GURU99

Software Requirements Analysis with Example

By : Matthew Martin Updated February 24, 2024

Software requirement is a functional or non-functional need to be implemented in the system. Functional means providing particular service to the user.

For example, in context to banking application the functional requirement will be when customer selects “View Balance” they must be able to look at their latest account balance.

Software requirement can also be a non-functional, it can be a performance requirement. For example, a non-functional requirement is where every page of the system should be visible to the users within 5 seconds.

So, basically **software requirement is a**

- Functional or
- Non-functional

need that has to be implemented into the system. **Software requirement are usually expressed as a statements.**

1 JIRA Software

Learn More

On Jira Software Website

Time Tracking

Drag & Drop	Yes
Free Trial	Forever Free Plan

2 **ClickUp**

Learn More

On ClickUp Website

Time Tracking	Yes
Drag & Drop	Yes
Free Trial	Forever Free Plan

Table of Content:

Types of Requirements

1. **Business requirements**: They are high-level requirements that are taken from the business case from the projects.For example, a mobile banking service system provides banking services to Southeast Asia. The business requirement that is decided for India is account summary and fund transfer while for China account summary and bill payment is decided as a business requirement

Country	Company providing Banking Functionalities or services
India	Account Summary and Fund Transfer
China	Account Summary and Bill Payment

2. **Architectural and Design requirements**: These requirements are more detailed than business requirements. It determines the overall design required to implement the business requirement.For our educational organization the architectural and design use cases would be login, course detail, etc. The requirement would be as shown below.

Banking use case	Requirement
Bill Payment	This use case describes how a customer can login into net banking and use the Bill Payment Facility.

Banking use case	Requirement
	The customer will can see a dashboard of outstanding bills of registered billers. He can add, modify, and delete a biller detail. The customer can configure SMS, email alerts for different billing actions. He can see history of past paid bills. The actors starting this use case are bank customers or support personnel.

3. **System and Integration requirements**: At the lowest level, we have system and integration requirements. It is detailed description of each and every requirement. It can be in form of user stories which is really describing everyday business language. The requirements are in abundant details so that developers can begin coding.Here in example of Bill Payment module where requirement will be mentioned for adding a Biller

Bill Payment	Requirements
Add Billers	• Utility Provider Name • Relationship Customer Number • Auto Payments – Yes/No • Pay Entire Bill – Yes/No • Auto Payment Limit – Do not pay if Bill is over specified amount

Sometimes for some project you might not receive any requirements or documents to work with. But still there are other sources of requirements that you can consider for the requirement or information, so that you can base your software or test design on these requirements. So the other sources for requirement you can rely on are

Other Sources of Requirements

- Knowledge transfer from colleagues or employees already working on that project
- Talk about project to business analyst, product manager, project lead and developers
- Analyze previous system version that is already implemented into the system
- Analyze the older requirement document of the project
- Look into the past Bug reports, some of the bug reports are turned into enhancement request which may be implemented into current version
- Look into installation guide if it is available to see what are the installation required
- Analyze the domain or industry knowledge that team is trying to implement

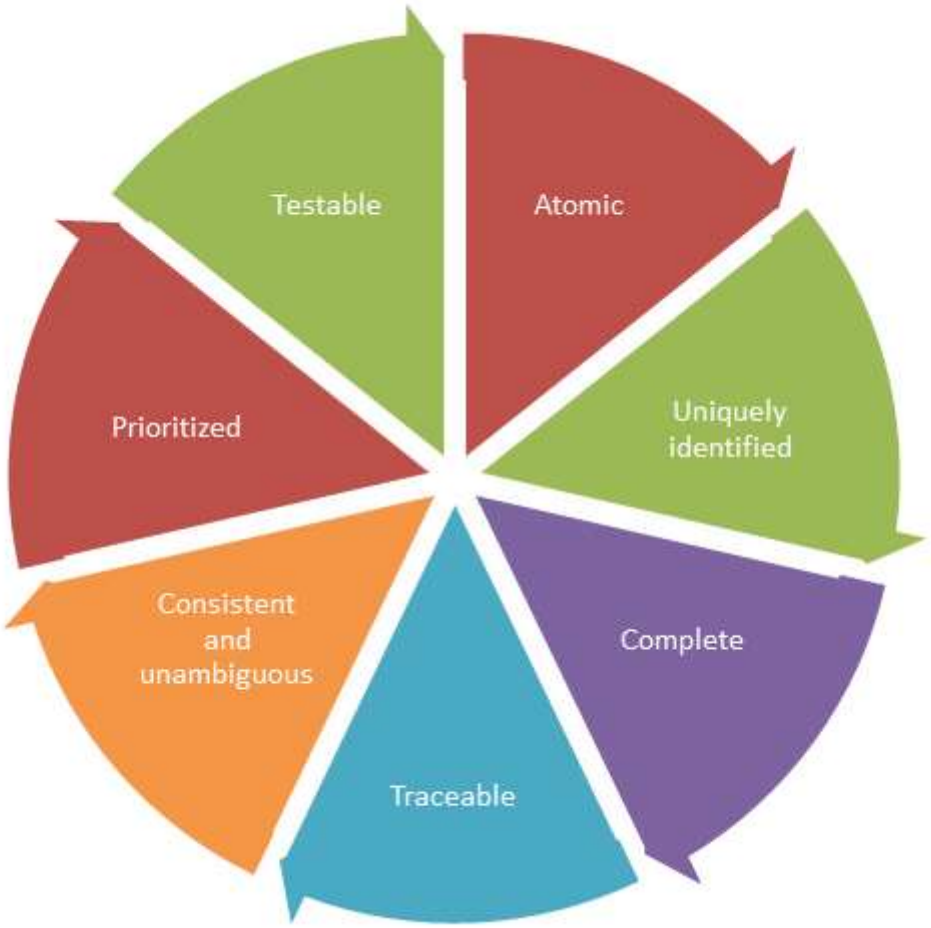
Whatever source of requirement you get make sure to document them in some form, get them reviewed from other experienced and knowledgeable team members.

