THE STUDY OF BLACK-BOX TESTING TECHNIQUE FOR COLLATERAL MANAGEMENT SYSTEM

CHUMPOL PONROD

A THEMATIC PAPER SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE (TECHNOLOGY OF INFORMATION SYSTEM MANAGEMENT) FACULTY OF GRADUATE STUDIES MAHIDOL UNIVERSITY 2014

Copyright by Mahidol University

COPYRIGHT OF MAHIDOL UNIVERSITY

Thematic Paper
entitled

THE STUDY OF BLACK-BOX TESTING TECHNIQUE FOR COLLATERAL MANAGEMENT SYSTEM

Chumpol Ponrod
.....
Mr. Chumpol Ponrod
Candidate

.....
Asst. Prof. Supaporn Kiattisin, Ph.D.
(Electrical and Computer Engineering)
Major advisor

Adisorn Leelasantitham
.....
Asst. Prof. Adisorn Leelasantitham, Ph.D.
(Electrical Engineering)
Co-advisor

B. Mahai
.....
Prof. Banchong Mahaisavariya,
M.D., Dip Thai Board of Orthopedics
Dean
Faculty of Graduate Studies
Mahidol University

.....
Asst. Prof. Supaporn Kiattisin,
Ph.D. (Electrical and Computer Engineering)
Program Director
Master of Science Program in
Technology of Information System
Management
Faculty of Engineering
Mahidol University

Copyright by Mahidol University

Thematic Paper
entitled

THE STUDY OF BLACK-BOX TESTING TECHNIQUE FOR COLLATERAL MANAGEMENT SYSTEM

was submitted to the Faculty of Graduate Studies, Mahidol University
for the degree of Master of Science
(Technology of Information System Management)
on
November 19, 2014

Chumpol Ponrod
.....
Mr. Chumpol Ponrod
Candidate

.....
Lect. Sotarat Thammaboosadee, Ph.D
(Information Technology)
Chair

.....
Asst. Prof. Supaporn Kiattisin, Ph.D.
(Electrical and Computer Engineering)
Member

Kairoek Choeychuen.
.....
Asst. Prof. Kairoek Choeychuen, Ph.D
(Electrical and Computer Engineering)
Member

Adisorn Leelasantitham
.....
Asst. Prof. Adisorn Leelasantitham, Ph.D.
(Electrical Engineering)
Member

B. Mahai
.....
Prof. Banchong Mahaisavariya,
M.D., Dip Thai Board of Orthopedics
Dean
Faculty of Graduate Studies
Mahidol University

.....
Lect. Worawit Israngkul,
M.S.(Technical Management)
Dean
Faculty of Engineering,
Mahidol University

Copyright by Mahidol University

ACKNOWLEDGEMENTS

The study of black-box testing technique for collateral management system would like completely. I thanks to my thematic advisor team, Supaporn Kiattisin, Ph.D. (Major advisor) and Adisorn Leelasantitham, Ph.D. (Co-advisor) for invaluable suggest, help and recommend throughout the course of this research

I also thank tester junior and tester manager of organization for help to teach black-box testing technique and suggest design testing process to complete.

I gratefully encouragement my parent, my friend and my co-worker for support of this research.

Chumpol Ponrod

Copyright by Mahidol University

THE STUDY OF BLACK-BOX TESTING TECHNIQUE FOR COLLATERAL MANAGEMENT SYSTEM

CHUMPOL PONROD 5336484 EGTI / M

M.Sc. (TEACHNOLOGY OF INFORMATION SYSTEM MANAGEMENT)

THEMATIC PAPER ADVISORY COMMITTEE: SUPAPORN KIATTISIN, Ph.D., ADISORN LEELASANTITHAM, Ph.D.

ABSTRACT

The study of Black-box testing technique for collateral management system ensures reliable operations and the use of proper testing techniques. Tester should have understand collateral management process of bank and testing technique for create business scenario, test case and test script that are complete all business role and function.

This research is qualitative, using three methodologies: documentary research, design testing and summary results.

The results confirm successful research because the IT manager and tester team were cooperating in consulting design testing and suggesting testing technique for the software development. Testing help to reduced manday and vulnerability for collateral management system.

KEY WORDS: BLACK-BOX TESTING TECHNIQUE / TESTING / TESTING TECHNIQUE

152 pages

Copyright by Mahidol University

ศึกษาเทคนิคการทดสอบรูปแบบ Black-box สำหรับระบบงานบริหารหลักประกันสินเชื่อ
THE STUDY OF BLACK-BOX TESTING TECHNIQUE FOR COLLATERAL MANAGEMENT SYSTEM

ชุมพล พอนรอด 5336484 EGTI / M

วท.ม. (เทคโนโลยีการจัดการระบบสารสนเทศ)

คณะกรรมการที่ปรึกษาสารนิพนธ์ : สุภาภรณ์ เกียรติสิน, Ph.D., อดิศร ลีลาสันติธรรม, Ph.D.

บทคัดย่อ

การทดสอบระบบงานโดยใช้เทคนิค Black-box สำหรับระบบบริหารจัดการหลักประกันขององค์กรนั้น เป็นกระบวนการที่จะต้องพิจารณาถึงกระบวนที่น่าเชื่อถือและวิธีการทดสอบที่เหมาะสม เมื่อมีการทดสอบระบบงานผู้ทดสอบควรทำความเข้าใจเกี่ยวกับกระบวนการบริหารจัดการหลักประกันของแต่ละธนาคาร แล้วจึงนำเทคนิคการทดสอบระบบมาใช้งาน โดยเริ่มจากการเขียน business scenario, test case และ test script ทั้งนี้ในการทดสอบระบบงานควรทำให้ครอบคลุมกับเงื่อนไขการบริหารจัดการหลักประกันและฟังก์ชั่นการทำงานของระบบงาน

จากการวิจัยครั้งนี้เป็นการวิจัยเชิงคุณภาพ โดยใช้วิธีการค้นคว้าวิจัยจากเอกสาร, ออกแบบการทดสอบระบบ และสรุปผลที่จากการศึกษาระบบงานและการทดสอบ

ผลการวิจัยพบว่า การวิจัยนี้ประสบผลสำเร็จเพราะความร่วมมือกันเป็นอย่างดีจากหัวหน้าหน่วยงานด้านสารสนเทศและทีมผู้ทดสอบระบบในให้คำปรึกษาเกี่ยวกับการออกแบบการทดสอบระบบและให้คำแนะนำเกี่ยวกับวิธีการทดสอบระบบงาน โดยการทดสอบระบบงานช่วยลดระยะเวลาการทำงาน ค่าใช้จ่ายและลดช่องโหว่บนระบบงานการบริหารจัดการหลักประกัน

152 หน้า

Copyright by Mahidol University

CONTENTS

	Page
ACKNOWLEDGEMENTS	**iii**
ABSTRACT (ENGLISH)	**iv**
ABSTRACT (THAI)	**v**
LIST OF TABLES	**x**
LIST OF FIGURES	**xiii**
CHAPTER I INTRODUCTION	**1**
1.1. Significance of the Problem	2
1.2. Objectives	3
1.3. Expected Outcomes/Benefits of Study	3
1.4. Scope	3
1.5. Duration of Study	4
1.6. Research Methodology	4
1.7. Definitions	5
CHAPTER II LITERATURE REVIEW	
2.1. Software testing	8
2.2. Introduction to testing	8
2.3. Basics of Software Testing	9
2.4. Six Types of Testing	10
2.4.1. Unit Testing	11
2.4.2. Integration testing	11
2.4.3. Functional and system testing	12
2.4.4. Acceptance testing	13
2.4.5. Regression testing	13
2.4.6. Beta testing	14
2.5. Black Box Testing Based on Requirement Analysis and Design Specifications	16
2.5.1. Requirement Analysis	16

Copyright by Mahidol University

CONTENTS (cont.)

	Page
2.5.2. Design Specification	16
2.5.3. Previous Work	17
2.5.4. Test Planning	17
2.5.5. Testing as Part of the Development Process	18
2.6. Performing Black Box Testing	18
2.6.1. Anatomy of a Test Case	19
2.6.2. Clear Descriptions	20
2.7. Strategies for Black Box Testing	22
2.7.1. Tests of Customer Requirements	23
2.7.2. Equivalence Partitioning	23
2.7.3. Boundary Value Analysis	24
2.8. Decision Table Testing	24
2.9 Acceptance Testing	25
2.10. Black Box Test Case Automation	26
2.11. Bridge between Black Box, White Box and Gray Box Testing Technique	27
2.11.1. White Box Testing Technique	27
2.11.2. Black box testing Technique	28
2.11.3. Grey box testing technique	30
CHAPTER III MATERIALS AND METHODS	
3.1. Project Sampling	34
3.1.1. Collateral Data System	34
3.1.2. Asset Classification	41
3.1.3. Provision	43
3.2. Problem of Testing	56
3.3. Design Software Testing and Apply Black Box Testing Technique	57
3.3.1. Test Planning	57

Copyright by Mahidol University

CONTENTS (cont.)

	Page
3.3.2. Test Analysis	58
3.3.3. Test Design	59
3.3.4. Test Implementation	60
3.3.5. Test Execution	60
3.3.6. Evaluating Exit Criteria and Test Reporting	61
3.3.7. Test Closure Activities	61
3.4. Evaluate result	61
3.5. Research Methodology	63
3.6. Data Collection	64
CHAPTER IV RESULTS	
4.1. Test Planning of Collateral Data System	65
4.2. Test Analysis of Collateral Data System	65
4.2.1. Test Scenario	66
4.3. Test Design	66
4.3.1. Test case	67
4.3.2. Test script	69
4.3.3. Test Summary	103
CHAPTER V DISCUSSION	
5.1. Discussion summary	102
5.2. Assessment	104
REFERENCES	**110**
APPENDICES	**111**
Appendix A	112
Appendix B	150
BIOGRAPHY	**152**

Copyright by Mahidol University

LIST OF TABLES

Table	Page
1.1. Duration of Study	4
2.1 Test Case Planning Template	19
2.2. Poor specification of Test Case	20
2.3 Preferred Specification of a Test Case	21
2.4 Decision Table	25
2.5 White Box Testing Technique Advantages and Disadvantages	27
2.6 Black Box Testing Technique Advantages and Disadvantages	29
2.7 Grey Box Testing Technique Advantages and Disadvantages (Cont.)	31
2.8 Grey Box Testing Technique Advantages and Disadvantages (Cont.)	32
2.9 Comparison between three forms of testing techniques	33
3.1 Collateral Type and Collateral Sub Category (Cont)	35
3.2 Collateral Type and Collateral Sub Category (Cont)	36
3.3 Describe Collateral Type and Primary Key (Cont.)	39
3.4 Describe Collateral Type and Primary Key (Cont.)	40
3.5 Describe Collateral Type and Primary Key (Cont.)	41
3.6 Describe asset classification	42
3.7 Provision Rate by Asset Classification (Cont.)	43
3.8 Provision Rate by Asset Classification (Cont.)	44
3.9 Set the PV % for every collateral type (Cont.)	44
3.10 Set the PV % for every collateral type (Cont.)	45
3.11 Set the PV % for every collateral type (Cont.)	46
3.12 Set the PV % for every collateral type (Cont.)	47
3.13 Set the PV % for every collateral type (Cont.)	48
3.14 Set the PV % for every collateral type (Cont.)	49
3.15 Set the PV % for every collateral type (Cont.)	50
3.16 Set the PV % for every collateral type (Cont.)	51
3.17 Set the PV % for every collateral type (Cont.)	52

Copyright by Mahidol University

LIST OF TABLES (cont.)

