Non-Functional Risk Assessment Framework to increase predictability of Non-Functional Defects

Vijayanand Chelliahdhas
HCL America, Inc.
Vijayanand.c@hcl.com

Abstract

Typically organizations tend to adapt a one-size-fits-all approach to validate a system for its non-functional (NF) requirements. Not surprisingly, a significant number of non-functional issues surface post go-live. Such unpredictability in the non-functional quality of a software implementation can be largely attributed to the lack of a focused approach to non-functional testing targeting the critical technical and non-functional risks in the system under test (SUT).

This challenge can be mitigated by conducting a Non-Functional Risk Assessment exercise to develop a comprehensive Risk Catalogue. As a result, the NF test strategy and scenarios would be directly mapped to all the technical and NF risks in the system.

The “NF Risk Assessment Framework” described in this paper enables to precisely determine the key vulnerable areas in the system and not just the application, addresses coverage across multiple non-functional domains of the SUT and not just performance, ensures the tests are designed to simulate production condition as close as possible and helps to prioritize test activities based on the criticality assigned.

Biography

Vijayanand (Vijay) Chelliahdhas is a Solutions Architect at HCL America, Inc. currently responsible for providing solutions around non-functional testing including Performance Testing and Engineering, Security Testing, Mobility Performance, Testing on the Cloud and Service Virtualization across the US geography.

For more than a decade, Vijay has been involved in evaluating performance for a variety of systems including OLTP, batch programs, middleware and package implementations across multiple technologies. Vijay has extensive experience in devising performance engineering strategies for multi-year multi-release IT programs, setting up performance test centers of excellence and providing consulting services to optimize the ways of working in the performance testing and engineering lifecycle. Additionally, Vijay also specializes in conducting maturity assessment for performance testing organizations and risk assessment for multi component distributed systems.

1 Introduction

The standard approach to non-functional testing, which in most cases is limited to performance testing only, is to identify a subset of use cases, which are anticipated to be executed frequently in production, and conduct a set of non-functional tests. Such tests could include Load Tests, Stress Tests, Endurance Tests et al using any renowned commercial off the shelf (COTS) tool and then publish the test results.

Where the test results indicate that certain transactions do not meet the stipulated Service Level Agreements (SLAs) (if they were defined in the first place) then either an exception is sought to go-live as per schedule or execute a re-test post tuning. Such tuning attempts are predominantly limited to modifying certain configuration values either at the application server or database server layer or both to alleviate the symptoms of performance issues.

This superficial approach to non-functional testing tends to leave gaping holes in the system and exposes it to serious performance and non-functional quality issues when the “rubber hits the tarmac” in production.

Given that there could be various reasons why the non-functional issues were not detected in a pre-production environment, one of the most critical elements is the ability to simulate the anticipated risks in a production like pre-production environment. To accomplish the same, the system under test should be systematically studied and all the potential risks thoroughly understood in the context of historical production non-functional issues and futuristic workload and deployment changes.

This white paper explains how to undertake a Non-Functional Risk Assessment exercise to ensure the non-functional test strategy and scenarios have a fool-proof coverage of all the technical and non-functional risks to maximize predictability of non-functional quality attributes of the system.

2 Objective and Outcome of Non-Functional Risk Assessment

The driver for conducting a Non-Functional Risk Assessment is to technically assess the System Under Test (SUT) for potential non-functional risks and thereby develop a Non-Functional Risk Catalogue, which would be the basis for developing the non-functional test strategy and the test scenarios.

The Non-Functional Risk Catalogue is a collection of detailed risks and their impact to the non-functional quality attributes of the SUT. This catalogue is derived from a Risk Matrix which is essentially an intersection of the various potential “Threats” and “Focus Areas” for the SUT.

The following sections detail the method to study a software system, assess it from the various non-functional attributes perspective and arrive at a detailed risk catalogue.

