

Memo

To: Tim Smith
From: Eric Wellmaker
cc: XXXXXXXX
Date: November 13, 2024
Re: Written Assignment 12.2

Preface

Per the project scenario, Tim Smith is impressed. For this assignment, I will explain the differences between SDLC and Dev Ops. I will also go over what DevOps is, if it replaces an SDLC and if DevOps replaces or augments phases in an SDLC.

11/13/2024

Executive Summary

Objective

Tim Smith needs clarification about the differences between SDLC and DevOps. To accomplish this, I will explain the differences between SDLC (Software Development Life Cycle) and DevOps (Development Operations). I will also discuss whether DevOps replaces an SDLC and also inform how DevOps relates to the phases of the SDLC.

Target Audience

Tim Smith, owner of Downtown Car Repair Business. Tim makes a profit by giving services to large and small auto repair jobs. Small jobs include oil changes and tire rotations, while large jobs include transmission and engine swaps. Tim bids on these repairs with the customers and if both parties are in agreement, then Tim begins the work. Payments are either paid in full or by monthly installments. If a monthly installment is late one month late, it will accrue ten percent late fee. Two months late, the payment will go to a collection agency. If the collection agency cannot collect the payment by the end of the third month, then Tim isn't charged by the collection agency for their services.

Competition

As aforementioned in the previous section, Tim's competition is the collection agency. Once payments reach two months late, then the collection agency initiates to attempt

to collect a payment for non-payment of Tims services. If non-payment exceeds three months, then Tim does not have to share the profits with the collection agency.

What is DevOPs?

DevOps (Development and Operations) uses a system of collaboration and continuous feedback from customers. This approach ensures the software creation process is streamlined from imagination to creation. This streamline process is accomplished by the customers keeping in contact with the software developers at each of the eight phases in the creation cycle. These cycles include Planning, Coding, Building, Testing, Releasing, Deploying, Operating, and Monitoring. Peering deeper into the creation cycle (The eight phases of DevOps., 2024).

- Planning includes deciding how the product will look overall. This means developers will start building the source code here.
- Coding will be completed after the developers and customers come to an agreement on how the software will be written.
- Building occurs after the source code is written and tested.
- Testing involves white and black box (sometimes grey box) testing to verify the software is ready to be released to the customer.
- Release is the second half of DevOps phase. It is here that the goal is to ensure the code that was written is properly configured for a successful release.
- Deployment is placing into production the software that was created for the clients' use.

- Operate is developers keeping the product operational.
- Monitor is obtaining clients feedback and monitoring the software that was created. This is the final phase of DevOps, so it is advised that feedback given from the client be used at the next software creation cycle.
(The eight phases of DevOps., 2024).

Does DevOps replace SDLC?

Development Operations (DevOps) doesn't replace the Software Development Lifecycle (SDLC). Instead, this process integrates the waterfall approach of SDLC into DevOps. This means collaboration and feedback is always at the forefront meaning clients stay in contact with developers. This approach results in timelines not being missed, and if changes do occur, it can be accomplished without wasting time or costs (Nalawade, P., 2024).

How DevOps augments SDLC

DevOps was created in such a way that it can be used at the same time as SDLC. DevOps is where clients and developers work together to create a product. Here is how DevOps overlaps SDLC.

- SDLC analysis phase overlaps the planning stage of DevOps. Both are focused on planning and understanding what the clients want from the product.

- The Design phase of SDLC is the system design, security protocols and type of programming language. Overlapping wise, DevOps doesn't apply here.
- Development is ensuring the code is executable. DevOps aligns here through the coding and build phase.
- Testing from the SDLC overlaps DevOps here. Both focus on ensuring the code is ready for the client.
- Deployment from the SDLC and Deploy from DevOps are two sides of the same coin. Both ensure the product is presented to the client.

Conclusion

DevOps and the SDLC are needed to ensure a smooth creation process for software development. The goal purpose is to keep in contact with the client so the product that is created is what was agreed upon. SDLC's waterfall approach is an A-to-Z method where processes go from one phase to the next. DevOps, these processes instead are communicated to the client to keep them informed and to also apply changes if necessary. While different in scope both DevOps and SDLC are needed to present a product to a client.

References

Nalawade, P. (2024). SDLC vs DevOps. Medium.

<https://medium.com/@nalawade1000work/sdlc-vs-devops-25ff2dd0acdf>

The eight phases of DevOps. (2024). PlanetScale.

<https://planetscale.com/docs/devops/intro-to-the-eight-phases-of-devops>