UNIVERSITI PUTRA MALAYSIA

METHOD AND TOOL TO GENERATE REQUIREMENT TRACEABILITY MATRIX FOR SCRUM DEVELOPMENT METHODOLOGY

GUNAVATHI DURAI SAMY

FSKTM 2014 27
METHOD AND TOOL TO GENERATE REQUIREMENT TRACEABILITY MATRIX FOR SCRUM DEVELOPMENT METHODOLOGY

By

GUNAVATHI DURAISAMY

Thesis Submitted to the School of Graduate Studies, Universiti Putra Malaysia, in Fulfillment of the Requirement for the Degree of Master of Science

October 2014
All material contained within the thesis, including without limitation text, logos, icons, photographs and all other artwork, is copyright material of Universiti Putra Malaysia unless otherwise stated. Use may be made of any material contained within the thesis for non-commercial purposes from the copyright holder. Commercial use of material may only be made with the express, prior, written permission of Universiti Putra Malaysia.

Copyright © Universiti Putra Malaysia
DEDICATION

To my parents, family members and friends for the endless support, encouragement and patience, and also those who made this study possible.
Abstract of thesis presented to the Senate of Universiti Putra Malaysia in fulfillment of the requirement for the degree of Master of Science

METHOD AND TOOL TO GENERATE REQUIREMENT TRACEABILITY MATRIX FOR SCRUM DEVELOPMENT METHODOLOGY

By

GUNAVATHI DURAISAMY

October 2014

Chairman: Associate Professor Rodziah Atan, PhD

Faculty: Computer Science and Information Technology

Requirement traceability matrix is a table that captures the complete user and system requirement for a system. It helps to trace from requirement till testing in order to verify that the requirement is fulfilled. In SCRUM development methodology, requirement traceability matrix is used to capture the linkage of user stories between product backlog and sprint backlog documents. The linkages between the requirements are retrieved through these two documents. However, unstructured format of both documents do not help in requirement traceability. Thus, requirement traceability has become an issue for SCRUM practitioners especially for system development and maintenance phases. Therefore, this study introduce structured format of available artifacts which can create and maintain traceability link between those documents and develops a tracing tool to generate requirement traceability matrix automatically. Both the documents used in this study have to be prepared by using the proposed structured format and the developed traceability tool is able to generate the requirement traceability matrix automatically. Two case studies have been used for pre-test and post-test experiments. The result shows that the introduced structured format is very useful and it has increased the efficiency of retrieving the matrix far better than manual process. By using the proposed method and tool, it's statistically shows significant result of time saving, completeness and correctness to generate requirement traceability matrix. Thus, the proposed method and the developed tool help in achieving requirement traceability in SCRUM methodology.
Abstrak tesis yang dikemukakan kepada Senat Universiti Putra Malaysia sebagai memenuhi keperluan untuk ijazah Master Sains

KAEDAH DAN ALAT UNTUK MENJANA MATRIK KEOBELEHKESANAN KEPERLUAN BAGI METODOLOGI PEMBANGUNAN SCRUM

Oleh

GUNAVATHI DURAIAMY

October 2014

Pengerusi: Profesor Madya Rodziah Atan, PhD

Fakulti: Sains Komputer dan Teknologi Maklumat

ACKNOWLEDGEMENTS

Thanks to the Almighty God, for giving me the strength and commitment to successfully complete this study. First of all, thanks to my supervisor, Assoc. Prof. Dr. Rodziah Atan who gives me full support, courage, ideas, advices and knowledgeable supervision throughout this study.

Special thanks to my strongest source of motivation and inspiration throughout the entire process of study: my respectful mother, Nallamah. The never ending supports from my siblings, Kalaiwani and Thilaghawati. The support and understanding from my friends Dhanam and Jayanthi. The courage and motivation from my superior at work Jenny Ng.

Next, I would like to thank management of IXEON MSC for the support and understanding and also give me the change and opportunity to conduct both my experiments in their organization.

Last but not least, I would like to dedicate my sincere gratitude and appreciation to all my friends and all who is directly or indirectly has contributed to the success of this study. Cherish for all your support, guidance and help.
I certify that an Examination Committee has met on 8 October 2014 to conduct the final examination of Gunavathi a/p Duraisamy on her thesis entitled "METHOD AND TOOL TO GENERATE REQUIREMENT TRACEABILITY MATRIX FOR SCRUM DEVELOPMENT METHODOLOGY" in accordance with the Universities and University Colleges Act 1971 and the Constitution of the Universiti Putra Malaysia [P.U. (A) 106] 15 March 1998. The Committee recommends that the student be awarded the Master of Science.

