XRAY: Inspector Tools For Designers

Leave Authors Anonymous
for Submission
City, Country
e-mail address

Leave Authors Anonymous
for Submission
City, Country
e-mail address

Leave Authors Anonymous
for Submission
City, Country
e-mail address

ABSTRACT

Numerous tools have been created to facilitate developers' borrowing of code from live websites and design examples. However, little work has been done to improve the experience of designers working on partially-developed or live sites. This paper introduces XRAY, inspector tools for designers (Figure 6). Unlike traditional inspector tools, XRAY allows the designer to adjust fonts, colors, margins, and padding within the browser without ever needing to look at HTML or CSS, making this technical, traditionally code-based task more approachable for designers. XRAY promotes the use of design systems by only suggesting styling options that exist in the current design system and highlighting where current aesthetics violate the design system. XRAY also improves designer-developer communication by allowing people in different locations are able to make live edits to a website collaboratively. XRAY allows users to export a document with all changes at the end of their session. Moreover, a 12 person user study with novices and a 12-person user study with professional designers showed that people were xx% more efficient, yy% more successful, and experimented more by using zz% more styles when they used XRAY, than when they used the standard industry tools.

ACM Classification Keywords

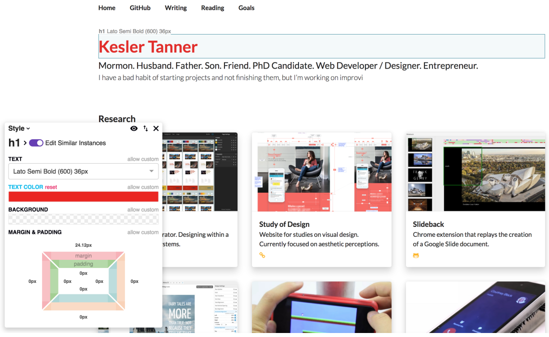
H.5.m. Information Interfaces and Presentation (e.g. HCI): Miscellaneous; See http://acm.org/about/class/1998/ for the full list of ACM classifiers. This section is required.

Author Keywords

Web design; design systems; inspector tools; experimentation; human factors; developers; designers.

INTRODUCTION

By definition, designers plan the UI of software and developers write the code. Although some designers can code and some developers are artistically savvy, most professionals in industry specialize. Creating a finished projected is a collaborative and iterative process. For example, if a company was preparing to launch a new website, the designer would create a mockup of how the product should look and send it to the developer. The developer would code it, and send it back to the designer. No first draft is flawless; the designer would next communicate the mistakes to the developer.



h1 Lato Semi Bold (600) 36px

Kesler Tanner

Mormon. Husband. Father. Son. Friend. PhD Candidate. Web Developer / Designer. Entrepreneur.

I have a bad habit of starting projects and not finishing them, but I'm working on improvi

h1 > Edit Similar Instances

TEXT allow custom: Lato Semi Bold (600) 36px

TEXT COLOR reset allow custom

BACKGROUND allow custom

MARGIN & PADDING allow custom: 24.12px, 0px, 0px, 0px, 0px, 0px, 0px, 0px

rator. Designing within a stems.

Study of Design
Website for studies on visual design. Currently focused on aesthetic perceptions.

Slideback
Chrome extension that replays the creation of a Google Slide document.

Figure 1. A screenshot of the XRAY user interface and the menu for [fonts? colors? margin/padding?]

