Smart City Visualisation: Interactive Data Visualisation for New York Taxi Operational Data

Qiru Wang
689404@swansea.ac.uk
Department of Computer Science
Swansea University, United Kingdom

Abstract

70 per cent of the world's population will move to cities by 2050, this challenges the existing ways of governing city's resources. Smart City, namely the integration of information & communication technology into governances, is seen as one of the most effective way to achieve that. The data generated from information & communication technology carries valuable insights that can be utilised to optimise city's current operation, e.g. improve infrastructure efficiency, enrich healthcare quality or reduce traffic congestion. Data Visualisation conveys abstract data into meaningful graphic representations, which is an effective way that helps in analysing the data collected.

This project aims to develop a data visualisation toolset that helps to extract meaningful information, insights and patterns from the taxi operational data recorded by the New York City Taxi & Limousine Commission. The toolset is developed using Java for data pre-processing and a web application developed in HTML and JavaScript for visualisation presentation. It provides interactive features to dynamically visualise the data, delivers the effectual visualisations to unveil useful information hidden in the data.

Swansea University
Prifysgol Abertawe

Contents

1 Introduction

Pressure are being applied on the existing way of governancing cities, 70 per cent of the world's population will be expected to move into cities by 2050 [Uni14]. The rapid expansion of cities mandates the implantation of Smart City, the definition of Smart City may vary from country to country, but the common interest behind is to improve life quality of its citizens by the optimisation of current resources [LR12], to mitigate the problems produced by rapid growth of population (including ageing population) and rapid urbanisation [Cho+11]. In every city, operational data of the city is being collected via Information and Communication Technology (ICT) [DA11], while investing insufficient effort into gathering useful insights and knowledge that the data hold. One of the reasons behind is that raw data is laborious to analyse with human eyes, without effective data visualisation techniques, which extracts the meaningful information by transforming raw data into graphical representations that convey more information via human visual cognition [WGK10].

Since sight is one of our key senses for information understanding, the initiative of this project is to develop an interactive visualisation toolset, revealing the meaningful insights underlie the taxi operational data in New York, collected by New York City Taxi & Limousine Commission, in conjunction with real-time traffic monitoring data provided by Mapbox. The toolset will be beneficial to users across different levels.

The toolset will enable users to dynamically explore, analyse and present the dataset imported, whether it is multivariate, geospatial or time-oriented. The performance of this toolset will be a key benchmark for its success, the target is to deliver a smooth and lightweight toolset with cross-platform compatibility, thus the form of Web Application is chosen. During the planning phase, the existing tools will be reviewed to find out their advantages and disadvantages, and those will be thoroughly considered for development.

One challenge faced in this project will be the data itself. Having different data source providers will result in different data formats, that often require tedious process for pre-processing. In order to overcome that, a Java program will be written specifically for pre-processing data into JSON format, which is both human readable and program friendly. Another challenge will be the tight schedule, some trade-off in interface design will be made in order to ensure the delivery of functionalities.

2 Background

The definition of Smart City has been adapting over the decades, the fundamentals of Smart City are to improve a city's efficiency sustainably [CDN11], thus improving the life quality of its citizens. To achieve sustainability means the optimisation of current resources is critical, by:

- Collecting operational data of the city via Information and Communication Technology (ICT) [DA11]
- Pre-processing the raw data into suitable formats
- Harvesting useful information via data analysis

- Policy-making base on the analytical results [LR12]

In order to carry out effective data analysis, the data collected need to be rigorously pre-processed and visualisations should be generated to convey the underlie information. This further requires appropriate Data Visualisation techniques.

2.1 Literature Review

This section reviews related literatures in the field of Smart City and Data Visualisation.

2.1.1 Understanding Smart Cities: An Integrative Framework

Understand Smart Cities [Cho+11] is a comprehensive paper about smart city. The paper discusses the motivations and purposes behind smart city and gives a set of working definitions of a smart city, a city that monitors and connects its physical infrastructures with ICTs to leverage the efficiency, sustainability, equitability and liveability of the city.

The paper describes a framework with eight crucial factors of a successful smart city:

- Management and organisation
- Technology
- Governance
- Policy
- People and communities
- Economy
- Build infrastructure
- Natural environment

For each of the eight factors, authors detail the challenges faced and the possible strategies to counter them. In particular, the paper emphasises that the inclusions of complex analytics to make better policies, requires visualisation techniques in the decision-making process.

Authors acknowledge that the capture of data from various ICT infrastructures is fundamental to a successful smart city, the decisions made upon those data acquired will offer huge potentials in optimising the operation of the city.

By gaining deep understanding of what a smart city is, this paper inspires the development of this project, precisely what the requirement specification of visualisation tool that suits the needs of a smart city.

2.1.2 Interactive Data Visualization: Foundations, Techniques, and Applications

Interactive Data Visualization is a book [WGK10] introducing the fundamentals of data visualisation concepts and techniques. The book covers the spectrum of usage for data visualisation across different industries.

The book defines visualisation as "the communication of information using graphical representations", the use of data visualisation techniques is prominent to convey more information underlie the dataset given via human visual cognition system. As mentioned by authors in the book, human as visual beings, absorbing information contained in graphs is more efficient than in text or in other formats [WGK10, pp. 3-5].

Figure 1: Examples of the four types of display of hypothetical clinical trial data [Elt+99]

Data visualisation also plays an important role in the decision-making process. The same data presented in different visualisations will have different impact on the final decision. In Figure 1. is the study conducted by [Elt+99] on the influence of visualisation types on clinical decisions, it uses a table, a stacked bar graph, a pie chart and an icon display showing exactly the same dataset, the results of a clinical trial from both conventional and investigational treatments. When a specially chosen group of competent physicians were asked to make a decision whether the clinical trial should be stopped or not, their decision accuracies were interestingly fluctuating from four visualisations, as seen in Table 2.1.2.

Visualisation Type	Decision Time	Decision Accuracy	Preference Rate
Table	35 seconds	68%	61.7%
Stacked bar graph	34 seconds	43%	23.5%
Pie chart	36 seconds	56%	14.8%
Icon display	37 seconds	82%	0%

Table 1: Performance on four types of visualisation in the study conducted [Elt+99]

Despite having the highest decision accuracy with similar time taken for making the decision, icon display was not preferred by any of the participants. Apart from the foundations of data visualisation, there are three chapters that are particularly valuable for this project:

- Visualization Techniques for Geospatial Data
- Visualization Techniques for Time-Oriented Data
- Visualization Techniques for Multivariate Data

In each chapter, not only that the authors present the techniques for visualisations, but also list down the issues faced during visualising the data. Since the proposed toolset that will be mainly used for visualising those three types of data, this book is then extremely beneficial to the development.

2.1.2.1 Visualization Techniques for Geospatial Data

One of the planned visualizations will be to visualise the origin and destination of taxi dataset on a geographic flow map with interactive features. However, the points are often clustering together as shown in Figure 2, in order to generate a geographic flow map that effectively conveys meaningful message, in this chapter authors suggest that those points should be generalised. The process of generalisation involves simplifying points, which essentially means removing or combining the points that are not separately visible on a map with suitable scale [WGK10, pp. 247-249].

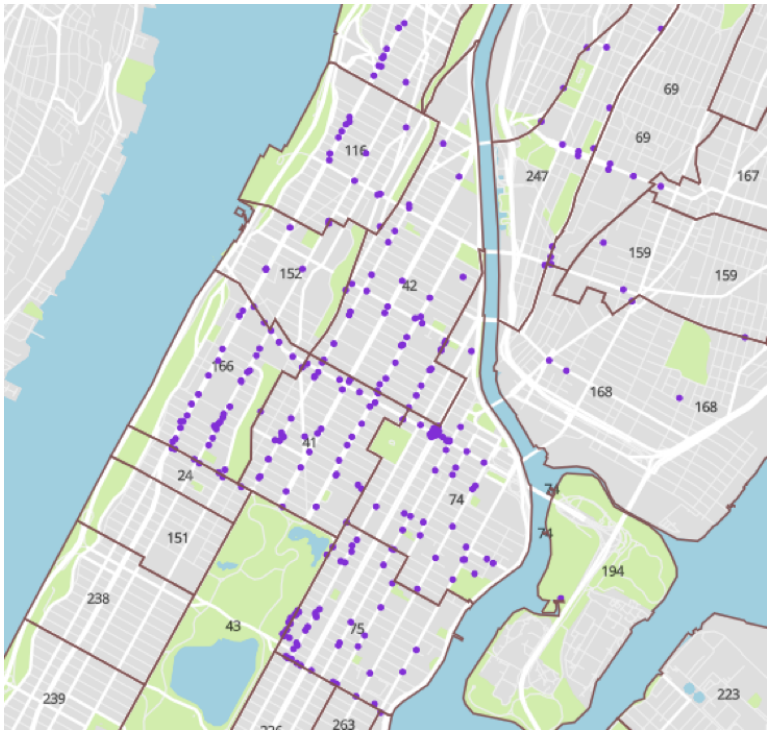

Figure 2: Visualisation of taxi pick-up locations in New York on a prototype of the proposed toolset using Mapbox [Map17a]

2.1.2.2 Visualization Techniques for Time-Oriented Data

According to the authors, time itself is a dimension of any dataset with distinct characteristics, the importance of revealing patterns and relationships of time dimension with other dimensions is illustrated in this chapter. [WGK10, p. 254].

