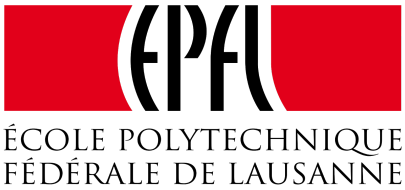


Colorful Traces

Process Book

By Alexandre Dumur and Raphael Strebel

Summary

Abstract

In the following we present the process book, a weekly journal that describes the step-by-step evolution of our project, along with issues encountered and screenshots. This work is done by two EPFL students, Raphael Strebel and Alexandre Dumur, both master students in Communication Systems. This project is in the field of *Data Visualization* a course given by Dr. Kirell Benzi, whose goal is to teach us to make interactive data visualization systems (see [1] for more information).

Introduction

The project we have chosen is called *Colorful Traces* and is provided by the *Smood* affiliation [2], a food delivery company in Switzerland. This company provides us with a dataset containing, amongst others, GPS traces in the region of Lausanne. The goal of the project is to find and visualize patterns or global trends in the dataset. The reason we chose this task is because we find it interesting to work with networks in a visualization. Networks provide many features to invent and explore, and the only limit is our imagination.

Initial Objective

In this section we present the objective as we see it today. As this is only the first week of the project and we do not yet have access to the dataset, it is only a first view on its final draw.
We would like to present the dataset of the *Smood* affiliation as a map of Lausanne, where one can see the most and least used roads in this network. The user could select one particular departure point, or select a departure area, and see the commonly used routes from there. A user would also have the possibility to select an arrival radius, or arrival area, and see the commonly used routes leading to this area, as well as the departure points from where it came.
It would also be interesting to see some statistics on the roads used. When clicking on a specific road, you would have a window showing some statistics on it, for example how many trips this particular road is used in an average week, or in a period of the year, as well as the distribution of the trip duration in a week, month or year. Then we could answer some questions such as: are more people ordering food in the winter or in the summer? In the week or on weekends? Do the routes used by the drivers change with time?
Another setting that the user would be able to set is the coarse attribute of the road according to its passage frequency (this would be dependent of the GPS data provided).
We would like to add an overlay, showing a "heatmap" where the most frequented areas, the most frequent drop-offs and pick-ups are highlighted.
Finally it would be useful to derive some global statistics on the drivers, for example their average duration trip, as well as some insight on how much food the drivers are able to carry per trip.
We would also like to add the feature of selecting which roads you see on the map, for example you could choose to see only roads that are used in 5% of trips or more.
We could also implement a feature where the user selects a departure point and an arrival point and we show a ranking of the roads used (if

any) in the network according to selectable settings (trip duration or trip distance).
As we imagine it, a user would arrive on the website and see a normal map of Lausanne, where he will be able to have fun with the settings described above.
Of course, all the above features depend on the given dataset, which is not yet provided. We have thought of many features that we would like to implement, however realising all of them would be too much for the user, and might complicate the usability. Therefore we will have to focus on a subset of the described features.

Weekly Journal

Week 1 : 16 October - 23 October

This week we had to choose one amongst a long list of very interesting projects. We read them all and discussed for some time before agreeing on *Smood*'s "Colorful Traces" project. We chose that one because, although we are only beginners in Javascript, DOM3, HTML, css and all other tools this course teaches us, we found it interesting to work with maps and especially networks, which we think is a captivating way to tie it all together.
After choosing the project we set up our objectives, which we described in the last section, and finally we started this process book.
The main issue this week is that we do not have access to the dataset. However we already looked at ways to implement a map in Javascript and found that, not surprisingly, Google has an API for maps in Javascript with detailed documentation [3].
We started by doing the basic setup for the project on github, so simply three files, one for the website (.html), the second for the javascript and the last for the website styles.

Week 2 : 23 October - 30 October

Having no access to the dataset, we sent an email to the supervisor of the project and are now waiting for an answer. In the meantime we started to have a closer look at the possible options to implement a map of the region of Lausanne. We excluded the Google API as we found out it has recently become payable so we had to find another way. Thankfully there is a widely used open source library for interactive maps called *Leaflet* (see [4] for the official website). After a few tutorials and youtube videos, we have a world map in our local html file (see *figure 1* below).

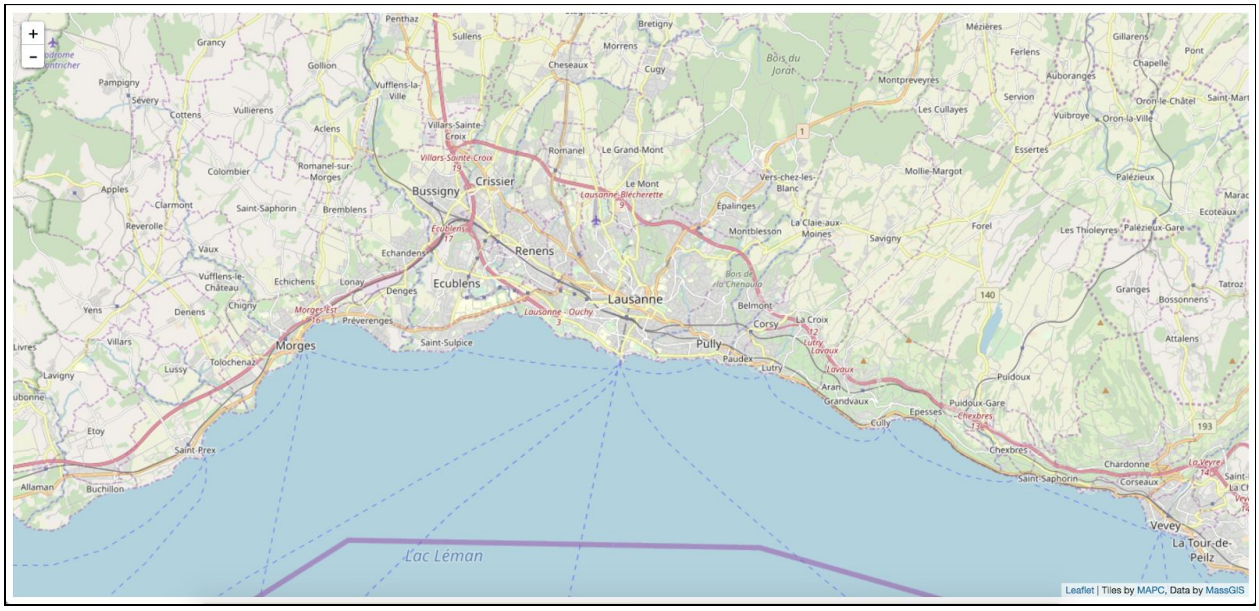


Figure 1

In the meantime, our project supervisor has answered our email and we have set a meeting for next week. We will then have access to the dataset and can start working on the features we set as objective.