3 Non-Functional Risk Assessment Framework

The proven framework for non-functional risk assessment comprises of the following 3 steps:

- System Appreciation and Technical Assessment
- Develop Risk Matrix
- Create Risk Catalogue

This framework was implemented to assess the non-functional risks of a Trading Product developed by a major UK based banking software provider. The following sections of the document will leverage the above said actual implementation experience for illustration purposes.

3.1 System Appreciation and Technical Assessment

This is the fundamental step in the framework, the output of which is critical in building the Risk Matrix subsequently. This is further broken down into 3 steps as follows:

- Study application architecture and design
- Historical non-functional incidents analysis
- Understand future deployment and workload characteristics

The architecture of the SUT is thoroughly studied from the logical and physical point of views, the technology landscape, including studying the various components of the architecture, the core business engine, how the complex algorithms were implemented, the synchronous and asynchronous communications, CPU or Memory or I/O intensive operations, what were the integrated components and how the data flows in integrated scenarios.

If the SUT has already been in production, then the information from the historical incidents reported in production is invaluable information to understand the maturity and weak areas of the system. In our example, at least 12 months of data from the Incident Management System was extracted for a thorough entry by entry analysis. Besides documenting various record keeping fields for each Incident shortlisted, the following data were also gathered, analyzed and documented. A sample entry is shown below:

Incident Description	Trunk End of Day (EoD) has a reproducible error where extract reports are requested in conjunction with re-use reports. The remaining steps are still unexecuted but the EoD job has completed.
Technical Analysis	When a job/step is bypassed, the next job/step should take over and execute. And EoD status should reflect this appropriately. However in actuality, due to bypass of a step, the EoD loses status integrity (suspected to be at least 2 causes - weblogic to database (DB) connection starvation, DB lock contention)
Non-Functional? (Y/N)	Y
NF Domain	Reliability
Class of Issue	Connection Pool Starvation, Database lock Contention
Sub-Class	Bypass a step in EoD (particularly a step in Extract Report)
Potential way of Detecting the issue	Schedule an EoD run with Extract Reports in conjunction with Reuse Reports. Bypass a step manually in the middle of the run (particularly a Step in Extract Report) Track the status of the next job, current job and overall EoD
Applicable NF Test	EoD Batch Test (peak daily volume)

Out of a total of 3000 incidents, about 212 non-functional incidents were identified for in-depth analysis. Each of those shortlisted entries were studied and documented in the above format.

After the historical incidents analysis, several discussions were conducted with two program teams who were involved in the two biggest implementations of the software product spanning over 36 months. The objective was to understand the technology stack, the infrastructure capacity details such as the number of nodes, VMs, data centers, latency between the architecture components, the latency between the end users and the datacenters, growth in business volume and therefore the projected workload, the growth in the number of end users, the rate of growth of the database size, the plan and algorithm for data archival and purging.

This comprised all the activities involved in System Appreciation and Technical Assessment which helped gather all the requisite data to arrive at the Risk Matrix.

3.2 Develop Risk Matrix

The Risk Matrix is essentially precipitated from the information analyzed in the above step. The objective of the risk matrix is to map the potential “Non-Functional Threats” to the “Focus Areas” of the SUT and to highlight the degree of impact of each Threat – Focus Area mapping.

