Social Overlays: Collectively Making Websites More Usable

Tao Dong, Mark S. Ackerman, Mark W. Newman, and Gaurav Paruthi

University of Michigan, Ann Arbor, MI, USA
{dongtao,ackerm,mwnewman,gparuthi}@umich.edu

Abstract. Many small organizations lack the expertise and resources to conduct usability evaluations of their websites. Social Overlays, presented here, is a new system that allows a community of users to collectively improve their website.

Social Overlays enables end–users to identify and repair common user interface problems through creating "overlays" on web pages as part of their regular use, thereby improving usability while reducing the need for professional services. In short, Social Overlays harnesses the diversity of experience and ideas within a community to "crowd source" usability.

To evaluate Social Overlays, we examined whether a group of community members without any usability training could use Social Overlays to identify and repair UI problems on their medium–sized community's website. We found that they could. Community users were able to uncover a large number of UI problems and formulate reasonable solutions to the problems they identified. In addition, we compared Social Overlays to two standard ways of assessing website usability: expert inspection and usability testing. We found that Social Overlays users identified more problems, and their reported problems differed in useful ways from those found by the experts and the usability testing team.

Keywords: Usability, community, peer production, social computing.

1 Introduction

Many small organizations' websites need usability improvement. For organizations like charities, museums, and schools, their constraints on technical resources and usability expertise keep them from sufficiently making their websites easy to use. For example:

- TriCounty GoodDeeds is rolling out its website. But, its web designer and webmaster are volunteers, and there are no spare resources for usability tests. How can they create a web site that is usable and helpful for the charity's constituents?
- The Tree City municipal government's financial services group created a new website for travel reimbursement. It is unusable, since it is full of jargon and appears to be meant for accounting professionals. How can staff in departments help the financial group with their usability?
- The History Department at the University of the Midwest recently created a new website using a content management system. The department has a well-meaning

P. Kotzé et al. (Eds.): INTERACT 2013, Part IV, LNCS 8120, pp. 280–297, 2013.
© IFIP International Federation for Information Processing 2013

web developer, who is an ex–computer science major. The students have to use the website to get information for courses, requirements, and school events, though they often feel the developer doesn't know how they actually use the website. How can the department create a usable, useful website?

At a high level, there are two standard approaches to evaluating usability on websites: usability testing and expert evaluation [13]. While these approaches have many variants, they all share the characteristic that usability experts play a critical role in determining the existence and nature of usability problems. An alternative approach is to solicit problem reports from users. Post–deployment usability approaches focus on collecting feedback from users in the field at the time the problem is encountered [7, 12]. However Chilana et al. found that few usability practitioners analyze or respond to such feedback [6]. Participatory techniques have been proposed for involving users in usability evaluation during the formative stages of system development as well [2, 11], though these techniques still depend on the participation of usability experts and product developers. In this paper, we investigate a novel approach to creating usable interactive systems: enabling community members themselves to collectively improve the system as part of their everyday interactions with it. Our approach is aimed primarily at websites built to serve small–to–medium sized organizations or communities—precisely the sites that often do not have the resources to hire usability professionals or implement frequent changes to a site. Our question, then, is how can we leverage the user community of a website to uncover and service usability problems?

We have designed and built a system called Social Overlays (SO). By harnessing the "wisdom of the crowd," SO allows a community to collectively evaluate and improve their website, without the need for formal usability methods or professional usability expertise. Using SO, community members can create "overlays," which effectively rewrite particular page elements (e.g., text, links, and tooltips), thereby improving the site's usability for subsequent visitors. SO also provides lightweight mechanisms for different community members to nominate potential problems, propose alternate fixes for the identified problems, and vote for the best solution. In short, SO harnesses the diversity of experience and ideas within a community to "crowd source" usability.

While, at a technical level, SO could be deployed on a wide range of sites, our initial focus is on a type of site that is particularly in need of and well-suited to the SO approach. Those are the sites that serve communities or organizations with a few dozen to several hundred members. While such communities are the ones most in need of a low-cost approach to usability improvement, we also expect them to be relatively cohesive and possessing members who have sufficient common ground in vocabulary, practices, and expectations [10]. In these communities, members often know one another, creating the grounds for altruism and self–policing [17]. In a word, those communities possess the desirable social properties that would allow the SO approach to be adopted most effectively.

To examine the viability of the SO approach, we conducted a study with thirteen members of a medium–sized academic community. We found that they were able to find and repair a large number of usability problems on the community's website. Moreover, compared with usability experts whom we asked to evaluate the same site

and an external usability team who conducted a conventional usability test, the community members reported more problems. The problems they found differed in systematic but useful ways from those found using standard usability methods. Thus, our study results argue for the feasibility of the SO approach, at least for small-to-medium sized communities and organizations.

The contribution of this paper, then, is twofold:

1. We present a novel approach to collectively improving website usability by enabling the site's community to not only identify but also repair usability problems.
2. We provide evidence that our approach leads to usability improvements for important types of communities that are comparable to standard but more costly methods such as expert inspection and usability testing.

In addition, we discuss how this approach can be extended for larger or less socially cohesive communities and how this approach is situated in the literature.

