

Req. Validation & Functional  
Decomposition for V&V  
Automation Testing

Lesson 3: Requirements Itemization

Capgemini

## Lesson Objectives



To understand the following topics:

- Requirements Itemization - How?
- Application Background
- Assumptions
- Explicit requirements
- Implicit requirements
- Interface Requirements
- Requirements Analysis to Test Scenarios
- Testable Items
- Non Testable Items
- Error Conditions
- Application Invocation
- Application Termination



## Lesson Objectives

To understand the following topics:

- File Handling
- Requirement Prioritization
- Case Study – Equipment Tracking System
- Template – 1 Equipment tracking System
- Template – 2 Equipment tracking System
- Summary
- Review Questions



### Requirements Itemization - How?



- Simplifies the requirements for better understanding
- Helps in identifying different Test Scenarios
- Helps in identifying testable items for the application
- The SRS document might contain non functional (e.g. performance) requirements along with Functional requirements
- Non functional requirements itemization must also be carried out

### Requirement Itemization is required for following :

1. To simplify the requirements for better understanding of the system user test
2. To identify different Test scenarios
3. To identify testable items for the application

Requirement Itemization needs to be done for both functional and non-functional requirements.

### Requirements Itemization - How?



#### Application Background

- Identify the background of the Application Under Test

#### Assumptions

- Identify the assumptions for the Application Under Test

#### Requirements Analysis and Prioritization

- Analyze and prioritize the requirements

#### Creating Test Scenarios from Requirements

- Create Test scenarios from the requirements given

**When splitting the requirements in smaller components, they can be split on the following categories:**

1. Background: Background of the application under test
2. Assumptions: Assumption those can be used while testing the application
3. Requirements: These can be divided further into Functional Requirements, Non Functional requirements and design objectives
4. Creating Test Scenarios from Requirements
5. Error conditions
6. Application Invocation
7. Application Termination
8. File Handling

## Requirements Itemization - How?



### Error Conditions

- Identify expected and unexpected error conditions

### Application Invocation

- Identify the different ways to start the application

### Application Termination

- Identify the different ways to end the application

### File Handling

- Analyze how file handling takes place in the application

1. **Error conditions** : Exceptions to be considered in the application
2. **Application Invocation** : Different ways to invoke the application
3. **Application Termination** : Different ways of terminating the application
4. **File Handling** : Different files that getting referred, used and updated through application

## Application Background



### System Requirements for the application

- Special requirements regarding operating system, size of the memory required for application to run etc.

### Module Under Test

- Modules to be tested and their scope in the application

### System Flow

- How the data will flow in the application and how different modules in the application are integrated

### Installation requirements

- Details of the test environment for Installation of the software

### Security Information – User Access rights

- This will include User access rights, authentications etc. What are the types of Users for the system under test and what will be rights given to the users

### **System Requirements for the application :**

For any application there could be special system requirements. These can be regarding operating system, size of the memory required to run the application etc.

### **Module under test :**

Find out which are the modules to be tested and their scope the application.

### **Flow of the system :**

How the data will flow in the system and how the modules are interconnected to each other?

### **Installation Requirements :**

The details on how the application will be installed for testing. Details of the test environment for Installation of the software.

### **Security information :**

This will include User access rights, authentications etc. What are the types of Users for the system under test and what will be rights given to the users.

## Assumptions



Assumptions may be detailed in the requirements document

These assumptions could be related to the users, functional specifications, input data etc.

- E.g. - Assumption in an equipment tracking system
  - Equipment entry is already in the system as Purchase phase is over
  - Equipment 'Use status' can be 'In Use' if it is installed and assigned to some user/Department or it can be 'In stock' if it is not put in Use

### **Assumptions are mentioned in the SRS**

These are the certain conditions or status, if it is present or existing, then the application under the test can perform its functionality.

In other words, it can be termed as pre-requisites for the application to carry out its functionality.



## Explicit requirements



Explicit Requirements are requirements

- Which are explicitly stated in the SRS
- These are requirements that are received from the customer
  - E.g. ATM machine shall validate PIN number entered by the customer

**The Explicit requirements** are the ones which the customer has specifically asked to deliver, it is the expected output or result of the system.

