DevOps for Product and Service Innovation (Research and Development) Teams: A Case Study

Dinuka Dayarathna
Faculty of Graduate Research and Studies
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
drdchathuranga@gmail.com

Lakmal Rupasinghe
Faculty of Graduate Research and Studies
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
lakmal.r@sliit.lk

***Abstract*—DevOps is a software development framework that extends the agile mindset to quickly produce software products and services as well as to improve operations performance and quality. DevOps is made up of the words ”development” and ”operations”. It was created to hasten the release of web-based solutions and swiftly provide users with additional value. Many web-based systems develop without a clear long-term objective in accordance with usage trends.Prior to the broad use of web services, the majority of software with a specific purpose was distributed as packages for end users to install on their own systems. The intervals between versions ranged from months to years, and new versions were released much less frequently. Large design, coding, and testing phases were separated into several development cycles, which culminated in the release of a new stable version. Even though the end goal is obvious within the context, the procedure to get there is unclear. In order to minimize risks for all stakeholders and to enable visibility of the current state of a project and steering it in the appropriate direction, working releases that capture the current state of the program must be provided often. This study investigates how DevOps principles can help with software development for innovation for product and services under research and development. We identify the parallels and divergences between research projects and the creation of other software products, and we talk about how the relevant approaches should be expanded. Here, specific problems are discussed together with created answers and unanswered issues. The research project best practices are recognized as lessons learnt. This research effort has taken to explore on introducing DevOps for research and development team in one of the largest technology companies is Sri Lanka. We are discussing the advantage we have taken from available literature and resources and the challenges and open areas for future efforts.**

***Index Terms*—DevOps, Product and Service Innovation, Agile Software Development**

I. Introduction

A. DevOps

DevOps is a software development approach which involves continuous development, continuous testing, continuous integration, continuous deployment and continuous monitoring throughout its development life cycle. In other words, it is a software development approach which involves continuous development, continuous testing, continuous integration, continuous deployment and continuous monitoring throughout its development life-cycle. DevOps is a culture, Is about implementing best practices with help of tools, Is about implementing agility in information, Is about automation. Automating infrastructure and workflows, continuously measuring application performance.

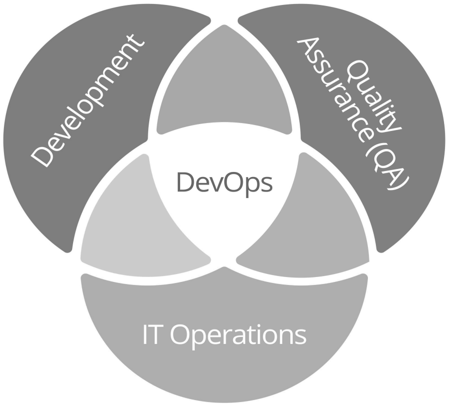

Fig. 1. DevOps Main Components

To improve the pace and caliber of software deployment, a DevOps team consists of developers and IT operations personnel that collaborate throughout the product life-cycle. Teams and the businesses they work for will be significantly impacted by this new method of working and cultural revolution. Tools are used by DevOps teams to automate and speed up activities, which increases reliability. Teams can manage crucial DevOps elements like continuous integration, continuous delivery, automation, and collaboration with the aid of a DevOps tool stack. Sometimes teams outside of development use DevOps values. Security becomes an active and integrated element of the development process when security teams use a DevOps methodology. DevSecOps is the term for this. We could depict the DevOps life cycle as below.

