FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Towards a Data Visualization Dashboard for Smart Cities

Pedro Miguel Dias Soares

Mestrado Integrado em Engenharia Informática e Computação

Supervisor: Rosaldo J. F. Rossetti, PhD

July 25, 2019

Towards a Data Visualization Dashboard for Smart Cities

Pedro Miguel Dias Soares

Mestrado Integrado em Engenharia Informática e Computação

Approved in oral examination by the committee:

Chair: Ana Paula Rocha, PhD
External Examiner: João Emílio Almeida, PhD
Supervisor: Rosaldo J. F. Rossetti, PhD

July 25, 2019

Abstract

Nowadays, the cities continue to grow and become more dynamic and complex than ever, generating a huge variety and amount of data from multiple sources. The social networks, part of modern life, a stream of data 24/7 almost in real time can be used to understand the forces that shape city dynamics. When aggregated with information retrieved from, for example, the transportation network, it can adjust the demand in transports. The problems have become multidisciplinary and requires different expertise making the extraction of information from multiple sources hard for smart cities; and specialized tools are less desirable since they are generally focused on limited number sub-systems and are not flexible enough to adapt to constant changing system or to integrate new domains. By providing a tool capable of extracting and visualizing this heterogeneous information can be helpful to the multiple city departments and managers, so their decisions are supported with real data, allowing them to recognize patterns or trends in multidisciplinary problems and adjust it to their needs. It is also important to refer that this kind of tools are not limited to them, and could be used by communities to access data and evaluate their local problems or in education by providing information for multiple subjects and support projects.

Currently, the available tools have some limitations. First, the information extraction is confined to one sub-system or a set of sub-systems that represent their domain. This can have consequences by not having access to the complete system, some relations may go unknown and full comprehension of the domain may become unreachable. Second, many have information in a format not compatible with other systems or not commonly used and in other cases, the data is not easily accessible because they require a complex navigation on the web. Finally, some of the tools do not allow the user to explore the data for the indicators that are looked for, making the usefulness of such tools near to null. This dissertation focuses on the visualization of extracted information from multiple sources such as sensors, data provided by government, private sector or communities, more precisely in multidimensional data by relating the spatial-temporal nature to the variety of potentially available knowledge merged into one tool.

The contribution of this dissertation is a review of multiple metaphors of multidimensional visualization and a framework flexible enough that supports the heterogeneity of urban data. This framework allows the visualization in a dashboard that contains data flow from the source to the end-user. The final product of this dissertation is capable of representing the data by allowing visualization, querying and exploration of urban data.

Keywords: Human-Computer Interaction, Multidimensional Data, Smart Cities, Visual Analysis, Spatio-temporal.

Resumo

Hoje em dia, as cidades continuam a crescer e a tornar-se cada vez mais dinâmicas e complexas que nunca gerando uma grande quantidade de dados das mais diversas fontes. As redes sociais, parte do estilo de vida moderno, geram uma fonte de dados 24/7 quase em tempo real que pode ser aplicada para compreender as forças que moldam uma cidade e quando agregada com outras fontes de informação, por exemplo, a rede de transportes, ajustar a oferta. Os problemas tornaram-se multidisciplinares e requerem uma diversidade de conhecimentos tornando a extração de informação difícil e ferramentas especializadas menos desejáveis sobretudo por se focarem num determinado grupo ou subconjunto da cidade. Além disso, não são flexíveis para se adaptarem as mudanças constantes do sistema, ou integrarem novos conceitos. Uma ferramenta capaz de extrair e visualizar esta heterogeneidade de informação pode ser útil aos diversos departamentos e gestores, tornando as suas decisões apoiadas por dados, permitindo que reconheçam padrões ou tendências em problemas multidisciplinares e a ajustar as suas necessidades. É importante referir que esta ferramenta não estaria limitada apenas a eles, também é aplicável as comunidades dando-lhes acesso a dados para poderem resolver os problemas locais ou na educação fornecendo informação para projetos.

Atualmente, as ferramentas disponibilizadas têm algumas limitações. Primeiro, a informação extraída esta restrita a um determinado subsistema ou a um conjunto de subsistemas que representam um domínio, mas isto pode ter consequências, algumas relações podem continuar desconhecidas e a compreensão do total do comportamento de uma cidade nunca ser alcançável. Segundo, muitas ferramentas usam formatos de informação não compatíveis com outros instrumentos ou pouco conhecidos, além disso, muitos dados não são facilmente acessíveis requerendo uma navegação complicada por pagina da Internet. Por último, muitas ferramentas não permitem ao utilizador selecionar as informações de interesse, tornando a sua utilidade quase nula. Esta dissertação foca-se na visualização de dados extraídos de múltiplas fontes como sensores, fornecidos por entidades públicas ou privadas, mais precisamente multidimensionais, relacionando a sua natureza espaço-temporal para possibilitar a extração de conhecimento numa só ferramenta.

Esta dissertação contribui com uma análise das múltiplas metáforas de visualização e uma ferramenta flexível capaz de suportar a diversidade de informação e fontes, permitindo a visualização de dados urbanos. Além disso, os dados são integrados desde a fonte ate à visualização por parte do utilizador. O produto final pode representar dados permitindo a visualização, pesquisa e exploração.

Palavras-chave: interação Humano-Computador, Dados Multidimensionais, Cidades Inteligentes, Análise Visual, Raciocínio Visual, Espaço-Temporal. .

Acknowledgements

Firstly, I would like to say a big thanks to my supervisor Professor Rosaldo Rossetti, whose expertise was invaluable and for all the support and encouragement he gave me. Especially, for being the guide through the changes, always confident on the new course and transmitting security. To Thiago Sobral, for the opportunity to work with his project and for the availability to help me in every question that I had and whose talks resulted in new ideas for this and future projects. To Sara Paiva for the contribution in the early stages, always on the scene and ready to help when was needed.

Some special words of gratitude go to my friends who are a major source of support, inspiration, and happiness. For their patient and encouragement was worth more than I can express in written words.

Finally, I must express my very profound gratitude to my family for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of writing this dissertation. This accomplishment would not have been possible without them.

Thank you very much, everyone!

Pedro Miguel Dias Soares

“The world that we have made as a result of the level of thinking we have done thus far creates problems that we cannot solve at the same level as the level we created them at.”

Albert Einstein

Contents

List of Figures

LIST OF FIGURES

List of Tables

LIST OF TABLES

Abbreviations

VAUD	Visual Analysis Approach for Exploring Spatio-Temporal Urban Data
WIMP	Windows, Icons, Menus, Pointer
Post-WIMP	Post-" Windows, Icons, Menus, Pointer"
VUMO	Visualization-oriented Urban Mobility Ontology

Chapter 1

Introduction

This chapter starts by describing the scope of the project with a short presentation of the current gap in the solutions available at this time. Then, in Section 1.2 the problem tackled and motivations are presented, followed by the definition of objectives and goals in Section 1.3. The chapter ends with Section 1.4, which contains the document outline shortly describing the content on each chapter.

1.1 Scope

Data is part of modern life, for most persons whatever happens to them or around generates data. Social networks become one of the information sources for smart cities. This type of data can have a huge value since multiple and complex forces shape city dynamics as well as the characteristics of different communities. Also, the extraction of this information can have a role on how cities are managed, by providing real-time information to the different departments responsible for managing a city, especially when services have a low budget and staff [RK15]. Such a tool is expected to support the decision-makers, as they can make informed and therefore better decisions relying on real data, and recognize patterns or trends in multidisciplinary problems and adjust it to meet their needs.

For the information to be useful, the method that is presented to the user plays a key role in how effective this kind of tools can be. If the important part of the available information is not on focus or hidden, the user may not recognize a problem or symptom, making it ineffective. It is also important to consider the interaction between the user and the dashboard because the relationship needs to be natural, otherwise the learning curve of the tool can be high and affect the receptiveness. In information visualization problem, it is also essential to reference the multiple challenges considering the diversity of available knowledge, along with the requirement to navigate throw multidimensional space and the presentation of data be effective.

1.2 Problem Statement and Motivation

Urban indicators have an enormous amount of data sources from which they can be generated making hard for each scenario to have the appropriate visualization. Also, a dashboard that supports all this variety of indicators represent some challenges. First, this kind of tools tend to be specific for each case and a generic solution is not standardized. Second, due to the variety of possible indicators, the dashboard needs to be extensible and flexible so data types support and visualization techniques can be implemented or removed has required. And the third challenge, related to human-computer interfaces where a user or operator must be able to process information that the dashboard generates and displays.

As for the potential contributions of such a dashboard one may identify its ability to support decision-making resorting to multiple sources of data to infer useful urban indicators. Also, appropriate visualization techniques can have an impact on and play an important role in policy making in urban planning and management, both by practitioners and by researchers. A natural outcome of such tools in practical terms is certainly an enhanced quality of life for the citizens, as such visual representations are a key ingredient in the process of devising, testing, and implementing appropriate measures to promote a sustainable city development.

1.3 Aim and Goals

The aim of this project is to apply data visualization techniques to urban data with the objective of turning the visualization more intuitive and a dashboard flexible enough to support the heterogeneity inside urban data. To achieve this, it is necessary to review the data visualization techniques available today and assess their applications in this type of data on a dashboard context. Also, for the dashboard, a review of current solutions and approaches, define and refine the components, analyze the result and future improvements.

1.4 Document Outline

Besides the introduction, five more chapters compose this document. The literature review on data visualization, interactions, urban indicators, and current solutions is presented in Chapter 2. Then, in Chapter 3 a list of requirements followed by the proposed architecture for the framework, that explained in detail each component. After Chapter 4 contains details for the current implementation of the solution. Then, in Chapter 5 two case studies and their results. For last, Chapter 6 contains main considerations, conclusions, and future work.

Chapter 2

Literature Review

The chapter focus on a comprehensive review in the four major components for this dissertation: urban indicators, data visualization metaphors, interactions and current available solutions. The Section 2.1 describes the concept of urban indicators and possible indicators to be included in the solution. Section 2.2 contains a review of visualization metaphors subdivided in multiple categories. Then on Section 2.3 a review of interaction composed by WIMP and Post-WIMP interactions. Finally, on Section 2.4 review the solutions available and does a comparison to understand the gap that exists nowdays.

2.1 Urban Indicators

2.1.1 Definition

Urban indicator can be defined has a variable or a set of variables aggregated that provide information about a condition, sign or tip about something of interest, can also be described has "an operational representation of an attribute (quality, characteristic, property) of a system" [Gal97]. For an indicator to be useful, need to have a reference value such as a goal, a norm or a benchmark [Gal97] which allows the comparison to a target value to retrieve the performance of a system. As show on the Figure 2.1 from data, indicators can be extracted [Seg02] to generate information and then, knowledge.

2.1.2 Classification

Using the previous definition, an indicator can represent almost any part of interest in a given system, making the classification of the whole set hard to realize. The proposed approach focus in select indicators that represent the sustainability of the system in study.

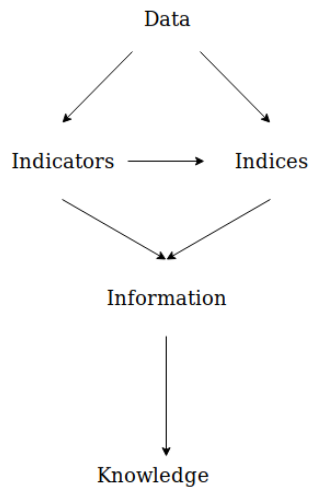


Figure 2.1: From data to knowledge adapted from [TW14]

2.1.2.1 Urban Sustainability indicators

Urban sustainability has been described in multiple ways since multiple disciplines uses different criteria and emphasis some characteristics. Many list of urban sustainability indicators have been promoted by multiple international organizations such as UN Habitat [Hab04], World Bank [Nat07], the European Foundation [FOU98]. These have been use as a reference for multiple cities, countries and communities around the world to build urban indicator systems.

A list of indicators called IUSIL have been proposed to be a comparative base to different sustainability projects [LYS11], dividing the 115 indicators in 4 categories: Environmental, Economic, Social and Governance and compares 9 cities urban indicators projects around the world. On the study is also suggested that in the first interaction for indicators inclusion a reduced number of indicators and add or eliminate indicators to match the constantly changing needs.

Taking into account the previous projects REFS a list of possible indicators to be integrated is presented on table 2.1 .

Table 2.1: Possible indicators based in previous projects

Visual Indicators	Social-Network Indicators	Transportation Network Sensors
Green areas	Topic trend	Traffic intensity
Small Vegetation zones	Traffic congestion	Network flow
Transportation network presence		

2.1.3 Indicator Quality Criteria

Hezri [Hez04] classified the quality criteria in four categories: robustness, democratic inclusion, longevity and relevance. T. Wass et al. [TW14] based on the quality criteria reviews of Gallopin [Gal97], Hezri [Hez04] and Van de kert et al. [GvdK08] and using Hezri classification identified the quality criteria presented on table 2.2. This criteria can be used as a scratch for checklist when assessing the inclusion or exclusion of indicators in the dashboard context. Its important to empathise that find a indicator that meet all criteria is hard to achieve and also, the quality criteria can't just be an checklist of all criteria because its purpose may not require all criteria [TW14].

Table 2.2: Quality indicators criteria adapted from [TW14]

Heinz Category	**Criteria**
Robustness	Scientifically credible Clear and standardized methodology to facilitate comparison Values should be measurable Data should be available/obtainable/reliable/up-to-date Sensitive to changes Practical focus such as a limited number of key issues Based on models with holistic perspective Appropriateness of scale No overlap Independence between indicators
Democratic Inclusion	Participation of and supported by stakeholders, experts and policymakers Openness with accessible sets of indicators, methods and explicit judgments
Longevity	Capacity for repeated measurement Iterative and adaptive to change Cost-effective Resource availability
Relevance	Institutional capacity for data collection Maintenance and documentation Meets the needs of audience and users Presentation in understandable structure Guided by clear vision of sustainability and relevant for its sustainability attributes

2.2 Data Visualization Metaphors

In this section, the reader is presented with a extended review of multidimensional data visualizations metaphors.

2.2.1 Classification

For this topic the visualization techniques are organized using the classification presented by Keim and Kriegel [Kei97] [KK96a] that divides them in six categories: geometric , icon-based , pixel-oriented , hierarchical , graph-based and hybrid techniques. All categories have a comprehensive review in the following sub-sections.

2.2.2 Geometric Techniques

Geometric techniques focus on the visualization multidimensional points using selected geometric shapes has axes.

2.2.2.1 Scatter Plot

Scatter Plots are a popular metaphor for two-dimensional or three-dimensional points that uses the Cartesian coordinate system. Brushing [BC87] and colored class points are two methods generally used to improve the recognition of patterns but glyphs, icons, color, and splatting have also been used to improve the utility. In three-dimension scatter plots by using animation, colors, shapes and interactions can be extended to higher dimensions.[PH01]

For a higher dimensional data a matrix of scatter-plots is the standard way to go, where an array of scatter plots are organized in a matrix showing all pairwise combinations of features. This technique allows to easily observe clusters, outliers, trends and correlations [PH01] but higher dimension patterns are hard to recognize.

2.2.2.2 Radial Coordinates Visualization

Also known as RadViz and their similar metaphors (PolyViz and GridViz) [PH01] are non-linear multidimensional visualization techniques that projects into 2-dimensional space data with three or higher dimension. The visualized variables are presented as anchor points equally spaced around the perimeter of the circle (RadViz) or other shape, the items are showed as points and their position is calculated based in a physic metaphor, where springs attached to each item feature, making items with equal values in each feature closer to the center.

2.2.2.3 Prosection Matrix

The Prosection technique combines slicing and projection by taking sections of the data set and projecting the items on the desired plane. Tweedie et al. [TS98] extended the method for a high-dimensional data by using multiple prosections views in a matrix allowing the visualization for higher dimensions

2.2.2.4 HyperSlice

HyperSlice [vWvL93] uses a matrix of orthogonal two-dimensional slices to represent scalar functions at focal points of interest, in other words, allows the user to visualize multidimensional space around a point.

