Report On DevOps:

Integrating DevOps into FinQuest's Current Development Approach

Jin Cherng Chong

Murdoch University

September 2022

Grade

1. Content and Research (75%) - Does the introduction describe the purpose of the report? - Is there evidence of adequate research and reading? - Is DevOps introduced? - Have the main benefits of integrating user experience design into agile development been identified and explained? - Have the main potential problems associated with integrating user experience design into agile development been identified and explained? - Have explicit recommendations for FinQuest been made? - Are the recommendations logical and clear? - Is sufficient detail provided to understand the recommendations? **Maximum score** 75	60 / 75
2. Reports Structure and Presentation (25%) - Does the report have a title that reflects the contents of the report? Does it look like a company report? - Does the table of contents have page numbers that match the page numbers on the report? Has it been generated using the Microsoft Word table of Contents functionality? - Is the writing fluent (correct grammar and spelling)? - Is the language appropriate for an organisational report? - Is material well linked to the aim of the report? - Does the body of the report follow the required structure? - Is a bibliography provided with all sources correctly presented using APA referencing? **Maximum score** 25	Good use of literature but you don't always need to include the page numbers. Use page numbers only when you use direct quotes. 20 / 25

80.00 / 100.00

Graded on	Saturday, 8 October 2022, 6:32 PM
Graded by	Fatuma Namisango

TABLE OF CONTENTS

1.   INTRODUCTION

FinQuest is a medium-sized company that delivers quality financial services to clients. The company adopts an agile development approach for the system development of their financial services. The justification for utilising this approach was to enable quality systems to be developed rapidly. Agile development methodology remains one of the most widely used methodologies (Abrahamsson et al., 2009, p. 281). DevOps, an alternative system development methodology, has recently seen rapid growth in popularity. Alternative system development methodologies should always be examined to ensure that FinQuest's current approach remains the most optimal approach and FinQuest's justification for its current approach remains valid.

As commissioned by the Chief Information Officer of FinQuest, this report seeks to provide guidance in successfully integrating DevOps with FinQuest's current system development approach. This report is structured as follows: Section 2 provides an overview of DevOps. Section 3 discusses the benefits of integrating DevOps with FinQuest's current system development approach. Section 4 discusses problems associated with integrating DevOps and the agile approach. Section 5 provides guidance in integrating both development approaches.

2.   OVERVIEW OF DEVOPS

Modern system development often relies on fast and frequent releases to maintain client satisfaction and adjust to changing client demands. Before the introduction of DevOps, having a standalone operation team and development team in distinct silos was the standard modus operandi (Süß et al., 2022, pp. 1495-1510). Having distinct silos for each team encouraged a lack of collaboration and communication between them (Díaz et al., 2021, pp. 1-50). This often resulted in fewer and slower frequent releases of system and software updates.

The general responsibility of a development team is to continuously develop new features and push them out to production, while the general responsibility of an operation team is to manage and maintain the deployed systems to ensure they operate correctly in production (Leite et al., 2019, pp. 1-35). These two distinct responsibilities tend to conflict; the development team values fast releases of frequent updates to production, while the operation team values stable systems in production. As a result, the operation teams often impede the development team's frequent releases (Leite et al., 2019, pp. 1-35).

DevOps, introduced in 2009, seeks to resolve the conflict (Almeida et al., 2022, p. 63). As described by Leite et al. (2019, pp. 1-35), DevOps combines the IT operation team with the development team. It is considered a cultural philosophy, tools, and practises that promote closer collaboration between the two teams and aim to increase the frequency and quality of system releases pushed to production (Abrahamsson et al., 2009, p. 281).

3.   POTENTIAL BENEFITS

Firstly, integrating DevOps and agile can speed up the FinQuest project's time to market (Almeida et al., 2022, p. 63). Currently, FinQuest utilises an agile-only system development approach to rapidly deploy systems to production. Integrating DevOps can further improve the rate at which a system is deployed to production. This is expressed in Senapathi et al. (2020, pp. 57-67) study, in which DevOps is acknowledged as enabling new functionalities to be deployed more quickly to production for end users. DevOps offers both continuous integration and deployment (Trigo et al., 2022, pp. 1-31). Continuous integration automates the manual process of merging code changes from various developers (Trigo et al., 2022, pp. 1-31). Similarly, continuous deployment automates the manual process of frequently delivering software releases to production (Trigo et al., 2022, pp. 1-31). Both continuous integration and deployment help make the FinQuest system deployment process more efficient by automating the current manual deployment process. This allows FinQuest projects to have a faster time to market.