How to Analyze Requirements

Consider example of an educational software system where a student can register for different courses.

Lets study how to analyze the requirements. The requirements must maintain a standard quality of its requirement, different types of requirement quality includes

- Atomic
- Uniquely identified
- Complete
- Consistent and unambiguous
- Traceable
- Prioritized
- Testable



Let understand this with an example, there are three columns in the table shown here,

1. The first column indicates- “requirement quality”
2. The second column indicates- “bad requirement with some problem”
3. The third column is same as second column but – “converted into a good requirement”.

Requirement Quality	Example of bad requirement	Example of good requirement
Atomic	• Students will be able to enroll to undergraduate and post graduate courses	• Students will be able to enroll to undergraduate courses • Students will be able to enroll to post-graduate courses
Uniquely identified	1- Students will be able to enroll to undergraduate courses1- Students will be able to enroll to post-graduate courses	1. Course Enrolment 2. Students will be able to enroll to undergraduate courses 3. Students will be able to enroll to post-graduate courses
Complete	A professor user will log into the system by providing his username, password, and other relevant information	A professor user will log into the system by providing his username, password and department code
Consistent and unambiguous	A student will have either undergraduate courses or post-graduate courses but not both. Some courses will be open to both under-graduate and post-graduate	A student will have either under-graduate or post graduates but not both
Traceable	Maintain student information-mapped to BRD req.ID?	Maintain student information-Mapped to ~~BRD~~ req ID 4.1

Requirement Quality	Example of bad requirement	Example of good requirement
Prioritized	Registered student-Priority 1Maintain User Information-Priority 1Enroll courses-Priority 1View Report Card-Priority 1	Register Student-Priority 1Maintain User Information-Priority 2Enroll courses-Priority 1View Report Card-Priority3
Testable	Each page of the system will load in an acceptable time-frame	Register student and enrol courses pages of the system will load within 5 seconds

Now let's understand each of these requirement in details starting with Atomic.

Atomic

Atomic	• Students will be able to enroll to undergraduate and post graduate courses	• Students will be able to enroll to undergraduate courses • Students will be able to enroll to post-graduate courses

So each and every requirement you have should be atomic, which means it should be at very low level of details it should not be possible to separated out into components. Here we will see the two examples for requirements, at Atomic and uniquely identified requirements levels.

So let us continue with example of system build for education domain. Here, the bad requirement is “Students will be able to enroll to undergraduate and post graduate courses” . This is a bad requirement because it is not atomic because it talks about two different entities undergraduates and post-graduates courses. So obviously it is not a good requirement but bad requirement, so correspondence good requirement would be to separate it out into two requirements. So one talks about the enrolment to undergraduate courses while the other talks about the enrolment to the post-graduate courses.

Uniquely Identified

Uniquely identified	1- Students will be able to enroll to undergraduate courses 1- Students will be able to enroll to post-graduate courses	1- Course Enrolment 1.1- Students will be able to enroll to undergraduate courses 1.2-Students will be able to enroll to post-graduate courses

Similarly the next requirement quality is to check for uniquely identified, here we have two separate requirement but they both have same ID#1. So, if we are referring our requirement with reference to ID#, but it is not clear which exact requirement we are referring to document or other part of the system as both have same ID#1. So separating out with unique id’s, so good requirement will be re-return as section 1- course enrolments, and it has two requirements 1.1 id is enrolment to undergraduate courses while 1.2 id is enrolment to postgraduate courses.

