Requirement Management in Testing

Anindita Sarkar
Project Manager
Infosys Technologies Limited, Bangalore

Abstract:

This paper discusses the issues we have faced in our testing projects in managing requirements, test cases, test conditions and how we have resolved them. It also discusses the status reporting for a testing project from the requirements perspective. For the sake of completeness this paper also discusses some aspects of the requirements change management process.

Introduction

Every software project arises out of a business problem. Requirements gathering and analysis try to identify the business problem to be solved and probable characteristic a software product needs to have as a solution to the business problem.

Requirements are the foundation stone on which a software product is built. Gathering and managing requirements is one of the biggest challenges a project manager faces in a project. Robust requirement management process is one of the stepping-stones of success of a project.

This paper discusses the issues we have faced in gathering; analyzing and testing these requirements and the workarounds we have taken to overcome those issues.

Gathering Requirements

Requirement gathering process can differ from project to project based on their nature, platform and technology. Testing teams will need to take care of the following aspects while gathering requirements for any kind of testing

Testability

One of the major challenges we have faced during requirements gathering is the testability of a requirement.

Very often customers come up with requirements that are not testable. To determine the testability of a requirement following questions can be asked

- Can we define the acceptance criteria for this requirement?

  If the answer is no then this requirement is not testable.
- Clearly state the assumption you have made on this requirement. Check if the assumption is conflicting with any other assumption/requirement made so far.

  If yes then the set of requirements are not testable
- Is this requirement clashing with any other requirement?

  If yes then the set of requirements are not testable
- Can it be broken into multiple requirements?

  If yes then the set of requirements are not testable. You will need to revisit the requirement again.

Completeness of the requirement

Very often it is difficult to find out whether a requirement is complete or not. Incomplete requirements may lead to rework during later stages of the project. To determine the completeness of a requirement following questions can be asked

- Who are the actors for the requirement?

  Requirements of actors of all categories need to be considered to make a requirement complete.
- Check about the alternate flows for a requirement.

  Most of the requirements are not complete because the alternate flows are not captured completely.

- Try to understand the Rationale behind a requirement.

  This will help in identifying gaps in the requirement.

Dependencies of Requirements

Clearly identify the dependency of your requirements on

- Any other requirement
- On systems which are outside the scope of the project. This is prevalent in batch testing environment where inputs comes as flat files from other system. Interface requirements need to be clearly documented and signed off by all the stakeholders.

Few Tips

- Identify the pre and post conditions for the requirement.

  This will help in test data setup and documenting the test results.

- Generate a Business workflow of the requirement.

  This will help the tester in understanding the system.

- Any information about existing functionality, which is not changing, should be clearly distinguished from the changing requirements. It is a good idea to put these in a new section called Regression test scenario section.
- Number all your requirements uniquely.

  This will help considerably in establishing tracebility.

- Get an independent review done to confirm your understanding of requirements.

Analyzing Requirements

Once the requirement has been gathered, one needs to analyse the requirement before arriving at testcases.

- Divide the requirements into various categories
  - Functional Requirement
  - Regression requirement.
  - Performance Requirement
  - Security Requirement
  - Technical Requirement

  Functional Requirement: These are the business needs of the proposed system.

  Regression Requirement: These are those functionalities of the system, which are not changing as a part of this project. We are recommending this distinction as we have burnt our fingers. In one project these requirements were not distinguished from functional requirements and we had spent considerable amount of time in analyzing these requirements, identifying testcases for testing this requirement, which was not necessary.

  Performance Requirement: These are related to the performance of the system.

  Security Requirement: These are related to the security of the system.

  Technical Requirement: These are those requirements, which are not related to the business functionality of the system, but needs to be tested for successful implementation of the application. This may include installation package testing interface testing.

- **Prioritize requirements** based on their business impact.

  Prioritizing requests will help the testing manager in coming up with a testing schedule, which is more likely to be in sync with the development team's expectation. All prioritization should be done based on the business criticality and availability of sources for testing. This will also help in testing critical functionality first, thereby giving more time for bug fixes.