E.g. Every user should be validated before he accesses the Accounting system by the username/password protection. This is specified by the customer, hence it is a explicit requirement.

### Implicit requirements



Implicit Requirements are also termed as derived requirements

These requirements are generated based on the functional requirements received from the customer

These are not explicitly given by the customer

E.g. Money can be withdrawn only if there is enough Cash available in ATM machine

This requirement is not specified in the SRS by the customer. But, it has to be validated for proper functioning of the system. So, this is a implicit requirement.

## Interface Requirements



Interface requirements are requirements related to the Internal and External interfaces of the system

- E.g. Displaying updating balance

Internal interfaces are interfaces between internal modules of the System under test

- E.g. Balance is updated after credit or debit of the account

External Interfaces relate to the given system's communication with other systems.

- E.g. Transaction record is printed if requested

### **Examples of interfaces are:**

1. Person-machine interface which includes operating system, language compilers and I-O facilities
2. Communication interfaces which include transmission of information between the computer and remote equipments
3. Program interfaces which include exchange of information whether on the same computer or distributed across multiple tiers application architecture

### Requirements Analysis to Test Scenarios



When analyzing the requirements, different Test Scenarios are identified for the requirement

- E.g. Valid User Login to Equipment Tracking System (Manage Equipment) & Update the Equipments

All requirements are analyzed from Tester's point of view. First different test scenarios are identified from the requirements.

Each requirement is analyzed whether it is testable or not.

### Requirements Analysis to Test Scenarios



Determine whether the requirement is Testable or not

Queries should be routed to either the Client or to the Subject Matter expert(s) or domain specialist(s)

All functional requirements are testable requirements. Depending on the scope of the project, we will need to decide whether the non-functional requirements can be tested or not.

There are bound to have some queries while analyzing the requirements document. These queries need to be routed to the concerned person. The queries about functionality, technical queries should be sent to the client or SME (subject matter expert). SME is technical expert on the project and he has domain knowledge. The queries about functionality should be sent to the client, technical queries should be sent to SME. A separate sheet is maintained for all types of queries and those are tracked.

### Testable Items



In the requirement document there are most of the requirements that can be tested

These are the requirements that are not frequently changing

These can be data validation, form validation, field validation, & so on

For E.g. integer value validation can be tested easily

### Non Testable Items



In the requirements documents there can be some requirements which can not be tested

These requirements can be performance related and hence they may not in the scope of functional testing

There can be some requirements which changes dynamically on web applications

Such requirements may be non testable requirements

For example if the session variable for the URL in web application is stored in some HTTP file and it is changing dynamically

Some implicit requirements where query needs to be raised may be non testable in the initial stage, but they may be testable items when the query is resolved

- For example: If it is not stated who are the users for the application and what are the access rights for those users in the requirements document then such items could not be tested unless the query is resolved. Once the query is resolved these non testable items will be converted into testable items.

## Error Conditions



Expected Error Conditions - Error conditions are stated in the requirements document. These are expected errors for any Erroneous input. E.g. Invalid user name or password is entered in the login screen

Unexpected Error Conditions - These are unexpected error Conditions like Server time out or link down. While analyzing requirements we should document these error conditions separately



### Application Invocation

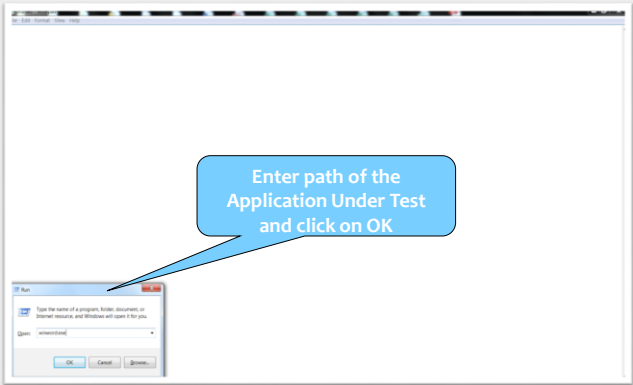


There can be number of ways in which an application can be invoked. E.g. through Windows Explorer, through Command Line, through a web form etc. Start up process involves the files required to start the application or the process involved in the application startup

### Application Invocation



Let us see how to invoke the AUT using Run option of Windows Explorer



## Application Termination



There can be number of ways in which the application can be terminated.  
E.g. Using the mouse, using hot keys etc.