Members of the Thesis Examination Committee were as follows:

**Abu Bakar bin Md Sultan, PhD**
Associate Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Chairman)

**Hazura bt. Zulzalil, PhD**
Senior Lecturer
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Internal Examiner)

**Yusmadi Yah bt Jusoh, PhD**
Senior Lecturer
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Internal Examiner)

**Wan Mohd. Nasir Wan Kadir, PhD**
Associate Professor
Faculty of Computer Science and Information Systems
Universiti Teknologi Malaysia
(External Examiner)

---

**ZULKARNAIN ZAINAL, PhD**
Professor and Deputy Dean
School of Graduate Studies
Universiti Putra Malaysia

Date:
This thesis was submitted to the Senate of Universiti Putra Malaysia and has been accepted as fulfillment of the requirement for the degree of Master of Science. The members of the Supervisory Committee were as follows:

Rodziah Atan, PhD  
Associate Professor  
Faculty of Computer Science and Information Technology  
Universiti Putra Malaysia  
(Chairman)

Abdul Azim Abdul Ghani, PhD  
Professor  
Faculty of Computer Science and Information Technology  
Universiti Putra Malaysia  
(Member)

BUJANG KIM HUAT, PhD  
Professor and Dean  
School of Graduate Studies  
Universiti Putra Malaysia
Declaration by graduate student

I hereby confirm that:
- this thesis is my original work;
- quotations, illustrations and citations have been duly referenced;
- this thesis has not been submitted previously or concurrently for any other degree at any other institutions;
- intellectual property from the thesis and copyright of thesis are fully-owned by Universiti Putra Malaysia, as according to the Universiti Putra Malaysia (Research) Rules 2012;
- written permission must be obtained from supervisor and the office of Deputy Vice-Chancellor (Research and Innovation) before thesis is published (in the form of written, printed or in electronic form) including books, journals, modules, proceedings, popular writings, seminar papers, manuscripts, posters, reports, lecture notes, learning modules or any other materials as stated in the Universiti Putra Malaysia (Research) Rules 2012;
- there is no plagiarism or data falsification/fabrication in the thesis, and scholarly integrity is upheld as according to the Universiti Putra Malaysia (Graduate Studies) Rules 2003 (Revision 2012-2013) and the Universiti Putra Malaysia (Research) Rules 2012. The thesis has undergone plagiarism detection software.

Signature: ___________________________  Date: ___________________

Name and Matric No.: __________________________________________
Declaration by Members of Supervisory Committee

This is to confirm that:
• the research conducted and the writing of this thesis was under our supervision;
• supervision responsibilities as stated in the Universiti Putra Malaysia (Graduate Studies) Rules 2003 (Revision 2012-2013) are adhered to.

Signature: ___________________  Signature: ___________________

Name of Chairman of Supervisory Committee: ___________________  Name of Member of Supervisory Committee: ___________________
**TABLE OF CONTENTS**

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT</td>
<td>i</td>
</tr>
<tr>
<td>ABSTRAK</td>
<td>ii</td>
</tr>
<tr>
<td>ACKNOWLEDGEMENTS</td>
<td>iii</td>
</tr>
<tr>
<td>APPROVAL</td>
<td>iv</td>
</tr>
<tr>
<td>DECLARATION</td>
<td>vi</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>x</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>xiii</td>
</tr>
<tr>
<td>LIST OF ABBREVIATIONS</td>
<td>xv</td>
</tr>
<tr>
<td>LIST OF APPENDICES</td>
<td>xvi</td>
</tr>
</tbody>
</table>

## CHAPTER

### 1 INTRODUCTION
1.1 Introduction                                           1
1.2 SCRUM Methodology                                      3
  1.2.1 SCRUM Documentation                                 3
  1.2.2 Traceability in SCRUM                                4
1.3 Problem Statement                                      4
1.4 Research Question                                      5
1.5 Objective                                              5
1.6 Scope and Assumption                                   6
1.7 Significance of the study                               6
1.8 Definition of Terms                                     6
1.9 Organization of Thesis                                  9

