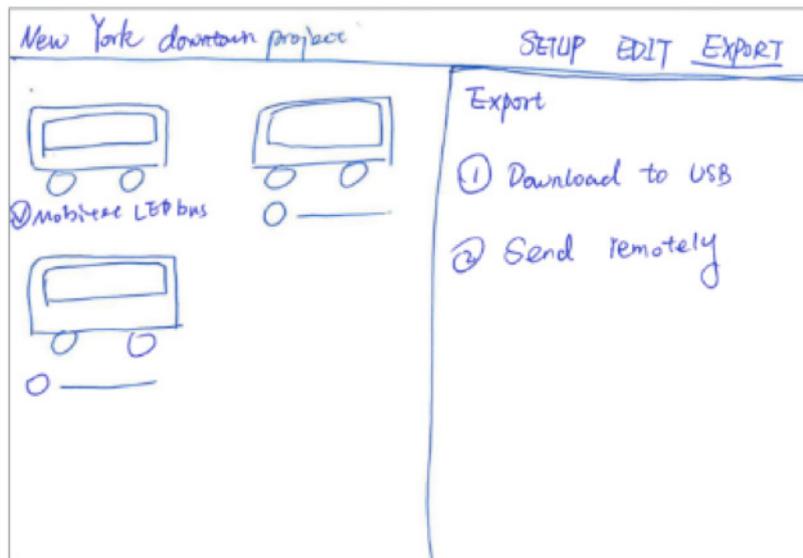
It is important that the user get visual feedback on which of the signs that have effects. Under the signs in the upper row they have dots underneath to indicate rotations.

Figure 6.8 Edit workflow in the first prototype.

Export



Select the vehicle you want to create an outfile for.



Select if you want to download to USB or send remotely.

Figure 6.9 Export workflow in the first prototype.

6.5 Design Assumptions

When creating iteration 1 prototype, many assumptions were made during the design process on what the software should contain and behave. To create a user-centered design, it is therefore important to list the assumptions made and then validate them to move further. In iteration 1, assumptions were made on:

- *High level use cases*
- *Software structure*
- *Functionality*

- *Main workflow and layout*

In appendix 6, the complete list of assumptions can be found. The assumptions questioned if the three main use cases are the most common things they do with the software, as well as if a vehicle oriented setup with one message listing and one universal outfile was is the best structure. Assumptions were also drawn about the functionality described above. and the main workflow and layout.

6.6 User validation and outcome

The assumptions were tested by demonstrating the prototype in both face-to-face and remote sessions with participants from the user types Customer support and Administrator from both Europe and US. Four sessions in total was conducted. No usability tests were done in this iteration because the prototype was not detailed enough. When choosing the set of participants a large variety was preferable, they should differ in the same way as the users differ in reality: responsibilities, background, computer knowledge, software skills and market. Below is the results from the sessions.

General

In general the participants thought they got a good overview of the concept and thought the prototype was heading towards the right direction. Many challenges were also risen. The issue of having a web-based software was risen with the comments that it would require the users to have connection all the time. Some users don't even have internet at all. Web-based software are the future and the problem could potentially be solved by allowing users to edit without connection and when they get connection it would auto-update. Some participants pointed out that the biggest challenges of the concept would develop when trying to combine what 5 different system can do today. "There are a lot of things to take care of" according to one participant in Sweden. Many of the participants also thought that they recognized the interface based on previous experience with other software.

Use cases and software structure

The three main use cases were validated and one message listing per project was preferred. When it came to the software structure there were many differences in opinion. If the software would be based on vehicle types, some argued that some customers want to have all the signs under one vehicle type and then generate one outfile (but signs have to have different addresses) and some customers prefer to have different

vehicle types. The software should therefore not limit the users but allow both setups. The concepts also suggested one universal outfile which could be plugged into any vehicle type from any brand. On the positive side it would save time when the user doesn't have to load several outfiles on several USB sticks and keep track on which stick to plug into which bus. "You have one stick which works for all buses" commented one customer support participant. The challenge with this idea from a technology perspective is that the system requires an identifier so that the bus knows what file and what info to take. For some systems it is difficult to achieve because some control units can't be updated. Also if you have a universal outfile it is more difficult to detect and solve issues on the outfile, because you can not see its content.

Functionality

Participants from US saw improvements with the functionality set in order to fulfill their needs. They pointed out that for them they want to be able to have dash sign support to be able to edit a small sign in the front window. It could be treated just as another sign. They also want to have PR message support and be able to customize them. In US they have an emergency switch in their buses and they want to be able to edit this message. Participants from Europe was satisfied with the functionality.

Setup interface

Some customer supports questioned having the setup module within the application so accessible. Having the less technical users in mind, they thought that having the setup interfaces so visible could be dangerous. “You don’t want users who don’t have enough knowledge play around with dangerous settings, then they might mess up the settings” said by a customer support from US. Many were also positive to the three setup possibilities: “If it’s possible to set-up with the three ways, that would be great”. Especially article number received positive feedback. It would be easier for the users who have little domain knowledge, they could just enter one number and be done with it. Especially when the software will be web-based, it could be connected to the new business system ABAS, with access to all product data within the group. Many customer supports pointed out that setting up with order number can be difficult to achieve in reality. Discussions were also risen about including the control unit in the setup face. When including all brands in the software, it might be necessary to define control unit in order to create accurate outfiles.

Edit interface

Users gave many input on the edit interface, especially they requested abilities to organize their workspace. For example, the ability to group signs by positions in the workspace and to turn on/off the default fast input. Some users wanted both overview of the signs and close view of the signs so that they can control the pixels close up. For rotations and scrolling was also important to get a good overview of to be able to see what rotations they have and what part is going to scroll. One user suggested having the buses and signs next to the message listing - similar to the layout they are used to.

Legacy or innovation mode

The biggest discussion of this prototype concerned whether to be compatible with old legacy products or allow new innovative ideas which would make bigger positive difference for the users. On one side there are a lot of users and technology already out there, and not designing for their systems would probably lose many customers, but designing for them puts a lot of limitations on the new software.

The innovation mode would be to set a date from which the software would support only the new modern signs, the so called IP signs for example, which are connected to internet directly. This would open up for possibilities that would probably change the software completely. The new signs allow for dynamic input and in the software the user would only have to define a template layout and where the information comes from. Then in the signs in the vehicles would have a dynamic dataflow with destination data that auto-formats according to your defined template. This automatic mode is probably the future trend. “We shouldn’t look at what we have today, that will only limit us because the current hardware is impossible to change. If we go for the future mode, then we have to set a date from what year we are supporting signs and not support too old technology”.

But problem is that the market is not in future mode yet. The signs and control units can last for 20 years, therefore it takes time to get rid legacy products. Designing for legacy mode would mean keeping addresses for example (“Why talking about addresses - they think about front, side..., leave it to the developer. Don’t let users care about address anymore, only position”) and many other limitations (“If we want to be backwards compatible then we are stuck with what we have”). Would one suggestion be to allow both automatic mode and static mode in the same software?

6.7 Conclusion

In the first Create phase, the foundation of the application was formed with software structure and functionality. Inspiration from existing software and the high level use cases created a main workflow which followed the mental model of users: setup project with vehicle fleet, create destination messages and finally export outfile with the messages. The main workflow was visualized in a low-fidelity paper prototype which was tested with external and internal users.

Based on the feedback of iteration one prototype these were the decisions and progress made:

- Design for the three main use cases: Buy new bus with LTG signs for the first time;
- Update message listing due to schedule changes and Make changes to bus fleet and update message listing;
- Design for legacy mode with a project and vehicle type based software with one message listing per project and outfiles per bus-types;

- Design for a integrated layout with three sections Setup, Edit and Export;
- Design for the three ways of setting up signs;
- Design for special needs of US market with support for editing of dashboard, PR message and drivers message.

These outcomes of Create phase 1 and above design decisions will be used as cornerstones for the next create phase, which helps to narrow the design space in the following design process. The next version of the prototype will be designed to incorporate above features to satisfy the user input gathered in this iteration.

07

CREATE PHASE 2: INTERACTIVE PROTOTYPE

In the second create phase, the focus was lied on developing the prototype to a more detailed level. An interaction framework was created to consciously design the software for its intentional use as a web based tool with which the user spends longer periods of time with. The necessary content and actions was also mapped out according to the high level use cases. The graphical design work of the software was initiated with the help of LTG Suite visual guidelines (these guidelines are confidential and will not be presented in this report).

7.1 Interaction Framework

“The interaction framework defines not only the high-level structure of screen layouts but also the product’s flow, behavior and organization” (Cooper et al, 2014). Defining the interaction framework followed these steps:

*Define form factor, posture, and input methods.
Define functional and data elements and their grouping.
Define key frames and main wireflows.*

Form factor, input methods and posture

Form factor and input method

What form factor the software have puts constraints on the design. In this case, the software is meant to be used on computers in a web application. The screen size should therefore be 1366x768px with high resolution. The usage situation would most likely take place in an office with sufficient lighting. The primary input methods will be mouse and keyboard. Some users will be using the keypad if they have laptops.

Posture

The software has a so called sovereign posture. Many job specific applications like Photoshop and Word are designed for sovereign posture. These software monopolize the attention of the user for long periods of time and they are used as the primary tool of the workflow. They should be designed for documents, meaning optimizing

the area in which the user is producing its content. In Photoshop this would be the artboard and in Word this would be the actual written document. Normally when designing for sovereign posture one should have in mind to target intermediate users. “Each user spends time as a novice but only a short period of time relative to the amount of time he will eventually spend using the product“ (Cooper et al, 2014). But in the case of destination editing software, many of the users have little computer knowledge and use it once a year. Novice users must also be designed for, as well as experts, even though intermediate users will be in focus. In this project it is therefore of great importance to achieve a good balance between support for beginners, intermediates and experts.

Also when designing for an application which will be used longer periods of time, it is important to be generous with screen real estate and optimize for full screen mode. Additionally, it is important to use a minimal visual style with a narrow color palette in muted colors. Tiny color details and accents of color will have more effect and be easier on the eye. Controls and tools can also be smaller than maybe expected: “The user will stare at the same palettes, menus and toolbars for many hours, gaining innate sense of where things are from sheer familiarity. This gives you, the designer, freedom to do more with fewer pixels” (Cooper et al, 2014).

Designing for sovereign posture also means keeping a balance of minimalistic style and providing rich visual feedback. Status bars, properties sections, system state indicators can be added and subtly presented so that they will not clutter the interface.

When designing for sovereign posture specific for web applications, there are several additions one can have in mind, even though web applications have many similarities as desktop applications. Consider responsiveness, should the application be possible to use on both laptop, mobile and tablet, and it is recommended to chose the most commonly used resolution. Use specialised panes or screen regions to group related functions and objects and create a feeling of being in an environment rather than navigating from page to page or place to place. The core strength of a web application is the collaboration by accessing the same data and functionality from the cloud. It is also possible to save continuously.

Functional and data elements and their grouping

“Functional and data elements represent functionality and information that are revealed to the user in the interface” (Cooper et al, 2014). These elements are a direct translation of the requirements gathered in the Understand Phase and described in the language of user interface representations. The data elements are the elements which users interact with and they are grouped together in so called containers to facilitate the user flow. The functional elements are the actions the users do with the data elements. In this project, the elements are divided into the three phases of the software: Setup, Edit and Export. Each phase was turned into a key frame consisting of several containers

with data elements. In the table 7.1 below, there is an example of how the containers, data and functional elements of each key-frame were laid out in the Setup phase. A complete list for all phases can be found in appendix 12.

Container	Data elements	Functional elements
Setup account	Account	Register account (username, password) Remember account Save account Edit account (username, account) Delete account
	License	Enter license number
Setup fleet	Import	Import fleet setup/database
	Vehicle	Define name Select Vehicle type from list Select brand from list Select control unit from a list Delete/copy/paste vehicle Change vehicle setting
	Sign	Select identification method (from list, article number or order number) If list: Select sign type from list Select position or enter manually Choose suggested address or enter manually Define sign name Choose template from list Preview sign Add another sign Delete/copy/paste sign

Table 7.1 One example of functional and data elements for setup.

Keyframes and main wireflows

The prototype from iteration 1 was updated and visualised in Sketch and XD based on the user input. The data elements and their actions were incorporated in this version as well as the guidelines for sovereign posture and LTG Suite. Following section describes the prototype produced in iteration two.

The software was structured into three main sections based on the main workflow, namely setup, edit and export. As illustrated in figure 7.1, by using the three buttons 'setup', 'edit' and 'export' in the top bar users could navigate to corresponding sections. The three screens are the key frames for the software. Setup is the foundation

of a destination project where relevant vehicles and signs are created according to real parameters. This procedure lets the software know what vehicles, signs and settings users have in the reality, and therefore generate correct configuration for each project. After setting up the project, users could edit the content of the signs in Edit section by entering the destination text in the table in the lower part of the edit screen. The toolbar can be used for detailed editing and effects like scrolling and rotations can be added in the properties section on the right side of the edit interface. Lastly, when all editing is finished, in Export section users could generate outfiles for control unit or print materials for administrative purpose. Below follows a more detailed description of the three keyframes.

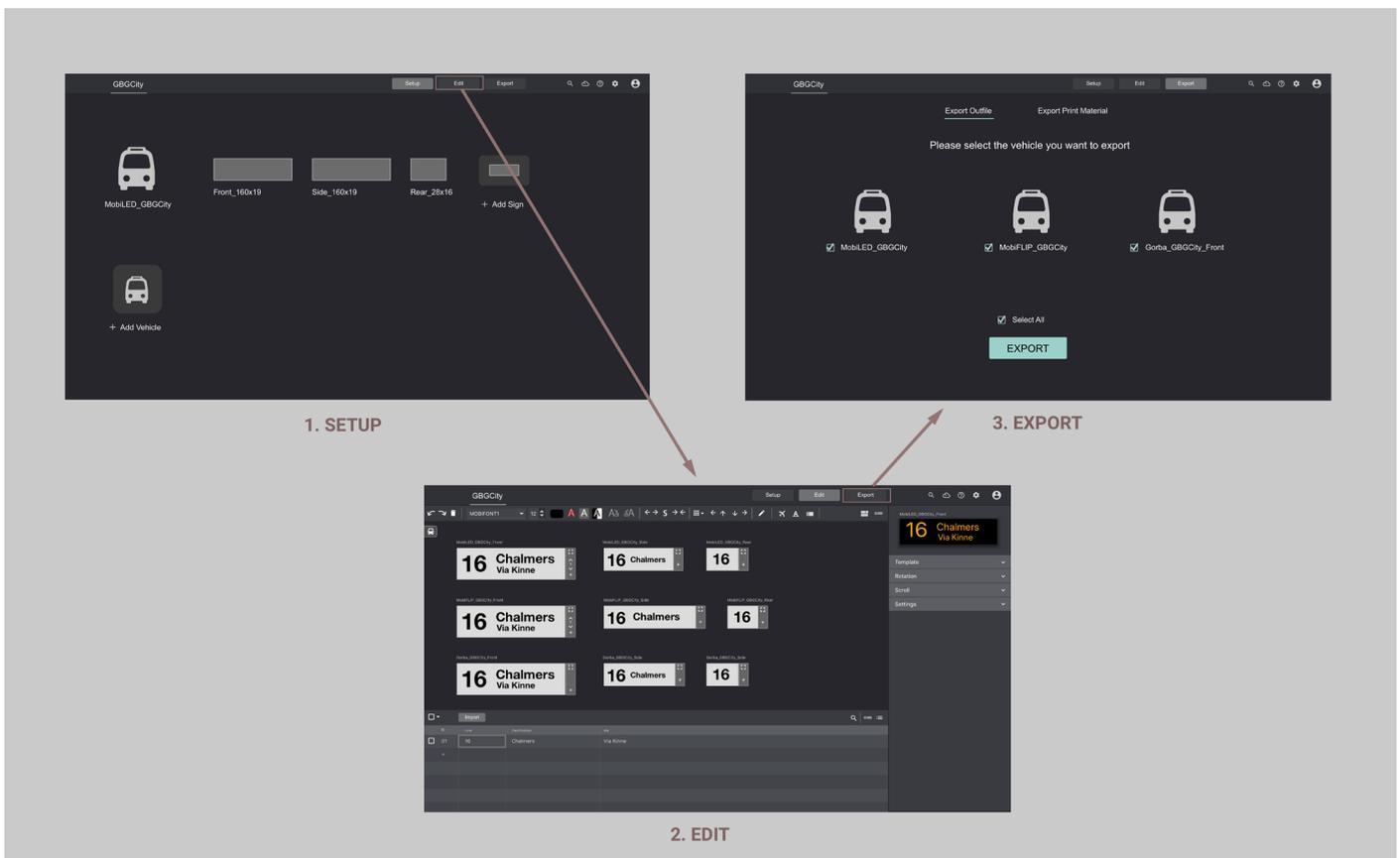


Figure 7.1 Visual representation of the second prototype navigation.

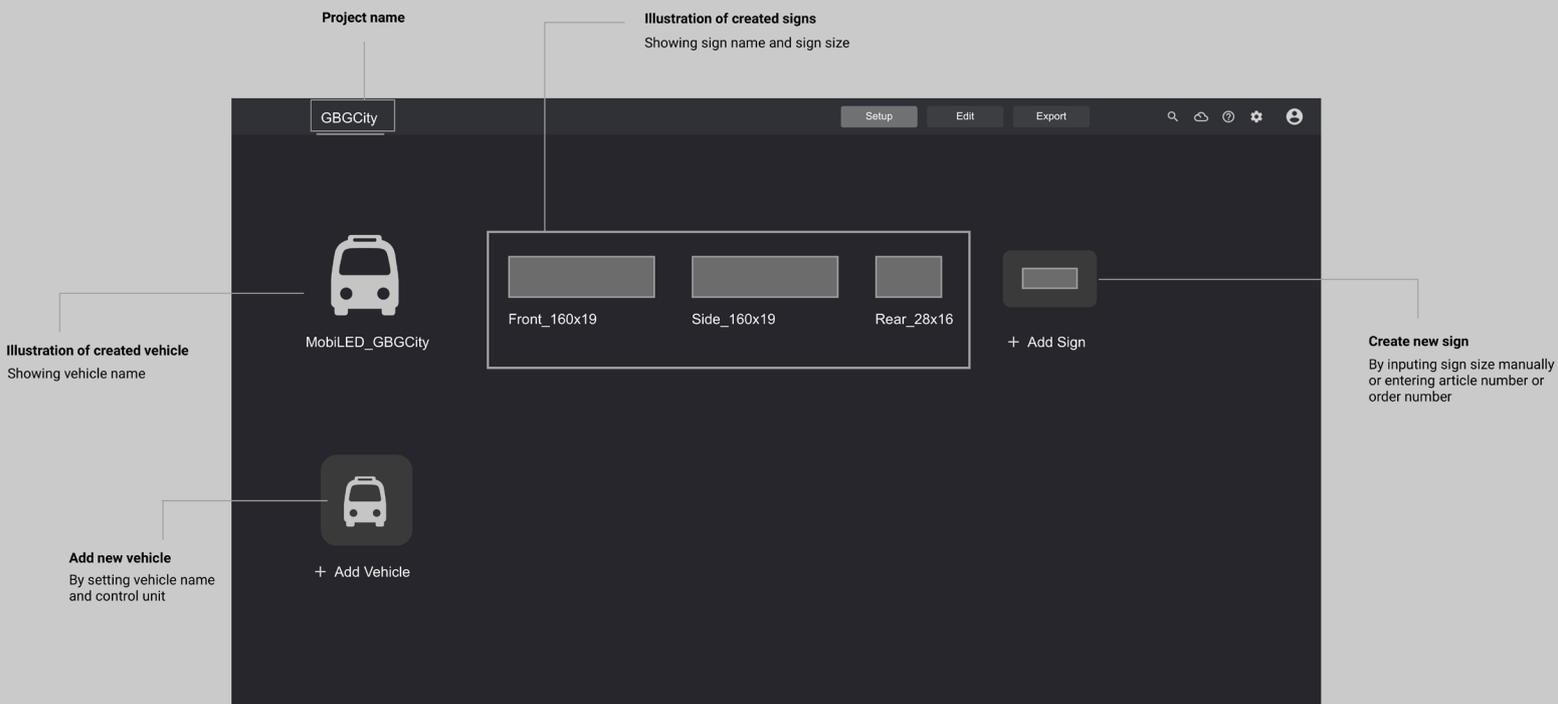


Figure 7.2 Setup keyframe in the second prototype.

In Setup interface, see figure 7.2, all the vehicles and signs created in current project are presented here in the form of graphics. By clicking on individual vehicle or sign, users could check the settings of them and make changes if needed. Moreover, according to high level use cases,

users could also maintain the project according to the changes of equipment in the reality, such as create new vehicle or assign new signs to existing vehicle.

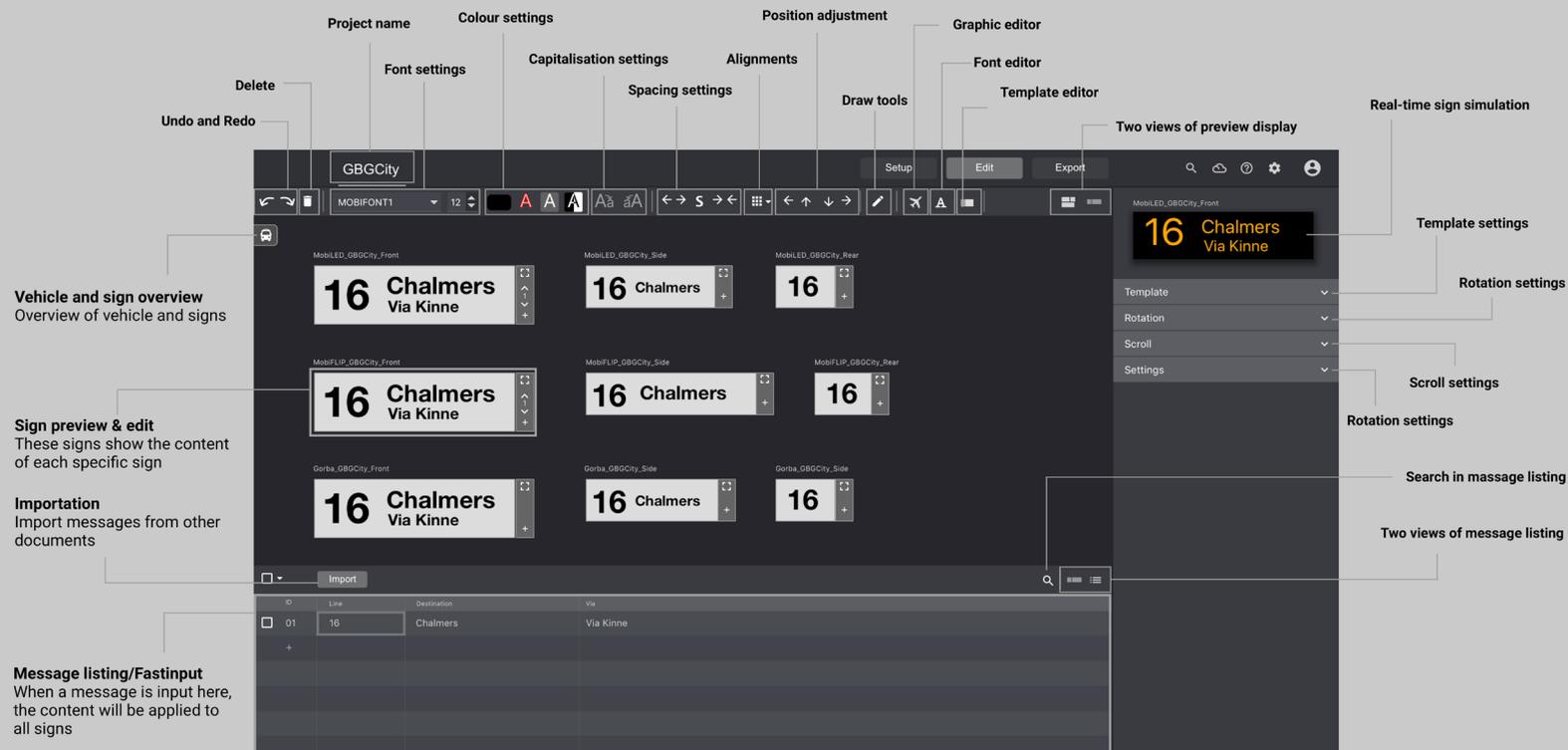


Figure 7.3 Edit keyframe in the second prototype.

In Edit section, see figure 7.3, users could edit content of the signs created in setup section. Edit interface is mainly divided into four parts: toolbar, property, message listing and workspace. Message listing works as Fastinput in MIE which means when a message is input here the content will appear on all signs and auto-formatting will be applied automatically. With the help of the message listing, all signs could be edited simultaneously in a quick and smart way. If Fastinput does not work successfully or changes on specific sign are needed, then users

could click on individual sign in the workspace and edit the content by using the toolbar. The property window at the right of the interface is specific to each individual sign, depending on which sign the user has selected. In property window, the real-time simulation of the sign could be seen on the top, below that, the template of each sign could be changed and special effects like rotation or scrolling could be added.

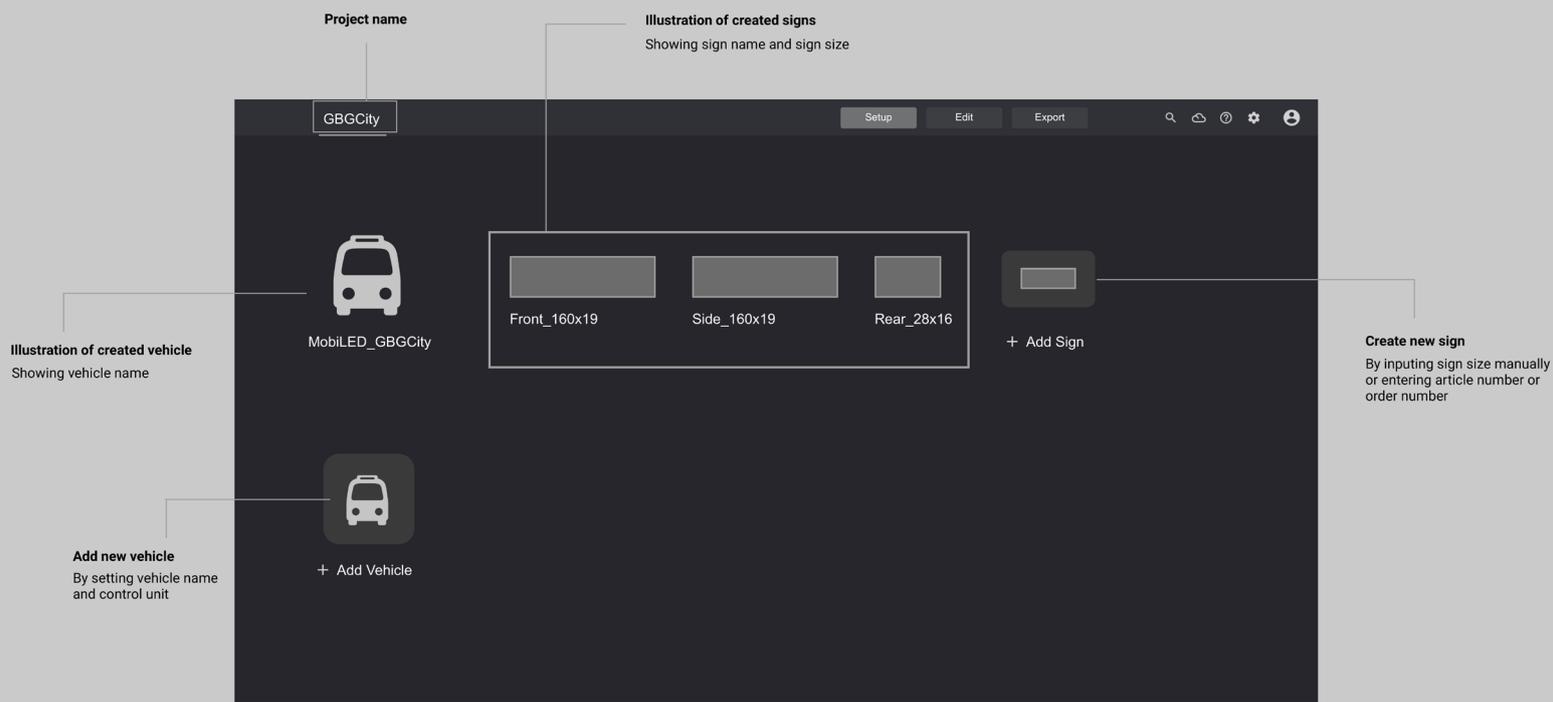


Figure 7.4 Export keyframe in the second prototype.

When all the editing are finished, in Export section, see figure 7.4, users could generate both outfiles for control unit and print material signs for administrative purposes. By ticking the vehicles and messages, users could select which vehicle(s) and message(s) they want to export. After that, define where the files should be stored and download will start accordingly.

In appendix 15, more details can be read on how the software concept works in the format of usage scenarios.

7.2 Design Assumptions

During the creation of the second prototype, assumptions were made on how the prototype should behave in order to produce the best user experience. This time, the assumptions were made on a more detailed level than in iteration one. The assumptions concerned naming and terminology to test whether labeling and button made sense or if there are better suggestions from users. It was also questioned if the design had the organization, if the information was grouped meaningfully and the position on data

elements made sense. The design also made assumptions on first time use and discoverability, if common items are easy to find and instructions are clear. Effectiveness is also important in the software, assumptions were made on if the functionality within the concept is enough to efficiently complete tasks. The complete list of assumptions can be found in appendix 7.

7.3 User Validation

The assumptions were then validated in usability tests and demo sessions both in person and in remote sessions depending on the location of the participants. Usability tests were executed with 5 participants: one Administrator (US), three Customer support (Sweden and UK) and one Assisting editor (Sweden) (more about the user types can be read in chapter 4). The demo sessions were done remotely with customer supports and developers at the sister companies of LTG in US, Switzerland and Germany, to validate the designs with the needs of different markets and ensure the technical compatibility with their software. Again, it was important to gather feedback and test the prototype with participants from different markets, different user types and different editing software experience. Because of availability one participant of the user type Mechanics (Switzerland) was demonstrated the prototype and not usability tested.

Because of availability, some usability tests were executed remotely. These usability tests were done by sharing the link to the prototype, the participants were then asked to share their screen and think out loud when performing the tasks. The quality of these test were of course affected by this factor, because facial expressions and presence tell much about the experience, but the main experience could still be gathered. More about this could be found in chapter 11. The usability tests had following structure, and

the complete session scripts can be found in appendix 4:

- *Introduction and explanation of test*
- *Scenario*
- *Task*
- *Post task question*
- *(repeat for 9 tasks)*
- *Post test evaluation questions*

The scenarios were given to the participant to create a content for the tasks. The tests had had different scenarios depending on the participants background. As an example if the participant was Customer support following scenario was given:

“You are working as a customer support at Mobitec. LTG has just launched a new editing software which is replacing [editing software]. You have barely got familiar with the software yourself and the support requests are already piling up in your email inbox with customers who wants help with the new software”.

The participants were given 9 tasks in total. The tasks tested different aspects of the application like: setup a new project, create new messages, add effects, create graphics and create outfiles. During the tasks the participants were evaluated on following criterias: help needed, use errors, confusion, unexpected user actions and task completion. After each task, the participants

were asked to evaluate their experience. After all tasks were completed, the participants were asked to answer questions related to the assumptions: terminology, workflow, placement, icons and the interaction qualities. Below is an example of the scenario and the first task which was tested with the Customer supports.

Customer support

You are working as a customer support at Mobitec. LTG has just launched a new editing software which is replacing the one you are currently using. You have barely got familiar with the software yourself and the support requests are already piling up in your email inbox with customers who wants help with the new software.

TASK 1: Setup - 2 concepts (Setup wizard and normal setup)

You receive a call from a customer who wants your help with the new software.

Set up a new project for GBGCity with Mobitec signs

Topic	Number	Explanation
<i>Help</i>		
<i>Errors made when performing the task</i>		
<i>Confusion / Unexpected user actions</i>		

Task completion?	Yes / No
------------------	----------

How was it to perform this task? Why?	
How intuitive was it to perform the task? Why?	

Table 7.2 An example of the scenario and the first task conducted in usability test.

7.4 User Validation Results

In general, the prototype received positive feedback both in tests and demonstration sessions. The main comments revolved around the interface. The participants thought it was a simple and modern interface that was easy to use and had good visibility without being cluttered. The participants were influenced by their experience of previous software when it came to terminology, layout and how they expected the new prototype to work. It was interesting to see how the participants applied their mental models to the prototype. No big mismatches between their mental models and the new prototypes could be detected. The main workflow in the prototype was the same as they were used to. Following data was collected from the usability tests and demonstration sessions.

Usability test statistics and evaluation

Of the 9 tasks which were performed by 5 participants, 87.5% were successfully completed during the usability test. The number indicates that the product has good guessability but one should have in mind that many factors affects the number both positively and negatively. Factors like the prototype itself which can be limiting (can not click anywhere) and it lights up hotspots which helps the user to know where to click. However, viewing the participants interacting with the prototype, the general impression was that the participants understood where

to find and how to perform tasks. In a real situation where the users can familiarize themselves and try several times in their own pace, it is very likely that they would perform even better.

The tasks which were not completed varied from participant to participant and each participant had one uncompleted task in average, for example add message in message listing, change lowercase letters to uppercase letters and create graphics. Help was given to the participants to for example change into Edit mode and where to find the uppercase button. Some participants struggled understanding what all the icons meant and asked for tooltips. Some confusion points involved how to handle the message listing (that they can input right into the table), why you can not change the template with the plus on the preview signs, how the scrolling area works (MIE users are not used to this function) and where to find graphic library.

In the post test questions, the participants were asked to evaluate the interaction qualities based on their impression of the application. Simple received an average of 4.5 with motivations like “Excellent”, “I can’t imagine how everything could be easier” and “4 or 5: very intuitive, less clutter screen, more modern looking [...] it’s a bit like comparing Apple with Windows”. They expressed that the ease of use is much better than in MIE, that it was less of an engineer tool

and more a non-engineer tool in which you do not need to know so much about the sign. Efficient received 4.25 with motivations like for example: “4, probably more efficient with the simple interface”. Powerful received 4.25 because the users were content with what the software seemed able to do. “It has everything what you can do with the signs [...], I can’t imagine if there is anything in MIE is missing here”. Another participant expressed: “I can do everything i want and I have everything in one software”. The software was also evaluated to have 4 in Customizable. The participants appreciated that they could change names and terminology to some extent to suit their mental models. In general, it can be stated that the interaction vision was validated, even though the scores probably were affected by the participants desire to please the designer and not give too negative feedback. But the scores were backed up by the participants test results and motivations.

Terminology and icons

The overall terminology, e.g. setup, edit, export, rotation, scrolling work was understandable for the participants because they could complete most tasks. Some differences in terminology was raised because of personal preferences and market habits. Especially when it came to line number which was called both line number, route number and service, and rotations which is also called alterations. Users pointed out that by utilizing an universal language, some confusion should be avoided. A common language could be applied by using “height and width” instead of x/y dots for example. Icon usage could also support this universal understanding and avoid confusion because of market differences. The icons used in the software were mostly understandable, but some could be improved and use tooltips.

Layout

The layout was validated when the organization between toolbar, workspace, message listing, properties, bus fleet made sense for the users. Especially MIE users recognize this organization well. Compared to IPS the organization is a bit different from the new editor and IPS users made comments on differences in the layout. For example the user wanted to know whether the message listing could be moved into the properties area in stead to get a tall and narrow message listing (like in IPS). Based on their comments, for all users there seemed to be no problems to get used to the layout of the new editor.

Functionality

Based on their comments, the prototype covers the main functionality of their current software. It was also appreciated that font and graphic editor are integrated in the same software (this was a recurring topic that many users expressed problems with during the user study). For some user who haven’t used MIE before, they were positively surprised by import and auto formatting functions.

There were many constructive feedback on details of the design. In the setup phase comments from customer supports were made on serial/article number “It is good for people who don’t know what they are doing”. The template editor should be more precise and allow definition of ratio and font selection for each block. It should also be possible to select templates in the setup phase. When it comes to defining scroll speed and rotation duration it should show numbers and not only slow, normal and fast. Questions were also raised about how to handle languages with special characters.

The participants got inspired by the prototype and came up with ideas for new features they would like to see in their future editor. From the Swiss perspective it was of great importance to be able to import destination lists based on sign position. “Sometimes user want different content in the side sign, sometimes front sign shows destination, side sign shows via station and it should be possible to import excel sheet

for side sign destination”. They also suggested error reports when the auto formatting doesn’t work, table overview of fleet and signs, master setting page where the users could customize the software and lock feature for the messages in the message listing. A complete list of potential improvements which were suggested in this iteration can be found in Appendix 10.

7.5 Conclusion

In the second iteration of designing the editing prototype, more details and functionality were added to the prototype. It was also translated into digital clickable prototype with the LTG look and feel. The concept was also designed for the correct interaction format, sovereign posture, to make sure the software can feel comfortable and work efficiently for longer periods of time and. The design was again with users in demonstration sessions and usability tests.

Below design decisions were made based on the outcome of the user validation in iteration 2:

- Let the setup be different when setting up a new project and changing settings later in the project by using a setup wizard when setting up a new project and provide a quicker sign setup for later in the project.
- Keep the division and main workflow of Setup, Edit and Export.
- Keep rotations, scrolling, template and settings in properties.
- Have Line, destination and via as message listing titles
- Template should be created by drag and drop.
- Have a black sign in property simulation when no target is selected to give user clues of the existence and how to use the properties area.

Following the same procedure as the last create phase, these design decisions were used as starting point for create phase 3. By involving user input continuously, it helped to make sure the design decisions were made based on the opinions and needs of users. In the next phase, above features and decisions will be incorporated in the prototype to bring the design into a more detailed level.

08

CREATE PHASE 3: LOOK AND FEEL

Following chapter describes how the prototype was taken to the next level based on the input from previous iteration. New features were added and final touches were made on the look and feel of the prototype. User feedback was collected to validate the new ideas. For a complete presentation of the final concept, please read chapter 9.

8.1 Final Details

The third iteration of developing the software prototype also involved implementing some new ideas of a support system (presented below) but most of the iteration was spent on finalising the look and feel and the clickable prototype. LTG Suite visual guidelines were used to com-

plete the look and make the software coherent with the rest of the applications in the suite. Figure 8.1 shows a sample of the final prototype: the main interface for creating destination messages.

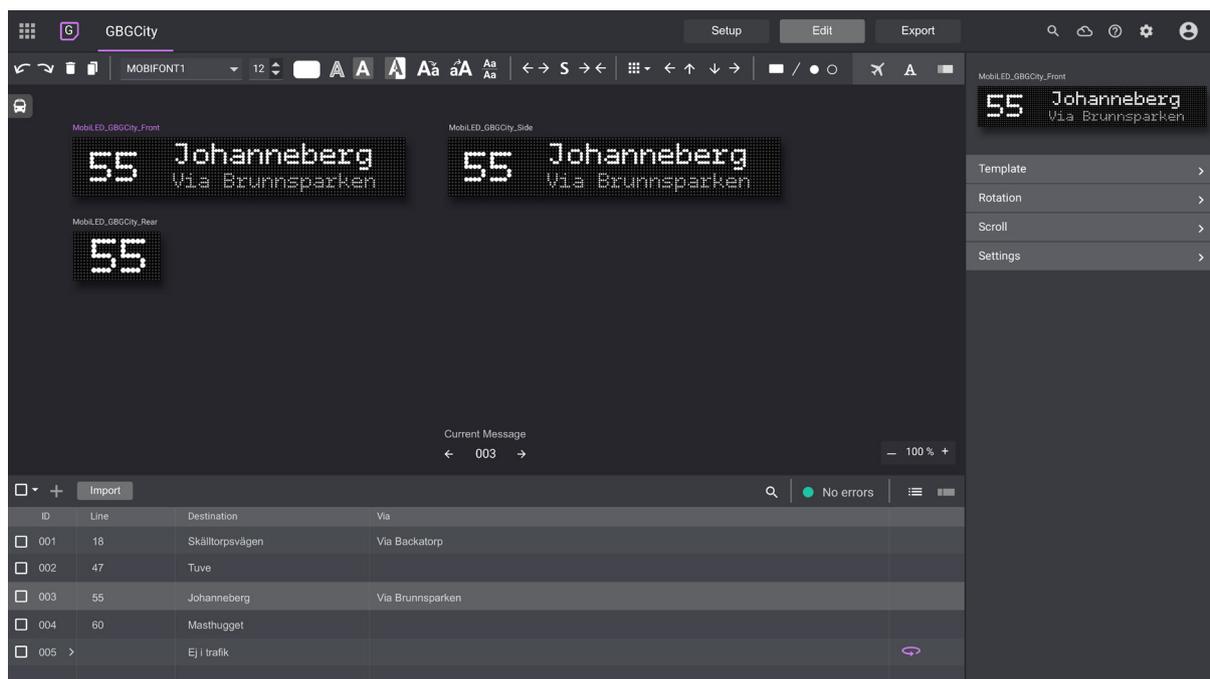


Figure 8.1 The main interface for creating destination messages in the final prototype.

Support system

The software is designed for three user types which are operating on different skill levels: Customer support (experts), Administrators (intermediate) and Mechanics (beginners/low skill level). All aspects of the design are carefully considered to achieve as good usability as possible, to make it possible for all users to achieve what they want with the software. A support system was developed to facilitate the users both during first time use and throughout future interaction.

To design for a short learning curve and provide extra support for the user types with less computer skills (Mechanics user type mainly) a first time guide was developed, see figure 8.2 below. After the user has gone through the setup wizard and enters the Edit interface for the first time, he/she will meet popup windows explaining all important interaction points like toolbar, workspace, properties and, as seen in the image 8.2, the message listing. The popups will explain what the different areas are and also provide video tutorial links where the users can get step by step explanations of how to work with the interaction points.

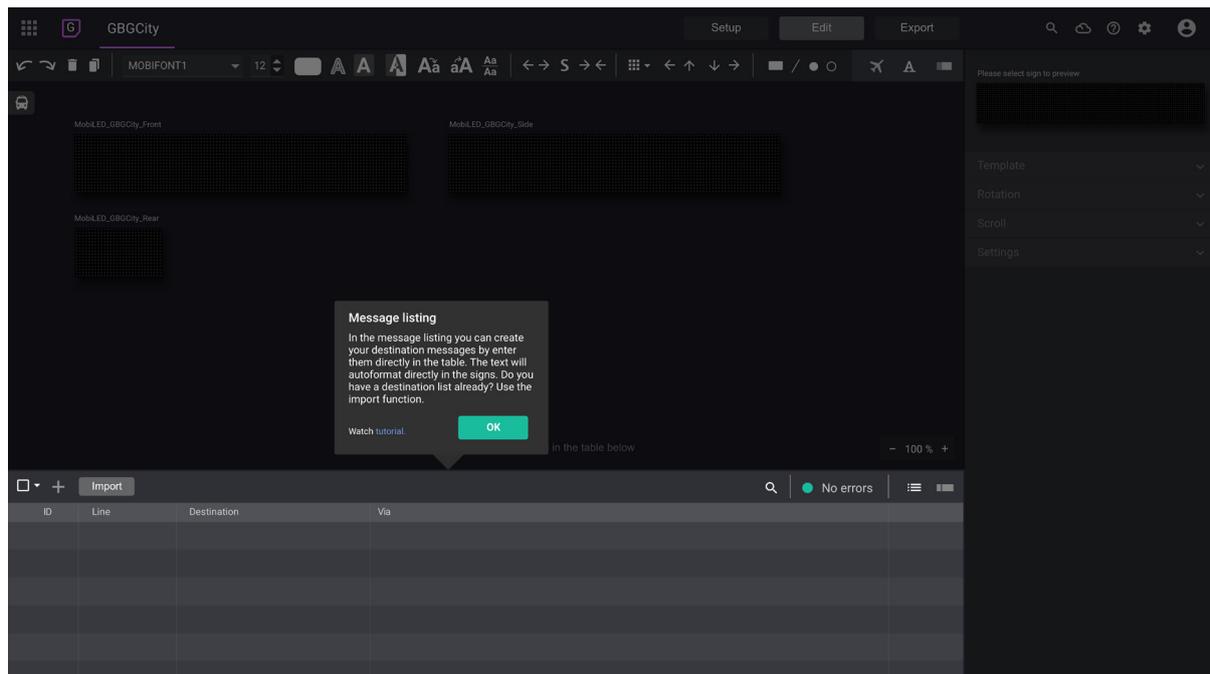


Figure 8.2 The first time guide interface in the final prototype.

Tooltips were designed for the first time and everyday use to help users navigate between the different buttons and elements in the interface, see figure 8.3. When the user hover over a button for example, a small text appears which explains what the button does. This feature is extra

valuable for many of the external users who use the software very seldomly. With the tooltips these users can get reminded of what they might have forgotten since last use.

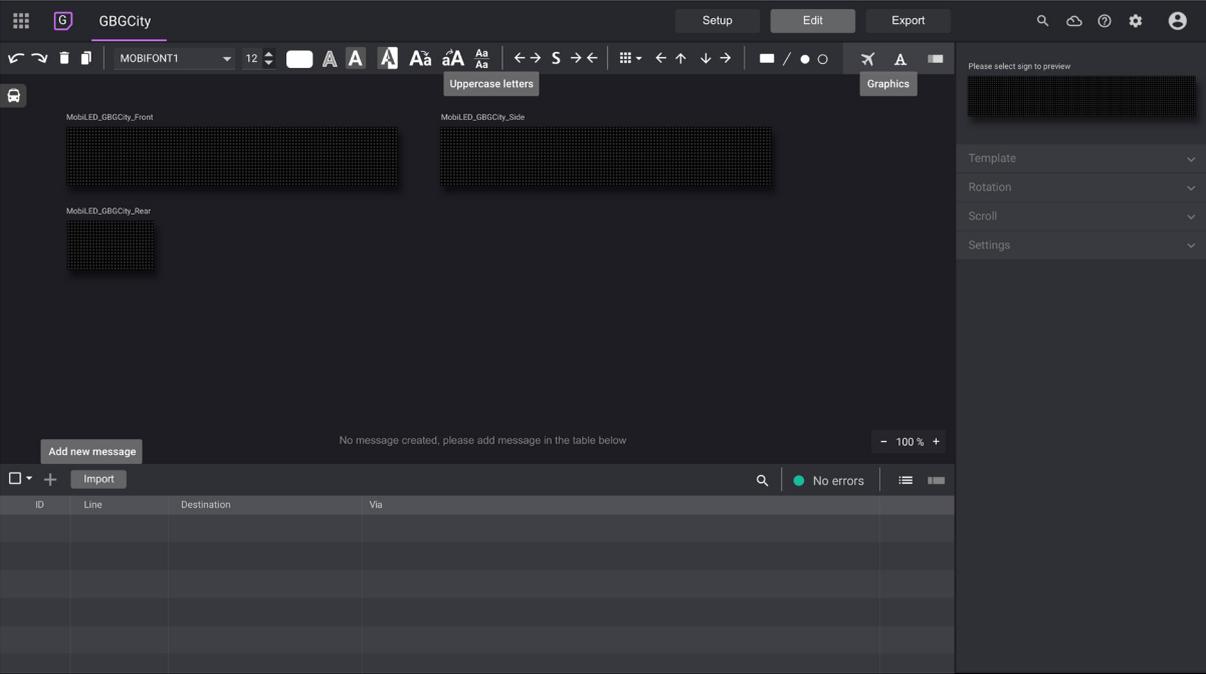


Figure 8.3 Examples of tooltips in the final prototype.

Lastly, a help center was developed to further support the users during their everyday use, see figure 8.4. If the user clicks at the question mark in the far right upper corner in the interface he/she will go to the help center. Here the user can find manuals and video tutorials explaining all

aspects and interaction in the application, see figure 8.5. He/she can also search for specific issues one might have, read about frequently asked questions (FAQ) and the forum where users can help each other and also chat with LTG customer support.

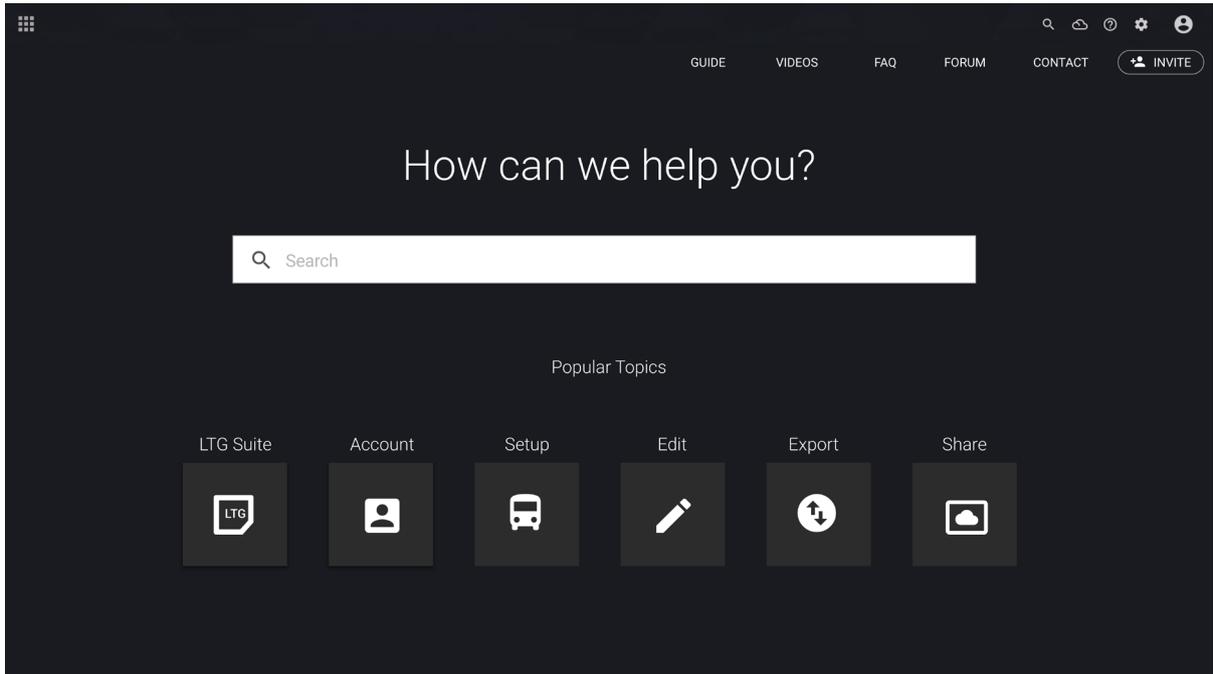


Figure 8.4 The help center interface in the final prototype.

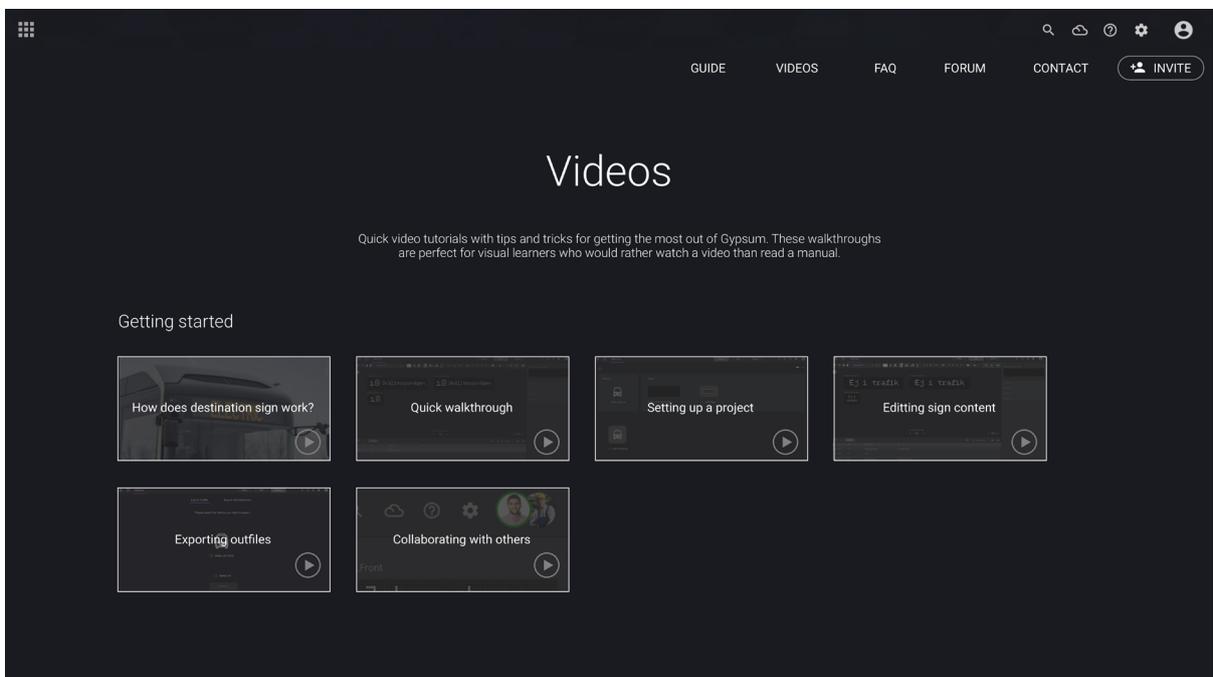


Figure 8.5 The video tutorial interface in the final prototype.

Today, when users encounter especially difficult problems they contact customer support so they can use software like TeamViewer to remote control users screen. The customer support can then solve the users problems. As a part of the support system, a collaboration feature was incorporated into the design. Users have the possibility to invite others into their project, see figure 8.6. The user can decide which access the

invitation has: read or edit. In this way, customers can easily invite customer supports to solve their issues directly in the project without having to send over a database or outfile. The feature also makes it possible for several users to help each other program destination lists at the same time. Figure 8.7 shows how the interface looks like when several users are working on the same project.

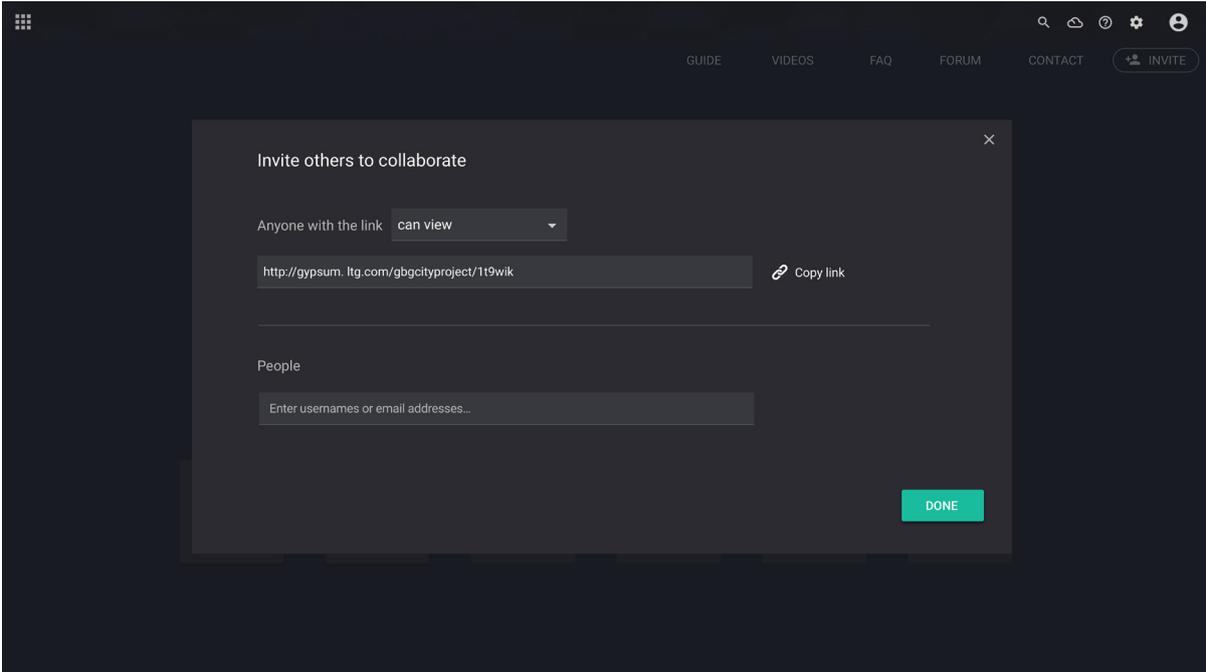
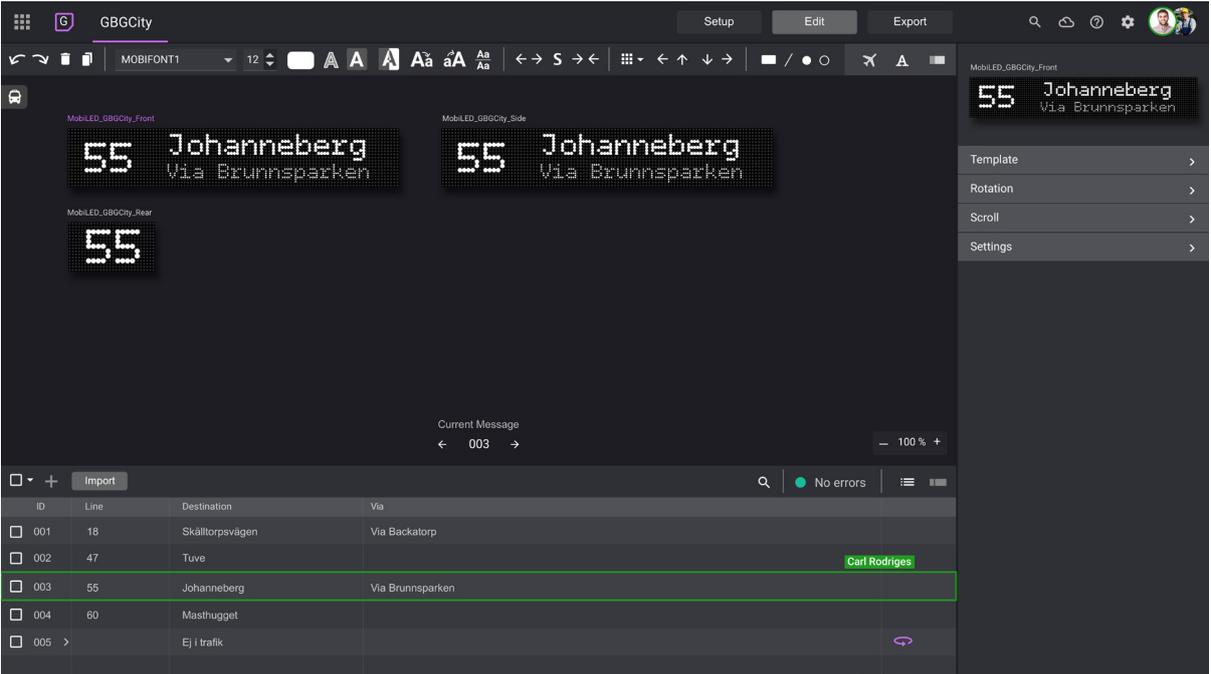


Figure 8.6 The collaboration invitation interface in the final prototype.



100 Figure 8.7 The example interface when several users are working on the same project.

8.2 Design Assumptions

Working with the final touches of the prototype based on previous iteration, more assumptions were made. In order to test the new ideas and the final look, the user validation sessions were fo-

cused on whether the new features are valuable for the users. Read the complete list of design assumptions in Appendix 8.

8.3 User Validation and Outcome

The assumptions and the final prototype was then tested in two demonstration sessions: one demonstration session with two expert customers in Germany (assisting editors) and one demonstration session with an external customer in UK (administrator). The first session was done with the help of a translator because of language limitations. Following was demonstrated in the sessions: setup a new project, create new messages, import destination list, use error report system, add effects, create graphics, create outfiles and the support system.

Result

In general the participants were impressed by the prototype. “Simple”, “easy to use” and “very pleasing, easy to use software with helpful functionality” were some of the comments which were expressed during the sessions. The participants initial impressions was that the software looks more modern and have a very intuitive user interface that is easy to understand. The participants stressed the fact that the main workflow is easy to understand. The participants expressed that it was difficult to give feedback without having tried it: “Our biggest wish it to get a working prototype as soon as possible to play with it. Then we can give you good feedback”.

Several points were raised about adding rotations during the sessions. First of all a better overview of the rotations was requested in the workspace. It was not sufficient with seeing the rotations in the message listing and the properties area. They referred to TEDPlus where all the rotations of a sign are displayed. Having all the rotations for one message displayed in the work area would make it possible to check them in one view. As it is now, you have to click several times in the properties to get the overview and that is not enough. They want to be able to search among the rotations and then copy and paste a display. The participants also asked for a lock function to copy text in following rotations (which is already implemented) and have a dedicated tool in the toolbar for it.

It was also discussed the differences between the use cases for rotations and scrolling. The participants confirmed that it was a good idea to have possibilities to add rotations to all signs with one message through the message listing and also to individual signs through the properties. The assumption was that scrolling differed from rotations in that way that there are no need for adding scrolling to several signs only to individual signs. It turned out that it was valuable for the users to apply scrolling to several signs at the same time. It also turned out that scrolling is not supported by all operating systems and one solution for this could be that unavailable functions can be gray and inactive.

The editors were among the attributes of the application which impressed the participants the most. Again they stressed the fact that it would have been valuable to try the software for real to try the different input methods. The participants expressed that in their previous software it was possible to use the left click on the mouse to light up pixels and the right click to remove pixels. This was something they thought should be in the new editor as well. The possibility to use images as underlay was also considered to be a great idea. They also liked the idea of using the keyboard arrows and spacebar. When it comes to the font editor, it was important to be able to import existing fonts. Many customers have company specific fonts and they would be upset if they would have to redo the fonts.