Table	Page
3.18 Set the PV % for every collateral type (Cont.)	53
3.19 Set the PV % for every collateral type (Cont.)	54
3.20 Set the PV % for every collateral type (Cont.)	55
3.21 Testing manday	56
3.22 Test Plan	58
3.23 Test Scenario	58
3.24 Test case example	59
3.25 Test script example	60
3.26 Evaluate result using questionnaire	62
4.1. Test Planning of Collateral Data System	65
4.2 Test Scenario	66
4.3 Collateral Management Test Case	67
4.4 Collateral Management business Case	68
4.5 Collateral Management Aging	68
4.6 Collateral Management Requirement Number	69
4.7 Menu Create Collateral Test script (Cont.)	69
4.8 Menu Create Collateral Test script (Cont.)	70
4.9 Tab General test script	71
4.10 Tab Collateral Information on Collateral management system test script (Cont.)	72
4.11 Tab Collateral Information on Collateral management system test script (Cont.)	73
4.12 Tab Collateral Owners on Collateral management system test script	74
4.13 Tab Insurance on Collateral management system test script	75
4.14 Tab Insurance and Menu Create Collateral on Collateral management system test script	76
4.15 Tab General on Collateral management system test script	77

Copyright by Mahidol University

LIST OF TABLES (cont.)

Table	Page
4.16 Tab Collateral Information on Collateral management system test script (Cont.)	78
4.17 Tab Collateral Information on Collateral management system test script (Cont.)	79
4.18 Tab Customer link on Collateral management system test script	80
4.19 Tab Account Link on Collateral management system test script	81
4.20 Tab Collateral Link on Collateral management system test script	82
4.21 Tab Provision / Reclassification by Provision Percentage = 1.00000 and C Class = 1 on Collateral management system test script (Cont.)	83
4.22 Tab Provision / Reclassification by Provision Percentage = 1.00000 and C Class = 1 on Collateral management system test script (Cont.)	84
4.23 Log in Web CSR of branch and click Tab Delinquency by Provision Percentage = 1.00000 and C Class = 1 on Collateral management system test script (Cont.)	85
4.24 Log in Web CSR of branch and click Tab Delinquency by Provision Percentage = 1.00000 and C Class = 1 on Collateral management system test script (Cont.)	86
4.25 Tab Provision / Reclassification by Provision Percentage = 2.00000 and C Class = 2 on Collateral management system test script	87
4.26 Log in Web CSR of branch and click Tab Delinquency by Provision Percentage = 2.00000 and C Class = 2on Collateral management system test script (Cont.)	88
4.27 Log in Web CSR of branch and click Tab Delinquency by Provision Percentage = 2.00000 and C Class = 2on Collateral management system test script (Cont.)	89
4.28 Tab Provision / Reclassification by Provision Percentage = 100.00000 and C Class = 3 on Collateral management system test script	90

Copyright by Mahidol University

LIST OF TABLES (cont.)

Table	Page
4.29 Tab Provision / Reclassification by Provision Percentage = 100.00000 and C Class = 3 on Collateral management system test script (Cont.)	91
4.30 Tab Provision / Reclassification by Provision Percentage = 100.00000 and C Class = 3 on Collateral management system test script (Cont.)	92
4.31 Log in Web CSR of branch and click Tab Delinquency by Provision Percentage = 100.00000 and C Class = 3 on Collateral management system test script	93
4.32 Tab Provision / Reclassification by Provision Percentage = 100.00000 and C Class = 4 on Collateral management system test script (Cont.)	94
4.33 Tab Provision / Reclassification by Provision Percentage = 100.00000 and C Class = 4 on Collateral management system test script (Cont.)	95
4.34 Log in Web CSR of branch and click Tab Delinquency by Provision Percentage = 100.00000 and C Class = 4 on Collateral management system test script	96
4.35 Log in Web CSR of branch and Input Non CRM Flag = Y on Collateral management system test script	97
4.36 Log in Web CSR of branch and Input Non CRM Flag = N on Collateral management system test script	98
4.37 Log in Web CSR of branch and click Tab Delinquency by Provision Percentage = 100.00000 and C Class = 5 on Collateral management system test script	99
4.38 Test summary (Cont.)	100
4.39 Test summary (Cont.)	101
5.1 Testing Manday summary	102
5.2 Actual Manday (After improvement)	103
5.3 Compare manday after using black-box testing technique	104
5.4 Evaluate results by senior (Cont.)	104
5.5 Evaluate results by senior (Cont.)	105

Copyright by Mahidol University

LIST OF TABLES (cont.)

Table	Page
5.6 Describe result assessment by senior (Cont.)	105
5.7 Describe result assessment by senior (Cont.)	106
5.8 Evaluate results by junior (Cont.)	106
5.9 Evaluate results by junior (Cont.)	107
5.10 Evaluate results by junior (Cont.)	107
5.11 Evaluate results by junior (Cont.)	108
5.12 Describe result assessment by junior	108
5.13 Average satisfy summary	109

Copyright by Mahidol University

LIST OF FIGURES

Figure	Page
2.1 - Level of Software Testing	15
2.2 Black Box Testing; A black-box test takes into account only the input and output of the software without regard to the internal code of the program.	19
2.3 Equivalence Classes for Player Money	24
2.4 Boundary Value Analysis. Test cases should be created for the boundaries (arrows) between equivalence classes.	24
2.5 Represent White Box Testing	27
2.6 Different Types of White Box Testing Techniques	28
2.7 Black Box Testing	29
2.8 Different Forms of Black Box Testing Techniques	29
2.9 Grey Box Testing	31
2.10 Different forms of grey box testing techniques	32
3.1 Collateral Data System Structure	34
3.2 One-to-one	37
3.3 Many-to-one	37
3.4 Many-to-many	38
3.5 Software Testing by Black Box Testing Technique	57
4.1 Loan Account Information criteria	66
4.2 Menu Create Collateral on system	70
4.3 Tab General on Collateral management system	72
4.4 Tab Collateral Information on Collateral management system	73
4.5 Tab Collateral Owners on Collateral management system	74
4.6 Add insurance	75
4.7 Show and submit create collateral	76
4.8 Tab General on Collateral management system	78
4.9 Tab Collateral Information on Collateral management system	79

Copyright by Mahidol University

LIST OF FIGURES (cont.)

Figure	Page
4.10 Show tab Collateral link on Collateral management system (Cont.)	80
4.11 Show tab Account Link on Collateral management system	81
4.12 Show Collateral Link on Collateral management system test script	83
4.13 Tab Provision / Reclassification by Provision Percentage = 1.00000 And C Class = 1 on Collateral management system	84
4.14 Log in Web CSR of branch and click Tab Delinquency by Provision Percentage = 1.00000 and C Class = 1 on Collateral management system	86
4.15 Tab Provision / Reclassification by Provision Percentage = 2.00000 and C Class = 2 on Collateral management system	88
4.16 Log in Web CSR of branch and click Tab Delinquency by Provision Percentage = 2.00000 and C Class = 2on Collateral management system	89
4.17 Tab Provision / Reclassification by Provision Percentage = 100.00000 and C Class = 3 on Collateral management system	91
4.18 Tab Provision / Reclassification by Provision Percentage = 100.00000 and C Class = 3 on Collateral management system	92
4.19 Log in Web CSR of branch and click Tab Delinquency by Provision Percentage = 100.00000 and C Class = 3 on Collateral management system	94
4.20 Tab Provision / Reclassification by Provision Percentage = 4.00000 and C Class = 4 on Collateral management system (Cont.)	95
4.21 Log in Web CSR of branch and click Tab Delinquency by Provision Percentage = 100.00000 and C Class = 4 on Collateral management system	97
4.22 Log in Web CSR of branch and click Tab Delinquency by Provision Percentage = 100.00000 and C Class = 4 on Collateral management system	100

Copyright by Mahidol University

CHAPTER I
INTRODUCTION

Loan information is vital for bank. Bank should preventive and detective data is security, confident, integrity and availability. Bank ensure necessary loan decision can be make for customer as loan information is sensitive because it has fee, collateral, interest rate on system then calculate is reliability and correct. If information is wrong, the bank may be gain impact in the form of payment of damages, negative reputation and customers is not service of bank.

Loan system is most extremely complicated of conditions. This is store customer sensitive data such as name, gender, marital status, occupation, Collateral and other. Bank develops Loan system for help Loan process in order that bank officer good service and approve loan.

Loan system relates to collateral management systems. Collateral management systems of the bank is starting the process of loan approval by enroll the Securities linked to the recording process appraised include approve loan then send to core banking system for open loan account and auto create agreements.

Collateral management system development is securing process that requirement, design, development, testing, deployment, maintenance and evaluation. System require maintain stability and ensure precise the privacy of information. Furthermore, business should be prepare plan when incident deploy to production for business can be continues and reduce effect to other system and operation.

Software development security starts to get requirement from user. Requirement should be complete and direct to operation affect design software. If requirement is frequent change, system design is vary and modify. Bank will be more pay change system while software development cost is increase. Software testing is importance. Testing should have function testing, non-function testing, confirmation testing and regression testing. Furthermore, Developer should be design test technique for system such white-box testing technique, black-box testing technique, experience-

Copyright by Mahidol University

based testing technique and other. White-box testing technique is design test condition or test case for tester to know software architecture and understand source code. There is level unit test such as statement coverage and decision coverage. Black-box testing is design test condition or test case then refers to function or feature defines identified requirement specification or design specification. There is system test and user acceptance test such as equivalence partitioning, boundary value analysis, decision tree and decision table. Experience-based testing technique is design test case by experience of Developer team such as Quality assurance tester, Developer, user and other who know defect opportunities in module such as error guessing and exploratory, accordingly there should complete test before deploy software to production. Collateral management system ensures prevent and detect information systems for less vulnerability and negative impact. There need to control system and security in order to prevent loss data to make create collateral on system.

