

Assignment Guidance and Front Sheet

This front sheet for assignments is designed to contain the brief, the submission instructions, and the actual student submission for any WMG assignment. As a result the sheet is completed by several people over time, and is therefore split up into sections explaining who completes what information and when. Yellow highlighted text indicates examples or further explanation of what is requested, and the highlight and instructions should be removed as you populate 'your' section.

This sheet is only to be used for components of assessment worth more than 3 CATS (e.g. for a 15 credit module, weighted more than 20%; or for a 10 credit module, weighted more than 30%).

To be completed by the student(s) prior to final submission:

Your actual submission should be written at the end of this cover sheet file, or attached with the cover sheet at the front if drafted in a separate file, program or application.

Student ID or IDs for group work	2286790
---	----------------

To be completed (highlighted parts only) by the programme administration after approval and prior to issuing of the assessment; to be consulted by the student(s) so that you know how and when to submit:

Date set	03/02/2023
Submission date (excluding extensions)	06/03/2023 by 12:00pm (UK time)
Submission guidance	To be submitted electronically via Tabula
Late submission policy	<p>If work is submitted late, penalties will be applied at the rate of 5 marks per University working day after the due date, up to a maximum of 10 working days late. After this period the mark for the work will be reduced to 0 (which is the maximum penalty). "Late" means after the submission deadline time as well as the date – work submitted after the given time even on the same day is counted as 1 day late.</p> <p>For Postgraduate students only, who started their current course before 1 August 2019, the daily penalty is 3 marks rather than 5.</p>
Resubmission policy	<p>If you fail this assignment or module, please be aware that the University allows students to remedy such failure (within certain limits). Decisions to authorise such resubmissions are made by Exam Boards. Normally these will be issued at specific times of the year, depending on your programme of study. More information can be found from your programme office if you are concerned.</p>

To be completed by the module owner/tutor prior to approval and issuing of the assessment; to be consulted by the student(s) so that you understand the assignment brief, its context within the module, and any specific criteria and advice from the tutor:

