Agile and NOVA Web Development

Jeffrey Elkner

University of the People

BUS 5114: Management and Information Technology

March 15, 2023

Agile and NOVA Web Development

While the task of this case study was to pick a company and describe how it effectively uses agile software development methodologies, I hope you will indulge me yet again in redefining the task (in the best agile spirit) a bit. I have long been a free software activist, and founded a web development worker cooperative, NOVA Web Development, back in 2012, with the goal of establishing a viable business developing free software (NOVA Web Development, n.d.). The cooperative has failed in several attempts to become a viable business, with "viable" defined as producing enough revenue to sustain its members, most recently last year (Jelkner's Weblog, 2022), but it refuses to die. At the beginning of this school year a foolhardy group of my senior project students convinced me that they wanted to take a try at keeping the cooperative going, so we are at it once more. Now that we are back in the game, the role that best practices and agile development methodologies will play in our latest reboot becomes a concrete concern. I was long ago convinced of the benefits and importance of agile methods, and for the remainder of this paper I will argue why I believe that we will need an effective agile strategy if we are going to have a shot at making NOVA Web Development succeed this time.

With a new group of young developers taking the reins, answering the question of how to effectively turn over development of our two main in-house projects, LibreOrganize (LibreOrganize, 2020) and Business Tracker (Business Tracker, 2020), becomes a central task, and the role best practices and agile software development methodologies will play in this is of particular importance. The quality and

maintainability of software tends to degrade over time. This phenomenon, known as code rot, has several causes which need to be actively and continuously fought. As Robert C. Martin, known as "Uncle Bob", says in the first episode of his Clean Code video series, "Here's the bottom line, you can not deal with a mess by running away from it. The only way to effectively deal with that mess is to turn around and face it, and find the rot and clean it." (2011, February, Clean Code, Timestamp 21:30). The new NOVA Web Development team, if they are going to be able to keep our systems clean once they start working on them, will have to learn best practices and begin implementing them as soon as they can. They will have to learn to recognize the four "code smells" identified by Uncle Bob as characteristic of code rot: rigidity (Timestamp 23:10), fragility (Timestamp 26:32), inseparability (Timestamp 29:53), and opacity (Timestamp 33:13). They will need to understand that "making messes is not the way you go fast, it's the way you go slow" (Timestamp 36:48), and that "if you want to go fast, you have to stay clean" (Timestamp 37:04).

In order to keep the code base clean, they will need to be able to use the best practices that Uncle Bob lays out in the Clean Code video series, including how to effectively use names to produce self-documenting code (Martin, R. C., 2011, February, Names++), how to keep functions small so that the code "reads like well written prose" (Martin, R. C., 2011, April), and how to use automated tests written *before the code* in a process called test-driven development to give them courage to make changes to the code (Martin, R. C., 2011, September).

The developers that originally created LibreOrganize and Business Tracker had studied agile practices and tried to apply them during development. Both applications have automated tests, and both are written with flexibility, robustness, separability, and transparency as goals. Both applications have achieved minimum viable product status, with LibreOrganize in use by several NOVA Web Development customers, and Business Tracker used by the cooperative itself to manage its business information, dogfooding its development (Chen, V., 2020, August 20).

For the applications to move forward, we will need to go from minimum viable product to minimum lovable product (Babych, M., 2021, December 8), continuing to add features our customers need in agile response to their requests. We can only do this effectively if we can keep our code from rotting.

The agile software development approach is fundamentally about values rather than a specific set of procedures. It is an approach which, as stated in the *Manifesto for Agile Software Development*, values individuals and interactions over processes and tools, working software over comprehensive documentation, customer collaboration over contract negotiation, and responding to change over following a plan (n.d.). These values align well with NOVA Web Development's stated value, "We want to help build the world we want to live in." (NOVA Web Development, n.d.), but values alone will not get us to responsive development of clean, maintainable, and changeable code.

With a new team of developers working with the cooperative, it is time for another focused study of clean code.

References

Babych, M. (2021, December 8). *A review of the minimum viable product approach*. Forbes.

https://www.forbes.com/sites/theyec/2021/12/08/a-review-of-the-minimum-viable-product-approach/

Chen, V. (2020, August 20). *What is dogfooding? And why it's still important for tech*. Bubble.

https://bubble.io/blog/dogfooding-startup-tech/

NOVA Web Development. (n.d.). Novawebdevelopment.org. Retrieved March 14, 2023, from

https://novawebdevelopment.org/

Jelkner's Weblog. (2022, July 6). Whither NOVA Web Development? Elkner.net. Retrieved

March 14, 2023, from https://weblog.elkner.net/blog/58/

Manifesto for Agile Software Development. (n.d.). Agilemanifesto.org. Retrieved March 14,

2023, from https://agilemanifesto.org/

Martin, R. C. (2011, February). *Clean code fundamentals: episode 1: Clean code* [Video].

https://cleancoders.com/episode/clean-code-episode-1

Martin, R. C. (2011, February). *Clean code fundamentals: episode 2: Names++* [Video].

https://cleancoders.com/episode/clean-code-episode-2

Martin, R. C. (2011, April). *Clean code fundamentals: episode 3: Functions* [Video].

https://cleancoders.com/episode/clean-code-episode-3

Martin, R. C. (2011, September). *Clean code fundamentals: episode 6 part 1: TDD* [Video].

https://cleancoders.com/episode/clean-code-episode-6-p1

Business Tracker. (2020). Businesstracker.org. https://businesstracker.org/

LibreOrganize. (2020). Libreorganize.org. https://libreorganize.org/