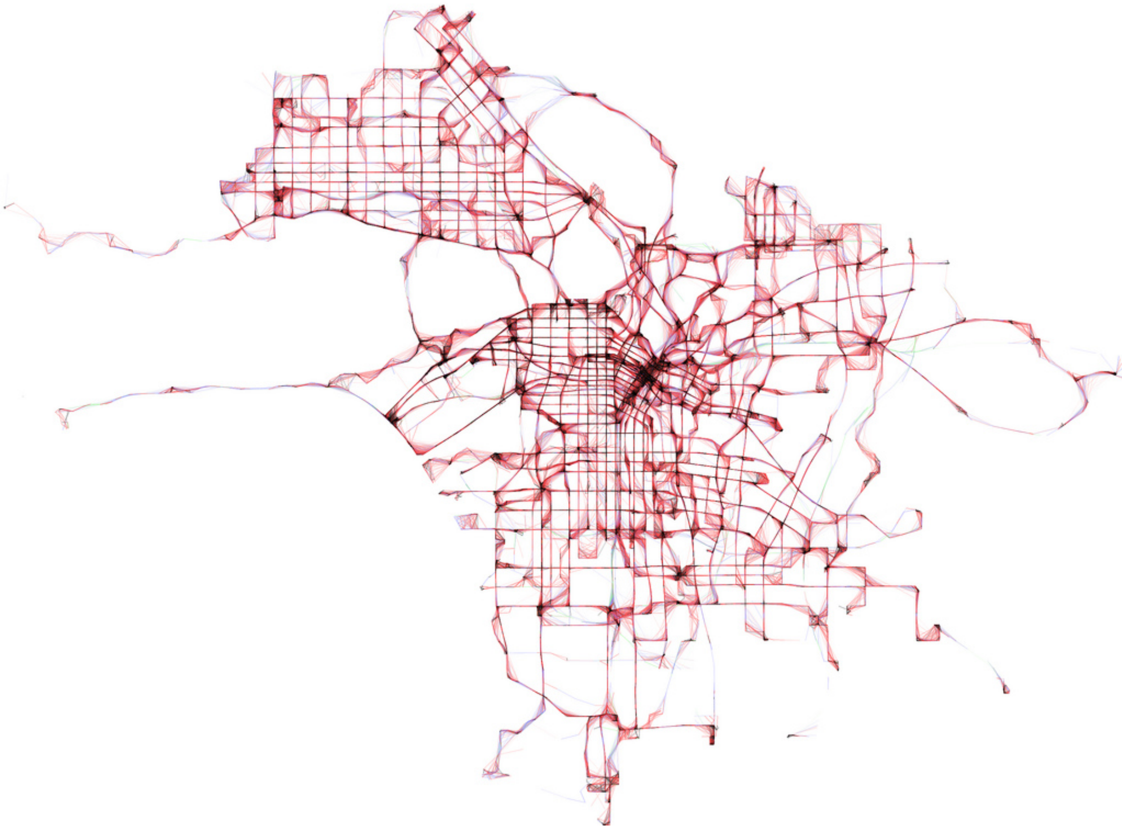

Geographical Data Visualization in Logistical Services

Visualizing Geographical Data to Facilitate Data Driven Decisions in Short Distance Logistical Services

Master's thesis in Interaction Design and Technologies

Oscar Blomqvist

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2017

Master's thesis 2018

Geographical Data Visualization in Logistical Services

Visualizing Geographical Data to Facilitate Data Driven Decisions in Short Distance Logistical Services

OSCAR BLOMQVIST

Department of Computer Science and Engineering
Interaction Design and Technologies
Chalmers University of Technology
University of Gothenburg
Gothenburg, Sweden 2018

Geographical Data Visualization in Logistical Services
Visualizing Geographical Data to Facilitate Data Driven Decisions in Short Distance Logistical Services
OSCAR BLOMQVIST

Supervisor: Olof Torgersson, Department of Computer Science and Engineering
Examiner: Staffan Björk, Department of Computer Science and Engineering

Master's Thesis 2018
Department of Computer Science and Engineering
Interaction Design and Technologies
Chalmers University of Technology and University of Gothenburg
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: A poly line visualization of transits of Los Angeles Metros

Typeset in LaTeX
Gothenburg, Sweden 2018

Geographical Data Visualization in Logistical Services
Visualizing Geographical Data to Facilitate Data Driven Decisions in Short Distance Logistical Services
OSCAR BLOMQVIST
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg

Abstract

Bzzt, a pod taxi company operating in central Stockholm, has a lot of geographical data such as customer bookings, destinations and taxi locations. To explore and discover insights using all of this data, a geographical data visualization system is required. In this paper, the requirements and important considerations for designing such a visualization system are explored. This research is done by interviewing users and stakeholders at Bzzt, observing Bzzt's operations and workflows, studying theory as well as analyzing existing geographical data visualization systems. The results of this investigation are the design of BGIS, a tool for visualizing, exploring and sharing geographical data. Further more, based on the discoveries during this design process, a set of design guidelines for designing geographical data science tools for logistical services are produced. Some of the most important findings are that the geo-data systems for logistical services needs to support a wide range of different geographical visualization methods, provide a high degree of flexibility in filtering data and support for exploring how data changes over time. The resulting design of BGIS, a web application for visualizing geo-data, includes these requirements. The paper also investigates technical consideration for implementing BGIS, how the design process may be continued as well future improvements of the design.

Keywords: Geographical Data Visualization, Data Science Tool, Taxi Services, Logistical Services

Acknowledgements

I wish to express my sincere appreciation to the people who helped me during this thesis project.
Everyone at Bzzt AB for your genuine interest in my work, and that you provided me with support, guidance and help throughout the project.
Johan Wingård for you always being there to give me feedback, guidance and constructive criticism. The results of this project would have had significantly lower quality if it was not for your help and competence.
Olof Torgersson for guiding me and making sure that I was progressing in the right direction throughout this thesis project.

Oscar Blomqvist, Gothenburg, June 2018

Dictionary

Nominal information coding - when a characteristic, such as color or shape, represents some information in a data visualization.

CIELuv - a color space where the colors are perceptually uniform; the distance between two colors are determined how different the colors are perceived to be.

Redundant Coding - when a symbol or an object is distinct on several feature channels, for example color and size.

Cornsweet contour - is when the edge of an element is enhanced by a color gradient.

Epistemic action - an activity intended to uncover new information [1].

WIP - is an acronym for Work In Process commonly used in Kanban.

SME - is an acronym for Subject Matter Expert. An expert in a specific domain.

UI - is an acronym for User Interface.

Stack - can have many meanings. In this thesis, stack refers to a *tech stack*, which is all tools and programming languages used to create a digital system.

NREL - is an acronym for National Renewable Energy Laboratory.

HCI - is an acronym for Human-Computer Interaction. This is a research field that focuses on the interaction between humans and computers.

Chromaticity - is an attribute of color that is independent of its luminance. It is based on two parameters: hue and saturation [2].

JSON - is an acronym for Javascript Object Notation, which as common way of formatting text which is stored, or exchanged between clients and servers.

CSV - is an acronym for comma-separated values, which is a commonly used file format for storing data records in plain text.

R - is a programming language specialized in statistical computing.

Shiny Dashboards - is an R package that allows for easy of creation of web application consisting of dashboards and data visualizations.

Contents

List of Figures

1
Introduction

The goals of this master thesis are to design and create a prototype of a geographical data visualization tool for the pod taxi company Bzzt. The tool should be modular enough to allow presenting a wide array of different data points. It should also be possible to display the same data in various ways, depending on the needs of the user. The purpose of the system is to optimize Bzzt's pod taxi operations and find potential customer and operational problems. This will be done by observing patterns and behaviours shown by the visualized data. Another important use of the tool is to share data visualizations, both with internal team members and external stakeholders, for the purpose of better communication and collaboration.
Apart from designing a data visualization tool for Bzzt, a secondary objective of this project is to create a set of design guidelines for designing geographical data visualization tools for logistical services, such as Bzzt as well as other taxi or delivery businesses. The insights that are discovered during the design process for Bzzt's geo-data tool, which are relevant in a more general context, will be the foundation for these guidelines. The purpose of the design guidelines is to help future designers and researchers working with geographical data visualization in logistical service domains.

1.1 Bzzt

Bzzt is the company which this Master Thesis is performed for. The core operation of Bzzt is providing taxi services using small pod taxis in urban areas. In February 2018 Bzzt only provides their taxi service in Stockholm [4], but the company is planning to expand to other cities in the near future. Some of the core characteristics of Bzzt are:

Simplicity: It should be simple and fast to order a pod taxi and travel to the desired destination.

Affordable: Bzzt's pod taxis have a fixed rate of three Swedish pennies (öre) per meter, making it an affordable travel option for inner city trips.

Environmental: All of the pod taxis use electrical engines and hence do not produce exhaust.

1.2 Stakeholders

An overview of the different stakeholders of this master thesis.

Oscar Blomqvist
The student performing the master thesis.

Bzzt
The company for which the master thesis is performed.

- **Johan Wingård:** User Experience director and Oscar Blomqvist's external supervisor at Bzzt. Potential future user of the tool.
- **The tech team at Bzzt:** Collaborators with Oscar Blomqvist during the thesis. Provides data and system operational necessities such as servers etc. Future users of the tool.
- **The operations and marketing team at Bzzt:** Potential future users of the tool.

Chalmers and Interaction Design technologies

The master thesis will be performed under the Interaction Design Technologies division under the Computer Science department at Chalmers university of technology. **Olof Torgersson** is Oscar's supervisor in the Interaction Design department at Chalmers.

1.3 The research questions

How can the pod taxi company Bzzt's geographical data be presented and explored to allow for business improvements and operational optimization?
What are important requirements, aspects and options when designing geographical data science systems for logistical services?

1.4 Planned result

The planned results of this master thesis are a set of design guidelines for designing geographical data science tools for logistical services, as well as a design for a geographical data science system for Bzzt AB. The design guidelines will be developed from designing the geographical data visualization tool for Bzzt. The functionality of the tool can be limited to a few, critical features, but it is important that it is modular enough to be expanded with additional data points and visualization options.

2

Background

An overview of existing tools and solutions used in exploration and visualization of geographical data.

2.1 Uber Data Science Workbench (DSW)

The company Uber previously had a wide range of different data visualization and analytic tools used by different areas in the Uber infrastructure [5]. Included areas are traffic information, route optimization, drop-off locations and user experience analysis. To make the data science more coherent and streamlined, Uber developed DSW, an all-in-one tool for data science and machine learning. DSW centralizes all aspects of data science to allow for fetching of data, exploration and visualization.

The reason Uber choose to develop their own data science tool instead of using a third party option was:

- Uber anticipated that no third party solution would be able to scale well enough to handle all future data.
- Third party platforms did not integrate as well with existing internal tools.
- Building their own platform would allow for optimization towards targeted use-cases which were the most relevant in Uber's operations.

When designing DSW the designers realized that they could not accommodate all of the different requirements across Uber's data science personnel; they needed to design a tool that provided the most essential parts required by most of the data scientists. After an extensive research phase the designers created a list of the most important requirements:

- Support for data exploration tools such as Jupyter Notebook, RStudio, Zeppelin Notebook and Shiny Dashboards.
- Support for common programming languages like R, Python and Scala.
- Support for distributed computing systems including Hive, Vertica and Spark.
- The tool needed to be fully hosted and require no manual setup from its users.
- The tool needed to ensure that several users could *share* the same data concurrently.
- A UI that allowed for easy collaboration and knowledge sharing.
- Integration with Uber's infrastructure, work flow management tools and data visualization tools.

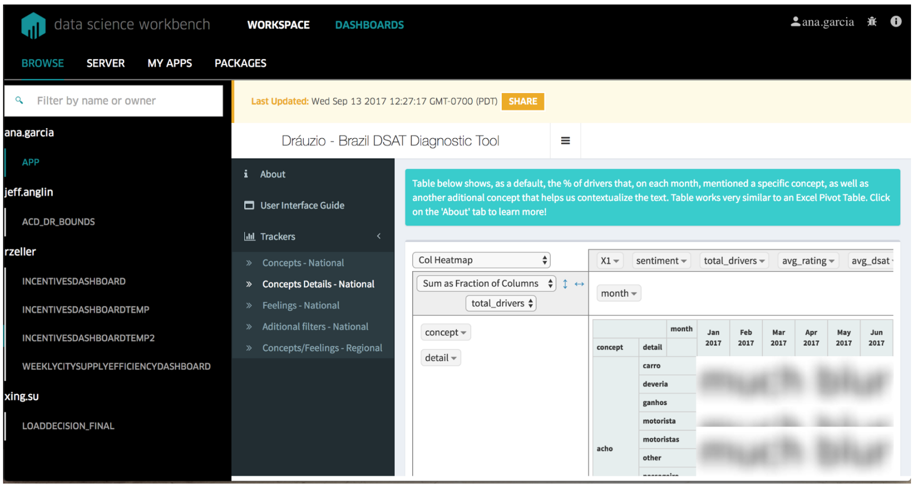

Figure 2.1: A screen shot from Uber's Data Science Workbench.

2.2 NREL TransAtlas

NREL's mission is to research which energy solutions that are the most viable options for a sustainable global infrastructure [6]. To support this they have developed geographical visualization tools to let users explore and apply their data-sets. Questions these tools can answer are, for example, how much electricity can be produced from solar panels in a specific location or what renewable energy sources are available in certain areas. One of their tools, *TransAtlas*, focuses on transportation data. This tool helps users explore where alternative fueling stations are located, among other things. A screen shot of TransAtlas is shown in Figure 2.2.

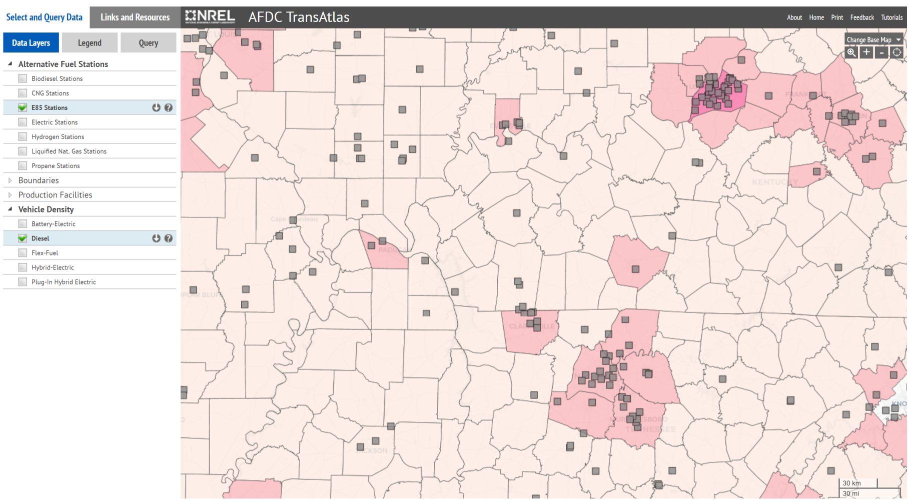

Figure 2.2: A screenshot from NREL's TransAtlas, showing gas station locations and diesel vehicle density.

2.3 Tableau

Tableau is a very common tool for data analytics and business intelligence [7]. Their primary product, Tableau Desktop, allows the user to explore data sets with a big variety of diagrams and options. The system also supports creation of interactive diagram visualizations. An example screen from Tableau displaying geographical data in a choropleth diagram is shown in Figure 2.3.
One of the features of Tableau that is most interesting in the context of this thesis is the functionality connected with maps and geospatial data. It is relatively easy to create interactive maps that display different kinds of geodata. Some of the supported geographical diagram types currently supported in Tableau are:

- Choropleth maps
- Proportional symbol maps
- Point distribution maps
- Flow maps
- Origin-destination spider maps
- Heat maps

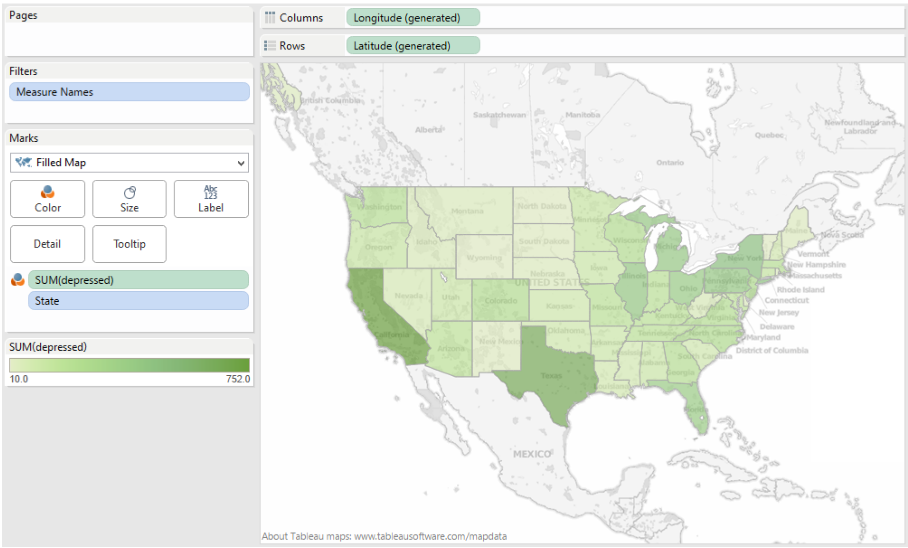

Figure 2.3: A screenshot of a geographical visualization in Tableau.

3

Theory

The main focus of this master thesis is information visualization, in particular geographical data visualization. For this reason, most of the theories described in this chapter comes from the information visualization field. In the end of the chapter a few common geographical visualization types are presented. The nature of these visualization types are described, as well as their benefits and disadvantages.

3.1 Color

Color is likely the most researched field in human perception[8], and some of the conclusions are useful in designing geographical visualizations. Some of the most important aspects of the use of color in that context are described in the sections below.

3.1.1 Color Blindness

According to Colin Ware [8], around 10% of the total population and 1% of the female population suffer from some form of color blindness. The most common types of color blindness result in the inability to distinguish color differences in the red and green color spectrum. To reduce issues connected to color blindness in terms of data visualization, Ware suggests against using both red and green as colors for identification of separate attributes.

3.1.2 Color for Labeling

Colors are often a preferred attribute for nominal information coding, as the alternatives are generally worse, according to Ware [8]. An important aspect when choosing which colors to use as nominal codes are distinctness between the colors. When specific colors needs to be distinguished rapidly Bauer et al. shows that each target color should lie outside of the convex hull of the other used colors in the CIELuv color space [3]. An example of this is shown in Figure 3.1 below.
Even though colors generally are good for displaying categorical information, there is a limit to how many different colors can be distinguished consistently. Healey estimates this number to between 5 and 10 [9]. Except for color choices, the size of the elements also influences the amount of possible color codes. In general, larger elements are more easily distinguished from other elements and the background, which allow for a higher number of color codes.

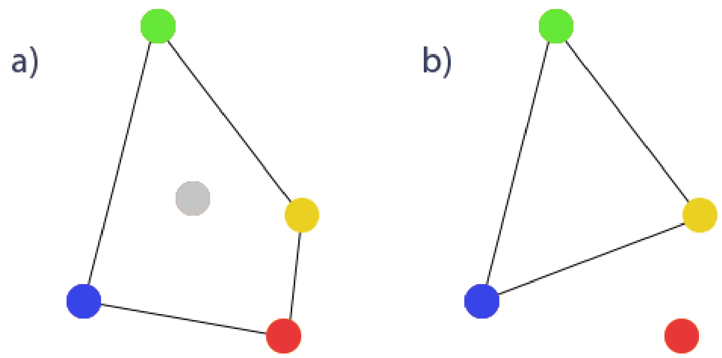

Figure 3.1: An example of distinguishable colors based on the theories of Bauer et al. and the CIELuv color space [3]. a) Gray is inside the convex hull of the other used colors, hence it is not distinguishable together with the other colors. b) Red is outside of the convex hull of the other used colors, so it should be sufficiently distinguishable.

The colors that Ware recommends for coding are red, green, yellow, blue, black, white, pink, cyan, grey, orange, brown and purple [8]. The reasons that these colors are recommended are that they have agreed upon names and are adequately far apart in the CIELuv color space. Some of these color combinations are although less suitable, for example red and green because of color blindness.

3.1.3 Background Color and Distinguishing Elements

It is important to make colored elements stand out from the background. This can be achieved by having luminance contrasts and large chromatic differences between elements and the background. Another way to make the colored elements even clearer is to give them thin black or white boarders. Ware suggests that light, low-saturation colors should be used for the background, and darker, high-saturation colors for the symbols and text [8].

3.1.4 Color Sequences

Color sequences are often used to shown continuously varying values in a map, for example temperature in a weather forecast or vegetation in a geographical map. What color sequence to use depends on the requirements of the visualization. However, the sequences red to yellow, yellow to green and green to blue are all increasing or decreasing on the red-green and yellow-blue color channels [8], which may make them good candidates.
There is a trade-off when choosing the amount of colors to use in a color sequence. By using the whole color spectrum in a heat map, correctly identifying the value of a point becomes easier and less prone to errors [10]. According to a study performed by Ware, there were a 2.5% error chance using spectrum approximation compared to 17% using grey scale. The reason for this is that a color sequence using spectrum

approximation has many more distinguishable steps. Another advantage of using several colors of the color spectrum in a color sequence is that there could be cultural aspects. An example of this is that many people identify red with hot and blue with cold. Because of this a blue to red color sequence may be appropriate when visualizing temperature.

The disadvantage of using a sequence with different colors, but no luminance variation, is that recognition of forms is harder. Studies have shown that using a grey scale color sequence is superior in terms of form recognition, since the luminance channel helps humans to identify forms and patterns. [10][11]

Spence et al. established that the most effective color sequences for identifying high and low points are coded with a combination of hue, saturation and luminance [12]. A good alternative to this may be to use a color sequence with a variety of colors, where each color is lighter than the previous one. Such a color sequence can potentially have the advantage of form identification based luminance difference, as well as the benefit in key value discerning because of the high number of distinguishable steps [10][13][14]. An example of such as color sequence is shown in Figure 3.2.

Figure 3.2: A color sequence where each color is lighter than the previous one, having the advantages of both luminance based and spectrum approximation color sequences.

3.1.5 Contours and Discrete Color Steps

An interval sequence is a sequence where each step represents an interval of values and has a change in magnitude equal to that of all other steps in the sequence. An example of this are contour maps showing height of terrain. Such height contour maps are effective at showing which parts of the terrain has equal height, but lacks in giving a clear understanding of the overall shape of the territory. One way of improving this aspect is to combine contour maps with pseudo-coloring, or perhaps even better, stepped pseudo-coloring [8], as shown in Figure 3.3.

3.1.6 Bi-variate Color Coding

Colors are three dimensional as they vary in hue, saturation and luminance. Thus several variables can be mapped simultaneously. As an example, one variable can be based on hue and another on luminance. While this is possible, maps using bi-variate color coding have been proved to be difficult to read [15]. Because of these difficulties, Ware suggests a solution using color to map one variable and texture for another [8].

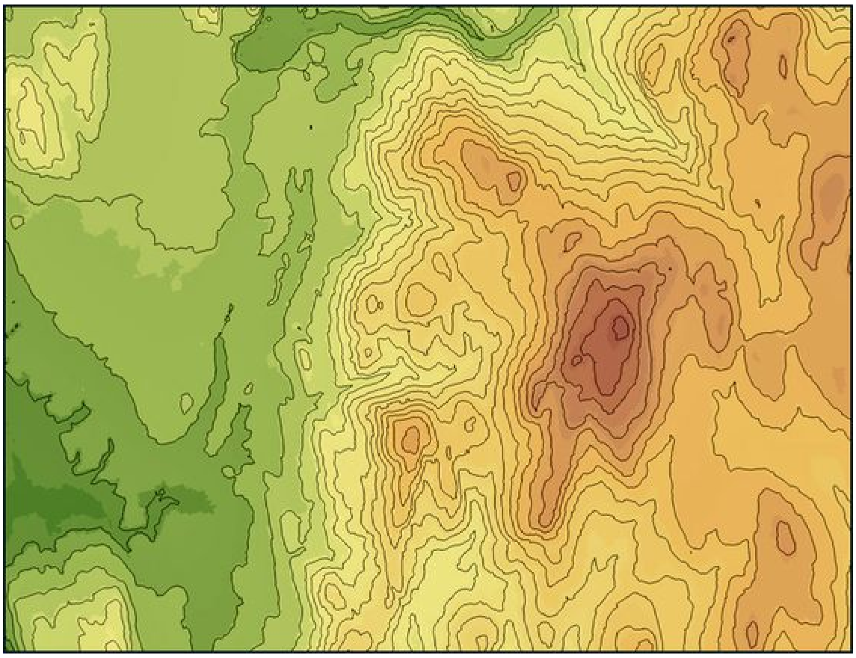

Figure 3.3: A pseudo-colored terrain height map providing effective ways of finding groupings, discerning individual points and overall form recognition.

3.2 Symbols and Text

To make text and symbols clearly visible on top of a background, the spatial frequency of the symbol or text should be distinctly different from the spatial frequency of the background [16]. It is also important to make symbols different from the background in terms of their orientation and form components.

3.2.1 Preattentive Processing

The concept of preattentive processing is that some elements are noticed faster by humans, based on their visual attributes [17]. Scientists first thought that we noticed such elements faster than our conscious attention, but more recent studies show that it is more likely integral. Figure 3.4 shows two graphs, one with and one without preattentive processing.

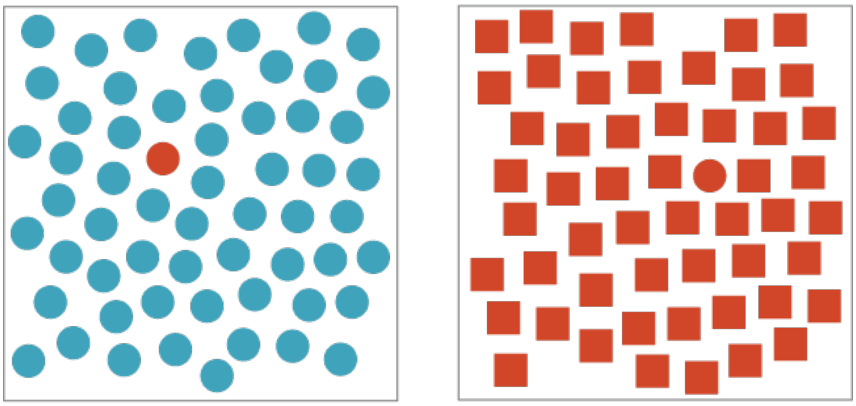

Figure 3.4: The red dot in the graph on the left is more distinct thanks to preattentive processing of color.

The different visual attributes that are preattentively processed can be organized in the following way [8]:

- Line orientation
- Line length
- Line width
- Size
- Curvature
- Spatial grouping
- Blur
- Added marks
- Numerosity (one, two, three objects)
- Color
- Hue
  - Intensity
- Motion
  - Flicker
  - Direction of motion
- Spatial position
  - Two-dimensional position
  - Stereoscopic depth
- Convex / Concave shape from shading

All of the listed visual attributes are preattentively processed, but their preattentive strength differ from each other. For example color, orientation and size tend to have stronger preattentive processing compared to line curvature. The two primary factors for determining how distinct a *target element* is are [18]:

- The degree of difference between the target elements and the non-target elements.
- The degree of difference of the non-target elements from each other.

An example of this is that highlighting a text in yellow can be effective when there are no yellow elements in the visualization. If there are a wide range of yellow objects, such highlighting method would be less effective. As a general rule, to make an element *pop up* to the observer, make it distinct in a particular feature channel.

3.2.2 Redundant Coding

Redundant coding is when elements are distinct with several visual properties, for example both color and size. The exact positive effect in this can be hard to measure, but there is almost always a benefit with redundant coding in terms of visual search [19].

3.2.3 Conjunction Search

A conjunction search is, for example, when an observer has to identify the red circles in a visualization composed of circles and squares, which be either red or blue. These types of searches have been shown to be slow [8]. However, there are some preattentive visual attributes that can be combined, allowing for preattentive conjunction search.

- Spatial grouping on the XY plane [20]
- Stereoscopic depth [21]
- Luminance polarity and shape [22]
- Convexity, concavity and color [23]
- Motion [24]

3.3 Gestalt Laws

Gestalt's laws are a set of principles describing the way humans see patterns in visualizations [25]. One fundamental grouping principle not included in the original Gestalt Laws is *connectedness* [26]. Linking connected elements with lines is a common way of showing relationships. This is for example used in node diagrams. It is easier to perceive the connections if lines with smooth curvatures are used [8].

Closure and common region is a good way of grouping elements that belong to each other, since humans have a strong tendency think in terms of *inside* and *outside* spaces [27]. An Euler diagram is a good example of this principle. To make each region in an Euler diagram more distinct, it is effective to display the regions using color, texture and Cornsweet contours.

3.4 Flow Visualization

One important aspect in visualizations of flows is to clearly show direction. In order to do this, without the use of movement, the vectors visualizing the flow needs to be asymmetrical. Fowler and Ware created a method of doing this by using vectors that are asymmetrical in both size and luminance, providing asymmetry on multiple levels [8].

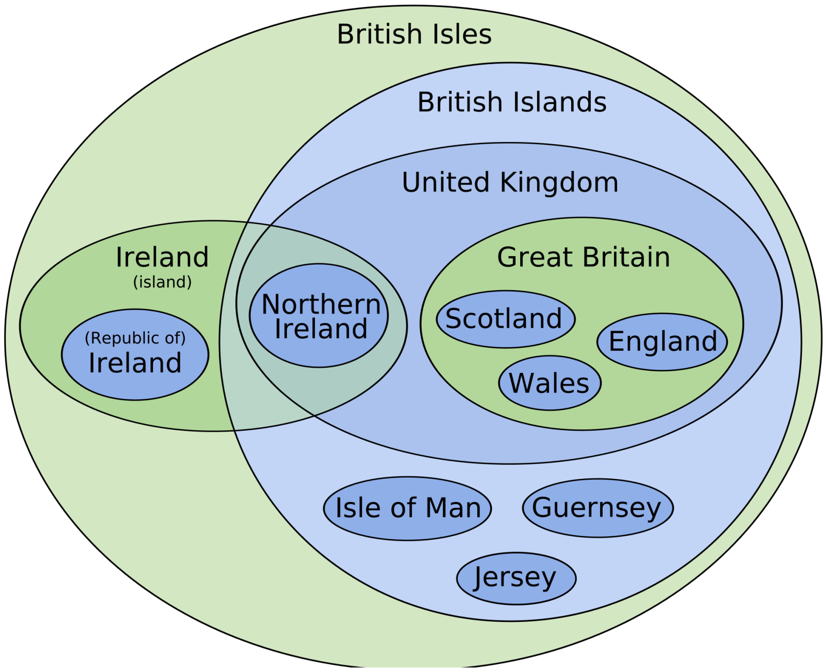

Figure 3.5: Using colors makes each region more visually distinct.