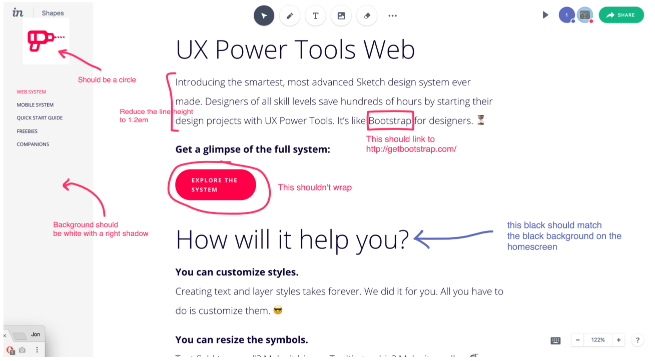
Some designers will meet with the developers, sitting next to them and explaining what needs to be changed. However, that is very time-consuming for both the developer and designer. Other designers might send an email with a checklist of what needs to be fixed, but matching the block of test to the website gets confusing very quickly. In reality, it is most common for the developer to "redline" by marking up a screenshot of the website using tools like Sketch or InVision, such as in Figure 2 as taken from Moore's 2017 Medium article [10].

The ability to write comments on a screen simplifies the revision process. Instead of writing an essay for each bullet point ("the icon of a drill on the top left of the home page should be enclosed by a circle instead of a square"), the designer can draw an arrow to the icon they are dissatisfied with and write "should be a circle."

However, when it comes to colors, padding, and other sizing issues, detail is key. Instead of saying "this black should match the black background on the homescreen", why not just tell the developer the hexcode should be #000000? Additionally, prior work shows the benefits of experimentation, yet writing notes on screenshots does not really provide the opportunity to expirement and try new ideas. Ideally, when the designer notices that the line height is too large, they would have the opportunity to see what the page would look like with 1.1em, 1.2em, and 1.3em before telling the developer which one is best.

CHI'16, May 07–12, 2016, San Jose, CA, USA

ISBN 123-4567-24-567/08/06...$15.00
DOI: http://dx.doi.org/10.475/123_4



UX Power Tools Web

Introducing the smartest, most advanced Sketch design system ever made. Designers of all skill levels save hundreds of hours by starting their design projects with UX Power Tools. It's like Bootstrap for designers. ⌛

Get a glimpse of the full system:

How will it help you?

You can customize styles.

Creating text and layer styles takes forever. We did it for you. All you have to do is customize them. 😎

You can resize the symbols.

Should be a circle

Reduce the line height to 1.2em

This should link to http://getbootstrap.com/

This shouldn't wrap

Background should be white with a right shadow

this black should match the black background on the homescreen

Figure 2. Example of a "redline" or markup of website that a designer might send to a developer

Currently, the developers are gatekeepers: they control the design and aesthetics of a finished product even though they do not have the knowledge or training that a designer does. The designers are in the uncomfortable position of knowing exactly how the final product should look, but are unable to create it themselves.

In order to provide designers with more control, many companies are creating design systems. Just as developers use a style guide in order to have a standardized way of writing code, designers create design systems to standardize the look and feel of a product.

Most companies strive to be consistent in their use of colors and fonts across websites, emails, advertising campaign, and apps [cite]. Design systems evolved from these style guides. Couldwell walks through how WeWork created a design system which contains not only font families, font sizes, and colors, but also templates, patterns (which include recurring elements or practices like navigation, cards, tables, or alerts), and components (pre-built widgets, such as buttons, sliders, or inputs, selects, toggles, avatars, tooltips, and etc) [4].

When developers and designers face seemingly small questions like "How should we format the time/date/currency?" or "What kind of color/typeface/button should I use?" or "Should this be sentence case or title case?", the design system can evolve to hold these answers [3]. Often a [file type] or [photoshop file], the design system can contain images of what these fonts, colors, and widgets look like, as well as the code to make them. This prevents developers from making the same buttons over and over for each website, and ensures that the color, shadow, and animated reaction for each click or hover over the button will be exactly the same [3].

Although the use of design systems is spreading rapidly in industry, we are not aware of any research regarding the creation or use of them. Companies including [list them] are currently using them; some began creating their design system as early as [year].

