<Template - Replace with Subject> - Performance Test Plan

CLONE This page and replace as necessary.

What are the objectives of the performance test phase

... identify the objectives of the tests ... template answers below

- Ensure that the defined Performance NFRs are satisfied for the application requests through the <X/Application>, <Y/Data> and <Z/Network> layers.
- Understand performance, stability and capacity of the Release <A> of the <X> application based on the observed and expected peak user volumes.
- Identify performance capabilities and problems as a result of the integration of <X>, <Y> and <Z>
- Support and guide during the investigation and debugging of performance issues.
- De-risk the occurrence of performance-related problems, when the release goes live.

What components constitute the system under test

... identify the business and technical components which constitute the system to be tested

What are the business transactions to be modelled

... identify the logical components and processes to be coded, this will allow you to define test scenarios

What are the NFRs

... To be able to calculate the pacing of the tests. The consequence of this is that if the test does not complete within the test period, it has failed the Non Functional Requirements

What are the performance acceptance criteria

... To allow you to identify if a test has passed or failed, requires further work or is good enough or good enough for now

What are the volumetrics of the system and especially the data through components under test

... identify the data and volumes thereof, the user and transaction load, operational profile including frequency and concurrency of data and processes, end-to-end, through the system

User Journey based volumetrics, depending on business definitions, one or more columns below may be redundant as they will be defined by two of the other columns.

Business Process	**Max No. Registered Users**	**Max No. Concurrent Users**	**Cases per User per time period**	**Normal Transactions per time period**	**Peak Transactions per time period**	**Max Transactions**	**SLA /response time**	**Activity Summary**	**Affected Systems / Data Flow**	**Performance Test**

What is the expected system availability: e.g. 24/7, 9-6, etc (required for modelling SOAK load, is volume spread across whole day or only in specific hours).

Transactions can be anything which provides input to the system, such as a user journey, external; system providing data, etc.

What does the capacity plan show about usage and growth over time

... At what point(s) in time do we model the tests for, including busy periods and taking into account growth of data and or usage over time

What type of tests are required

... Based on the objectives of the test, what needs to be proven, identify which one or more of the following tests are required:

Typical testing would include a subset of:

- Load – examine behaviour of the system under a specific load e.g. Normal, Peak, Outage, DoS or DDoS attacks
- Stress – find the upper limits of the system
- Soak – aka endurance testing, see if system can sustain a continuous load for a period of time
- Spike – suddenly increase load, see what happens
- Configuration – tuning a system through configuration changes
- Isolation – identify an issue and find the root cause, while bedding in a new system or release this is inadvertently and unavoidably the most common

What are the priorities of each test

... To allow you to prioritise and plan the tests to ensure the correct tests are completed with the resources and time available

What are the test entry criteria

... What needs to be in place and working before testing can commence, allows you to identify if we are actually ready to test, e.g. necessary business and technical documentation validated, pipe clean/smoke tests correctly identify state of system, identify pretest checklist, supporting resources/personnel, results collation procedure

What are the test exit criteria

... What needs to be completed before testing can complete, do we provide simply a report or a tuned system, which tests should/must be run - coverage, are defects completed, what level of confidence is required/risk is acceptable

What technology is required to test each component

... How can each component be tested, is there are specific requirement or restriction for a particular set of testing tools

The standard IPT Performance testing tool set will be used, this includes:

- Git for source code control
- Jenkins for job control
- Maestro for test control
- JMeter for test execution