Figure 3 below shows the visualisations of one time-oriented dataset for cases of influenza occurred daily in Germany in a period of three years. The left visualisation is a simple line plot to linearly visualise the data, it clearly shows the peak times of influenza occurrence but the patterns underlie the dataset is hard to conclude. The centre (with one cycle represents 24 days) and the right (with one cycle represents 28 days) use a cyclic visualisation which utilises a spiral-shaped time axis.

There is no pattern to be recognised in the centre visualisation. By adjusting the cycle length to fit into the data's natural interval (multiple of 7 days), the right visualisation is obtained, which instantly reveals that more influenza cases occurred in Mondays than any other days. This example demonstrates that the characteristics of time can significantly change "the expressiveness of visual representations" [WGK10, p. 254].

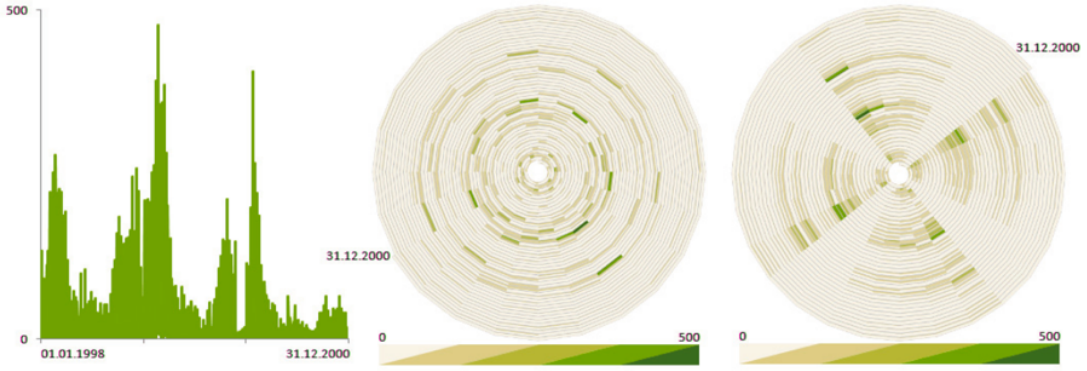

Figure 3: Linear vs cyclic visualisation of time-oriented data [WGK10, p. 255]

For the taxi dataset used for this project, discrete time data is introduced in multiple fields, e.g. taxi pick up time and drop off time. According to the authors, discrete time data is the most widely used quantitative time model. Such dataset can be studied to find out the hidden patterns and 3D visualisation often provides a better illustration of time-oriented dataset [WGK10, pp. 258-263].

2.1.2.3 Visualization Techniques for Multivariate Data

In the field of Data Visualisation, multivariate data refers to datasets that have more than three variables. Visualisation of multivariate data is referred by Liu [Liu17] as *Curse of Dimension* due to the fact that the effectiveness of retinal visual elements such as colour, shape and size will deteriorate as the number of variables increases.

This chapter introduces visualisation techniques that maintain characteristics of multivariate dataset while providing effective renderings. The taxi dataset contains multiple point plots as the main data format, therefore the techniques for visualising point-based multivariate data are beneficial to the development of the proposed toolset. Four point-based techniques are described [WGK10, pp. 285-292] in this chapter as follows:

- Dimension Subsetting – dimensions can be selectively displayed by the user or the toolset automatically selects the most meaningful dimensions to visualise.
- Dimension Reduction – dimensionality reduction algorithms such as principal component analysis (PCA) will be applied to project higher dimensional dataset into lower dimensions, meanwhile it tries to reduce the loss of information to the minimum during the process.
- Dimension Embedding – dimensions can be mapped to various graphical representations besides position, e.g. colour, size and shape. This is however limited as mentioned previously as *Curse of Dimension*.
- Multiple Displays – visualisations are superimposed or juxtaposed. The classic example of multiple displays is the use of scatterplot matrix.

2.2 Existing Software API

2.2.1 AnyMap.js

AnyMap.js is a JavaScript based geospatial visualisation library that was initially written for businesses who want to transform operational data into actionable information via data visualisation. The library requires a commercial license but it is available for personal evaluation for an unlimited period of time here: `https://static.anychart.com/cdn/js/7.13.1/anymap.min.js?download`.

It provides complete cross-platform compatibility with support for popular data formats such as XML and JSON. AnyMap.js is also embedded with a full set of JavaScript API and offers a variety of options to create geospatial visualization with interactivity and simplicity.

Figure 4, Figure 5 and Figure 6 are examples of Bubble Map,Connector Map and Choropleth Map that are enlightening for the development of this proposed toolset.

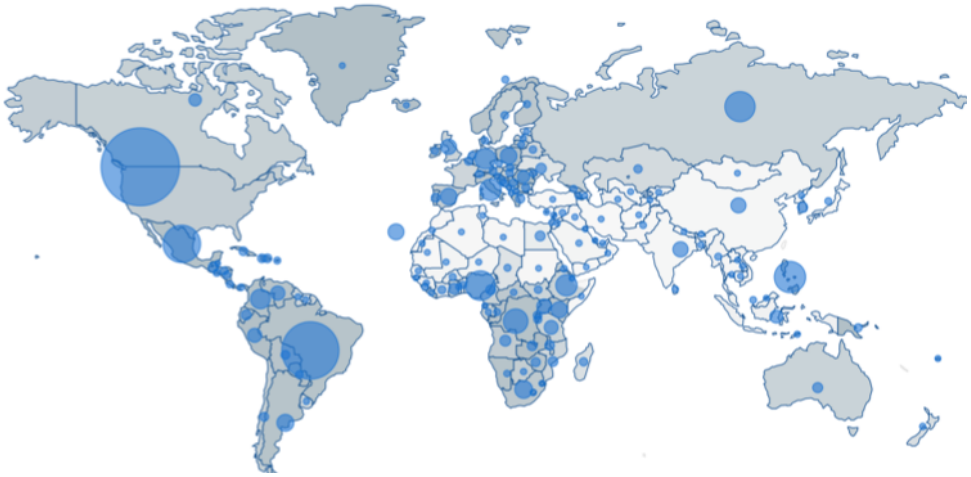

Figure 4: Visualisation using a Bubble Map with AnyMap.js [Any17a]

Figure 5: Visualisation using a Connector Map with AnyMap.js [Any17b]

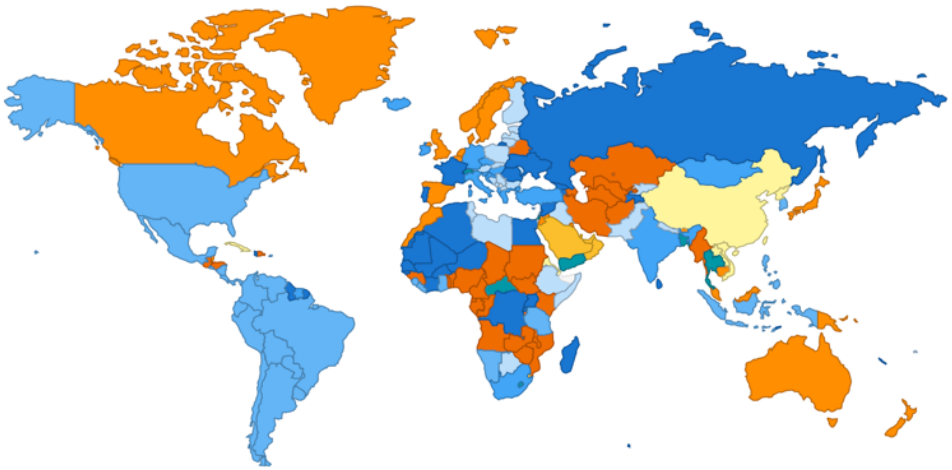

Figure 6: Visualisation using a Choropleth Map with AnyMap.js [Any17c]

2.2.2 Mapbox Studio and Mapbox.js

Mapbox is mapping platform for developers. Mapbox Studio is a web application that enables developers to create base map styles with ease. It allows the user to upload geospatial dataset in GeoJSON and CVS format, Mapbox Studio will then convert the dataset into a format called Tileset that stores both vector and raster data. Each tileset can be visualised on a base map as a separated layer. By doing this, a map can be modified to visualise arbitrary dimensions of data.

In Figure 7 below, six customised layers are visualised on a base map of New York.

1. Taxi zones are segmented using brown boundary.
2. Taxi pick-up locations are depicted using purple dots.
3. Low traffic congestion is mapped to lines in green.
4. Moderate traffic congestion is mapped to lines in yellow.
5. Heavy traffic congestion is mapped to lines in orange.
6. Severe traffic congestion is mapped to lines in red.

Those layers form a new base map and Mapbox Studio outputs a unique URL of that base map. By making API request containing the unique URL using Mapbox.js, the newly created base map can be initialised on different platforms as long as JavaScript and HTML are supported.