- **Identify the risk** associated with the requirements.

  Risk identification will help in identifying, which requirements will require more intensive testing compared to others. Incase of time constraints (which is always true for the testing team) this risk matrix and along with priority matrix goes a long way in deciding which requirements can be tested superficially or left out altogether (!!!).

  Many a time end users are not aware of the differentiating factors for the risk. In such cases it is better to make a questionnaire consisting of what-if questions and ask them to rank the answers in a scale of 1 to 5. Cumulative scores of the answers will give a fair insight into the risks associated with a requirement.

Requirements Traceability

Requirement traceability is a process of establishing the linkage between the requirements and the testcases. This helps the project in many ways.

- It indicates the extent of testing a requirement has undergone.
- It also gives a clear indication of how many requirements have gone live without any testing.
- It also helps the testing team in identifying the impact of a requirement change. For example if a requirement is getting tested using 10 testcases, a change request, will mean revisiting and retesting 10 testcases.

Defining a traceability matrix is a simple process however establishing requirement tracebility is not a simple exercise. From my experience most of the testers consider entering this additional information as an overhead.

- Your team will first need to understand the benefits of maintaining the requirement traceability
- This should be an integral part of your process as test planning, test data generation is.
- Keep it simple.
- Try to automate the redundancy. For example we create a requirement to test scenario mapping. Test scenario is a testable requirement. There may be a many to many mapping between the test scenario and requirements. This is done during requirements gathering stage. During test case creation we create a test scenario to test case mapping. We then automate the process of mapping between requirement and testcases using transitive relationship.

Appendix A provides a sample tracebility matrix.

Change Management

Managing requirement change is another major challenge for the testing manager. Change requests can come at various stages of testing life cycle. In my experience it is easier to absorb these requests during the test-planning phase than during test execution phase. It is absolutely necessary to set the expectation of other stakeholders right at the beginning of the project. We set two sets of requirement change threshold based on the LC stage of the project when it is requested.