2 System Description

The goal of Social Overlays (SO) is to allow users of a website to identify and repair usability problems during the course of their regular use of the site. We identified, through pre-studies, an initial set of requirements that SO must meet in order to accomplish this goal.

1. It should be easy for non-technical users to report and repair problems.
2. When a user is aware of a problem but does not know how to repair it, he/she should be able to request help from other community members or the webmaster.
3. Community members must be able to see the repairs made by other users and decide whether or not they prefer a peer-modified version to the original design of a UI element.
4. Community members and the webmaster need to be able to review and address requests made by other users.

In the rest of this section, we first walk through a scenario that illustrates the use of SO, and then describe the main features of SO.

2.1 Scenario

Chelsea, a master's student at the aforementioned history department, is looking up a class she is considering for next semester. On the page that allows her to search for courses, she finds herself annoyed by the weirdly labeled "Apply" button next to the course search field. She wonders, "Why couldn't they just put the word 'Search' on that search button?" Coming from a literature background, Chelsea often jokes that she is a "language snob."

Fortunately, Chelsea has a new browser extension called Social Overlays that allows her to revise the button's label. She does so, and sees the change immediately.

In more detail, Chelsea chooses the Text tool in the SO panel (see Fig. 1a). Now when her mouse hovers over a page element, it is highlighted with an orange dashed outline (see Fig. 1b). She clicks the "Apply" button that annoyed her on the page to

invoke SO's element editing dialog box, where she enters the text string "Search" to change the button's label (see Fig. 1c). Clicking the "Save" button instantly applies the change for her.

Within an hour, Chelsea finds that her overlay (the alternate label for the button) has received thumbs-up from 5 other SO users in her department. After several days, her department's IT person notified her that the website has implemented her change permanently, because of strong user preference shown on SO.

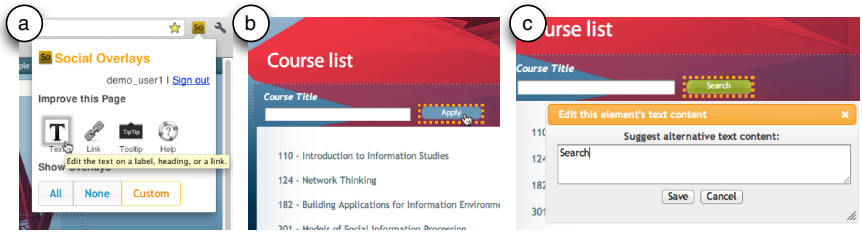

Fig. 1. Social Overlays has a three-step process of making modifications to an existing web page: a) select a modification type, b) select a page element, and c) specify the modification. For example, the user in this scenario renames a mislabeled button from "Apply" to "Search."

2.2 Overlays

In the above scenario, Chelsea repairs an unconventionally labeled button by creating an *overlay* using the Text tool provided by SO. Overlays are user–generated in–place modifications to existing web page elements. There are three types of overlays supported in the current version of SO: Text, Links, and Tooltips. Each type of overlay can be created and edited using tools provided by the SO extension panel (see Fig. 2).

As the scenario above has shown, the *Text* tool allows the user to revise inaccurate or unintuitive terms or languages used on buttons, links, or headers.

The *Link* tool allows the user to add a link to any element of a web page by entering a target URL. This can be useful for creating navigation shortcuts or pointing to additional information that might be helpful for a given task. In addition, the Link tool allows the user to edit the URL of a broken or outdated hyperlink.

The *Tooltip* tool allows users to create or rewrite tooltips (i.e., short messages that appear when an element is hovered over) that are attached to any page element. This tool allows a user to explain or clarify what an element does (or does not do) and how to make use of a feature on the site to his/her fellow users.

The Link tool and the Tooltip tool follow the same workflow as the Text tool to make overlays. A user can combine different types of overlays if needed. For example, the user can modify the text of a hyperlink, and also install a tooltip for it.

2.3 Help Requests

Although the changes that SO currently supports are relatively simple, some users might not feel confident enough to make a change. Additionally, some users might want a more sophisticated change that SO does not yet support. In such situations, a user can request help from the community or the webmaster using the Help button

(see Fig. 2). The Help button allows the user to attach a message to the element that needs to be fixed or improved through a process similar to editing a button's label. If the request is not related to a particular element, the user has the option to submit a general request for the page by selecting the "bucket" at the bottom–right corner.

2.4 Indicators

When a page loads, SO shows page modifications made by the user community on the current page. A modified element will flash for half a second to help a user distinguish it from the original element, unless a particular overlay has been previously approved by the current user or by a certain number of other community members. Inspired by Edit Wear [8], SO uses the side margin of the web browser to provide lightweight visual indicators of prior community activity on the current page (see Fig. 2). Each indicator corresponds to an element that has community–generated overlays or requests, and they are visually aligned on the same line. When the user hovers over an indicator, its associated element will be highlighted. The color of the indicator shows the status of the element. A green indicator signals that there are existing overlays for the element, while a red indicator signals that there are active requests to fix this element. If an element has both overlays and unresolved requests, a yellow indicator is displayed.

If the user wants to see the original version of the element, clicking on the indicator will toggle between the original version and the community–enhanced version. He can also toggle between the original page and the community–enhanced page using the "Show Overlays" switch on the SO panel.

