**Large scale changes to an application can be achieved using \_\_.**

Branch by Abstraction

**Capacity testing \_.**

is a critical testing and helps to verify technical aspects

**A build can be triggered by a version control tool.**

True

**A build that fails \_\_ stage will not be deployable.**

Acceptance

**\_\_\_\_\_\_\_\_\_\_\_\_ testing was derived from hardware testing.**

Smoke

**Accurev is a \_ system.**

Stream Based Version Control

**The continuous deployment must be followed to practice continuous delivery.**

True

**Collection of executable codes is known as .**

Binaries

**Release candidate is created after the change passes through the .**

Commit Stage

**Dependencies between components must be represented using .**

Directed acyclic graph

**Build-time dependency should be present when an application is run.**

False

**Build tools can be categorised as \_\_\_\_.**

Both the options

**People who do the deployment should be part of creating the deployment process.**

True

**Time taken to fix a broken build is measured using \_\_.**

Build repair rate

**New instance of \_\_\_ is created upon every check-in.**

All of the options

**Continuous deployment aims at making feedback loops as short as possible.**

True

**\_ is a central part of continuous deployment.**

Continuous Integration

**Environment constitutes \_\_\_.**

All of the options

**Functional testing is a verification activity.**

True

**is a non-functional test.**

None of the options

**Only positive scenarios are validated in \_\_\_ testing.**

Build Verification

**Static code analysis is performed in the \_.**

Commit stage

**Steps to be followed to release the application first time is part of \_\_\_.**

Both the options

**Value delivered by an application is validated through .**

User acceptance

**Deployment pipeline is an extension of .**

Configuration Management

**What are the principle(s) that are basic for an effective configuration management?**

Both the options

**User acceptance testing is done in the stage.**

Acceptance

**Automated implementation of an application's build, test, and deployment process is called as \_.**

Deployment Pipeline

**Famous problem of dependency management is known as \_\_\_.**

Dependency Hell

**Component is separated from codebase when .**

All of the options

**An environment that is in a properly deployed state is known as .**

Baseline

**Which of the activity/activities are part of continuous integration?**

All of the Options

**Testing done in a customer's environment is know as .**

Beta

**User acceptance testing is a verification activity.**

True

**The deployment pipeline is initiated by the creation of a release candidate.**

True

**\_\_\_ are known as hot-deployments.**

Zero downtime releases

**Configuration information can be supplied using .**

Registry settings

**\_\_\_\_ stage takes the delivery team beyond continuous integration.**

Acceptance

**Disciplined technique for restructuring a code without modifying its external behavior is \_.**

Refactoring

**Chef is an orchestration tool.**

True

**Response time of the requests are verified using \_\_\_\_.**

Scalability Testing

**\_ stage asserts that the system works at the technical level.**

Manual Test

**Number of transactions a system can process in a given timespan, is measured using \_ testing.**

Throughput

**Feature toggles is possible in version control systems.**

True