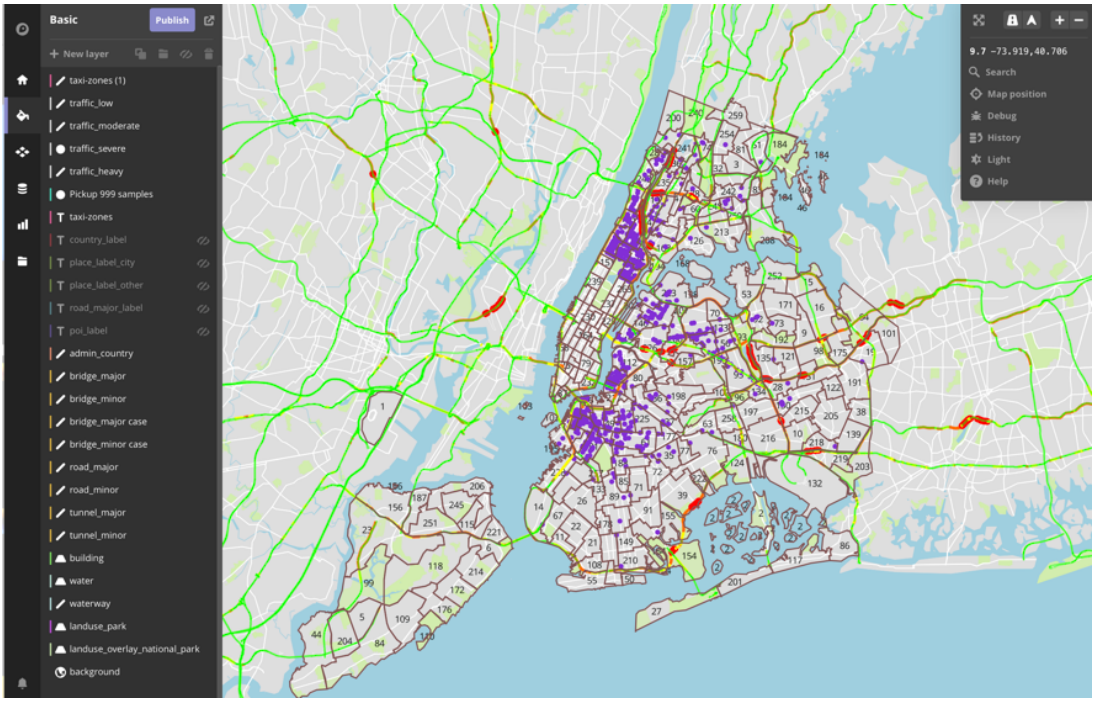

Figure 7: Interface of Mapbox Studio when creating a prototype base map [Map17a]

2.3 Similar Applications

2.3.1 MONiTOUR

MONiTOUR [Rat+16a] in Figure 8, is an e-waste tracker that visualises the travel routes of obsoleted printers, LCD and CRT monitors from the US. As a part of the joint project *e-trash Transparency Project*, MIT Senseable City Lab and Basel Action Network embedded GPS trackers into those e-wastes in order to obtain the travel routes.

The results are visualised on a world map, blue dots and red dots represent the origin and destination respectively, with white lines clearly depict that the majority of e-waste travelled to Asia. Filtering on device type and region. Figure 9 shows when a route is selected, the detailed travel history of the GPS tracker in steps.

The app is written in JavaScript with HTML 5 and the base map is from Mapbox, it is available for free at `http://senseable.mit.edu/monitour-app/`

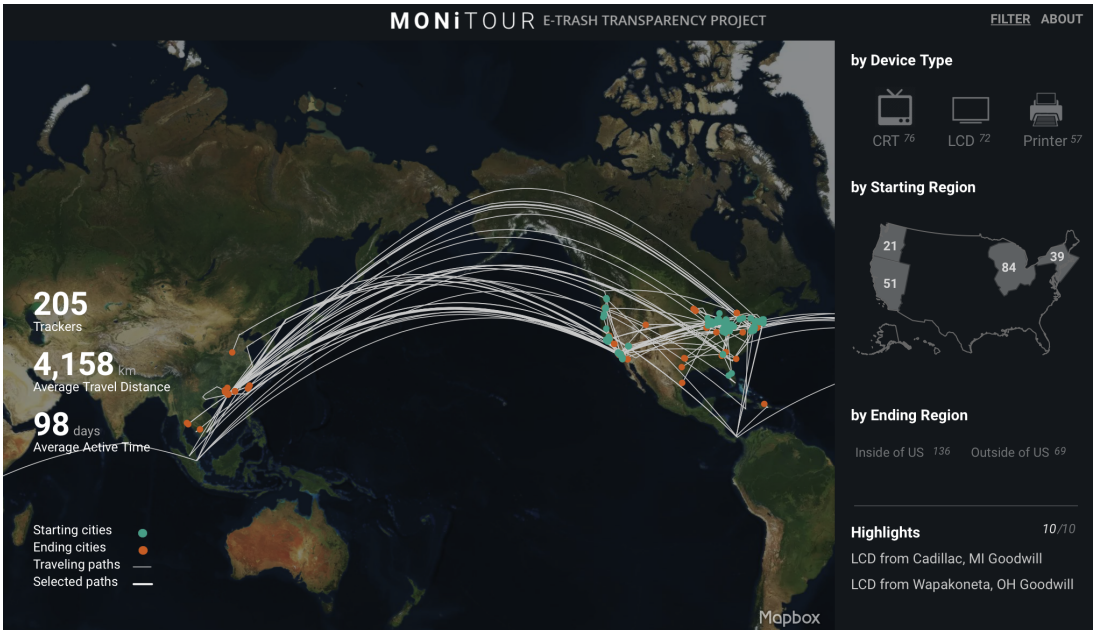

Figure 8: Interface of MONiTOUR visualising origins, destinations and paths of 205 GPS trackers embedded into printers, LCD and CRT monitors [Rat+16a].

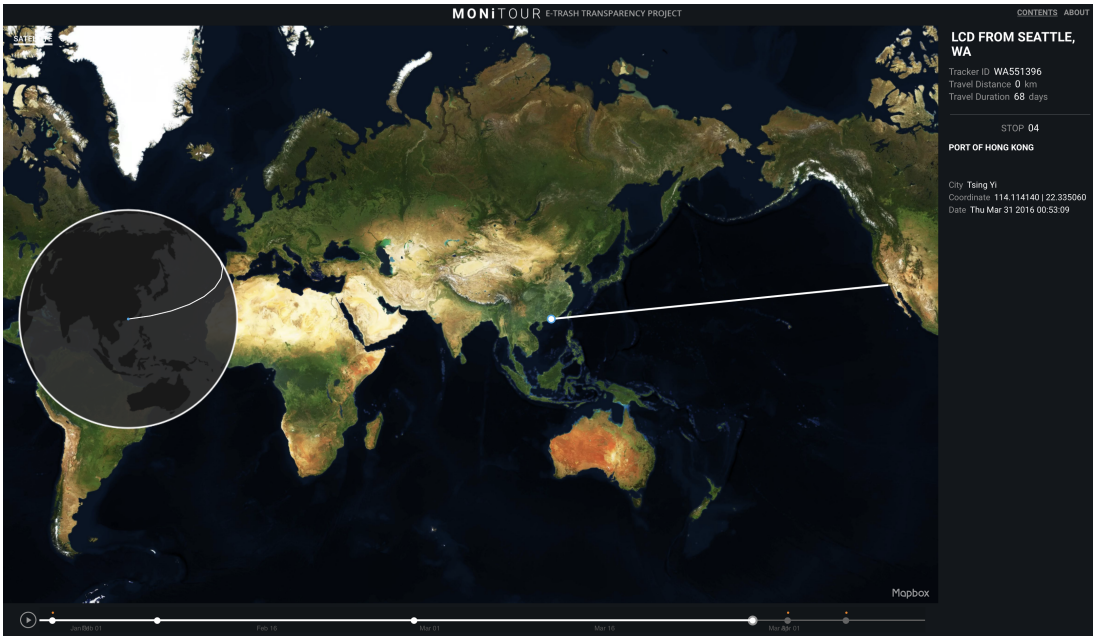

Figure 9: Interface of MONiTOUR visualising the travel history of a GPS tracker embedded into a LCD monitor, from Seattle to Hong Kong [Rat+16b].

2.3.2 Boston 311

Boston 311 [GLO12] in Figure 10 is a geospatial visualisation of 3-1-1 incident reports in Boston from October 2010 to June 2012.

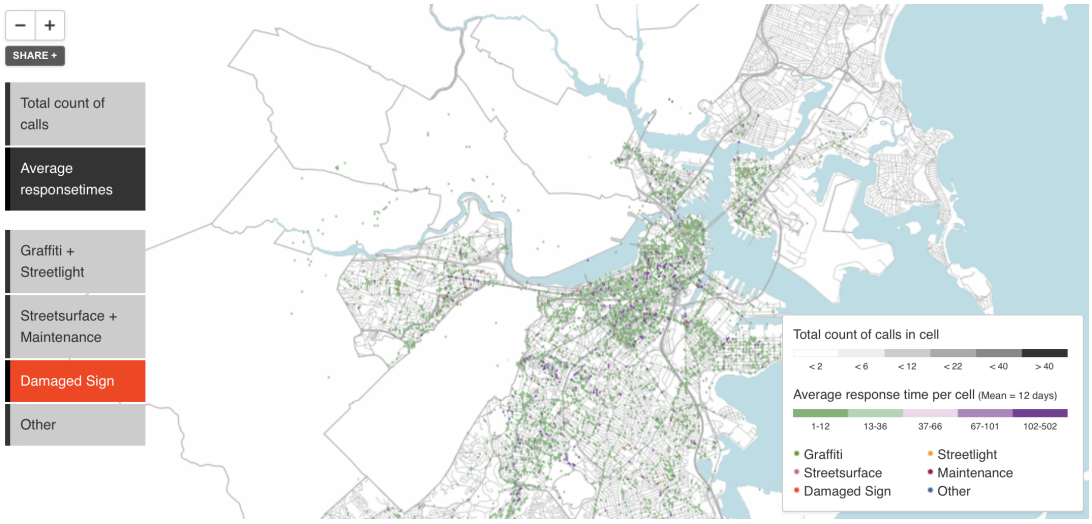

Figure 10: Interface of Boston311 visualising average response time against damaged sign in Boston [GLO12].

