**Project Requirements**

**<ReqM Tool>**

**NTT Data Romania**

**Street Constanta 19-21**

**Cluj Napoca City, 400158**

**2019**

**Table of Contents**

[1 Introduction 3](#_Toc6078502)

[2 Requirements 4](#_Toc6078503)

[2.1 XML Requirements Object members 5](#_Toc6078504)

[2.1.1 NeedsCoverage and ProvidesCoverage members: 5](#_Toc6078505)

[2.1.2 Description Member 6](#_Toc6078506)

[2.1.3 Status Member 6](#_Toc6078507)

[2.1.4 CreatedBy Member 6](#_Toc6078508)

[2.1.5 Priority Member 6](#_Toc6078509)

[2.1.6 Functional safety relevant 6](#_Toc6078510)

[2.1.7 Change Request Reference 7](#_Toc6078511)

[2.1.8 Review ID 7](#_Toc6078512)

[2.1.9 Type 7](#_Toc6078513)

[2.1.10 Chapter 7](#_Toc6078514)

[2.2 Version 7](#_Toc6078515)

[2.3 XML Document settings area. 7](#_Toc6078516)

[2.3.1 Document area members 7](#_Toc6078517)

[2.4 Start Tab 8](#_Toc6078518)

[2.5 Database Tab 9](#_Toc6078519)

[2.5.1 Default columns to display 9](#_Toc6078520)

[2.5.2 Filter functionality 9](#_Toc6078521)

[2.5.3 Save Settings for “filter option” 9](#_Toc6078522)

[2.5.4 Search functionality 9](#_Toc6078523)

[2.5.5 MenuStrip 9](#_Toc6078524)

[2.6 Coverage Tab 10](#_Toc6078525)

[2.6.1 Columns for the dataGrid 10](#_Toc6078526)

[2.6.2 Functionality 10](#_Toc6078527)

[2.7 Statistics Tab 11](#_Toc6078528)

[3 Functionality Requirements 12](#_Toc6078529)

[3.1 Requirements Version 12](#_Toc6078530)

# Introduction

The purpose of the Requirements Tool Project document is to outline the requirements for the Summer Students Project.

This document contains all of the software requirement specifics. It contains a general description of the types of users who will be using our application, how it is going to work, and what technologies we are using to make it work.

It will also outline and describe each component of the project.

# Requirements

| **Requirement ID** | **Description** | **Status** |
| --- | --- | --- |
| 1. | XML “needscoverage” member shall be implemented. Please see 3.1.1 chapter |  |
| 2. | XML “providescoverage” member shall be implemented. Please see 3.1.1 |  |
| 3. | Each requirement from the xml file shall support multiple members defined in chapter 3.1 |  |
| 4. | The xml file shall support a document settings area. Please see 3.2 chapter. |  |
| 5. | The C# Application shall have implemented a tab called “Start”. Please see chapter 3.3 |  |
| 6. | The C# Application shall have implemented a tab called “Database”. Please see chapter 3.4 |  |
| 7. | The C# Application shall have implemented a tab called “Coverage”. Please see chapter 3.5 |  |
| 8. | The C# Application shall have implemented a tab called “Statistics”. Please see chapter 3.6 |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## XML Requirements Object members

### NeedsCoverage and ProvidesCoverage members:

Description:

**User Requirements**

<id> REQ1

<needscoverage> REQ2

<providescoverage> N/A

**System Requirements**

<id> REQ2

<needscoverage> REQ3

<providescoverage> REQ1

**Software Requirements**

<id> REQ3

<needscoverage> src,tst

<providescoverage> REQ2

“src,tst” means that the REQ3 shall be implemented in the code (src) and also tested (tst).

The c# application will search in all code, “.c” and “.h” files, to check if the requirement REQ3 is implemented and tested.

