



The Challenges and Mitigation Strategies of using DevOps using Software Development

Nivedita A Gaonakar¹, Prof. P.V. Mitragotri²

Department of MCA, KLS Gogte Institute of Technology Belagavi-590008¹

Visveswaraya Technological University, Belagavi, Karnataka -590008, India²

Abstract: This report bolt on the challenges that DevOps brings to program improvement, and the strategies for relieving challenges. DevOps could be a set of organizational hones as well as a culture that tries to dispense with deterrents between the Devs and Ops groups, moving forward the interaction and communication among colleagues. Each and each industry, from fund to retail fabricating, depends on computer programs. Person Desktop programs, wide-ranging web applications, and portable apps are all illustrations of computer programs. In this think about, various challenges of embracing DevOps practices in-car computer program improvement ventures were recognized. A past inquiry about approximately both DevOps and car programs, existed at the time of composing, but ponders joining these two points were occasional. Distinctive organizations have embraced DevOps standards due to gigantic potentials, such as less than no time to generation, increased solidity, and steadiness. With today's headway within the field of innovation, the industry is drawing closer to the "DevOps" worldview of software development. Different dangers can be tended to and moderated, but without the correct structure, preparation, and capacities, numerous dangers go unevaluated. The dexterous computer program advancement technique proceeds to have eminent focal points over conventional computer program advancement and is continually evolving. One advancement is Engineer and Operations (DevOps) which centers on robotization, collaboration, and development. DevOps isn't an estimate shoe that fits all, and the appropriation of DevOps requires a sizable sum of methodology, exertion, and organization speculation. This thinks about implied to affirm and assess whether the issues and moderation measures of executing DevOps from precise writing inquire about are far-reaching in trade.

Keywords: DevOps, software development, culture, challenges, programs

I. INTRODUCTION

Conventional computer program companies have unmistakable divisions for computer program advancement, IT operations, and quality affirmation. Advancement and operations are regularly at chances when giving great program to clients routinely. Engineers are more curious about giving modern highlights or alterations to customers quickly, whereas operations need more prominent unwavering quality and security and recommend that they do not modify their items as routinely. It's troublesome for operations to endure the visit discharge of unused forms. These differences may block the headway of program. To bridge the hole between improvement and operations, DevOps combines the words "improvement" and "operations," which are utilized to change incentives and share strategies all through the complete improvement prepare. Improvement and operations must work together more effectively to handle significant issues all through the program improvement handle, and DevOps could be a collection of strategies for doing so Fear of alter and risky arrangement are key challenges. There's less of a partition between designers and operations work force and end-users as a result of DevOps. The computer program improvement prepare models utilized within the IT industry takes after a common system utilized to structure, plan and control the method of creating an data framework in an effective and beneficial way. There are number of program advancement life cycle models characterized and outlined to be practiced amid the program improvement prepare. Each prepare show takes after a arrangement of steps one of a kind to its sort, guaranteeing victory within the handle of computer program improvement. Conventional program improvement life cycle models are Waterfall Show, Iterative Show, Winding Show, V-Model, Big-Bang Demonstrate. In 2001 Dexterous was presented to the industry and companies utilized to hone dexterous concepts such as Scrum, Kanban. Few a long time prior industry presented DevOps which was an made strides adaptation of Spry centering on operational angles. DevOps may be a set of strategies in which designers and operations communicate and collaborate to provide program and administrations quickly, dependably and with higherquality. The word 'DevOps' has come up with Dev from Designers and Ops from Operation. DevOps is sharing of errands and obligations inside a group engaged with full responsibility of their benefit and its fundamental innovation stack; from improvement, to sending and back. DevOps is getting well known with each passing day due to the quicker application conveyance to the showcase with moo taken a toll and brief length.



The integration of advancement and operational groups guarantees fast sending and testing of program related with each construct cycle. Agreeing to Capgemini's yearly quality report 2016-17, 88% of the companies utilized or tried the DevOps standards. Yearly 'State of DevOps' reports uncovers that as DevOps hones spread, the number of DevOps groups has developed from 19% in 2015 to 22% in 2016 to 27% in 2017. Organizations have enormously grasped DevOps standards due to current customers' request for exceedingly accessible, continuous-release, and high-value applications that are useable anytime, anyplace at any stage. Driving wanders such as Google, Netflix, Amazon, LinkedIn, Spotify, Glint, and Etsy have embraced DevOps hones to discharge computer program with a better pace and way better quality. The most objective of the IT organization is continuously to bring unused and prevalent quality applications with more highlights to the client, be its an internal buyer of the organization or a customer within the advertise.