Secondly, integrating DevOps and agile can provide FinQuest with improved resource utilisation. Faster and more frequent releases of new system functionalities to production will result in more feedback for developers (Toh et al., 2019, pp. 173-177). The increased frequency of client feedback for developers is one advantage of using a DevOps approach (Toh et al., 2019, pp. 173-177). The resources will be spent only on what is necessary; the client can be more active in bringing attention to non-relevant features being developed. This encourages resources to be redirected to more important tasks. DevOps also promotes the automation of repetitive tasks, freeing up the resources normally allocated to the manual version of the task (Toh et al., 2019, pp. 173-177). The resources freed and made available may include-developers, testers and infrastructure.

4.   POTENTIAL PROBLEMS

First, cultural barriers can be a potential problem impeding the successful integration of DevOps and the agile development approach. Cultural barriers, as described by Rajendran et al. (2017, pp. 437-456), are the differences in culture between two parties that prevent effective communication between them. As evident in Smite et al. (2021, p. 106612) study, when the DevOps team members were geographically separated from each other, issues began to arise. The DevOps team had to adopt the agile development methodology. DevOps team members from India were reluctant to voice, to their Swedish counterparts, problems they faced with their work (Smite et al., 2004, p. 106612). The Indian DevOps team members explained the reluctance as being due to the Swedish

members often hijacking and doing their work whenever problems were raised (Smite et al., 2004, p. 106612). This was believed to impede the agile approach (Smite et al., 2004, p. 106612) since ineffective communication violates the agile way of working (Hummel et al., 2013, pp. 343-355).

Second, the suitability of developing projects using DevOps and the agile development approach might be a problem. Integrating DevOps with the current agile development methodology will force every future project to be developed using DevOps. This is expressed in Hemon-Hildgen et al. (2020, pp. 474-499) paper, which examined a firm that integrated both approaches. In this study, employees expressed that DevOps was required to be utilised without justification (Hemon-Hildgen et al., 2020, pp. 474-499). Certain projects are simply not suited to being developed using DevOps. For instance, when it comes to the US Navy's combat defence system, ensuring that the system remains secure is vital (Miller et al., 2022, pp. 22-48). The importance of making sure the combat system remains secure was one barrier stopping agile and DevOps adoption (Miller et al., 2022, pp. 22-48). An insecure system makes the system prone to being compromised and important information potentially being transferred to the wrong hands. By nature, security is a weak point with DevOps (Lampropoulos et al., 2019). DevOps promotes the sharing of data between the operation team and the development team, and a faster development process leads to greater system exposure to security risks (Miller et al., 2022, pp. 22-48). FinQuest, a company that delivers financial services, would also highly value security since financial companies would most likely be subjected to stringent regulations.

5.   RECOMMENDATIONS

One way to remove cultural barriers that may arise due to integrating DevOps and agile is to implement cultural training. Casey (2009, pp. 8-17) examined the impact of cultural differences on the development process. The case identified that cultural differences causing a misunderstanding of information resulted in employees leaving the organisation (Casey, 2009, pp. 8-17). Both management and staff realised that cultural training would have helped (Casey, 2009, pp. 8-17). Similarly, Smite et al. (2021, p. 106612) confirmed that cultural training is important in resolving cultural differences. For FinQuest, ensuring that the team members are located close to each other will help minimise or even eliminate major cultural differences. This could mean having the teams in the same country. However, implementing cultural training would be appropriate and recommended if that isn't possible. FinQuest's cultural training can easily be facilitated through a ninety-minute lecture-based approach (Pruegger & Rogers, 1994, pp. 369-387). As outlined in Pruegger and Roger's (1994, pp. 369-387) article, the lecture would explore the relevant cultures and they impact behaviour and communications. In addition, the lecture would include specific relevant examples. The lecture-based approach is advantageous because it is the most cost-effective approach to deliver and the easiest to create (Bhui et al., 2007, pp. 1-10).

Second, developing a criterion to assess whether a project is suitable for DevOps and Agile is recommended. There are various criteria and selection frameworks available. Bakhtouchi and Rahmouni (2018, pp. 211-216) offer a tree-decision approach for selecting the development methodology. Cockburn's (2000, pp. 64-71) article presents a framework selection methodology

grid. Churchill's (2017) paper outlines a framework model based on the analytic hierarchy process to calculate the most appropriate methodology for a project. All three approaches are based on underlying factors influencing a project's appropriateness for certain methodologies. The common underlying factors also apply to FinQuest. For instance, FinQuest criteria factors could include project size, project cost, project timeframe, project importance, project quality, and project security. Future FinQuest projects should be assessed against these factors. Simple FinQuest projects would no longer utilise the DevOps approach.