When the developer implements a line of code, is needed to add a comment about the requirement that is implemented. Please check the following example:



Default Value for needscoverage shall be: TBD

Default Value for providesciverage shall be: TBD

TDB: Providescoverage member needs to be able to open (go to) the requirement from which is derived via hyperlink.

### Description Member

TBD:

Format:TBD

Default Values: TBD

Description shall have the functionality to display pictures.

Format of the picture text:TBD

Location of the pictures in the project folder tree: TBD

### Status Member

Description:

When you start working on some already existing requirements, you should mark them as In work and check in the xml into version manager in order for the team to be aware that those requirements will be changed.

When you finalized update on requirements, all changed requirements should be set in Ready for review state.

After review is done with no findings, all requirements will be switched to acce

Default value shall be: TBD

Values supported:

* Accepted by project
* Ready for review
* Discarded by project

### CreatedBy Member

Description:

TBD

Default value shall be: TBD

### Priority Member

Description:

TBD

Default value shall be: TBD

### Functional safety relevant

Description:

TBD

Default value shall be: TBD

Values supported:

NA

QM

ASIL A

ASIL B

ASIL C

ASIL D

### Change Request Reference

Description:

Here you should add the number of ticket based on what you make the changes (JIRA ID in our case)

Default value shall be: TBD

### Review ID

Description:

Here you should add the number of ticket used to make the review for requirements

Default value shall be: TBD

### Type

Description:

TBD

Default value shall be: TBD

Possible values:

* Description
* Requirement

### Chapter

Description:

Here you should add the name of chapter for specific requirement ( for ex: req1 is part of Chapter DCM )

Default value shall be: TBD

### Version

Description:

TBD

Default value shall be: TBD

## XML Document settings area.

### Document area members

TBD

## Start Tab

1. Start Tab shall include the title of the project. TBD
2. Start Tab shall include the NTT logo. TBD (position)
3. Start Tab shall include a short description of the project. TBD

TBD

## Database Tab

### Default columns to display

TBD: Which columns should be displayed at the start-up

### Filter functionality

1. TBD: a button to filter the columns to be showed. Save as view
2. TBD: filter for some values for a selected column

### Save Settings for “filter option”

TBD: after the user saves the columns to be showed, the settings shall be saved into XML document.

### Search functionality

TBD: to have a search list from where you can filter the rows.

### MenuStrip

MenuStrip shall have the following buttons:

* File
* Edit

#### File Button

“File” button shall have the following options(buttons):

* Open

TBD: functionality

* Save

TBD: functionality

* SaveAs

TBD: functionality

* Publish

The application shall be able to save the xml in a user friendly format.

The requirements needs to be in a format easy to read by the customer.

TBD

* Close

TBD: functionality

#### Edit Button

“Edit” button shall have the following options(buttons):

* Add row

TBD: functionality

* Delete Row

TBD: functionality

## Coverage Tab

### Columns for the dataGrid

TBD: what columns we’ll have in the Coverage Tab. With which functionality?

### Functionality

TBD: How we’ll cover all requirements.

User Req

System Req

Software Req

#### MenuStrip

Menu strip shall have the following buttons:

* Settings
* Tools

##### Settings Button

“Settings” button shall have the following options(buttons):

TBD

##### Tools Button

“Tools” button shall have the following options(buttons):

* Make Coverage

Functionality:

* + - Search in the code each ID to check if is implemented or not.
* Publish Coverage
  + - Save the coverage raport into a Excel file. This file needs to be versioned on the server.

## Statistics Tab

TBD: What statistics will be calculated

# Functionality Requirements

## Requirements Version

When a version is changed from a “lower requirement” that covers another “higher requirement, the “higher requirement” shall become “uncovered”.

TDB

When you add a new requirement, tool should check before save if any column is TBD. In case is TBD, a pop up warning should appear to ask you to check again. Save will not be done.

Tool should be able to export data like:

How many requirements are based on a filter (filter will be a merge of columns for ex: status :