Week 3 : 31 October - 7 November

This week started by meeting with the project supervisor to have more information about the project. It was quiet brief and we didn't gain much information. He also mentioned that it would be better to not use map and think about the data as a network. Which doesn't really match our previous thoughts regarding how to carry on this project.

Later this week we received the data set, and did some statistical analysis on it to have a better understanding of what we are dealing with.
For this purpose we created a jupyter notebook and used the pandas library on python. This is a useful tool as it provides instinctive and in-depth ways to handle data in a so-called dataframe.

Here is our first global view of the data :

One data point contains of 6 attributes standing for:

Plat : Pickup latitude
Plng : Pickup longitude
Dlat : Drop off latitude
Dlng : Drop off longitude
t : time. The time of the hour
Road : List of Open streetmap node id. The road that was taken by the driver

The data contains 2000 entries, which is much less than we expected. We will have to adapt our initial goals to this restriction since it will be more difficult to come up with strong statistical results on little amount of data.

In *figure 2* below we show some statistics on the data :

	plat	plng	dlat	dlng	LengthRoad	pnode	dnode
count	2000.000000	2000.000000	2000.000000	2000.000000	2000.000000	2.000000e+03	2.000000e+03
mean	46.520077	6.633191	46.534785	6.644947	210.462500	2.089708e+09	1.792157e+09
std	0.003142	0.005639	0.018449	0.031695	105.143822	1.506250e+09	1.541335e+09
min	46.507833	6.581138	46.487959	6.489375	12.000000	5.040500e+05	2.806610e+05
25%	46.518914	6.630342	46.519167	6.625845	123.000000	5.630213e+08	5.615095e+08
50%	46.520649	6.635173	46.529105	6.639434	190.500000	2.223092e+09	6.224273e+08
75%	46.520649	6.635173	46.558792	6.680104	316.000000	2.223092e+09	3.357409e+09
max	46.536840	6.652842	46.618108	6.741719	651.000000	5.542663e+09	5.623968e+09

Figure 2

By taking the four geographic points $(minlat, minlng)$, $(minlat, maxlng)$, $(maxlat, minlng)$ and $(maxlat, maxlng)$ we are able to define the geographic boundaries of our map.

The geographical boundaries are hence defined by :
$(46.487959, 6.489375), (46.487959, 6.741719), (46.618108, 6.489375), (46.618108, 6.741719)$

Week 4 : 8 November - 15 November

The geographical boundaries are highlighted in *Figure 3*:

Figure 3

So we are able to set the initial view of the map of our website which will be the one above. (If keeping our project on a map).

If we look at the time attribute, we see that it is of the form hh:mm:00 and that it is rounded to the nearest 15 seconds. Furthermore, as the Smood deliveries are done by foot, by bicycle and by car and since we do not have this information in the dataset, the time attribute is not very precise. It might, however, be useful to give an idea of the path duration, for example we could highlight the paths by taking into account it's duration.

Week 5 : 16 November - 23 November

After a more thorough analysis of the data, there is no correlation between the length of the roads and the time needed to arrive at the destination. Also, the nodes at the beginning of a road and the nodes at the end are more closely located on the openstreetmap, so grouping them might be a good idea. However we don't want to take any rash decisions by suppressing data that might be useful later.

After analyzing the dataset and deriving statistics on it in the previous weeks, we have come to the matter of whether or not we want to use a map.
On the one hand a map is useful, especially with a low amount of data. Furthermore the data consists of geographical points, so a map comes as the most natural answer for a visualization of this dataset.
On the other hand, it would be interesting to see what we can show on the data without the use of a map.
After discussion, we have decided that we want to give a user both options : keep the map hidden but have the option to see a node on a map (by clicking on it for example).
By default, the idea is to have a network of nodes (displayed by their geographical localizations but not on a map), and the option to see them connected through edges when they are connected in the dataset. We would also give the option to show only pickup or dropoff nodes.
By just displaying the nodes as described we lose the number of times a node is used, which is crucial in our vision of the project. To still be able to keep this information, we want to color the nodes according to number of times they are present in the dataset. That way we have both the network of nodes and a colored and instinctive visualization of which areas are the most used and which are not.

In the following pictures we show the design worksheets we have filled:

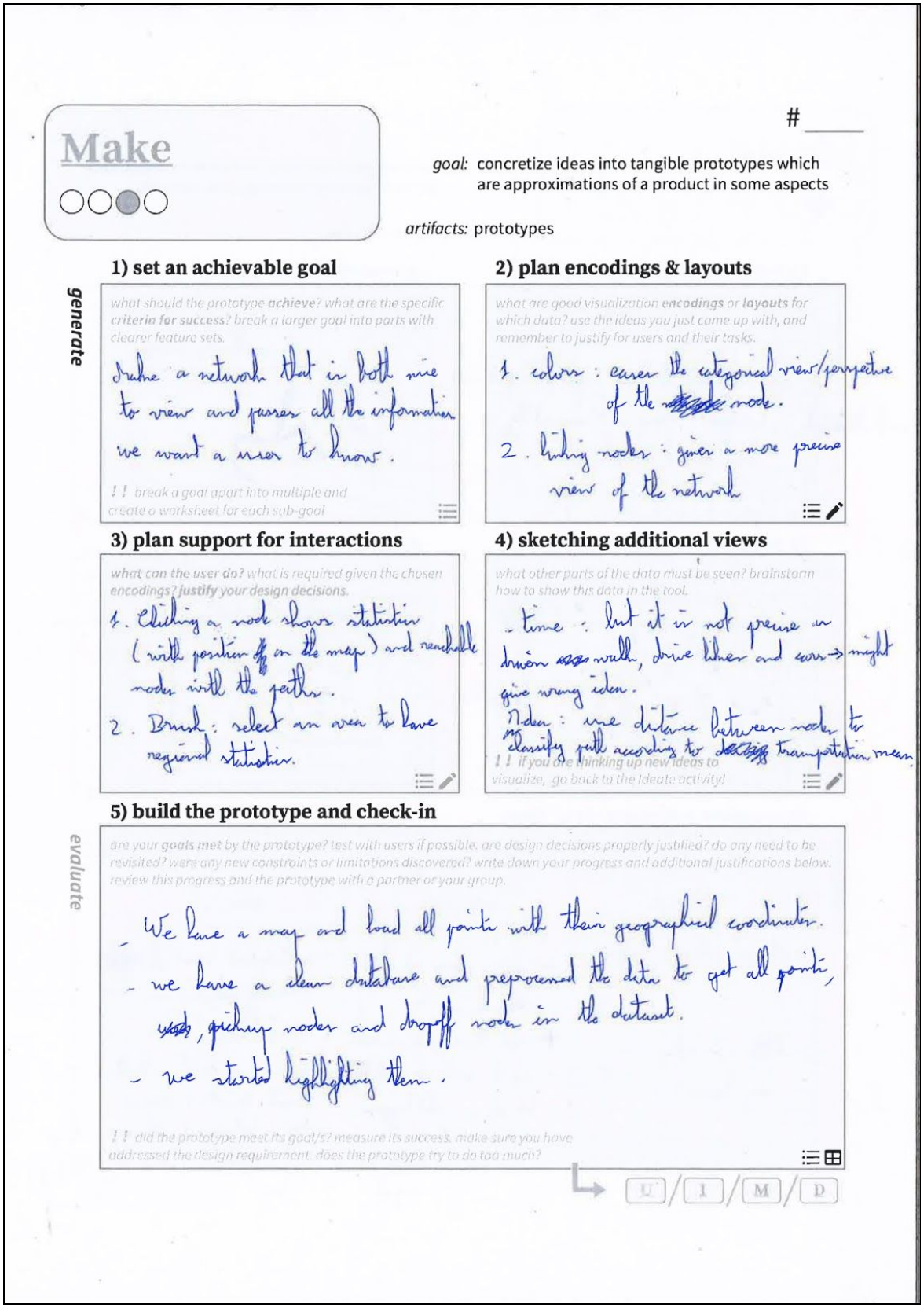

Make
#
goal: concretize ideas into tangible prototypes which are approximations of a product in some aspects
artifacts: prototypes
generate
1) set an achievable goal
what should the prototype achieve? what are the specific criterion for success? break a larger goal into parts with clearer feature sets.
Make a network that is both nice to view and passes all the information we want a user to know.
!! break a goal apart into multiple and create a worksheet for each sub-goal
2) plan encodings & layouts
what are good visualization encodings or layouts for which data? use the ideas you just came up with, and remember to justify for users and their tasks.
1. colors: easier the categorical view/perspective of the node.
2. linking nodes: gives a more precise view of the network
3) plan support for interactions
what can the user do? what is required given the chosen encodings? justify your design decisions.
1. Clicking a node shows statistics (with position on the map) and reachable nodes with the path.
2. Brush: select an area to have regional statistics.
4) sketching additional views
what other parts of the data must be seen? brainstorm how to show this data in the tool.
- time: but it is not precise as driven walk, drive biker and cars → might give wrong idea.
Idea: use distance between nodes to classify path according to transportation mean.
!! if you are thinking up new ideas to visualize, go back to the Ideate activity!
5) build the prototype and check-in
evaluate
are your goals met by the prototype? test with users if possible. are design decisions properly justified? do any need to be revisited? were any new constraints or limitations discovered? write down your progress and additional justifications below. review this progress and the prototype with a partner or your group.
- We have a map and load all points with their geographical coordinates.
- we have a clean database and preprocessed the data to get all points, pickup nodes and dropoff node in the dataset.
- we started highlighting them.
!! did the prototype meet its goal/s? measure its success. make sure you have addressed the design requirement. does the prototype try to do too much?
U / I / M / D

Figure 4

Ideate

#____

goal: generate good concepts and ideas for supporting some of the project's design requirements

artifacts: ideas & sketches

generate

1) select a design requirement

how might we address the challenge using the requirement? which questions would a user ask? revisit this worksheet for each important design requirement.

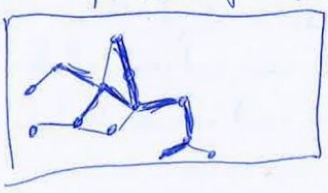
The ability to see specific pattern with Network without being influenced by a background map.

!! revisit this worksheet for all important design requirements for your project

2) sketch first idea

*show how to address this requirement using an **informal sketch** - focus on the big idea not the details.*

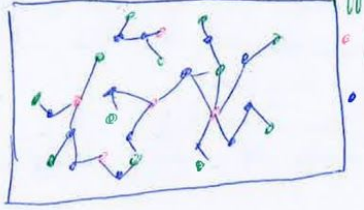
path width by usage



3) sketch another idea

*try another **sketch**, think of a new perspective, be different, do not build off of your previous sketch.*

dropoff
pickup
path



4) sketch a final idea

*think of a different abstraction. challenge constraints and assumptions to **draw** something new or surprising.*

medium used nodes
very used nodes

!! is three enough? not always. have other ideas? fill out another worksheet!

evaluate

5) compare and relate your ideas

*for each sketch, break apart **what works well (+)** and **what doesn't (-)** in the **table** below. make connections, reflect on best parts. can you **combine ideas**? review the table with a partner or group.*

sketch #1	sketch #2	sketch #3
+ shows network traffic - little information - very basic	+ nice view of the network + see the complete network + can be combined with #2 - too dense, what happens if a node is pickup and dropoff ?	+ instinctive view of node density in an area + readability + can be combined with #2 if we decide to show edges - don't differentiate between pickup and dropoff (not nice if it is in bad)

!! combining ideas and sketches is not easy, sometimes it may open up new possibilities and ideas - guess what, ideate again!

