USABILITY BASED RISK ASSESSMENT MODEL FOR SOFTWARE DEVELOPMENT PROCESS

JAYALETCHUMI A/P T. SAMBANTHA MOORTHY

UNIVERSITI TEKNOLOGI MALAYSIA
USABILITY BASED RISK ASSESSMENT MODEL FOR SOFTWARE DEVELOPMENT PROCESS

JAYALETCHUMI A/P T. SAMBANTHA MOORTHY

A thesis submitted in fulfilment of the requirements for the award of the degree of Doctor of Philosophy

Advanced Informatics School
Universiti Teknologi Malaysia

JUNE 2016
Dedicated to my family especially my son, Kuhen who has been a strong and steadfast support in my PhD journey. They taught me the value of life and faithful love. Most of all, I cannot fully express in words for priceless love and encouragement that my son has brought in my life.
ACKNOWLEDGEMENT

Firstly, I would like to thank god for giving me determination and strength in completing this thesis.

Next, I would like to express immeasurable appreciation and deepest gratitude to my supervisors, Prof. Dr. Suhaimi Bin Ibrahim and Dr. Mohd Naz’ri Bin Mahrin for their untiring support, constructive input, advices, words of encouragement and guidance in making this study possible.

I also would like to thank my close friends, Dr. Huma Khan, Dr. Sugumaran Nallusamy, Ms. Mariayee Doraisamy and Ms. Sri Lakshmi Kanniah for their inspiring discussions and continual motivation which encouraged me further in completing this study successfully. My special gratitude goes to my parent, family members and friends for their help which has ease my PhD journey.

My special thanks also go to all the experts and respondents involved directly and indirectly in my research, especially Mr. Kathirrasan and Mr. Subramani from MAMPU for their continuous advice and support during data collection for my research. Last but not the least, I would like to thank the staff at Advanced Informatics School (AIS) and other co-researchers at who have helped me in many ways to complete this research.
ABSTRACT

Software usability is an important factor in ensuring the development of quality and usable software. Ignorance, unawareness and failure to address usability during the software development process lead to poor quality software that is associated with potential usability risks. Risk management can be used to assess and control these usability risks. However, currently knowledge on usability risks is still insufficient and model to assess these risks is also lacking, leading to ignorance in managing usability risks in the software development lifecycle process (SDLC). This thesis proposes to develop a new Usability Risk Assessment Model to assessment of usability risks during the SDLC. Initially, elements of the Usability Risk Assessment Model were identified using Systematic Literature Review (SLR) whereby five major elements, namely, Risk Identification, Risk Analysis, Risk Prioritization, Risk Classification and Risk Mitigation were included in the model. Subsequently, feedback from 270 respondents of a survey questionnaire was utilized to identify 38 possible usability risk factors, which were then used to define 42 potential usability risks. These usability risks were used as keywords in identifying 85 initial usability vulnerabilities from the literature, which were grouped into four main categories that influence software development outcomes: Institutional Context, Software Project Content, People and Action, and Development Processes. The above usability risks and their vulnerabilities were then validated by four selected experts from the Public Sector. After validation, a total of 88 distinct usability vulnerabilities for various usability risks were identified. The usability risks were analysed using the Delphi method, involving seven experts to identify the probability of occurrences, impact on SDLC phases and mitigation plans for usability risks. Aided by the probability of occurrences and impact on SDLC phases, the usability risk exposure level was quantified, and used to classify and prioritize usability risks on SDLC phases. A Web-based Usability Risk Assessment Tool as a proof-of-concept was developed using ASP.Net to automate detailed elements in order to support the implementation of the model. Using this tool, multiple case study evaluations on four software projects in the Public Sector of Malaysia had demonstrated an inverse relationship between number of usability risks and usability of software. Thus, with the proposed Usability Risk Assessment Model, usability risks can effectively identified, analysed, prioritized, classified and mitigated during software development process to reduce these risks in order to enhance the usability of software. The contributions of this research are; first, a validated list of potential usability risks, usability vulnerabilities and possible mitigation plans for the usability risks; second, classification and prioritization of usability risks on SDLC phases; and third, empirically evaluated the Usability Risk Assessment Model.
ABSTRAK

## TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>CHAPTER</th>
<th>TITLE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>DECLARATION</td>
<td></td>
<td>ii</td>
</tr>
<tr>
<td>DEDICATION</td>
<td></td>
<td>iii</td>
</tr>
<tr>
<td>ACKNOWLEDGEMENT</td>
<td></td>
<td>iv</td>
</tr>
<tr>
<td>ABSTRACT</td>
<td></td>
<td>v</td>
</tr>
<tr>
<td>ABSTRAK</td>
<td></td>
<td>vi</td>
</tr>
<tr>
<td>TABLE OF CONTENTS</td>
<td></td>
<td>vii</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td></td>
<td>xiv</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td></td>
<td>xviii</td>
</tr>
<tr>
<td>LIST OF ABBREVIATIONS</td>
<td></td>
<td>xxi</td>
</tr>
<tr>
<td>LIST OF APPENDICES</td>
<td></td>
<td>xxii</td>
</tr>
</tbody>
</table>

### 1 INTRODUCTION

1.1 Background of the Problem

1.2 Problem Statement

1.3 Research Questions

1.4 Research Objectives

1.5 Scope of the Study

1.6 Significance of the Study

1.7 Glossary

1.8 Outline of the Thesis

1.9 Chapter Summary
2 LITERATURE REVIEW

2.1 Usability
   2.1.1 The Concept and Definition of Usability
   2.1.2 Usability Models
   2.1.3 Usability Attribute and Sub Attribute
   2.1.4 Usability Problem and Usability risk

2.2 Software Risk Management
   2.2.1 Software Risk Management Model
   2.2.2 Software Risk Management Process

2.3 Software Risk Assessment
   2.3.1 Existing Software Risk Assessment Model
   2.3.2 Existing Software Risk Assessment Tool

2.4 Current Issues on Usability Risk Assessment Model
   2.4.1 Integrated Usability Risk Assessment Model

2.5 Review on Possible Methods to be adopted
   2.5.1 Review of Literature
   2.5.2 Opinion/Inputs from experts
   2.5.3 Responses from population

