Free Questions for SAFe-DevOps by certsdeals

Shared by Farmer on 22-03-2024

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

What does the %C&A metric measure in the Continuous Delivery Pipeline?

Options:

A- The percent concurrent and accurate process times of each pipeline activity

B- The percent complete and average times of each pipeline activity

C- The percent of change averages of each pipeline activity

D- The percent of time downstream customers receive work that is usable as-is

Answer:

D

Explanation:

The %C&A metric measures the percent of time downstream customers receive work that is usable as-is in the Continuous Delivery Pipeline. The %C&A metric is a measure of the quality of the work output from a process step. It indicates the percentage of time that the downstream customers receive work that is acceptable as is, without any rework or errors. To calculate the %C&A metric, the people responsible for the next step need to inspect the work output from the previous step and determine whether it meets the quality standards and expectations. They also need to provide feedback to the previous step on the defects or issues they find and how they affect the value stream. By measuring the %C&A metric, the enterprise can gain insight into the current state of the delivery process, such as the lead time, cycle time, throughput, quality, and waste. This insight can help the enterprise identify bottlenecks, dependencies, handoffs, delays, and inefficiencies that affect the flow of value. The %C&A metric can also help the enterprise understand how the flow of value can be improved by applying the principles and practices of DevOps, such as culture, automation, lean flow, measurement, and recovery.By improving the %C&A metric, the enterprise can increase customer satisfaction, reduce costs, accelerate time-to-market, and enhance business agility

Question 2

Question Type: **MultipleChoice**

Who is responsible for ensuring quality is built into the code in SAFe?

Options:

A- Product Owners

B- Agile Teams

C- Testers - xx

D- Developers

Answer:

B

Explanation:

Agile Teams are responsible for ensuring quality is built into the code in SAFe. SAFe is a framework for scaling agile practices across the enterprise, based on the principles of Lean, Agile, and DevOps. SAFe organizes the enterprise into Agile Release Trains (ARTs), which are teams of Agile Teams that deliver value in a regular cadence. Agile Teams are the fundamental building blocks of SAFe, and they are cross-functional, self-organizing, and self-managing teams that deliver value in short iterations. Agile Teams are responsible for ensuring quality is built into the code in SAFe, by applying the following practices:

Test-first -- Test-first is a practice that involves writing tests before writing code, to ensure that the code meets the requirements and standards, and does not introduce any defects or vulnerabilities. Test-first helps to improve the design and maintainability of the code, and to accelerate the feedback and validation process. Test-first can be implemented using various techniques, such as Test-Driven Development (TDD), Behavior-Driven Development (BDD), or Acceptance Test-Driven Development (ATDD).

Built-in quality -- Built-in quality is a practice that involves applying quality standards and checks throughout the solution lifecycle, rather than inspecting quality after the fact. Built-in quality helps to prevent defects from escaping to downstream stages or customers, and to reduce the cost and risk of rework and waste. Built-in quality can be achieved using various methods, such as code quality and security analysis, code review, pair programming, refactoring, and continuous testing.

Continuous testing -- Continuous testing is a practice that involves automating and executing tests at every stage of the Continuous Delivery Pipeline, to verify that the solution meets the functional and nonfunctional requirements and standards, and to detect and resolve any issues or defects as early as possible. Continuous testing helps to ensure the reliability and performance of the solution, and to support the delivery and deployment of value to the customer.Continuous testing can be performed using various tools and frameworks, such as unit testing, integration testing, system testing, acceptance testing, performance testing, and security testing

Question 3

Question Type: **MultipleChoice**

What are two activities performed as part of defining the hypothesis in Continuous Exploration? (Choose two.)

Options:

A- Elicit feedback

B- Identify Metrics based on leading indicators

C- Use value stream mapping

D- Define the minimum viable product

E- Develop a detailed business case

Answer:

B, D

Explanation:

Two activities performed as part of defining the hypothesis in Continuous Exploration are identifying metrics based on leading indicators and defining the minimum viable product. Continuous Exploration (CE) is an aspect of the Continuous Delivery Pipeline that drives innovation and fosters alignment on what should be built by continually exploring the market and customer needs, defining a vision, roadmap, and set of features for a solution that addresses those needs. CE is based on applying customer-centric and design thinking to understand and create alignment on new development opportunities, while recognizing that all such ideas are hypotheses that need to be validated. CE involves four activities: hypothesize, collaborate and research, synthesize, and validate. The hypothesize activity describes the practices for generating ideas and the measurements needed to validate them with customers. The hypothesize activity involves the following practices:

Identifying metrics based on leading indicators -- Leading indicators are metrics that measure the expected outcomes and benefits of the solution, such as customer satisfaction, retention, engagement, and revenue. Leading indicators help to evaluate the validity of the hypotheses and assumptions about the customer value proposition, and to guide the decision making and prioritization of the features. Leading indicators are also known as key performance indicators (KPIs) or objectives and key results (OKRs).

