



Objective

Deliver the final version of your project, plus report and video.

The submission is done via Moodle by just **one element of the group**.

You need to submit **on 16th November until the end of the day (23:59)**.

We will check the date/time.

Projects delivered **late will not be graded (you will fail the course)**.

Requirements/Deliverables

For this final submission you need to prepare three things:

1. **A folder containing your project.** It should be ready to run as-is: just unzip to a local folder, open an HTML file and we're ready to go! It must include all scripts, data files, etc. Any special requirements/instructions should be given in a README.txt file. This cannot include the installation of servers, access to remotely hosted services, etc. Again: I turn off the Internet, unzip, open a local file and everything works. At most, we'll launch a minimal python server to be run as shown in the first d3 tutorial lab class. Also, you should always use d3 version 7. You will be penalized for each different version used within the project.
2. **A .pdf file with your report** according to the instructions provided in this lab guide.
3. **A video file (h264 codec, MP4 format), between 2 and 3 minutes, demonstrating your project.** You have seen several throughout the lectures so you know what is expected but, in a nutshell: the first part should describe the visualization, its different components, and how it works. The second should highlight some use cases and demonstrate its usefulness vis-à-vis the questions you defined in the first checkpoint. You can look at examples of good videos in the course's **Hall of Fame**.

Not following the required formats: ZIP for the file (not 7ZIP, not ARJ, not LHA, ...); MP3/H264 for the video (not MKV, WEBM, WMV, ...) or PDF for the document (not DOCX, PPTX, ODF, ...) will be penalized with at least **one grade point** (for each fault...). However, in the worst case, if we are unable to promptly open the wrong format (we will not be searching for installing codec packs or other software "just for you"), you will get a zero on that component!

Create a zip file with the three things. The name should be:

"VI-<group>.zip".

For instance, Group 5 names the file “**VI-05.zip**”. Upload it to Moodle. For student workers: you don’t have a group number, please use your initials instead (ex: “VI-DG.zip”).

The project will be graded in due time and the grade posted on the page’s course.

Do note: there is an upload size limit on Moodle, which we do not control. Leave the upload to the last minutes at your own peril. If something goes wrong, the obvious fix will be to reencode the video at a different quality setting. If strictly necessary, you can include the URL for a high-quality version of the video in your README.txt file but the Moodle submission needs to be complete, even if you have to include a low-quality version in it.

The Report

For the final report of your InfoVis project, you should write a document using the provided template (**CP5-template.docx**), with no **fewer than 6 and no more than 8 pages long**. You can adapt the sections to what is relevant for your project, but you should at the very least focus on the following things:

1. Introduction

Describe the problem domain you are going to address. Why is it relevant? What questions are unanswered? What is the motivation for addressing it? Don’t current tools already do what you need? What did you expect to gain from your approach? What questions do you propose to allow (from Checkpoint I)? New ones, even?

IMPORTANT: explicitly list the questions you defined in Checkpoint I

The intro should give someone who didn’t know your project enough background to understand it.

Generic text about how nice this was, how it allowed you to overcome interesting challenges, etc. Fluff that I could transplant to any other such report is, most likely, not good. Be concrete. Here, and in the rest of the report!

[NOTE: Part of this comes from what you decided in Checkpoint I]

2. Related Work

What similar works did you find to yours? Where did you get your inspiration? Scientific papers would be great, but at least online references should be here. You should be describing other works that visualize the same problem domain (or similar ones) and explain why they don’t really do what you need.

[NOTE: be honest here. Mentioning something you already saw, even if it is like what you ended up implementing, is not considered a bad thing. You won’t get downgraded because you implemented a similar solution (only if you copied someone else’s code...)]

On the other hand, you WILL be downgraded if there is some obvious previous system (or it is just a google search away) and you neglect to mention it]

3. The Data

Where did you get your data from? What challenges did you face? Did you have to correlate sources? Did you have to clean it up? What data did you think you’d get and ended up not finding? Which compromises did you make? Also, be sure to mention scalability issues. Did you have to filter things out? Aggregate them? Derived measures? Go beyond Checkpoint II: after it, did you need to revisit the data and its format? Make changes? Describe them and why.

[NOTE: We don't need an exhaustive description such as the one in Checkpoint II. We want a narrative, a story of what you had to do and why, not lists of column names, etc. Copying and pasting the things from Checkpoint II here won't work.]

4. Visualization

Be sure to profusely illustrate this section!

4.1 Overall Description

What is your solution? Start with an overview of the system (layout, etc.), how it works (how data can be filtered, selected, etc) and then move on to describing the different visualization techniques, showing how each works and gets the job done.

4.2 Rationale

Why did you think your techniques would work? What visual encodings did you use and why (and why not others)? What alternatives did you consider, even if they turned out not to work? Especially, discuss how you managed the complexity of real data, and matters of scalability. Also, include in your discussion the evolution of the prototype, from the initial sketches to the last version highlighting what you learned from version to version and how that influenced your design.

4.3 Demonstrate the Potential

Describe for at least a couple of cases (from the questions you promised you'd answer before) where, step by step (illustrated with screenshots), you find the answers you seek. In short, demonstrate the potential of your solution! Does your visualization provide insights on data that *you were not expecting / that are not common knowledge*? **These are pure gold!** Be sure to include them!

5. Implementation Details

What challenges did you find and overcome? How did you implement the links between the views (incl. brushing, etc.)? What algorithms did you use? What techniques did you adapt, or implement, from scratch? (instead of just copying & pasting them from the D3 examples page...)

6. Conclusion & Future Work

What did you learn? Were you able to address all the questions? If you were to start over, what would you have done differently? Also, if you now had 1 more month and €3000 to spend on this, what else would you do to enrich your solution?

Penalties

- Use of d3 versions other than version 7: **2 grade points penalty for each different version.**
- Existence of external dependencies: **2 grade points penalty for each dependency.**
- Documents under 6 or over 8 pages long: **1 grade point penalty per missing or extra page.**
- Document template altered (wider margins, smaller font, etc.): **1 grade point penalty.**
- Videos over time: **1 grade point.**
- Not following the required formats: **1 grade point penalty for each wrong format.**
- Zip with project, document and video uploaded after the deadline: **0.5 grade points per hour of delay.**

Grading

Your work will be graded according to the following parameters:/We will have a detailed grading rubric that will focus on the following aspects:

Prototype [60%]

- Completeness

- Coherence
- Usefulness (follows questions set in Checkpoint I, more?)
- Layout & graphic design
- Interactivity
- Wow factor

Video [20%]

- Duration
- Explanation of the Visualizations
- Demonstration of Usefulness
- “Standaloneness” (i.e., if someone just watches the video, no report, no prototype, is the project understandable?)

Report [20%]

- Readability / Understandability
- Structure (adherence to the content sections)
- Content (depth, thoroughness, discussion of alternatives, justification of choices, etc.)