U / I / M

Figure 5

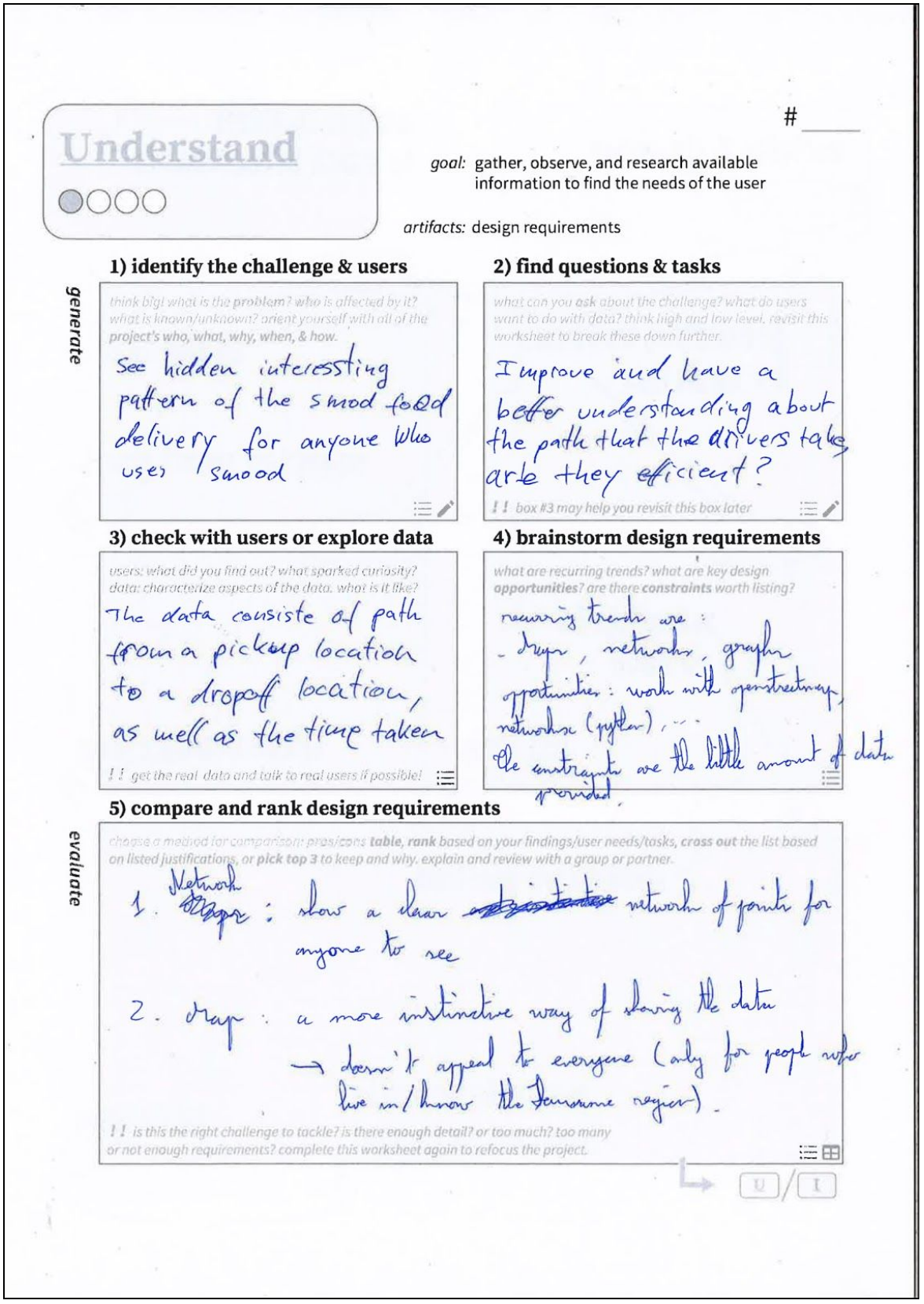

Understand
#
goal: gather, observe, and research available information to find the needs of the user
artifacts: design requirements
generate
1) identify the challenge & users
think big! what is the problem? who is affected by it? what is known/unknown? orient yourself with all of the project's who, what, why, when, & how.
See hidden interessting pattern of the smod food delivery for anyone who uses smood
2) find questions & tasks
what can you ask about the challenge? what do users want to do with data? think high and low level. revisit this worksheet to break these down further.
Improve and have a better understanding about the path that the drivers take, are they efficient?
!! box #3 may help you revisit this box later
3) check with users or explore data
users: what did you find out? what sparked curiosity? data: characterize aspects of the data. what is it like?
The data consiste of path from a pickup location to a dropoff location, as well as the time taken
!! get the real data and talk to real users if possible!
4) brainstorm design requirements
what are recurring trends? what are key design opportunities? are there constraints worth listing?
recurring trends are :
- map, networks, graphs
opportunities : work with openstreetmap, networkx (python), ...
the contraints are the little amount of data provided.
5) compare and rank design requirements
evaluate
choose a method for comparison: pros/cons table, rank based on your findings/user needs/tasks, cross out the list based on listed justifications, or pick top 3 to keep and why. explain and review with a group or partner.
1. Network Map : show a clear networks of points for anyone to see
2. Map : a more instinctive way of showing the data
→ doesn't appeal to everyone (only for people who live in/know the Vancouver region).
!! is this the right challenge to tackle? is there enough detail? or too much? too many or not enough requirements? complete this worksheet again to refocus the project.

Figure 6

The following picture shows all our sketched ideas for the visualization until now :