Two dimensions can be visualised simultaneously by selecting them in the left panel. The first dimension can be either total count of calls or average response time, whereas the second dimension can be chosen from graffiti and streetlight, streetsurface and maintenance, damaged sign or other.

2.4 Limitation of Similar Applications

2.4.1 MONiTOUR

The performance of MONiTOUR is a major concern for visualisation of millions of taxi operational data entries. During the experiment of MONiTOUR, even there are only 205 distinct paths visualised on the map, the performance is not optimal. The web app lags significantly when applying zoom and pan operations, using Chrome (Version 58.0.3029), Firefox (Version 52.1.0) and Safari (Version 10.1). The initialisation of the application also takes more than ten seconds with a fast internet connection.

Certain level of filtering is allowed in MONiTOUR but is far from sufficient for visualisation of taxi operational data, due to the high dimensionality.

Visualisation of time-oriented data is not built into this application.

2.4.2 Boston 311

Filtering is extremely limited in Boston 311, apart from selecting different dataset, users are unable to select subset of data for visualisation.

Visualisation of time-oriented data is not built into this application.

3 Project Specification

The primary objectives of this project is to develop a visualisation toolset for the taxi operational data. The toolset should contain the following major features:

- Converting large chunk of taxi operational data in CSV format into JSON and GeoJSON formats, to be used for different software APIs
- Exploring the data via various types of visualisation
- Analysing the visualisation via different interactive functions such as filtering of data dimension or volume.

The project should help the author to gain deeper understanding of information visualisation, the usefulness and the difficulties in obtaining such visualisation. Due to the massive amount of data used, data pre-processing will also be focused during the development phase.

3.1 Data Characteristics

3.1.1 Taxi Operational Data recorded by the New York City Taxi & Limousine Commission (TLC)

Taxicab & Livery Passenger Enhancement Programs (TPEP/LPEP) was introduced in March 2004 by the TLC, the primary objective this program is to improve New York taxi service by implementing specific technology to collect taxi operational data, including pick-up and drop-off location, trip distance and time, passenger counts and others. Prior to the TPEP/LPEP, this data collection process was manually done by the drivers using a hand-written log book, the introduction of the program significantly improved accuracy of the data and efficiency of the data collection process [NYC].

Table 3.1.1 shows the description of the data provided by TLC at `http://www.nyc.gov/html/tlc/html/about/trip_record_data.shtml`.

For each month, three datasets in CSV format are provided, Yellow Taxi, Green Taxi and For-hire Vehicle (FHV). Due to the nature of FHV service, its dataset contains only pick-up time and pick-up borough ID, thus the dataset is not used in this project.

Table 3.1.1 shows the difference between three datasets in terms of dataset size and number of records.

Dataset	**Dataset size**	**Number of records**
Yellow Taxi	1800 MB	11,500,000
Green Taxi	200 MB	1,500,000
For-Hire Vehicle (FHV)	300 MB	8,500,000

Table 2: Taxi operational datasets [NYC17]. Each month the size and number of records vary slightly, all data shown in the table are the estimated average.

Column Name	Data Type	Data Description	Sample
VendorID	Integer	Vendor that provided the data 1= Creative Mobile Technologies, LLC 2= VeriFone Inc.	1
Lpep_pickup_datetime	Datetime	Taxi pick-up date time	01/01/2015 02:21
Lpep_dropoff_datetime	Datetime	Taxi drop-off date time	01/01/2015 03:21
RateCodeID	Integer	Rate code for the trip 1= Standard rate 2=JFK 3=Newark 4=Nassau or Westchester 5=Negotiated fare 6=Group ride	1
Pickup_longitude	Longitude	Taxi pick-up longitude	-73.86389923
Pickup_latitude	Latitude	Taxi pick-up latitude	40.72250748
Dropoff_longitude	Longitude	Taxi drop-off longitude	-73.94389923
Dropoff_latitude	Latitude	Taxi drop-off latitude	40.70850748
Passenger_count	Integer	Number of passengers for the trip	5
Trip_distance	Double	Total distance for the trip	14.01
Fare_amount	Double	Total fare for the trip	16.55
Extra	Double	Extra surcharge for the trip	0.5
MTA_tax	Double	Metropolitan Transit Authority Tax for the trip	0.5
Tip_amount	Double	Total tip amount for the trip	3.5
Tolls_amount	Double	Total toll amount for the trip	5.33
Improvement_surcharge	Double	Taxi Improvement Fund surcharge	0.3
Total_amount	Double	Total charge amount for the trip (Fare + all surcharges)	20.55
Payment_type	Integer	Payment type used for the trip 1= Credit card 2= Cash 3= No charge 4= Dispute 5= Unknown 6= Voided trip	1

Table 3: Description of taxi operational data collected by TLC [NYC15].

Starting from July 2016, instead of the coordinates, only pick-up and drop-off borough ID are provided, as the result only data prior to that are used in this project. Figure 11 and Figure 12 are screenshots of Green taxi operational dataset from January 2015.

The dataset contains both geospatial (longitude and latitude) and discrete time-oriented data (pick-up and drop-off time).

VendorID	lpep_pickup_datetime	Lpep_dropoff_datetime	RateCodeID	Pickup_longitude	Pickup_latitude	Dropoff_longitude	Dropoff_latitude
2	01/01/2015 00:34	01/01/2015 00:38	1	-73.92259216	40.75452805	-73.91363525	40.765522
2	01/01/2015 00:34	01/01/2015 00:47	1	-73.95275116	40.67771149	-73.98152924	40.65897751
1	01/01/2015 00:34	01/01/2015 00:38	1	-73.84300995	40.71905518	-73.84658051	40.71156693
2	01/01/2015 00:34	01/01/2015 00:38	1	-73.86082458	40.75779343	-73.85404205	40.74982071
2	01/01/2015 00:34	01/01/2015 01:09	1	-73.9451828	40.78332138	-73.98962402	40.76544952
1	01/01/2015 00:34	01/01/2015 00:40	1	-73.96681213	40.7146759	-73.94940948	40.71843719
1	01/01/2015 00:34	01/01/2015 00:53	1	-73.93048859	40.85013199	-73.97805786	40.78905869
2	01/01/2015 00:35	01/01/2015 00:35	5	-73.86389923	40.89543915	-73.86187744	40.89477921
2	01/01/2015 00:35	01/01/2015 00:41	1	-73.91712952	40.76488876	-73.92797852	40.76147079

Figure 11: Partial screenshot of Green Taxi operational dataset from January 2015 [NYC17].

Passenger_count	Trip_distance	Fare_amount	Extra	MTA_tax	Tip_amount	Tolls_amount	improvement_surcharge	Total_amount	Payment_type
1	0.88	5	0.5	0.5	0	0	0.3	6.3	2
1	3.08	12	0.5	0.5	0	0	0.3	13.3	2
1	0.9	5	0.5	0.5	1.8	0	0	7.8	1
1	0.85	5	0.5	0.5	0	0	0.3	6.3	2
1	4.91	24.5	0.5	0.5	0	0	0.3	25.8	2
4	1.2	6.5	0.5	0.5	0	0	0.3	7.8	2
1	6.6	22	0.5	0.5	0	0	0.3	23.3	2
1	0.13	15	0	0	0	0	0	15	1
5	1.18	6.5	0.5	0.5	1.4	0	0.3	9.2	1

Figure 12: Partial screenshot of Green Taxi operational dataset from January 2015 [NYC17].

A .shp file containing boundaries of 354 taxi zones provided by TLC can be downloaded from `https://s3.amazonaws.com/nyc-tlc/misc/taxi_zones.zip`. .shp format is a popular geospatial vector data format for geographic information system (GIS) software.

3.2 Feature Specification

The proposed toolset will have the following features

1. An interactive and responsive web application that contains
   (a) A tool to generate a map that predominantly as a geospatial visualisation but also contains time-oriented dimension
   (b) A tool to generate a chord diagram for taxi trip origin and destination visualisation, see Figure 13

Figure 13: A chord diagram using D3.js [Bos17].