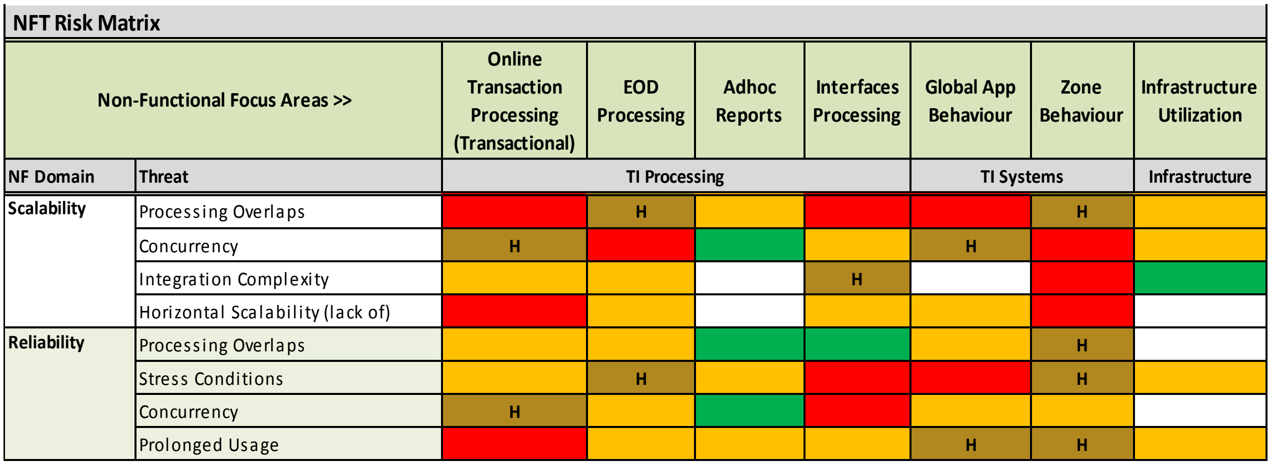
A “Threat” is a technical attribute or event that can impact the non-functional quality of the SUT. Examples of threats classified in the illustration include: processing overlaps, concurrency, JVM sharing, multi-geo access, VM crash, incorrect error handling etc.

A “Focus Area” is a component or set of functionalities in the SUT that is critical to the non-functional quality of the SUT. Examples of focus areas classified in the illustration include: OLTP scenarios, EoD processing, interfaces processing, global app, infrastructure utilization etc.

In our particular illustration, 14 non-functional domains were initially identified which was later crystallized to 7 at the end of the System Appreciation and Technical Assessment phase. The 7 Non-Functional domains included Scalability, Reliability, Performance, Resilience & Recoverability, Capacity, Interoperability and Compatibility.

There were a total of 31 Threats identified across the 7 Non-Functional domains and these were mapped to 7 Focus Areas. This resulted in a mapping of 217 non-functional risks in total which was then classified to 5 different types as indicated below.

Following is a snippet of the Risk Matrix, displaying a sub-set of Threats mapped to the Focus Areas.



NFT Risk Matrix

Non-Functional Focus Areas >>		Online Transaction Processing (Transactional)	EOD Processing	Adhoc Reports	Interfaces Processing	Global App Behaviour	Zone Behaviour	Infrastructure Utilization
NF Domain	Threat	TI Processing				TI Systems		Infrastructure
Scalability	Processing Overlaps	🟥	🟫 H	🟧	🟥	🟥	🟫 H	🟧
	Concurrency	🟫 H	🟥	🟩	🟧	🟫 H	🟥	🟧
	Integration Complexity	🟧	🟧	⬜	🟫 H	⬜	🟥	🟩
	Horizontal Scalability (lack of)	🟥	🟧	⬜	🟧	🟧	🟥	⬜
Reliability	Processing Overlaps	🟧	🟧	🟩	🟩	🟧	🟫 H	⬜
	Stress Conditions	🟧	🟫 H	🟧	🟥	🟥	🟫 H	🟧
	Concurrency	🟫 H	🟧	🟩	🟥	🟧	🟧	⬜
	Prolonged Usage	🟥	🟧	🟧	🟧	🟫 H	🟫 H	🟧

Figure: Non-Functional Risk Matrix

Rating	Description
H	High Impact + High Probability of Occurrence
🟥	High Impact
🟨	Medium Impact
🟩	Low Impact
⬜	Negligible Impact