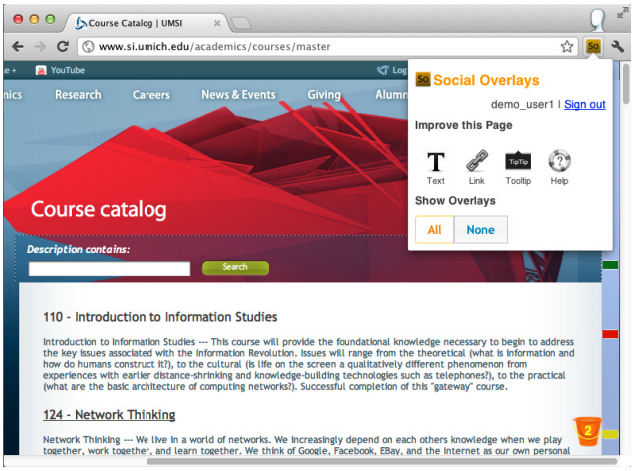

Fig. 2. The Social Overlays browser extension can be accessed by clicking its icon on the browser's toolbar. The icons from left to right represent the Text tool, the Link tool, Tooltip tool, and the Help button. The green, red, and yellow indicators on the right margin are used to indicate overlays, help requests, and both overlays and requests, respectively, made on a page element on the same horizontal line. The bucket at the right–bottom corner is a repository for page–wide issues.

2.5 Selection Rules

When multiple users have defined overlays for an element, one of them must be chosen to display. We implemented a voting mechanism that allows each user to indicate whether an overlay works for him or not. The most approved overlay for an element is automatically displayed unless the current user has approved a different one. In addition, if an overlay's number of votes reaches a community–specific threshold, it will no longer flash to draw the user's attention when the page loads, since it is likely to be working for her as well.

2.6 Implementation Details

We implemented SO as an extension for the Chrome web browser[1] that is coupled to a server–side application. Every time the user visits a new web page, the SO extension injects a set of JavaScript files into the current page to access and manipulate the page's HTML Document Object Model (DOM). With the full control of the target page's DOM, the SO extension enables a set of augmented capabilities within the browser, allowing the user to identify usability problems, suggest localized changes, or make quick modifications to the current page. The SO extension sends the changes made by the user to the SO server's repository of overlays. Upon a subsequent page load, the SO extension retrieves the overlays and requests made by all the community members associated with the current page from the server and applies these changes by modifying the DOM of the rendered web page.

This implementation allows us to gain complete control over the rendering of each web page. It also allows easy porting to extensions for other browsers or a bookmarklet application.

3 Study Design, Site, and Data Collection

After creating Social Overlays, we wanted to know whether it was a feasible approach. To evaluate the feasibility of Social Overlays, we conducted a three-phase study. In the first phase, we determined whether community members could use SO to identify and repair usability problems. We learned that people could use SO effectively, but we could not tell how good SO's results were compared to those generated by standard usability methods. Thus, in the next two phases we compared SO to two standard usability methods: expert inspection and usability testing. The results of all three phases show that SO is a feasible approach to website usability.

3.1 SO Evaluation

First, we needed to know whether community members without training in usability could identify and repair usability deficiencies on their website using SO. Therefore, in the initial phase of our study (which we will call the "SO evaluation"), we asked 13

[1] `http://developer.chrome.com/extensions/`

information science students, none of whom had any formal usability education or experience, to use SO in the process of completing four tasks on the website of their school (and ours), the University of X School of X (UXSX).

We did not choose the website and users as a matter of convenience. UXSX's website had been recently overhauled and had many flaws and problems, thus providing a good testbed. In addition, by using UXSX and its users, we could emulate website users coming to the webpages they use everyday and finding problems based on their normal tasks. Moreover, UXSX is an organization similar to those foreseen as our target communities. It consists of approximately 400 people, including students, staff, and faculty, and is reasonably cohesive socially.

We selected four common information–seeking tasks (see Appendix I) and asked participants to carry out the tasks, using SO to identify and, if possible, repair any issues they encountered as they went along. Each task had known usability issues associated with the website. While the four tasks selected covered only a small subset of the site's possible interactions, they were enough to observe the community process of discovering, noting, fixing and voting on changes to usability problems within a tractable timeframe. We anticipate that, if SO were deployed over a longer time period, a larger number of community members would visit many more pages, ultimately covering all of the most common interaction paths.

To emulate people coming to a web site over time, we ran our participants consecutively. Thus, the overlays (UI modifications) and requests made by a participant were available to all participants after him/her. Participants were first given a demonstration of using SO to fix four example usability problems, and then they were instructed to identify usability problems while solving the four tasks. The participants were asked to state their ideal solution to a problem, and then solve the issue using SO. If a problem could not be repaired by SO's Text, Link, and Tooltip tools, the participant had the option of submitting a request in SO or not doing anything. Each session lasted about an hour and each participant received a $10 coffee shop gift card.

3.2 Expert Inspection and Usability Test

As mentioned, to address whether SO worked as well as standard usability methods, we conducted two additional evaluations. In the second phase (which we will term the "expert inspection"), we asked four usability specialists with at least four years of professional experience to conduct an expert walkthrough (as described in [15]) of the same four tasks that were used in the SO evaluation. The experts were given the same demonstration of SO in the beginning of each session. They were asked to identify as many problems as they could while walking through the four tasks, and envision how SO could be used for implementing their suggested solutions. Each expert was compensated $50 for participating an hour-long session.