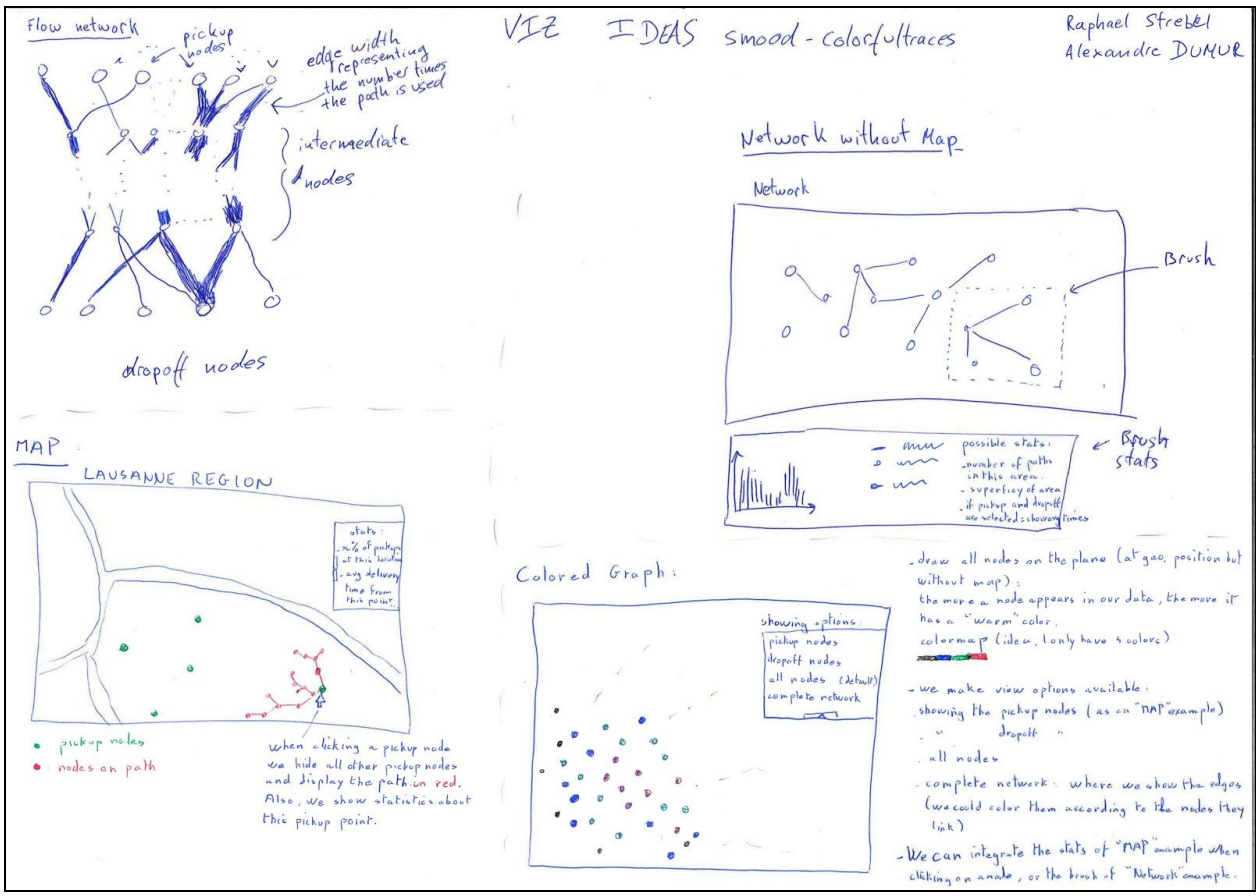


Figure 7

Though after discussing the options, we will drop the “Flow Network” idea as there are too many dropoff points (more than 500) and the paths may well cross each other multiple times, which renders the visualization very hard to understand.
We have decided to focus on the “Colored Graph” visualization but keeping some ideas of the “Map” and the “Network Without Map” options. For example when clicking on a node we could show it’s statistics on the side and show it’s localization on a map. We could also “brush” an area of the network and show area regional statistics.

We have chosen the “Colored Graph” idea because we think it reflects much of what the course teaches us, both on graphs, color maps and maps. Of course we might not be able to load all 12’600 nodes, but we could display the pickup and dropoff nodes and maybe some other nodes if needed.

As we have the map and the dataset, the next mission is to highlight the nodes. To do this we need the points as a longitude latitude tuple instead of an OSM node ID. We have found a way to get this information from a node ID, by doing requests to the page “https://api.openstreetmap.org/api/0.6/node/ID” where *ID* is the node ID, we then parse the XML result to find the longitude latitude pair.
However one issue that we came across is that some OSM nodes that are present in the Smood database do not exist in openstreetmap anymore.
For example when searching for node “598114225” by typing :
https://www.openstreetmap.org/node/598114225
We get the following result :

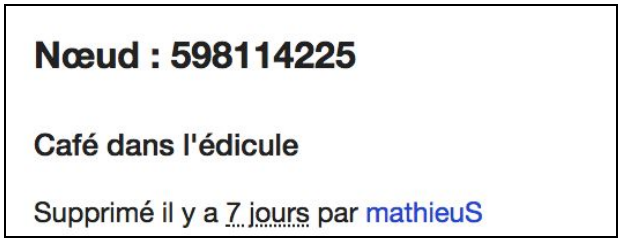


Figure 8

Which shows that this particular node is not present on openstreetmap anymore.
We have multiple nodes that are nonexistent, so we first have to list them before deleting them from our database. Only then will we have a complete mapping from node ID to longitude latitude.

After building the mapping of OSM node ID to longitude latitude tuples we have found 22 nodes that have been deleted from openstreetmap.

We have managed to highlight the pickup and dropoff nodes, but as expected loading 12'600 nodes was too much for the leaflet map. The result can be seen in the image below (the pickup points are highlighted in red and the dropoff nodes are in blue) :