Figure: Legend for the Non-Functional Risk Matrix

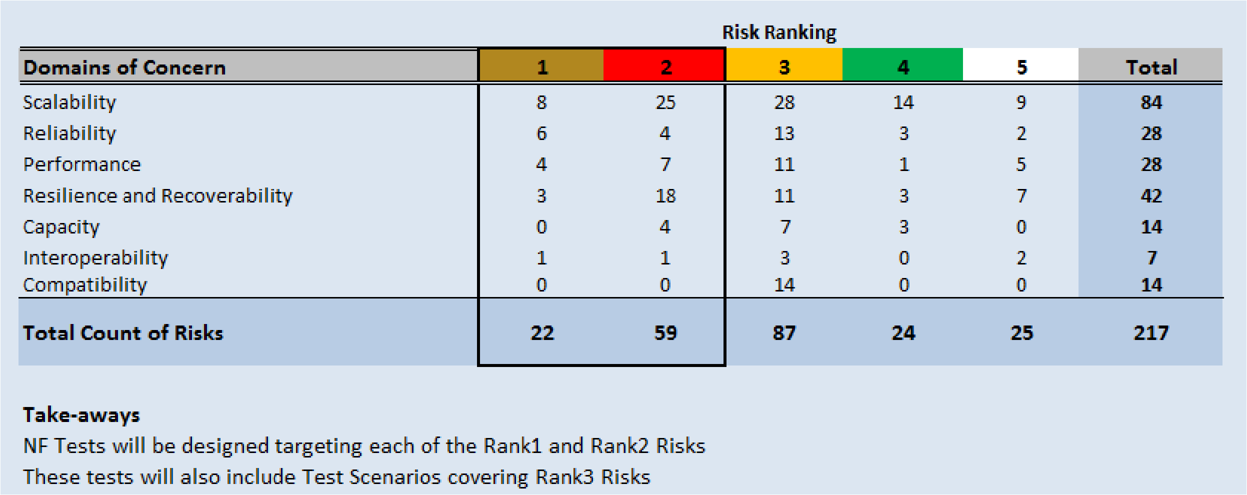
3.3 Create Risk Catalogue

Once the Risk Matrix is developed, the next step of creating the Risk Catalogue is essentially a task of detailing the risks in accordance with each Threat – Focus Area mapping. In our particular illustration, out of the total 217 risks, about 81 risks which fell under the first 2 categories (brown & red) were documented in detail in the form of a Risk Catalogue.

For each Risk Catalogue entry, besides the risk and impact, the sub-threats and parameters to measure were also gathered and documented which could be directly converted to Test Scenarios. Following are 2 sample entries from the Risk Catalogue.

#	1	2
NF Domain	Scalability	Performance
Threat	Processing Overlap	Multi Geo Access
Sub-Threats	Intra Zone Processing	-
Risk	Two or more Multi Bank Entities (MBE) within a Zone could be performing different operations at the same time, leveraging the same application/OS/database resources and processing the same data set or accessing from the same data source (table/schema/database)	User sites are spread across the globe, however all user access have to pass through the Global Single Sign On (SSO). There will be only one primary instance of Global App in one location and all users will be routed through this single Global app.
Impact	There will be intermittent delays in online transaction processing (OLTP) or delays in Message transmission into the Transport Client	User accesses from multiple geographies to the global App and the response therefore will potentially be slow, influenced by the bandwidth congestion over the wide area network (WAN) between the user sites and the global app
Parameters to Measure	OLTP Response Time	Global Dashboard Response Time
Focus Area	Zone Behavior	Global App (SSO, Dashboard)

Eventually the Non-Functional Risks were grouped by the NF Domain and summarized as part of this exercise conducted for the Banking Product.