6.   REFERENCE LIST

Almeida, F., Simões, J., & Lopes, S. (2022). Exploring the benefits of combining DevOps and agile. *Future Internet, 14*(2), 63. https://doi.org/10.3390/fi14020063

Bakhtouchi, A., & Rahmouni, R. (2018, October). A Tree Decision Based Approach for Selecting Software Development Methodology. In *2018 International Conference on Smart Communications in Network Technologies (SaCoNeT)* (pp. 211-216). IEEE

Bhui, K., Warfa, N., Edonya, P., McKenzie, K., & Bhugra, D. (2007). Cultural competence in mental health care: A review of model evaluations. *BMC Health Services Research, 7*(1), 1-10. https://doi.org/10.1186/1472-6963-7-15

Casey, V. (2009, July). Leveraging or exploiting cultural difference?. In *2009 Fourth IEEE international conference on global software engineering* (pp. 8-17). IEEE.

Churchill, B. J. (2017). *Selecting a software development life cycle methodology*

Cockburn, A. (2000). Selecting a project's methodology. *IEEE Software, 17*(4), 64-71. https://doi.org/10.1109/52.854070

Díaz, J., López-Fernández, D., Pérez, J., & González-Prieto, Á. (2021). Why are many businesses instilling a DevOps culture into their organization?. *Empirical Software Engineering, 26*(2), 1-50. https://doi.org/10.1007/s10664-020-09919-3

Hemon-Hildgen, A., Rowe, F., & Monnier-Senicourt, L. (2020). Orchestrating automation and sharing in DevOps teams: A revelatory case of job satisfaction factors, risk and work conditions. *European Journal of Information Systems*, *29*(5), 474-499. https://doi.org/10.1080/0960085X.2020.1782276

Hummel, M., Rosenkranz, C., & Holten, R. (2013). The role of communication in agile systems development: An analysis of the state of the art. *Business & Information Systems Engineering*, *5*(5), 343-355. https://doi.org/10.1007/s12599-013-0282-4

Lampropoulos, G., Morcavallo, A., Salvi, L., Spiralska-Golak, I., & Siakas, K. DevOps: The New Frontier of Industrial Software Development

Leite, L., Rocha, C., Kon, F., Milojicic, D., & Meirelles, P. (2019). A survey of DevOps concepts and challenges. *ACM Computing Surveys (CSUR), 52*(6), 1-35. https://doi.org/10.1145/3359981

Miller, A. W., Giachetti, R. E., & Van Bossuyt, D. L. (2022). Challenges of adopting DevOps for combat systems development environment. *Defense AR Journal*, *29*(1), 22-48. https://doi.org/10.22594/dau.21-870.29.01

Rajendran, D., Farquharson, K., & Hewege, C. (2017). Workplace integration: The lived experiences of highly skilled migrants in australia. *Equality, Diversity and Inclusion*, *36*(5), 437-456. https://doi.org/10.1108/EDI-11-2016-0094

Pruegger, V. J., & Rogers, T. B. (1994). Cross-cultural sensitivity training: Methods and assessment. *International Journal of Intercultural Relations, 18*(3), 369-387. https://doi.org/10.1016/0147-1767(94)90038-8

Senapathi, M., Buchan, J., & Osman, H. (2018, June). DevOps capabilities, practices, and challenges: Insights from a case study. In *Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering* 2018 (pp. 57-67).

Smite, D., Moe, N. B., & Gonzalez-Huerta, J. (2021). Overcoming cultural barriers to being agile in distributed teams. *Information and Software Technology*, *138*, 106612. https://doi.org/10.1016/j.infsof.2021.106612

Süß, J. G., Swift, S., & Escott, E. (2022). Using DevOps toolchains in agile model-driven engineering. *Software and Systems Modeling, 21*(4), 1495-1510. https://doi.org/10.1007/s10270-022-01003-2

Toh, M. Z., Sahibuddin, S., & Mahrin, M. N. R. (2019, February). Adoption issues in DevOps from the perspective of continuous delivery pipeline. In *Proceedings of the 2019 8th International Conference on Software and Computer Applications* (pp. 173-177).

Trigo, A., Varajão, J., & Sousa, L. (2022). DevOps adoption: Insights from a large European Telco. *Cogent Engineering, 9*(1), 1-31. https://doi.org/10.1080/23311916.2022.2083474