The strength, or magnitude, of the flow is another important feature to visualize. A good way to do this is to use variation in the length and width of the vectors; longer and wider vectors visualize a stronger flow [28].

3.5 Movement and Animation

It is often desired to visualize motion of objects, for example flow of vectors over time. Humans have been shown the most sensitive to motion with speed of 0.5 to 4.0 cm per second, when the motion is happening on a screen at normal distance [29].

3.5.1 Discerning Animated Elements

Humans have a inherent ability to understand events and see relationship through motion alone. In an experiment performed by Gunnar Johansson [30], it was shown that different humans consistently could identify activities such as walking or dancing through observing moving white dots. Johansson's experiments also showed that humans were able to identify such activities in a tenth of a second. The test participant could also identify additional information such as genders and emotional states.

3.5.2 Animated vs Static Visualizations

There have been several studies with the goal of measuring the effectiveness of animated visualizations compared to static ones. Many of these studies have not been able to identify any strong advantage with animated visualizations [31][32].

In one study on this subject, performed by Mayer et al. [32], the visual and static visualizations contained precisely the same information. The results of this study showed that participants observing the static visualization retained information better and had greater ability to generalize about the learned concepts, compared to the animated visualization.
Another study by Palmiter et al.[33] compared instructions to procedural tasks; one of the instructions was animated while the other was written text. It was shown the animated instruction provided better short term results, but that the observers of the written instruction retained the knowledge better. Their theory about the cause of these results was that observers of the animated instruction could mimic the animated characters, providing better result in the short term, while interpreting the written instructions embedded the knowledge deeper in long term memory, produced superior results in the long run.

3.6 Interaction

According to Colin Ware [8], a good visualization allows the user to observe data at several levels of detail. The user first observes an overview of the data, and zooms in to better understand a particular detail. Ben Shneidermann created a guideline for visual information seeking interface design, known as *Shneidermann's Mantra* that goes: *Overview first, zoom and filter, then details on demand* [34]. A good computer-based visualization interface should dynamically display different types of data on different level of detail, based on the user's needs.
A user exploring an interactive visualization is generally engaged in three different cognitive tasks at different levels [8]. These levels / tasks are:

- Lowest level: Selecting and manipulating data through hand-eye coordination.
- Intermediate level: Exploring and grouping data.
- Highest level: Data analysis, including formation and refinement of hypotheses.

3.6.1 Hover queries

A common and simple epistemic action is to hover the mouse cursor over an element to display additional information connected to the element. To prevent the information from being displayed when the user moves the mouse cursor over several elements is to add a small delay. The effects of actions triggered by hover queries should generally not be overly distracting [8].

3.6.2 Learning Systems and Interfaces

As with any system, the operators normally become more skilled the more they used it. Card et al. described *The Power Law of Practice*; a simple expression describing how the speed of doing a task is increased over time of doing the task.

$$\log T_n = C - a \log n$$

where $\log T_n$ is the time it takes to perform the n:th trail, a is a constant representing the learning curve steepness and $C = \log T_n$ is based on time it takes to perform the first trial [35].

3.7 Data Science

Data Science is a field of scientific methods and systems to acquire knowledge through analysis of often big quantities of information, referred to as data. The activity of data science can be broken down into three stages: *design for data, collection of data* and *analysis of data* [36]. The goal of data science is to analyze and understand real phenomena with data, or as Hayashi write on page 41 in the conference proceedings *Data Science, Classification, and Related Methods*: *"...the aim of data science is to reveal the features or the hidden structure of complicated natural, human and social phenomena with data from a different points of view from the established or traditional theory and method. This point of view implies multi-dimensional, dynamic and flexible ways of thinking"* [36].

3.8 Geographical Data Visualization Types

There are a wide variety of ways to visualize data on maps, and each type of visualization has their own advantages and drawbacks. In this section some of the more common ways of visualizing geographical data is described.

3.8.1 Heat Map

A heat map is a way of visualizing the general shapes and concentration of geospatial data by the use of different colors on a map [37]. Heat maps displays data as being continuous, even though the data may itself may not be continuous. This can make heat maps a poor visualization option for data that does not vary continuously. The creation of heat maps involves algorithmic extrapolation, so the exact value at any given point may not be reliable [38]. The main advantage of heat maps is that distributions and shapes are interpreted relatively easily. An example of a heat map visualization is shown in Figure 3.6

3.8.2 Chloropleth Map

A chloropleth map visualizes data by shading defined regions with different colors. Each color represents a specific value or a range of values [39]. This types of visualization is good for intuitively showing general patterns and distributions between areas. The downside of chloropleth maps is that the difference in size between regions can be miss-leading, and that they generally have a low level of details as the regions tend to be quite large in the context. Figure 3.7 shows geographic data visualized as a choropleth diagram.

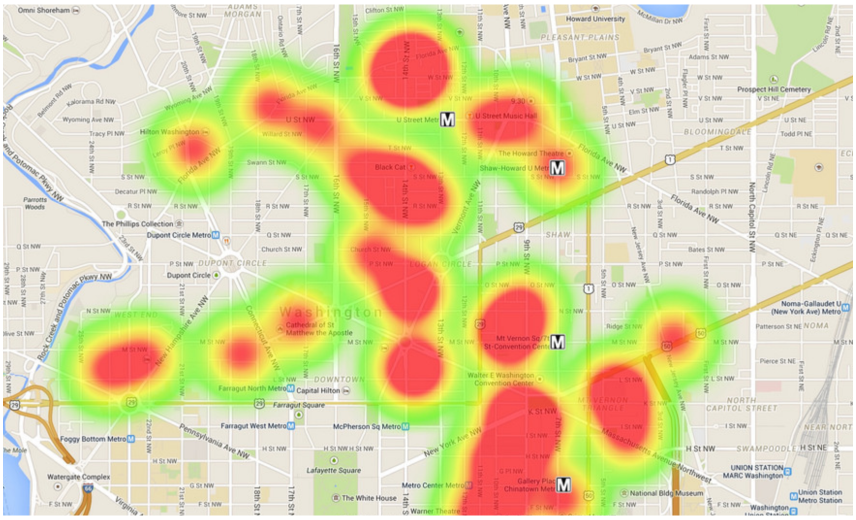

Figure 3.6: An example of a heat map showing the density of gyms in Washington DC.

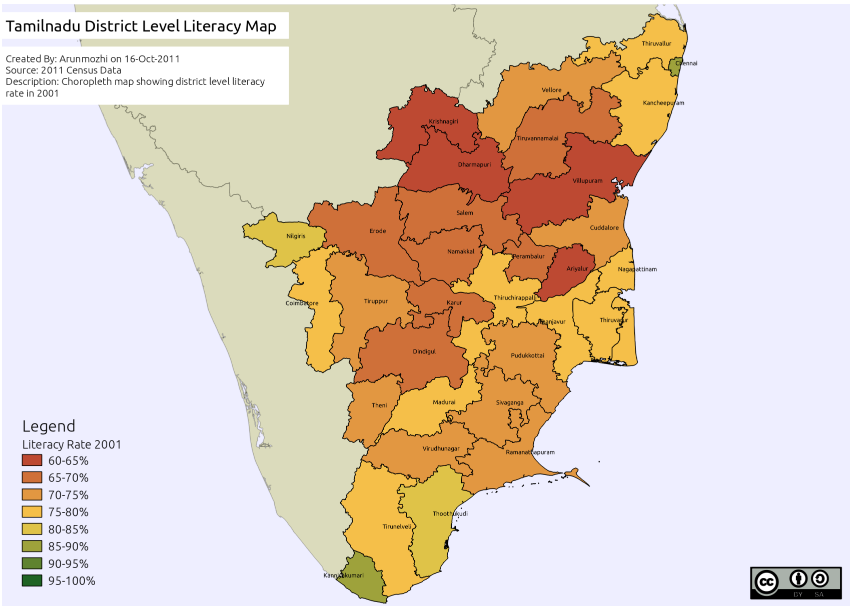

Figure 3.7: A choropleth map from Tableau.

3.8.3 Hexagonal Binning

This visualization method is based around a hexagonal grid in a map of an area. Each hexagon in the grid can then be colored by the same logic as a chloropleth map [39]. Since there is no extrapolation in this type of distribution, the accuracy of a point in the visualization tend to be more exact than, for example, a heat map. The size of the hexagons can be decreased for greater level of detail. An example visualization of hexagonal binning is displayed in Figure 3.8.

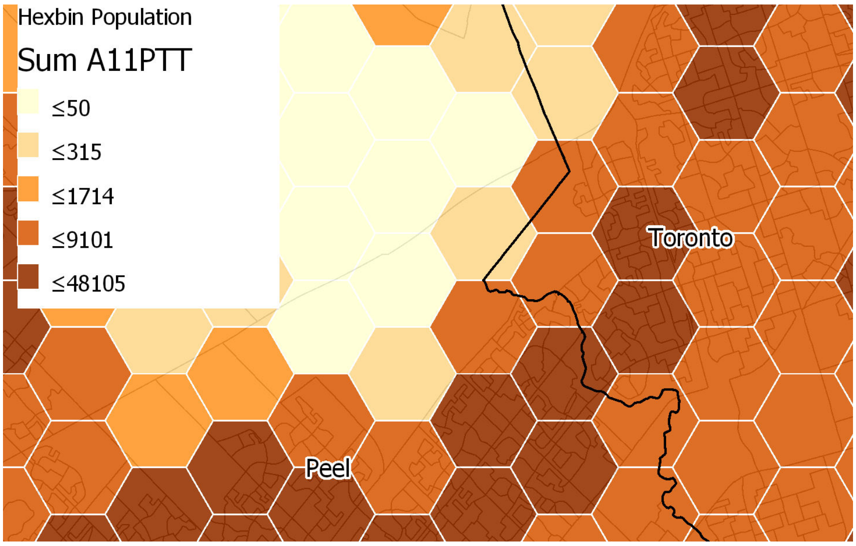

Figure 3.8: An example of hexagonal binning.

3.8.4 Point Distribution Map

A point distribution map (also called a dot map) represents a data point with a dot on map; it is basically a scatter-plot on a map. Point distribution maps are useful for showing spatial patterns such as clusters of data points.
A dot map (also called dot distribution map or dot density map) uses a dot to indicate the presence of a variable. Dot maps are essentially scatterplots on a map and are useful for showing spatial patterns. Each dot represents the precise location of a variable, so it provides a high degree of accuracy. As several dots can overlap it is less advantageous for showing densities, even though it can give a general understanding of relative density between areas. Figure 3.9 presents an example of a point distribution map.

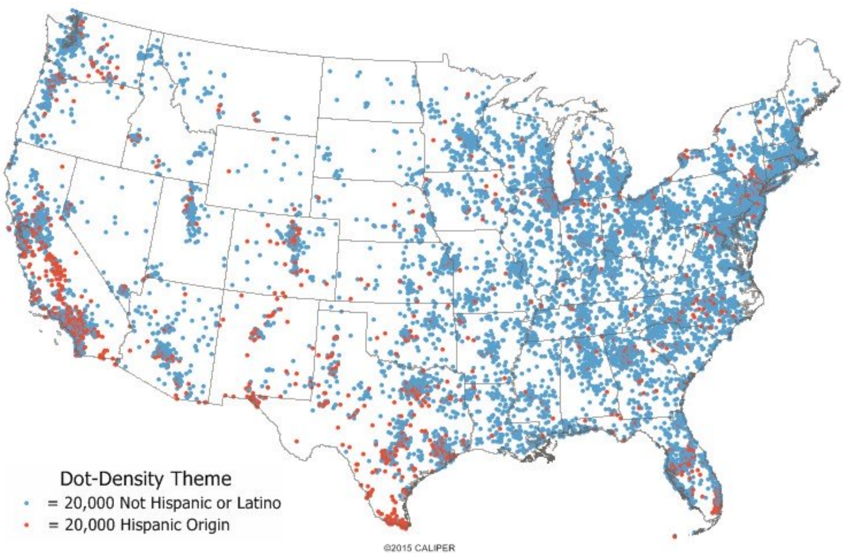

Figure 3.9: An example of point distribution map.

3.9 Visualizing Geographical Data over Time

According to Nick Rabinowitz, a data visualization developer specialized in visualizing geographical data over time, there are several challenges with visualizing geodata over time [40]. There is generally a trade-off between good geographical display and showing data trends over time. Rabinowitz has developed a javascript framework, timemap.js, for displaying geographical data over time [41]. A few examples of the most common approaches are provided at `http://www.nickrabinowitz.com/projects/sxsw/sxsw_slides/`. According to Rabinowitz, *Many good visualizations combine a number of these techniques (but use too many and it gets overwhelming)*[40]. In Figure 3.10 an example where a slider provides the possibility of filtering a point distribution map in terms of time is presented.

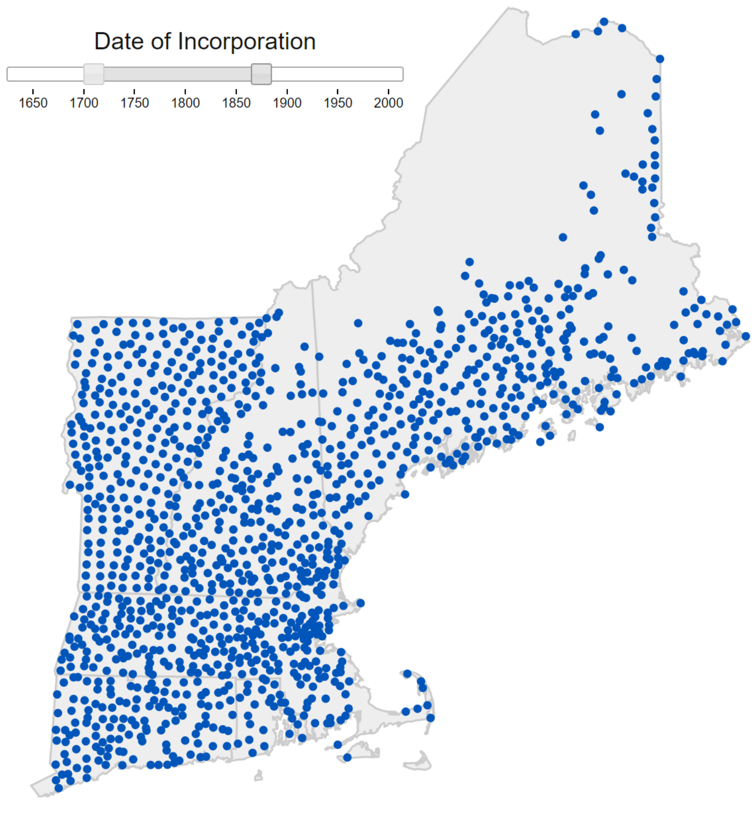

Figure 3.10: An example of a point distribution map filtered by time.

4

Methodology

The methodology section describes work processes, techniques, approaches and design research methodologies. It is a mix between methods which *will* be used and methods which *could* be used. In the end of this section, the chosen work process of this master thesis is presented.

4.1 Work Processes

In this section different project work processes are described. Some of the processes are aimed specifically at design projects (Goal Directed Design, John Chris Jones Design Methodology), while others are more generally applicable (Kanban, Scrum).

4.1.1 Goal Directed Design

In the book *About Face*, Alan Cooper et al. outlined a simple premise: *"If we design and construct products in such a way that the people who use them achieve their goals, these people will be satisfied, effective, and happy..."* [42]. With this premise as foundation Cooper et al. developed the design process goal directed design. They argued that many of the current digital products lacked in understanding user needs, has conflicting interests and did not have a clear design process. To create good digital products a deep understanding of its users as well as cognitive principles are required. This can be done in a systematic way, and perhaps the most critical aspect of this is *understanding the user's goals* [43].

User goals

The goals of a user can often be ambiguous and not what designers first assumed them to be. They tend to be more personal than business oriented. An example from the book *About Face* is the goals of an accounting clerk [42]. It is natural to assume that the main goal of the clerk is efficiently process invoices, but in reality the clerk is more interested in things such as:

- Appear competent at his work and avoiding mistakes.
- Keeping himself occupied and stimulated while doing repetitive tasks.
- Becoming more competent in his field.
- Set a good example for others.

A good system needs to address both personal and business goals. When the personal goals are met, users will much more effectively achieve the goals of the business [42].

Difference between goals, tasks and activities

A goal is the end condition, while tasks and activities are steps that needs to be taken in order to achieve the goal. Influenced by the work of Donald Norman and Activity-Centered Design [44], the hierarchy is as follow:

1. **Goals:** The desired end condition, driven by human motivations.
2. **Activities:** To reach the goal, one or several activities needs to be performed. Normally based on the technologies and tools at hand.
3. **Tasks:** Activities can be broken down into tasks, which are smaller steps included in an activity.
4. **Operations:** A task is broken down to the smallest component: operations. These can be as simple as moving the mouse or clicking a button.

Goal Directed Design Process Overview

The Goal Directed Design process can be broken down into six sequential steps. These steps can be repeated in an iterative manner, and previous steps can be re-explored to accommodate for new information. These steps are:

1. **Research:** Understand the users and the domain.
2. **Modelling:** Model the users and context of activities.
3. **Requirements:** Define user, business and technical needs and restrictions. The basic concept of the design is created in this step.
4. **Frameworks:** Define the overall structure and flow of the design.
5. **Refinement:** Refine behaviours, content and form.
6. **Support:** Aid the development process.

4.1.2 Requirement Definition Process

Requirement Definition Process is a process for coming up with and prioritize the requirements of the design developed by Cooper et al. [42]. This process consists of five steps, of can be performed several times iteratively:

1. Creating a problem- and a vision-statement. This is done to outline the direction of the design.
2. Brainstorming
3. Identifying user expectations and mental models.
4. Creating context scenarios. These are stories about how users interact with the product on a high level, and focuses on the *what*, not the *how*.
5. Identifying requirements.

4.1.3 John Chris Jones Design Methodology

Jones breaks down the design process into three steps: *Divergence*, *Transformation* and *Convergence* [45].

Divergence

The initial phase in Jones Design Methodology is the Divergence phase. This stage revolves around expanding the design space to have, according to Jones: *"...a large enough,and fruitful enough, search space in which to seek a solution"* [45]. At this phase in the design process the final goals and requirements are still unstable and unclear. A lot of effort goes into finding information and understand contexts and relationships. Important questions in the Divergence phase are:

- What is valuable?
- What are the risks?
- What is feasible?
- Who are the stakeholders and users?
- What relationships and dependencies are there?
- Are the right questions being asked?

Transformation

When the research phase is close to done and a model of the domain and its users is developed, the design process enters the Transformation stage. The primary activity in this phase is what Jones calls Pattern-Making: *"...the creative act of turning a complicated problem into a simple one by ... deciding what to emphasize and what to overlook"* [45]. In practice this revolves around identifying the most critical aspects, recognize important constraints and define a foundation for the design. Another important task in the Transformation stage is to divide the problem into smaller sub-problems, which optimally can be solved in isolation by different team members. Jones also states that it is important to choose one, clear design solution instead of a mix of several solutions.

Convergence

The final stage of the design process begins once the problem and overall design solution have been clearly defined. There should be a good understanding of important variables, constraints and objectives. It also preferable that the relative priority of these elements are agreed upon.

The goal of the Convergence phase is to create a concrete and detailed design. This stage is the most rational and concrete phase in the design process; vagueness and flexibility are undesired. If critical, unforeseen problems arise, the best solution is to shift back the design process to the Transformation stage.

4.1.4 Kanban

Kanban is development process and production control system developed by the Toyota Motor corporation [46]. It is a system created for *just-in-time* production and aimed to reduce the overhead costs of production control and adhere to rapidly changing information. Kanban revolves around the concept of a *Kanban Board* to visualize the work flow, as shown in Figure 4.1. The process can be broken down into four core principles [47]:

1. **Visualize Work:** Visualizing the flow of the work leads to increased communication and more synchronized mental models. Tasks that are blocking, acts as bottlenecks and tasks queues can be identified faster and easier.
2. **Limit Work in Process:** The cost of stitching between different tasks has been researched and proven lower efficiency [48][49]. Kanban decreases this issue by lowering the amount of unfinished work tasks in the pipeline at any point in time. This also makes continuous improvements to the system more steady.
3. **Focus on Flow:** By lowering the amount of WIP and having a relatively constant flow of working task in the development pipeline, data metrics such as throughput and bottlenecks can be collected.
4. **Continuous Improvement:** Once the Kanban system is integrated in the work process the teams efficiency can be measured through analysis of collected data. New policies can be tested and evaluated.

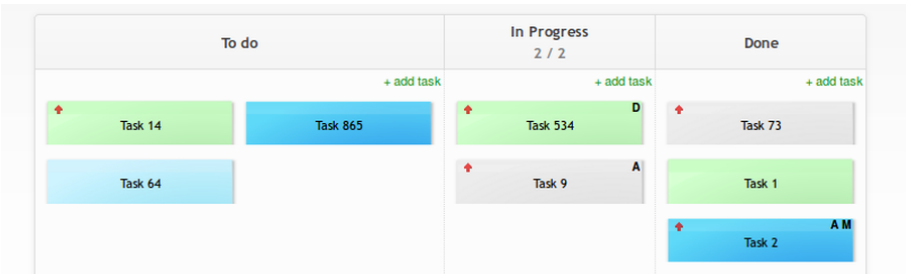

Figure 4.1: A basic Kanban board.

4.1.5 Scrum

Scrum is an extension to the iterative/incremental object oriented development process. It is a work development process that assumes that system development is inherently unpredictable and treats different parts of the process as controlled black boxes [50]. This makes the Scrum development process flexible and able to deal with unpredictable results and changes in requirements.

Phases of Scrum

A project developed with Scrum can be divided into three major phases. These phases are:

1. **Planning Phase / Pregame:** The initial phase of the Scrum development process. In this phase the high level requirements and system architecture are

explored. A high level backlog is created, including a rough definition of how these backlog items will be implemented. *Definition of Done* is defined, and costs and a time schedule are estimated.

2. **Sprints / Game:** The majority of the time in a Scrum development process are spent in this phase. It consists of multiple *sprint* cycles: a fixed time period with development goals revolving around specific functionality. Each of these sprint cycles attempts to respect and work around the changing variables of requirements, cost, time and quality.
3. **Closure / Postgame:** The final phase which focuses on finalizing the project. This phase includes activities such as final documentation, release preparation, final testing and product release.

4.2 Techniques and Approaches

In this section different design techniques and approaches which are relevant to the project, and design projects in general, are described.

4.2.1 Affinity Diagrams

This technique revolve around organizing design elements and concepts in clusters in order to identify relationships. The clusters can be based on things such as similarity, dependence, characteristics or proximity. This method can reveal design opportunities and provide a model for design elements [51].

4.2.2 Design for Intermediates

In general, the experience level of users of a system tends to follow a traditional bell curve common in statistics [42]. There are few beginners and experts and a majority of intermediates, users that have some experience with working with the system but do not know enough to be classified as experts. A big reason for this is that beginners tend to move onto the intermediate group quite quickly, but intermediate users often stay in that category during the complete lifetime of the system. This is also true, although to a lesser extent, for expert users; many do not not maintain their expertise over long periods of time. Based on this it is a good design principle to optimize the system for intermediate users [42]. A good of example of an UX element that fit well with intermediates are *Tooltips*. *Tooltips* do not explain the scope or purpose, but provide a short description of the function.

4.2.3 Five Whys

Five Why's is a simple technique used by IDEO [51]. By asking *Why?* to five consecutive answers, the underlying reasons for a behaviours can be explored. This can be used during interviews as well as brainstorming sessions.

4.2.4 Interviews

Interviews is one of the most common techniques for acquiring qualitative data in design research [42]. Interviews can be roughly categorized depending on the interviewee:

- **Stakeholder interviews:** A stakeholder often is a person with responsibilities or authority over the product or service being designed. Interviews with different stakeholders is important in order to understand business and technical context in the domain of the design. Important questions that stakeholder interviews may answer are:
  - The vision of the product.
  - Does the stakeholders have the same mental vision?
  - Schedule and budget of the design.
  - Technical constraints and opportunities.
  - What are the business goals?
  - Useful insight of the users.
- **SME interviews:** Subject-matter experts are often expert users in the domain of the system. As stakeholders, they can have valuable insight into desired requirements and user behaviour. However, it is important to remember that SME's can have complex mental models due their long expertise, and may focus on aspects that are too advanced for the average user.
- **User interviews:** The potential users of a product are generally the most important group to focus on during the design process. It is they who will use the system to achieve their goals. Some useful information designers can learn from user interviews are:
  - How will the product fit in the context of their lives? How, why and when will it be used?
  - What is important for the users to know in the domain?
  - How are the users currently performing these activities?
  - What are the user motivations and goals for using the product?
  - What are the expectations and mental models of the activities of the design?
  - Problems with currently used products.

4.2.5 Paper Prototyping

One of the fastest way to communicate to articulate and visualize interaction design concepts is to rapidly sketch them using pens and paper. This is often good as an initial prototyping method when the focus is on basic usability [51]. It allows for quick iteration and communication about the design within a team.

4.2.6 Scenario Testing

A testing technique that can be combined with paper prototyping. By showing the users a set of *screens*, and allowing them to interact and share their reactions, early and rough concepts can be evaluated [51].

4.2.7 Summative User Testing

A type of user test which focuses on quality assurance of a design. Summative user testing requires working prototypes of the design, and is therefor primarily done during the later stages of a design project. A goal of summative user testing is often to measure metrics such as how long time it takes for a user to finish a task, or the success rates of users for finishing a task. [52]

4.2.8 Moscow Method

A requirement prioritization method which prioritizes requirements as *Must have, Should have, Could have* and *Would have* [53].

4.3 Design Research

This section covers methodology in design research projects. In his book *Do it yourself social research* Yoland Wadsworth outlines several points which are present in good social research [54]:

- Focuses on the central research question.
- The shape of the research plan has a *shape* that strikes a balance between flexibility and structure. There should be rough step-by-step tasks that drives the research, and these should be connected to research question and its purpose.
- The purpose of the research, and the context of the purpose, should be clear.
- A focus on essential, simple and unambiguous questions.
- Involves the correct stakeholders.
- It should be clear what the evidence is able to eventually conclude.
- A high degree of integrity. An unbiased devotion to find the objective truth, even in minor details.
- Is understandable and can be translated into action.

4.3.1 Scientific Standards in Design Research

Design fields have becoming increasingly integrated with HCI. This has produced questions in the design community whether there are sufficient standards in research through design [55]. In the paper *What Should We Expect From Research Through Design?* William Gaver argues that implementation of such standards should be approached with caution [56]. Development in the design field may come from increased agreement, but also derive from discursiveness and elaboration. As such, to strive from consensus and disciplinary norms of process may constrain the development of design research.

Design theories tend to be provisional, contingent and aspirational in their nature [56]. Gaver argues that this can be an advantage, and that the foundation of the field is *an endless string of design examples*. By grounding the theories in sets of detailed design examples, something Gaver refers to as annotated portfolios, an area in the design space can be established. These annotated portfolios allows, according to Gaver, theoretical work to be viewed from multiple perspectives.

5 Planning

The different work processes described in the Methodology section are not mutually exclusive. Goal-directed Design and Jones Design Methodology are ways of organizing a design process in different phases, each with specific activities and goals. Kanban and Scrum on the other hand, are more general work processes for prioritizing tasks and controlling the work-flow.

In this master thesis the design process foundation will be based on Jones Design Methodology. Jones' three phased design processed has worked well in many typical design projects. Since this master thesis share many of the characteristics with a design project, it is probable that Jones design process will provide a good foundation. However, certain techniques from Cooper's Goal Directed Design process will also be used, such as the *Requirement Definition Process* [42].

A light weight version of Kanban will be the work-flow process used during the project. It is primarily visualizing the work-flow, as well as limiting concurrent working tasks, that are the benefits of Kanban desired in this project. The main reasons why Kanban will be used are its flexibility and that it is suitable for projects with few team members.

5.1 Time Plan

This master thesis will consist of three phases: a planning phase, a design phase and report finalization phase. A visual overview of the time plan is presented in Figure 5.1, and the length and focus of these phases are described below.

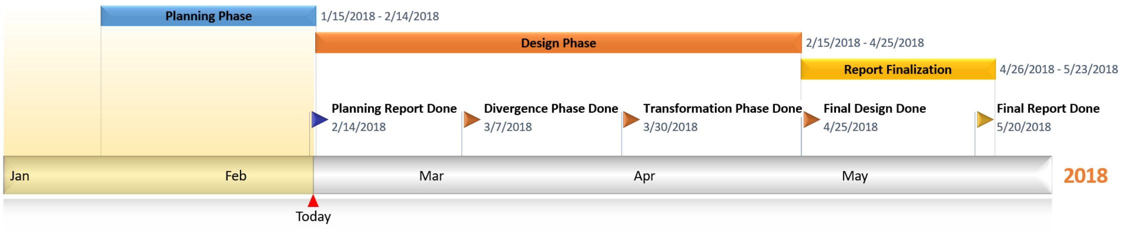

Figure 5.1: A general time plan for this master thesis.

5.1.1 Planning Phase (week 1 - 5)

The initial phase of the master thesis where efforts will be focused on planning the thesis work and researching relevant areas.

5.1.2 Design and development phase (week 5 - 17)

During the design and development phase most energy will be put into designing and developing a prototype for the system. At this point I should have a lot of theoretical background information from the planning phase. As Jones Design Methodology will be the foundation of the design work in this thesis, this step can be broken down into Divergence, Transformation and Convergence.
In the beginning of this phase most focus will be on design, will during the end development will be prioritized. However, both design and development will be intertwined in an iterative manner during the whole duration of this stage of the master thesis.
The report will also be continuously updated during the full duration of this phase. The text will likely be rough and informal at this stage, as much of the purpose is to just document progress and interesting challenges and observations. The report text will later be polished during the last four weeks.