### 2 LITERATURE REVIEW
2.1 Introduction                                           10
2.2 Agile Methodology                                      10
2.3 SCRUM Methodology                                      12
2.4 Requirement Traceability in SCRUM                       14
2.5 Importance of Requirement Traceability                  15
2.6 Requirement Traceability Matrix                         16
2.7 Requirement Traceability Tool                           17
2.8 Requirement Traceability Technique                      19
2.9 Requirement Traceability Approach                       19
2.10 Summary                                               21

### 3 RESEARCH METHODOLOGY
3.1 Introduction                                           22
3.2 Research methodology                                   22
3.3 Formulation of research problem                         28
3.4 Literature Review                                       31
3.5 Experiment Design                                       31
3.6 Clarification of problem                                37
3.7 Conduct Pre-test Experiment                             38
3.8 Proposed Solution                                       42
3.9 Design and Implementation 42
3.10 Evaluation 43
3.11 Hardware and Software Requirement 44
3.12 Summary 45

4 PROPOSED METHOD AND TOOL 46
4.1 Introduction 46
4.2 Proposed Solution 46
4.3 Design for generating RTM 57
4.4 User Interface Design 60
4.5 System Implementation 61
4.6 Summary 64

5 RESULT AND DISCUSSION 65
5.1 Introduction 65
5.2 TraceMatrix User Acceptance Testing 65
5.3 Results of Efficiency 67
5.4 Results of completeness and correctness 91
5.5 Summary 102

6 CONCLUSION AND FUTURE WORK 103
6.1 Conclusion 103
6.2 Future Work 104

REFERENCES 106
APPENDICES 112
BIODATA OF STUDENT 173
LIST OF PUBLICATION 174
# LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Comparison between Agile and Waterfall</td>
<td>12</td>
</tr>
<tr>
<td>2.2</td>
<td>Existing traceability tool on market</td>
<td>18</td>
</tr>
<tr>
<td>3.1</td>
<td>Experiment Design</td>
<td>31</td>
</tr>
<tr>
<td>3.2</td>
<td>Scope of Interview Question</td>
<td>37</td>
</tr>
<tr>
<td>3.3</td>
<td>Tested Hypotheses for time taken</td>
<td>39</td>
</tr>
<tr>
<td>3.4</td>
<td>Tested Hypotheses for precision and recall</td>
<td>40</td>
</tr>
<tr>
<td>3.5</td>
<td>Keyword and correct links for forward traceability</td>
<td>41</td>
</tr>
<tr>
<td>3.6</td>
<td>Keyword and correct links for backward traceability</td>
<td>41</td>
</tr>
<tr>
<td>3.7</td>
<td>Test Case Template</td>
<td>43</td>
</tr>
<tr>
<td>3.8</td>
<td>Software and Hardware Requirement</td>
<td>44</td>
</tr>
<tr>
<td>4.1</td>
<td>Columns for product backlog</td>
<td>49</td>
</tr>
<tr>
<td>4.2</td>
<td>Columns for sprint backlog</td>
<td>49</td>
</tr>
<tr>
<td>5.1</td>
<td>Test case result for UAT</td>
<td>65</td>
</tr>
<tr>
<td>5.2</td>
<td>Number of test cases pass and fail</td>
<td>66</td>
</tr>
<tr>
<td>5.3</td>
<td>Time taken to prepare each document for VCR</td>
<td>67</td>
</tr>
<tr>
<td>5.4</td>
<td>Paired samples statistics-time taken to prepare document for VCR</td>
<td>68</td>
</tr>
<tr>
<td>5.5</td>
<td>Paired samples test for time taken to prepare document for VCR</td>
<td>68</td>
</tr>
<tr>
<td>5.6</td>
<td>Time taken to prepare each document for SAMS</td>
<td>69</td>
</tr>
<tr>
<td>5.7</td>
<td>Paired samples statistic-time taken to prepare document for SAMS</td>
<td>70</td>
</tr>
<tr>
<td>5.8</td>
<td>Paired samples test for time taken to prepare document for SAMS</td>
<td>70</td>
</tr>
<tr>
<td>5.9</td>
<td>Time taken to store document for VCR</td>
<td>71</td>
</tr>
<tr>
<td>5.10</td>
<td>Paired samples statistics-time taken to store document for VCR</td>
<td>72</td>
</tr>
<tr>
<td>5.11</td>
<td>Paired samples test for time taken to store document for VCR</td>
<td>72</td>
</tr>
</tbody>
</table>
5.12 Time taken to store document for SAMS  