In order to 1. improve the experience of web designers, 2. improve their collaboration with developers, and 3. to promote the use of design systems, we built XRAY (Figure 6), design-oriented inspector tools for developers. Unlike traditional inspector tools, XRAY allows the designer to adjust fonts, colors, margins, and padding without ever needing to look at HTML or CSS, making this technical, traditionally code-based task more approachable for designers.

XRAY makes the following contributions:

- A novel set of direct manipulation web tools (inspector tools??), designed for those who are more familiar with the UI/UX of graphic design tools and do not need to understand how HTML and CSS work. XRAY allows users to adjust fonts, colors, margins, and padding on live sites without ever looking at any code.

- XRAY focuses on design systems. By only suggesting choices that are in the current design system, it helps novice designers be consistent. XRAY provides the option to override these design guidelines, allowing users to expand the design system by adding new rules. When users redefine an aspect of their design system, XRAY automates consistency, letting the changes percolate through not only the current page but the entire website, persisting even after page reloads. XRAY highlights where a site breaks from the design systems, letting users quickly identify possible issues.

- Designer-developer communication is improved because people in different locations are able to make live edits to a website collaboratively. At the end of a session, designers can export a document with all of their final changes noted in HTML and CSS to give to the developer. This exported document of changes can then be uploaded by the developer (or any other user) on a different computer to see what the live site would look like with those recommended changes.

- A within-subject evaluation of XRAY in comparison to the industry standard of using Sketch(c) to note flaws in a website. We had a total of 24 partcipants, half of whom were design professionals with a minimum of five years' experience, and half of whom were design students at a competetive private university. Our results showed that people were xx% more efficient, yy% more successful, and experimented more by using zz% more styles when they used XRAY, than when they used the standard industry tools.

RELATED WORKS

1. understanding (existing code) tutorons [6] javawhyline [8] firecrystal [11] webcrystal [2]

2. modifying (existing code)

3. tools for novices or non-programmers

4. finding/using example code bricolage [9] chickenfoot [1] copystyler [5]

   telescope [7]

5. data systems/stylesheets (empty bucket)

6. place in a bucket

INTERFACE AND FEATURES OF XRAY

Created to assist designers working on websites, XRAY is a chrome plugin written in JavaScript [check that] and is similar to inspector tools. “WebCrystal accesses the document object model (DOM) of the web page, and treats every DOM node in the web page as a web element.” Users can access XRAY by [pressing the extension button in the top right of their screen], as shown in Figure 6. Once XRAY is turned on, users can alter the HTML and CSS of a website of their choice by using buttons and menus, without looking at a single line of code.

Fonts (family, weight, color)

In addition to providing the font families, weights, and colors that are already in the design system, XRAY automatically downloads Google fonts to make them available on the website. This is often a very tedious task that requires the designer to either install the fonts locally or have access to the code.

Whitespace (margin and padding)

Currently, when designers are redlining a document, they only see the whitespace. XRAY shows the padding and margin so that designers can better learn the difference in order to more effectively communicate with developers.

Design system

XRAY preforms a live audit by highlighting aspects of a website that fall outside of the existing design system to make them easier to identify. This gives developers a change to decide if they should expand the design system by including this outlier, or if they should bring the design into alignment by being more consistent and fixing the outlier.

Experimentation

In order to increase experimentation, XRAY lets users copy and paste styles from one element to the next. These changes do not need to be from the same webpage. Users can toggle on and off aspects of those changes, and aspects from several sources can be pasted into the same element. XRAY automates consistency, letting the changes percolate through not only the current page but the entire website, persisting even after page reloads. Users get an instant preview of those changes, which allows them to experiment more because they no longer need to adjust each different heading to see how it would look.

Collaboration

Two people can make live edits to a website collaboratively. While our users were mixed about the practicality of this feature, one benefit is that designers can share the session with a developer so they can have a live editor with the changes, instead of a static list of the differences.