Defining the minimum viable product (MVP) -- An MVP is a version of the solution that has just enough features to test the hypotheses and assumptions about the customer value proposition, and to elicit feedback from the customer. An MVP is not a fully functional or polished product, but rather a learning vehicle that allows the enterprise to validate the problem-solution fit and the product-market fit.An MVP helps to reduce the uncertainty and risk of developing the wrong solution, and to accelerate the learning and discovery process11

Question 4

Question Type: **MultipleChoice**

What are two benefits of DevOps? (Choose two.)

Options:

A- Less frequent deployments

B- More lead time

C- Less time spent on new work

D- Fewer defects

E- Less time spent fixing security issues

Answer:

D, E

Explanation:

Two benefits of DevOps are fewer defects and less time spent fixing security issues. DevOps is a mindset, culture, and set of technical practices that supports the integration, automation, and collaboration needed to effectively develop and operate a solution. DevOps helps break down organizational silos and develop a Continuous Delivery Pipeline --- a high-performance innovation engine capable of delivering market-leading solutions at the speed of business. DevOps has many benefits for the enterprise, such as:

Fewer defects -- DevOps improves the quality and consistency of the solution by enforcing frequent testing and validation throughout the solution lifecycle. DevOps applies various testing techniques and tools, such as unit testing, integration testing, acceptance testing, performance testing, and security testing, to verify that the solution meets the functional and nonfunctional requirements and standards. DevOps also enables early detection and resolution of defects, by implementing fast and frequent feedback loops within and across the value stream. DevOps reduces the cost and risk of defects, by shifting quality left and building quality in, rather than inspecting quality out.

Less time spent fixing security issues -- DevOps enhances the security and compliance of the solution by integrating security practices into the value stream. DevOps applies the DevSecOps approach, which emphasizes the importance of proper information security practices in the pursuit of continuous delivery. DevSecOps involves applying automated tools and processes to detect and respond to security threats and vulnerabilities in the development and production environments, and ensuring compliance with security policies and standards. DevSecOps also involves collaborating with security teams and stakeholders, to foster a culture of shared responsibility and awareness for security.DevSecOps reduces the time and effort spent on fixing security issues, by shifting security left and building security in, rather than bolting security on8

Question 5

Question Type: **MultipleChoice**

What is a core DevOps principle?

Options:

A- Lean Budgets

B- Culture

C- Servant leadership

D- Iteration review

Answer:

B

Explanation:

Culture is a core DevOps principle. DevOps is not only a set of technical practices, but also a mindset and a culture that supports the integration, automation, and collaboration needed to effectively develop and operate a solution. DevOps culture is based on the following values and behaviors:

Shared responsibility -- DevOps culture fosters a sense of shared ownership and accountability for the entire solution lifecycle, from ideation to operation. Development and operations teams work together, not only to help each other, but also to ensure that the overall organization succeeds. DevOps culture eliminates the blame game and the silo mentality, and encourages mutual trust and respect among all stakeholders.

Continuous learning -- DevOps culture promotes a culture of continuous learning and improvement, where teams are constantly seeking feedback, experimenting with new ideas, and learning from failures. DevOps culture embraces a growth mindset, where teams are not afraid to try new things, challenge the status quo, and learn from their mistakes. DevOps culture also supports a learning organization, where teams share their knowledge and best practices, and leverage the collective intelligence of the whole enterprise.

Customer focus -- DevOps culture emphasizes the importance of delivering value to the customer, and aligning the solution to the customer needs and expectations. DevOps culture applies customer-centric and design thinking approaches, such as personas, empathy maps, and customer journeys, to understand and empathize with the customer problems and desires. DevOps culture also validates the assumptions and hypotheses about the customer value proposition, by collecting and analyzing data and feedback from the customer.