2.2.2.5 Parallel Coordinates

Parallel coordinates for multidimensional visualization was proposed by Inselberg [Ins85] where the data characteristics are represented by n vertical parallel lines and each item is defined by a polygon line that intersect the multiple axes. When the coordinated are too close makes difficult to understand the data structure. Also, when the set of items is large the interpretation of the results are hard to achieve. Hierarchical parallel coordinates is a variation of parallel coordinates [FWR99] that focus on that problem by aggregating data in clusters that are displayed using brushing and color intensity. Another variation is the circular parallel coordinates that instead of using the vertical axes uses circular ones.

2.2.2.6 Andrews Curve

Andrew plot can be described has a smoothed version of a parallel coordinates plot since the items are now represented by curves instead of polygon line. Andrew plot uses a finite Fourier Series that makes computational expensive to display large data. Common uses include quality control and outliers detection in a times series.

2.2.2.7 Star Coordinates

Star coordinates [Kan00] is similar to a scatter plot but for higher dimensions. The items are show as point and each axe represents a feature. This technique allows scale and rotation transformations allowing to change the contribution of each feature and change the correlation between a feature and the others.

2.2.3 Pixel-Oriented

This category of techniques focus on represent as many data objects as possible on the screen at the same time by mapping each data value to a pixel of the screen and arranging the pixels adequately. This techniques can be divided in two subcategories [Kei96]: query-independent techniques and query-dependent techniques. The first category, try to present many data as possible, only limited by the number of pixels available and the second focus on visualize the data to give users feedback on their queries, in a specific context.

2.2.3.1 Space-Filling Curves

Space-filling curves are query-independent and provide a continuous curve that visit every point in regular spatial region [Kei96], some well-know curves are Peano, Hilbert and Morton. The space-filling can map multidimensional space to one dimensional sequence allowing to find patterns or clusters. One disadvantage of this set of techniques is only applicable in $2^i * 2^i$ shapes.

2.2.3.2 Recursive Pattern

Recursive Pattern is also a query-dependent which try to solve the square shape requirement of space-filing curves. It keeps the clustering properties but allowing the user to arrange the pixels turning the result semantically meaningful [Kei96].

2.2.3.3 Spiral and Axes Techniques

Spiral and axes are both query-dependent. In the spiral techniques the answers to the query are sorted by relevance in a wave spiral shape around the center, where the most relevant data is located. [Kei96]. The axes techniques are similar but improves the spiral technique by integrating some feedback on displacement [Kei96],in other words, assigning two variables to the axes and dividing the data in four quadrants and arrange it using partial spirals, the amount items in each quadrant can vary depending on the data set.

2.2.3.4 Circle Segment

As the name suggest divide the circle in n-dimensions (segments), the items attributes are places in the same position in each segment and the colors and position calculate taking account the relevance in the query context.

2.2.3.5 Pixel Bar Chart

Pixel bar chart [KHDH02] is an extension of bar charts but instead of aggregating data into few values, each data item is represented in the bars as a pixel with the appropriate color. The data is ordered by like in the old fashion x-y diagrams.

2.2.4 Hierarchical Techniques

Hierarchical techniques focus visualize subspaces by using hierarchical partitioning [Kei97], with the exception of treemap this techniques focus on multivariate functions making them not the best candidate for data mining [KK96a].

2.2.4.1 Dimensional Stacking

Dimensional Stacking represents high-dimensional data into lower dimensions by stacking dimension inside other dimensions [Kei97]. In a two-dimensional representation, each dimension will have an assigned orientation, horizontal or vertical, and order the items according to that axes. Each item is subdivided in two-dimension and represent the next dimensions, the process repeat until all dimensions are represented.

2.2.4.2 Worlds with Worlds

Worlds with Worlds divides the n-dimensional space into three-dimensional subspaces [Kei97] generating an interactive representation that allow the exploration of n-dimensional subspaces where the most important parameters are represented in the innermost world.

2.2.4.3 Treemap

Treemap is a space-filling method [SW01] that represents data as a set of nested rectangles where each branch of the tree is a shape and each sub-branch is inside that shape and the node is represented by the area of occupied. The size and color are used to encode the features. Usually, this method is applied for pattern recognition over large hierarchical data sets.

2.2.5 Iconography

Multidimensional data is not only about mapping higher-dimension data to two or three dimensions but to understand the data behaviour. The iconography, icon-based or glyph methods each object is defined by a glyph where color, shape and location can be used to map the item features. Common to all multidimensional icons, the relation between the task and the glyph used as a significant impact in the effectiveness of the visualization.

2.2.5.1 Chernoff Faces

Chernoff faces [Che73] maps two features for the glyph position and the rest are mapped to facial features such as nose,eye size or face, mouth shape, eyebrows orientation. This visualization has several disadvantages: the different items are not easily compared [OL03], can only be used for limited amount of data items [KK96b],

2.2.5.2 Star Glyph

Star Plot maps data features to a stylized star. The features are represented as angular axes from the center and their length represent the value. They can be used to compare low and medium size data or used for performance metrics visualization. The start glyph can be expanded to encode more features by using each axe as a source of information, per example a box-plot.

2.2.5.3 Stick Figure

As in the Chernoff Faces technique, Stick Figure [PG88] maps two features to the axes and the remaining by changing the rotation, size, color or thickness of the figure. In relation to a display context, when dealing with high density creates patterns that can be easily detected. This technique does not have order for the mapping, making the process of mapping the features subjective and reduce the usefulness of this technique.

2.2.5.4 Shape Coding

Shape coding [Bed90] uses a matrix made of array of pixel. Each data item is represented by an array and each attribute is one pixel that is mapped with a color scale, black for missing values, blue for low values, red for mid values and green for high values. The array is generally placed in a rectangle arrangement. The major advantage of this method is the computational speed, making the time for the visualization generation low on the other hand the detail showed is lower in comparison to other methods.

2.2.5.5 Color Icons

Color Icons merges color, shape, and texture perception coding multiple parameters into a single integrate image [Lev91]. In comparison with the shape coding, the pixel are replace by array of colors that represents the different features.

2.2.6 Graph-based Techniques

Graph-based techniques target the visualization of high amounts of nodes and edges in a way that the information can be extracted clearly and quickly. The different methods can be divided in three categories [Kei97]: Basic Graph, which includes the well know two-dimensions graph drawing; Specific Graphs that focus in particular problems like cluster optimization and symmetry optimization; Graph Systems [AHK06] [GN00] that try to represent high complex interconnected data by representing data attributes in the nodes or edge figures by using color, shape, size, and text information.

2.2.7 Hybrid Techniques

Hybrid techniques combines multiple concepts from the previous categories trying to provide solutions for specific scientific areas or use cases to improve the interpretation in a intuitive way. Many examples of concepts and implementations and probably the category with more techniques published: FlinaPlots [CvW11]is a c ombination of parallel c oordinates arranged with scatterplots and histograms that allows rapid visual correlations but have the downside of being a complex technique; Angular histograms [GPS+11] also allows the detection of correlations by analysing the slopes and density mapping of the histogram; Event River [LYK+12] where events are presented in oval shapes that represent the duration of the event and intensity over a timeline; Whisper [CLS+12] a spatio-temporal visualization that shows the tweeter messages sentiments over a radial layout, with timestamp and location.

2.3 Interactions

Interactions are important as the presentation of information for understanding and analysis, they can be described has “the communication between user and the system” [DFAB03]. Some classifications for this topic appear during the last years. In 2014, Liu et al. [SL14] provided an update to the previous classification Yi et al [JSY07] by adding state-of-the art techniques and dividing them in two categories main categories: WIMP interactions and post-WIMP interactions.

2.3.1 WIMP interactions

This type of interactions techniques target the study of solutions for comparison, navigation and feature visualization in a window,mouse, pointer, keyboard paradigm. Examples of this kind of interactions include selection, that is used in combination to others techniques to enhance the navigation [JSY07]; filtering ; brushing and some more complex interactions such as visual comparison, interest-driven navigation. In the table a summary of interaction techniques based on main application.

Table 2.3: Interactions

Comparison / Relation	Navigation	Interactive feature specification
2-tone coloring [SMY+05] Juxtaposed Views [JKM01] [WS09] Difference Views [LKH10]	Search Zooming [CKB09] Panning [YaKS07] Focus-based navigation [DRRD12] Faceted-navigation [DRRD12] Interest-driven navigation [HD12] Undo-redo [JSY07]	Brushing [KLM+08] [KMG+06] Transfer function [WS09] [AFM06]

2.3.2 Post-WIMP interactions

In comparison to WIMP, this kind of techniques attempt to go beyond the paradigm of windows, icons, menus and pointing interfaces and try to develop new techniques in touch-enabled, gesture-enabled devices and other devices that don not use mouse and keyboard input. This techniques are interesting in the point that allow to expand the utility of the dashboard to devices like smartphones or tablets and to increase the proximity between the user and the dashboard and allow new use cases.

2.4 Current Solutions

At this time, multiple solutions for different problems are already deployed. Some tools are focused in collecting and analysing data from urban life from a high level of detail to a combination of indicators available in a web site for every citizen. Another branch tries to bring new methods of visualization and exploration and create a combined solution for multiple cities. For last, the framework focused more in business metrics and data that are know for be flexible for many context in the business domain. In this section is a overview of the current solutions followed by a comparison between them.

2.4.1 Urban Control Rooms

Nowadays many cities have already real-time data analytics to manage subsystems of the city by collecting data from sensor networks, transponders, cameras, and other sources [BAR15, SRM+16]. Computer Vision has specially played an important role in this task, in different ways and for different purposes [LRB09, LKR+16, NSR18]. The solutions are centralized in a hub allowing to monitor and adjust the traffic flow, police surveillance or automatic administer for penalties for traffic violations. One example for this type of system is Centro De Operações Prefeitura Do Rio in Rio de Janeiro, Brazil inaugurated in 2010 merges multiple agencies data in one centralized hub and collects municipal, services, traffic, weather forecast data and others [KMM16]. Then, the data is analysed by algorithms and analysts and it also stored and agglomerated to generate periodic insights and create prediction models. The system also permits data sharing between institutions making the coordination. The second example is Kashiwa-no-ha Smart Center in Kashiwa, Japan is a smaller scale system focused in the energy management of the city. The center monitors electricity usage from residences, commercial buildings and manage energy information during disasters. The evaluation of performance metrics of control rooms usually resorts to the concept of Artificial Transportation Systems and Simulation [RFBO08], whose information can also feed such dashboards.

2.4.2 City Dashboards

This type of solution focus on objective, rational and neutral data exposed to the public in a comprehensive way that every citizen can understand, evaluate and monitor multiple municipal services or policies making then an important tool not only for citizen but also for the decision-makers because it creates a neutral field of empirical observations, since most of then in recent times are managed by external sources to local governments like scientists and researchers. We can say that as data showcases showing data from both public and private services using live data streams from official sources, social-media and others allowing the user to get a realtime showcase of the city. Many examples of such solutions exist around the world, and the most recent one is the City of Sidney's Dashboard, also known as CityDash. Such dashboard combines multiple feeds such as weather conditions, public transport information, air quality, live traffic feed, twitter trends, and it also includes an interactive map that let users to visualize realtime transports information [PLJ17]. In the same line of research, other efforts have been reported in the literature which explore crowdsensing and social media as an important source of information for city dashboards [CSR10, RSR15, PPSR17, PPS+17, USRS16, KFC+15].

2.4.3 Frameworks/Platforms

2.4.3.1 Visual Analysis Approach for Exploring Spatio-Temporal Urban Data

Also know as VAUD is a visual analysis framework for exploration and comprehension of heterogeneous urban data [CHW+18] . To represent the data two types of structures where implemented:

the Object Based composed by identification tag, spatial information, temporal data, descriptive information and their relations with other objects and the Space-time cube that stores all spatio-temporal data. In relation to data like points of interest is stored in a database and indexed by their spatial attribute. For the data exploration implements a interactive drag-and-drop system that allows a intuitive query construction with direct effect on visualization . The system supports multiple types of analyses that include geo-spatial, social-network, temporal, statistical.

2.4.3.2 MEGA-WEB

MEGA-WEB [GSM17] is an open Internet-based GIS platform focused in various cities from Africa and Asia that born from the absence of a tool for local stakeholders for a sustainable urbanization and includes information of 42 major cities from this two continents. The platform focus on data sharing, visualizations and spatial analysis by providing a geodatabase of indicators that includes population density, energy use and environmental metrics. Also the platform allow three types of analysis: city analysis, that focus the selected city with the available information, the city comparison that allows to compare the same indicators between two cities and an overview mode that allow the comparison between multiple cities and their relations with multidimensional exploration.

2.4.4 Dashboard Framework

This category has a wide offers already available, the distinguish for supporting multiple dashboard that can contain data from multiple sources but are focused in the business domain. The more complex systems allow query, visualization and set alerts of metrics can be considered limited in terms of visualizing geospatial data.

One of the most know is Grafana, released in 2014 and at this timed used by numerous amount of companies. The main characteristics for this projects are open-source; allows visualization of metrics;possibility of aggregating data multiple sources; runs as a web application; can be deployed in a container and requires some advance to configure it.

Another interesting solution is the Cube.js that offers a complete solution in terms of visualization of metrics, but has the disadvantage of requiring programming for the setup of the visualizations.

2.4.5 Solutions Comparison

In this section a comparison of the different solutions by analysing different factors. Most of solutions in the urban control rooms and city dashboards integrate multiple city subsystems like transportation, social networks, weather and have visualizations for each type however the visualization is focused for specialist or the data is in low detail only allowing a very superficial analysis. One interesting solution with intuitive interface that allow more intuitive data exploration without many previous knowledge was the VAUD interface by using box and connections to build the queries. The solution that can be considered more complete and complex solution is the Rio de

Janeiro integrating dozens of subsystems an many deep and detail visualizations but is not open for the public use. The framework such as the Grafana and Cube.js offer a flexibility for each context that the VAUD, MEGA-WEB and city dashboards doesn't allow but lack support for urban data.

On the table 2.4 is presented an overview of the comparison between the solutions found.

Table 2.4: Comparison between different solutions

	Rio de Janeiro	Kashiwa	Sidney	London	VAUD	MEGA-WEB	Grafana	Cube.js
Integrates Multiple City Subsystems	+		+	+				
Spatio-Temporal Visualizations	+	+	+	+	+	+	+	
Other Multidimensional Visualizations	+	+						
Metrics Visualization			+				+	+
Real Time Data	+	+	+				+	+
Open to Public			+	+		+	+	+
Extensible by the public			+				+	+
Allows data exploration	+			+	+			
Allows micro view of data	+			+	+		+	+
Allows macro view of data	+		+	+	+		+	+

2.5 Summary

The chapter starts by reviewing urban indicators, the definition and the process from data to indicators to knowledge. Then, focus specifically in the sustainability indicators where a comparison study between multiple cities projects is presented, where the authors created a list of indicators from it that can be used has a base for a full city overview implementation in combination with the indicator quality criteria method presented right after.

After, a comprehensive review of data visualizations method starting for the most well-know and ending with more recent and many times more complex methods that try to solve problems using many techniques combinations and getting interesting results but require complex implementations to put in practice. Human-Computer interactions are important as the data visualization itself since connect the user and the machine, two categories can divide this techniques, WIMP interactions study the interactions by using a mouse and a keyboard however since is studied for many years many approaches appear to solve comparison problem, navigation or visualization of a specific feature. The second category, the Post-WIMP interaction go outside the windows, mouse, keyboard paradigm and try the use of different devices like sensor gloves to explore new ways of visualization and navigation, making it a quite interesting category for out-of-the-box approaches.

