**SEGR 5910 Midterm Exam Key**

This is the exam key for the 18WQ mid-term exam.

1. **In your opinion, does DevOps reduce or increase development and / or operational costs. Defend your opinion.**

*This is an open question. It’s your opinion, which means it can’t really be incorrect, but it can be ill-defended. What I’m looking for here is an understanding of what we’ve talked about so far this quarter, a position, defense of that position in the terms that we’ve been using.*

*Here’s one.*

*In many cases DevOps will have a positive impact on development and operational costs, that is, it’ll reduce them and, in many cases, reduce them significantly. However, there will be an immediate increase in costs as you turn your more expensive development resources to work on these issues. Keep in mind that a developer is likely to cost nearly 50% more than your average operations engineer. At the outset, and until the repeatable operations are automated, you’ll be spending 50% more. In general, you’ll need to “automate away” 1/3 of your work before you hit a break-even point. Over time, the increased efficiencies from automation, (re)work reduction, and resource churn (keeping operations engineers happy is hard, they tend to have a short shelf life, which increases your training costs), will balance, and you’ll see a net positive impact.*

1. **A goal of DevOps is to reduce the time to market of a given product or feature. How does it do this and in what ways might it increase time market?**

*Simply put, time-to-market is decreased through those same efficiencies mentioned above: reduction in mistakes and rework, and shortened time between “commit” and “release”. However, while your development team is now able to release features faster or get them through the core CI/CD pipeline faster, there’s still the issue that some of those developers are focused on DevOps work and not on “feature” work. Clearly, if those developers can’t build new features, there’s no way those features can get to market.*

1. **What are the NIST cloud computing models? Give concrete exams of each and explain how and when an organization might adopt one or more of them.**

*I’m looking for the following three computing models: Software as a Service (SAAS), Platform as a Service (PAAS), and Infrastructure as a Service (IAAS).*

|  |  |
| --- | --- |
| SAAS | The capability provided to the consumer is to use the provider’s applications running on a cloud infrastructure. The applications are accessible from various client devices through either a thin client interface, such as a web browser (e.g., web-based email), or a program interface. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or even individual application capabilities, with the possible exception of limited user specific application configuration settings. |
| PAAS | The capability provided to the consumer is to deploy onto the cloud infrastructure consumer-created or acquired applications created using programming languages, libraries, services, and tools supported by the provider. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly configuration settings for the application-hosting environment. |
| IAAS | The capability provided to the consumer is to provision processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications. The consumer does not manage or control the underlying cloud infrastructure but has control over operating systems, storage, and deployed applications; and possibly limited control of select networking components (e.g., host firewalls). |

*You will likely choose SAAS when you’re looking for a solution and are not going to build your own application. Think Google Docs, Office365, Salesforce, Workday, Atlassian.*

*You will likely choose PAAS when you’re building an application but are not focused on getting the absolute most out of the infrastructure. Or, you don’t have the development or operations resources and skills necessary to configure your own networking stack, storage configuration, or operating systems. In this case you’re relying on a third party to provide this functionality for you. Azure and most AWS offerings are of this variety.*

*Finally, you’ll choose IAAS if you’re building software from the ground up, either to be run internally or to be sold as a SAAS offering. Both Microsoft and Amazon sell IAAS products.*

1. **NIST identifies five essential cloud computing characteristics. What are they? Do not simply enumerate the list.**

***On-demand self-service****. A consumer can unilaterally provision computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with each service provider.*

***Broad network access****. Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g., mobile phones, tablets, laptops, and workstations).*

***Resource pooling****. The provider’s computing resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand. There is a sense of location independence in that the customer generally has no control or knowledge over the exact location of the provided resources but may be able to specify location at a higher level of abstraction (e.g., country, state, or datacenter). Examples of resources include storage, processing, memory, and network bandwidth.*

***Rapid elasticity****. Capabilities can be elastically provisioned and released, in some cases automatically, to scale rapidly outward and inward commensurate with demand. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be appropriated in any quantity at any time.*

***Measured service****. Cloud systems automatically control and optimize resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts). Resource usage can be monitored, controlled, and reported, providing transparency for both the provider and consumer of the utilized service.*

1. **How does DevOps use and define an error budget? What are some tensions that cause DevOps to use this tool?**

*Product Management defines the service level objective, this is the overall application availability measurement. Then development measures that metric (remember, this is a unit of measurement, a value, and a timestamp) using some neutral third party (a monitoring solution, for example). We track this over time and calculate the difference between planned and actual. This gives us an idea of how many units of that measurement are remaining. As long as there are some remaining units, development is free to ship a release.*

*Now, that’s pretty theoretical, and it’s wide open to interpretation because I’m skipping out on the part where Product Management has to decide what “availability” means to them, or whatever other measurement they deem important to the business.*

*The tensions are easy to discuss. Development is under pressure to release product as fast as a possible to as many customers as possible. There are any number of reasons for this, but the reasons ore actually unimportant to the discussion. Operations, on the other hand, do not like to touch a running system. This is based on the solid belief (truth?) that nearly all breakages can be traced back to some change. Operations are tasked with keeping a system running and available at all times. Therefore, Dev’s desire to change the system coupled with Op’s desire to not change the system creates an inherent conflict of interest. Error budgets are a tool used to balance this conflict.*

1. **Early this quarter we talked about the DevOps process and how it relates to other SDLCs. Using this graphic as a launching point, explain how we layer the DevOps process on it.**

*This question can be directly answered from a slide presented the first evening. Overall, the DevOps process looks and acts very much like another other SDLC. Each of the above steps in the process should involve DevOps in the following ways:*

|  |  |
| --- | --- |
| Requirements | Treat operations personnel as first-class stakeholders  Get their input when developing requirements |
| Development | Small teams  Limited coordination  Unit tests |
| Build | Build tools  Supports continuous integration |
| Testing | Automated testing  User acceptance testing |
| Deployment | Deployment tools  Supports continuous deployment |
| Execution | Monitoring  Responding to error conditions |