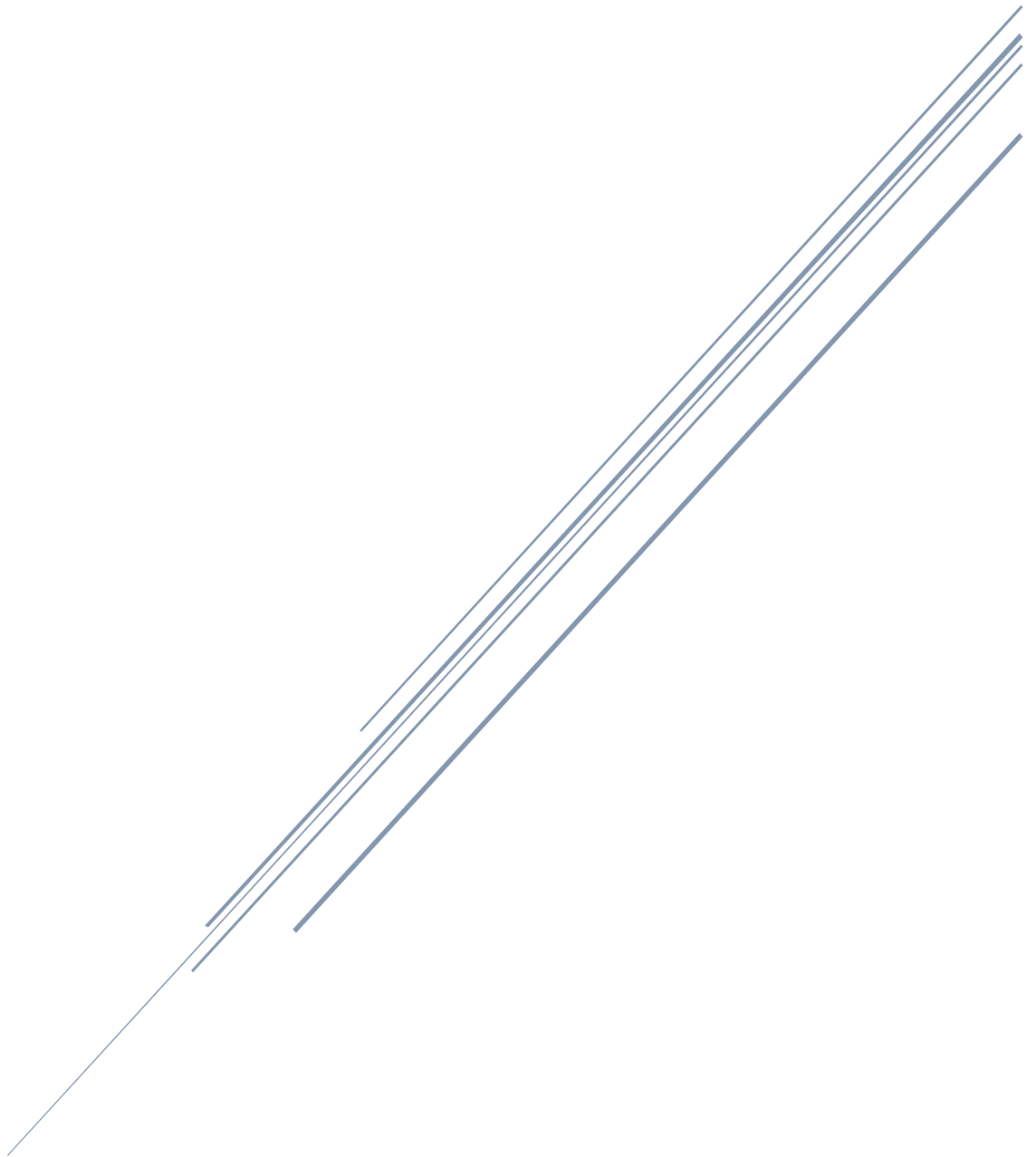
2286790

Business Report for PharmaC

Module title & code	WM9B6-15 Cloud Native Computing
Module owner	Michael Mortenson
Module tutor	Michael Mortenson
Assessment type	Essay
Weighting of mark	80%

ADOPTING CLOUD-NATIVE

PharmaC Inc.



2286790

Contents

Executive Summary	4
Methodology	5
Scope	5
Objectives.....	5
Assumptions.....	5
Challenges of the current Architecture.....	6
Agile + DevOps.....	6
Containerisation	8
Orchestration.....	9
DevOps Practices	9
Infrastructure as Code (IaC)	11
Monitoring and Logging.....	11
Timelines	12
Recommendations	14
Conclusion	14

Executive Summary

The report offers advice on how to adopt cloud-native and explains why it is crucial for the PharmaC's continued growth. The company believes that transition to the cloud is a strategic direction, enabling a rapid pace of change in the future of the company. Adopting to Agile along with DevOps practises will boost the company's efficiency, cost-effectiveness, and collaboration while maintaining compliance and security. As there are many advantages of adopting the suggested cloud native architecture, there are several risks that are associated to them and must be mitigated.

Adapting to a cloud-native architecture has advantages such as increased agility, scalability, and cost savings. The software development lifecycle can be accelerated using agile and DevOps methodologies.

The company must aim to migrate 80% of current infrastructure to cloud native by the end of this year.

Summary of Challenges:

- Lack of In-house Expertise: Cloud-native solutions need high level of expertise dues to the complex nature.
- Inability to leverage the full potential of cloud infrastructure and services.
- Business Development: Releasing the latest product on-time in the market is challenging.
- Culture Shift from Agile to DevOps may raise some resistance to acceptance.

Summary of Recommendations:

- Invest in training to upskill the current employees or hire cloud-native developers.
- Adopt Containerisation and Microservices architecture to break the code into small parts and reuse them wherever possible.
- Automate the deployment, scaling and management of containerised applications using Kubernetes.
- Adopt cloud-native development tools like Jenkins for automation server and Gitlab for continuous integration and deployment (CI/CD).
- Using Infrastructure as a Code- Automated network monitoring using Terraform.
- Develop a roadmap for cloud-native adoption with specific timeline.
- Encourage collaboration between the development and operations teams by implementing DevOps practises.

Introduction

PharmaC Inc. is a mid-sized leading pharmaceutical company in the UK that specializes in the development and production of innovative medicines and healthcare products. The company was founded in 2000 and over the years gained recognition. The company has offices worldwide with operations in Europe, and Asia, and employees over 10,000 people worldwide. They produce, drugs, vaccines and medical devices. PharmaC follows strict regulatory compliance and standards to maintain quality of their products. The business is an early adopter of cloud computing and big data analysis.

The company has invested in cloud practises like hosting applications on cloud, managing backups and using data centres. The company also use agile software delivery model - Scrum, in which the teams work in short sprints and ensure continuous and collaborative improvement.

Even after all this, PharmaC is still unable to quickly deliver high-quality products to the market. Although the company has partially migrated to cloud, they didn't optimise the cloud infrastructure to achieve efficiency, scalability and automation. All the software delivery process along with testing is done manually. Due to a lack of automation and cloud-native practices PharmaC is unable to satisfy the demands of the market and customers. In order to reduce the bottlenecks and gain a competitive edge in the market, the company understands the necessity of implementing cloud-native development practices.