In the final section the current solutions is presented, five solutions are presented and compared, Rio de Janeiro solution is the most complete in representing subsystems of the city however lack integration with other user groups like researchers on the other hand VAUD present quite interesting and intuitive system for data exploration system with the drag-and-drop interaction to build the queries. From the comparison presented is possible to understand that currently there is a gap between the capability of representing urban data and the flexibility of business data frameworks. In the next chapters a solution is that tries to bring this flexibility to urban data visualization is showed.

Chapter 3

Methodological Approach

In this chapter a detailed characterization of the project, starting by detailing on Section 3.1, the context of the solution to be achieved. Followed by the system and client requirements on Section 3.2. Then, on Section 3.3 an explanation of all the process from the data extraction to the information visualization on the dashboard, discussing possible alternatives for multiple sub-problems the methods applied. Section 3.4 has a detailed description of each component of the architecture. Section 3.5 contains the method of service integration and Section 3.6 describes the method for the selection of visualization integrated.

By the end of the chapter, it should be possible for the reader to understand the objectives and general scheme of the project.

3.1 Problem Statement

The current solutions do not allow the user in a context of urban data adapt the a solution to met their requirements. In the topic of displaying data, one of current approaches requires experience in data analyses like in urban control rooms. The other show limited information that only allow a very specific use case or only allows a very superficial analyses. Also, many system do not allow flexibility in the visualizations as show in many city dashboards. In the opposite side, the frameworks for business data visualization that have the flexibility but does not have urban data support.

A solution for a tool flexible that merge all users from decision-makers, researchers to the regular citizen that can be extended as the needs required can add value to this kind of tool by on a long term, a working solution that can represent multiple subsystems in a smart city in a explained and intuitive for each type of user. The proposed solution is a framework with an modular application design to do not compromise further implementations of visualization or data type supports are added to the system. This work follows up and complements previous efforts to devise the proposed solution [RB99, ZRC14, SR15, PSR18, CMRR18].

3.2 System Requirements

After the identification of the current gap in the offered solutions, this section presents the requirements of a solution that fills that space. The requirements grouped into two groups architecture and client. For each one, a detailed list was formed with the identified requirements.

3.2.1 Architecture Requirements

- **Integrate new services without disrupting the others already available –** Many tools have been developed in other projects for specific subsystems. Also, some of then focused on the extraction. By having the ability to include such tools, makes easier to share data between projects and aggregate multiple information to retrieve new knowledge.

- **Support multiple data types –** Each domain has is own specific data that require data structures to be understandable and converted to be supported by multiple visualizations, that have it is on data requirements.

- **Ability to remove services without major disruptions –** A source of data or information can longer not be worth the resources available or no longer fit the requirements. The capability of being able to remove without affecting the system is important to give flexibility in dynamic problems.

- **Extensible to other platforms –** The services at an initial stage can be limited to one platform but, a full-scale implementation independent from the platform can offer access to data in different situations and support a completely new range of scenarios.

- **Extensible in the features provided to the users –** Not only the possibility to extend the data sources are important, the ability to give the user new features to fit their requirements can improve the effectiveness of the solution.

3.2.2 Client Requirements

- **Support multiple visualizations metaphors –** The user should be able to visualize a variety of data, each one requires a different visualization per example, geo-located data combine well with maps but from non geo-referenced can extract more information from a type of chart.

- **Multiple visualization technologies supported –** The current offer of data visualization techniques is disperse in a variety of technologies that have their on communities focused in specific problems. For that reason, the system need multiple technologies to be able to support geo-located data visualizations, animations, and plots.

- **Customize the dashboard layout –** A dashboard is composed by a combination of visualizations, if their location and size are fixed, the user can't focus on a certain data during the period of interest and the resize it back to the original position. Another case is when the user want to rearrange the visualization. This give flexibility to adapt to each user and their momentary or more long term requirements.

- **Change of the color scheme –** One of features widely require by multiple modern services and application, is the possibility to change the overall application look with one click. Normally, this features is composed at least, by a normal scheme and a dark mode.
- **Configuration of visualizations –** Each data demands a specific set of configurations and the user should be able to set up each one individually to maximize flexibility.
- **Load specific data from the services –** Not all data is required at all the time, maybe only requires a set of data from one source. The user should be able to select which data is loaded.
- **Add external data –** Not all data is integrated in the service and be available to the user, by that reason users should be able to add their own sources.
- **Add new visualizations to the dashboard –** A new visualization may be required to understand some aspect of some data and the user should be able to add it and position it in the dashboard layout.
- **Remove visualization from the dashboard –** A visualization in the dashboard may no longer be required and the user should be able to delete it.
- **Have multiple dashboards –** A user may have more than one dashboard depending on the use case or the focus on a particular topic. In order to avoid to reconfigure the same dashboard all the times, the user should be able to keep then stored.
- **Filter data presented on visualizations –** The user may want to focus on a specific aspect in one visualization for that need to filter some irrelevant parameters or objects from the visualization.
- **Get visualization recommendation for the data –** The get the most from data the visualization need to be appropriate, by getting a recommendation the user can see a filtered selection to choose from.
- **Rate a visualization –** The user can be a source of data if rates the visualization containing data from a determinate theme, allowing to help other user choose.

3.3 Proposed Solution

An important variable to have into account when conceiving the architecture is the ability of scaling. To analyze it, the three dimensions, represented in figure 3.1, that need to be approached:

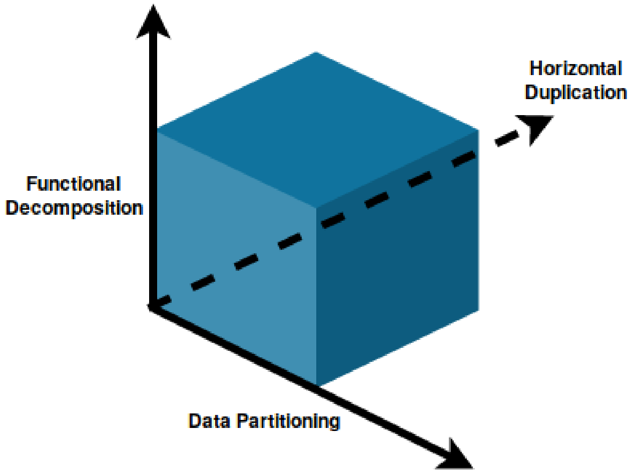


Figure 3.1: Dimensions of scaling adapted from [AF15]

- **Function decomposition**: split the task into smaller sub-tasks, this can be achieved by divide into multiple services where each one is responsible by one domain, use a verb-based decomposition and each service is responsible by one use case, or a combination of both methods mentioned.

- **Horizontal duplication**: can be described has run multiple copies of the same application behind a component that distributes the requests for the multiple copies. This dimension requires some special attention since multiple copies can possibility access and edit the same data.

- **Data partitioning**: similar to the horizontal scaling but each service copy is only responsible for part of the data, this requires some component to reroute the request for the service with the required data.

In recent times, multiple companies have managed their business functional complexity by slicing it into many components. The microservice architecture come as an alternative to the monolithic approach, dividing into multiple autonomous services that fulfill the requirements and giving scalability in a functional decomposition dimension. This approach has several advantages:

- Allow for a continuous delivery and deployment, specially important when the system start to grow in complexity.

- Each service is focused and not so complex to manage and understand.

- No technological commitment to any stack

- The service domain is confined to a specific context
- If a service fail not necessarily means that all system goes down

On other hand, this solution has some drawbacks to be considered:

- Since every service is now independent, there is a increase of complexity because of the distributed system and requires special attention to handle requests between then, if they exist.
- The deployment can be more complex because may require coordination between modules due to operational requirements.
- The resources required for the system increase since it replaces the a application instance for multiple services instances.

In relation to the horizontal scaling is not fully explored in this proposal but a possible simple solution is to run multiple copies of the same service hidden by load balancer to solve possible bottlenecks.

For last, the data partitioning is similar to the horizontal scaling but each service copy is only responsible for part of the data, this requires some component to reroute the request for the service with the required data.

The proposed methodology tries to find solution for the requirements of such framework listen on Section 3.2 that can be extended by adding support for new sub-systems or extend already integrated features

In relation to the integration of new services or removing, the solution relies on the microservice architecture and the use of some patterns that follows. The first pattern is a database per service, this not necessarily mean a provision for a database for each service, the persistence data is private to each service and if one service want some information it needs to ask for it instead of assessing it directly. The second one is the decomposition by sub-domain where each service is composed by closely related domain functions, this allows the development of services by small teams with no big help needed or coordination from other teams.

A visualization can have many shapes from 2D to 3D combined with animations, maps and objects. To covert this variety many specialized technologies are required to achieve a good user experience and cover a wide range of possible visualizations.

For the support of multiple data types they need to be defined both on the client and on the service level and compatibility between types should be described in a way that allow the conversion to match the specifications in both sides.

About the visualization, to support a wide variety they need to be contained in a component and an interface described for the implementation. The information provided to the visualizations are listed bellow.

- Width of the container visualization

- Height of the container visualization
- The data related
- Specific configurations

With this, it is possible to get it encapsulated into a modular structure, allowing for an easier integration and removal of visualization techniques easier. Another condition for the support for the visualization metaphors integration, three possibilities related with rendering where considered:

- **Client Side Rendering**: The data selected is transmitted to the client and then it renders the visualization. This approach has the disadvantage of requiring a better user machine and also, the data is not available as required, need to request it to the service and becomes inefficient with a significant amount of user view the same visualization with same data.
- **Mixed Rendering**: The service pre-renders a set of images for a fixed viewpoints and store then. As requested, the client receives multiple images and merge then, creating a composition. This method has the advances of letting merge data from multiple services but has the inconvenience of the increment in complexity of the implementation.
- **Server Side Rendering**: the visualization is rendered in the server and is streamed to the client, this have a advantage that the data is always available, the user machine requirements are lower but the required larger service infrastructure and for a large user base visualizing the same thing turns this approach efficient.

Taking it into account the focus on a user centered choice in terms of visualization, usage of data, and time restrictions, the selected method was the client side rendering. This option does not limit the support for other methods in a further development, but simplifies the implementation and the integration of range of visualization.

Respecting to the customization many components can be taken into account, from the configuration of a visualization, selection of datasets or even the layout of the dashboard, the proposed are presented on table 3.1. The planned visual customization only affect the appearance in the screen, not changing the data presented or the dashboard state. On other hand, the functional customization change the data visualized.

Table 3.1: Customization

Visual Customization	**Functional Customization**
Client Color Scheme	Filter Functions
Visualization Color Scheme	Dataset Selection
	Visualization Configuration
	Dashboard Layout Arrangement

Lastly, the current trends in visualization focus on cloud platforms, web technologies, and a considerable amount of creative content is created based on Javascript and WebGL [1], the major advantage is the possibility to use a variety of devices with the same implementation. Also, the current offer can support a broad variety of requirements in the data visualization domain.

3.4 Description of Architectural Modules

Taking into account the several requirements and objectives already presented, a general approach is proposed. The main objective is to achieve a prototype capable of representing the scenarios and flexible for the necessary adaptations not only at the problem itself but also at the user level. The conceptual architecture, in figure 3.2, is divided into the following modules.

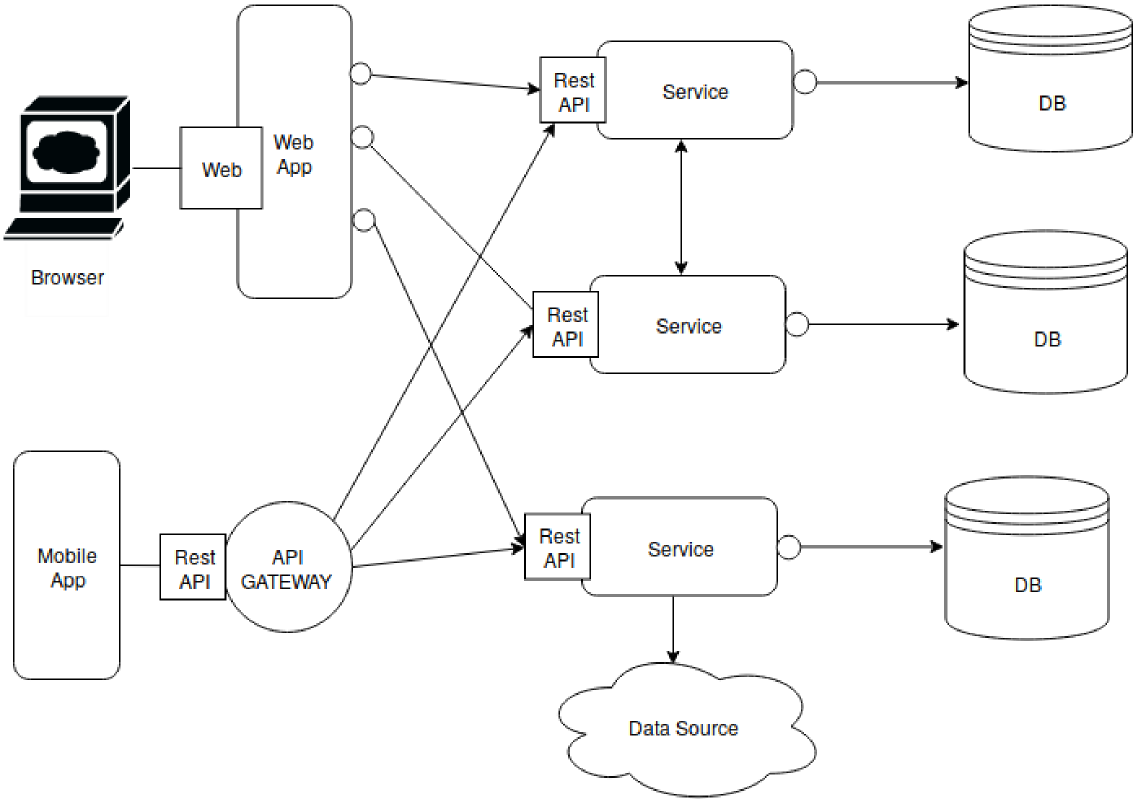


Figure 3.2: Conceptual Architecture

1. **Service**: The tasks addressed to the module are extraction, collection, processing of data and provide access to information. Each service should be limited to one data source, especially if they are collecting information from external sources, to avoid disruption in the collection from other sources. This concern is also valid in a more management aspect if determinate data is no longer worth collecting or by limitation of resources.

[1] https://get.webgl.org/

Each server can possibly communicate with each other there are many scenarios that can be useful: a service that aggregates data or processed data from multiple services and even with external sources to generate new information and transmit it to a client. Another point, important to refer is the possibility that the service is not directly implemented in the system and be treated as another source of information for the client.

2. **Client**: A user interface and client-side logic that allows the user to select a set of visualizations and data that he desires to view, can be a web, mobile application or even a desktop application. Some more simple features can include visualization auto-arrange, export data, snapshots of visualizations or even the export the client configuration with a share option.

3. **API Gateway**: Different user and clients will fetch a diversified data selection and commonly the interface provided by services does not match their requirements also the division of function in the system should be hidden from the client. The solution is to concentrate the access to a single entry point for all clients, one advantage is allowing the implementation of a security check to know if the client can perform the request but also the services instances and their location can change over time and thus allow the client to communicate all the time.

4. **Databases**: Permanent storage for collected information or data. Ideally, the way to go is a database per service and only allowing the access to the data throw their API, with this the different services can adapt to their storages requirements, can be deployed independently and allow the scaling by data partitioning.

3.5 Service Integration Methodology

For the integration of a certain service, the data flow represented in figure 3.3 is critical, particularly in the relation between the request to the service and the information aggregated. How the information transmitted is structured, the output format must be known for each request type and if there is a new data type be described to the client and other services that require information. Furthermore, another important point to be described is the compatibility with other formats for the automation of conversions and simplify possible aggregations functions.

In relation to performance, there are some requirements, the time from the request to the answer should be short enough to don't affect the user experience and have large time with loading screen or spinners. Also, the amount of information transmitted has to be carefully thought to send large amounts can cause the client to crash and also making the time for visualization loading huge.