5.1.3 Report finalization phase (week 16 - 21)

At the final stage of the master thesis the focus will be on finalizing the report, performing a report presentation and opposition.

- Finalize report
- Presentation
- Opposition

6
Execution

The Execution chapter describes the ongoing and iterative design process of this master thesis. This master thesis design project will be based on John Chriss Jones Design Methodology, so the sections will be mapped to the different design phases: *Divergence, Transformation* and *Convergence.*

6.1 The Divergence Phase

The initial phase of the design process is the Divergence Phase, based on Jones' Design Methodology. The first activity of the Divergence Phase was to get a clear understanding of what questions requires answering. Through reading different literature and talking to the supervisors of the project *(Johan Wingård, Olof Torgersson)*, the following list of questions was produced:

- Understand the domain of the design: Bzzt's operations. The domain should be understood on an overview level, but also specific areas and how they relate to each other.
- Understand goals, motivations and problems of the Stakeholders.
- Get a specific understanding of who the users are.
- Understand goals, motivations and problems of the users.
- Understand existing similar products and solutions. What are their benefits and disadvantages?
- Understand the overall purpose of the design.

The second activity was to plan which methods that should be used to answer these questions. To get a good understanding of existing tools, the chosen approach was to research the tools online as well as, when it was possible, trying them out myself. The primary method of answering the other questions was interviewing users and stakeholders. The interviews were extended with an additional task: the interviewee had to rank 13 potential data-points, from most to least important.

6.1.1 Interviews

Interviews were the most important method for acquiring information in this design process. The reason why interviews were particularly useful was that there were few potential users for the tool. Additionally, many of the potential users were also stakeholders of the design.

Before any interviews were performed the complete interview approach was planned. Based on information primarily from Cooper et al. *About Face*[42], an interview methodology was outlined. Some examples of the takeaways from researching interview theory are:

- Focus on goals before tasks.
- Avoid leading questions.
- Avoid a fixed set of questions; let the answers of the interviewee guide the direction of the interview.

Johan Wingård emphasized the importance of focusing each interview on the context of the interviewee's role and competence, as well as some additional important aspects when conducting user interviews. Based on the information from literature and Wingård, interview templates for each stakeholder and user were created. The interviews were approximately one hour long.

6.1.2 Data-point Prioritization Exercise

In the beginning of the Divergence Phase 13 different data points were identified as potentially relevant to the tool. This was done by informal discussions with stakeholders at Bzzt and researching other similar business, such as Uber and Lyft. These data points are:

- Destination
- Waiting Time
- Pickup Spot
- App Openings
- Pickup-to-destination Path
- Vehicle State
- Vehicle Driving Path
- Vehicle duration in commercial zone.
- Vehicle Efficiency (% of time occupied)
- Trip Efficiency (duration / cost)
- Amount of people in the streets.
- Impressions
- Vehicle Location

Each of these data-points were written on a post-it notes. After the interview, the interviewee ordered these data-point notes by how important he or she considered supporting them where. A photo of the arranged post-it notes is shown in Figure 6.1.

6.1.3 Requirements Creation

After all of the interviews were concluded, the final stage of the divergence phase was to synthesize all of the information into a list of requirements, constraints and important aspects. This was done by going through all the interviews and identifying goals, patterns and recurring problems.

A long list of requirements and potential data points were created through this approach. The requirements were not categorized or prioritized, since those were tasks which were planned to be done later in the *Transformation* phase. A couple of examples of requirements and data points are:

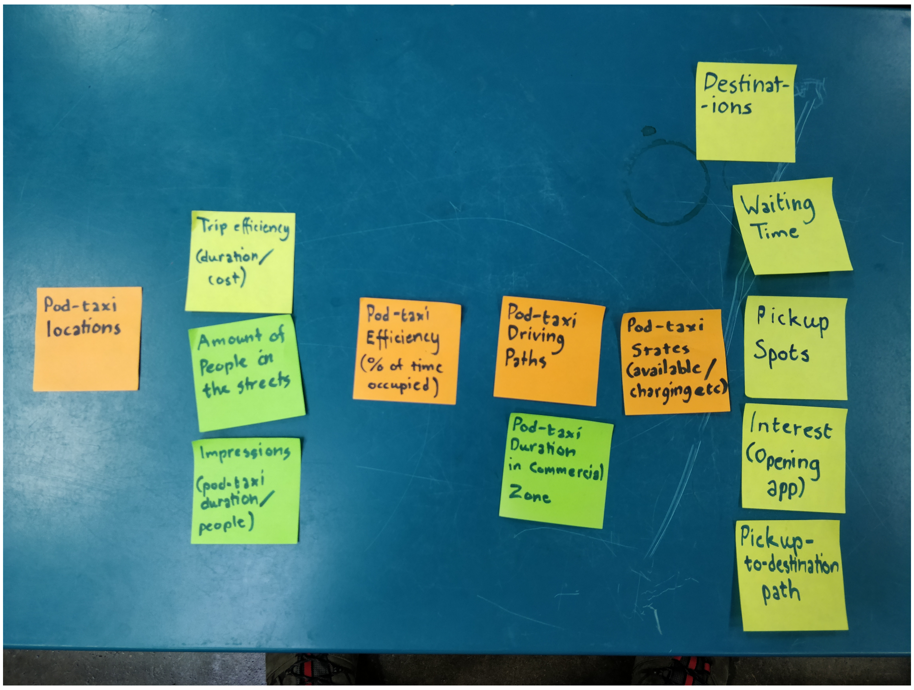

Figure 6.1: An example of data points prioritized by importance according to the interviewee. The most critical data-points are to the right, while the least important ones are to the left.

Requirements

- Simple to use and intuitive system.
- Possible to combine data points.
- Showing the same data with different granularity and in different ways.
- Possible to identify patterns over time.

Data Points

- Estimated time (point).
- Pickup location (point).
- Flow of vehicles (path).

6.1.4 Target User Identification

Apart from the list of requirements and data points, the interviews provided several insights and helped in making a variety of design decisions. A list of the discoveries from the Divergence phase can be found in appendix D. One major decision the interviews generated was specifying the target user.

It was decided that the primary target users of the design are non-technical Bzzt employees which are directly involved in making business decisions. A secondary target user was also identified as a non-technical employee who is involved in business and operational discussions, although making business decisions is not their primary responsibility. An example of a primary user is *Sven Wolf*, CEO of Bzzt, and a secondary user is Anders Hegefors, creative and partner at Bzzt.

The reasons for these decisions regarding the targets users originated from several insights discovered during the stakeholder interviews. A crucial reason is that several interviewees expressed that it was difficult to acquire and explore the data which Bzzt currently possessed. There was a need for data exploration that did not require significant work and resources.

6.2 The Transformation Phase

The transformation phase is the most creative phase in Jones' Design Methodology [45]. It is during this stage that the actual design start to take shape. The general goal of the transformation stage is to transform all of the information acquired in the Divergence phase into the foundation of the design.

6.2.1 Requirement Definition Process

One of the main activities during Transformation phase was to perform a Requirement Definition Process. The end results of this process was a list of concrete requirements. These requirements were also prioritized using the *Moscow Method*. Roughly 15 - 20 requirements within each category were identified. A few examples of the most critical requirements are (the complete requirement list can be found in Appendix A):

- View data points on a map with zoom functionality.
- Filter data on time.
- See data in different aggregation and geographical granularity.

Additional to requirements, the Requirements Definition Process also produced several design concepts. This was mostly the result of a brainstorming session with myself and my supervisor Johan Wingård.

6.2.2 Redefining the Scope

There were a lot of discussions about the specific nature and scope of the design with the Bzzt technology team, in particular the head of technology Johan Lindberg. The main points being discussed during these sessions were how the design would fit the overall data science strategy at Bzzt as well as how much engineering resources could be assigned for implementation of the product.

The result of these discussion was a redefined design scope. Previously the goal of the design were to allow users to perform data queries as well as visualizing and filtering the data in meaningful ways. To limit the scope, reduce the implementation workload and better integrate the design with existing technology, it was decided that the functionality of performing data queries would be discarded. Instead, the product will receive data as input, and provide the user with ways of filtering and viewing the data.

6.3 The Convergence Phase

The transformation phase had produced a list of prioritized requirements. These are the *what* of the design; the focus of the Convergence phase is to figure out the *how*. The chosen approach was to produce quick, iterative sketches and prototypes and receive feedback on these as fast as possible.

6.3.1 First Design Sketch

The initial sketch of the product was made as a simple paper sketch. This was a good choice as the design could be rapidly updated and experimented with. Several designs were produced, each with minor changes and improvements. After a few iterations a higher quality design sketch was produced, which was reviewed by the project supervisor Johan Wingård. Wingård provided feedback regarding several critical design choices, and another sketch was produced which accommodated these points. Some of Wingård's points are:

- It was not possible to compare dates from several years simultaneously.
- A share settings feature would be useful.
- Provide options to remove UI components to reduce clutter on the interface.

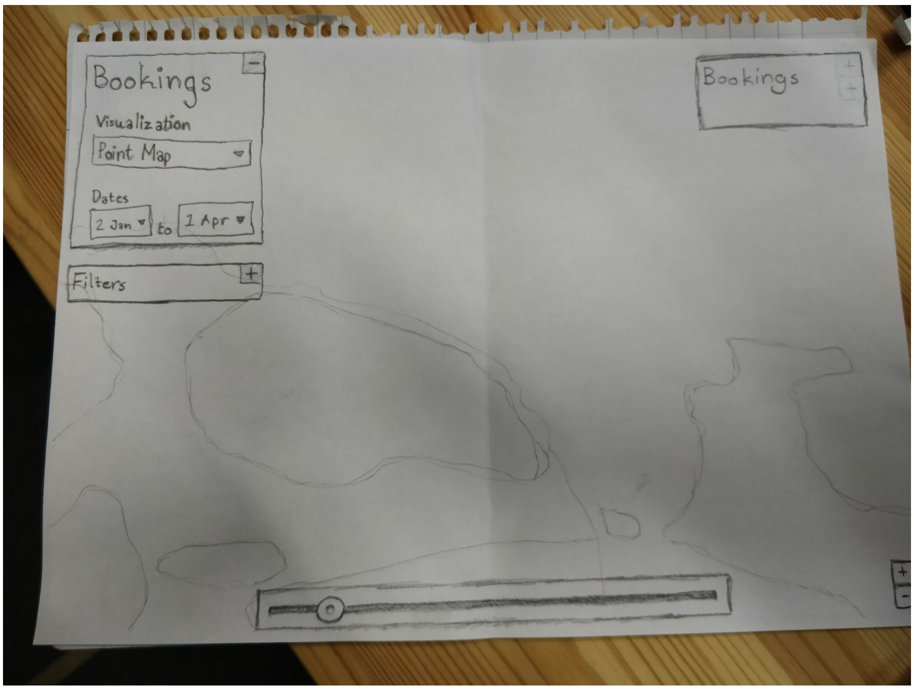

Figure 6.2: One of the first and very basic sketches of the design.

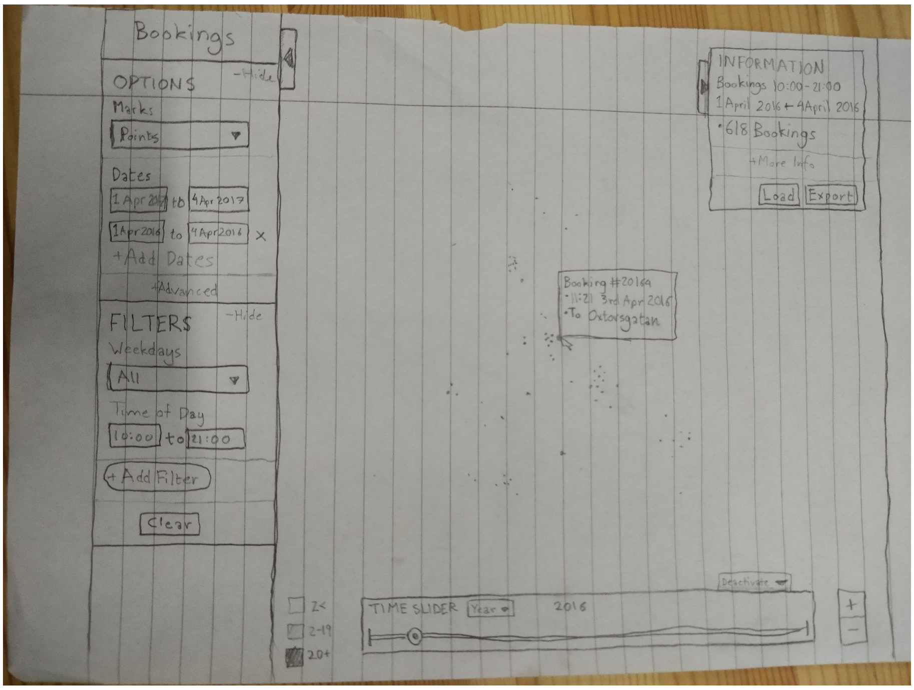

Figure 6.3: A later iteration of the design, updated to accommodate to Johan Wingård's feedback.

6.3.2 Low Fidelity Digital Sketch

Once the general look of the product had been decided upon and visualized as a paper sketch, the next step was to produce a low fidelity digital sketch with basic interactivity. Balsamiq[57] was chosen as the tool to design this low fidelity sketch. The reasons for this were that Balsamiq is a lightweight program which allows for fairly rapid sketching, and the rough look of Balsamiq's wire-frames clearly takes away focus from the look & feel of the UI.

Wireframes of four screens of the product were created in Balsamiq: one main screen and three screens visualizing three different important interactions. The screen sketches were connected with the link tool in Balsamiq, and the interactive design was tested.. These tests were not formal and organized; different people at Bzzt where spontaneously asked to look at the design and give their feedback. The feedback which the design received was generally positive, but with a few concerns and improvement suggestions which were recorded.

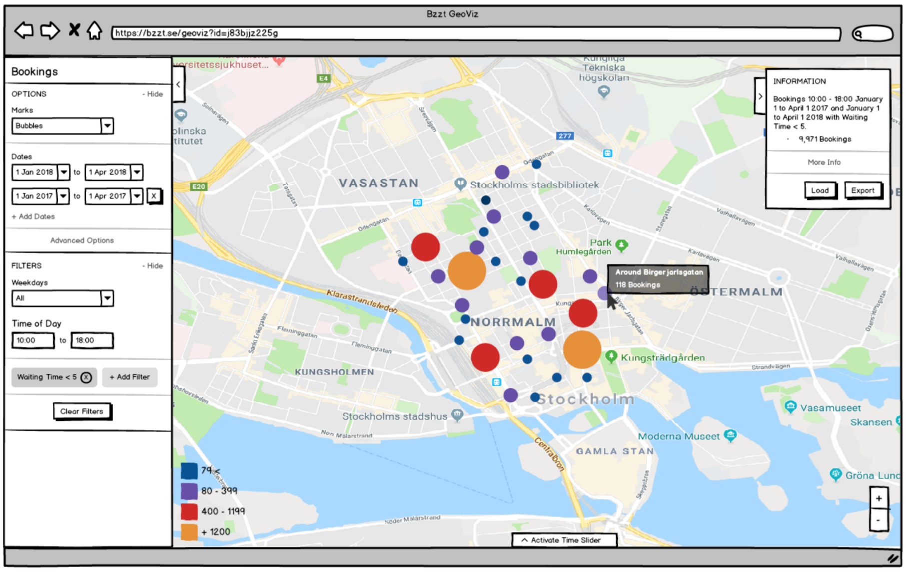

Figure 6.4: The main screen of the low fidelity design made in Balsamiq.

6.3.3 High Fidelity Digital Sketch

The final step in the design process was to design high fidelity wire-frames and screens sketches for the tool. This was done using the tool Adobe XD [58]. Adobe XD is UX/UI tool for designing web sites, mobile apps, software applications and more. One of the benefits of using Adobe XD is the possibility to create interactive prototypes, something which was beneficial for communication and evaluation purposes.

The high fidelity design was based on the low fidelity sketches made in Balsamiq. Some design details, such as button positions and copy, was changed based on the feedback which was received during the user interviews and evaluation sessions performed during the design of the low fidelity sketches.

The sketches made in Balsamiq only showed a couple of central screens of the tool, with a basic and rough look and feel. With the high fidelity screens however, the goal was to design almost all of the different views and interactions. An attractive look and feel was also an objective, although this was not prioritized as highly as the usability and functionality.

The process of creating the high fidelity screens was very similar to that of the low fidelity ones. It was a highly iterative process of design and user feedback. A set of screens and interactions were created and linked together in Adobe XD, followed by an informal interview with a user at the Bzzt office. The purpose of these interview sessions was not get an extensive evaluation of the design, but rather a test to evaluate if the design as a whole was progressing in the right direction. A secondary benefit of these recurrent *talks* with the target users of the tool was ideation: through the frequent discussions small improvements and new ideas were produced.

Over 50 different screens for the high fidelity interface were created in order to design every view and interaction, and to show how different settings could be combined. However, every different combination of settings was not modelled in this way, as it would provide very little value. It is enough to show how one or two examples work in order demonstrate how the interaction functions as a whole.

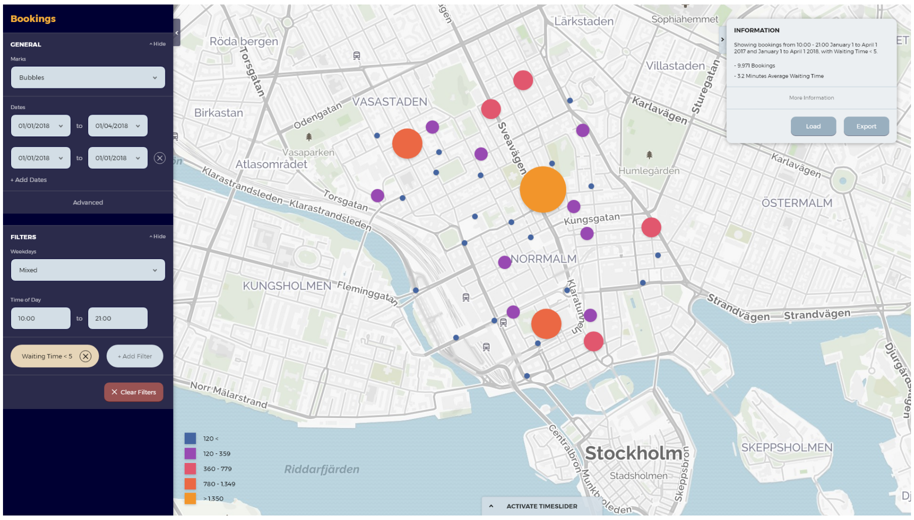

Figure 6.5: An early version of the high fidelity interface.

7

Results

This chapter outlines the results of this master thesis and answers the research questions of this project, which are:
How can the pod taxi company Bzzt's geographical data be presented and explored to allow for business improvements and operational optimization?
The answer to the first research question will primarily be provided as a description of the final version of a design for a geographical data science tool designed for Bzzt AB. The working name of this tool is *BGIS*, which is an acronym for *Bzzt Geographical Information System*, with GIS being a common and universally accepted acronym for tools which displays geographical data.
What are important requirements, aspects and options when designing geographical data science systems for logistical services?
The second research question is answered with a set of general design guidelines for designing geographical data visualization tools for logistical services.
Except for the description of the final design and the design guidelines, this section also describes some of the major insights which were the foundation of the design of BGIS. These insights are specific to Bzzt and not generally applicable to other logistical service domains.

7.1 Design Insights Specific to Bzzt

Doing this project has provided a deep understanding regarding Bzzt's needs in terms of working with geographical data. In this section discoveries which are specific to Bzzt are presented, while insights which are more generally applicable are described in a later section outlining universal guidelines for designing geographical data visualization systems.

7.1.1 Data Accessibility

An important aspect that has become apparent is to make data available to a wide range of stakeholders in an organization. At Bzzt, employees who primarily work with software have a much greater access to data. This has several reasons:

- They develop the systems that collect and handles data.
- They have easy access to these systems in their daily work environment.
- They have a deep understanding in how to write queries and scripts that fetches and handles data.

- Because of experience and the nature of their work, they are more adapt at interpreting data and working in data-driven ways. There are many exceptions to this however, as many non-developers are highly knowledgeable in statistics and making data-driven decisions.

While exploring data is highly important for members of the software team, there are other areas in the organization which are even more dependant on it. The Operations, Marketing and Finance teams can use data to discover insights about the business that is relevant to their particular field, or use data to communicate about real world behaviours to their colleagues, customers or other external stakeholders. Making data easily accessible to the whole organization is therefor highly desirable and this was identified as a primary goal in this design project.

7.1.2 Bzzt's Long-term Data Science Strategy

To achieve easy and complete access to data, Bzzt has created a strategy regarding data which should be implemented continuously as the company grows. Data Science at Bzzt can essentially be divided into two parts:

1. **Technical Data Science**: This type of data science is performed by *Data Scientists*, people who's primary work responsibility is to model, interpret and explore data. It requires a high degree of freedom in terms of querying, combining and modelling data. This work is usually performed mostly through scripting with tools such as *Jupyter Notebook*[59], a web application for working with data and machine learning.
2. **Non-technical Data Science**: As many non-technical people can benefit greatly from working with data, it is important that the systems supports non-technical data science. This is performed by people from operations, marketing, sales, finance and other teams which main responsibilities are not specifically related to data. Non-technical data science is primarily done through the use of graphical user interfaces consisting of sliders, buttons and other similar input methods to filter the data. This requirement for increased simplicity inhibit the flexibility of asking questions, although it is important that the most critical and frequent information can easily be acquired.

7.1.3 Target User Group

The goal of BGIS is to first support the second, non-technical way of working with data, even though advanced data scientists may also benefit from the tool. This is means that the target user group of the tool are people in Bzzt's management, operations, marketing, design and finance teams. The reason for this is that people in those teams, who often benefit the most from data, have less access to it. This core goal heavily influences the design. The tool needs to be simple enough to allow for non-technical users without rigorous training to use it, while still offer enough flexibility to allow for these teams to investigate the most important questions in their respective field.

7.2 Final Design

This section describes the final design of BGIS, a tool for exploring and visualizing tempo-spatial data, designed for Bzzt AB. The design is outlined at a high level of detail, briefly illustrating the tools concepts and features. A more comprehensive description is provided in the appendix *BGIS Design Document*.

BGIS is a web application which allows users to explore and work with different geographical data-sets. The tool only supports working with one type of data set simultaneously. In the Bzzt domain, two examples of such data sets are *bookings* and *trips*. BGIS also allows for the data to be easily exported and shared with others.

7.2.1 Look & Feel

While the look & feel of tool has not been a priority, it has received some consideration. The main goal of the look & feel is to make the tool feel as functional as possible. The focus of working in BGIS is the actual data visualization, and it is often desired that this part of the interface attracts the most attention of the users. To achieve this, markings are designed to stand out by the use of vibrant, high saturation colors, while menus are toned down with a less contrasting color scheme. To fit the theme of *data science*, fonts, menus and other interface elements has been designed in a modern, slightly futuristic style.

7.2.2 Interface Overview

BGIS graphical user interface consists of a map with visualized data, as well as several menus. The menus allows the data to be filtered in different ways, such as only viewing data during a specific time period. Except for visualizing the data as marks on a map, information about the visualized data can also be read as text.

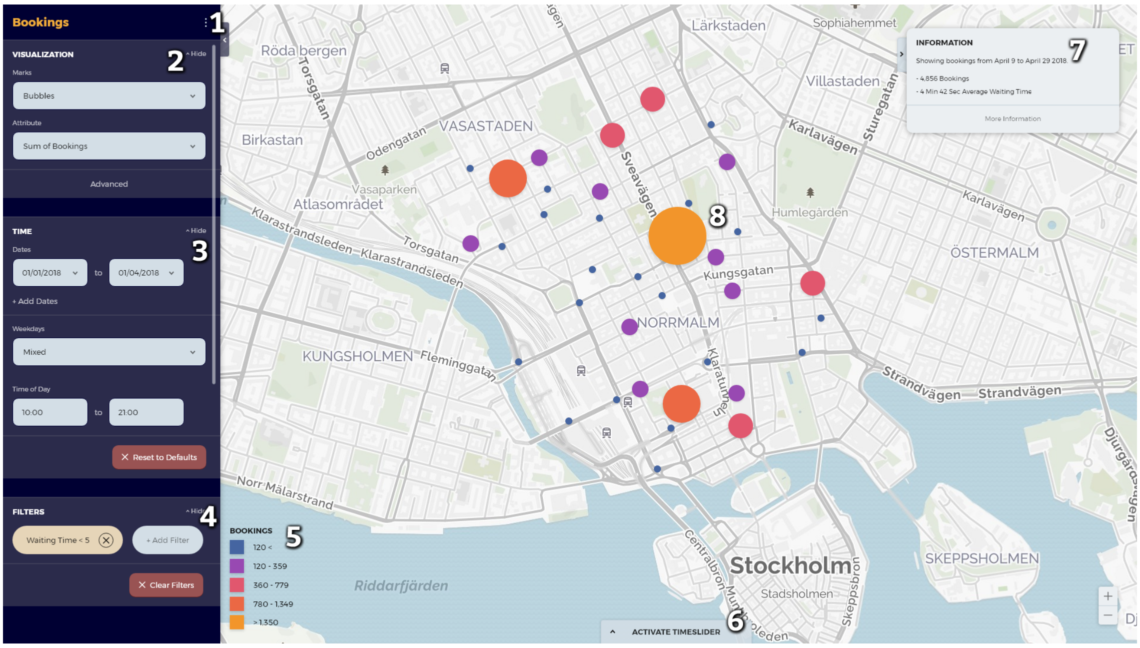

Figure 7.1: An annotated version of the main view of the tool which is currently showing bookings data.

1. **Left View**: The left most view of the screen is used to define which data to show, and how to show it.
2. **Visualization Menu**: In the visualization view it is possible to define how data should be shown. This can be done by setting the mark-type, for example bubbles, heat-map or dot map. The attribute setting controls which variables in the data that should be shown in the visualization, such as different colored markings depending on waiting time.
3. **Time Menu**: Controls time specific settings of what data to show. This includes date intervals, weekdays and time of day.
4. **Filter Menu**: Handles filtering of data depending on the data's attributes. Examples of this are to only show bookings with a specific waiting time or trips with a certain distance.
5. **Legends**: Explains what the colors of the markings mean.
6. **Time Slider**: The *Time Slider* is a feature that allows the data to be iterated through in time steps.
7. **Information Window**: Shows information of what data is currently being showed, as well as information regarding the data's attributes. For bookings this may be average waiting time or total cost.
8. **Marks**: The actual data visualized in different way. In the picture above bookings are visualized in bubbles; a higher number of adjacent bookings results in a bigger bubble.

7.2.3 Data Sets

BGIS is always used with one data set: the type of data which should currently be visualized. An example from Bzzt's pod taxi domain of data points is *bookings*, events where a user booked a taxi with the Bzzt app. Another example of a data set is *opened apps*, events where the user opened the Bzzt smart phone app. A limitation of the BGIS system is that only one data set can be used at one time; it is not possible to explore both *opened apps* and *bookings* simultaneously in one application instance. The only constraints of supported data is that it should contain positional and time information, such as a lat-long position and a time stamp. Every data type have several attributes, which is additional information connected with each data point in the set. This concept is described previously in the *Results* section.

7.2.4 Data Points

The foundation of BGIS is to visualize different geographical data points, and allow the user to explore this data in various ways. During the research phases of the project the relative importance of certain data points was explored through interviews and a data point evaluation exercise with future users and stakeholders. This investigation resulted in a set of data points which were deemed most important to support. These data points are:

- Bookings & Destinations
- Trips
- Opened Apps

The initial design of BGIS are focused on supporting these data points. However, a fundamental concept of the design is that it should be modular. The overall design philosophy should allow for additional data points to be seamlessly integrated in the tool. For this reason, functionality that is overly specific to a certain data point has been avoided. While the research phase has almost exclusively focused on the needs of Bzzt, it is reasonable to assume that these data points are highly important to other logistical service firms.

Attributes

Every data set of a particular data point, which henceforth will be referred to as a *data set*, has attributes which are specific to that type of data. The easiest way to describe this, is by providing an example. Consider the data set of bookings, which are events where a user booked a taxi. This data type contains additional information, which are called attributes:

- Estimated Waiting Time: How long the system informed the user that she had to wait before the pod taxi would pick her up.
- Waiting Time: How long the user actually had to wait before the taxi picked her up.
- Destination: Where the user want to go.
- Estimated Cost: How much the system estimated the trip to cost.

BGIS allows for exploration and filtering of all of these different attributes. For example, it is possible to visualize the markings based on the values of one of the attributes, or to filter the data so only data with certain attribute-values are displayed.

Bookings & Destinations

Bookings are events of a user booking a pod taxi through the Bzzt smart phone app, while *Destinations* are events of when taxi arrives to the final destination of a customer trip. It is an interesting data point from a tempo-spatial viewpoint, as it tells as story about where Bzzt's customers are located, when they are using the service and where they wish to go. This makes it perhaps the most central data point in Bzzt's business.

Trips

Trips are all of the trips a pod taxi drives. Trips can be done in different states, such as *occupied*, *available*, *unavailable* to name a few. Trips are interesting in the context of data science as they tell a story regarding the driving patterns of Bzzt's pod taxis; which routes are common, how long the trips are and how much how the taxis drive in the various states. This can be useful for many reasons. For example, it can help determining where garages and recharge stations should be positioned.
A relatively new business concept of Bzzt is *Bzzt Banners*, which is a technical commercial service. Businesses can buy commercial time on digital roof signs on Bzzt's taxi pods. Which commercial that should be shown is handled digitally in a centralized manner. This allows for high flexibility in showing different commercials at various locations in a city. Trips data allows for Bzzt to get highly accurate information of how much time their taxis spend at individual streets and areas, which can be used to calculate *impressions* and communicate this to their customers.

Opened Apps

Opened apps are events when a user opens up the Bzzt customer app in her smart phone. This data point is relevant as it can tell a story about demand for the Bzzt service. Understanding demand at different locations, times and under various circumstances is essential to optimizing Bzzt's service.