2.6 Development of Usability Risk Assessment Model

2.7 Summary

3 RESEARCH METHODOLOGY

3.1 Research Design

3.2 Research Phases and Process
   3.2.1 Phase 1: Identification of Elements and Methods for Usability Risk Assessment Model
      3.2.1.1 Systematic Literature Review (SLR)
      3.2.1.2 Literature Review using Secondary Data Analysis
3.2.2 Phase 2: Identification of Potential Usability Risks

3.2.2.1 Defining Usability Risk Factors using Secondary Data Analysis

3.2.2.2 Survey Methodology

3.2.2.3 Identification of Usability Risk from Usability Risk Factors

3.2.2.4 Identification of Usability Vulnerabilities

3.2.2.5 Expert Review

3.2.3 Phase 3: Analysis and Prioritization of Potential Usability Risk

3.2.3.1 Delphi Method

3.2.4 Phase 4: Development of Usability Risk Assessment Model

3.2.5 Phase 5: Evaluation of Usability Risk Assessment Model

3.2.5.1 Case study

3.3 Chapter Summary

4 IDENTIFICATION OF ELEMENT AND METHODS IN USABILITY RISK ASSESSMENT MODEL

4.1 Identification of elements in Usability Risk Assessment Model

4.1.1 Research Question (RQ1)

4.1.2 Search Strategy

4.1.3 Study Selection Procedure

4.1.4 Publication Year

4.1.5 Focus of the Articles

4.1.6 Units of analysis

4.1.7 Result and Discussion
4.1.7.1 RQ1: What are the elements in Usability Risk Assessment Model? 119
4.1.7.2 Method of Analysis 121
4.1.7.3 Integration with SDLC 122
4.1.8 Analysing Validity Threats 123
4.1.8.1 Investigator Bias 123
4.1.8.2 Publication Bias 123
4.1.8.3 Primary Studies Threats 124
4.1.8.4 Data Extraction Threats 124
4.1.9 Summary 125
4.2 Identification of Methods to Implement Elements in Usability Risk Assessment Model 125
4.3 Chapter Summary 127

5 IDENTIFICATION OF USABILITY RISKS 128

5.1 Survey Conduction 128
5.1.1 Construction of Questionnaire 129
5.1.2 Pilot Survey and Reliability Test 132
5.1.3 Actual Survey 133
5.2 Result and Analysis 134
5.2.1 Descriptive Statistics of Respondents 134
5.2.2 Identification of Usability Risk Factors 138
5.2.2.1 Quantitative Analysis of the Responses 138
5.2.2.2 Qualitative Analysis of the Responses 141
5.3 Identification of Potential Usability Risk 149
5.4 Identification of Usability Vulnerabilities 151
5.5 Description of Usability Risks 152
5.6 Validation by Expert 152
5.7 Chapter Summary 161
6 ANALYSIS AND INCORPORATION OF POTENTIAL USABILITY RISK IN SOFTWARE DEVELOPMENT LIFE CYCLE (SDLC) PHASES

6.1 The Delphi Method
   6.1.1 The Delphi Rounds
   6.1.2 The Delphi Experts
   6.1.3 Duration of Delphi Study
   6.1.4 Development of Questionnaire for Delphi Study
      6.1.4.1 Phase 1 Pilot Study
      6.1.4.2 Phase 2 Pilot Study
      6.1.4.3 Phase 3 Pilot Study
      6.1.4.4 Conclusion on Pilot Study
   6.1.5 Actual Delphi Study
      6.1.5.1 Phase 1: Determining New Usability Risk and Probability of Occurrences of Usability Risk
      6.1.5.2 Phase 2 Delphi study: Determining Impact of Usability Risk on SDLC Phases
      6.1.5.3 Phase 3: Determining Mitigation Plans for Usability Risk
   6.1.6 Prioritization and Classification of Usability Risks Based on Risk Exposure Level

6.2 Discussion
   6.2.1 Probability of Occurrences of Usability Risk
   6.2.2 Impact of Usability Risk on SDLC Phases
   6.2.3 Risk exposure level, Prioritization and Classification of Usability Risks
   6.2.4 Mitigation Plans for Usability Risk

6.3 Chapter Summary
7 DEVELOPMENT OF USABILITY RISK ASSESSMENT MODEL 206

7.1 Model Development techniques 206
7.2 Usability Risk Assessment Model 211
    7.2.1 Incorporation of usability risks into SDLC phases 214
7.3 Usability Risk Assessment Tool 217
    7.3.1 Development Environment 218
    7.3.2 Architecture of the Usability Risk Assessment Tool 218
        7.3.2.1 Presentation Layer 219
        7.3.2.2 Business Layer 226
        7.3.2.3 Data Layer 229
7.4 Testing of Usability Risk Assessment Tool 229
7.5 Chapter Summary 230

8 EVALUATION OF USABILITY RISK ASSESSMENT MODEL 231

8.1 Conducting Case Study 232
8.2 Result and Analysis 234
    8.2.1 Usability of Software 234
    8.2.2 Number of Potential Usability Risk 236
    8.2.3 Perceived Usefulness, Ease of Use and Self-Predicted Future Usage of the Tool 246
        8.2.3.1 Reliability Test 246
        8.2.3.2 Perceived Usefulness 247
        8.2.3.3 Ease of Use 248
        8.2.3.4 Correlating Perceived Usefulness, Ease of Use, and Self-predicted Future Usage 249
8.3 Discussion 250
8.4 Study Validity 252
9  DISCUSSIONS AND FUTURE WORK  256

9.1  Research Summary  256
  9.1.1  RQ1: What are the elements and methods in Usability Risk Assessment Model?  257
    9.1.1.1  Objective 1: To investigate elements and methods in Usability Risk Assessment Model  257
  9.1.2  RQ2: How potential usability risks can be incorporated into SDLC phases?  258
    9.1.2.1  Objective 2: To identify potential usability risk that influence usability of software  258
    9.1.2.2  Objective 3: To analyse and prioritize potential usability risks in SDLC phases  259
  9.1.3  RQ3: How a suitable Usability Risk Assessment Model can be proposed based on the above findings?  260
    9.1.3.1  Objective 4: To develop the proposed Usability Risk Assessment Model  260
    9.1.3.2  Objective 5: To evaluate the proposed Usability Risk Assessment Model  261