The template editor was questioned in terms of how it should be constructed. Currently the editor is designed according to the assumptions that templates can and need only to be designed blocks in the template are defined. Talking to the

users, they questioned the design by pointing out that often their users wants to define their input areas by the resolution of the sign and therefore by pixels. It was suggested that the user could choose between ratio and pixel. It was also requested to have better control of the sizes of the blocks in the template preview, for example by showing the ratio/pixel size.

Another aspect of the application which received some attention during the sessions was the collaboration feature. For them, it was important with the access control. Often they would like to share their project with only reading rights and not editing permissions. They did see value in the feature and therefore validated the assumption.

A complete list of potential improvements for this iteration can be read in Appendix 11.

8.4 Conclusion

In the third and final iteration the final touches were made on the prototype in terms of coherency, pixel position and workflow. Some additional functionality was designed. A help center with manuals, video tutorials and collaboration features was meant to provide better support

for especially beginner to shorten the learning curve. The prototype was tested and received positive results with comments such as simple and easy to use software. There were also some suggestion for improvements, see section 8.3.

09

FINAL DESIGN

This chapter describes and explains the final outcome of this project. The final delivery is a clickable prototype covering the main functionalities and features for destination sign editing, named Gypsum. The chapter is initiated with a short introduction explaining the foundation of the software, followed by descriptions of the software structure and main workflow with regards to three key interfaces. Thereafter, it is described how the final design is adapted to user differences involving the three key user types and the needs of a global market. Below is an image of the final design, Gypsum. More interfaces of the final design can be found in Appendix 14 and 16. The online prototype of Gypsum could be found at <https://sketch.cloud/s/ke21o/all/page-1/suite-overview/play>.

9.1 Design Foundation

This project resulted in a clickable prototype of a web-based editing software which provides a way of creating the destination text in exterior signs in LTG equipped vehicles, as shown in figure 9.1. The main users of the software are public transportation authorities, like Västtrafik in Sweden. The software covers the main functionalities and features a destination sign editing software should have to cover the needs of a global market. Gypsum also combines the valuable functionality of current software within the group and makes it possible to edit all LTG signs in the same software. It is a web-based application within LTG's digital product portfolio, the Luminator Suite.

According to the outcomes of user studies, the software will be used both internally and externally by LTG technicians and LTG customers.

Therefore, the software was designed with focus on satisfying the needs of three key user personas with different attributes that were presented in Chapter 5, namely Support Carl, Admin Mary and Fixer Rob. The software was adapted to three high level use cases that summarize when and why the software would be used, they are:

High level use case 1: Buy a new bus with LTG signs for the first time

High level use case 2: Update message listing due to schedule changes

High level use case 3: Make changes to bus fleet and update message listing



Figure 9.1 One example of destination sign on Gothenburg bus 16.

According to the project's initial requirements, the software is meant to be used as a web-based application, thereby the screen size of the software was set as 1366x768px with high resolution. The primary input methods the software supports are mouse, keyboard and trackpad. The software is designed for sovereign posture because it occupies the attention of the user for long periods of time and therefore follows some design principles for sovereign applications that are defined in the book About Face (Cooper et al, 2014), including:

Principle 1: Optimize for intermediates

Principle 2: Generous with screen real estate

Principle 3: Use minimal visual style

Principle 4: Provide users rich visual feedback

Principle 5: Support rich input

Principle 6: Design for documents

Following Cooper's design principles for sovereign software, intermediate users and common use cases were the main focus in the design of Gypsum. However, according to the user studies and three key user personas, see figure 9.2, a large number of users are beginners and experts. Thereby, addressing the needs of both beginning users and expert users was also considered in

the design process.

Besides, since users in different markets have formed different usage habits, satisfying the preferences of a global market, maintaining current users but also attract new users was another fundamental focus of the software design.

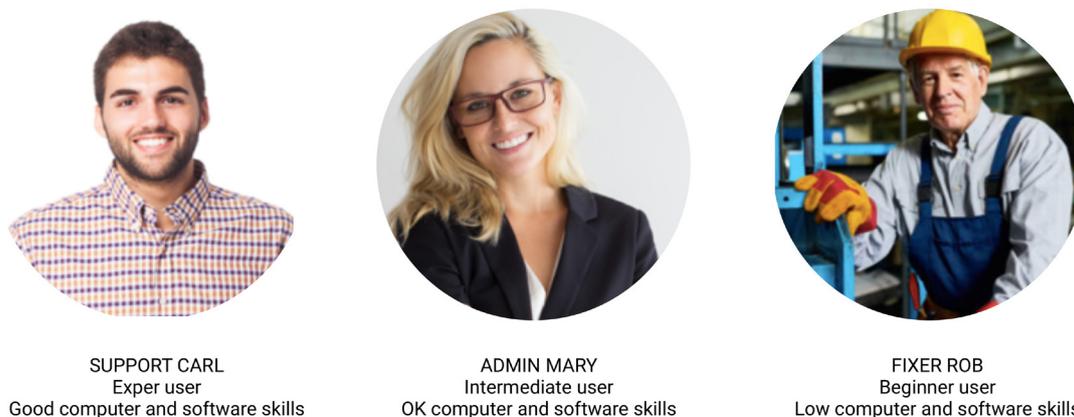


Figure 9.2 Three key user personas of Gypsum.

9.2 Software Structure

If a bus which runs in a city is used as an example, then it usually runs several routes with different sign content. This means all buses that are operated within the same area are expected to have the same program list in the control unit so that bus drivers could switch between different sign messages easily. For this reason, the software structure in Gypsum is designed to be project-based, and each project is supposed to contain all the vehicles that work in the same area. Considering ‘Gothenburg City’ project as an example, all buses that operate in Gothenburg city area should be included in this

project and share the same list with destination messages even though most often they are equipped with different destination sign sets in terms of sign type or sign size. Due to the technical and hardware limitations of existing signs, buses with different sign sets and control unit require different configurations on the outfiles. This means that each vehicle type require their own outfile. Therefore in Gypsum, the user has to select with vehicle type to export for and the software will generate separate outfiles for each vehicle, as illustrated in figure 9.3.

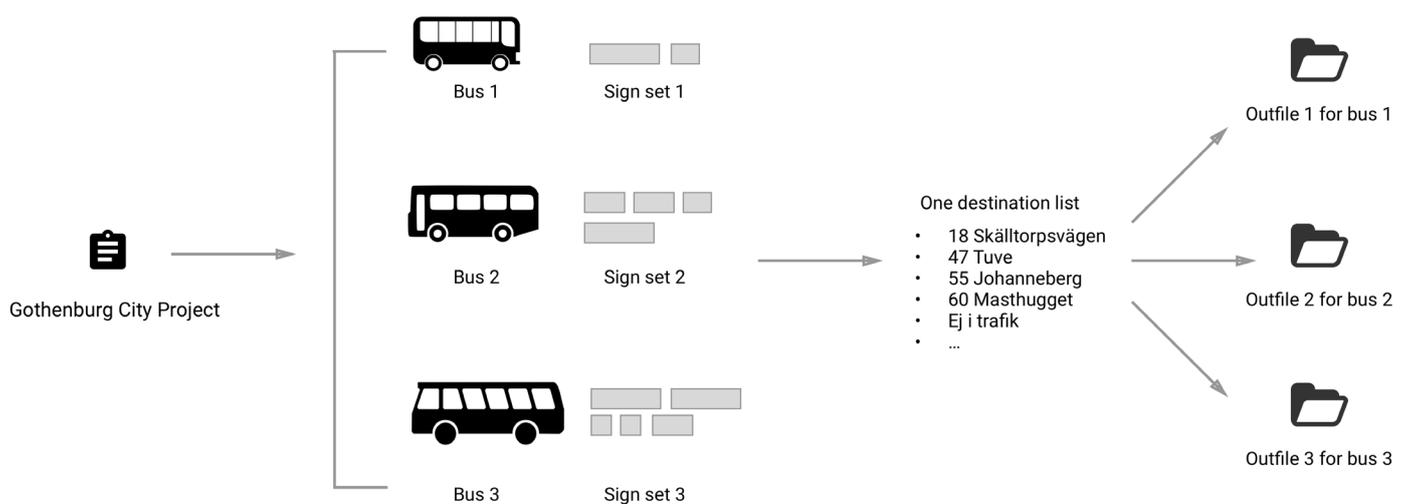


Figure 9.3 Visual representation of software structure.

9.3 Software Functionality

The software has a three-step-workflow which is adapted to existing software to support the mental models of current users. The three steps are Setup, Edit and Export. The software is designed for certain use cases which can be performed in each step, as figure 9.4 shows. By

clicking on the three buttons on the top bar in figure 9.5, users could navigate between the three key frames and perform the use cases, which will be described in following sections.



Figure 9.4 Software workflow and main tasks in each section.



Figure 9.5 Navigation buttons for setup, edit and export sections.

Setup

In the setup section, the users creates a new project with the vehicles and sign technology they have in their vehicle fleet (most commonly bus fleets). They then add the different vehicles types (different sign setup) in the vehicle fleet to the project by inputting configuration data including vehicle name, vehicle type, sign brand and control unit. An example of two different vehicle types could be one bus with LTG Sweden sign system and one bus with LTG Rastatt sign technology. This is followed by adding

signs to the vehicles through three ways, select from sign library, input sign article number and input order number. After setting up, all the vehicles and signs will be presented in the form of graphics to give users a clear overview, as shown in figure 9.6. By clicking on individual vehicle or sign, users can check relevant configuration data and make changes on them. If there is any changes of equipment, users can also come to setup interface and maintain configuration data here, such as create a new vehicle, delete signs or add new sign to existing vehicle.

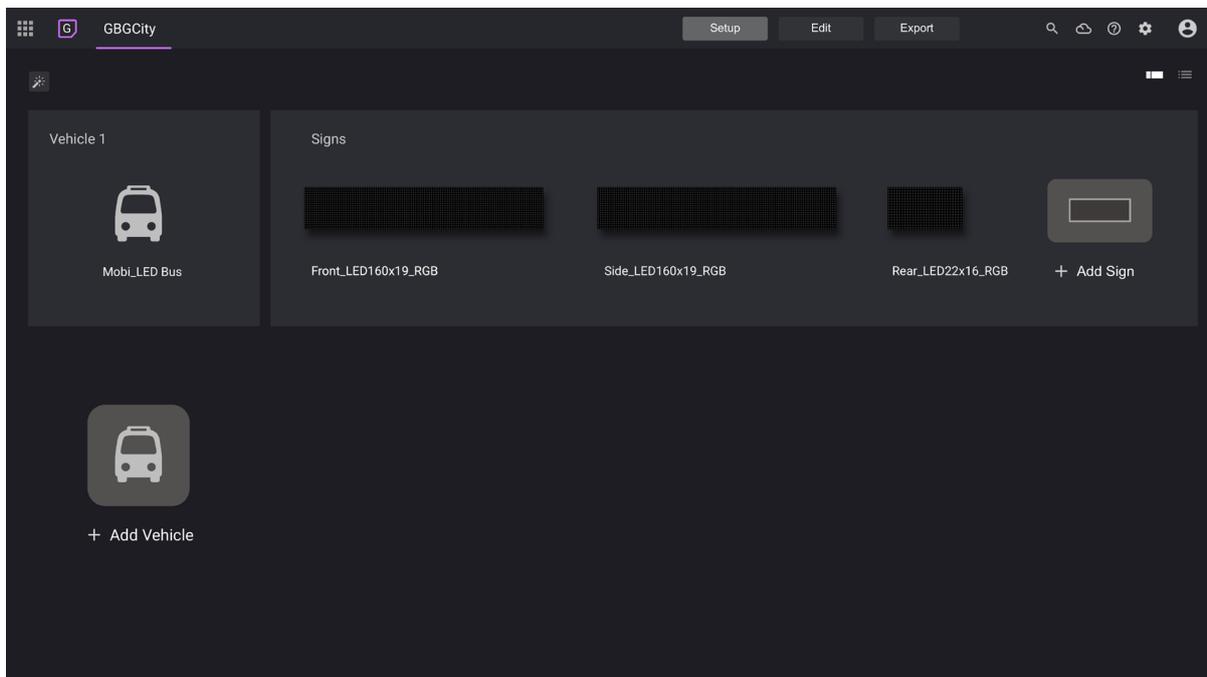


Figure 9.6 Setup keyframe in the final design.

Edit

In the Edit section, the content of the signs can be created. This interface, see figure 9.7, is divided into six parts, vehicle overview, toolbar, property, message listing, workspace and editors, more details about the six parts will be presented in appendix 14 software overview section. To create destination messages in the

signs, the user can input the text in the table at the bottom of the interface, the so called message listing. The table is divided into line, destination and via information. Depending on in which column the user input text, the sign will choose suitable template layout, see examples in figure 9.8 and 9.9. The text will then be auto-formatted into all the signs by adapting text size and font so that the message fits in the sign.

The message listing allows users to edit all signs simultaneously without doing repetitive work. If the auto-formatting would not work, the software is equipped with an error report system, see figure 9.7. In reality, most users receive an Excel sheet with destination messages to program in the editing software. To support this behavior, the software has an import function. If users want to make changes on specific signs, then

they can click on individual sign and edit it with toolbar and properties to the right part of the interface. The user can also add effects like scrolling and signs with several displays (rotations) here in the Edit section. Gypsum also have some additional customization possibilities such as Graphic, Font and Template editors, more about the editors can be found in appendix 16.

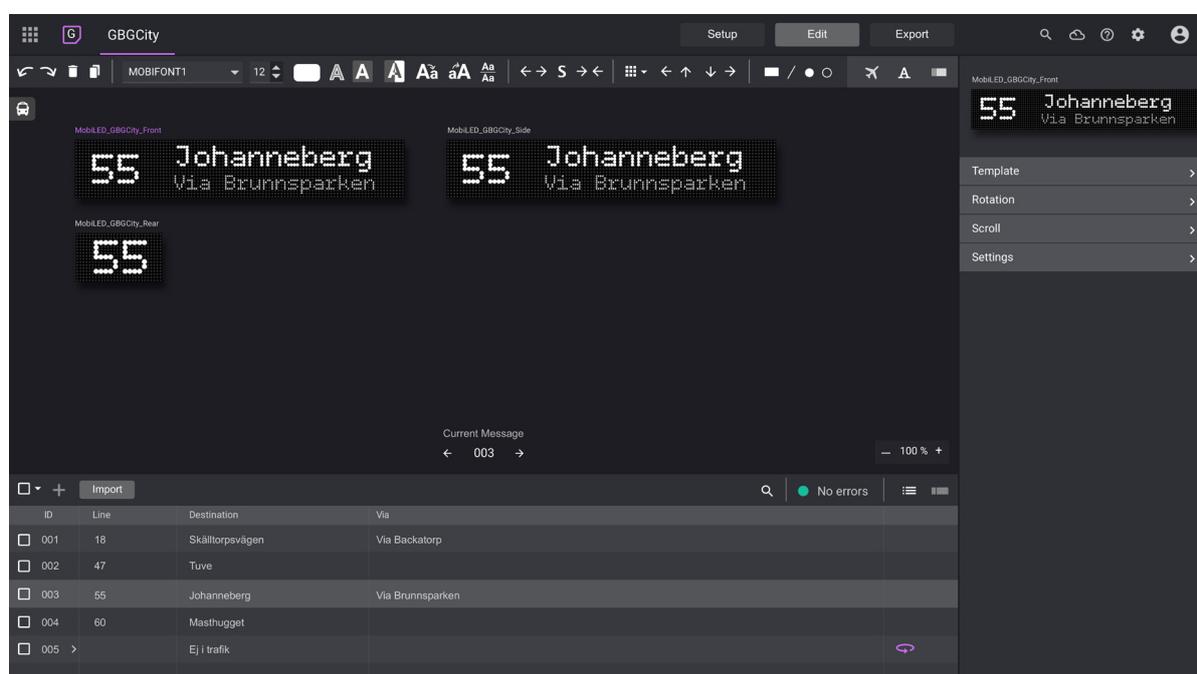


Figure 9.7 Edit keyframe in the final design.

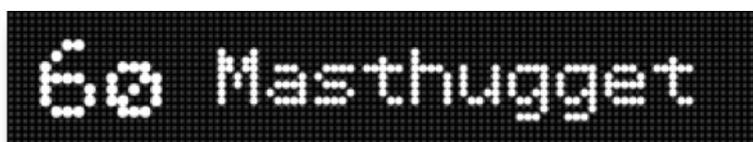


Figure 9.8 Sign display for message 004 with line and destination content.



Figure 9.9 Sign display for message 005 with only destination content.

Export

When all the editing for all signs are finished, in Export section, users can export their destination messages in a format which can be plugged in to the vehicles in their fleet, see figure 9.10. The software will export individual outfiles for each vehicle type. By ticking the vehicles and messages, users could select which vehicle and message they want to export and the download

will start accordingly. Users can also print documents for other purposes, such as PDF file of the destination messages or a complete list of sign previews of how the messages looks in the sign for each message. This function is useful for bus drivers when they select a message ID to enter in the control unit in the vehicle. The signs in the vehicle will then display that message in the signs.

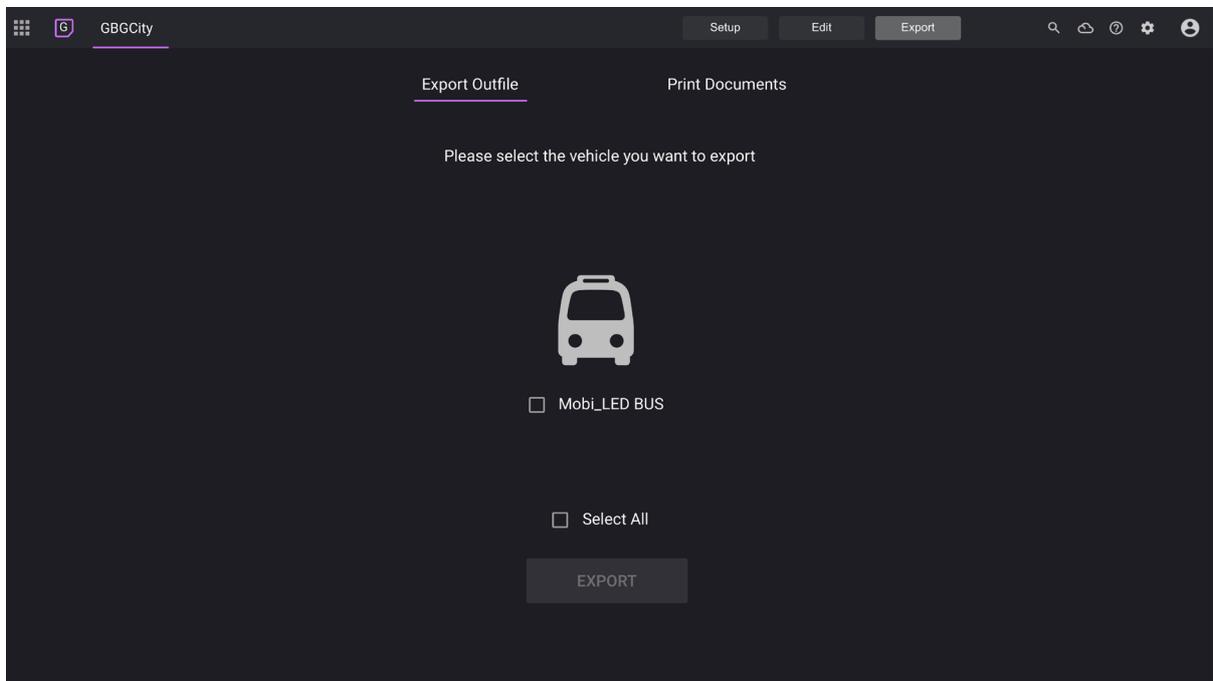


Figure 9.10 Export keyframe in final design.

9.4 Design for Differences

This section discusses how Gypsum is designed to adapt to the differences in terms of user types and markets by presenting more functionality within the application.

Design for beginners

Beginners refer to the users who have limited or no experience with the software. In this project, users who have used the software for a long time but still have problems with using the destination editors, such as the Mechanics user types, are also referred to as beginners. Typically, beginners are sensitive and can get demoralised in the first time usage. For beginners, the design of the software should help them get familiar with the software and move to intermediates rapidly. In order to satisfy the needs of beginners, three strategies were developed and followed to make the interactions between beginners and the software more intuitive and effective, they are:

- *Provide instructions and clues without disturbing current tasks*
- *Provide step by step guidance for first time usage*
- *Offer assistance whenever users need it*

Provide instructions and clues without disturbing current tasks

Instructions and clues are provided for beginners to let them understand and learn the interface without disturbing their current work. In the final design, instructions and clues are provided in the forms of graphical feedback and guiding

text without disturbing current tasks.

In digital product design, providing visual cues is a good approach to ease the interactions between users and the product. Considering many beginning users of Gypsum have relatively low computer skills, graphical cues are largely used in the design of interfaces. For instance, when setting up a project, the buttons for adding vehicle, importing databases and adding signs are visualized with both graphics and text, as figure 9.11 shows, to give beginners clear clues regarding what these buttons could do.



Figure 9.11 Graphical buttons in the final design.

Another example is that when signs and vehicle are created, visual clues will be presented in the setup interface to offer beginners straightforward visual feedback on what was created, as figure 9.12 shows.

Also, in the edit interface, when no message is created or no sign is selected then there will be short text descriptions to tell users how to active then, as shown in figure 9.13.



Figure 9.12 Graphical feedback for created vehicles and signs

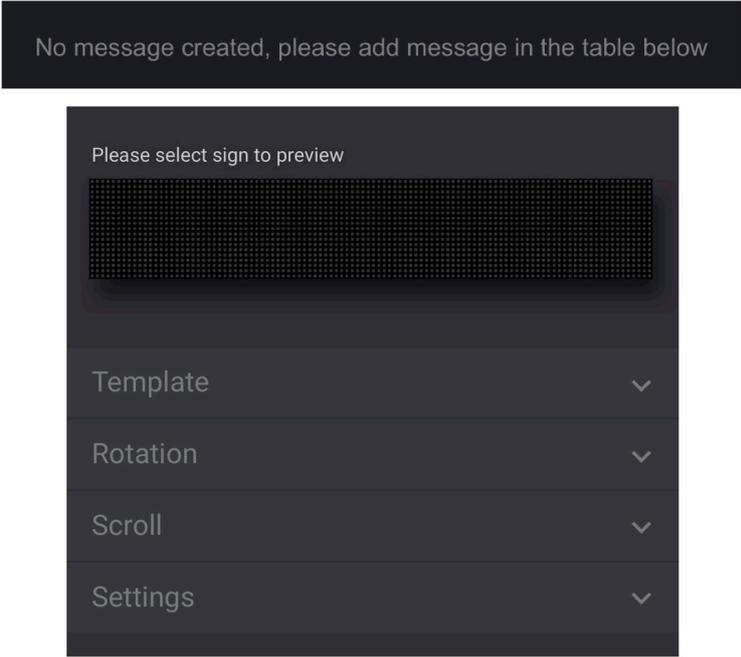


Figure 9.13 Text instructions and clues when no messages is created or no sign is selected.

Provide step by step guidance for first time usage

In a digital world, it is always crucial to let first time users understand the big picture by answering ‘what does the software do’, ‘what is the main flow’ and ‘how to get started’. In Gypsum, step by step guidance offers understanding of the big picture for first time users. The guidance is designed from three aspects, first time guided tour, setup wizard and concentrated view.

First time guided tours

Guided tour appears when a user enters Gypsum for the first time to provide explanation of features and interface behaviours through a sequential set of text boxes. Figure 9.14 illustrates

one example of the message listing in Gypsum. Each box is composed of current section name, what users could do with it and a link to tutorial videos. Users can grasp the main workflow and expected interactions rapidly by taking the users through a sequential process, setup, edit and export, as well as providing description for the basic functions in each screen. Of course, for users who want to get straight to work without visiting each interface they could click on the blurred background to skip the tour.

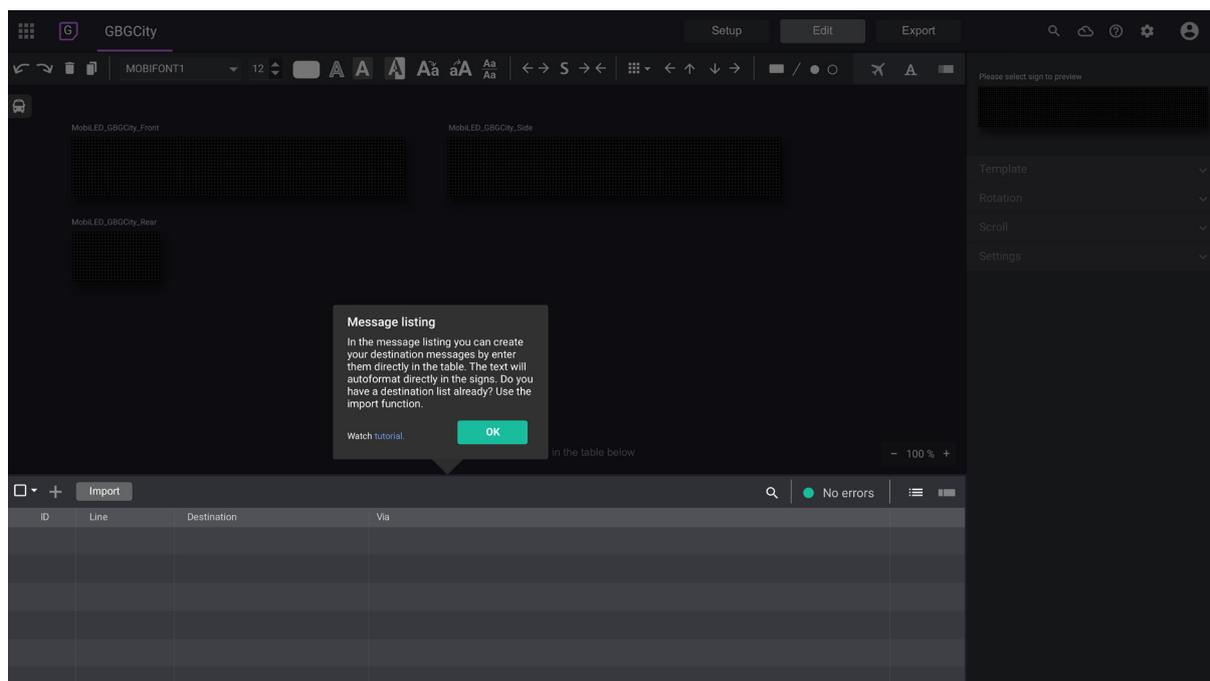


Figure 9.14 First time guided tour in the final design.

Setup wizard

In the user studies, it turned out that the setup process is often difficult for the users. Therefore, a setup wizard will provide a step by step guide tool help users to setup vehicles and signs for the first time. Different from normal setup, when the users enter the project for the first time, a centralised setup wizard will appear and ask the users to input specific information in each screen, as shown in figure 9.15. In this way,

users do not need to bother with the big interface and can focus on the current task only. For intermediate or expert users, they can exit setup wizard by clicking the cross at the upper right corner and get a better overview in the normal setup interface. In the normal setup interface, by clicking the wizard button at the upper left corner users could come back to setup wizard mode. For more details on the setup wizard can be found in Appendix 16.

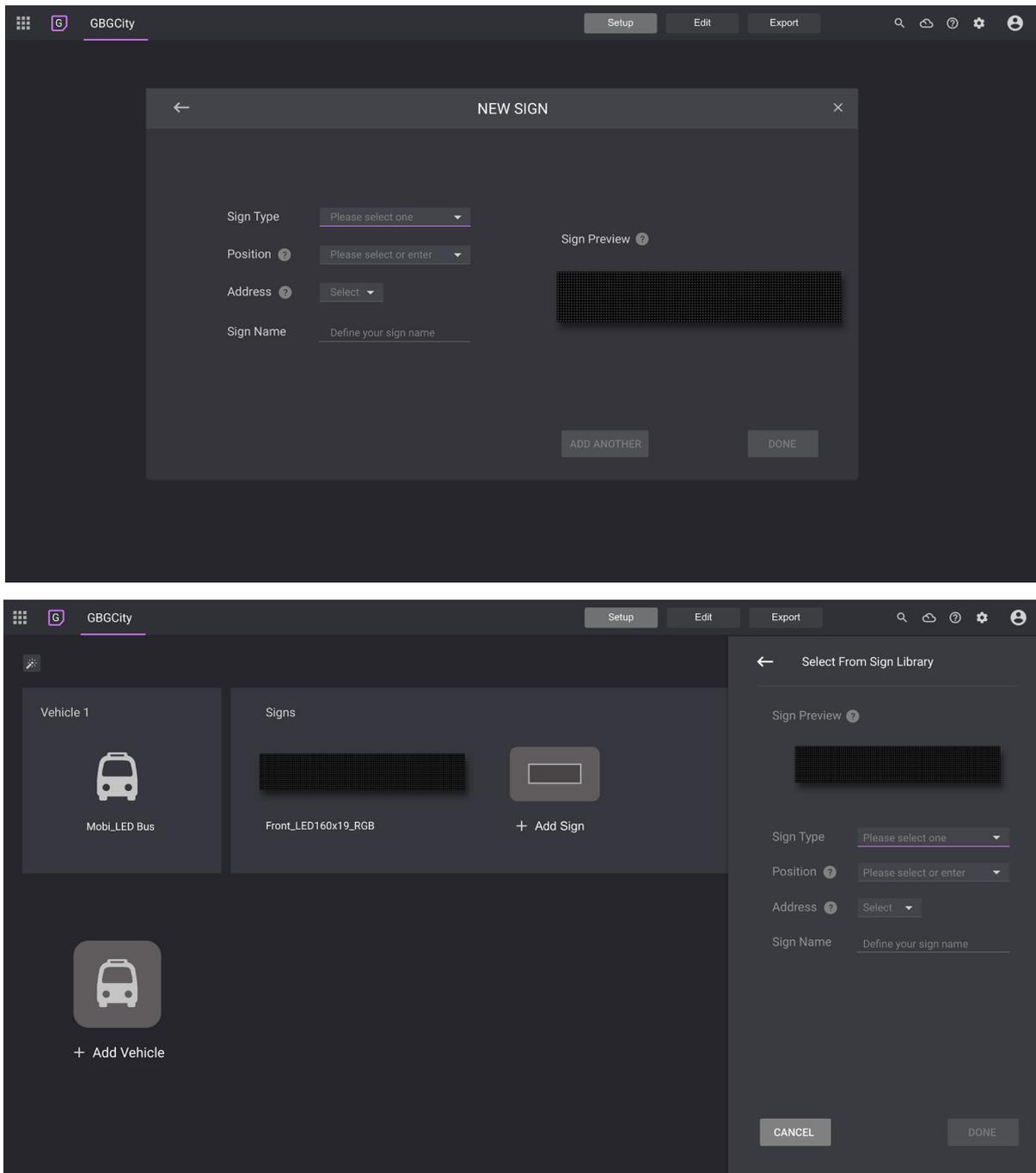


Figure 9.15 Setup wizard v.s. normal setup in the final design.

Focused view

Designing for sovereign posture often involves presenting much information and actions at the same time and this can be an obstacle for beginners. In order to avoid this situation, Gypsum offers Focused view and Table view in setup and message listing interfaces to let users choose the one they prefer. Taking message listing for

example, as shown in figure 9.16, Table view shows a list of messages while Focused view shows one message at the time. For users with different experience levels, they could switch between the two views to get a more concentrated workspace for one message or a better overview for all messages.

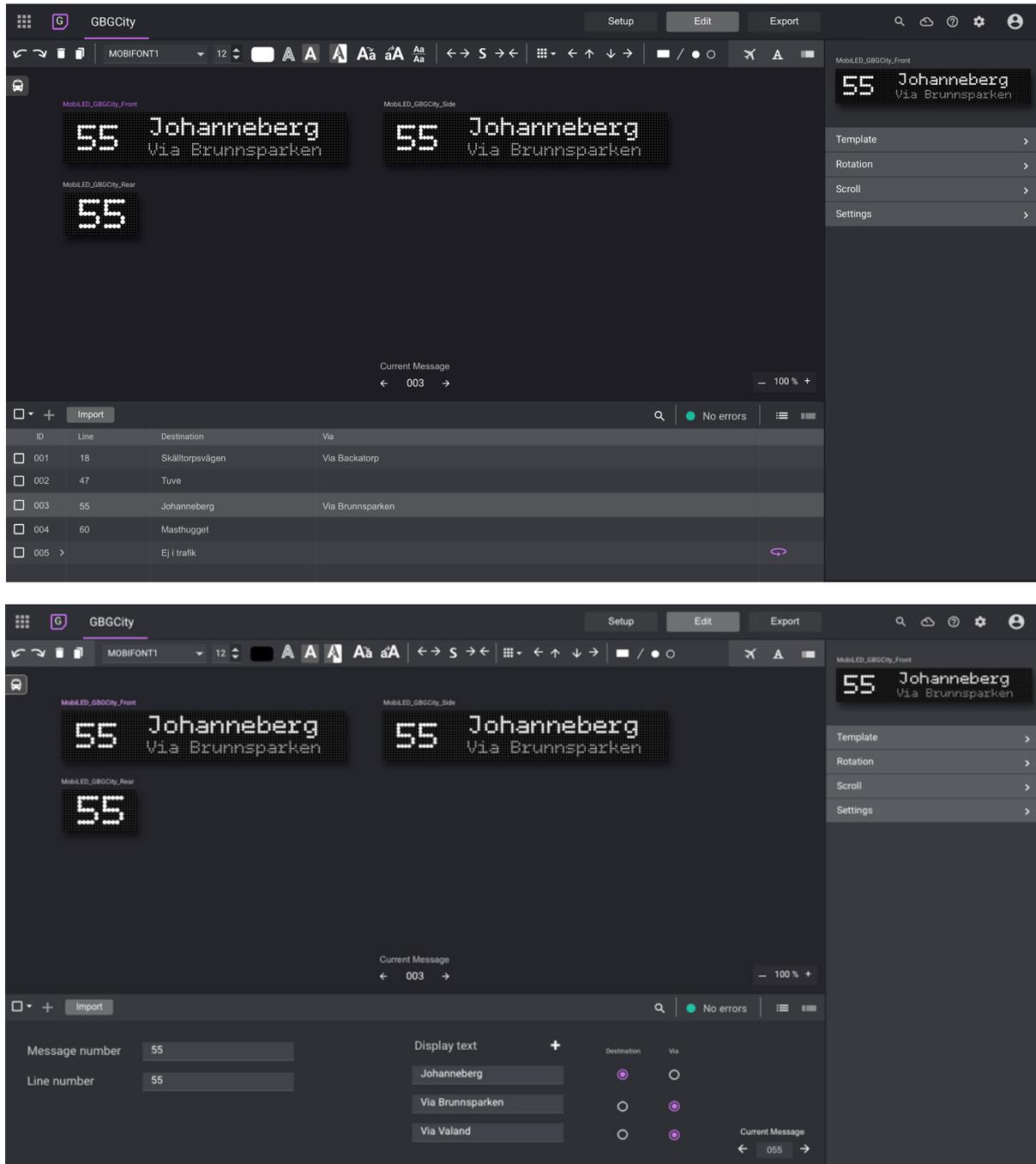


Figure 9.16 Table view v.s. focused view for message listing in the final design.

Offer assistance whenever users need it

A good software design is expected to provide assistance when and where users need them. Lack of assistance is also a pain point of current LTG destination sign editing software. Today, external customers often have to contact LTG technicians directly to seek help which can be very frustrating and increases the actual costs to both sides. To change this situation, Gypsum, offers various ways of assistance including explanation labels, help center and collaboration feature are developed to provide assistance to users whenever they need it.

Explanation labels

According to the outcome of user studies, destination sign editing software requires users to

have good knowledge of the hardware and relevant terminologies such as pixel, pitch, address, and sign type. For beginning users or people who do not have this domain knowledge setting up a project without help from LTG technicians it can be extremely difficult. In order to deal with this pain point, explanation labels were designed in the setup interface to help beginners understand the terminology and guide them on how to find required information. Considering sign article number in setup interface as an example, as shown in figure 9.17, when the user clicks on the question mark icon an explain text box will appear to tell users what article number is and where to find it on the real sign.

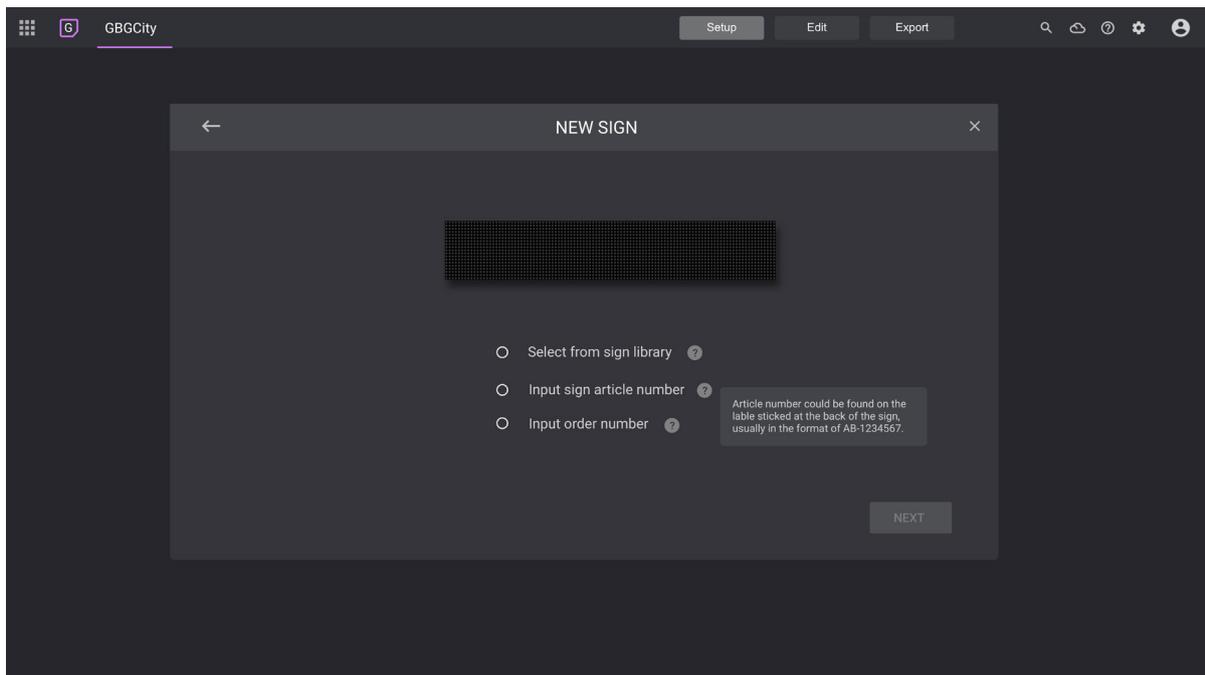


Figure 9.17 Explanation label in setup interface in the final design.

Help center

Help center is designed to support users in their everyday usage. By clicking on the question icon on the top navigation bar users could enter the help center that offers various kinds of support, as illustrated in figure 9.18. Here users could find step by step guidance, video tutorials, FAQ, forum and contact LTG directly. They could also search for specific topics in the whole

website.

Figure 9.19 is an example interface for the video section that presents all tutorial videos with tips and tricks for getting the most out of Gypsum. Help center is not only designed for beginners but also for intermediates and experts who want to develop more thorough understanding of the software.

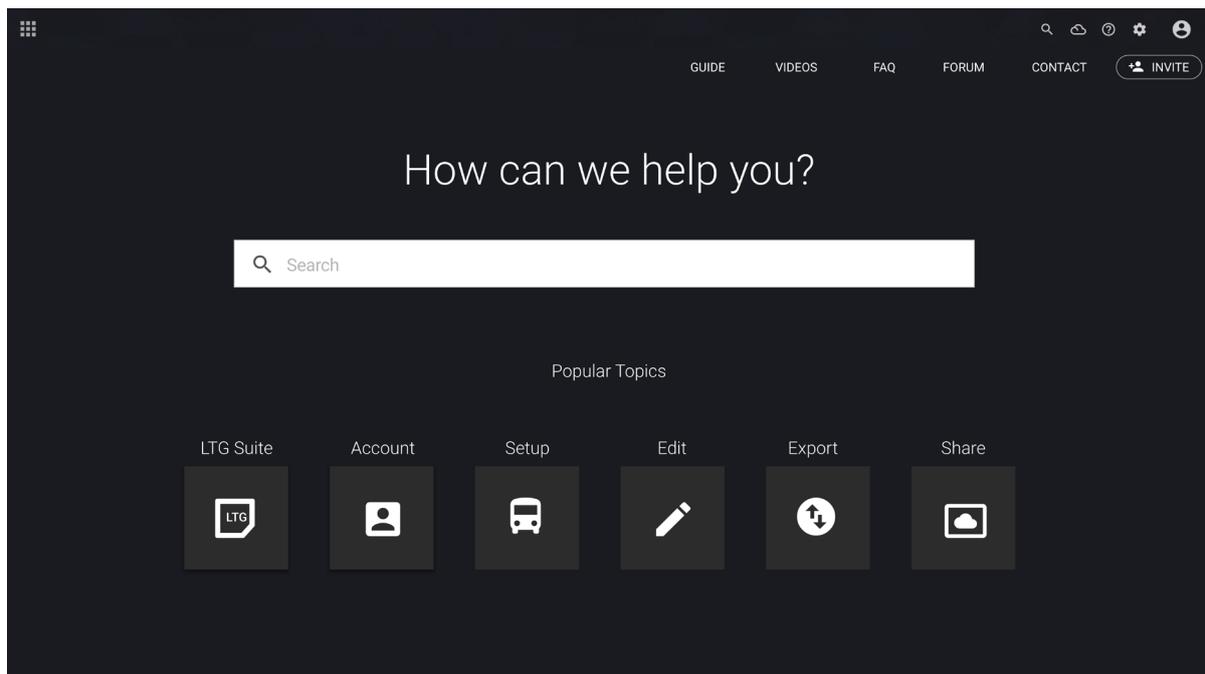


Figure 9.18 The help center interface in the final design.

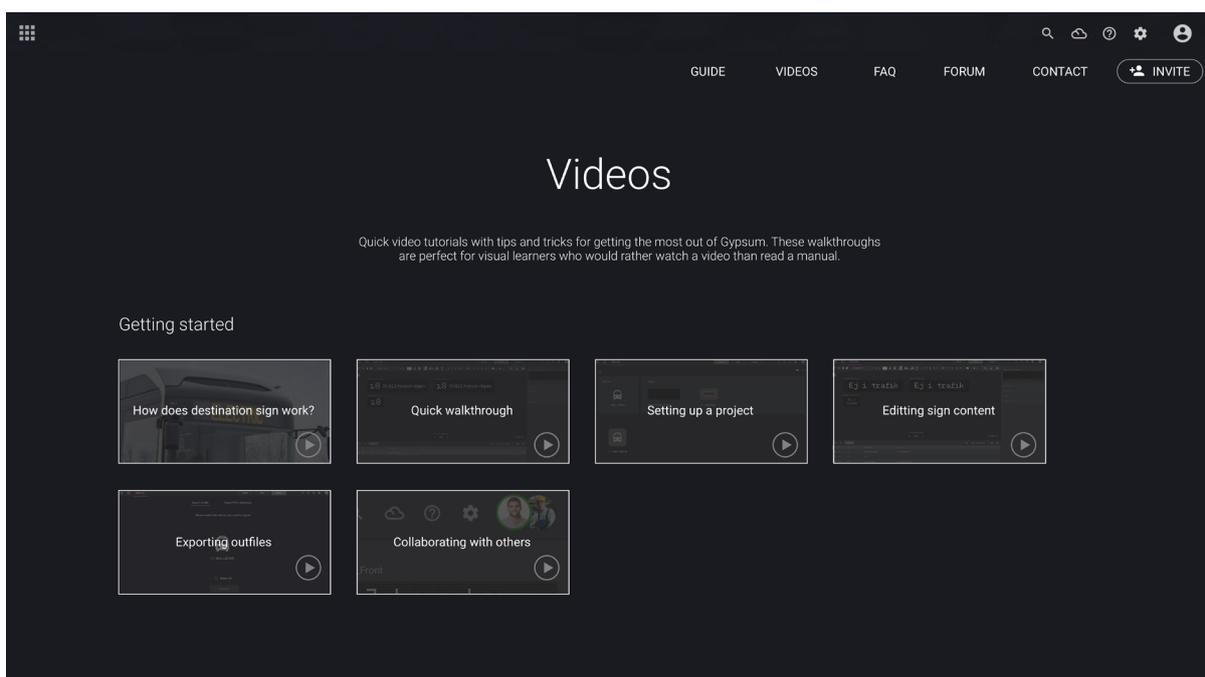


Figure 9.19 The video tutorial interface in the final prototype.

Collaboration

Collaboration feature is developed to enhance the collaboration between customers and LTG technicians in terms of seeking help or trouble shooting. In help center, users have the possibility to invite others into their project and give them different access permissions such as

view only, comment only or edit permission, as shown in figure 9.20. The feature also makes it possible for several users to help each other program destination lists at the same time. Figure 9.21 shows how the interface will look like if several people are working in the same project.

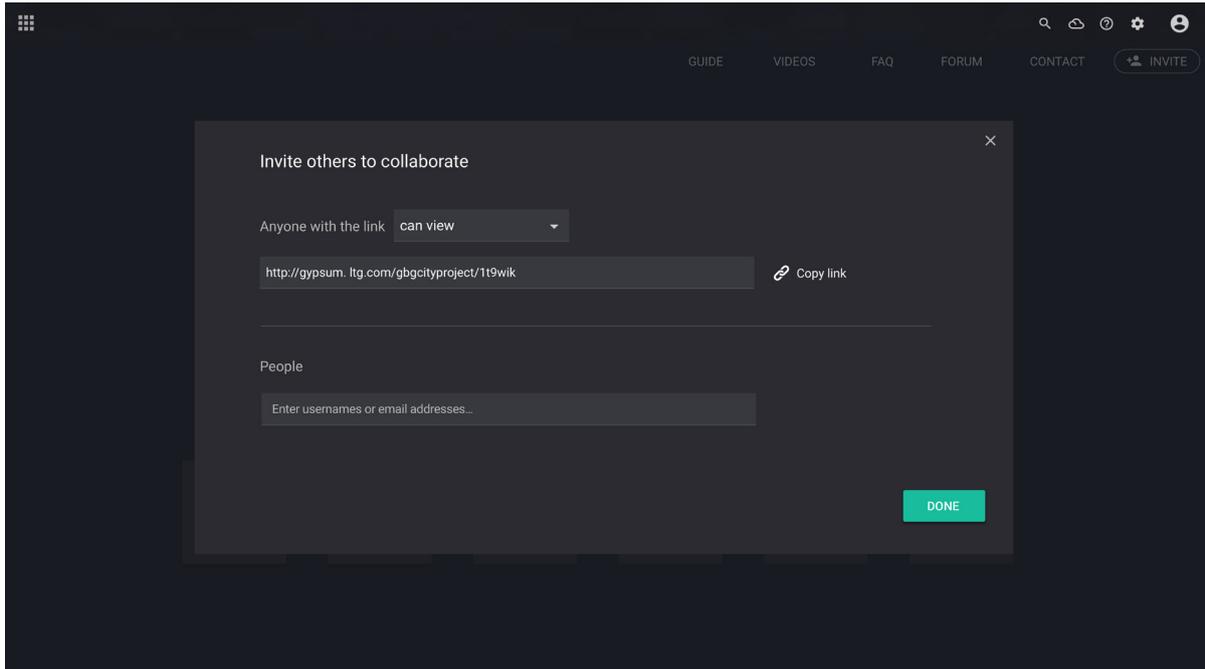


Figure 9.20 The collaboration invitation interface in the final prototype.

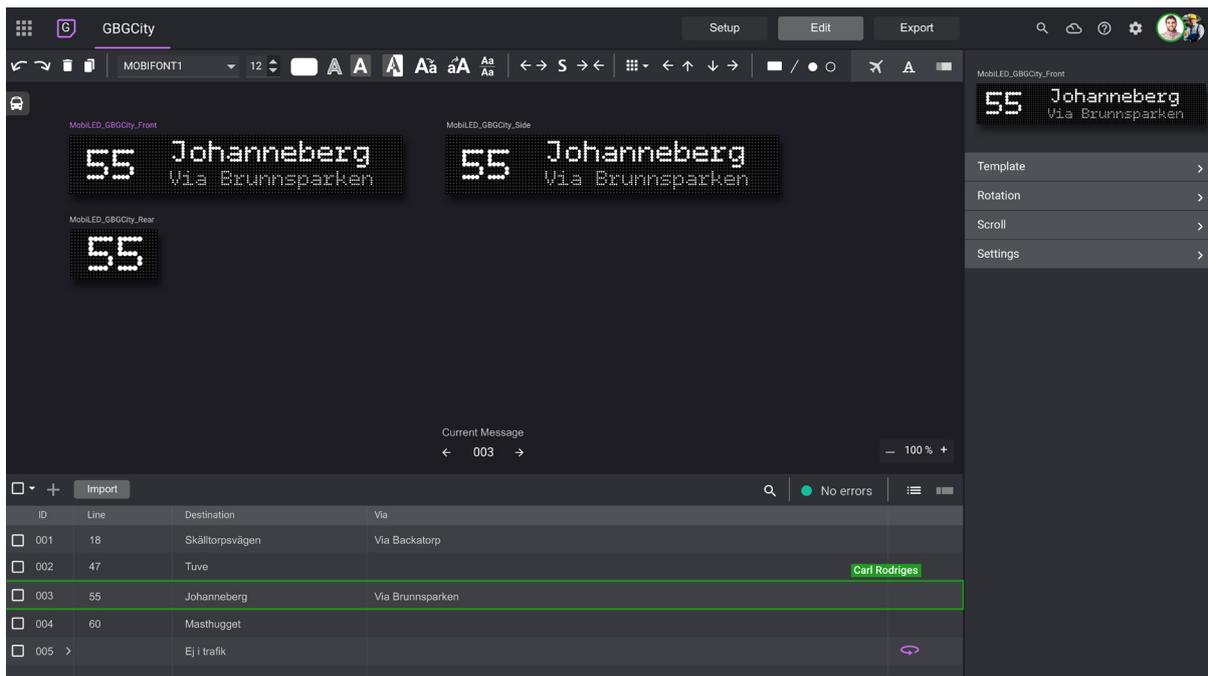


Figure 9.21 The example interface when several users are working on the same project.

Design for intermediates

Following Cooper’s design principles, in general Gypsum is designed for intermediate users (the Administrator user type is considered to be an intermediate user). This is reflected in covering the most common use cases, optimizing the interface for full screen use and following the workflow the intermediates already used to. Besides, some others features were also developed to optimize for perpetual intermediates specially, including

- *Tooltips*
- *Good overview of massive data*
- *High customization level*

Tooltips

A tooltip refers to the small text box with information about the item being hover over. Compared to beginners, intermediates do not need software scope and meaning explanation as they already know this things. Instead, tooltips say nothing about the scope and meaning, they only state function with little time and space consumption, which makes tooltips become the perfect intermediate tool. Figure 9.22 provides some examples on how tooltips should look like in this software. Adding a tooltip showing the full names of each buttons offers not only intermediates but also, beginners and experts a quick reminder about the functionality and when do they need them.

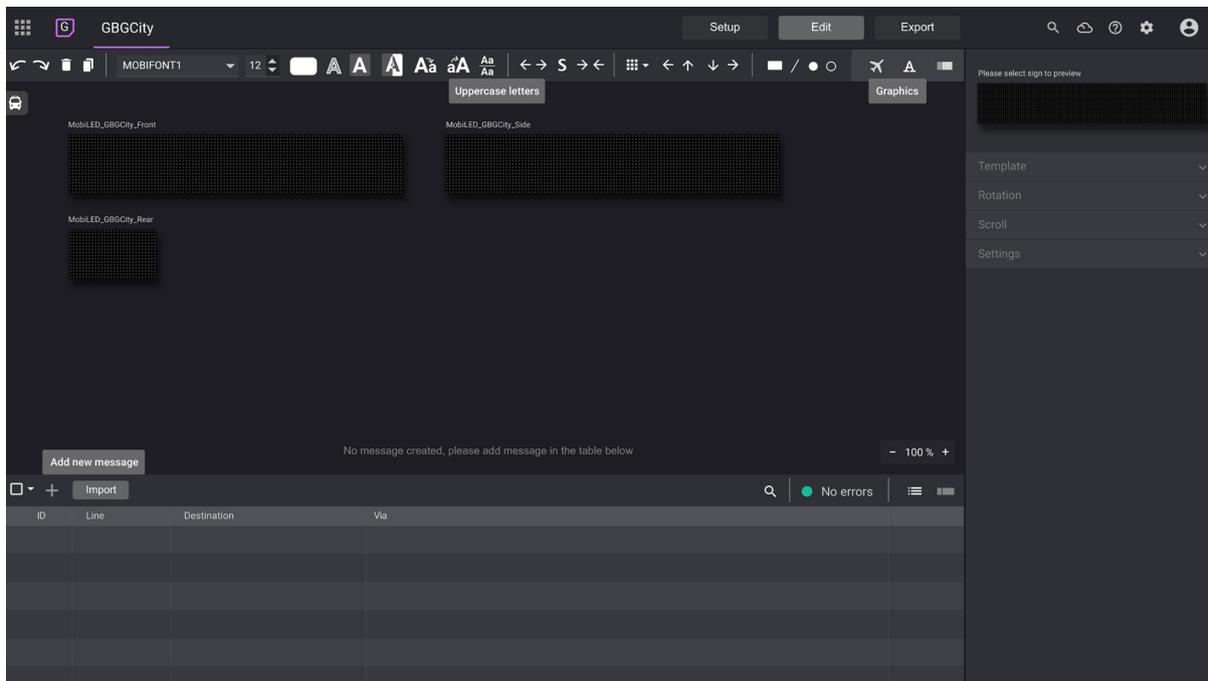


Figure 9.22 Examples of tooltips in the final design.

Good overview of massive data

As discussed in Design for beginners section, Gypsum offers two types of view, Focused view and Table view in setup and message listing interfaces to let users choose the one they prefer. Figure 9.23 shows the differences between Focused view and Table view in setup interface. The Table view is designed for intermediates and experts who want to get an overview of massive data rapidly. The Table view does not have much graphics and displays more detailed information about created vehicles and signs, in this way, intermediates and experts could go through them and identify potential problems quickly.

High customization and automation level

Compared to beginners, intermediates have higher requirements on the flexibility and customization of the software. Beside destination text editing, some features that give them the freedom to do more advanced editing are developed to satisfy their needs, such as to let users create their own templates, graphics and fonts instead of only using the default ones. Furthermore, Gypsum also has a high automation level by providing functions such as auto-formatting and import for message listing, upload image as underlay in graphic editor, error report to find errors quickly and so on. More details about these features will be presented in appendix 16.

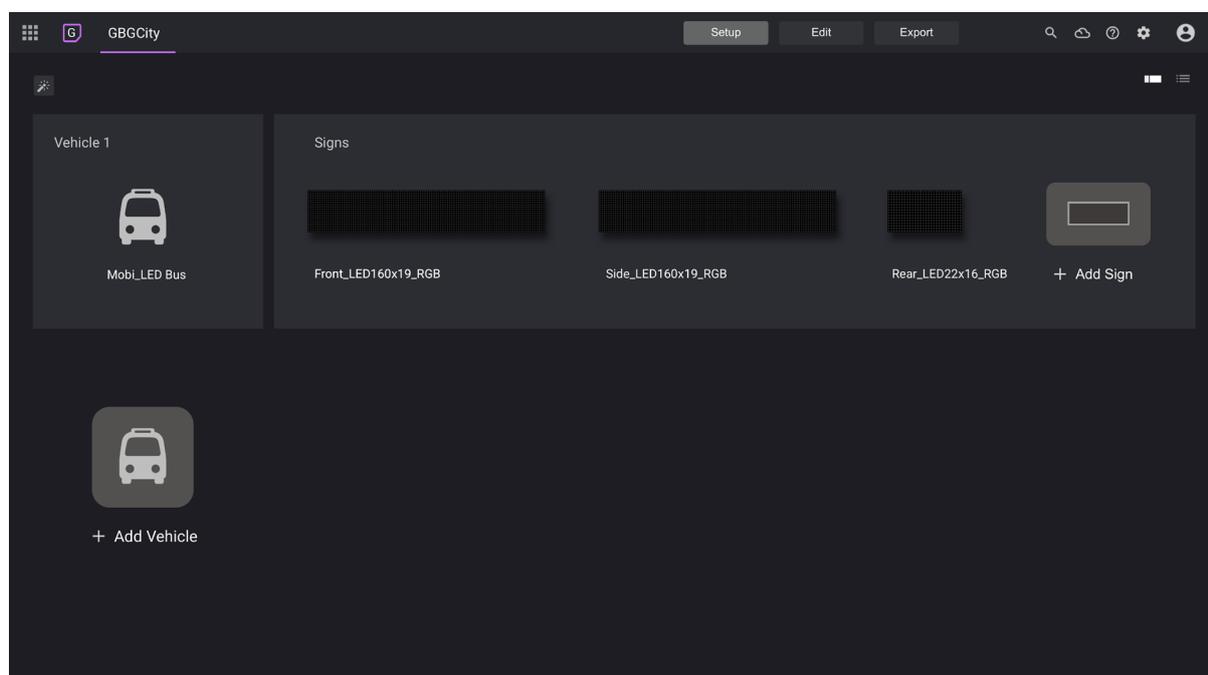
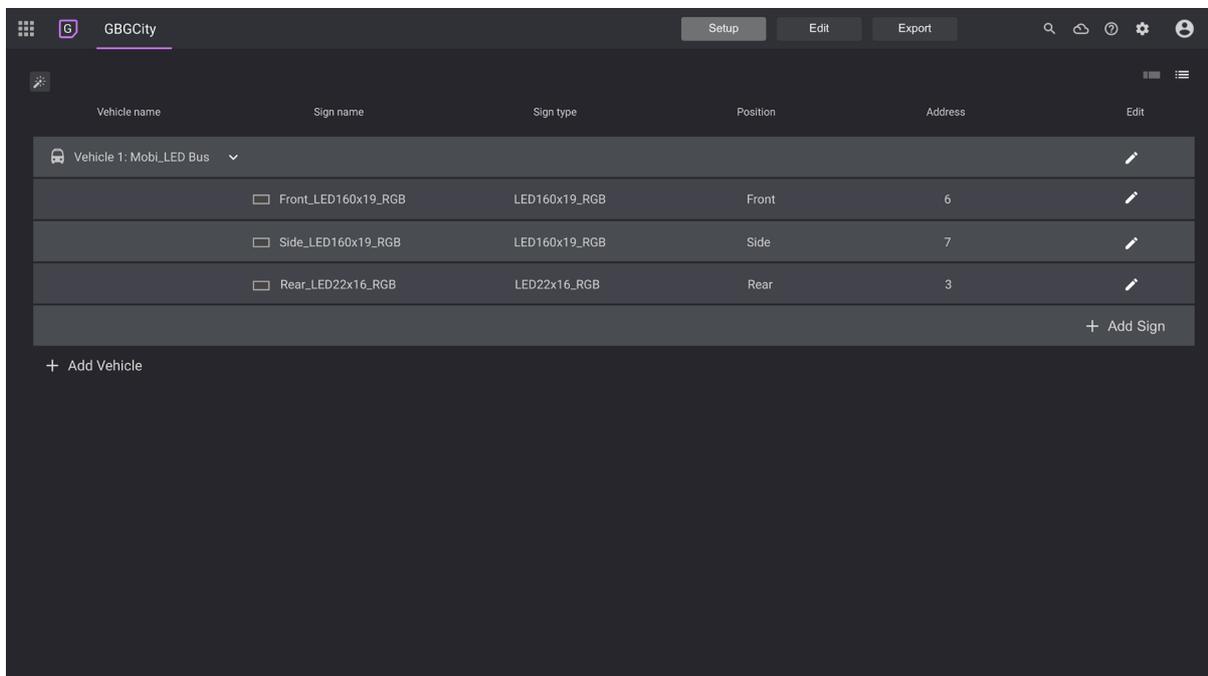


Figure 9.23 Table view v.s. focused view for setup interface in the final design.

Design for experts

The software will be used by both internal and external users which means a big part of Gypsum users will be internal expert users who have high computer and software skills. The user type Customer supports can be considered as experts. Designing for experts who already formed good understanding of destination sign editing software is also vital in this project. However, intermediates and experts often share some similarities in terms of rich data display, customization and automation, many features mentioned in design for intermediates also apply to experts. This section presents how the design of Gypsum is optimized for experts as supplements to the points mentioned above from three perspectives,

- *Support rich input methods*
- *Give users different access permissions*
- *Advanced settings for experts*

Support rich input methods

Expert users usually use the software for hours in a day and want their work get done quickly and efficiently. Different from beginners and intermediates, experts know the functionality of the software by heart and want shortcuts to everything. Gypsum currently supports mouse, keyboard and trackpad inputs. Based on this, some keyboard shortcuts are developed to allow experts to achieve some common tasks without using mouse so that they could keep their working path on keyboard. The hotkeys in Gypsum were developed by following the international standards for common actions and initials for other actions. Below are some examples of the hotkeys in Gypsum.