In the third and final phase (the usability test), we commissioned a team of external usability evaluators to conduct a conventional lab-based usability test on the UXSX site, and asked them to report the problems they found along with recommendations to address those problems. The usability team consisted of two graduate students with formal training in usability testing and one usability professional with formal training and three years of professional experience.

The usability test followed the standard protocol described in Rubin's widely-used textbook [16], which consists of pre–test and post–test questionnaires, task observation, and debriefing. Eight additional UXSX students participated in the test and tackled the same four tasks used in the SO evaluation and expert inspection. Each session in the usability test lasted about an hour and each participant received $20.

4 Evaluation Results

In this section, we answer the following questions:

- Within our evaluation study, could community members use SO to report a substantial number of usability problems on their website?
- In addition to identifying problems, were community members able to use SO to repair at least some of them in a helpful way?
- How well did the SO approach work, in comparison with expert inspection and usability testing?
- How did community members collaborate informally in using SO to improve their site and benefit from one another's efforts?

In short, is the SO approach likely to be viable?

4.1 Community–Based Usability Improvements

The results of our evaluation show that community members without training in usability can identify a large quantity of usability deficiencies on their website, at least in this community.

In our data analysis, we tallied all problems for which the SO evaluation participants either made an overlay or submitted a help request. We then manually verified these issues on the UXSX website, confirming the existence of identified problems and eliminating duplicates. As a group, they documented 47 unique problems in the process of solving the 4 evaluation tasks. These included issues that could and could not be fixed with SO, but excluded issues that were verbally reported only as well as issues that were similar to those used as examples in the SO demo. In the rest of this subsection, we describe the problems identified by community members in the SO evaluation, and how they as a collective addressed those problems.

Types and Characteristics of Overlays. Using the Text, Link, and Tooltip tools provided by SO, the participants in the SO evaluation made 50 overlays (i.e. page modifications) to address 27 (57.4%) of the total 47 problems they documented. Among the 50 overlays, 10 were alternative text or labels, 11 were hyperlinks attached to existing elements, and 29 were tooltips.

As expected, the Text tool was often used to correct or clarify a link's label. For example, P3 changed a link's label from "Course schedule" to "Course schedule by term," as she thought people might expect the linked schedule to be organized by week. Other uses of the Text tool included replacing an unfamiliar term, correcting typos, and appending a commonly-used acronym.

The Link tool was generally used to shorten navigational paths. For example, five different participants linked 5 static course titles on a degree requirement page to their respective course information pages, after they found it took too many clicks to check course information from the requirement page.

Usage of the Tooltip tool was more varied and interesting. Most of the tooltips were added to links in order to help users decide whether or not to click on them. First, tooltips were created for a link to hint what information could be expected in the linked page. For example, P7 attached the following tooltip to the "guest speakers" link on the Media page: "job candidate talks are accessible through this link." Second, tooltips were used to clarify community-specific jargon. For example, P3 added the following tooltip to explain "faculty guest lectures" as "another way of saying 'job talks'." Lastly, P1 and P10 used tooltips to give specific directions to aid navigation. Fig. 3 shows a tooltip P10 created to direct users to other course schedule viewing options that were hard to find due to the poor information architecture.

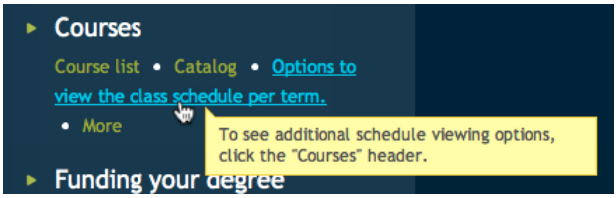

Fig. 3. P10 added a tooltip to indicate that there are other options to view the class schedule (e.g. by week) on the page linked to the "Courses" header.

Help Requests. Though the remaining 20 problems were not "overlaid" with page modifications, they were identified via 33 help requests from the SO evaluation participants. When did participants choose to submit a request rather than create an overlay? We found that 10 of those unresolved problems fell out of the scope of tools like SO. They were related to backend issues, missing content, or missing a feature. Among the remaining 10 unresolved problems, 3 of them could possibly be repaired with the current version of SO, while the other 7 issues could potentially become repairable using an improved version of SO outlined in the Discussion section.

We also examined those requests' quality, placements on the page, and intended audiences. First, close to 90% of help requests included specific recommendations for improving the website, which suggests that participants generally had pretty clear ideas about how the problems they identified could be resolved.

Second, participants appear to be able to find an appropriate element to associate their requests to. Among the 33 requests submitted, 21 requests were attached to a specific page element, while 12 went into the "bucket" as page–wide issues or general comments. Most requests were attached to an appropriate element that helped us understand what the issues were.

Third and finally, many requests appeared to be added in the hope that webmasters would fix them from the server. However, there were also 4 requests intended to go to other community members, asking whether a change was accurate or dividing up the work of making a series of similar changes.

