A Proposal for a Unified Interface Production Environment for Independent Developers

Craig Earley
Earlham College Computer Science
801 National Rd W
Richmond, Indiana - 47374
cjearley13@earlham.edu

1. INTRODUCTION

Software applications and websites facilitate interactions between a user and some data. Research into human-computer interaction has shown that changes to the presentation of objects in an interface affects how users respond to the information - in other words, design is not value-neutral from the perspective of user behavior.

Several challenges face independent developers in this area. They lack the requisite resources to conduct large-scale testing, for example. Technology, however, increasingly disconnects work from geography. In many cases, at the scale independent developers work, simply referring to an extension, sending a little bit of information out via email, and waiting twenty-four hours to collect anonymized data may serve their purpose.

This is the problem I propose to solve. In this paper I describe the need for a framework to allow independent developers to explore interface design quickly and easily in a web context. I also explain how it might be implemented and sketch a timeline for completion of the project.

2. BACKGROUND RESEARCH

In a literature review submitted earlier, I discussed the theoretical background that has explored how users and computers interact. The following is a summary of the relevant research, including those portions relevant to the project itself.

2.1 Theory

HCI as an academic discipline in computer science emphasizes communication. Haynes et al., for example, conducted a series of interviews with HCI design researchers about how design facilitates understanding. They described several established and emerging social theories. Of particular concern in this context is their description of the established, fundamental frames of reference cited by their interviewees. Most important are causality, instrumentality, and prediction, all of which are relevant to considering user responses within a given interface.[2] This paper is more general than the direct topic of concern here, but emphasizes that computers communicate with people, an assumption that underlies the rest of this review.

My research draws most on Oinas-Kukkonen's research about how interfaces on a variety of platforms change human behavior, with an explicit emphasis on the human's side of the interaction. I will in particular draw on his heuristic series of specific changes an interface may want to produce: changes in attitude, in behavior, or in compliance. He emphasized that this is a cross-platform phenomenon.[6] I will focus on compliance (completion, in some amount of time), though as I will describe this software could easily examine behavior change as well.

2.2 Design Heuristics

Within a given platform, it is necessary to control the factors presented to the user. Liu et al.'s examination of behavioral change matters here. They found that, when experiencing a visual warning nudge, users were less likely to click through and more likely to employ a "dismiss" feature than when no warning was activated.[4] I will also heed the findings of Kalnikaite et. al., who found that an overwhelming interface presenting too many options or indicators is excessive. Reaction time is improved with an emphasis on fewer factors rather than more. [3]

Cockburn and Gutwin's experiments in different "shapes" for interfaces is a reminder to ensure that this work is responsive. They conducted a series of trials with "grid," "linear," and "binary" interfaces. They concluded that information structure and user experience directly affects the time required to complete a set of tasks through an interface.[1]

2.3 Popular Research

The high-level background for this topic must also incorporate design research from the private sector. Of particular value is *The Design of Everyday Things* by Donald Norman on the topic of usability (in physical as well as digital contexts). He emphasizes the responsibility of the designer - e.g., that "human error" is often a consequence of design flaws instead of clumsiness - and in particular describes the effect of design on the psychology of the user.[5] His insights will guide both my proposed software, which I will describe below, and potential test cases to consider.

The combined academic and popular research, coupled with the timetable on which this project will be conducted (see below) emphasize some common principles: ease of use, simplicity, and clarity. These design principles guide the

© 2016 ACM. ISBN 978-1-4503-2138-9.
DOI: 10.1145/1235

general project. Further research will examine more carefully the details of particular features, but for current purposes this general review is sufficient.

3. PRELIMINARY RESULTS AND DEFINING THE PROBLEM

In short, design of interfaces is of critical importance. This is true for the private sector for obvious reasons: increased click-through rates, more pleasant experiences, and greater command of user behavior are critical objectives to increasing the financial success of companies.

However, it is of value in academia as well, in that academics wish to guide students, research participants, and others to succeed in a given set of goals. Academics lack access to the often proprietary resources of the private sector. This is an even more acute problem for students and independent developers, who lack the resources for usability testing of any scale.

Major firms have application programming interfaces (API's), internal guidelines, the credibility to organize testing and focus groups, and other advantages that those not employed by them do not enjoy. The application of the principles of human-computer interaction, thus, require extensive setup work.

4. PROPOSED SOLUTION

I propose a framework for conveniently choosing, displaying, and analyzing interface options for comparison, for use by smaller organizations, academics, and other developers not affiliated with major technology firms. For now, this will focus on speed, the feature most easily checked using built-in system features and accessible software.

