Design Sprints for Online and On-Campus Visualization Courses

Johanna Beyer*
Harvard University

Hanspeter Pfister†
Harvard University

Figure 1: Different steps in a visualization design sprint. From left to right: map (identifying questions and audience), sketch (pen and paper visualization designs), decide, prototype & test, present.

Abstract

We present how we have integrated the concept of *Design Sprints* and project-based learning into teaching a range of introductory visualization courses. Design sprints describe a time-constrained, interdisciplinary process based on rapid prototyping and testing with customers to define goals quickly, validate ideas, and decide on a product roadmap before starting costly development. The well-defined, interactive, and time-constrained design cycle makes design sprints a promising option for teaching project-based courses to increase student engagement and hands-on experience. We have adjusted the original design sprint method for teaching a range of visualization courses. This paper presents a detailed guide on how to incorporate design sprints into both large undergraduate and small professional development online and on-campus courses. Design sprint results, including quantitative and qualitative student feedback, show that design sprints engage students and are useful in practicing and applying visualization and design skills. Finally, we provide design sprint teaching materials, show examples of student-created work, and discuss lessons learned.

1 Introduction

In recent years, innovative pedagogical techniques such as active learning and flipped classrooms have become increasingly popular for teaching university-level courses. Active learning encourages students to do more than just listen to a lecture. They become actively engaged in higher-order thinking tasks, such as analysis, synthesis, and evaluation [5, 18].

In the past five years, we have transformed our large undergraduate visualization course at Harvard University, with more than 200 students, from a lecture-centered course to a learner-centered course [4]. We initially switched to an active learning model of teaching in spring 2016 to make the class more engaging for both students and instructors. We also included a single large team-based capstone project towards the end of the semester. Since fall 2018, we have added design sprints into our visualization courses to formalize how we teach a user-centered design process, teamwork [9, 17], and how to run team-based projects. To date, in summer 2020, we have used design sprints in four courses overall, including our semester-long visualization course CS171 (`www.cs171.org`) and a two-day long visualization workshop for professionals (`www.dataviscourse.org`). In total, we have used design sprints nine times on-campus and six times online for a total of fifteen times in four different settings (CS171 and workshops, both on-campus and online, respectively). Using active learning and a flipped classroom with in-class activities and multi-week team-based projects leaves less time for content coverage. However, our students gain more hands-on-experience and skills to apply their knowledge in a real-world project, leading to a noticeable improvement in the quality of student projects.

Design sprints are time-constrained, interdisciplinary projects that rely heavily on rapid prototyping and testing with customers to quickly define goals and validate ideas [10]. Design sprints originated from Google Ventures and focus on exploring business ideas within one 40-hour workweek. Compared to other strategies for quick prototyping, such as Hackathons, design sprints follow a clearly defined five-step process that starts and ends with users. Design sprints have recently been explored in classroom settings [8], but to our knowledge, we are the first to explore design sprints for visualization courses. We use design sprints in our courses to teach students visualization skills, a structured user-centered design process, and the necessary skills for team-based projects.

In the semester-long courses, we use design sprints for an 8-week long guided project at the beginning of the semester and a 5-week long final capstone project. In the 2-day professional workshops, we use design sprints for one guided project that lasts a few hours for the online course and two days for the on-campus class. In addition to adjusting the length of the original design sprint, which is one 40-hour week, we have also emphasized the visualization components at each step of the process. Furthermore, to scaffold design sprints in our guided projects, we provide project ideas and cleaned datasets instead of having students work with real users [6]. We let students implement their own ideas, including choosing users and finding data, for their 5-week long final projects.

This paper provides a detailed outline of how we conduct design sprints in our visualization courses. We show several examples of how to incorporate design sprints into different course settings. Along the way, we will discuss common pitfalls and present several lessons learned. We will give guidance on how to prepare teaching materials, and we are freely providing our teaching materials on GitHub (`https://github.com/CS171/design_sprint_material`). Overall, we believe that design sprints are useful and suitable for active learning and project-based courses that prioritize the hands-on-learning of visualization skills.