7.2.5 Data Marks

The core of BGIS is to display data as different marks on a map. The types of marks can be adjusted to meet the current needs of the user. If positional accuracy of a single event is needed, the data can be visualized as dots, where each event is represented by a dot on the map. Different data sets support different types of marks. Event based data with a single position can be visualized in different ways compared to route data.
Settings regarding how the data should be visualized is performed in the visualization menu. The two primary options are which mark type to use, and which attribute should be visualized by the marks color. For example, bookings data can be visualized as *dots, hexagons, bubbles, heat map* or *areas (choropleth).* The default attribute option of bookings is to vary the color of the marks depending on the

frequency of bookings events, but this can be changed to map the average waiting time or trip cost.

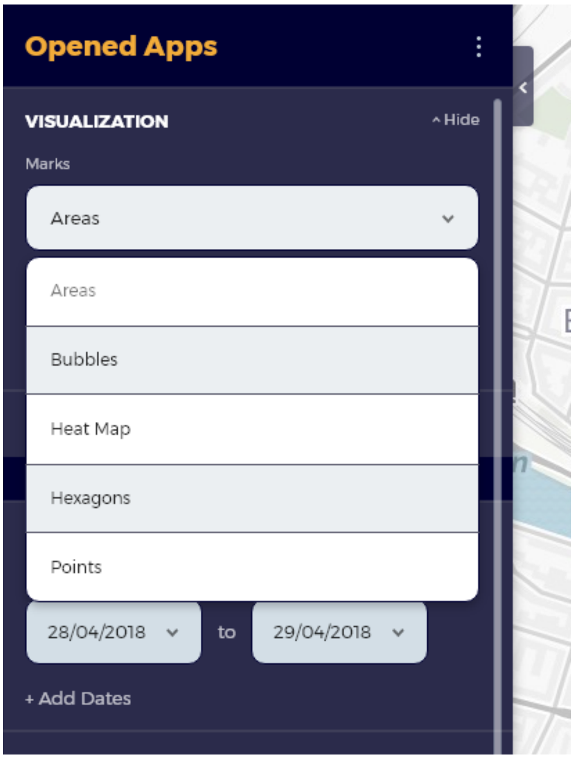

Figure 7.2: Drop-down for choosing which mark type to use for visualizing bookings.

One or several marks can be selected by the user. By doing this, the user can see additional information about the selected events. What information to show depends on the current visualization type. This is useful to get a more accurate understanding of behaviours in a smaller area of the visualization.

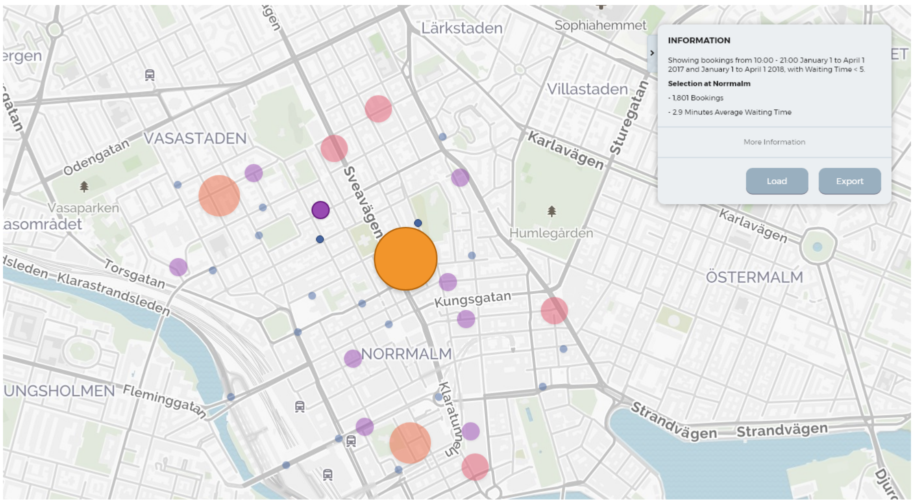

Figure 7.3: It is possible to select one or multiple marks to see more detailed information about them. The screen shot is from an older version of BGIS, when the *Load* and *Export* buttons where located in the information window.

Legends

In the bottom left corner of the map *legends* for the marks are located. These legends describe the numerical values the different colors represent.

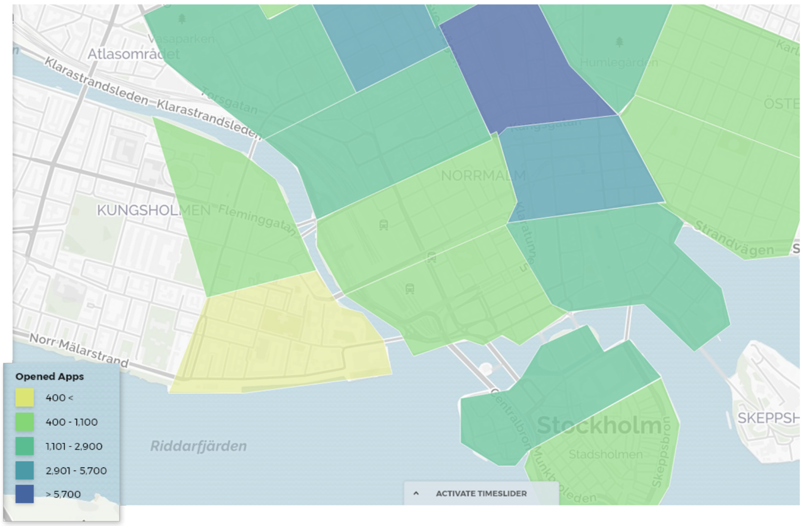

Figure 7.4: Legends describe what the colors represent in numerical values.

7.2.6 Filters

A central part of BGIS is to filter data with different parameters. The most common way of filtering is by defining a time period. This can be done by setting one or more date intervals. Time can also be filtered on days of the week or time of day. For example, a user may apply filters to only show bookings that were made between 8:00 - 11:00 on weekends on Mars 1 - 31.

It is also possible to filter the data on its attributes, such as only displaying bookings with an estimated waiting time of 7 minutes or more. It is possible to combine several time and attribute filters.

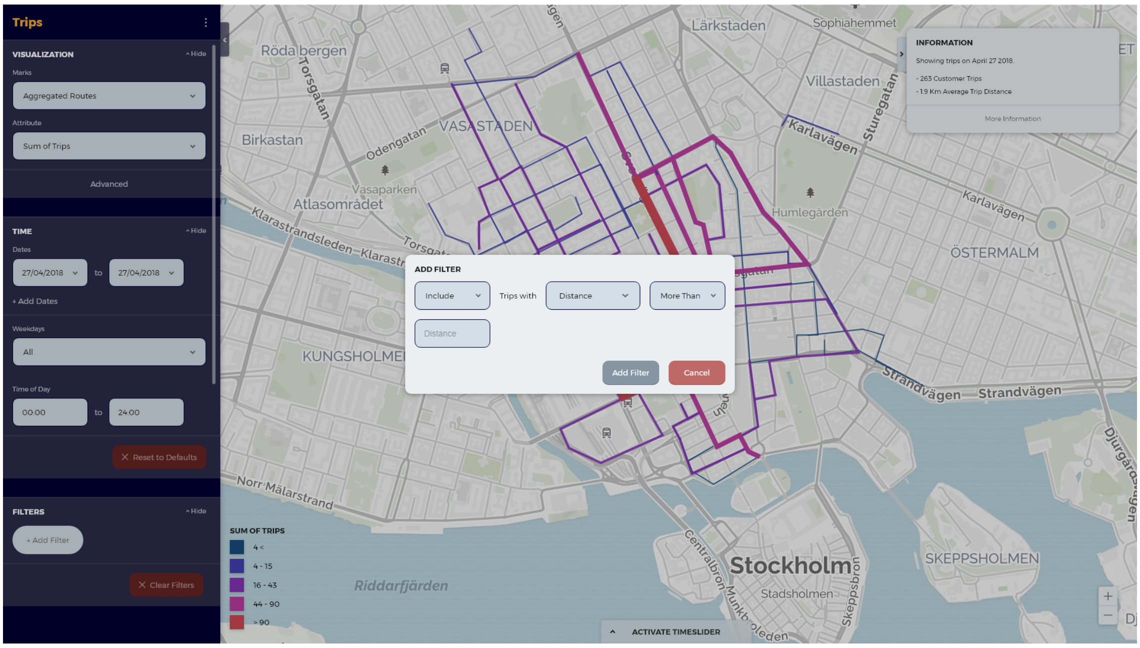

Figure 7.5: The dialog for adding an attribute filter.

7.2.7 Time Slider

The *Time Slider* is a feature which allows the user to easily go through a longer period of time in smaller time steps. The main purpose of this feature is to explore how data changes over time in a quick and easy way. To change the length of each time step (hours, days, months etc.), the user selects the desired time step in a drop-down menu. Figure 7.18 shows an example of the time slider feature.

7.2.8 Information Window

The information window is located in the top right corner of the screen and displays information about the data which is currently visualized. The purposes of the information window are to explain, in text, precisely what data is currently displayed on the map and give central information about the data's attributes. The information can be expanded by clicking on the *More Information* label, which shows a potentially very long list of information about the currently visualized or selected data.

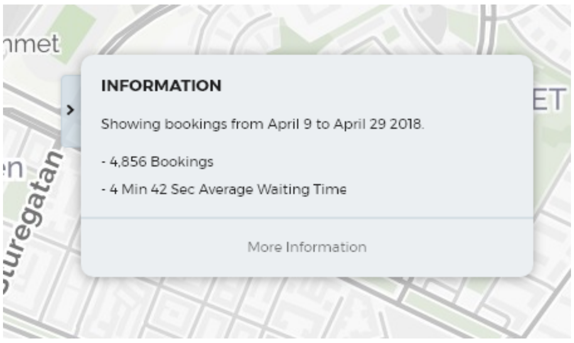

Figure 7.6: The Information Window displays what data is currently being showed and general information about the data.

7.2.9 Sharing & Exporting

Sharing or exporting visualizations and data is an important feature in BGIS. This is performed using a *Share* dialog. It is possible to export the raw data of the current visualization to an excel document. An image of the current visualization can also be exported as a PDF document. Excel and PDF are the two chosen examples of exporting methods in the high fidelity design, but this can easily be changed to other formats. For example, raw data could be exported as CSV, and the visualization as a PNG image file.
A secondary feature in the Share dialog is to share a link of the current visualization. This link contains contains a hash of the current visualization and settings as a URL parameter, which will open the BGIS tool with the exact settings as when it was exported. This is useful for sharing visualizations and settings with other team members, which allows for faster and easier collaboration.

Figure 7.7: The *Share* dialog allows for the current visualization and data to be exported and shared in various ways.

7.2.10 Scenarios

Based on the advice of Olof Torgersson and Johan Wingård, the two supervisors of the project, three scenarios were created to show examples of the functionality of BGIS. The three scenarios show how a fictive user works with the tool. Each of the scenarios is based on a realistic use case in Bzzt's operations. The purpose of the scenarios is not to describe every detail of how the tool can be used, but give an example of each major feature and to get people to reflect on the possibilities that BGIS offers. The text description of these use case scenarios can be read in Appendix B *Use Case Scenarios*.

Scenario 1: General Exploration of Bookings

In the first scenario Martin, a fictive user in Bzzt's operations team, explores bookings data from the previous week. He first investigates the amount and concentrations of bookings, and then examines the waiting times. Martin notices that there was an unusually high waiting time in the south east of Stockholm, and further analyze this by using the *Time Slider* feature. By narrowing down the increased waiting times to a few smaller areas between specific time periods, he is able to conclude that the reason for this anomaly is a running event in Stockholm called *Run Stockholm*.

Scenario 2: Investigate Demand Theory

Martin wishes to explore how the event *Run Stockholm* impacted demand for the Bzzt service. To investigate this, Martin visualizes *Opened Apps* in Stockholm during the dates and times of the event. He notices two clusters with increased demand, and inspect these areas further. To make sure that the increased demand is unusual, he compares it with the same times during other weeks. By doing this, his theory is proven correct, and he finally uses the share feature to share his findings with his colleague Emma.

Scenario 3: Investigate Dishonest Driving Behaviour

Martin has received a tip that one of the drivers in the Bzzt taxi fleet makes private trips with the Bzzt vehicle during working hours. To investigate this, Martin visualizes trips and filters the data to only show trips made by the vehicle of the suspect driver. Martin also filters to only show trips made in the *Unavailable* state. He notices that several longer trips have been made while being unavailable, and analyzes this more by looking at the raw data.

7.2.11 Technical Implementation Considerations

Technical implementation factors have been considered during the design of BGIS, although it has not been a priority in the project. One of the implementation philosophies of Bzzt is that applied in data to the system, which is likely to be in

the form of JSON, will define what visualization and filtering options are available. For example, a JSON blob with bookings data will specify what kind of data this is, which will determine what information to show and what visualization and filtering options should be available in the tool. The reason for this is to make the tool as modular as possible. It should be relatively easy to create a new data set which the tool should seamlessly be able to handle, as long as the JSON is correctly formatted and contains the required fields.
There are several challenges with implementing BGIS. One of these challenges is to display a high number of data points simultaneously without the requirement pre-aggregating data before it is sent to BGIS. One technology that can improve performance in this regard is WebGL[60], a Javascript API for rendering graphics and performing calculations on the graphics card in a web browser.
Another implementation challenge is how often data should be sent to BGIS. The easiest solution is to load the web application with a huge amount of the data at start-up, and then let the front-end handle all of the data logic. The main downside of this is that it requires a lot of bandwidth and processing power of the client computer. The user is perhaps only interested in data from a specific week, but the tool still has to fetch and handle much larger amounts of data, potentially several years. A better, but likely more technically challenging approach, is to make servers handle a lot of the data logic. The BGIS client only requests the data which the user wants to visualize. When the user wants to work with new data, from a different time period for example, the tool needs to make a new request to the server. This may increase loading times, as several requests are necessary, but will heavily reduce the bandwidth and processing requirements of the client computer.

7.3 Geographical Data Visualization Design Guidelines for Logistical Services

Based on the insights gained from carrying out this project, a set of design guidelines and important considerations when designing geographical data visualization solutions for logistical services have been created. While the foundation for this information are derived from research focused on creating a design for the Bzzt domain, much of the findings are more universally applicable. The purpose of these guidelines is to provide rules of thumb for designing geographical data visualization tools for domains similar to Bzzt, as well as describing some relevant aspects and design options.

List of Design Guidelines and Important Aspects

- Understand Who the Target Users Are
- Geographical Visualization Types
- Mark Colors
- Clarity in Currently Presented Data
- Possible to Visualize Data in Various Ways
- Focus User Attention on the Data

- The Default View should provide an Overview of the Domain
- High Degree of Flexibility in terms of Filtering the Data
- Time Filtering
- Possible to Receive more Information about one or several Events
- Easy to Change Filters and Settings
- Possible to Visualize Change in the Data over Time
- Exporting & Sharing
- View the Raw Data
- Visualize and Compare Several Different Data Points

7.3.1 Understand Who the Target Users Are

One of the most important aspects of any interaction design project is to clearly determine who the target users of the design are. Still, it is worth to emphasis that this is usually the first task when designing a geographical data visualization tool. It is especially critical to determine how technically adept the target users of the tool are. If the target audience are data scientists who have a lot of experience with data bases, scripting languages (Python, R) and data science tools (Jupyter Notebook, Shiny Dashboards), the tool likely should be designed to offer a great deal of flexibility and advanced features. On the other hand, if the target users of the tool are identified as non-technical users with professional backgrounds in operations, marketing or finance for example, the requirements of the design will probably be significantly different. The operations needs to be more intuitive and it should be easier to perform the most common use cases, even though some flexibility necessarily needs to be sacrificed.

7.3.2 Geographical Visualization Types

Geographical data points can symbolize an endless variation of things, but it is still possible to categorize them in three distinct ways: *points, lines* or *areas.*

Points

Points data are events that have one precise position. For example, a booking is an event of a customer booking a taxi from one location; it is not possible to book a taxi from several positions simultaneously. These types of events can be visualized as a single dot on a map, or aggregated in different ways. Four ways of aggregating points, each with different strengths and weaknesses, are:

- **Bubbles**: Nearby events get aggregated as *bubbles*, with the bubbles size depending on the number of aggregated events. This visualization type is useful for identifying clusters of events. An example of bubbles visualization in BGIS is shown in Figure 7.8
- **Hexagonal Binning**: Provides both a good overall pattern identification of the data, as well the possibility to determine the value of a single point with fairly high accuracy. This visualization technique is shown in Figure 7.9.

- **Heat Map**: Very useful visualization technique for distinguishing the pattern of the visualized data. This visualization technique is shown in Figure 7.10.

- **Choropleth Map**: Advantageous for providing an overview of data, with the possibility of mapping sub-areas that are relevant in the domain. This visualization technique is shown in Figure 7.11.

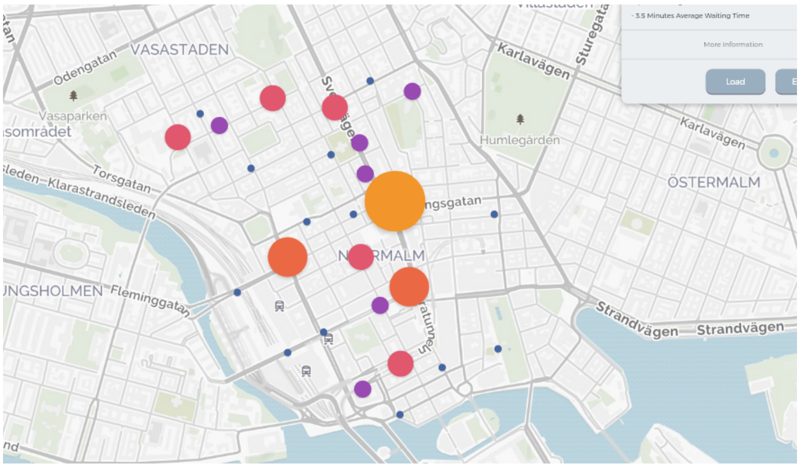

Figure 7.8: Bubbles visualization in BGIS.

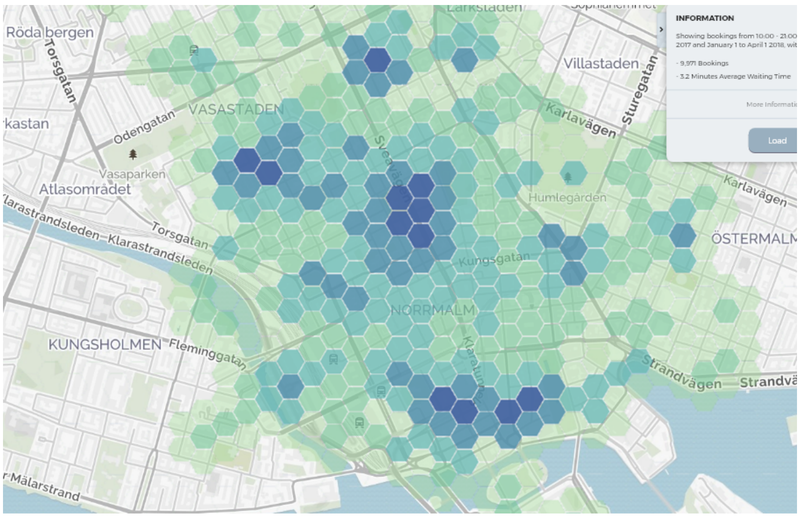

Figure 7.9: Hexagonal binning visualization in BGIS.

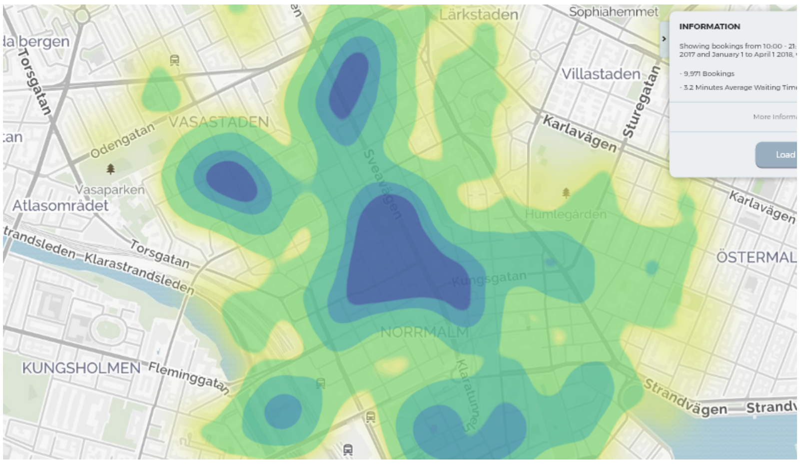

Figure 7.10: Heat map visualization in BGIS.

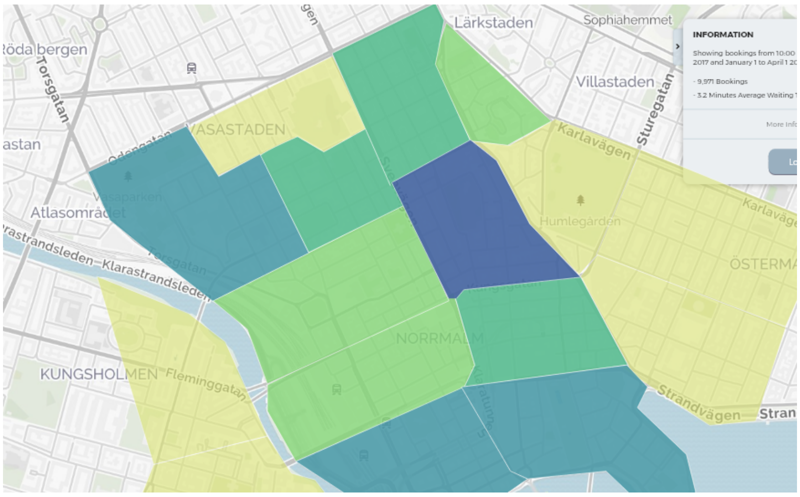

Figure 7.11: Choropleth (areas) visualization in BGIS.

Lines

Line, or route, data consists of several positions. When these positional points are connected they together form a poly-line. A poly-line may be visualized as a simple line on a map, but they can also be aggregated in various ways. A common way of aggregating route data is to make high concentration areas be thicker compared to low concentration areas. In Figure 7.12 an example of this from BGIS is shown.

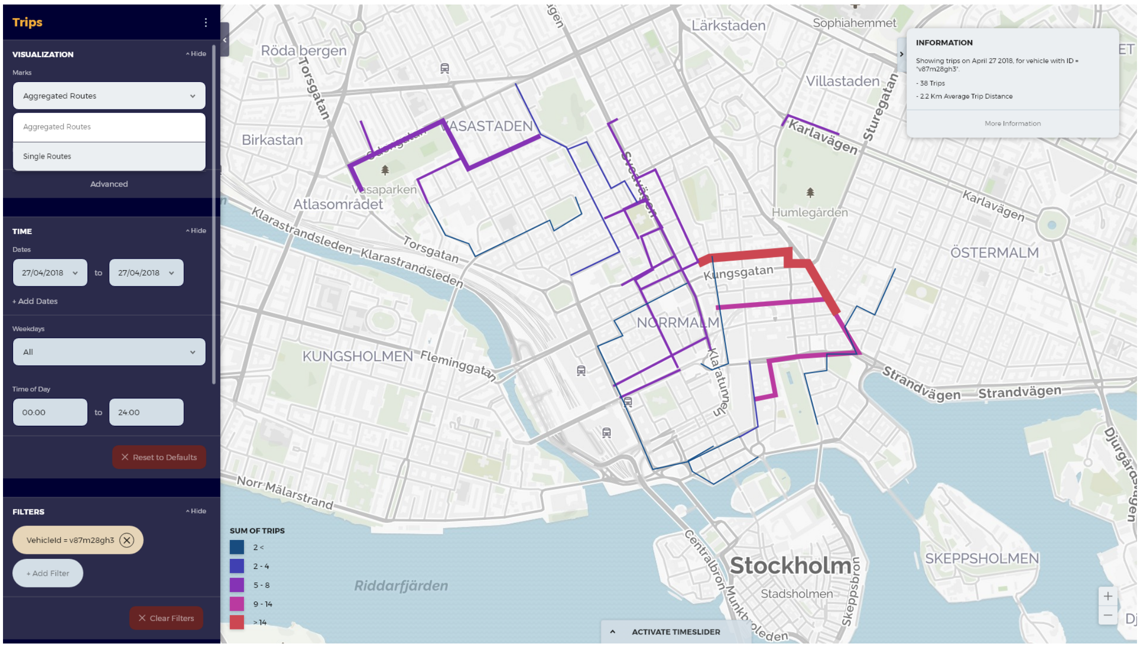

Figure 7.12: A visualization of trips as aggregated routes.

Routes data sometimes also has another general attribute: direction. Visualizing a few directional poly lines clearly is relatively easy, but it gets very difficult with a high number of overlapping poly-lines. Three visualization techniques that can make poly-line direction more clear are:

- **Color Gradient**: Make the color of the line a gradient in the direction of the line.
- **Animation**: Animate the background of the line so that background gradient *flows* in the direction of the line.
- **Start and End Symbols**: Add start and end symbols to each poly line. Note that this can clutter the visualization, especially if there are a high number of simultaneously visualized lines.

In Figure 7.13 an example of a directional poly-line from BGIS is shown.

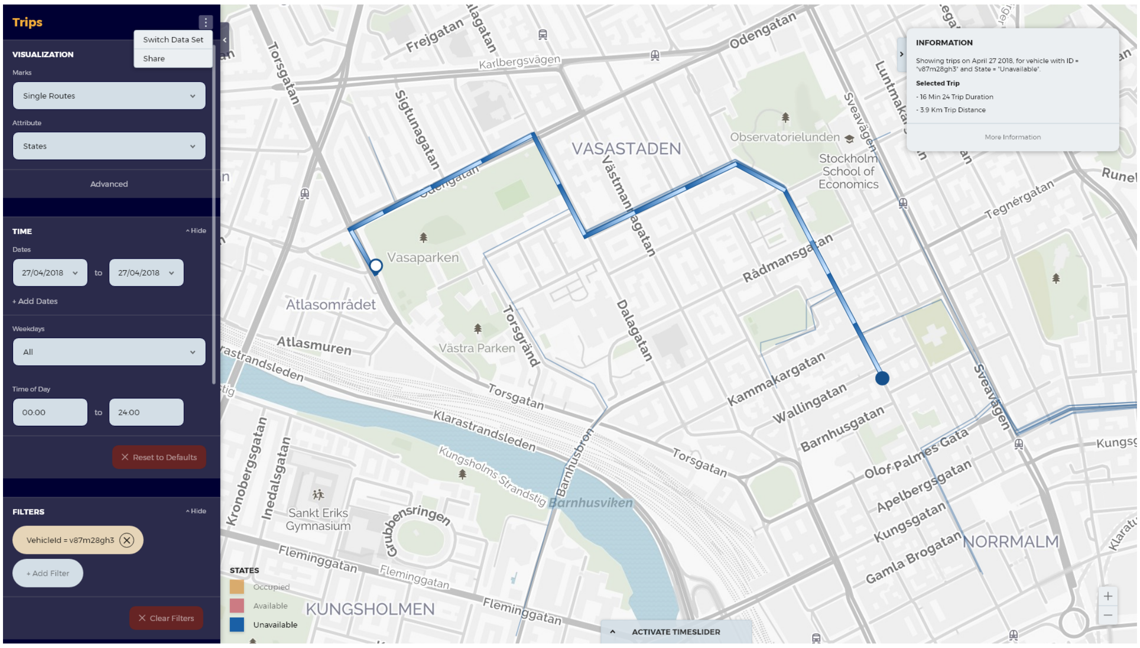

Figure 7.13: A directional poly line, visualizing a trip driven from start to destination. Gradient colors, as well as start and end symbols, are used to clarify the direction of the trip.

Areas

Finally there is area-type data. This data type represents information, perhaps an aggregate of several events, in an area. This type of data is often visualized as a polygon over the represented area on a map. Areas are most commonly visualized as choropleth maps, but could also be aggregated with hexagonal binning or heat map visualization techniques.

7.3.3 Mark Colors

Colors are critical in data visualization. The data marks should be the elements that capture the user's attention the most. In map visualization, a general rule is that data elements should have a sharp contrast compared to the map background. Additionally, the data elements should usually have significantly more vibrant and saturated colors. It is also important that the color palettes and gradients should be designed to make differences in the data as clear as possible, both in identifying the value of a single point in the visualization and to see the overall pattern of the data. To achieve this, multi-variate color gradients should be used: the colors used should differ in several color channels, such as hue and lightness. It is also important that the colors accommodate for color blindness. To further increase the likelihood that the mark colors work well for a wide range of users, different options for color palettes can be supported. The figures below show several palettes used in BGIS, which all follow the aforementioned guidelines for marks color in geographical data visualization.

Figure 7.14: One of the color palettes in BGIS, which uses multi-variate color coding in hue and lightness.

Figure 7.15: Another color palette in BGIS.

Figure 7.16: The color palette in BGIS which is used for aggregated poly line visualizations.

Figure 7.17: The color palette in BGIS which is used to visualize different states.

7.3.4 Clear what Data is Presented Currently

It should be evident to users what data is currently being shown on the map. The easier and faster this is to understand, the better. It is useful to always describe what data is currently being shown as text. This should be done as clearly as possibly.

7.3.5 Possible to Visualize Data in Various Ways

In section 7.3.2, several methods of visualizing different kinds of geographical data were described. It is important that the tool supports a lot of customizability in terms of what visualization method to use. The reasons for this are that different visualization techniques excel in different ways and that people have varying preferences regarding how data should be displayed.

7.3.6 Focus User Attention on the Data

In a geographical data visualization system the most important elements are those which show the actual data visualization, such as data marks on a map. It is desirable that the attention of the user should be primarily drawn to these elements. To achieve this, use visually distinct and highly contrasting graphics for data elements. For secondary parts of the interface such as menus, input elements and similar, use discrete graphics. Also, make it possible to minimize menus and windows to allow the user to focus on the data visualization even more.