(c) A tool to generate a sun-burst diagram for visualising multivariate data into sequential order, see Figure 14

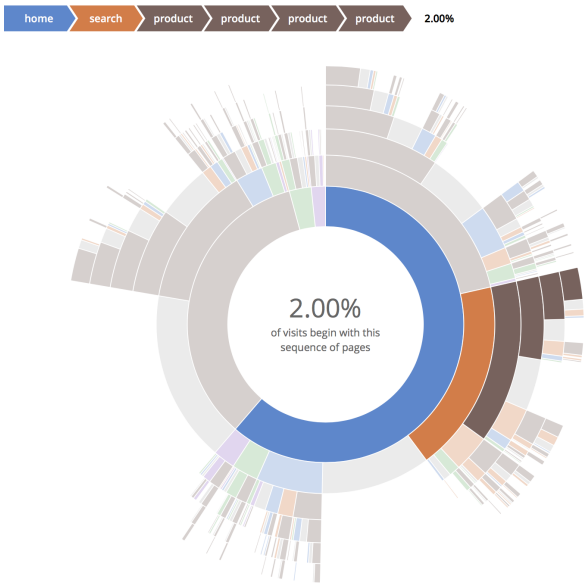

Figure 14: A sun-burst diagram using D3.js [Rod17].

2. A function to adjust time in order to present time-oriented data dimension
3. A function to filter the data to be visualised
4. A function to export the visualisation in image format

3.3 Technology Choices

This section introduces the fundamental technology used, different technology will be used during different phase of the development, to suite the needs during the phase.

The primary hardware for the development will be on a MacBook Pro embedded with an Intel i7-6920HQ CPU and 16GB of RAM, with the operating system of macOS Sierra 10.12.4. This will ensure that the large dataset can be pre-processed efficiently and the hardware is sufficient to run virtual machines of other operating system for testing purpose.

3.3.1 Data Pre-processing

Java is chosen as the primary language for data pre-processing. Java is one of the most popular programming language that has great cross platform support with detailed documentation. The author also has years of experience in Java, thus Java is chosen to expedite the data pre-processing phase.

JSON-simple [Fan] is a Java library that enables Java for manipulating JSON format in accordance with JSON specification RFC4627 [Cro06]. Taxi operational data in CSV is piped into a Java program written for producing GeoJSON format follows the specification RFC7946 [But+16] that Mapbox Studio accepts.

The IDE for Java will be Eclipse version 4.6.3 with JDK build 1.8.0_121b1.

3.3.2 Web Application Development

The toolset will be presented in the form of Web application, all interactive functions will be manipulated by users via browsers. The application will mainly use HTML and CSS for presentation and JavaScript for functionality.

This combination is supported by any modern device with a screen, there are many existing frameworks such as AngularJS and Bootstrap to expedite the development process.

Microsoft Visual Studio IDE was considered to be the IDE for the development of Web Application. The IDE is fully-featured with development, testing and publication of the application. However, it only supports Windows and the publication and hosting of the application is tied to Windows Azure. Therefore, Microsoft Visual Studio Code, the code editor that inherits a certain level of functionality from the IDE, will be used for the development.

Visual Studio Code is an open source (MIT License) code editor and has a huge library of extensions with active communities, it supports all major operating system. More importantly, it is capable of handling all the tasks involved in the development.

Visual Studio Code version 1.11.2 will be used for completing the web application.

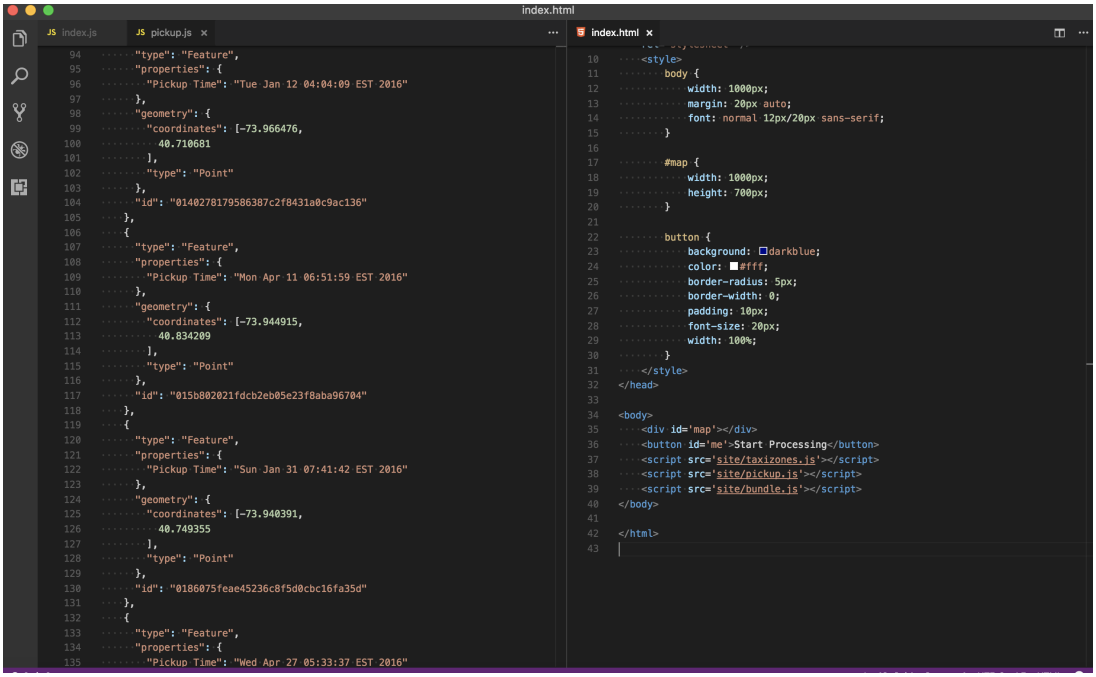

Figure 15: Interface of Visual Studio Code editing HTML and JavaScript files.

3.3.3 Others

3.3.3.1 Version Control System

Git is the most widely used version control system that is free (GNU General Public License v2) and supports all major operating systems.

The free Git repository hosting service GitHub will be used to store all the code. GitHub will also serve as a regular backup warehouse to prevent any accident that may result in lost or corrupted files.

Git version 2.11.0 with the command line tools will be used throughout the whole development lifecycle.

3.3.3.2 Testing and Documentation

Testing and documentation are two essential steps for software development. According to Vliet, the cost of repairing errors made at an early stage is extremely high if they are discovered at a late stage of the development [Vli07, p. 386], thus the testing should be frequently conducted during the development to minimise the negative impact of errors and bugs.

Unit Testing is a testing process applied on every testable parts of a software, instead of testing the software as a whole, the focus of unit testing is the smaller building blocks called *unit*, such as classes and procedures [MTS04, p. 85]. The benefits of unit testing are apparent:

- Each unit and combinations of units are covered easily
- Reflecting impacts from code modifications

- Locating errors precisely to its source

JUnit is a framework to conduct unit testing for Java, Emma is an open source (Common Public License v1.0) JUnit tool for Eclipse [Rou10], it is capable of automating unit testing and will be used the development and testing stage of this project. Similarly, the parts of implementation using JavaScript will be unit tested using QUnit [QUn].

Doxygen [Dox] will be used as the documentation generating tool, it is free under the GNU General Public License v2. Following Bob's Concise Introduction to Doxygen [Lar11a], Doxygen is capable of generating well structured documents base on the comments in the code. It will also product class hierarchy and collaboration diagrams. A HTML version will also be generated, makes it easier to publish the documentation online.

Using Doxygen also encourages developers to comment the code properly during development, this improves code quality and reduces bugs.

4 Project Plan and Timetable

4.1 Development Strategy

These are five essential stages of software development [Lar11b; Vli07, p .11].

Five sections are included in the gantt chart:

1. Requirements specification
2. Software design
3. Implementation
4. Testing
5. Documentation

Arnuphaptrairong [Arn11] discovered that project planning is the area where the most number of risks originated. Therefore, this project will strictly adopt the Scrum Agile software development framework to plan and control the project development throughout the entire lifecycle.

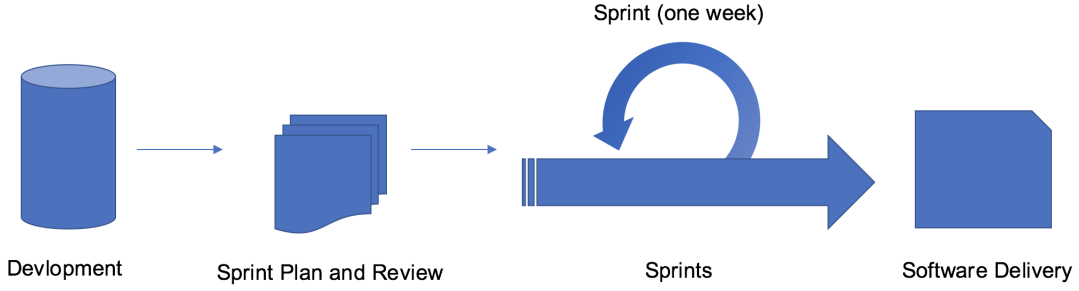

Figure 16: Slightly modified Scrum workflow for this project.

Scrum is the most widely adopted Agile framework. Figure 16 shows a Scrum workflow tailored for this project. Development is broken into smaller Sprints, each Sprint takes one week to develop, completion of all Sprints signals the shipment of software. The original Scrum workflow requires a daily team meeting during each Spring, since the author is the sole developer in the team, a daily meeting with team members during each Spring is cancelled. Sprint requires planning ahead and reviewing after, at the end of each Sprint, the development team will review it with stakeholders to identify improvements or corrections needed.

The flexibility of Scrum allows the author to embrace changes to requirements and correct misinterpretation of requirements within an acceptable timeframe.