Methodology



Scope

This report aims to analyse the problems faced by PharmaC to adopt cloud-native practices and provides recommendation for overcoming them. As the company already follows agile practices for application development, the report emphasises the importance of DevOps practices in enabling a smooth transition to cloud-native development.

Objectives

The purpose of this report is to give PharmaC an improvised and reliable cloud native solution and architecture that will meet their needs and serve as a bridge for them to successfully adopt cloud native.

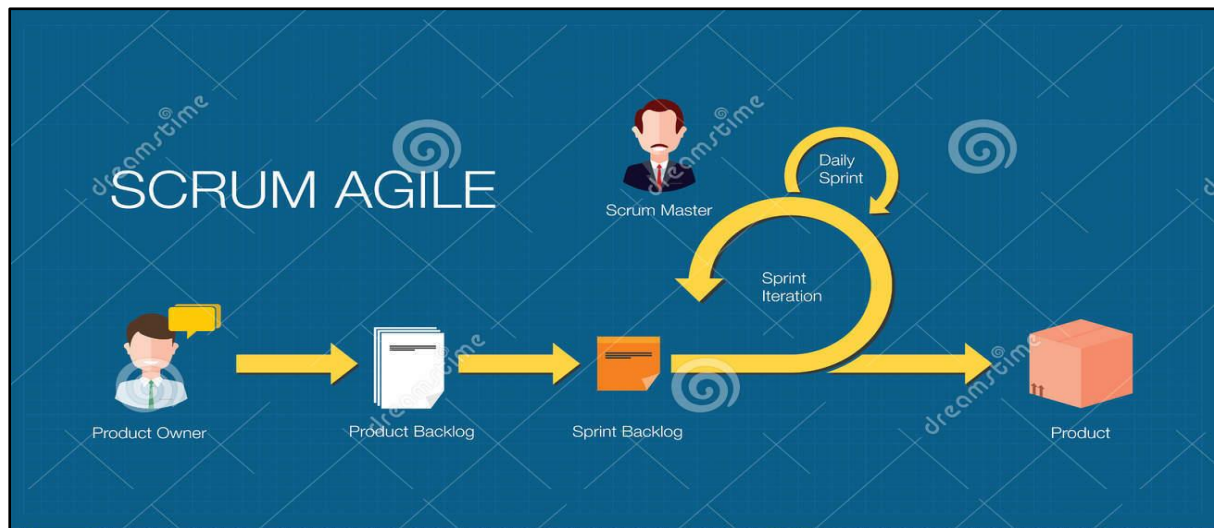
Assumptions

- It is assumed that the solutions and recommendations provided below are feasible.
- It is also assumed that the company is interested in adopting cloud native practices and is willing to invest in the same.

Challenges of the current Architecture

After the analysis of all the existing practices in the company, the following challenges can be drawn:

- Inefficient Application Development: The application development and deployment process does not meet the needs of market and is impacting the company's ability to keep up with the competition.
- Manual Testing: The company does the manual testing locally which can be time consuming.
- The applications might or might not be able to scale the infrastructure up and down as quickly and effectively as required.
- Slower Performance and increased costs as the company due to a lack of containerization and microservices architecture.
- Limited Collaboration Between the teams
- Digital transformation: Today's businesses must deal with constantly shifting demands. If a sudden emergency occurs, such as the COVID19 apocalypse in 2020, and urgent drug development is required, the current infrastructure must be able to handle the load.



Current Infrastructure for software delivery

Agile + DevOps

DevOps along with agile methodologies should be considered by businesses that want to transform into cloud-native organisations.¹ The company currently follows Scrum, which is an agile practice.

Some of areas of improvement in the existing infrastructure:

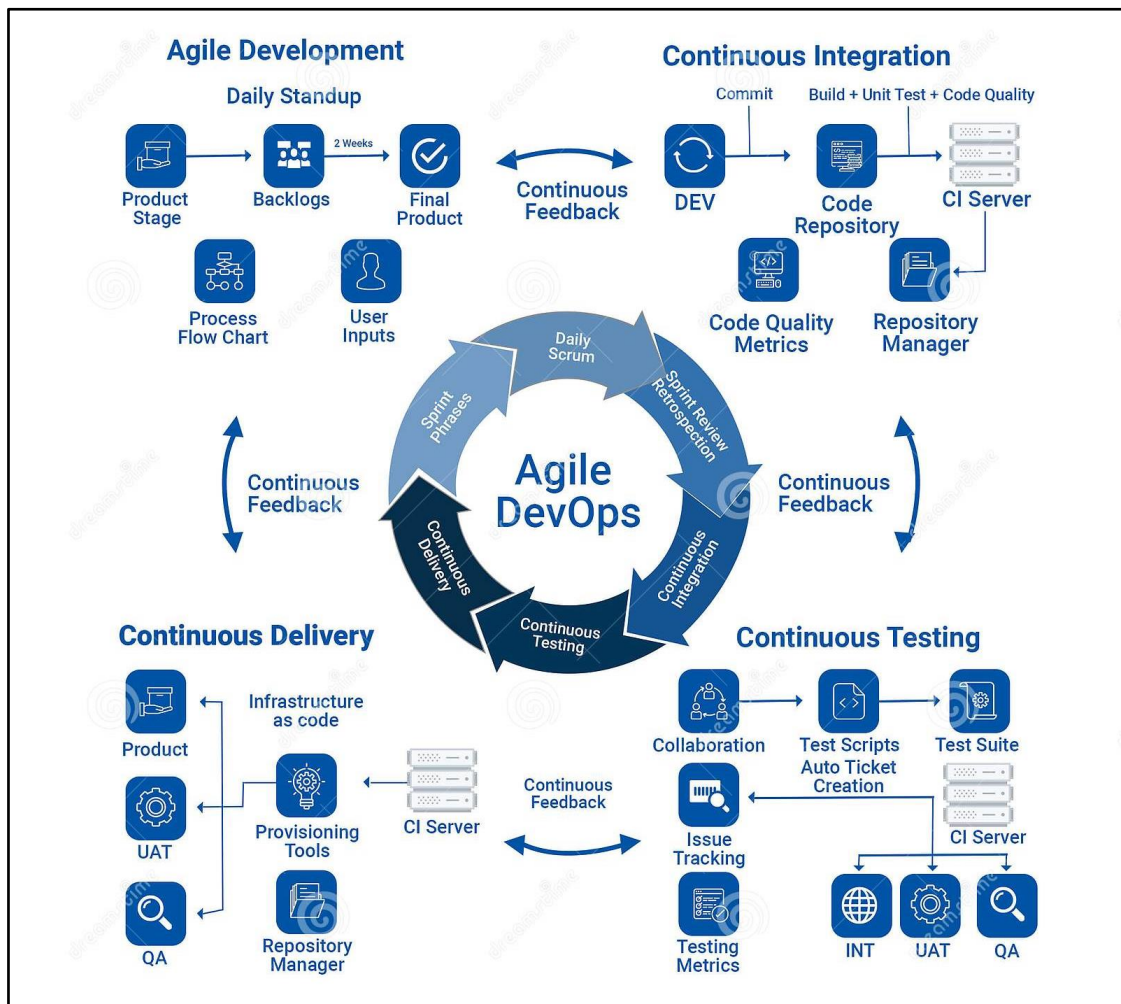
- Less focus on non-functional needs - As the company is currently using agile (Scrum), it is more focused on software development. But other factors like automation, testing, scalability and reliability are less focused on.
- Lack of Automation – The company performs the testing manually, which is expensive and time-consuming, and even then, mistakes can happen.
- Limited communication with Operations team- As the Agile practices focuses only on software development, they may not have much communication with operation's team.

¹ Laszewski, T., Arora, K., Farr, E. and Piyum Zonooz (2018). *Cloud Native Architectures*. [online] Packt Publishing Ltd. Available at: <https://www.perlego.com/book/800670/cloud-native-architectures-design-highavailability-and-costeffective-applications-for-the-cloud-pdf>

For these reasons, it is recommended to use a conjunction of DevOps and agile, as these bridge the gaps of agile production.

The main advantages of combining Agile and DevOps (Development and Operations) software delivery methodologies are:

- Improved Collaboration – Both Agile and DevOps focusses on collaboration and communication between Developers, Testers and Ops Teams. This will ensure that the company to release software more quickly and frequently while also improving team collaboration.²
- Better Software Quality – By leveraging automated and continuous testing, DevOps can help improve the quality of software and ensure there are no bugs and errors.
- Faster Product Delivery – If the products delivery speed is increased using these practices, they will be delivered faster to the market and in turn help with increased customer satisfaction.
- Increase Business Agility – As both Agile and DevOps help the company increase their adaption to the market, they can provide better products efficiently and in time. This will help them to stay up in the competition.



Proposed Agile + DevOps Architecture

² Hemon, A., Lyonnet, B., Rowe, F. and Fitzgerald, B. (2019). From Agile to DevOps: Smart Skills and Collaborations. *Information Systems Frontiers*, [online] 22(4), pp.927–945. doi:<https://doi.org/10.1007/s10796-019-09905-1>.

Cloud Native Solutions

The company must first ensure that their people and processes align with the cloud-native model.