Complete

Complete	A professor user will log into the system by providing his username, password, and other relevant information	A professor user will log into the system by providing his username, password and department code

Also, each and every requirement should be complete. For example, here the bad requirement says a “professor user will log into the system by providing his username, password and other relevant information”. Here the other relevant information is not clear, so the other relevant information should be spelt out in good requirement to make the requirement complete.

Consistent and Unambiguous

Consistent and unambiguous	A student will have either undergraduate courses or post-graduate courses but not both. Some courses will be open to both under-graduate and post-graduate	A student will have either under-graduate or post graduates but not both

Next each and every requirement should be consistent and unambiguous, so here for instance we have requirements “A student will have either undergraduate courses or post-graduate courses but not both” this is one requirement there is some other requirement that says “Some courses will be open to both under-graduate and post-graduate students”.

The problem in this requirement is that from the first requirement it seems that the courses are divided into two categories under graduate courses and post graduate courses and student can opt either of two but not both. But when you read other requirement it conflicts with the first requirement and it tells that some courses will open to both post-graduate and under-graduate.

So it is obvious to convert this bad requirement into good requirement which is “A student will have either under-graduate courses or post-graduate courses but not both”. Which means that every course will be marked either being as under-graduate course or post-graduate course

Traceable

Traceable	Maintain student information-mapped to BRD req.ID?	Maintain student information-Mapped to BRD req ID 4.1

Each and every requirement should be traceable because there are already different levels of requirement, we already saw that at the top we had business requirements, and then we have an architectural and design requirements followed by system integration requirements.

Now when we convert business requirement into architectural and design requirements or we convert architectural and design requirements to system integration requirements there has to be traceability. Which means that we should be able to take each and every business requirements and map it to the corresponding one or more software architectural and design requirement. So here

is an example of bad requirement that says “Maintain student information – mapped to BRD req ID?” the requirement id is not given over here.

So converting it to a good requirement it says same thing but it is mapped with the requirement id 4.1. So mapping should be there for each and every requirement. Same way we have high level and low level mapping requirement, the mapping is also there between system and integration requirement to the code that implements that requirement and also there is a mapping between the system and integration requirement to the test case which test that particular requirement.

So this traceability is all across entire project

Prioritized

Prioritized	Registered student-Priority 1 Maintain User Information-Priority 1 Enroll courses-Priority 1 View Report Card-Priority 1	Register Student-Priority 1 Maintain User Information-Priority 2 Enroll courses-Priority 1 View Report Card-Priority3

Then each and every requirement must be prioritized, so the team has guideline so which requirement that able to implement first and which can be done later on. Here you can see the bad priority has register student, maintain user information and each and every requirement has given priority-1. Everything cannot be at same priority, so requirement can be prioritized. So the example of good requirement over here is the register student and enroll courses is given the highest priority 1, while maintain user information comes below at priority 2 and then we have view report card at priority-3

Testable

Testable	Each page of the system will load in an acceptable time-frame	Register student and enrol courses pages of the system will load within 5 seconds

Each and every requirement should be testable, here the bad requirement is “each page of the system will load in an acceptable time frame”. Now there are two problems with this requirement first is that each page meaning that there can be many pages, which going to blow up the testing efforts. The other problem is that it say the page is going to load in acceptable time frame, now what is acceptable time frame? Acceptable to whom. So we have to convert the non-testable argument into a testable argument, which specifically tells about which page we are talking about “register student and enroll courses pages” and the acceptable time frame is also given which is 5 seconds.

Conclusion

So this is how we have to look at each and every requirement at appropriate level. For example, if we are going to build a software with regards to system and integration requirements. We have to look in system and integration requirements given in the software requirement specifications or user stories and apply to each and every requirement quality. Then check whether each and every requirement is atomic, uniquely identified, and complete and so on.

You Might Like:

- Business Analyst PDF (Free Download)
- 13 BEST Evernote Alternatives and Competitors (2024)
- 10 BEST Decision Making Tools for Business (2024 Update)
- 13 Best FREE OCR Software (2024 Update)
- 13 Best FREE Flowchart Software (2024)
- 17 Best Adobe Acrobat Alternatives in 2024 [Free/Paid]
- 11 Best FREE Gantt Chart Software Online (2024)
- 15 Best FREE Office Software (2024 Update)

Prev Report a Bug Next

Guru99's Headquarters

4023 Kennett Pike #50286,
Wilmington, Delaware,
United States

English