

CityBikes

Final Report - Team 5

<http://vickymoser.github.io/citybikes/>

Anreiter Simon*
Computer Science Faculty
University Vienna

Eitenberger Christoph†
Computer Science Faculty
University Vienna

Moser Victoria‡
Computer Science Faculty
University Vienna

Polzer Mikél§
Computer Science Faculty
University Vienna

VU Visualization
University Vienna

1 Abstract

Vienna Citybike is a public bike rental service providing people from Vienna and outside with sustainable and low-priced transportation [5]. In order to guarantee a good quality of service, the rental system depends on regular maintenance, which is best realized by precise planning. In the context of our project for the Visualization course at the University of Vienna, we implemented an interactive tool designed to aid the planning process by providing visualized predictions for the upcoming bike usage based on historic bike usage data. We will first introduce you to the problem and requirements resulting from it (chapters 2 and 3), present the main mock-ups the design is based on and provide a description and explanation of our final design choices (chapter 5), describe a characteristic scenario of use (chapter 7), and finally discuss our result and state the most important lessons learned during the process (chapters 8 and 9).

2 Motivation

The aim of our visualization is to provide historic statistics to employees of the *vienna citybike* organization. We chose this project due to the fact that we have all utilized the service before and found it interesting to analyze the data behind the service. After we analyzed the new data set we came to the conclusion that our ideas are realizable. However, after some consideration, we decided to design our tool for City Bike employees, as we thought that there are already many applications out there designed for City Bike users, and we wanted to be innovative.

Our defined goal was therefore:

Enabling CityBike Employees to forecast and plan ahead by providing them with an interactive and adjustable tool visualizing historic trends and usage. Aiding users to predict the usage for a future time period but also allowing them to adjust certain parameters changing the automatically generated prediction.

2.1 Problem

The Vienna Citybike system consists of 120 bike stations and 1400 bikes [5]. As these bikes are popular and in current use, the rental service requires constant maintenance. City Bike employees in charge with the technical support of

the bikes need to plan ahead in order to be able to come up with fast and good solutions for their task. In order to successfully fulfill this task, the planning should be based on a good estimations about the city bike usage best one that can be influenced and manipulated by the experience of the employees themselves.

2.2 Users and their Tasks

For the sake of presentiveness, we introduce a persona called Matilda¹, who is a technician at *vienna citybike*. Her task is to react to outages of city bike stations as good as possible, which is best done by accurate planning in advance. Our task is to provide her with a tool that helps her do her job. Our application helps her to make predictions about the usage of City Bike stations based on historical data about station usage over time.

2.3 Data

The data this project is driven by contains information about bike usage, rental stations and rental time. The rental station numbers are unique and the nomenclature makes it possible to infer the district in which each station is placed. Accessibility and availability: Part of the Vienna CityBike data is openly available at <http://www.citybikewien.at/>. Some not-open data was provided in the context of the course.

3 Related work

3.1 Other visualization solutions to this problem or similar problems

There is another team in our class that worked with the same data. However, we did not come across any other application designed for employees of *vienna citybike* yet (this might also be due to the fact that such applications are most likely proprietary and thus not freely available).

3.2 Any previous visualization ideas that you incorporated into your solution

We oriented ourselves on the D3 tutorial from class part of the code is embedded in the example code. We also used existing code for the implementation of the calendar for time selection [3] as well as or the map [10], [11].

¹At first, our user was depicted as male. As we are of the opinion that a) language is a mean through which we construct realities and b) women are underrepresented within technical and engineering domains, we decided to talk about a female user instead in order to make women within the field of Technics and Computer Science more visible.

*e-mail:anreiter.simon@gmail.com

†e-mail:a1248272@unet.univie.ac.at

‡e-mail:vmoser@gmail.com

§e-mail:m.polzer@gmx.net

4 Approach

4.1 Description of your visualization design

We developed an interactive tool which allows the user to explore the *vienna citybike* data in terms of adjusting a graph (the prediction line).