The company can follow this roadmap towards cloud-native in phases as outlined below:



Containerisation

The first step is Containerisation. Containers have higher density, faster start-up times, and better portability. Additionally, it can enhance security and compliance as it provides a layer of abstraction between applications and their dependencies.³ As PharmaC already follow agile practices like Scrum, applications made using containers can be moved around in different environments like development, production or testing and are highly reproducible and reusable. And conjunction of Containers with Microservices will complement DevOps Practices of Continuous Integration and Continuous Deployment, allowing businesses to fully automate the software development lifecycle.

The company can further use Kanban to improve their agile adoption and collaboration between these teams. These steps can help PharmaC to enhance their existing agile practices and overcome the challenges of inefficiency in application development and insufficient collaboration between the teams.

A study done on a similar Indonesian pharmaceutical company shows that agile methodologies, particularly Scrum, Kanban and DevOps practices, can effectively address the challenges faced by the pharmaceutical industry in their IT infrastructure projects.⁴

³ Hajou, A., et al. "Method æ, the Agile Software Development Method Tailored for the Pharmaceutical Industry." *Lecture Notes on Software Engineering*, vol. 3, no. 4, 2015, pp. 251–262

⁴ Fitriani, A.N., Raharjo, T., Hardian, B. and Prasetyo, A. (2021). IT Infrastructure Agile Adoption for SD-WAN Project Implementation in Pharmaceutical Industry: Case Study of an Indonesian Company. *2021 IEEE International IOT, Electronics and Mechatronics Conference*

Orchestration

After the applications are containerised, Kubernetes⁵ must be implemented. Docker will provide and maintain the containers and Kubernetes will deploy them. For businesses like PharmaC that want to enhance the scalability, efficiency and reliability to perform IT operations, using both Docker and Kubernetes can be a powerful combination.

It will increase company's scalability, efficiency and. One of the benefits of Kubernetes is that there are automatic load-balancers, that can transfer traffic to other instances in times of peak demand. This also means that the applications will be highly available even without any human intervention.

Kubernetes will be managed and run on Kubernetes Cluster which is a collection of physical and virtual machines. It also provides the facility of auto-scaling the applications and offer self-healing properties (Automatically replace the damaged containers).

Although the business can choose another orchestration platform like Docker Swarm or Apache Mesos, Kubernetes is advised due to its popularity, extensive feature set, and community support.

DevOps Practices

Agile methodologies prioritise "working software over comprehensive documentation" and is an iterative and collaborative approach centred on continuous improvement and customer satisfaction.⁶ In addition to Agile practices, implementing DevOps practices will speed-up the process of deployment of new services and applications. This will enable the company to launch products in the market on-time. DevOps also helps with automated testing and deployment and helps reduce the risk of errors and bugs in the tech-stack. This will increase the quality of the applications developed.

Continuous Integration and Continuous Delivery, and automated testing will improve software quality and reduce time to market.

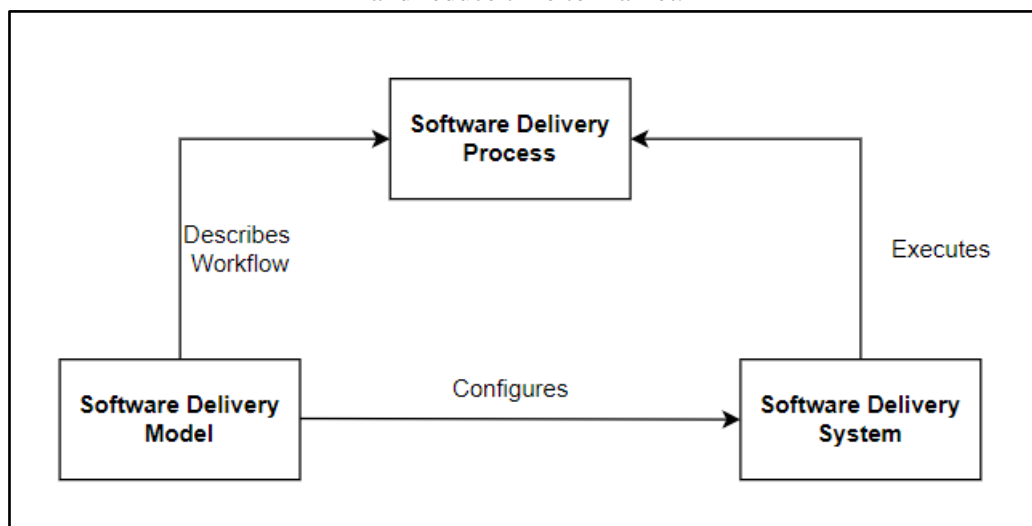


Figure 1 Continuous Integration and Continuous Development⁷

⁵ Production-Grade Container Orchestration (2023). *Production-Grade Container Orchestration*. [online] Kubernetes. Available at: <https://kubernetes.io/>

⁶ Hajou, A., Batenburg, R. and Jansen, S. (2014). How the Pharmaceutical Industry and Agile Software Development Methods Conflict: A Systematic Literature Review. *2014 14th International Conference on Computational Science and Its Applications*.