1.1. Significance of the Problem

Testing design technique is critical to develop process because bank system has important loan data from customer, collateral information and sensitive information. Collateral management system has collateral information, contract information and insurance information on system then there are keep, record, calculate and other. Collateral information enroll collateral detail, information support Bank of Thailand calculate Basel II and IAS39 and legal owner information. Contract information record primary contract, sub contract, Borrowers information and connect to core banking system. Insurance information can search insurance and collateral gain benefit and connect information to insurance company accordingly collateral information connect to vary system such as core banking system (CBS), Loan system etc. so software development is priority. Development should find vulnerability from most testing in order that system reduce deploy problem to production and solve problem before go-live. Quality assurance ensures software according to requirement and design.

Copyright by Mahidol University

Researcher study black-box testing technique because refers to function or feature defines identify requirement specification or design specification. Tester and Quality assurance tester should have understand collateral management process of

bank and testing technique for create business scenario, test case and test script that are complete all business role and function. Weaknesses may result in testing operation vulnerability for systems. Therefore, the study of black-box testing technique for collateral management system to help security by covering all application and recommend test methodology. Help is available to ensure the effectiveness and efficiency of testing systems than reduce testing man-day.

These testing problems such as redundant testing, business scenarios are not cover scope, time delay, testing summary and increased cost. Because researcher study black-box testing technique help to improve testing process.

1.2. Objectives

1.2.1. To know root cause and problem of testing.

1.2.2. To create black-box testing guidelines for collateral management system.

1.2.3. To ensure that Quality assurance tester have knowledge, understanding and awareness about information black-box testing.

1.3. Expected Outcomes/Benefits of Study

1.3.1. Reduced vulnerability for collateral management system.

1.3.2. Application of black-box testing guidelines.

1.3.3. Knowledge, understanding and awareness of black-box testing guidelines for Quality assurance tester.

1.4. Scope

The scope of this research was aimed at studying the black-box testing for collateral management system and black-box testing guidelines then creating a testing manual for Quality assurance tester by applying software development and testing for collateral management system.

Copyright© by Mahidol University

1.5. Duration of Study

Table1.1. Duration of Study

Details	Month								
	Apr 2014	May 2014	Jun 2014	Jul 2014	Aug 2014	Sep 2014	Oct 2014	Nov 2014	Dec 2014
Feasibility of Study	←	→							
Data Collection			←	→					
Create business scenario				←	→				
Create test case and test script					←	→			
Using black-box testing							←	→	
Summary and Analysis								←	→

1.6. Research Methodology

1.6.1. To study test design technique practices by using framework and standards.

1.6.2. To study collateral management system.

1.6.3. Select and determine test design technique for collateral management system.

1.6.4. Creation of business scenario for collateral management system.

1.6.5. Creation of test case and test script for collateral management system.

1.6.6. Improvement of business scenario, test case and test script for collateral management system.

1.6.7. Using black-box testing on system.

Copyright by Mahidol University

1.6.8. Summary black-box testing on system

1.7. Definitions

1.7.1. Availability

Availability is one of the system quality goals under the stable and security.

1.7.2 Black-box testing

Method of software testing that examines the functionality of an application without peering into its internal architecture or workings.

1.7.3. Effectiveness

Information is effective if it meets the needs of the information consumer who uses the information for a specific task. If the information consumer can perform the task with the information, then the information is effective. This corresponds to the following information quality goals: appropriate amount, relevance, understandability, interpretability, objectivity.

1.7.4. Efficiency

Whereas effectiveness considers the information as a product, efficiency relates more to the process of obtaining and using information, so it aligns to the 'information as a service' view. If information that meets the needs of the information consumer is obtained and used in an easy way (i.e., it takes few resources—physical effort, cognitive effort, time, money), then the use of information is efficient. This corresponds to the following information quality goals: believability, accessibility, ease of operation, reputation.

1.7.5. Information

A data that, like other important and essential bank asset. It can exist in many forms: printed or written on paper, stored electronically, transmitted by post or electronically, shown on films, or spoken in conversation.

Copyright by Mahidol University

1.7.6. Integrity

If information has integrity, then it is free of error and complete. It corresponds to the following information quality goals: completeness, accuracy.

1.7.7. Process

Generally, a collection of practices influenced by the enterprise's policies and procedures that takes inputs from a number of sources (including other processes), manipulates the inputs and produces outputs (e.g., products, services)

1.7.8. Reliability

Reliability is often seen as a synonym of accuracy; however, it can also be said that information is reliable if it is regarded as true and credible. Compared to integrity, reliability is more subjective, more related to perception, and not just factual. It corresponds to the following information quality goals: believability, reputation, objectivity.

1.7.9. Security

The protection of information from hacking system, change system by unauthorized, or abuse and destruction or corruption of data.

1.7.10. Sensitive personal information

Sensitive personal information is customer information such as loan information, collateral information, contract information, insurance information and other that requires an extra level of protection and a higher duty of care

1.7.11. System

A collateral management system consists of five key principles organized to achieve a specified objective. The five principles are: infrastructure (facilities, equipment, and networks); software (systems, applications, and utilities); people (developers, operators, users, and managers); procedures (automated and manual); and data (transaction streams, files, databases, and tables).

Copyright by Mahidol University

1.7.12. Test case

Set of conditions or variables which tester will define that application, software system. Function is working as it was originally established for it to do

1.7.13. Test script

Set of instructions that will be performed on system for testing before deploy system to production.

1.7.14. Test plan

There is document detailing a systematic approach to testing on system and show testing workflow. Test is internal project only. If test plan is finish, there should review within development team.

Copyright by Mahidol University

CHAPTER II
LITERATURE REVIEW

2.1. Software testing

Software testing is significant technique for estimate quality of software product. It will explain the following:

- The basis of software testing is verification, validation practice and all software development lifecycle. The techniques of software testing suggest black-box testing and white-box testing.
- Black-box and white-box techniques are six type of testing.
- Method for writing fewer test cases and finding as many faults as possible.
- Using a template for writing repeatable
- Define test cases

2.2. Introduction to testing

Software testing process is analyzing a software subject to detect differences between existing, required conditions and estimate features of software item. Software testing activity should be done all development process. There is focus on verification and validation such as inspections and pair programming. Software practices will be discussed; Verification is process of evaluating system to resolve that product of a given development phase satisfy conditions imposed at the start of that phase. Verification activity includes review and testing such as Monopoly game should be verified that two players cannot own the same house.

Validation is the process of assessing a system or end of the development process to define development process to determine whichever it realize specified requirements. At the end of development validation activities are used to evaluate whether the features that have been built into the software satisfy the customer requirements and are traceable to customer requirements. For example, we validate

Copyright by Mahidol University

that when a player lands on “Free Parking,” they get all the money that was collected. Boehm has informally defined verification and validation as follows:

Software testing is helpful for understanding these concepts;

- Mistake is human error because produce incorrect result.

- Fault is wrong process or data disposal in an application or program.

- Failure is inefficiency of program operate and desire function within specified requirement.

- Error is difference between computed value or condition and the true correct value or condition.

- Specification is a document that specifies in a complete, accurate, design behavior, component of system and procedure define satisfied.

2.3. Basics of Software Testing

Software testing basics are divided into two classes, namely, white box testing and black box testing. The basic difference between the definition of white box testing and black box testing is explained below:

Black box testing is functional testing that ignores the internal mechanisms of a system and focuses on output generated for response to input and execute conditions.

White box testing is also structural testing that realizes the internal mechanisms of systems.

The classes of testing are denoted by colors depicting the opacity of the code testers. With black box testing, the software tester does not (or should not) have access to the source code itself. The code is considered a “big black box” to the tester who is unable to see inside the box. The tester knows only that information can be entered into to the black box and the black box will transmit information out. Based on the knowledge requirements, the tester knows what to expect as the black box sends out and tests to assure that the black box transmits what it is supposed to send out. Alternatively, white box testing focuses on the internal structure of the software code. The white box tester (most often the developer of the code) knows what the code looks like and writes test cases by executing methods with certain parameters. In the language of V&V, black box testing is often used for validation (Are we building the

Copyright by Mahidol University

right software?) and white box testing is often used for verification (Are we building the software right?). This chapter focuses on black box testing. All software testing is done with executable codes. Thus, it might be necessary to create scaffolding code. Scaffolding is defined as computer programs and data files built to support software development and testing but not intended to be included in the final product. Scaffolding code is code that simulates the functions of components not existing yet and allowing the program to execute. Scaffolding code involves the creation of stubs and test drivers. Stubs are modules that simulate components that have not been written yet and are formally defined as computer program statements substituting for the body of a software module that is or will be defined elsewhere. For example, a skeleton of a method might be written with just the method signature and a hard-coded but valid return value. Test drivers are defined as software modules used to involve modules being testing and frequently providing test input, controls, monitor execution and report test results. Test drivers simulate the calling components (e.g. hard-coded method calls) and perhaps the entire environment under which the component is to be tested. Another concept is mock objects. Mock objects are temporary substitutes for domain codes that emulate the real code. For example, if the program is to interface with a database the programmer might not want to wait for the database to be fully designed and created before you writing and testing a partial program. A mock object of the database can be created for the program to use temporarily. The interface of the mock object and the real object are the same. The implementation of the object matures from a dummy implementation to an actual database.

2.4. Six Types of Testing

Several types of testing should be done on a large software system. Each type of test has a “specification” that defines the correct behavior examined by the test to ensure that incorrect behavior (an observed failure) can be identified. The six types and origin of specifications (what to look at to develop tests) involved in the test type are now discussed. There are two issues to consider in these types of testing – one is the opacity of the tester’s view of the code (whether it is white or black box testing).

Copyright by Mahidol University

The other issue is scale (Is the tester examining a small bit of code or the entire system and its environment?).

2.4.1. Unit Testing

Opacity: White box testing

Specifications: Low-level design and/or code structure

Unit testing is the testing of individual hardware, or software units, or groups of related units. Using white box testing techniques, testers (usually the developers creating the code implementation) verify that the code does what it is intended to do at a very low structural level. For example, the tester writes test code that will call a method with certain parameters and ensure that the return value of this method is as expected.

Looking at the code itself, the tester might notice that there is a branch (an if-then) and might write a second test case to go down the path not executed by the first test case. When available, the tester will examine the low-level design of the code; otherwise, the tester will examine the structure of the code by looking at the code itself. Unit testing is generally done within a class or a component.

