Annie Gesellchen
Ben Celsi
Project 1: Pitch

Bug Reporter Edge Extension

THE RELATIVE COST OF FIXING A FLAW AT DIFFERENT STAGES OF THE SDLC
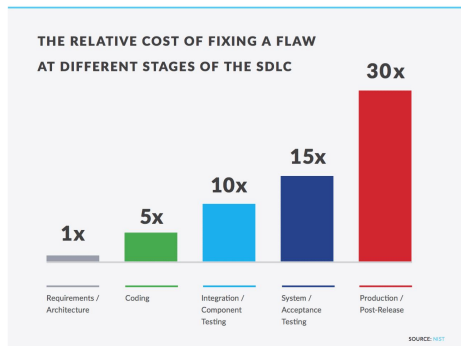
Motivation
Users encounter bugs in the websites they visit every day, but reporting these bugs can be difficult and not worth a user's time; thus, they rarely report these bugs. Instead, bugs often go unnoticed until developers find it themselves, which means that bugs generally affect users for a long time before they're fixed. Website developers find bugs via dogfooding, error logs, and the occasional user report. This method finds big bugs quickly, but is much slower at detecting minor bugs that affect a small subset of users. Our goal is to reduce the amount of time that bugs go unnoticed by making bug reporting easy for users.

As an example, take a TV streaming platform such as Netflix. Imagine that one episode of a show won't play for some reason. It could take the website developers weeks, or months, to realize on their own that one episode of one season of one show amongst hundreds or thousands of shows and movies doesn't work right. Most users, who aren't watching the show, would never know; but to the few who are, this could potentially be a frustrating, long-term bug. With a way to easily report a bug, users are more likely to log this error, and the streaming service will be notified and take action far faster than they would have otherwise.

Easy user error reporting will improve software quality, decrease cost, and improve user experiences with websites by fixing annoying, and possibly costly, bugs in the codebase. As seen in the figure above, bugs that make it to production cost approximately 30 times as much as a bug caught during the requirements stage; a tool that allows users to seamlessly report a bug without going too far out of their way to do so will decrease the cost of the bug as well as improve the lives of both users and developers.

Approach

Our solution is an Edge browser extension that allows users to easily report bugs they encounter. The extension lets users describe the bug and optionally upload a screen capture of it. The extension may also automatically store some information about the state of the site at the time of the bug (such as error logs). This information is then stored, and sent to companies (either automatically or by request.)

Previous browser extensions have been directed towards developers and clients who are purposefully bug testing, not towards general users. Additionally, there aren't any options for Edge at the moment; most options are for Chrome.

One challenge of this approach is a lack of motivation for most users to report bugs. A browser extension may make it easier to report a bug than the standard way (sending an email), but there is still fundamentally a lack of motivation for most bugs. However, while the majority of users might not be motivated, this extension would allow people who enjoy finding bugs an easy to use, centralized platform to report these bugs, which will hopefully encourage them to report bugs more often.

Limitations to our approach include:
- Not being able to take screenshots directly (Edge doesn't allow this; users will have to upload any screen shots that they take)
- Developers will likely have to request data; we have no good way to send collected user error reports to developers directly
- It may not be in-scope to capture details about the state of the website (such as error logs), depending on Edge's extension API.

Challenges and Risks

The single greatest challenge to developing this product on-schedule is possible limitations of the Edge extension API. It could potentially limit the data that we can collect (ex.: error logs), or cause issues when interfacing with the back-end. As it's a newer browser, there isn't as much documentation or information about creating an extension or its limitations.

Another challenge is filtering valid bug reports from reports that are unhelpful (although, this could potentially be something handled by the companies rather than our software)

Risks:
If users are taking screenshots of bugs, there is a potential security risk that users will take screenshots that include sensitive information. A way to protect against this would be to remind users not to post sensitive information when uploading a screen capture.

Hours: 3