The idea was to provide the user with an interface for both the pre-selection of data to be included in the graph, as well as the interactive graph itself. See for a more detailed description in the next section, where we also explain why we made these design choices.

4.2 Mockups

4.3 Reasons for your design choices

Our tool is graphically and functionally divided into two subparts: the filter selection area, where the user can include and exclude data to be analyzed (districts and stations), and the prediction area, where the user gets to actually play with the prediction line and thus can explore the preselected data.

- Why we divided these sections in the first place: By the division, we wanted to avoid a visual overload. However, we are aware of the issues raised by the demand for scrolling (see section Weaknesses). - Why we chose a vertical placement of selection and prediction area: we thought that vertical scrolling is more user-friendly as it is widely used, compared to horizontal scrolling. This assumption is backed up by the following usability guideline:

When scrolling is used, a Web site should be designed to allow the fastest possible scrolling. Users only should have to scroll through a few screenfuls, and not lengthy pages. Designers should never require users to scroll horizontally. [4]

In the following, we provide a more detailed description of these two subparts and a short explanation for the design choices that were made for each of them:

Selection area: The selection area is again divided into two tabs, one for the selection of districts (broad selection) and one for the selection of stations (narrow selection).

- Why one can select for districts: The reason for this is that we assumed that most people living in Vienna for some time have a mental representation of the city in terms of its districts, and thus can easily navigate on a map which is zoned by districts. However, we also added a second way of selecting districts, namely by checkboxes. We implemented this solution as well to be on the safe side (i.e. in case employees are new to Vienna and not yet familiar with a geographical representation of its districts). The checkboxes and the map are connected, i.e. if one clicks on the first district in any of the two selection tools, it will automatically be selected (or de-selected) in the other tool as well. Also, districts are a good way of grouping 120 stations, which otherwise would have to be implemented as a huge list, making the selection task time consuming and frustrating especially for new employees.

- Why one can select for stations: We assumed that for example highly frequented stations are not always necessarily grouped by district, but can cross district boundaries.

Example: Matilda may want to only include those stations near the University of Vienna she can then select the first and the ninth district (the University area), and then exclude stations that are further away. Other examples would

be the analysis of a single station, the comparison of two stations against each other, and so on.

- Why we used tabs: We wanted to avoid the need for more scrolling. By using tabs, we omit one of the two selection options (districts or stations). This is not a problem, since the usage order is district selection station selection, and the selected districts are reflected in the station selection area (see Scenarios of use).

- Choice of colors: we chose light colors for the basic design, and strong colors (red) only for highlighting the selected districts on the map.

Prediction area: This area is where the actual visualization takes place. It contains the graph area displaying the prediction line derived from mathematical functions which the user can choose and adjust by changing the weight of the function parameters, the green historical line (mean station usage), the standard deviation of the historical line (displayed as light green shadow of historical line), and the outliers (blue dots). The user can add functions to adjust the prediction line by selecting the function type (linear, squared or gaussian) and clicking on add new function. Theoretically, she should also be able to save prediction views, but we haven't yet managed to successfully implement this feature into our application.

- Why sliders: We chose to use sliders as input tools for changing the parameters that allow for the adjustment of the prediction line, as well as the bandwidth of the green shadow, as we thought sliders are very intuitive, especially for a generation of smartphone users. But we included the feedback in our evaluation of these input elements for the purpose of adjusting the prediction line (see section Weaknesses).

- Choice of colors: we chose the strongest color (red) for the most important element (the prediction line to be adjusted), as this element conveys the most important information. The rest of the visualization is held in light-weight green and blue colors which are distinguishable from each other without being disharmonic or too strong and thus distracting.

5 Implementation

5.1 Brief description of how the system was implemented (toolkits, languages, platforms)

The webproject is based on HTML/CSS and uses Javascript (JS) as well as JS libraries such as JQuery and D3 [7]. Our repository was administered on GitHub [8].