5.13 Paired samples statistics-time taken to store document for SAMS  

5.14 Paired samples test for time taken to store document for SAMS  

5.15 Time taken to retrieve document for VCR  

5.16 Paired samples statistics-time taken to retrieve document for VCR  

5.17 Paired samples test for time taken to retrieve document for VCR  

5.18 Time taken to retrieve document for SAMS  

5.19 Paired samples statistic-time taken to retrieve document for SAMS  

5.20 Paired samples test for time taken to retrieve document for SAMS  

5.21 Time taken to generate RTM for forward traceability for each keyword in VCR  

5.22 Paired samples statistics-time taken to generate RTM for forward traceability for each keyword in VCR  

5.23 Paired samples test for time taken to generate RTM for forward traceability for each keyword in VCR  

5.24 Time taken to generate RTM for forward traceability for each keyword in SAMS  

5.25 Paired samples statistics-time taken to generate RTM for forward traceability for each keyword in SAMS  

5.26 Paired samples test for time taken to generate RTM for forward traceability for each keyword in SAMS  

5.27 Time taken to generate RTM for backward traceability for each keyword in VCR  

5.28 Paired samples statistics-time taken to generate RTM for backward traceability for each keyword in VCR  

5.29 Paired samples test for time taken to generate RTM for backward traceability for each keyword in VCR  

5.30 Time taken to generate RTM for backward traceability for each keyword in SAMS  

5.31 Paired samples statistics-time taken to generate RTM for backward traceability for each keyword in SAMS  

5.32 Paired samples test for time taken to generate RTM for backward traceability for each keyword in SAMS
5.33  Paired samples statistics-precision and recall for both treatments on forward traceability VCR 91
5.34  Paired samples test-precision and recall for both treatments on forward traceability VCR 93
5.35  Paired samples statistics-precision and recall for both treatments on backward traceability VCR 95
5.36  Paired samples test-precision and recall for both treatments on backward traceability VCR 96
5.37  Paired samples statistics-precision and recall for both treatments on forward traceability SAMS 97
5.38  Paired samples test-precision and recall for both treatments on forward traceability SAMS 98
5.39  Paired samples statistics-precision and recall for both treatments on backward traceability SAMS 100
5.40  Paired samples test-precision and recall for both treatments on backward traceability SAMS 101
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Sample Requirement Traceability Matrix</td>
<td>7</td>
</tr>
<tr>
<td>1.2</td>
<td>Sample Product Backlog</td>
<td>8</td>
</tr>
<tr>
<td>1.3</td>
<td>Sample Sprint Backlog</td>
<td>8</td>
</tr>
<tr>
<td>2.1</td>
<td>Perceived IT project success rates by paradigm</td>
<td>11</td>
</tr>
<tr>
<td>2.2</td>
<td>Process of SCRUM methodology</td>
<td>13</td>
</tr>
<tr>
<td>3.1</td>
<td>Diagram of research methodology</td>
<td>23</td>
</tr>
<tr>
<td>3.2</td>
<td>Phase 1 activities</td>
<td>24</td>
</tr>
<tr>
<td>3.3</td>
<td>Phase 2 activities</td>
<td>25</td>
</tr>
<tr>
<td>3.4</td>
<td>Phase 3 activities</td>
<td>26</td>
</tr>
<tr>
<td>3.5</td>
<td>Phase 4 activities</td>
<td>27</td>
</tr>
<tr>
<td>3.6</td>
<td>Manual process for forward traceability</td>
<td>29</td>
</tr>
<tr>
<td>3.7</td>
<td>Manual process for backward traceability</td>
<td>30</td>
</tr>
<tr>
<td>3.8</td>
<td>Overview of manual activity</td>
<td>34</td>
</tr>
<tr>
<td>3.9</td>
<td>Detail activities for manual operation</td>
