Prospectus

The Advising Assistant

(Technical Topic)

Semiotic Approaches & Software Development

(STS Topic)

By

Scott Lutz

11/22/19

On my honor as a University student, I have neither given nor received unauthorized aid on this assignment as defined by the Honor Guidelines for Thesis-Related Assignments.

Signed: _____

Approved: _____ Date 12/16/2019

Benjamin Laugelli, Department of Engineering and Society

Approved: _____ Date 25 Nov 2019

Luther Tychonievich, Department of Computer Science

Introduction

University of Virginia's (UVA) Student Information System (SIS) is a website used by students and faculty. On SIS students pay for tuition, sign up for classes, and check their course requirements and transcripts. Creating a schedule and planning curriculum is a frustrating task for UVA students. Students are tasked with looking at a selection of classes, adding it to a schedule and then returning to a list of classes to add more without being able to see the schedule built thus far (Weigand, 2019). This creates frustration because students cannot visualize the timing of the schedule being created. Another point of frustration stems from the fact that students have no way to verify whether they are expected to graduate on time ("Student information system"). This creates anxiety and feelings of uncertainty for some students.

To address this problem, I am designing a web application, for students, that will create feelings of satisfaction and eliminate frustration. The web application will encompass all of the functionality within SIS used to build a schedule in a more practical way. The design will be focused around creating an experience that matches how users work through the process of curriculum planning. Currently there are tools for students to use outside of SIS such as, Lou's List and UVA schedule | me; however, none of these tools offer all of the functionality students need. My design would incorporate these tools into one along with additional functionality that students have never had access to before, such as curriculum plan verification and suggested courses based on interests. The current tools created by students work as a simple technical fix, but they are inadequate to address the broad problem fully. The broad problem is that students have a frustrating experience creating a schedule. To address this problem fully it is important to keep in mind that technology acts as a script for users to follow but sometimes conflicts with the users needs and practices when the designer has incorrect ideas about the user that get embedded

into the technology (Lindsay, 2003). It is necessary to examine how users work during the process of designing a schedule. Providing all of the functionality a user needs is not sufficient. A failure to take into account users needs and practices when developing software would result in a product that does not have a good workflow. By addressing both the technical and social aspects of this problem, developers stand to gain an understanding of how to build software that users enjoy using. Creating a satisfying experience for users is a socio-technical problem in nature because the software must work correctly, be intuitive to use, and match the desires of the user. The technical aspects of the solution require understanding what functionality users need in order to build a schedule and plan their curriculum. The social aspects of the solution require understanding why users have resisted the design of SIS using the STS framework of user configuration. I will analyze the tool students have created for themselves to better understand the differences between the designers' and users' scripts.

Technical Problem

UVA has over 20,000 students currently enrolled, and every semester they need to sign up for classes ("Statistics"). Students are able to sign up for classes by adding them to a cart, and when their designated enrollment time comes along, they can add them to their schedule; however, there is a major issue with SIS. SIS was created using iframes. Iframes allow a webpage to have another webpage embedded inside of it ("The inline frame element", 2019). This makes SIS a context sensitive website. This is poor practice because information about what a user is looking at is no longer stored in the URL. This leads to not being able to have multiple tabs open at a time and not being able to use the browser's back button (Smirnov, 2019). All of this creates a very frustrating experience for users. This becomes especially apparent when students are trying to build a schedule. Students have to navigate back and forth between the list

of classes, requirements, and their schedule. This involves a lot of repetitive actions and searches when trying to build the perfect schedule and it creates a frustrating experience.

Many students have elected to not use SIS at all while creating their schedule. There is an unofficial website called Lou's List that was created by a professor at the University (Bloomfield). The website scrapes SIS for information about all of the courses offered each semester and presents it in a way that is much easier to digest. It also allows for users to have multiple tabs open, so they can compare courses from different departments. Using Lou's List, some students create Excel spreadsheets so that they can visualize how the time slots for their schedule will line up. There is another tool that was developed by students called UVA schedule | me ("uva schedule | me"). This website allows users to search for classes and add them to a schedule. It displays the schedule visually and updates it dynamically to give users an idea of how certain course sections will line up, so they can avoid conflicts. Users can also select classes and have the website auto generate many different schedules to choose from. The searching functionality on Lou's List is much more powerful, with advanced searches. According to Google Trends related queries, most students tend to use these tools together ("Google trends: Lou's list"). After they have crafted their schedule students have to go back to SIS and add each of the classes they want into their cart.

SIS does not support the workflow required to build a schedule. This major glaring issue has caused students to seek out or create their own tools in order to have an easier time creating one; however, none of these tools single-handedly fix the problem. New students are generally unaware that these tools exist, and having to use multiple different tools to accomplish a simple goal can still be frustrating and these tools do not offer all of the functionality students need.