	Risk Ranking					
Domains of Concern	**1**	**2**	**3**	**4**	**5**	**Total**
Scalability	8	25	28	14	9	**84**
Reliability	6	4	13	3	2	**28**
Performance	4	7	11	1	5	**28**
Resilience and Recoverability	3	18	11	3	7	**42**
Capacity	0	4	7	3	0	**14**
Interoperability	1	1	3	0	2	**7**
Compatibility	0	0	14	0	0	**14**
Total Count of Risks	**22**	**59**	**87**	**24**	**25**	**217**

Take-aways

NF Tests will be designed targeting each of the Rank1 and Rank2 Risks

These tests will also include Test Scenarios covering Rank3 Risks

Figure: Non-Functional and Technical Risks Summary

The Non-Functional Risks in the catalogue were then mapped to the identified NF tests to ensure traceability and coverage of the potential risks.

3.4 Advantages & Benefits

The following benefits could be potentially reaped by conducting a comprehensive non-functional risk assessment exercise:

- Gain a precise understanding of the vulnerable areas and use cases (risks) of the SUT
- Develop an exhaustive repository of non-functional test cases/scenarios
- Ability to design tests focused on simulating the specific technical and non-functional risks
- Ensure maximum possible coverage and traceability of the NF risks in the SUT
- Predictability into all probable outcomes in production in the event of a technical failure or an unexpected workload situation or projected business growth

4 Conclusion

The Risk Assessment exercise illustrated here was conducted by 2 non-functional consultants over a 6 week period at the client site. Given the effort it takes to conduct such a comprehensive exercise, it may not be a feasible solution for short term projects or low complex applications or highly matured systems. However, for software product vendors and applications with large scale usage with unanticipated workload patterns and business growth scenarios, the qualitative and the quantitative benefits of conducting a Non-Functional Risk Assessment exercise clearly outweighs this modest effort and investment.

This was a consulting exercise where the primary objective was to conduct the Risk Assessment and develop the Risk Catalogue. Subsequent to which, the existing delivery team on the customer side had the responsibility to conduct the test execution. We did not have access to any additional data points or supporting facts post production to substantiate any further benefits due to this exercise.

5 Glossary

#	Term	Description
1	NFT	Non-functional Testing
2	Load Test	A test conducted by simulating a concurrent workload on the SUT to determine whether it can process the expected workload within the stipulated processing time
3	Stress Test	A test conducted by simulating an incremental concurrent workload on the SUT to determine whether the system performance begins to degrade in proportion to the increase in workload or the maximum resource utilization is achieved
4	Endurance Test	A test conducted by simulating a prolonged concurrent workload on the SUT to determine whether system performance sustains or degrades on continuous usage over a period of time
5	COTS	Commercial Off The Shelf
6	SUT	System Under Test
7	SLA	Service Level Agreement
8	EoD	End of Day
9	JVM	Java Virtual Machine
10	VM	Virtual Machine
11	OLTP	Online Transaction Processing
12	SSO	Single Sign On
13	MBE	Multi Bank Entity

15	WAN	Wide Area Network
16	Scalability	Indicates the ability of the SUT to sustain system behavior when the scale of operations (number of instances, amount of workload, integration complexity, capacity and geographical spread) increases exponentially/linearly in a larger scale deployment
17	Performance	Indicates the ability of the SUT to meet stipulated performance requirements (at the minimum, Throughput and Response Time) when the system is subjected to production like workload
18	Reliability	Indicates the ability of the SUT to function as intended for a prolonged duration of time under average, stress and unanticipated business workload conditions with minimal failures
19	Resilience & Recoverability	Indicates the ability of the SUT to handle faults gracefully and systematically. In the event of a hardware or software failure, the system should have the ability to repair and resume business as usual with minimal end user impact
20	Capacity	Indicates the amount of hardware capacity (in terms of CPU, Memory, Heap, Disk et al) required to run the system for business as usual and for futuristic workload requirements
21	Compatibility	Indicates the ability of the SUT to operate as intended on various flavors of the computing environment, comprising of multiple technology platforms and databases
22	Interoperability	Indicates the ability of the system to interact and exchange information seamlessly with other systems across different protocols and transmission formats in heavy workload, high stress conditions