<td>35</td>
</tr>
<tr>
<td>4.1</td>
<td>Product Backlog for new project in Excel</td>
<td>47</td>
</tr>
<tr>
<td>4.2</td>
<td>SCRUM board for research project using sticky note</td>
<td>48</td>
</tr>
<tr>
<td>4.3</td>
<td>Sprint backlog in Word</td>
<td>48</td>
</tr>
<tr>
<td>4.4</td>
<td>New format for Product Backlog</td>
<td>50</td>
</tr>
<tr>
<td>4.5</td>
<td>New format for Product Backlog</td>
<td>50</td>
</tr>
<tr>
<td>4.6</td>
<td>Linkage between product backlog and sprint backlog</td>
<td>51</td>
</tr>
<tr>
<td>4.7</td>
<td>Relationship among the documents</td>
<td>52</td>
</tr>
<tr>
<td>4.8</td>
<td>Traceability tool design</td>
<td>53</td>
</tr>
<tr>
<td>4.9</td>
<td>Add Project flow</td>
<td>54</td>
</tr>
<tr>
<td>Section</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>---------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>4.10</td>
<td>Search Project</td>
<td>55</td>
</tr>
<tr>
<td>4.11</td>
<td>Upload Document</td>
<td>55</td>
</tr>
<tr>
<td>4.12</td>
<td>View Document</td>
<td>56</td>
</tr>
<tr>
<td>4.13</td>
<td>Delete document</td>
<td>56</td>
</tr>
<tr>
<td>4.14</td>
<td>Generate Requirement Traceability Matrix</td>
<td>57</td>
</tr>
<tr>
<td>4.15</td>
<td>Forward traceability</td>
<td>58</td>
</tr>
<tr>
<td>4.16</td>
<td>Backward Traceability</td>
<td>59</td>
</tr>
<tr>
<td>4.17</td>
<td>User Interface Design 1</td>
<td>60</td>
</tr>
<tr>
<td>4.18</td>
<td>User Interface Design 2</td>
<td>60</td>
</tr>
<tr>
<td>4.19</td>
<td>Home page of the traceability tool</td>
<td>62</td>
</tr>
<tr>
<td>4.20</td>
<td>Search Project</td>
<td>62</td>
</tr>
<tr>
<td>4.21</td>
<td>Add Project</td>
<td>63</td>
</tr>
<tr>
<td>4.22</td>
<td>Upload Document</td>
<td>63</td>
</tr>
<tr>
<td>4.23</td>
<td>Generate Requirement Traceability Matrix</td>
<td>63</td>
</tr>
<tr>
<td>5.1</td>
<td>Number of test cases pass and fail</td>
<td>66</td>
</tr>
<tr>
<td>5.2</td>
<td>Time taken to prepare document for VCR</td>
<td>68</td>
</tr>
<tr>
<td>5.3</td>
<td>Time taken to prepare document for SAMS</td>
<td>70</td>
</tr>
<tr>
<td>5.4</td>
<td>Time taken to store document for VCR</td>
<td>71</td>
</tr>
<tr>
<td>5.5</td>
<td>Time taken to store document for SAMS</td>
<td>73</td>
</tr>
<tr>
<td>5.6</td>
<td>Time taken to retrieve document for VCR</td>
<td>74</td>
</tr>
<tr>
<td>5.7</td>
<td>Time taken to retrieve document for SAMS</td>
<td>76</td>
</tr>
</tbody>
</table>
# LIST OF ABBREVIATIONS

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Full Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMMI</td>
<td>Capability Maturity Model Integrated</td>
</tr>
<tr>
<td>UML</td>
<td>Unified Modeling Language</td>
</tr>
<tr>
<td>XP</td>
<td>eXtreme Programming</td>
</tr>
<tr>
<td>CASE</td>
<td>Computer Aided Software Engineering</td>
</tr>
<tr>
<td>MTTC</td>
<td>Mean Time To Close</td>
</tr>
<tr>
<td>ERD</td>
<td>Entity Relationship Diagram</td>
</tr>
<tr>
<td>PHP</td>
<td>Hypertext Preprocessor</td>
</tr>
<tr>
<td>HTTP</td>
<td>Hypertext Transfer Protocol</td>
</tr>
<tr>
<td>RDBMS</td>
<td>Relational Database Management System</td>
</tr>
<tr>
<td>RAM</td>
<td>Random Access Memory</td>
</tr>
<tr>
<td>UI</td>
<td>User Interface</td>
</tr>
<tr>
<td>RTM</td>
<td>Requirement Traceability Matrix</td>
</tr>
<tr>
<td>UAT</td>
<td>User Acceptance Test</td>
</tr>
<tr>
<td>PMO</td>
<td>Project Management Officer</td>
</tr>
<tr>
<td>SIT</td>
<td>System Integration Testing</td>
</tr>
<tr>
<td>VCR</td>
<td>Visual Custom Report</td>
</tr>
<tr>
<td>SAMS</td>
<td>Sports Administrative Management System</td>
</tr>
</tbody>
</table>
## LIST OF APPENDIES