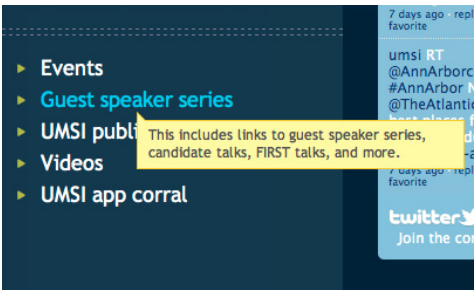

Fig. 4. P1 added a tooltip to the "Guest speaker series" link to indicate that users can find faculty candidate talks in the linked page. This tooltip significantly reduced the number of clicks taken by subsequent participants once they reached this page.

Quality of Community–Generated Usability Enhancements. Did the participants in the SO evaluation make helpful changes? Our analysis suggests that at least some user-generated overlays made subsequent participants more efficient in solving tasks. For example, one of these helpful overlays was P1's tooltip added to the "Guest speaker series" link on the "Events and News" page, as Fig. 4 shows. The tooltip explained what was in the linked page, which was linked to the faculty candidate talks that participants were asked to find in Task 1. The mouse click counts in Fig. 5 shows that Task 1 became substantially easier after P1 created that tooltip.

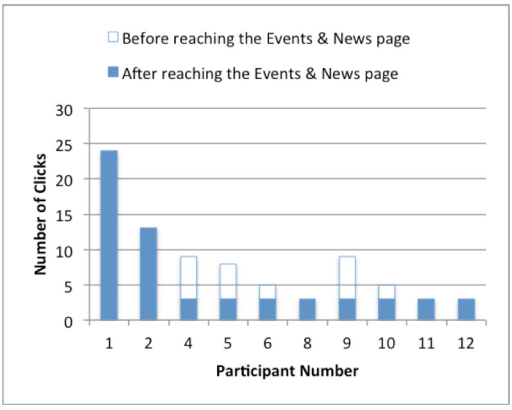

Fig. 5. The number of mouse clicks users made decreased dramatically after P1 added a tooltip to clarify a critical link on the Events & News page. Participants who did not take this route were omitted from the figure.

The participants made helpful overlays to simplify other tasks as well. For example, after completing task 3, P7 linked a course title to a page that provided additional information about the course, and then P12 came across it and said, *"So someone added a link. Oh man, it doesn't tell me if it has PEP credits or not. So someone*

helpfully probably put this link in. [Clicking the link]. That was helpful, that was totally helpful."

However, not all overlays were well crafted. For example, there were tooltips added to clarify the difference between the course list and the course catalog, but apparently these tooltips still lacked detail that would help P12 pick one course page over another. Although the quality of the user–generated usability enhancements varied, they rarely made the website harder to use.

In summary, we found that Social Overlays could lead to a substantially improved website. However, this improvement was not uniform, and much of the improvement showed satisficing behavior.

4.2 Differences between Social Overlays and Standard Usability Methods

To assess and contextualize the problems identified by participants in the SO evaluation, we first compare them with the results of the expert inspection and then with those of the usability test. Specifically, we went through the list of issues reported in the SO evaluation and checked if each of them was covered by the expert inspection or the usability test.

To our surprise, participants in the SO evaluation documented 52% more problems than the 31 problems reported by the expert inspectors. Only 12 out of the 47 problems documented in the SO evaluation were also identified by the experts (see Fig. 6). How could the participants in the SO evaluation find these 35 additional problems? As members of the community, they appear to have advantages in three aspects, based on our analysis:

First, the community members in the SO evaluation leveraged their lived experiences in the organization during problem identification. For example, P3 pointed out that the Tracking and Planning Sheet, a useful PDF file that she could print out and use as a resource in completing task 4, was buried too deeply in the site. She was aware of this problem, because of her prior experience in planning courses as a master's student at UXSX.

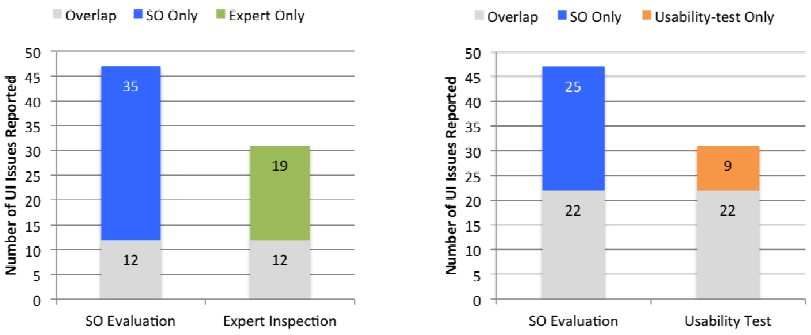

Fig. 6. The above bar charts show that community members using Social Overlays reported not only more but also different usability issues than the expert inspectors and the usability testing team. We describe those differences in section 4.2.

Second, community members uncovered similar problems that occurred in different places in the website, since as a whole, they were exposed to more pages and explored different paths than the group of experts.

Third, the community members were able to project the needs and preferences of the sub–community to which they belonged. For example, two masters' students, P4 and P11, believed that other students would want to find student club events in the school's official Events page.