4.2 Gantt Chart

Figure 17 shows the gantt chart for this project.

4.3 Risk Analysis

This analysis outlines the potential risks that will have catastrophic damage on the development or severely affect the progress of development and strategies to counter them.

4.3.1 Risk Identification

The identification of risks followed the research carried out by Arnuphaptrairong [Arn11]. Seven risks are rated as the most frequently encountered in software development lifecycle in the research as follows:

1. Misunderstanding of requirements
2. Lack of top management commitment and support
3. Lack of adequate user involvement
4. Failure to gain user commitment
5. Failure to manage end user expectation
6. Changes to requirements
7. Lack of an effective project management methodology

4.3.2 Risk Mitigation

This section outlines the mitigation measures against the risks identified.

Table 4.3.2 shows the risk identified by the author. The first risk, the involvement of new technology with a high level of technical complexity is rated as the riskiest factor. Obstacles are expected in the project due to the novelty of the author, the involved technology will be carefully selected by the following factors:

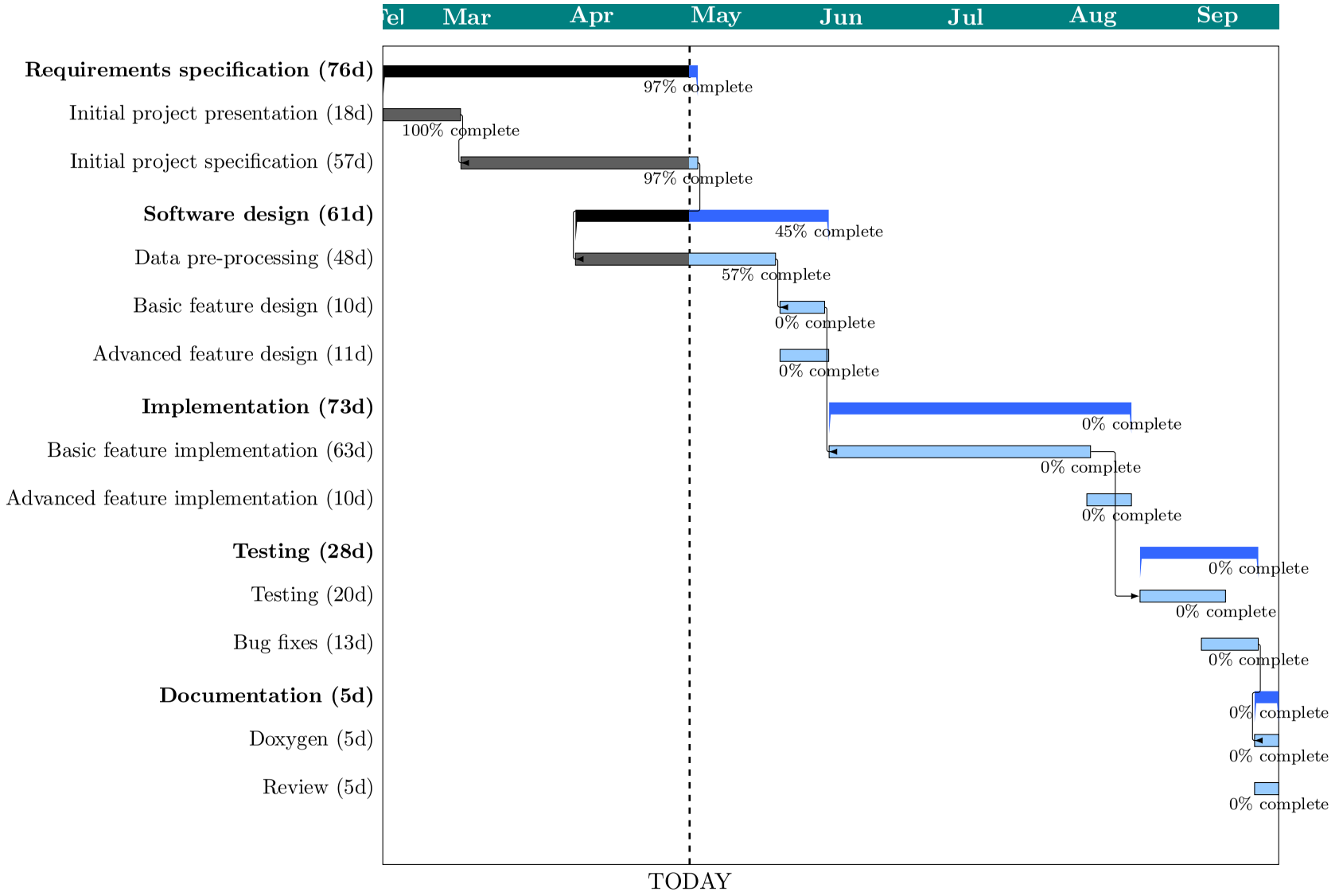

Figure 17: Project gantt chart as of May 9, 2017

- Language structure and syntax are close to the author's prior experience
- Detailed documentation for the technology
- Preferably free and open source
- Active communities are in place
- Successful software built upon the technology can be accessed and reviewed

The second risk, misinterpretation of requirements has serious consequence, it mitigated by conducting regular meeting with supervisor. Scrum workflow also helps minimise the risk.

The third risk, Java and JavaScript are the main languages used for this project, they are both well-documented and have active communities to provide technical support. This mitigates the risk from inadequate programming skills of the author.

The fourth risk, the development will start with high-value features with low-risk, proceeding to low-value features with low-risk and eliminating or substituting low-value features with high-risk. This ensures that the final project will deliver useable features.

The fifth risk, two volunteers with Data Visualisation background are chosen to test the features at the end of each or several Sprints, provide insightful feedback to minimise the risk brought by insufficient user involvement and feedback.

The sixth risk, GitHub, Dropbox and a local external harddrive are used for backup to ensure any equipment failure will not result in delay of the development.

The last risk is personal illness which is unavoidable, ensuring GP Surgeries are within reach can help minimise the damage.

5 Initial Implementation

This section describes the initial stage of the implementation, primarily on data pre-processing.

5.1 Data Pre-processing

There are several steps involved in preparing or enhancing the taxi dataset.

1. Filtering out columns that are not used for this project
2. Correcting fields with errors
3. Handling missing values
4. Deriving new columns

	Risk	Probability	Impact	Mitigation measures
1	Involvement of new technology with a high level of technical complexity	High	High	Choose the technology involved carefully, avoid unrealistic or unnecessary features and invest more time into exploring the technology used
2	Misinterpretation of requirements	Medium	High	Regular meeting with supervisor for consultations and progress monitoring
3	Inadequate skills and knowledge in programming	Medium	High	Experienced programming language with detailed documentation and active community is used
4	The final software developed has serious bugs that makes it unusable	Medium	High	Features are developed in an order ranked by their value/risk ratio
5	Insufficient user involvement and feedback	Medium	Medium	Volunteers with data visualisation background are regularly surveyed when each version rolls out
6	Equipment failure resulting in loss of files	Low	High	GitHub and Dropbox are used for regular backup of files. Laptop is backup daily using an external harddrive
7	Lack of effective project management skill	Low	High	Adopt Scrum Agile software development framework strictly
8	Personal illness	Low	Medium	GP Surgeries are within reach

Table 4: Risk analysis and mitigation.

In the first step, unnecessary columns such as VendorID, RateCodeID and improvement_surcharge are irrelevant, are filtered out to optimise storage and improve performance of the toolset.

No errors were found during the second step, due to the standardised and digitised method used by TLC to collect those data.

Some empty fields were filled with a selected value. For example, some records are missing Payment_type, the field is irrelevant to geospatial visualisation presented by the toolset, however, in chord diagram and sun-burst diagram, the field can be used as one dimension itself. Therefore, empty fields were taken as 5-Unknown.

A Java program was written to convert data into a format that Mapbox Studio accepts. The program reads all records and write into GeoJSON format, Lpep_pickup_datetime and Lpep_dropoff_datetime fields are converted from Unix time into Datatime. At this stage, base map layer can already be created using Mapbox Studio.

However, some columns need to be derived for better visualisation of the data. The original dataset does not contain taxi zone names for pick-up and drop-off. Taxi zones are polygons highly customised by TLC, therefore reverse geocoding from popular geospatial APIs such as Google Maps or Mapbox is not capable of returning the correct taxi zones. In order to obtain this column and add it into the dataset, all coordinates must be first visualised on the map in one layer together with a layer that contains all taxi zone polygons. Taxi zone polygons were obtained in .shp format, Ogre [Har16] was used to convert it into GeoJSON format.

A JavaScript program based on Leaflet's Point in Polygon [Map17b] was written to determine in which taxi zone polygon that the specific coordinate is located within, and add the taxi zone name into dataset.

More details about the process are illustrated in Figure 18.