<table>
<thead>
<tr>
<th>Appendix</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Agile Methodology</td>
<td>112</td>
</tr>
<tr>
<td>B</td>
<td>Research Planning</td>
<td>120</td>
</tr>
<tr>
<td>C</td>
<td>Interview Question &amp; Review</td>
<td>123</td>
</tr>
<tr>
<td>D</td>
<td>UML Diagram &amp; Codes</td>
<td>126</td>
</tr>
<tr>
<td>E</td>
<td>Test Cases &amp; Defect Tracking Log</td>
<td>147</td>
</tr>
<tr>
<td>F</td>
<td>User Manual of Traceability Tool</td>
<td>155</td>
</tr>
</tbody>
</table>
CHAPTER 1

INTRODUCTION

1.1 Introduction

Software production increases in conjunction with fastest growing computing technology. It is getting larger day by day with increased number of artifacts along with the source codes. Software development processes have to go through several development life cycles since it is a complex endeavor. Each life cycle focuses on different level of software development activity such as planning, requirement gathering, requirement elicitation, analysis, design, development, implementation, validation and maintenance. Different kind of documentation is created for each level by different people and it requires a high level of maintenance to ensure these documents are up-to-date.

Varieties of studies conducted to measure the success rate of software development projects and numerous software development methodologies were introduced to achieve delivering software on time. However, projects are often delivered over schedule with many quality issues. Agile development methodology is new and one of the preferred methodology in recent years (Scott, 2011) as it focuses on improving the development process. It is designed in a way that the end product delivered continuously with frequent changes.

Agile is a philosophy was initiated by 17 signatories in 2001. A group of software practitioners and consultants (Lee & Xia, 2010; Cockburn, 2006; Beck et al., 2001) publish four agile manifestos as below. More details about these manifestos are described in Appendix A.

“We are uncovering better ways of developing software by doing it and helping others do it. We value:

1. Individual and interaction over process and tools.
2. Working software over comprehensive documentation
3. Customer collaboration over contract negotiation
4. Responding to change over following a plan
That is, while there is value in the items on the right, we value items on the left more.” (Agile Alliance, 2001b).

Documentation is one of the important activities in software engineering. Software documents explain how the software operates or how to use it and it means differently to people in different role. Large amount of associated documents are being generated for large software development projects. The common types of documentation being prepared are requirement, design, technical, end user and marketing specification. Creating and maintaining these documentations need a higher amount of costs. Therefore, the software development team has to give an equal attention and effort for documentation as of to the software development.

However, this is different in organization that is practicing agile methodology. Agile methodology focuses more on working software than the comprehensive documentation, but it doesn't mean that agile not producing any document. Agilists create documentations only when there is a need at appropriate point in the life cycle. Therefore tracing requirement from the available document is hard and sometimes impossible for developers (Cleland-Huang, 2012).

Traceability is defined as “the degree to which a relationship can be established between two or more products of the development process, especially products having predecessor-successor or master-subordinate relationship to one another” by IEEE (2004). Requirement traceability is “the ability to describe and follow the life of a requirement, both forward and backward direction, ideally through the entire system lifecycle” (Mäder & Gotel, 2012; Cleland-Huang, Gotel, Hayes, Mäder, & Zisman, 2014; Hemalatha & Prakash, 2013). Requirement can be traced in two directions (Mäder & Gotel, 2012; Hemalatha & Prakash, 2013):

i. Backward traceability: “following the requirement back to its origin”.
ii. Forward traceability: “tracing the requirement to the modules and functions by which it is implemented”.

In case of traceability is missing among the requirement or documentation being prepared, the main issue or problem faced by project team is the hardness to get the product to meet the original or core requirement in sequence. When there is a missing linkage in traceability between requirement and design specification, there are possibilities of delivering an
end product which does not meet the intended outcome. This will lead to low quality in the delivered software.

1.2 SCRUM Methodology

SCRUM defined by Ken Schwaber as “a process that accepts that the development process is unpredictable, formalizing the “do what it takes” mentality, and has found success with numerous independent software vendors.” (Schwaber & Beedle, 2002). SCRUM has become increasingly popular in the past decade (Schwaber & Beedle, 2002). It practices time-boxed process as:

i. Iterative: the product is produced during the small cycles called iterations.
ii. Incremental: the functionality of the product increase during each iteration by adding new properties.