By consolidating everything students need to plan their curriculum into a single tool it will make the process much easier and more enjoyable. They will need less help from their advisors and it will be less frustrating. I am proposing a web application that helps students plan out their curriculum for all 4 years, recommends elective classes to students based on their interest, and verifies that students are meeting all of their requirements to graduate on time. This website will be created following good user experience design guidelines and with the primary workflow in mind, creating schedules. The website will be developed primarily with Python and JavaScript. Students will test the design and their feedback will help shape it into a final product that everyone can use. The first iteration of the design is intended for Computer Science majors, but I plan to eventually incorporate all majors and minors at the University of Virginia.

STS Problem

At UVA students use SIS, a website, to sign up for classes. Students search for a class, add it to the cart, view the schedule created by adding that class, and then repeat for every additional class. This is extremely repetitive and tedious, so students have created workarounds in order to avoid using SIS directly while building their schedule. The major purpose of SIS for students is to create a schedule, and it has all of the desired functionality to do that. Users can search for classes, add classes to a schedule, view a picture of their schedule, and reference a list of required classes. It would seem that the technology would be useful since it has everything a user could need, but it is designed without the capabilities of the user in mind. SIS's design has a flawed understanding of the user's identity built into its core. SIS has been configured to a user that does not require visual aid while building a schedule. As a user is building a schedule and selecting classes it is not possible to look at the classes already selected. Users have to navigate through many different pages to find this information, remember it or write it down, and then

search for a new class and decide whether or not to add it. By exploring the workflow of schedule creation in SIS and workarounds students use, developers will gain a better understanding of the way unintentional and implicit bias has affected the design.

Drawing on the STS concept of user configuration, I argue that the designers unintentionally embedded biases about the users' thought process in the technology's design. In the case of SIS, designers did not consider how a typical user would build a schedule. Instead they imagined a user who works in a linear fashion and never needs to double check something on a previous page. The concept of user configuration will allow me to analyze SIS as a technological script. User configuration examines ways designers embed ideas about users into technologies, which then function as scripts that direct what users can or cannot do (Akrich, 1992; Woolgar, 1991). The script, then can be described as the workflow forced upon the user by design. The user follows the script in order to use the software; however, in the case of SIS the users' script does not match the technological script. This conflict causes the user to resist and create workarounds. The concept of antiprogram, the extent that designers' and users' scripts conflict, will allow me to analyze how the tools created by users, such as UVA schedule | me, were an expression of user agency to address major issues in the design of SIS (Oudshoorn & Pinch, 2003, p.7-11). I will also draw on the concept of user agency, which analyzes how users take action to modify existing technology to fit their needs. In this case, the users have created tools such as Lou's List and UVA schedule | me to help them use SIS more efficiently.

Conclusion

The technical project will deliver a new platform for students to create and customize schedules and a plan for their curriculum over the next 4 years. It will also offer recommendations and validation for the plans they create. This will consolidate all of the tools

students need into one location to create a satisfying experience. The STS project will deliver a better understanding of how SIS is configuring the student user in ways that conflict with the needs and practices of actual users. Understanding the social factors involved in student schedule creation will help ensure a design that fulfills the users' desires and matches their workflow. By combining the results from each of these projects, it will be possible to create a design that can satisfy users and eliminate frustration because users will have all of the functionality they need and using the software will be intuitive and not require dependence on additional tools outside of SIS. Users will have access to everything they need so they can easily make decisions about their schedule as they are building it.

Word Count: 1819

References

Akrich, M. (1992). The de-scription of technical objects. *Shaping technology/building society*, 205–224.

Bloomfield, L. (n.d.). UVa class schedules. Retrieved from https://rabi.phys.virginia.edu/mySIS/CS2/.

Google trends: Lou's list. (n.d.). Retrieved from https://trends.google.com/trends/explore?date=today 5-y&geo=US&q=lous list,uva lous list.

Lindsay, C. (2003). From the shadows: Users as designers, producers, marketers, distributors, and technical support. In N. Oudshoorn and T. Pinch (Eds.), *How users matter* (pp. 29-51).

Oudshoorn, N. & Pinch, T. (2003*). Introduction: how users and non-users matter*. In N. Oudshoorn and T. Pinch (Eds.), *How users matter* (pp. 1-25).

Smirnov, A. (2019, April 30). 3 Reasons you might not want to use iframes. Retrieved from https://www.ostraining.com/blog/webdesign/against-using-iframes/.

Statistics. (n.d.). Retrieved from https://admission.virginia.edu/admission/statistics.

Student information system. (n.d.). Retrieved from https://msisuva.admin.virginia.edu/.

The inline frame element. (2019, August 28). Retrieved from https://developer.mozilla.org/en-US/docs/Web/HTML/Element/iframe.

uva schedule | me. (n.d.). Retrieved from https://www.uvaschedule.me/.

Weigand, S. (2019, May 20). UX case study: process of redesigning UVA's student information system (SIS) page. Retrieved from https://medium.com/@smw5gn/ux-case-study-process-of-redesigning-uvas-student-information-system-sis-page-7ee7ecfabc88.

Woolgar, S. (1991). Configuring the user: The case of usability trials. In *A sociology of monsters: Essays on power, technology and domination* (pp. 58–99). London: Routledge.