2 Visualization Design Sprints

Our design sprints' learning goals are twofold: First, we want students to practice the skills they have learned in lectures, labs, and design exercises through more substantial projects. Second, we want to teach them how to conduct group projects using a suitable design

*e-mail: jbeyer@g.harvard.edu
†e-mail: pfister@g.harvard.edu

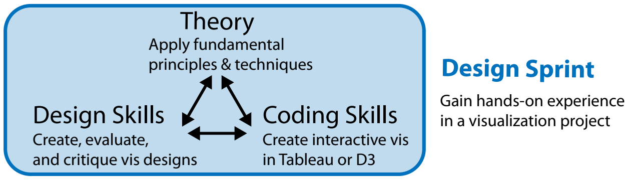

Figure 2: Learning goals. Design sprints allow students to apply their knowledge of visualization theory, design, and coding skills to gain hands-on experience in a visualization project.

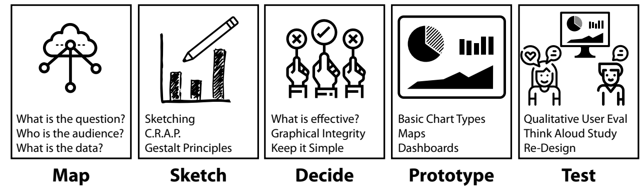

Figure 3: The design sprint process (map, sketch, decide, prototype, test) and the main visualization-related activities in each step.

process for bigger, independent visualization projects. Fig. 2 summarizes our learning goals. The design sprint combines visualization theory, design skills, and coding skills, and allows students to learn by doing. In our semester-long visualization course [1], we run two design sprints per semester. The first sprint is guided and takes place in the last 20 minutes of each lecture during the first weeks of the semester. This allows us to give students direct feedback in class and teaches students the design sprint process. The second sprint takes place towards the end of the semester. Students work on the second sprint mostly outside of class, and hand in the result as a final project. For teaching shorter classes, such as multi-day workshops, we only run a single guided design sprint.

The original design sprint process consists of five distinct steps: Map, Sketch, Decide, Prototype, and Test [10]. In the following, we outline each of these steps and describe how we adjusted them to fit our teaching goals and course settings. Figure 3 gives an overview of our steps and the main visualization activities in each of them. One main adjustment that we added to the design sprint process is that we emphasize the importance of iterations in visualization design. Therefore, we typically give students the option to refine their initial ideas after having received feedback from peers and course staff. Our courses also cover other visualization topics, such as perception and color theory. Here we just focus on the essential elements for visualization design sprints. For a complete overview of CS171 and our workshops, please see the respective websites [1, 2].

Design Sprint Preparation. Before starting a design sprint in any of our courses, we first outline the format and, more importantly, explain the motivation of following the design sprint process. This introduction primes students for their projects. Next, we start with our team formation process. For all of our guided projects, we randomly assign students to teams to create diverse groups of students with different design and coding backgrounds. Additionally, in college courses, we make sure to talk about effective teamwork and have students sign *team expectation agreements* that outline their responsibilities, such as their method of communication, team meeting times, and sharing materials and code. We found that this initial team formation step is crucial for setting up a positive team dynamic and prevents many issues student teams often face. Finally, every team creates a *process book*, a shared digital lab notebook (we use Google docs) that documents every step of the design sprint in great detail. Process books are part of the design sprint deliverables and allow course staff to give direct feedback to all teams on an ongoing basis, which is crucial in online courses.

Map. The first step in a design sprint focuses on mapping out the problem space. Before this step, we introduce the user-centered design process [12, 19] to our students and give them access to the datasets they will be using during the sprint. Furthermore, in this step, we introduce different data types (nominal, ordinal, and quantitative) based on Steven's taxonomy [20]. In their teams, they then focus on three main questions: *What is the data?*, *What is the question?*, and, *Who is the audience?*. In this step, students should get a sense of their data and come up with questions about their data to answer for their audience through visualizations.