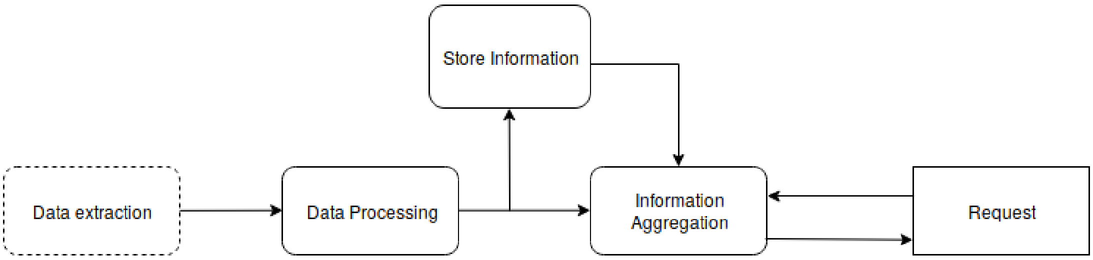


Figure 3.3: Service basic data flow

Regarding which services and by what order were integrated, the process was iterative, based on the data available, compatibility with visualizations and the ones how had to impact for the collection of information.

3.6 Visualization Integration Methodology

Essentially, the data flow also plays an important role, a scheme for this flow is presented in figure 3.4. The data has two possible paths, does not need conversion and is not any configured transformation the information flows directly for the visualization need to pass throw a series of pipes and functions after flow to the visualization.

In relation to performance, there can be some issues. The time for the application of the conversions and filters have a heavy role in the user experience. One possible approach to reduce the time in the exploration phase is the memorization of steps through the transformation functions.

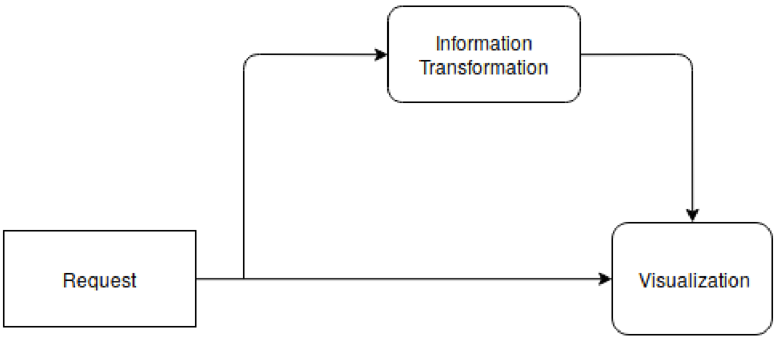


Figure 3.4: Client information flow

Regarding which services and by what order were integrated, was again an interactive process based specifically on the information possibly compatible with the visualization and which offer better flexibility for future services.

3.7 Summary

The objective of this chapter is to describe the methodology to build a system capable of supporting the requirements of a variety of domains and also customizable at a user level. The previously presented methodology currently looks versatile and encouraging. The workflow is created in a way that allows the components to operate independently, this is important especially with service no longer makes sense of being available and the cost of removing it is lower. Also, the proposed methodology focuses primarily on functional dimensional scalability by using microservice architecture and tries to solve some subproblems by using some design patterns, such as database per service.

Chapter 4

Implementation

This chapter describes the implementation of the project. Starts by explaining how each service integrated, the technological choices and how problems that occurred were approached. Also, presents the client by detailing each component and functionality. At the end of is section, the reader should be able to understand how the system was implemented and how it works together.

4.1 Current Solution

The current implementation presented in figure 4.1 with the technological stack is composed of several services and one client. This section is divided into two parts. The first describes the services and the second one focus on explaining the client and its main features.

All the services and the client were developed and tested using containers. More specifically Docker[1] with this technology is possible to build services and applications with all their dependencies and settings files allowing to run it in a much easier way in a variety of environments.

4.1.1 Service Modules

At this time, the system is composed of four services. They are divided into three categories: information providers, information gathering and knowledge gathering. The first focus on giving access to data and allows the client to interrogate for data that match certain conditions. Second, registers the interactions between the user and a client, to understand these relations. Finally, this target in collection knowledge from the users and process it to improve future users decisions. The details about each one are presented below in this subsection.

[1] https://www.docker.com

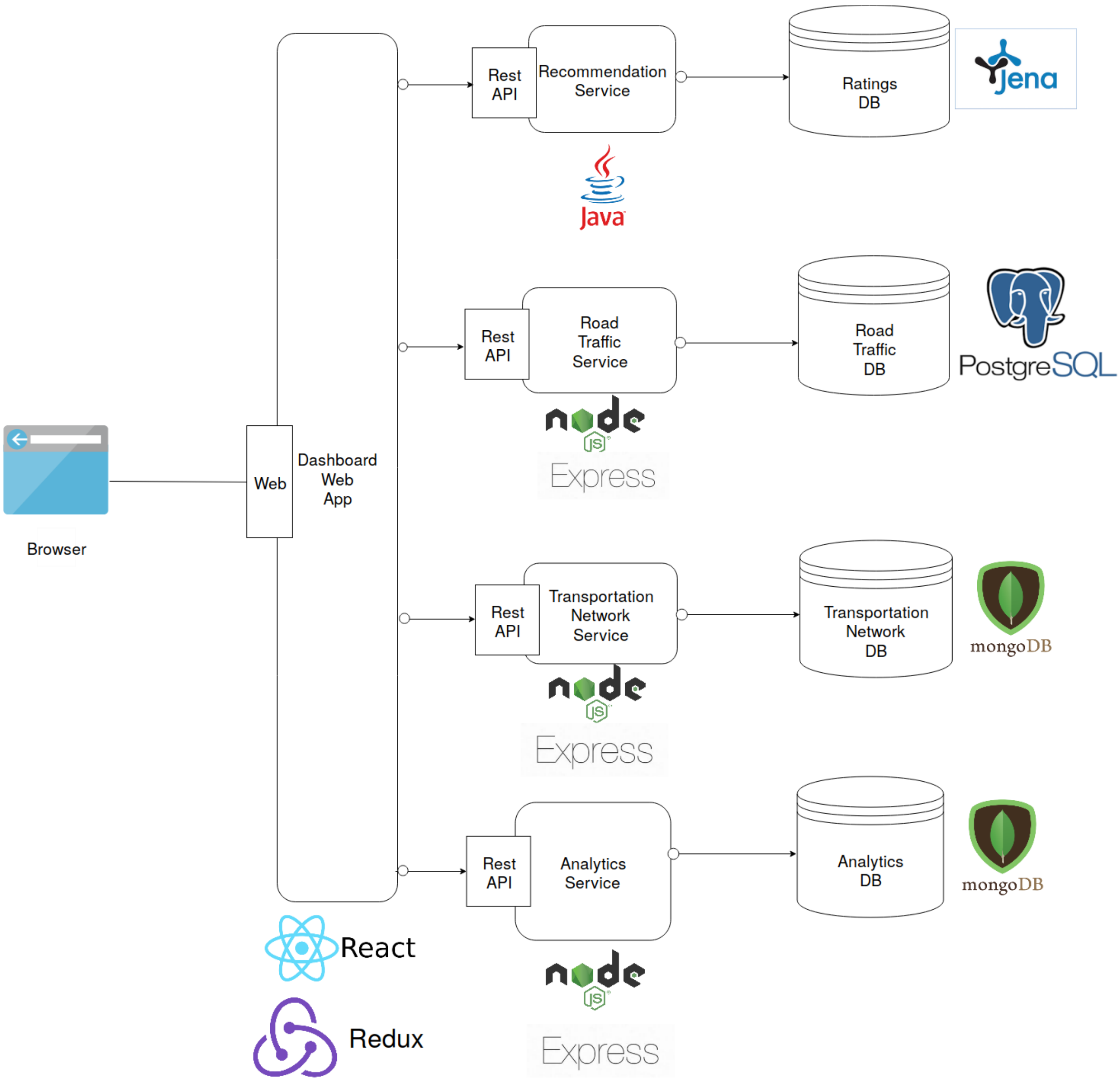


Figure 4.1: Current System Implementation

4.1.1.1 Road Traffic Service

This service provides information about road traffic data previously collected such has sensor position, traffic flow during a specific time. The data initially was imported to a relational database, PostgreSQL [2]. The data inserted is was the following structure.

- Location coordinates:
  - Location Name
  - Latitude and Longitude

[2] https://www.postgresql.org/

- Flux between zones:
  - Initial Zone
  - Destination Zone
  - Flux Volume
- Sensors information
  - Equipment Identification Number
  - Record Starting Time
  - Length of the Record
  - Lane Direction
  - Total Volume
  - Average Vehicle Speed
  - Lanes Occupancy

This data is accessible to the client throw a REST API implemented using Node.js [3] and Express [4] that provides the routes for each type of data and query to match a certain condition.

In summary, the service falls in the category of information provider containing traffic data from a highway collected during a determined period and allows the data filtering during a period of time and transforms it into the supported format by the visualization.

4.1.1.2 Transportation Network Service

At current state, the service contains the topology of the different transportation networks with information relative to each node allowing to search for specific characteristics in the network. This data is accessible throw a REST API implemented using Node.js and Express that provides a unique route that has as parameter the name or id of the network.

Every network is composed of nodes and edges. Each node is formed by geolocation and descriptors. Edges are composed by starting node, ending node, and descriptors. The descriptors can be anything of interest that only has to follow a JSON format.

In summary, the service is an information provider containing data multiple networks with from a city to country size, that provides direct access to a full network at one request.

4.1.1.3 Analytics Service

Focused on recording anonymously, the user interactions with the system allowing to understand which features are more used, by what order and find where the system can be improved. Also, speed up and increase the accuracy of the data collected during the tests phase.

When a user executes an action in the client a record transmitted to the service, an example of data presented below.

[3] https://nodejs.org/en/
[4] https://expressjs.com/

```
{
    interaction: "ADDED_VISUALIZION",
    visualizaton_type: "BAR_CHART",
    timestamp: "2019-06-18T23:00:00.000Z"
}
```

Listing 4.1: Example of recorded interaction

With this is possible to understand per example which visualization the users focus more, how many visualizations is the ideal, how the dashboard is arranged by the user and also comprehend which features can be removed.

In summary, the services belong to the information gathering category collecting multiples interactions from multiple clients and the main reason for him to exist is necessary to understand which features are essential and how the user interacts with the system.

4.1.1.4 Recommendations Service

For the implementation the ontology used was the Visualization-oriented Urban Mobility Ontology [SGB16] or VUMO more specifically the Visualization and the Domain User concepts, allowing a visualization system the annotation of visualization techniques and the feedback from the user about this visualizations. In relation to the visualization techniques, the ontology specifies multiple components for a detailed description from the input variables, spatial dimension, temporal arrangement and representation and the type of data required as detailed by [AMST11]. Also, the DomainUser define the user feedback characteristics such as user profile, a recommended topic for the visualization, visually appealing or effectiveness.

A rating has presented on Listing 4.2 is composed of the related visualization and the rating components. The aggregation is done by calculating the average of each visualization in the different rating components returning to the client a list with the results.

```
:Rating1 a  :TechniqueRating ;
    :isAboutVisualizationTechnique  :barchart;
    :hasRatingStatement [
        :hasRatingComponent          :RecommendedTheme;
        :hasRatingCategoricalValue  :TravelIntention
        ].
```

Listing 4.2: Rating example for the barchart. visualization

On submission of a new rating, using system validates the request and tries to infer new relations with the rules of [SGB16] and stores along with the new rating, the discovered relations.

In summary, the system focused on collect knowledge from the user base to help the decision of what is the best fit visualization for the data related to a determined topic and store it over an RDF graph.

4.1.2 Client Modules

The client presented on detail in figure 4.2, is a web application that supports multiple visualization technologies 2D and 3D and let the user customize some aspects such as windows arrangements, which visualization with what data, visualization dimensions, and overall color schema.

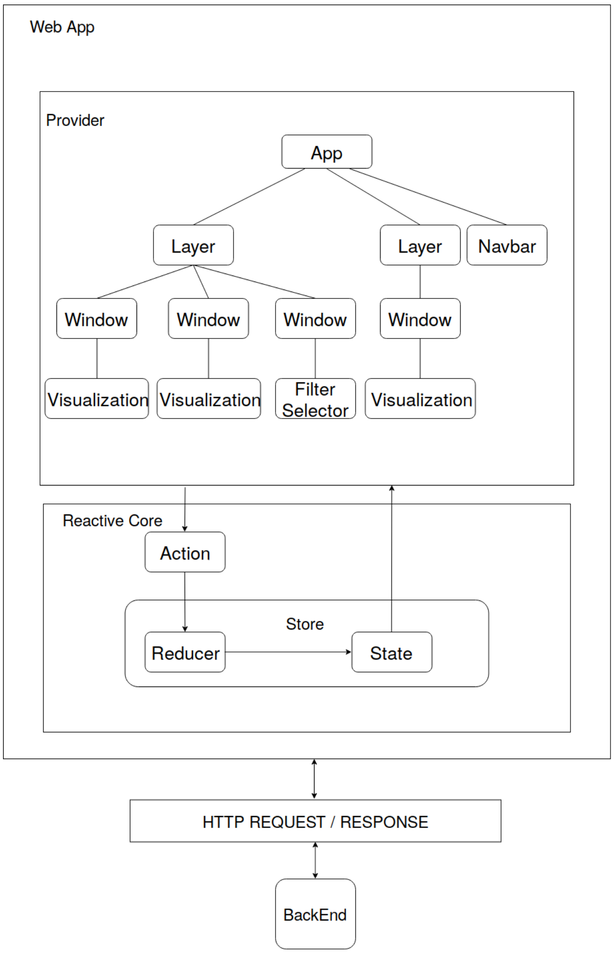


Figure 4.2: Client Architecture

4.1.2.1 Client Components

In a more architectural aspect, the client is composed of multiple layers to enabling the user to have and keep multiple dashboards. Each layer can have any number of windows and each window can have one or none visualization or an interactive tool like the filter selector, the overall result of this combination can be seen in Figure 4.3. The function of each component is described below.

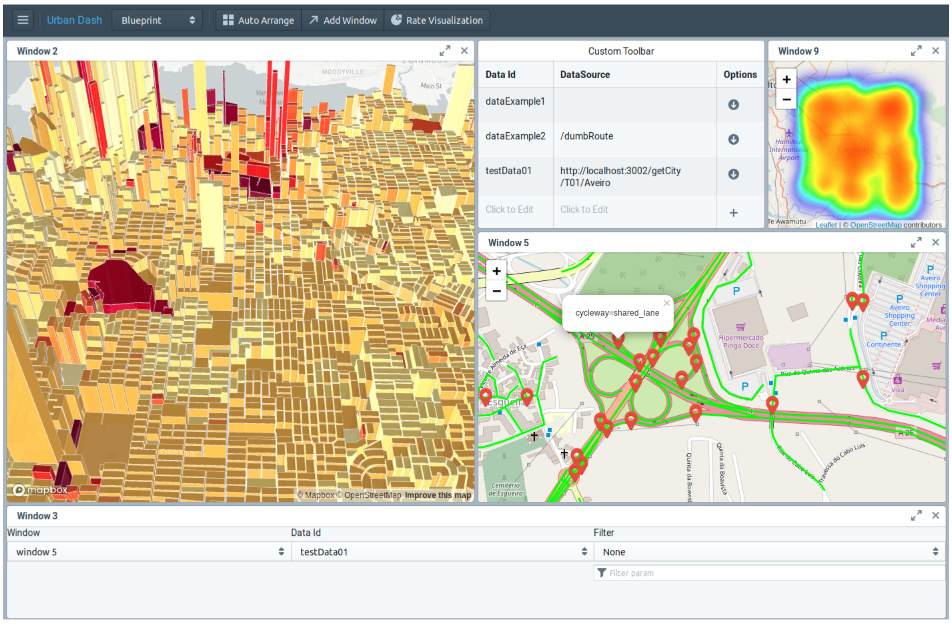


Figure 4.3: Client Screenshot

Layer – Responsible for the organization and arrangement of a set of windows. Does not have a visual representation but is the core of the whole visualization system because allows the support for multi-dashboard and space awareness to represent visually the different windows.

```
{
  currentNode: {
    direction: 'row',
    first: 2,
    second: {
      direction: 'column',
      first: 1,
      second: 3,
    },
    splitPercentage: 40,
  },
  name: "Dashboard 1",
  windowCount:3,
},
```

Figure 4.4: Default Layer State

The layer state in figure 4.4 represents the distribution of windows using a binary tree where a node is a window, represented by their identification number, or with two children. The space that each window take is represented by the splitPercentage that is the total space taken in the specified direction.