II. PROBLEM STATEMENT

The most issue that this paper will address is to investigate the challenges and moderation methodologies when embracing DevOps amid program advancement. It's not unprecedented for advancement and operations to clash when a company is giving valuable modern program to its clients. To resolve the pressure between the advancement and operations groups, the creating thought of DevOps has been displayed. DevOps is being utilized by an expanding number of businesses and organizations. The idea of DevOps is still moderately unused, so it's critical to know what issues it might offer assistance illuminate as well as how to moderate them.

III. LITERATURE REVIEW

A. *DevOps: A Brief History*

When Patrick Debois was working on a extend including improvement and operations groups, he realized that there had to be distant better; a much better; a higher; a stronger; an improved">a much better strategy to handle the debate that emerge between the two universes of Dev and Ops. This was in 2007. Dexterous Foundation was a well-known theme of talk amid the 2008 Spry Conference in Toronto, much obliged to Patrick Debois. A little gather of individuals met after the conference to investigate ways to bridge the improvement and operational holes. Amid the Speed conference 2009, John All spaw and Paul Hammond displayed their well-known address entitled 10 arrangements per day Dev & ops participation at Flickr on June 23rd, 2009. Patrick, John, and Paul met together after trading messages on Twitter to conversation almost DevOps. Patrick concluded that the occasion required a title that included both improvement and operations, subsequently DevOps days was born.

DevOps pulled in frameworks directors, designers, and directors from all around the globe. Indeed in spite of the fact that participants scattered to all regions of the world once the conference finished, the subject carried over to Twitter. Since Twitter as it were permits 140 characters per tweet, the DevOps hashtag is utilized rather than the DevOps day's hashtag. Inevitably, the conferences became a repetitive around the world arrangement delivered by the DevOps community that's a key driving constrain within the field. The #DevOps Twitter hashtag creates into a wealthy and critical data stream on the social media location. As DevOps develops, it enters the workplace, where well-known companies like Target, Nordstrom, and LEGO have grasped it. DevOps could be a blend of designs planning to move forward collaboration between advancement and operations. DevOps addresses shared objectives and motivating forces as well as shared forms and devices. Since of the characteristic clashes among diverse bunches, shared objectives and motivations may not continuously be achievable. Be that as it may, they ought to at slightest be adjusted with one another. Primary objective of the DevOps is to distinguish and kill the crevices between the Advancement group and Operations group. Computer program improvements experts regularly tend to come with an viewpoint where product change is what they are ordered to achieve. State of DevOps Report 2016 has recognized that DevOps has progressed organization's execution, incomes and productivity. DevOps groups have expanded from 16% in 2014 to 19% in 2015 at that point 22% in 2016. Tall performing companies have sent 200 times more regularly than moo entertainers at the side 2,555 times speedier lead times. Expansive companies such as Amazon, Netflix have sent changes thousands of times per day. These actualities suggest why companies have moved to hone DevOps.

B. *DevOps*

The concept of DevOps has gotten small consideration within the scholarly writing. Built on dexterous improvement standards, it permits for speedy improvement and sending cycles. DevOps has no all-around agreed-upon definition. According to researchers, DevOps may be a program advancement technique that integrates quality confirmation and operations into advancement strategies. DevOps may be a corporate methodology that will be utilized to characterize more successful participation including program improvement and foundation administration specialists. Concurring to IBM Cloud, receiving common DevOps innovations empowers participation between engineers, analyzers, and administrators and encourages the conveyance of program persistently by permitting collaborative testing and persistent



observing over the improvement, integration, and division situations. Utilizing the correct instruments may offer assistance with anything from form control to framework setup to coordinating to checking to containerization to mechanization. It's been since 2011 that the DevOps community has made open-source instruments like Manikin (for mechanizing the setup of virtual advancement situations) and Chef (for mechanizing the setup of physical improvement situations). DevOps selection isn't comparative for all organizations. When beginning DevOps hones, the company does not know different realities with respect to the venture, items and strategies are not clearly caught on. Numerous challenges are recognized when receiving the DevOps approach, engineers and the Operational group got to learn modern innovations, apparatuses, and strategies, and critical endeavors are required to begin the arrangement prepare consequently. The greatest drawback of DevOps development models is less documentation to relate DevOps strategy to moving forward innovation conveyance and way better trade results. DevOps and Dexterous are the same, but in a few vital facets, they are disparate. Spry reflects a alter in considering, whereas DevOps could be a conceptual system that forces changes in organizational culture. DevOps does not have strategies of spry techniques such as scrum and extraordinary programming; DevOps contains a wide and assorted extend of strategies and rules effectively implemented in a specific environment. The littler the number of individuals in a little organization, the less time it'll take to receive unused thoughts.