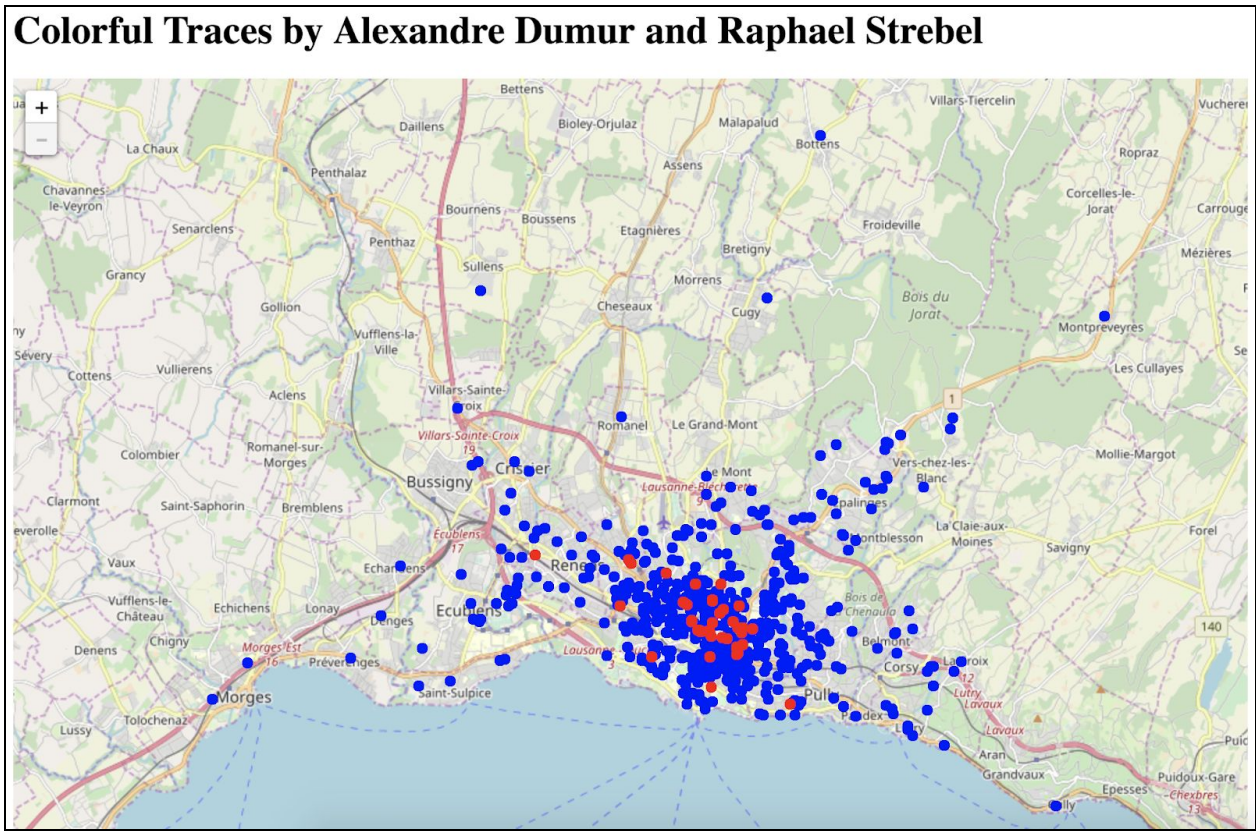


Figure 9

The present viz is of course not finished, what we want now is to have a network of nodes without the map in background. However it is useful to have the map as we will want the option to display the localization of a node on the map (when clicking on it for exemple).

As stated above, the next step is to display the nodes without the map. To do so we can use a Mercator projection since, as seen in class, the Mercator projection does very small distortions on the projection of points when dealing with small areas.

Week 6 : 24 November - 30 November

After much thought we have decided to focus our efforts on the svg. So we will leave the background map behind (for now) and we will draw a network of geographically located nodes without the map.
In the picture below is the result we get when showing all pickup and dropoff nodes, in red and blue respectively. When clicking on the text “pickup”, we show only the pickup nodes, same for “dropoff” and “pickup and dropoff” (we followed the tutorials in [5] and [6]).

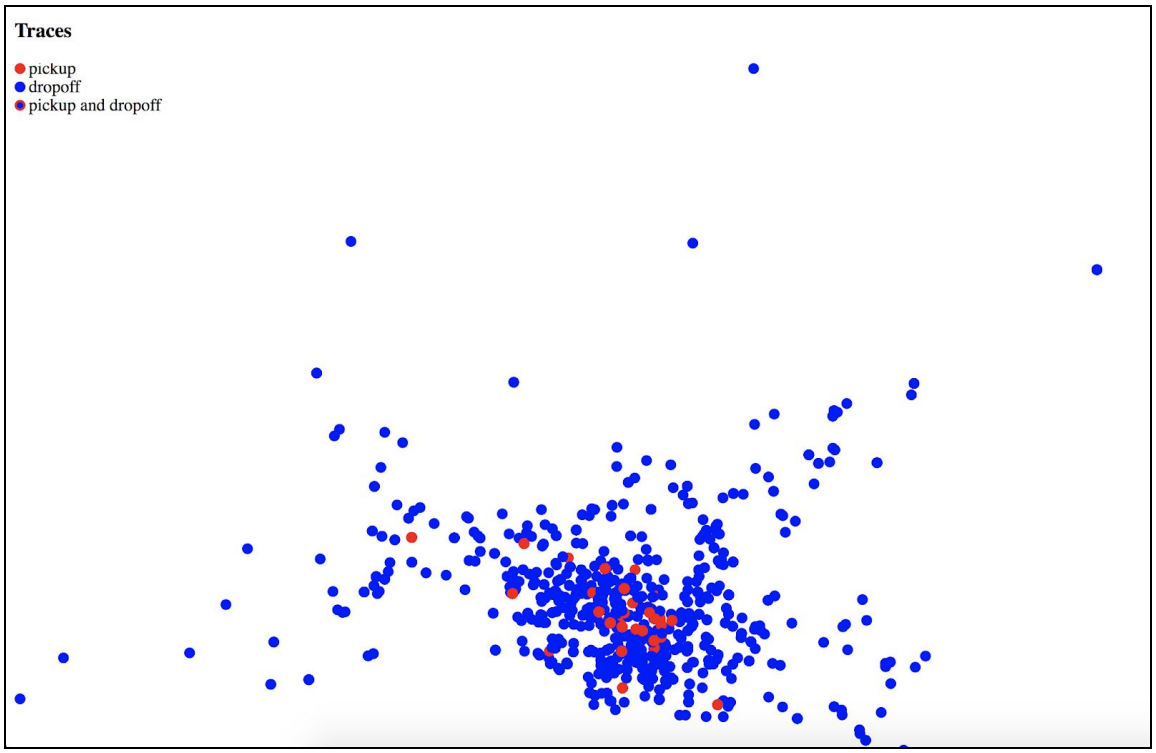


Figure 10

A nice feature we implemented was to have the latitude, longitude pair of a node when hovering over it (see *figure 11*). This was inspired from [7].

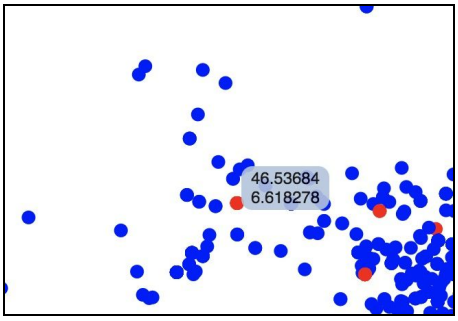


figure 11

The next feature we want to implement is when clicking on a pickup node, we show all paths beginning at this node.

Week 7 : 31 November - 7 December

We have slightly changed our “hovering over a node” feature to depict not the latitude longitude (which we plan to show on a map) but the number of times the node is used as pickup in the case of a pickup node, or a dropoff in the case of a dropoff node. We find this more interesting to visualize.
The harder part of our week was showing paths when clicking on a pickup node but after achieving this we also implemented a nice transition for showing nodes. We thought it would be nice to make pickup and dropoff nodes appear from nowhere, and we made paths expand from the pickup point which was clicked. The result is shown in the figure below. However we had some trouble finding a way to display them according to the path duration.