2.4.2. Integration testing

Opacity: Black- and white-box testing

Specifications: Low- and high-level designs

Integration testing is where software components, hardware components or both are combined and tested to evaluate interactions between the two [11]. Using both black and white box testing techniques, the tester (still usually the software developer) verifies that units work together when integrated into a larger code base. The fact that the components work individually does not mean the components all work together when assembled or integrated. For example, data might be lost across an interface, messages might not be properly forwarded, or interfaces might not be implemented as specified. To plan these integration test cases, testers look at high- and low-level design documents.

Copyright by Mahidol University

2.4.3. Functional and system testing

Opacity: Black-box testing

Specifications: high-level design/requirements specification

Using black box testing techniques, testers examine the high-level design and the customer requirements specifications to plan the test cases and ensure the code does what it is intended to do. Functional testing involves ensuring that the functionality specified in the requirement specification works. System testing involves putting the new program in many different environments to ensure the program works in typical customer environments with various versions and types of operating systems and/or applications. System testing is testing conducted on a complete, integrated system to evaluate the system compliance with its specified requirements. Because system testing is performed with a full system implementation and environment, several classes of testing can be done that can examine the non-functional properties of the system. It is best when function and system testing is performed from an unbiased, independent perspective (e.g. not the programmer).

- Stress testing - testing conducted to evaluate a system or component at or beyond the limits of its specification or requirements. For example, if the team is developing software to run cash registers, a non-functional requirement might state that the server can handle up to 30 cash registers simultaneously searching for prices. Stress testing might occur in a room of 30 actual cash registers running automated test transactions repeatedly for 12 hours. There also might be a few more cash registers in the test lab to see if the system can exceed its stated requirements.

- Performance testing – testing conducted to evaluate the compliance of a system or component with specified performance requirements. To continue the above example, a performance requirement might state that the price lookup must be completed in less than a second. Performance testing evaluates whether or not the system can look up prices in less than a second (even if there are 30 cash registers running simultaneously).

- Usability testing – testing conducted to evaluate the extent to which a user can learn to operate, prepare input, and interpret the output of a system or component. While stress and usability testing can be and is frequently automated, usability testing is performed by human-computer interaction specialists who observe humans interacting with the system.

Copyright by Mahidol University

2.4.4. Acceptance testing

Opacity: Black-box testing

Specifications: requirements specifications

After functional and system testing, the product is delivered to a customer and the customer runs black box acceptance tests based on functionality expectations. Acceptance testing is formal testing conducted to determine whether or not a system satisfies its acceptance criteria (the criteria for the system must satisfy to be accepted by a customer), and to enable the customer to determine whether or not to accept the system. These tests are often pre-specified by the customer and given to the test team to run before attempting to deliver the product. The customer reserves the right to refuse delivery of the software if the acceptance test cases do not pass. However, customers are not trained software testers. Customers generally do not specify a "complete" set of acceptance test cases. The test cases are no substitute for creating a set of functional/system test cases. The customer is probably very good at specifying at most one good test case for each requirement. As you will learn below, many more tests are required. Whenever possible, we customer acceptance test cases should be run to increase confidence in functionality at the customer location.

2.4.5. Regression testing

Opacity: Black- and white-box testing

Specifications: Any changed documentation, high-level design

Regression test cases are run throughout all testing cycles. Regression testing is selective retesting of a system or component to verify that modifications have not caused unintended effects and that the system or component continues to comply with its specified requirements. Regression tests are a subset of the original set of test cases. These test cases are frequently re-run after any significant changes (bug fixes or enhancements) are made to the code. The purpose of running the regression test case is to make a "spot check" to determine whether or not the new code works properly and to assure that the new code has not damaged any previously-working functionality by propagating unintended side effects. Most frequently, it is impractical to re-run all the test cases when changes are made. Since regression tests are run throughout the development cycle, white

Copyright by Mahidol University

box regression tests can be performed at the unit and integration levels with black box tests at the integration, function, system and acceptance test levels.

The following guidelines should be used when choosing a set of regression tests (also referred to as the regression test suite):

- Choose a representative sample of tests exercising all of the existing software functions;

- Choose tests focused on software components/functions that have been changed; and

- Choose additional test cases that focus on the software functions most likely to be affected by the changes. A subset of the regression test cases can be set aside as smoke tests. A smoke test is a group of test cases establishing that the system is stable and all major functionality is present and works under “normal” conditions. Smoke tests are often automated, and the selection of the test cases is broad in scope. The smoke tests might be run before deciding to proceed with further testing (Why dedicate resources to testing if the system is very unstable?). The purpose of smoke tests is to demonstrate stability, not to find bugs with the system.

2.4.6. Beta testing

Opacity: Black-box testing

Specification: None.

When an advanced partial or full version of a software package is available, the development organization can offer it free to one or more (and sometimes thousands) potential users or beta testers. These users install the software and use it as they wish with the understanding that they will report any errors revealed during usage back to the development organization. These users are usually chosen because they are experienced users of prior versions or competitive products. The advantages of running beta tests are as follows:

- Identification of unexpected errors because the beta testers use the software in unexpected ways.

- A wider population search for errors in a variety of environments (different operating systems with a variety of service releases and with a multitude of other applications running).

Copyright by Mahidol University

- Low costs because the beta testers generally get free software but are not compensated.

The disadvantages of beta testing are as follows:

- No systematic testing because each user uses the product according to preference.

- Low quality error reports because the users may not actually report errors or may report errors with insufficient details.

- Much effort is necessary to examine error reports, particularly when there are many beta testers.

Throughout all testing cycles, regression test cases are run. Regression testing is selective retesting of a system or component to verify that modifications have not caused unintended effects and that the system or component continues to comply with its specified requirements.

Testing Type	Specification	General Scope	Opacity	Who generally does it?
Unit	Low-Level Design Actual Code Structure	Small unit of code no larger than a class	White Box	Programmer who wrote code
Integration	Low-Level Design High-Level Design	Multiple classes	White Box Black Box	Programmers who wrote code
Functional	High Level Design	Whole product	Black Box	Independent tester
System	Requirements Analysis	Whole product in representative environments	Black Box	Independent tester
Acceptance	Requirements Analysis	Whole product in customer's environment	Black Box	Customer
Beta	Ad hoc	Whole product in customer's environment	Black box	Customer
Regression	Changed Documentation High-Level Design	Any of the above	Black Box White Box	Programmer(s) or independent testers

Figure 2.1 Level of Software Testing

It is best to find a fault as early in the development process as possible. When a test case fails, the symptoms of the failure are apparent and there is a necessity to find the fault that caused the failure. The further the investigator delves into the development process, the harder it is to track down the cause of the failure. If unit tests are frequently performed, a new failure is likely to be in the code just written/tested and should be reasonably easy to find. If the tester waits for system or acceptance testing, a failure could be anywhere in the system – the investigator will have to be an astute detective to find the fault at this point.

Copyright by Mahidol University

2.5. Black Box Testing Based on Requirement Analysis and Design Specifications

2.5.1. Requirement Analysis

The first step to starting a project is to gather requirements from the customers. These requirements serve as a project objective. Requirements gathered from a customer may be either explicit or implied. These implied requirements must be transformed to the stated requirements. These requirements are then analyzed in order to know whether they are feasible and achievable. The purpose of requirement analysis is to refine the customer requirements on the basis of performance, functions and constraints. Requirements can be categorized as customer requirements, derived requirements, functional requirements, performance requirements and design requirements.

Analyzed requirements form the base of the design phase during the software development process. Therefore, the post-analysis requirements must be achievable, verifiable, clear and consistent. The status of the requirements must be traceable at any time and unambiguous. These requirements are then documented properly. Thus, activities performed during requirement analysis are as follows:

- Requirement gathering
- Requirement analysis
- Requirement documentation

2.5.2. Design Specification

Software design is an iterative process through which requirements are translated into a 'blueprint' for constructing the software. All of the gathered project requirements and objectives are transformed into a project design which is then used by the developer for the coding phase. Thus, software design acts as a bridge between the requirement analysis and development phase. A good design should incorporate all of the explicit and implicit requirements of the customer. During the design phase, designers decide how the requirements can be translated into a working project. Therefore, this phase of software development deals with deciding what hardware and software will be used to build the required product, what algorithms will be used, what data structures will be used, what the flow of data between the interfaces will be, what processes will be followed during

Copyright by Mahidol University

coding and what the criteria of testing and acceptability of the product will be. All of these details are documented properly. The objective of design documentation is to provide an efficient, modular design that will reduce the system complexity and result in an easy implementation. Output from design specifications is used by the development team.

2.5.3. Previous Work

The present work is a part of our larger endeavor to develop an automated test case generator. The system generates both types of test cases via black box and white box. The generation of test cases via white box testing has already been carried out using cellular automata. The generation of test cases by not considering the internal code was accomplished using artificial life. There is a need to compare the test cases generated via artificial life and Lenten's loop with another technique. This paper proposes that very technique. The results obtained thus far are encouraging. The system that generates overall test cases has two components. The first component generates test cases on the basis of the code. The present work intends to replace the second component.

2.5.4. Test Planning

Test planning should be done throughout the development cycle, especially early in the development cycle. A test plan is a document describing the scope, approach, resources and schedule of intended test activities. It identifies test items, the features to be tested, the testing tasks, who will perform each task and any risks requiring contingency plans. An important component of the test plan is the individual test cases. A test case is a set of test inputs, execution conditions and expected results developed for a particular objective, such as exercising a particular program path or verifying compliance with a specific requirement. Write the test plan early in the development cycle when everything generally continues to go fairly smoothly and calmly. This allows you to think through a thorough set of test cases. If you wait until the end of the cycle to write and execute test cases, you might be in a very chaotic, hurried time period. Often good test cases are not written in this hurried environment and ad hoc testing takes place. With ad hoc testing, people start trying anything they can think of without any rational roadmap through the customer requirements. The tests done in this manner are not repeatable.

Copyright by Mahidol University

2.5.5. Testing as Part of the Development Process

In testing, it is essential that planning be started as soon as the necessary artifact is available. For example, as soon as customer requirement analysis has been completed, the test team should start writing black box test cases against the requirements document. By doing so this early, the testers might realize that the requirements are incomplete. The team may ask questions of the customer to clarify the requirements in order to write a specific test case. The answer to the question is also helpful to the code developer. Additionally, the tester may request (of the programmer) that the code be designed and developed to allow some automated test execution. To summarize, the earlier testing is planned at all levels, the better. It is also very important to consider test planning and test execution as iterative processes. As soon as requirement documentation is available, it is best to begin to write functional and system test cases. When requirements change, revise the test cases. As soon as some code is available, execute test cases. When code changes, run the test cases again. By knowing how many and which test cases actually run, the progress of the project can be accurately tracked. All in all, testing should be considered an iterative and essential part of the entire development process.