Copy (Ctrl+C)
Paste (Ctrl+V)
Add new message (Ctrl+M)
Add new rotation (Ctrl+R)
Uppercase letters (Ctrl + U)

Besides, in the message listing, font and graphic editors, instead of using right and left clicks users could also use the tab key, direction keys and spacebar to control the movement of blocks and the drawing of pixels.

Give user different access permissions

Today, depending on customers' preferences and computer skills, some customers prefer LTG customer support to setup the project for them and hand in the file to them and they continue the editing work, while some customers want to do the entire project by themselves. Gypsum offers the possibility for customers to setup the project by themselves, however, some internal expert users have the concern that if novice ex-

ternal users were given access to settings then they might mess up the project. In order to deal with this project, Gypsum is designed with the feature that allows internal users to give external users different access permissions to the software. For instance, LTG experts could lock the setting sections for them and do all the setup for them, as shown in figure 9.24.

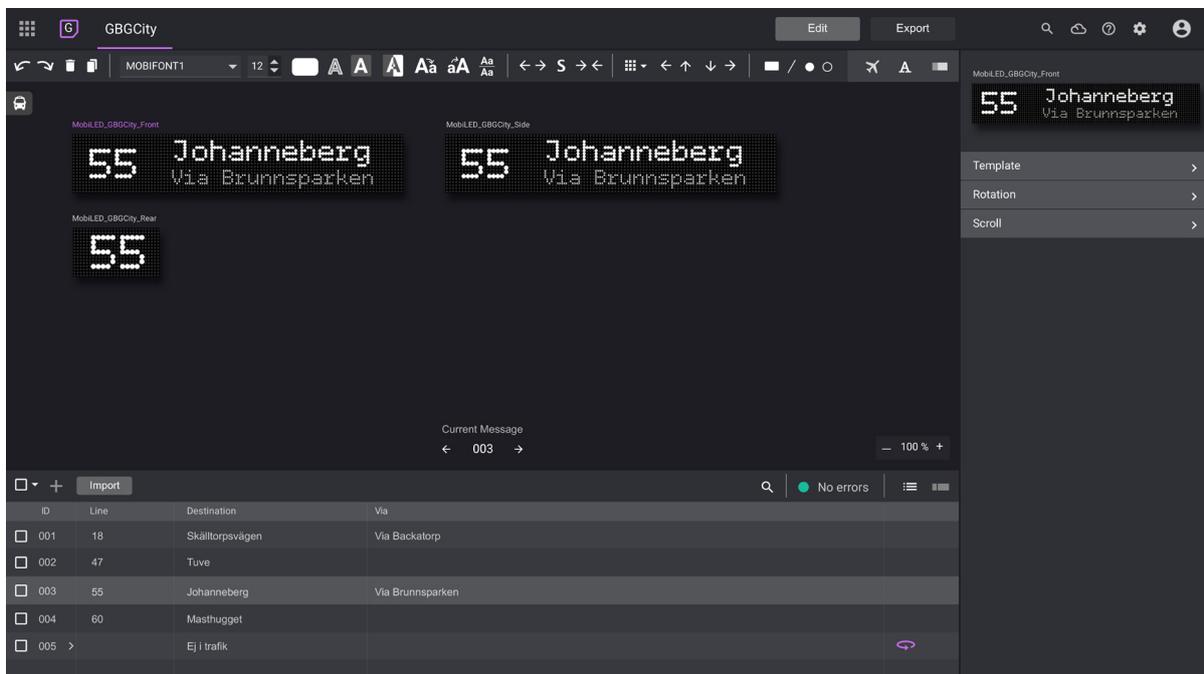


Figure 9.24 Limited setting permission interface for novice users without setup tab and settings tab in properties

Master settings

Based on the user studies of internal expert users, many of them want full control of the editor and make everything customizable. In order to design for this, a master setting feature is developed to allow users customize everything, example settings could be define the template for specific sign size instead of using the one suggested by the software, define the default font for message listing, or the turn on and turn off of

some functions. With the help of this master setting function, experts could customize the way Gypsum work to the ways they prefer.

Design for global market

LTG offers several sign brands for their customers, currently the main market is Europe and US, but also some customers in South America, Asia and Australia. It is important to design for the global market and adapt the software to local needs. This section discusses how Gypsum is designed to achieve this from three aspects,

- *Support all sign brands in LTG*
- *Adapt the interface according to the preferences of different markets*
- *Support various usage scenarios*

Support all sign brands in LTG

Currently, there are five sign brands in LTG that are used in different markets and each brand requires their own editor for content editing. As a

replacement of current editors, Gypsum is designed to support all sign brand in LTG to maintain the users in current markets. Therefore, as shown in figure 9.25, when setting up a vehicle, users are required to select the sign brand that current vehicle is equipped with so that the editor could configure for that brand. And in the export stage, if users have different sign brands, then the editor will generate different formats of outfiles for different brands.

In this way, users that are using other LTG editors currently could switch to the Gypsum easily and still keep working on their previous databases by importing them. Supporting all sign brands is a vital function of Gypsum to let existing users accept it as well as make it used globally.

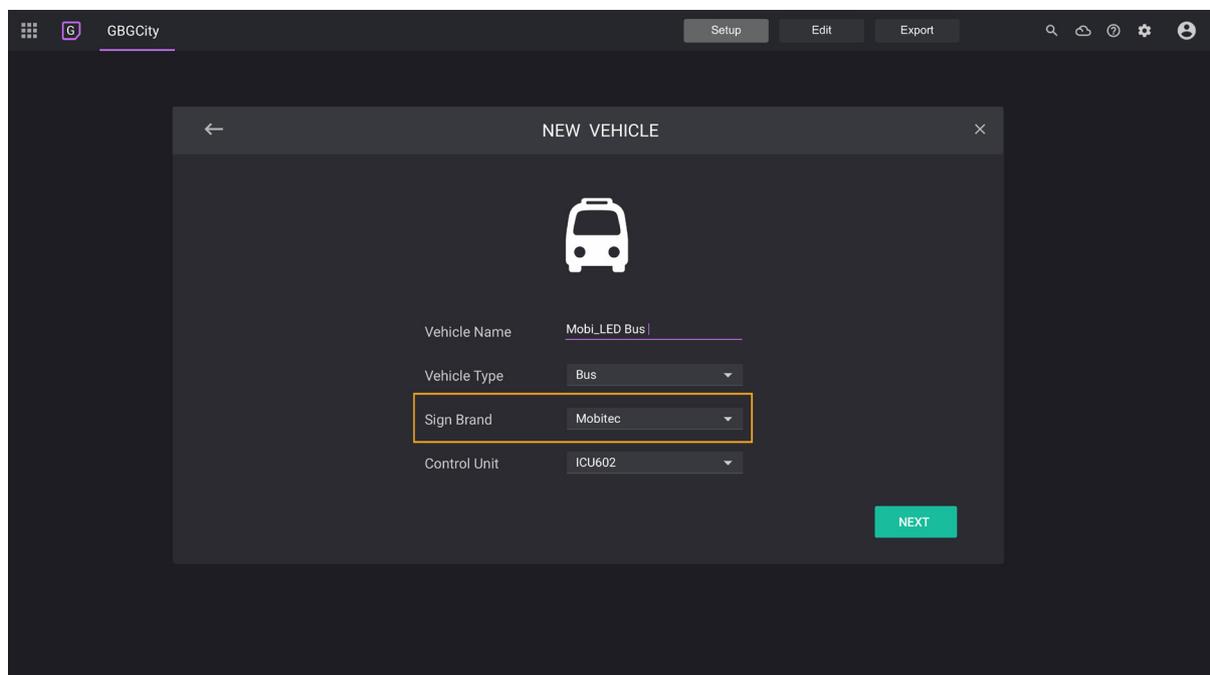


Figure 9.25 Sign brand selection in setup interface in the final design.

Adapt the interface according to the preferences of different markets

According to the previous user studies, even though the main workflow of destination sign editors are similar, users in different markets are used to different terminologies and having different usage habits, for example US users require PR message display that shows some greeting text and this is not used in Europe. In order to satisfy special needs of different markets, Gypsum is designed to change its inter-

faces slightly depending on the user's location. When registering LTG account, the user will be required to select the country and based on this information, the interfaces and terminologies could be changed to the way this market prefer. Taking PR message as an example, if a user is from US market then his/her message listing will have a message type selection that allows them to switch between PR messages and normal messages, as shown in figure 9.26.

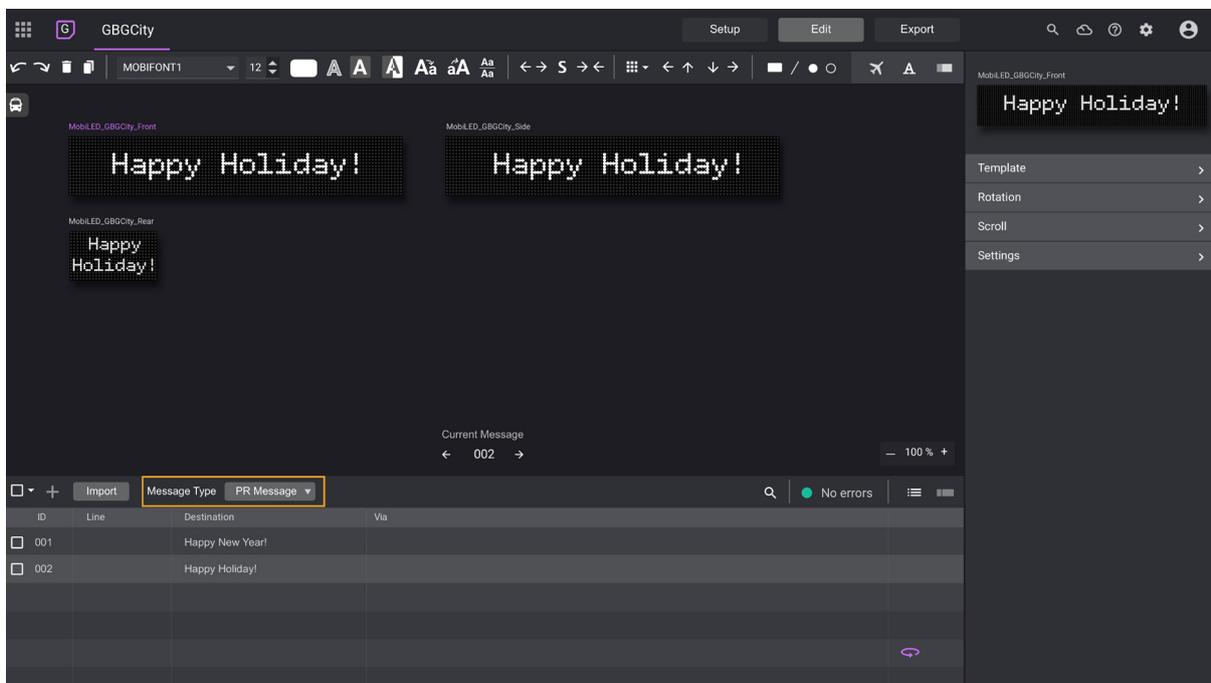


Figure 9.26 US version of Gypsum that shows message type selection in the edit interface.

Support various usage scenarios

As discussed in chapter 2, there is a large variety between markets in terms of sign layout, effect, colour, regulation and language. In order to satisfy the editing needs all over the world, Gypsum is designed to support various usage scenarios by offering users various edit tools and freedom of defining the layout and adding effect to whatever they want.

Taking add scrolling effect as an example, as

shown in figure 9.27, the software provides various settings such as scroll speed, scroll direction and scroll area adjustment, which could satisfy the needs of market that prefers lots of effect such as Brazilian market. Moreover, Gypsum is also designed with no limitations when it comes to the usage of font and effect, which means users could use as many fonts and effects as they. More usage scenarios such as add rotation to the entire message, customize font, customize graphics could be found in appendix 16.

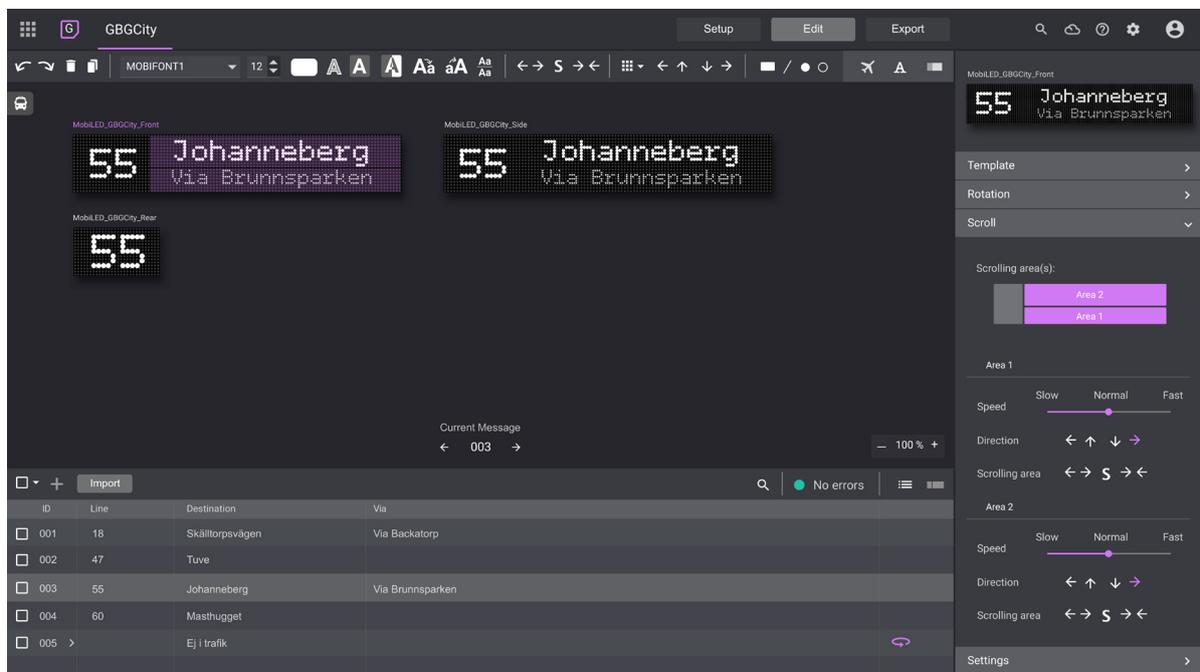


Figure 9.27 Adding scroll effect interface in the final design.

10

EVALUATION

In this chapter, Gypsum will be evaluated in order to determine whether the design fulfilled the initial requirements of the project and the requirements which were formed throughout the project such as interaction vision and brand persona. The project outcome was supposed to improve the user experience when creating content for destination signs for a global market. The final concept should be designed for a web based application presented in a clickable prototype with the look and feel according to Luminator Suite. The concept should also allow multi-usage and be compatible with iPad and iPhone.

10 Evaluation

Scope

According to the project requirements, Gypsum is a clickable prototype designed for the web and it is designed with support from visual guidelines of Luminator Suite. The interface creates an improved user experience by solving many of the pain points which were raised in the user study such as low automation level, long learning curve and many different software to achieve one output. Gypsum is also designed for a global market because it was collaboratively developed with input from users from all around the world. The scope also required the software to allow multi-usage, however, during the project it became clear that the software is not relevant for iPad or iPhone based on several reasons. Firstly, the work to be done requires a software of sovereign posture with a good overview, support rich input methods (mouse, keyboard and trackpad) and precise editing. All of these are more suitable for desktop applications. Secondly, handling legacy technology requires USB port, which is much more difficult to achieve with iPad and iPhone. Therefore, current technology work better with laptops but maybe in the future it would be relevant with an iPhone application to upload destination lists remotely.

User types and personas

The three user types and personas were used throughout the design process to guarantee that Gypsum satisfies the needs of key users: Customer Support Carl, Administrator Mary and Mechanics Robert. The end product was developed with these personas in mind and adopting for their differences in terms of usage and skill level, see chapter 9 Final design - Design for differences for more details of how Gypsum was designed for the key users. Gypsum is also designed according to the use cases of these user types.

Interaction vision and user feedback

The interaction vision was created after the user studies to guide the development of the application. The interaction should be simple, efficient, powerful and customizable. Based on the user input throughout the create phase the interaction vision is fulfilled, see for example user validation input in chapter 7 Create phase 2, where all qualities received grade 4 of 5 at least. To further validate the interaction vision: the software has a simple layout and clear main workflow; features like import, auto-formatting message listing and all-in-one software makes a efficient and powerful software; the editors and the toolbar creates customization possibilities. The prototype concept received positive feedback and many small suggestions of improvement. The users think it covers the functionality they require to do their work.

Brand persona

LTG is striving to become the preferred global provider of innovative and qualitative technology for passenger information systems which are easy to use. To give some examples of their branding material: their core value is “Global reach - Local touch” and they value teamwork and customer focus in everything they do. Gypsum is designed to fit these values as well as the brand persona Mark and the brand map, see chapter 4 Understand phase to read more about this. The application is designed for a global market and the local touch is achieved by offering good support of individual customers with the help center section in the application. The project was executed in a transparent and involving manner which not only benefited the teamwork between different sites within the Group but also set an example of how to collaborate in a global company. When creating Mark and the brand map it became clear that it would be valuable for the Group to work more towards dominance and freedom. Gypsum pushes the market towards a more modern, user friendly and innovative position in the market, which definitely goes in line with how LTG should develop.

11

DISCUSSION

Most product development processes starts on a broad level, understanding the context of the project, its elements and dynamics. Moving forward in the development process, the focus narrows to solve a smaller problem within the context by developing a solution. This project was situated in the context of public transportation where passengers and technology collaborate everyday to make it possible for people to get where they want to go. A software called Gypsum was developed to create the content of the destination signs guiding the passengers. At this point in the report, it is time to widen the focus again and discuss what implications the project outcome has for the users, for the public transportation and for LTG. Additionally, this chapter also discusses how the process and methods affected the final outcome and recommendations for future work.

11.1 Results

Gypsum and its users

Before Gypsum, there was no united way to edit destination signs. All sign brands within LTG had their own software to control the content and many users had to use several systems and softwares. Gypsum provides a way to edit all sign brands within the same software, reducing repetitive work and frustrations with previous software. Before Gypsum, the digital tools within LTG were developed back in the days where software often was developed by programmers who focused on powerful functionality and ease of code rather than great user experiences. Designing a software with a user centered approach has made a big difference on their tools. Functions like importation of old database, high automation level and collaboration features are all examples of the software solving current interaction problems and even consider the transition from previous software into the new one. All in all, by designing for the users and their use cases it has improved the user experience and created a modern software with a look and functionality they can be proud to show their competition. The software is designed for users from different markets needs and behavior and allowing them to practise creativity and produce high quality signs.

Gypsum and LTG

The LTG Suite is new and will require much time and effort to complete. There are many different systems within the Group which need to

be unified in order for the Luminator Suite to work, but done successfully, the Suite can have a huge difference. Destination signs can be considered to be the core business of LTG, therefore the new editor is a very important module in the Suite. The project to create Gypsum improves the foundation of the suite and, maybe more importantly, understanding of their customers. For LTG, a great destination editor would secure and improve their position on the market against competition. It would probably be beneficial with one software for the job so that all sites can collaborate more easily and reduce development costs.

Gypsum and public transportation

Designing for public transportation turned out to be complicated with many elements, stakeholders and complicated technology to consider. The act of programming the destination signs happens in the background, and maybe doesn't affect the passengers directly, but it does influence the quality of the passenger information system they are using to travel. It contains error recovery systems like error report, to provide an efficient way of checking editing work for mistakes. From a societal perspective, the software makes it possible to improve the quality of public transportation by improving the information displayed on the public vehicles. There is also a possibility of this leading to an increased number of passengers who choose to use public transportation, and thereby reducing the environmental pollution caused by cars.

11.2 Process and Methods

This project was heavily focused on user experience and it was half research and half development based. In this case the planning process had great influence on the project outcome. Below are some of the process decisions which made the biggest difference in this project:

- *Extensive user study*
- *Kept in the loop*
- *Continuous user input*

Extensive user study

One of the biggest challenges of designing is to understand the context which you are designing for. It is close to impossible to design a tool in a context with users you know neither about. It was therefore important to spend a large part of the beginning of the project to understand, and it paid off. Not only did it bring the knowledge to understand what needed to be designed but also it sped up the create process.

It also made a huge difference for the project to meet the users in their work environment in the abroad user studies. Before the travel many assumptions were made on who the user are. For example, first when meeting the smaller transit authorities and the Mechanics user types, it was clear why Customer supports have expressed that some of their users have low computer skills. Meeting them in person made it possible later in the create phase to have a much better understanding of what solutions they needed.

The project also had requirements to adapt to an global market, the travel made it possible gain knowledge not only about Swedish usage.

Many remote sessions were made with abroad users but these would not have been enough to understand their user situation. The travel also created a crucial contact network to utilize later in the project. The remote sessions were necessary but they influenced the quality of the data. Technical issues and the fact that you can not see users' facial expressions and body language was limiting, especially during usability tests. However, it was surprising how much information you can get by using screen sharing and think aloud techniques.

Looking back at the project, the focus was mostly on abroad users and few external swedish user. Question is if it would have brought better understanding to involve more Swedish external users? From a big perspective, different methods were used to collect data from different perspectives and it was more important to understand the big picture rather than collect exact data. More resources and time would of course made it possible to be more thorough, but in this project it was not relevant.

Continuous user input

This project was executed iteratively by choice. The Understand phase was the first one and then Create phase was divided into several smaller iterations. The setup made it possible in the Create phase to quickly illustrate and demonstrate ideas in discussion sessions and usability tests to validate the ideas and move on. It sped up the process of designing value with confidence. The prototype worked as a mediating object and made it possible to give valuable and straight to the point feedback, but was also limiting. In usability tests it was confusing not being able to click everywhere for example. A working prototype would of course be a better alternative but it would have required time and resources that were not available in this project. The iterations also helped to test the prototype with the entire spectra of users, from extreme experts who wants to do very special use cases and users with low software skills who are very uncomfortable using the software.

The iterations are a quick approach using available data sources rather than being very thorough with participants and data collection. It meant a faster approach and captured the main opinions but how would it turn out with a more thorough user study? More time for research but less time for concept development. This would have been difficult to motivate for the stakeholders because they were of course more interested in something they can use (the prototype) rather than just a understanding they might not be able to handle. It was better to absorb than focus on exact data and let our ideas can be validated later in the feedback sessions.

Kept in the loop and Uniting LTG

LTG as a group has quite recently been formed and when working with LTG it became clear that the different companies within the Group are still working on how to collaborate. There were many efforts which were not aligned and many sites turned out to work on overlapping projects,

not only destination sign editing. No “we”-feeling was detected and it is probably difficult to let go of the competitive mindset (they used to be competitors for not so long ago). For example, it felt like different sites had a hard time accepting a design that they know only came from one company. Knowing that this mindset exists, it became clear that in order for the design in this project to get accepted, it was important to be very transparent. Continuous feedback sessions with relevant people at the different sites and stakeholder meetings were held in order to gather useful input but also make people feel involved. Would this conscious decision of involvement make a difference for the acceptance of the project outcome - “The New Editor”? Hopefully.

The sessions made it possible to make sure the project focused on what output is important for them. But working with the company, it became clear that UX is not an established mindset or methodology and with many old digital tool in need of a update, they are in need of many more projects like this within the group. Therefore, in the beginning of the project, it was discussed how to scope the project in order to make the output valuable in the available time frame. A bigger scope would mean a less detailed prototype and a smaller scope would mean a more detailed prototype covering less use cases. Focusing on creating destination sign content turned out to be a good balance because the output covers the most important use cases and the prototype is still detailed enough so programmers know how it shall be programmed. However, it can be dangerous to involve stakeholders throughout the project as well. It happened several times that they wished to expand the scope to involve both stationary signs and dynamic sign editing but it was decided not to involve them eventually. Both modules would have required more research and studies in order to be incorporated in a valuable way. It was not even clear if they would have been modules within the editor or would suit better as applications of their own in the Suite.

11.3 Recommendations for future work

Currently, Gypsum is designed to cover most use cases gathered in the user study but there is still much more work to be done in the future. Many details have to be worked out how the application is going to work backend. How will the transfer of old databases work with the different brand systems? This consideration would be necessary in order to attract current users. How would a connection with the new business system ABAS work and what benefits could it potentially bring? How would Gypsum work with more modern technology with which outfiles can be sent remotely? Should route handling in MIE be incorporated in Gypsum or should it be an application of its own? In a close future should Gypsum be programmed into a beta version so that users can test it for a longer period of time. A prototype can give a good understanding of a product, but like discussed before it has its limitations. The experience can feel totally different when everything works.

After reflecting on the project, there have been many aspects of the project which have been demanding. It required time and effort to handle that many software and that many users. Sometimes it felt like too much focus was put on trying to please the needs of internal expert users. Taking Gypsum to the next step would require a better balance between internal and external users with different skill level. The prototype has the basic functionality and now it is time to fine

tune the details so that it is accessible to users of all skill level. The LTG has to be designed in collaboration with the users from different markets. Historically some of the software within the Group have had problems with adding too much functionality in the wrong way. Taking icenter for example, many functionalities has been added to the software without knowing where and how to incorporate them, with the result of a complicated and unintuitive software. When developing Gypsum further it is important to keep this in mind: question new functionality and leave space for future technology that the market haven't adopted yet.

12

CONCLUSION

In the final chapter, the conclusions drawn from the project are presented here, together with a summary of the final design and key reflection.

12 Conclusion

The purpose of this master thesis project has been to understand the interactions between users and current destination sign editing software in Luminator Technology Group and thereby develop a more user friendly version of the software for the global market. The conducted user studies show that three user types are mainly involved in the usage of the software, internal customer support, external administrator users and external mechanism users. The destination signs are also found to be designed differently in terms of layout, colour, language and effect among different countries. Therefore, addressing the needs different user types and different markets was the main focus in the design process.

As a result of the project, a destination sign editor named Gypsum was designed. The software makes it possible to edit all sign brands within Luminator Technology Group in the same software. It is also designed to support database importation to maintain existing old software users. It demonstrates a balance between efficiency, customization, simplicity and powerful functionality to serve the needs of a global market that have users with various skill level. Through usability tests and user validation with different user types, Gypsum was identified to be easy to understand, intuitive to use and contains the main functions they need to achieve their tasks. However, due to the limitation of

non-programmed prototype, a beta version of the software is expected to be developed in the near future to test with real databases and to gather more user input.

In conclusion, by conducting this project, Luminator Technology Group got a chance to know better about their users and thereby could offer them more user centered solutions for them in the future. For public transportation system, it is valuable to have Gypsum as future destination sign editing software to provide programmers and passengers better user experience.

REFERENCES

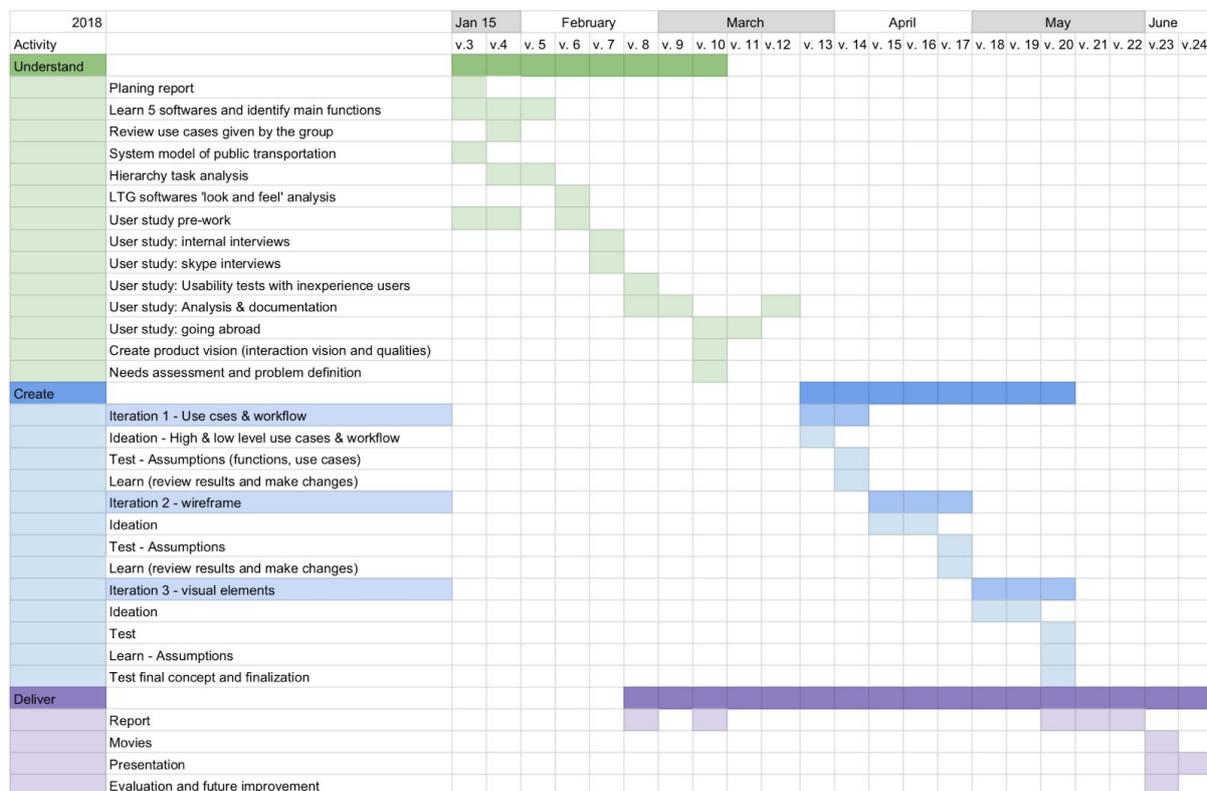
- Annett, N. (2006). Hierarchical task analysis. *Handbook of Cognitive Task Design*, pp. 17-35.
- Blackwell, A. H.; Manar, E., eds. (2015). "Prototype". *UXL Encyclopedia of Science* (3rd ed.).
- Cooper Alan (2014). *About face: The essentials of interaction design* (4th ed.). John Wiley & Sons Inc. Publications.
- Draper, Janet (2004). The relationship between research question and research design. In: Crookes, Patrick A. and Davies, Sue eds. *Research into Practice: Essential Skills for Reading and Applying Research in Nursing and Health Care*, Volume 2nd Ed. Edinburgh: Bailliere Tindall, pp. 69–84.
- Gall, M. D., Gall, J.P., & Borg, W.R. (2003). *Educational research: An introduction* (7th ed.). Boston, MA: A&B Publications.
- Leidenkrantz, T, 2017. *Building a strong brand platform*. Lecture 170918, Chalmers University of Technology.
- Luminator Technology Group. 2018. *Capabilities*. <https://www.ltgglobal.com/capabilities/> (Retrieved 2018-02-20)
- Mobitec. 2018. *About us*. LTG Sweden AB. <http://www.mobitec.eu/about-mobitec/> (Retrieved 2018-02-01)
- Raymond Scupin (1997). *The KJ Method: A Technique for Analyzing Data Derived from Japanese Ethnology*. *Human Organization*: Summer 1997, Vol. 56, No. 2, pp. 233-237.
- Usabilitygeek, 2016. *Smart-UX: Use Cases*. <https://usabilitygeek.com/smart-ux-use-cases/> (Retrieved 2018-05-28)
- Usabilitygeek, 2016. *Usability Testing With Prototypes*. <https://usabilitygeek.com/usability-testing-prototypes/> (Retrieved 2018-05-28)
- William Lidwell; Kritina Holden; Jill Butler (1 January 2010), *Universal Principles of Design*, Rockport Publishers, p. 182, ISBN 978-1-61058-065-6.

APPENDICES

- Appendix 1: Gantt chart
- Appendix 2: Interview script for software users
- Appendix 3: Pre-interview questionnaire for remote software users
- Appendix 4: Usability test script
- Appendix 5: Detailed evaluation of each software
- Appendix 6: Design assumptions in create phase 1
- Appendix 7: Design assumptions in create phase 2
- Appendix 8: Design assumptions in create phase 3
- Appendix 9: Potential improvements gathered from user studies
- Appendix 10: Potential improvements for prototype in create phase 2
- Appendix 11: Potential improvements for final prototype
- Appendix 12: Functional and data elements and their grouping
- Appendix 13: Low level use cases
- Appendix 14: Overview of the final design
- Appendix 15: Usage scenarios of the prototype in create phase 2
- Appendix 16: Usage scenarios of the final prototype

Gantt chart

Appendix 1 - Gantt Chart of this project



Interview and usability test scripts

Appendix 2 - Interview script for software users

This script was used with small changes to adapt to different software, 'the software' in this script could refer to MIE/TedPlus/ALPHA-NT/IPS/Elyse, depending on which software user is being interviewed.

Ok to record sound and take some photos?

Talk about aim of session

- Master thesis - develop interface for destination editing software, combine and improve
- Interested in user input

- The output is not set in stone, only project....

Introduction (SHORT)

- Tell us SHORTLY about yourself (Age, gender, hobbies, working hours, workplace)
- How long have you worked here? How many year of experience with the software?
- What's your position in the company? Main responsibility?

Sketch system model

- Sketching
 - How does the software work?
 - How does the software work with the rest of Public Transportation (customers, vehicles, drivers)? What happens with the outfile?
- Brief discussion

Questions

- General/Goals
 - What do you think about the software?
 - What are you trying to accomplish in the software?/Why do you use the software?
 - What makes a good day? A bad day?
 - What do you enjoy most about your job? What do you always tackle first?
- Teamwork - What people are you depending on to do your work? How?
- Domain knowledge - What do you need to know to do your work?
- First impressions
 - How was it to learn the software in the beginning?
 - Compared to now?
 - If MIE is a person, what words would you use to describe this person? e.g. Successful, charming, dull, intelligent
- Function
 - What are the most common things you do with the software?
 - What main basic tasks do you do?
 - in case of no info: Tell them a scenario: Set up a new customers
 - Show us one of your recent projects [DEMO, take pics]
 - What happens after your work with the software?
 - How do you check if your work is done in a correct way?
- Frequency
 - What parts of the software do you use the most?
 - How often do you interact with the software?
 - What parts aren't you using? Why?
- Preference/failure
 - What's your favourite aspects of the software?
 - What drives you crazy?
 - What problems do you encounter? Give example
 - How do you work around problems?
 - How could the software be improved?

Thank you!

Ok if we contact you again during the development phase?

Appendix 3 - Pre-interview Questionnaire for remote software users

This pre-interview questionnaire was used to gather the basic information and software usage situation about remote users.

This form is for us to collect basic data before our interview session and for you to start thinking about the destination editing software you are using before the session.

The form is divided in two parts: a question part and one photo part!

General

- What's your name?
- What's your gender?
- Where do you live at the moment?
- Which company are you working for and what are you working with?

Software

- Ok, so what destination software are you using?
- What do think of the software? (Rate from 1 to 6)
- Why?
- What do you use the software for?
- What were the latest things you did in the software?
- What is most important for you when you work in the software?

Evaluation

- What are your favorite aspects of the software?
- What aspects of the software drive you crazy?
- What would you add in future the software?

Photo study

- We are also very interested how public transportation looks like where you live. Could you take some pictures of for example destination signs, control units, buses and trams, which you feel are relevant for the software?

Anything else you would like to add?

Amazing, thanks :)

See you on the interview sessions!

Appendix 4 - Usability test script

Script template

[Task name]

[Scenario]

[Task]

<i>Topic</i>	Number	Explanation
<i>Help</i>		
<i>Errors made when performing the task</i>		
<i>Confusion / Unexpected user actions</i>		

Task completion?	Yes / No
------------------	----------

How was it to perform this task? Why?	
How intuitive was it to perform the task? Why?	

Customer support

You are working as a customer support at Mobitec. LTG has just launched a new editing software which is replacing XX. You have barely got familiar with the software yourself and the support requests are already piling up in your email inbox with customer who wants help with the new software.

TASK 1: Setup - 2 concepts (*Setup wizard and normal setup*)

You receive a call from a customer who wants your help with the new software.

- Set up a new project for GBGCity with Mobitec signs.

<i>Topic</i>	<i>Number</i>	<i>Explanation</i>
<i>Help</i>		
<i>Errors made when performing the task</i>		
<i>Confusion / Unexpected user actions</i>		

<i>Task completion?</i>	<i>Yes / No</i>
-------------------------	-----------------

<i>How was it to perform this task? Why?</i>	
<i>How intuitive was it to perform the task? Why?</i>	

TASK 2: Import

The customer emailed you a destination list which he wants you to import into the project.

- Import the excel sheet from the customer into the project.

<i>Topic</i>	Number	Explanation
<i>Help</i>		
<i>Errors made when performing the task</i>		
<i>Confusion / Unexpected user actions</i>		

Task completion?	Yes / No
------------------	----------

How was it to perform this task? Why?	
How intuitive was it to perform the task? Why?	

TASK 3a: Enter message manually

You realise that the excel sheet is missing one message that you and the customer was speaking about over the phone. You decide to enter it manually.

- Add the message "1 Chalmers via Central Station".

<i>Topic</i>	Number	Explanation
<i>Help</i>		
<i>Errors made when performing the task</i>		
<i>Confusion / Unexpected user actions</i>		

Task completion?	Yes / No
------------------	----------

How was it to perform this task? Why?	
How intuitive was it to perform the task? Why?	

TASK 4: Enter message in Mattias mockup

You realise that the excel sheet is missing one message that you and the customer was speaking about over the phone. You decide to enter it manually.

- Add the message "1 Chalmers via Central Station".

How was it to perform this task? Why?	
How intuitive was it to perform the task? Why?	

TASK 5: Add rotations

The customer also wants the front sign to also show the biggest bus stop on the line.

- Show both “via Central Station” and “via Bus Terminal”.

<i>Topic</i>	Number	Explanation
<i>Help</i>		
<i>Errors made when performing the task</i>		
<i>Confusion / Unexpected user actions</i>		

Task completion?	Yes / No
------------------	----------

How was it to perform this task? Why?	
How intuitive was it to perform the task? Why?	

TASK 6: Add Scrolling

On the side sign the customer wants the text to be moving.

- Make the text in the side sign move from left to right.

<i>Topic</i>	Number	Explanation
<i>Help</i>		
<i>Errors made when performing the task</i>		
<i>Confusion / Unexpected user actions</i>		

Task completion?	Yes / No
------------------	----------

How was it to perform this task? Why?	
How intuitive was it to perform the task? Why?	

TASK 7: Capital letters

The customer wants all the letters to be capital.

- Change all small letters to big capital letters.

<i>Topic</i>	<i>Number</i>	<i>Explanation</i>
<i>Help</i>		
<i>Errors made when performing the task</i>		
<i>Confusion / Unexpected user actions</i>		

<i>Task completion?</i>	<i>Yes / No</i>
-------------------------	-----------------

<i>How was it to perform this task? Why?</i>	
<i>How intuitive was it to perform the task? Why?</i>	

TASK 8: Graphic plane

The customer also wants a graphic to visualise that the line is going to the airport.

- Insert a plane graphic on the place of number "1" in front sign.

<i>Topic</i>	Number	Explanation
<i>Help</i>		
<i>Errors made when performing the task</i>		
<i>Confusion / Unexpected user actions</i>		

Task completion?	Yes / No
------------------	----------

How was it to perform this task? Why?	
How intuitive was it to perform the task? Why?	

TASK 9: Outfile

Now when all editing is done it's time to send your work to the customer.

- Create an outfile for all bus types.

<i>Topic</i>	Number	Explanation
<i>Help</i>		
<i>Errors made when performing the task</i>		
<i>Confusion / Unexpected user actions</i>		

Task completion?	Yes / No
------------------	----------

How was it to perform this task? Why?	
How intuitive was it to perform the task? Why?	

Post test questions:

- How was the overall terminology?
- Did you understand what actions were available?
- How was the division between setup, edit, export? Did this main flow make sense?
- How was the organization between toolbar, workspace, message listing, properties, bus fleet?
- Did you feel like you found them where you expected them to be?
- You think the placement of rotations and scrolling and template in properties made sense?
- What did you think about the icons?
- Do you think the main flow make sense?
- On a scale from (not at all) 1 to 5 (very), how simple would you experience the software?
- On a scale from (not at all) 1 to 5 (very), how powerful would you experience the software?
- On a scale from (not at all) 1 to 5 (very), how efficient would you experience the software?
- On a scale from (not at all) 1 to 5 (very), how customizable would you experience the software?

External user

You are working at XXX and you are responsible for creating destination messages. LTG has just launched a new editing software which is replacing XX. In this software you can edit all the signs from all brands within LTG. You are very new to the software.

TASK 1: Setup

Setup a new project with your bus fleet.

- Set up a new project with one bus and three signs

<i>Topic</i>	<i>Number</i>	<i>Explanation</i>
<i>Help</i>		
<i>Errors made when performing the task</i>		
<i>Confusion / Unexpected user actions</i>		

Task completion?	Yes / No
------------------	----------

How was it to perform this task? Why?	
How intuitive was it to perform the task? Why?	

TASK 2: Import

You want to use the same destination list as in your old software. You have an excel sheet contains all destinations in your computer.

- Import your excel sheet with destinations.

<i>Topic</i>	Number	Explanation
<i>Help</i>		
<i>Errors made when performing the task</i>		
<i>Confusion / Unexpected user actions</i>		

Task completion?	Yes / No
------------------	----------

How was it to perform this task? Why?	
How intuitive was it to perform the task? Why?	

TASK 3a: Enter message manually

You realise that the excel sheet is missing one message. Now you need to add this message manually.

- Add the message "1 Chalmers via Central Station".

<i>Topic</i>	Number	Explanation
<i>Help</i>		
<i>Errors made when performing the task</i>		
<i>Confusion / Unexpected user actions</i>		

Task completion?	Yes / No
------------------	----------

How was it to perform this task? Why?	
How intuitive was it to perform the task? Why?	

TASK 4: Add rotations

You want the front sign to also show the biggest bus stop on the line.

- Show both “via Central Station” and “via Bus terminal”.

<i>Topic</i>	Number	Explanation
<i>Help</i>		
<i>Errors made when performing the task</i>		
<i>Confusion / Unexpected user actions</i>		

Task completion?	Yes / No
------------------	----------

How was it to perform this task? Why?	
How intuitive was it to perform the task? Why?	

TASK 6: Add Scrolling

On the side sign you want the text to be moving.

- Make the text in the side sign move from left to right.

<i>Topic</i>	<i>Number</i>	<i>Explanation</i>
<i>Help</i>		
<i>Errors made when performing the task</i>		
<i>Confusion / Unexpected user actions</i>		

<i>Task completion?</i>	<i>Yes / No</i>
-------------------------	-----------------

<i>How was it to perform this task? Why?</i>	
<i>How intuitive was it to perform the task? Why?</i>	

TASK 7: Capital letters

You realise that the message would be much clearer if the letters would be in capital.

- Change all small letters to big capital letters.

<i>Topic</i>	Number	Explanation
<i>Help</i>		
<i>Errors made when performing the task</i>		
<i>Confusion / Unexpected user actions</i>		

Task completion?	Yes / No
------------------	----------

How was it to perform this task? Why?	
How intuitive was it to perform the task? Why?	

TASK 8: Graphic plane

You want a graphic to visualise that the line is going to the airport.

- Insert a plane graphic.

<i>Topic</i>	<i>Number</i>	<i>Explanation</i>
<i>Help</i>		
<i>Errors made when performing the task</i>		
<i>Confusion / Unexpected user actions</i>		

Task completion?	Yes / No
------------------	----------

How was it to perform this task? Why?	
How intuitive was it to perform the task? Why?	

TASK 9: Add Scrolling

Now when all editing is done it's time wrap up your work, so you can plug in the file into the buses..

- Create an outfile for all bus types.

<i>Topic</i>	Number	Explanation
<i>Help</i>		
<i>Errors made when performing the task</i>		
<i>Confusion / Unexpected user actions</i>		

Task completion?	Yes / No
------------------	----------

How was it to perform this task? Why?	
How intuitive was it to perform the task? Why?	

Post test questions:

- How was the overall terminology?
- Did you understand what actions were available?
- How was the division between setup, edit, export? Did this main flow make sense?
- How was the organization between toolbar, workspace, message listing, properties, bus fleet?
- Did you feel like you found them where you expected them to be?
- You think the placement of rotations and scrolling and template in properties made sense?
- What did you think about the icons?
- Do you think the main flow make sense?
- On a scale from (not at all) 1 to 5 (very), how simple would you experience the software?
- On a scale from (not at all) 1 to 5 (very), how powerful would you experience the software?
- On a scale from (not at all) 1 to 5 (very), how efficient would you experience the software?
- On a scale from (not at all) 1 to 5 (very), how customizable would you experience the software?

Software evaluation

Appendix 5 - Detailed evaluation of each software

- MIE
 - Good points
 - General
 - Customers are generally happy with what you can do with MIE
 - MIE basic - Easy to use, simple, straightforward, powerful
 - Get things done quick - the ability to transfer something from a small sign to a big sign, the ability to import from excel file quick set up template and font, auto formatting, export
 - “It’s a really great software I have to say”
 - A lot of time saving functions, automation level is quite high - don’t need to do things manually
 - “Intelligent person because such powerful with much knowledge, a range of vary skills because MIE can do so much, quite efficient, intelligent” - Matt
 - Stable
 - Usability

- Learnability
 - quick and easy to learn
- Icons and graphics
 - Straightforward, easy to understand and refer to
 - Clear, they help you get your things done
 - Good visual control
- Layout
 - Simple
 - Logical, makes sense
 - all the buttons are grouped where I want them
- Functionality
 - Auto - Formatting
 - Import - simple and good
 - Copy - copy a new program, copy to bus
 - Preview - you can have a look at it before it goes to the buses
- Pain points
 - General
 - MIE has many good features, but it doesn't help users to know the existence of them and how to use them
 - Technical issues
 - Bugs
 - Sometimes it doesn't work as it should be
 - The software shows possibility of doing things, but doesn't work in reality
 - Not 100% stable
 - Response speed
 - Import or export large database, takes long time, and crash sometimes
 - Transfer database
 - Transfer database from old computer to new computer can crash
 - Usability
 - Guidance
 - Only one very old user manual, plain text, not step by step guide, e.g. Indian users create a step by step manual by themselves
 - It doesn't give you any clues how to do stuff
 - e.g. right click on the stop, now the choice are gray, showing possibility but doesn't show where to active the function, maybe add more choices when user right click on it
 - You have to enter things in a certain way, otherwise it won't work
 - When the software is updated, no info or tip tell you what has been updated and how to use new features

- Knowledge
 - Lack of basic knowledge for sign and bus systems
 - Software terminology, takes time to understand
 - Some customers lack of common computer skill
- Layout
 - Hidden features, e.g. import, autoformatting
 - Have seen the functions but can't find them afterwards
 - There are some functions which aren't useful and only confuse customers
- Efficiency
 - Some functions require many steps
- Error report
 - Error report doesn't help you INSERT SCREENSHOT
 - You get warnings and errors when generate outfile and don't show any solutions, wrong time
- Functionality
 - Handling of bus types
 - Many customers have many different types of bus, double deckers for e.g., and many sizes of signs, put all buses together can be very time consuming. When they need to add a new service, they need to do individually on each bus, can be very time consuming.
 - Fonts and graphics
 - Too few fonts compared to competitors
 - Lack of customization
 - Crappy fonts, they are all over the place
 - No font editor
 - no graphical editor
 - Template
 - Can't use free template
 - Default templates are limiting in terms of not good for title case lettering e.g. gjy, too thin line numbers and not emphasis on the top line
 - Simulation
 - no showing the correct proportion
 - No real effect
 - Limitation
 - not flexible enough when it comes to different user habits, different users use MIE in different ways. e.g. Scrolling, rotation effect in Brazil and US market
 - Rotations are limited to 20, in Singapore the buses have very long journey. They use front sign to show stoplist
 - There's a limitation of 15 signs in a bus
 - Can't have several same name and addresses e.g. some customers could have over 15 signs
- MIE advance

- Requires pre-knowledge
 - Quiet bloated
 - You need to make lots of adjustments, settings conditions
Eastriggers door opening, you have to have a lot of data before you can do anything, e.g. GPS coordinates, equipment in the bus
 - Default setting doesn't work well sometimes
 - Reusability - you have to work with it continuously otherwise you forget.
 - Importing GTFS files on advanced version takes much time
 - map destination codes to route programs takes much time
 - Exporting outfiles with route and voice data takes much time
 - need of a major reconfiguration as to how databases are structured as it can run really slow
 - Advanced mode is complicated due to the setting of GPRS
 - have to be very concentrated
 - Hard to import data from different sources together
- TEDPlus
 - General
 -
 - Good points
 - Easy to handle, She understands it very fast
 - It's easy, the user interface is well organised
 - They like the editing bar under input bar, change fonts, good overview, well organised
 - Right click on the font to open font editor
 - Mostly happy with the software, They have a good base to work with it
 - Pain points
 - Handling configuration files in several software
 - Configuration files (limits customer do to setup, puts workload on customer support)
 - Many configurations software, more easy to make mistakes and hard to detect mistake
 - Configuration files from LAWO's perspective: adds much workload for them, hard to write invoice, they can't do everything for customers
 - They have a new bus and need new configuration
 - Can't transfer database using old configuration file into a new configuration file
 - Could be easier to import signs, when they get a new bus, it's a new configuration, should be easier to transfer data from old configuration to new configuration

- In MIE, it's easier to convert, she creates a new sign, then MIE asks do you want to convert from existing signs, then it works.
 - Config files limits ability to setup - eg Address changing - now it's in configuration file, should be done in the main interface, for people don't know how to edit configuration file can take long time of work
 - Many softwares to create the result
- Card reader and card handling
 - Older devices requires SRAM cards - the reader and the card is expensive
 - The handling of SRAM card, post to customer
 - takes time, sometimes delivery is slow
 - mie customers only use usb sticks
 - sometimes they need to sent twice the card
 - Needs a card - customers need to spend extra money on purchasing card and card reader
 - Sometimes the cards doesn't works on some control units, sometime don't have battery
- Low level of automation
 - Destination editing of many destinations (no import, have to do all one by one)
 - One destination takes 3 mins - (takes long time)
 - Should contain auto-complete, you type in the first letters then there is a drop down list showing a city list in that country
 - Creating customer fonts take a lot of time, they often do it for customers
- Functionality
 - You could only scrolling one block on the sign, he wants to scroll two blocks, bus line and destination text line on some special signs, e.g. bus line express35
 - Missing functions - copy paste very basic, auto align function(always align to the left), template for destination layout
 - Undo and redo won't work
 - System messages are embedded in message listing
- Usability
 - Reusability - Customers edit signs seldomly, they forget totally and call him
 - There are some hidden functions in TEDPlus, e.g. shortcut, right click to open font editor
 - Lots of movement of mouse and much type work
- Knowledge
 - Lack of understanding of the whole system - Some customers have good knowledge of the whole system, know what documents they should send to customer support, some don't
- Acceptance of new software with the customers

- In TEDPlus, if two persons work with the same database, can be a bit risky

- Alpha-NT

- General

- Difficult in the beginning, but once learned it's good, you understand the logic behind
- font limitation
- Level of automation
- Copy function dangerous
- Difficult and unintuitive setup

- Good point

- Both likes it
- In the beginning its a bit complicated but then you get used to it
- After you have the basic, e.g. resolutions, setup, then it's very easy, once it works then it works
- They can simulate drive with a bus to see if the sign changes or not
- Can program almost everything
- Once you know it then you can handle it
- Easy to handle - user interface is logic, symbols and names is clear, the points are from top to bottom and follows his workflow
- Functionality is ok
- He thinks it is effective enough for him

- Pain points

- Learning software
 - In the beginning its a bit complicated but then you get used to it, In the beginning, it's too complicated, normally you download a new software, you have tutorial, pictures to guide you. Now you have nothing, you have a manual, but it's complicated
 - Problem to find where to find more fonts
- Technical issues
 - Not stable enough - When it works, it's okay, but if it's broken, they need to rebuild the database. Gorba programmed special feature for RVBW, they changed the database setting for them.
 - Not running with some units
- Font limitation
 - They have the problem with only ten fonts - he problem if is you have different resolutions you need different fonts, Max 10 fonts - bad
- Low level of automation
 - The software itself is not complicated, but filling it in takes time, the software is stupid, you have to tell it hey do this... it doesn't do anything automatically

- You have a lot to do, nothing works automatically
- Functionality
 - When you transfer the data, you can't specify specific folder, disk only today
 - No modify date for each database, could have dates on the top bar
 - Rotations are not clear enough, can't get an overview of all the layers
 - Copy function is dangerous, the default selection is the most dangerous one - Possible to copy over all signs – creates problems for the customers
 - Routes handling - very complex now, not many customers are using it
 - Unable to simulate ibis protocol
 - Can't import exported file(ltd file) to the software, but could import database, can't export whole database every time, the file will be too large.
 - Stored in another folder than Alpha-NT (because old software)
- Setup
 - Setup for signs and dimensions are a bit too complex, they need a guide on how to set up
 - Another bus company only used Z before, now they get a new board computer, they need to change to L&Z, then they need to do everything again. If you created basics wrong, not thinking about future, you have to take your time to do it again, create a new database, copy and paste, you can't copy Z to L
 - Basic settings can sometimes be hard to change - forcing you to do everything again. eg LZ
- Automatic mode

Many big bus companies are using onboard computer to handling all info automatically. In this way, they don't need to edit signs manually in AlphaNT but only load excel file into control unit, They only use AlphaNT to configure vehicles.

 - Main problem with having two systems (manual in AlphaNT and automatic with excel)
 - They have 10 bus types in Alpha-NT - have to configure differently for all bustype
- Pictures /graphics, support drag and drop to adjust graphic positions - IMPROVEMENT
- ICenter
 - General
 - Good points
 - Looks nice, modern/flat style,

- Not windows opening all the time
- In beginning/basic with the specialist, the software is intuitive
- Flat style
- “Ridiculously powerful”
- Pain points
 - Unintuitive and complex
 - He has to fight through the program
 - Alex: iCenter is still a bit complicated even though they want to make it simple. There is no guide for the software. LTG ignores customers who use the software for the first time. Not like iPhone. When you first touch this software, you know how to use it
 - In general it’s nice but at the end it’s annoying, this is not the value LTG wants to promote
 - He thinks iCenter is not intuitive, not suitable to deliver to customers
 - iCenter works quite well in the beginning, then it gets overloaded, this bloating is everywhere.
- Elyse
 - General
 - IPs and Elyse will not be updated anymore, can’t support beyond
 - Good overview of all signs at the same time
 - “ it’s not an easy system allow you to make quick changes”
 - Functions users could achieve in standard mode is limited but valuable (automatic), forcing them to switch to freestyle mode, can’t go back
 - Very unstable, crashes often and have to save every 5 min
 - Biggest complain ‘font limitation’
 - Good points
 - Layout/structure
 - Sign overview gives you overview (able to see ALL signs) - especially good when dealing with different bus types (see all but can also see which one sign goes to which)
 - Showing all signs at the same time - It’s good to be able to switch between different bus types without having to produce another listing
 - Has both standard and freestyle mode. Elyse is more efficient because of the standard mode
 - Setup/preferences
 - Pretty easy to setup the signs
 - Easy to set them up and they use the same database
 - Pretty easy to update because of easy access to preferences
 - Functionality
 - Dash sign support

- System messages - Pre-programmed
- Colour sign support
- Print - Elyse produce sign preview signs for printing
- Pain points
 - Technical issues
 - Freestyle works fine, the problem/limitation is she can only put dots on the bottom, no direct pixel editing, it's a nice feature to have (BUG)
 - "Hard to find 1GB memory stick for Elyse" due to hardware limitation
 - Crash sometime - have to save all the time
 - Instability of software - have to save all the time
 - Low level of automation
 - The automation is there but only go to certain points, not fully
 - Output files different for each configuration - no master output file for all
 - Build image for configurations impacts output file (it's not one main file for all sign models - separate setup for different gen of signs)
 - Standard vs. Freestyle mode
 - Start in standard, then have to switch over and then can't go back for that message
 - Standard mode is not automatic enough and doesn't give users much freedom while freestyle requires much manual work, for many tasks, e.g. graphic, users need to freestyle
 - Limits for editing fonts in standard mode forcing to freestyle mode
 - Font limitation
 - Biggest complain 'font amount limitation'
 - No font editor
 - Several systems
 - They have to go outside the software to program for the bus, it is expected to configure within the software
 - Export
 - Everytime when we want to export it, there's some kind of issues, need to go back and make changes
- Only limitation is when they want to do adhoc? For service changing, it's time consuming, it's not an easy system allow you to make quick changes

- IPS

- General
 - IPS and Elyse will not be updated anymore, can't support beyond
 - Customizable, flexible, allows high creativity
 - Good working program

- The biggest problem is the age of the software, 20 year old with new signs, the concern is can the software support the signs in the future?
 - Much manual work, when you apply the text 99% the text doesn't fit the sign, user needs to check signs individually and manually
 - Good points
 - Layout
 - IPS has a list showing all messages, which allows you to see all signs, more clear
 - Drivers interface is very easy - input easy
 - Functionality
 - Lock/unlock feature. When she's done with a sign, she locks the sign. So once she sees the sign is locked, she knows it's done.
 - Search function in messaging list
 - Users can set more specific settings in IPS
 - Print
 - Message classes
 - A, B, C, E classes
 - Color sign support
 - Dash sign support
 - Graphic/font editor

Old dos software which is used for internal font and graphic editing allows pixel by pixel editing. Some super customers are also using it.

 - Font editor
 - Graphics editor
 - Good with pixel by pixel editing
 - Pain points
 - Usability
 - Not intuitive enough, they have to cross two 'obstacles' to get into message list
 - When it's totally new file, nothing is in the list and background
 - Get past log in window
 - Open correct project
 - IPS is based on dos, you need to type in special dos language for some functions, e.g. ^ in the input bar for rotations
 - Reusability
 - Have to relearn it every time
 - Build everything from scratch is difficult, not for the beginner
 - Starting a new project. In ips to build a new file, you have to build sign set up, then template, then editing. The foundation of the software is complicated.
 - Lacks ease of use for editing w/o training
 - Automation level

- Elyse is more efficient because of the standard mode, IPS is time consuming, especially work with two lines
- Functionality
 - It's a very old program, you can't insert a route in between two routes, can only add at the end of the list
- Technical issues
 - Can't move their software to a faster machine, has to use internal card reader

Overall

Pain

- Can't support different bus types in the same project - repetitive work
- Can't edit several sign sets from different companies in the same software - repetitive work
- All software relies on domain system knowledge - understand of sign, dimension, address
- Inefficient / manual work - IPS, TEDPlus, AlphaNT
- Relearnability is a problem
- Hidden features
- Each software was programed for their own signs, but many customers they are using several kinds of signs
- General
 - Difficult in the beginning, but once learned it's good, you understand the logic behind
 - All require understanding of the whole system
 - acceptance
 - easier work between databases, bus types....
 - Repetitive work, very manual depending on the software (elyse and MIE more automated)
 - Can't edit the signs from different companies in the same software - have to do repetitive work if they have several sign sets
 - Also repetitive work between bus types....
 - The software doesn't allow LED and flipdot in the same project, you have to create two projects, Back time it was time consuming since they have both led and flip dots signs

Design assumptions

Appendix 6 - Design assumptions in create phase 1

Use cases

- IF the product is designed to match the three main use cases, THEN the software will cover the most important functionality, BECAUSE the three use cases are the most common things they do.

Software structure

- IF the product has a vehicle oriented setup, THEN this reflects how they want to organize their work, BECAUSE this is how their fleet is organized in reality.
- IF the product has one message listing per project, THEN this is how the user wants to work, BECAUSE it saves time.
- IF the product can produce one universal outfile, THEN it would save time and reduce potential use errors, BECAUSE the user don't have to produce one outfile per bustype.

Functionality

- IF the product contains the main functionality described in **XXXX**, THEN the user would be satisfied, BECAUSE it provides the tools so that the user can accomplish what he/she want to be able to do in the software.

Main workflow and layout

- IF the product has a workflow which matches the three main use cases described in **XXXX**, THEN the interaction with the software would feel natural for the user, BECAUSE it follows their mental model of how to create signage on vehicles.

Appendix 7 - Design assumptions in create phase 2

- **Naming/terminology** - Section/buttons makes sense? Better words?
- **Organization** - Info grouped meaningfully? Workspace and positions of containers/elements makes sense?
- **First time use/discoverability** - Common items easy to find? Instructions clear? Too much Instructions?
- **Effectiveness** - Can they efficiently complete tasks? Missteps? Where? How often?
- **Workflow** - Does the workflow make sense?
- **Qualities** - Simple, effective, powerful, customizable

Naming/terminology

- IF the product is using the names: Message, fleet management, setup, edit, export, import, properties, template, scrolling, rotations, THEN the users will find and understand the available actions, BECAUSE they recognize the terms.

Organization

- IF the product is divided into Setup, Edit, Export, THEN the user thinks it makes sense and use the tabs when needed, BECAUSE it follow the main workflow.

- IF the Edit interface is structured into toolbar, workspace, message listing, properties, bus fleet, THEN the user will understand and find what they are looking for, BECAUSE they recognize the layout from previous experience with editing software.

First time use/Discoverability

- IF the tabs Setup, Edit, Export are located upper right, THEN the user will find where to go, BECAUSE the user will look for XXXXX.
- IF rotations and scrolling is located in the properties, THEN the location makes sense and the actions can be found, BECAUSE tools and effects differ from each other.
- IF the interface uses mostly icons for actions, THEN the user will understand how to use and find the available actions, BECAUSE the icons are clear enough and enough information.
- IF templates can be changed in properties, THEN the user will find the action, BECAUSE the location makes sense.

