“A supporting conceptual plugin for manage bugs and change requirements in agile software development”

P U K Perera
179474P

Supervisor: Mr. Charmen Wijesiriwardhana

Dissertation submitted to the Faculty of Information Technology, University of Moratuwa, Sri Lanka for the fulfillment of the requirements of Degree of Master of Science in Information Technology.

June 2020

Declaration

I declare that this thesis is my work and has not been submitted in any form for another degree or diploma at any university or other institution of tertiary education. Information derived from the published or unpublished work of others has been acknowledged in the text and a list of references is given.

Name of Student: Miss P U K Perera Student ID – 179474P

Signature of Student: ………………………

Date: …………………………………

Supervised by

Name of Supervisor: Mr. Charmen Wijesiriwardana

Signature of Supervisor: …………………

Date:………………

Acknowledgments

First of all, I would like to thank my supervisor Mr. Charmen Wijesiriwardhana for his valuable contribution to this thesis through insightful discussions, guidance, and support throughout the process. And his expert advice and feedback on my thesis.

In addition to all of the other lectures that give their guidance and advice to improve the quality of this thesis.

I must acknowledge the contributions of the large supporting cast from the organizations involved in this research; the people who were interviewed, allowed access to their work artifacts, speculated about the challenges they were facing as well as share their views regarding the role of collaborative technologies. Without their support and co-operation, there would have been nothing to report.This thesis would not have been possible without the support of my parents. Your continuous encouragement, support

Finally hope this thesis has benefited from the help and support from several people who deserve my respect and acknowledgment. I am grateful to everyone who has been directly or indirectly involved in this process.

Table of Contents

Declaration i

Acknowledgements iii

Table of Contents vi

List of Tables... vii

List of Figures.. ix

Abstract 1

Chapter 1. Introduction 3

1.1 Research Background 6

1.2 Research Aims & Objectives 8

1.3 Thesis Structure 9

1.4 Chapter summery.. 10

Chapter 2. Background & Literature Review... 11

2.1 Literature Review 11

2.2 Limitations of earlier studies... 18

2.3 Chapter Summary 19

Chapter 3. Research Design and Methodology 20

3.1 Research Question Selection 20

3.2 Deriving Solutions... 22

3.3 Chapter summary.. 33

Chapter 4. Analysis and findings.. 34

4.1 Logical Design of the Solution... 34

4.2 Logical Functionalities 28

4.3 The method of the solution logic.. 29

4.4 Analyzing the sample data for test the system 35

4.5 Prototype the solution 45

4.6 Chapter Summary 48

Chapter 5. Implementation 49

5.1 Implementation Design Architecture 49

5.2 Technologies used for the implementation of 51

5.3 Implementation of presentation layer 54

5.4 Implementation of the Service/ Logical layer 54

5.5 Implementation of the database layer 58

5.6 Limitations and the issues related to the implementation of 59

5.6 Chapter Summary 60

Chapter 6. Conclusions 61

6.1 Motivation Revisited 61

6.2 Research limitations 65

6.3Analysis of the System advantage 68

6.4 Future work of the project.. 69

6.5 Chapter Summary 70

References: 71

Appendices 73

Appendix A Existing UI in the current JIRA tool 76

Appendix B Existing UI in the current CONFLUENCE tool. 78

Appendix C Actual UI screens 92

Appendix D Detail code on main logics 83

Appendix E SQL queries... 112

Appendix 7 Class diagram for Database 115

List of Tables

Table 2 -1 : Benefits of using an RCM model 14

Table 2-2: An evaluation of the Proposed RCM_GSD framework................................ 16

Table 2- 3: limitations of earlier studies 18

Table 4-1: Sample set of test data.. 43

List of Figures

Figure 2-1: Typical RCM process 12

Figure 2-2 : Feature tree model 17

Figure 3-1: User-based functionality 27

Figure 3-2 : functionality Context 29

Figure 3-3 : Modules of the Solution.. 30

Figure 3-4: Core Component hierarchical structure 32

Figure 4-1 : Logical design of the solution.. 34

Figure 4-2 : How to derived requirement into components 36

Figure 4-3 : Derived issue 01 for component levels... 37

Figure 4-4 : Derived issue 02 for component levels…..……………………………… 43

Figure 4-5 : Solution logic diagram ………………………………………………… 45

Figure 4-6: How to define Bug and CR ………..………………………..…………… 48

Figure 4-7: Prototyping development model …….……..……………………………... 65

Figure 5-1: Three-layered architecture………..………………………………………. 49

Figure 5-2: Presentation tier…….……………………………………………………….. 51

Figure 5-3: service tier ……………………………. …..………………………………54

Figure 5 -4: Data-tier... ………………………...…………………………………….54

Figure 7 -1 :Jira Create project UI 01……… ……..……………………………………… 76

Figure 7-2 :Jira Create project UI 02…...……………….…………………………..76

Figure 7 -3 Jira Create project UI 03 ………….…………………………………………… 77

Figure 7 -4: Jira Create project UI 04………….…. ..………………………………… 77

Figure 7- 5:Jira Create tasks, test, bugs UI …….……………………………………… 78

Figure 7- 5:Jira Manage permission UI……………. …..……………………………78

Figure 7- 6 : Jira Create roles 79

Figure 7 -7 :Confluence Manage permissions.. 79

Figure 7- 8 :Confluence Web page UI of requirement details…… ………………….. 8 0

Figure 7-9 :Confluence UI for uploaded documents..……………..……..…………80

Figure 7-10: Client login page... 81

Figure 7-12 :Select interface 82

Figure 7-13: Select Component type 82

Figure 7 -14 :Select the main component 83

Figure 7 -15 :Select the subcomponent…………………………………………83

Figure 7- 16 :Select the mini component 83

Figure 7- 17 :Add Project... 64

Figure 7- 18 :Create Module 84

Figure 7 -19 :Create Interface... 85

Figure 7- 20 :Select Component type 56

Figure 7 -21 :Create the Sub component.. 86

Figure 7-22 :Create the Mini component 86

Figure 7- 23 :Summary report UI ……………………………………………….87

Figure 7 -24 :BA login page... 88

Figure 7 -25 :Select Project... 89

Figure 7- 26 :Select Module... 90

Figure 7 -27 :Select Interface 90

Figure 7 -29 :Select Main Component.. 91

Figure 7- 38 :Add Requirement... 91

Figure 7 -39 :Issue allocation…..…………………………………………………91

Figure 7 - 40 :Issue detail view 92

Figure 7- 42 :Assign task 94

Abstract

This thesis reports a comprehensive investigation of the modern software industry, it is widely considered the concept of software as a service. In this concept, they always use the agile methodologies, and they get involved with customer representatives and valued their ideas for the final product. Even after completing their projects, the companies are agreed to do the changes according to user personas and the dependencies of the system environments by getting some agreements. At this point requirements of the system start evolving to address new business needs, usability study results, changing assumptions. As a result, the project scope may extend for quite several reasons. The modern world is changing very fast, and today fixed project requirements are a thing of the past. So this identified the challenges encountered in managing requirements change and investigates supports of the project management tools to simplify the identification of **Bugs and Change Requirements**