Termination of application will involve shutdown process for the application



## File Handling



Different types of files are created / updated within the application

- Some files may be required before the application is started
- Some files may be required to be present while the application is being invoked
- Some files may be created or updated as the application executes
- Some files may need to be uploaded or downloaded for application execution

There are some files which are required to be present while application invocation. If any of the files is not present or is invalid then application will not invoke.

Similarly there are some files which are created or updated as the application executes. All these files must be separately documented in File Handling part from the requirements document.

## Introduction to Requirement Prioritization



Most of the software development projects consists of many candidate requirements that needs to be realized within the project constraints like time, cost and other resources

Requirement Prioritization helps to identify the most essential requirements from this set by distinguishing the critical few from the trivial many

When customer expectations are high, timelines are short and resources are limited, we need to make sure that the software system features all required functionalities

Requirement prioritization can help project teams to attain above mentioned goal by making a right choice of requirements at right time

Requirement Prioritization is the process of managing the relative importance and urgency of different requirements to cope with limited resources of projects

Prioritization must be done in collaboration with stakeholders

## Introduction to Requirement Prioritization

Most of the software development projects consists of many candidate requirements that needs to be realized within the time and cost constraint. However, if the development team cannot deliver every requirement by the scheduled initial delivery date, the project stakeholders must agree on which subset to implement first. Requirement prioritization or establishing relative priorities of the requested functionalities is an essential activity that needs to be carried out by every project, especially those are with limited resources and cost.

Requirement prioritization must be done in collaboration with stakeholders – Customers, Product Owner and Users. It is recommended to have developers and stakeholders working together in prioritization process as they always have conflicting views and needs about requirement and their importance. A common trap is to let the stakeholders choose the priorities without any guidance. In those situations, the stakeholders likely tag most requirements as being critical. To avoid this situation the business analyst must guide the customers through the proper requirement prioritization process.

The typical participants in the prioritization process include:

1. The project manager
2. Key customer representatives
3. Development representatives, such as team technical leads

### Why Requirement Prioritization?



Following are some of the aspects that signifies the importance of prioritization process in projects

- Varied levels of importance : Not all requirements are equally important, and the many different stakeholders in the system typically will not agree as to which requirements are most important.
- Limited Project Resources : All projects have limited resources in terms of budget, staff, and schedule. It is impractical to implement all the requirements during the system's current release. The incremental development approach is preferred to ensure that even trivial requirements are realized during product development.
- Long Running Projects : Incrementally developed systems requires months to years to develop during which requirements are subject to significant iteration as the business environment changes, business needs change, and new requirements are identified. These situations further increases the need for prioritization.
- Penalty : It is possible to evaluate the penalty that is introduced if some of the requirements are not fulfilled. For example, on-conform to a standard could incur a high penalty even if it is of low importance for the customer.

### Why Requirement Prioritization?

The quality of the software system is often determined by its capability of satisfying the needs of the customers & users. Therefore, eliciting and specifying the correct set of requirements and planning appropriate suitable releases with the right functionality is a major step towards the success of a project or product. Prioritization helps to identify the most valuable requirements and implement them first.

One of the important feature of a good requirement is having to be explicitly prioritized. When customer expectations are relatively high and project team required to ensure that within limited timelines and resources the software system exhibits the most essential functionalities. Establishing priorities for important functionalities lets you order the development activities to provide the greatest product value at the lowest cost.

The slide above throws light on some of the project facts that make the prioritization of requirements a critically important part of requirements analysis that every requirements engineer must perform.

## Benefits of Requirement Prioritization



Properly prioritizing requirements provides the following significant benefits to the project:

- It enables the development team and stakeholders to plan and design optimal set of software requirements for implementation in successive releases
- It helps project team to trade off desired project scope against sometimes conflicting constraints such as schedule, budget, resources, time to market, and quality
- Helps in choosing only a subset of the requirements and still produce a system that will satisfy the customer
- Ensures increase in customer satisfaction by implementing & delivering customer's most important requirements first
- It lessens the risk of project cancellation as valuable progress is being demonstrated with each increment
- Prioritizing requirements is a good way to force stakeholders to address all requirements and not just their own
- It also helps in minimizing rework and schedule slippage