- Plan - Identify the objectives/ deliveries of the project and resources allocated for the project and then plan the tasks accordingly. (Trello, JIRA) [11]
- Code - Based on the requirement, it is required to select most suitable programming language to build the software. Probably it may be one or more programming languages for optimum solution. Tools required for source code management and version management. (VS Code,

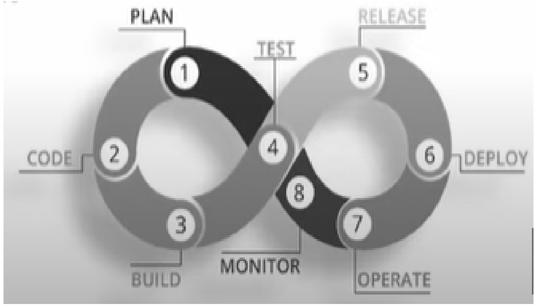

Fig. 2. DevOps Life-Cycle

Git, bitbucket) [12] enables distributed version control and facilitates data assurance through circulated non-linear workflows.

- Build - Stable and executable version of application need to build with support of coding and version management tools. Source code may get added or modified based issues on functional testing, integrations and packaging the source code for application functionality. (gradle, Mavan) [10]
- Test - Package released to test servers are then undergoes through quality assurance process through simulation to identify bugs and issues in functional and non functional aspects. This can be fully automated in DevOps to save time and effort based on the test scenarios and scope of work. (Selenium, JUnit, JMeter) [13]
- Release - Once the identified bugs fixed continuously, more stable and bug-free package will be released to Production environment. Releasing and configurations required can be fully automated to maintain high availability. This phase need to be always active and reliable once application is in live status. (Jenkins, GitLab, Bamboo) [14]
- Deploy - In configuration management process containerized applications are required for continuous deployment of release packages. (ANSIBLE, CHEF) [15]
- Operate - Least complex and shortest phase in DevOps lifecycle. Complete automation of release process and provide relevant updates and patches to achieve continuous operations and quick go-to-market timeline.
- Monitor - To maintain the high availability, reliability and productivity, continuously monitoring the performance of production application and environment by analyzing the usage data and be alert on threats or unusual behavior and automate proactive and preventive the actions required. (Nagios, Sensu) [16]

B. Product and Service Innovation Team

Product and Service Innovation unit has been established inside one of the largest tech companies in Sri Lanka. It is the innovation arm of the company to innovate the new products and services and also support and work with nationwide innovative community for the betterment of the Country. There are around 10-12 research and solution engineers works on innovations and productizing their innovations to external customers through solution engineers. All Research and Development (RD) engineers and solution engineers have a annual budget grant from the company and annual Key Performance Index (KPI) to measure their annual performance (How many new product or service concepts presented, how many Proof of Concept (POC) or prototypes build from them and how much Minimum Viable Product (MVP) or prototypes launch to the market.

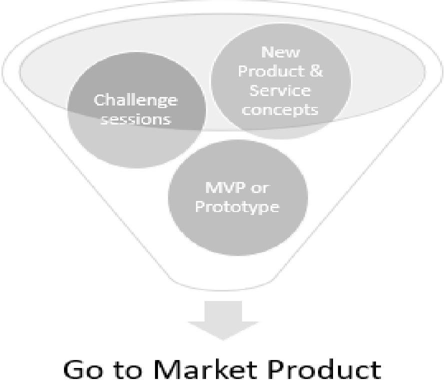

Fig. 3. New Product or Service Life Cycle

RD engineers and solution engineers are working on wide spectrum of technology stack based on their interest and organizational goals. They are working on disruptive technologies like Robotic Process Automation(RPA), Machine Learning (ML) and Artificial Intelligence(AI), Internet of Things (IoT), Metaverse based product and services targeting the current market needs. Based on the requirement end product or service will be connecting the customer through either web or mobile interfaces of combination of both. To making concepts to MVP or Prototype and them to a go to market product and then it's operation support has been supported by a team of software developers. Its kind of a internal development funnel. They were practicing some of the agile ceremonies like daily standups and planning sessions as required, but not fully practicing or adhering to any other framework. Based on the available literature, we see a gap of introducing DevOps concepts to RD area and various literature on introducing to the other software development teams and based on this nature of the team and their work culture and scope we tried out establishing some DevOPs based tools stack and life cycle to this team and evaluate the results.