Communication among team members, team collaboration in completing the task, faster way of exchanging information between team members, teamwork, workable end product and flexibility are the key attributes of SCRUM which differ from other traditional methodologies. SCRUM allows frequent changes in specification as per end user requirement.

1.2.1 SCRUM Documentation

As the second manifesto of agile methodology (working software over comprehensive documentation) is being concerned, document or artifacts produced in SCRUM methodology are limited to:

1. Product Backlog: This document contains the whole list of business and technical functionality to be developed.
2. Sprint Backlog: Details for each item from product backlog are logged in this document. The list consists of business and technology features, enhancements and defect which are planned for current iteration.
3. Burndown Chart: Remaining hours to complete for each of the item in sprint backlog are graphed in this chart.
1.2.2 Traceability in SCRUM

As the requirement changes frequently at the end of each iteration, these changes have to be considered to be included in the next iteration. When there is no proper documentation has been prepared, the sprint backlog and product backlog have to be updated with the current changes required or take place as well as with the existing requirement list. At this point of time, SCRUM team faces requirement traceability linkage issues to see the impacted requirements which are already implemented and to be implemented.

1.3 Problem Statement

Traceability is essential in assuring that system is conform to requirements with terms are defined and used consistently which corresponding to the structures of models (Alexander, 2002). Even though, various standards and government agencies (Asuncion, François & Taylor, 2007; Leffingwell & Widrig, 2002) encouraging and mandating traceability, many organization take it infeasible to incorporate traceability into their practices (Alexander, 2002). Currently the information about implemented requirements, code changes or test results are traced manually (Azmi, Ibrahim & Mahrin, 2011) and it is time consuming (Brinkkemper, 2004).

Traceability links are typically stored “in a trace matrix” that is constructed manually by team members during system development (Port, Nikora, Hihn & Huang, 2011; Borg, Runeson & Ardö, 2013). Building and maintaining complete and accurate trace matrices is “arduous and effort consuming” and so practitioners often fail to implement consistent and effective traceability processes (Espinoza & Garbajosa, 2011; Cleland-Huang, 2012). The “sheer number of artifacts” produced in a project, the differing levels of formality and specifically between various artifact types and the complex interrelationships between artifacts (Mäder & Gotel, 2012; Alexander, 2002) are the main reason of traceability links problem. SCRUM documents are only limited to product backlog and sprint backlog, and codes are treated as the final specification. Limited content of the documents in SCRUM leads to limited references in requirement traceability.

Developers view traceability as a heavyweight and burdensome activity (Cleland-Huang, 2012; Zhang et al., 2010). To understand the requirement,
developers are depending on the documentations. Without documentation, source codes are the only reference point for them which consume their time more and they tend to make mistakes easier (Azmi, Ibrahim & Mahrin, 2011). Traceability is “the ability to link between various artifacts in software development phases linking requirements, design, source code and testing artefacts” (De Lucia, Marcus, Oliveto & Poshylvanyk, 2012; Ramesh & Jarke, 2001). Unfortunately, difficulties in using and maintaining traceability links causing many organizations failed to implement effective traceability. On top of that, developers rarely apply formal requirement specification techniques in practice while creating or maintaining requirement documentation. Due to this there are no proper linkage and identification in the documents in order to create or maintain the traceability links in SCRUM.

All three problems which are, creating and maintaining traceability links among documents, time consuming in current manual method and informal structure of the document showed the necessity of proposed solution of this research. This study proposed a method to create and maintain links between product backlog and sprint backlog documents which eventually will reduce the time taken in generating requirement traceability matrix.

1.4 Research Question

Below are the research questions identified for the stated research problems:

1. How to structure the existing format and content?
2. How to create traceability reference links among the documents?
3. How to retrieve requirement traceability matrix efficiently?

1.5 Objective

In this research, impact and benefits of traceability in SCRUM software development methodology will be studied to enable the development team to create and maintain requirement traceability links.

Therefore, the study aims to achieve the following objectives:
1. To propose a method that is able to effectively create and maintain requirement traceability links in SCRUM.

2. To validate and demonstrate the applicability of the proposed method by developing its supported tool and conducting experiments.