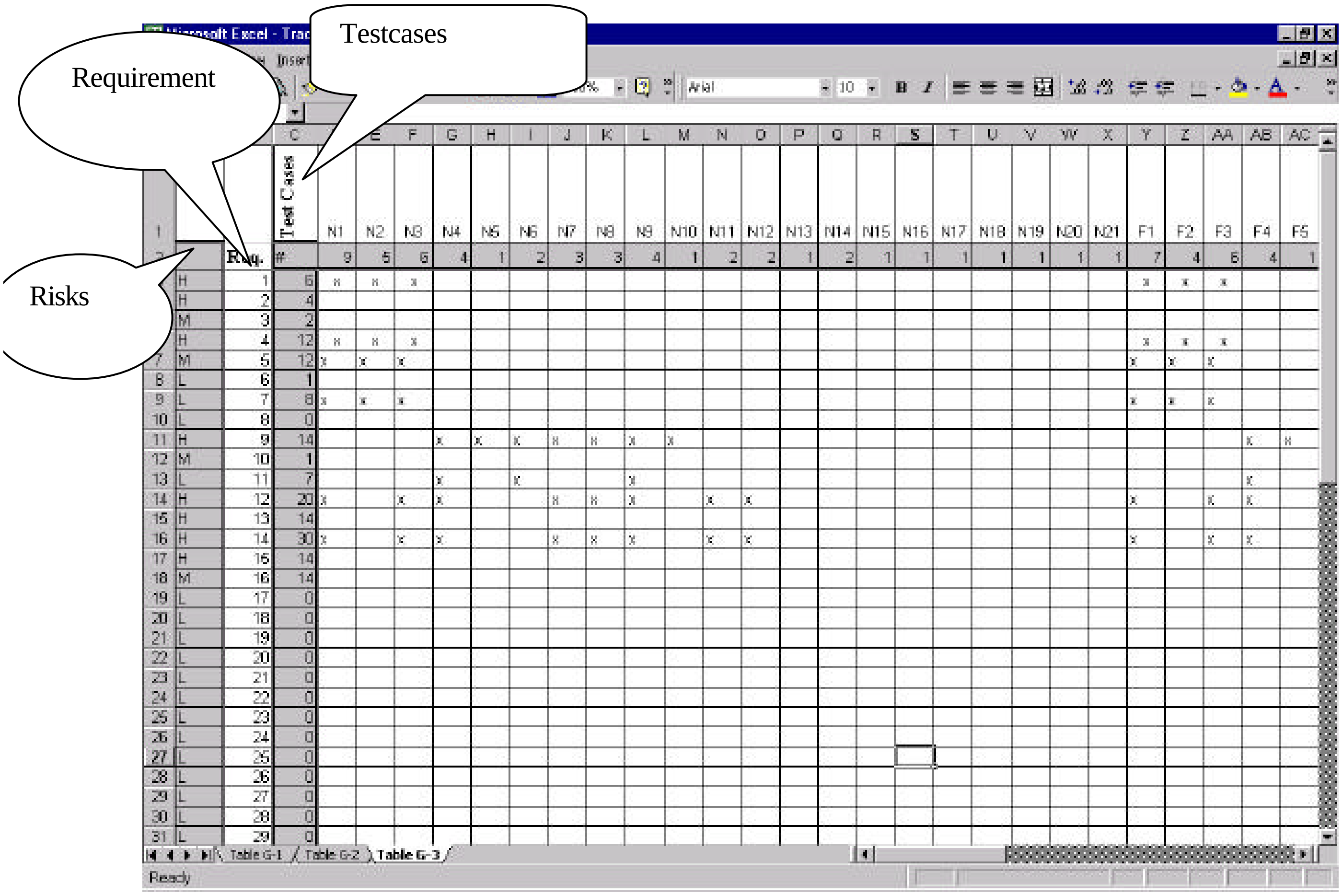
Threshold during test execution is much smaller compared to threshold during test planning stage. For example a requirement change requiring 2 person days of effort reported during test planning phase may get absorbed in the schedule without impacting the deadline. The same change request when generated during change execution may lead to a change in schedule.

Project Status Reporting

Project progress reporting becomes easier if requirements are managed well. Few reports, which can be produced easily, are

- Requirements status: How many requirements have been analyzed, converted to testcases
- Risk based progress of testing. How many testcases of each category has been tested vis-à-vis total number of testcases
- Priority based progress of testing. How many testcases of each category has been tested vis-à-vis total number of testcases
- Risk based bug detection. How many bugs were detected under each testing category.

Appendix A



			C		E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC
1			Test Cases	N1	N2	N3	N4	N5	N6	N7	N8	N9	N10	N11	N12	N13	N14	N15	N16	N17	N18	N19	N20	N21	F1	F2	F3	F4	F5
		Req.	#	9	5	6	4	1	2	3	3	4	1	2	2	1	2	1	1	1	1	1	1	1	7	4	6	4	1
	H	1	6	x	x	x																			x	x	x		
	H	2	4																										
	M	3	2																										
	H	4	12	x	x	x																			x	x	x		
7	M	5	12	x	x	x																			x	x	x		
8	L	6	1																										
9	L	7	8	x	x	x																			x	x	x		
10	L	8	0																										
11	H	9	14				x	x	x	x	x	x	x															x	x
12	M	10	1																										
13	L	11	7				x		x			x																x	
14	H	12	20	x		x	x			x	x	x		x	x										x		x	x	
15	H	13	14																										
16	H	14	30	x		x	x			x	x	x		x	x										x		x	x	
17	H	15	14																										
18	M	16	14																										
19	L	17	0																										
20	L	18	0																										
21	L	19	0																										
22	L	20	0																										
23	L	21	0																										
24	L	22	0																										
25	L	23	0																										
26	L	24	0																										
27	L	25	0																										
28	L	26	0																										
29	L	27	0																										
30	L	28	0																										
31	L	29	0																										