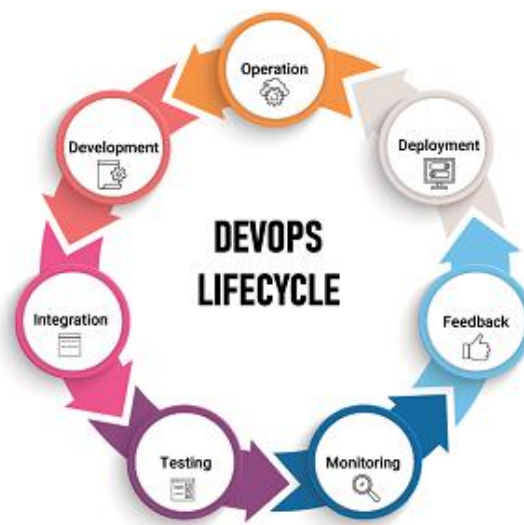


Fig.1 Six Phases of DevOps Model

C. *Benefits of DevOps*

As a entirety, the key objective of DevOps is to induce the leading conceivable return on venture whereas too guaranteeing highquality program and assembly client requests. To permit visit and fast computer program discharges, DevOps endeavors to construct a persistent pipeline that includes computerized testing cycles to empower ceaseless computer program conveyance. Utilizing DevOps, you'll too react rapidly to client needs that alter. With DevOps, designers and administrators may to Co-ordinated all organizational forms, streamline testing and quality affirmation, and ease the move between improvement and operations. DevOps eliminates organizational and social issues by joining improvement and operations and reduces the fetched of fault discovery within the early stages. Computer program abandons are right away settled early on within the advancement lifecycle within the DevOps environment due to the nonstop arrangement of program builds.

D. *Major DevOps Challenges*

The start of DevOps can be linked to the need to get rid of separate groups within an organization in order to work together and improve the products being produced. Progress are being made in various areas such as technology, education, and medicine. New innovations and discoveries are constantly being made, improving our lives and society. This continuous improvement benefits everyone and contributes to the advancement of our world. Operations are the two most important parts of the company area. DevOps is when software development and operations teams work together from the start of creating the software to when it is being used. This strategy aims to make deliveries faster and for the person delivering to take more responsibility for the final product, resulting in better quality. With DevOps, companies can serve their customers better by constantly delivering software that is of better quality. Although there are many benefits to using DevOps, there can also be many challenges when implementing it. Ensure smooth collaboration and communication between development and operations teams. It also requires businesses to adapt to cultural changes and implement new technologies and tools effectively. However, DevOps can bring numerous benefits



to businesses, including improved efficiency, faster time to market for products and services, and better quality and stability of software. Encourage working together in different areas or updating old ways of doing things that are more than ten years old. Before you start using DevOps, it's important to understand the problems you might encounter and find ways to solve them.

1. *Obstacles and Resistance to Change*

One problem with adopting DevOps is that there aren't enough good plans, so important people don't want to change. Some people who are relevant to DevOps may not want to use it because they are afraid that they will lose their jobs or because they don't want to work together with others. Different teams may have different abilities to understand and approach tasks, which can make it difficult for them to work together. Development and operations teams have different goals that contradict each other. Developers usually want new features and bug fixes to be released quickly. At the same time, the operations team wants to limit the number of releases to make sure that the system runs smoothly and stays stable and reliable. From a different point of view, the higher-ups in the company are not accepting the implementation of DevOps. The literature talks about the issues that can happen because of senior management. For instance, if the advantages of using DevOps are not clear, higher-ups in the company may hesitate to adopt DevOps by questioning its effectiveness and usefulness. DevOps automation tools can be costly and require careful handling, which might discourage managers from fully grasping the benefits of DevOps. Sometimes, the process of adopting DevOps can be influenced by the way things are organized within a company and how bosses manage their teams. The researchers talked about the low amount of work being done at the start of the adoption process. This might make senior managers think there is a problem. However, top-level managers may hesitate to accept DevOps because it might bring about problems with productivity.

