DevOps Maturity Assessment

CAMS

Culture: creating a cross-functional team guided by empathy, transparency, respect, and alignment on producing the same outcome

1. Big, up-front plans are created with lists of expected features; a delivery date is mandated
2. A Product Owner is clearly identified and engaged
3. S
4. TEST AUTOMATING ENABLERS AND CAPABILITIES: Turning to test automation pre-requisites and enablers, how would you rate your current capabilities in the following areas?
   1. Capturing/modelling business requirements
      1. 0 = Severely lacking in general
      2. 1 = Effective but inefficient
      3. 2 = Effective and efficient
   2. Generation of test cases from business requirements
      1. 0 = Severely lacking in general
      2. 1 = Effective but inefficient
      3. 2 = Effective and efficient
   3. Ensuring maximum test coverage, with the least amount of test cases
      1. 0 = Severely lacking in general
      2. 1 = Effective but inefficient
      3. 2 = Effective and efficient
   4. Adapting test cases quickly as requirements are changed
      1. 0 = Severely lacking in general
      2. 1 = Effective but inefficient
      3. 2 = Effective and efficient
   5. Ability to do in-Sprint test automation for new features/functionality
      1. 0 = Severely lacking in general
      2. 1 = Effective but inefficient
      3. 2 = Effective and efficient
   6. Progressing code to the next stage based upon the passing of test cases
      1. 0 = Severely lacking in general
      2. 1 = Effective but inefficient
      3. 2 = Effective and efficient
   7. Obtaining access to required systems or environments for testing purposes
      1. 0 = Severely lacking in general
      2. 1 = Effective but inefficient
      3. 2 = Effective and efficient
   8. Ensuring test data is available when needed (e.g. on demand)
      1. 0 = Severely lacking in general
      2. 1 = Effective but inefficient
      3. 2 = Effective and efficient
   9. Avoiding exposure of PII (personally identifiable information) in test data
      1. 0 = Severely lacking in general
      2. 1 = Effective but inefficient
      3. 2 = Effective and efficient
5. ORGANIZATION CULTURE AND PEOPLE: How would you rate your company in relation to the following?
   1. Do you have a defined service ownership model across development and Operation community?
   2. Achieving the necessary cultural and mindset shift:
   3. Management appreciation of the need for software quality
      1. 0 = Little or no achievement
      2. 1 = Partial achievement of goal
      3. 2 = Full achievement of goal
   4. Securing management buy-in to implement modern methods
      1. 0 = Little or no achievement
      2. 1 = Partial achievement of goal
      3. 2 = Full achievement of goal
   5. Securing the budget required to fund necessary improvements
      1. 0 = Little or no achievement
      2. 1 = Partial achievement of goal
      3. 2 = Full achievement of goal
   6. IT practitioner appreciation of need for software quality
      1. 0 = Little or no achievement
      2. 1 = Partial achievement of goal
      3. 2 = Full achievement of goal
   7. Individual IT practitioner accountability for quality of output
      1. 0 = Little or no achievement
      2. 1 = Partial achievement of goal
      3. 2 = Full achievement of goal
   8. Effective team collaboration on testing activities
      1. 0 = Little or no achievement
      2. 1 = Partial achievement of goal
      3. 2 = Full achievement of goal
6. Which of the following best describes the state of your code at the end of each iteration?
   1. Ready for testing
   2. Partially tested, ready for additional integration, performance, security, and/or other testing
   3. Fully tested, documented, and ready for production delivery or GA release (modulo translation work or legal approvals)
7. How quickly can your team pivot (complete a feature in progress and start working on a newly-arrived, high-priority feature)?
   1. 3 months or longer
   2. less than 3 months
   3. less than one month
   4. less than one week
8. How quickly are you able to change a line of code and deliver to customers as part of a fully-tested, non-fix pack release?
   1. 12 months or longer
   2. less than 6 months
   3. less than 3 months
   4. less than one month
   5. less than one week
   6. less than one day
9. What is the cost (in person-hours) of executing a full functional regression test?
10. How long does it take for developers to find out that they have committed a source code change that breaks a critical function?
    1. One week or longer
    2. 1 to 7 days
    3. 12 to 24 hours
    4. 3 to 12 hours
    5. 1 to 3 hours
    6. Less than one hour
11. Which of the following best describes the state of your deployment automation for the environments used for testing?
    1. We have no deployment automation.  Setting up our test environments is entirely manual
    2. We have some deployment automation, but manual intervention is typically required (for example, to provision machines, setup dependencies, or to complete the process)
    3. We create fully-configured test environments from scratch and we reliably deploy into those environments without manual intervention
    4. We create fully-configured production-congruent environments from scratch and we reliably deploy into those environments **without manual intervention**.
12. (If your product is SaaS) Which of the following best describes the state of your deployment automation for your staging and production environments?
    1. We have no deployment automation.  Setting up our staging and production environments is entirely manual
    2. We have some deployment automation, but manual intervention is typically required (for example, to provision machines, setup dependencies, or to complete the process)
    3. We create fully configured staging and production environments from scratch and we reliably deploy into those environments **without manual intervention**.
13. (If your product is SaaS) Are you able to make business decisions based on data provided by infrastructure, application, and customer experience monitoring?
    1. Yes
    2. No
14. (If your product is SaaS) How much downtime is generally required to deploy a new version into production?
    1. 4 hours or longer
    2. 1- 4 hours
    3. Less than 1 hour
    4. No downtime is needed
15. (If your product is SaaS) How often do problems occur when deploying a new version into production?
    1. Problems always occur
    2. Problems occur about 50% of the time
    3. Problems occur about 25% of the time
    4. Problems occur about 10% of the time
    5. Problems are rare
16. D
17. D
18. D
19. S
20. S
21. s