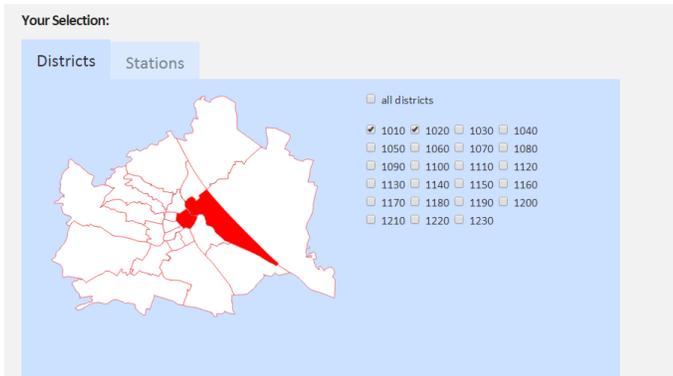
5.2 Any serious implementation challenges you encountered and how you handled them

The biggest challenge was that we all lacked enough experience in JS and especially with D3 in order to freely use these tools for implementing our ideas. We relied on the info and the material from the tutorial from our class [6].

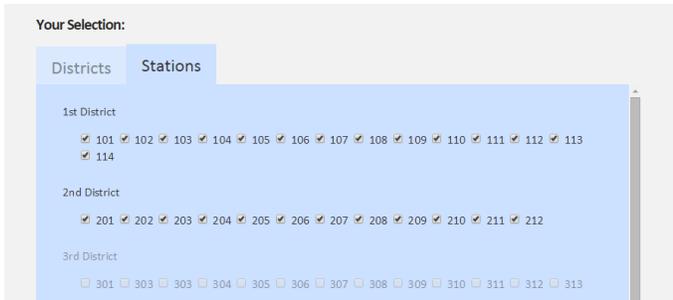
6 Results

6.1 Scenario(s) of use, including screen shots of the system being used

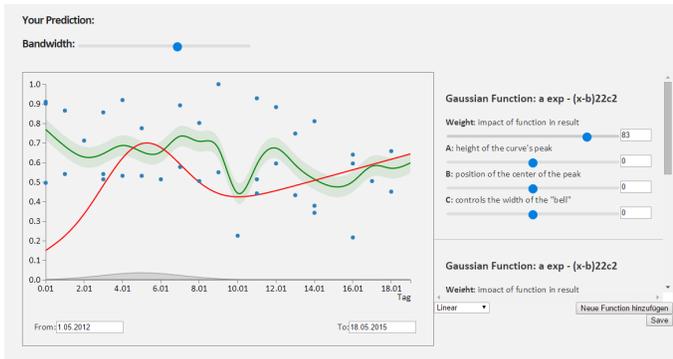
Scenario: Matilda wants to make a prediction for the first and second district. Here is a step-by-step view of what she does:



1. Use Case District Selection.



2. Use Case Station Selection. As you can see, ticked stations depend on ticked districts: 1st and 2nd district are selected in the preceding use case, therefore the stations of 1st and 2nd district are pre-selected in this use case.



3. Use case adjusting prediction line and saving view: adjustment of bandwidth (through slider on top), function parameters (through clicking on the respective function and using sliders) and date (through clicking on date input field). By clicking on save, the current view is stored and displayed at the bottom of the prediction area for further editing/deletion (see next use case).

#	View	Date	Edit
1	Parameter 1: 85, Parameter 2: 45, Parameter 3: 97	26.05.2015	edit delete
2	Parameter 1: 85, Parameter 2: 45, Parameter 3: 97	26.05.2015	edit delete

4. Use Case editing/deleting views: the user can select previous views and continue playing with them. She can also delete saved views, both via clicking on the buttons edit or delete.

6.2 Performance of the system

The tool performs well, we had no issues with delay or overload in data retrieval or display. So far, the application is stable.

6.3 Feedback from the evaluations about the design and functionality of your tool.

We included the feedback in the next section (strengths and weaknesses), as we inferred these aspects of our tool directly from the feedback.