2.6. Performing Black Box Testing

Black box testing, also called functional testing and behavioral testing, focuses on determining whether or not a program does what it is supposed to do based on its functional requirements. Black box testing attempts to find errors in the external behavior of the code in the following categories:

- Incorrect or missing functionality
- Interface errors
- Errors in data structures used by interfaces
- Behavior or performance errors
- Initialization and termination errors.

Through this testing, we can determine whether the functions appear to work according to specifications. However, it is important to note that no amount of testing can unequivocally demonstrate the absence of errors and defects in the code. It is best if the person who plans and executes black box testing is not the programmer of

Copyright by Mahidol University

the code and knows nothing about the code structure. The programmers of the code are innately biased and likely to test whether or not the program does what they programmed it to do. There is a need for tests to make certain that the program does what the customer wants it to do. As a result, most organizations have independent testing groups to perform black box testing. These testers are not the developers and are often referred to as third-party testers. Testers should only be able to understand and specify the desired output for a given input into the program as shown in Figure 2.1.

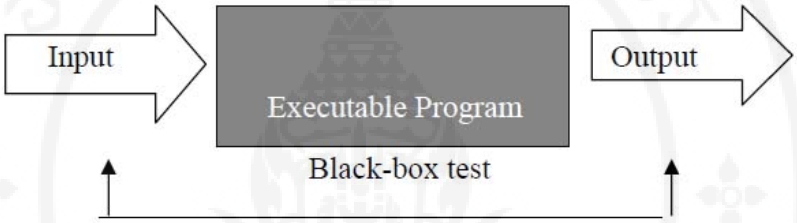


Figure 2.2 Black Box Testing; A black-box test takes into account only the input and output of the software without regard to the internal code of the program.

2.6.1. Anatomy of a Test Case

The format of your test case design is very important. We will use a particular format for our test cases as shown in Table 2.2 We recommend that the following template be used in test planning.

Table 2.1 Test Case Planning Template

Test ID	Description	Expected Results	Actual Results

First, give each test case a unique identifier. When tracking large projects, there might be a need to itemize those test cases that have not yet passed. This identifier is recorded in the first column. For example, certain statements like, “All my test cases are running except playerMovement1. I’m working on that one today,”

Copyright by Mahidol University

might need to be made. Next in the second column of the table, specifically describe the set of steps and/or input for the particular condition to be tested (including what needs to be done to prepare for the test case to be run). The third column is the expected results for an input/output oracle – what is expected to come out of the "black box" based upon the input (as described in the "description"). An oracle is any program, process or body of data that specifies the expected outcome of a set of tests as applied to a tested object. An input/output oracle is an oracle that specifies the expected output for a specified input. In the last column, the actual results are recorded after the tests are run. If a test passes, the actual results will indicate "Pass." If a test fails, it is helpful to record "Fail" and a description of the failure ("what came out") in the actual results column.

2.6.2. Clear Descriptions

It is of prime importance that the test case description be very clear and specific so that the test case execution is repeatable. Even if the same person will always execute the test cases, that person will need to pretend the test planning document is being passed on to someone else to perform the tests. Sufficiently clear instructions are needed for the other person to be able to follow the directions explicitly so the exact same test is executed every time. For example, consider a basic test case to ensure that players can move on a Monopoly board. Examples of poorly specified test cases are shown in Table 2.2

Table2.2 Poor specification of Test Case

Test ID	Description	Expected Results	Actual Results
1	Player 1 rolls dice and moves.	Player 1 moves on board.	
2	Player 2 rolls dice and moves.	Player 2 moves on board.	

The problem is that the description does not provide exact values on how many spaces the players have moved. This is an overly simplistic problem, but the program might crash for some reason when Player 1 and Player 2 land on the same

Copyright by Mahidol University

spot. If the player cannot remember what was actually rolled (let the rolls be determined randomly and do not record the rolls), the player might never be able to cause the problem to recur because the player cannot remember the circumstances leading up to the problem. Recreating the problem is essentially important in testing to ensure that identified problems can be repeated and corrected. Instead, write specific descriptions as shown in Table 2.3

Table 2.3 Preferred Specification of a Test Case

Test ID	Description	Expected Results	Actual Results
3	Precondition: Game is in test mode, SimpleGameBoard is loaded, and game begins. Number of players: 2 Money for player 1: $1200 Money for player 2: $1200 Player 1 dice roll: 3	Player 1 is located at Blue 3.	
4	Precondition: Test case 3 has successfully completed Player 2 dice roll: 3	Player 1 is located on Blue 3. Player 2 is located on Blue 3.	

A few aspects should be noted about the test cases in Table 2.3. First, notice the precondition in the description field. The precondition defines what has to happen before the test case can run properly. There may be an order of execution whereby a test case may depend upon another test case running successfully and leaving the system in a state such that the second test case can be successfully executed. For example, one test case (call it Test 1) might test whether a new user can create an ID in a system. Another test case (call it Test 2) may depend upon this new user logging in. Therefore, Test 1 must run before Test 2 can run. Additionally, if Test 1 fails, then Test 2 cannot be run yet. Alternately, perhaps Test 1 passes but Test 2 fails. Later, when the functionality is fixed, Test 1 must be re-run before the testers try to re-run Test 2. Or perhaps a database or system needs to be re-initialized before a test case can be run. Another important element to notice is the preconditions for Test Case 3 in Table 2.3. How can the test case ensure the player rolled a “three” when the value the dice rolls needs to be random in the real game? A bit of extra functionality is sometimes required to put a program in “test mode” in order to ensure that test cases can be run in a repeatable manner and so we can easily force a condition to happen.

Copyright by Mahidol University

For example, we may want to test what happens when a player lands on “Go” or on “Go to Jail” and want to force this situation to occur. The Monopoly programmers needed to create a test mode in which the dice rolls could be input manually and the amount of money each player starts with is input manually. It is also important to run some non-repeatable test cases in the regular game mode to test whether or not random dice input appears to change expected behavior. The expected results must also be written in a very specific way as in Table 2.3. The ideal program output should be recorded given a particular input/set of steps. Otherwise, how can the tester know whether or not the answer is correct (every time it is run) if the tester does not know what the answer is supposed to be? The program might perform mathematical calculations, in which cases a calculator needs to be taken out to perform some calculations by hand. Then the answers need to be put in the expected results field. What the program is supposed to do can be pre-determined ahead of time in order to instantly know whether or not the program responds properly.

2.7. Strategies for Black Box Testing

Ideally, every possible test should be run with a program. As previously stated, however, writing and executing test cases is expensive. The tester needs to make certain that test cases are definitely written for the kinds of things customers perform most frequently or even fairly often. The objective is to find as many defects as possible in as few test cases as possible. To accomplish this objective, certain strategies that will be discussed in this subsection need to be used. The goal is to avoid writing redundant test cases that will not reveal anything new (due to conditions similar to other test cases already written). Each test case should probe a different mode of failure. Another goal is to design the simplest test cases that could possibly reveal this mode of failure – test cases themselves can be error-prone if this is not kept in mind.

Copyright by Mahidol University

2.7.1 - Tests of Customer Requirements

Black box test cases are based on customer requirements and begin by looking at each customer requirement. To start, the tester needs to be certain that every single customer requirement has been tested at least once. As a result, every requirement can be traced to its test case and every test case can be traced back to its stated customer requirement. The first test case to be written for any given requirement is the most-used success path for that requirement. Success path means that some desirable functionality (something the customer wants to work) without any error conditions needs to be tested. The tester proceeds by planning more success path test cases based on other ways the customer wants to use the functionality and some test cases that execute failure paths. Intuitively, failure paths intentionally contain some kind of errors such as errors users can accidentally input. The tester needs to be certain that the program behaves predictably and gracefully in the face of these errors. Finally, the tester should plan the execution of the tests in order to ensure that the most troublesome, risky requirements are tested first. This would allow more time for fixing problems before delivering the product to the customer. It would be devastating to find a critical flaw right before the product is due to be delivered. Hence, the tester should start with one basic requirement. Many test cases can be written based on this one requirement which follows below. As previously stated, it is impossible to test every single possible combination of input. Hence, an incomplete sampling of test cases and reason will be outlined in this section.

2.7.2. Equivalence Partitioning

To minimize testing costs, the tester does not want to write several test cases testing the same aspect of a program. A good test case uncovers a different class of errors (e.g., incorrect processing of all character data) than has been uncovered by prior test cases. Equivalence partitioning is a strategy that can be used to reduce the number of test cases that need to be developed. Equivalence partitioning divides the input domain of a Testing Overview and Black-Box Testing Techniques program into classes. For each of these equivalence classes, the set of data should be treated the same by the module under test and produce the same answer. Test cases should be designed to ensure that the inputs lie within these equivalence classes. For example,

Copyright by Mahidol University

for tests of “Go to Jail” the most important thing is whether the player has enough money to pay the $50 fine. Therefore, the two equivalence classes can be partitioned as shown in Figure 2.3

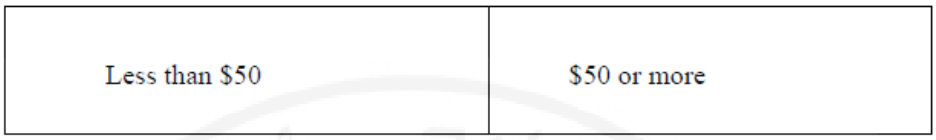


Figure 2.3 Equivalence Classes for Player Money

2.7.3. Boundary Value Analysis

Boris Beizer, well-known author of testing books advises, “Bugs lurk in corners and congregate at boundaries.” Programmers often make mistakes on the boundaries of the equivalence classes/input domain. As a result, focus needs to be placed on testing at these boundaries. This type of testing is called Boundary Value Analysis (BVA) and guides the tester to create test cases at the “edge” of the equivalence classes. Boundary value is defined as a data value that corresponds to a minimum or maximum input, internal, or output value specified for a system or component. In the above example, the boundary of the class is at 50 as shown in Figure 2.4. Test cases should be created for Player 1 at $49, $50 and $51. These test cases will help find common off-by-one errors caused by errors such as using >= when > should be used.

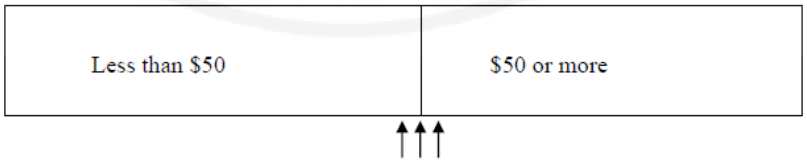


Figure 2.4 Boundary Value Analysis. Test cases should be created for the boundaries (arrows) between equivalence classes.