5th 75% generation of high speed digital

II. Literature Survey

In the research done by Mirna and Mario [7] presents a solution to reinforce DevOps after understanding the practical implementation issues when implementing DevOps in organizations. They have represented results of a case study they have tried out in small scale company to prove their guide supported to implement DevOps which have clearly mentioned the gaps, better understand the concepts and how to facilitate DevOps.

Achieving a DevOps environment requires more than just installing automation tools; it also entails a cultural shift that indicates a method of team collaboration for project accomplishment. DevOps information is widely available, which makes it challenging to execute, with the major challenge being to locate and modify that information to produce a specified workflow. The advice provided in this research paper outlines two ways to begin implementing a DevOps approach within an organization: by defining a reinforced DevOps process, or by modifying the current process to obtain a reinforced DevOps process. Output of this case study as follows,

- The review is necessary to understand the organization's condition and to support them in need.
- The guidance framework makes it easy to integrate the reinforced DevOps. Risk-free implementation since it already beyond the needs of the company.
- After initial implementation, company improved some aspects after initial cultural change and it was easier to adopt from there.
- Training is required and it results some improvements in the company.

In another qualitative research conducted in 2017 by Erich, Amrit and Daneva [5] discuss about the systematic literature review and represented the results of exploratory research done across 6 companies of different sizes and fields. They have observed more positive outcome from results than negative experience regarding implementing DevOps. DevOps has been defined as the interplay of development and operations at the level of the individual, the team, and the department. Using this definition, organizations can clarify how they talk about and think about DevOps. The highest level codes are listed here which derived from research and interviews, which businesses may use as a tool to assess their own DevOps adoption.

Lusiano [3] in another research conducted in 2021, is making hypothesis on defining the job description for all the roles inside the DevOps framework may reduce the time to implement the processes and reduce the issues between team members. He has implemented the hypothesis based changes in practical and presented results. He has identified some parameters to measure for evaluating the performance of the DevOps teams as Average time for a component delivery, Release frequency, average recovery time from a failure point, KPIs defined by the organization are some of the key parameters he has used for his research.

A Case study approach done by Rajavi and Nisha [4] on establish security in DevOps as a best practice. They have highlighted that to implement security in all stages of DevOps, it is required to understand all the stage of DevOps and related practices and tools. This implementation is also interpret as DevSecOps which is a specified version of DevOps. Based on their literature review and proposed methodology they have identified three main attributes which can support effectively to implement security in DevOps. Those are training programs to provide right knowledge to relevant team members, integration of security policies or tools and preventive measures.Further as preventive measures following steps can be establish for further establishment of DevSecOps.

- Define, maintain and practice general compliance and protective measures.
- Security inbuilt to coding practices. Security by design.
- Continuous monitoring and feedback evaluation.

Another research effort has been taken in 2016 by Pulasthi, Madhushi and Indika [8] on evaluating the impact of DevOps practices in Software Development companies in Sri Lanka. Their main intention is to identify a relationship between business oriented delivery, Agility and increased quality factors and establish DevOps framework. Also another objective is to identify the issues they have faced and what are the optimum solutions to overcome those issues. Based on their evaluation they have confirmed a strong linear relationship between mentioned factors and implementing DevOps framework in Sri Lankan Software companies. Increasing quality of delivery has really supported the companies since most of the clients are resides in overseas and their satisfaction created more revenue in return. Also they are highlighting that the effort for identifying right tools to automate the functions to reduce error and improve quality and provide relevant training to operate with automated tools.

In 2019, Yarlagadda [2] has taken research effort to highlight the relationship between DevOps and the software development quality. CAMS [2] Framework (Community, Automation, Measurement and Sharing) has been named as a vital concept to adopt to DevOps to increase quality of the delivery. It focuses on people first rather than tools and automation when adopting DevOps which is a success factor to clear misunderstanding among team members and resistance to change. Additionally in the research emphasise the working relationship between development and operations which is another success factor of establishing DevOps in software companies to increase overall performance of the teams.