⁷ ACM Conferences. (2018). *Designing a next-generation continuous software delivery system | Proceedings of the 4th International Workshop on Rapid Continuous Software Engineering*.

The aim for using Continuous Integration and Continuous Deployment (CI/CD) in PharmaC is to ensure that if there are any updates or changes in the code of the application, they are tested and then deployed quickly and reliably without human intervention. The following diagram provides an insight into the tools that can be used in CI/CD system in PharmaC.

The CI/CD can be divided into 3 parts⁸:

- Software Delivery Model:** This refers to the practices like Agile, DevOps and DevSecOps. These describes the methodology used in software development life cycle.
- Software Delivery Process:** These includes all the process in software development such as coding, testing and delivering the software.
- Software Delivery System:** This is the underlying technical infrastructure which ensure continuous delivery of high-quality software. These includes the feedback management tools, monitoring tools etc.

There are obviously other tools that can be used, but the most recommended and widely used are mentioned below :

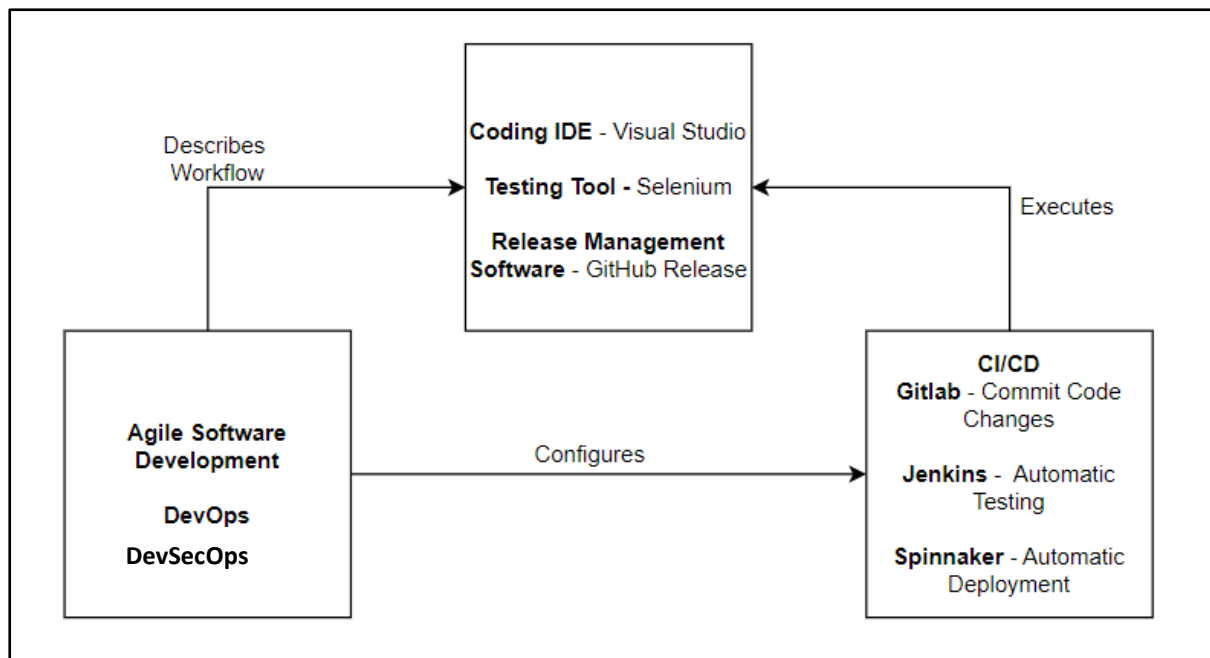


Figure 2 The recommended CI/CD flow for PharmaC

Rather than being an afterthought or a collection of isolated technical procedures, security should be a fundamental aspect of an organization's culture and values.⁹ And to enhance the existing security posture, DevSecOps will be implemented. Although the Company's main business is developing software's, but security should be considered at every phase of the cloud-native application development process. It should be considered as a culture.¹⁰ DevSecOps will reduce the security incidents and can be mitigated in the early stages of production, to avoid loss of time and revenue.

⁸ ibid

⁹ Sánchez-Gordón, M. and Colomo-Palacios, R. (2020). Security as Culture. *Proceedings of the IEEE/ACM 42nd International Conference on Software Engineering Workshops*.

¹⁰ ACM Conferences. (2018). *Designing a next-generation continuous software delivery system / Proceedings of the 4th International Workshop on Rapid Continuous Software Engineering*.