Nevertheless, the SO evaluation participants did not identify 19 problems that were found in the expert inspection. It appears that the experts were able to spot these additional issues by following general principles and heuristics. For example, the experts were sensitive to inconsistencies on the website such as showing Prof. A's profile on the same page of a course taught by Prof. B. The experts also seemed to evaluate the site following best practices in usability. For example, E1, E2, and E3 mentioned that the website failed to visually separate groups of content on several pages.

Next, we compare the results of the SO evaluation with those of the usability test. The usability testing team reported 31 problems, some of which were broad statements and covered multiple narrower problems documented by the participants in the SO evaluation. In total, the usability test covered 22 of the 47 problems documented by the participants in the SO evaluation (see Fig. 6), representing a larger overlap than that between the SO evaluation and the expert inspection. This increased similarity is not surprising, since usability testing could also access community members' lived experience and local knowledge about the site.

But why did SO users report 25 problems that were not found by the usability testing team? From our analysis, there were two main reasons. First, participants showed empathy with their peers in the UXSX community, pointing out 12 issues that did not prevent themselves from solving tasks but they believed could be frustrating or confusing to less experienced peers. For example, P3 found it a potential problem that the acronym "PEP" was not linked to its full description page, though she already knew what PEP stood for ("Practical Engagement Program"). Second, participants in the SO evaluation pointed out 7 more issues related to missing information or features that they expected to see on the site.

In short, we found that community members reported more problems compared with usability experts and the external usability testing team, and that their reported problems differed in systematic but useful ways from the results generated by the expert inspection and the usability test.

4.3 Community Processes in SO

We also wanted to assess whether SO would show preliminary evidence of community activity. The most important community activity, as mentioned in section 4.1, was getting assistance from the webpage improvements made by other users. We also observed community members giving feedback on the overlays made by others. Six of our participants used the voting feature of SO, and they gave 15 thumbs–up and 6 thumbs–down to overlays made by prior participants. We did not observe conflicting votes on any of these changes. Other users verbally commented on the helpfulness of

other users' overlays, but they did not click the voting button in SO. Four of the 6 thumbs–down were given to point out errors made by P8 and P9, while the other 2 thumbs–down were cast by P7 and P9, when they disliked existing overlays and sought to replace them with one of their own.

We also observed some evidence of social dysfunctionalities that might become more critical in larger communities or with more use. P4 showed self-serving tendencies, declaring, "*I would make that a link, but it would be motivated by my convenience and not out of altruism for other users.*" Another participant did not like all of the changes, but did not revert or modify those changes. (This might also be tacit approval of the change's sufficiency, however.) As well, participants did not vote as much as we hoped. We will return to these issues in the Discussion section.

4.4 Subjective Perception of the Utility of SO

Participants in the SO evaluation commented on what aspects of SO they found particularly helpful. First of all, participants liked the ability to immediately see the change they made using SO. Many of them were delightedly surprised when they saw problems repaired instantly. Second, participants liked that SO provides peripheral awareness of community activities through the indicators on the side margin (see Fig. 2), as P5 mentioned, "*I really like the scroll bar [indicators] that change colors. That's the first thing I looked to, besides the blinking.*" Third, participants appreciated that the changes they made were shared with other community members. P3 enthusiastically commented, "*It can only be in the long term a very big asset to the community, especially the social element ... where people vote up and down changes*."

In summary, the above results of the SO evaluation and our comparisons with expert inspection and usability test show that:

1. With Social Overlays, community users identified a substantial number of problems as they interacted with the site. They were able to repair many of them by creating overlays helpful to others and address the rest by submitting constructive requests.
2. With Social Overlays, community users uncovered problems that existed along multiple paths of browsing and brought to the problem identification and solving process their lived experience and ideas that only members of the community would have.
3. With Social Overlays, community users seemed to take on a more active role in reporting UI problems than the participants in the usability test.

Our findings argue that Social Overlays is useful and feasible for communities, as least those similar to UXSX in size and cohesiveness, to collectively identify and address usability problems on their website. Moreover, the results of using SO were comparable to standard methods that are more expensive, such as expert inspection and usability testing, though each of the three approaches showed different strengths and priorities in our study.

5 Discussion

To summarize, this paper:

- Introduced a new community–based approach for addressing web usability problems without requiring expertise in usability evaluation or interface design.
- Presented a new system called Social Overlays that embodies this approach of collaborative usability improvement and demonstrated its technical feasibility.
- Provided an evaluation that showed that Social Overlays and the approach it enables are useful and feasible for an important set of communities and websites.

Despite SO's success, we found five issues with it and the approach in our evaluation. First, our evaluation showed that community users grounded problems in their lived experience inside their community, covered a larger number of web pages, and made quick responses to UI issues. This, we have argued, is critical for many organizations without access to usability professionals. However, we also saw that users using SO might need help formulating holistic and broad redesign recommendations. An interesting question we would like to explore in future work is how SO can be used with some lightweight expert involvement, e.g. with an expert synthesizing or gatekeeping modifications proposed by users.

Second, although our evaluation argued that SO was likely to be helpful for communities with a few dozen to several hundred members, there are many other types of communities with many more members, less social cohesion, and potentially less trust. Addressing these communities will require additional functionality. Some public-facing websites are subject to spam, problematic content, and attacks, requiring more security mechanisms. Larger or more public websites and communities may suffer more motivation issues, although they also potentially have more users for SO. All of these issues remain to be addressed in future work, and we plan to do so in subsequent efforts. Nonetheless, we were pleased that SO appears to be able to work in at least an important subset of small-to-medium communities. The same type of community exists in many educational settings, community organizations, charities, and the like. Nonetheless, we will not know all the issues until this tool is released into the wild and used by other sites and communities.