2.8. Decision Table Testing

Copyright by Mahidol University

Decision tables are used to record complex business rules that must be implemented in the program and, therefore, tested. A sample decision table is found in Table 2.4. In the table, the conditions represent possible input conditions. The actions

are the events that should trigger depending upon the makeup of the input conditions. Each column in the table is a unique combination of input conditions (called a rule) that results in triggering the action associated with the rule. Each rule (or column) should become a test case.

If a Player (A) lands on property owned by another player (B), A must pay rent to B. If A does not have enough money to pay B, A is out of the game.

Table 2.4 Decision Table

	Rule 1	Rule 2	Rule 3
Conditions			
A lands on B's property	Yes	Yes	No
A has enough money to pay rent	Yes	No	--
Actions			
A stays in game	Yes	No	Yes

2.9 Acceptance Testing

Acceptance test cases are written by the customer. In custom software development, contracts between the customer and the development organization frequently state that the customer can refuse to accept delivery of the product if the acceptance test cases do not run properly in the customer's own (software and hardware) environment. Sometimes the customer shares the acceptance test cases with the team, which gives them a shared specific goal. Other times, the customer hides the acceptance test cases from the developers and runs them after receiving the code (in the same way as a teacher often foregoes telling the students the test cases they will run to grade their class projects). It is believed that it is much more productive for the customer and the development team to work openly and collaboratively on the creation of the acceptance test cases. Next, the customer and development team have a similar vision of what the software needs to look like for the customer to be happy. In the researcher's experience, the collaborative acceptance test case creation serves as an excellent means of clarifying requirements by making requirements specified in a way that is quantifiable, measurable and unambiguous long before testing commences. Likewise, the aforementioned can

Copyright by Mahidol University

track the progress of system development together as the team tells the customer which acceptance test cases pass.

2.10. Black Box Test Case Automation

By nature, black box test cases are designed and run by people who do not see the inner workings of the code. Ultimately, system and acceptance cases are intended to be run through the product user interface to show that the whole product really works.

Test automation can be difficult because the developer has no knowledge of the inner workings of the software and because system and acceptance cases must be run through the UI. However, the more automated testing can be, the easier it is to run and repeat the test cases again and again. The simpler it is to run a suite of tests, the more often those tests will be run. The more the tests are run, the faster any deviation from those tests will be found. If the tester's role on the team is as a software developer, it is always good to consider the types of black box test cases (functional, system and acceptance) that will ultimately be run on the code and to automate test cases to test the logic (separate from the UI logic) behind these black box test cases. Once they have been written, automated test cases can be frequently run with minimal time investment. The following can be achieved by automating the testing of the logic behind the black box test cases:

- Assurance that the "behind the scenes" logic is working properly to ensure that inevitable black box test cases can run smoothly through the UI by the testers and customers;

- More motivation to decouple program/business logic separate from the UI logic (which is always a good design technique).

When test cases are automated, they can then become compile-able and executable documentation.

Copyright by Mahidol University

2.11. Bridge between Black Box, White Box and Gray Box Testing Technique

2.11.1. White Box Testing Technique

White Box Testing Technique: The detailed investigation of internal logic and structure of the code. In white box testing, the tester requires full knowledge of the source code.

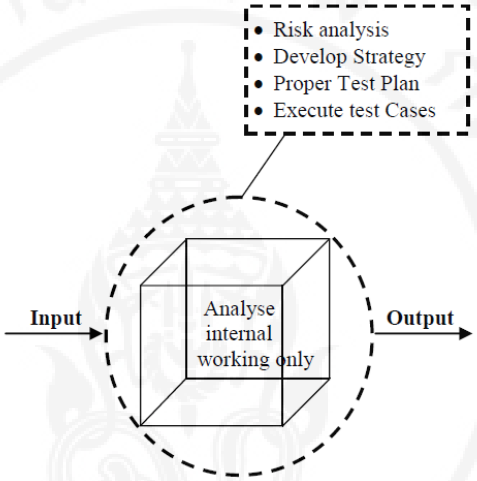


Figure 2.5 Represent White Box Testing

Some of the advantages and disadvantages of the white box testing technique are listed below:

Table 2.5 White Box Testing Technique Advantages and Disadvantages

Advantages	Disadvantages
1. Reveals error in hidden code by removing extra lines of code. 2. Side effects are beneficial. 3. Maximum coverage is attained during test scenario writing.	1. Very expensive as a skilled tester is required to perform testing. 2. Many paths will remain untested as it is very difficult to look into every nook and cranny to detect hidden errors. 3. Some of the codes omitted in the code could be missed.

Copyright by Mahidol University

Some important types of white box testing techniques are briefly described below:

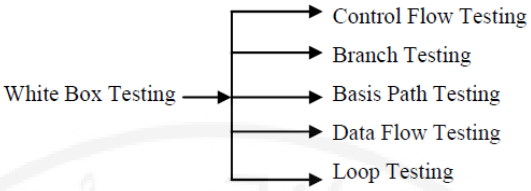


Figure 2.6 Different Types of White Box Testing Techniques

2.11.1.1. Control Flow Testing: It is a structural testing strategy that uses the program control flow as a model control flow and favors more but simpler paths over fewer but complicated paths.

2.11.1.2. Branch Testing: Branch testing has the objective of testing every option (true or false) on every control statement which also includes compound decisions.

2.11.1.3. Basis Path Testing: Basis path testing allows the test case designer to produce a logical complexity measure of procedural design and then uses this measure as an approach for outlining a basic set of execution paths.

2.11.1.4. Data Flow Testing: In this type of testing, the control flow graph is annotated with the information about how the program variables are defined and used.

2.11.1.5. Loop Testing: Focuses exclusively on the validity of the loop construct.

2.11.2. Black box testing Technique

Black box testing treats the software as a “Black Box” – without any knowledge of internal working and only examines the fundamental aspects of the system. While performing black box testing, the tester must know the system architecture and will not have access to the source code.

Copyright by Mahidol University

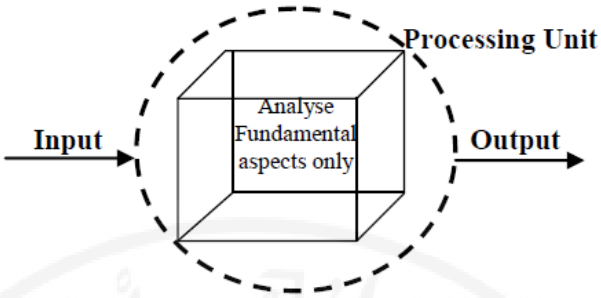


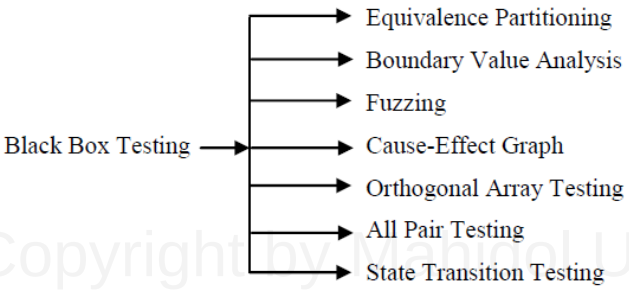
Figure 2.7 Black Box Testing

Some of the advantages and disadvantages of the black box testing technique are listed below:

Table 2.6 Black Box Testing Technique Advantages and Disadvantages

Advantages	Disadvantages
1. Efficient for large code segments. 2. Tester perception is very simple. 3. User perspectives are clearly separated from developer perspectives (programmer and tester are independent of each other). 4. Faster test case development.	1. Only a selected number of test scenarios are actually performed. As a result, there is only limited coverage. 2. Without clear specification, test cases are difficult to design. 3. Inefficient testing.

Some of the synonyms of the black box testing technique are opaque testing, functional testing, close box testing and behavioral testing. Some important types of black box testing techniques are briefly described below:



Copyright by Mahidol University

Figure 2.8 Different Forms of Black Box Testing Techniques

2.11.2.1. Equivalence Partitioning: Able to reduce the number of test cases as it divides the input data of a software unit into the partition of data from which test cases can be derived.

2.11.2.2. Boundary Value Analysis: Focuses more on testing at boundaries, or where the extreme boundary values are chosen. It includes minimum, maximum, just inside/outside boundaries, error values and typical values.

2.11.2.3. Fuzzing: Fuzz testing is used for finding implementation bugs, using malformed/semi-malformed data injection in an automated or semi-automated session.

2.11.2.4. Cause-Effect Graph: A testing technique in which testing begins by creating a graph and establishing the relation between the effect and its causes. Identity, negation, logic OR and logic AND are the four basic symbols expressing the interdependency between cause and effect.

2.11.2.5. Orthogonal Array Testing: OAT can be applied to problems in which the input domain is relatively small, but too large to accommodate exhaustive testing.

2.11.2.6. All Pair Testing: In the all pair testing technique, test cases are designed to execute all possible discrete combinations of each pair of input parameters. The main objective is to have a set of test cases covering all pairs.

2.11.2.7. State Transition Testing: This type of testing is useful for testing state machines and navigating graphical user interfaces.

2.11.3. Grey box testing technique

Grey box testing technique will increase the testing coverage by allowing the tester to focus on all layers of any complex system through the combination of all existing white box and black box testing.

Copyright by Mahidol University

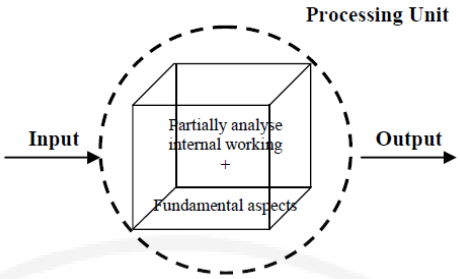


Figure 2.9 Grey Box Testing

In grey box testing, the tester requires knowledge of internal data structures and algorithms for the purpose of designing test cases. Examples of grey box testing technique include the following:

-Architectural Model

-Unified Modeling language (UML)

-State Model (Finite State Machine)

In grey box testing, the codes of two modules are studied (white box testing method) for the design of test cases. Actual tests are performed in the interfaces exposed (black box testing method).

Table 2.7 Grey Box Testing Technique Advantages and Disadvantages (Cont.)

Advantages	Disadvantages
1. Grey box testing provides the combined benefits of the white and black box testing techniques. 2. In grey box testing, the tester relies on interface definition and functional specification rather than source code. 3. In grey box testing, the tester can design excellent test scenarios.	1. Test coverage is limited as the access to the source code is not available. 2. It is difficult to associate defect identification in distributed applications. 3. Many program paths remain untested.

Copyright by Mahidol University

Table 2.8 Grey Box Testing Technique Advantages and Disadvantages (Cont.)

Advantages	Disadvantages
4. The test is done from the user's point of view rather than the designer's point of view. 5. Creates an intelligent test authoring. 6. Unbiased testing.	4. If the software designer has already run a test case, the tests can be redundant.