Sketch. After getting a clearer picture of the problem space, this step exposes students to the ideation and convergence process by sketching possible visualizations on paper. Students *individually* create sketches of visualization designs that answer questions about the data that they have identified in the previous step. During this step, we also introduce students to sketching, the CRAP design principles [22], Gestalt principles [11], and basic color design.

Decide. In this step, students review all proposed sketches of their group, critique them, and vote to pick the most promising designs. They then combine the highest-rated sketches into a visual dashboard design. During this step, we discuss visual effectiveness [14], Tufte's principles of graphical integrity and design [21], and when it is okay to break those design guidelines.

Prototype. In this step, students prototype their visualization designs using Tableau in our guided projects or D3 in our final projects, respectively. We discuss basic chart types to visualize comparisons, trends, distributions, and correlations during this step. We also discuss the basic map visualizations.

Before this step, we make sure that students are sufficiently competent using prototyping tools. In our on-campus courses, we introduce Tableau and teach students to implement basic chart types and maps. In CS171, we use self-guided weekly programming labs to teach students the basics of web programming with HTML, CSS, Javascript, and D3. Since we have students with varying levels of programming experience, we adjust software or IDE choice based on the students' technical background. Students can also prototype their designs using pen and paper drawings.

Test. This final step aims to have outside users test the teams' visualization dashboards to collect feedback. Each group performs qualitative user evaluation through a think-aloud study with students from other teams. Students then analyze the feedback and come up with possible improvements to their initial prototypes. For final projects, teams then spend another week implementing these improvements. During this step, we present testing and evaluation methods, including a live demonstration of a think-aloud study.

Wrap-up. Unlike the typical design sprint process, we have added a wrap-up step to the CS171 design sprints. We ask students to fill out peer evaluations to rate their own and their teammates' performance during the design sprint. If there are strong discrepancies between a students' self-evaluation and their peers' ratings, we reach out to the student and discuss possible causes and remedies. During the guided projects, these evaluations are formative. In contrast, they could affect a students' grade in their final projects. Since introducing peer evaluations, we have to deal with much fewer typical team issues.

For final projects, this step also involves cleaning up the process book, producing a 2-minute video, and presenting their visualization dashboards in class. To add an element of suspense to the project presentations, we hand out best final project prizes (chocolate) to the top three teams as determined by the teaching staff. These prizes do not influence students' grades, but students love having bragging rights and being added to the hall of fame on our course website.

3 Case Studies

The steps of a visualization design sprint outlined in Sec. 2 stay the same for all different types of courses we teach (e.g., on-campus vs. online, large undergraduate course vs. small professional development workshop). However, we make small adjustments for different courses to fit the course objectives and available time better. In this section, we discuss several case studies of design sprints