The tools that can be used are Static Application Security Testing (SAST) like SonarQube and Dynamic Application Security Testing (DAST) like OWASP ZAP.

Implementing DevSecOps can be initially expensive, but the long term benefits can outweigh the cost easily. Additionally, it is crucial to comply with standards because the company handles private and sensitive data like patient records and drug management systems. While ensuring regulatory compliance, the advantages of quicker delivery and better quality can be realised using these technologies.¹¹

Infrastructure as Code (IaC)

After all the development and deployment, comes the provisioning and the management of all the infrastructure. As the company was an early adopter of cloud, automating the infrastructure provisioning process will be easier. Instead of manually configuring resources like servers, databases, and networks, they are defined and configured using code. Using IaC will ensure that the code is reliable and efficient as the whole process will be automated and therefore the risk of errors and down-time is eliminated.

A company can easily specify their desired configuration in the code if they choose to add more servers in the future. The code can be versioned, tested, and automated according to the requirements. Additionally, the server will be automatically configured in accordance with that configuration using tools like Terraform.

This can help with keeping the infrastructure consistent and minimise any unpredictable activity without any human intervention, for example, It is critical to confirm where the server is running (regions) and choose to abide by the relevant rules and data privacy laws like HIPPA (Health Insurance Portability and Accountability Act) and GDPR (General Data Protection Regulation). And with the help of IaC, the company can use the same code to replicate the servers in different environments. This complies with the company's aim to improve scalability, efficiency and reliability. Moreover, Infrastructure code can benefit from version control and automated testing to increase its dependability and quality.¹²

Apart from all these benefits and functionalities, implementing IaC will cut down the expenses of huge infrastructure management which is done manually. These benefits will out-weight the initial cost and provide efficiency to the whole cloud native model.

Monitoring and Logging

The last phase would be ensuring that the applications work the way they are intended.

Logging - One popular and user-friendly way to logging is Elasticsearch, Logstash and Kibana (ELK) stack. The company will be able to collect, process, and analyse logs generated by its infrastructure and applications, allowing for real-time visibility into the system¹³. Application logs are stored and indexed in Elasticsearch, collected and processed in Logstash, and visualised and analysed in Kibana. This will provide a centralised logging platform, making it easier to identify and troubleshoot issues as they arise.¹⁴

¹¹ Laukkarinen, T., Kuusinen, K. and Mikkonen, T. (2017). DevOps in Regulated Software Development: Case Medical Devices. 2017 IEEE/ACM 39th International Conference on Software Engineering: New Ideas and Emerging Technologies Results Track (ICSE-NIER).

¹² Artac, M., Borovssak, T., Di Nitto, E., Guerriero, M. and Tamburri, D.A. (2017). DevOps: Introducing Infrastructure-as-Code. 2017 IEEE/ACM 39th International Conference on Software Engineering Companion (ICSE-C).

¹³ Ahmed, F., Jahangir, U., Rahim, H., Ali, K. and Agha, D.-S. (2020). Centralized Log Management Using Elasticsearch, Logstash and Kibana. 2020 International Conference on Information Science and Communication Technology (ICISCT).

¹⁴ ibid

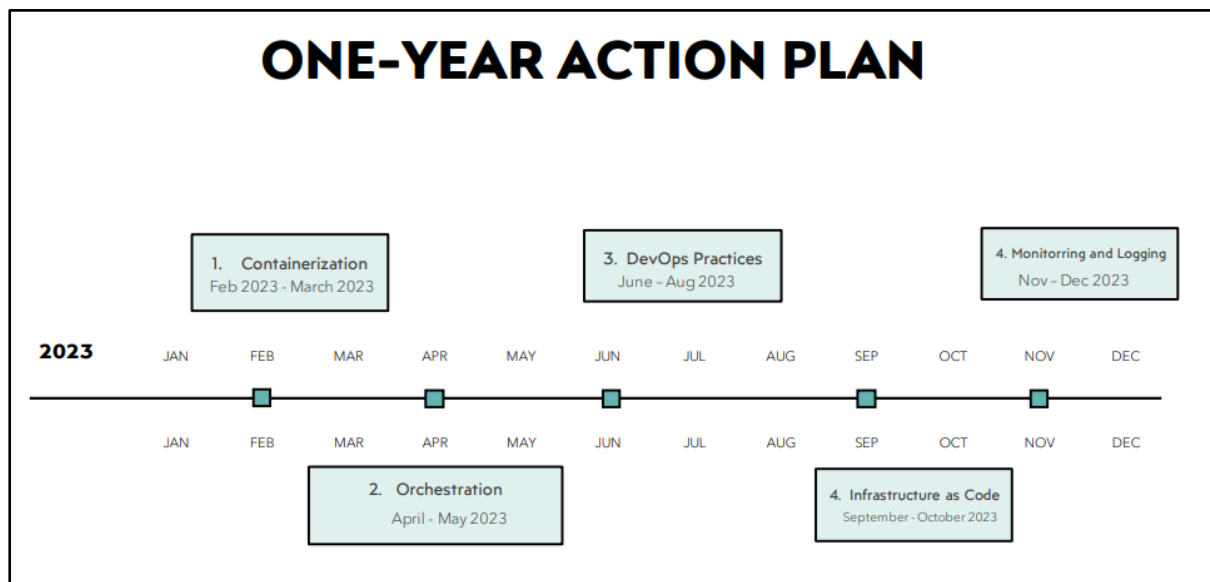
Infrastructure Monitoring – This will allow the Operations, Development, and Security teams to monitor the system's status. Key metrics such as system health, CPU usage, file system usage, and network activity will be tracked by the infrastructure monitoring system.¹⁵