2. *Lack of Standard Guidelines and Knowledge Skills*

The information suggests that DevOps doesn't have a clear and specific way or set of rules for how to put it into practice. The findings indicate that companies had trouble adopting DevOps because their employees lacked proper education, which made it challenging. But many articles about DevOps challenges talk about the problem of not having clear rules, abilities, and knowledge related to DevOps. These articles show that there is a problem with people not having enough skills, knowledge, education, or clear instructions on how to use DevOps. In one situation, a company that adopted DevOps had problems with the technologies and platforms they used in their work. Even in other companies where employees were highly skilled in technology and knowledge, the DevOps practitioners had difficulties with the DevOps approach. In a different situation, a study found that it is extremely important to hire the right people with the right skills. But, not knowing enough and not having the necessary skills caused problems when trying to use and do DevOps. People who are not qualified and don't have the right skills can make it harder to use DevOps because they don't know what to do when there are problems. Four main studies found that one of the biggest challenges with DevOps is that there aren't clear rules, understanding, and expertise. The researchers found that there aren't enough guidelines for DevOps adoption, and this can cause delays in releasing software (Figure 2).

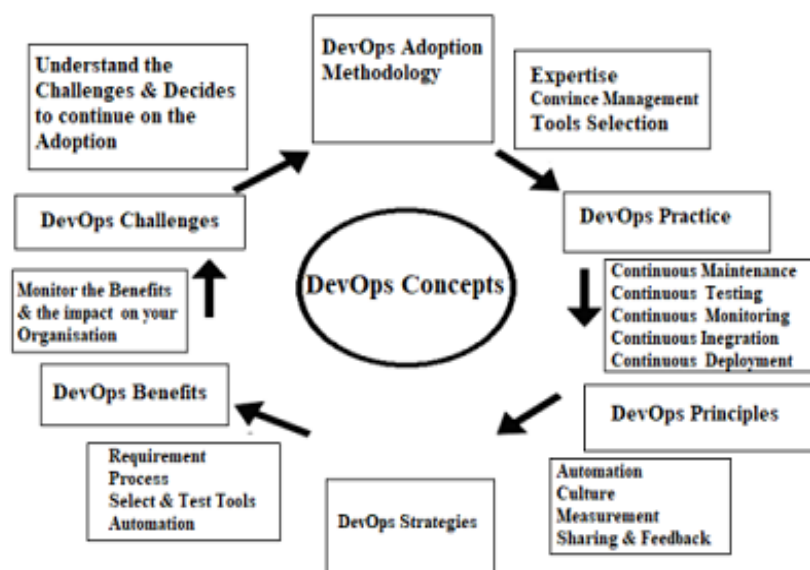


Fig. 2 DevOps Loop-Flow Framework Model



3. *Avoiding Risks Associated with Containers and Other Tools*

When you work in a DevOps environment, you often use cloud infrastructure and deployments. If you don't put in place the right security measures, these can make your application more likely to have security problems. In the DevOps world, many new and creative technologies are used. Small errors or settings mistakes in the quick-moving process of DevOps can cause big problems, like when companies accidentally expose sensitive information or when an entire company fails. There are many different tools that a DevOps team uses, such as Ansible, Salt, Chef, Puppet, and others. Containers are commonly used tools/technologies by DevOps teams and are widely used everywhere. Containment systems are lightweight and portable platforms used to package software programs so that they can be easily installed and used. It may be difficult for security teams to accurately assess the safety of these containers. We are thinking about using libraries that are known to be safe, and also creating services that have good protection. Are secrets being kept safe. If we don't address these concerns or give satisfactory answers, using containers could be dangerous for a company. Although containers pose a big challenge, it is essential to deal with and safeguard all the tools used for deployment. These tools are crucial for creating and managing the application and environment once it is deployed. Often, the keys to access important things are connected to a specific software. Before using this software, it is important to thoroughly check if it is safe and secure.

