Thesis Project Portfolio

Software Development Internship Experiential Learning Takeaways and Observations

(Technical Report)

Investigating the Effect of Of Actors Within the Scrum Methodology

(STS Research Paper)

An Undergraduate Thesis

Presented to the Faculty of the School of Engineering and Applied Science

University of Virginia • Charlottesville, Virginia

In Fulfillment of the Requirements for the Degree

Bachelor of Science, School of Engineering

James Nathan Barnette

Spring, 2024

Department of Computer Science

Table of Contents

Sociotechnical Synthesis **2**

Software Development Internship Experiential Learning Takeaways and Observations **5**

Investigating the Effect of Of Actors Within the Scrum Methodology **9**

Prospectus **24**

Sociotechnical Synthesis

Technical Report Summary

During my time at MessageGears as a software development intern, I gained valuable insight into how modern day software development teams function and work together. MessageGears is a midsized SaaS tech startup in the marketing and advertising space, whose main product is a marketing communications service which large logo brands use to send personalized advertisements and messages to consumers at scale. I contributed to their product development pipeline working on many subtasks as part of larger product initiatives. My work initially focused on fixing bugs in the backend, coded in Java. I also learned how to write automated tests on my written code, working in multiple frameworks like JUnit, Mockito, and Groovy. As I took time to learn about the codebased I eventually gained responsibility to work on the Angular and React frontend as well, becoming a full stack contributor. Working on front end technologies required me to learn TypeScript, Selenium, HTML, and CSS styling as well. I contributed roughly 1800 lines of production code shipped in Summer's 2022 product release and gained additional responsibilities in 2023 to help plan the intern program. My technical report details all of these contributions in full and explains what technical abilities I gained during employment.

While I learned many useful skills during this internship, compared to my academic studies the main source of academic outgrowth I received from this experience was my exposure to agile development methodologies. These methodologies dictated how our team interacted with each other and what our day to day schedule looked like. My team worked in a Scrum environment, a type of agile methodology which timeboxed a set amount of work for our team to complete into two week cycles. I learned a lot about the Scrum workflow and got better at it as

time went on. However, even though I saw merits in using this methodology, I still felt as though our team was wasting time and resources as a whole.

STS Paper Summary

Deficiencies in the workflow processes used at MessageGears led to frustrating decreases in overall productivity, causing me to be interested in the workflow methodologies themselves. Scrum, characterized by its iterative, flexible approach and minimal upfront research, has become the dominant framework in modern software development due to its adaptability and efficiency. However, despite its popularity, Scrum faces criticism for potential drawbacks such as increased risk of employee burnout, unpredictability in project timelines, and less thorough documentation which can hinder long-term project sustainability (Agile, 2022).

To explore the benefits and drawbacks of Scrum, an Actor Network Theory analysis into the innermost workings of a Scrum network provides context by showing the power dynamics and communication channels between actors in Scrum. This analysis shows that the interactions between various human and non-human actors within the Scrum framework—such as the Scrum Master, Product Owner, Development Team, and the artifacts and tools they use—help shape the development process. The way these actors negotiate and align their goals is crucial for the smooth functioning of the Scrum process. For instance, the Product Owner's priorities must be clearly understood and accepted by the Development Team, and the Scrum Master plays a key role in facilitating this understanding (Sutherland, 2014).

A successful implementation of Scrum requires careful management of team dynamics and continuous adaptation to project demands. Shortcomings in communication and alignment can lead to inefficiencies and dissatisfaction among team members. Enhancing the integration

and functionality of non-human actors, such as project management tools and documentation practices, can significantly improve the effectiveness of Scrum teams. A thorough understanding of the actor-network within Scrum environments, combined with improvements in team dynamics and tool integration, can lead to better software development practices. In promoting successful agile development methodologies, this analysis hopes to improve overall software development efficiency and effectiveness as well.