Effectiveness

- IF the product has the current design, THEN the user will be able to efficiently perform:

Tasks to be performed

- Setup project with vehicles and signs
- Import excel sheet with destinations
- Add message (Mattias mockup)
- Add rotation
- Add scrolling
- Make all text in capital letters
- Insert airplane graphic
- Export outfile

Demo and ask for feedback

- Font editor
- Graphic editor
- Template editor

Workflow

- IF current design, THEN the user will appreciate the main workflow, BECAUSE the flow follows their mental models.

Qualities

- IF current design, THEN the user will find the interface simple to handle, BECAUSE he/she can find and understand actions and able to perform main tasks.
- IF current design, THEN the user will find the functionality of the software powerful, BECAUSE she/he can get their work done.
- IF current design, THEN the user will find that he/she have great possibilities for customization when editing messages, BECAUSE the software provide pixel-by-pixel editing.

- IF current design, THEN the user will find the software efficient to work with, BECAUSE she/he can get their work done fast and without too much effort.

Appendix 8 - Design assumptions in create phase 3

New features - Does report system and a new way of adding rotations bring value and make sense?

- **Support system** - How can the product support users so they can get their jobs done?
- **Look and feel** - Is the design attractive?

Assumptions

New features

- IF the product has a error report system, THEN this will help the users produce the sign content they want and reduce sign errors, BECAUSE it makes the user aware of sign content problems and provide a way to find them.
- IF the product provide the possibility to add rotations on entire messages, THEN this would be of valuable, BECAUSE the users want to apply rotations on entire messages and not only on individual signs.
- IF the product provide the possibility to add rotations on entire messages, THEN user will understand how to do it, BECAUSE the user can discover and understand to click on the rotation icon to activate rotation.

Support system

- IF the product provides first time guide, THEN the learning curve will shorten, BECAUSE the new user will quickly understand the basic concept of the software and how to do basic tasks.
- IF the product has tooltips when hovering over buttons, THEN the user will be able to perform their tasks, BECAUSE the user will understand what the buttons do and where to click.
- IF the product provides video tutorials on common tasks, THEN this will help the user to handle the software, BECAUSE the tutorials provide step by step explanation of tasks.
- IF the product has collaboration possibilities to invite people to your project, THEN this will solve many problems user have with the software, BECAUSE users could invite customer supports to solve issues.

Look and feel

- IF the product has a grey scale colored, flat style user interface according to LTG Suite guidelines, THEN the design would enhance the user experience, BECAUSE the users will appreciate the look and feel professional when using it.

Software potential improvements

Appendix 9 - Potential improvements gathered from user studies

Technical

- Improve process speed, faster response time
- Create 64 bit version, better handling of memory
- Possible to transfer fonts in export file to signs
- Documentation of the software development should start from the beginning
- Vector based TTF fonts

Support

- Help beginners understand the whole system
- A help button/guide for beginners to help them to understand and also find where the function is
- Good manual with step by step pictures
- Video tutorials
 - e.g. short video explain address, program, and how they work
- First time typeform guide
- When the software is updated with new features, it should show guide tell users what functions have been added and how to use them
- Error prevention and recovery
 - Hide dangerous actions
 - Error report
 - Provide solutions
 - Undo
 - Access to older versions
 - Autosave
- Possible to give feedback and improvement ideas to developers since users might have very good ideas
- FAQ database, links to tutorial videos under each question

Collaboration

- Collaboration function into our design
- Invite customer support to work with database directly in customers project rather than having to send the database

Overall structure

- Support all sign brands in LTG Group

- Generate result in one software
 - Sign setup
 - Handling an dsetup of data management
 - Alignment operating programs
 - Editing destinations
- Easy switch between databases
- Faster to handle different bus types
- Make terminologies closer to user's mental model
- Prioritization - Make good features extra visible e.g. import, autoconfiguration
- Make all possible actions visible
- For tasks that require long time, it should show process bar, e.g. when import/export big files, show estimated time of the task
- Support all languages (middle east, asian, european.....)
- Good and clear overview of workspace
- Less mouse cursor movement

Setup

- Import/transfer data
 - Import databases from legacy software
 - Import bus settings
 - Import sign settings
 - Import message listing
 - Import destination files in many different formats, e.g. gtf files from Rebus, Excel, txt files, xml, csv
 - import destination files directly from traffic planning system itself
 - Make import smarter and customizable
- Settings
 - Should be possible to reuse settings and preferences
 - Set master template for all signs e.g. minimum size of text
 - Suggest name and suggestions should be based on saved settings for signs
 - No position and address limitation
 - Color defaults - both on individual signs but also overriding all template settings
- Wizard
- Select country of usage and get better more adapted functionality based on market preferences, e.g. autofill stops in your country

Editing

- Open project
 - Show last modified date
- Font, graphics and characters
 - Font & graphic editor
 - More font & graphic options
 - No font amount limitation
 - Word fonts
 - Support draw and search

- Preselected fonts, add/remove
- Create pixel by pixel
 - space bar and keyboard arrows
- Import graphic image and make it pixelated
- Shortcuts
 - For everything
 - Tab between template input
 - Shortcut for autoconfiguration
 - Customizable hotkeys
- Template
 - Company specific templates / set up your own template
 - Free template
 - Customizable template (set fonts, sizes, input areas)
 - Template should be smarter and handle title case letters well e.g. gjy
 - Pre-designed set of templates to choose from
- Layout
 - Allow customizable workplace
- Effects
 - Many effects like scrolling rotations and colours
 - Customizable effect in terms of speed, direction, no number limitation
- Preview
 - be able to check real-time effect/animation
 - You can see at the same time of editing what the text look like
- Simulation
 - See how the signs look like in the bus (3D, rotation, switch between bus type and messages)
- Undo
- Message listing
 - Default message and then change per sign
 - Reorder
 - Search
 - Show modified date
 - Possibilities to structure messages into groups
 - Provide fixed pre-programmed messages, difficult to edit
- Copy paste within one project or between several projects
 - Text
 - Graphics
 - Block wise
 - All
 - Programs/messages
 - ...
- Message input
 - Input in preview
 - Input in messaging list
 - If the text is too long, it should scroll automatically
 - Move around text blocks flexibly

- Support all keyboard characters
- Generate arrows → automatically and similar characters
 - Support draw and search
- Autofill with destination text (cities and locations)
- Lock/unlock messages
- Autoformat
 - Active by default but possible to disable
 - Active by default for both fastinput and manual input
- Fastinput
 - Could be combined with auto-formatting
 - Should be aware of when fastinput or auto formatting don't work
- Tools
 - Auto-generate capital casing a----A

Export

- Wizard
- Automatic update sign content in the bus remotely
- Specify where to save it - "USB" or "send to signs/or update signs"
- Suggest name
- Easy to share with colleagues and customers
 - Sample
 - Project

Appendix 10 - Potential improvements for the prototype in create phase 2

Following are potential improvements suggested in create phase 2:

- Add possibility to import destination list in setup wizard
- Preview signs in the workspace should be closer to reality with black color and pixelated font
- Template selection in setup stage, should also allow create new template at this stage
- Should be able to define the font for each blocks in template editor
- Improve and test icons, for example aircraft icon for graphic library is not clear enough for first time usage
- Add a error report which tells the user if any of the messages doesn't fit the signs, it should not disturb but rather tell users what the problem is and make it easy to solve it
- Add possibility to print material for bus driver
- Connect to the business system ABAS when setup of project to make it smarter, eg. list with existing signs or use order number
- Add lock function in message listing, located in the right end of the table
- Add message ID in import interface

- Allow users to import for sign position
- Add font library with different language packages
- Add operating system selection when users have Lawo signs
- Internal users could give customers different access to different sections of the software to avoid customers messing up the setup, the software slightly could change slightly to suit different markets' preferences like message listing titles, PR messages and and dashboard support

Appendix 11 - Potential improvements for the final prototype

After iteration three the prototype was coming was not updated based on the input from feedback session in iteration three. The reason was that the project was coming to an end. Therefore no design decisions were made, but the feedback was collected in following potential improvements:

- Scrolling should be possible to apply to entire messages from the message listing, possibly in the same way as rotations in current prototype.
- Scrolling and rotation effects could also be applied by marking text in the sign (apply only on that sign) or in the message listing (apply to entire message) and then click on a button in the toolbar.
- Provide better overview of rotations in the preview.
- Make it possible to use left and right mouse clicks to add and remove pixels in the graphic and font editors.
- Make it possible to import fonts from external sources in the font editor.
- Make it possible to create template block by pixels and ratio in the template editor and provide more data in the template overview of the sizes of the blocks.

Software structure and elements

Appendix 12 - Functional and data elements and their grouping

Setup

Container	Data elements	Functional elements
Setup account	Account	Register account (username, password) Remember account Save account Edit account (username, account)

		Delete account
	License	Enter license number
Suite overview	Applications	Select and enter application Add new application
Project overview	Existing projects	Open existing project
	New project	Create a new project Name new project
	Recent project	View recent projects in list Open recent project
Setup fleet	Import	Import fleet setup/database
	Vehicle	Define name Select Vehicle type from list Select brand from list Select control unit from a list Delete/copy/paste vehicle Change vehicle setting
	Sign	Select identification method (from list, article number or order number) If list: Select sign type from list Select position or enter manually Choose suggested address or enter manually Define sign name Choose template from list Preview sign Add another sign Delete/copy/paste sign

Edit

Container	Data elements	Functional elements
Topbar	Application suite	Go back to application suite
	Application icon	Visual feedback of current application
	Project tabs	Visual feedback of current projects Click on tabs to switch between projects Copy and paste vehicles, signs and messages between projects

	Setup tab	Click to enter fleet setup interface
	Edit tab	Click to enter edit interface
	Export tab	Click to enter export interface
	General controls	Search in software Access to cloud Access to software settings Help button Access to account settings
Toolbar	Undo/redo	Click to undo/redo action
	Trash bin	Delete active object
	Fonts	Select font from list Select font size from list
	Color	[If RGB sign otherwise unavailable] Select text color Select text outline color Select background color
	Upper/lower case letters	Change text into upper/lower case letters
	Double stack letters	Divide sentence on two rows
	Spacing	Increase spacing between letters Decrease spacing between letters Use standard spacing
	Positioning	Align object in nine positions within the template area Adjust pixel by pixel position of object with keyboard arrows
	Editors	Path to Graphic library and graphic editor Path to Font editor Path to Template editor
Workspace	Preview signs	'What you see is what you get' - overview of message specific sign content Change text content for specific sign Add rotation to specific sign Flip through rotations Enlarge selected sign Select sign to activate properties are for that sign
	Message display	Feedback of which message currently

		displayed Change displayed message by enter number or arrows
	Zoom	Increase and decrease the size of the signs
Message listing	Message	Enter sign content in the table Click "+" to add new message Select message to delete or duplicate Select message and drag to change order
	Message in single view	Enter message in the different entry areas
	Rotation	Hover over message to activate rotation mode Use "+" to add rotation and enter the text Use lock function to freeze cells in the default message to copy the cell to following rotations Deactivate rotations by clicking at the icon
	Import destination list	Click on "import" and select file in popup window
	Search	Search for specific message or content
Properties	Sign simulation	Simulation feedback of the selected sign with effects Feedback of sign name Click on name to change
	Template selection	Feedback of what template the sign is currently using for that message Change current message Path to template editor
	Rotation	Add rotation to selected sign (not for entire message) Delete rotation Order rotations Change duration of rotations
	Scroll	Select/deselect scrolling area to activate/deactivate scrolling to selected sign (not entire message) Adjust scrolling speed Select scrolling direction
	Setting	Feedback of sign name, size, position and address Click to change settings
Graphics	Graphic library	Select and add graphic from library

		Import graphic
	Graphic editor	Name graphic Create graphic by pixel by pixel editing Import and pixelate image Import image and use as underlay Select and duplicate/delete area of pixels
Font editor	Existing font	Make pixel by pixel changes to existing font
	New font	Select pixel size Create new font pixel by pixel Import font to use as underlay
Template editor	Input blocks	Drag and drop input blocks to template Change default name of input block Define font for input block Change size of input block by either pulling the area block or enter number manually Select/deselect dynamic line width

Export

Container	Data elements	Functional elements
Export	Export outfile	Select all/group/one vehicle types to export Select where to save Autodetect USB
Print	Print	Select what to print (eg. drivers notes, message list, preview) Select all/group/one messages to print

Appendix 13 - Low level use cases

Setup section

- Account
 - Enter license number
 - Register account by defining username and password
 - Remember account
 - Help with lost password
 - Log in
- Suite overview

- Overview of what software are available
- Select application
- Project overview
 - Overview of existing projects
 - Overview of recent projects
 - Create a new project
- Setup fleet
 - Import database from previous editing software
 - Setup vehicle (name, brand, control unit)
 - Import vehicle settings from other projects
 - Create vehicle
 - Define vehicle name
 - Select vehicle type from a list
 - Select sign brand from a list
 - Select control unit from a list
 - Change vehicle setting
 - Change name by inputting text
 - Change sign brand by selecting from a list
 - Change control by selecting from a list
 - Delete/Duplicate/copy and paste vehicle to reuse vehicles
 - Setup signs (name, size, resolution, pitch, address, brand, position, graphic representation of sign, drivers message)
 - Add new signs
 - Select from list of different sign sizes
 - US version: should offer dashboard size in the list
 - (Enter manually)
 - Enter serial/article/part number
 - Order number
 - Copy and paste and duplicate signs
 - Select position (front, side, rear, drivers message)
 - US version: should offer dashboard in the list
 - Select sign address from a list, should offer suggestions based on sign brand and position selection
 - Select template from a list or navigate to template editor to create new template
 - Add split signs
 - Copy and paste and duplicate to reuse signs
 - Change sign settings
 - Delete signs
 - Overview of fleet and settings
 - Graphical view
 - Table view (Think Cinnabar)
 - Search sign or vehicle

Edit section

Message listing - Similar as Fast input, apply the content to all the signs

- Import messages
 - Import from excel, svg..... documents
 - Import from traffic planning system
 - Define import message range
 - Define import all messages or skip existing ones in the message listing
 - Import by sign position
- Add new message manually
 - Add message with default template
- Edit message
 - Add graphic to one or several template areas/blocks
 - Effect
 - Add multiple rotations to one message
 - Add multiple rotations with different templates
 - Add multiple rotations with freeze template areas function
 - Add scrolling to one or several template areas/blocks in one message
 - Get an overview of messages and their effects
 - Delete effect
- Settings
 - Change order of messages
 - Delete message/s
 - Duplicate message/s
 - Define bounded/unbounded
 - Add new column to the message listing???
 - Lock/unlock message

Preview signs (signs, sign name, sign dimension, effect display)

- Enter message valid only for that sign
- Change message content
- Select sign to get info/effects in properties
- Select object
 - Double click to edit
 - Single click to select block/object
- Edit blocks with tools
- Move blocks
- Delete/copy/paste/duplicate blocks
- Zoom in one individual signs
- Adjust size of sign perviews
- Display current message code, go to previous one and go to next one
- Add rotation on selected sign
- Go through current rotations of selected sign

Properties

- Simulation

- Check real-time effect simulation
- Template
 - Display the current template of selected sign
 - Change template from default list or personal templates
- Rotation
 - Activate rotation
 - Lock which blocks which should stay the same in next rotation
 - Add new rotation
 - Adjust rotation duration
 - Display all rotations of selected sign
 - Change order of rotations
 - Copy/duplicate/delete rotations
 - Determine which blocks to be locked
- Scrolling
 - Activate scrolling
 - Select which block(s) to scroll
 - Adjust scroll speed
 - Define scroll direction
 - Adjust scroll area
- Setting
 - Change sign name
 - Change sign size
 - Change sign position
 - Change sign address

Font editor

- Import fonts (e.g. windows)
- Create new font from scratch
- Create new fonts based on default font
 - Check font preview
 - Pixel editing for each letter

Fonts library

- Select from default list
- Add fonts from font library to font list
- Remove fonts from font list
- Add font language package into default list
- Sort fonts
- Search font

Graphic editor

- Create new
- Name new graphic
- Define name
- Define size
- Import pixelated graphic and make changes

- Import normal picture and use as underlay
- Select from graphic library and make changes
- Pixel edit the graphic by turning on and off pixels
 - Mouse cursor
 - Keyboard arrows

Graphic library

- Overview of all default graphics
- Search graphic
- Create new graphic - navigate to graphic editor
- Insert graphic in message listing or individual signs from default graphics

Template editor

- Make changes to current templates
- Define name
- Create template from scratch
 - Add content blocks (Line, Destination, Via, free)
- Save as new
- Overwrite existing template

Toolbar

- General
 - Redo
 - Undo
 - Duplicate
 - Delete
- Text
 - Font list
 - Font size
 - Background colour
 - Text colour
 - Outline colour
 - Background colour inverse
 - Upper case
 - Lower case
 - Double stack text in two rows
- Spacing
 - Increase space
 - Decrease space
 - Standard space
- Position
 - Align
 - Up/down/left/right arrows for pixel movement
- Create tool
 - Text block
 - Pen tool

- Line
- Dots
- Rectangle
- Circle
- Pixel erase

Fleet overview

- Overview of all buses and signs
- Select what to be displayed in the workspace

Export section

Export

- Select bus types to export for
- Export to USB
- Export and transfer wirelessly

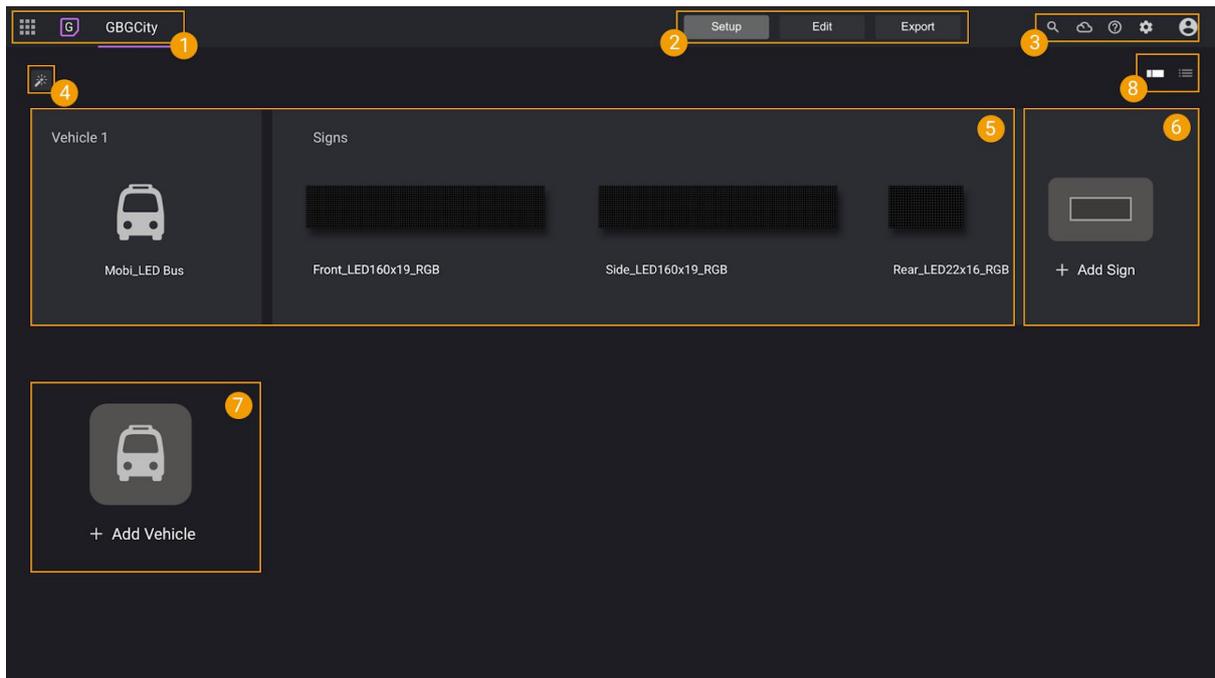
Print

- Select what to print (drivers notes, preview, destination list)
- Preview print job

Appendix 14 - Overview of the final design

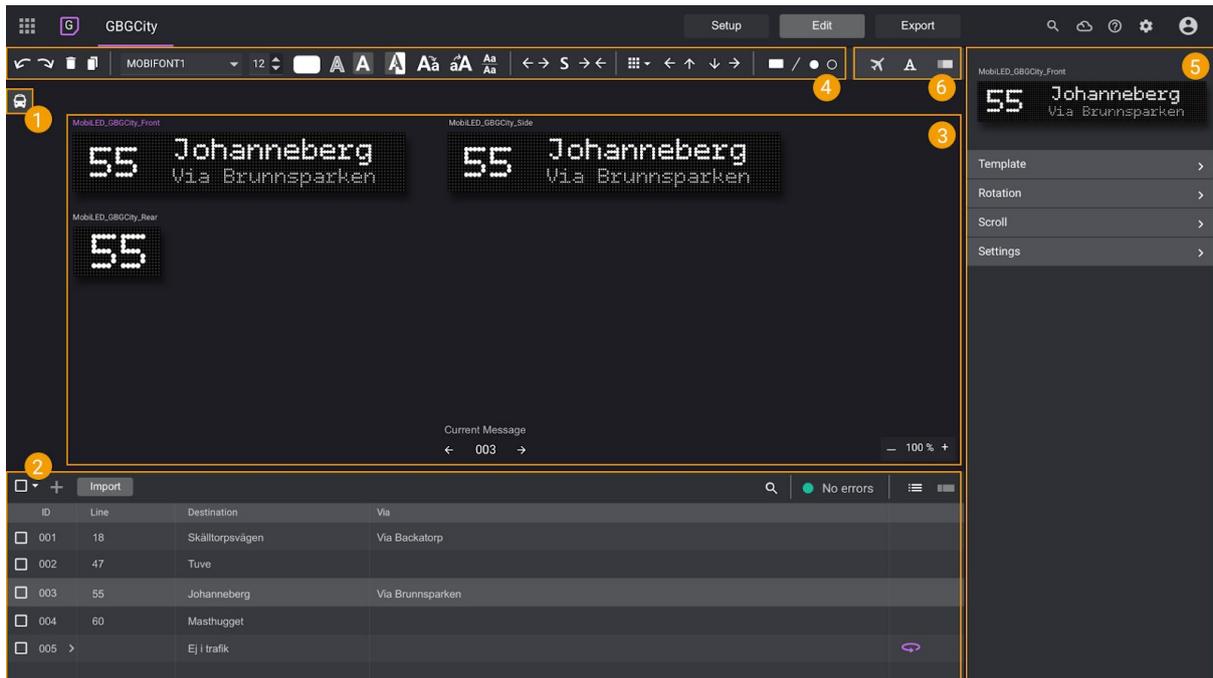
This section presents the data elements and their functions in the three main interfaces, *setup*, *edit* and *export*.

Setup



1. LTG Suite center and current project
2. Navigation buttons between three key usage phases, *setup*, *edit* and *export*
3. LTG Suite general settings including search, cloud service, help center, settings and user profile
4. Setup wizard for novice users that provides step by step setup guidance
5. Vehicle and sign overview that offers an overview of current vehicles and signs in a graphical way
6. Create new sign for current vehicle
7. Create new vehicle

Edit



1. Foldable fleet overview that provides an overview of created vehicles and signs, by ticking and ticking them users could control what signs are visible in the workspace
2. Message listing - Message input area that applies input content to all signs, it is divided into line, destination and via columns by default but could be customized by creating new template

Import

- a. Import - import destination list from Excel document

+

- b. + - add a new message manually

● No errors

- c. Error detector that reports the error status of current project

🔍

- d. Search - Search keywords among messages

☰

- e. Two message display views: multiple message list view and single message view

ID

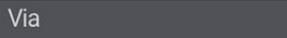
- f. ID - message ID number

Line

- g. Line - Line number input column that corresponding to line blocks in templates

Destination

h. Destination - Destination input column that corresponding to destination blocks in templates



i. Via - Via input column that corresponding to via blocks in templates



j. Rotation icon - Add rotation to current message, it only appears when the user moves or hovers the pointer over the message

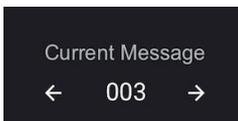


k. > - Fold and unfold rotations

3. Workspace - Work area where the preview of signs are displayed and changes to individual signs could be made



a. Sign preview shows the basic information and content of each sign



b. message number that indicates which message is displayed currently



c. Zoom in/out that allows users to check signs in different distance views

4. Toolbar that includes all tools for sign editing



a. General tools: Undo, redo, delete, duplicate



b. Text tools: Font selection, size selection, text colour, outline colour, background colour, text and background colour reverse, uppercase reverse, lowercase reverse, Double stack into two rows



c. Spacing tools: increase space, standard space, decrease space

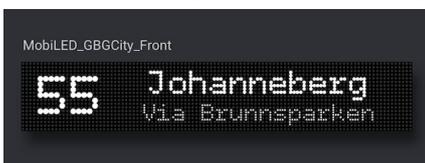


d. Position tools: Align in 9 ways, up/down/left/right pixel movement



e. Create tools: text block, pen tool, draw, erase

5. Property settings for selected sign



a. Real time simulation of selected sign



b. Template setting for selected sign



c. Rotation setting for selected sign



d. Scroll settings for selected sign



e. Setup settings for selected sign

6. Editors that allow users to check available font, graphics and template as well as create new ones



Graphic library and editor which allows users to insert graphics from default graphic library and create new graphics from scratch

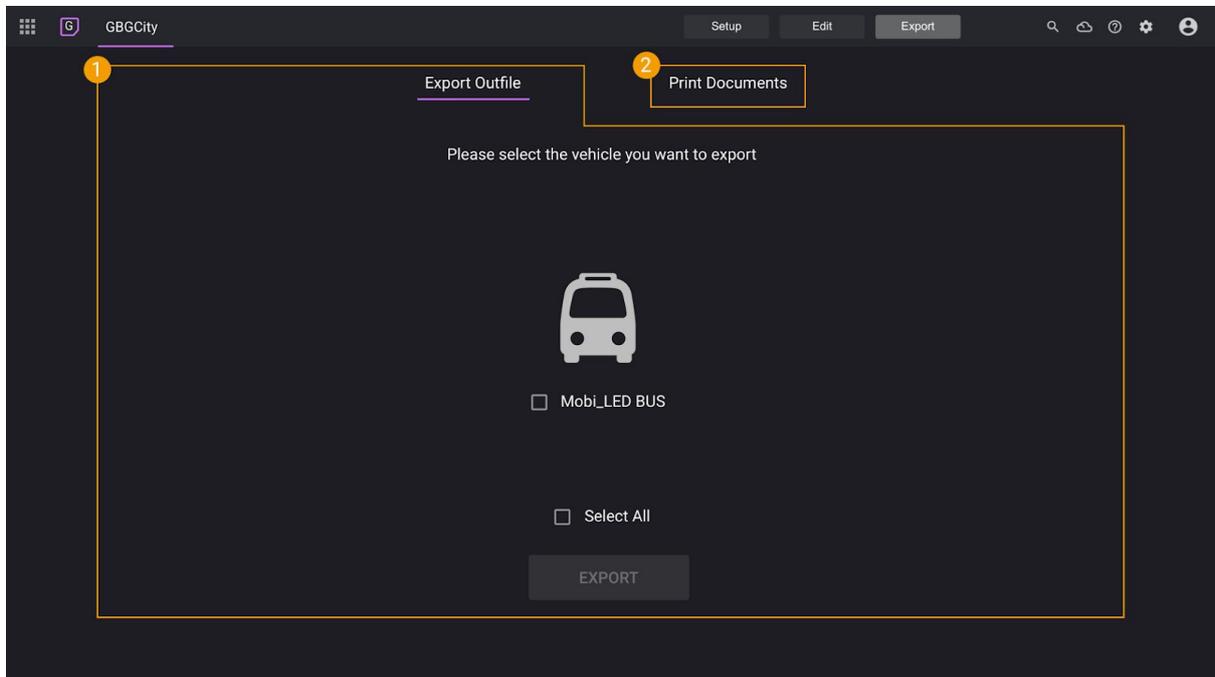


Font editor



Template editor

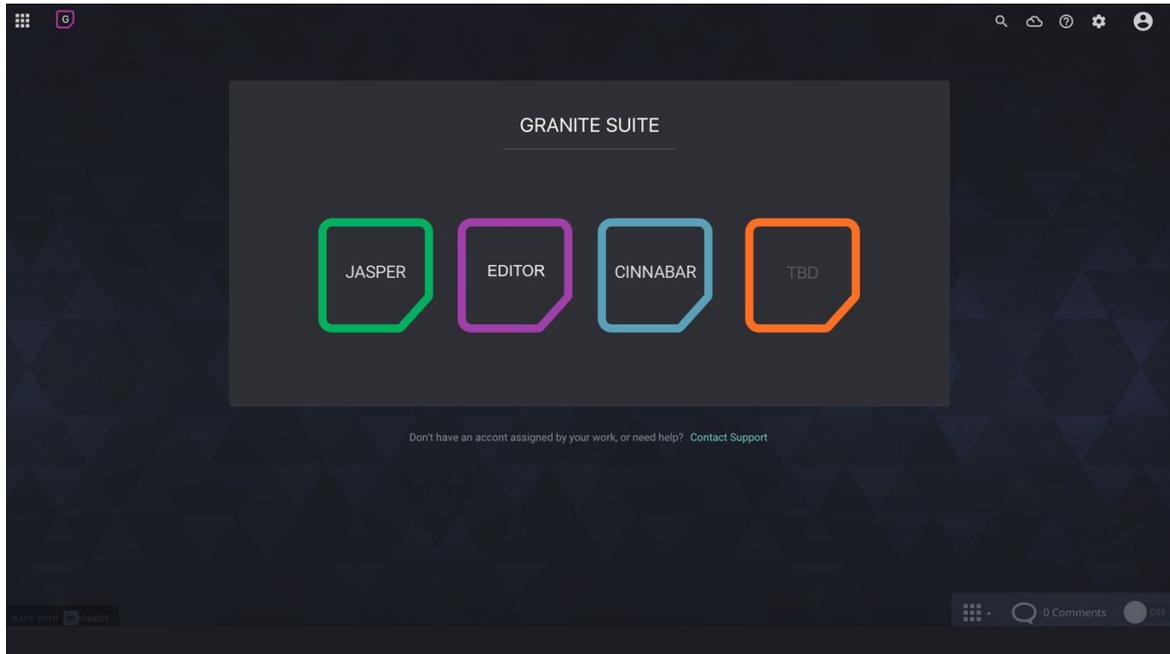
Export



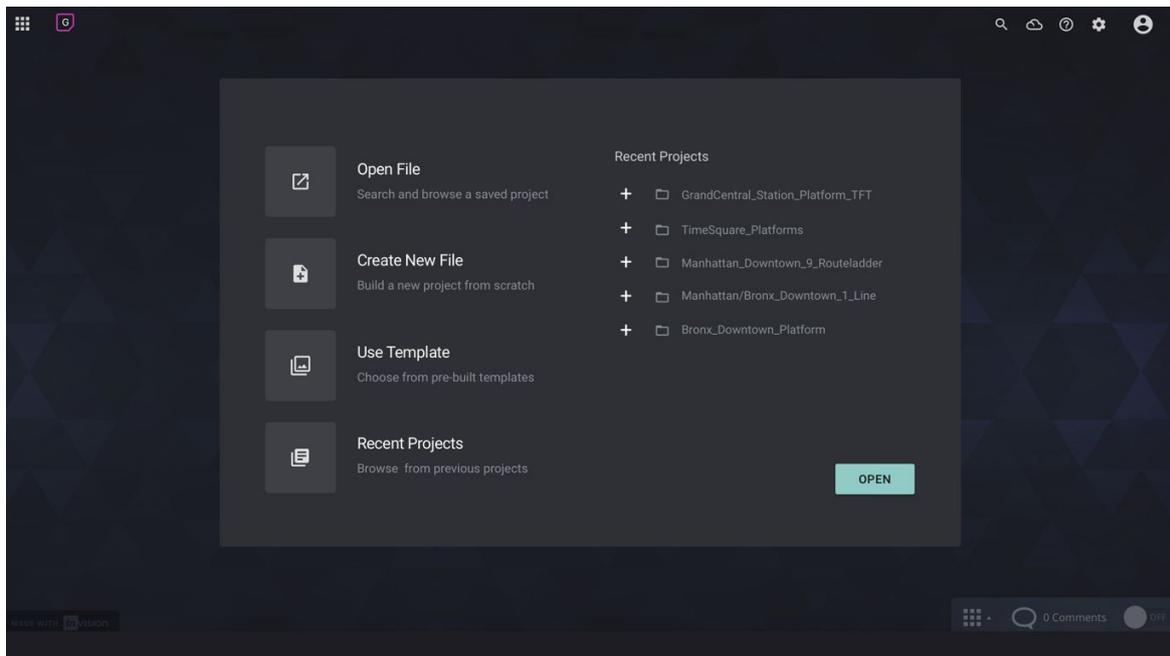
1. Export outfiles that are contains sign content information to load into vehicles
2. Export print documents for administrative purpose

Usage scenarios

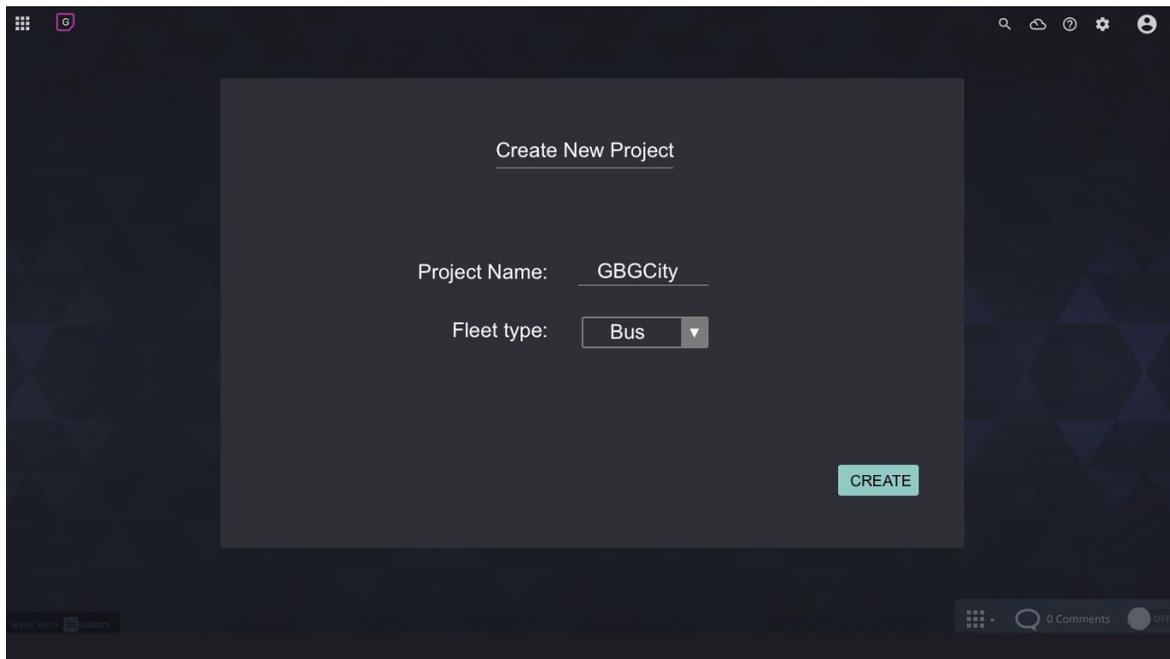
Appendix 15 - Usage scenarios of the prototype in create phase 2



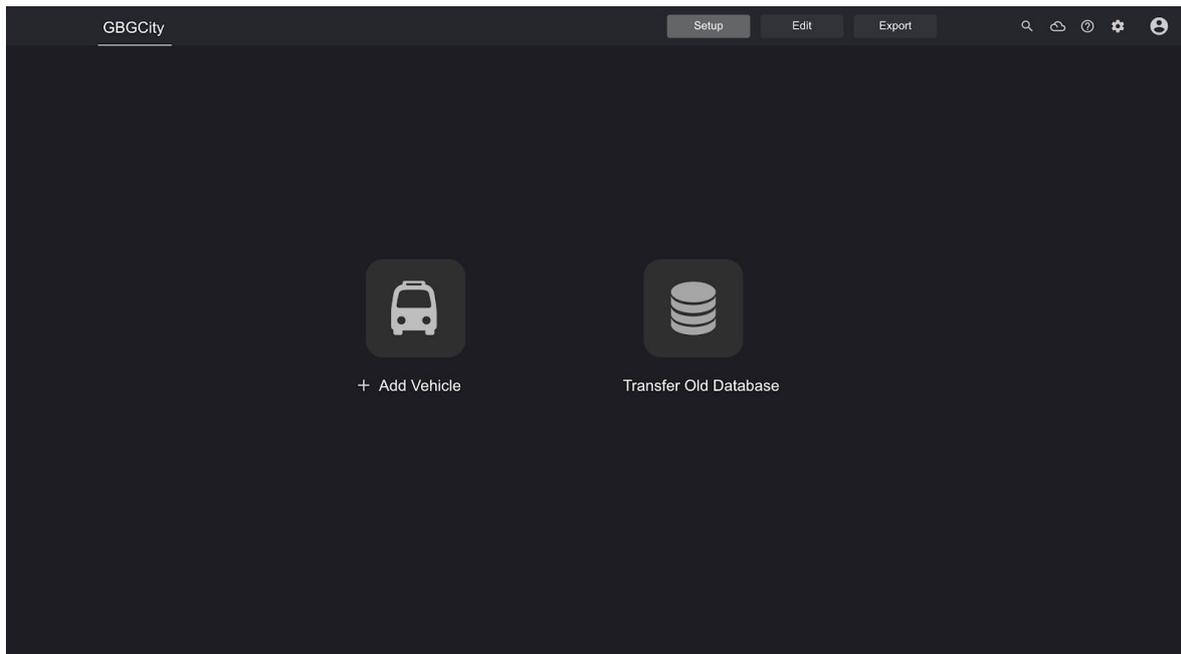
1. LTG Suite overview where all software within LTG suite are presented here.



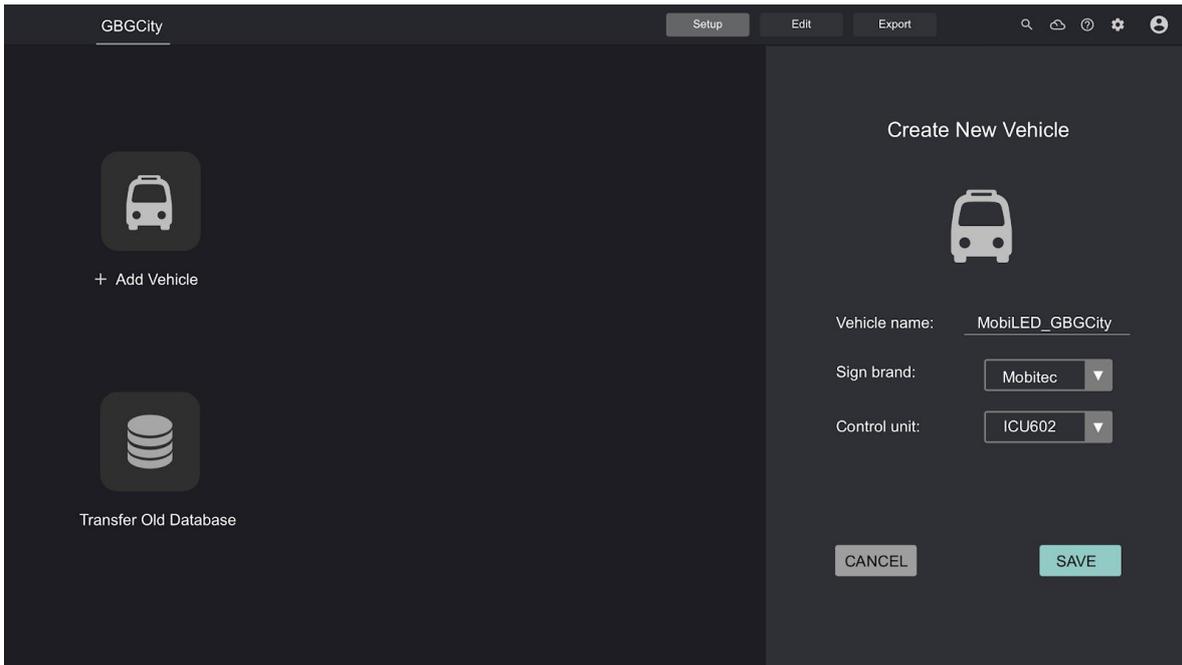
2. Project overview which allows users to open existing projects, create new project and go through recent projects.



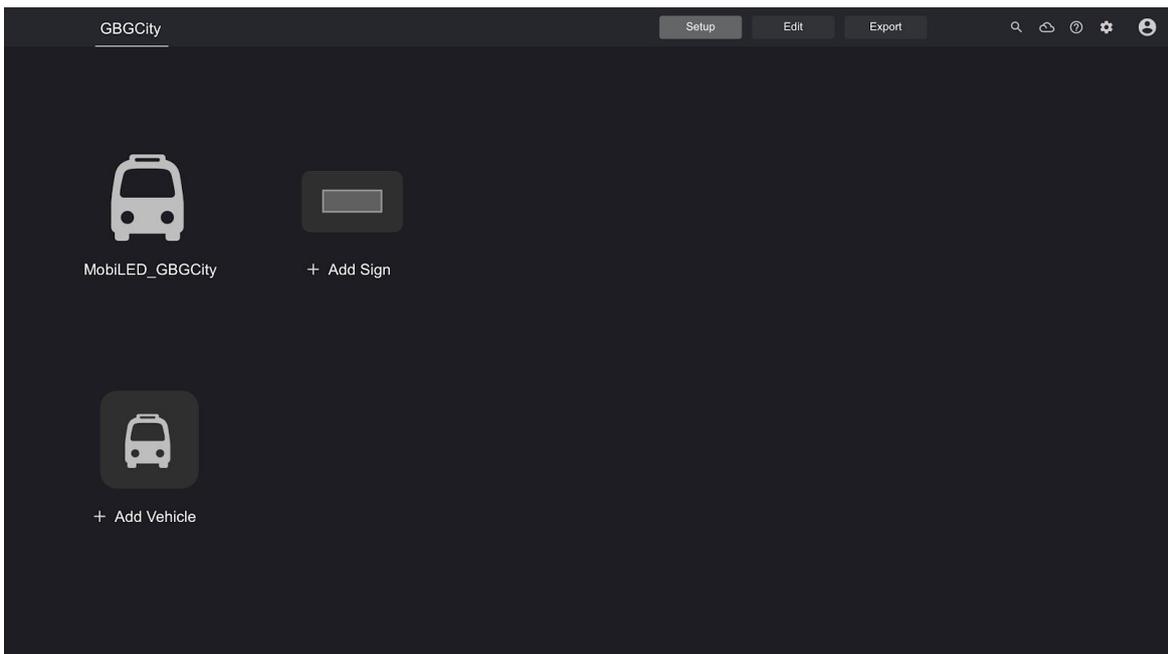
3. Create a new project by entering project name and selecting fleet type from the drop-down list.



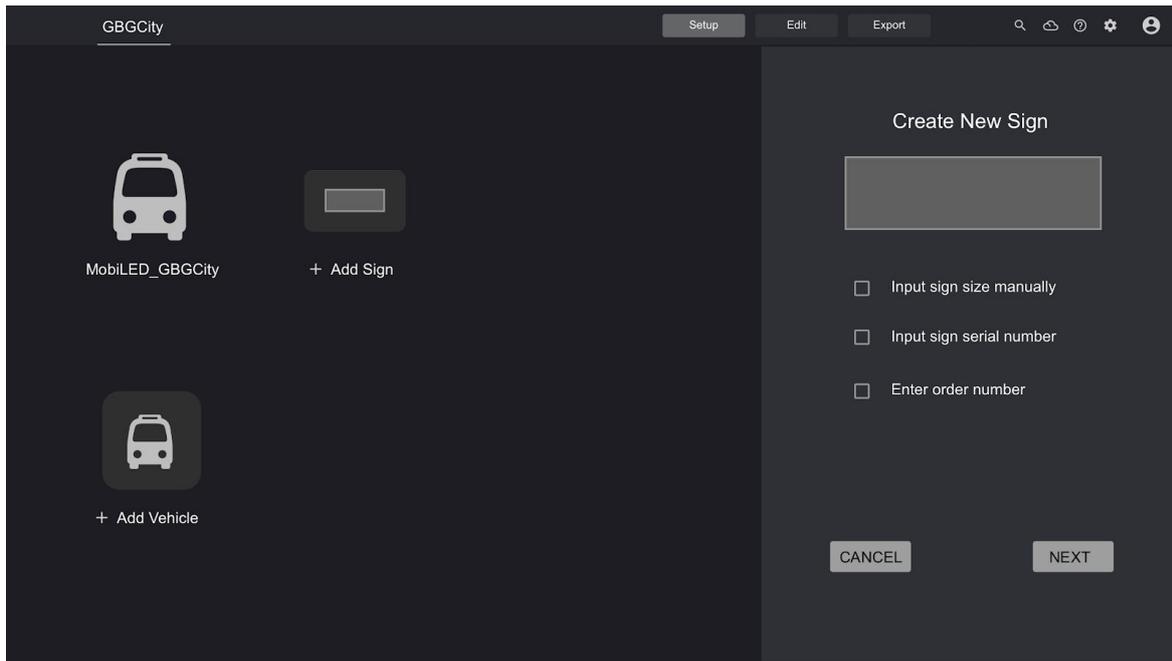
4. Setup the fleet by adding vehicle manually and transfer databases from previous software.



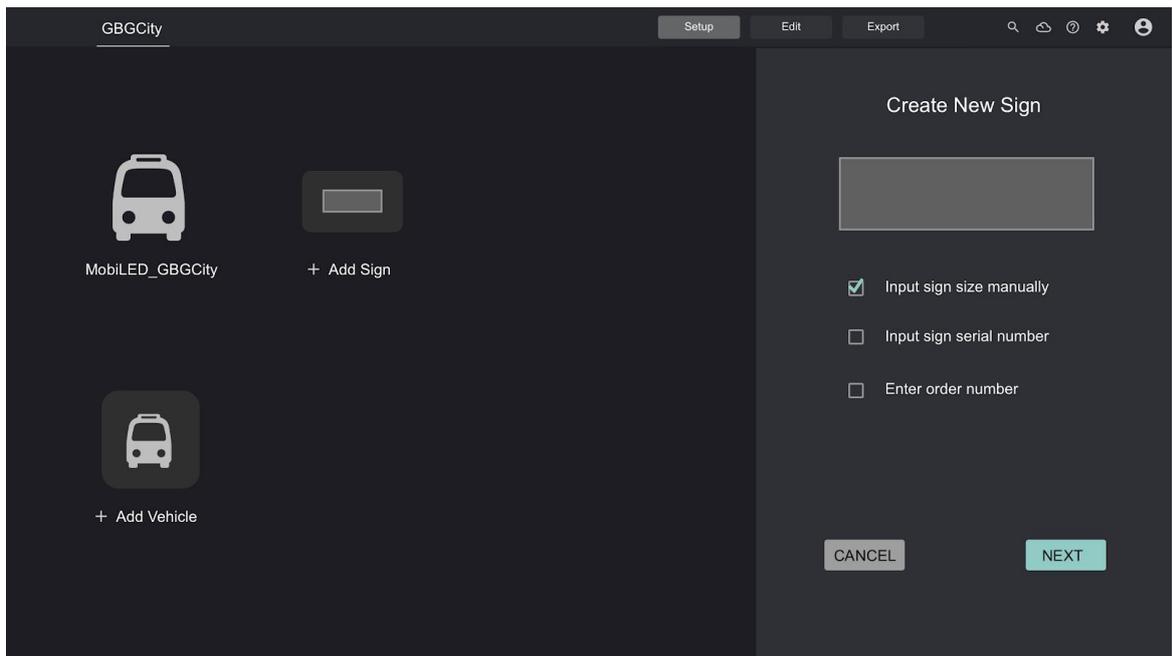
5. Create new vehicle manually by entering vehicle name, selecting sign brand and control unit type.



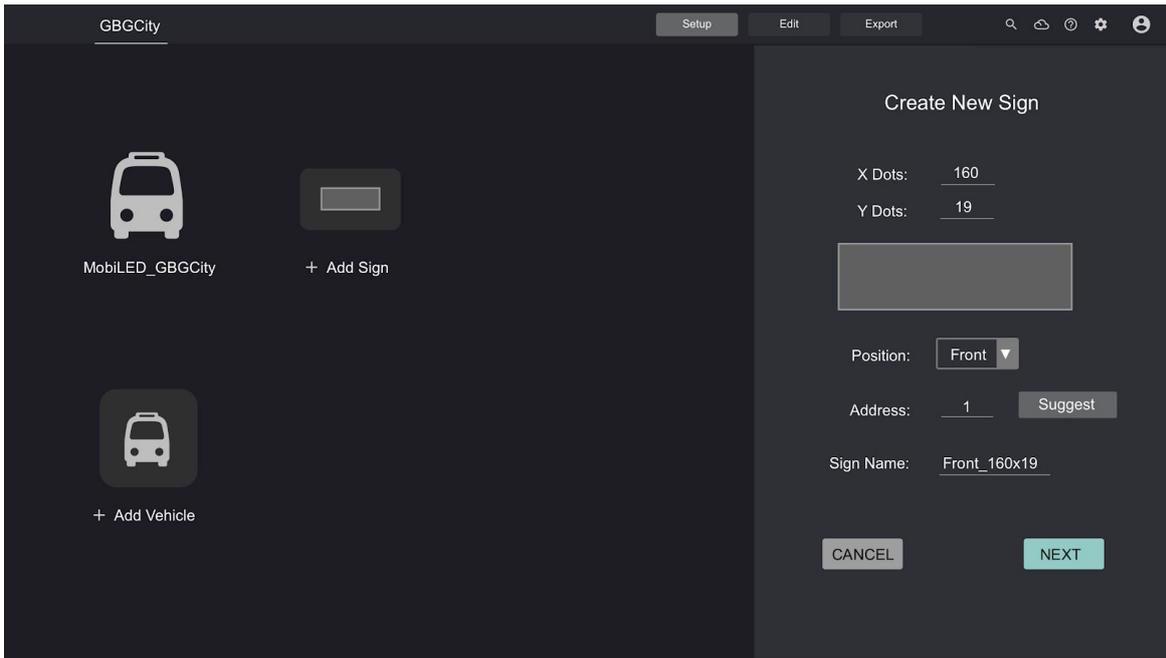
6. A new vehicle 'MobiLED_GBGCity' was created, click 'add sign' to create signs for this vehicle.



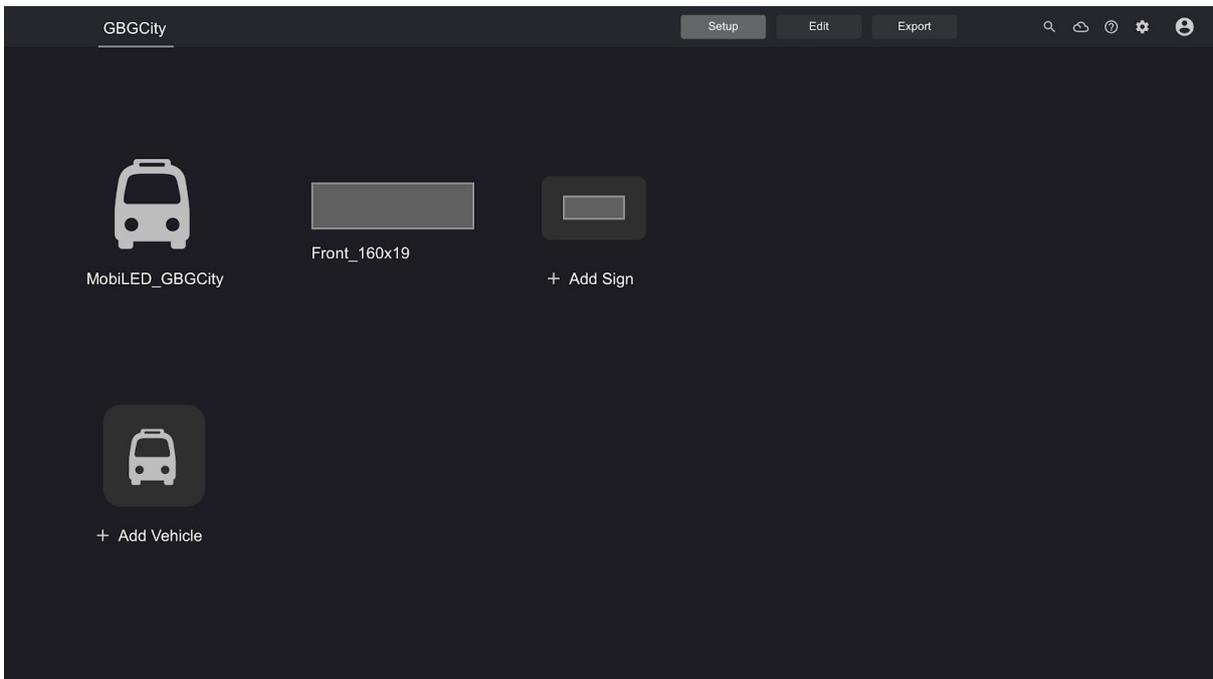
7. There are three ways to create new signs, namely input sign size manually, input sign serial number or enter order number. Users could select the way that they prefer, depending on their knowledge of the sign and accessibility to the hardware.



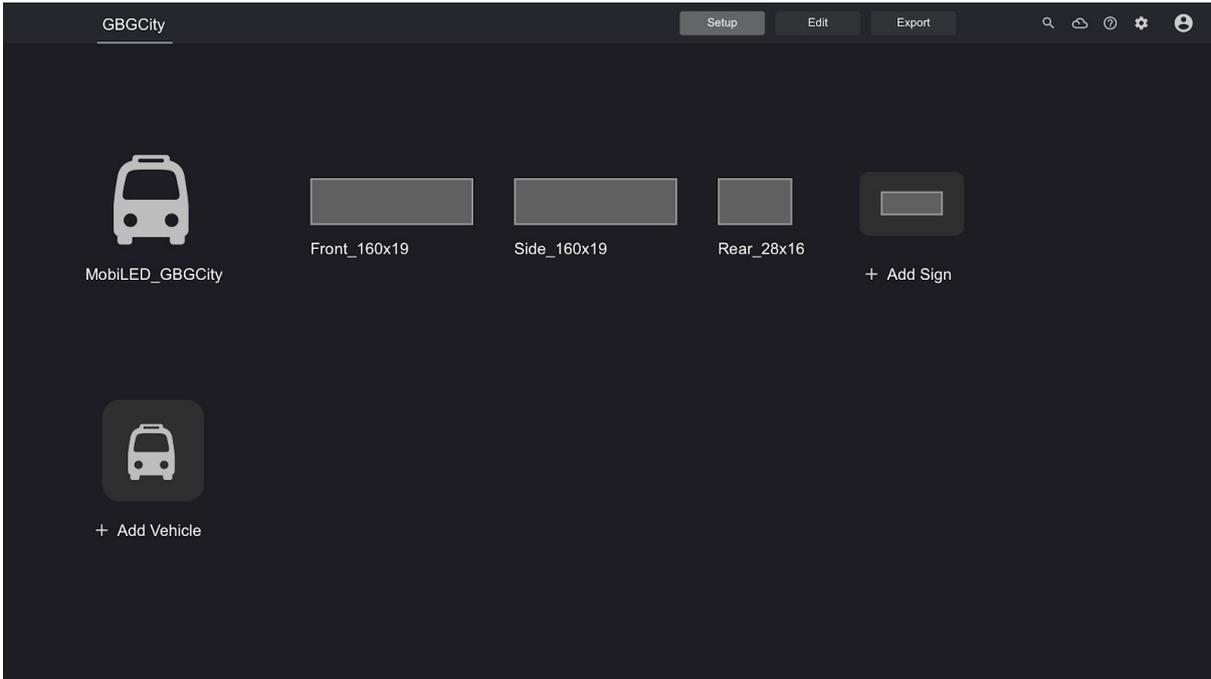
8. Select input sign size manually



9. In order to create a new sign manually, users need to input sign size by the number dots in X and Y dimensions, then select sign position, input address number or ask the software to suggest one according to the position selection. Based on above settings, the software will generate a default sign name automatically, such as 'Front_160x19', if users are not satisfied with the name then they could edit it manually.

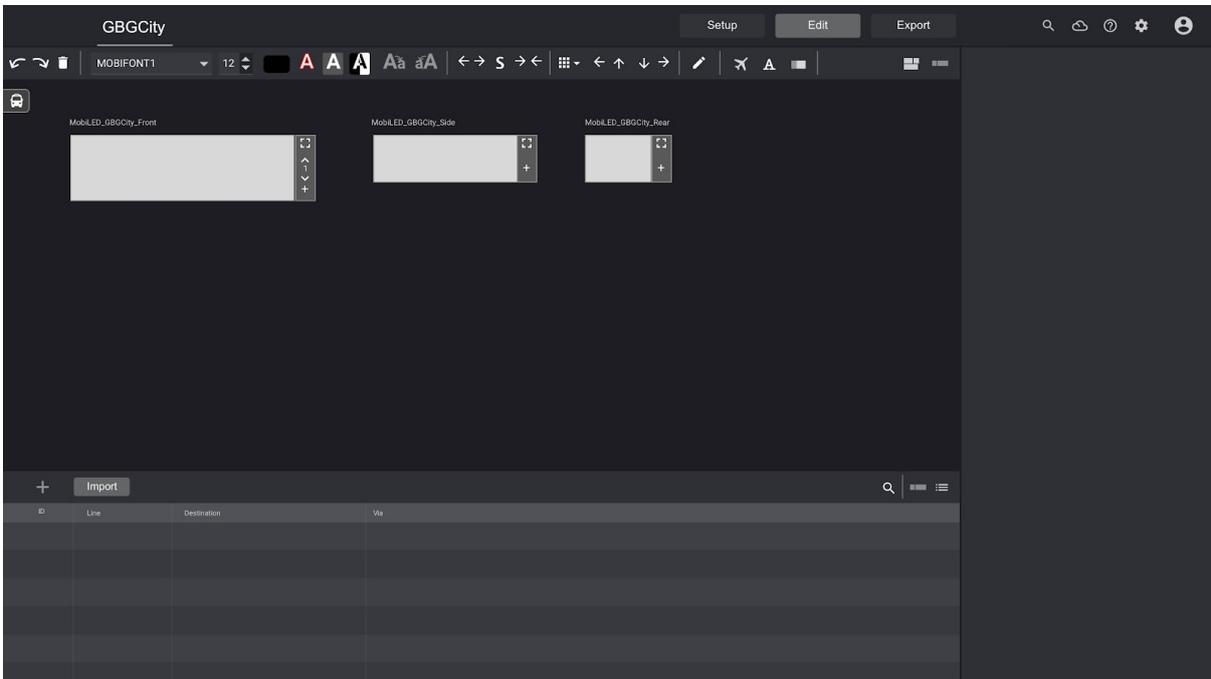


10. A front sign was created successfully

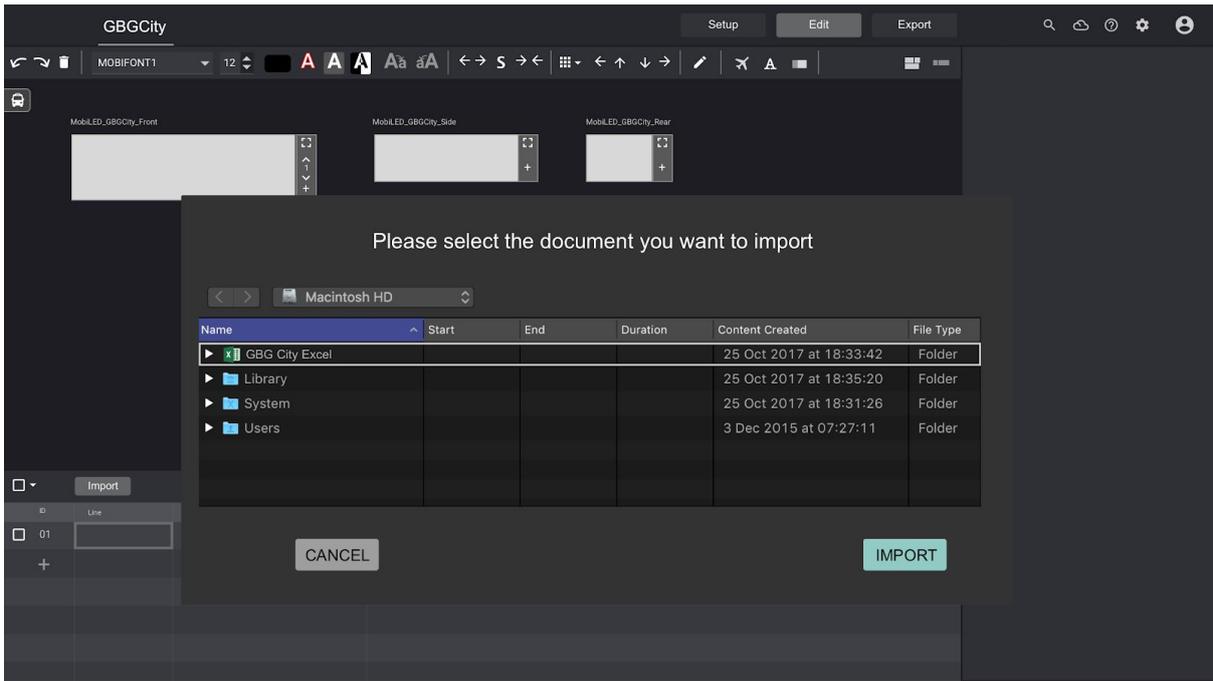


11. Following the same procedure, side sign and rear sign were created. Now the setup of vehicle 'MobiLED_GBGCity' was done, click 'edit' in the top bar to edit these signs.