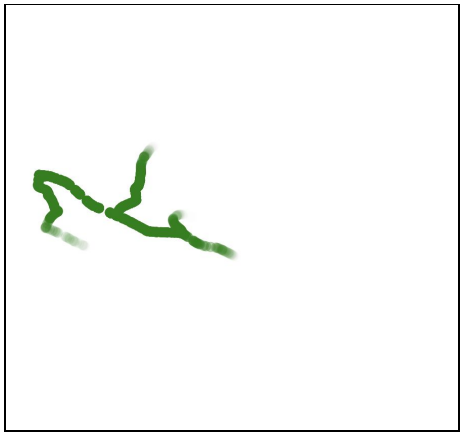


figure 12

Week 8 : 8 December - 15 December

This week we came back to the OSM map. When clicking on a node, we now show it's location on a small map, so that users have an idea of where they are located. This was a necessary option since a user has no idea of what he or she is looking at when opening the website. However we did not want to have the map as background since our goal is to show how nodes are displayed without the distraction of a map behind them.
Some pickup nodes take a lot of time to load their paths, we did a little downsampling (as described in *Week 5*, with nodes that are nonexistent in OSM) but we wanted the network to be as representative as possible by making paths follow "real" roads as closely as they can.

figure 13

We also did the brushing, so now a user can select an area and we zoom into this area to show only the nodes inside it. When double clicking we get a normal view of the network. The map also adapts on the area selected by recentering and setting the appropriate zoom level. This brushing took a lot of time but [10] was a great guide through this process.

Week 9 : 16 December - 23 December

This week is the deadline week, so we have done a lot more than in the last few weeks.
First of all we have decided to make two SVG's to visualize more than just the interactive map. We built the first showing all nodes by coloring them according to their number of occurrences, the result is shown below.

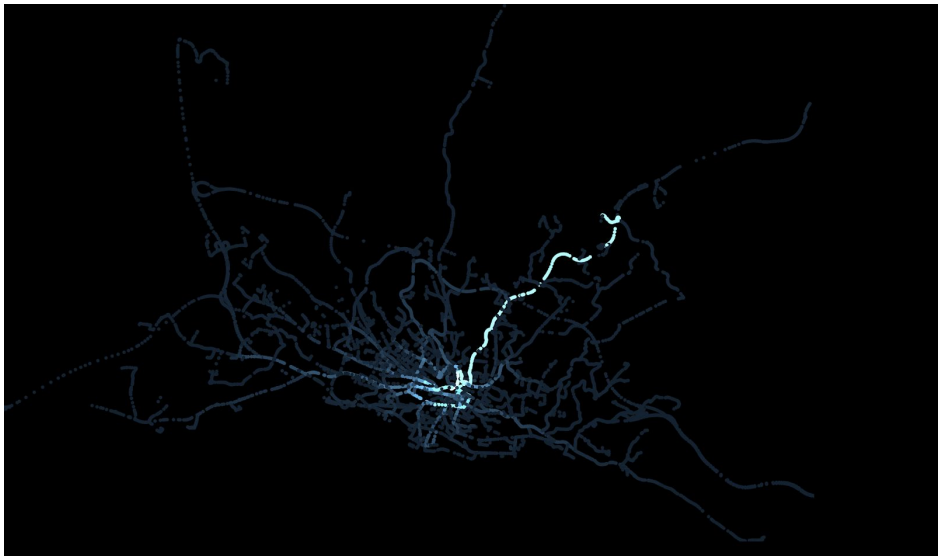


figure 14

The scale used is a linear scale from a dark blue (#112231) for 0 occurrences going through a lighter blue (#3C769D) when the number of occurrences reaches 40 and to white linearly.
As expected there are many nodes used in the center of lausanne, but surprisingly we can also see a path that is used almost one hundred times and heads to the outer region of Lausanne.

We added an option to change the opacity of the nodes in the network of *figure 14* with a “slider”. We used the examples described in [8] and [9] to achieve this. However after further discussion of this option we have decided that it does not add any information to the visualization, and thus shouldn’t be included.

The second visualization network is the interactive map we have been working on since multiple weeks. We made pickup and dropoff nodes appear red and blue even when on a path, so that a user can click on them and display a new path (erasing the last). We also made the network of the interactive map appear only when it is “entering” in the window.
An issue we are facing in this second network is that we had trouble getting the nodes inside a brushing. As the callback function is executed on “end” of the brush, we still have the nodes coordinates just

before rescaling the network. We fixed this by using a “registerListener” and executing the function to get a list of all nodes inside the brushed area only after 1.5 seconds. It is not the most elegant solution and it took us quite some time to achieve this result but it works.
We also made a bar chart to show the number of pickups and the number of dropoffs in the selected area, so that we show at least some information to the user.

We also worked on some additional features that might not be noticed at first glance, specifically when a user resizes the window, the networks adapt to the new window size. This makes the networks nice to see on any screen, as we adapt the size of the network to fill the entire page.

Security is a matter of choice when deploying a system, so we decided to make our website available through HTTPS. We got our intermediate certificate signed by *Let's Encrypt Authority X3 CA*.

In the following images we show the website as it is now.

