Abstract

Software developers spend a significant proportion of their time searching for and making sense of complex information online in order to accomplish their goals, whether it is comparing and choosing different APIs, adapting code snippets found on the Internet to meet their needs, or trying to learn unfamiliar code to fix an error or add a new feature. While there have been many tools proposed by the research community to support the process of finding these relevant information, in many cases, the answers to developer' questions involve trade-offs among multiple valid options and not just a single solution. Indeed, prior work as well as research conducted in this thesis have pointed out that developers express a desire for help with decision-making and navigating through the complex information space to understand trade-offs.

Furthermore, after each sensemaking episode in which a developer collects, synthesizes, and gains information and knowledge for themselves, their work is essentially lost, with no one else benefiting, resulting in the challenges that: 1) without proper documentation, it is hard for future developers to understand the underlying design rationale: what options were considered, what criteria or constraints should be met, what the resulting trade-offs are, and what was deemed to be the most important and why; 2) subsequent developers who may share similar needs and goals would have to start their sensemaking from scratch rather than reuse the knowledge summarized by previous developers.

In this thesis, I[1] explore interactive systems that scaffold developers' sensemaking and decision making processes on the web. I hypothesize that tools designed to help a developer capture information and make decisions about trade-offs can provide crucial benefits for both the developer and others who want to understand their design rationale. To probe this hypothesis, I designed the Unakite system, which enables developers to collect, organize, and keep track of information about trade-offs and build a comparison table, which can be saved as a design rationale for later use. To address the potential high cost of manually collecting and organizing information, I built Crystalline, which leverages natural language processing (NLP) and passive behavioral signals such as mouse movement and dwell time to automatically collect and organize information into tabular structures as a user searches and browses the web. To facilitate the evaluation and reuse of previous design decisions and rationale in such tabular formats, I developed both a framework and the Strata system that collects and visualizes key signals about the context, trustworthiness, and thoroughness of previous design decisions for subsequent developers to explore and evaluate.

Although my existing work found that knowledge organized in a tabular structure (e.g., a comparison table) is appropriate and effective for both organizing information while searching and browsing and reading and digesting later by someone else, empirical research has suggested that a table might not always be the most efficient organizational structure to use, especially during early stages of sensemaking when people do not necessarily have a fixed mental model, or their mental model might change rapidly as new information comes in. Therefore, in my proposed work, I would like to explore alternative knowledge organizational structures (e.g., lists, mind maps, affinity diagrams, etc.) that would support developers' sensemaking processes. I

[1] Although this thesis proposal is based on research projects that I have personally led, this document predominantly contains the pronoun "we" out of respect to all of my collaborators who have contributed to the research.

would like to leverage the intelligent techniques and recent advances in machine learning and NLP to 1) minimize the cost of generating and curating these structures on the go and 2) facilitate fluid transformations between these different structures. Similar to my previous work, I plan to evaluate this new system through a series of lab and field studies with developers solving their real-world problems.

The series of work introduced in this thesis points to the importance of having tool support that helps developers efficiently organize and manage information as they find it in a way that could also be beneficial to others, and therefore bootstrapping the virtuous cycle of developers being able to build on each other's sensemaking results, fostering efficient collaboration and knowledge reuse.