Noel Miranda

April 14, 2025

DevOps

Module 3.2 Assignment

Case Study: Strangler Pattern at Blackboard Learn (2011)

Introduction

Before diving into the case study, I would like to briefly introduce what the strangler fig pattern is because initially I had no idea what it was and what it could be with such an interesting name. When it comes to software development, it is actually a pretty smart way to upgrade old systems. The idea comes from how the strangler fig tree grows around older trees and slowly takes over. In the same way, this pattern helps teams replace parts of a large, outdated system little by little instead of all at once. To be more specific, it allows developers to build new features in a modern way, while the old system still works in the background. Over time, more and more of the new system takes over, and the older parts are retired.

This approach can be helpful to the many companies that continue to rely on systems that are too old to keep up with new demands. Usually, these systems are part of an enormous, tightly coupled monolithic architecture, which makes updates tedious and error prone. To combat such issues, the strangler fig pattern offers a safer and more controlled approach to make changes without needing to rebuild everything at once. According to The DevOps Handbook (Kim et al., 2021), this method gives teams a chance to reduce risk and make progress in a steady and manageable way. On top of being flexible, it also helps teams avoid the pressure and uncertainty of completely replacing a system in one go.

Summary of the Blackboard Learn Case Study

According to The DevOps Handbook (Kim et al., 2021), in 2011, Blackboard Learn, which is a company known for its educational tools, found itself stuck with a system that was falling behind. The system was getting harder to manage, updates were slower, and users were not happy. At the same time, competitors were moving faster. Something had to change.

Under such conditions, the team at Blackboard decided to try the strangler fig pattern (Kim et al., 2021). They did not try to rebuild the entire platform from scratch because it would have been risky and expensive. Instead, they started pulling pieces out of the old system and replacing them with newer, better ones. For example, they focused on rebuilding the parts of the platform that users interacted with the most (Kim et al., 2021). This gave them a chance to improve the experience quickly while keeping the core system stable. I personally think that was a smart move because it demonstrated that they understood their users' needs and did not just think about the technology.

On top of that, they also changed the way their teams worked. They moved toward smaller, more independent teams that could work on new features without being slowed down by the older system. This was possible due to the construction of APIs for the system to communicate within its differing components. Over time, the monolithic system transformed into a microservices system. As a result, this allowed them to test changes more quickly and fix problems faster.

Lessons Learned

One major lesson from this case study is that big changes do not always need to happen all at once. Taking small steps can actually make more progress in the long run. According to The DevOps Handbook (Kim et al., 2021), the strangler fig pattern helped Blackboard reduce downtime, avoid major disruptions, and keep delivering value during the process, which are

crucial in keeping the company's user relationship in a healthy state. It also allowed teams to learn as they went, thus guaranteeing better work quality over time.

Another lesson I took from it is the importance of flexibility. The Blackboard team did not lock themselves into a strict plan. They adjusted along the way based on what they learned and what the users needed. That kind of thinking is valuable no matter what kind of project one is working on. In addition, this shows how important communication is between teams because everyone had to be on the same page for this to work.

Finally, the case study made it clear that DevOps is not just about tools. It is also about people and how they work together. By giving teams more ownership and freedom to deliver updates, the company was able to build a system that was faster, more reliable, and better for users.

Conclusion

To sum it up, the strangler fig pattern is a smart way to modernize systems without creating chaos. In view of the Blackboard Learn case study, it demonstrates that this approach can really work if done with care and good planning. On top of that, this architectural pattern allowed Blackboard to move away from a legacy system without shutting everything down or starting from zero. More than that, it helped them deliver better service to their users. Taking all into consideration, this case study proves that change does not need to be frightening if one has the right strategy. Slow and steady, in this case, really did win the race.

References

Kim, G., Humble, J., Debois, P., Willis, J., & Forsgren, N. (2021). *The DevOps Handbook, Second Edition*. IT Revolution.