Hemon, Lyonnet, Rowe and Fitzgerald in their research on Agile to DevOps: Smart Skills and Collaboration [1] suggests three milestones between the journey from Agile to DevOps. Agile itself, Continuous integration and delivery. They also proving that team members are improving their soft skills over experience in working DevOps environment. They are more flexible, fast adopting and highly dynamic teams. Their article discussed the practical implications of several approaches, including whether teams using a waterfall plan-driven approach may begin at automation L3 right away or whether it is possible to skip from automation L1 to automation L3. The three outlined stages offer a fairly straightforward method to visualize a path leading to DevOps, but as we emphasized, they also imply that a direct transition to DevOps is feasible.

In another research done by Agarwal and Rawat [9] on DevOps as a new approach for testing and cloud development, their objective is to move DevOps environment setup to cloud and increase agility of software development and operation.

Also they are highlighting that DevOps leads to significant process change in software development. They are highlighting some key points to adopt for pilot DevOps projects to make them a success.

- Amplify innovation among operation and development teams to fast-track the delivery of components.
- All developers are on alert for any change in software components and they could act fast on resolutions.
- Continuous monitoring and feedback evaluation.
- Automated tools and standardized production leads to fast and quality output.

III. Methodology

A. Proposed Implementation Process

Based on the available literature we have identified that team collaboration and mindset and cultural change are key factors to start DevOps implementation. By keeping it in mind we have study the current setup including current processes, tools and practices and the scope of work related to the team. Then we have collaborated with teams on building a proposed DevOps infrastructure setup. Prior to implementing DevOps, we have briefed the senior management on the proposed architecture and the DevOps framework we are going to implement and requested their support to assess the teams, tools, and procedures it presently employs and, if necessary, change or eliminate them. It entails creating the appropriate infrastructure to provide teams the independence to develop, launch, and manage their products without overly depending on outside teams. In parallel, we have started training and mentoring sessions for teams based on the observations with help of other experienced teams, Agile coaches in the organization. Along with the training we have setup the finalized infrastructure setup for automation. Once the teams and infrastructure is ready we have started practicing and most importantly with implementation of agile ceremonies we could get continuous feedback from teams for continuous improvement.

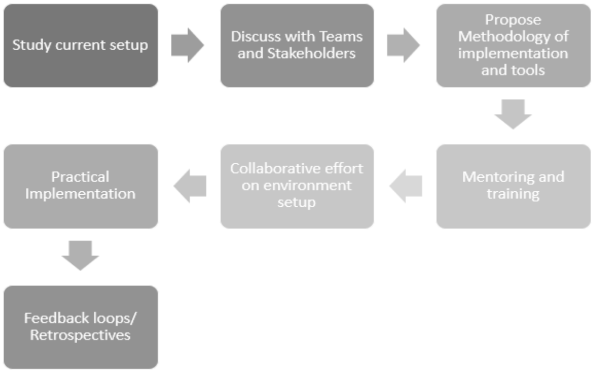
Study current setup
Discuss with Teams and Stakeholders
Propose Methodology of implementation and tools
Mentoring and training
Collaborative effort on environment setup
Practical Implementation
Feedback loops/ Retrospectives

Fig. 4. Proposed Implementation Process

B. Proposed DevOps Infrastructure Setup

DevOps has been the most discussed subject these days, especially in Continuous Integration/ Continuous Delivery (CI/CD). Therefore, CI/CD has been the main component of the software development cycle, with so many configurations and tools available. We will use Amazon Web Services (AWS) as the cloud platform, Github for the code repository, Jenkins for the Continuous Integration (CI), and AWS CodeDeploy service for the Continuous Delivery (CD). Since other software development teams in the organization is also using the similar setup, we could get support and training from them and avoid many infrastructure setup related issues.