1.6 Scope and Assumption

This study will focus on Agile – SCRUM methodology for requirement traceability among functional requirement and design. The requirement traceability will be based on the documents prepared by selected SCRUM team which are product backlog and sprint backlog.

It is presumed by this study that the documents used are only SCRUM documents prepared by organization which currently practices the SCRUM development methodology.

1.7 Significance of the study

As this research aims to develop a requirement traceability method and tool for SCRUM methodology, the outcome of this study is to reduce the time taken to retrieving requirement traceability matrix automatically whenever needed by SCRUM team members. By achieving this, it would be beneficial for the organization who practicing SCRUM to maintain the bidirectional traceability among requirement even with limited documentation.

1.8 Definition of Terms

This part describes the several definitions that are used in this research. These definitions are by no means comprehensive, but it will provide a focal point for terms to be used in the following chapters. These definitions are as follows:
Definition 1: Requirement Traceability

“A characteristic of a system in which the requirements are clearly linked to their sources and to the artifacts created during the system development life cycle based on these requirements”. (Mäder & Gotel, 2012; Hemalatha & Prakash, 2013; Ramesh & Jarke, 2001).

Definition 2: Requirement Traceability Matrix

A table consists of features or requirement traces in both forward and backwards direction. A sample of requirement traceability matrix is shown in Figure 1.1.

<table>
<thead>
<tr>
<th>Requirement Source</th>
<th>Product Requirements</th>
<th>HLD Section #</th>
<th>LLD Section #</th>
<th>Code Unit</th>
<th>UTS Case #</th>
<th>STS Case #</th>
<th>User Manual</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business Rule #1</td>
<td>R00120 Credit Card Types</td>
<td>4.1 Parse Mag Strip</td>
<td>4.1.1 Read Card Type</td>
<td>Read_Card _Type.c</td>
<td>UT 4.1.032</td>
<td>ST 120.020</td>
<td>Section 12</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>UT 4.1.033</td>
<td>ST 120.021</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>UT 4.1.038</td>
<td>ST 120.022</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>UT 4.1.043</td>
<td>ST 120.023</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>ST 120.024</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Use Case #132</td>
<td>R00230 Read Gas Flow</td>
<td>7.2.2 Gas Flow Meter Interface</td>
<td>7.2.2 Read Gas Flow Indicator</td>
<td>Read_Gas Flow Indicator.c</td>
<td>UT 7.2.043</td>
<td>ST 230.002</td>
<td>Section 21.1.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>UT 7.2.044</td>
<td>ST 230.003</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>ST 230.004</td>
<td>ST 230.005</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>ST 230.006</td>
<td>ST 230.007</td>
<td></td>
</tr>
</tbody>
</table>

Source of sample image: www.westfallteam.com

Figure 1.1: Sample Requirement Traceability Matrix

Definition 3: Product Backlog

Product backlog contains list of features and description of the functionality. Product owner have to maintain this documents and responsible to prioritize the product backlog items. It comprises features, bugs, technical work and knowledge acquisition as shown in Figure 1.2.
Definition 4: Sprint Backlog

Sprint backlog consist of list of task to be completed for particular sprint or iteration as shown in Figure 1.3. Team selects the product backlog items and identifies the tasks to be done. After estimate the effort needed, the task will be assigned to the team members.
1.9 Organization of thesis

This thesis is organized into six (6) chapters. It starts with Chapter 1 which covers the introduction, background of the research, problem statements, objectives, scope, significance of the study, and definition of terms.

Chapter 2 describes the literature review of this project. The agile methodology, SCRUM, traceability methods, requirement traceability, traceability tool, and documents which are related to this study will be described in this chapter.

Chapter 3 describes the research methodology used in this study. The step by step activities have been explained in detail in this chapter.

Chapter 4 describes the system design and implementation of the developed traceability tool. In this chapter, the detail design of the tool has been drawn and explained.

Chapter 5 shows the results and discussion on the developed tool based on the metric and attribute set. The result gathered for pre-test and post-test for both case studies.

Chapter 6 summarizes the overall study as a conclusion and some outline for the future work.
REFERENCES


Brinkkemper, S. (2004). Requirements engineering research the industry is and is not waiting for. In Proceedings of the 10th anniversary international workshop on requirements engineering: foundation for software quality (REFSQ'04), Riga, Latvia (pp. 41-54).


Cleland-Huang, J. (2006) Just enough requirements traceability. COMPSAC 1, 41–42