Window: Container for the visualization or a utility tool with a quick toolbar on the top, allowing resize of a visualization in the dashboard space or deletion of itself. This component does not have a state, his purpose is space awareness for the component to be represented.

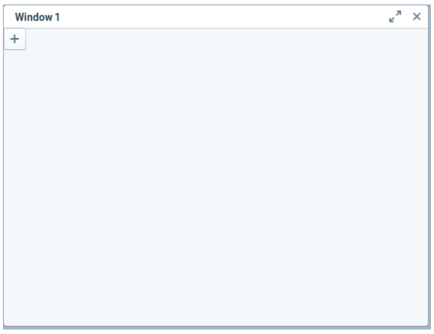


Figure 4.5: Empty window

Visualization: A graphics and interactive component, that receives the space from the window and renders the content accordingly. Currently, the supported visualizations are charts and maps, to achieve these multiple technologies where used react-vis [5] for the charts. In relation to the maps, leaflet[6], and mapbox[7] and for the 3D visualizations deck.gl[8].

This component has the most complex interface besides the Reactive Core due to the variety of types of configuration parameters required to obtain a visualization. One remark for the configurations of the filter, that is stored was to allow the sharing of raw data between visualizations.

```
export interface VisualizationConfig {
    type: Visualization_Types,
    dataId: string,
    nodeId: number,
    xAxis?: xAxis,
    yAxis?: boolean,
    bars?: BarConfig[],
    lines?: LineConfig[],
    tooltip?: boolean,
    legend?:boolean,
    cartesianGrid: {
        active: boolean,
        strokeDasharray?: string,
    },
    tileLayer?: boolean
    filters?: Filter[],
}
```

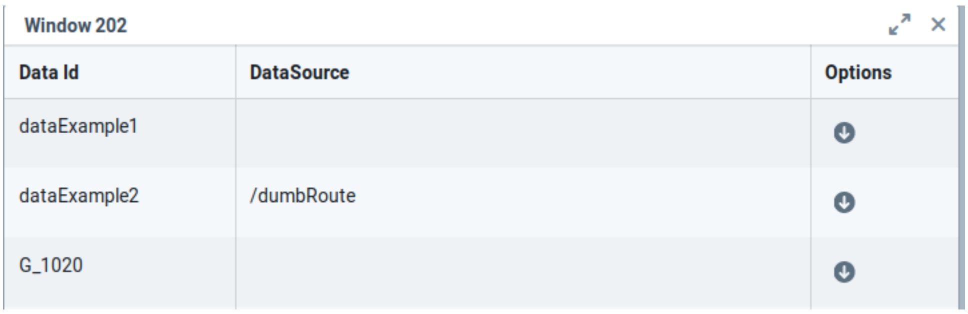
Figure 4.6: Interface for Visualization

[5] https://uber.github.io/react-vis/
[6] https://leafletjs.com/
[7] https://www.mapbox.com/
[8] https://deck.gl/

Navigation bar: A top navigation bar with easy access to buttons and tools. In figure 4.7 from left to right, the three line button gives access to the dashboard menu that allows changing dashboard; the selector with theme options that change the overall look of the dashboard; the four square buttons with the text "Auto-Arrange" automatically organizes the dashboard for the user; the arrow-top-right button with the description "Add window" inserts a node in the deeper level of the tree representation of the layer; the send-to button with the text "Rate Visualization" opens the panel to submit ratings.

Figure 4.7: Navigation Bar

DataSources Panel: Let the user add new sources, after adding the data is available for all visualization. The components are a list divided into three column's, has the reader can see in figure 4.8. The first is a human-readable identification label so the user can identify the data to posteriors use it in a visualization, the second is related to the origin of the data, it is local no information appears otherwise the remote locations are displayed. Finally, the column of options that as functionalities associated, such as download the data, upload data.


Window 202

Data Id	DataSource	Options
dataExample1		
dataExample2	/dumbRoute	
G_1020		

Figure 4.8: DataSources example

Filter Selector: Allows the user to apply transformations to the data interactively allow to set a sequence of filters and pipes. To use it, need to select the window where the visualization is allocated and then can apply a sequence of the filter by selecting fill a parameter and repeat. The component also allows changing the data for the picked visualization. These three previously referred selectors can be analyzed in figure 4.9.

Figure 4.9: Filter Component

4.1.2.2 Client functionalities

Dashboards selection –The client state is composed of a given number of layers, each layer represents a dashboard. To select a dashboard the user press the top left corner menu and the management of the layers panel appears, letting the user choose which dashboard want to see. On selection, the default visualized dashboard is changed to the selected one, without refreshing the page. This menu only features the dashboard selection but can incorporate other future functions.

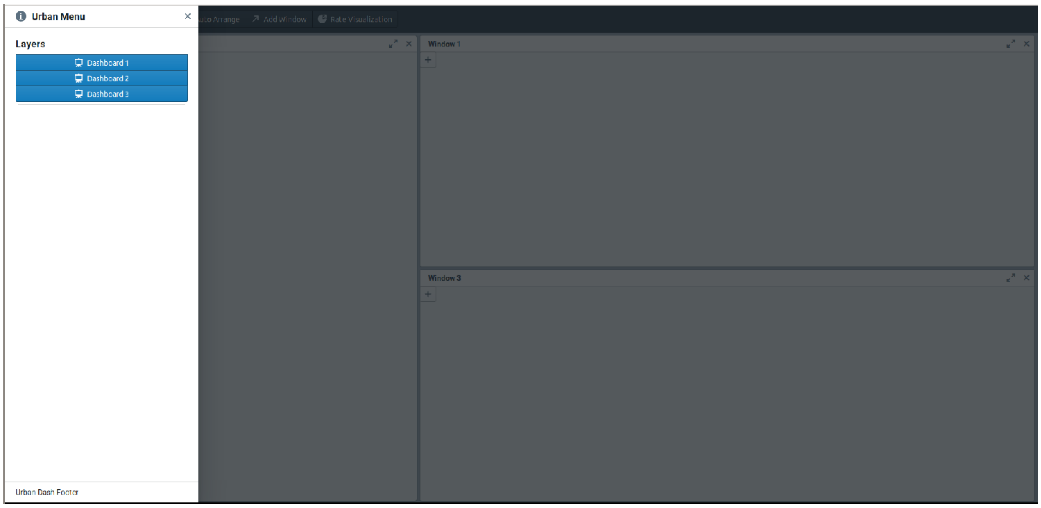
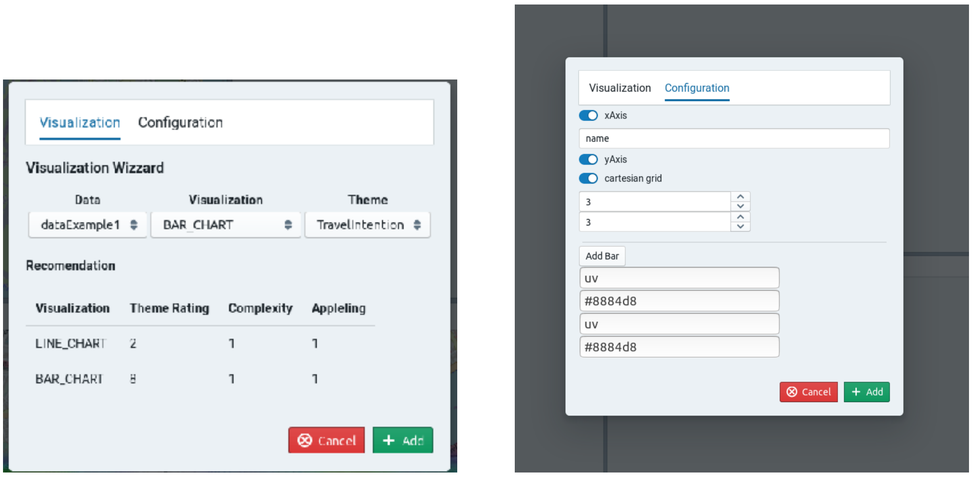


Figure 4.10: Dashboard Menu

Add Visualization – This functionality is divided into two different operations. First, the selection of the visualization, where a user can choose data and visualization. Second, the configuration of the visualization that a specific display for each visualization. The presented in figure 4.11 is an example for adding a bar chart where the user can choose display the axis, place a grid his density and select which data to show and their respective colors.

Recommendation– Composed of two parts, the recommendation, and the visualization rating. The first presented in figure 4.12 is incorporated on the add visualization feature. When the user selects a determinate data and a theme, the client sends a sample data and theme to the service that if have any record the system displays a list with the rating components available.

The rating visualization available at the top of the navbar and is a panel where the user can submit their review to a determinate visualization where can rate any component implemented in



(a) Visualization Selection (b) Visualization Configuration

Figure 4.11: Add Visualization

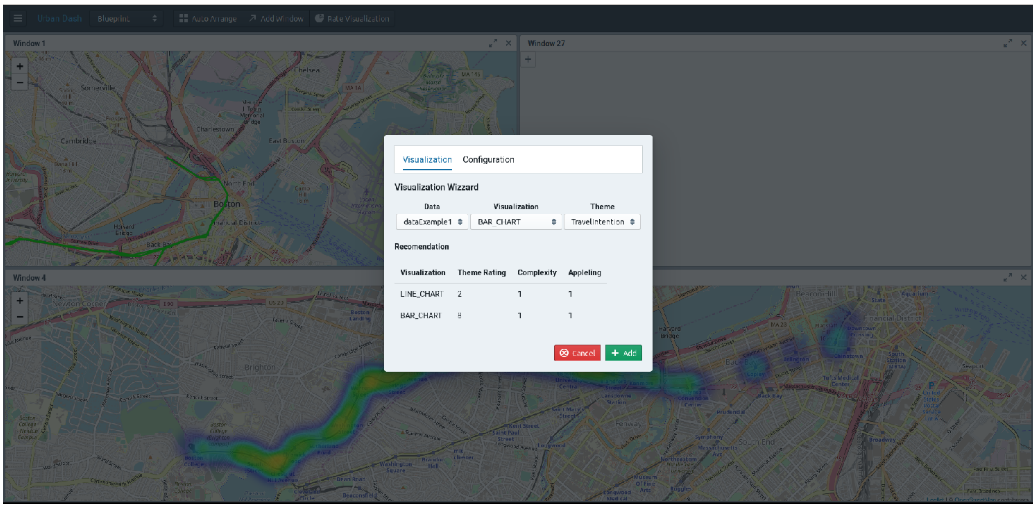


Figure 4.12: Recommendations

the recommendation service. The current state only allows submitting one rating at each time. An example of rating is presented on figure 4.13.

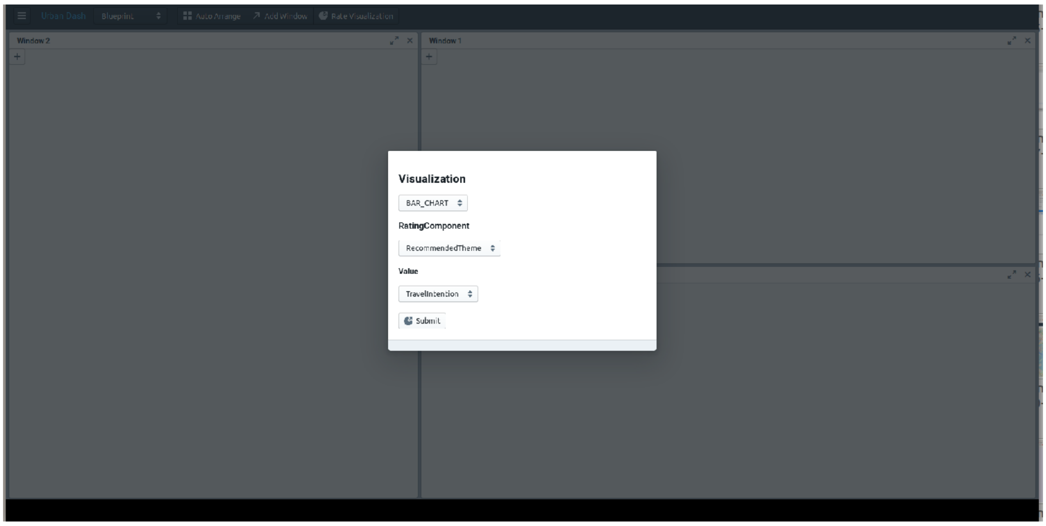


Figure 4.13: Rating Visualization

Toolbar functions– Quick access functions present in each window, all off them affect the layout of the visualizations.

- **Remove Visualization –** As the name suggests, remove a visualization from the dashboard. This is done by one button pressing in the window toolbar, present in every visualization.
- **Visualization Expansion –** Increases the size of the visualizations and readjust all the windows to fit.
- **Custom Toolbar –** Allows the creation specific content for each visualization providing quick access for the relevant tools. A comparison between the two modes is presented in figure 4.14

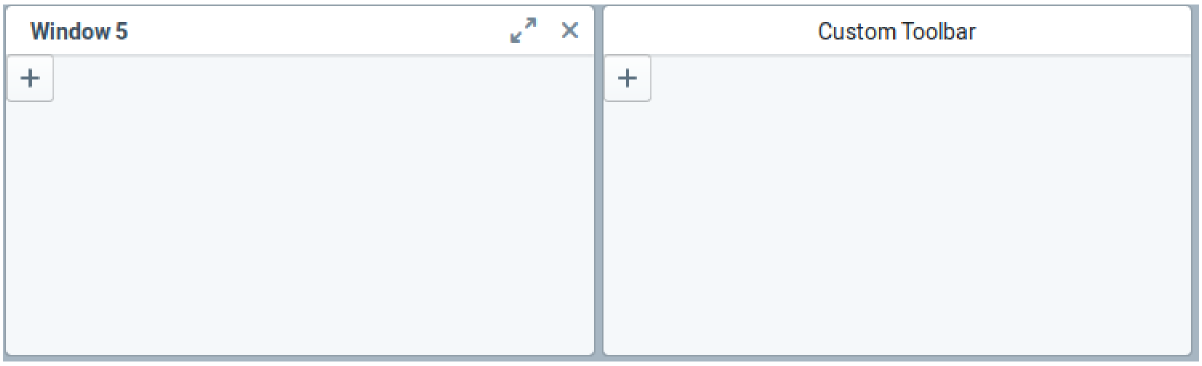


Figure 4.14: Normal toolbar vs Custom toolbar

Auto-Arrange Layout – Sometimes it is hard to arrange the windows by ourselves, this function tries to rearrange the size and positions of the windows by increasing the size of a possibly more relevant visualization and rearrange the rest accordingly.

Color Schemes – Also, know as themes, they define the hole colors of the dashboard, this includes the visualization in some conditions to maximize the possible customization. They are available in the top corner, near the layer menu.

Drag and Drop – Let the user grab a window and drop it in another location. This function is mainly used for the rearrangement of a layer layout. The user grabs a window and drops it on the desired location a preview of the final arrangement is visible by a blue transparent shape representing the new position.

Add new datasource – Not all data comes from the internal services some data can just be upload by a user or from a remote service, this lets the user add data as required.