Third, the current version of SO provided only three simple modification tools. Even with these limited capabilities, our participants were able to address common and important usability problems with ease: confusing labels and vocabulary issues, broken and cumbersome navigational paths, and unclear site functionality. We believe, based on our evaluation, that additional tools would make SO even more helpful. We are currently investigating capabilities to insert new links into a page, convert the highlighted part of text into a hyperlink, and support the navigational use of tooltips with better hyperlinking, and address higher-level issues of page and site organization. We are also examining how to address potential information overload problems that would arise if SO were to scale, including new awareness visualizations and mechanisms for issue consolidation.

While it is technically possible to include more powerful and sophisticated tools such as a full-fledged style editor, we believe this would run the risk of having naïve

users freeze or make too many mistakes. We also plan to examine this risk in future studies.

Fourth, based on even our limited study, we suspect there will be resulting problems as SO use grows in a site. As the number of users grows, it will increasingly be a concern that different subcommunities inhabit different language worlds, and individuals may wish to place markers for themselves that would be inappropriate for the entire community. We are currently investigating mechanisms to allow both subcommunities and individual personalization.

Finally, as mentioned, we had less usage of the voting feature than we had hoped. In our observations, non-voting behavior often implied agreement, and there were very few overlays or requests that were perceived negatively by participants. We expect both the use and the utility of the voting feature to increase as more users participate in an SO deployment; however, we are investigating new user interface designs to make voting easier and more meaningful.

We must also acknowledge the many limitations in the current study. Our tasks were artificial and scoped, as with any lab–based study. As well, the study was short–term and used selected pages. As mentioned, we will not know all the issues until SO is released into the wild and used by other communities—work that is presently ongoing. The current system and evaluation reported here, however, has led us to believe the usefulness and stability of SO are such that such explorations can fruitfully occur.

6 Related Work

The SO system and the approach it enables, as presented above, aim to tackle the challenge of post-deployment usability which is important but has been undervalued in the user-centered design practice [6, 19]. A number of commercial offerings have sprung up that facilitate the collection of user feedback on deployed websites, either through web forms (e.g., Feedback Army[2] and UserVoice[3]) or message boards (e.g., SuggestionBox[4]). Additionally, web annotation systems (e.g., Diigo[5] and AnnotateIt[6]) could potentially be employed to obtain user feedback, though most of them are focused on learning and information management. It is evident that, while being reasonable solutions for collecting user feedback, all of these tools require users to depend on website supporting staff to assess the severity of the problem, design a solution, and implement the change. In contrast, SO addresses usability by enabling a website's user community to not only identify problems but also repair many of them immediately.

To design a system like SO, we draw on three bodies of technical work: *web-rewriting systems*, *collaborative web accessibility tools*, and *community-based help systems*. We will describe each of them in turn.

[2] http://www.feedbackarmy.com
[3] http://www.uservoice.com/
[4] http://www.suggestionbox.com/
[5] http://www.diigo.com/
[6] http://annotateit.org/

First, we build upon *web-rewriting systems* that allow users to alter the design or behavior of a webpage at runtime. General-purpose web-rewriting systems like Chickenfoot [4] and GreaseMonkey [14] support such rewriting, and they have online repositories whereby users can share, discover, and make use of scripts created by others. However, the versatile capability of these systems creates difficulties for most users, since many users do not know how or want to inspect, edit, and debug scripts. SO leverages some of the same technologies used in web rewriting systems (e.g., JavaScript injection), but SO nonetheless has been designed specially to allow naive users to contribute.

Second, collaborative web accessibility tools (e.g. AccessMonkey [3] and Social Accessibility [18]), represent a specialized class of web-rewriting systems—one that shares SO's high-level goal to make websites easier to navigate and use. Accessibility systems, though, are focused on helping particular subsets of users who share particular disabilities. For example, the Social Accessibility system [18] allows volunteers to enter missing metadata that can then be consumed by visually impaired users who subsequently visit the augmented sites using screen readers. Though the underlying technologies that enable SO and Social Accessibility are similar, the sociality afforded by the SO approach is fundamentally different, because in SO both producers and consumers of collaborative usability enhancements belong to the same group. Our study has shown that many usability problems can best be identified and addressed by users who share lived experience relevant to the community of the site.

Therefore, the sociality supported by SO is closer to that studied in the third body of work we draw upon: *community-based help systems.* These systems assist users by providing information generated by other members of the community. Community-based help systems, such as Answer Garden [1], QuME [20], IP–QAT [9], and LemonAid [5], have focused on making it easier for users to ask questions and find answers, and these systems have shown resoundingly that users can collectively create useful information. Answer Garden builds a store of questions and answers, but accessing it requires using a separate system, resulting in a potential distraction from the core tasks users are trying to accomplish. IP-QAT and LemonAid associate questions with related UI elements, and display answers as part of the system for which help is sought. IP-QAT and LemonAid, however, do not seek to actually improve the user experience of the system by effecting design changes, as SO does, which has the additional benefit of reducing the need for seeking help in the future.