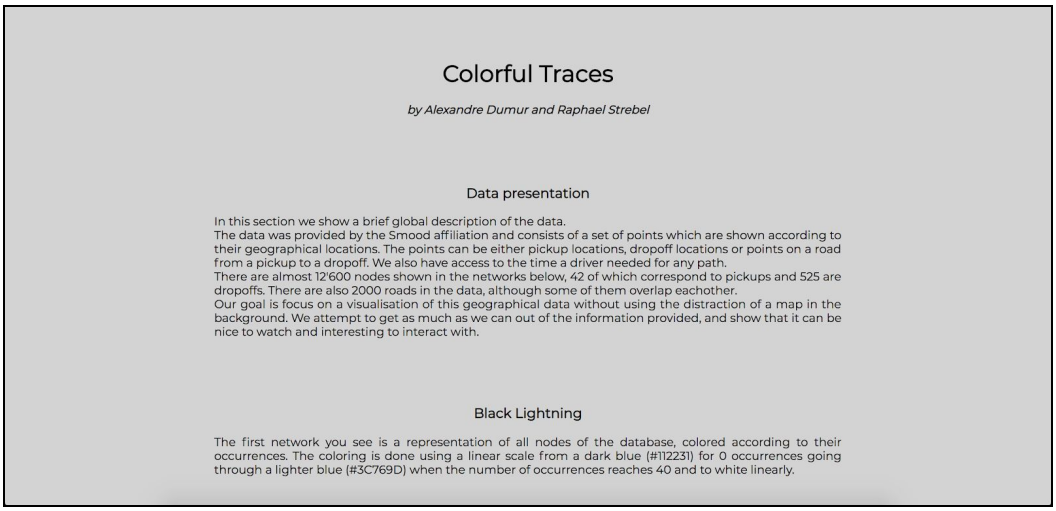


figure 15

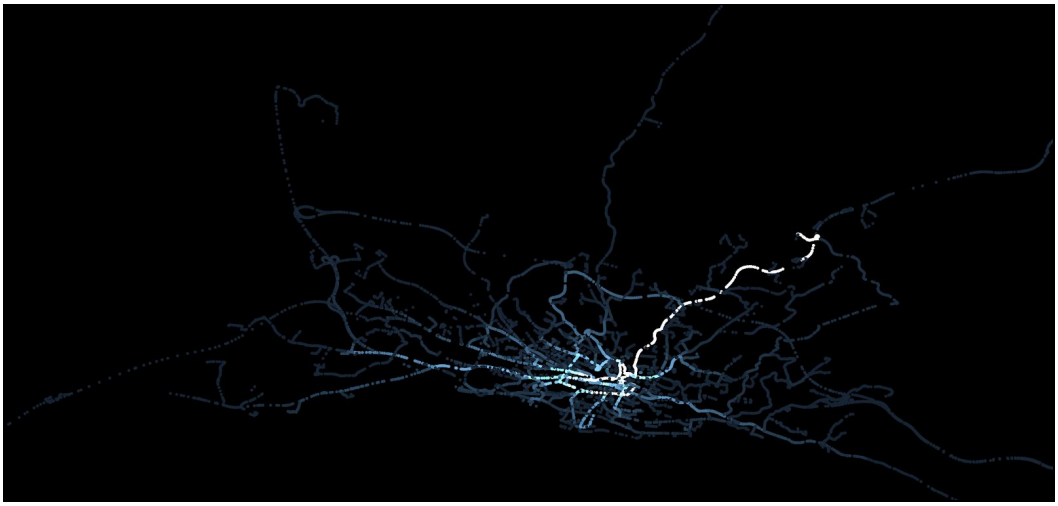


figure 16

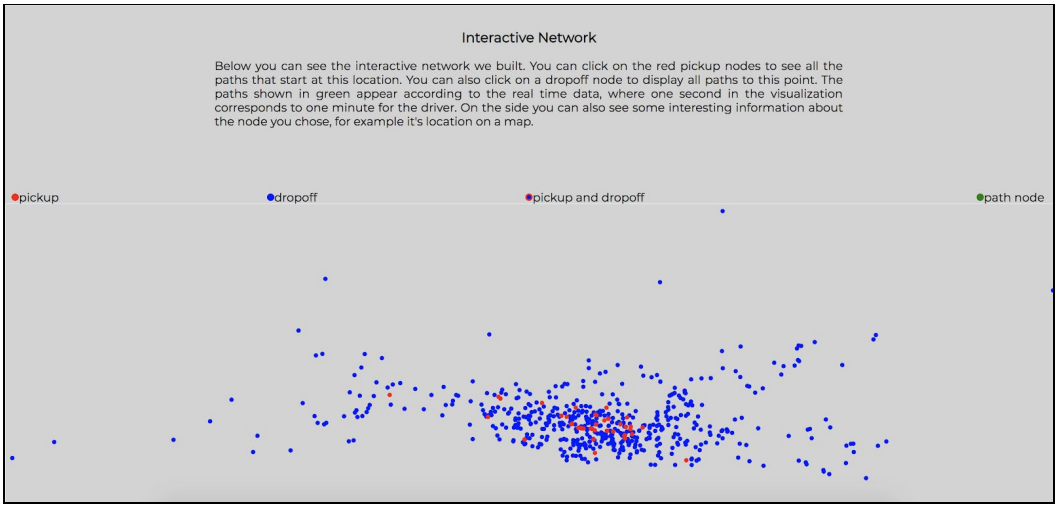


figure 17

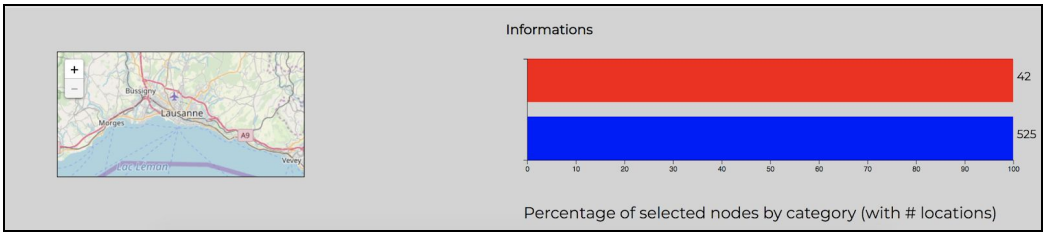


figure 18

Conclusion

Constructing a visualization of a dataset requires a lot of work, especially when it is the first time two students work with javascript (and D3). However it was very rewarding, not only from everything we learned but also by seeing what we built from scratch.

We deviated a lot from our initial project ideas (see *section 3*) for a variety of reasons. First of all because we did not have access to the dataset when we wrote the first section. As the dataset is relatively small (2000 lines), the statistics we initially wanted to show are not so representative and should not be generalized. We also wanted to show the roads arriving and departing an area, but as we went on with our project we left behind the background map and we thought it would be more interesting to show the nodes in a brushed area. We did however implemented many features that we had initially set as objective, for example the heatmap of node occurrences or clicking on a point highlights the paths from or to this point as well as the trip duration.

The main challenges we faced was deciding what we want to show with the dataset, as we both had many ideas and had to make compromises. We also regret that the project is done only from week 5, which leaves little time to work on it, but since we both needed to get some experience with javascript before starting the project it is not that bad.

We mainly enjoyed to see progress so rapidly when working on the visualization. As D3 is instinctive user-friendly and has a big community of programmers willing to teach, a lot can be done in a relatively little amount of time.

As future work, we could better handle pickup nodes that generate a large amount of path. In our current interactive graph those nodes take some time to load their paths. This could be avoided by further cleaning the database, for example putting a “pathID” instead of a path when this path is used in multiple roads. That way we could append the paths instead of drawing the same nodes multiple times.

Bibliography

[1] http://edu.epfl.ch/coursebook/fr/data-visualization-COM-480

[2] https://www.smood.ch/fr/

[3]https://developers.google.com/maps/documentation/javascript/localization

[4] https://leafletjs.com/index.html

[5] http://bl.ocks.org/WilliamQLiu/76ae20060e19bf42d774

[6] http://bl.ocks.org/d3noob/5d621a60e2d1d02086bf

[7] http://bl.ocks.org/d3noob/a22c42db65eb00d4e369

[8] https://bl.ocks.org/officeofjane/47d2b0bfeecfcb41d2212d06d095c763

[9] https://bl.ocks.org/mbostock/6452972

[10]https://bl.ocks.org/mbostock/f48fcdb929a620ed97877e4678ab15e6