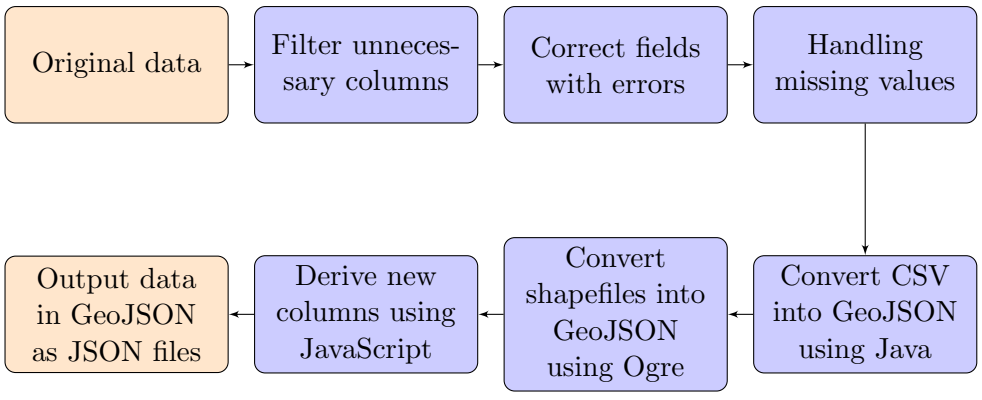

Figure 18: Steps for initial data pre-processing.

5.2 Geospatial Visualisation

After obtained the processed dataset from the previous section, base map layer can be created by uploading the processed dataset along with the converted taxi zone

polygons data, both are JSON files containing data in GeoJSON format. Figure 19 and Figure 18 shows the base maps obtained.

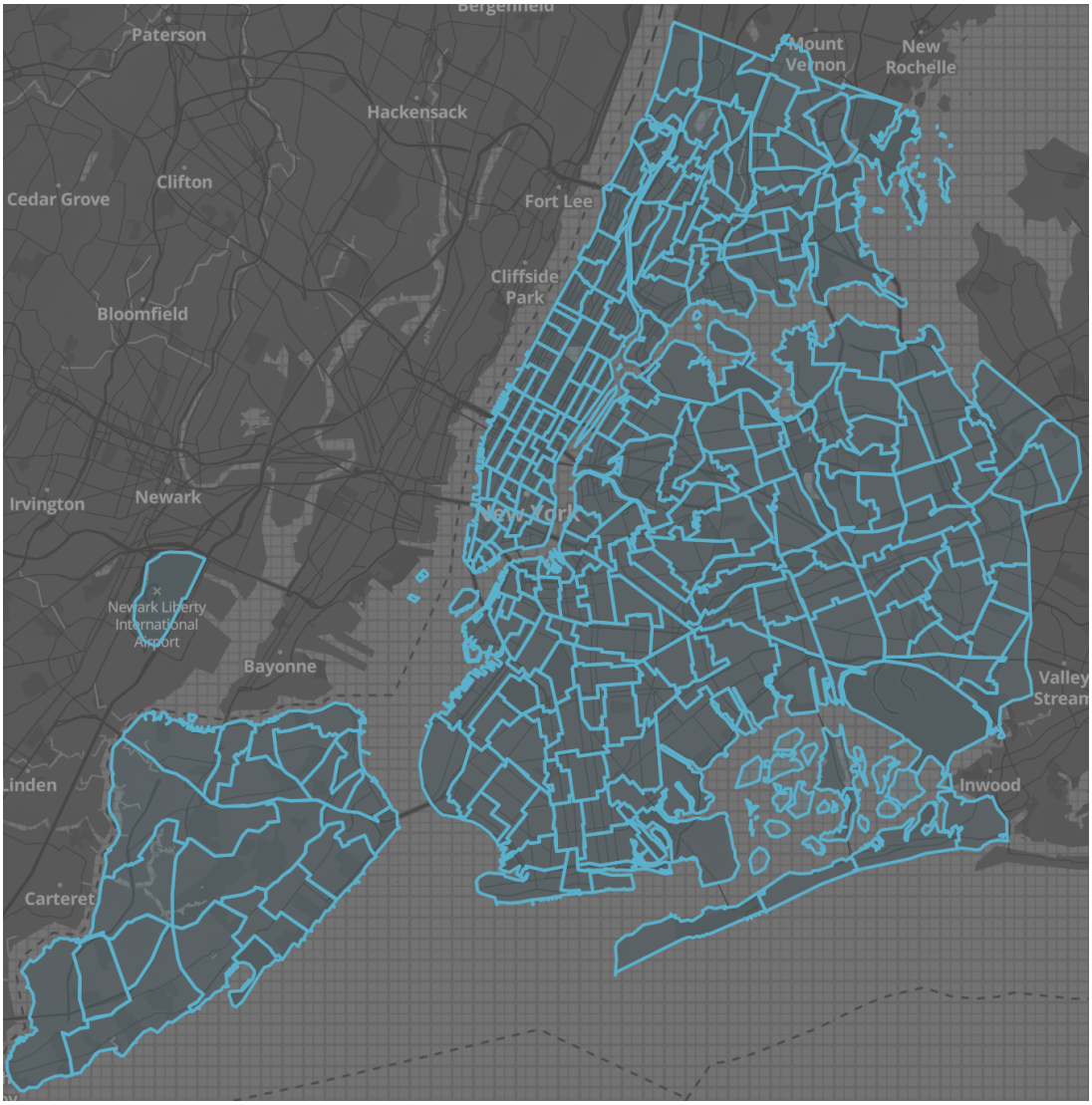

Figure 19: A base map layer containing a visualisation of 354 taxi zones polygons using Mapbox Studio [Map17a].

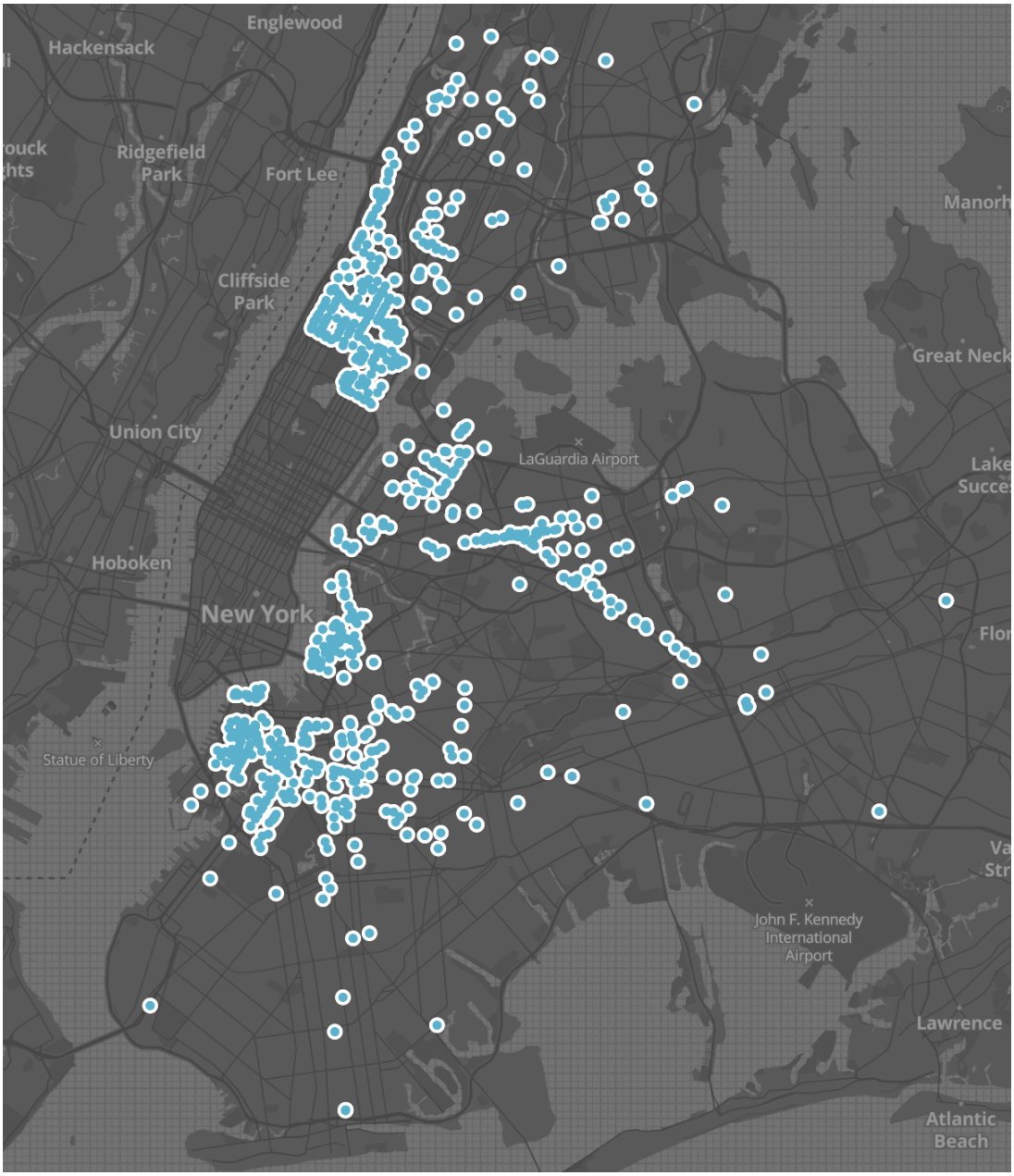

Figure 20: A base map layer containing a visualisation of 1,000 taxi pick-up locations using Mapbox Studio [Map17a].

6 Conclusion

The project aims to provide an interactive visualisation toolset to visualise the taxi operational data collected by the New York City Taxi & Limousine Commission (TLC), that enables users to select and filter data dimension to present.