What does the Business environment look like

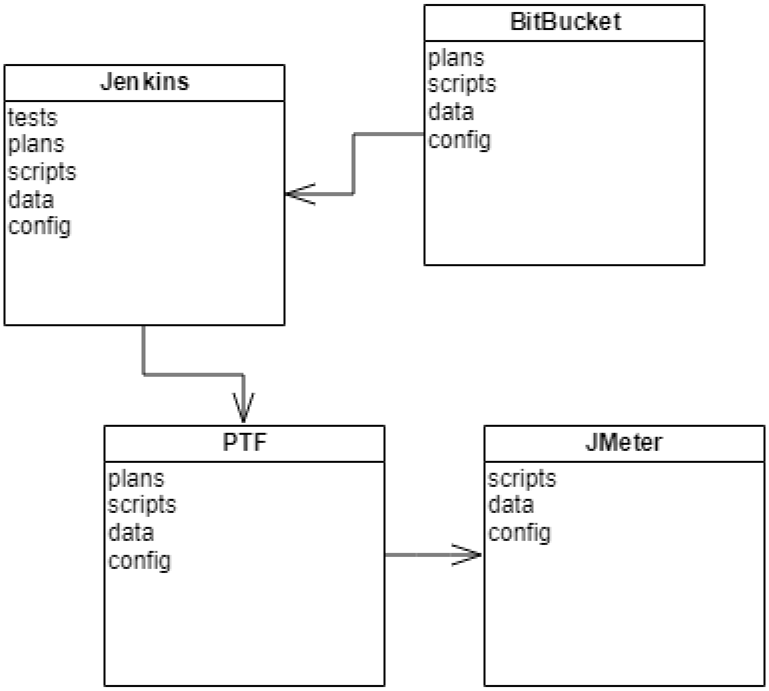
... Understand components required to make the system work

- Availability / Operational Hours
- User Base
- Licencing restrictions (max users on a component)

What does the Test environment look like

... Understand components available during testing

The test infrastructure is pictured in the diagram below.



What does the Live environment look like

... Understand components which implement the Live system

- Operational Bandwidth (component restrictions)
- Network Bandwidth

- Technical restrictions

What are the differences between Test and Live

... Identify risks, restrictions and limitations (including components, configuration and sizing) due to differences between Test and Live, things which may not be tested or require scaling of some sort

Is the test system shared with other teams

... Will there be any restrictions on the use of the environment, especially regards to:

- *modifications of configuration or data*
- *access to files and servers*
- *dates and times when one can test*

What tools are there for maintaining the system in a known state

... To allow you to start each test from a well know position so as to make tests repeatable and directly comparable, scripts or processes to allow the system state to be set to a known point, such as delete scripts, file/database backup and recovery, database Flashback (Oracle) etc

What is the key data within the system

... To allow you to identify important data entities, which may need to conform to certain standards, formats, ranges or values and may also be primary database keys which much maintain referential integrity across tables or even databases. This will allow you to identity what test data is required and how it needs to be generated

What metrics should be collected during the test

... To allow you to identify if performance acceptance criteria have passed, may involve measurements on transactions, cpu usage, memory usage, network io, disk io, db transaction times etc. Collecting the correct metrics will allow identification of issues and also provide supporting evidence of the test results

What are the deliverables of the performance tests

- The Performance Test regression pack.
- The Performance Test Report.
- Updated <replace with link to appropriate Test Status Report>

- -

End of Template

Notes - Quick check List

- Get confirmation on the scope of the performance tests .
    - Transaction set including type, protocol (REST/SOAP) responses,
    - Define number of concurrent operations of each type.
    - Define user blend (relative percentage of concurrent operations performed by each user)
    - Get the type of performance testing (Spike, Stress, Soak, Normal, Peak).
    - Get an understanding of the unique identifiers (correlation ids), linked business keys, unique business keys.
    - Include that the scope will only cover happy path testing.

    - Confirm existence of NFRs.
    - If no to above then define the transaction volumes.
    - Get the scope defined on a confluence page.(IIAS PERF Tset approach)
- Check the following on System Test:
    - If the services are going to be HTTP or HTTPS?
    - Check if there are going to be any additional Security setup (any encryption on payload) on the environment.

- Check if the Simulator can be used for performance testing.
- Check the target environment is live-like (especially with regards to HA and networking)

ⓘ See https://www.gov.uk/service-manual/operations/load-and-performance-testing.html

https://en.wikipedia.org/wiki/Load_testing

https://en.wikipedia.org/wiki/Software_performance_testing

https://msdn.microsoft.com/en-gb/library/bb924356.aspx

The 7P's: Proper Planning and Preparation Prevents Pi$$ Poor Performance

Related articles

- EPIC Creation - Performance Test Plan
- Performance Testing Knowledge Base
- SSB Troubleshooting
- AppDynamics
- Analysing Tests