9.2  Contribution and Significance of Study  261
9.3  Limitations of the Research  264
9.4  Conclusion and Future Work  265

REFERENCES  267

Appendices A- M  289-328
# LIST OF TABLES

<table>
<thead>
<tr>
<th>TABLE NO.</th>
<th>TITLE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Comparison of quality attributes in various quality models</td>
<td>16</td>
</tr>
<tr>
<td>2.2</td>
<td>Usability attributes defined by Alonso-Ríos (2009) and Dubey (2012)</td>
<td>19</td>
</tr>
<tr>
<td>2.3</td>
<td>Usability attributes and sub-attributes by Dubey (2012)</td>
<td>20</td>
</tr>
<tr>
<td>2.4</td>
<td>List of top ten software risks</td>
<td>24</td>
</tr>
<tr>
<td>2.5</td>
<td>Comparison of risk management models</td>
<td>28</td>
</tr>
<tr>
<td>2.6</td>
<td>Summary of software risks assessed by existing risk assessment models</td>
<td>34</td>
</tr>
<tr>
<td>3.1</td>
<td>Research framework</td>
<td>50</td>
</tr>
<tr>
<td>3.2</td>
<td>Synonyms for keyword</td>
<td>56</td>
</tr>
<tr>
<td>3.3</td>
<td>Form used for recording data from primary studies</td>
<td>58</td>
</tr>
<tr>
<td>3.4</td>
<td>Modified form used for recording data from primary studies</td>
<td>59</td>
</tr>
<tr>
<td>3.5</td>
<td>Study quality assessment checklist</td>
<td>61</td>
</tr>
<tr>
<td>3.6</td>
<td>Scoring for questions in quality assessment checklist</td>
<td>61</td>
</tr>
<tr>
<td>3.7</td>
<td>An example of data extracted from a study in IEEE</td>
<td>63</td>
</tr>
<tr>
<td>3.8</td>
<td>Example of usability risk factors identified from usability attributes and sub-attributes</td>
<td>66</td>
</tr>
<tr>
<td>3.9</td>
<td>Questions to identify and characterize the target audience</td>
<td>67</td>
</tr>
<tr>
<td>3.10</td>
<td>Question to design the questionnaire</td>
<td>69</td>
</tr>
<tr>
<td>3.11</td>
<td>Example of data coding, constant comparison, and memoing implementation</td>
<td>73</td>
</tr>
<tr>
<td>3.12</td>
<td>Identification of usability risk from usability risk factors</td>
<td>74</td>
</tr>
<tr>
<td>3.13</td>
<td>Categories of usability vulnerabilities</td>
<td>74</td>
</tr>
<tr>
<td>Section</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>---------</td>
<td>-----------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>3.14</td>
<td>Interpretation of Coefficient of Variation (CV)</td>
<td>84</td>
</tr>
<tr>
<td>3.15</td>
<td>Criteria for expert selection</td>
<td>88</td>
</tr>
<tr>
<td>3.16</td>
<td>Items in Software Usability Scale (SUS) 98</td>
<td>93</td>
</tr>
<tr>
<td>3.17</td>
<td>Items in questionnaire to measure perceived usefulness, ease of use and self-predicted future usage</td>
<td>95</td>
</tr>
<tr>
<td>3.18</td>
<td>Pilot project selection criteria</td>
<td>97</td>
</tr>
<tr>
<td>3.19</td>
<td>Selected software projects based on category</td>
<td>98</td>
</tr>
<tr>
<td>3.20</td>
<td>Checklist for case study planning</td>
<td>100</td>
</tr>
<tr>
<td>4.1</td>
<td>Retrieved studies based on selection procedure</td>
<td>108</td>
</tr>
<tr>
<td>4.2</td>
<td>Retrieved studies based on source of articles</td>
<td>109</td>
</tr>
<tr>
<td>4.3</td>
<td>Focus of primary articles</td>
<td>110</td>
</tr>
<tr>
<td>4.4</td>
<td>Information recorded from retrieved articles</td>
<td>111</td>
</tr>
<tr>
<td>4.5</td>
<td>Method of analysis of software risk assessment model</td>
<td>121</td>
</tr>
<tr>
<td>5.1</td>
<td>Attributes and sub-attributes of usability</td>
<td>129</td>
</tr>
<tr>
<td>5.2</td>
<td>Usability risk factors</td>
<td>130</td>
</tr>
<tr>
<td>5.3</td>
<td>Cronbach’s Alpha value for constructs in questionnaire</td>
<td>132</td>
</tr>
<tr>
<td>5.4</td>
<td>Distribution of respondents based on ministries</td>
<td>135</td>
</tr>
<tr>
<td>5.5</td>
<td>Distribution of respondents based on age</td>
<td>135</td>
</tr>
<tr>
<td>5.6</td>
<td>Distribution of respondents based on service scheme</td>
<td>136</td>
</tr>
<tr>
<td>5.7</td>
<td>Distribution of respondents based on experience</td>
<td>137</td>
</tr>
<tr>
<td>5.8</td>
<td>Distribution of respondents based on knowledge on usability</td>
<td>137</td>
</tr>
<tr>
<td>5.9</td>
<td>Agreement on occurrence of usability risk factors</td>
<td>138</td>
</tr>
<tr>
<td>5.10</td>
<td>Mapping of respondents’ opinion on possible usability risk factors</td>
<td>143</td>
</tr>
<tr>
<td>5.11</td>
<td>Finalised list of usability risk factors</td>
<td>148</td>
</tr>
<tr>
<td>5.12</td>
<td>Potential usability risk identified from usability risk factors</td>
<td>149</td>
</tr>
<tr>
<td>5.13</td>
<td>Details of experts involved in expert review</td>
<td>153</td>
</tr>
<tr>
<td>5.14</td>
<td>List of experts’ recommendations and action taken after expert review</td>
<td>154</td>
</tr>
<tr>
<td>6.1</td>
<td>Three phases of Delphi study</td>
<td>163</td>
</tr>
<tr>
<td>6.2</td>
<td>Details of experts involved in Delphi study</td>
<td>165</td>
</tr>
</tbody>
</table>
6.3 Feedback and action taken after Phase 1 Pilot Study 167
6.4 Feedback and action taken after Phase 2 Pilot study 168
6.5 Suggestion on new usability risk 170
6.6 Constant Comparison and Memoing of suggested usability risk 171
6.7 Initial probability of occurrences of usability risks 173
6.8 Probability of occurrences of usability risks 175
6.9 Initial impact of usability risks on SDLC phases 177
6.10 Impact of usability risks to SDLC phases 179
6.11 CV for each SDLC phases 181
6.12 Finalized mitigation plans for usability risks 184
6.13 Risk Level Matrix 194
6.14 Risk exposure level of usability risks 195
6.15 Usability risk grouped based on its probabilities of occurrences 197
6.16 Impact of usability risks based on SDLC phases 199
6.17 Usability risks with ‘very high’ impact 201
