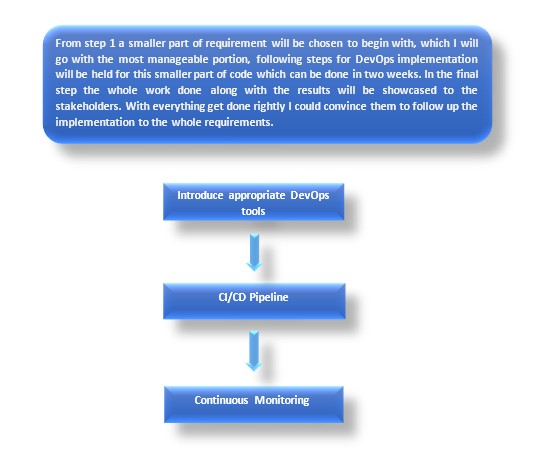
DEVOPS IMPLEMENTATION

To adopt DevOps to the company I have designed a simple strategy by integrating DevOps tools, processes, and data into traditional the work culture followed in the company. For that I have broken down the process into steps tackling one DevOps aspect at a time as follows:

**Step 1: Gather the requirements.** Start from collecting requirements, identify the important aspects such as knowing the current work culture of the company, various phases and structures of codes used, tools and infrastructures, and the bottlenecks for downturn to get an idea of what went wrong. As I have strict time bound for this, I will have a heads up with Pat and the stakeholders which in this case can be the company managers who wants the company to rise up from the dire strait, to sum up the important aspects stated earlier in this step.

**From this step a smaller part of requirement will be chosen to begin with, which I will go with the most manageable portion, following steps for DevOps implementation will be held for this smaller part of code which can be done in two weeks. In the final step the whole work done along with the results will be showcased to the stakeholders. With everything get done rightly I could convince them to follow up the implementation to the whole requirements.**



**Step 2: Introduce appropriate DevOps tools.** After collecting the requirements having an idea on how the company works, the initial step for actual DevOps process can be brought to picture with the introduction of tools. I prefer to stick with the popular and efficient DevOps tools available in market for various phases in the DevOps cycle. For example, Git Hub, Selenium, Jenkins, Docker, etc. for collaborative coding, testing, CI/CD pipeline, IaC respectively. Why I prefer to stick up with these tools is because developing a custom DevOps solution would be time-consuming which in this case is strictly constrained and would likely result in integration and scalability problems.

A team meeting will be set up with three of us to get a buy in from each team member. Ren will be entirely new to the DevOps culture but he will be well known about the traditional culture going on in the company, it would help us to crack on how and where to integrate the DevOps tools. Coming to Jalen he will be an enthusiast to introduce change to an existing system with fresh skillset which can be well utilized to play with the tools. Taking their inputs, I could well organise and efficiently develop the integration of DevOps tools.

**Step 3: CI/CD Pipeline**. Now having the appropriate DevOps tools in place, start developing the CI/CD pipeline one step at a time incrementally. A typical CI/CD pipeline can be followed which would consist of stages where code gets pushed to the repository, the build gets triggered, build is tested and deployed to the production environment. For each phase we have the automated DevOps tools integrated in the previous step.

Examining each phase in detail, before pushing to repository the code needs to be modernized as it’s a legacy code and it’s in urge for modernization process as per the given case. For that a modernization framework would be set up and implemented to the existing code and then it goes to the repository. Progressing to next phase build gets triggered and deployed into a test environment where automated tests are executed and tailed by continuous feedback testing loop. Then the final code is deployed to production environments with the appropriate IaC tool assigned in step 2.

**Step 4: Continuous Monitoring.** The deployment should be followed by continuous and proactive monitoring as it is essential to optimize the performance. The stakeholders can be brought to this step where I can **hand over the deployed system showcasing the work** my team, along with that they could monitor whether their requirements and the system delivered are going the right path. Several DevOps tools can be used here for example Splunk, DataDog, etc. which are quite popular and helps to attain alerts, graphs, reports and visualizations.

**CODE MODERNIZATION**

Modernising a legacy code involves challenges as we will be dealing with older code and the code which we didn’t write, can be an unfamiliar programming language, may be a poorly organized structure which is difficult to understand and so on. So here I am building a modernization process which helps overcome these challenges and implement the built process for fixing the given piece of broken PowerShell code.