In short, none of the above systems are focused on post-deployment usability assessment and correction, though we draw inspirations and learn lessons from them to inform different aspects of SO design. By specifically designing for the collaborative work around post-deployment usability, Social Overlays provides new capabilities of community interaction and assistance.

7 Conclusions

In this paper, we presented the Social Overlays system (SO) and its evaluation. SO enables end–users to identify and repair common user interface problems on a website

by making "overlays" that rewrite specific page elements. Moreover, SO displays page modifications to others who also have this extension installed.

We also reported findings from an evaluation of Social Overlays. In our study, we found that a group of community members without any usability training could use SO to identify and fix UI problems on their medium–sized community's website. We also compared the results from those community members to the assessments of usability experts as well as the results of a usability test, and found that SO produced a larger number of issues, and that these issues partially overlapped with those found through standard methods. To summarize, we found that Social Overlays can be a viable approach. By having users improve a website's usability as a part of their use of that site, a community can collectively make their website more usable.

Acknowledgements. This work was supported in part by the National Science Foundation (IIS 0905460), the University of Michigan Rackham Graduate School, and kynamatrix Research Network. We also thank Sonali Mishra, Pei-Yao Hung, and Chuan-Che Huang for their help with this study.

References

1. Ackerman, M.S.: Augmenting the organizational memory: a field study of answer garden. In: Proc. CSCW 1994, pp. 243–252. ACM, New York (1994)
2. Bias, R.G.: The pluralistic usability walkthrough: coordinated empathies. In: Nielsen, J., Mack, R.L. (eds.) Usability Inspection Methods, pp. 63–76. John Wiley & Sons, Inc., New York (1994)
3. Bigham, J.P., Ladner, R.E.: Accessmonkey: a collaborative scripting framework for web users and developers. In: Proc. W4A 2007, pp. 25–34. ACM, New York (2007)
4. Bolin, M., et al.: Automation and customization of rendered web pages. In: Proc. UIST 2005, pp. 163–172. ACM, New York (2005)
5. Chilana, P.K., et al.: LemonAid: selection-based crowdsourced contextual help for web applications. In: Proc. CHI 2012, pp. 1549–1558. ACM, New York (2012)
6. Chilana, P.K., et al.: Post-deployment usability: a survey of current practices. In: Proc. CHI 2011, pp. 2243–2246. ACM, New York (2011)
7. Hartson, H.R., Castillo, J.C.: Remote evaluation for post-deployment usability improvement. In: Proc. AVI 1998, pp. 22–29. ACM, New York (1998)
8. Hill, W.C., et al.: Edit wear and read wear. In: Proc. CHI 1992, pp. 3–9 (1992)
9. Matejka, J., et al.: IP-QAT: in-product questions, answers, & tips. In: Proc. UIST 2011, pp. 175–184. ACM, New York (2011)
10. McGrath, D., Spear, M.B.: The Academic Crisis of the Community College. SUNY Press (1991)
11. Muller, M.J., et al.: Methods & tools: participatory heuristic evaluation. Interactions 5(5), 13–18 (1998)
12. Nichols, D.M., et al.: Participatory usability: supporting proactive users. In: Proc. CHINZ 2003, pp. 63–68. ACM, New York (2003)
13. Nielsen, J.: Usability Engineering. Morgan Kaufmann Publishers Inc., San Francisco (1993)
14. McFarlane, N.: Fixing Web Sites with GreaseMonkey, `http://www.linuxjournal.com/article/8215?page=0,0`

15. Oppermann, R., Reiterer, H.: Software evaluation using the 9241 evaluator. Behav. Inf. Technol. 16(4-5), 232–245 (1997)
16. Rubin, J., Chisnell, D.: Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Tests. Wiley (2008)
17. Sproull, L., et al.: Connections: New Ways of Working in the Networked Organization. MIT Press (1992)
18. Takagi, H., et al.: Social accessibility: achieving accessibility through collaborative metadata authoring. In: Proc. ASSETS 2008, pp. 193–200. ACM, New York (2008)
19. Wagner, E.L., Piccoli, G.: Moving beyond user participation to achieve successful IS design. Commun. ACM 50(12), 51–55 (2007)
20. Zhang, J., et al.: QuME: a mechanism to support expertise finding in online help-seeking communities. In: Proc. UIST 2007, pp. 111–114. ACM, New York (2007)

Appendix I: Evaluation Tasks

1. Imagine you're a UXSX faculty member who has traveled a lot recently. As a result, you missed several faculty candidate talks. Now you want to find the video recordings of the following candidates' talks on the UXSX website: Jiang Chen, John Smyth, and Amit Gupta [names changed for publications]
2. Imagine you're a master's student at UXSX who stays in town this summer. You'd like to find out what's happening at UXSX during the summer.
3. Imagine that you're a new master's student. As part of the program requirements, you need to earn a certain number of PEP (Practical Engagement Program) credits. You're thinking about taking SX 622, so you'd like to find out how many PEP credits SX 622 offers.
4. Imagine you're a second–year master's student in the Human Computer Interaction (HCI) specialization. You're in the process of planning coursework for next semester. You'd like to find three HCI elective courses you're interested in and find out who is teaching each of these classes.