The other name of grey box testing is translucent testing. Different forms of grey box testing techniques are briefly described below:

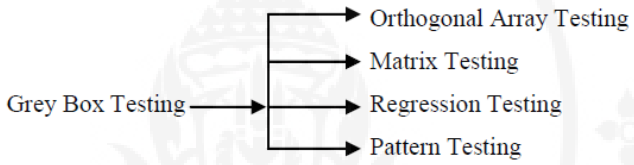


Figure 2.10 Different forms of grey box testing techniques

2.11.3.1. Orthogonal Array Testing: This type of testing is used as subset of all possible combinations.

2.11.3.2. Matrix Testing: In matrix testing, the status report of the project is stated.

2.11.3.3. Regression Testing: If new changes are made in software, regression testing implies the running of test cases.

2.11.3.4. Pattern Testing: Pattern testing verifies good application for architecture and design.

Copyright by Mahidol University

Table 2.9 Comparison between three forms of testing techniques

No.	Black Box Testing	Grey Box Testing	White Box Testing
1.	Analyses fundamental aspects only i.e. no proved edge of internal working	Partial knowledge of internal working	Full knowledge of internal working
2.	Granularity is low	Granularity is medium	Granularity is high
3.	Performed by end users and also by tester and developers (user acceptance testing)	Performed by end users and also by tester and developers (user acceptance testing)	It is performed by developers and testers
4.	Testing is based on external exceptions – internal behaviour of the program is ignored	Test design is based on high level database diagrams, data flow diagrams, internal states, knowledge of algorithm and architecture	Internal are fully known
5.	It is least exhaustive and time consuming	It is somewhere in between	Potentially most exhaustive and time consuming
6.	It can test only by trial and error method	Data domains and internal boundaries can be tested and over flow, if known	Test better: data domains and internal boundaries
7.	Not suited for algorithm testing	Not suited for algorithm testing	It is suited for algorithm testing (suited for all)

Copyright by Mahidol University

CHAPTER III
MATERIALS AND METHODS

3.1. Project Sampling

3.1.1. Collateral Data System

The collateral data system in the profile is a the central database system for all collateral for which correlations can be displayed and linked with credit (including rent-to-buy loans) and customer data for the benefit of considering credit approval or the use of data for debt collection and litigation enforcement or links for the purpose of using data on collateral and loan money to calculate reserve funds and send the data for reports to the Bank of Thailand.

3.1.1.1. Collateral Data System Structure

The data collected in the collateral data system is composed of data on contracts, customers, loan accounts, collateral, proprietary rights to collateral and data on insurance for collateral with the following structure:

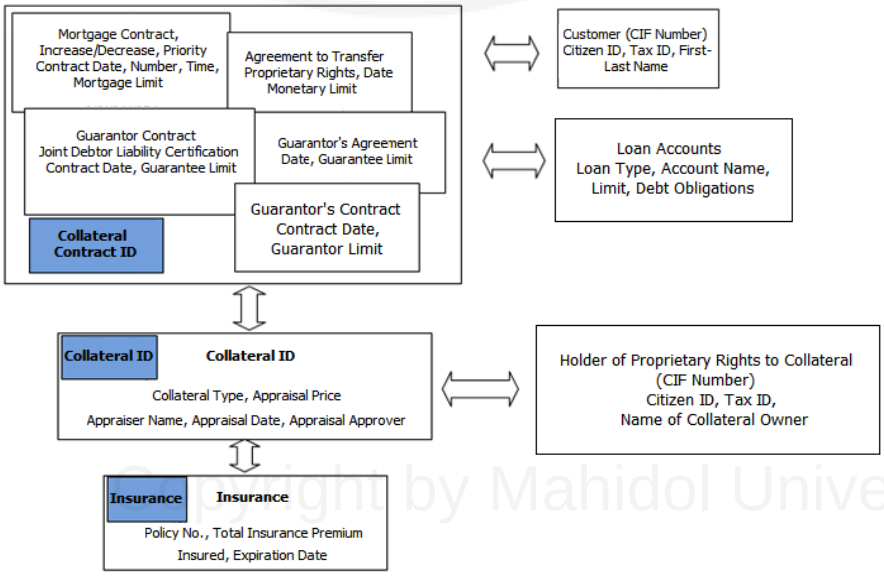


Figure 3.1 Collateral Data System Structure

Copyright by Mahidol University

- Mortgage/Pledging/Guarantor Agreements are key information used to link various parts of data together which might be pledging, mortgage, guarantor, transfer of proprietary rights, mortgage increase agreement, mortgage decrease agreement, collateral/asset increase contract, priority mortgage contract, which contains key information such as the contract date, redemption date, mortgage/pledging/guarantor (collateral) limit.

- Customer - information linked to mortgage/pledging/ guarantor (collateral) contracts to be stored as identification of the customer who is a loan account owner.

- Loan Account - Information linked with mortgage/pledging/ guarantor) collateral contracts to be stored as information on loan accounts and/or loan overdrafts by the debtor in the contract.

- Collateral - Information linked with mortgage/pledging/guarantor (collateral) contracts to be stored as information identifying each type of collateral. The bank has divided collateral types as follows:

Table 3.1 Collateral Type and Collateral Sub Category (Cont)

Collateral Type	**Collateral Sub -Category**
1. Land/Land with Structures	1.1 Land Title Deed
	1.2 Nor. Sor. 3 Kor.
	1.3 Booking/Reservation
	1.4 Nor. Sor. 3
	1.5 Nor. Sor. 3 Khor.
2. Structures	No Sub-Category
3. Machinery	No Sub-Category
4. Depository Account	No Sub-Category
5. Bonds	No Sub-Category
6. Stock Certificates	No Sub-Category
7. Guarantor/Juristic Person Guarantor	No Sub-Category
8. Condominium	No Sub-Category

Copyright by Mahidol University

Table 3.2 Collateral Type and Collateral Sub Category (Cont)

Collateral Type	**Collateral Sub -Category**
9. Concession Right Transfer	No Sub-Category
10. Automobile/Motorcycle	1. Automobile
	2. Motorcycle
11. Rental Rights	No Sub-Category
12. Right to Transfer Assets into Capital	1. Title Document Certification
	2. License
	3. Patent
	4. Petty Patent
	5. Trademark
13. Guarantor Certificate	1. TCG
	2. TCG (Portfolio Guarantee Scheme)
	3. Inherited Pension Fund
	4. Ministry of Finance Guarantor Certificate
14. Lottery	No Sub-Category
15. Other	No Sub-Category

- Holder of Proprietary Rights to Collateral - Information linked to the collateral to be stored with the information on the holder of proprietary rights to the collateral who does not necessarily need to be a bank customer.

- Collateral Insurance Information - Information linked to the collateral to be stored as information on the insurance for the collateral; the collateral is composed of the following four parts:

Policy - Information, e.g. insurance company, policy number, policy date and expiration date; insurance premium;

Copyright by Mahidol University

Insured ID - Information, e.g. name of insured; personal ID number, taxpayer ID number, contact address;

Collateral ID - Information, e.g. address of insured property, sum of building insurance, sum of product insurance;

Insurance Premium Payment - Information, e.g. insurance premium payment date, paid by, amount of money paid.

3.1.1.2. Types of Information Relationships in the System

Loan Account Information (collateral contract, collateral and insurance) can be related in a number of ways as follows:

- One-to-one

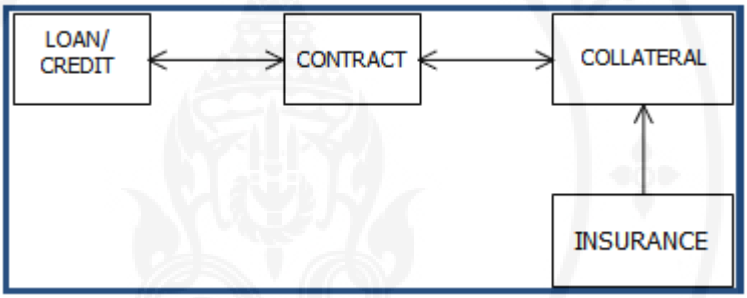


Figure 3.2 One-to-one

- Many-to-one

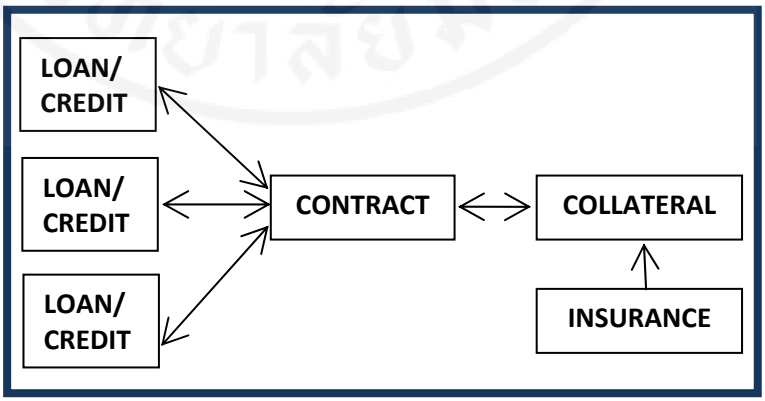


Figure 3.3 Many-to-one

Copyright by Mahidol University

- Many-to-many

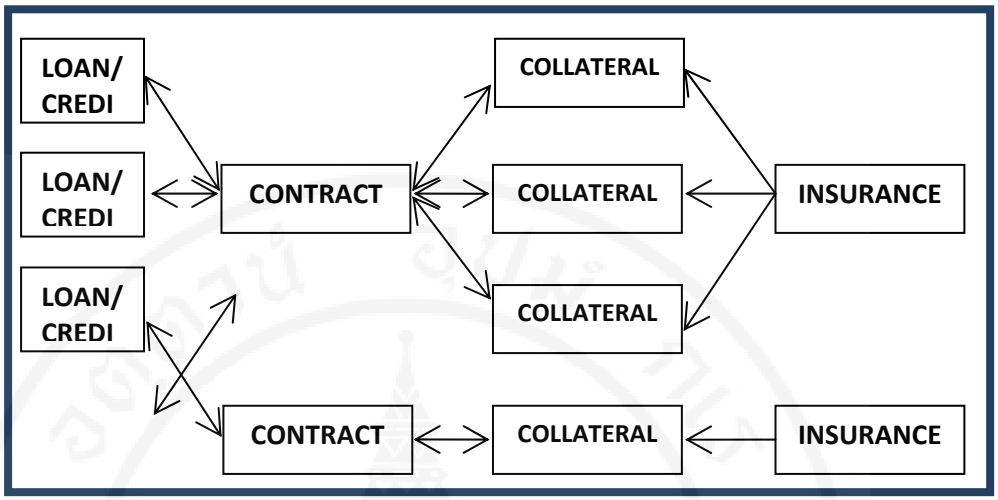


Figure 3.4 Many-to-many

3.1.1.3. Collateral Management System Description

The system has a monitor display for recording collateral categorized by collateral type according to information specific to each part or piece of the collateral, collateral appraisal price and appraisal date;

The system can record contracts according to mortgage/pledging/ guarantor (collateral) contracts. The contracts can be recorded and mortgage contracts can be linked for cases of additional contracts from the following primary contracts:

- Priority Mortgage
- Mortgage Increase
- Mortgage Decrease
- Additional Mortgage Securities
- Partial Mortgage
- Joint Obligor Liability Certificate

Each set of data concerning collateral contracts, including the mortgage/ pledging under each collateral contract can be recorded.