## Requirement Priority Scales



The effective requirement prioritization demands the use of a ranking scheme

The number of different scales are used in practice to indicate the relative importance of a requirement like categorical scales, linear and non-linear numeric scales

A project team is responsible for deciding the ranking scheme

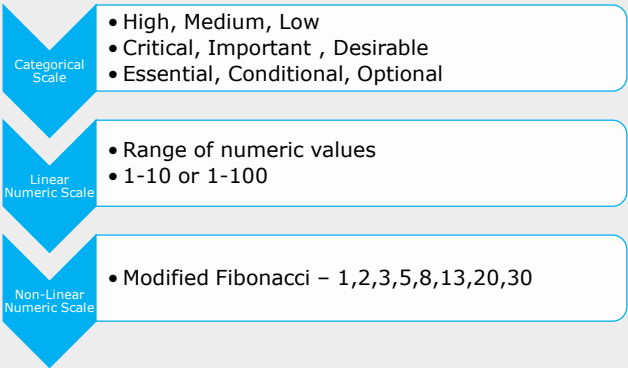
Initially, a simple categorical scale can be used to triage requirements

Then a numeric scale can be applied to further prioritize the requirements

Once the requirements are prioritized the list is ordered and implementation starts with the most important ones



### Requirement Priority Scales



### Requirement Priority Scales



It is important for all stakeholders to understand the meaning of each priority value

For a numeric scale, a small value means a low priority reduced necessity and less urgency , while a large value indicates a high priority necessary and urgent

For categorical scales, a definition of each categorical value needs to be established

Priority	Meaning
High/Critical/Essential	A critical requirement without which the product is not acceptable to the stakeholders
Medium/Important/Desirable	A necessary but deferrable requirement which makes the product less usable but still functional
Low/Conditional/Optional	A nice feature to have if there are resources but the product functions well without it

### Requirement Prioritization Process Guidelines



The requirement team needs to convince the stakeholders regarding the importance of prioritizing requirements

The stakeholders should be trained on requirement prioritization process

Categorize raw potential requirements during requirement elicitation phase into actual requirements and desirable requirements so that the actual requirements can be prioritized

During requirement analysis, requirement team should work closely with stakeholders to prioritize the actual requirements

This includes negotiation with the stakeholders to develop a consensus and validation of the resultant priorities with them

The development team that must actually implement the requirements creates and records realistic estimates of the effort required to implement each requirement

The requirement team with management & development team schedules development based on priorities of the requirements

### Requirement Prioritization Process Guidelines



During requirements management, the requirements team should work with the requirements stakeholders to maintain the requirements parameters as they change

This will typically include storing the priority as metadata in the requirements repository, and then updating the value of the priority as it changes

## Prioritization Techniques



Following are some of the known requirement prioritization techniques used to determine, negotiate, and develop a consensus regarding the priorities of the requirements:

- Business Case Analysis / Return On Investment (ROI) estimation
- Pair-wise comparisons
- Prioritization working groups
- Scale of 1-to-10 rankings
- Voting schemes (e.g., give each stakeholder a specific number of votes to distribute amongst the requirements or classes of requirements being prioritized)
- Weightings (e.g., weight the votes of different stakeholders)
- Value-Based Software Engineering [Boehm 2003]
- WIN-WIN [Boehm 2001]
- Quality Function Deployment (QFD)

Case Study – Equipment Tracking System



Case Study for requirements Itemization for an equipment tracking system

### Case Study - Step 1



Separate the application background from Requirements Document. Also separate module under test and flow of the System

- E.g. Background Of the application
- System is Equipment tracking
- The functionality is available on fat Client and on web

Module Under test - Manage equipments

Flow of the system - Purchase equipment -> Manage Equipment -> Track financials -> Retire equipment

## Case Study - Step 2



List down assumptions from the requirements document specified for the system under test

- E.g. Assumptions:
  - Equipment entry is already in the system as Purchase phase is over
  - Equipment 'Use status' can be 'In Use' if it is installed and assigned to some user/Department or it can be 'In stock' if it is not put in Use



### Case Study - Step 3



Identify the following:

- Different Scenarios (Update, Print, Report etc.)
- Testability of the scenarios and
- Type of the requirement.
  - (E.g. on the next slide)