Three visualisation types will be provided:

1. A geospatial visualisation in the form of a map, that also contains a time-oriented

dimension.

2. A chord diagram that visualises the inter-relationships between entities within a given dimension.

3. A sun-burst diagram that visualises dimensions into sequential order.

The toolset aims to help users in harvesting the useful insights from abundant data for better decision-making process.

The project is still at a very early stage, this interim document may change along the development lifecycle. However, one key objective will stay the same throughout the whole project, that is for the author to gain deeper understanding of data visualisation and its application in solving real world problems. The development process of this novel toolset also serves as a chance to practise programming skills and data processing techniques.

7 Acknowledgements

I would like to express my gratitude to my supervisor Robert S. Laramee from Swansea University, for his guidance and support during this project.

My sincere appreciation is extended to Anton G. Setzer from Swansea University, for his guidance and support for project research methods.

I thank Liam McNabb from Swansea University, for providing help on the topic of Smart City Visualisation.

I also thank Bonie Zhang and Yang Liu from Rutgers Business School, for providing help during the data acquisition stage.

I would also like to thank Swansea University for providing researching resources as well as the knowledge needed for conducting this research.

8 References

[Any17a] AnyChart. *Bubble Christian Map — Bubble Maps — AnyMap Gallery — AnyChart*. 2017. URL: http://www.anychart.com/products/anymap/gallery/Maps%7B%5C_%7DBubble/Bubble%7B%5C_%7DChristian%7B%5C_%7DMap.php (visited on 04/20/2017).

[Any17b] AnyChart. *Busiest Routes From Heathrow Airport — Connectors Maps — AnyMap Gallery — AnyChart*. 2017. URL: http://www.anychart.com/products/anymap/gallery/Maps%7B%5C_%7DConnectors/Busiest%7B%5C_%7DRoutes%7B%5C_%7DFrom%7B%5C_%7DHeathrow%7B%5C_%7DAirport.php (visited on 04/20/2017).

[Any17c] AnyChart. *World Governments Map — Choropleth Maps — AnyMap Gallery — AnyChart*. 2017. URL: http://www.anychart.com/products/anymap/gallery/Maps%7B%5C_%7DChoropleth/World%7B%5C_%7DGovernments%7B%5C_%7DMap.php (visited on 04/20/2017).

[Arn11] Tharwon Arnuphaptrairong. "Top Ten Lists of Software Project Risks: Evidence from the Literature Survey". In: *Proceedings of the International MultiConference of Engineers and Computer Scientists (IMECS2011)* I (2011). URL: http://www.iaeng.org/publication/IMECS2011/IMECS2011%7B%5C_%7Dpp732-737.pdf.

[Bos17] Mike Bostock. *Chord Diagram - bl.ocks.org*. 2017. URL: https://bl.ocks.org/mbostock/1046712 (visited on 04/25/2017).

[But+16] H. Butler et al. *The GeoJSON Format*. 2016. DOI: 10.17487/RFC7946. URL: https://tools.ietf.org/html/rfc7946.

[CDN11] Andrea Caragliu, Chiara Del Bo, and Peter Nijkamp. "Smart Cities in Europe". In: *Journal of Urban Technology* 18.2 (2011), pp. 65–82. ISSN: 1063-0732. DOI: 10.1080/10630732.2011.601117. arXiv: arXiv:1011.1669v3.

[Cho+11] Hafedh Chourabi et al. "Understanding Smart Cities: An Integrative Framework". In: *Proceedings of the Annual Hawaii International Conference on System Sciences* (2011), pp. 2289–2297. ISSN: 15301605. DOI: 10.1109/HICSS.2012.615.

[Cro06] Douglas Crockford. *The application/json Media Type for JavaScript Object Notation*. 2006. DOI: 10.17487/rfc4627. (Visited on 04/26/2017).

[DA11] Mark Deakin and Husam Al Waer. "From intelligent to smart cities". In: *Intelligent Buildings International* 3.3 (2011), pp. 133–139. ISSN: 1750-8975. DOI: 10.1080/17508975.2011.586673.

[Dox] Doxygen. *Doxygen: Main Page*. URL: http://www.stack.nl/%7B~%7Ddimitri/doxygen/index.html%20http://www.stack.nl/%7B~%7Ddimitri/doxygen/ (visited on 04/29/2017).

[Elt+99] L S Elting et al. "Influence of data display formats on physician investigators' decisions to stop clinical trials: prospective trial with repeated measures." In: *BMJ (Clinical research ed.)* 318.7197 (1999), pp. 1527–1531. ISSN: 0959-8138. DOI: 10.1136/bmj.319.7216.1070.

[Fan] Yidong Fang. *JSON-simple*. URL: https://code.google.com/archive/p/json-simple/ (visited on 04/25/2017).

[GLO12] Benedikt Gross, Joseph K. Lee, and Dietmar Offenhuber. *Boston 3-1-1 Incident Reports*. 2012. URL: http://senseable.mit.edu/bos311/ (visited on 04/23/2017).

[Har16] Marc Harter. *Ogre - ogr2ogr web client*. 2016. URL: https://github.com/wavded/ogre (visited on 05/01/2017).

[Lar11a] Robert S. Laramee. "Bob's Concise Introduction to Doxygen". In: (2011), pp. 1–6. URL: http://cs.swan.ac.uk/%7B~%7Dcsbob/teaching/laramee07commentConvention.pdf.

[Lar11b] Robert S. Laramee. "Bob's project guidelines: Writing a dissertation for a BSc. in computer science". In: *ITALICS Innovations in Teaching and Learning in Information and Computer Sciences* 10.1 (2011), pp. 43–54. ISSN: 14737507. DOI: 10.11120/ital.2011.10010043.

[Liu17] Yan Liu. *Visualization of Multivariate Data*. 2017. URL: http://people.stat.sc.edu/hansont/stat730/MultivariateDataVisualization.pdf (visited on 04/17/2017).

[LR12] George Cristian Lazaroiu and Mariacristina Roscia. "Definition methodology for the smart cities model". In: *Energy* 47.1 (2012), pp. 326–332. ISSN: 03605442. DOI: 10.1016/j.energy.2012.09.028.

[Map17a] Mapbox. *Mapbox Studio*. 2017. URL: https://www.mapbox.com/studio/ (visited on 04/20/2017).

[Map17b] Mapbox. *point in polygon for Leaflet*. 2017. URL: https://github.com/mapbox/leaflet-pip (visited on 05/01/2017).

[MTS04] Glenford J. Myers, Todd M Thomas, and Corey Sandler. *The Art of Software Testing*. Vol. 1. John Wiley & Sons, 2004, p. 255. ISBN: 0471469122. DOI: 10.1002/stvr.321. arXiv: 9809069v1 [arXiv:gr-qc]. URL: http://www.worldcat.org/oclc/65982801.

[NYC] NYC Taxi & Limousine Commission. *NYC Taxi & Limousine Commission - Taxicab Passenger Enhancements Project (TPEP) Archive*. URL: http://www.nyc.gov/html/tlc/html/industry/taxicab%7B%5C_%7Dserv%7B%5C_%7Denh%7B%5C_%7Darchive.shtml (visited on 04/24/2017).

[NYC15] NYC Taxi & Limousine Commission. *Data Dictionary - SHL Trip Records*. 2015. URL: http://www.nyc.gov/html/tlc/html/about/trip%7B%5C_%7Drecord%7B%5C_%7Ddata.shtml. (visited on 04/25/2017).

[NYC17] NYC Taxi & Limousine Commission. *NYC Taxi & Limousine Commission - Trip Record Data*. 2017. URL: http://www.nyc.gov/html/tlc/html/about/trip%7B%5C_%7Drecord%7B%5C_%7Ddata.shtml (visited on 04/24/2017).

[QUn] QUnit. *QUnit: A JavaScript Unit Testing framework*. URL: https://qunitjs.com/ (visited on 04/29/2017).

[Rat+16a] Carlo Ratti et al. *MONITOUR*. 2016. URL: http://senseable.mit.edu/monitour-app/ (visited on 04/23/2017).

[Rat+16b] Carlo Ratti et al. *MONITOUR*. 2016. URL: http://senseable.mit.edu/monitour-app/%7B%5C#%7Dwa551396 (visited on 04/23/2017).

[Rod17] Kerry Rodden. *Sequences sunburst - bl.ocks.org*. 2017. URL: https://bl.ocks.org/kerryrodden/7090426 (visited on 04/25/2017).

[Rou10] Vlad Roubtsov. *EMMA: a free Java code coverage tool*. 2010. URL: http://emma.sourceforge.net/ (visited on 04/29/2017).

[Uni14] Population Department United Nations, Department of Economic and Social Affairs. *World Urbanization Prospects*. Tech. rep. 9. 2014, p. 32. DOI: 10.4054/DemRes.2005.12.9. arXiv: arXiv:1011.1669v3. URL: https://esa.un.org/unpd/wup/publications/files/wup2014-highlights.Pdf%20http://esa.un.org/unpd/wup/Highlights/WUP2014-Highlights.pdf.

[Vli07] Hans Van Vliet. *Software Engineering: Principles and Practice*. Wiley, 2007.

[WGK10] MO Ward, G Grinstein, and D Keim. *Interactive Data Visualization: Foundations, Techniques, and Applications*. 2010, pp. 1–5. ISBN: 9781482257380.