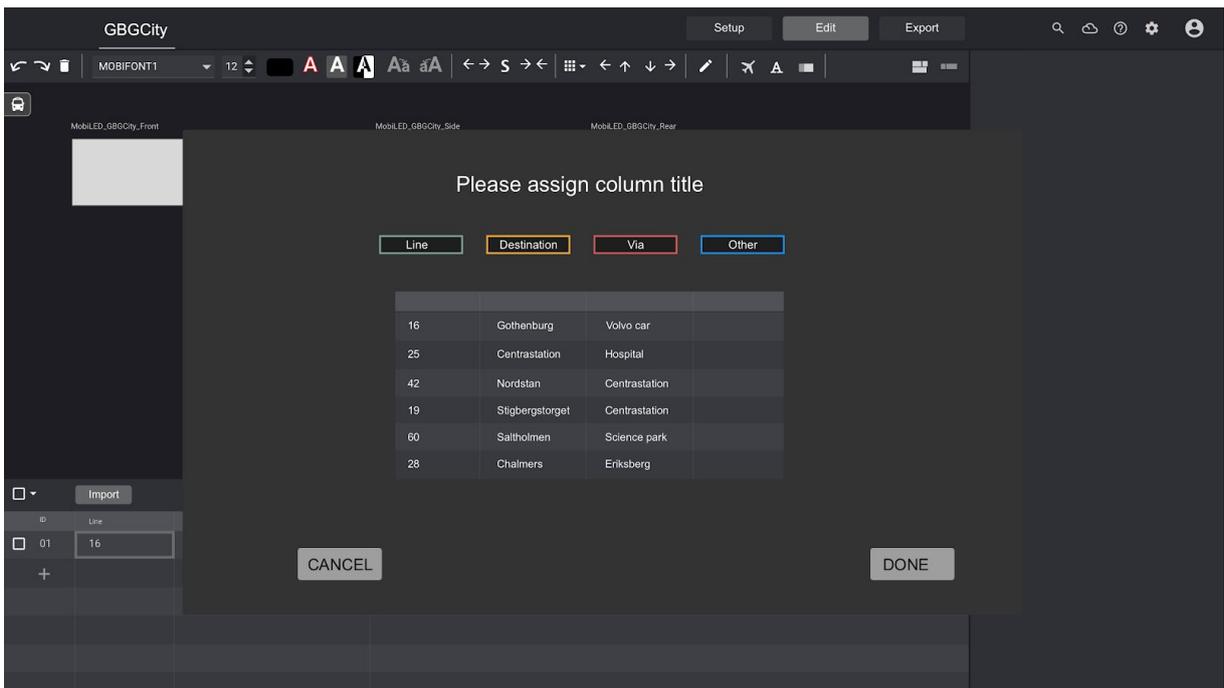
Scenario 2 - Import destination list



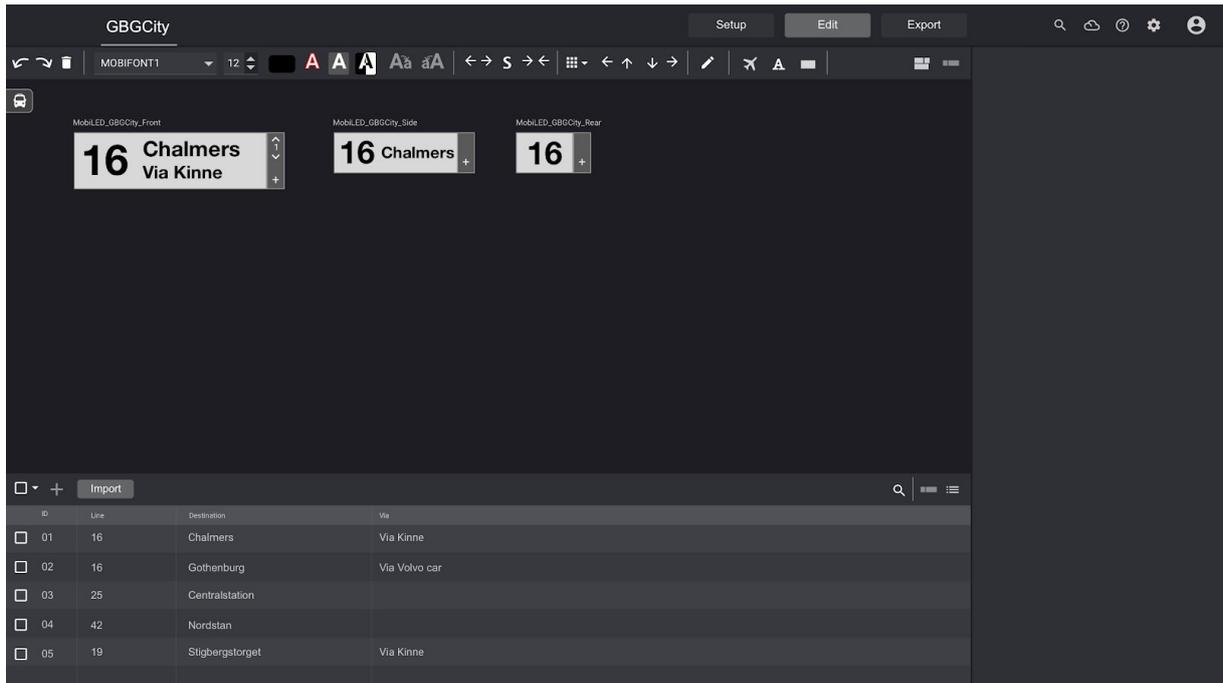
12. Switch to 'edit' mode, the edit interface was divided into fleet overview, workspace, toolbar, message listing and property areas.



13. Click 'import' to import a list of destinations to message listing. Considering the Excel sheet 'GBG City Excel' as an example.

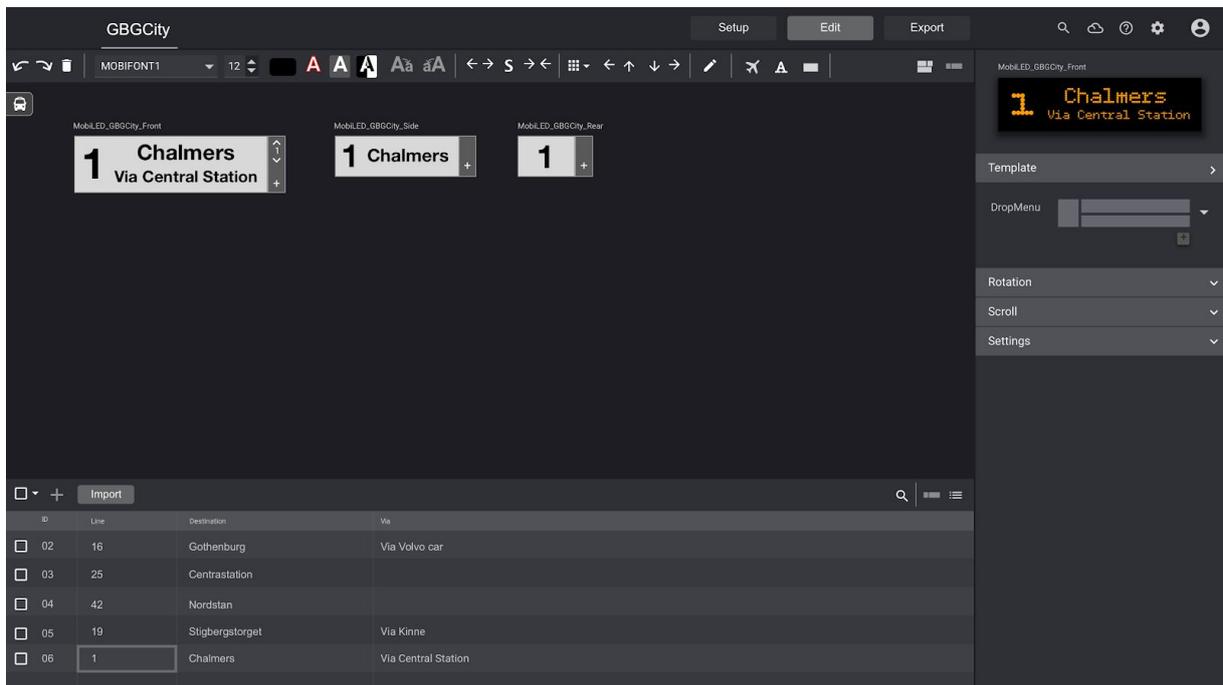


14. Assign titles each column to let the software know where to display the content.



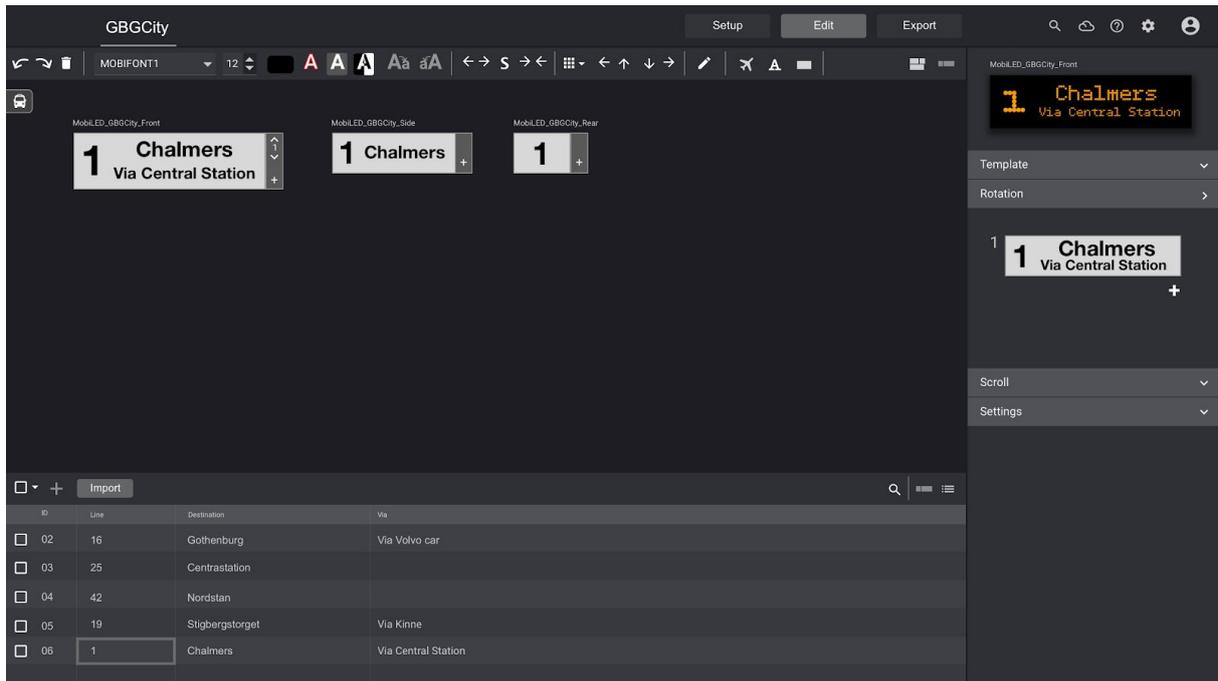
15. The document was imported successfully, thereby the content in the Excel document was filled into message listing and workspace automatically.

Scenario 3 - Change template

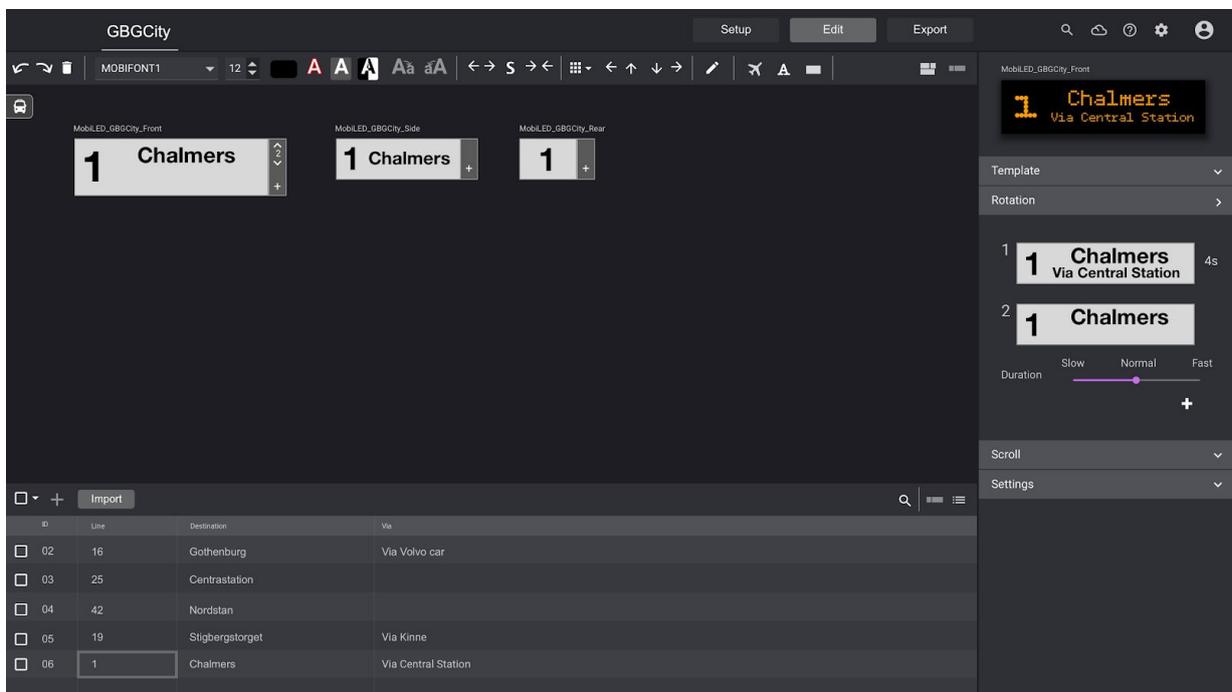


16. By clicking on individual signs in workspace, the property interface changes accordingly which allows users to make changes including template, rotating and scrolling effect and sign information on specific signs as well as check real-time simulating effect. In the template section, users could change the current template of the front sign by selecting from the drop-down list.

Scenario 4 - Add rotation

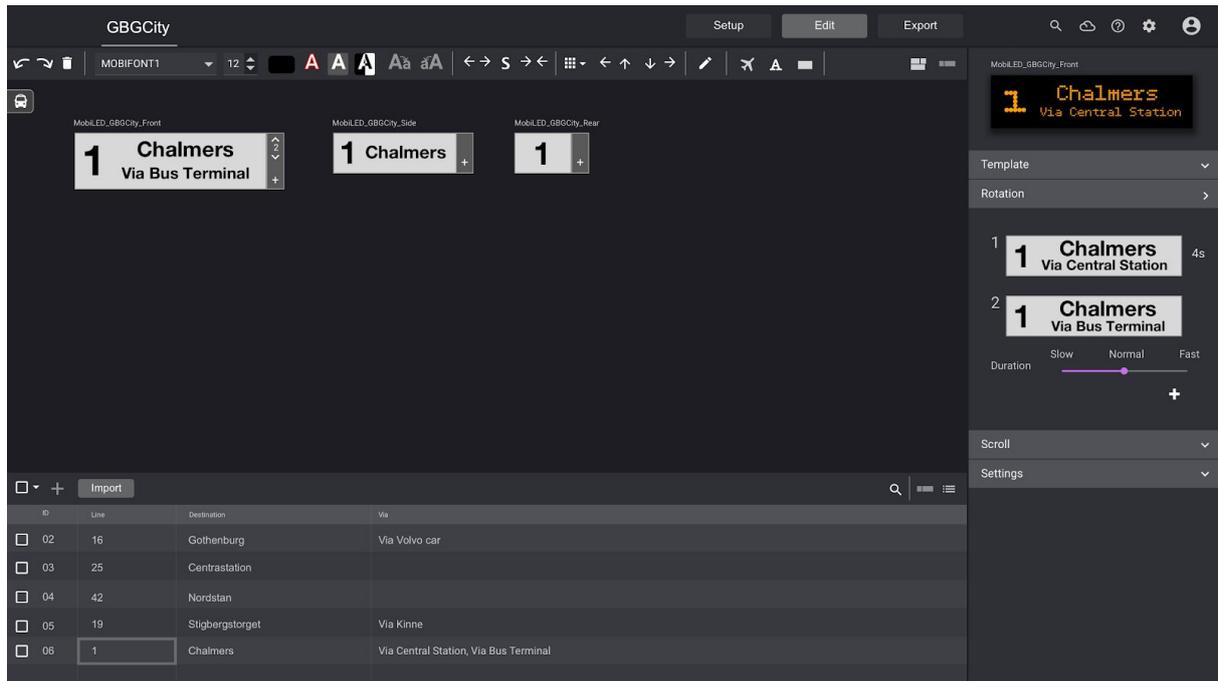


17. In the rotation section, by clicking the '+' button, a new rotation will be created on the selected sign.

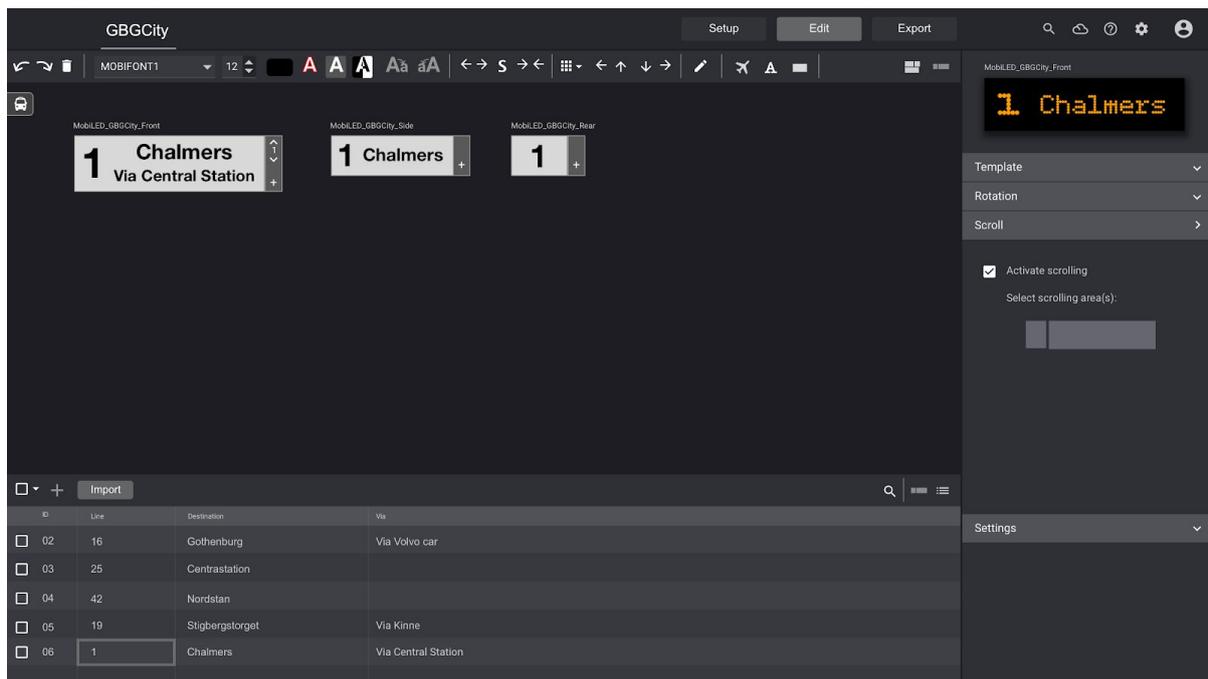


18. In the workspace, users could fill in the content in the new generated rotation and get an overview of all rotations and change rotating duration in the property area.

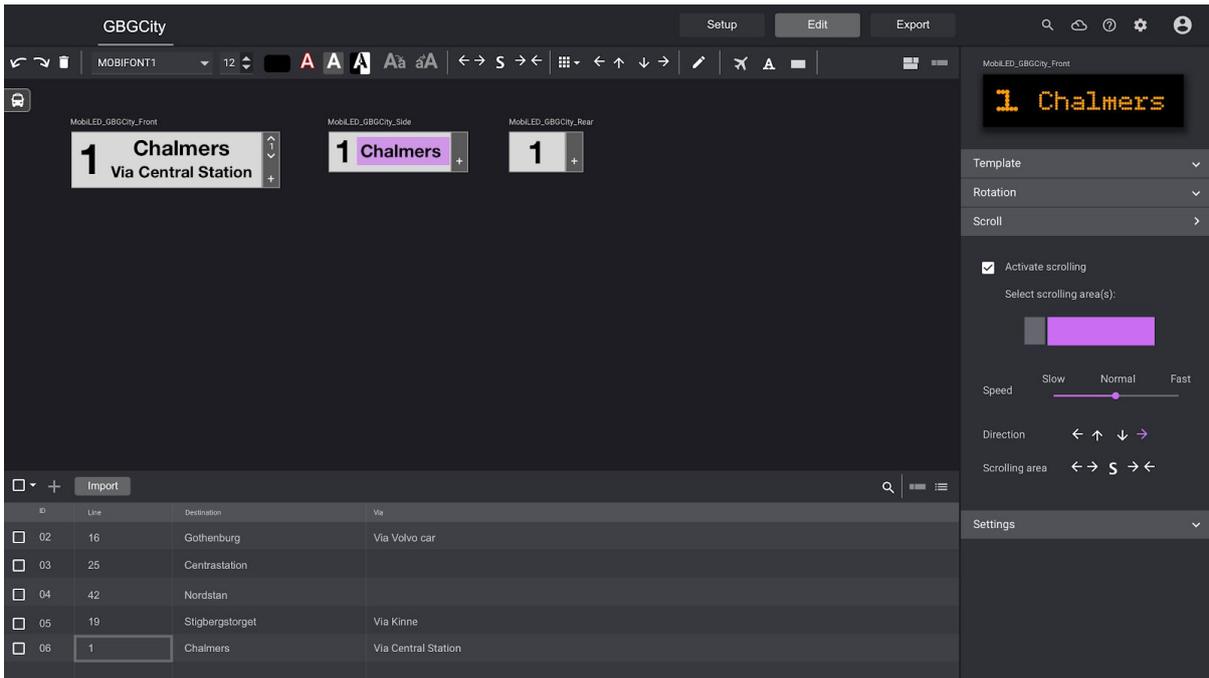
Scenario 5 - Add scrolling



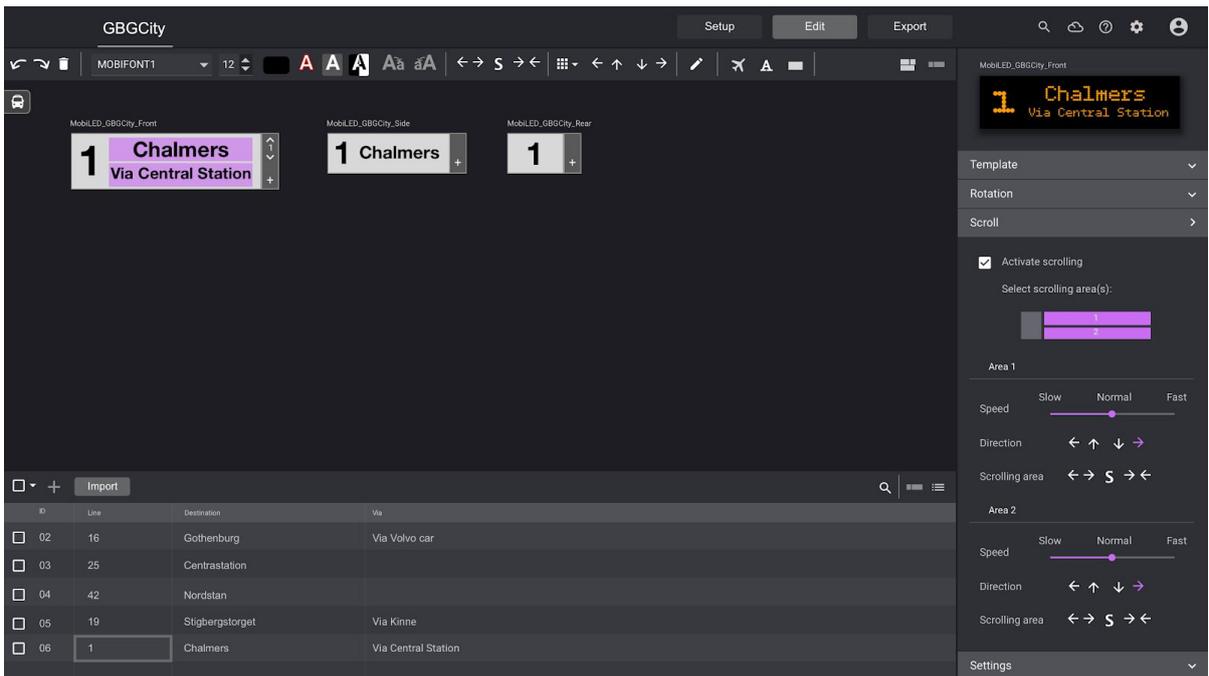
19. Inputting 'Via Bus Terminal' in the second rotation as an example.



20. Considering the side sign as an example. In the scroll section of property window, users could activate the scrolling effect.

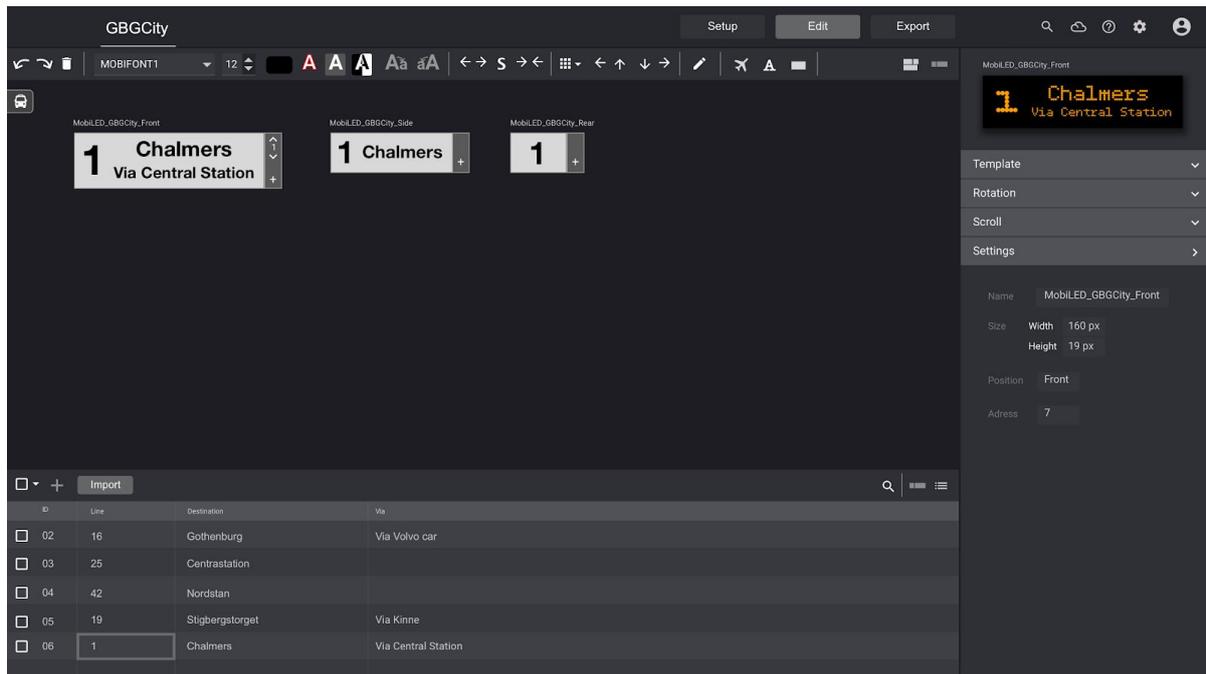


21. After activating scrolling, users are required to select scrolling area(s) and adjust the settings for scrolling, including scroll speed, scroll direction and scroll area. The scroll area was also marked in the sign in workspace to provide users visual feedback.



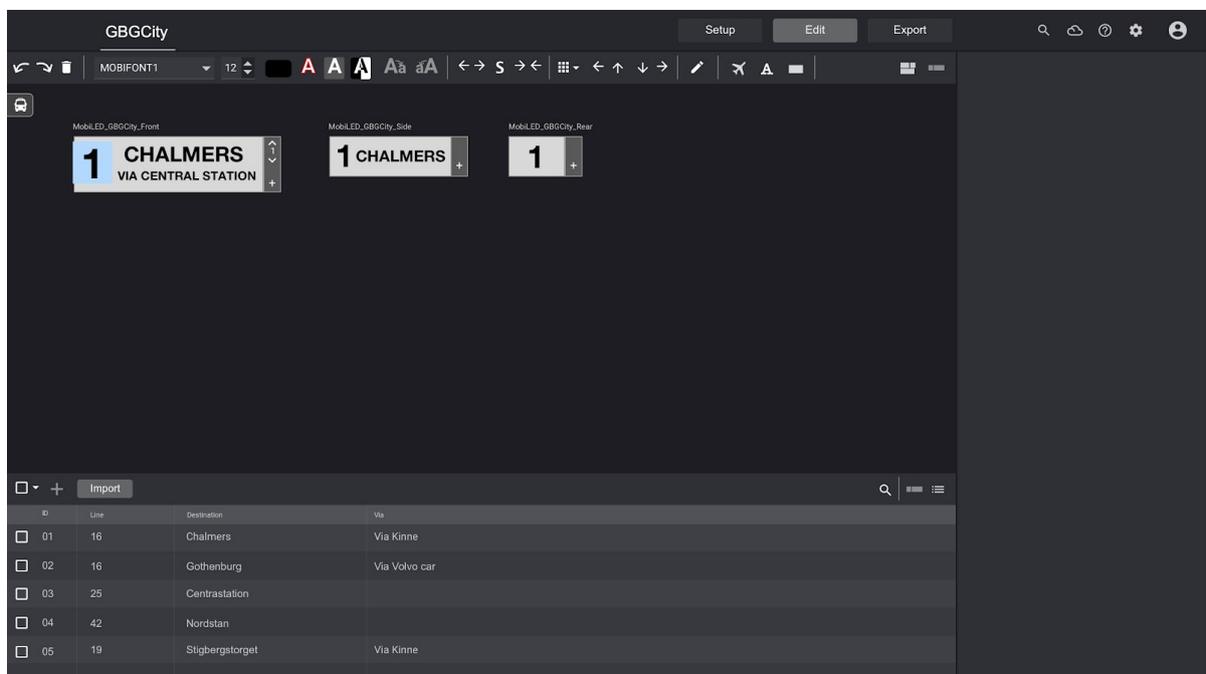
22. If multi scrolling areas were selected, then multi scroll setting sections will appear in the property window.

Scenario 6 - Sign settings

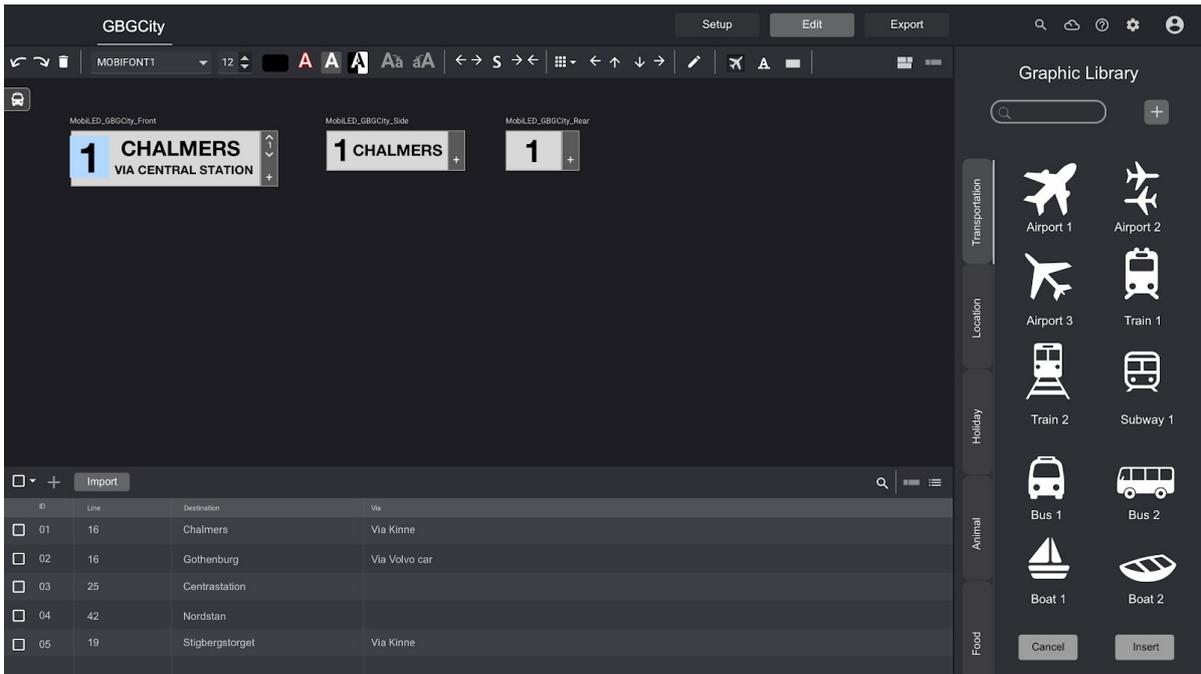


23. In the 'settings' section, the setup information about the sign are displayed and allows users to make changes.

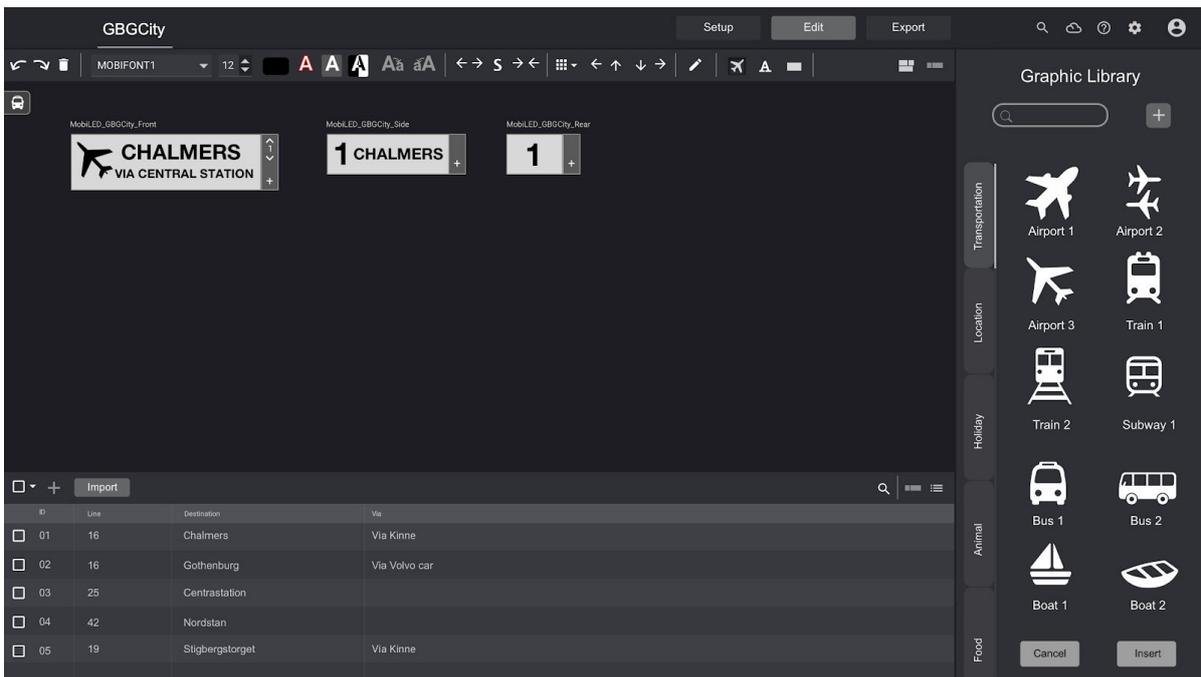
Scenario 7 - Graphic library and editor



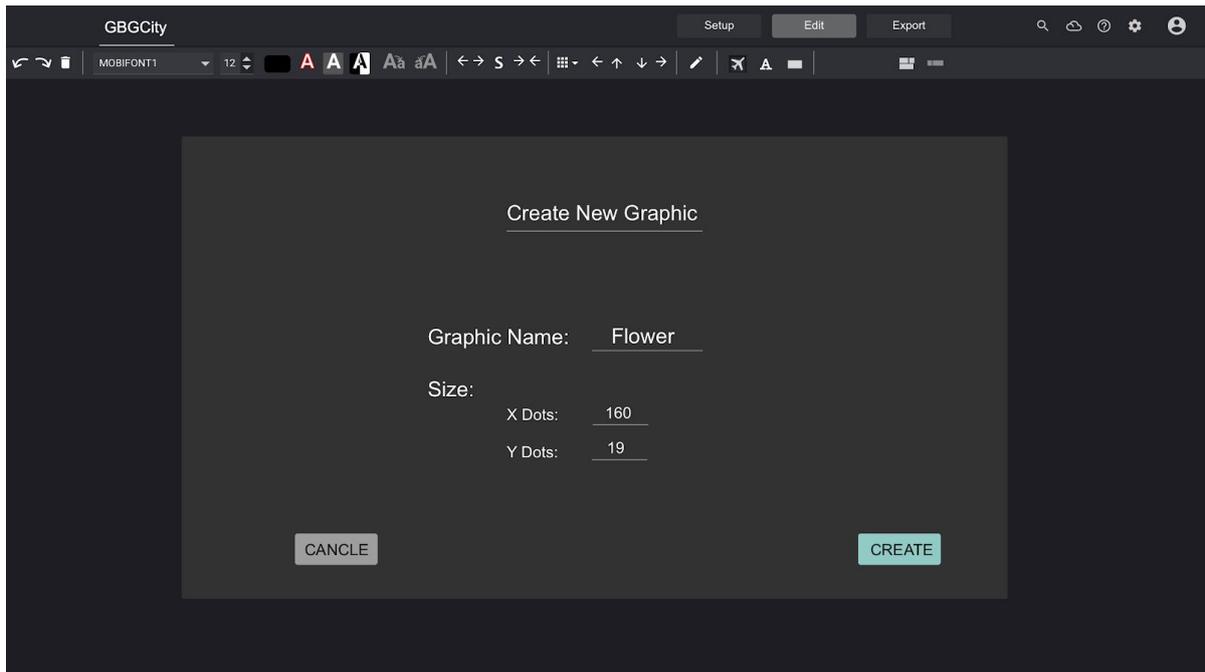
24. In order to insert graphics, users could mark the text and then click 'graph library' icon in the toolbar.



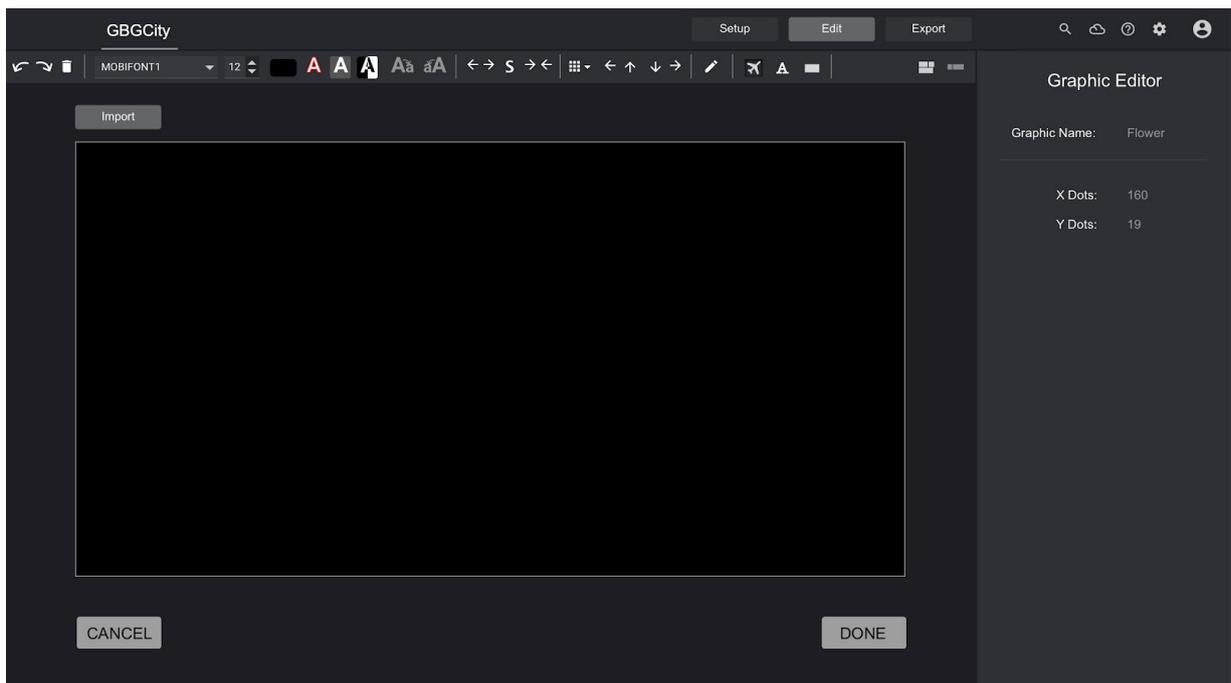
25. The graphics in the graphic library are grouped into different categories which allows users to find the graphic they need quickly. For instance, if the user wants to replace '1' with 'Airport 3' icon, then users could either drag and drop the icon to the workplace directly or click insert.



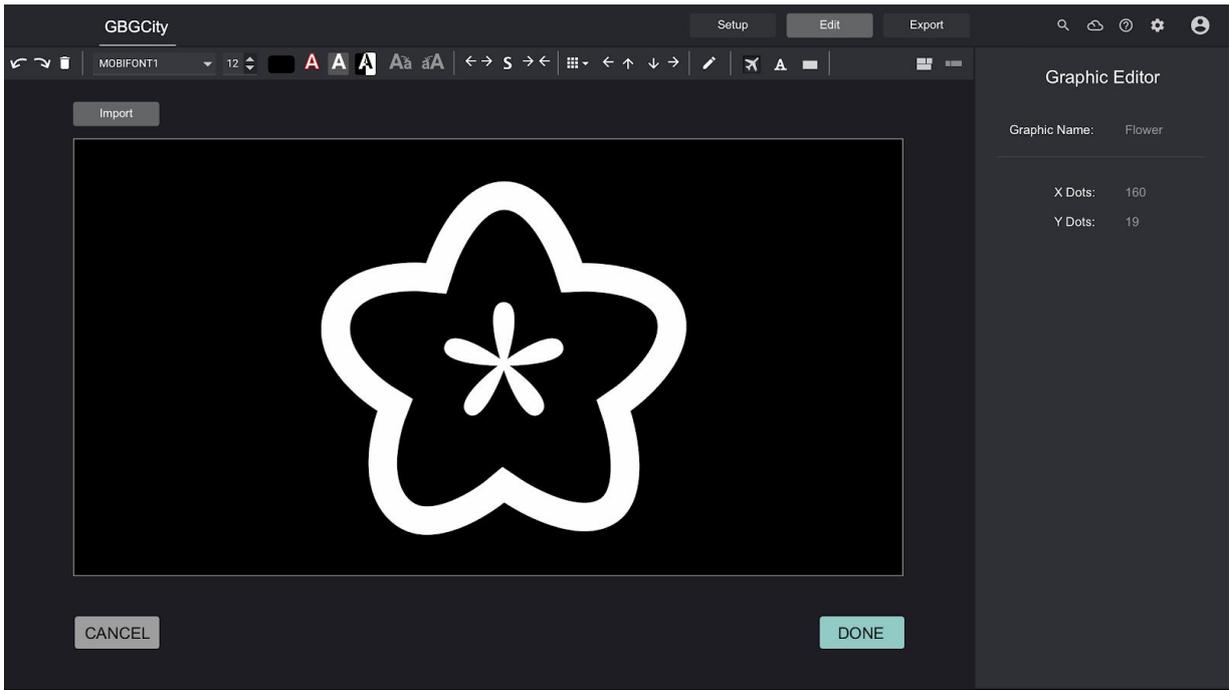
26. 'Airport 3' icon inserted successfully.



27. If users could not find a suitable graphic in the default library, then they could create new graphics by themselves according to their needs. By clicking '+', inputting graphic name and size, users could enter graphic editor and create a new graphic manually.

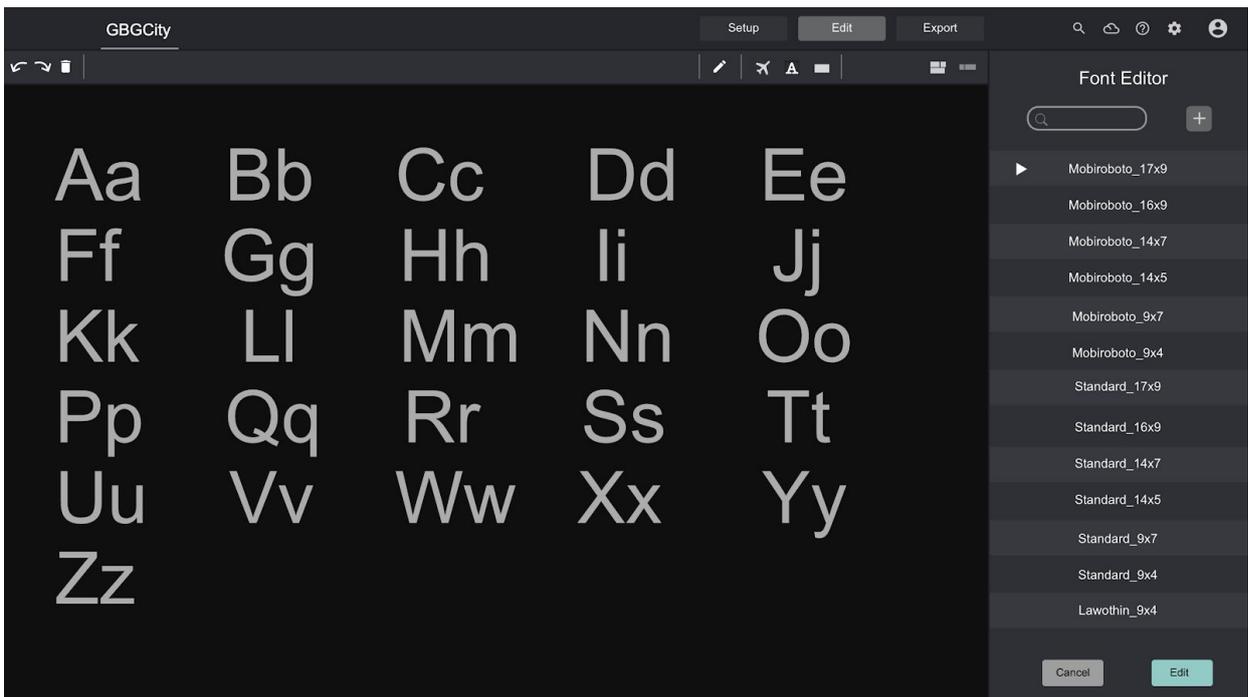


28. Here users could make pixel by pixel editing to create new graphic.

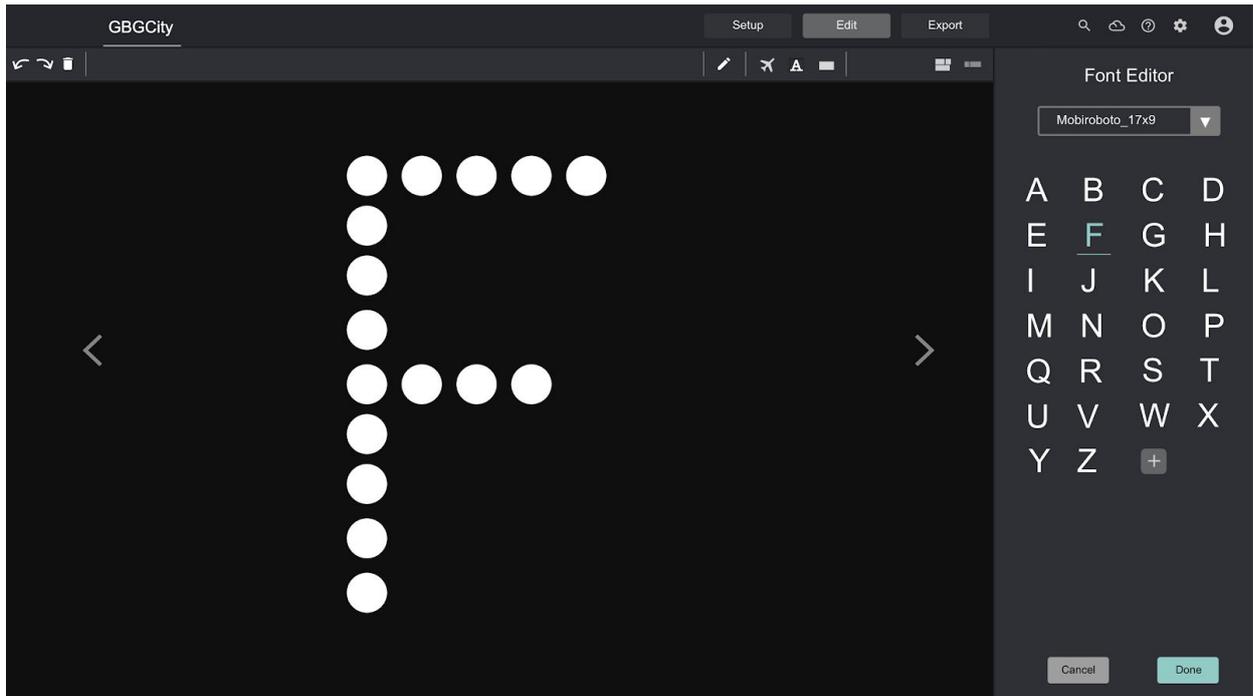


30. The new graphic 'Flower' was created successfully.

Scenario 8 - Font editor

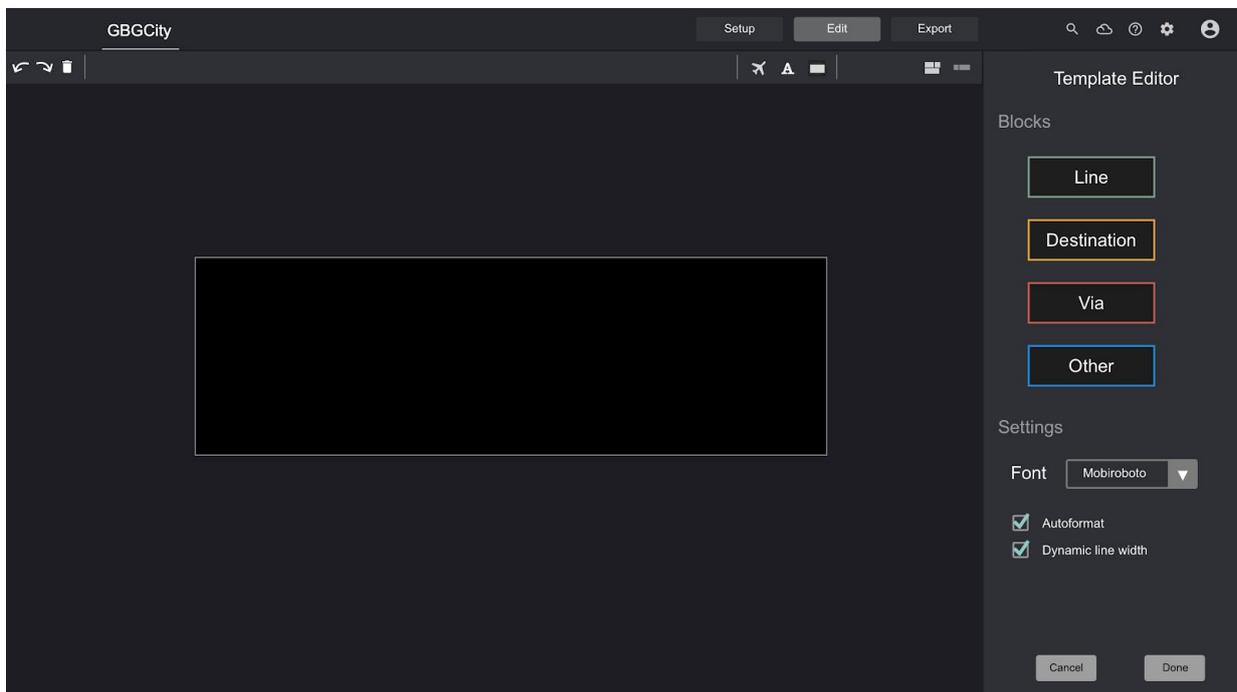


31. By clicking the 'font editor' icon in the toolbar, users could enter font editor. In most cases, users want to create a new font by making changes based on an old font. Therefore, in the font editor users could go through all fonts that are available by default, check their effects and pick one as foundation.

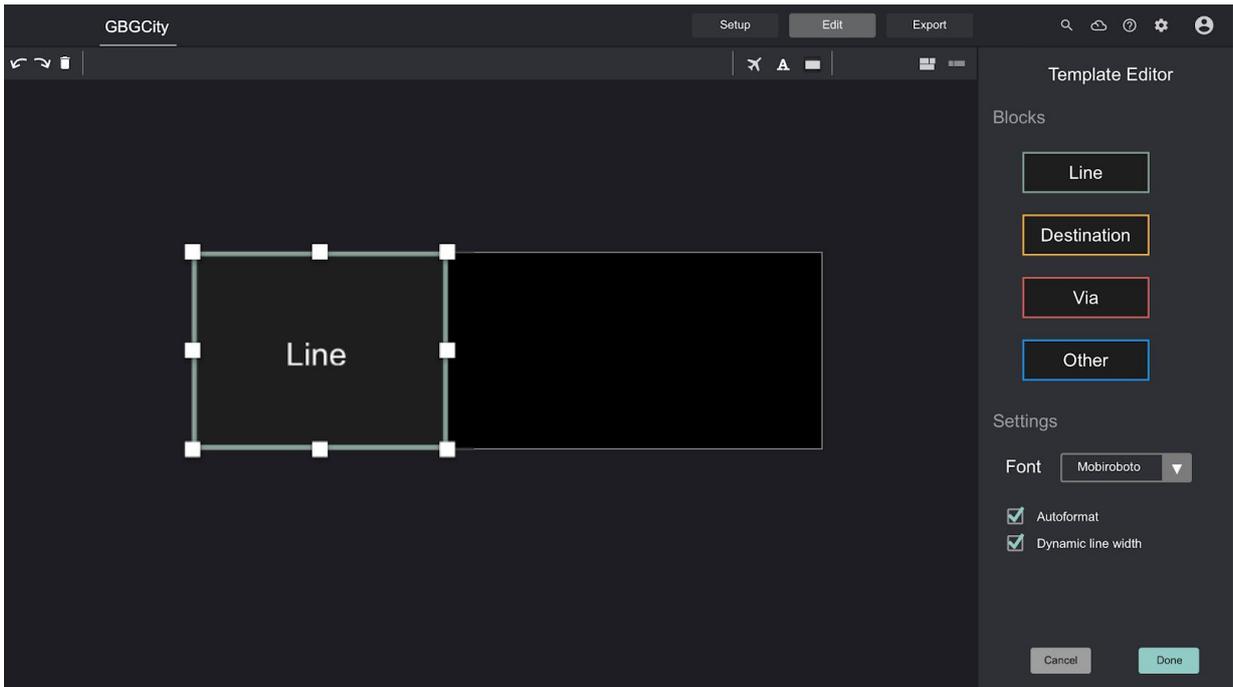


32. In the font editor, users could edit each letter pixel by pixel. Also, they could create new letters from scratch by clicking the '+' button.

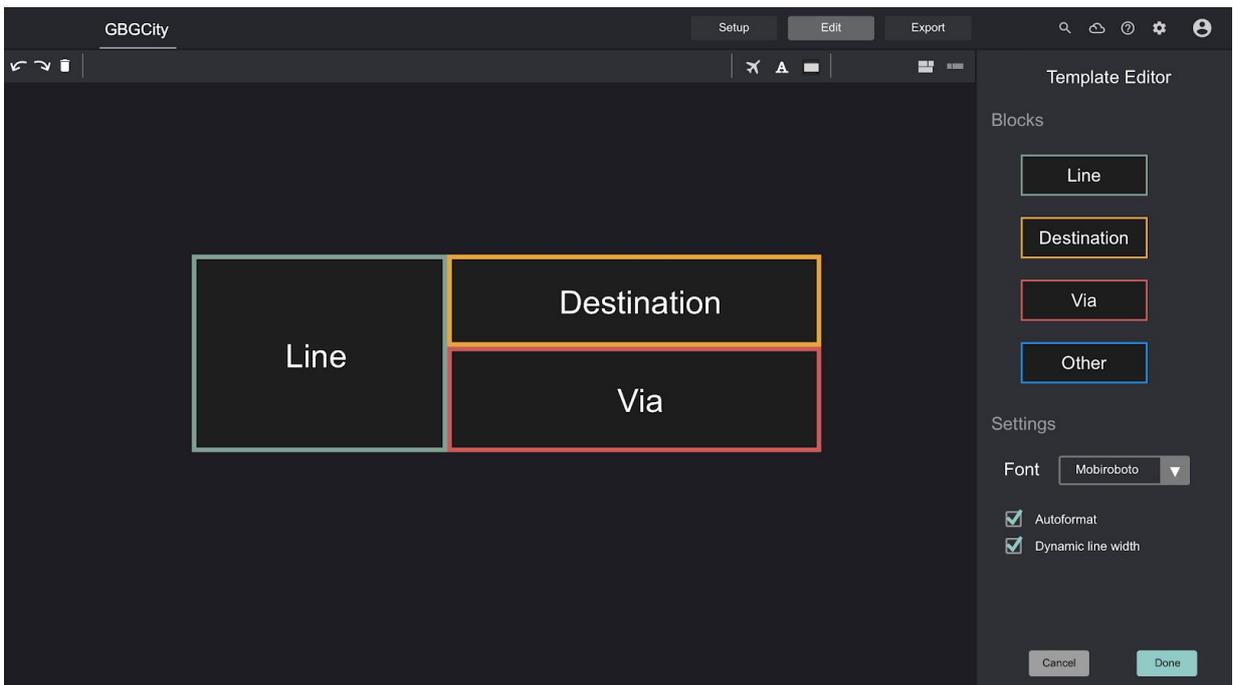
Scenario 9 - Template editor



33. By clicking the 'template editor' icon in the toolbar or the '+' button in template section, users could enter template editor where they could create customized templates.

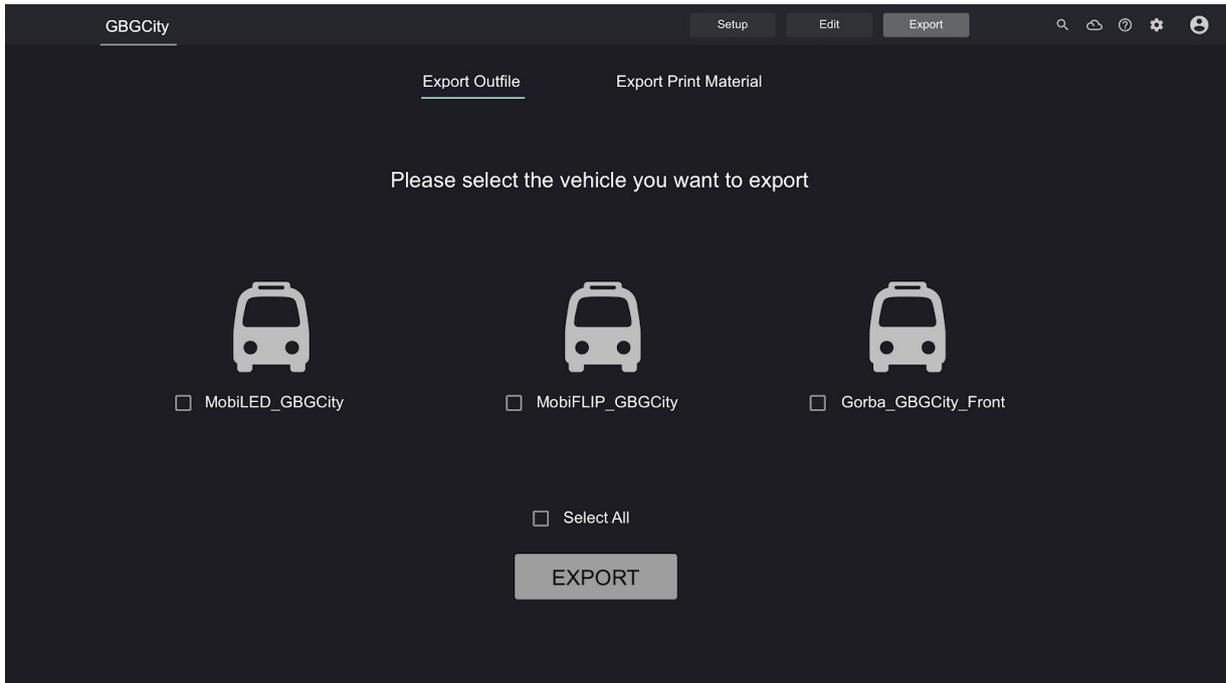


34. Four area blocks 'line', 'destination', 'via' and 'other' are available by default, users could drag and drop these blocks into workspace to create a new template. Below the blocks, users could change the settings for this template such as font adjustment, turn on/off auto-format and dynamic line width.