7.3.7 The Default View should provide an Overview of the Domain

The default view of the system should show the user a high-level overview of the visualized data. For logistical services, this generally means displaying the whole area of the logistical service. For example, Bzzt is a taxi company which is only active in Stockholm, and in BGIS the default visualization when the tool is initiated is a zoomed out view of Stockholm, where events from the previous month are visualized. This concept is derived from the first part of Schneiderman's Mantra[34].

7.3.8 High Degree of Flexibility in terms of Filtering the Data

While the default settings of the systems should provide an overview, it is important the tool supports the data to be narrowed down significantly, and in multiple ways. By the use of filters the user can focus on very specific aspects of the data. An obvious way of filtering are time periods, but each data point may have specific attributes which could be used to filter on as well.

7.3.9 Time Filtering

Early on the research phase, time was identified as a very important factor for geographical data in any logistical service. It should be possible to see data at various time periods. A few examples of this are:

- Explore data from the previous week.
- Explore data from every Thursday in May.
- Explore data during the Christmas holidays over several years.
- Explore how data changes from day to day in the previous month.
- Explore how data changes hour to hour on the 7th of November.

7.3.10 Possible to Receive more Information about one or several Events

A common way of working with data visualization is to begin with observing the data on an overview level, and then go in to the details of interesting aspects [34]. To support this work flow, it is important that it is possible to acquire additional information about smaller parts of the visualized data. Since geographical data generally is displayed on a map, zoom functionality is one method of providing this. Two other methods that help with this are to show tool-tips with more information when a user mouse over a data mark, and to allow the user to select one or several marks to receive more information about these events. Another possibility is to have a secondary view that displays the data in a structured text format.

7.3.11 Easy to Change Filters and Settings

A common scenario is that a user sets some input parameters in order to view a sub set of the data. The user continues to explore different sub sets by tweaking and changing individual parameters. To support this exploratory work flow, it is essential that it easy to adjust settings, filters and other parameters. Changing such a value should require as few number of operations as possible (an operation in this context is, for example, clicking a button). It is also critical that an easy way of resetting all filters and restoring the setting to default values is readily available to the user.

7.3.12 Possible to Visualize Change in the Data over Time

When working with geographical data, or data in general, it is often interesting and relevant to observe how the data changes over time. While this is not essential in all domains, it is a common requirement which a geographical data visualization and exploration system likely should support.

There are many ways of visualizing how geographical data behaves over time. An important factor when designing this is how many data points are going to be visualized simultaneously, as well as the size of the markings. If there are few data points, each point can be visualized with a small mark and there are significant geographical spread between the points, all of the data points can clearly be shown at the same time. To show the time variable of each point, color coding could then be used. Note that this visualization method is not pre-attentively processed [17].

Often however, these prerequisites are not satisfied, or at least it is not certain that they are. The most consistently successful way of visualizing how geographical data changes over time is to remove and add data points to the map based on the position in a time axis. This can either be done automatically through animating how the data changes from the start to the end timestamp, or it can be done manually by letting a user control some input element which decides the time period to visualize. In BGIS, the second method is used. An example of how a user can observe how data changes over time by using the *Time Slider* is shown in Figure 7.18. This decision was based on the disadvantages of animated visualizations, which can be read in Chapter 3.

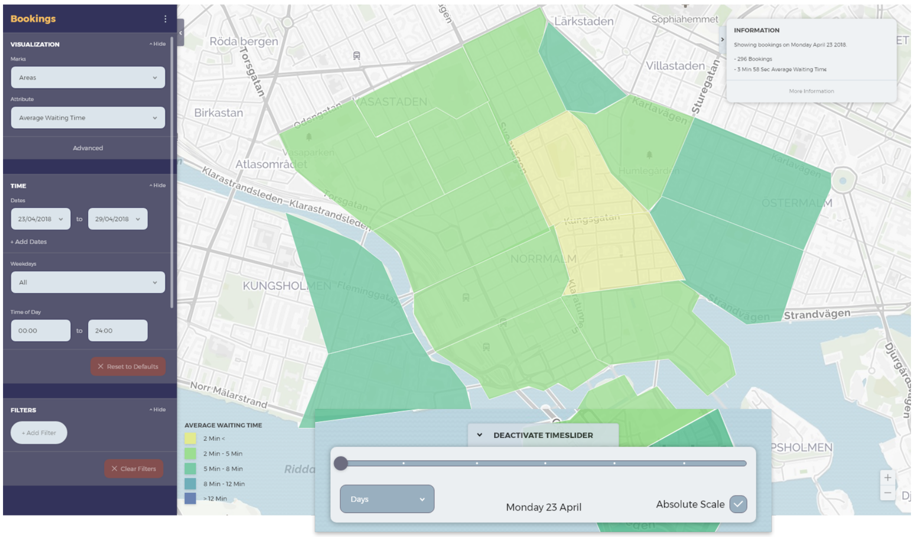

Figure 7.18: The Time Slider in BGIS: a tool for iterating through data in smaller time steps to visualize how the data changes over time.

7.3.13 Exporting & Sharing

Except for investigating and directly working with data, another crucial requirement is to share data in different ways. This can be done internally through sharing data findings to allow for better collaboration and communication within a team. Another use case is sharing data externally to outside stakeholders like partners, investors, customers or users. Depending on the goal and recipient, the data needs to be visualized and shared in different ways.

When working with data science within a team, a common use case is to share the visualization settings so that several people can load exactly the same visualization on several clients. Another frequent requirement is to export data in the form of reports. In most cases it is essential that the design support both of these features.

7.3.14 View the Raw Data

It is often desired to investigate the raw data in text form. Marks on a map are great for providing an overview of events, but the accuracy and level of detail are often limited. A common scenario is that a user finds something interesting in the map visualization, and in order to get a better understanding, wishes to go through each data point in structured text format. This should be supported by the system, although to what extent may vary depending on the specific requirement of the domain.

The simplest way of providing this feature is to make it possible to export the data as structured text, for example CSV. This is an easy solution, but it has several disadvantages: it requires the users to work with several tools simultaneously (one for the geographical visualization, and one for exploring the data as text), and it is cumbersome and inflexible if the user is required to export the data frequently with different parameters.

Another approach is to make the system support viewing the data as text in a separate view. This is even more powerful if the view allows the text data to be easily filtered and searched. A middle ground between the two aforementioned approaches are to make the tool display the most important data in text form, but if the user needs to explore details of the raw data, exporting may be required.

7.3.15 Visualize and Compare Several Different Data Points

It is often desired to explore the relationship between two or more different data points. For example, how does the amount of rain influence the frequency of taxi rides. To allow users to analyze such questions, a good approach is to make it possible to visualize several different data types simultaneously.
While this very useful, it is also notoriously difficult to visualize several geographical data points at the same time while still making the visualization clear to the user. The reason for this is primarily that identifying both the overall shape and single points in bi-variate geographical visualizations has been shown to be difficult[61]. To improve this, the amount of comparable data points could be limited to two. Further more, use varied visualization methods (example: choropleth for one data point, and bubbles for the other) or make the data points different on several visual channels (shape, color etc.).

8
Discussion

The discussion chapter will reflect on the various aspects of the design project. This includes reflection on the final design, the design guidelines, the design process used in the project, ethical and legal considerations as well as future improvement of the design.

8.1 Final Design

Overall, the result of the final design is satisfactory. The objective was to design a tool which can help Bzzt AB to work with and share geographical data. Based on the feedback received on the design, BGIS accomplishes these goal. However, it is difficult to properly evaluate how well BGIS actually meets the different requirements, since no parts of the system have been implemented. The ambition of the project was to create a rough prototype with limited functionality in order to test how difficult BGIS is to implement and how well it meets the needs of the users. Unfortunately, because of time constraints, no such prototype has been implemented. This makes the reflection on BGIS's strengths and weaknesses slightly more speculative.
The main challenge when designing BGIS was to strike a balance between ease of use and flexibility. The target users of the systems are non-technical operators in Bzzt's operations, management, marketing, design and finance teams. It is critical that the presented model of the system matches their mental model of how it should work as closely as possible, and not the underlying implementation model. Changing settings, filtering data and selecting events need to be intuitive, and it should not be necessary to have prior knowledge in scripting or how data bases work. However, it also important to note that the target user group is limited to internal personnel, which makes it more acceptable if the tool requires some training and time to learn.

8.1.1 Advantages

BGIS offers quite a lot of flexibility in terms of what questions a user can ask, while still remaining fairly simple. In particular, there is a high degree of elasticity in filtering data in different time periods. Data can be filtered on several date intervals, on certain days of the week and on a specific time during the day. Additionally by using the *Time Slider*, it is possible to easily see how data changes over time, with interchangeable time steps.

The Information Window ensures that the user always has information regarding what data is currently being visualized or selected, since this information is provided as plain text. While this may take some time to get used to, it should reduce the ambiguity in terms of what data is being shown in the visualization. Following *Shneiderman's Mantra* [34], the default view of the tool is an overview of the data, but the user may explore single data elements with greater accuracy and detail by selecting data elements, using the *More Information* feature and exporting the raw data.

A lot of consideration has been focused on the actual data visualization. The user has several options regarding what mark type to visualize the data with. The various mark types offers different benefits and disadvantages: dot maps provide the greatest positional accuracy, heat maps present a clear overview and pattern of many data points and hexagonal binning gives both a fairly good single point accuracy with decent pattern recognition. Colors palettes are another aspect that have received a lot of attention and time. The goal is that marks, by the user of differences in size and colors, should be pre-attentively processed by the users. The color palettes have been optimized with this in mind, making each color as visually distinct as possible compared to the background as well as other colors in the palette. For example, color gradients differ on both the hue and the lightness channel.

Attributes offer additional flexibility in what aspect of the data to visualize. It is possible to add several attribute filters. With this feature it is possible, for example, to only show trips that were made in the *Occupied* state and that were longer than three kilometers. It is also possible to set the colors of the marks to represent a particular attribute.

Finally, the share visualization feature allows for visualizations to be shared to collaborators in an easy way. By simple sharing a link, the visualization can be accessed on another client in an effortless way.

8.1.2 Disadvantages

The most significant weakness of BGIS is that it can only handle one data type simultaneously. For example, it not possible to explore both *destinations* and *trips* at the same time in one instance of BGIS. This significantly reduces the flexibility of the tool, as there are many use cases when a user wishes to explore the relationship between two different data sets. There are several reasons that this is not supported in the current version of the BGIS design. The primary reason is that visualizing several different geographical data points simultaneously is notoriously difficult in the context of making the visualization clear to the user. However, it is likely that many users would happily accept the reduced clarity for the increased flexibility of many simultaneous data points.

Another reason is more technical. To better fit into Bzzt's current data pipeline and data science strategy, the Bzzt software team suggested that the tool could work similarly to other graph tools which are currently being used. A fundamental implementation aspect of these graphs is that they are initiated with a set of data, and are unable to fetch additional data without being recreated. For the sake of consistency, BGIS was therefore designed with this constraint. If BGIS were to support multiple simultaneous data points, the tool would therefor need to be initiated with a much larger amount of data. This would increase the necessary bandwidth and client processing resources.

The final reason is that it were a risk that the scope of the project would be to large. If the tool would be able to support multiple simultaneous data points, this would likely have a big impact on the design of the overall interface. New problems and corner cases would likely be discovered, and the quality of the delivered design would perhaps suffer from this increase in scope.

There are other, less severe limitations in the tool. BGIS does not support an easy way of viewing the raw data as text. This is possible, but it requires the data to be exported to CSV or Excel files. An improvement might be a view that shows the raw data as entries directly in the application. It would likely be even more useful if these data entries could be filtered and searched. There are several other areas in which BGIS could be polished and improved further, but these are minor aspects which will not be described specifically in this chapter.

8.2 General Design Guidelines for Logistical Services

Except for the design of BGIS, the other major result of this project were a set of guidelines and important aspects for designing geographical data visualization tools for logistical services. These guidelines and considerations were produced based on the findings of this design project, which primarily focused on the Bzzt domain. An important question therefor is how generally applicable these discoveries are. My belief is that many of the findings and requirements identified in this project are relevant in a much larger context. No matter what types of geographical data points a tool supports, the core requirements and functions should remain fairly constant. Examples of some generally relevant functions are:

- Visualize distinct marks on a map.
- Visualize how data behaves over time.
- Filter the data in different time periods.
- View the raw data as structured text.
- Share and export visualizations.
- Select data elements to get more information about the specified data points.

One aspect which varies greatly between different domains is the relative importance of different functions. For one domain, exploring how the data changes over time might be the most critical use case, while in another this is a rarely used feature. This principle has been a consideration when the guidelines were written. The design considerations and guidelines are supposed to be rules of thumb for designing geographical data science solutions, not absolute truths. The priority of features and requirement details will necessarily be different depending on which domain and what users the visualization tool targets.

8.3 The Design Process

In this project John Chris Jones Design process was used, together with the Kanban work flow process and many methods from Cooper's Goal Directed Design. Overall this has worked very well. The three phases of Jones design process has provided a clear road-map for the project, and there were never any issues with regards to what tasks to perform next.

I personally appreciate the methods of Cooper's Goal Directed Design. His principle of primarily focus on the underlying goals of the users makes a lot of sense in my opinion. The reason for this is that a person's goals provides the most fundamental motivation as to why that person performs tasks and actions. The *Requirement Definition Process* has been particularly useful for converting unstructured qualitative data from interviews into concrete and measurable requirements.

Kanban has been the work flow process which I have successfully been using for several years, and it was used with satisfactory results in this project as well. The process is simple enough so that it does not create any time consuming and unnecessary overhead, but still allows for tasks to be properly organized and prioritized. It is also useful for retrospective, since it is easy to reflect on previously done tasks which are automatically ordered by their time of completion.

A huge benefit of this project is that it primarily has been conducted at the Bzzt office. This has allowed me to acquire greater insight in Bzzt's business, quickly become aware of changing requirements and get to know many of the stakeholders and target users. It has also made it significantly easier to get feedback on the progression of the design. The cooperation and frequent feedback from Johan Wingård, my supervisor at Bzzt, has been especially useful. Without his input I believe that the results of this project would have had significantly lower quality.

8.3.1 Improvements of the Design Process

While the design process has worked fairly well, there are several aspects which can be improved. The design process as a whole could be done in a more iterative way. This project has been composed of three big phases: *Divergence, Transformation* and*Convergence*. A likely better a approach is to have several shorter cycles of these three steps. Working in this way would allow feedback from early designs to drive future research phases, and would probably resulted in better end results. Figure 8.1 shows an example of how such a process could look like, with three design iterations each consisting of 17 days.

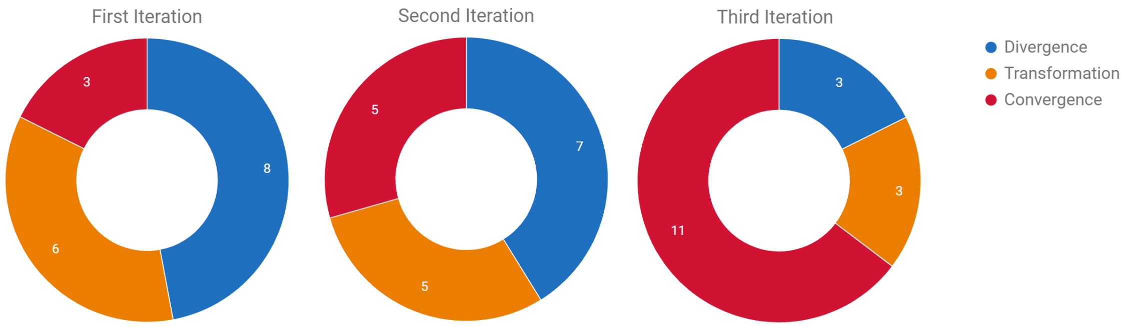

Figure 8.1: An alternative, more iterative design process approach. There are three design iterations, all containing each of Jones' three phases, with varying number of days.

In the first phase of the design process most work were devoted interviewing potential users and stakeholders. All relevant employees at Bzzt's headquarter in Gothenburg were interviewed, and this activity yielded a lot of valuable insights and qualitative data. However, the actual taxi operations are carried out by the daughter company *Bzzt Stockholm*. While the management and personnel at Bzzt Stockholm are not the target users of the design, these people almost certainly have unique insights in critical aspects and problems of the Bzzt business. No one of the employees at Bzzt Stockholm were interviewed. To get even better information to guide the design, interviews with key people in the Stockholm office should have been conducted.
Throughout the later stages of the design process, when wire-frames of BGIS were iteratively designed, frequent feedback sessions were conducted with target users. However, no summative user tests were performed since the prototype did not have enough interactions to properly support this. Summative user tests would provide value to the design. By performing summative user tests it would be possible to better evaluate how well BGIS design matches the mental model of the target users. Additionally, errors in details of the design might be discovered, such as unclear copy.

8.4 Applying the Results in a General Context

BGIS was designed based on research of the needs of employees at Bzzt AB, but many of the uses of the system should be applicable to any similar business, such as other taxi companies or delivery services. A fundamental design philosophy is that the tool should be highly modular; it should be easy to add new data sets. Since the only requirement of the data supported by Bzzt is that it should have a position and a time stamp, the range of potential data the tool can handle is almost endless. All of the features in BGIS can be applied to a very broad array of use cases, and while Bzzt is primarily a taxi company there are no functionality in BGIS that is limited to taxi operations. BGIS could, without the need of any fundamental changes to the design, handle data sets such as *flights* or *delivery vehicles*.
One limitation of the current design of BGIS in terms of applying the system to completely new data types are the supported ways of visualizing the data. When visu-

alizing a public transportation system for example, a common visualization method is *origin-destination maps*. Figure 8.2 shows an example of such a visualization in Tableau. This type of visualization is currently not supported in BGIS.

Figure 8.2: An example of an *origin-destination map* in Tableau. This type of visualization is commonly used for visualizing public transportation systems.

8.5 Ethical Considerations

Since BGIS is a tool that revolves around data, the main ethical consideration of the tool is that it complies with EU's General Data Protection Regulations (GDPR)[62]. Since BGIS does not directly store any data, the sections of GDPR that focuses on *data portability* and *right to erasure* does not impact the design. *Data portability* refers to a user's right to request data connected to them, which should be delivered in a structured format such as CSV[63]. *Right to erasure* is that a user has the right to request that personal data that is concerning him or her, which is collected by the target company, should be deleted[64].

A rule of GDPR that does influence the design of BGIS is that personal data, defined as information which *can be used to directly or indirectly identify an individual*, can only be collected with *specified, explicit, and legitimate purposes*. This is called *The Purpose Limitation Principle* [65]. Therefor, Bzzt needs to explicitly state the specific use cases where personal data should be used. With the implementation of GDPR it is no longer acceptable to collect personal data for experimental and vaguely defined research purposes. It is also necessary for Bzzt to acquire consent from the users, which needs to be given in a unambiguous and affirmative way [66]. Bzzt needs to clearly communicate the purpose of their data science using personal data to all of the impacted people, and receive their permission to proceed.

8.6 Future Improvements

With additional time, the next natural step in the design process of BGIS would be to create a prototype of the tool in code. It is likely that this provides a lot of value to the design, both in terms of discovering new problems, technical difficulties and to allow the design to be tested in meaningful ways. My personal preference is to build a prototype using the JavaScript framework React[67], map and location service Mapbox[68] and data analysis and visualization framework DeckGL[69]. I would choose these tools because they fit well with BGIS' requirements and have good historical track records.
Another important activity that was not conducted are summative user tests. BGIS is primarily a data exploration tool. For this reason, it is likely that it would be worthwhile to implement a prototype where a user has a wide range of options and possible operations before conducting summative tests. Once the foundation of such a prototype existed, summative user tests would preferable be performed frequently and iteratively, accompanying implementation of additional features in the prototype. Following the advice of Cooper et. al.[42]: *user tests should be performed late enough in the design process when there is a concrete design to test, and early enough to allow for adjustments.*

8.6.1 Extensions of the Design

As stated previously in this paper, BGIS has been designed as a modular tool which allows for easy addition of new data sets. A natural way of extending the tool therefor, is to add more types of data. The data types *bookings, destinations, trips* and *opened apps* were chosen as the first supported data sets based on their importance to Bzzt, but there are data points which are interesting for Bzzt as well, such as:

- **Suggested routes** between pickup and destination points, provided by third-party service such as Google. These could be compared to the actual driven routes, and difference might reveal interesting insights.
- **General traffic data** might be interesting for optimizing vehicle locations at different times.
- **Amount of people in the street** can help in both predicting demand and impressions of the Bzzt commercial service.

The feature that would probably increase the uses of BGIS the most is support for multiple simultaneous data sets, or at least the possibility of visualizing two different data types at the same time. Exploring how support for this could be added to BGIS is perhaps the most critical extension of BGIS' design. Support for even two different data points simultaneously is a major change to the system, and would influence several aspects of the design such as menu interfaces, color palettes, selections, filters and what information to show in the *Information Window.*
Apart from simultaneous multi data set support, an improved way of viewing the raw data as text in a structured way is possibly the second largest enhancement to the tool. In the current state of BGIS' design it is possible to view the raw data if you export the visualization to some text format, but it would be convenient if

viewing the raw data was a feature directly supported in the web application.

9
Conclusion

In this master thesis project two research questions were explored. In this chapter, the conclusions to these research questions are presented.
How can the pod taxi company Bzzt's geographical data be presented and explored to allow for business improvements and operational optimization?
To answer this the geographical data science tool BGIS has been designed for Bzzt AB. Based on the feedback received from users and stakeholders at Bzzt, the resulting design succeeds in reaching the project's goal of presenting geographical data in way that allows for improvements and optimization in Bzzt's operations. An example screen from the BGIS design is shown in Figure 9.1. The main deliverables to Bzzt from this project are a design document, wire-frames and use case scenarios which outlines examples of BGIS functionality. The ambition of the project was that a prototype of the design would be implemented in code in order to evaluate the system more thoroughly. Unfortunately, due to time constraints, no such programmatic prototype has been created.

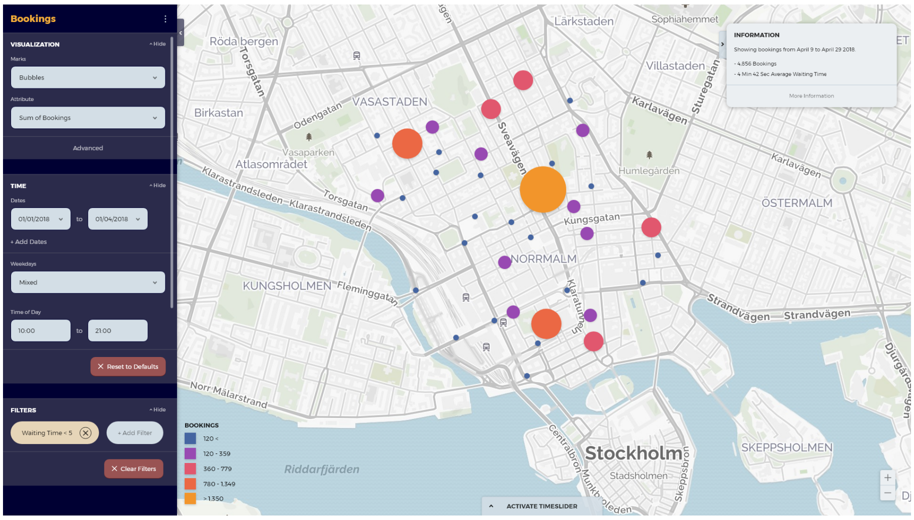

Figure 9.1: Bookings visualized as bubbles in a screen from the BGIS design.

An important conclusion from the research phase was that the people most in need of a geographical data science tool at Bzzt were non-technical personnel in the Operations, Management, Marketing and Design teams. This group of people were therefor chosen as the target users of the design. The data points that were discovered to be most important to Bzzt were *Customer Bookings, Destinations, Trips* and *Opened Apps*. An important goal is that the system should be highly modu-

lar and have the possibility of being extended with additional data points in the future.
The most important requirements discovered during the Divergence phase were that the tool should be easy to use, allow for filtering on time and other attributes, support several visualization types, show data points as both visual elements and plain text, support exporting and sharing of data, and contain functionality which allows the user to understand how data changes over time.
The future steps in the continued design of BGIS are to create a prototype in code and to test this prototype with target users. In terms of extension of the tool, the two most critical features, based on research and user interviews, are support for multiple simultaneous data points and functionality for viewing, searching and filtering the data as text. It is preferable to perform summative user tests before beginning to extending the tool, since these tests might reveal new problems and even more critical requirements.
What are important requirements, aspects and options when designing geographical data science systems for logistical services?
To answer the second research question a set of guidelines and characteristics for designing geographical data science tools for logistical services were produced. The purpose of these guidelines is to provide a foundation for designers who are designing or researching similar systems. Some of the proposed rules are very specific *(use highly contrasting and saturated colors for data marks on a map)*, while others are more general *(it is critical to determine the level of technical competence of the target user, as this have significant impact on the design).* The rules and aspects of the guidelines are purposely not prioritized, as this is likely highly individual to the domain that the geographical visualization tool is designed for.
The produced design guidelines are not absolute truths, but rather rules-of-thumb and important considerations. Some aspects to focus on are: what are the most common work flows, which visualization types are used in different scenarios, how often does users desire to compare two or more data points and how commonly does users want to explore the raw data in structured text form.

Bibliography

[1] David Kirsh and P. Maglio. On distinguishing epistemic from pragmatic action. *Cognitive Science*, 18(4):513–49, 1994.

[2] Michael R. Peres. *The Focal Encyclopedia of Photography (Fourth Edition)*. Focal Press, Boston, fourth edition edition, 2007.

[3] Pierre Jolicoeur Ben Bauer and William B Cowan. Distractor heterogeneity versus linear separability in colour visual search. *Perception*, 25:1281–1293, 1996.

[4] Bzzt official website. `https://www.bzzt.se/`. Accessed: 2018-02-64.

[5] Joshi N. and Geracioti I. Turbocharging analytics at uber with our data science workbench. 10 2017.

[6] National renewable energy laboratory: Geospatial data science. `https://www.nrel.gov/gis/tools.html`. Accessed: 2018-02-11.

[7] Tableau: Data science and business intelligence solutions. `https://www.tableau.com/`. Accessed: 2018-02-14.

[8] Colin Ware. *Information Visualization: perception for design*. Elsevier, 225 Wyman Street, Waltham, MA 02451, USA, 3 edition, 2012.

[9] Christopher G. Healey. Choosing effective colours for data visualization. San Francisco, CA, USA, 1996. Visualization '96. Proceedings, IEEE.

[10] Colin Ware. Color sequences for univariate maps: Theory, experiments, and principles. *IEEE Computer Graphics and Applications*, pages 41–49, September 1988.

[11] Gordon Kindlmann and Carl fredrik Westin. Diffusion tensor visualization with glyph packing. *IEEE Transactions on Visualization and Computer Graphics*, November 2006.

[12] Natasha Kutlesa Ian Spence and David L. Rose. Using color to code quantity in spatial displays. *Journal of Experimental Psychology: Applied*, 5, 1999.

[13] Haim Levkowitz and Gabor T. Herman. The design and evaluation of color scales for image data. *IEEE COMPUTER GRAPHICS AND APPLICATIONS*, 12:72–80, 1992.

[14] Gordon Kindlmann, Erik Reinhard, and Sarah Creem. Face-based luminance matching for perceptual colormap generation. In *Proceedings of the Conference on Visualization '02*, VIS '02, pages 299–306, Washington, DC, USA, 2002. IEEE Computer Society.

[15] Howard Wainer and Carl M. Francolini. An empirical inquiry concerning human understanding of two-variable color maps. 34:81–93, 05 1980.

[16] Joshua A. Solomon and Denis Pelli. The visual filter mediating letter identification. 369:395–7, 07 1994.

[17] Anne Treisman. Preattentive processing in vision. *Computer Vision, Graphics, and Image Processing*, 31(2):156 – 177, 1985.

[18] Philip T. Quinlan and Glyn W. Humphreys. Visual search for targets defined by combinations of color, shape, and size: An examination of the task constraints on feature and conjunction searches. *Perception & Psychophysics*, 41(5):455–472, Sep 1987.

[19] Charles W. Eriksen and Harold W. Hake. Absolute judgments as a function of stimulus range and number of stimulus and response categories. *Journal of Experimental Psychology*, 49(5):323, 1955.

[20] Anne Treisman and Stephen Gormican. Feature analysis in early vision: Evidence from search asymmetries. 95:15–48, 02 1988.

[21] Ken Nakayama and Gerald H. Silverman. Serial and parallel processing of visual feature conjunctions. 320:264–5, 03 1986.

[22] Jan Theeuwes and Frank L. Kooi. Parallel search for a conjunction of shape and contrast polarity. 34:3013–6, 12 1994.

[23] Michael D'Zmura, Peter Lennie, and Carlo Tiana. Color search and visual field segregation. *Perception & Psychophysics*, 59(3):381–388, Apr 1997.

[24] Jon Driver, Peter McLeod, and Zoltan Dienes. Motion coherence and conjunction search: Implications for guided search theory. *Perception & Psychophysics*, 51(1):79–85, Jan 1992.

[25] K. Koffka. *Principles of Gestalt Psychology*. International library of psychology, philosophy, and scientific method. Routledge & K. Paul, 1955.

[26] Stephen Palmer and Irvin Rock. Rethinking perceptual organization: The role of uniform connectedness. *Psychonomic Bulletin & Review*, 1(1):29–55, Mar 1994.

[27] Stephen Palmer. Common region: A new principle of perceptual grouping. 24:436–447, 08 1992.

[28] Colin Ware, Peter Mitchell, and John G. W. Kelley. Designing flow visualizations for oceanography and meteorology using interactive design space hill climbing. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, San Antonio, TX, USA, 11-14 October 2009*, pages 355–361, 2009.

[29] Ehtibar N. Dzhafarov, Robert Sekuler, and Jüri Allik. Detection of changes in speed and direction of motion: Reaction time analysis. *Perception & Psychophysics*, 54(6):733–750, Nov 1993.

[30] Gunnar Johansson. Visual perception of biological motion and a model for its analysis. *Perception & Psychophysics*, 14(2):201–211, Jun 1973.

[31] Barbara Tversky, Julie Morrison, and Mireille Bétrancourt. Animation: Can it facilitate? 57:247–262, 10 2002.

[32] Richard Mayer, Mary Hegarty, Sarah Mayer, and Julie Campbell. When static media promote active learning: Annotated illustrations versus narrated animations in multimedia instruction. 11:256–65, 01 2006.