6.18 Prioritization and classification of usability risks on SDLC 202
7.1 Classification of usability risk in SDLC phases 214
8.1 Unit of analysis in case study 232
8.2 Usability of software 235
8.3 Possible usability vulnerabilities of Software Project A 236
8.4 Potential usability risks of Software Project A 237
8.5 Possible usability vulnerabilities of Software Project B 238
8.6 Potential usability risks of Software Project B 239
8.7 Possible usability vulnerabilities of Software Project C 241
8.8 Potential usability risks of Software Project C 242
8.9 Possible usability vulnerabilities of Software Project D 243
8.10 Potential usability risks of Software Project D 245
8.11 Number of usability vulnerabilities and potential usability risk identified software projects 246
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.12</td>
<td>Cronbach’s Alpha value for <em>Perceive Usefulness</em> and <em>Ease of Use</em></td>
<td>246</td>
</tr>
<tr>
<td>8.13</td>
<td>Constructs of <em>Perceived Usefulness</em></td>
<td>247</td>
</tr>
<tr>
<td>8.14</td>
<td>Constructs of <em>ease of use</em></td>
<td>248</td>
</tr>
<tr>
<td>8.15</td>
<td>Correlation between <em>perceived usefulness, ease of use</em>, and <em>self-predicted future usage</em></td>
<td>250</td>
</tr>
<tr>
<td>8.16</td>
<td>Usability and number of potential usability risk of software projects</td>
<td>250</td>
</tr>
</tbody>
</table>
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>FIGURE NO.</th>
<th>TITLE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Development of problem statement</td>
<td>6</td>
</tr>
<tr>
<td>2.1</td>
<td>Quality as defined in <em>ISO/IEC 25010</em></td>
<td>15</td>
</tr>
<tr>
<td>2.2</td>
<td>Boehm’s Risk Management Model (Win-Win)</td>
<td>26</td>
</tr>
<tr>
<td>2.3</td>
<td>SEI’s Software Risk Management Model 30</td>
<td>27</td>
</tr>
<tr>
<td>2.4</td>
<td>Riskit Methodology 30</td>
<td>27</td>
</tr>
<tr>
<td>2.5</td>
<td>Software risk management processes</td>
<td>29</td>
</tr>
<tr>
<td>2.6</td>
<td>Overview of SLR steps 43</td>
<td>40</td>
</tr>
<tr>
<td>2.7</td>
<td>Three phases in data coding</td>
<td>41</td>
</tr>
<tr>
<td>2.8</td>
<td>Steps to conduct expert review</td>
<td>44</td>
</tr>
<tr>
<td>2.9</td>
<td>Seven-stage processes in survey</td>
<td>45</td>
</tr>
<tr>
<td>2.10</td>
<td>Standards, guidelines and models used in model development</td>
<td>48</td>
</tr>
<tr>
<td>3.1</td>
<td>Research phases</td>
<td>52</td>
</tr>
<tr>
<td>3.2</td>
<td>Research process</td>
<td>53</td>
</tr>
<tr>
<td>3.3</td>
<td>Selection process of primary studies</td>
<td>60</td>
</tr>
<tr>
<td>3.4</td>
<td>The hierarchical structure of usability risk factors, usability risks and usability vulnerabilities</td>
<td>65</td>
</tr>
<tr>
<td>3.5</td>
<td>Analysing survey results</td>
<td>70</td>
</tr>
<tr>
<td></td>
<td>Implementation of Grounded Theory to analyse respondent’s suggestion on possible usability risk factors</td>
<td>72</td>
</tr>
<tr>
<td>3.6</td>
<td>Format used to describe usability risk</td>
<td>75</td>
</tr>
<tr>
<td>3.7</td>
<td>First snap shot of the statements taken from Kothari (2010)</td>
<td>75</td>
</tr>
<tr>
<td>Section</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>-----------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>3.9</td>
<td>Second snap shot of the statements taken from Kothari (2010)</td>
<td></td>
</tr>
<tr>
<td>3.10</td>
<td>Delphi method adopted in this research</td>
<td></td>
</tr>
<tr>
<td>3.11</td>
<td>Pilot studies that pilots all rounds before the actual Delphi Process</td>
<td></td>
</tr>
<tr>
<td>3.12</td>
<td>Pilot studies that pilots each round separately</td>
<td></td>
</tr>
<tr>
<td>3.13</td>
<td>Expert selection process</td>
<td></td>
</tr>
<tr>
<td>3.14</td>
<td>Guidelines to conduct case study (Kitchenham, Pickard, &amp; Pfleeger, 1995)</td>
<td></td>
</tr>
<tr>
<td>3.15</td>
<td>Example of email sent to the experts</td>
<td></td>
</tr>
<tr>
<td>4.1</td>
<td>Distribution of articles according to publication year</td>
<td></td>
</tr>
<tr>
<td>4.2</td>
<td>Elements in software risk assessment models</td>
<td></td>
</tr>
<tr>
<td>7.1</td>
<td>Element and methods in Usability Risk Assessment Model</td>
<td></td>
</tr>
<tr>
<td>7.2</td>
<td>Conceptual model of usability risk assessment</td>
<td></td>
</tr>
<tr>
<td>7.3</td>
<td>Prioritization of usability risks based on SDLC phases</td>
<td></td>
</tr>
<tr>
<td>7.4</td>
<td>The architecture of Usability Risk Assessment Tool</td>
<td></td>
</tr>
<tr>
<td>7.5</td>
<td>Login page to access Usability Risk Assessment Tool</td>
<td></td>
</tr>
<tr>
<td>7.6</td>
<td>Main page of Usability Risk Assessment Tool</td>
<td></td>
</tr>
<tr>
<td>7.7</td>
<td>Screen shots of interface showing registered software projects</td>
<td></td>
</tr>
<tr>
<td>7.8</td>
<td>Screen shots of interface to add new software projects</td>
<td></td>
</tr>
<tr>
<td>7.9</td>
<td>Screen shot of interface to insert new usability risk</td>
<td></td>
</tr>
<tr>
<td>7.10</td>
<td>Screen shot of interface to identify usability vulnerabilities</td>
<td></td>
</tr>
<tr>
<td>7.11</td>
<td>Screen shot of interface to identify usability risk</td>
<td></td>
</tr>
<tr>
<td>7.12</td>
<td>Screen shot of interface that classify usability risk based on SDLC phases</td>
<td></td>
</tr>
<tr>
<td>7.13</td>
<td>Screen shot of interface that prioritize usability risk within a SDLC phase</td>
<td></td>
</tr>
<tr>
<td>7.14</td>
<td>Screen shot of interface that provides mitigation plans for the selected usability risk</td>
<td></td>
</tr>
<tr>
<td>7.15</td>
<td>Classes with its attributes and functions</td>
<td></td>
</tr>
</tbody>
</table>
8.1 Null hypothesis in our case study