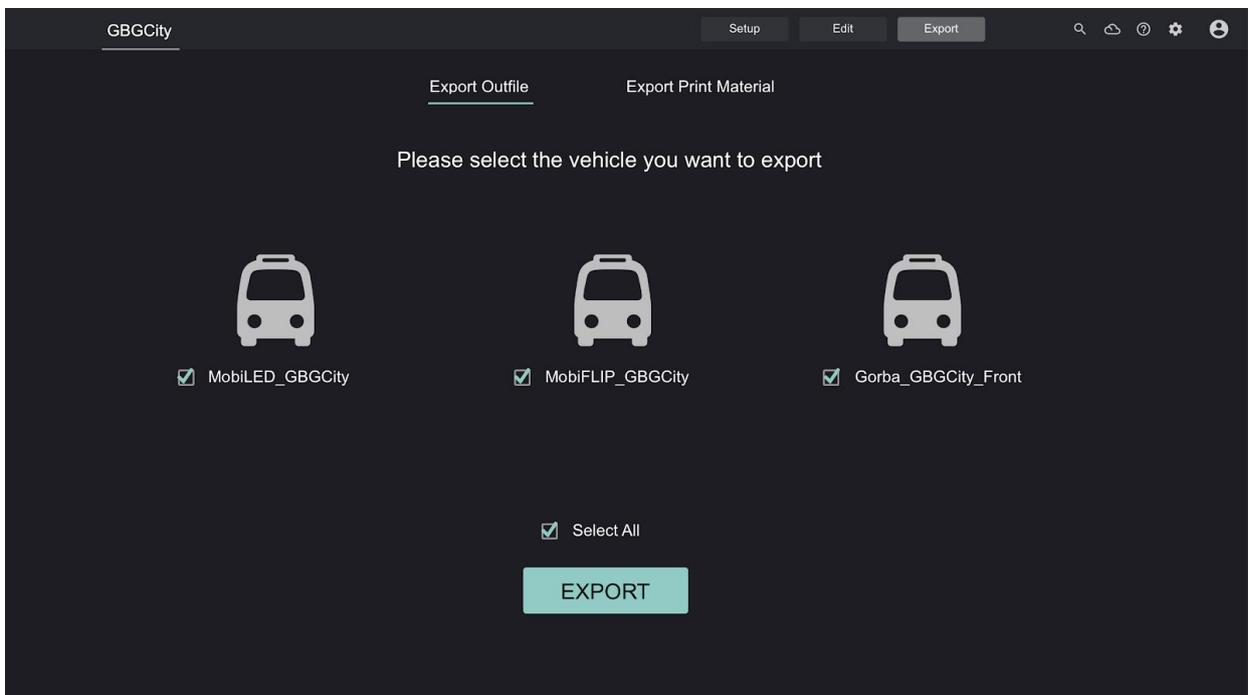


35. A new template composed of 'line', 'destination' and 'via' blocks was created.

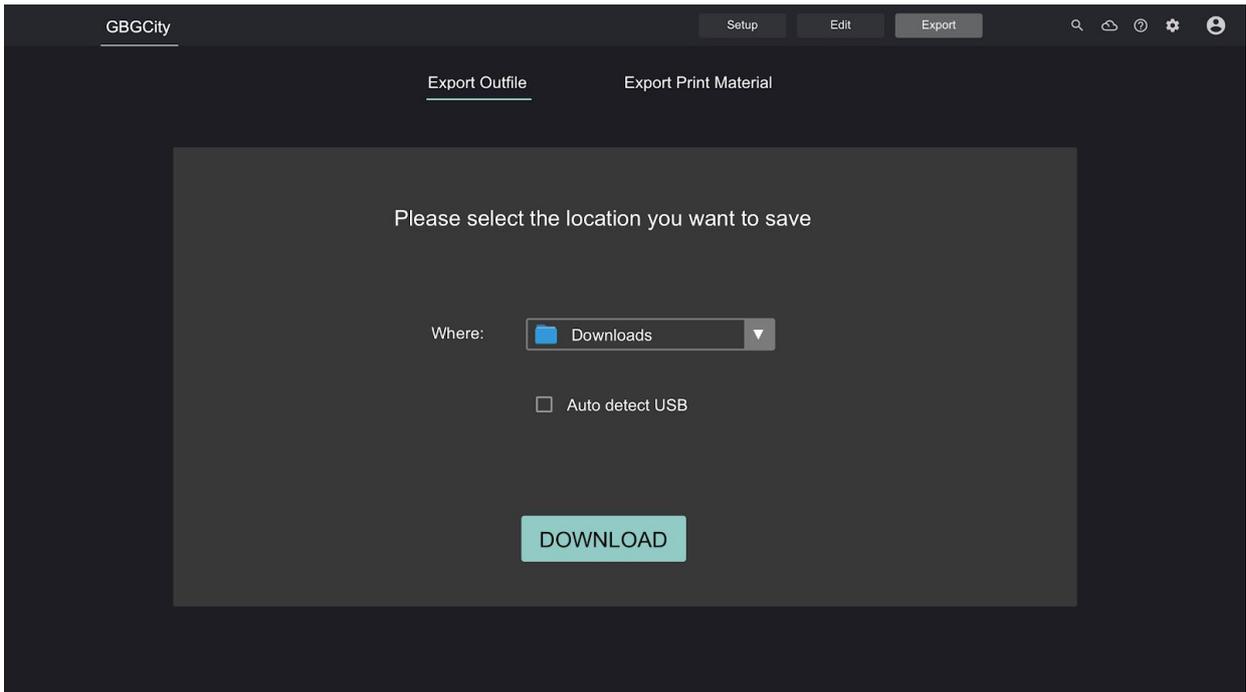
Scenario 10 - Export



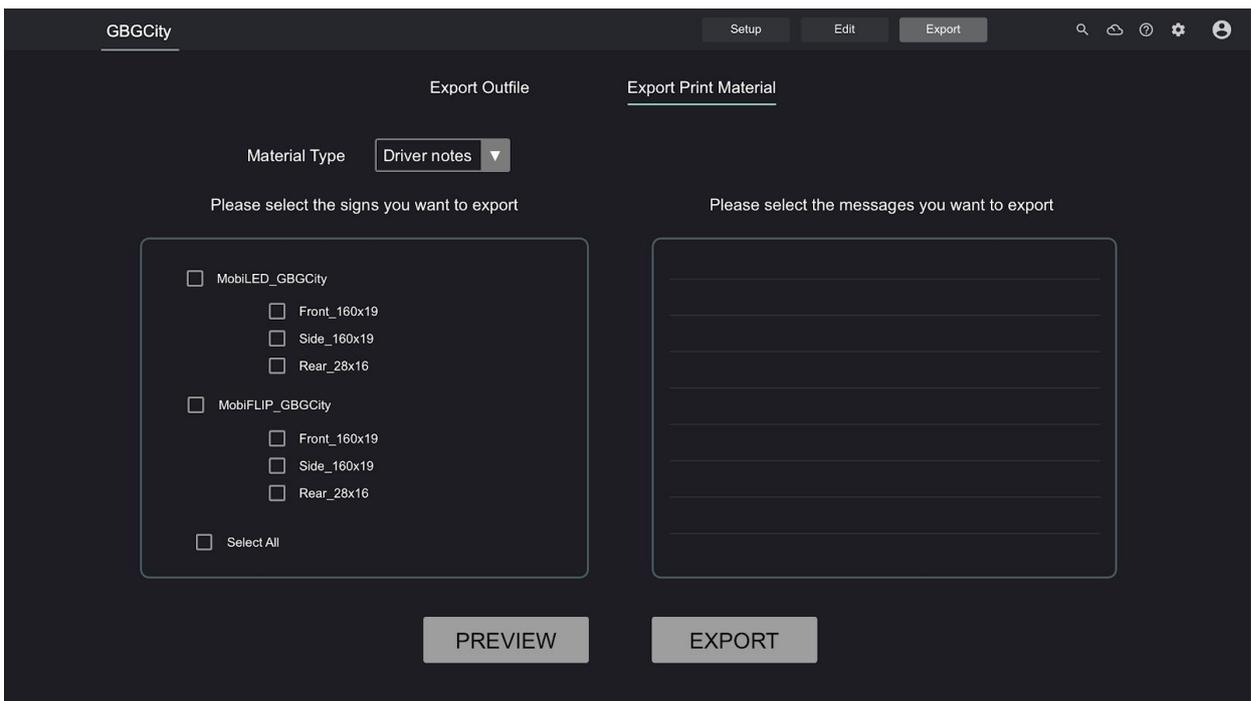
36. When the editing is finished, click 'export' in the top bar to generate outfiles. Here users were asked to select the vehicle(s) they want to generate outfile for.



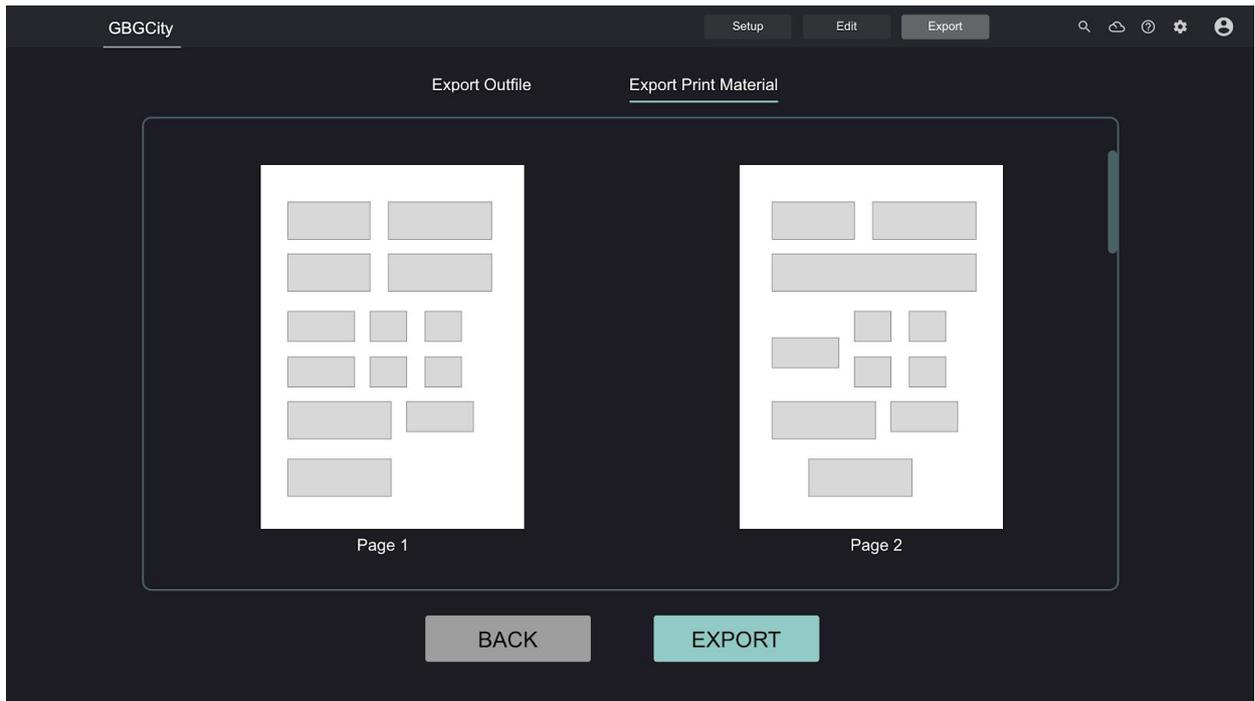
37. Tick 'Select All' to create outfile for all vehicles in this project. Then click 'export' button.



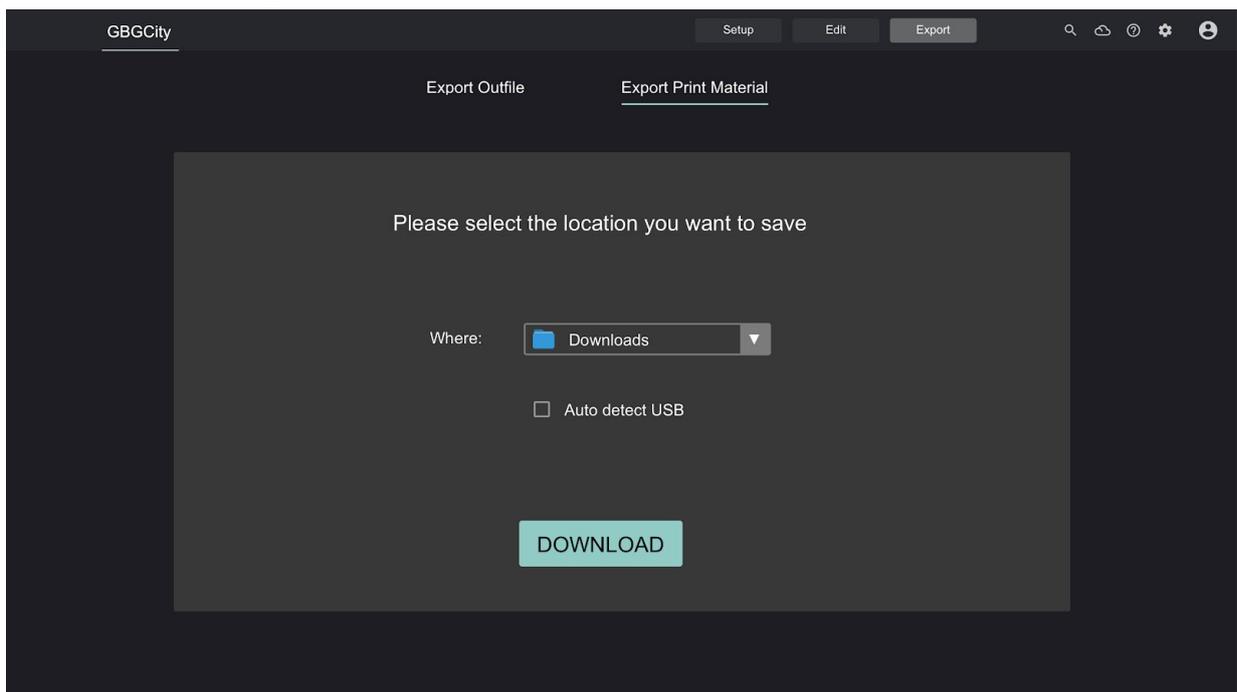
38. Define where the outfiles should be stored and click 'download'.



39. Creating print materials for administrative purpose is another important needs of software users. Therefore, in export interface, users could also export print materials by selecting different material types. Here 'Driver notes' was used as an example of print materials. Then click 'preview'.



40. Preview of 'Driver notes' was generated here which allows users to check the layout and effect. Then click 'export'.



41. Define where the document should be stored and click 'download'.

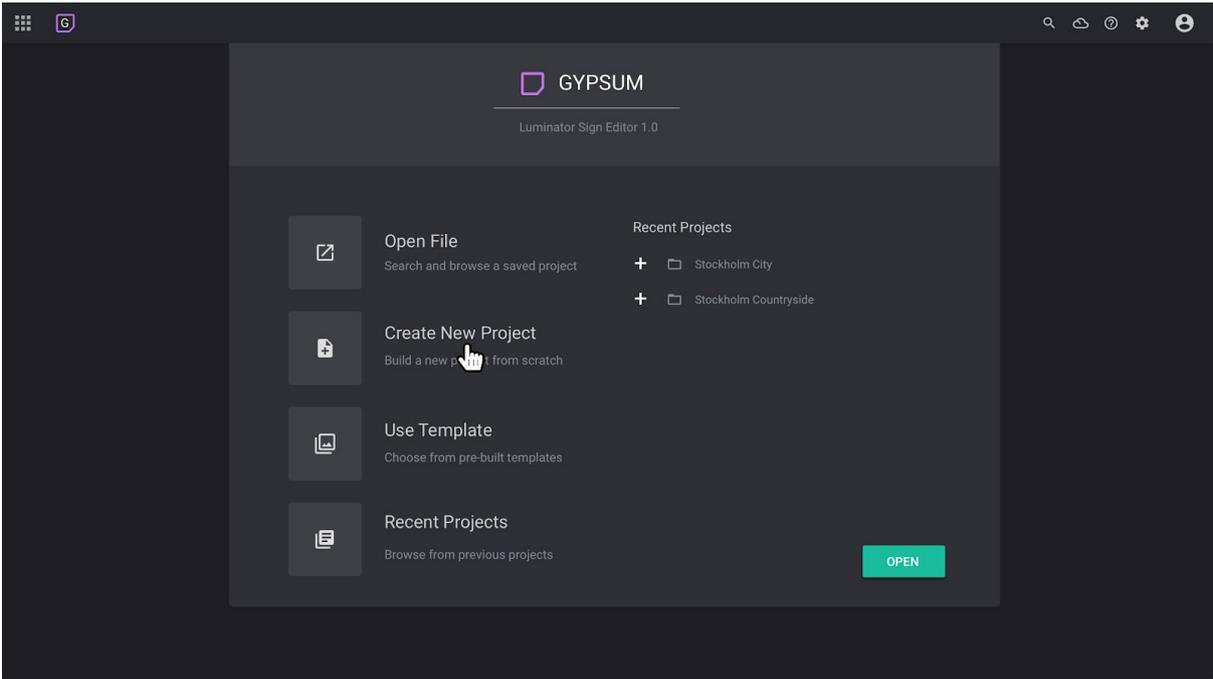
Appendix 16 - Usage scenarios of the final prototype

This section presents the detail design of Gypsum by presenting the interfaces under different usage scenarios.

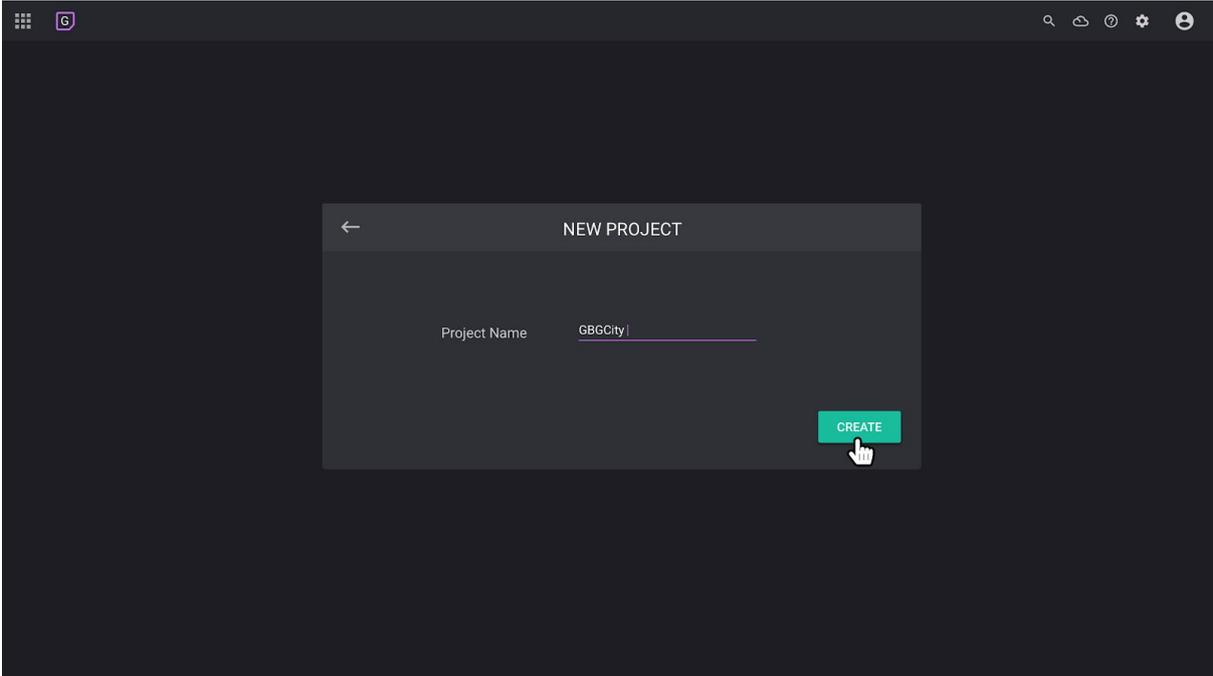
Scenario 1 - Setup a new project named 'GBGCity', add one bus and three signs into this project



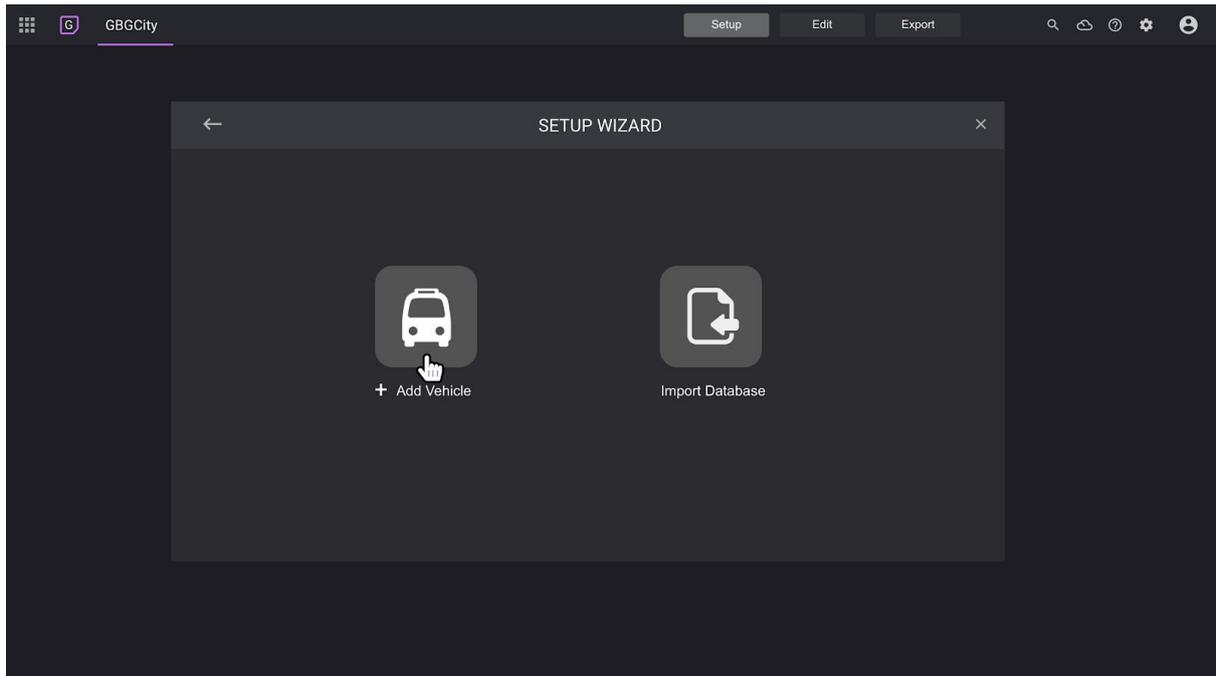
1. LTG suite overview presenting all LTG software. *Click 'GYPSUM'*



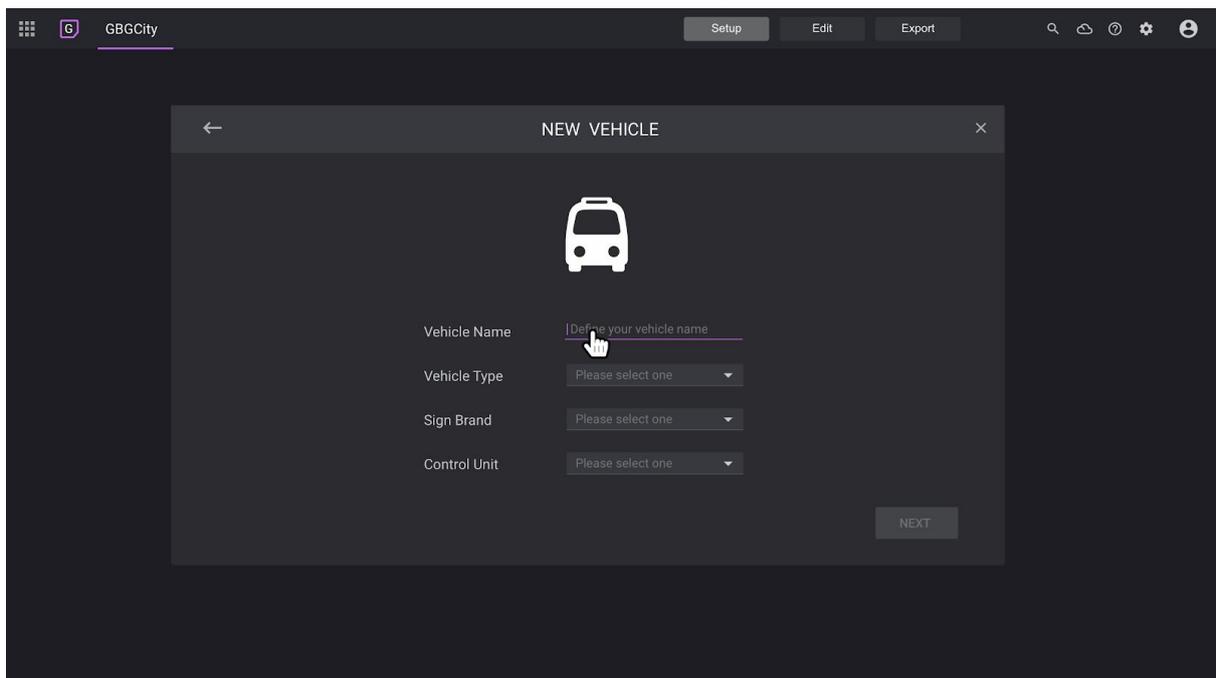
2. Project overview showing recent projects but also allows users to open file, create new project, or use template. *Click 'Create New Project'*



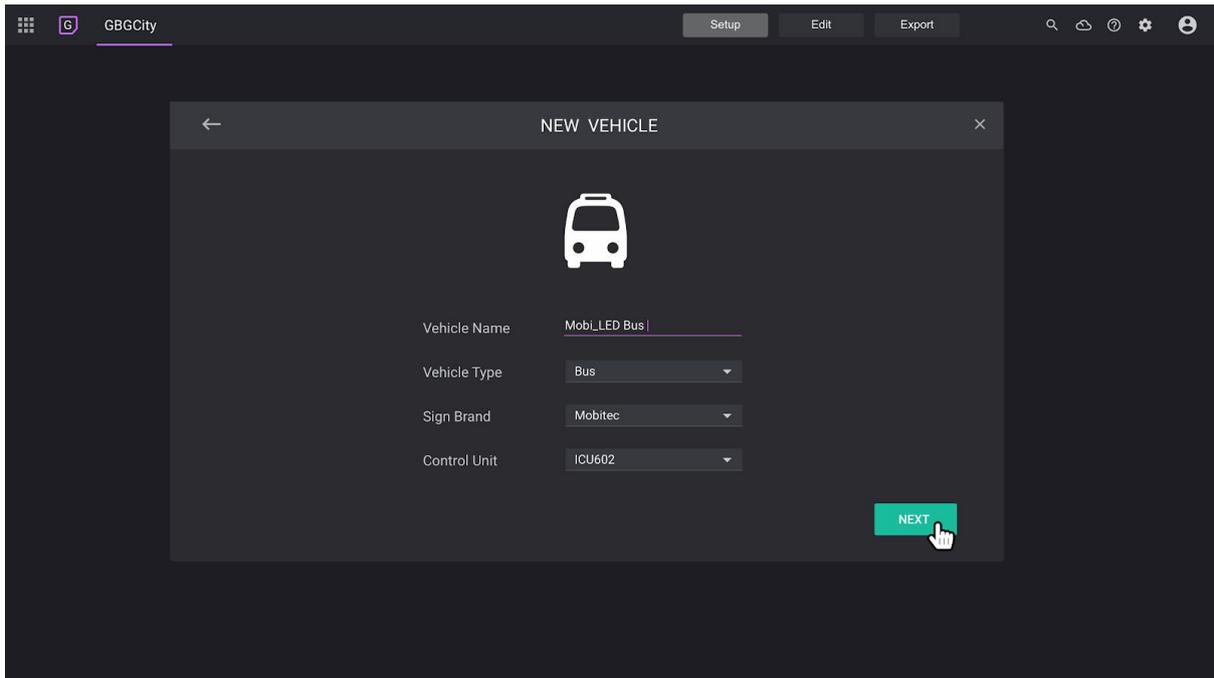
3. Input project name 'GBGCity'. Click 'create'



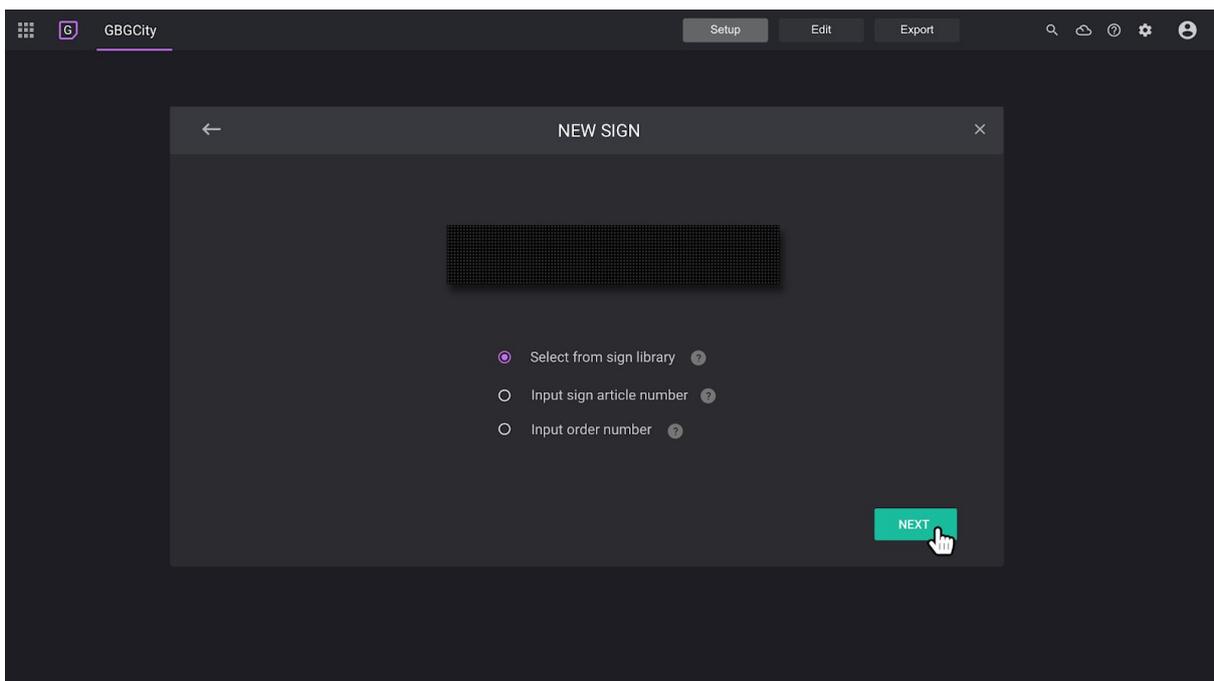
4. For users who used other LTG destination sign editors before, Gypsum supports database transfer which allows users to continue work on their old databases with the new interface. Click 'add vehicle' to create a vehicle manually



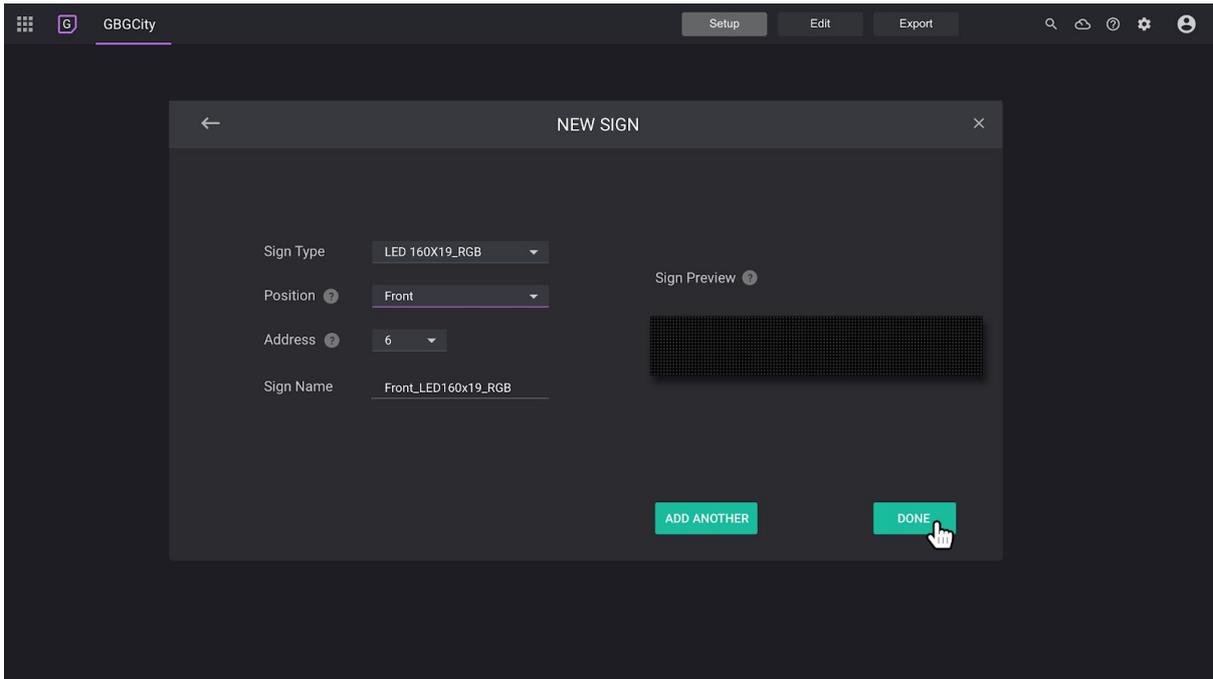
5. For users who enter the project for the first time, setup wizard appears defaultly to help users setup the project step by step. Users could also click the cross on the right corner to exit setup wizard. Input *vehicle name*, select *vehicle type*, *sign brand* and *control unit*.



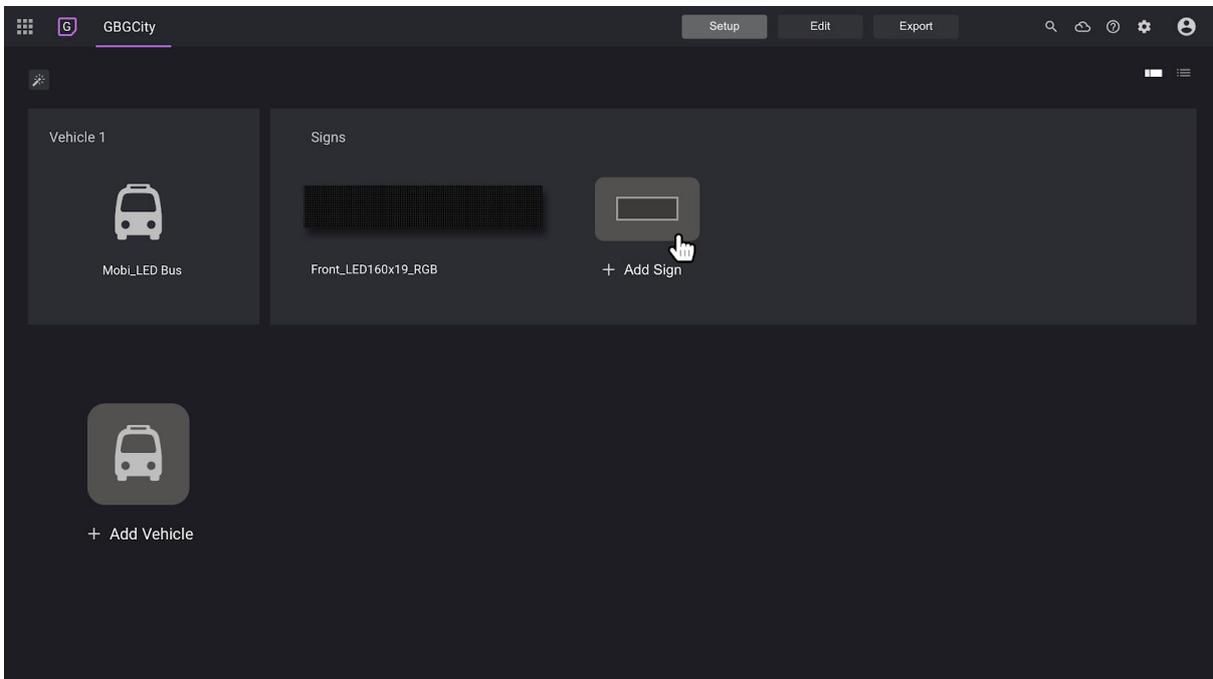
6. Vehicle data are filled in. *Click 'next'*



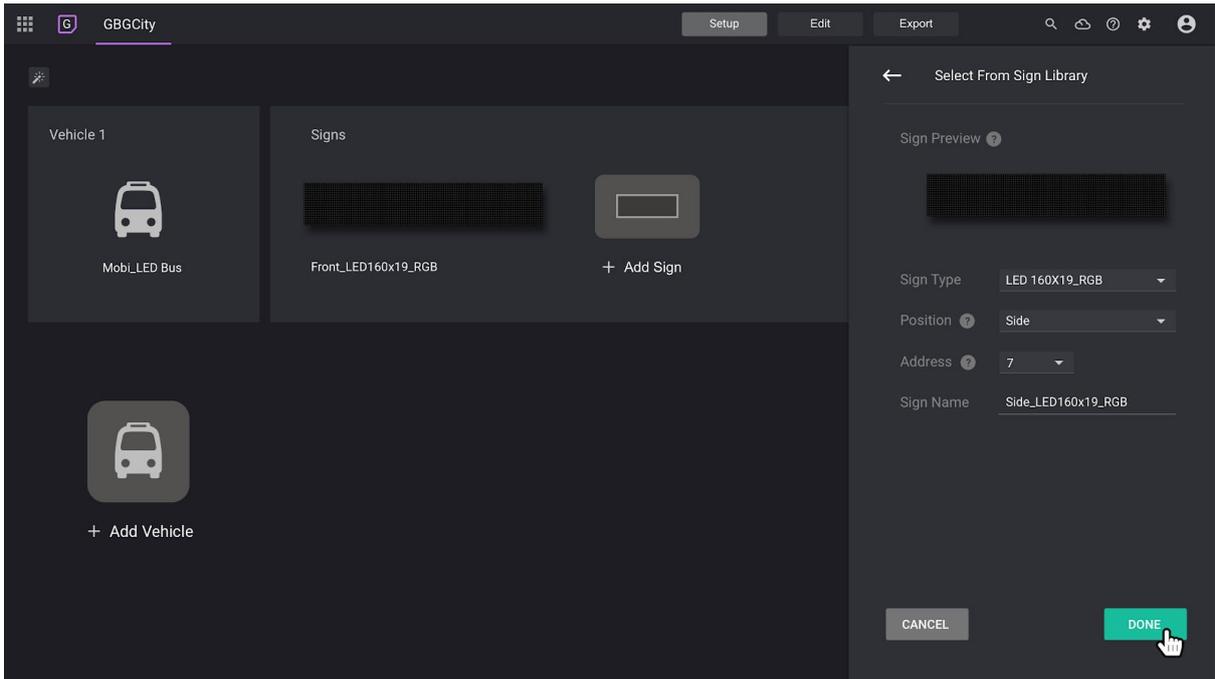
7. Gypsum offer three ways to create new signs, users could select the way they prefer according to the information they have. *Select 'select from sign library' and click 'next'*



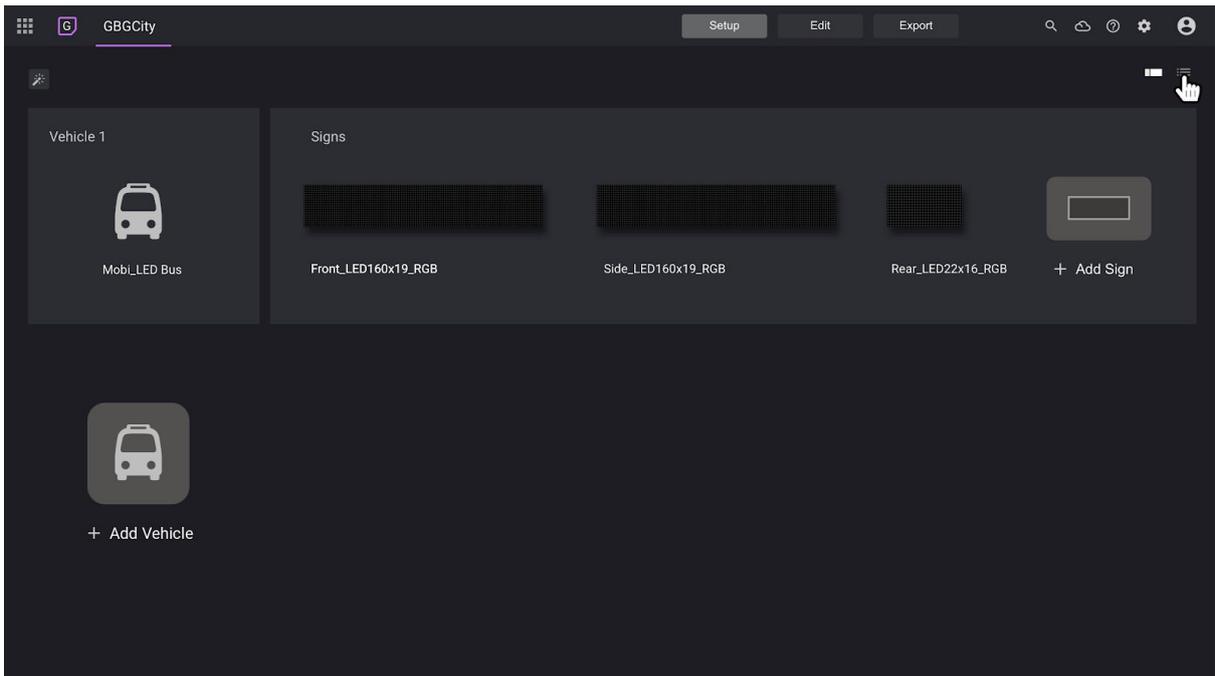
8. In sign setup interface, users are asked to select sign type and sign position, then based on users' selections, sign address and sign name will be generated automatically but still allow users to change them manually. Now sign data are filled in. *Click 'done' to exit setup wizard and enter setup overview*



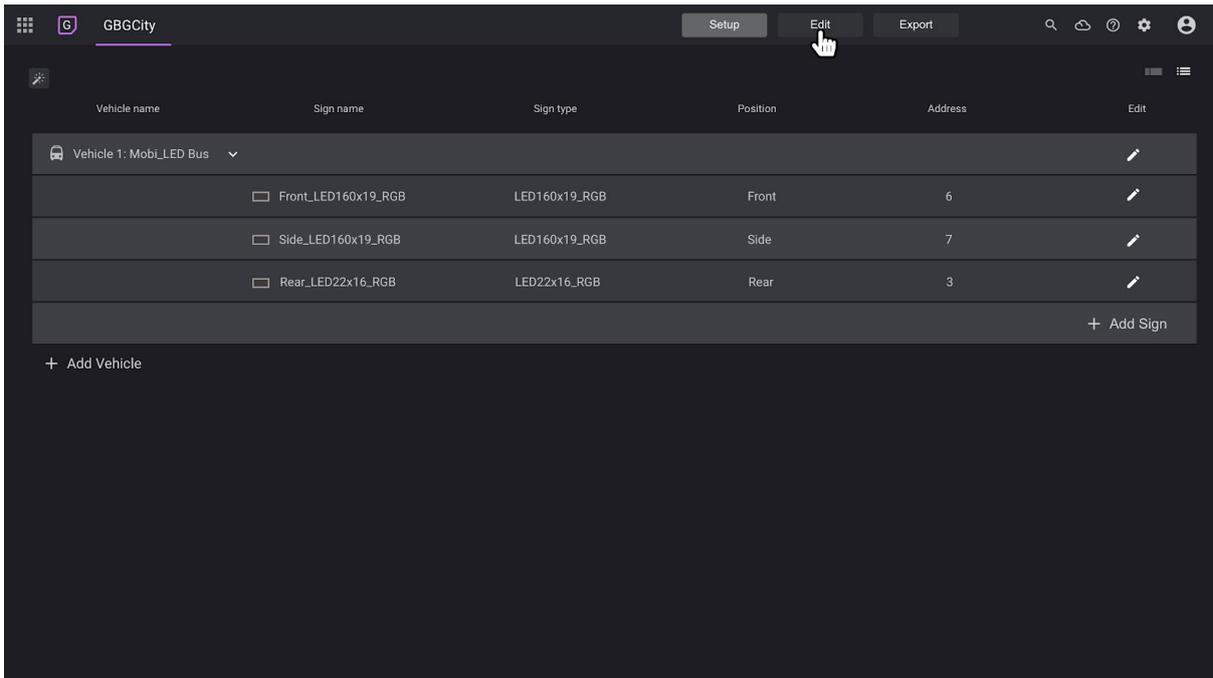
9. In the setup overview interface, all of the created vehicles and signs are presented here. *Click on 'add sign' button to continue add sign in the normal setup mode*



10. In the normal sign setup interface, following the same procedure input configuration data for side sign and rear sign. *Click on the 'done' button*

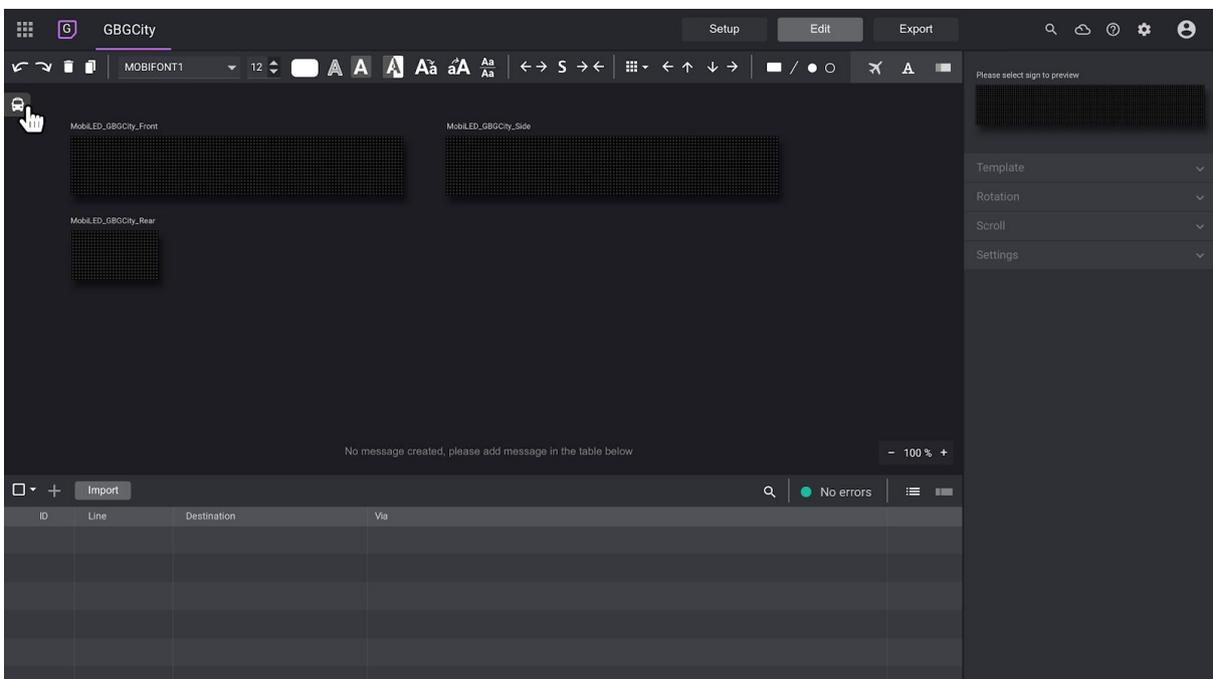


11. The vehicle and all three signs are created, users could also switch to the table view to get a detailed overview of the data. *Click on the table icon*

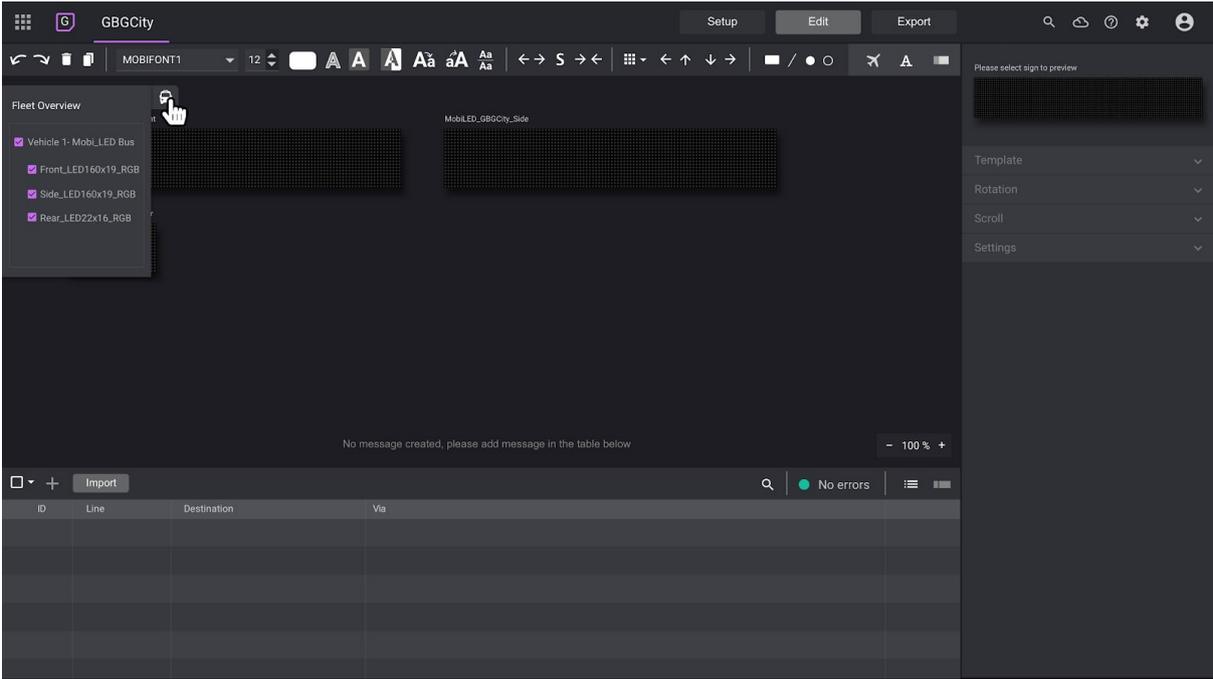


12. Table view is designed for intermediate and expert users who want an overview of the detailed setting information about the vehicle and signs for purposes like administration. Now the setup for this project is finished. *Click on the edit button in the top bar to enter edit section*

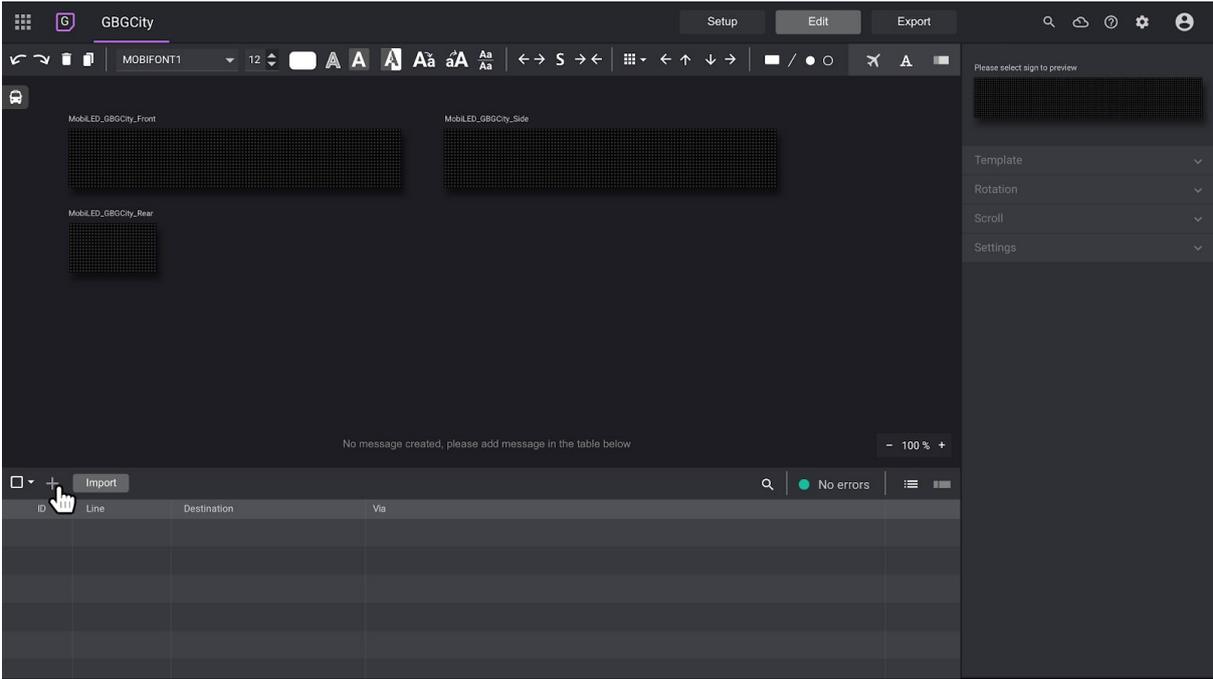
Scenario 2 - Add message '18 Skälltorpsvägen via Backatorp' manually



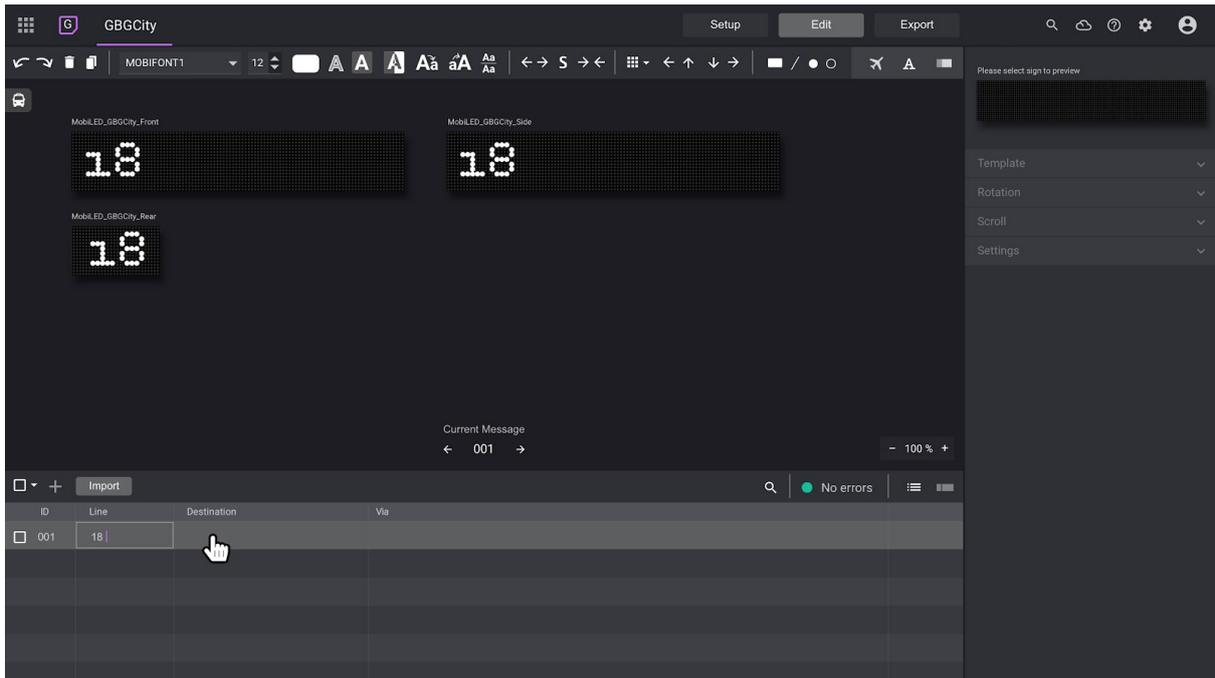
1. This is the blank edit interface when no messages are created. *Click or pull the bus icon to unfold fleet overview*



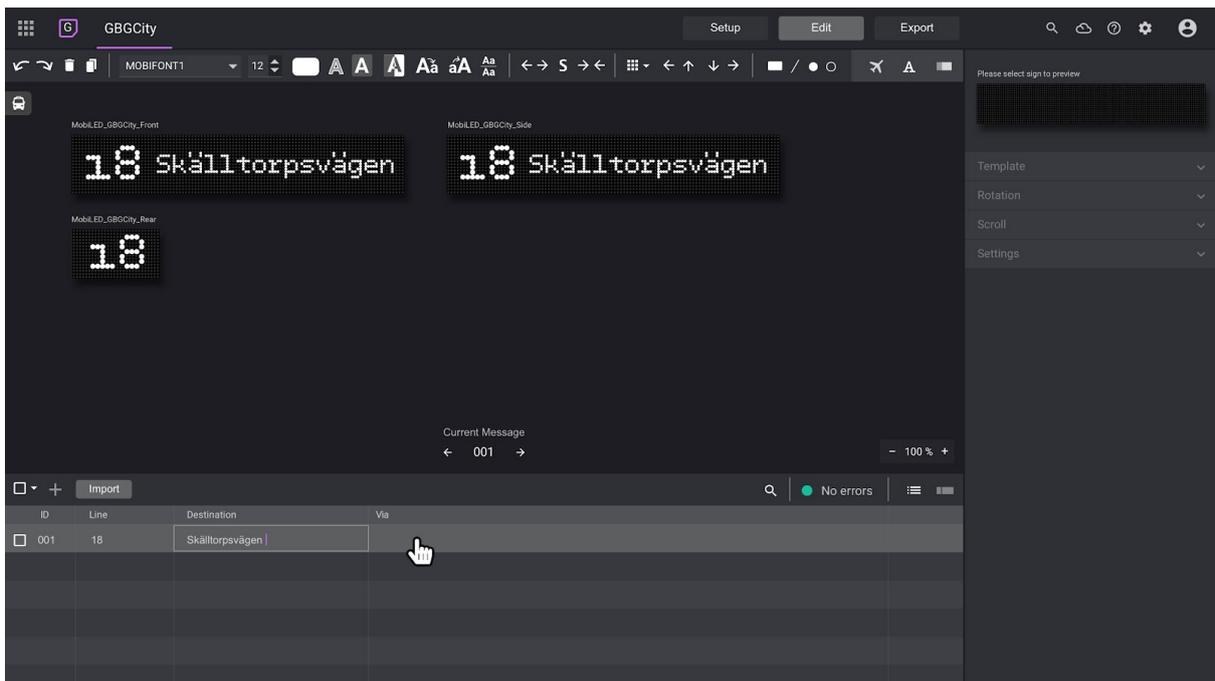
2. In fleet overview section, users could tick or untick vehicles and signs to control what to be displayed in the work area. *Click or pull the but icon to fold fleet overview*



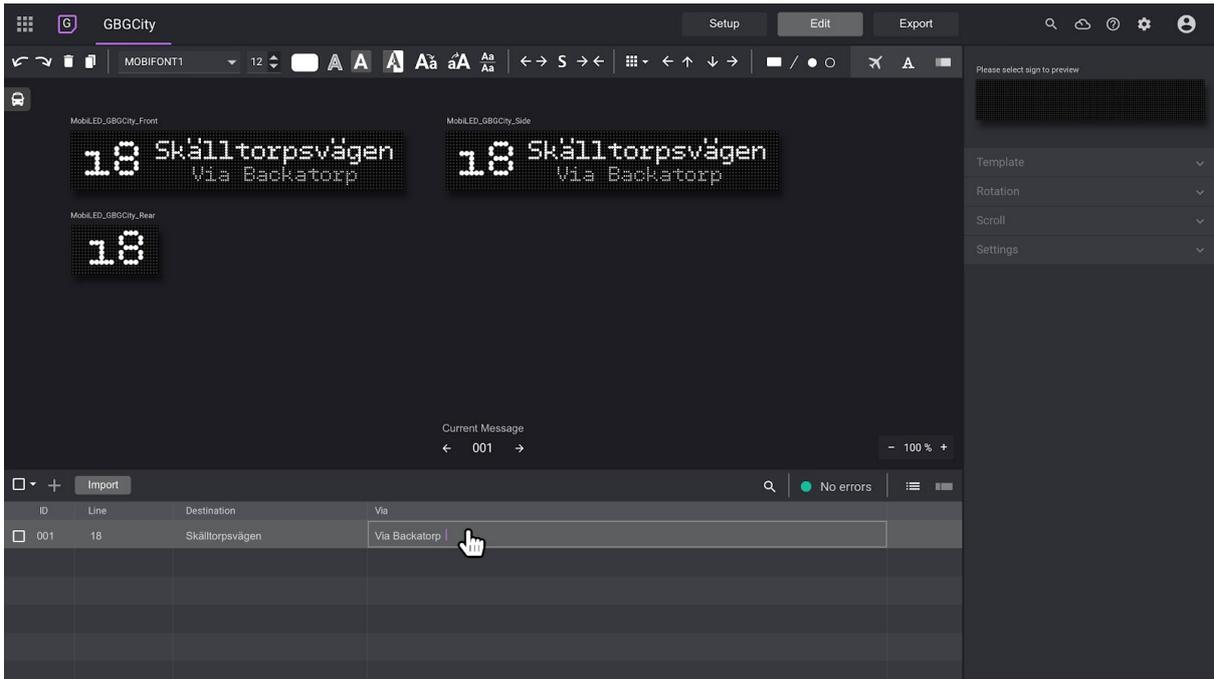
3. *Click on the '+' icon to create a new message*