4.2 Summary

The framework takes advantages of technologies and techniques that already exist in a wide variety of software already deployed to provide the user with a solution capable of being complete as a visualization tool. At this time, the implementation is composed of four services and one client supporting a variety of data types and visualizations. At this point becomes important to comprehend how this tool works with the user and if the proposed objectives were achieved. For this, the next chapter describes the method to assess the concept and presents the results and constraints from the system.

Chapter 5

Results and Discussion

The chapter Study, Result and Discussion details how the current solution fulfills the characteristics previously proposed. Start by explaining the test method step by step and the scenarios used. After, a presentation of the results obtained followed by an explanation and evaluation of what has found taking into account the previous literature and the proposed research questions.

5.1 Case Studies

To experiment the concept two tests were conducted with a given number of persons evaluating two major components: understand the usability of core features, capabilities and value for the user and the application of such tool in a given problem. The section starts by detailing the macro steps taken in both cases and after presents both study cases and their results.

5.1.1 Study Protocol

To get relevant data is important to define a specific sequence of actions to reduce the probability of data get invalid or not usable. To solve this a study protocol was developed for the user study cases to collect data within a controlled environment. For this, every individual test needs to be performed with the same equipment, working as expected and well-defined actions. The actions are taken with every test subject were the following.

Explain the study – Starts by describing the concept of the tool and what were the main objectives with the test to the user. This includes the overall dissertation objective and their contribution to the overall work.

Initialization – Each test started in the same initial condition, so each user had the same initial client state and a script of tasks where provided.

Tasks execution – Each user tried to execute the sequence of tasks presented on the script. The tasks covert all major functionalities available at the time of the study and is adapted for the context presented.

Free time – After the execution of the task the user was free to use the tool as they wish.

Fill the form – The form is divided into three parts, the first was based on guidelines for the user interface from Nielsen [NM90] to understand the perception of the user when interacting with the client. The second focused on the features provided by the system and how the user felt by using it, it matches the requirements or it was something still missing and for last, a series of question to comprehend how the user sees the framework in the context. All the forms use the same scale 1 to 5 to rate the parameters in the study and some open section when the user can write suggestions.

Thanks for the participant – After completing the form, we thanks the participant for the cooperation.

5.1.2 User Case 1 - Description

In this case, the study was conducted with 54 participants. The context is transport networks where the user has access to a bunch of data stored in the transportation network service that can explore to complete the tasks. The data provided to the user composed by nodes and edges where the user was free to choose from the data available, presented on table 5.1.

Table 5.1: Data provided in Test Case 1

City	Number of Nodes	Number of Edges
Aveiro	3294	3135
Braga	20063	20371
Coimbra	21262	21729
Ermesinde	2949	3134
Fafe	2383	2504
Gondomar	7340	7504
Lisboa	74622	91587
Maia	8505	8692
Porto	10176	10765
Viseu	6193	6175

Each user has the same given set of tasks to complete. The first is to add visualizations to the dashboard and rearrange it has desired, then has to insert the data from the ones provided and add the data to the visualizations. Finally, the user is asked to explore the data by using filter functions and visualization navigation. The list of task is presented in table 5.2.

Table 5.2: Task Sequence - User Study 1

Task Number	Task Description
1	Add a Map Visualization
2	Add a Filter Selector
3	Add a DataSource Panel
4	Rearrange the Windows
5	Add one of the dataset provided to the client
6	Add the data to the Map Visualization
7	Configure a filter
8	Explore the result

After this sequence was executed the user was invited experimented other available visualizations to better understand the capabilities of the system and test with not used data previously. Also, other features that the users did not try during the experiment such as different filters or the color schemas.

5.1.3 User Case 1 - Results

The results collected using A.1 for this case are divided into 3 sections, framework usability, functionalities implementation, and application of the tool in the context, each one composed by several questions focused in a particular aspect.

5.1.3.1 Framework Usability

For this section, the participant was asked about the overall feeling of framework divided into 5 parameters, allowing to get a macro view of the system, the results are presented in figure A.1.

In relation to simplicity, 41 of 54 the participants considered the system relatively simple or easy to use but the rest considered the complexity has average, not having no grades below 3, this can be related with the participant evaluation in some of the functionalities that got lower ratings. For the visibility of the system status, the participant had a good reception on how the client delivered feedback, from the visual direct feedback from the actions and from the pop-up system used to notify about asynchronous functions, with a 46 of 56 above rating 3 but there is some room for improvement. Regarding the user control above 90% of the user rated above average, this can be related to the level of customization provided from the visualization, the color scheme of client and layout is quite complete. The user freedom has a lower result and the user control and this can be explained by the fact that not all customization were fully implemented at the time of this test limiting the user choices during the test. Finally, the overall design that highest rate of 5 with more than half of the sample. The result is an indicator that the accessibility for every tool composing the system is well placed and the met the expectations of the user in relation to such a framework.

5.1.3.2 Functionalities

In the second part focus on the core features available asking in each question how the participant rate each feature, the results are available in figure A.2.

First, the ability to rearrange the dashboard as desired the users felt comfortable using it, the overall reception to the current state is positive with 90% of the participants rating it over 3. In some cases detected some abnormal behavior in special conditions such as dragging the window outside the browser space and drag in over the navigation bar, that let the client to not update the layout, that reflects the lower rate of 5. Next comes the adding visualization, that many users complained to be complex and required many steps but evaluated it positively over half the population rated with the max score. For the remove visualization, many users wanted only to remove the visualization, not the window this contributed to some frustration to the user and reflect in the evaluation. In relation to adding new data and using filters, the users were receptive with the concept. In case of adding data, the current offer did not fulfill all the user requirements, such as load local data as drag the file over the browser. For the filters, the participants asked for ways to implement new ones outside the implementing it directly in the client code, something like injection functions into the system. Finally, in the change dataset functionality, the users felt that works well but wanted the users to want it to be placed in the window toolbar as a visualization editor panel.

5.1.3.3 Applicability of the tool in this Context

In the final section, the results obtained can be analyzed in figure A.3. For the question relative to the contribution of this for the context, 40 of 54 rated 5, in other words, over 70% of the sample considered it better than solutions they have used previously. In relation to the information provided and visualizations available were to understand with the requirements for the problem in terms of offered solutions were there and to understand what as missing, overall half of the user considered that the requirements were completely full-filled for the problem. In the open section, many users suggested edition on the color of the visualization, some interactive features with maps such as calculating the path between two points taking into account a set of provided parameters and visualization of a solution and possible paths over a network.

5.1.4 User Case 2 - Description

In this second case, the study was conducted with 25 participants and the data collected using the questionnaire in B.2. The main goal was to evaluate the tool in a transport management context, where the user has access to a public transport record of a period of time around 20 days composed by schedules, transports delays and geo-location of vehicles. Some bugs detected during the first use case were fixed previously to this use case that includes some abnormal behavior when dragging in the windows. some heavy datasets let some visualizations crash and some styling errors. Each user as the same given set of task to complete, similar to the one in the first case. The first is to add visualizations to the dashboard and positioning has the user felt the best, then the user selected the data that wants to visualize. Finally, the user tries the filters to dynamically filter the data. The list of the task is presented in Table 5.3.

Table 5.3: Task Sequence - User Study 2

Task Number	**Task Description**
1	Add a Map Visualization
2	Add a Heap Map
3	Add a Filter Selector
4	Add a DataSource Panel
5	Rearrange the Windows
6	Add one of the dataset provided to the client
7	Add the data to the Map Visualization
8	Configure a filter
9	Explore the result

After the sequence, the user was invited experimented other available visualizations to better understand the capabilities of the system and test with not used data previously and some filters.

5.1.5 User Case 2 - Result

The result was collected using A.2 divided into three sections, framework usability, functionalities implementation, and application of the framework in the context, each one constituted for several questions about different aspects of the system and the interactions with the user.

5.1.5.1 Framework Usability

In this first part of the questionnaire the objective is to understand the overall relationship between the user and framework, the results obtained can be seen at A.4. The distribution for all categories are quite similar and improved in relation to the first use case. In relation to the simplicity of the system, 11 of 25 considered the simplicity at very good and all the participants considered to be average or better their previous experiences. For the visibility of the system status, user control, and user freedom, the result is similar to the first test and can be justified by nothing then the system changes to improve these parameters.

5.1.5.2 Functionalities

For the second part, the user was asked about the core features and it's possible to compare it with the previous test to see if the changes delivered improvements, the results can be seen at A.5. The biggest difference is related to the customization in relation to the dashboard layout, with the corrections implemented, 18 of 25 participants rated as a very good experience. Some improvements in system response time had a positive effect on removing visualization where the system has to rearrange and render the new layout. There no other significant results to report in this category.

5.1.5.3 Applicability of the tool in this Context

This is the final section of the questionnaire as in User Case 1 focus in understand if the framework works in the context and if the requirements for that context are full filled, the result can be seen in A.6. More than 80% of the users rated the tool above average for the problem, this is a good indicator of the current functionalities in the system. In relation, for the information provided and visualization, the participant considered above average but many requested a way to merge visualization.

5.1.6 Approach Limitations

The current results are limited to specific problems and do not include other uses cases not allowing to assess the effectiveness in a more broad scope or in aggregation with other domains, further user cases are required. Besides that, allowed to get an initial perception of the receptiveness of such framework, what features should be improved or integrated into the next interactions and perception of which are the requirements for a framework. Another restriction to have into account has the amount of data available for each problem and their diversity in sources and structure letting sometimes the users lose some time understanding the data.

5.2 Summary

In this chapter was demonstrated that the framework met the requirements proposed since can be considered somehow complete in terms of data supported, visualization integrated and customization. The user cases study allowed to understand what was the problems in the framework and its current limitations, especially when dealing with heavy amounts of data.

The offer of visualizations can cover a variety of data and was satisfactory for most of the participants, in comparison, the data available felt sometimes short in details. In relation to the level of customization that added complexity to the system, especially for adding visualization, the user felt comfortable on having such power of choice but wanted some automation or suggestion that could require left effort executing the task. Also, the study demonstrated that the level of details taken into the client has a heavy impact on how the user views the solution and how comfortable use it in the presented problems.

Overall, the result for the study is positive demonstrating that is capable of offering from data to visualization a quick solution to the user examine the data. Some of the problems found are described as a proposal for future work in Section 6.3.

Chapter 6

Conclusions

This last chapter is the result of the all process, a combination of all that as learned with it and is divided in three parts. First, the main difficulties during the process, then the major contributions and conclusion. For last, a list of possible paths for future work on this project, to improve is contribution as a tool and in a more scientific way, or the start point for others project with other focus.

6.1 Main Contributions

The main contribution as referred to in Chapter 3, a solution for the visualization of data from the source to the point is presented to the user, having the following characteristics.

- **Integration of new services without disrupting the others already available –** other words, the ability to add new sources and new functionalities that required backend processing and storage without necessarily meaning that have to disrupt the normal function of a deployed service.

- **Ability to remove services without major disturbance –** similar to the first but, in the opposite direction. Removing a service without causing abnormal behaviors in the services and clients.

- **Support multiple data structures –** multiple sources have diverse ways to structure the data but the visualization required specific data format. For this reason, the data need to be converted to specific structures.

- **Support multiple visualizations metaphors –** the diversity of data cannot be visualized all in the same way, for this a diversity of visualizations need to be available to cover a range of contexts.

- **Customizable at the user level –** the requirement change user by user and even if two user visualizes the same data, their perception depends on how the data is visualized, from the arrangement of visualizations, overall color scheme, or even how which filters are applied to the data.

- **Multiple visualization technologies supported –** at this time, there is a complex variety of visualization implemented in a variety of technologies by integrating such tools the framework can easily be extended to fit new requirements.

Also, a scientific paper resulted from this work, which was submitted and is currently under review for possible presentation and publication at the fifth IEEE Annual International Smart Cities Conference (ISC2 2019)[1].

6.2 Final Remarks

This framework focused on urban data based on a microservice architecture capable of representing a complex system, integrate previous tools and support a range of visualizations. The proposal seems to versatile and promising, the concept allows each component to work independently, this is especially important because each user will have different requirements. Also, can support a project focused on the information extraction simplifying the data visualization and sharing of the findings or data.

The results obtained are encouraging, the user satisfaction with the state of the framework is positive with the requirements of the studied cases accomplished with the current offer of data structures and visualizations. It is important to refer there some room for improvements and explore some new feature, that is described as future work in Section 6.3.

Another conclusion drawn is the integration of data sources require a big effort, especially in the data processing and at each data source integrated increases the complexity of a service or the overall system. If adding a new domain possibly requires expansion of current data structures or implementation of new ones and support for it.

We believe the contribution of this framework is clear, at this time there is no current solution that merges the solutions focused in urban data such as the control rooms presented in Section 2.4.1 with the city dashboards presented 2.4.2 and also combined the dashboards framework that allows custom dashboards 2.4.3 that can combine a large community with different focused in one framework.

One of the biggest challenges was how to integrate such variety of data and processes that many time is the result of other projects or require completely different technologies to handle it properly, adding a level of complexity and limitations of the proposed solution. Inevitably, this brings a problem of compatibility between services and clients that require multiple data structures to be defined and know by the parts involved in a communication.

[1] https://ieee-isc2.org/

Another adversity was related to the recommendation system, the current data had to be produced and limited the test of the current implementation and tests it with the user. Also, the technologies from this project did not fulfill the expectations being limited in documentation and performance is to low to be used to be available in the client.

In addition, an expected setback associated with the adaptability required to the system adapts to each user added multiple problems and increased complexity by requiring a combination of solutions to achieve a decent result.

6.3 Further Developments

Keeping in mind, that was the initial approach for the conception of a framework for such a complex system, several features require further deeper tests, investigation and may be considered interesting for research projects. Some of them are discussed as follows.

Automated multi-render visualization method system – In short words, adaptation to the data, visualization, and user machine. The main concept is a system capable to decide which method is better from the client-side, server-side or a mixed one for the best experience. If a dataset is large and may take several minutes to download and the client still has to process it. This affects the user experience and productivity, to reduce this waiting time the visualization is rendered in the server-side and transmitted to a client but in the opposite way if the dataset is small and the visualization has low hardware requirements may be rendered in the client and make available resources for other clients.

Automation of visualization configuration – Many users found the configuration of a visualization tedious, especially in more complex ones. The automation would be by filling the parameters for the visualization configurations decreasing the time the user spends, this probably requires to analyze the dataset for attributes, relations, or something of interest to determinate which combination of colors, shapes to use to represent fit the user objectives. This idea could be extended for the whole dashboard, by giving an objective and the data sources generate a number of visualizations and organize it for the user.

Improvement and creation of dashboards and visualizations interactions – The essay that took place only analyzed some interactions and already know interaction could be implemented and tested to specific contexts. Also, some new interactions were an option to experiment in this framework, making it a platform for research.

Extend the implementation for other platforms – The current implementation is restricted to a browser, however, extending it to tablets or mobile phones could be an attractive option to have data on request with easy access for quick decisions or even give a lower entrance for many users for data exploration.

Extend the implementation to other city domains – There are still many sources to be explored and aggregated with the ones already in the system. By contributing for a more complete solution, deeper studies in city behaviors could take place and tests, for the advantages and disadvantages of visualizations in multiple domains could be taken farther.

Improve and explore the recommendation system – One major issue is what visualization fits the data and the objective of the user. The VUMO ontology allowed to build an initial recommendation system but still many improvements and methods explored to deliver a better user experience.

The actual approach only shows the user a rating based on his objective and number of evaluation by other users and but can be improved by combining with the automatic visualization configuration to generate a visualization.

Extend the visualization offer and explore new visualizations metaphors – the visualizations available sometimes felt sort for users, also there is space to experiment new visualizations, especially in scenarios where no visualization meets the user requirement.