Case Study - Step 3



Req · Id.	Scenario s	Description	Post Condition	Type	Testa ble Y /N
1	Login	Only authorized users and system can update equipment record	Functionali ty is accessible	Explicit	Yes
2	Login	Unauthorized users should not be able to login to the system	Functionali ty is not accessible	Implicit	Yes
3	Verify	Check for the equipment type properties for updating	Properties are checked	Explicit	Yes
4	Update	Updates when end date is not expired. Only valid locations and valid users are assigned to equipment record.	Updating is done	Explicit	Yes
5	Update	Update equipment record when end date is expired.	Updating is done	Explicit	Yes

Case Study - Step 4



While identifying scenarios you need to maintain a query log  
Separate queries for Client and for subject matter experts

Client	Subject matter Expert
Which functionality will be on FAT client and which on web?	How the users are categorized into : Inventory personnel, equipment auditors, service personnel, maintenance personal and Equipment Tracking personnel
What is an external system?	There is contradiction between statements 3.2.1 and 3.2.2. Equipment tag is numeric or character?
	If the equipment tag is all numeric then how 'AD' will be present in position 1 and 2 which needs to be removed

Case Study Step 5



Document the Expected and Unexpected Error conditions specified in requirements document

Req . Id.	Scenarios	Description	Post Condition	Type	Testable Y /N
9.4	Check errors generated	Check error generated when equipment tag is not found in equipment tracking system	Error generated is verified	Explicit	Yes
		Check error generated when last scan date stored in equipment tracking system is greater than last scan date stored in comp track system	Error generated is verified	Explicit	Yes

### Template - 1



This template consists of only one sheet

Requirements are separated as per different categories. E.g.

- Invocation, Termination, File Handling, error handling, system requirements, Security requirements, explicit, Implicit, interface requirements etc.

Each requirement Id will have requirements converted into test conditions and its post conditions

Query log will be maintained separately



## Template - 1

[illegible]

Template – 1 Equipment tracking System



Sr.No	Req. ID	Scenarios	Requirement	Expected Result	Category	Testable Y/N
1	2	Update	Only selected user roles (list given) can update equipment record	Selected user are able to update equipment record	Explicit	Y
2		Update	The user roles which are not in the list, can not update equipment records	Users those are not part of the given list of authorized users are not able to update equipment record	Implicit	Y
3	3.5	Query	Users can query for an equipment by "User ID"	Equipment can be queried based on "User ID"	Explicit	Y
4	3.6	Query	Users can query for an equipment by "Location" selected from a list	Equipment can be queried based on "Location" selected from a list	Explicit	Y
5	4, 4.1 to 4.6	Query	A list of equipment matching the query criteria (result set) should be displayed containing following fields : Equipment Tag, Quantity, Seq Number, User ID, Location, Equipment Type	The list of equipments displayed with mentioned fields	Explicit	Y
6	5.1	Update	Users can modify equipment record of their state only	Users are able modify equipment record successfully of their state only	Explicit	Y

Template – 1 Equipment tracking System



Sr.No	Req. ID	Scenarios	Requirement	Expected Result	Category	Testable Y/N
7		Update	Users can not modify equipment record of other state	Users are not able modify equipment record of other state	Implicit	Y
8	5.2	Update	For retired equipments : only comments field can be modified	Updation can be done to only comments field for a retired equipment	Explicit	Y
9		Update	For retired equipments all fields can not be modified except comments	Updation of all fields except comments field is restricted for a retired equipment	Implicit	Y
10	10.1 to 10.4	Report	CSV file (With given fields) should be created for equipments received in current month, excluding any "Retired" equipments	CSV file created successfully for equipments received in current month, excluding any "Retired" equipments	Explicit	Y
11		Report	CSV file (With given fields) should not contain data for equipments received in non-current month and including any "Retired" equipments	CSV file is not getting created for equipments received in non current month, including any "Retired" equipments	Implicit	Y



Query Log – Equipment tracking System



Client	SME
What is authorized system? How those will update equipment record?	What are most recent rules about Seq Number? What is the format, Number or Character?
	What is the meaning of - Manually update specified data for a single equipment record?
	What is the meaning of - Manually update specified data for multiple equipments record simultaneously?
	What is unassociated equipments identified as "Spare Part" ?
	How Seq Number is related to the equipment type?

