Takeaways from an Internship; How to Expand Upon Learned Skills to Full-time Employment

A Technical Report submitted to the Department of Computer Science

Presented to the Faculty of the School of Engineering and Applied Science
University of Virginia • Charlottesville, Virginia

In Partial Fulfillment of the Requirements for the Degree
Bachelor of Science, School of Engineering

Sarah Gould

Spring, 2023

On my honor as a University Student, I have neither given nor received unauthorized aid on this assignment as defined by the Honor Guidelines for Thesis-Related Assignments

Briana Morrison, Department of Computer Science

Takeaways from an Internship; How to Expand Upon Learned Skills to Full-time Employment

CS4991 Capstone Report, Fall 2022
Sarah Gould
Computer Science
The University of Virginia
School of Engineering and Applied Science
Charlottesville, Virginia USA
sg4sgz@virginia.edu

Abstract

As an intern satisfying a software development role, my driving force was to create a better user experience by enhancing a software feature that allowed users to create data tables from scratch with ease. I utilized my ability to think critically and problem-solve efficiently coupled with my social skills to make the greatest impact on my team and contribute properly. More specifically, my prior technical skills, such as my familiarity with codebases, various coding languages and technical problems served as a solid base of understanding to build from when paired with my eagerness to learn more. Beyond just technical skills, my soft skills included experience working in large groups, the ability to receive feedback and to network with others. I completed a number of influential tickets for the product and closed a number of pull requests that got merged into the product that was pushed out to the users. To build *upon my experience* the next steps would be to apply the knowledge learned and the skills gained to future ventures, specifically any full-time jobs. Furthermore, I can apply to a hybrid workplace making use of my ability to be successful while fully remote.

1. Introduction

The basis of my internship and the problem to be solved was building and enhancing a low-code platform for users. Users can consist of: individuals, groups, or companies. The goal is to provide a versatile and diverse application, in terms of who the intended user would be and its functional capabilities. To further define what a low-code platform is, low-code is the idea that creators can develop software and applications with little to no coding knowledge or application. It removes the need for traditional coding and creates a faster and easier experience for any user to create software or application from scratch.

The purpose of providing such a product is to make software development more feasible for general users and to expand the capabilities of many companies. In our modern age of technology, the Internet and computers, both software and hardware are so involved in everyday life and the ability to build products incorporating those aspects of technology is highly advantageous. Therefore, removing a time-consuming software development step and providing the user with the building blocks to do so significantly advances efficiency for many users and groups.

As an intern working on the feature enhancements to this low-code product, it was important to have the capabilities of traditional coding and understand the traditional software development lifecycle. My role in the development of the process was being the back-end producer of the building blocks given to the user. More specifically, I worked on providing functionality to a low-code data modeling

feature that allowed users to create a data table, connected to a database, solely from scratch.

2. Related Works

Waszkiowski (2019) breaks down the importance of low-code platforms and its significance to businesses. He details the foundation of the low-code ideology and its primary purposes and focuses. The particular example described was focused on automating processes, which differs from my database-based product, did not emphasize solely automation but focused on a unified low-code platform that incorporated work flow, automation and process mining.

Sahinaslan, et al. (2021) investigates the role that low-code platforms have played in satisfying the demand for software development applications. In response to the exponential growth of technologies and more specifically software, low-code platforms are an inexpensive and flexible solution to meet those demands. The publication focuses on a singular low-code platform, SetXRM, to understand the benefits of low-code platforms in the realm of software development.

3. Process Design

Similar to any other internship throughout my experience, I was challenged to understand the innerworkings of the company and the typical task process from start to finish. Being aware of these expectations and the structure of the platform is essential to being efficient and successful in the workplace. To provide greater context to the code structure and the software development life cycle, the following will describe those topics.

3.1 Review of System Architecture

The internal process within the design of the products within the company was characterized as a monolithic architecture. In the world of computing, two main architecture forms exist: microservices and monolithic architectures. A monolithic architecture is a traditional model for software development, in which the product scales on one dimension. I completed the work on the internship product in a phased product-release schedule. All the groups within the company had a similar goal and worked on individual portions to then send out jointly to the users.