232

8.2 Graph of number of usability risk vs. usability

251
# LIST OF ABBREVIATIONS

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Full Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACM</td>
<td>Association for Computing Machinery</td>
</tr>
<tr>
<td>ADO.Net</td>
<td>ActiveX Data Objects</td>
</tr>
<tr>
<td>ASP.Net</td>
<td>Active Server Pages .Net</td>
</tr>
<tr>
<td>CV</td>
<td>Coefficient of Variation</td>
</tr>
<tr>
<td>ICT</td>
<td>Information Communication and Technology</td>
</tr>
<tr>
<td>IEEE</td>
<td>Institute of Electrical and Electronics Engineers</td>
</tr>
<tr>
<td>IIS</td>
<td>Internet Information Server</td>
</tr>
<tr>
<td>MAMPU</td>
<td>Malaysian Administrative Modernization and Management Planning Unit</td>
</tr>
<tr>
<td>REBOK</td>
<td>Requirement Engineering Body Of Knowledge</td>
</tr>
<tr>
<td>SDLC</td>
<td>Software Development Life Cycle</td>
</tr>
<tr>
<td>SEI</td>
<td>Software Engineering Institute</td>
</tr>
<tr>
<td>SLR</td>
<td>Systematic Literature Review</td>
</tr>
<tr>
<td>SPSS</td>
<td>Statistical Package for Social Science</td>
</tr>
<tr>
<td>SQL</td>
<td>Structured Query Language</td>
</tr>
<tr>
<td>SRS</td>
<td>Software Requirement Specification</td>
</tr>
<tr>
<td>SWEBOK</td>
<td>Software Engineering Body Of Knowledge</td>
</tr>
<tr>
<td>WWW</td>
<td>World Wide Web</td>
</tr>
<tr>
<td>APPENDIX</td>
<td>TITLE</td>
</tr>
<tr>
<td>----------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>A</td>
<td>Quality Assessment Scores for Selected Articles</td>
</tr>
<tr>
<td>B</td>
<td>Complete List of Usability Risk Factors</td>
</tr>
<tr>
<td>C</td>
<td>Questionnaire to Identify Usability Risk Factors</td>
</tr>
<tr>
<td>D</td>
<td>System Usability Scale (SUS) Questionnaire</td>
</tr>
<tr>
<td>E</td>
<td>Software Application Details Form (for Case Study)</td>
</tr>
<tr>
<td>F</td>
<td>Finalize list of usability risk and its descriptions, usability vulnerabilities with its descriptions, and category of usability vulnerabilities after expert review</td>
</tr>
<tr>
<td>G</td>
<td>Selection of Experts to participate in Delphi Method</td>
</tr>
<tr>
<td>H</td>
<td>Questionnaire for Phase 1 Delphi Study</td>
</tr>
<tr>
<td>I</td>
<td>Questionnaire for Phase 2 Delphi Study</td>
</tr>
<tr>
<td>J</td>
<td>Questionnaire for Phase 3 Delphi Study</td>
</tr>
<tr>
<td>K</td>
<td>Functional Testing</td>
</tr>
</tbody>
</table>
CHAPTER 1

INTRODUCTION

This chapter presents introduction on development of a usability risk assessment model as a mechanism to reduce usability problems in software. The first section explains the background of the research problem, followed by the problem statement, research questions, objectives, and scope of the research. The final section presents the significance of this research, and provides a brief description on key terms applied throughout the thesis.

1.1 Background of the Problem

The demand for faster, larger-scale software with better performance has increased during the past couple of years. However, the dependency on software has created great concern and criticism on quality problems in software currently being used. Meeting users’ expectations for quality software has been a tedious task for software developers (Okonta, Ojugo, Wemembu, & Ajani, 2013). This is attributable to unanticipated problems such as missing the deadlines, poorly defined software requirements, budgets being overrun and failing to deliver the expected business value (i.e., ROI) (Al-Ahmad et al., 2009; R. Kaur & Sengupta, 2013). Consequently, the existence of quality problems in software has led to many software failures (R. Kaur & Sengupta, 2013; Verner, Sampson, & Cerpa, 2008), in line with reports by Standish Group (2011), which demonstrates that only 37% of software projects developed between 2002 and 2010 were successful (Schwaber & Sutherland, 2012)
Users’ expectations for quality software are related to product quality (characteristics of software) and quality in use (the interaction of different users with software to meet their needs). In order to fulfill users’ expectations, *Usability* appears as a highly relevant quality attribute in achieving product quality and quality in the use of software, since usability can only be measured when the software are being used. The ISO/IEC 25010 (2011) standard defined usability as “*the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.*” In addition, studies on existing quality models (Dromey, 1995; ISO/IEC25010, 2011; McCall, Richards, & Walters, 1977) also recognized usability as an important quality factor, and it has always been stated even in the first model of software quality, referred to as *McCall Factors Criteria Metrics (FCM)* (McCall, 1977).