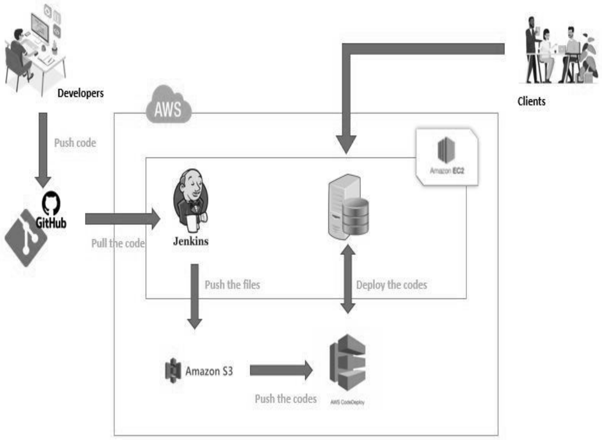

Fig. 5. Proposed Architecture

C. Proactive Measurements for Success

Even the DevOps framework have been applied for other software development and operation teams within the organization it was new to apply in highly dynamic area like research and innovation area. As a case study we have taken proactive measurements in the beginning of the research and below are some of the key points to note.

- Refer available literature and feedback from experts to identify the challenges and make proactive measures to overcome them to minimize the risk.
- Collaborative effort with team to finalize the infrastructure setup and implementation methodology.
- Organizational readiness/ Process level changes are managed by justifications in early stages.
- Minimize knowledge gaps for methodology and infrastructure setup by oriented training.
- Continuous feedback from teams and stakeholders to continuous improvement.

D. Results

Once the agile concepts were fully adhered to the team with DevOps, We have executed a week long Sprints with initial planning and other agile ceremonies like backlog refinement, Daily stand-up meetings and Sprint review and retrospectives. With them and the use right tools we have built the transparency of the work completed by the teams to the other stakeholders. It was highly appreciated by the management and the automated setup increased the quality of the work and team response for change management was remarkable. Below is a sample velocity chart exported from the past sprints. As you can observe we have some estimation issues in the beginning, but team has settled down over time to increase their productivity over time.

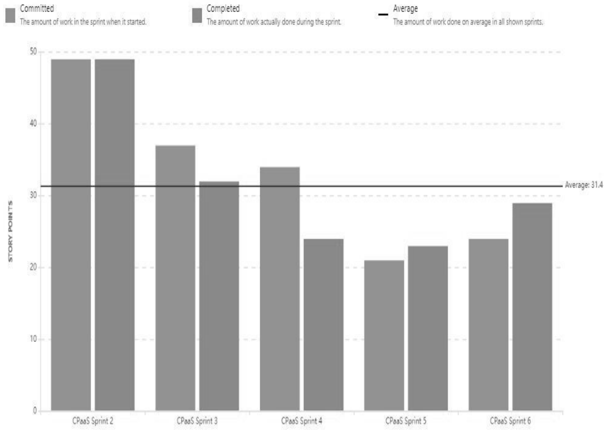

Fig. 6. Velocity Chart from past Sprints

E. Challenges and Future Work

As a research effort we have faced some challenges based on the domain specific facts which is significant to note in here for future work.

- Development teams are working on broad spectrum of technology stack which may not compatible to undergo same kind of automated testing, integration and deployment. We need to explore more on specific scientific areas for best DevOps approach.
- There are some scenarios, these RD teams are collaboratively work with 3rd party institutions like universities, school laboratories and other government research agencies. Since they cannot on boarded directly to this setup there were some issues in communications and delivery.
- Even we have received support from experts to setup infrastructure, there were some additional effort required scenarios in implementation.

CONCLUSION

We discussed our experiences creating novel MVPs and prototypes and go to market products and services utilizing cutting-edge methodologies like Agile, DevOps and cloud computing in this paper. We determined the features that agile web application development and delivery have in common. There were discussions on specific problems, the solutions we came up with, and the unanswered questions. For those who are not much familiar with the tool stack, software and hardware setup issues can be quite frustrating. There may be some issues due to the technology stack uses to develop products which may not generally support the overall infrastructure setup. Also there may be gaps in process changes between stakeholders and 3rd party on boarding issues. But as a research effort, proactive measures and results we could say DevOps implementation in RnD environment may leads to more successful direction.