4. *Poor access controls and secret management procedures*

One of the main security problems in DevOps is having weak controls for accessing information and not having well-managed procedures for keeping secrets safe. These problems can cause different security issues, such as information being stolen, people getting into systems without permission, and losing money. It's very important to keep things a secret and have strict control over who can access them when using highly automated development and deployment processes. API tokens, SSH keys, privileged account credentials, and other similar pieces of information are considered secrets. Containers are things that can be used by many different types of people and organizations, like businesses or individuals, to help them with their work. There are also services, which are things that provide a specific type of help or assistance. People who work in these containers or services are called personnel. There are also many other things that can make use of these containers and services. These important passwords and keys are often not handled properly, which makes them an easy target for hackers on the internet. Furthermore, in order to keep works running smoothly and effectively, DevOps teams often allow unrestricted access to important accounts like admin and root. It is more likely that people will misuse these extra rights when many users share passwords for secret accounts and when processes have a lot of access.

E. *DevOps Mitigation Strategies in Software Development*

DevOps can bring problems with security and compatibility among different teams in software development, but there are ways to solve these issues. Think about using these strategies in your business to make DevOps more secure while keeping a good balance between different teams and the need to be quick and flexible. DevOps mitigation strategies are plans or methods used to lower the chances of risks occurring and minimize their negative effects in software development. These ideas can be used in many different areas. By using helpful DevOps strategies, organizations can decrease the chances and effects of risks in software development. This can make the software better, faster, and more secure, and lower the chance of expensive security issues.

1. *Enforce Security-Oriented Policies*

When it comes to creating safe and secure places, it's important to have rules and good communication. You need to create a set of simple and clear processes and rules to protect your cybersecurity. These should cover things like restrictions on accessing information, checking code, using firewalls, and managing settings. to ensure that all security measures are followed.

Work together in a positive way to build a safe application. Moreover, the concept of "infrastructure as code (IaC)" is a fundamental aspect of the DevOps approach and implementation. Setting up and arranging virtual machines, configuring networks, distributing workloads, and establishing connections are all described using code that is tracked and updated similarly to how DevOps teams handle their application code. IaC means Infrastructure as Code evaluated for security vulnerabilities. This can help identify any weaknesses and make improvements, ultimately making everything more secure. Make it easier to check if the environment is set up correctly for the situation. When we use an IaC model, it means that if we have the same code, we will get the same results. It helps us recreate the same environment that we had before. The use of IaC solves the problem of differences between environments in the delivery process by using distributed computing. If there was no Infrastructure as Code (IaC), teams would need to control the settings of each deployment environment by hand. Problems can happen when there are differences in different places. IaC makes it easier for DevOps teams to handle the security of their apps and environments.



In order to keep everything running smoothly and get things done efficiently, DevOps teams that use IaC work together and follow the same security rules and tools to work together.

2. *Identify and Manage Vulnerabilities Successfully*

Joining security into the program improvement lifecycle (SDLC) from the beginning makes it less demanding to spot and address vulnerabilities early on. To bargain with the identified vulnerabilities, you'll require a successful powerlessness administration framework that can screen and prioritize the numerous approaches that ought to be taken to settle each issue (remediation, acknowledgment, exchange, etc). In a powerlessness administration program, there are four major steps to consider:

- Decide the criticality of a resource, the proprietors of the resource, the recurrence of filtering, and the plan for repair that can be accomplished inside a sensible time period.
- Find and stock resources on the arrange as they are found.
- Recognize any vulnerability within the resources that have been identified.
- Recognize and address any vulnerability that has been found.

Once you first begin working with a powerlessness administration application, you will realize that you just have a or maybe tall powerlessness score, which comes about in time-consuming repair cycles. The imperative thing is to illustrate improvement from quarter to quarter and year to year. With expanded recognition and instruction around the defenselessness administration program, groups ought to be able to diminish the sum of time went through remediating vulnerabilities and bringing down their helplessness scores. These programs persistently grasp and comply with the organization's most current chance decrease targets.

3. *Adopt a DevSecOps model of operation*

To achieve good DevOps security, it is important to encourage collaboration between different teams involved in the entire DevOps process, from building to running software. When trying to make things safer, teams that focus on both development and operations should work together and be involved at every stage of the process. Security shouldn't just be the job of one team. Instead, it should be something everyone in the organization cares about and is involved in. The term "DevSecOps" means that security is built into the way an organization operates. It is a practice in businesses where everyone takes responsibility for following security rules and procedures. DevSecOps is a set of cyber security actions that aim to reduce the chances of security breaches due to weak account restrictions and other vulnerabilities. This goes far beyond just using advanced tools and programs to make sure that security is thought of as an important idea for an organization. DevSecOps promotes the understanding of basic security principles among teams of any size. It is suggested that everyone on a team should receive basic security training. In addition to learning in school, developers should also learn how to use special tools and software to quickly check the security of their code. Security experts should know how to write code and work with APIs. This will help them create scripts and automate security checks, which can be very helpful in environments that use Infrastructure as Code (IaC). Participation of individual security teams can create secure versions of infrastructure for the development team to use. They can also make sure the settings are followed by automatically checking the code that controls the infrastructure, which can be done using computer programs.