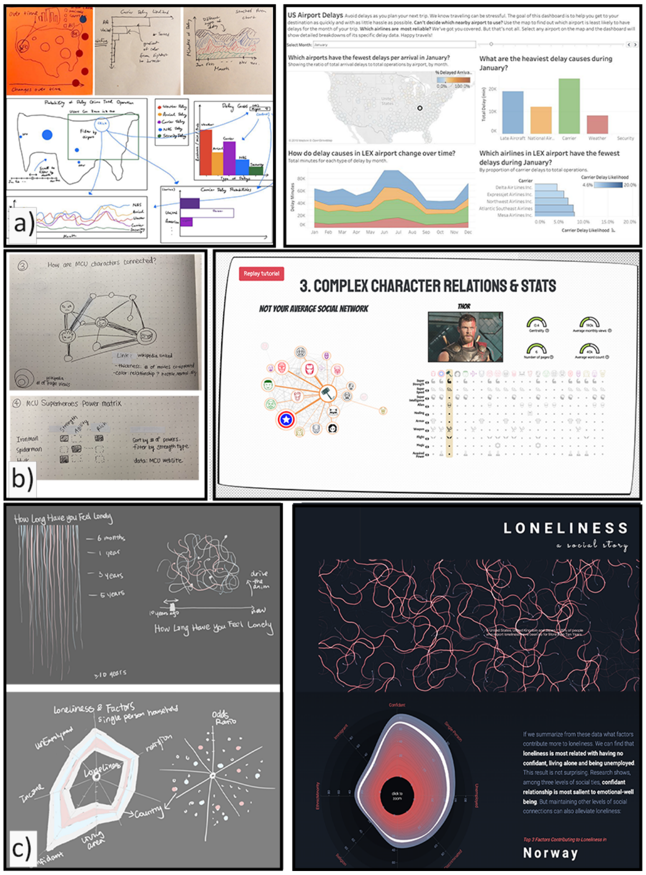

Figure 4: Visualization design sprint case studies. a) CS171 guided project in Tableau using US flight delays data. b) CS171 final project in D3 about Marvel superheroes. c) CS171 final project in D3 about 'Loneliness'. The left column shows student sketches, the right column shows their final dashboards and websites.

performed in a variety of different visualization courses (see Fig. 4). Our supplemental material [3] contains datasets, teaching materials, and grading guidelines we have curated over the past couple of years for our design sprints. We also include examples of student-created process books in the supplemental material.

3.1 On-Campus Courses

Guided Projects in Tableau. We start our guided sprint in the second week of the semester. We assign students randomly into teams, aiming for diverse teams in terms of programming and design experience. We prepare project briefs for one to four different datasets that each team can choose from for guided projects. The project briefs include a dataset description, defines the audience for the final visualization prototype, and states high-level goals. Alternatively, we could let students define their audience and goals. Each week, students work on one step of the design sprint and document their progress in their process book. Students should complete most of the work during class time, but, if necessary, they finish their work at home. In class, several teaching assistants walk around and give feedback to groups. Fig. 4a shows a design sprint on flight delay data. The team created a dashboard to help casual travelers minimize flight delays. The different views are linked to each other and follow an overview-and-detail navigation pattern.

Final Capstone Projects in D3. The second design sprint is less guided and gives students more independence. Students get to pick their teams and have to find a topic and dataset they want to visualize. Every week students have to submit their sprint's current state and get timely feedback from their project mentor, a teaching assistant assigned to be their main point of contact for questions or feedback during the sprint. We have observed that allowing students to pick their teams and projects leads to highly motivated student groups. This has led to a wide variety of topics over the years, spanning from projects on obesity, to the war in Syria, to cryptocurrency, and Marvel superheroes. Fig 4b shows a student project on the interconnection of Marvel superheroes, and Fig 4c shows a project on the topic of loneliness. Both projects were implemented in D3 and incorporated visual storytelling and multiple connected views.

3.2 Online Courses

We have also taught design sprints successfully online, both in semester-long courses and in two-day workshops. The main pitfall in online design sprints is that students need regular and immediate feedback. Therefore, we schedule design sprint sessions to be synchronous and have teaching staff available to give feedback and answer any team questions. One project mentor typically oversees 4-5 groups. In larger online classes, additional peer feedback mechanisms could be used. We slightly reduce the content for online courses to consider that teamwork and group activities tend to take longer in online settings. Especially the two-day workshop requires careful planning and scheduling. We give students a clean dataset and also allow them to work solely with pen and paper instead of Tableau. This approach enables students without Tableau experience to be highly productive in a limited amount of time.

4 Discussion and Lessons Learned

Visualization design sprints are a powerful technique to allow students to gain hands-on experience in visualization projects. This section highlights our experience with design sprints, discusses common pitfalls, and reports on student feedback.

Overall Lessons. Design sprints allow students to apply what they have learned in class immediately. This works very well in visualization courses where students can directly apply high-level concepts, such as Tufte's design principles, to refine their visualization designs. Additionally, by using sketching and tools such as Tableau, students do not need to learn more complex visualization libraries (e.g., D3) before practicing their visualization skills. In contrast to typical design sprints, we also stress the importance of iterative refinement in our visualization classes and extend the amount of time they have for implementing their prototypes.