REFERENCES

[1] Hemon, A., Lyonnet, B., Rowe, F. et al. From Agile to DevOps: Smart Skills and Collaborations. Inf Syst Front 22, 927–945 (2020). https://doi.org/10.1007/s10796-019-09905-1

[2] Yarlagadda, Ravi Teja, How DevOps Enhances the Software Development Quality (August 3, 2019). International Journal of Creative Research Thoughts (IJCRT), ISSN:2320-2882, Volume 7, Issue 3 August 2019, Available at :http://www.ijcrt.org/papers/IJCRT1133932.pdf, Available at SSRN: https://ssrn.com/abstract=3798270.

[3] L. de Aguiar Monteiro, "A Proposal to Systematize Introducing DevOps into the Software Development Process," 2021 IEEE/ACM 43rd International Conference on Software Engineering: Companion Proceedings (ICSE-Companion), 2021, pp. 269-271, doi: 10.1109/ICSE-Companion52605.2021.00124.

[4] Desai Rajavi, Nisha T. N.. "Best Practices for Ensuring Security in DevOps: A Case Study Approach." Journal of Physics: Conference Series (2021) doi: 10.1088/1742-6596/1964/4/042045.

[5] Erich, FMA, Amrit, C, Daneva, M. A qualitative study of DevOps usage in practice. J Softw Evol Proc. 2017; 29: e1885. https://doi.org/10.1002/smr.1885.

[6] Mireille Blay-Fornarino, Günther Jungbluth, Sébastien Mosser. Applying DevOps to Machine Learning: ROCKFlows, a Story from the Trenches. 2018. ffhal-01792775.

[7] Mirna Muñoz;Mario Negrete Rodríguez; (2021). A guidance to implement or reinforce a DevOps approach in organizations: A case study . Journal of Software: Evolution and Process, doi:10.1002/smr.2342

[8] P. Perera, M. Bandara and I. Perera, "Evaluating the impact of DevOps practice in Sri Lankan software development organizations," 2016 Sixteenth International Conference on Advances in ICT for Emerging Regions (ICTer), 2016, pp. 281-287, doi: 10.1109/ICTER.2016.7829932.

[9] P. Agrawal and N. Rawat, "Devops, A New Approach To Cloud Development Testing," 2019 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT), 2019, pp. 1-4, doi: 10.1109/ICICT46931.2019.8977662.

[10] Davis, A. L. (2019). Gradle. In Learning Groovy 3 (pp. 105-114). Apress, Berkeley, CA.

[11] Kaur, A. (2018). App Review: Trello. Journal of Hospital Librarianship, 18(1), 95-101.

[12] Fricke, C. (2021). Standalone Web Diagrams and Lightweight Plugins for Web-IDEs such as Visual Studio Code and Theia.

[13] Collins, E., Dias-Neto, A., de Lucena Jr, V. F. (2012, July). Strategies for agile software testing automation: An industrial experience. In 2012 IEEE 36th Annual Computer Software and Applications Conference Workshops (pp. 440-445). IEEE.

[14] Singh, C., Gaba, N. S., Kaur, M., Kaur, B. (2019, January). Comparison of different CI/CD tools integrated with cloud platform. In 2019 9th International Conference on Cloud Computing, Data Science Engineering (Confluence) (pp. 7-12). IEEE.

[15] Rahman, A., Rahman, M. R., Parnin, C., Williams, L. (2021). Security smells in ansible and chef scripts: A replication study. ACM Transactions on Software Engineering and Methodology (TOSEM), 30(1), 1-31.

[16] Bovina, S., Michelotto, D. (2017, October). The evolution of monitoring system: the INFN-CNAF case study. In Journal of Physics: Conference Series (Vol. 898, No. 9, p. 092029). IOP Publishing.