[33] Susan Palmiter, Jay Elkerton, and Patricia Baggett. Animated demonstrations vs written instructions for learning procedural tasks: A preliminary investigation. 34:687–701, 05 1991.

[34] Ben Shneiderman. *Designing the User Interface: Strategies for Effective Human-Computer Interaction.* Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 3rd edition, 1997.

[35] Stuart K. Card, Allen Newell, and Thomas P. Moran. *The Psychology of Human-Computer Interaction.* L. Erlbaum Associates Inc., Hillsdale, NJ, USA, 1983.

[36] Chikio Hayashi. What is data science ? fundamental concepts and a heuristic example. In Chikio Hayashi, Keiji Yajima, Hans-Hermann Bock, Noboru Ohsumi, Yutaka Tanaka, and Yasumasa Baba, editors, *Data Science, Classification, and Related Methods*, pages 40–51, Tokyo, 1998. Springer Japan.

[37] Mike DeBoer. Understanding the heat map. *Cartographic Perspectives*, 0(80), 2015.

[38] Christopher Wesson. Pros and cons of various geo-visualization techniques. 10 2014. Accessed: 2018-02-14.

[39] Vasavi Ayalasomayajula. 7 techniques to visualize geospatial data. 10 2016. Accessed: 2018-02-14.

[40] Nick Rabinowitz. *Visualisation of spatio-temporal data set?* discussion thread. `https://gis.stackexchange.com/questions/408/visualisation-of-spatio-temporal-data-set`. Accessed: 2018-03-16.

[41] Nick Rabinowitz. Nick rabinowitz web site. `http://nickrabinowitz.com/`. Accessed: 2018-03-16.

[42] Alan Cooper, Robert Reimann, David Cronin, and Christopher Noessel. *About Face: The Essentials of Interaction Design.* Wiley Publishing, 4th edition, 2014.

[43] Frank Rudolf. Model-based user interface design: Successive transformations of a task/object model. *CRC Press*, 1998.

[44] Donald A. Norman. *The Design of Everyday Things.* Basic Books, Inc., New York, NY, USA, 2002.

[45] J. Christopher Jones. *Design methods : seeds of human futures / J. Christopher Jones.* Wiley-Interscience London ; New York, 1970.

[46] Y. SUGIMORI, K. KUSUNOKI, F. CHO, and S. UCHIKAWA. Toyota production system and kanban system materialization of just-in-time and respect-for-human system. *International Journal of Production Research*, 15(6):553–564, 1977.

[47] What is kanban? `https://leankit.com/learn/kanban/what-is-kanban`. Accessed: 2018-02-10.

[48] A Allport and Glenn Wylie. Task switching, stimulus-response bindings, and negative priming. 18:33–70, 01 2000.

[49] Stephen Monsell. Task switching. *Trends in Cognitive Sciences*, 7(3):134–140, 2003.

[50] Ken Schwaber. Scrum development process. In Jeff Sutherland, Cory Casanave, Joaquin Miller, Philip Patel, and Glenn Hollowell, editors, *Business Object Design and Implementation*, pages 117–134, London, 1997. Springer London.

[51] Ideo. *IDEO Method Cards: 51 Ways to Inspire Design.* William Stout, 2003.

[52] David Travis. Usability test reporting. `https://www.userfocus.co.uk/articles/cif.html`. Accessed: 2018-06-10.

[53] Duncan Haughey. Moscow method. https://www.projectsmart.co.uk/moscow-method.php. Accessed: 2018-06-09.

[54] Yoland. Wadsworth. *Do it yourself social research / Yoland Wadsworth*. Allen and Unwin Crows Nest, N.S.W, 3rd ed. edition, 2011.

[55] Jodi Forlizzi, Carl Disalvo, Jeffrey Bardzell, Ilpo Koskinen, and Stephan Wensveen. Quality control: a panel on the critique and criticism of design research. pages 823–826, 01 2011.

[56] William Gaver. What should we expect from research through design? In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 937–946, New York, NY, USA, 2012. ACM.

[57] Balsamiq official web site. https://balsamiq.com/. Accessed: 2018-04-09.

[58] Adobe xd official web site. https://www.adobe.com/products/xd.html. Accessed: 2018-05-08.

[59] Project jupyter official web site. http://jupyter.org/. Accessed: 2018-05-08.

[60] Mozilla's official webgl web site. https://developer.mozilla.org/en-US/docs/Web/API/WebGL_API. Accessed: 2018-05-10.

[61] Cynthia A. Brewer. Guidelines for selecting colors for diverging schemes on maps. *The Cartographic Journal*, 33(2):79–86, 1996.

[62] Gdpr portal: information page of eu gdpr. https://www.eugdpr.org/. Accessed: 2018-02-14.

[63] Art. 20 gdpr right to data portability. https://gdpr-info.eu/art-20-gdpr/, 2018. Accessed: 2018-05-11.

[64] Art. 17 gdpr right to erasure ('right to be forgotten'). https://gdpr-info.eu/art-17-gdpr/, 2018. Accessed: 2018-05-11.

[65] Art. 5 gdpr principles relating to processing of personal data. https://gdpr-info.eu/art-5-gdpr/, 2018. Accessed: 2018-05-11.

[66] Art. 7 gdpr conditions for consent. https://gdpr-info.eu/art-5-gdpr/, 2018. Accessed: 2018-05-11.

[67] React official web site. https://reactjs.org/. Accessed: 2018-05-11.

[68] Mapbox official web site. https://www.mapbox.com/. Accessed: 2018-05-11.

[69] Deckgl official web site. http://uber.github.io/deck.gl/. Accessed: 2018-05-11.

Appendices

A
Requirements

The requirements of the tool. The requirements are prioritized using the Moscow method, which divides the requirement in four different categories depending on their importance. The categories are *Must have, Should have, Could have* and *Would have.*

A.1 Must Have

- View data points on a map.
- Map zoom functionality.
- Filter data on time.
- See data in different aggregation and geographical granularity.
- Show individual events.
- Show additional information about individual and aggregated data points and events.
- Heatmap
- Point map
- Hexagon map
- Polylines
- Polyline heat map
- Visualize polyline direction (static or animated).
- Display clearly in text what data is currently showing.
- Legends that explain data visualization.
- Map shows streets, names of places etc.
- No setup required.
- Easy to use.

A.2 Should Have

- See data changes over time. (Slider)
- See data changes over time with dynamic time steps.
- Choropleth map
- Circle Map
- Polyline heat map
- Filter data points geographically.
- Two step time filtering (ex: first dates, then only certain weekdays).

- Display data in text.
- Data variables displayed in several ways, for example color, height and size.
- Dynamic variable numbers depending on shown data.
- Go in and back from a details view of a data point.
- Initially shows the most relevant parts of the data.
- Easy way to clear all current filter options.
- Nice & clean graphic design.
- Fast and responsive.
- Modular: it should be easy to extend the tool.
- Minimize/maximize windows and options.
- Visualisera extremvärden tydligare.
- Possibility to change what kinds of user control is allowed for each graph.

A.3 Could Have

- See data changes over time. (Animation)
- Aggregate data automatically depending on zoom level.
- Take formatted screenshots to visualize data well in a picture.
- 3D map visualization.
- Export raw data. (excel for example)
- Different variable options (example: colorize areas based on amount or average waiting time).
- Undo button.
- Autoplay slider.
- Display relative data information (example: 55
- Animations that helps in showing functionality.
- Change graphical focus of data underlying map.
- Filter on specific data points, for example only show vehicles of a certain state, one specific taxi vehicle or customer id.
- GDPR connection.
- Fun (graphical language, animations sounds & earcons etc.)
- Switch between relative and absolute variables.
- Switch between logarithmic, linear, exponential variable scales etc.
- Control over what variables show.
- Dynamically change visualization type depending on amount of events.
- Users can choose variable colors.
- Switch between showing numbers and percentage.
- Save/load filter settings.
- Med slidern, visa data från tidigare steg med lägre opacitet.

A.4 Would Have

- Make persistent comments in the data visualization.
- Two different uncorrelated data showed simultaneously overlapped.
- Comparing two different filter options.

- Double variables (example: saturation for amount and hue for waiting time).
- Post to slack feature.
- Night / day cycle
- Specialized graphs (example: Graph to optimize vehicle location in accordance to demand).
- English & Swedish
- Export to animated gif

B
Use Case Scenarios

Actor

Martin, an operations manager at Bzzt.

Date

Monday 30th of April.

B.1 Scenario 1: General Exploration of Bookings

Martin wishes to explore the bookings during the last week and notices an anomaly.

Showed Requirements

- Time Slider
- Select aggregated positional marker.
- Read Information about showed data.
- Change attribute.

Part 1: Explore bookings last week

1. Martin opens the bookings visualization, with choropleth visualization.
2. Martin sets the dates to last week.
3. Martin notes that there where x amounts of bookings.

Part 2: Explore Waiting Time last week

1. Martin is interested in the waiting time, so he switches attribute to waiting time.
2. He notices that there where particularly high waiting time in the south east of Stockholm.
3. By selecting both of the areas he sees that the average waiting time were 9 min, which is quite high.
4. Martin deselects the areas.

Part 3: Explore day by day with Time Slider

1. Martin wants to understand if it was a high waiting time every day, so he activates the time slider.
2. He slides through the week, and notices that the waiting time is only high in the south east of Stockholm during Saturday and Sunday.

Part 4: External Research

1. By making a Google search Martin finds out that large areas of south east of Stockholm has been closed down because of Run Stockholm, a running event (`http://www.runstockholm.se/om-banan2/`).

B.2 Scenario 2: Investigate Demand Theory

Martin wishes to explore how running events influences demand.

Showed Requirements

- Time Slider
- Change time granularity in time slider.
- Some different position visualizations.
- Bubbles
- Choropleth
- Select Aggregated Positional Marker
- Read Information about showed data.
- Filter on certain days of week.
- Filter on time of day.
- Share visualization link.

Part 1: Check Demand during Run Stockholm

1. The Run Stockholm event has peeked Martin's interest. He wants to investigate how the race influenced demand.
2. Martin switches to the Opened Apps visualization.
3. He filters the time to show last week's weekend, with time of day set to 8:00 - 17:00.
4. Martin notice that it was an increased demand in south east Stockholm.

Part 2: Switch visualization method.

1. In order to get a more accurate understanding regarding where the demand were higher, Martin switches to Bubble visualization.
2. Martin notices two high demand clusters.

Part 3: Select cluster.

1. Martin selects the biggest cluster to see additional information.
2. Martin notices that there were X amount of opened apps in the area.
3. Martin clicks on More Information.
4. Martin discovers that 40% of opened apps was made by new users. He thinks it is unfortunate that they were not able to supply new users with better service.
5. Martin deselects the bubble.

Part 4: Compare to other weekends.

1. Martin wants to check whether the high demand is an anomaly compared to other weekends. He increases the time period to the last two months.
2. He opens the time slider and sets the granularity to week.
3. He scrolls through the weekends and confirms that the high demand last week is indeed anomaly.

Part 5: Share findings with Emma.

1. Martin thinks this is interesting and wishes to share his findings with his colleague Emma.
2. He shares the link to the visualization to Emma.

B.3 Scenario 3: Investigate Dishonest Driving Behavior

Martin wants to investigate if a driver has made private trips during working hours.

Showed Requirements

- Aggregated Polylines
- Single Route Polylines
- Select Single Polyline
- Read Information about showed data.
- Filter on vehicle id.
- Legend Filtering
- Share visualization to colleagues.
- Change attribute visualization.

Part 1: Investigate routes data visualization.

1. Martin opens up the route visualization. He has received a tip that one of the drivers are take unofficial trips during working hours. This driver's id is 34289e31.

2. He filters the driver id to include 34289e31.
3. He filters the time to a day he received the tip on.

Part 2: Single Route Visualization

1. To see every individual route the driver made, Martin switches to single route visualization.

Part 3: Visualize states

1. To better see the different states of the trips, Martin switches the visualization attribute to states.
2. Martin notices that many of the trips are made in the unavailable state.

Part 4: Legend Filtering

1. To only see trips made from the unavailable state, Martin hides the other states through legend filtering.
2. He notices how the driver has made several long trips in the unavailable state.

Part 5: Legend Filtering

1. To only get a better look on individual trips, Martin zooms in the visualization.
2. He selects one of the unavailable trips, and notices how it is 3.8 km long.
3. Something seems off with this many longer trips.

Part 6: Raw Data Export

1. Martin shows data from all states.
2. Martin exports the raw data to excel to be do further analysis of the raw numbers.

C

Design Document

This is a design document for the BGIS, an abbreviation for Bzzt Geographic Information System. BGIS is a tool for exploring and visualizing tempo-spatial data, designed for Bzzt AB.
BGIS is a web application which allows users to explore work with different geographical data-sets. The tool only supports working with one type of data set simultaneously. In the Bzzt domain, two examples of such data sets are *bookings* and *trips*. BGIS also allows for the data to be easily exported and shared with others.

C.1 Interface Overview

BGIS graphical user interface consists of a map with visualized data, as well as several menus. The menus allows the data to be filtered in different ways, such as only viewing data during a specific time period. Except for visualizing the data as marks on a map, information about the visualized data can also be read as text.

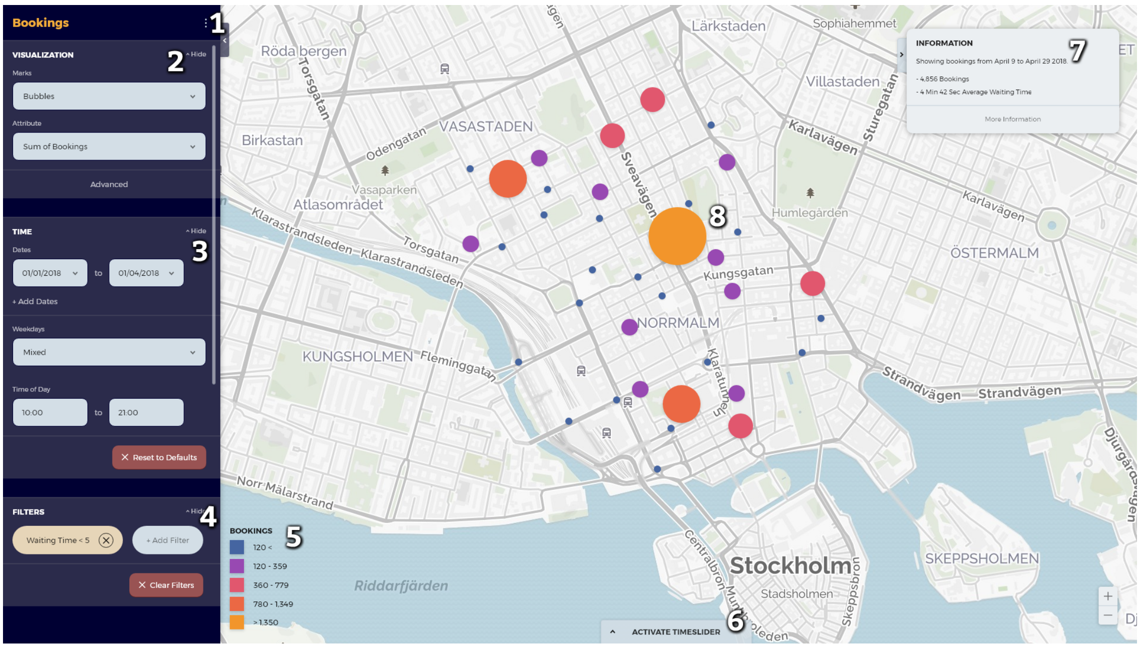

Figure C.1: An annotated version of the main view of the tool which is currently showing bookings data.

1. **Settings View**: The left most view of the screen is used to define which data to show, and how to show it.
2. **Visualization Menu**: In the visualization view it is possible to define how data should be shown. This can be done by setting the mark-type, for example bubbles, heat-map or dot map. The attribute setting controls which variables in the data that should be shown in the visualization, such as different colored markings depending on waiting time.
3. **Time Menu**: Controls time specific settings of what data to show. This includes date intervals, weekdays and time of day.
4. **Filter Menu**: Handles filtering of data depending on the data's attributes. Examples of this are to only show bookings with a specific waiting time or trips with a certain distance.
5. **Legends**: Explains what the colors of the markings mean.
6. **Time Slider**: The *Time Slider* is a feature that allows the data to be iterated through in time steps.
7. **Information Window**: Shows information of what data is currently being showed, as well as information regarding the data's attributes. For bookings this may be average waiting time or total cost.
8. **Marks**: The actual data visualized in different way. In the picture above bookings are visualized in bubbles; a higher number of adjacent bookings results in a bigger bubble.

C.2 Concepts

In this section the central concepts of BGIS are described at a general level.

C.2.1 Data Sets

BGIS is always used together with one data set: the type of data which should currently be visualized. An example from Bzzt's pod taxi domain of data points is *bookings*, events where a user booked a taxi with the Bzzt app. Another example of a data set is *opened apps*, events where the user opened the Bzzt smart phone app. A limitation of the BGIS system is that only one data set can be used at one time; it is not possible to explore both *opened apps* and *bookings* simultaneously in one application instance. The only constraints of supported data is that it should contain positional and time information, such as a lat-long position and a time stamp.

C.2.2 Attributes

Every data set has attributes which are specific to that data type. The easiest way to describe this, is by providing an example. Consider the data set of *bookings*, events where a user books a taxi. This data type has several additional information, or *attributes*, such as:

- **Estimated Waiting Time**: How long the system informed the user that she had to wait before the pod taxi would pick her up.
- **Waiting Time**: How long the user actually had to wait before the taxi picked him up.
- **Destination**: Where the user want to go.
- **Estimated Cost**: How much the system estimated the trip to cost.

BGIS allows for exploration and filtering of all of these different attributes. For example, it is possible to visualize the markings based on the values of one of the attributes, or to filter the data so only data with certain attribute-values are displayed.

C.2.3 Filters

A central part of BGIS is to filter data with different parameters. The most common way of filtering is by defining a time period. This can be done by setting one or more data intervals. Time can also be filtered on days of the week or time of day. For example, a user may filter to only show bookings that was made between 8:00 - 11:00 on weekends on 1 - 31 Mars.

It is also possible to filter the data on its attributes, such as only displaying bookings with an estimated waiting time of 7 minutes or more. It is possible to combine several time and attribute filters.

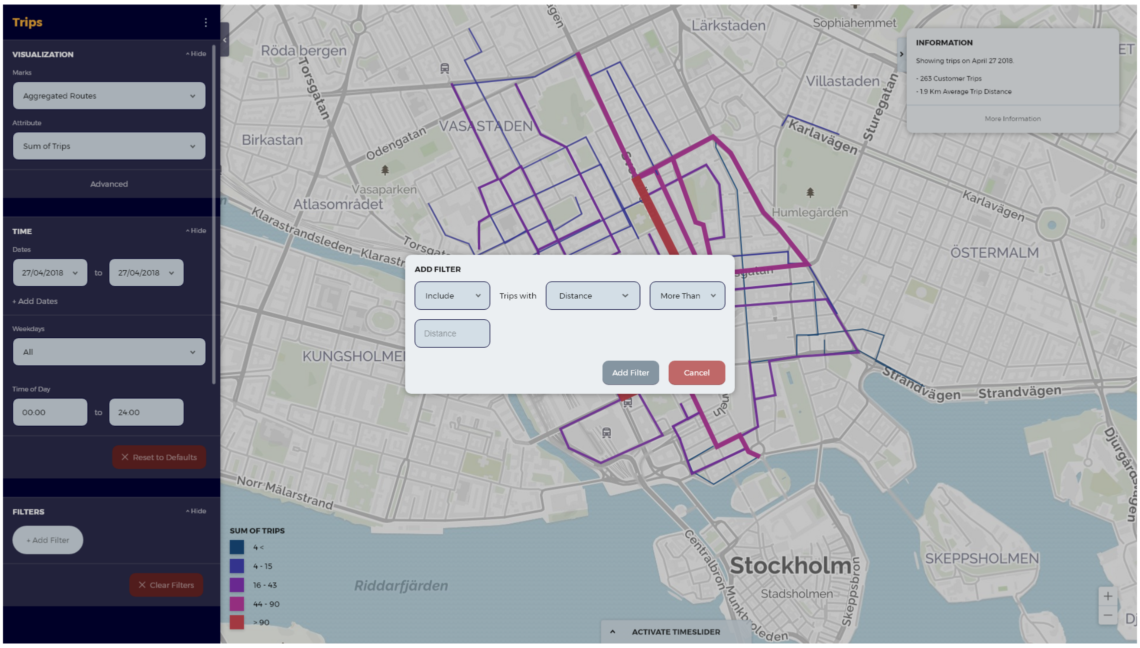

Figure C.2: The dialog for adding an attribute filter.

C.2.4 Time Slider

The *Time Slider* is a feature which allows the user to easily go through a longer period of time in smaller time steps. The main purpose of this feature is to explore how data changes over time in a quick and easy way. To change the length of each time step (hours, days, months etc.), the user selects the desired time step in a drop-down menu. Figure C.3 shows an example of the time slider feature.

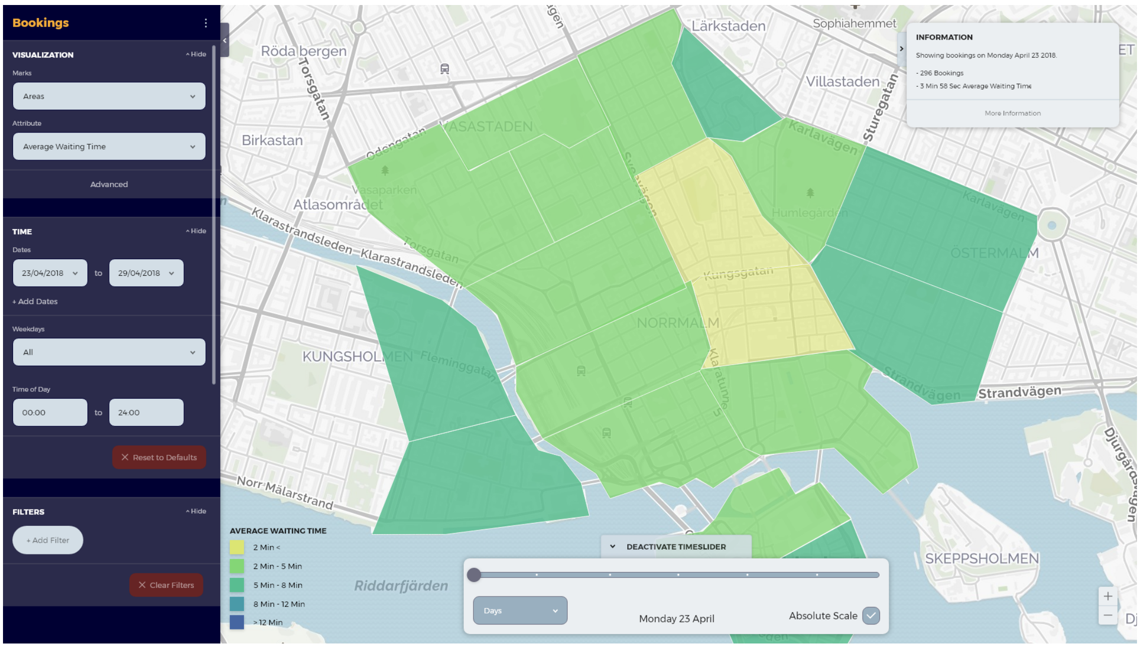

Figure C.3: The Time Slider: a tool for iterating through data in smaller time steps to visualize how the data changes over time.

C.2.5 Marks & Selections

The core of BGIS is to display data as different marks on a map. The types of marks can be adjusted to meet the current needs of the user. If positional accuracy of a single event is needed, the data can be visualized as dots, where each event is represented by a dot on the map. Different data sets supports different types of marks. Event based data with a single position can be visualized in different ways compared to route data.

Settings regarding how the data should be visualized is performed in the visualization menu. The two primary options are which mark type to use, and which attribute should be visualized by the marks color. For example, bookings data can be visualized as *dots, hexagons, bubbles, heat map* or *areas (choropleth)*. The default attribute option of bookings is to vary the color of the marks depending on the frequency of bookings events, but this can be changed to map the average waiting time or trip cost.

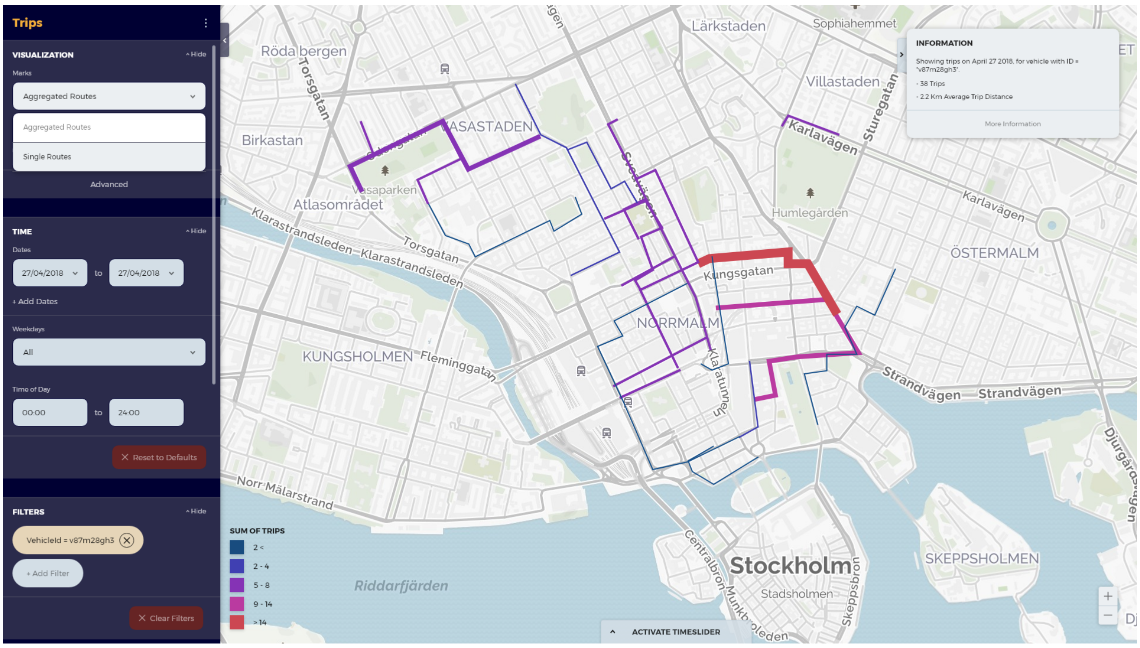

Figure C.4: A visualization of trips as aggregated routes.

One or several marks can be selected by the user. By doing this, the user can see additional information about the selected events. What information to show depends on the current visualization type. This is useful to get a more accurate understanding of behaviours in a smaller area of the visualization.

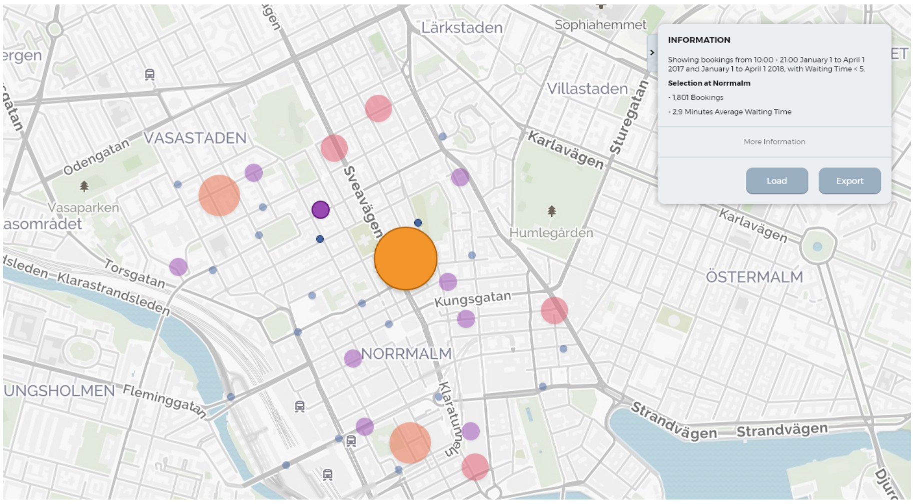

Figure C.5: It is possible to select one or multiple marks to see more detailed information about them. The screen shot is from an older version of BGIS, when the *Load* and *Export* buttons where located in the information window.

C.2.6 Information Window

The information window is located in the top right corner of the screen and displays information about the data which is currently visualized. The purposes of the information window are to explain, in text, precisely what data is currently displayed on the map and give central information about the data's attributes. The information can be expanded by clicking on the *More Information* label, which shows a potentially very long list of information about the currently visualized or selected data.

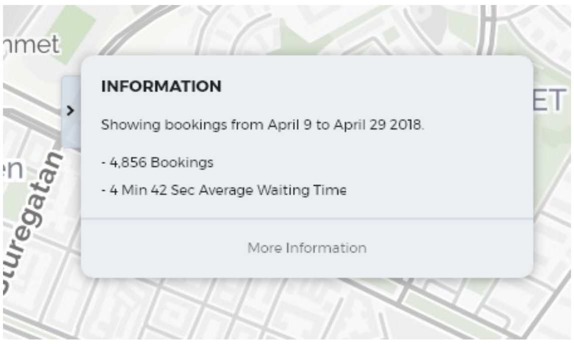

Figure C.6: The Information Window displays what data is currently being showed and general information about the data.

C.2.7 Sharing & Exporting

Sharing and exporting visualizations and data is an important feature in BGIS. This is performed using a *Share* dialog. It is possible to export the raw data of the current visualization to an excel document. An image of the current visualization can also be export as a PDF document. Excel and PDF are the two chosen examples of exporting methods in the high fidelity design, but this can easily be changed to other formats. For example, raw data could be exported as CSV, and the visualization as a PNG image file.

A secondary feature in the Share dialog is to share a link of the current visualization. This links contains contains a hash of the current visualization and settings as a URL parameter, which will open the BGIS tool with the exact settings as when it was exported. This is useful for sharing visualizations and settings with other team members, which allows for faster and easier collaboration.

Figure C.7: The *Share* dialog allows for the current visualization and data to be exported and shared in various ways.