However, the inability of a development team to establish this factor causes software to be plagued by usability problems that have negative impact on the overall quality of the software (Farooq & Quadri, 2012; Hua & Gong, 2013). One of the main causes behind usability problems is ignorance and unawareness about existing usability standard (S.K. Dubey & Rana, 2010). Furthermore, product design with less usability activities, inadequate usability skills and knowledge, unawareness usability engineering life-cycle activities, and inappropriately applying usability methods have led to many usability problems in software (Jokela, 2005). Examples of usability problems are increase in broken links, short of interactive features and accessibility features and slower accessibility speed (Isa, Suhami, Safie, & Semsudin, 2011). Moreover, usability problems underlying operating process, interface design, and product structure have resulted in lower efficiency, effectiveness, and difficulty of use for end users (Liang, Deng, & Wang, 2009). Usability problems in software influence the overall usage level of the software (Inversini, Cantoni, & Bolchini, 2011), reaffirmed in a 2011 survey report by the Malaysian Administrative Modernization and Management Planning Unit (MAMPU), which reported that the usage of online services by the Malaysian government was merely 40%, suggesting higher usability problems in online services (MAMPU, 2011). Any effort to improve software usability after its development is not recommended, since it only increases costs, and involves, to a certain degree, in the remaking of the product (Sharma, Kalia, & Singh, 2012).
Generally, the existence of usability problems could lead to software failure; by reducing these problems, more usable and quality software could be produced (Ahmed Seffah, Donyae, Kline, & Padda, 2006).

Numerous approaches to decrease usability problems in software have been suggested in past studies in the related literature. Usability evaluation activities such as empirical testing, inspection, and metrics for usability standards were used to evaluate the usability of software (Humayoun, 2012). This only evaluates a completed system, and does not intervene in earlier stages of the development process (Lilja, Laakso, & Palomki, 2011). Alternatively, Den Ouden (2006) discovered that most problems related to software products can be traced back to design decisions made throughout the development process. As the quality of a product is greatly influenced by the quality of the process used to develop it, it is critical to tightly integrate usability with the Software Development Life Cycle (SDLC) to develop software with fewer usability problems (Lindgaard, 1994).

A great deal of effort has been taken to incorporate formal usability standards, processes, techniques and practices into SDLC in order to improve interaction and software quality (Durrani & Qureshi, 2012; Ferre, 2003; Fischer, 2012; Heiskari, Kauppinen, Runonen, & Mannisto, 2009). However, usability practices are only integrated in certain phases of the software development process, e.g., the requirements and design phase (Carlshamre & Rantzer, 2001; A. Seffah, Djouab, & Antunes, 2001). In fact, practical implementation of this integration is lacking (Durrani & Qureshi, 2012). Furthermore, in most software development projects, usability requirements are not part of the software requirements, causing incomplete, confusing, and contradictory requirements for developers, which have resulted in difficulties for development teams to avoid usability problems (Heiskari et al., 2009). Meanwhile, ISO 13407 (1999), an international standard, proposed a framework to integrate usability in all software development phases with use of a User-Centred Design (UCD) approach. Even so, usability problems still seem to reoccur in software. This reoccurrence is possibly attributable to the presence of usability risks in all software development phases, each of which needs to be addressed independently.
Generally, a *usability risk* can be understood as the potential action or activity that leads to undesirable outcomes that impact the usability of software. Unmanaged usability risks affect the software development process, and in turn lead to various usability problems in end products. However, this term is not widely used in the literature. The majority of prior work uses the term *usability problems* and not *usability risks*. The term *usability risk* was first introduced in the e-commerce and World Wide Web services context (Platt, 1999). Some studies on the sources and consequences of risks related to mobile applications used this term in their research (Dey & Häkkilä, 2008; Jin, Ko, Mun, & Ji, 2007; Ketola, 2002). Further investigation also revealed that knowledge on usability risks is still lacking, whereby there is a lack of effort in identifying and unifying usability risks as a unique entity, either in the form of checklists, models or others. Consequently, inadequate knowledge on usability risks has caused constraints in managing these risks during the software development process, which hinders the effort to reduce usability problems associated with the end product. Furthermore, the approach of usability risk management can only be entirely implemented once adequate knowledge on the associated risks is acquired. Since knowledge on usability risks is still lacking, there has been great ignorance in prior work in managing usability risks compared to other risks such as those of market, technology and money (Platt, 1999). Thus, there is a need to perform risk assessment prior to risk control in order to gain adequate knowledge on these risks (Boehm, 1991).

Usability risk assessment is defined as “a systematic process to identify, analyse and prioritise usability risks that can affect the achievement of project objectives in the aspect of usability”. Existing software risk assessment models were found to assess various risks in various domains such as project risks (Bazaz, Gupta, PrakashRishi, & Sharma, 2012), security risks (Mkpong-Ruffin, Umphress, Hamilton, & Gilbert, 2007), technical risks (Loutchkina, Jain, Nguyen, & Nesterov, 2013), managerial risks (Yan-qiu, Chi, & Ying, 2012), and cultural risks (Wattanapokasin & Rivepiboon, 2009). Most existing software risk assessment models focus on project risks and process risks, not on risk based on the quality attributes. Only Mofleh and Zahary (2011) proposed a framework called SPRMQ (Software Risk Management based on Quality attributes and Operational Life Cycle), which manages risk based on quality attributes such as Functionality, Reliability, Efficiency, Performance, and
Maintainability. Usability was not taken into consideration in this model. Since there is lack in models used to assess usability risks, there is a need for the development of a risk assessment model based on usability risks, which aims to reduce usability problems in software.