4. Input 18 in line column and move to the next destination block

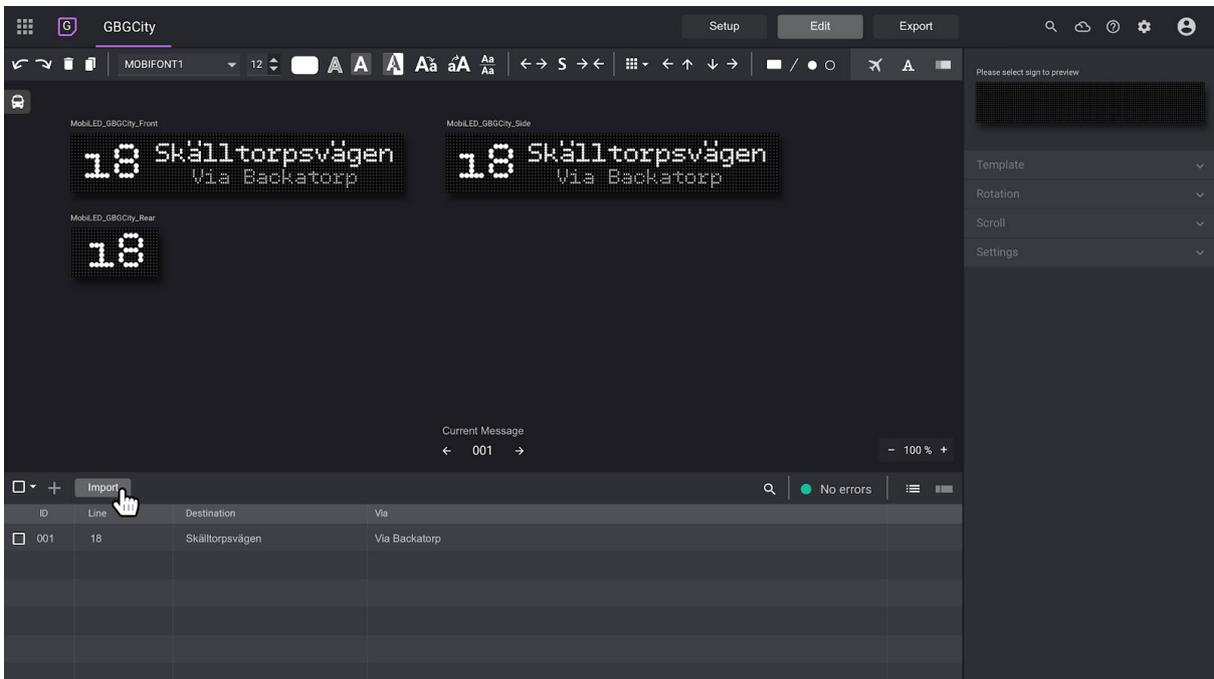


5. Input Skälltorpsvägen in destination column and move to the next via block

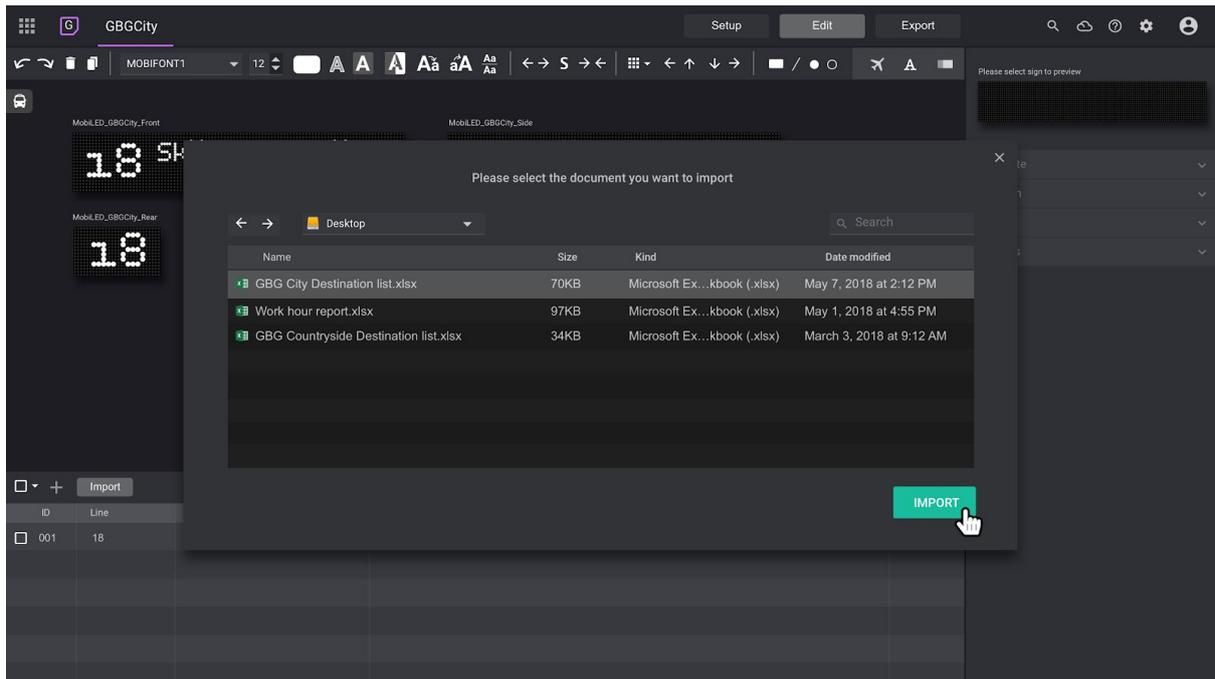


6. Input Via Backatorp in via column and now the message is created successfully

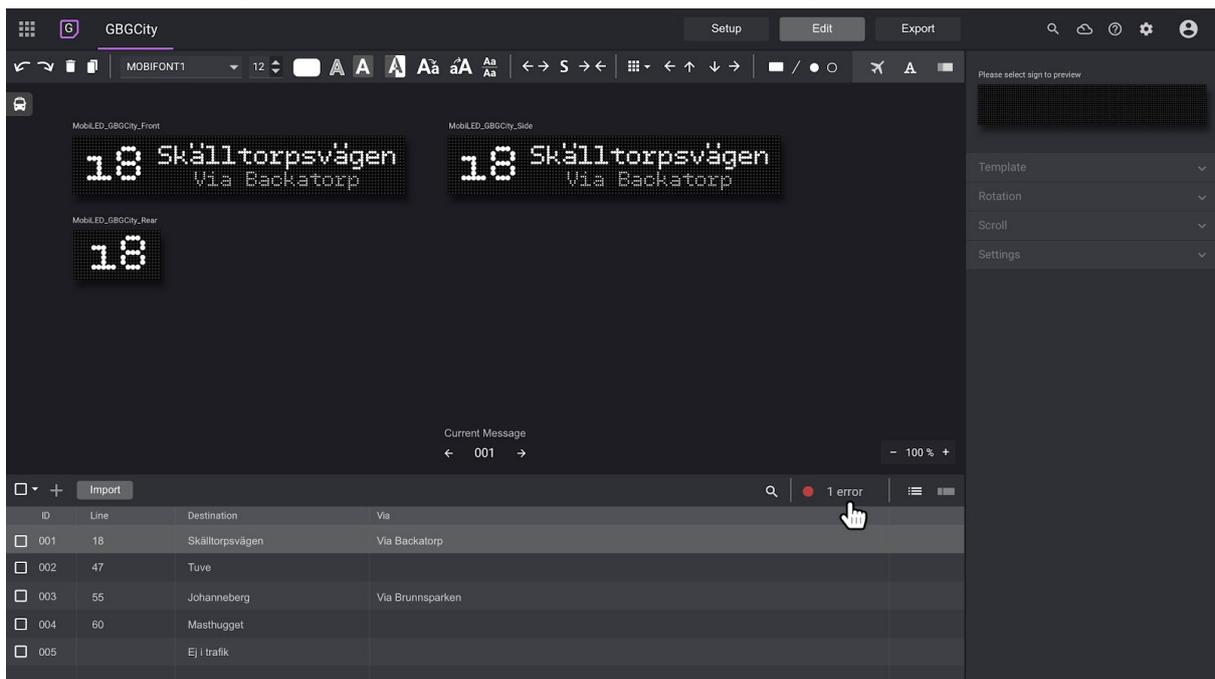
Scenario 3 - Import destinations from Excel sheet and fix error report



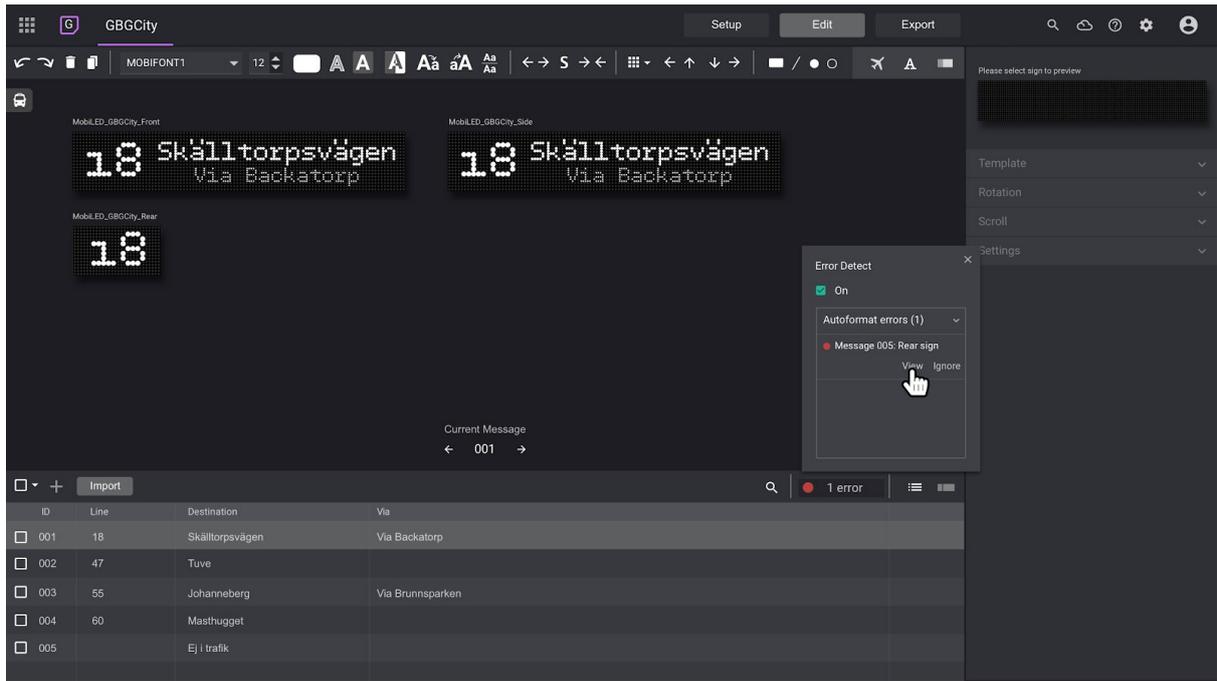
1. Click 'import' to open import window



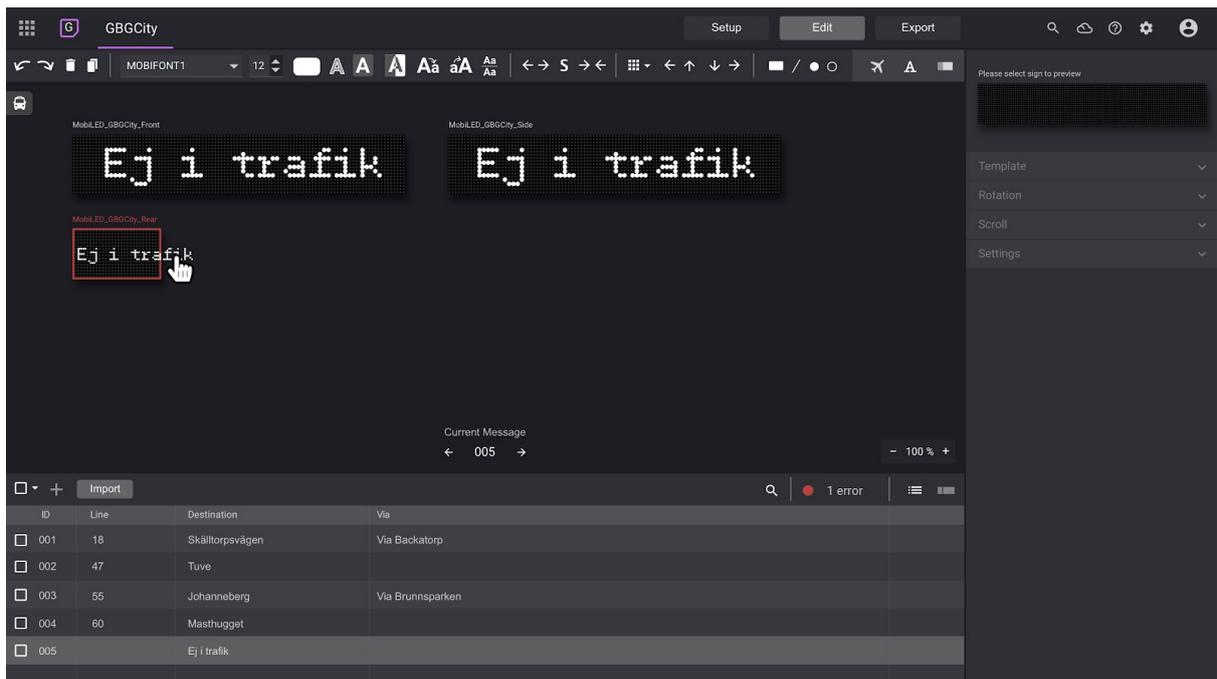
2. Select 'GBG City Destination list' document and click on 'import'



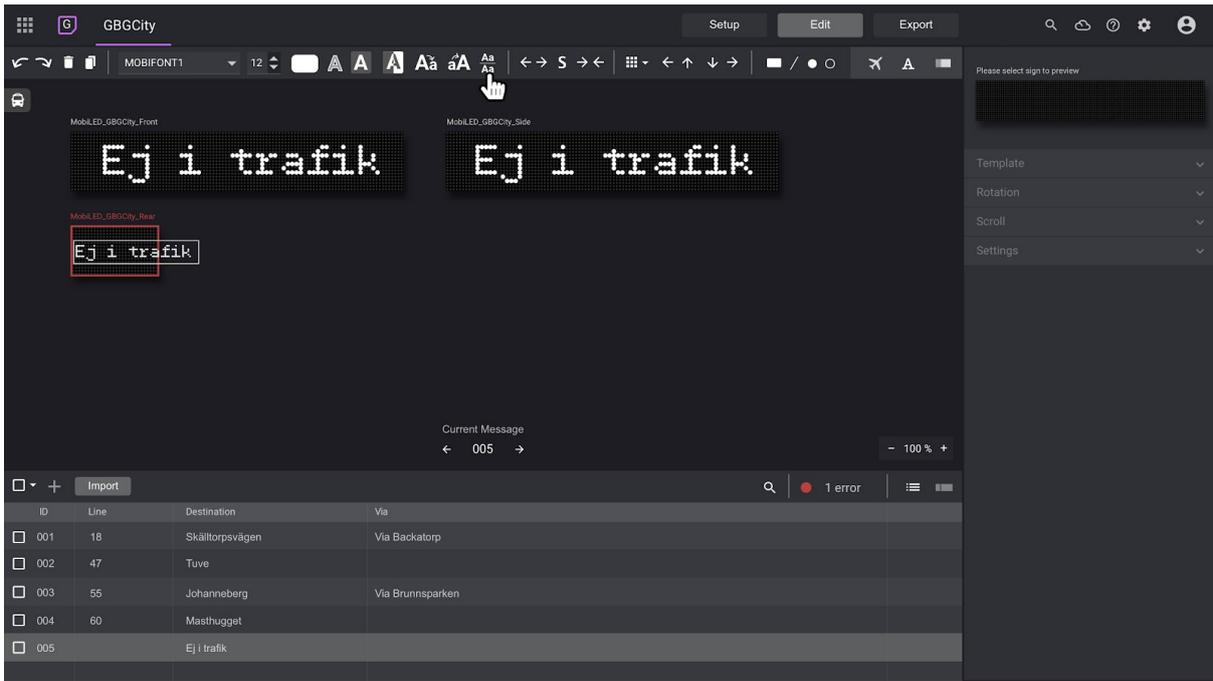
3. The document is imported successfully, however, the error report indicates there is one error. Click error report to check what is wrong



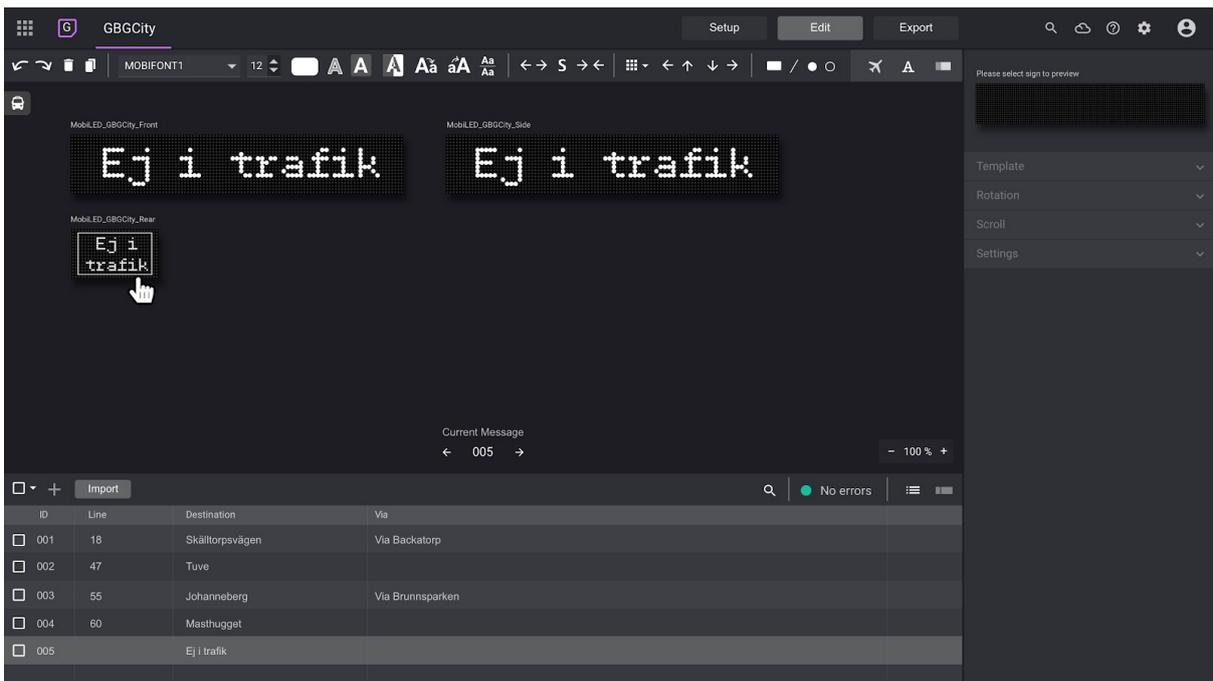
4. The error report shows there is one auto-formatting error with the rear sign in message 005. Click on 'view' to go to that message



5. The text block 'Ej i trafik' is too long to fix the sign. Now click on this block to mark it

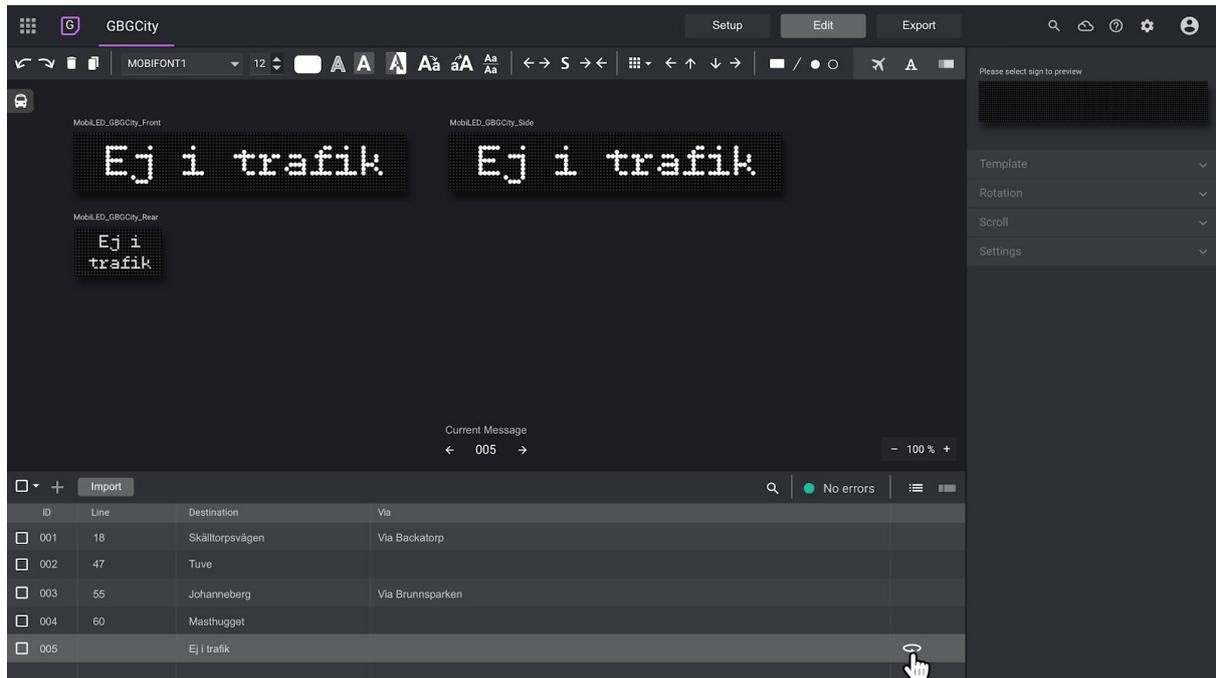


6. The text block is marked, now you want to divide this block into two line to fit the sign. Click on the double stack tool in the toolbar to achieve this

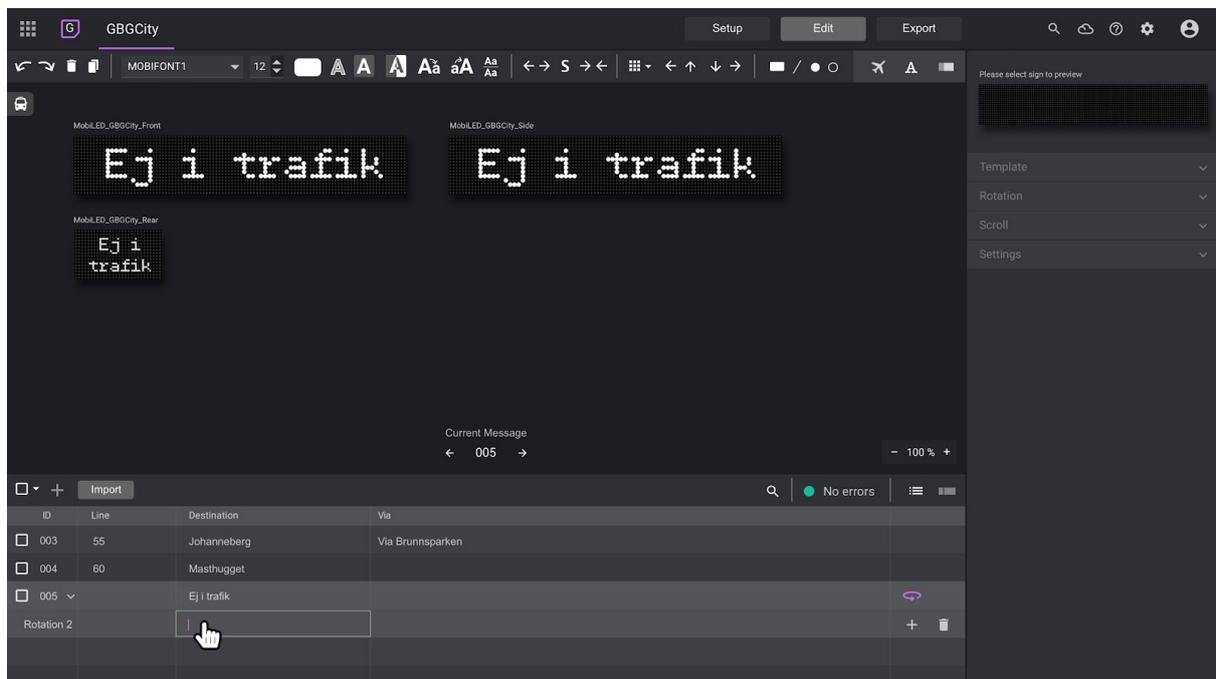


7. Now the error is fixed.

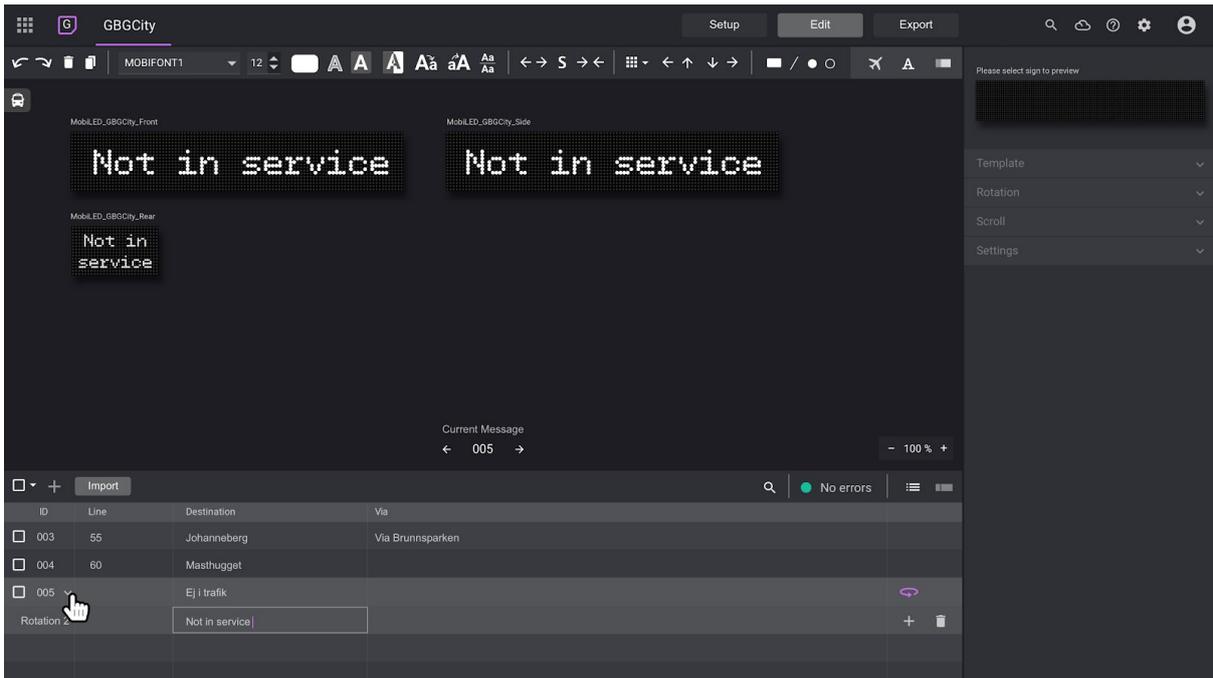
Scenario 4 - Add a rotation 'Not in service' to message 005



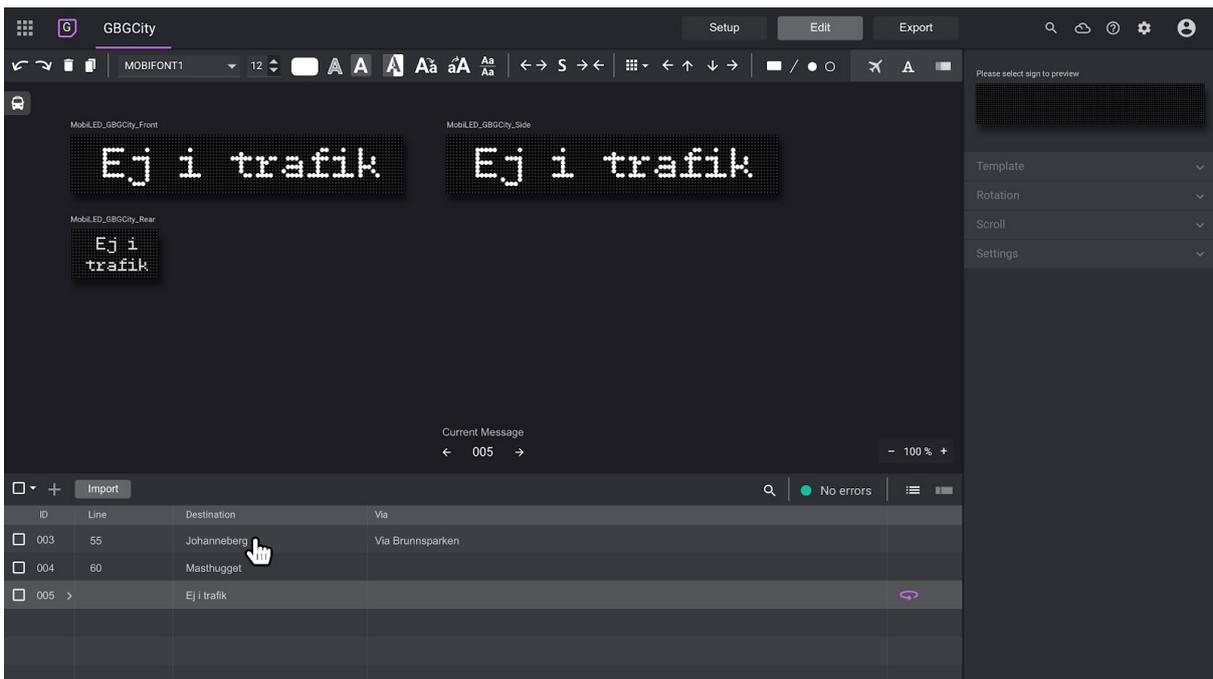
1. Hover the pointer on message 005, and the rotation icon appear according. *Click on the this icon to active it and create rotation for the entire message*



2. The 'Rotation 2' row appears. *Click on the destination block to input text*

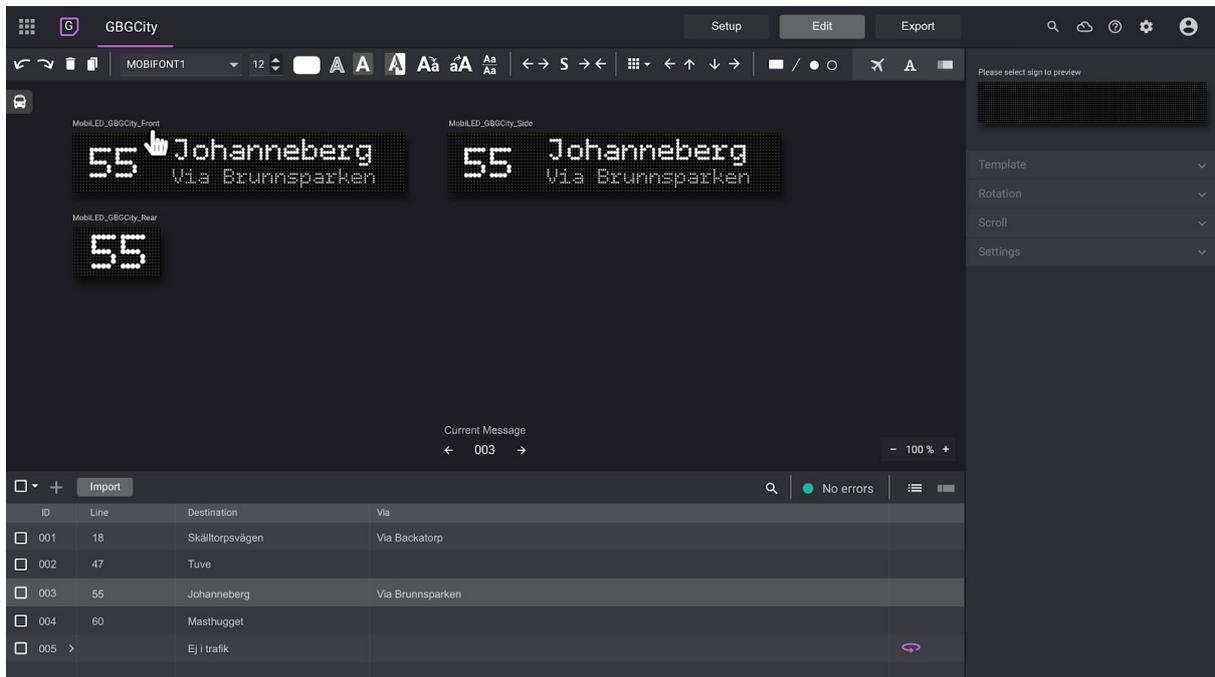


3. Input 'Not in service' in destination block, now the rotation editing is finished. *Click on the fold icon to fold or unfold rotations*

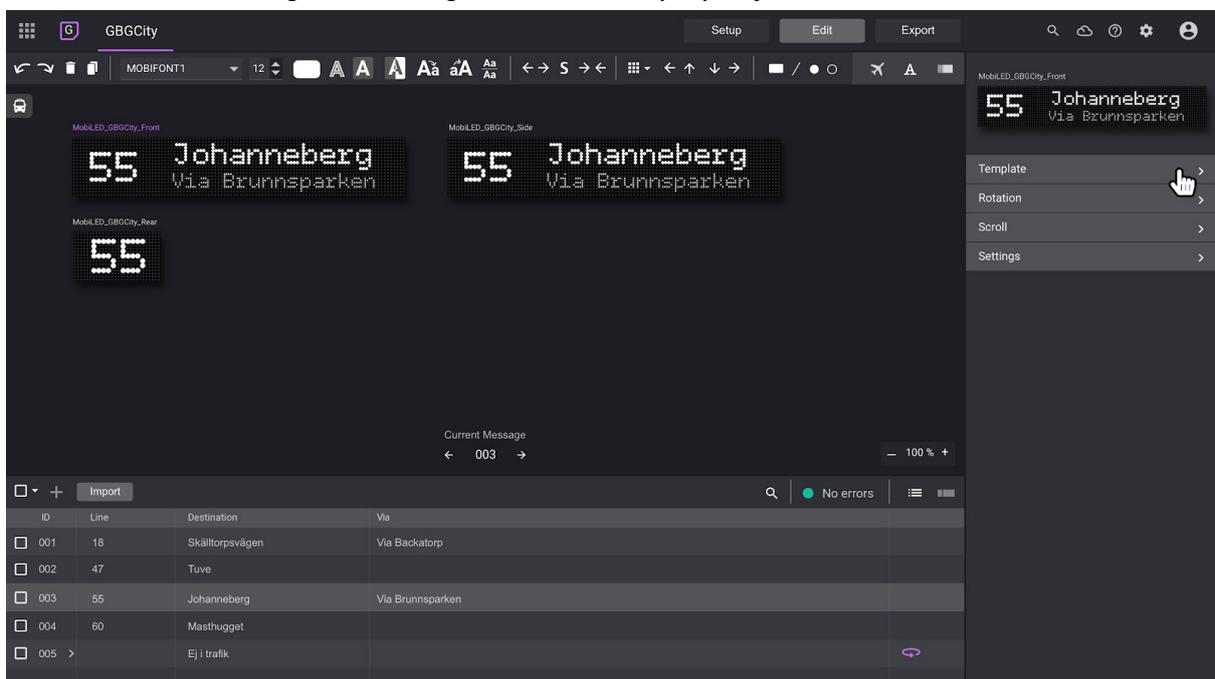


4. The the rotations are folded up. Now in message 005, the rotation icon is lit up to indication this message contains rotations.

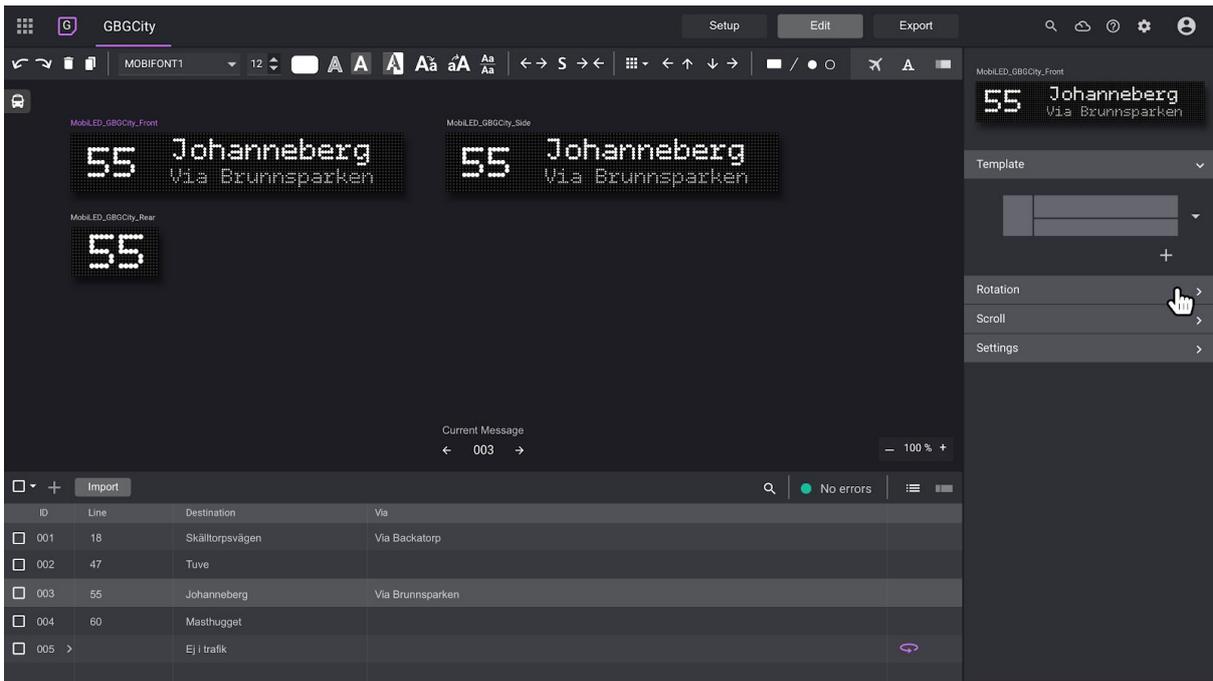
Scenario 5 - Edit for individual sign, check the properties of the front sign in message 003



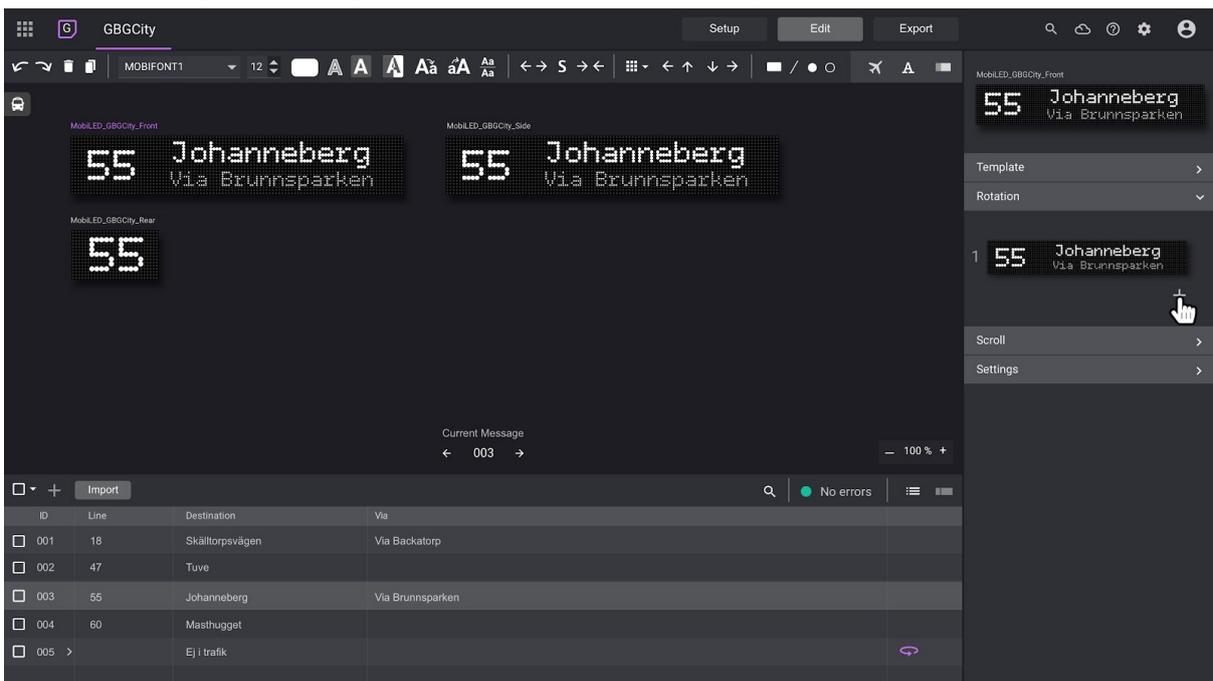
1. Click on the front sign in message 003 to active property section.



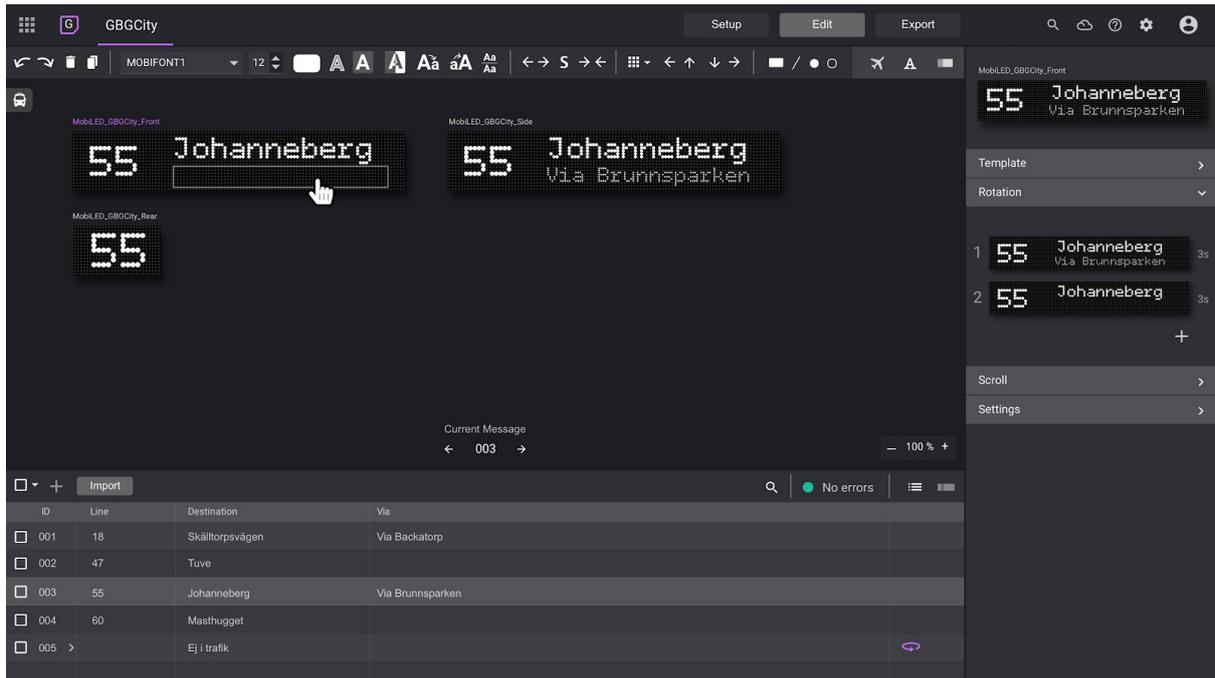
2. Now the property section is activated, here users could check the real-time simulation, change template, add rotation, add scroll or change the setup settings of selected sign. Click 'template' to check current template.



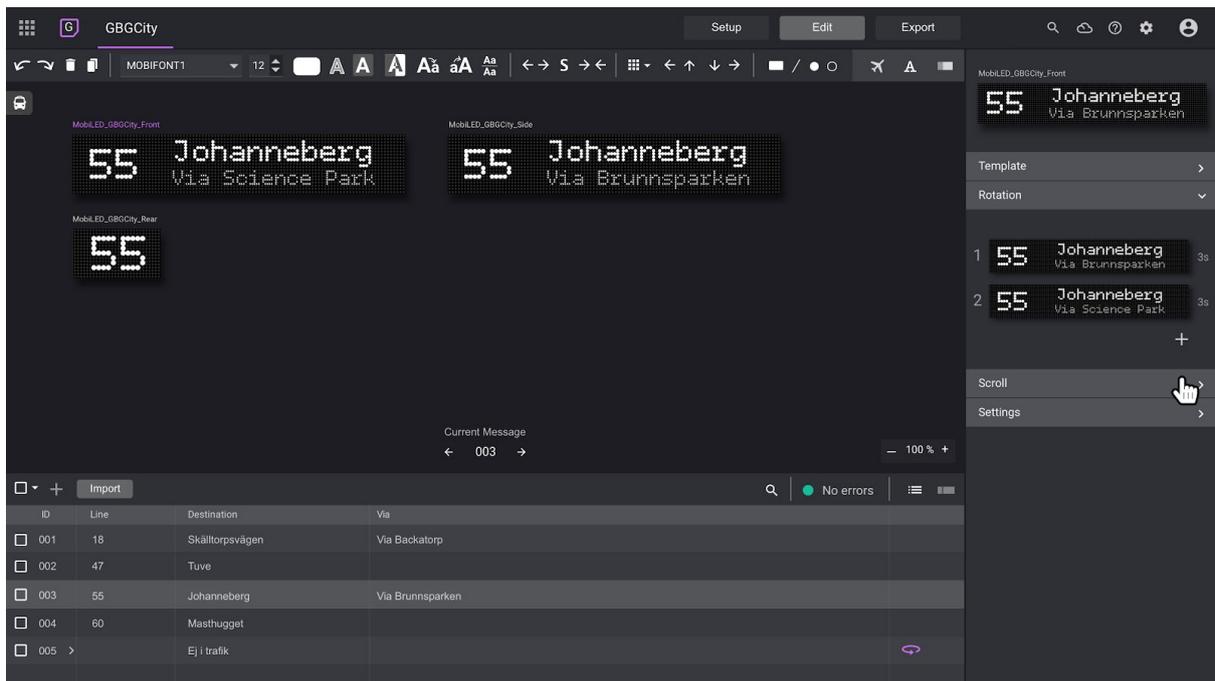
3. In template section, users could change current template by selecting from template list or create new template in template editor. *Click 'rotation' to open rotation section.*



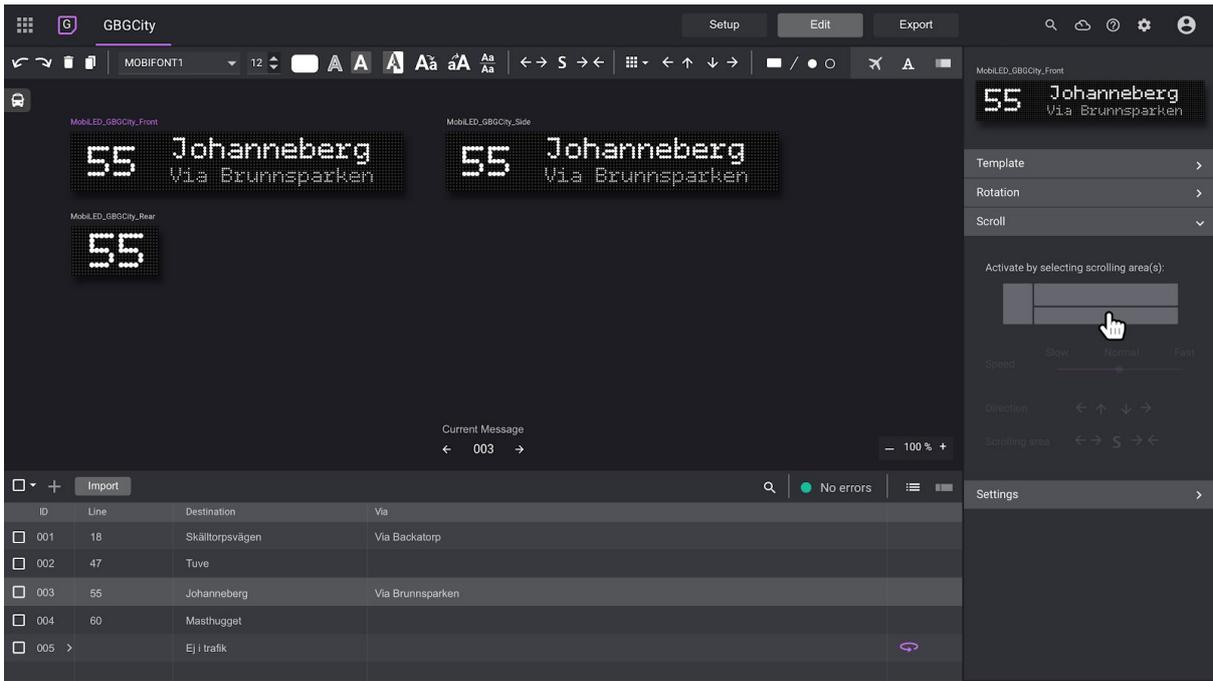
4. In rotation section, users could create new rotations for individual signs. *Click on the '+' button to add a new rotation.*



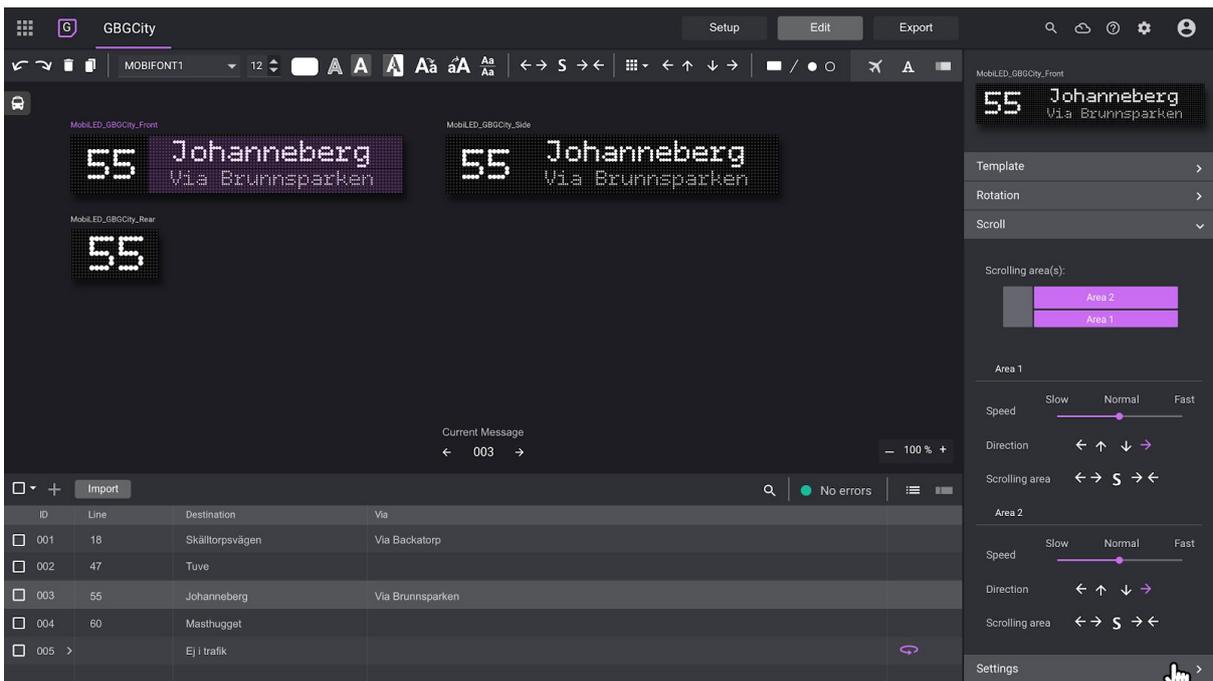
5. A blank rotation is created. *Input the text 'via Science Park' in the blank box.*



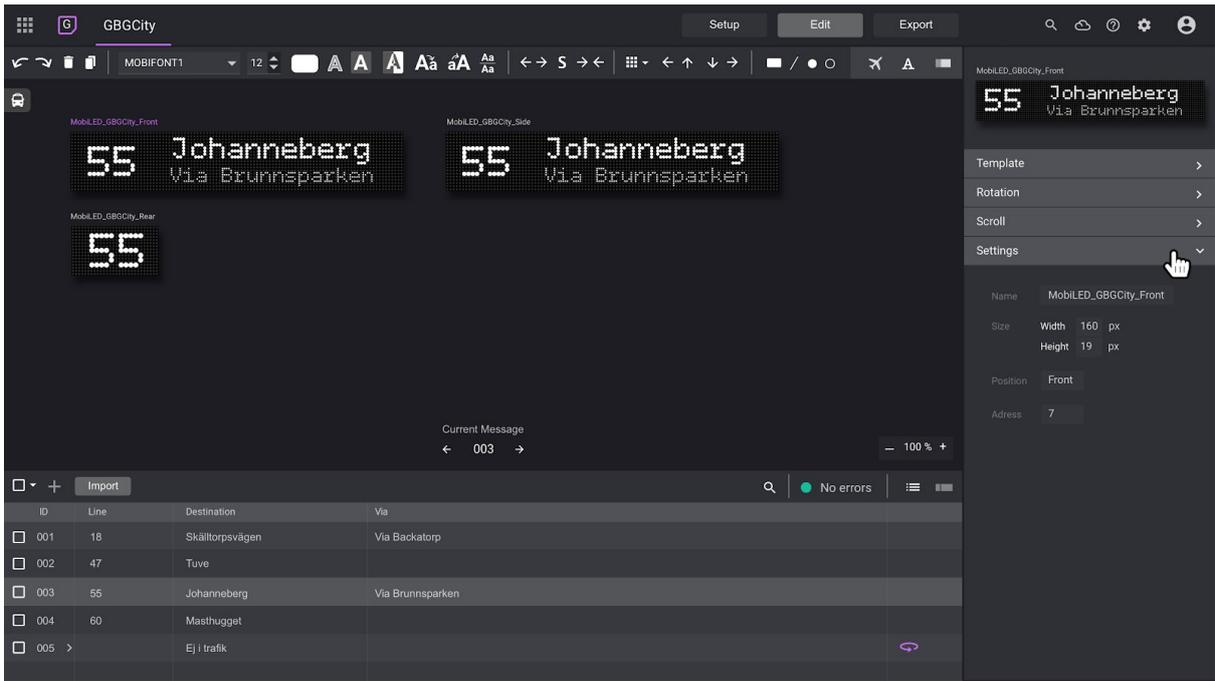
6. Now a rotation for the selected sign is created successfully, in the property section users could get an overview of all rotations for this sign and also set the duration for each rotation. *Click 'scroll' to open scroll section.*



7. In scroll section, users could activate scrolling text for different parts of the sign. Assuming that the user wants two areas to be scrolled, then *click on the two areas*

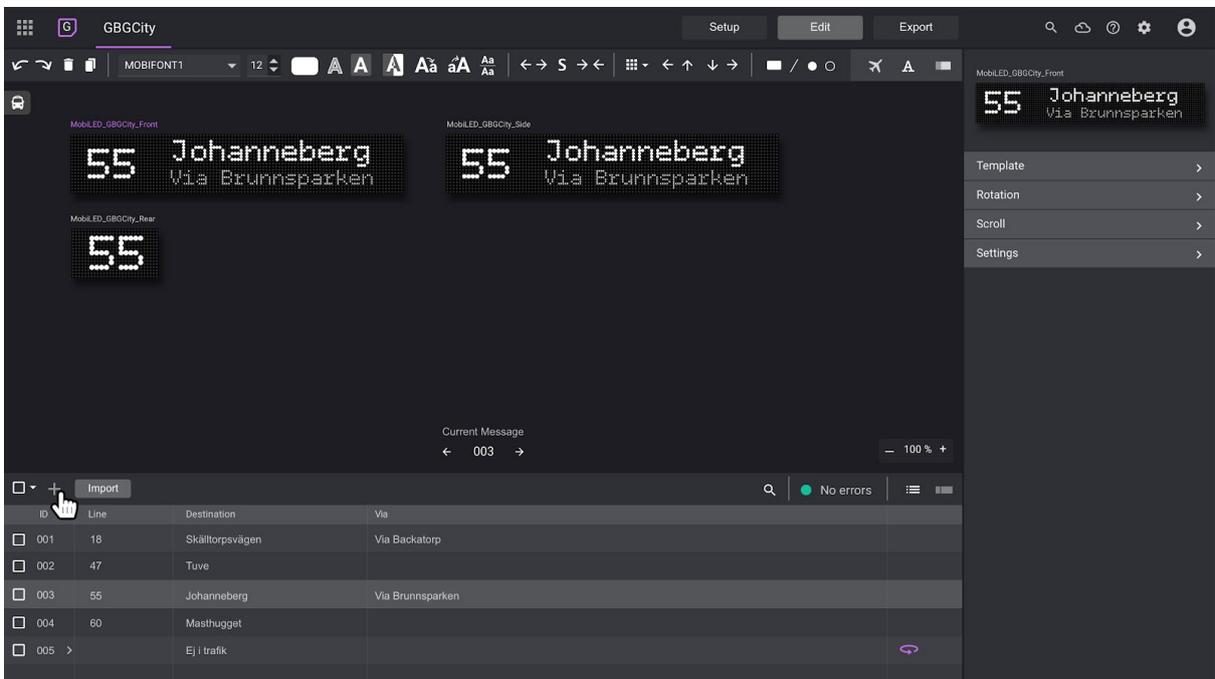


8. Now two areas are marked, in the scroll section users could also adjust the settings for these two areas separately such as scrolling speed, scrolling direction and scrolling area size. Moreover, in the preview sign in the work area, the selected scrolling areas will be marked with purple background at the same time to indicate the scrolling feature. *Click 'settings' to open the setting section.*

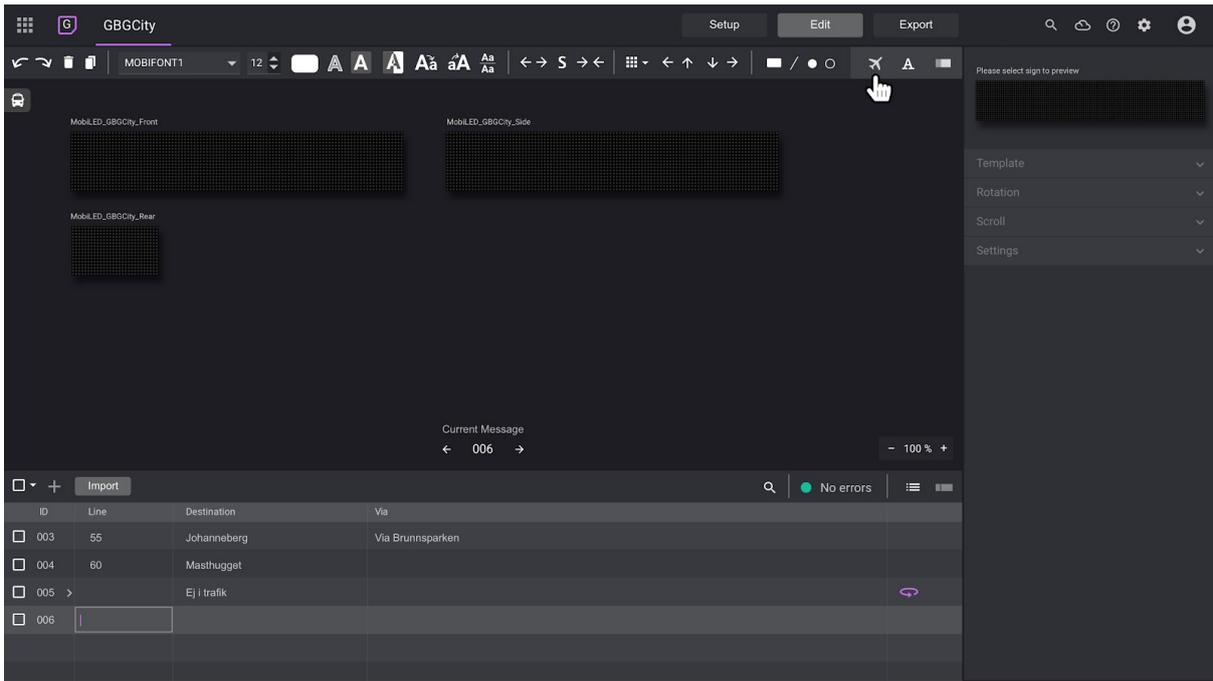


9. In the setting section, users could check and adjust the detailed setting information for selected sign including sign name, sign size, position and address. *Click 'settings' again to fold this section.*

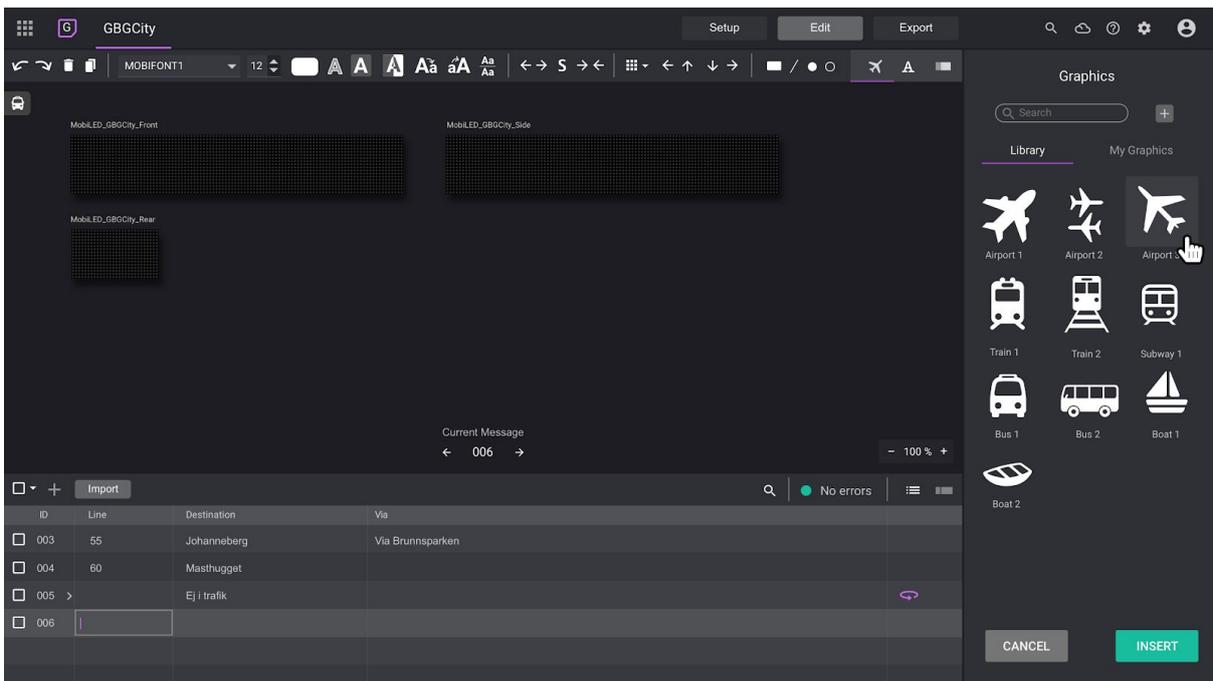
Scenario 6 - Add a new message with aircraft icon



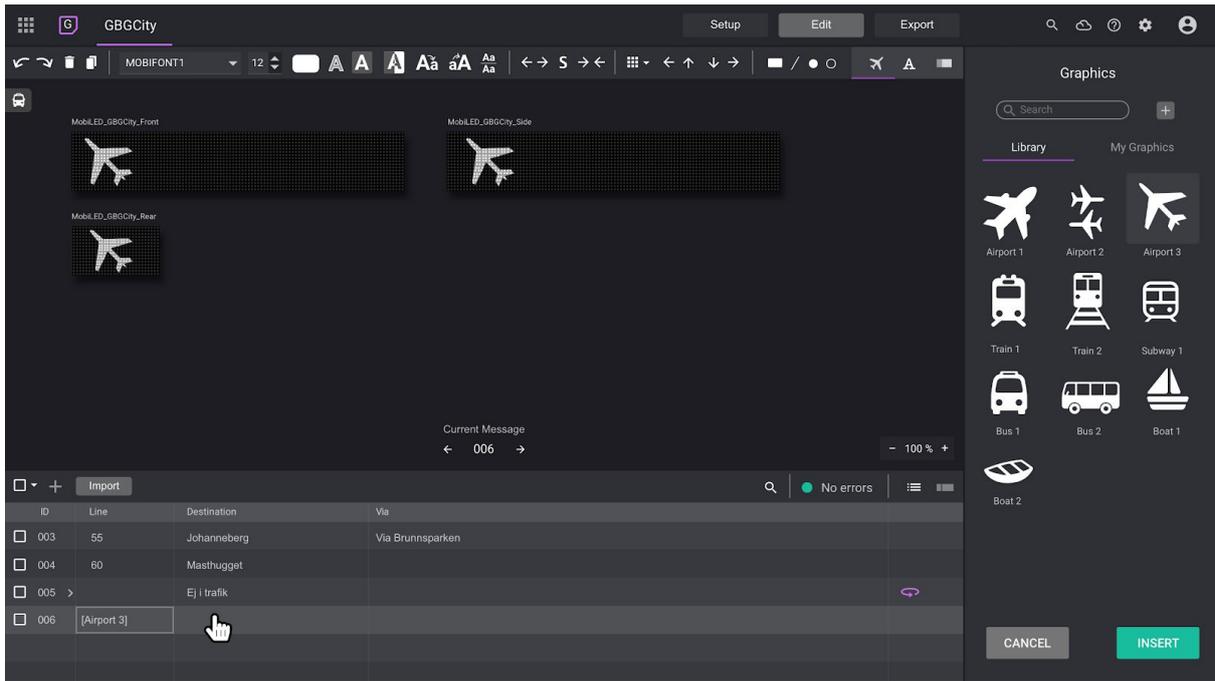
1. Click on the '+' icon to create a new message.