Error Conditions – Equipment tracking System



Req. ID	Scenarios	Description	Category	Post Condition	Testable Y/N
8	Update	Check error generated when the external system fails while updating equipment record (Unexpected Error)	Implicit		N
8		Check error generated when the Comp Track system fails while receiving automatic updates (Unexpected Error)	Implicit		N
		Server time-out, Link down	Implicit		Y
7		Check error generated when various validations fail during updation (Unexpected Error)	Implicit		Y

### Pros and Cons (Template - 1)



#### Pros

- Useful when requirements are for smaller projects
- Requirements can be filtered category wise for creating test cases
- Only one sheet needs to be maintained

#### CONS

- Can not be used for larger project requirements

Template - 2



Separate sheets will be maintained for different categories  
For each category, the Requirement Id with the test conditions and post conditions is written  
Query sheet will be maintained separately

Page 01-45

Template – 2 Equipment tracking System



For a case study given in lab book, in Security tab, following will be the input

Sr.	Req. ID	Scenario	Description	Post Condition	Testable Y/N
1	1	Login	Authorized users and systems must have the ability to record updates to equipment records	Records are updated	Y

Template – 2 Equipment tracking System



For Explicit tab, all the mentioned functionalities will be included

Sr.	Req. ID	Scenario	Description	Post Condition	Testable Y/N
1	5.1	Update	Users are restricted to modify equipment records only within their state	Records are updated	Y

Template – 2 Equipment tracking System



For Implicit tab, all the derived functionalities will be included

Sr.	Req. ID	Scenario	Description	Post Condition	Testable Y/N
1	5.1	Update	Users are modifying equipment records other than their state	Records are not getting modified	Y



Template – 2 Equipment tracking System



For Invocation tab, all the ways to invoke an application are to be captured

Sr.	Req. ID	Scenario	Description	Post Condition	Testable Y/N
1		Application Invocation	Through IE	Application getting invoked	Y

Template – 2 Equipment tracking System



For Termination tab, all the ways to close an application are to be captured

Sr.	Req. ID	Scenario	Description	Post Condition	Testable Y/N
1		Application Termination	Click on Close button (X) of an application	Application getting closed	Y

## Template – 2 Equipment tracking System



File handling tab will be taking care of all the files getting used through out the application

Error handling tab will be considering all the fatal and non fatal error condition for an application

System tab will be talking about all the system specification requirements as per SRS

Query tab will be taking care of queries to be forwarded to client as well as SME

### Pros and Cons (Template - 2)



#### Pros

- Useful when requirements are for larger projects
- Separate sheets are available for different categories

#### Cons

- Multiple sheets need to be maintained
- Time consuming
- Some requirements might get repeated if they fall in two or more different categories

Case Study: Query Log



Equipment Tracking System

Client	SME
What is authorized system? How these will update equipment record?	What are most recent rules about Seq Number? What is the format, Num or Char?
	What is the meaning of manually update specified data for a single equipment record?
	What is the meaning of manually update specified data for a multiple equipment record?
	What is unassociated equipment identified as spare parts?
	How Seq Number is related to the equipment type?

## Summary



In this lesson, you have learnt:

- How to perform requirement itemization?
- Requirements Analysis to Test Scenarios
- Testable Items
- Non Testable Items
- Requirement Prioritization
- Templates used in requirement itemization



**Answers:****Question1:**  
True**Question2:**  
True**Question3:**  
Implicit  
Requirements**Question4:**  
Interface  
requirements**Question5:** All  
of the above**Review Question**

Question 1: Interface requirements are requirements related to the Internal and External interfaces of the system.

- True/ False

Question 2: Requirement Prioritization is the process of managing the relative importance and urgency of different requirements to cope with limited resources of projects.

- True/ False

Question 3: \_\_\_\_\_ are also termed as derived requirements.

Question 4: \_\_\_\_\_ are requirements related to the Internal and External interfaces of the system.



### Review Question



Question 5: Which of the following are requirement prioritization techniques?

- Option 1: Pair-wise comparisons
- Option 2: Prioritization working groups
- Option 3: Scale of 1-to-10 rankings
- Option 4: All of the above