C.3 Data Sets

As previously explained, BGIS can be loaded with different data sets, depending on what data the user wants to explore. A foundation of BGIS is that it should be modular; it should be relatively easy to expand the tool with the new functionality and support for new types of data. When developing BGIS for Bzzt, four important data points were defined for the initial design of the tool: *bookings, trips & opened apps.*

C.3.1 Bookings

Bookings are events where a user books a pod taxi through the Bzzt smart phone app. It is an interesting data point from a tempo-spatial viewpoint, as it tells as story about where Bzzt's customers are located, and when they are using the service.

Attributes

- `Cost [number]`
- `Destination [position]`
- `Estimated Cost [number]`
- `Estimated Waiting Time [number]`
- `New User [boolean]`
- `Time Stamp [timestamp]`
- `Waiting Time [number]`

Supported Mark Types

- `Choropleth Map`
- `Heat Map`
- `Bubbles`
- `Hexagons`
- `Dot Map`

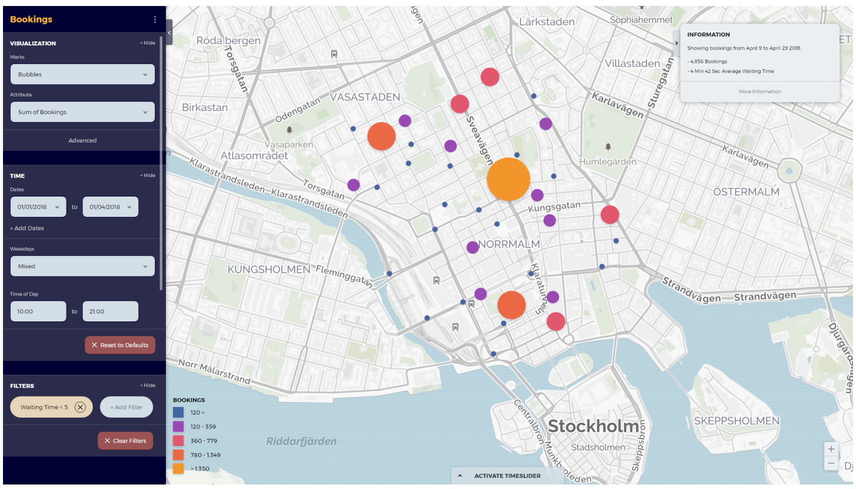

Figure C.8: BGIS displaying bookings data.

C.3.2 Trips

Trips are all of the trips a pod taxi drives. Trips can be done in different states, such as *occupied*, *available*, *unavailable* etc. Trips are interesting from a data science context as they tell as story of how Bzzt's pod taxi drives; which routes are common, how long the trips are and how much how the taxis drive in the various states.

Attributes

- `Cost [number]`
- `Distance [number]`
- `Duration [number]`
- `Start Position [position]`
- `End Position [position]`
- `Shown Commercial [enum]`
- `State [enum]`
- `Start Time Stamp [timestamp]`
- `End Time Stamp [timestamp]`

Supported Mark Types

- `Poly Lines`
- `Aggregated Poly Lines`
- `Choropleth Map`

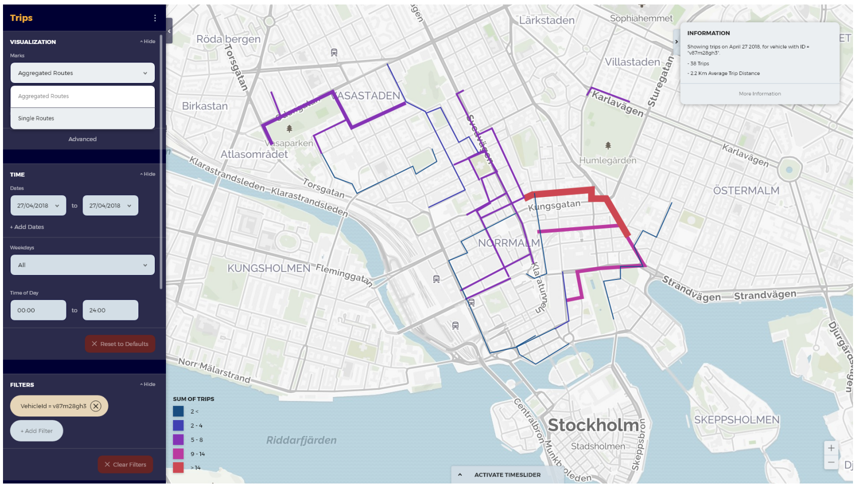

Figure C.9: BGIS displaying trips data.

C.3.3 Opened Apps

Opened apps are events when a user opens up the Bzzt customer app in her smart phone. This data point is relevant as it can tell a story about demand for the Bzzt service at different locations and at various times.

Attributes

- `New User [boolean]`
- `Time Stamp [timestamp]`

Supported Mark Types

- `Choropleth Map`
- `Heat Map`
- `Bubbles`
- `Hexagons`

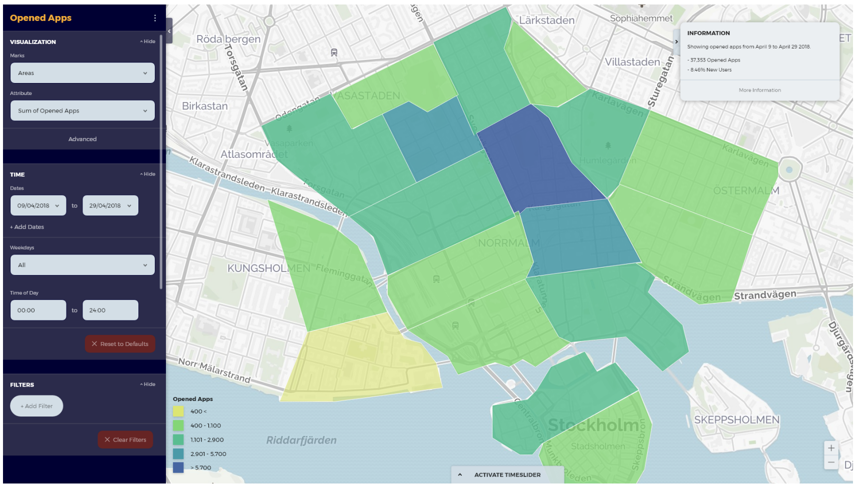

Figure C.10: BGIS displaying opened apps data.

C.4 Views

This section describes each view in the design in a more detail.

C.4.1 Map

The core of BGIS is the map containing the data marks. The map can bee zoomed in and out, either by the mouse scroll wheel or by using the zoom button located in the bottom right part of the interface (show in Figure C.11).

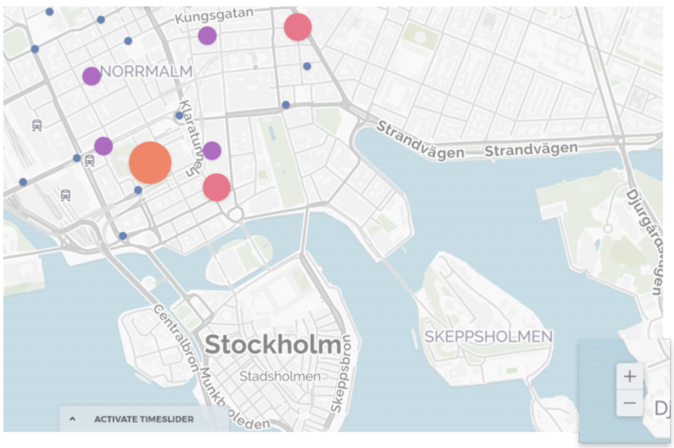

Figure C.11: The zoom buttons are used for zooming the map, and are located in the bottom right area of the interface.

Legends Marks on the map symbolizes data points. In the bottom left corner of the map *legends* for the marks are located. These legends describes the numerical values which the colors of the marks represents.

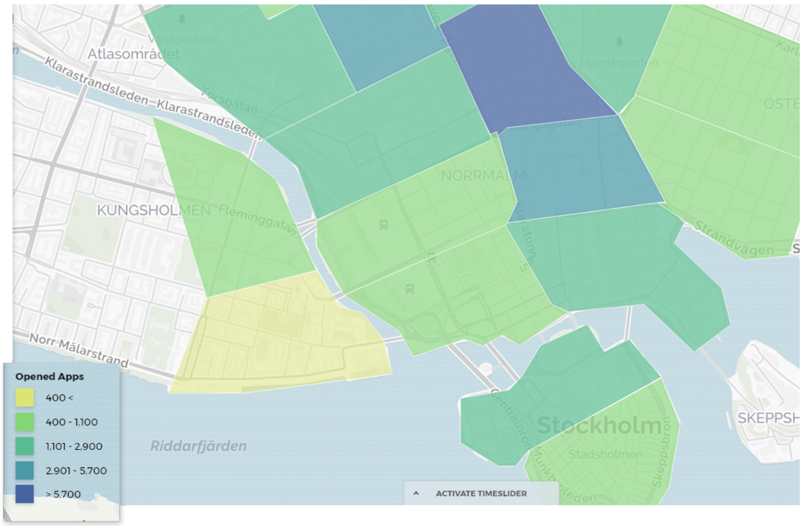

Figure C.12: Legends describe what the colors represents in numerical values.

C.4.2 Settings Menu

The Settings Menu, located to the left of the interface, contains settings for defining how and what data to visualize. It consists of three sub-menus: the visualization menu, the time menu and the filter menu.

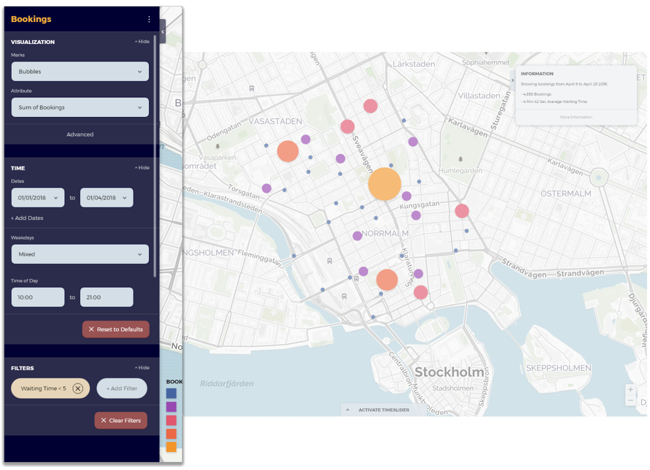

Figure C.13: The left menu.

Visualization Menu The visualization menu handles how data should be displayed. This includes what mark type to use, which attribute should be mapped to the mark's color variable and which color palette to display the data with.

Change Mark Type It is possible to change the mark type of the data visualization with the *Marks* drop-down menu.

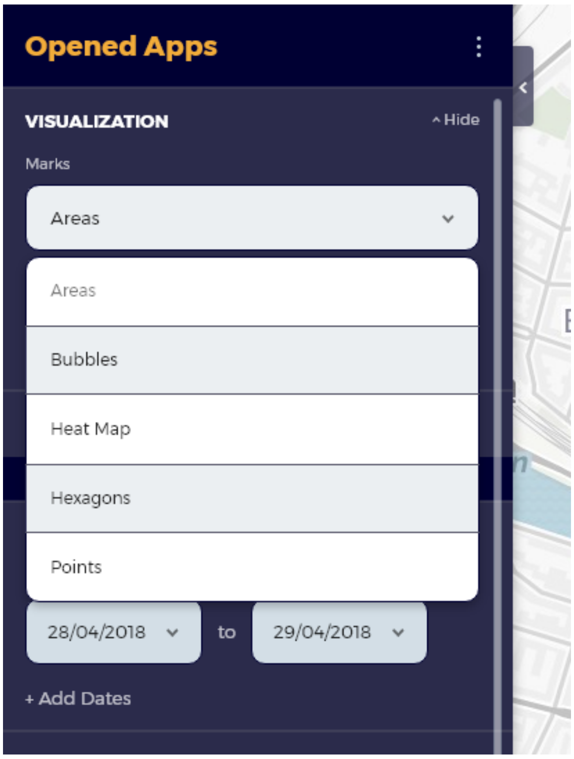

Figure C.14: Drop-down menu for changing the mark type used to display the data.

Change Attribute In the *Change Attribute* dropdown menu the user may change what attribute that the colors of the data marks should visualize. The default value is that the colors visualize the amount of events, but is also possible to for example show *Average Waiting Time* for bookings data or *States* for trips data.

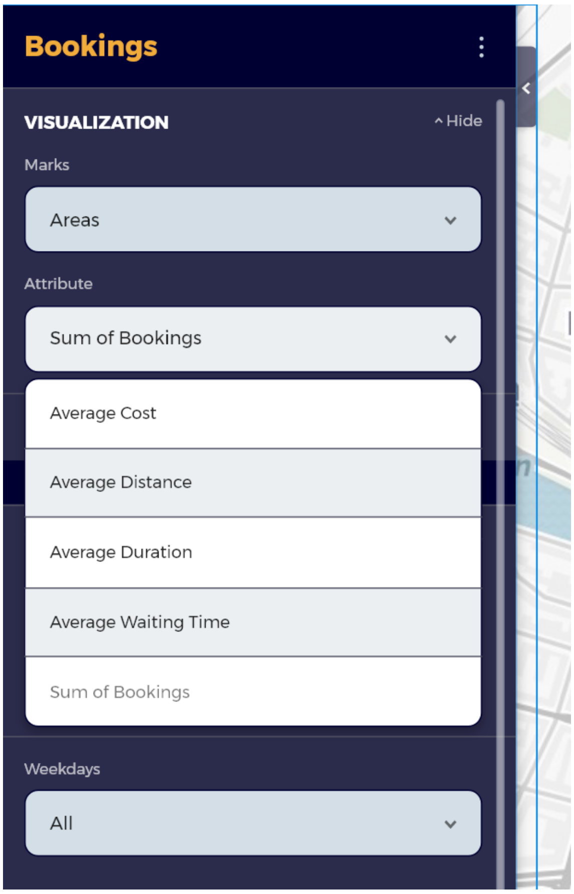

Figure C.15: Drop-down menu for changing the attribute that is represented by the colors of the marks.

Advanced Expanding *Advanced* in the Visualization menu shows additional, rarely used visualization options. In the current state of the design the only component hidden under *Advanced* is a drop down for changing the color palette of the data marks.

Time Menu The most central way of filtering data in BGIS are different time periods. Filtering data is performed at three levels of detail: *Dates Intervals, Days of Week* and *Time of Day*.

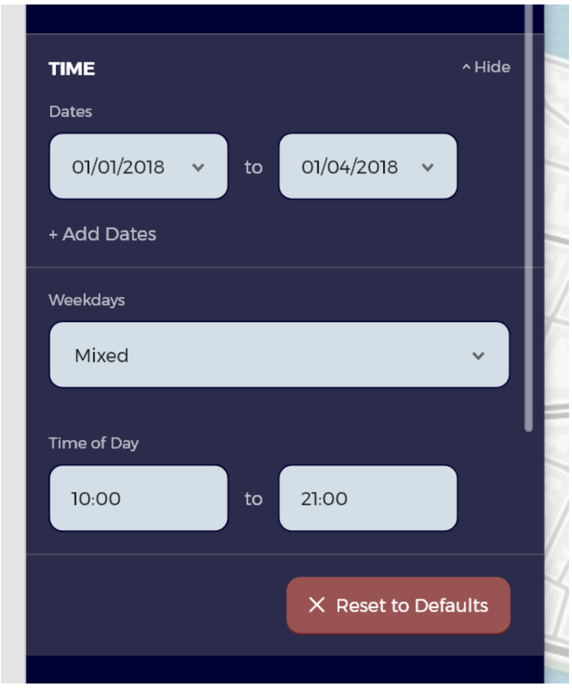

Figure C.16: The time menu contains settings for filtering the data in time periods.

Change & Add Date Intervals It is possible set precisely which dates you want to explore data from. Further more, a user may specify several such intervals simultaneously. This is useful to, for example, view data from the same period from several different years. By clicking the *+ Add Dates* Label a new date interval is added. It is also possible to remove date intervals by clicking on the cross symbol to the right of the interval, although you can never remove the last interval.

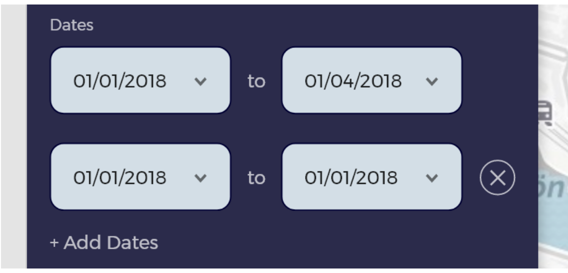

Figure C.17: BGIS supports filtering data on several simultaneous dates intervals.

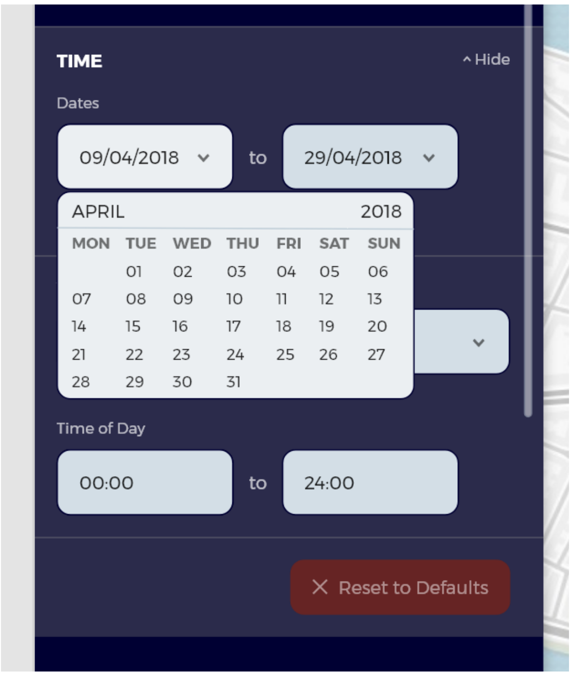

Figure C.18: The choose date dialog in BGIS.

Change Weekdays The *Weekdays* checkbox-dropdown allows the user to define which days of the week he wants to see data from. An example of this is shown in Figure C.19.

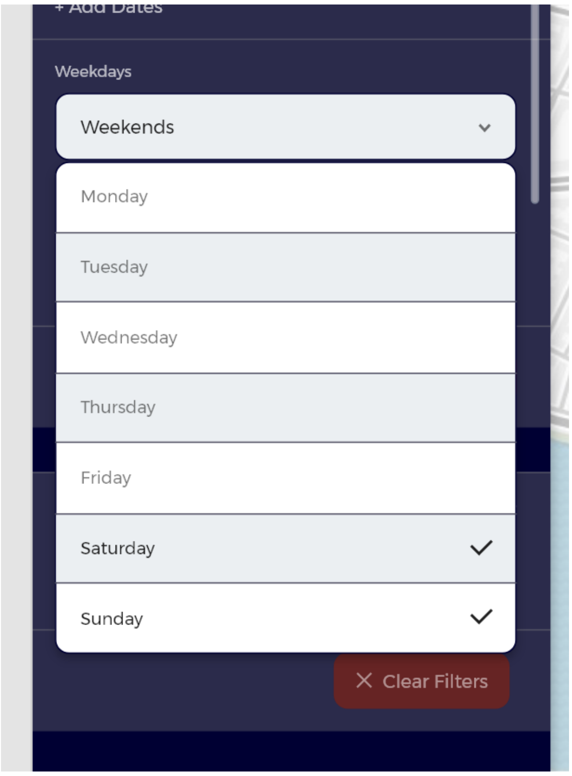

Figure C.19: BGIS supports filtering data on certain days of the week.

Change Time of Day The of day is simply defined by the user by writing a time with the keyboard or using a dropdown menu. A start time and an end time can be specified. This works similarly to how time input in a Google Calendar event works.

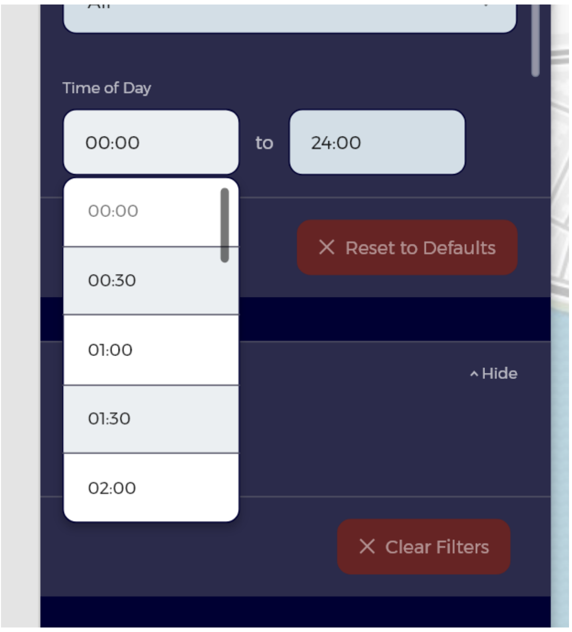

Figure C.20: Time input in BGIS works similar to Google Calendar.

Reset to Default The *Reset to Default* button resets all of the time input options to the default values. This is defined as all full days during the previous three weeks.

Filter Menu The filter menu contains attribute filters, meaning filters on attributes specific to the visualized data. It is possible to add several filters at the same time. Note however that filters are performed in the order that they are added. This means that the order of which filters are added might impact how the data is filtered.

The *Add Filter* button opens up the *Add Filter Dialog* which is used to add a new filter. This dialog is described later in the design document.

Clicking on an existing filter opens up the *Modify Filter Dialog*, which looks identical to the *Add Filter Dialog*. The only difference is that it modifies the clicked filter instead of adding a new one.

Clicking on the cross icon to the right in a filter removes the filter.

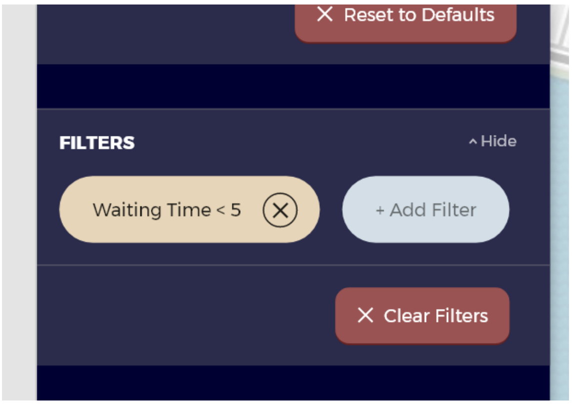

Figure C.21: The filters menu handles data filtering on aspects other than time.

Clear Filters A button which simply removes all current filters.

Add Filter Dialog In the add filter dialog the user can filter the data visualization based on the attributes of the currently loaded data point. For example, *trips* can be filtered on *Cost, Vehicle State, Vehicle ID* etc. The filter dialog is shown in Figure C.22.

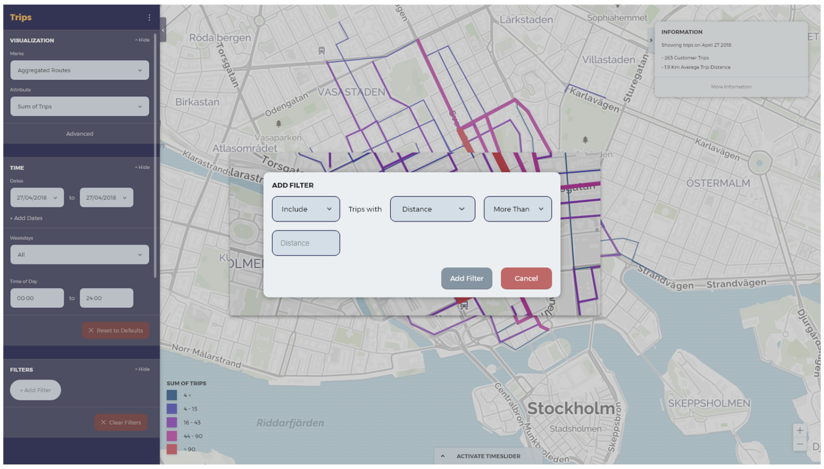

Figure C.22: The filter dialog is used to filter the data based on its attributes.

The filter dialog contains several input elements used to specify how the data should be filtered.

Include/Exclude Dropdown The first dropdown is used for specifying if the filter should include or exclude data. *Exclude* simply excludes the specified data from the visualization.
Include is not as simple as simple as *Exclude*. The default mode of the tool is to include all data, regardless of their attributes. If an *include filter* is added however,

only the data specified in that include filter will be included in the visualization. If two include filters are added, the data specified by both these filters will be included, and so on. Also, as stated before, the order of the applied filters also matters to end outcome of the combined filters.

Attribute Dropdown The second dropdown is the *Attribute Dropdown*, where a user chooses which attribute should be filtered. The attributes are specific to the currently visualized data type. Examples of attributes for trips are shown in Figure C.23.

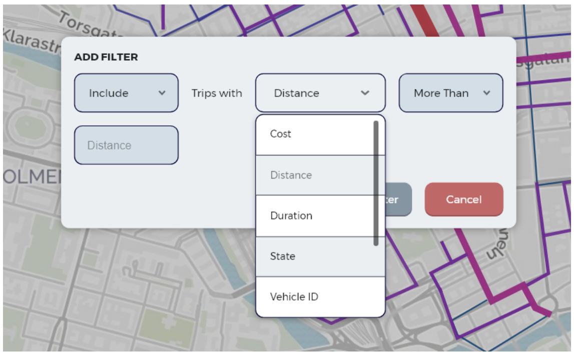

Figure C.23: The attribute dropdown menu allows the user to choose an attribute to filter the data on.

Conditional Operator Dropdown The condition of the filter. This is dependant on the attribute type. For numerical attributes the conditions *More Than, Less Than, Is Between* and *Equal To* exists. For text type attributes, conditions such as *Contains, Begins With, Ends With* or *Equal To* are supported. Exactly what conditions exists for each type of attribute has not been explicitly designed.

Attribute Specific Input The final input elements in the *Add Filter Dialog* are specific to the attribute the user is filtering on. In Figure C.23, the attribute *Distance* is filtered with the *More Than* condition. In this case, the remaining information that the user need to define is the numerical value on which the data should be filtered with. Consider if the *Is Between* condition had been used instead. In that case, two numerical values had to be specified: one lower end value and one higher end value.

Additional Options In the top right corner of the *Settings Menu* a three dotted icon is located, which expands the additional options menu. This menu contains two alternatives: *Switch Data Set* and *Share*.

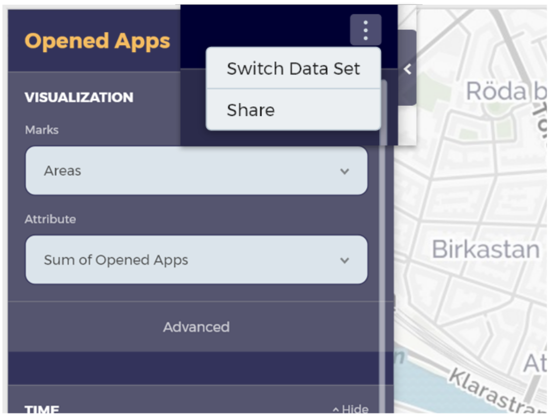

Figure C.24: The three dot icon in the top right corner of the *Settings Menu* expands the additional options menu.

Switch Data Set *Switch Data Set* simply lets the user switch the current data set of BGIS, for example from *Bookings* to *Opened Apps*. The interface how precisely to switch data sets has not been designed. The reason for this is that it is still slightly unclear if this is the preferred way of switching data sets.

Share Clicking on *Share* opens up the *Share Dialog*. In this dialog the user can export the current visualization in different formats. In current state of the design, these formats are *Excel* (Structured text) and *PDF* (a formatted document including an image of the visualization). It is entirely possible that these formats are not optimal. Another alternative to Excel is, for example, CSV.

It is also possible to share a link of BGIS. The link contains a hash of the current visualization settings. By opening the link, BGIS is initiated with the same visualization, filters and such. This is useful when several people are collaborating using the same data.

Figure C.25: In the *Share Dialog* the user can export and share the current visualization.

C.4.3 Information Window

The purpose of the *Information Window* are to clearly in text describe what data is currently being visualized, as well as providing the user with more information about the data and its attributes. Exactly what information that is shown in the information window is specific to the currently used data set. In Figure C.26 the *Information Window* is shown for *Bookings* data.

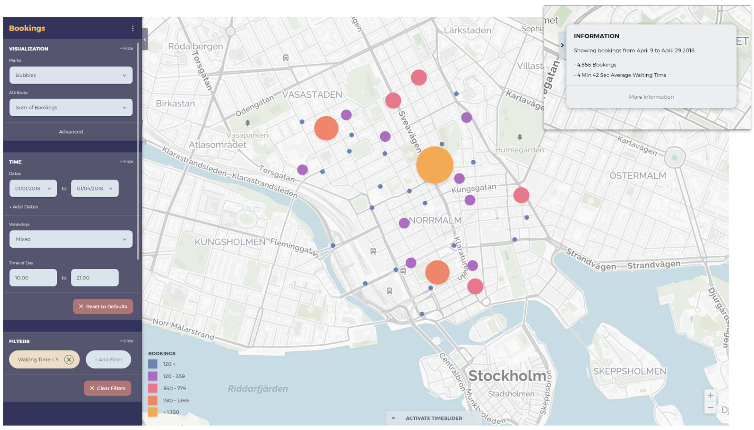

Figure C.26: The *Information Window* shows information about the current visualization in text.

The *Information Window* contains a *More Information* label. By clicking this, the *Information Window* is expanded to show additional information. When the *Information Window* is expanded in can potentially be so long that it requires a scroll bar, since a high number of relevant information may be shown.

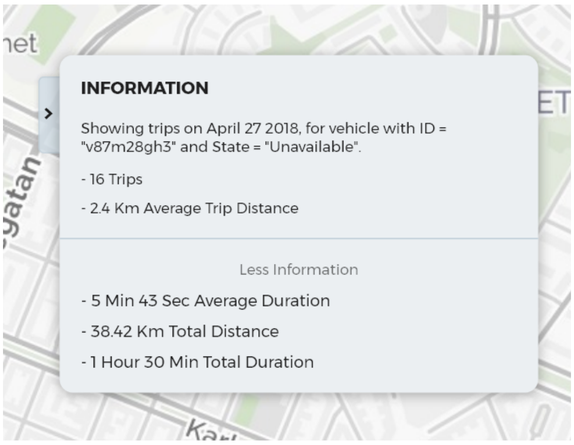

Figure C.27: By clicking the *More Information* label, additional information about the visualization are shown.