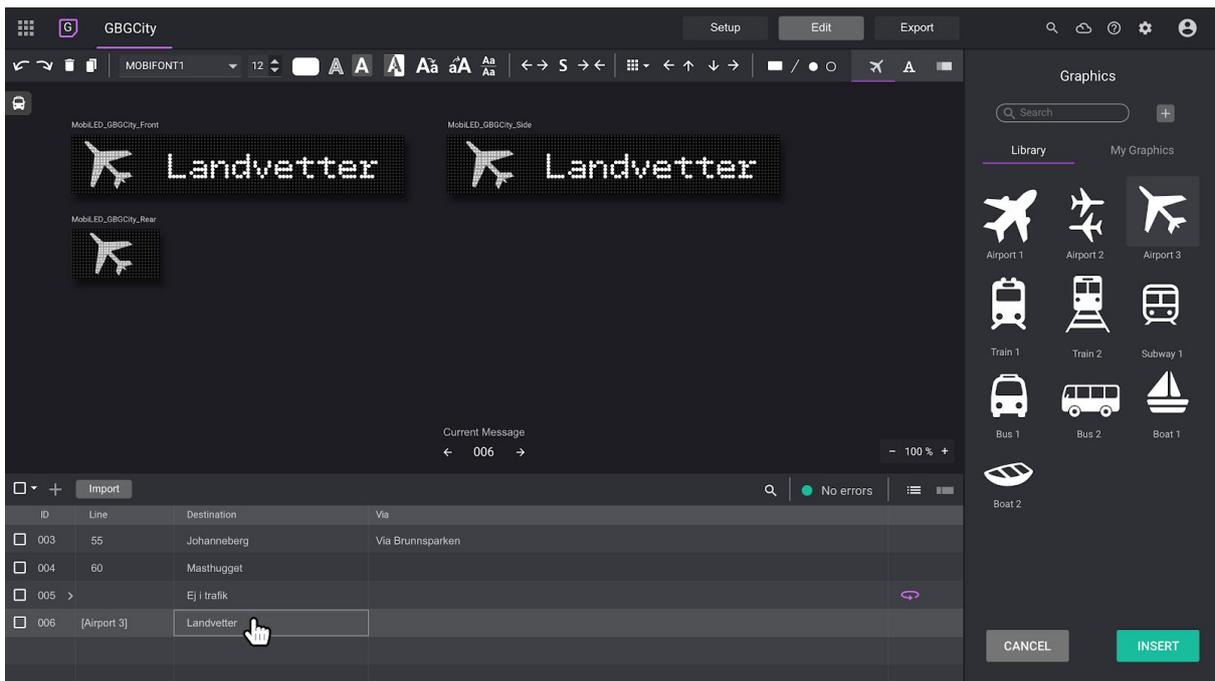
2. Click on the 'aircraft' icon to enter graphic library.



3. Here is the graphic library where users could go through all graphics offered by default, search for specific graphics or create customized graphics. Now 'airport 3' icon looks good, drag and drop it into message listing or select and click 'insert'.

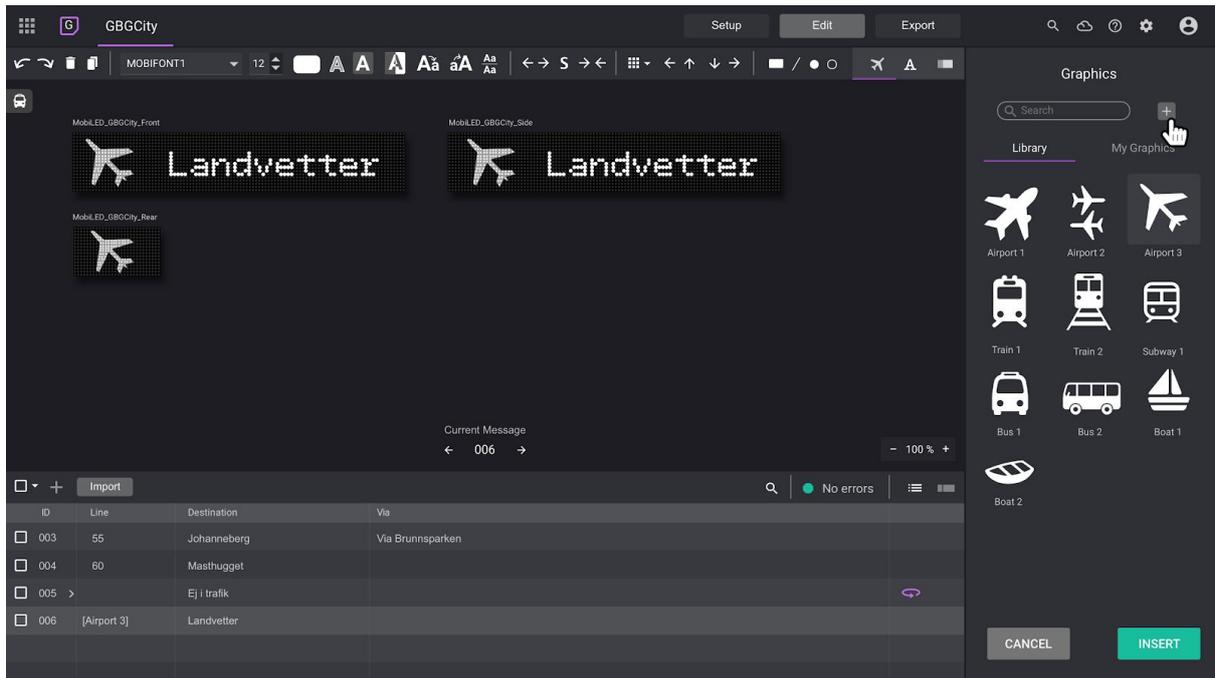


4. Now the icon is applied to all signs in the line block. *Input text 'Landvetter' in the destination block.*

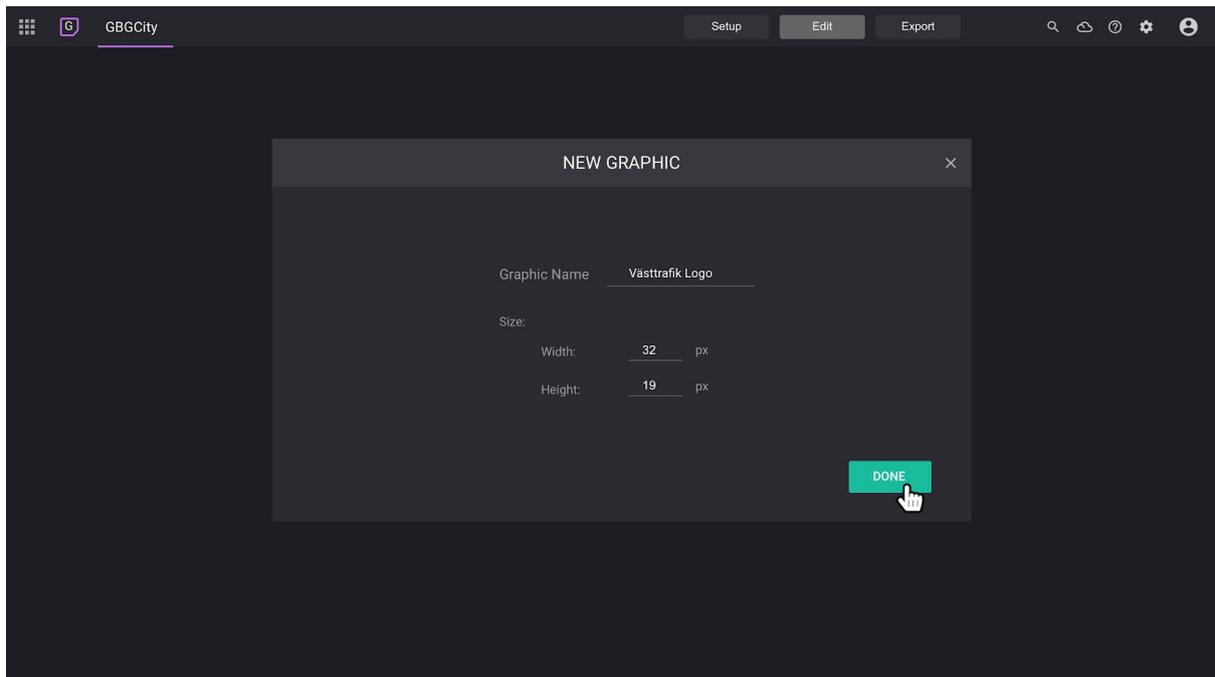


5. The airport message is created successfully.

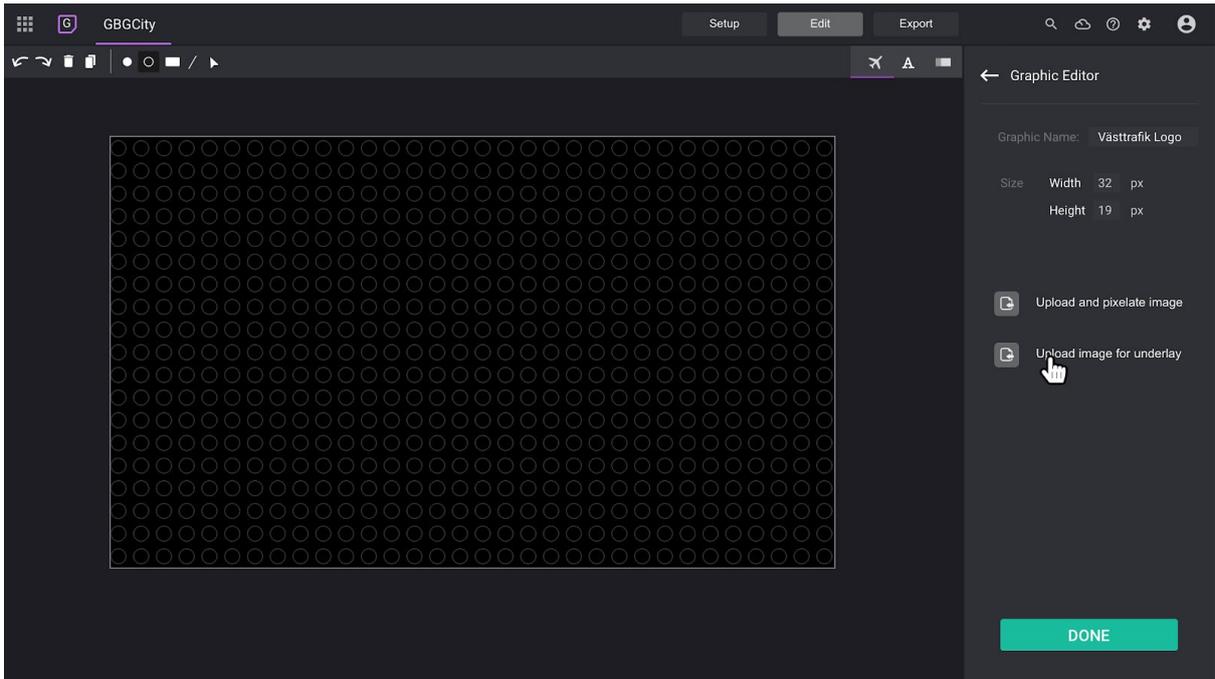
Scenario 7 - Create Västtrafik logo in graphic editor



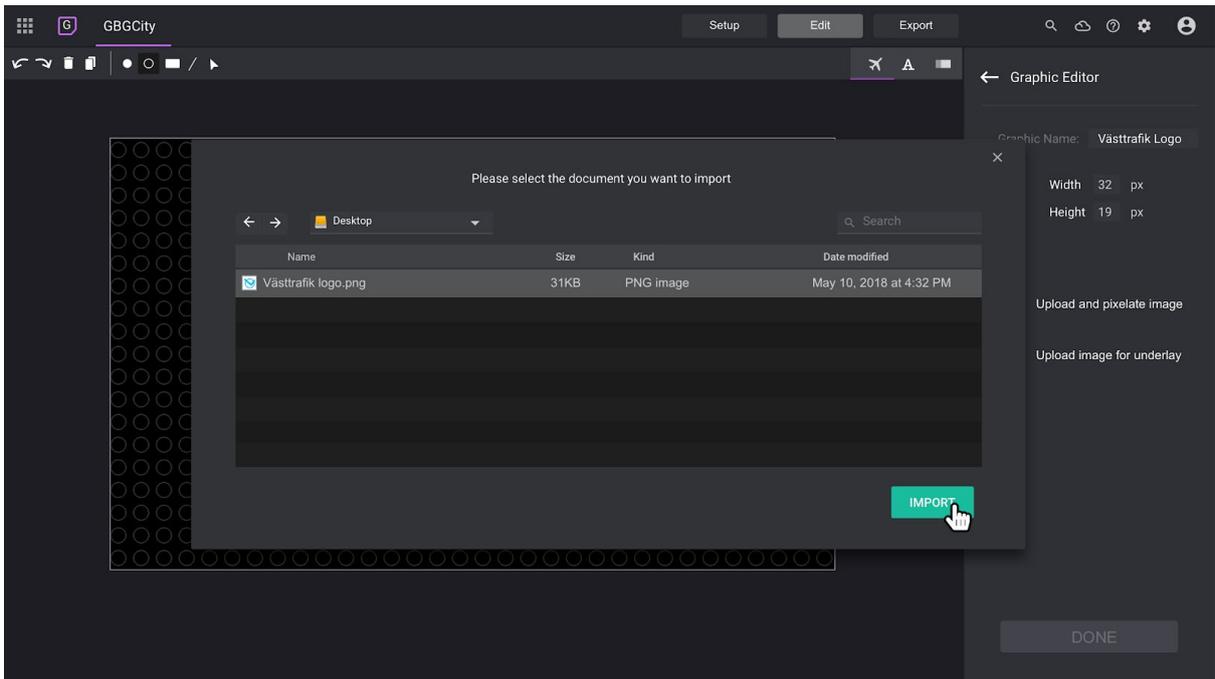
1. Click on the '+' icon to enter graphic editor.



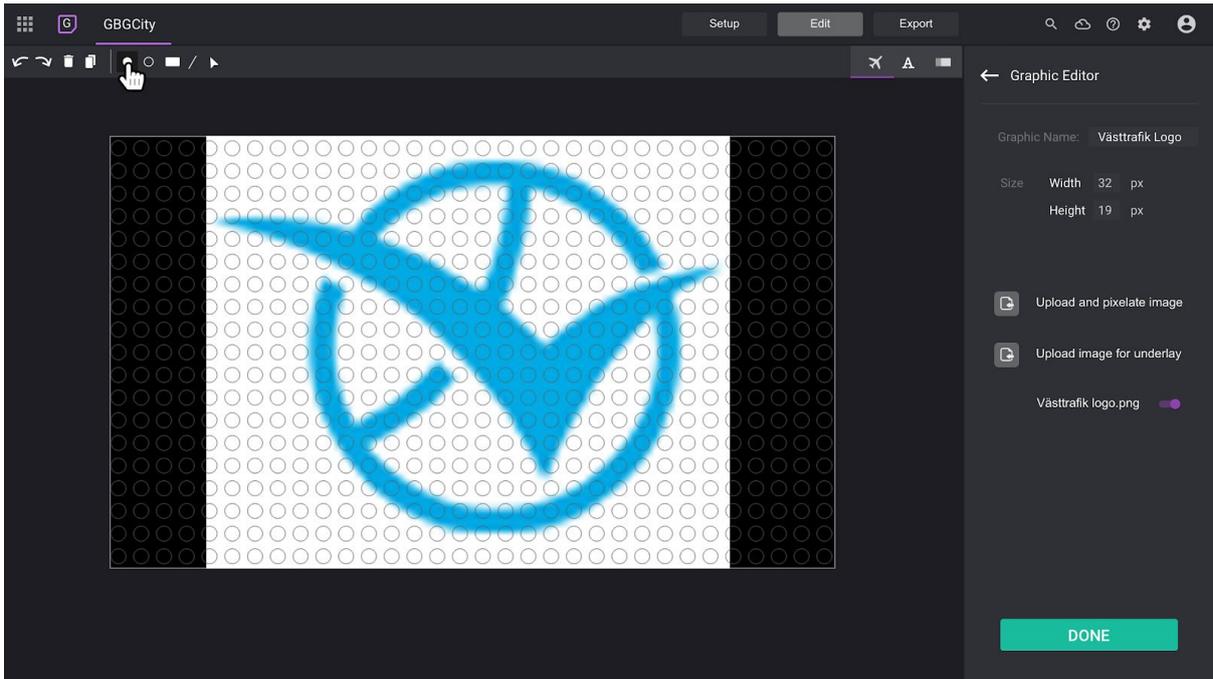
2. Define graphic name and size. Click 'done'.



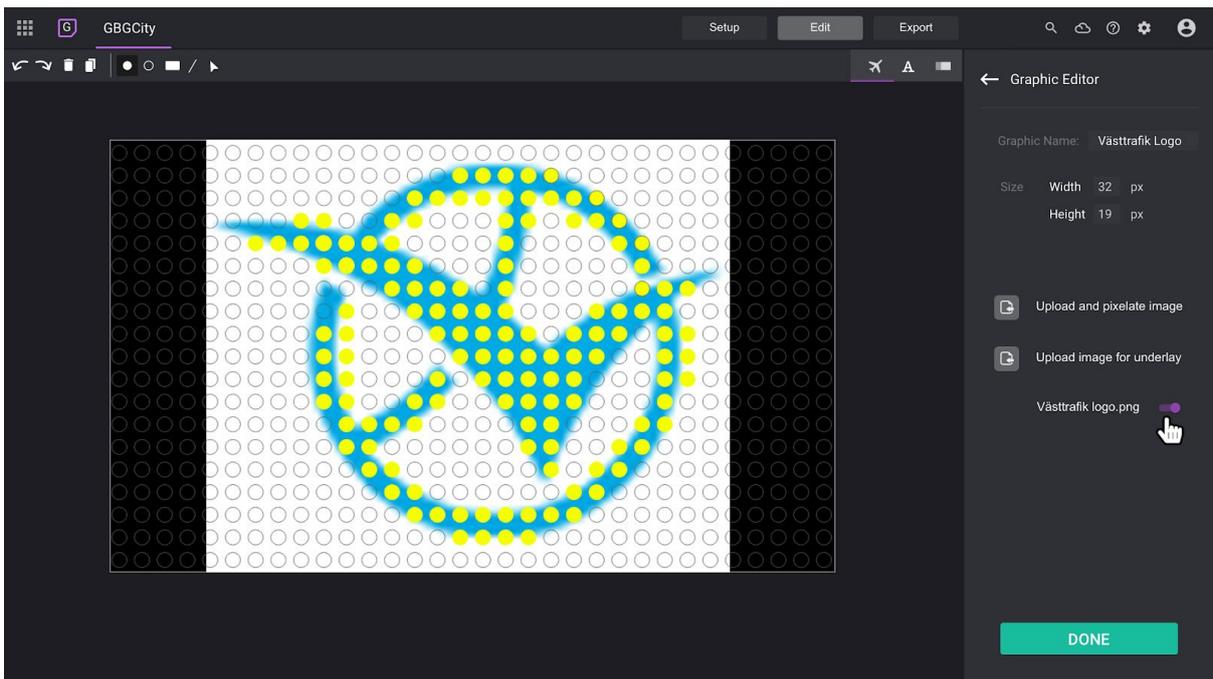
3. This is the main interface of graphic editor, the center area is the work space that allows users to create new graphics with the tools in the toolbar. And in the left window, users could check the settings of current graphic, upload and pixelate image or upload image for underlay. *Click 'upload image for underlay' to assist the drawing.*



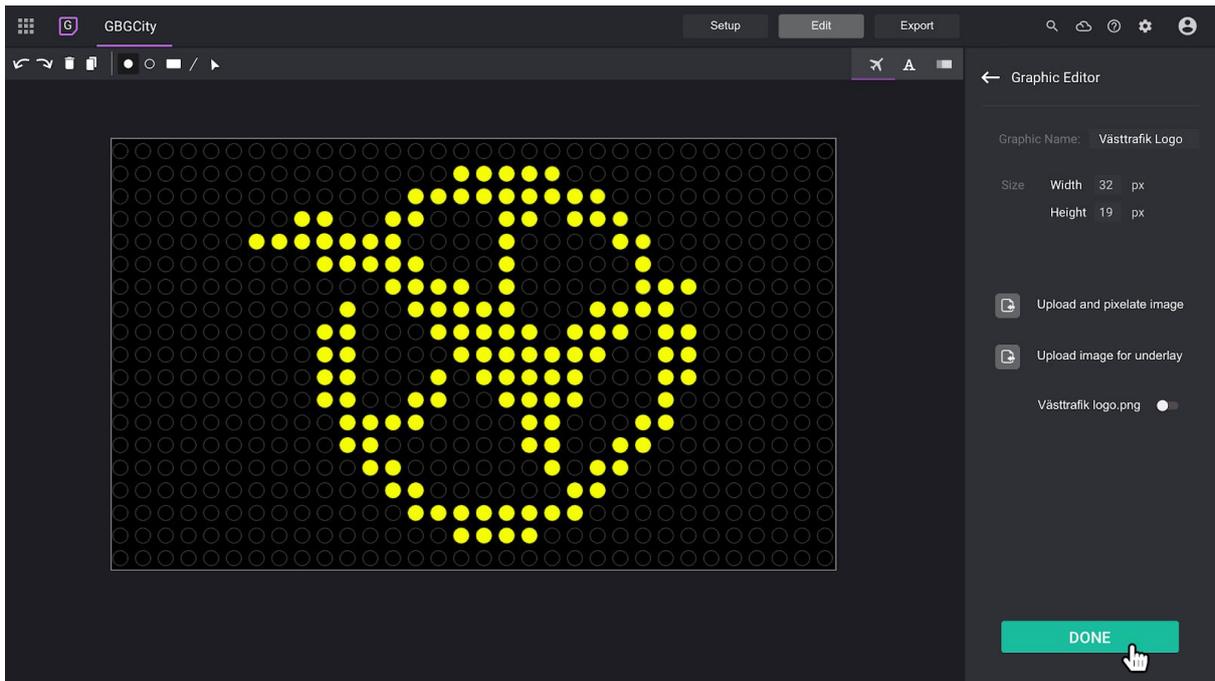
4. *Select the image 'Västtrafik logo' and click import.*



5. Now Västtrafik logo is uploaded successfully, adjust and move it to desired size and position. Use the dot tool to draw pixels in the workspace.

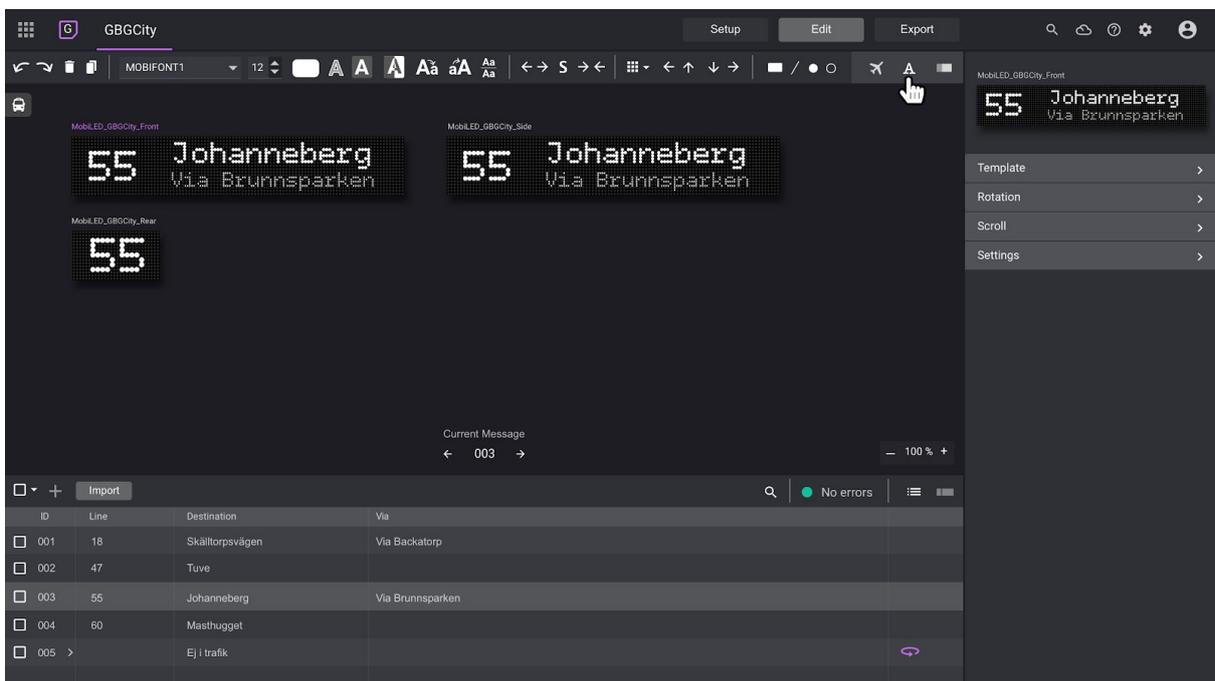


6. The drawing is finished. Turn the underlay image off to check the effect.

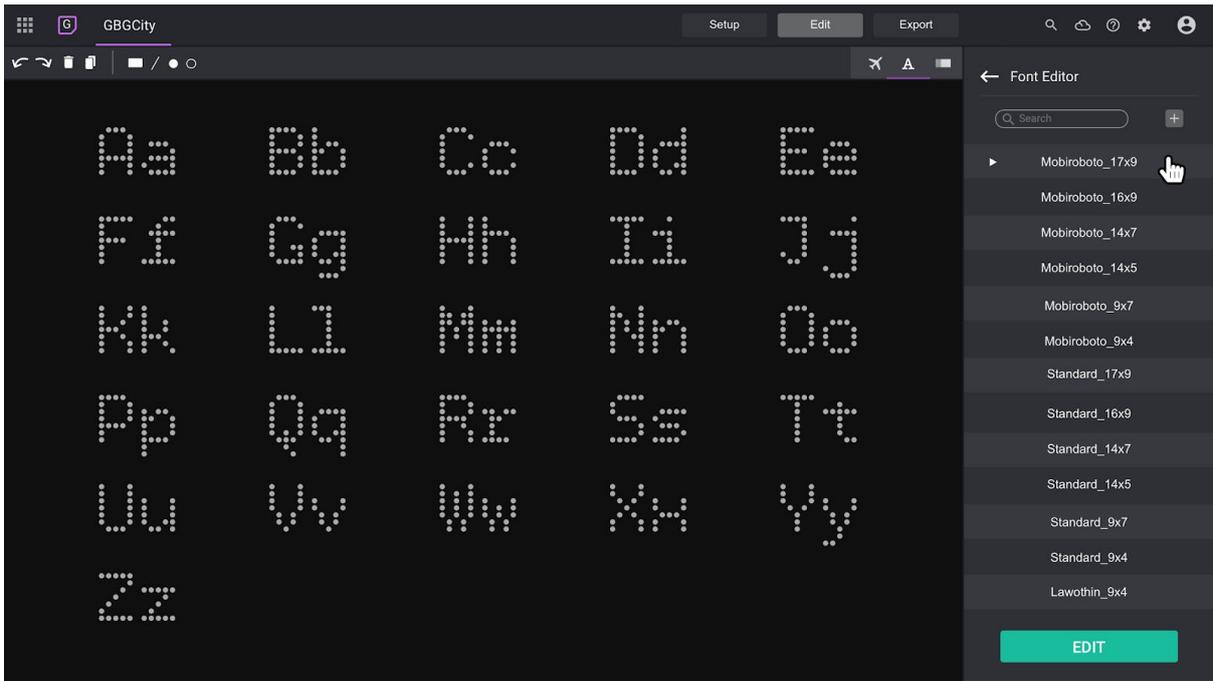


7. Good job, the logo looks great! Click 'done' to exit graphic editor.

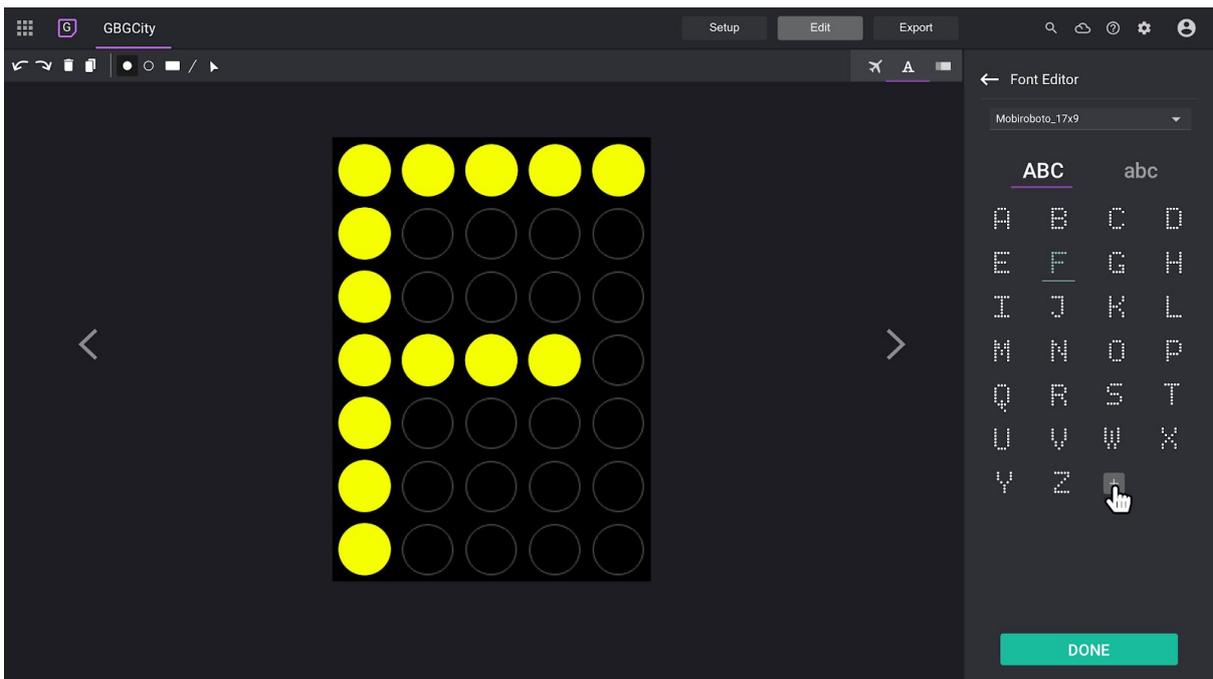
Scenario 8 - Create a new font by making adjustments on an existing font



1. Click on the 'A' icon to enter font editor.

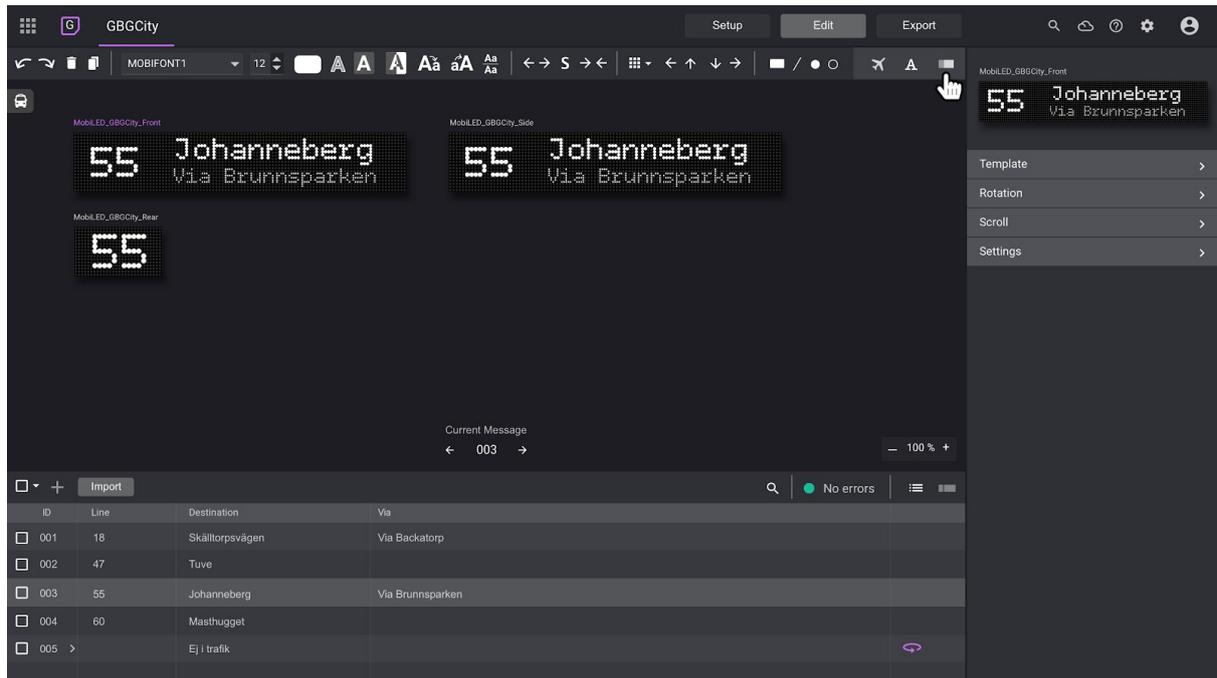


2. Go through the default font list in the software on the right side and check their effects in the main work space. The font 'Mobiroboto_17x9' looks good, *double click on it to edit.*

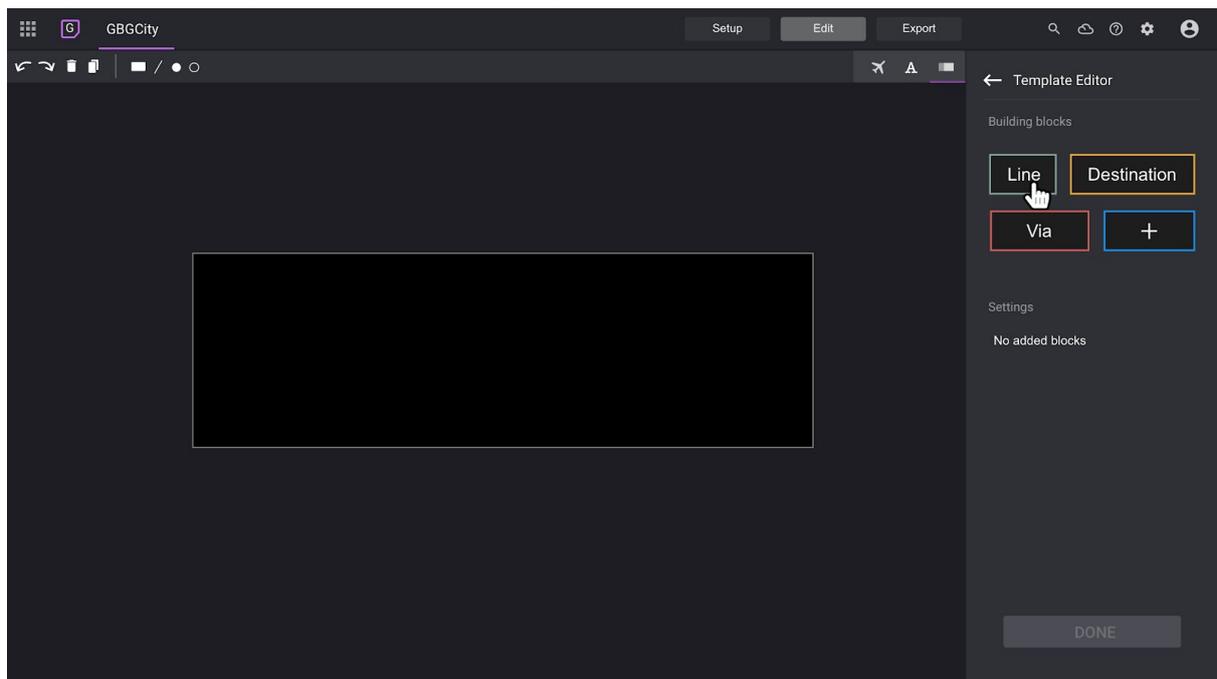


3. Here is the editing interface for each letter, users could select the individual letter they want to edit on the right side and it will appear in the edit space. ABC stands for uppercase letters and abc is for lowercase letters. By right click and left click, the dots in the workspace could be turned on and off. Users could also click on the '+' button to create new letters from scratch.

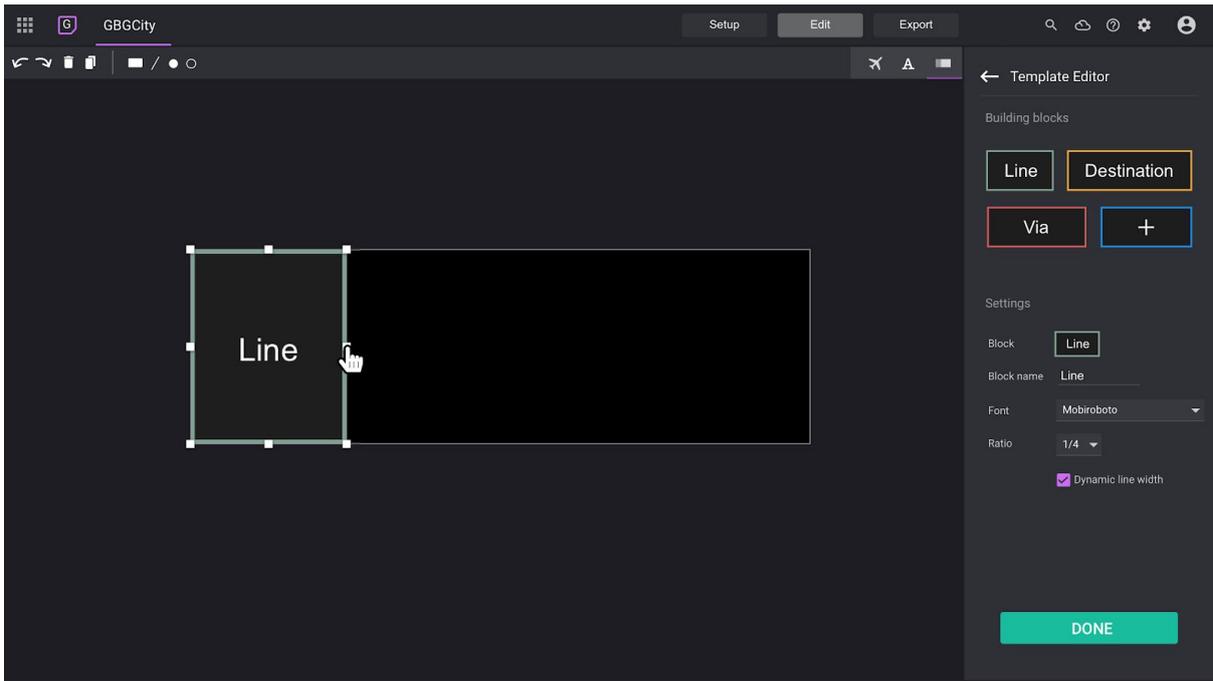
Scenario 9 - Create a new template in template editor



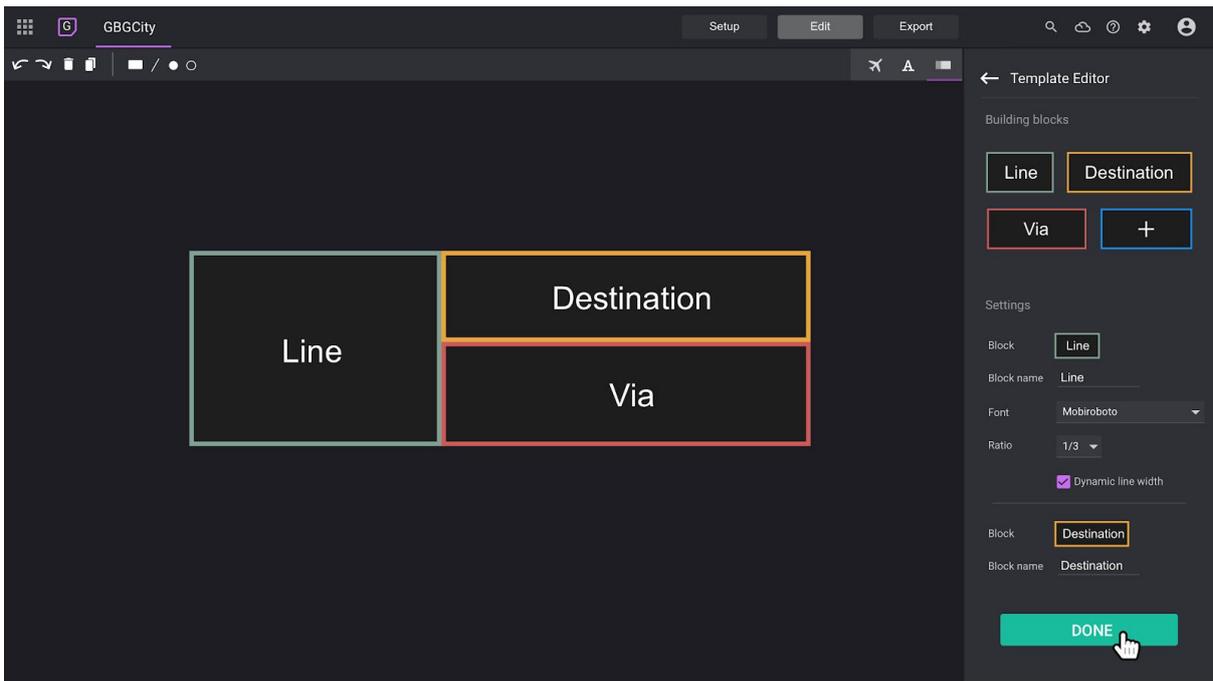
1. Click on the 'template' icon to enter template editor.



2. Here is the template editor. On the right side there are different types of blocks such as line, destination, via and other, by dragging and dropping them into the center workspace users could create their own template. *Drag and drop the line block.*

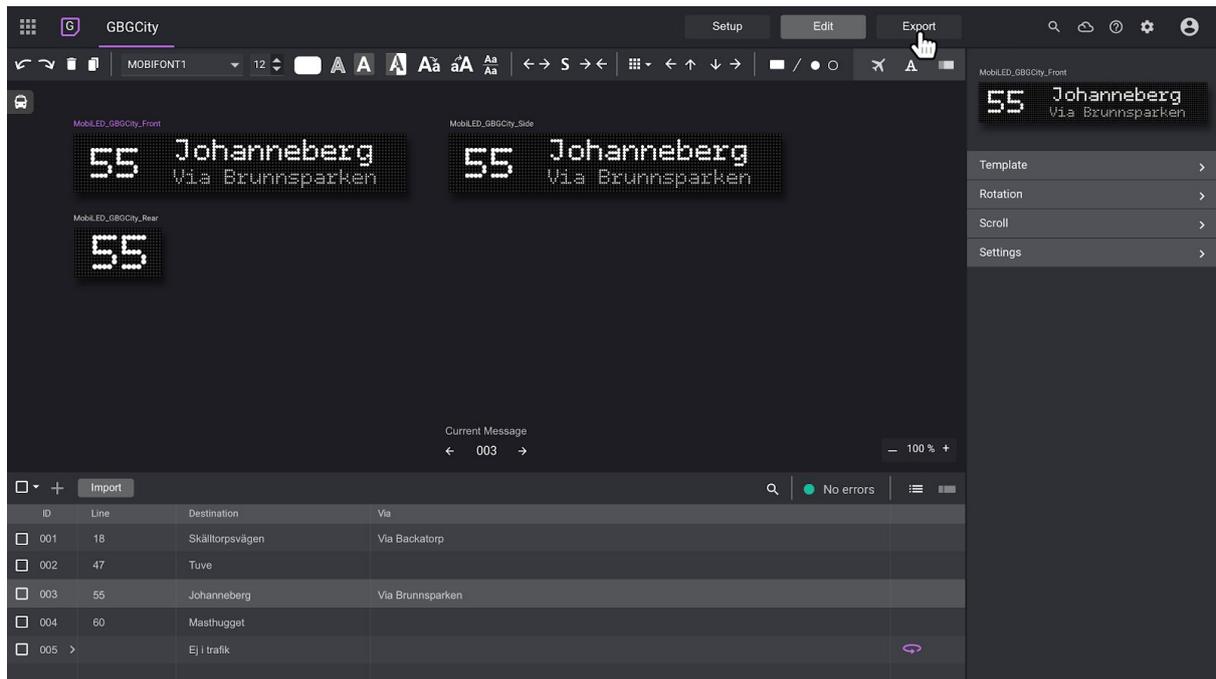


3. Now the line block appears in the template space, users could also adjust the size of it by pulling the edges. In the right side, the settings for this block such as font, ratio could be configured. *Continue drag and drop other blocks.*

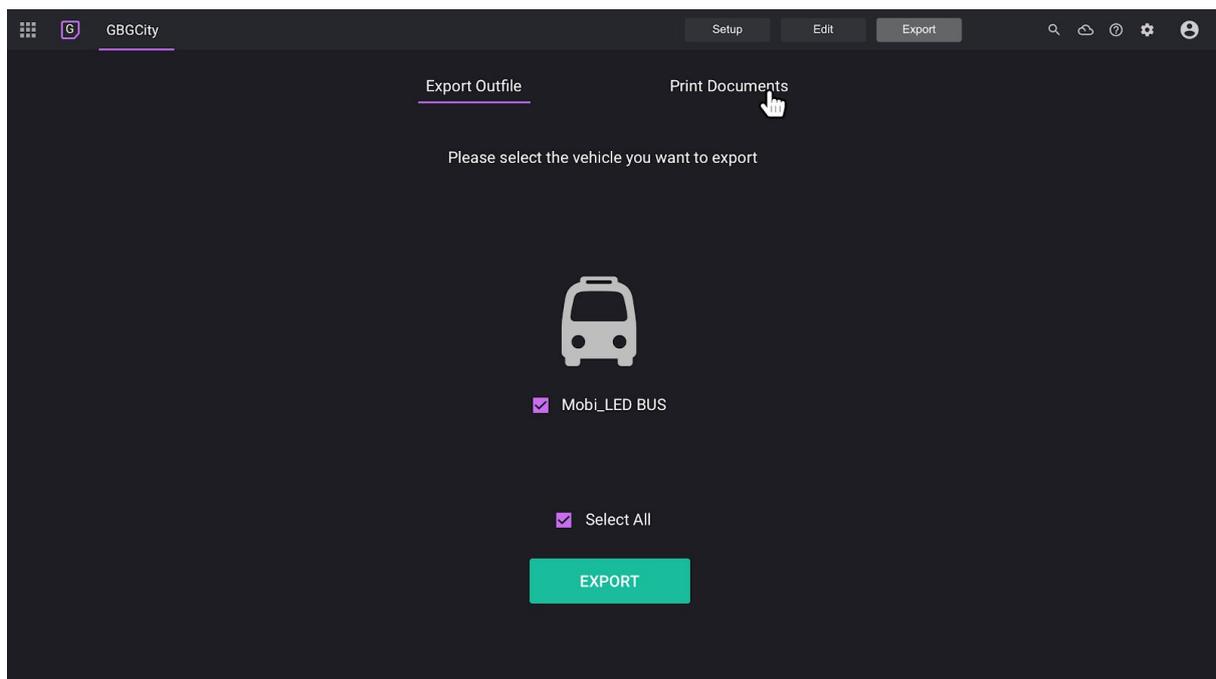


4. Following the same procedure, a new template composed of line, destination and via blocks is created. *Click 'done' to exit template editor.*

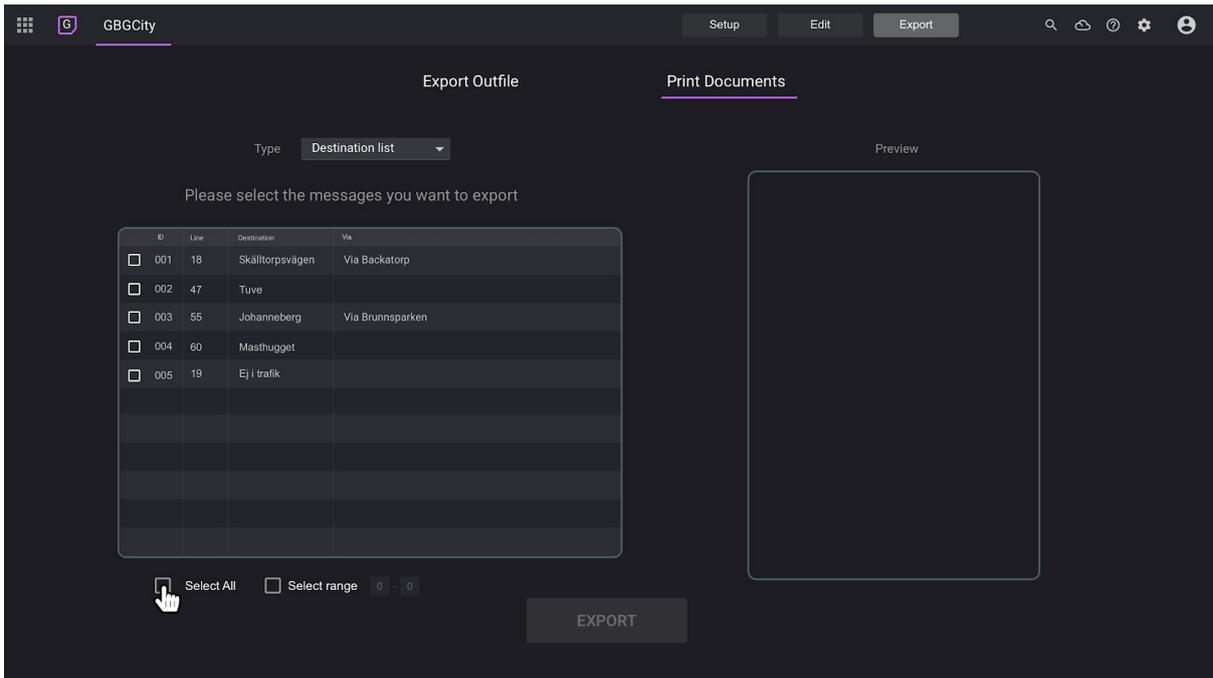
Scenario 10 - Export outfiles and print materials



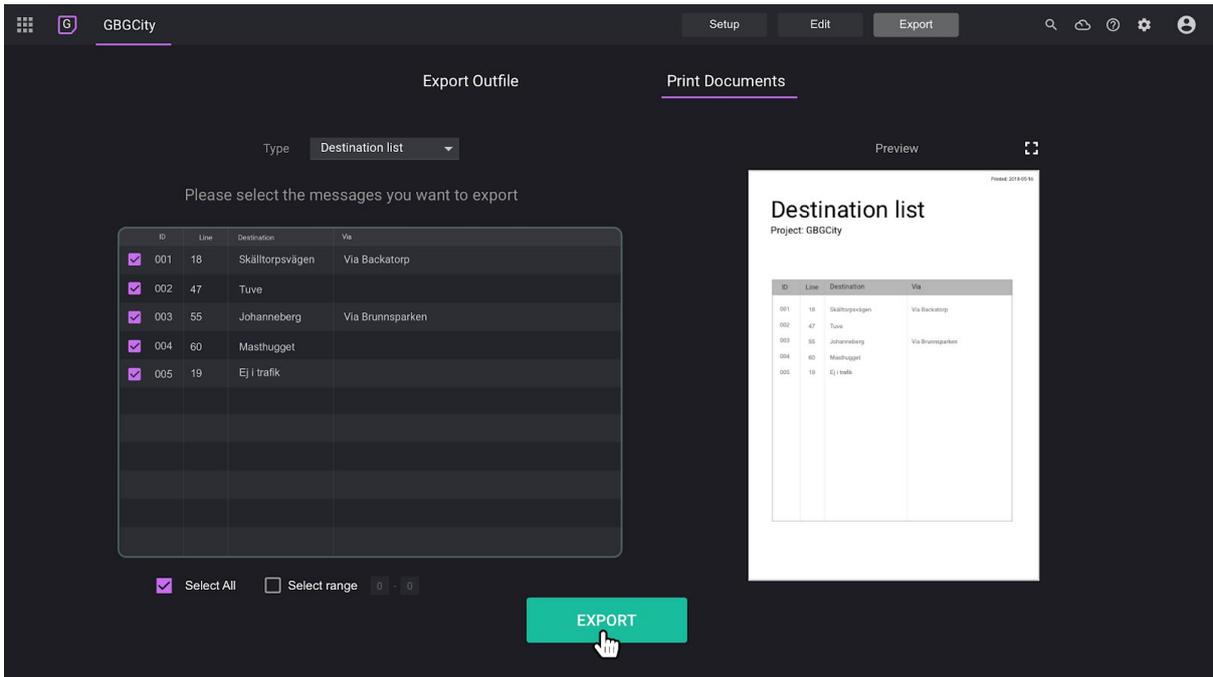
1. Click on the 'export' button to enter export interface.



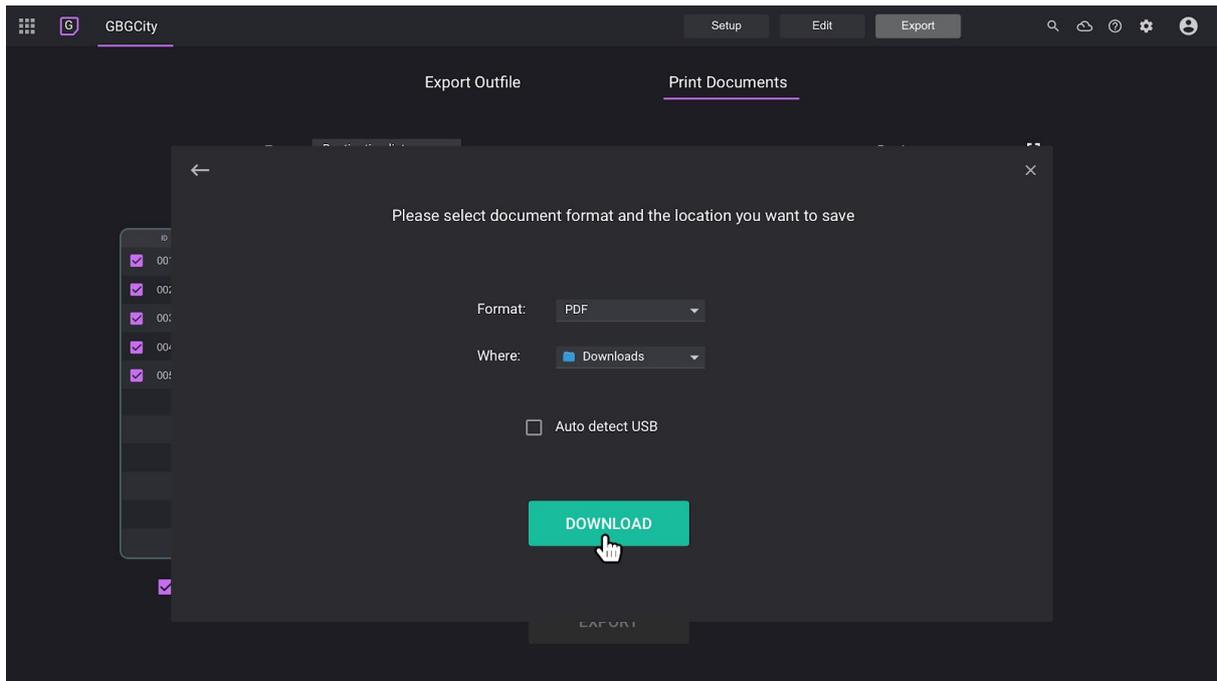
2. The export interface is composed of two sections, export file and print documents. In export outfile section, users could select which vehicle(s) to generate outfile for, then click 'export' and define where to store the document. Click 'print documents' to enter this section.



3. in the print document section, users could generate documents for various purposes such as administration or driver notes. In the left area the messages that users want to include in the document could be selected and in the right area a preview will be generated automatically for users to check the effect and make adjustments if needed. *Click 'select all'*



4. A preview is generated, it look good! *Click 'export'.*



5. Define the document format and storage location. Click 'download' to download this document.