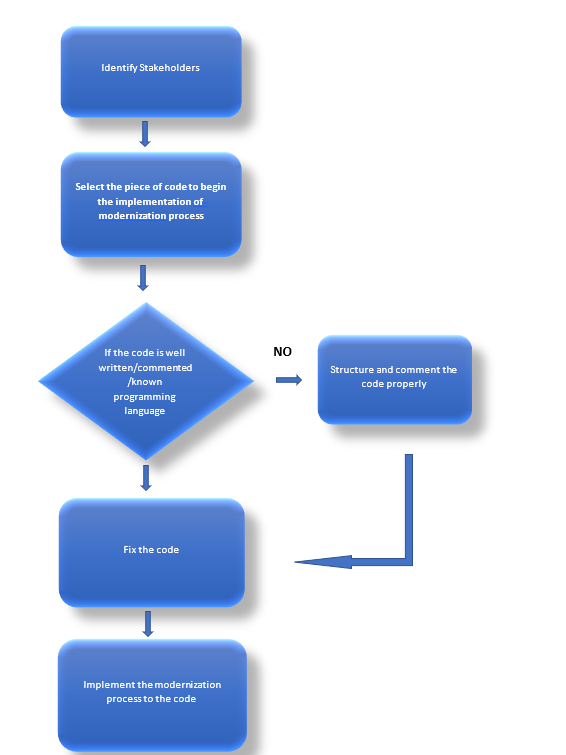
My strategy for developing this process involves certain steps and scenarios where the initial steps would be same and on progressing it may change according to different scenarios. Firstly, examine the piece of code and categorize it into one of the following scenarios.

**Scenario 1:** Well written and understandable/appropriate comments/ team member knows the language.

**Scenario 2:** NotWell written and understandable/no proper comments or structure/ no team members know the language

**Step 1**: Identify the stakeholders and gain contacts with help of Ren. They will be knowing the who knows the **key things to be fixed, prioritized, categorize into critical one or which can bring in more revenue with low cost of running.**

**From this step a smaller piece of code will be chosen to begin with, which I could select with the inputs I gathered from the stakeholders. By modernizing that smaller piece of code and getting it right and efficient I can convince the stakeholders to modernize the whole code.**



**Step 3:** If the code comes under scenario 2, first and foremost we need to gain people who knows the programming language in which the code is written. This can be done with help of Ren. After that with the help of that person the code can be structured and commented properly in a way that anyone could understand the code. Here we can use the skill set of Jalen who will catch up with code faster and could easily provide efficient inputs for modernizing the code.

**Step 4:** Review, Analyse, organize a plan within the team to fix the code and modernize. Small meetings can be held in between to discuss on various inputs each of the team member needs to provide. Utilize Jalen to integrate new technologies.

The given PowerShell code can be fixed by following the previous steps and can be modernized further by:

* Restructuring and optimizing the code and thus to remove technical debt and improve non-functional attributes.
* The data and functions can be encapsulated and make them available as services via an API.
* Code can be altered in such a way as to shift into new architectures and exploit new and better capabilities.

CONCLUSION

This assignment helped me to explore various aspects of DevOps implementation and code modernization and integrating these concepts to a running company, that too under certain constraints. Designing a DevOps implementation process paved the way to research more on DevOps processes and various tools used in DevOps, to efficiently implement the tools. Various techniques to modernize a legacy code was learnt throughout the assignment.

My strategy to build the DevOps process was to break down the process into several steps taking one DevOps aspect at a time and I had started the whole process with smaller part of the requirements and then the idea was to progress it into the whole requirements after convincing the stakeholders with the smaller implementation done. Same approach was taken for the code modernization process as well, taking a smaller piece of code initially and then progressing to the whole code. Modernization process was implemented step by step through different scenarios. Several techniques to modernize a legacy code were used in the process and was implemented to the given broken PowerShell code as well.

With the given scenario of Shinty Software Company, several challenges were given to be taken care of while designing and implementing the DevOps and code Modernization. Time constraints, downturns, traditional work culture, small team of two people with entirely different characteristics – one is experienced, other is a fresher which again could raise some conflicts in various situations, all these were the challenges that had to be addressed while designing each step of both the DevOps implementation process and the code modernization process.

Creating a framework under several constraints were quite challenging but it has made a major impact on my approaches towards DevOps processes and I have become more efficient than earlier which would help in future as anything can happen in real time projects and in that cases definitely the impact attained by this assignment will help me face the challenges and design an efficient DevOps framework.

References

Gartner. (2019). 7 Options to Modernize Legacy Systems. [online] Available at: https://www.gartner.com/smarterwithgartner/7-options-to-modernize-legacy-systems [Accessed 24 Nov. 2022].

Gill, Navdeep Singh. (2021). DevOps Implementation and Adoption Strategy [online] XENONSTACK. Available at: https://www.xenonstack.com/blog/devops-implementation [Accessed 24 Nov. 2022].

Kalinin, Konstantin. (2022). Why DevOps strategy is a must for your Business Implementation Steps, Plan and Examples [online] topflight. Available at: https://topflightapps.com/ideas/how-to-implement-devops/#6 [Accessed 24 Nov. 2022].