4. *Implement an effective Privileged Access Management system*

You can greatly reduce the chances by limiting who has special access to the account. This change means that regular computer users will not be able to use accounts with special powers or high levels of access. You should carefully monitor all sessions using special accounts to make sure they are legitimate and follow your company's rules. Additionally, the access for development teams to specific development, production, and management areas is limited. In simpler words, "enforcing a privileged model" means implementing a system where certain individuals have more advantages or benefits. However, they should still have the ability to create pictures and machines from approved models, put them into action, make alterations to them, and resolve any weaknesses in the system with the required access and permissions. Think about using a high-tech system like Open IAM to manage special access to things. It can do things automatically, like controlling, watching, and recording access, from the start to the end of a project. We need to keep track of important credentials and secrets from the moment they are created until they are no longer needed. This means that both people who work inside a company and people who attack a company from the outside can cause problems with cyber security. People who work inside the company might abuse their power to do bad things, and people who attack from the outside might trick or steal information from people inside the company. In both cases, people are the ones who make mistakes and cause problems with cyber security. Privileged access management helps organizations ensure that individuals only have the required access levels to perform their tasks. Privileged Access Management helps security teams find and stop bad actions connected to abusing special access and then quickly fix the problem to reduce risk.



It's important to watch out for and find unusual things happening in a place. However, if we don't pay attention to the biggest risks, like when important access isn't managed, watched, or protected, the business will still be unsafe. By using PAM as part of a thorough security and risk management plan, organizations can keep track of all actions related to important IT systems and sensitive data. This helps them make audits and meeting compliance rules easier.

Companies that focus on Privileged Access Management programs as part of their overall cyber security plan can gain many benefits. These benefits include lowering the risk of security breaches, decreasing the potential for cyber-attacks, saving money and reducing complications, improving the company's understanding of what is happening and being more prepared for different situations and meeting government regulations more effectively.

IV. CONCLUSIONS

This study looked at past writings and the business world to find out about the issues and ways to solve them when using DevOps in software development. This text is about development. The main aim of this paper is to provide a detailed analysis and explanation of the issues that will occur when using DevOps. In this research, we use a review of the literature to identify and discuss the issues faced when trying to implement DevOps in the software development process. We also provide solutions and strategies to overcome these challenges. The main goal of this paper is to study and determine if software quality gets better when using DevOps methods. We collected information by asking questions online and talking to experts in software development known as DevOps. Right now, more and more software development is using DevOps worldwide. In short, we have studied how DevOps affects the quality of software. In this process, we looked at different books and articles about the topic using different websites like Google scholar and IEEE Xplore. The review discovered the advantages, difficulties, and ways to resolve problems in DevOps. The main research used in this review indicates that the use of DevOps is currently being implemented in software development all around the world. We looked at the good things, difficulties, and rules for using DevOps. During this process, we looked at different articles about the advantages, difficulties, and ways to reduce problems in DevOps. We used various databases like IEEE, Google Scholar, and Science Direct. One of the biggest issues with using the DevOps approach is that many companies struggle to include security measures. However, we need to think about safety at every step. Adding security measures to the development and operations process as soon as possible can help to quickly find and fix weaknesses and mistakes before they cause issues. Including security at the beginning of the DevOps process helps ensure that it continues to be present and working well throughout the entire life of the product. With this security measure, we will keep the code safe from hackers and prevent unauthorized access to our data.