7 Discussion

7.1 Strengths and weaknesses of your approach and implementation

Weaknesses:

- Regarding the division of the tool into two areas: 1. As already mentioned in section XY, our tool is divided into two subparts: the selection area and the prediction area. As explained there, the idea behind this is that this division in positioning on the page (two different areas) also reflects the division in functionality (two different use cases, namely pre-selecting and playing). However, this division lacks of consistency, as the time selection which is also a filter selection and thus should be placed in the filter selection area is currently implemented within the prediction area. This makes it easier for the user to quickly adjust the time selection so the lack of consistency is the price we paid for that, so to say.

2. As suggested in the feedback after our presentation in class, the placement of these two subparts (one above the other) may be suboptimal, as the user may have to scroll up and down in order to be able to make an instant connection between her selection and her prediction (especially for complex selections, such as combinations of stations from various districts).

- Regarding the functionality of the tool: Our tool does convey the idea behind it, but it lacks some of the functionalities we wanted it to have:
- it is not yet possible to actually include selected stations as filters
- it is not yet possible to select only weekdays or weekends
- it is not yet possible to save views

Strengths:

- Our tool is overall clear and easy to use and can thus be said to be generally user-friendly.
- We believe that although direct manipulation of the graph would have been even a better solution for the adjustment of the prediction line, the sliders still do provide a solution to this challenge.

- Our tool is to a certain degree barrier-free for dyslexic people, as the selection of districts is possible on the map and the sliders dont require reading or typing either (unfortunately, this doesn't yet hold for the function selection).
- Possibly the greatest strength of our tool draws upon the popular concept of Gamification [2]: its playfulness. Our tool animates the user to play with it, it challenges her to make the prediction line fit the historic line. It can thus be fun to use it, which really adds to its quality, especially because it is designed to assist employees during their work. This aspect is also based on the feedback we got from colleagues of our class.

8 Lessons you learned

- Things that seem trivial at first require time and evaluation, such as for example the right selection and placement of input tools on the page.

- Things that seem to require little effort in order to be implemented do often take more time than expected, even when it comes to simple functions especially if one lacks experience with the programming language or library.

- An uneven distribution in the skill set of the individual team members is most likely reflected in the division of the tasks within the team, leading to a setting where those with more experience in skills required for the project also tend to do more of the required work. This is, as the two statements above, nothing new to the four of us, as we have already worked on projects with differently skilled people. This is why we also would highly appreciate to have more introductory parts in the tools most likely needed for the course, such as Javascript and D3 (instead of Tableau, which again is not accessible to all of us and thus was excluded as an alternative to JS in our case). For most of us, working on the project was very much about implementing something in a language we are all not so familiar with (but some a bit more than the others), so that the engagement with the visualization design itself faded a bit into the background, which was a pity, as the course should be about exactly that rather than for example dealing with the import of .csv files. Regarding the distribution of work load, we created a table in order to make the division of tasks more transparent for the teachers (see https://docs.google.com/spreadsheets/d/1QXzQFIFPGSghCFK-oI_Td3QLD1hoE1AnZYbLqEKQBKE/edit#gid=0).

References

- [1] Our web project on GitHub, <http://vickymoser.github.io/citybikes/>
- [2] S. Deterding et al., From game design elements to gamefulness: Defining "gamification". In: Proc. 15th Int. Acad. MindTrek Conf., 2011. pp. 915.
- [3] Code and Design Studio, <http://www.codeanddesign-studio.com/2011/03/02/40-free-date-and-time-pickers/>
- [4] Usability Guidelines, <http://guidelines.usability.gov/>
- [5] Vienna CityBike, <http://www.citybikewien.at/>
- [6] The tutorial from the class, see <https://github.com/gabysbrain/D3-Demo>
- [7] D3 JS, <http://d3js.org/>
- [8] Github, <http://github.com/>
- [9] Visualization Analysis and Design: Abstractions, Principles, and Methods, CRC Press, 2014.
- [10] GeoJSON-Daten sterreich, <http://www.strategieanalysen.at/wahlen/geojson/>
- [11] Open Government Wien, <https://open.wien.at/site/datensatz/?id=2ee6b8bf-6292-413c-bb8b-bd22dbb2ad4b>