Automation -- DevOps culture leverages automation to improve the efficiency, quality, and reliability of the solution delivery process. DevOps culture applies automation to various aspects of the Continuous Delivery Pipeline, such as testing, integration, deployment, monitoring, and security.DevOps culture also uses automation to reduce manual work, eliminate human errors, and accelerate feedback loops5

Question 6

Question Type: **MultipleChoice**

What falls outside the scope of the Stabilize activity?

Options:

A- Continuous security monitoring is done

B- Features are monitored after release

C- Blue/green deployment

D- Failover and recovery processes are in place

Answer:

C

Explanation:

Blue/green deployment falls outside the scope of the Stabilize activity. The Stabilize activity is part of the Release on Demand aspect of the Continuous Delivery Pipeline, which is responsible for releasing new functionality to end users, either immediately or incrementally, based on business and customer needs. The Stabilize activity ensures that the solution is working well from a functional and nonfunctional perspective, and that it can be operated and supported effectively. The Stabilize activity involves the following practices:

Continuous security monitoring -- Applying automated tools and processes to detect and respond to security threats and vulnerabilities in the production environment, and ensuring compliance with security policies and standards.

Failover and recovery processes -- Establishing and testing backup and restore mechanisms, disaster recovery plans, and business continuity procedures, to ensure the availability and resilience of the solution in case of failures or disruptions.

Features monitoring -- Collecting and analyzing data on the usage, performance, and outcomes of the released features, to measure their value and impact, and to identify any issues or defects that need to be fixed or improved.

Support and maintenance -- Providing ongoing support and maintenance for the solution, such as resolving incidents, handling requests, applying patches, and performing upgrades, to ensure the reliability and quality of the solution.

Blue/green deployment, on the other hand, is a technical practice that belongs to the Continuous Deployment aspect of the Continuous Delivery Pipeline, which is responsible for deploying new functionality into the production environment, where it can be tested and validated. Blue/green deployment is a change management strategy that reduces the downtime and risk of deploying new versions of software. It involves having two identical but separate environments: one is the active environment that serves the user traffic (blue), and the other is the inactive environment that hosts the new version of the software (green). The deployment process consists of switching a small portion of the user traffic from the blue environment to the green environment, after verifying that the new version is working properly. The portion of users who receive the new version are called canaries, as they serve as early indicators of the quality and performance of the new version. If the canary release is successful, the entire user traffic is gradually switched to the green environment, which becomes the new active environment. If the canary release fails, the user traffic is switched back to the blue environment, which remains the active environment. Blue/green deployment has several benefits, such as:

It allows for fast and reliable rollback, in case of any issues or errors in the new version, by simply switching back to the active environment.

It eliminates the need for complex and error-prone migration scripts, as the inactive environment can be prepared and tested in advance, without affecting the active environment.

It enables testing and experimentation of the new version with a subset of users, by directing some user traffic to the inactive environment, before switching completely.

It facilitates continuous delivery and deployment, by automating the switching process and reducing the transaction cost and risk of moving changes to production

Question 7

Question Type: **MultipleChoice**

Which two areas should be monitored in the Release on Demand aspect to support DevOps and Continuous Delivery? (Choose two.)

Options:

A- Full-stack system behavior

B- Build status

C- Agile Release Train velocity

D- Deployment cycle time

E- Business Metrics

Answer:

B, D

Explanation:

Two areas that should be monitored in the Release on Demand aspect to support DevOps and Continuous Delivery are the build status and the deployment cycle time. The build status is the measure of whether the code and components can be successfully compiled, linked, packaged, and verified into deployable binaries. The build status indicates the quality and consistency of the code and the readiness for deployment. Monitoring the build status helps to support the Release on Demand aspect in SAFe by providing valuable information for the following purposes:

To identify and fix any errors or defects that may prevent the code from being deployed or released

To ensure that the code meets the quality standards and security checks, such as static code analysis, code coverage, and code review

To verify that the code and components are integrated and merged correctly into the trunk

To track the progress and status of the features and capabilities that are being developed and delivered

The deployment cycle time is the measure of how long it takes to deploy the code and components from the source control system to the production environment. The deployment cycle time indicates the efficiency and reliability of the deployment process and the speed of delivery. Monitoring the deployment cycle time helps to support the Release on Demand aspect in SAFe by providing valuable information for the following purposes:

To optimize the deployment process and reduce the lead time and variability

To automate the deployment process and eliminate manual steps and errors

To align the deployment process with the market demand and release strategy

To evaluate the impact and value of the deployed features and capabilities7

20%
DISCOUNT