Shareable Dashboards – A quick way to share the layout or visualizations, the state of the dashboard is a relatively short JSON object, this is also true for the visualizations. There are many possibilities in this, one is to simply download and send the JSON file but other more interesting is a short link, that can be shared in one messaging app and then when the other opens it, the dashboard opens with all the settings set up.

Research the dashboard composition layout – Usually in information visualization focuses on the development of new visualizations, in this case, is the exploration of combinations visualization to find what is the best arrangement for a given problem. Per example, which is the ideal number of visualizations, by what order, which amount of space they should occupy, all of this in a more future work contribute to an automatically generated dashboard.

References

[AF15] Martin L. Abbott and Michael T. Fisher. *The Art of Scalability: Scalable Web Architecture, Processes, and Organizations for the Modern Enterprise*. Addison-Wesley Professional, 2nd edition, 2015.

[AFM06] Hiroshi Akiba, Nathaniel Fout, and Kwan-Liu Ma. Simultaneous classification of time-varying volume data based on the time histogram. In *EuroVis*, volume 6, pages 1–8, 2006.

[AHK06] James Abello, Frank Ham, and Neeraj Krishnan. Ask-graphview: A large scale graph visualization system. ieee trans vis comput graph. *IEEE transactions on visualization and computer graphics*, 12:669–76, 09 2006.

[AMST11] Wolfgang Aigner, Silvia Miksch, Heidrun Schumann, and Christian Tominski. *Visualization of Time-Oriented Data*. Springer Publishing Company, Incorporated, 1st edition, 2011.

[BAR15] J. Barros, M. Araujo, and R. J. F. Rossetti. Short-term real-time traffic prediction methods: A survey. In *2015 International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*, pages 132–139, June 2015.

[BC87] Richard A Becker and William S Cleveland. Brushing Scatterplots. *Technometrics*, 29(2):127–142, 1987.

[Bed90] Jeff Beddow. Shape coding of multidimensional data on a microcomputer display. In *Proceedings of the 1st Conference on Visualization '90*, VIS '90, pages 238–246, Los Alamitos, CA, USA, 1990. IEEE Computer Society Press.

[Che73] H. Chernoff. The use of faces to represent points in k-dimensional space graphically. *Journal of the American Statistical Association*, 68:361–368, 1973.

[CHW+18] Wei Chen, Zhaosong Huang, Feiran Wu, Minfeng Zhu, Huihua Guan, and Ross Maciejewski. VAUD: A Visual Analysis Approach for Exploring Spatio-Temporal Urban Data. *IEEE Transactions on Visualization and Computer Graphics*, 24(9):2636–2648, 2018.

[CKB09] Andy Cockburn, Amy Karlson, and Benjamin B Bederson. A review of overview+ detail, zooming, and focus+ context interfaces. *ACM Computing Surveys (CSUR)*, 41(1):2, 2009.

[CLS+12] N. Cao, Y. Lin, X. Sun, D. Lazer, S. Liu, and H. Qu. Whisper: Tracing the spatiotemporal process of information diffusion in real time. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2649–2658, Dec 2012.

[CMRR18] I. Caldas, J. Moreira, J. Rebelo, and R. J. F. Rossetti. Exploring visualization metaphors in macroscopic traffic simulation. In *2018 IEEE International Smart Cities Conference (ISC2)*, pages 1–6, Sept 2018.

[CSR10] Sara Carvalho, Luís Sarmento, and Rosaldo J. F. Rossetti. Real-time sensing of traffic information in twitter messages. In *4th Workshop on Artificial Transportation Systems and Simulation (ATSS), 13th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 1–4, Funchal, Portugal, Sept 2010.

[CvW11] J. H. T. Claessen and J. J. van Wijk. Flexible linked axes for multivariate data visualization. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2310–2316, Dec 2011.

[DFAB03] Alan Dix, Janet E. Finlay, Gregory D. Abowd, and Russell Beale. *Human-Computer Interaction (3rd Edition)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2003.

[DRRD12] M. Dörk, N. Henry Riche, G. Ramos, and S. Dumais. Pivotpaths: Strolling through faceted information spaces. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2709–2718, Dec 2012.

[FOU98] EUROPEAN FOUNDATION. Urban sustainability indicators, October 1998.

[FWR99] Ying-Huey Fua, Matthew O. Ward, and Elke A. Rundensteiner. Hierarchical parallel coordinates for exploration of large datasets. *IEEE Computer Society Press*, pages 43–50, 1999.

[Gal97] G.C. Gallopin. Indicators and their use: information for decision-making. *Moldan, B., Billharz, S. (Eds.), Sustainability Indicators*, pages 13–27, April 1997.

[GN00] Emden R. Gansner and Stephen C. North. An open graph visualization system and its applications to software engineering. *Softw., Pract. Exper.*, 30:1203–1233, 2000.

[GPS+11] Z. Geng, Z. Peng, R. S.Laramee, J. C. Roberts, and R. Walker. Angular histograms: Frequency-based visualizations for large, high dimensional data. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2572–2580, Dec 2011.

[GSM17] Hao Gong, Matamyo Simwanda, and Yuji Murayama. An Internet-Based GIS Platform Providing Data for Visualization and Spatial Analysis of Urbanization in Major Asian and African Cities. *ISPRS International Journal of Geo-Information*, 6(8):257, 2017.

[GvdK08] Arthur R. Manuel Geurt van de Kerk. A comprehensive index for a sustainable society: The ssi – the sustainable society index. *Journal of Ecological Economics*, pages 228–242, 2008.

[Hab04] UN Habitat. Urban indicators guidelines, August 2004.

[HD12] C. G. Healey and B. M. Dennis. Interest driven navigation in visualization. *IEEE Transactions on Visualization and Computer Graphics*, 18(10):1744–1756, Oct 2012.

[Hez04] A. Hezri. Sustainability indicator system and policy processes in malaysia: A framework for utilisation and learning. *Journal of Enviromental Management*, pages 357–371, 2004.

[Ins85] Alfred Inselberg. The plane with parallel coordinates. *The Visual Computer*, 1:69–91, 08 1985.

[JKM01] TJ Jankun-Kelly and Kwan-Liu Ma. Visualization exploration and encapsulation via a spreadsheet-like interface. *IEEE Transactions on Visualization and Computer Graphics*, 7(3):275–287, 2001.

[JSY07] John T. Stasko Julie A. Jacko Ji Soo Yi, Youn ah Kang. Toward a deeper understanding of the role of interaction in information visualization. *IEEE Transactions on Visualization and Computer Graphics*, pages 1224–1231, October 2007.

[Kan00] Eser Kandogan. Star coordinates: A multi-dimensional visualization technique with uniform treatment of dimensions. In *In Proceedings of the IEEE Information Visualization Symposium, Late Breaking Hot Topics*, pages 9–12, 2000.

[Kei96] Daniel Keim. Pixel-oriented visualization techniques for exploring very large data bases. *Journal of Computational and Graphical Statistics - J COMPUT GRAPH STAT*, 5:58–77, 03 1996.

[Kei97] Daniel A. Keim. Visual techniques for exploring databases. In *Intern. Conference on Knowledge Discovery in Databases (KDD'97), Newport Beach, CA, USA, November, 1997*, 1997.

[KFC+15] Zafeiris Kokkinogenis, João Filguieras, Sara Carvalho, Luís Sarmento, and Rosaldo J.F. Rossetti. Chapter 12 - mobility network evaluation in the user perspective: Real-time sensing of traffic information in twitter messages. In Rosaldo J. F. Rossetti and Ronghui Liu, editors, *Advances in Artificial Transportation Systems and Simulation*, pages 219 – 234. Academic Press, Boston, 2015.

[KHDH02] Daniel A Keim, Ming C Hao, Umesh Dayal, and Meichun Hsu. Pixel Bar Charts: A Visualization Technique for Very Large Multi-Attribute Data Sets. *Information Visualization*, 1(1):20–34, 2002.

[KK96a] D. A. Keim and H. . Kriegel. Visualization techniques for mining large databases: a comparison. *IEEE Transactions on Knowledge and Data Engineering*, 8(6):923–938, Dec 1996.

[KK96b] Daniel A. Keim and Hans-Peter Kriegel. Visualization techniques for mining large databases: A comparison. *IEEE Trans. on Knowl. and Data Eng.*, 8(6):923–938, December 1996.

[KLM+08] Johannes Kehrer, Florian Ladstädter, Philipp Muigg, Helmut Doleisch, Andrea Steiner, and Helwig Hauser. Hypothesis generation in climate research with interactive visual data exploration. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1579–1586, 2008.

[KMG+06] Zoltan Konyha, Kresimir Matkovic, Denis Gracanin, Mario Jelovic, and Helwig Hauser. Interactive visual analysis of families of function graphs. *IEEE transactions on Visualization and Computer Graphics*, 12(6):1373–1385, 2006.

[KMM16] Rob Kitchin, Sophia Maalsen, and Gavin McArdle. The praxis and politics of building urban dashboards. *Geoforum*, 77:93–101, 2016.

[Lev91] Haim Levkowitz. Color icons: Merging color and texture perception for integrated visualization of multiple parameters. In *Proceedings of the 2Nd Conference on Visualization '91*, VIS '91, pages 164–170, Los Alamitos, CA, USA, 1991. IEEE Computer Society Press.

[LKH10] Ove Daae Lampe, Johannes Kehrer, and Helwig Hauser. Visual analysis of multivariate movement data using interactive difference views. In *VMV*, volume 10, pages 315–322, 2010.

[LKR$^+$16] G. Lira, Z. Kokkinogenis, R. J. F. Rossetti, D. C. Moura, and T. Rúbio. A computer-vision approach to traffic analysis over intersections. In *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, pages 47–53, Nov 2016.

[LRB09] P. F. Q. Loureiro, R. J. F. Rossetti, and R. A. M. Braga. Video processing techniques for traffic information acquisition using uncontrolled video streams. In *2009 12th International IEEE Conference on Intelligent Transportation Systems*, pages 1–7, Oct 2009.

[LYK$^+$12] D. Luo, J. Yang, M. Krstajic, W. Ribarsky, and D. Keim. Eventriver: Visually exploring text collections with temporal references. *IEEE Transactions on Visualization and Computer Graphics*, 18(1):93–105, Jan 2012.

[LYS11] Mona N. Shah Xiaoling Zhang Li Yin Shen, J. Jorge Ochoa. The application of urban sustainability indicators - a comparison between various practices. *Habitat International*, pages 17–29, 2011.

[Nat07] United Nations. Indicators of sustainable development: Guidelines and methodologies, October 2007.

[NM90] Jakob Nielsen and Rolf Molich. Heuristic evaluation of user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '90, pages 249–256, New York, NY, USA, 1990. ACM.

[NSR18] J. Neto, D. Santos, and R. J. F. Rossetti. Computer-vision-based surveillance of intelligent transportation systems. In *2018 13th Iberian Conference on Information Systems and Technologies (CISTI)*, pages 1–5, June 2018.

[OL03] Maria Cristina Oliveira and H Levkowitz. From visual data exploration to visual data mining: A survey. *Visualization and Computer Graphics, IEEE Transactions on*, 9:378 – 394, 08 2003.

[PG88] Ronald Pickett and Georges Grinstein. Iconographic displays for visualizing multidimensional data. *Proc. IEEE Conf. Syst. Man Cybern.*, 1:514–519, September 1988.

[PH01] Georges Grinstein Patrick Hoffman. Visualizations for high dimensional data mining - table visualizations. *Information visualization in data mining and knowledge discovery*, 2001.

[PLJ17] Christopher Pettit, Scott N Lieske, and Murad Jamal. Citydash: Visualising a changing city using open data. In *International Conference on Computers in Urban Planning and Urban Management*, pages 337–353. Springer, 2017.

[PPS+17] J. Pereira, A. Pasquali, P. Saleiro, R. Rossetti, and N. Cacho. Characterizing geo-located tweets in brazilian megacities. In *2017 International Smart Cities Conference (ISC2)*, pages 1–6, Sept 2017.

[PPSR17] João Pereira, Arian Pasquali, Pedro Saleiro, and Rosaldo Rossetti. Transportation in social media: An automatic classifier for travel-related tweets. In Eugénio Oliveira, João Gama, Zita Vale, and Henrique Lopes Cardoso, editors, *Progress in Artificial Intelligence*, pages 355–366, Cham, 2017. Springer International Publishing.

[PSR18] S. Paiva, D. Santos, and R. J. F. Rossetti. A methodological approach for inferring urban indicators through computer vision. In *2018 IEEE International Smart Cities Conference (ISC2)*, pages 1–7, Sept 2018.

[RB99] Rosaldo J. F. Rossetti and Sergio Bampi. A software environment to integrate urban traffic simulation tasks. *Journal of Geographic Information and Decision Analysis*, 3(1):56–63, 1999.

[RFBO08] R. J. F. Rossetti, P. A. F. Ferreira, R. A. M. Braga, and E. C. Oliveira. Towards an artificial traffic control system. In *2008 11th International IEEE Conference on Intelligent Transportation Systems*, pages 14–19, Oct 2008.

[RK15] Gavin McArdle Rob Kitchin, Tracey P. Lauriault. Knowing and governing cities through urban indicators, city benchmarking and real-time dashboards. *Regional Studies, Regional Science , 2015, Vol. 2, No. 1*, pages 6–28, October 2015.

[RSR15] F. Rebelo, C. Soares, and R. J. F. Rossetti. Twitterjam: Identification of mobility patterns in urban centers based on tweets. In *2015 IEEE First International Smart Cities Conference (ISC2)*, pages 1–6, Oct 2015.

[Seg02] L. Segnestam. Indicators of environment and sustainable development. *The World Bank: Washington, DC, USA*, 2002.

[SGB16] T. Sobral, T. Galvão, and J. Borges. Vumo: Towards an ontology of urban mobility events for supporting semi-automatic visualization tools. In *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, pages 1700–1705, Nov 2016.

[SL14] Yingcai Wu Mengchen Liu Shixia Liu, Weiwei Cui. A survey on information visualization: recent advances and challenges. *Visual Computers*, October 2014.

[SMY+05] T. Saito, H. N. Miyamura, M. Yamamoto, H. Saito, Y. Hoshiya, and T. Kaseda. Two-tone pseudo coloring: compact visualization for one-dimensional data. In *IEEE Symposium on Information Visualization, 2005. INFOVIS 2005.*, pages 173–180, Oct 2005.

[SR15] A. Seliverstov and R. J. F. Rossetti. An ontological approach to spatio-temporal information modelling in transportation. In *2015 IEEE First International Smart Cities Conference (ISC2)*, pages 1–6, Oct 2015.

[SRM+16] M. Sandim, R. J. F. Rossetti, D. C. Moura, Z. Kokkinogenis, and T. R. P. M. Rúbio. Using gps-based avl data to calculate and predict traffic network performance metrics: A systematic review. In *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, pages 1692–1699, Nov 2016.

[SW01] Ben Shneiderman and Martin Wattenberg. Ordered treemap layouts. In *Proceedings of the IEEE Symposium on Information Visualization 2001 (INFOVIS'01)*, INFOVIS '01, pages 73–, Washington, DC, USA, 2001. IEEE Computer Society.

[TS98] Lisa Tweedie and Robert Spence. The prosection matrix: A tool to support the interactive exploration of statistical models and data. *Computational Statistics*, 13, 01 1998.

[TW14] Thomas Block Tarah Wright Francisco Benitez-Capistros Aviel Verbruggen Jean Hugé Tom Waas, Jean Hugé. Sustainability assessment and indicators: Tools in a decision-making strategy for sustainable development. *Sustainability (Switzerland)*, 6:5512–5534, 2014.

[USRS16] D. Ulloa, P. Saleiro, R. J. F. Rossetti, and E. R. Silva. Mining social media for open innovation in transportation systems. In *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, pages 169–174, Nov 2016.

[vWvL93] J. J. van Wijk and R. van Liere. Hyperslice. In *Proceedings Visualization '93*, pages 119–125, Oct 1993.