Download/explore final changes

Instead of a series of redlined screenshots, users can download the final visual changelog from XRAY to give to the developer. Instead of descriptions of requested changes, the changelog is code, making it easy for the developer to make the website look exacly as it did when the designer finished tweaking and adjusting it using XRAY. This will eliminate the need of further back-and-forth communications.

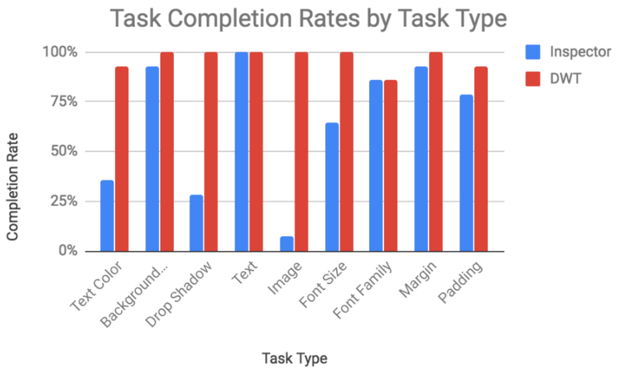


Figure 3. more people finished the tasks when they used our tool

USER STUDY

We surveyed designers and found that xx% of them said task AA was frequently part of their work and yy% of them said that BB described their work well, therefore we designed the user study

Study Design

Participants

We had twelve different students participate in our user study. Of them, xx% were female. YY% were undergraduate students. We only selected participants who had ranked their skill [in design??] as a 3 or higher out of 5 where a 3 was a [quantification] (or had inspector tool experience??). The study was approved by our IRB and no adverse affects were reported during the study. The study took approximately 60 minutes to complete and we paid students $ZZ for participating.

Tasks

All users were given a design system and the same nine steps/tasks to complete in order to bring an existing website into [fitting the system]. The tasks included switching font family, changing background color, editing text, adding whitespace, adjusting shadows on buttons, and swapping out an image.

Procedure

And fill in this section too

Results

CONCLUSION/DISCUSSION

Discuss benefits...

- promotes use of design system
- improves designer-developer communication
- this gets designers into the same medium that developers are using (ie, “writing code” except they are still in their comfort zone”)
- keeps the designers in their element: they can ignore the code entirely