Application Monitoring – The company will be able to monitor Continuous Integration and Continuous Delivery (CI/CD), this involves monitoring the different stages of the pipeline, from code commit to production deployment. Other processes, such as the resources used by the applications and the error rate, can also be tracked.

Prometheus¹⁶ can be used for both Infrastructure Monitoring and Application Monitoring as it provides alerting and data visualisation facility. It is simple and can be integrated with any platform.

Timelines

The entire timeline for the company to completely migrate to cloud-native methodology will be between 7-8 months. Although this does not include the time needed to train the professionals in order to adapt to cloud native environment.



¹⁵ ibid

¹⁶ Prometheus (2014). *Prometheus - Monitoring system & time series database*. [online] Prometheus.io. Available at: <https://prometheus.io/>

Risks Associated with Migrating to Cloud-Native Architecture:

Risks	Risk Description	Mitigation	Risk Likelihood	Risk Severity
Compliance and Regulation Violations	Pharmaceutical companies like PharmaC, deal with a lot of sensitive data such as patient records and therefore must adhere to the strict regulatory standards like HIPPA and GDPR. And the fines associated to the violation of these standards can be huge.	To avoid this risk, PharmaC can do an assessment of the regulatory standards and fill in the gaps accordingly. They can also setup compliance monitoring and reporting.	High	High
Data Breaches	Cloud-Native architectures are vulnerable to data breaches and cyber-attacks, and the loss of sensitive data can result in significant financial losses and fines.	<ul style="list-style-type: none"> - Implement encryption. - Two-Factor Authentication - Access Control 	High	High
Vendor Lock-In	When the company migrates to cloud-native and select a suitable vendor, it will be very difficult to change the vendor later.	To avoid this situation, the company can adopt multi-cloud approach instead of relying on one.	Moderate	Moderate
Loss of control over the infrastructure	In cloud-native infrastructure, the company won't have any transparency about the infrastructure used as it will be managed by the cloud provider. This will make the company dependent on the cloud provider.	Training the staff or hire cloud native skilled employees.	Moderate	Moderate
Resistance to cultural shift	As the company will shift from agile to DevOps, the work culture will change and if the employees are not well versed with the cloud-native culture, they may find it hard to work with.	Provide training to employees so that they can shift to cloud-native comfortably.	High	High

Recommendations

Apart from the mitigations that are mentioned in the table, the company can follow the recommendations below:

- Gain a clear understanding of cloud native architecture - Because the company currently uses Scrum (agile), it's crucial to gain a better understanding of how to use cloud native architecture. And this can be accomplished through routine staff training, progress monitoring, and logging.
- Conduct Thorough risk assessments – As there can be some unseen risks associated with this migration like legacy softwares or incompatibility of traditional hardware with cloud-native architecture.
- Invest in employee training - Because the transition from Agile to Agile+DevOps can be challenging for staff members, adequate training is necessary to prevent any unintended consequences of human error.
- Take Security Measures - Because software development is not a pharmaceutical company's primary business, sometimes businesses tend to ignore the significant problems it raises. To ensure the protection of sensitive data and information, the company should implement strong security.

Conclusion

Migrating to cloud native technologies will benefit PharmaC with all the “ilities” that are required by a pharmaceutical company to be competitive in the market. Although there are some risks related to these, but in a longer run, with proper training and implementation, these risks can be mitigated.

To successfully complete the transition the cloud native, the company and follow the plan as well as timeline. They should work with the cloud provider closely and be clear about the necessary support and infrastructure . Pharmaceutical businesses can maintain the highest levels of security, compliance, and quality while also remaining competitive and providing better customer service by implementing a cloud native architecture and DevOps.