REFERENCES

- [1] M. Ammar, "Application of Artificial Intelligence and Computer Vision Techniques to Signatory Recognition", *Information Technology Journal*, vol. 2, no. 1, pp. 44-51, 2002.
- [2] V. Gupta, P. Kapur and D. Kumar, "Modeling and measuring attributes influencing DevOps implementation in an enterprise using structural equation modeling", *Information and Software Technology*, vol. 92, pp. 75-91, 2017.
- [3] J. Humble, D. Farley, *Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation* (Adobe Reader). Pearson Education; 2010 Jul 27.
- [4] H. Papadopoulos, A. Andreou and M. Bramer, *Artificial Intelligence Applications and Innovations*. Berlin, Heidelberg: IFIP International Federation for Information Processing, 2010.
- [5] L. Bass, I. Weber and L. Zhu, *DevOps: A Software Architect's Perspective*. Pearson Education, Inc., 2015.
- [6] L. Iliadis, I. Maglogiannis and H. Papadopoulos, *Artificial intelligence applications and innovations*. Berlin: Springer, 2012.
- [7] E. Dornenburg, "The Path to DevOps," *IEEE Software*, vol. 35, no. 5, pp. 71-75, 2018, DOI: 10.1109/MS.2018.290110337.
- [8] R. Jabbari, N. bin Ali, K. Petersen, and B. Tanveer, "What is DevOps? A systematic mapping study on definitions and practices," *ACM International Conference Proceeding Series*, vol. 24-May-201, May 2016, DOI: 10.1145/2962695.2962707.
- [10] R. Jabbari, N. bin Ali, K. Petersen, and B. Tanveer, "What is DevOps? A systematic mapping study on definitions and practices," in *ACM International Conference Proceeding Series*, 2016, vol. 24-May-201, pp. 1-11. DOI: 10.1145/2962695.2962707.
- [11] M. Daneva and R. Bolscher, "What we know about software architecture styles in continuous delivery and devops?," in *Communications in Computer and Information Science*, vol. 1250 CCIS, 2020, pp. 26-39. DOI: 10.1007/978-3-030-52991-8_2.
- [12] R. Rapaport, *A short history of DevOps*, 2014, (Accessed 28 December 2017).