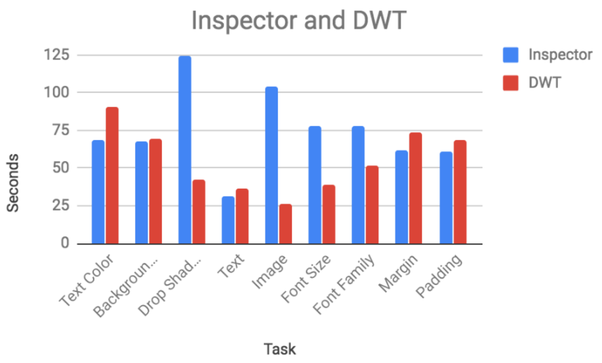


Figure 4. people were faster when they used our tool

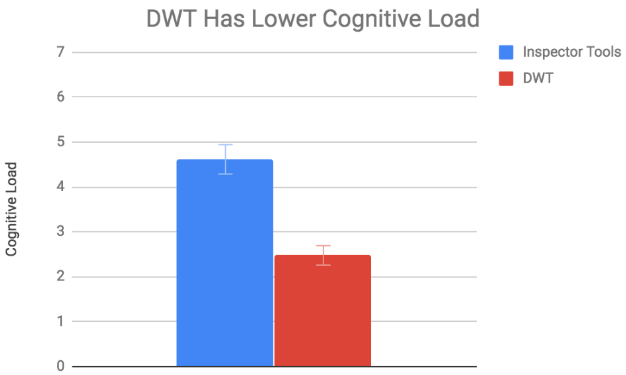


Figure 5. it was cognitively lighter to finish with the XRAY tool

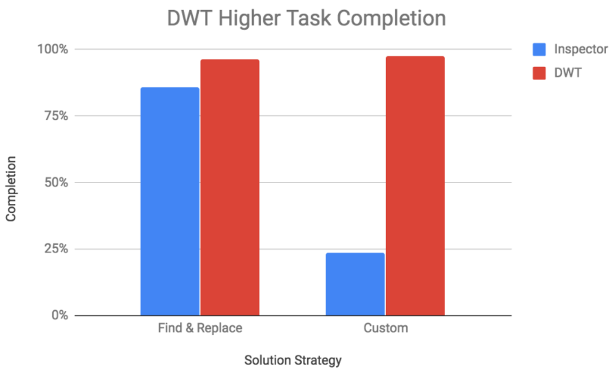


Figure 6. Depending on what kind of solution was required, this tool could be espeically essential.

- lets the developers get back in their element: designers are giving them code to implement instead of vague directions

Discuss harm/risks/dangers/drawbacks...

- ...I need to think about this...

Discuss limitations of our work

- margin/padding issue is not yet solved
- there needs to be more conversation about design systems
- does not support absolute positioning (cannot grab a piece of the page and move it to a different part of the web page)
- we didn't actually implement, but COULD implement some kind of live/real-time collaborative editing thing
- talk about whether or not we are really facilitating experimentation

Discuss future work/room to grow...

- see the limitations and drawback sections above.

REFERENCES

1. Michael Bolin, Matthew Webber, Philip Rha, Tom Wilson, and Robert C. Miller. 2005. Automation and Customization of Rendered Web Pages. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology (UIST '05)*. ACM, New York, NY, USA, 163–172. DOI: http://dx.doi.org/10.1145/1095034.1095062

2. Kerry Shih-Ping Chang and Brad A. Myers. 2012. WebCrystal: Understanding and Reusing Examples in Web Authoring. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, New York, NY, USA, 3205–3214. DOI: http://dx.doi.org/10.1145/2207676.2208740

3. Andrew Couldwell. 2017a. Laying the foundations for system design: Great digital products are built upon solid foundations. *Medium* (Dec 2017). https://medium.com/owl-studios/laying-the-foundations-7e503ef2120f

4. Andrew Couldwell. 2017b. Plasma design system: Creating and documenting a product design system. *Medium* (Jan 2017). https://medium.com/owl-studios/plasma-design-system-4d63fb6c1afc

5. Michael J. Fitzgerald. 2008. *CopyStyler : Web design by example*. Master's thesis. Massachusetts Institute of Technology.

6. Andrew Head and Marti A. Hearst. Tutorons: Generating Context-Relevant, On-Demand Explanations and Demonstrations of Online Code. (????).

7. Joshua Hibschman and Haoqi Zhang. 2016. Telescope: Fine-Tuned Discovery of Interactive Web UI Feature Implementation. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16)*. ACM, New York, NY, USA, 233–245. DOI: http://dx.doi.org/10.1145/2984511.2984570

8. Andrew J. Ko and Brad A. Myers. 2009. Finding Causes of Program Output with the Java Whyline. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09)*. ACM, New York, NY, USA, 1569–1578. DOI: http://dx.doi.org/10.1145/1518701.1518942

9. Ranjitha Kumar, Jerry O. Talton, Salman Ahmad, and Scott R. Klemmer. 2011. Bricolage: Example-based Retargeting for Web Design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, New York, NY, USA, 2197–2206. DOI: http://dx.doi.org/10.1145/1978942.1979262

10. Jon Moore. 2017. InVision Freehand: A 21st Century Case for Digital Whiteboards. *Medium* (Aug 2017). https://medium.com/ux-power-tools/invision-freehand-digital-whiteboarding-bf83639c1184

11. S. Oney and B. Myers. 2009. FireCrystal: Understanding interactive behaviors in dynamic web pages. In *2009 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*. 105–108. DOI: http://dx.doi.org/10.1109/VLHCC.2009.5295287