For design sprints to be effective teaching tools, students must get immediate and direct feedback. We have tackled this problem by assigning *project mentors* to each team that guides them through the design sprint and makes students aware of the strengths and weaknesses in their visualization designs. For example, in the 2-day workshops, we read through each team's project book overnight and leave comments that students can address the next day. Being able to track student progress in a shared process book is essential, especially in the class's online formats. Students also seem to like having a dedicated project mentor: "*Having a project mentor is a great idea — my group learned a ton from our mentor!*"

Student Engagement is High. We found that the optimal team size for design sprints in visualization courses is around 3-4 students. This ensures that everyone is engaged, simplifies coordination, and students will call each other out if someone is not pulling their weight. We also found that our peer feedback mechanism further helps in keeping student engagement high.

Project Quality Overall is High. In the previous lecture version of the course, we noticed that projects would go wrong in the expected ways: teams missed deadlines, did not follow a structured process [15], ignored lessons of good visualization design, and some students were hitchhikers or couch potatoes [16]. With design sprints, almost all of these problems went away. The clear and simple process makes it easier for students to know what is expected from them each week, and there is less risk of students falling behind.

We also found it helpful to share good examples from previous years (i.e., process books and prototypes) with students.

Finally, adding a small element of competition to the design sprint (i.e., a 'best project award') is appreciated by our students and our teaching fellows. Students want to beef up their resume, and teaching fellows also take pride in mentoring the winning team, so they are more involved during the design sprint.

Lessons for Online Teaching. An essential difference in online teaching is that students feel less connected and, therefore, clear communication is even more critical than in on-campus settings. That means that everything needs to be clearly explained and documented plenty of time ahead. For example, we clearly communicate how teams will be assembled, when and how teams are expected to meet, and we explain each step and milestone in the design sprint process. This allows online students to plan ahead and reduces student anxiety. Networking does not work as well in online settings. Therefore, proper asynchronous communication channels (e.g., Slack or Piazza) are essential to connect students and teaching staff. To avoid overwhelming students, we limit the number of different collaborative tools we use and provide usage instructions for each tool. We mainly use Google docs for collaboration and only introduce additional tools if they add a significant benefit that cannot be achieved by video conferencing and Google docs alone.

Short Online Courses. An important consideration for short (i.e., single-, or multi-day) online courses is the necessity for strict time management. Team activities and giving feedback online takes longer than in person. We found it challenging to fit a Tableau-based design sprint into our two-day online workshop. A better alternative might be to just use pen and paper prototyping in those scenarios.

We also found that the busy schedule of online workshop participants makes it hard for them to complete homework before and during the course. This makes timely and continuous feedback throughout the course even more vital. We also found it crucial to have technical support staff to manage our online video conferencing tool, set up breakout rooms. and help students with technical issues.

Student feedback. Students have overwhelmingly liked our design sprints. In particular, the final project is the favorite part of the course for many students. We have received feedback such as "*The final group project was a great way to combine all the skills you learned throughout the semester and turn it into some complex visualizations about real issues and topics, which is something great to be added to a portfolio.*" and "*The final project especially showed me that I'm capable of creating unique visualizations from just the simple building blocks of D3 shapes and areas*".

When we originally switched our large undergraduate on-campus class of roughly 200 students to active learning in 2016, our student evaluation (Q scores) increased by 0.8 points to 3.8 (out of 5) compared to the earlier lecture-based course, which had a maximum Q Score of 3.0. With the switch to design sprints in 2018, the Q Score of CS171 has increased another 0.4 points to 4.17 out of 5. More specifically, on a 5-point Likert scale, 72% of all students agreed or strongly agreed that the guided design sprint helped them learn how to apply visualization principles and a structured design process. 77% of all students agreed or strongly agreed that using Tableau in the guided sprint allowed them to focus on good visualization design and visual effectiveness early on.

5 Conclusions

We will continue to use design sprints in our visualization courses, both online and in person. We believe that design sprints are an excellent technique for students to gain hands-on experience in visualization design. We distinguish between guided design sprints, that are performed mainly during class time, and more independent sprints that can serve as a final capstone project. For our guided design sprints, we use Tableau but might switch to an all sketching approach for shorter classes and workshops.