- [13] T. Laukkarinen, K. Kuusinen, and T. Mikkonen, "Regulated software meets DevOps," *Information and Software Technology*, vol. 97, pp. 176–178, May 2018, DOI: 10.1016/j.infsof.2018.01.011.
- [14] S. Mandepudi, "Communication Challenges in DevOps & Mitigation Strategies," 2019, [Online]. Available: <http://www.diva-portal.org/smash/record.jsf?pid=diva2:1368400>
- [15] N.A.P.O. Perera, *An Analysis on Implementing DevOps in Software Companies in Sri Lanka* (Doctoral dissertation), 2016, Available at <http://dl.lib.mrt.ac.lk/handle/123/12372>.
- [16] V. Lalsing, S. Kishnah, and S. Pudaruth, "People factors in agile software development and project management." *International Journal of Software Engineering & Applications*, 3(1), 2012, p.117.
- [17] A. Benoist, 2013 "Influence of release frequency in software development.", [available at] <https://hal.archives-ouvertes.fr/hal-00832011v2> [17] Iso.org, "ISO/IEC 9126-1:2001 - Software engineering -- Product quality -- Part 1:Quality model", 2016. [Online]. Available:http://www.iso.org/iso/catalogue_detail.htm?csnumber=22749.
- [18] V. Sugumaran, *Distributed artificial intelligence, agent technology and collaborative applications*. Hershey, PA: Information Science Reference, 2009.
- [19] Y. Jiang, "Analysis on the Application of Artificial Intelligence Technology in Modern Physical Education", *Information Technology Journal*, vol. 13, no. 3, pp. 477-484, 2014. [11] Y. Nakajima, M. Ptaszynski, H. Honma and F. Masui, "Automatic extraction of future references from news using morphosemantic patterns with application to future trend prediction", *AI Matters*, vol. 2, no. 4, pp. 13-15, 2016.
- [20] T. Bradley and T. Bradley, "Why DevOps means the end of the world as we know it", *TechSpective*, 2016. [Online]. Available: <https://techspective.net/2015/08/16/why-devopsmeans-the-end-of-the-world-as-we-know-it/>.
- [21] K. Hirasawa, "Trend on application of AI technologies to industry. From the latest international workshop on AI applications.", *IEEJ Transactions on Industry Applications*, vol. 108, no. 10, pp. 868-871, 1988.
- [22] H. Izadkhah, "Transforming Source Code to Mathematical Relations for Performance Evaluation", *Annales Universitatis Mariae Curie-Sklodowska, sectio AI – Informatica*, vol. 15, no. 2, p. 7, 2015.
- [22] L. Lopes, N. Lau, P. Mariano and L. Rocha, *Progress in Artificial Intelligence*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009.
- [23] D. Linthicum, "What is DevOps? DevOps Explained [Microsoft Azure", *Azure.microsoft.com*, 2016. [Online]. Available: <https://azure.microsoft.com/en-us/overview/whatis-devops/>.
- [24] L. Rendell, "A new basis for state-space learning systems and a successful implementation", *Artificial Intelligence*, vol. 20, no. 4, pp. 369-392, 1983.
- [25] G. Simov, "Artificial intelligence and intelligent systems: the implications", *Information and Software Technology*, vol. 32, no. 3, p. 229, 1990.
- [26] H. Salzman, "Engineering perspectives and technology design in the United States", *AI & Society*, vol. 5, no. 4, pp. 339-356, 1991.
- [27] L. Bass, I. Weber and L. Zhu, *DevOps: A Software Architect's Perspective*. Pearson Education, Inc., 2015.
- [28] M. Huttermann, "Beginning DevOps for Developers", *DevOps for Developers*, vol., 2012, pp. 3-13.
- [30] R. Conejo, M. Urretavizcaya and J. Prez-de-la-Cruz, *Current topics in artificial intelligence*. Berlin: Springer, 2004.
- [31] Mazeiar Salehie, Liliana Pasquale, Inah Omoronyia, Raian Ali, and Bashar Nuseibeh. 2012. Requirements-driven adaptive security: Protecting variable assets at runtime. In *Requirements Engineering Conference (RE), 2012 20th IEEE International*. IEEE, 111–120.
- [32] Bradley Schmerl, Javier Cámara, Jeffrey Gennari, David Garlan, Paulo Casanova, Gabriel A. Moreno, Thomas J. Glazierr, and Jeffrey M. Barnes. 2014. ArchitectureBased Self-Protection: Composing and Reasoning about Denial-of-Service Mitigations. In *Proceedings of Symposium and Bootcamp on the Science of Security*. 2:1–2:12.
- [33] Martina Maggio, Henry Hoffmann, Alessandro V Papadopoulos, Jacopo Panerati, Marco D Santambrogio, Anant Agarwal, and Alberto Leva. 2012. Comparison of decision-making strategies for self-optimization in autonomic computing systems. *ACM Transactions on Autonomous and Adaptive Systems (TAAS)* 7, 4 (2012), 36. [34] Yar Rouf, Mark Shtern, Marios Fokaefs, and Marin Litoiu. 2017. A Hierarchical Architecture for Distributed Security Control of Large Scale Systems. In *Proceedings of the 39th International Conference on Software Engineering Companion (ICSE-C '17)*. IEEE Press, Piscataway, NJ, USA, 118–120. <https://doi.org/10.1109/ICSE-C.2017.64>.
- [35] François Gauthier and Ettore Merlo. 2012. Fast Detection of Access Control Vulnerabilities in PHP Applications. In *19th Working Conference on Reverse Engineering, WCRE 2012, Kingston, ON, Canada, October 15-18, 2012*. 247–256.
- [36] Díaz, J., Perez, J. E., Yague, A., Villegas, A., & de Antona, A. (2019). DevOps in Practice A Preliminary Analysis of Two Multinational Companies. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 11915 LNCS, 323 330. https://doi.org/10.1007/978-3-030-35333-9_23.



- [37] Knauss, E., Pelliccione, P., Heldal, R., undefinedgren, M., Hellman, S., & Maniette, D. (2016). Continuous Integration Beyond the Team: A Tooling Perspective on Challenges in the Automotive Industry. In Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement. New York, NY, USA: Association for Computing Machinery. <https://doi.org/10.1145/2961111.2962639>.
- [38] R. K. Gupta, M. Venkatachalapathy, and F. K. Jeberla, "Challenges in Adopting Continuous Delivery and DevOps in a Globally Distributed Product Team: A Case Study of a Healthcare Organization," in 2019 ACM/IEEE 14th International Conference on Global Software Engineering (ICGSE), Montreal, QC, Canada, May 2019, pp. 30–34. doi: 10.1109/ICGSE.2019.00020.
- [39] Elshamy, A., & Elssamadisy, A. (2007). Applying agile to large projects: new agile software development practices for large projects. In Agile Processes in Software Engineering and Extreme Programming: 8th International Conference, XP 2007, Como, Italy, June 18-22, 2007. Proceedings 8 (pp. 46-53). Springer Berlin Heidelberg.
- [40] M. Hüttermann, Beginning devops for developers, in DevOps for Developers. New York, NY, USA: Springer, 2012, pp. 3–13.