Project Proposal

Douglas Garcia Torres (1036847)
M. Bilal Zahid (1035291)
Pradyumna Majumder (1036151)

14-02-2017

Contents

1 Global description

This open data project will be focused on giving any user the tools to analyze and get insights from traffic violation information. In order to develop the project, we selected the Traffic Violations dataset of the Montgomery County of Maryland, USA, to use it as a model for the structure of the data.

This dataset can be obtained in the U.S. Government's Open Data Portal at catalog.data.gov. It contains traffic violation information from all electronic traffic violations issued in the specified County. The information in this dataset includes:

- Date and time of stop
- Location information (state, district, latitude, longitude)
- Description of the violation
- Arrest type
- Vehicle information (type, make, model, color)
- Driver information (race, gender, city, state)
- Other boolean fields (accident, belt use, injuries involved, alcohol consumption, etc.)

It is relevant to notice that any information that can be used to uniquely identify the vehicle, the vehicle owner or the officer issuing the violation is not specified.

We expect that our resulting software will help to analyze large amounts of data through easy-to read charts and graphs. For this, we have selected the D3.js technology that will allow us to bind the data to a Document Object Model (DOM), and then apply data-driven transformations to this document.

2 Specifications

2.1 Functionality

We consider that we are going to add more functionalities as the project will proceed, but for the time being; following are the basic functionalities that will be including:

Map charts: as our dataset has geolocalization data (latitude and longitude), we will be able to populate spatial data showing the figures (e.g. number of traffic violations) of every location (i.e. state, city, country).

Bubble charts: encode data in the area of circles. Although less perceptually-accurate than bar charts, they can pack hundreds of values into a small space.

Enclosure diagrams: use containment to represent the hierarchy. Although circle packing is not as space-efficient as a treemap, it better reveals the hierarchy.

Sankey diagrams: visualize the magnitude of flow between nodes in a network. The fully automatic layout is convenient for rapid visualization.

Time-series: is offered in an easy to use format - just give it the class name of the container in which you would like it to render, some data, and whether or not to use the brush. Each datapoint is an object so you can easily add your own fields and extend functionality.

Trend chart: shows loading times across releases of a hypothetical web application. It is a mixture of a line chart and an area chart, with the daily median loading time indicated by the line and percentile ranges indicated by the surrounding areas.

Time Data availability chart: this chart allows a quick insight into which periods of time a time-dependent dataset covers. It is visually similar to a Gantt chart and allows easy identification of missing pieces and gaps in large datasets.

Grouped Horizontal Bar Chart: this is a horizontally grouped bar chart. It allows you to easily see the difference between different series of data.

Time series with brush: time series is offered in an easy to use format. We just need to give it the class name of the container in which you would like it to render, some data, and whether or not to use the brush. Each datapoint is an object so you can easily add your own fields and extend functionality.

Treemap chart: A treemap recursively subdivides area into rectangles; the area of any node in the tree corresponds to its value.

Data Comparison: we think a very useful functionality to get insights from the data would be to divide the visualization space into various panels with the possibility to select different form of visualizations and different sub-datasets (through filtering) facilitating a comparative analysis.

2.2 Interaction

There will be various use-case scenarios, through user will be able to interact with our tool. We will provide several user friendly controllers for all the implemented functionalities.

Select visualization: the user will be able to select the type of visualization (from a list of available options) that is more suitable for the data analysis he/she wants to perform.

Hovering: the user could check brief tooltip information about any part of the visualization by hovering over with the mouse.

Drill-down: into detailed information by clicking on any part of the visual components with additional details available. After a left mouse click, a new panel will populate with the details.

Filtering: we will provide the visualization tool with a panel with filtering controls by any of the available attributes for allowing the user to perform data analysis at any possible level.

Drag and Drop: we will evaluate the flexibility and feasibility of implementing drag and drop features with D3.js, because we consider it as feature that would give high usability to our visualization.

2.3 Presentation

For presentation purposes, we will publish the visualization tool online. The hosting options we have considered so far are the following:

- Google App Engine
- GitHub Pages pages.github.com (free HTML/JavaScript website hosting)
- eatj.com (free web hosting solution)

In addition, we will generate a user manual and a screencast demo explaining all the possible functionalities that can be performed with the tool.

3 Planning

We plan to develop this project separating the design/planification tasks from the execution tasks. This way we will have a clearer defined picture of what we want to implement and therefore execute it faster.

After getting the feedback on this proposal, we plan to spend an adequate amount of time defining the solutions we want to implement. In that point we will have a clearer idea of what is feasible to execute with our knowledge of D3.js. Immediately we will focus on implementing a prototype of our visualization tool to be able to present in 1st demo presentation. Then, we plan to invest time on adjustments from the feedback we get in order to deliver the Concept Version report.

Further we will continue implementing the functionalities as gradually we will explore our dataset. After the planned review of March 23, we will make the required adjustments, prepare a screencast and a presentation for the March's 28 demo day. Finally, we will perform the final testing, create the required user manual and the final report. The details on these tasks and the planned timing can be seen below.

Task name	Start date	Duration day	Progress	Assigned
Total Estimate	*13/02/2017 00:0*	*53.00*		
Visualization Development	**13/02/2017 00:0**	**53.00**	**0%**	
Proposal	13/02/2017 00:0	2.00	15%	
Select topic	13/02/2017 00:0	1.00	31%	ALL
Write proposal	14/02/2017 00:0	1.00	0%	ALL
Evaluation of proposal	15/02/2017 00:0	6.00	0%	Andrei Jalba
D3.js tutorials	15/02/2017 00:0	6.00	0%	ALL
Submission I	21/02/2017 00:0	22.00	0%	
Design	21/02/2017 00:0	3.00	0%	ALL
Development	24/02/2017 00:0	12.00	0%	ALL
Presentation	08/03/2017 00:0	2.00	0%	Douglas Garcia Tc
Adjustments	10/03/2017 00:0	4.00	0%	ALL
Report: Concept Version	13/03/2017 00:0	2.00	0%	ALL
Submission II	15/03/2017 00:0	23.00	0%	
Development	15/03/2017 00:0	8.00	0%	ALL
Review	23/03/2017 00:0	1.00	0%	Andrei Jalba
Adjustments	24/03/2017 00:0	2.00	0%	ALL
Screencast	26/03/2017 00:0	1.00	0%	ALL
Presentation	27/03/2017 00:0	1.00	0%	Douglas Garcia Tc
Demo	28/03/2017 00:0	1.00	0%	ALL
Tests	29/03/2017 00:0	3.00	0%	ALL
Manual	01/04/2017 00:0	3.00	0%	ALL
Final Report	04/04/2017 00:0	3.00	0%	ALL