We have noticed that design sprints lead to projects that adhere more closely to good visualization design principles, but often at the cost of less creative and more uniform looking projects. We believe that this is not necessarily due to design sprints but instead from students internalizing and applying effective visualization design principles. Different visualization tools certainly also influence the diversity of student solutions. We counter this by explicitly encouraging students to come up with novel visual designs and by teaching them how to implement more creative ideas in their chosen visualization software or library. We also plan to emphasize more playful approaches to visualization, such as the "Dear Data" project [13] or the work of Nadie Bremer [7], in future courses.

Finally, design sprints are not just more engaging for students; they are also fun for instructors and teaching staff. They have made our teaching more rewarding, and we believe that design sprints will benefit visualization courses at other institutions.

References

[1] CS 171 - Harvard University. http://cs171.org.

[2] Data Visualization Course - Harvard Extension School. http://vcg.github.io/dataviscourse/.

[3] J. Beyer and H. Pfister. Visualization Design Sprint Supplemental Material. https://github.com/CS171/design_sprint_material.

[4] J. Beyer, H. Strobelt, M. Oppermann, L. Deslauriers, and H. Pfister. Teaching Visualization for Large and Diverse Classes on Campus and Online. In *Pedagogy of Data Visualization Workshop (PDVW) at IEEE VIS*, 2016.

[5] C. C. Bonwell and J. A. Eison. *Active Learning : Creating Excitement in the Classroom*. ASHE-ERIC Higher Education Reports. George Washington University, Washington, D.C, 1991.

[6] M. A. Borkin, Z. Yan, B. Horn, L. Roe, and B. Berkey. Visualization Education Through Social Impact: A Service-Learning Approach for Visualization Pedagogy. In *Pedagogy of Data Visualization Workshop (PDVW) at IEEE VIS 2017*, 2017.

[7] N. Bremer. Visual Cinnamon. https://www.visualcinnamon.com/.

[8] V. Ferreira and E. Canedo. Design sprint in classroom: Exploring new active learning tools for project-based learning approach. *Journal of Ambient Intelligence and Humanized Computing*, p. 3, Mar. 2019.

[9] T. B. Hilburn and W. S. Humphrey. Teaching Teamwork. *IEEE Software*, 19(5):72–77, Sept. 2002.

[10] J. Knapp, J. Zeratsky, and B. Kowitz. *Sprint: How to Solve Big Problems and Test New Ideas in Just Five Days*. Simon and Schuster, 2016.

[11] K. Koffka. *Principles of Gestalt Psychology.* Principles of Gestalt Psychology. Harcourt, Brace, Oxford, England, 1935.

[12] T. Lowdermilk. *User-Centered Design*. O'Reilly, 2013.

[13] G. Lupi and S. Posavec. *Dear Data*. Particular Books, 2016.

[14] J. Mackinlay. Automating the design of graphical presentations of relational information. *ACM Transactions on Graphics (TOG)*, 5(2):110–141, Apr. 1986.

[15] T. Munzner. *Visualization Analysis and Design*. AK Peters Visualization Series. CRC Press, 2015.

[16] B. Oakley. It Takes Two to Tango: How 'Good' Students Enable Problematic Behavior in Teams. *Journal of Student Centered Learning*, Jan. 2002.

[17] B. Oakley, R. Brent, R. Felder, and I. Elhajj. Turning student groups into effective teams. *Journal of Student Centered Learning*, 2:9–34, Jan. 2004.

[18] M. Prince. Does Active Learning Work? A Review of the Research. *Journal of Engineering Education*, 93(3):223–231, 2004.

[19] M. Sedlmair, M. Meyer, and T. Munzner. Design Study Methodology : Reflections from the Trenches and the Stacks. *IEEE Trans. Vis. and Comp. Graphics (Proc. InfoVis)*, 18(12):2431–2440, 2012.

[20] S. S. Stevens. On the Theory of Scales of Measurement. *Science, New Series*, 103(2684):677–680, 1946.

[21] E. R. Tufte. *The Visual Display of Quantitative Information*. Graphics Press, second ed., 2001.

[22] R. Williams, 1953. *The Non-Designer's Design Book : Design and Typographic Principles for the Visual Novice*. Peachpit Press, 1994.