Selecting Marks The marks displayed on the map can be selected by the user. It is possible to select one mark or several marks simultaneously, using the mouse drag or *ctrl/shift-click*. When one or several marks are selected, the text in the *More Information* will change to only show information about the selected marks. An example of this is shown in Figure C.28. By clicking anywhere on the map not containing a selected mark, the marks are deselected.

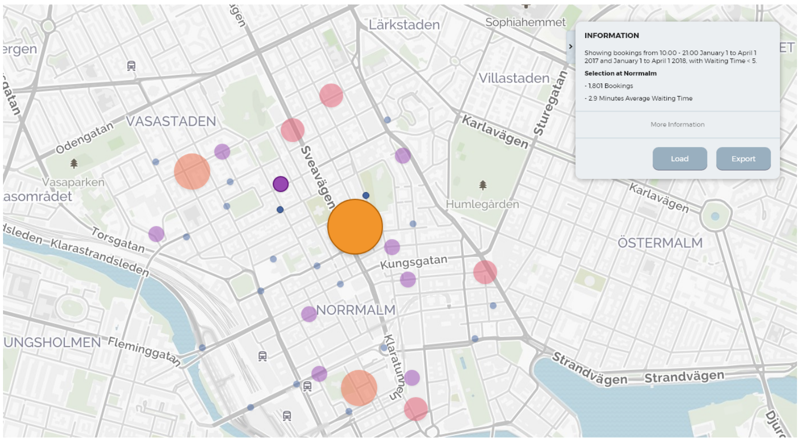

Figure C.28: It is possible to select one or several marks to see additional information about them.

C.4.4 Time Slider

One important requirement of BGIS is to support exploration of how data changes over time. The main tool of achieving this the *Time Slider*. The *Time Slider* allows the user to go through the current data (with the dates and time specified in the time menu) in smaller time steps. For example: consider bookings data currently filtered to show data from week January. By using the time slider, and setting the time step to *weeks*, the user can explore difference in the data week by week by dragging the slider.

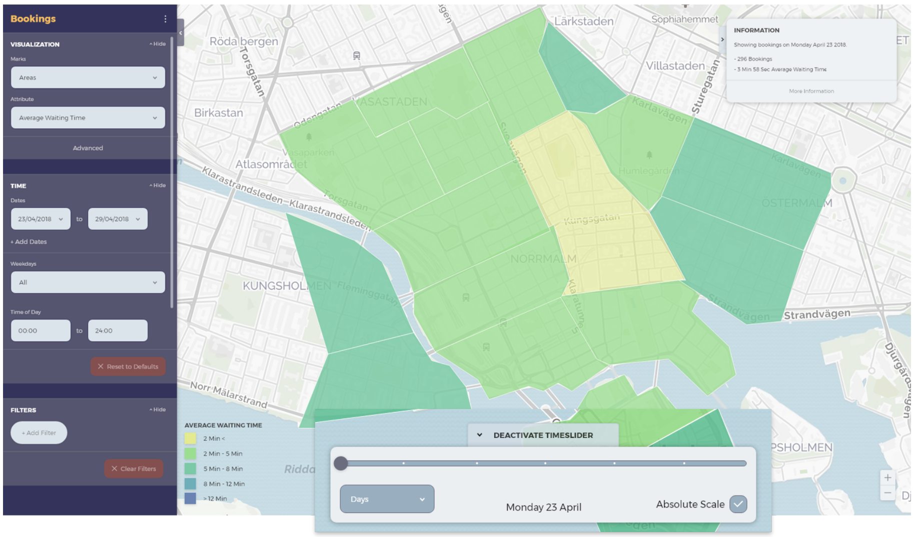

Figure C.29: The *Time Slider* allows the user to explore how data changes over time.

Time Step Dropdown The *Time Step Dropdown*, located in the bottom left corner of the *Time Slider*, allows the user to specify the length of the time steps for the slider. The allowed time steps are dependant on the length in time of the current data. If you are currently only showing data on the 24th of December, the maximum time step is *Hours*. It would not make any sense to have *Days* or *Weeks* as time steps, as the data is limited to a single day. This limitation also exists from the opposite direction. If the tool currently shows data from the whole years 2016, 2017 and 2018, the minimum time step would be weeks. Having smaller time steps than weeks would result in to many individual steps in the slider. An example of the *Time Step Dropdown* is shown in Figure C.30.

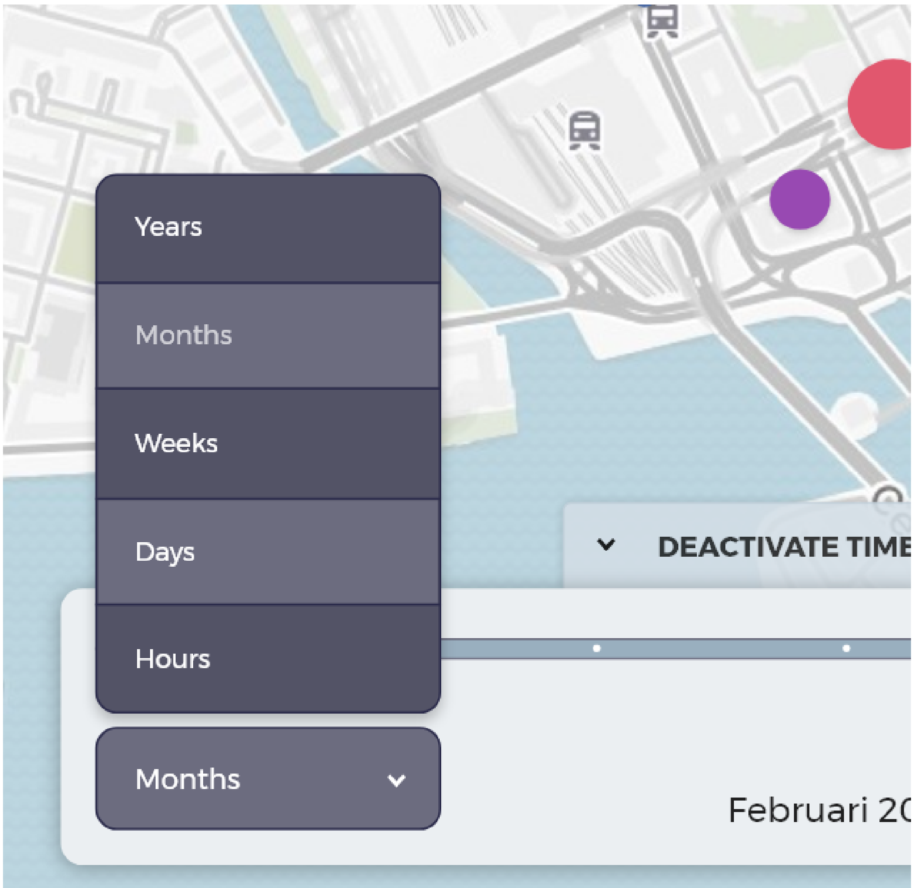

Figure C.30: Using the time step dropdown the user can specify the length of each time step in the slider.

Absolute Scale The *Absolute Scale Checkbox* decides the numerical values of the mark colors. If the *Absolute Scale Checkbox* is enabled, the threshold values of the mark colors will be the same as if the *Time Slider* was disabled. If the *Absolute Scale Checkbox* is disabled, the threshold values will be re-scaled to better match visualized data of the time steps.

C.4.5 Mark Types

In this section the different visualization methods currently designed for BGIS will be described.

Bubbles Bubbles visualization contains two variables: one for the size of the bubbles and another for the color. In BGIS, the size of the bubbles are determined by the amount of nearby events. If there are many events closely together, a larger bubble will be created. Bubbles visualization is useful for finding clusters in the data.
The color variable can represent various information. The default setting is that the color is also depending on the amount nearby events, but this can be changed in the *Settings Menu* so that the color instead shows some other attribute variable.

Bubbles are the only visualization type that changes depending on map zoom. When the map is zoomed out a lot, a larger geographical area of events will be aggregated into bubbles. The opposite applies when the map is highly zoomed in. *Bubbles can be selected.*

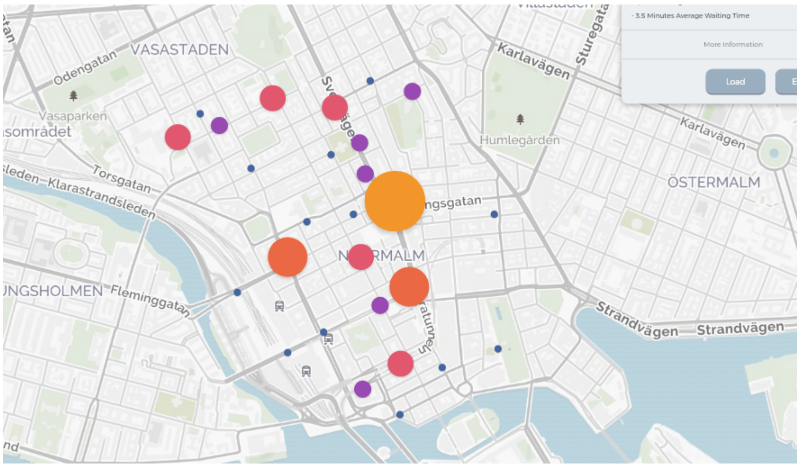

Figure C.31: Bubbles visualization.

Choropleth This visualization type is called *Areas* in BGIS. It uses a set of predefined areas. Choropleth visualization is useful for mapping data to areas that are relevant in the domain, and to see an overall pattern. Choropleth visualization only contains one variable, the color of the polygon. The default variable that the colors represents is the number of events in the area. *Areas can be selected.*

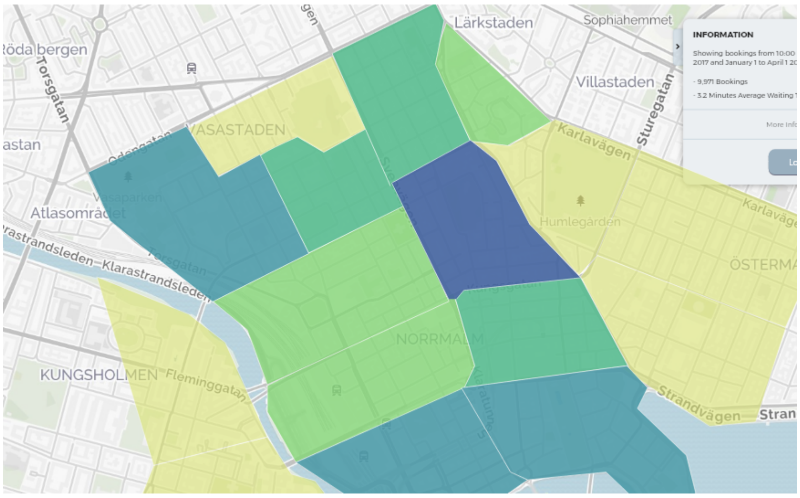

Figure C.32: Choropleth (areas) visualization.

Heat Map In a heat map does not correspond to any geographical boundaries, as choropleth maps do. The areas of the heat map are instead dynamically created depending on the aggregated data. Heat map visualization is great for visualizing the overall pattern of the data. The default variable that the colors of a heat map visualization represents is the number of events.

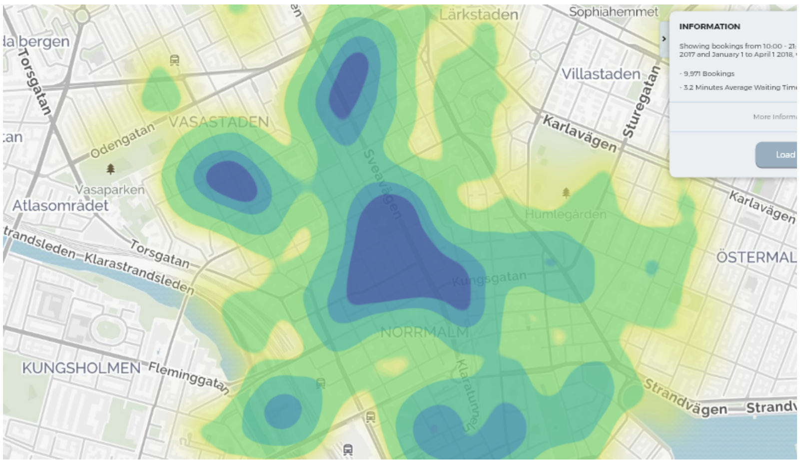

Figure C.33: Heat map visualization.

Hexagons Hexagon visualization is also called *Hexagonal Binning*. This visualization type is capable of both showing the overall pattern of the data and allows for fairly high accuracy in identifying a single point in the data. The map is simply divided into equally large hexagons, which contains a color variable. The default variable that the colors of a hexagon visualization represents is the number of events. *Hexagons can be selected.*

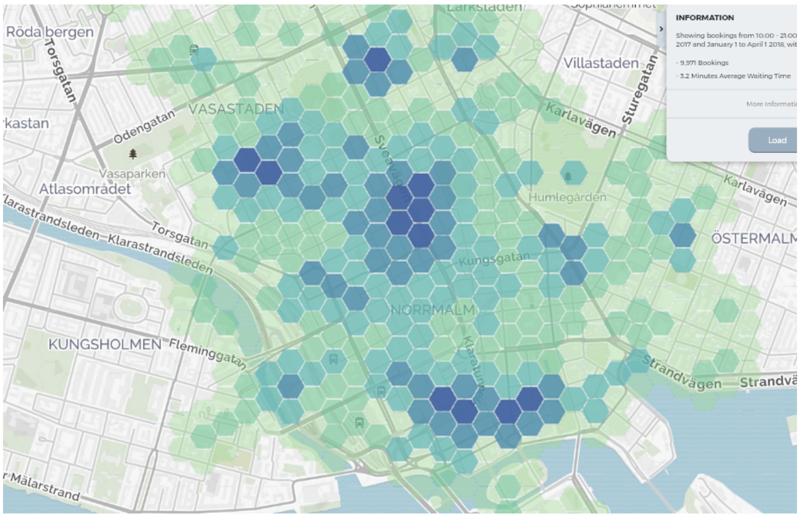

Figure C.34: Hexagonal binning visualization.

Dot Map In BGIS the dot map visualization method is simply called *Points*. Dot maps simply visualize each event as a dot on map. The dots are slightly transparent, so several overlapping dots results in somewhat darker color. The colors of the dots can be mapped to attributes in the data using the *Settings Menu*. Dot maps provide an exact accuracy, both can be ambiguous when visualizing data with many overlapping events. *Points can be selected.*

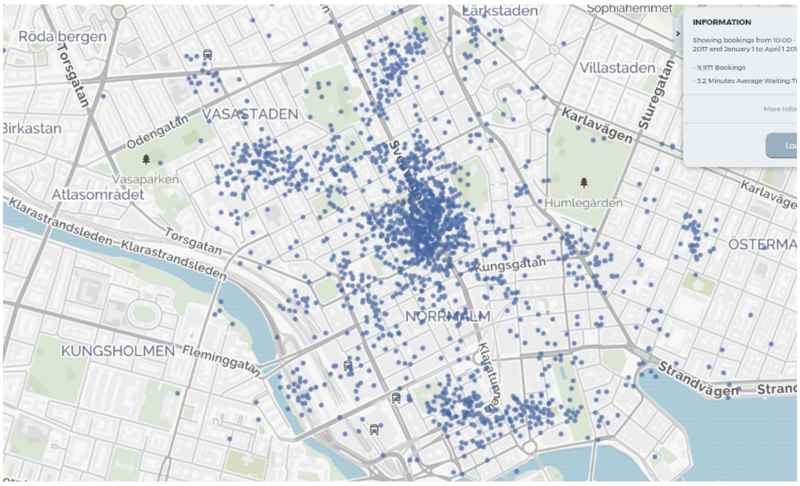

Figure C.35: Dot map visualization.

Poly Lines Poly lines are used for visualizing route type data, such as taxi trips. Similar to a dot map, each route is visualized as slightly transparent poly-line. This visualization method works well with few routes, but quickly becomes cluttered with to much data. The colors of the lines can be mapped to attributes in the data using the *Settings Menu. Lines can be selected, as shown in Figure C.38.*

Figure C.36: Single poly lines visualization.

Aggregated Poly Lines A better option of visualizing a high number of routes is to use aggregated poly lines. With this visualization method, routes are aggregated so that many routes forms a wider poly line. Aggregated poly lines have two variables: width and color. The width variable always represents the number of aggregated routes, and this is also the default variable of colors. However, the color variable can be changed to visualize other attributes, such as vehicle speed in the example of trips.

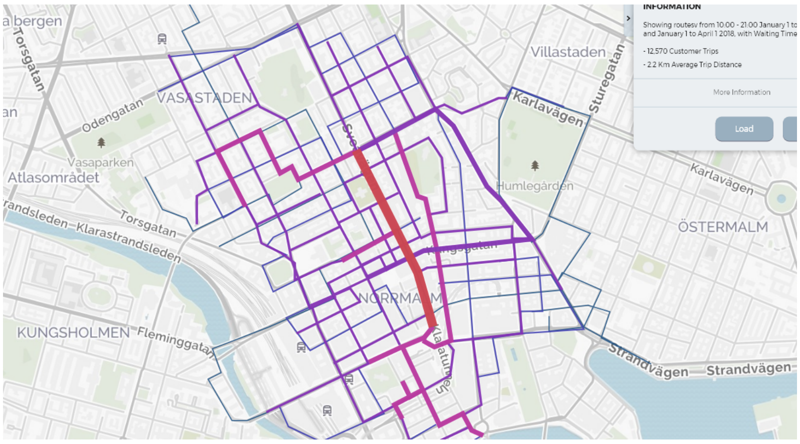

Figure C.37: Aggregated poly line visualization.

Selected Poly Line Single poly lines can be selected. When selected, the poly line is visualized with a direction. The start and end positions of the line are marked with different symbols, and a gradient is used to better show the direction of the route.

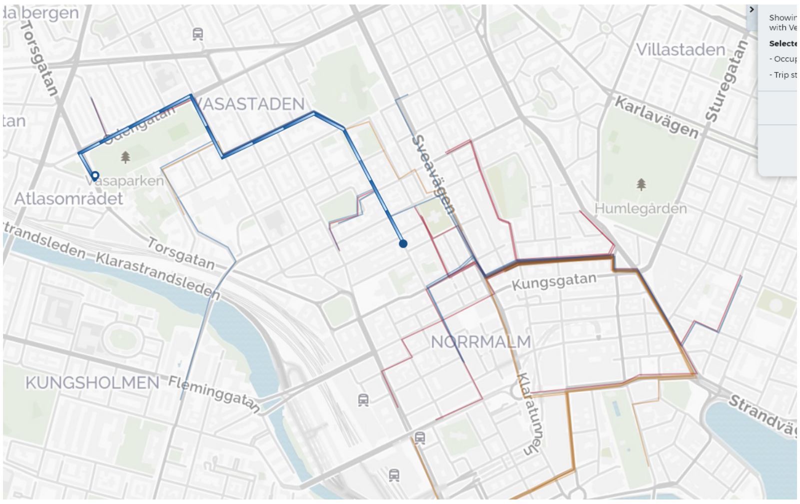

Figure C.38: A selected poly line which is visualized with a direction.

C.5 Look & Feel

This section briefly describes the look & feel of BGIS. This has not been the focus of the design (and not my area of expertise), although it has received some considerations. The goal of the look & feel is to make BGIS look minimalistic, sleek and modern. Most of the attention of the user should be focused on the actual data marks.
All of the details of the BGIS Look & Feel (including exact metrics, shadows, borders, margins etc.) will not be described in this document. This information is available in the BGIS Adobe XD Design Files.

C.5.1 Typography

BGIS only use one font family: **Montserrat**. It is a simple and modern font created by Julieta Ulanovsky. It is also free. BGIS contains four different text styles, all using Montserrat but with different sizes and modes. These text types are described below.

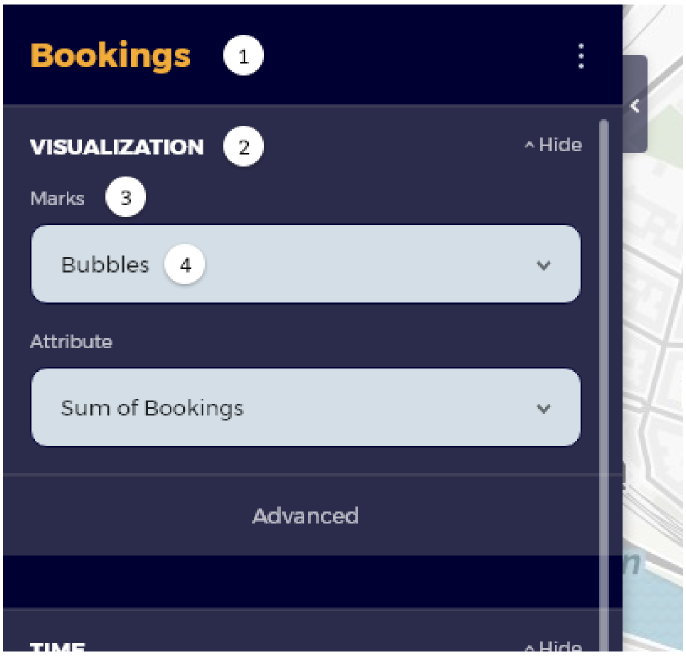

Figure C.39: The top part of the *Settings Menu*, where the different text styles are annotated.

1. **Title**: Montserrat Bold, 20 pt.
2. **Header**: Montserrat Bold, All Caps, 13 pt.
3. **Small Header**: Montserrat Regular, 11 pt.
4. **Normal Text**: Montserrat Regular, 13 pt.

C.5.2 Colors

In this section the colors used in the BGIS design are described.

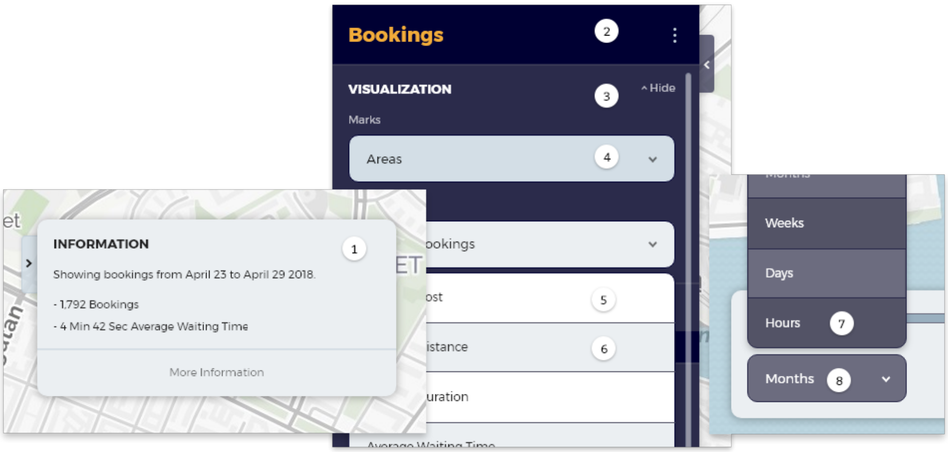

Figure C.40: Various parts of the BGIS interface with annotated background colors.

1. **Standard Grey Background**: #EBEFF2
2. **Settings Menu Background**: #000033
3. **Settings Sub-menu Background**: #2B2B4B
4. **UI Element Light Background**: #D3DEE6
5. **Dropdown Light Background 1**: #FFFFFF
6. **Dropdown Light Background 2**: #EBEFF2
7. **Dropdown Dark Background 1**: #6C6C7F
8. **UI Element Dark / Dropdown Dark Background 2**: #535366

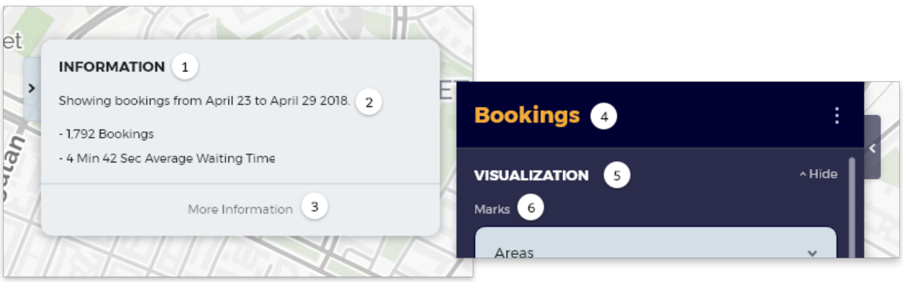

Figure C.41: Annotated text styles in BGIS.

Text Colors

1. **Title Black Text**: #000000 100% Opacity
2. **Normal Black Text**: #000000 83% Opacity
3. **Secondary Black Text**: #000000 54% Opacity
4. **Title Orange**: #F2AC3A
5. **Normal White Text**: #FFFFFF 100% Opacity

6. **Secondary White Text**: #FFFFFF 70% Opacity

Data Marks Color Palettes The data color palettes in BGIS uses multi-variate color coding; the colors used differs in several color channels, such as hue and lightness.

Blue - Green - Yellow Color Gradient

1. #4565A1
2. #4A9AA8
3. #59BD91
4. #81D672
5. #DCE570

Figure C.42: One of the color palettes in BGIS, which uses multi-variate color coding in hue and lightness.

Blue - Red - Orange Color Gradient

1. #4565A1
2. #9849B2
3. #E05A70
4. #EB6844
5. #F2962C

Figure C.43: Another color palette in BGIS.

Dark Red - Blue Color Gradient

1. #CC4752
2. #BA3AA0
3. #8533B5
4. #4040B2
5. #184C7F

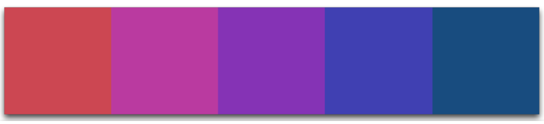

Figure C.44: The color palette in BGIS which is used for aggregated poly line visualizations.

States Color Gradient

1. #3CAB8F
2. #9E0E26
3. #B86200
4. #16518C

Figure C.45: The color palette in BGIS which is used to visualize different states.

D

Research Discoveries

Various discoveries from the divergence phase, primarily from interviews. The findings are roughly categorized as *requirements, data-points, goals, insights* and *research.*

D.1 Target User

Non-technical operator.
Is involved in making business decisions for Bzzt.
Understands Bzzt operations quite well.
Uses data for insights and support in business decisions.
Uses data as communication tool, for example between team member or for investors.

D.2 Requirements

Easy to access.
No setup required.
Fun
Simple to use & intuitive.
Shneiderman's Mantra: Overview first, with the ability to zoom in and get details on demand.
Good looking.
Possible to combine data.
Possibility to filter data.
Map with possibility of zooming.
Showing the same data in different granularity/aggregation.
Aggregate flow/directional/path data.
Dynamic time. For example only show events from Thursdays.
Show data over time with a time slider.
Simulate data over time.
3D Map
Choropleth Maps
Choropleth Map with differently scaled regions.
Dot map
Heat map

Hexagon grid
Polylines
Fat aggregated polylines.
Only show data for specific region.
Explain the data in detail.
Show the numerical values.
Different Layers of data.
Data should be easily available to the right people.
Simulate several data points simultaneously. Explore different constellations of data.
Possible to play around with the data.
Modular: possible to add new features and data-points.
Visualize the same data in different ways.
Relative and absolute visualizations of data points and data aggregations.
Show position differences.
Difference between pickup spot and estimated pickup spot.
Minority report L&F
Swedish & English
Clearly explain data attributes
Display data in plain text.
Dynamic aggregation of data depending on zoom level.
Filter on week day.
Filter on month of year.
Filter on days of year.
The ability to easily export visualization to documents such as pdf or powerpoint.
Maximize/Minimize settings windows.

D.3 Data Points

Estimated time [pos]
Estimated time that leads to a booking [pos]
Estimated time that does not lead to a booking [pos]
Waiting time [pos].
Diff in estimated time and actual waiting time [pos]
Filter on retention, only shows trips, bookings that is made by people that has already made x amount of trips.
Destination [pos]
Pickup [pos]
Pickup to destination path (bird's way) [line]
Interest (opened apps) [pos]
Trip Efficiency (revenue/duration) [path]
Weather
Waiting time [pos]
Pod-taxi driving paths [path]
Suggested driving paths [path]

Diff between suggested and actual path [path]
Pod-taxi efficiency (trips/hour or revenue/hour) [single-pos]
Pod-taxi locations [pos]
Pod-taxi states [single-pos]
People in street [area-pos]
Pod-taxi duration in commercial zone [area-pos]
Impressions [area-pos]
Traffic
Pod time-to-customer [single pos]
Price [polyline]
Estimated price [polyline]
Difference between price and estimated price [polyline]
Ordered pickup spot [pos]
Booking availability / paid time [single pos]
Cancelled bookings [pos]
Refused booking requests [pos]
Paths to take break [polyline]
Paths to refuel [polyline]
Bookings / demand [area pos]
Flow of vehicles [directional polylines]

D.4 Goals

The main purpose of Bzzt is that people have emotional trust for Bzzt. They trust us to be the fastest, safest, most ecological option for inner city travels. And also bloody cheap. -Anders
Get insight into peoples and the service behaviour.
Understand the flow of pods and people during different times and circumstances.
A customer should not need to wait more than four minutes.
Evaluate hypothesis with data.
Data visualization can be very useful as a communication tool.
Make it easier for non-technical people to explore and talk about data.
Understand the real world; how does things work out there?
Optimize KPI's.
Get a better understanding of the commercial service: impressions, people, pods etc.
Understand high demand cluster during different times.
Make things as easy as possible for the drivers.
The most central goal of the tool is making it easy to acquire the most essential data.
Controlling the vehicle fleet as a whole in a better way.
Better re-fuel and break planning.
Track behaviours of drivers.

D.5 Insights

Waiting time & Supply / Demand are two critical data points; it is critical to optimize these.
Pretty much all of the data points proposed are interesting.
Some are hard to get however, such as people in the streets.
Time and flows are important factors in a geographical visualization.

D.6 Questions & Research

Which data fit each other for combination/filtering?
What is the best way to visualize aggregated flows?
What is the best way to visualize geo data over time?