3.2 *Key Components*

Typical patterns of work exist from end-to-end of a product release. Each new feature begins with critical planning and the creation of tickets to complete that feature. Following the ticket creation, the components of the feature are broken down into sprint-wise pieces, with each piece being manageable for completion within a two-week timeframe. For each specific ticket, there is a general development lifecycle. A user-experience designer will construct the vision for that feature enhancement and then the software developer takes on completing that vision. To ensure clean and concise code, code is peer-reviewed and a quality engineer is tasked with verifying the feature enhancement works as expected without any additional bugs.

The specific feature that was the focus of my internship was focused on enabling users to create data tables from scratch. My work included user experience changes, feature enhancements and error handling.

3.3 C*hallenges*

The largest obstacle to overcome while working through the code platform was understanding the code structure of the large monolithic architecture of the system. The company had its own proprietary language that was essential to all of the front-end work to be completed for the product. Understanding how to navigate and locate the

necessary files was a challenge that can be expected when entering any new software development position. To overcome these challenges, it was vital to maintain a positive mindset and to recognize that asking questions is not a bad thing. As an intern, it is important to gain as much knowledge as possible and to ask clarifying questions in order to do so.

4. Results

From the work completed during my internship, the user experience of creating a data table from scratch has been made cleaner and more straightforward to understand. By implementing tooltips and adding error-handling dialogue, the user can have an overall better experience when encountering any sources of confusion. The functionality of the feature has also been amplified, allowing for a greater ability to create relationships and make connections between different data tables. Prior to the feature, creating a many-to-one relationship between different tables took a substantially longer time; now the user can do so instantaneously by clicking a single button. The overall goal of the low-code product is to minimize the brunt work of computing for the user and allow them to create a seamless application quickly.

5. Conclusion

Low-code platforms have positively changed the world of software development, allowing any individual or group the ability to create a comprehensive application without any prior programming knowledge. The specific feature within my company's low-code platform I worked on is enhancing the user experience and adding greater capabilities with each quarter release. The time required by a user to create a data table, with the desired functionality, was previously extensive and could warrant inconsistencies. Following my summer internship, this feature is now more robust, with more relationship abilities and an overall elevated user experience.

Moving forward, my understanding of this platform and the non-technical skills learned will be beneficial when implementing that knowledge in my future full-time employment. Following graduation, I will be working for the same company and furthering the development of this low-code platform. Specific technical skills, such as prioritizing the understanding of the code structure and the company's software development lifecycle, will be important takeaways within any future software endeavors. In addition, knowing how to work in a team environment fully remote and transitioning that knowledge to a hybrid workplace will be a crucial transition, if not an easier one. Altogether, the technical and non-technical skills learned throughout my internship will be significant takeaways for any full-time employment opportunities.

6. Future Work

To expand upon the work completed during this internship, any functional enhancements or additions to the feature to make it more user-friendly would be useful. One such example would be to offer larger support with third-party database platforms within the low-code platform. Doing so will diversify the customer's abilities to use existing database tables. Additionally, enhancing the platform to provide more user clarity, such as with more tooltips, more documentation, or providing alternatives for the user once an error is encountered, are all options to allow users without any prior computing knowledge to successfully understand and make use of this platform. Simplifying the user experience and making the process of creating a data table from start to finish as efficient and as straightforward as possible is the overall end goal.

References

Waszkowski, R. (2019). Low-code platform for automatin business processes in manufacturing. *IFAC-PapersOnLine, 52*(10),376-381. https://doi.org/10.1016/j.ifacol.2019.10.060

Sahinaslan, E., Sahinaslan, O., and Sabancioglu, M. (2021). Low-code application platform in meeting increasing software demands quickly: SetXRM. AIP Conference Proceedings, 2334(1), 070007. https://doi.org/10.1063/5.0042213