1.2 Problem Statement

Development teams currently face challenges in assessing potential usability risks during the SDLC for several reasons. Firstly, inadequate knowledge on usability risks has become a hindrance for development teams in reducing software usability problems. Subsequently, knowledge about usability risk factors and vulnerabilities that can further explain this concept is also lacking. Many studies have focused on risks underlying projects and processes meanwhile only a limited number of studies focused on risks underlying quality attributes. Overall, studies that focus on usability risks are still lacking. Secondly, the lack of knowledge on usability risks has created constraints in assessing them during the software development process. Without proper assessment, a risk control approach cannot be effectively implemented. In terms of risk assessment, there is a lack of models that assess usability risks during the SDLC. If development teams continue to develop software without assessing usability risks, the chances of producing software with more usability problems are higher.

Hence, we propose to develop a Usability Risk Assessment Model to assess usability risks during the SDLC. With usability-based risk assessment model, usability risks are identified, analysed and prioritized on software development phases by which the existence of usability risks during software development process could be made evident and better understood by the development team. Little effort in past studies in identifying, analysing and prioritizing potential usability risks in software development phases has provided evidence that a usability based risk assessment model is certainly needed in order to assist development teams to produce software with less usability problems, and in turn fulfil the expectation of end users. This can improve software quality and reduces the risk of software failure.
Usability Risk Assessment Model

**Existing Problem**
- Affects users’ expectation and acceptance of software
- Low usage of software
- Poor quality software
- Presence of usability risk in SDLC phases
- Usability risk has not been identified and unified as an entity
- Inadequate knowledge on usability risks has created constraints in their management
- Great ignorance on managing usability risks
- Lack of models that assess risk based on quality attributes
- Lack of models that assess usability risks

**Aim of study**
- Improving users’ expectations and acceptance of software
- Increasing usage of software
- Improving quality software
- Reducing presence of usability risks in SDLC phases
- Identifying and unifying usability risk as an entity
- Increasing knowledge on usability risks by identifying usability risk factors and usability vulnerabilities
- Reducing ignorance in managing usability risks
- Creating a model that assess risk based on quality attribute
- Developing a model that assess usability risks

**Figure 1.1 Development of problem statement**
1.3 Research Questions (RQ)

This research aims to assess potential usability risks during the software development process by using the proposed Usability Risk Assessment Model. This research is expected to help development teams in identifying, analysing and prioritizing potential usability risks during the software development process to reduce usability problem. Based on the specified research problem, three research questions have been formulated, which are as follows:

RQ 1: What are the elements and methods in the Usability Risk Assessment Model?
This question answers the concern on the elements that constitutes in the proposed model and the methods to implement these elements.

RQ 2: How potential usability risks can be incorporated into SDLC phases?
This question answers the concern on how usability risks can be incorporated into SDLC phases.

RQ 3: How a suitable Usability Risk Assessment Model can be proposed based on the above findings?
This question answers the concern on how the findings from RQ1 and RQ2 can be used to develop the proposed model.

1.4 Research Objectives (RO)

The objectives of this research have been formulated based on the five research questions as follows:

RO 1: To investigate elements and methods in the Usability Risk Assessment Model
This objective determines the elements and method used in the proposed model.
RO 2: To identify potential usability risks that influences the usability of software
This objective determines the potential usability risks which effects software usability.

RO 3: To analyse and prioritize potential usability risks in SDLC phases
This objective analyses and prioritises the identified usability risks (from RO 2) in order to incorporate usability risks in SDLC phases.

RO 4: To proposed aUsability Risk Assessment Model
This objective aims to develop the proposed model using the findings from RO 1, RO 2 and RO 3.

RO 5: To evaluate the proposed Usability Risk Assessment Model
This objective evaluates the developed model (from RO 4) to ensure it achieves aim of this research

1.5 Scope of the Study

This section presents the limitations of this study, which mainly includes the software quality and risk assessment approach. A further description on the software quality and risk assessment approach is provided as follows.

- Software quality scope
This study emphasizes the concept of software quality as explained in the quality standard: ISO/IEC 25010: 2011 Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) -- System and software quality models. Additionally, this study focuses on usability as one of the important attributes that contribute to software quality. This study aims to improve process quality by integrating usability risks into SDLC, which directly contributes to an improvement in product quality, and subsequently improves software quality in use.
Risk assessment approach scope

Although software risk management processes involve risk assessment and risk control, this study only focuses on the risk assessment process. This is because current knowledge on usability risks is still lacking, and it is difficult to control a risk without conducting a proper assessment process. The advantage of a risk assessment process is that it identifies, analyses, and prioritizes usability risks in SDLC phases, which could potentially increase the knowledge and awareness of development teams to develop more usable software. This study proposed a usability based risk assessment model that involves three core elements: Risk Identification, Risk Analysis and Risk Prioritization. However, the elements of Risk Classification and Risk Mitigation were included in this model to add value to the proposed model.

Since usability problem is also a common problem faced in Malaysian Public Sector, respondents and experts involved in this study were selected from Malaysian Public Sector. Furthermore, possible usability risks during software development process vary among private and public sector. Thus, focus of this study remains on potential usability risks in software development process at public sector.

1.6 Significance of the Study

Achieving the objectives of this study provides theoretical and practical significance.

Theoretical significance

The main aim of this study is to provide a usability based assessment model. This model focuses on usability as a quality factor that integrates well with the software development process. The proposed model contributes to the area of knowledge in the Software Engineering Body of Knowledge (SWEBOK) (P.Bourque and R.E. Fairley, 2014), particularly in Software Quality (Section 1.3-Models and Quality Characteristics and Subsection 1.3.1/1.3.2-Software
Process Quality/Software Product Quality), Software Engineering Management (Section 3-Software Project Planning and Subsection 2.5-Risk Management), and Software Engineering Process (Section 2.2-Software Life Cycle Models). Since this study focuses on usability, it highlights knowledge on usability risks, usability risk factors and usability vulnerabilities.