[WS09] Jonathan Woodring and Han-Wei Shen. Multiscale time activity data exploration via temporal clustering visualization spreadsheet. *IEEE Transactions on Visualization and Computer Graphics*, 15(1):123–137, 2009.

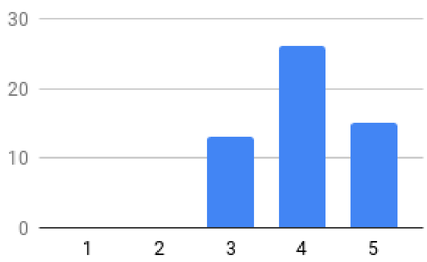
[YaKS07] J. S. Yi, Y. a. Kang, and J. Stasko. Toward a deeper understanding of the role of interaction in information visualization. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1224–1231, Nov 2007.

[ZRC14] A. Zaiat, R. J. F. Rossetti, and R. J. S. Coelho. Towards an integrated multimodal transportation dashboard. In *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 145–150, Oct 2014.

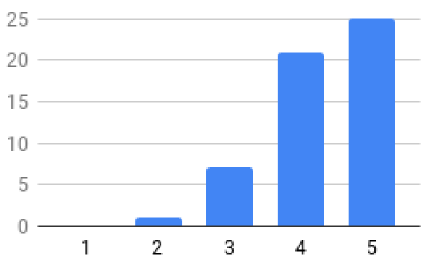
Appendix A

Results

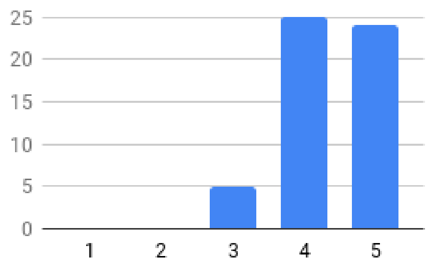
A.1 Use case 1



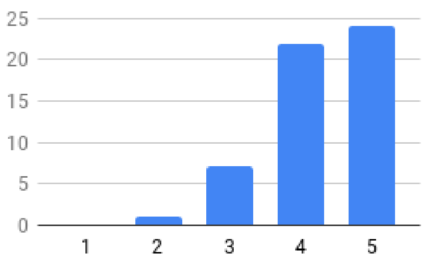
(a) Simplicity of the system



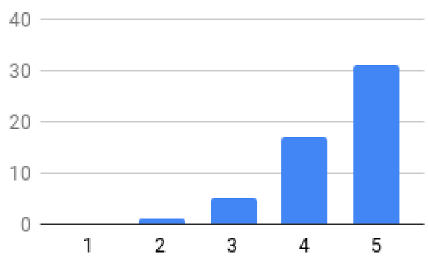
(b) Visibility of the system status



(c) User control

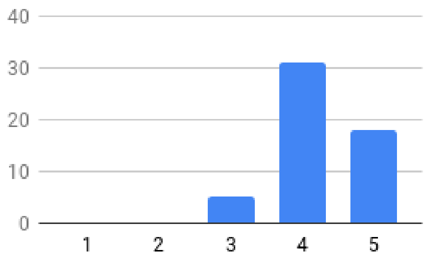


(d) User freedom

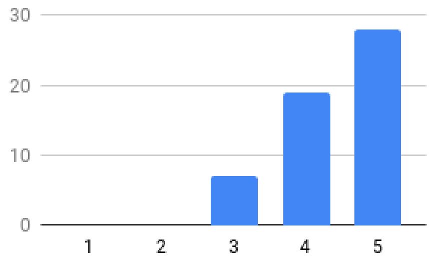


(e) Overall design

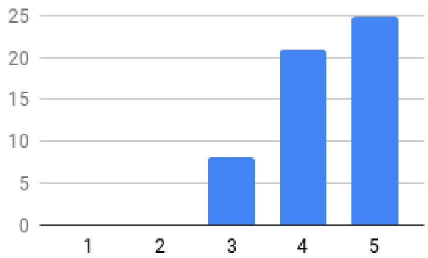
Figure A.1: Framework Usability - Use case 1



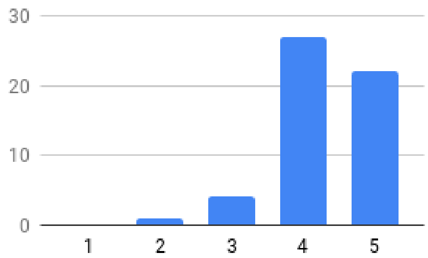
(a) Reorganize the visualizations



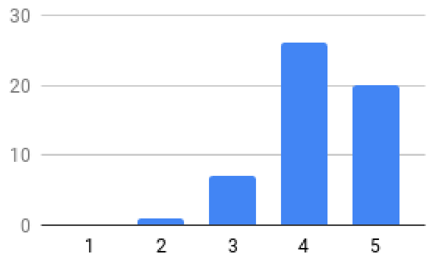
(b) Add visualization



(c) Remove visualization

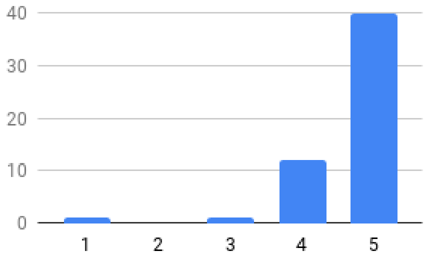


(d) Add new dataset

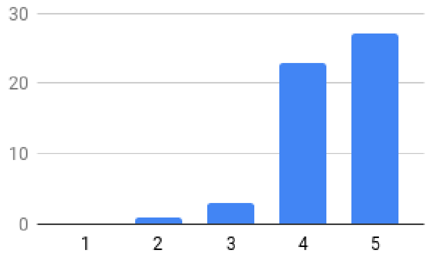


(e) Apply a filter function to a dataset

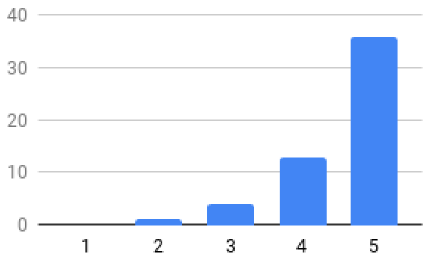
Figure A.2: Functionalities Results - User Case 1



(a) Value of information provided



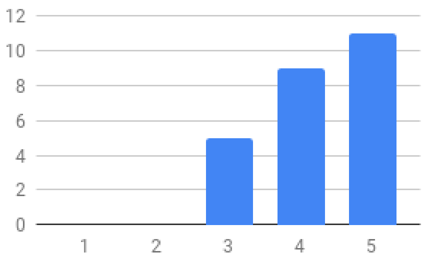
(b) Value of information provided



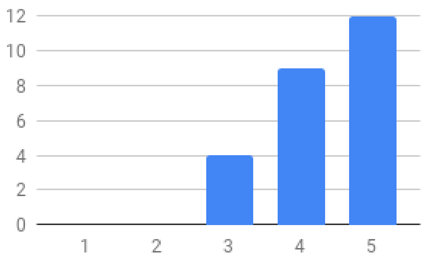
(c) Visualizations presented

Figure A.3: Framework Contribution - User Case 1

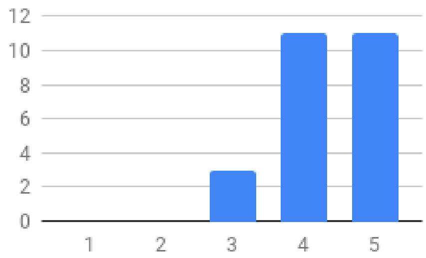
A.2 Use case 2



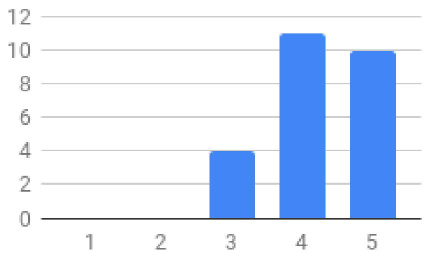
(a) Simplicity of the system



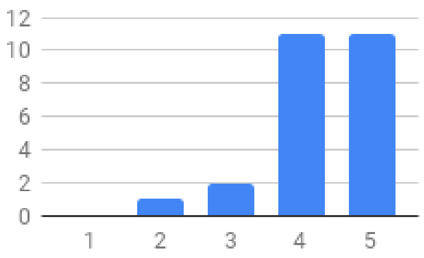
(b) Visibility of the system status



(c) User control

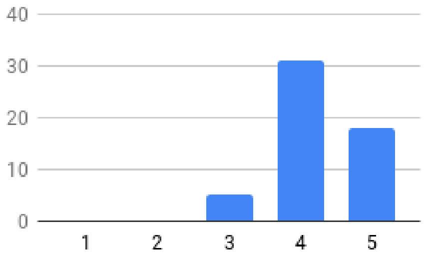


(d) User freedom

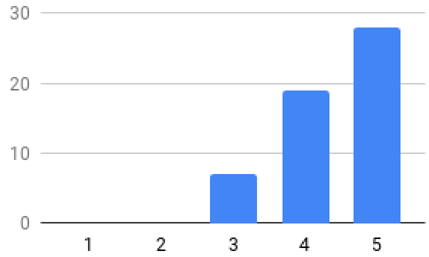


(e) Overall design

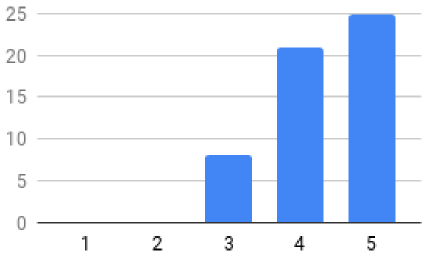
Figure A.4: Framework Usability - Use case 2



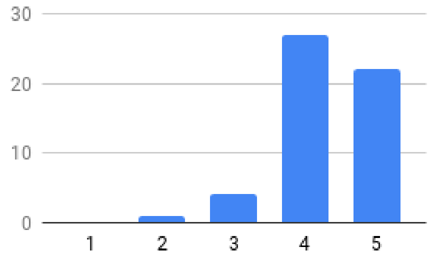
(a) Reorganize the visualizations



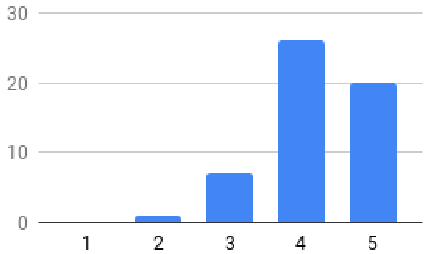
(b) Add visualization



(c) Remove visualization

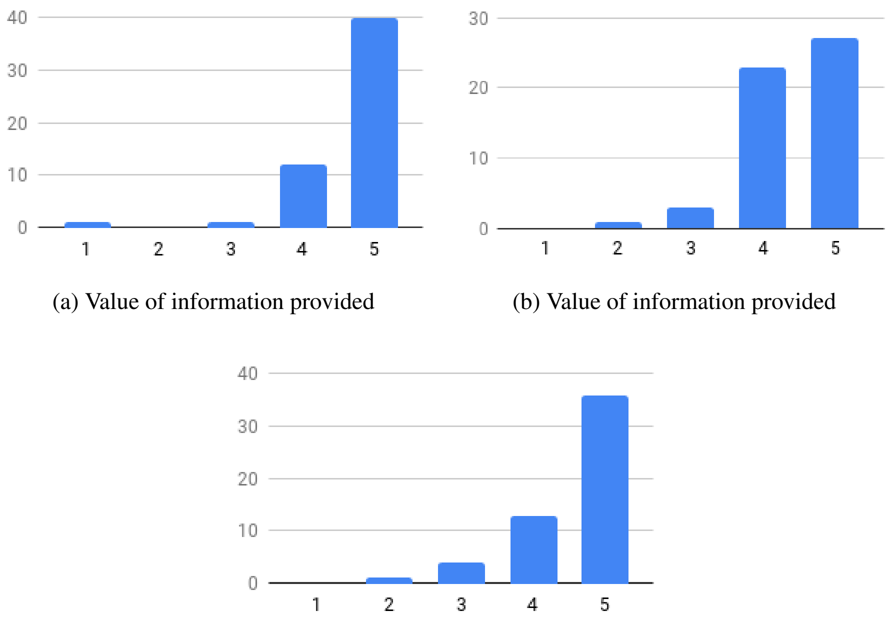


(d) Add new dataset



(e) Apply a filter function to a dataset

Figure A.5: Functionalities Results - User Case 2



(a) Value of information provided

(b) Value of information provided

(c) Visualizations presented

Figure A.6: Framework Contribution - User Case 2

Appendix B

Forms

The following documents are copies of the Google Form document that each participant answered in the respective tests has described in a stage of the study protocol with the objective to collect data for further analysis.

B.1 Use case 1

Dashboard Prototype

The UrbanDash is a generic tool that allows the integration of visualizations and reuse to fit multiple problem allowing the development of solutions for specific use cases.

*Obrigatório

Usability

1. **How do you rate .... ***

   Marcar apenas uma oval por linha.

	1 (Bad)	2	3 (Average)	4	5 (Very good)
the simplicity of the system?					
the visibility of the system status?					
user control ?					
user freedom?					

Tasks

Rate the task execution and overall design

2. **How do you rate .... ***

   Marcar apenas uma oval por linha.

	1 (Bad)	2	3 (Average)	4	5 (Very good)
the overall design?					
drag and drop ?					
add Visualization ?					
remove Visualization ?					
add new datasets?					
filter functions?					
change visualization dataset?					

3. **What part(s) of design should be improved? ***

4. **What feature(s) should be improved?** *

5. **Which feature(s) do you think that should be included?** *

Transportation Networks

Question specific related to the use case

6. **How do you rate ....** *

Marcar apenas uma oval por linha.

	1 (Bad)	2	3 (Average)	4	5 (Very good)
the contribution of such tool for CAL?					
the value of the information provided?					
the visualization(s) presented?					

7. **Which visualization(s) are adequate for this case?** *

8. **Which feature(s) do you think that should be included?** *

B.2 Use Case 2

Dashboard Prototype

The UrbanDash is a generic tool that allows the integration of visualizations and reuse to fit multiple problem allowing the development of solutions for specific use cases.

*Obrigatório

Usability

1. **How do you rate .... ***

   Marcar apenas uma oval por linha.

	1 (Bad)	2	3 (Average)	4	5 (Very good)
the simplicity of the system?	⬭	⬭	⬭	⬭	⬭
the visibility of the system status?	⬭	⬭	⬭	⬭	⬭
user control ?	⬭	⬭	⬭	⬭	⬭
user freedom?	⬭	⬭	⬭	⬭	⬭
the overall design?	⬭	⬭	⬭	⬭	⬭

Tasks

Rate the task execution and overall design

2. **How do you rate .... ***

   Marcar apenas uma oval por linha.

	1 (Bad)	2	3 (Average)	4	5 (Very good)
drag and drop ?	⬭	⬭	⬭	⬭	⬭
add Visualization ?	⬭	⬭	⬭	⬭	⬭
remove Visualization ?	⬭	⬭	⬭	⬭	⬭
add new datasets?	⬭	⬭	⬭	⬭	⬭
filter functions?	⬭	⬭	⬭	⬭	⬭
change visualization dataset?	⬭	⬭	⬭	⬭	⬭

3. **What part(s) of design should be improved? (N/A if you don't know) ***

4. **What feature(s) should be improved? (N/A if you don't know)** *

5. **Which feature(s) do you think that should be included?(N/A if you don't know)** *

Boston Case

Question specific related to the use case

6. **How do you rate ....** *
Marcar apenas uma oval por linha.

	1 (Bad)	2	3 (Average)	4	5 (Very good)
the contribution of such tool for transportation management?					
the value of the information provided?					
the visualization(s) presented?					

7. **Which visualization(s) are adequate for this case? (N/A if you don't know)** *

8. **Which feature(s) do you think that should be included? (N/A if you don't know)** *