Importantly, this solution is not preferable to a full-scale usability test or A/B test. Given the proper resources, such tests are superior and can produce substantially greater insights. The vision for this project is to allow projects on an individual or small-group scale to be quickly and easily tested. As they scale, more robust tests will still be necessary.

4.1 Hardware

I require minimal hardware. I will make use of the Earlham College Computer Science Department's cluster computing network, as well as my own Macintosh laptop, to design and implement this tool. These computers include a web server and other system tools in the event that I wish to produce and host custom web pages for testing.

4.2 Software

The core of the project is the software application, which will be web-based. This serves several purposes. First, it is straightforward to design browser extensions, which have a robust API ecosystem and an active developer community. Second, every computer user has web access and can easily download the necessary browser extension, so such disparities as iOS/Android in mobile applications do not apply. Third, user behavior on the web is a ubiquitous case in both commerce and academia. Together, they make this project simple and broadly useful.

The initial software will be a Google Chrome extension, meant to be minimal enough to be adaptable for other browsers in the future. It would open a tab containing a text box that would take three inputs: a start URL, a stop URL, and a natural number specifying which version the user chose. (Standard software features such as exception handling will be included.) This could then be exported for quick A/B testing of web features of either a large or small scale. Any data collected would be anonymized and stored in a CSV.

4.3 Data Collection

This project will culminate in requesting the members of the class to examine the software for fixed URL pairs. They will be timed, anonymously, as they each complete the process for one interface between a fixed pair of URL's. The anonomized data can then be analyzed using basic statistics to see which interface tended to be completed more quickly, if any. This will require the submission of a form to the Institutional Review Board, which I will complete within the week.

4.4 Long-Term Prospects

In the future, this software may be expanded to support evaluation of user choices - e.g., given a starting URL and a set of one or more stop URL's, which stop URL does a user tend to land on given a particular interface? This data could be collected with similar anonymization.

It should also be developed for browsers other than Chrome. Firefox has a strong development community and many "addons" of its own and would be a likely candidate for a followup application.

5. TIMELINE

This solution is necessarily limited in scope. This project will be conducted over the course of three calendar months. For the remainder of October, I will complete my consumption of all research material, including annotations and comments, and make the design and specification decisions that will drive the rest of the project. It is likely I will also begin the research paper's first draft.

I anticipate spending around 8 hours per week on the core of the project, including supplementary readings, software development, and writing.

The following is a tangible timeline (all dates are deadlines):

- October 26: Software design complete (i.e. the design is robust enough to implement a preliminary version with no additional design); submit form to the IRB in the event the software is complete enough to test
- November 2: Paper fully outlined and key topics understood
- November 9: Initial software written; poster outlined
- November 16: First draft completed and handed in
- November 30: Be prepared to present (may also end up presenting December 4)
- December 12: Second draft complete
- December 16: Final draft complete, best version of software done

This will be modified as project specifications are changed.

I will leave notes, likely in the form of a post on the class website, about the process, the goals, and the progress that

has been made by the end of the semester. This will act as a resource in the event that future students wish to consider this topic in greater depth. By the time I finish December, I intend to complete a base case of this project that others may adapt and revise in the future. I will submit a second draft on or before December 12 and a third on or before December 16.

6. ABOUT THE AUTHOR

Craig J. Earley is a December 2016 graduate of the Earlham College Computer Science Department.

7. REFERENCES

[1] A. Cockburn and C. Gutwin. A model of novice and expert navigation performance in constrained-input interfaces. *ACM Trans. Comput.-Hum. Interact.*, 17(3):13:1–13:38, July 2010.

[2] S. R. Haynes, J. M. Carroll, T. G. Kannampallil, L. Xiao, and P. M. Bach. Design research as explanation: Perceptions in the field. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, pages 1121–1130, New York, NY, USA, 2009. ACM.

[3] V. Kalnikaitė, J. Bird, and Y. Rogers. Decision-making in the aisles: Informing, overwhelming or nudging supermarket shoppers? *Personal Ubiquitous Comput.*, 17(6):1247–1259, Aug. 2013.

[4] J. Liu, S. Ruohomaa, K. Athukorala, G. Jacucci, N. Asokan, and J. Lindqvist. Groupsourcing: Nudging users away from unsafe content. In *Proceedings of the 8th Nordic Conference on Human-Computer Interaction: Fun, Fast, Foundational*, NordiCHI '14, pages 883–886, New York, NY, USA, 2014. ACM.

[5] D. Norman. *The Design of Everyday Things*. Basic Books, 2002.

[6] H. Oinas-Kukkonen. A foundation for the study of behavior change support systems. *Personal Ubiquitous Comput.*, 17(6):1223–1235, Aug. 2013.