- **Practical significance**
  
  The proposed model could aid software development teams, including project managers, quality managers and risk management teams to conduct usability risk assessment processes that identify, analyse, and prioritise potential usability risks that impact SDLC phases. This assessment process influences the development of quality software. Organisations could meet users’ expectations on quality by ensuring the identified usability risks are handled well in each phase during the SDLC. Knowledge and understanding on potential usability risks could also proactively facilitate the development of usability requirements for software, and support the evaluation of software implementation against its requirements.

1.7 **Glossary**

This section explains several key terms that have been used throughout this thesis. A detailed discussion for each term is provided in Chapter 2.

(a) **Usability problem** is perceived as an aspect of the system and/or a demand on the user which makes it unpleasant, inefficient, onerous, perturbing or impossible for the user to achieve their goals in typical contexts of use (Lavery, 1997).

(b) **Usability risk factor** is a cause or characteristic that typically influences the possibility of a risk event occurrence (Islam, 2009).

(c) **Risk** is considered a function of the likelihood of a given threat-source’s exercising a particular potential vulnerability, and the resulting impact of that adverse event on the organization (Stoneburner et al., 2002)
(d) **Risk event** can be certain or uncertain, and can be influenced by a single occurrence or a series of occurrences. There exists a cause for the occurrence of a risk event, which is known as a risk factor (Islam, 2009).

(e) **Usability risk** is the potential that a chosen action or activity leads to a loss or an undesirable outcome which could impact usability of a software (Naik, 2013).

(f) **Usability vulnerability** is a weakness that can be accidentally triggered or intentionally exploited, which creates the potential for harm to software (Stoneburner et al., 2002).

(g) **Threat** is a potential for a person or natural event to exercise (accidentally trigger or intentionally exploit) specific vulnerabilities (Stoneburner et al., 2002).

(h) **Probability of occurrence** refers to the probability that a potential vulnerability may be exercised within the construct of the associated threat environment (Stoneburner et al., 2002).

(i) **Impact** is the amount of potential losses that an organization could suffer from a negative or harmful risk event (Stoneburner et al., 2002).

(j) **Effectiveness** is the degree to which specified users can achieve specified goals with accuracy and completeness in a specified context of use (ISO/IEC25010, 2011).

(k) **Efficiency** is the degree to which specified users expend appropriate amounts of resources in relation to the effectiveness achieved in a specified context of use use (ISO/IEC25010, 2011).

(l) **Satisfaction** is the degree to which users are satisfied in a specified context of use (ISO/IEC25010, 2011).
1.8 Outline of the Thesis

This thesis is organised in the following chapters:

**Chapter 1** presents introduction to the research, and discusses the background of the research, problem statement, goals, research questions, objectives, scope and significance of the study.

**Chapter 2** provides a comprehensive review of related studies in the existing body of literature. Initially, state-of-the-art on models, processes and tools relevant to software risk assessment and software risk management are reviewed. This chapter elaborates usability as one of the significant factors in producing quality software.

**Chapter 3** discusses the phases of the research design and methodology in detail. Explanation of the research phases includes related activities and deliverables. This chapter also discusses the research instruments and the evaluation criteria which were adopted in this work.

**Chapter 4** documents and illustrates the data collection process using Systematic Literature Review (SLR), and the adoption of an existing mapping study. Using these, elements and methods in the Usability Risk Assessment Model are identified to achieve the first objective of the research: *To investigate elements and methods in the Usability Risk Assessment Model*.

**Chapter 5** explains the processes of identifying usability risk and usability vulnerabilities, and provides respective descriptions. The process of expert validation on usability risk and its vulnerabilities are also explained.

**Chapter 6** documents and illustrates the analysis of usability risks, as well as their incorporation into the SDLC phases using a Delphi based risk analysis method. Using this method, seven experts were requested to determine the probability of occurrences, impact on SDLC phases, classification and prioritization through risk exposure levels, and mitigation plan for each of the identified usability risks.
Chapter 7 describes the conceptual model of the Usability Risk Assessment Model, and explains in detail the design of the Usability Risk Assessment Tool.

Chapter 8 presents the evaluation of the Usability Risk Assessment Model using a quantitative case study.

Chapter 9 concludes this study by providing the research summary and achievements. The contributions and limitations of this research are also presented. Finally, some suggestions for future work are provided.

1.9 Chapter Summary

This chapter described the background of the problem by explaining current usability problems in software, approaches taken to reduce usability problems and the challenges associated with the implementation of these approaches. Usability problems have been considered risk factors that contribute to usability risks. By effectively managing usability risks, more usable software can be produced. The problem statement suggests that, before usability risks are managed, they should first be identified, analysed and prioritized. This chapter also described the research questions, objectives, scope and significance. The next chapter reviews the current state-of-the-art in the related literature, specifically in the areas of usability, risk management and risk assessment.
REFERENCES


2012 International Conference on Advances in Engineering, Science and Management (ICAESM).


Brereton, Pearl, Kitchenham, Barbara A, Budgen, David, Turner, Mark, & Khalil, Mohamed. (2007). Lessons from applying the systematic literature review process within the software engineering domain. *Journal of Systems and Software*, 80(4), 571-583.


Glaser, Barney. (1967). The Discovery Of Grounded Theory: Strategies For Qualitative Research, Publisher: Al.


Khan, Khalid S, Ter Riet, Gerben, Glanville, Julie, Sowden, Amanda J, & Kleijnen, Jos. (2001). *Undertaking systematic reviews of research on effectiveness: CRD's guidance for carrying out or commissioning reviews*: NHS Centre for Reviews and Dissemination.
Kitchenham, B. (2007). Guidelines for performing systematic literature reviews in software engineering. School of Computer Science and Mathematics, Keele University and Department of Computer Science, University of Durham.


Peansupap, Vachara, & Walker, Derek HT. (2006). Information communication technology (ICT) implementation constraints: a construction industry


Proceedings of the 17th International Conference on Evaluation and Assessment in Software Engineering, Porto de Galinhas, Brazil.