The system prevents basic errors in recording each type of collateral data. For example, real estate collateral has a control file for the provincial, district and sub-district codes nationwide in the form of Land Department codes in which districts are divided differently from governing districts.

Copyright by Mahidol University

The system calculates the mortgages of collateral contracts used in order to handle to provisional funds in compliance with IAS39 criteria.

The system has a warning system for notification about which accounts have no collateral obligations (daily and monthly reports).

The system can organize and display the codes of the agencies owning the collateral by the branch or agency recording the information for the first time and the field where the documents are stored which must be recorded in order to know where the original documents are kept, e.g. main office or branch office, etc. Changes and corrections in collateral descriptions can only be made by the branch or agency in charge of the collateral.

When information on collateral already existing in the system is recorded, the system can check and display a warning as a notification that the aforementioned collateral is already obligated. Checking for repetitive collateral depends on the collateral types in the following table:

Table 3.3 Describe Collateral Type and Primary Key (Cont.)

Collateral Type		**Collateral Sub-Category Type**		**Primary Key**
1	Land/Land & Structures Structures	1	Land Title Deed	Coll Type + Coll Sub Type + No. + Province + District + Sub-district
		2	Nor. Sor. 3 Kor.	Coll Type + Coll Sub Type + No. + Province + District + Sub-district
		3	Booking/Reservation	Coll Type + Coll Sub Type + No. + Province + District + Sub-district
		4	Nor. Sor.3	Coll Type + Coll Sub Type + Moo + Book number + Page Number + Sequence + Province + District + Sub-district
		5	Nor. Sor. 3 Khor.	Coll Type + Coll Sub Type + Moo + Book number + Page Number + Sequence + Province + District + Sub-district
2	Structure	0	No Sub-Category	Coll Type + Coll Sub Type + Building Code + Province + District + Sub-district

Copyright by Mahidol University

Table 3.4 Describe Collateral Type and Primary Key (Cont.)

Collateral Type		**Collateral Sub-Category Type**		**Primary Key**
3	Machinery	0	No Sub-Category	Coll Type + Coll Sub Type + Machine start number
4	Depository Account	0	No Sub-Category	Coll Type + Coll Sub Type + A/C number
5	Bond	0	No Sub-Category	Coll Type + Coll Sub Type + Bond code + Issuer Year + Bond Sequence + Bond start ID
6	Stock Certificate	0	No Sub-Category	Coll Type + Coll Sub Type + Tax ID + Certificate number
7	Guarantor/ Juristic Person Guarantor	0	No Sub-Category	Coll Type + Coll SubType + CIF
8	Condominium	0	No Sub-Category	Coll Type + Coll Sub Type + Room number + Building register number + Province
9	Concession Right Transfer	0	No Sub-Category	Coll Type + Coll SubType + Branch T/R + Concession contract right number + Concession contract date
10	Automobile/ Motorcycle	1	Automobile	Coll Type + Coll Sub Type + Serial No. + Province + District + Sub-district
		2	Motorcycle	Coll Type + Coll Sub Type + Serial No. + Province + District + Sub-district
11	Rental Rights	0	No Sub-Category	Coll Type + Coll Sub Type + License No. + Province + District + Sub-district
12	Right to Transfer Assets into Capital	1	Entitlement Document Certificate	Coll Type + Coll Sub Type + Book No. + Issuance Date
		2	License	Coll Type + Coll Sub Type + Cert. No. + Date of Issuance

Copyright by Mahidol University

Table 3.5 Describe Collateral Type and Primary Key (Cont.)

Collateral Type		**Collateral Sub-Category Type**		**Primary Key**
		3	Patent	Coll Type + Coll Sub Type + Cert. No. + Date of Issuance
		4	Petty Patent	Coll Type + Coll Sub Type + Cert. No. + Date of Issuance
		5	Trademark	Coll Type + Coll Sub Type + Cert. No. + Date of Issuance
13	Guarantor Certificate	1	TCG	Coll Type + Coll Sub Type + Cert. No.
		2	TCG (Portfolio Guarantee Scheme)	Coll Type + Coll Sub Type + Cert. No.
		3	Inherited Pension Fund	Coll Type + Coll Sub Type + Guarantor Cert. No.
		4	Ministry of Finance Guarantor Cert.	Coll Type + Coll Sub Type + Guarantor Cert. No. + Date of Guarantor Cert. Issuance
14	Lottery Ticket (Lots)	0	No Sub-Category	Coll Type + Coll Sub Type + Lottery Registration + Beginning No. + Cat. No. + Installment No. + Ref. No.
15	Other	0	No Sub-Category	Coll Type + Coll SubType + Branch code + System date + Coll.ID

3.1.2. Asset Classification

When debtors miss payments, the system will use the method of counting the number of months for which payment is outstanding on the principle or interest amount as announced by the Bank of Thailand and the quality of the assets according to the criteria for asset classification.

Copyright by Mahidol University

According to the Bank of Thailand notification, the terms for asset clarification used to calculate the number of outstanding principle or interest payment months are the criteria for consideration as follows:

Hence, the criteria for asset classification for both monthly and yearly debts will be based on the same standards.

Table 3.6 Describe asset classification

Asset Classification	**Number of Outstanding Payment Months**
Standard – Normal Classification	Outstanding principle or interest payment for no more than one month.
Special mention	Outstanding principle or interest payment for more than one, but no more than three months.
Sub Standard	Outstanding principle or interest payment for more than three, but no more than six months.
Doubtful	Outstanding principle or interest payment for more than six, but no more than twelve months.
Doubtful of loss	Outstanding principle or interest payment for more than twelve months and up.
Loss	Account with Loss Status

A Non-Performing Loan (NPL) is a loan classified as 3 – 5.

The system will classify assets according to the criteria for counting the outstanding payment period for the principle or interest payment by dividing into one to five classes. For debts written off as losses, the system will store the information for the doubtful debt account separately by recording through the coded items in order to set the account as a loss.

In addition, for asset classification according to the quality criteria, the system also handles the storage of separate information recorded by employees due to the use of criteria for considering loan quality as the criteria for considerations over and above the criteria for the outstanding payment period. Hence, a person should be used to give considerations to matters outside the system and a subsequent record should be made later.

The criteria for processing the results of asset classification by the system will differ according to the loan type as follows:

Copyright© by Mahidol University

3.1.2.1. For Loan Accounts

The system considers asset classification by calculating the number of outstanding payment months for interest or principle payments in the bills for collecting outstanding payments processing the results per batch on a monthly basis.

3.1.2.2. For rent-to-buy loan accounts

The system considers debt classification by calculating based on the number of outstanding payment months exceeding the criteria by processing the results by batch on a monthly basis.

3.1.3. Provision

The system for handling the setting of account provisions by establishing criteria for the setting of provision, namely, using the collateral value according to the Bank of Thailand guidelines for handling IAS 39 to be deducted from the account debt (%PV). A provision must be set at a ratio commensurate with the reason for the debt. Hence, different asset classifications have amounts requiring the setting of different provisions by using asset classification according to the payment or quality criteria in the criteria for setting provisions. In cases involving adjustments in asset classification according to quality criteria, the system will use asset classification according to the quality criteria rather than asset classification according to outstanding payment criteria (consideration in combination with asset classification according to the debt restructuring). At present, the bank does not use the collateral to deduct before calculating the setting of provisions in which the system sets the %PV for every piece of collateral at 0%.

In cases involving loan debts, the government policy or loan amount released by the hired bank will have no provision for doubtful debts.

Table 3.7 Provision Rate by Asset Classification (Cont.)

Provision Rate Categorized by Asset Classification	
Asset Classification	**Provision Setting (%)**
1 - Normal	1
2 - Special Mention	2
3 - Sub-standard	100

Copyright by Mahidol University

Table 3.8 Provision Rate by Asset Classification (Cont.)

Provision Rate Categorized by Asset Classification	
Asset Classification	**Provision Setting (%)**
4 - Doubtful	100
5 - Doubtful of Loss	100
6 - Loss	Write off (No provision necessary)

The criteria for considering provisional settings differ according to loan type as follows:

1. For loan accounts, the provision setting is calculated based on the book value - deductible collateral value.

Book Value = Principle

2. For loan overdrafts, the provision setting is calculated based on the book value - deductible collateral value.

Book Value = Principle – Cumulative Interest + Intra-Monthly Interest (if any)

3. For rent-to-buy loan accounts, the provision setting is calculated based on the book value - deductible collateral value.

Book Value = Outstanding Rent-to-Buy Debtor– Interest Pending Deduction

After calculating value of the provisional setting for the account, the system can go through the provisional account for doubtful debts and doubtful debts in the batch at the end of every month, record the accounts with differences in provisions by recording on a monthly basis and transmitting the information on the account records for the SAP system in which the average totals in the SAP system will need to be adjusted. Set the PV % for every collateral type as follows:

Table 3.9 Set the PV % for every collateral type (Cont.)

Collateral Type	Subtype	Aging	Legal status	%PV	
				Evaluation cost is less than 3 year	Evaluation cost is more than 3 year
1.Land/Land & Structures	All	C1 - Normal	No litigation	90.00	90.00

Copyright by Mahidol University

Table 3.10 Set the PV % for every collateral type (Cont.)

Collateral Type	Subtype	Aging	Legal status	%PV	
				Evaluation cost is less than 3 year	Evaluation cost is more than 3 year
		C2 - Special Mention		90.00	90.00
		C3 - Sub-standard		62.00	0.00
		C4 - Doubtful		62.00	0.00
		C5 - Doubtful of Loss		62.00	0.00
1.Land/Land & Structures Structures	All subtype	C1 - Normal	Litigation is not consider.	90.00	90.00
		C2 - Special Mention		90.00	90.00
		C3 - Sub-standard	In progress litigation Adjudge /Sequester Auction	62.00	0.00
		C4 - Doubtful		66.38	0.00
		C5 - Doubtful of Loss		71.02	0.00
2.Structures	-	C1 - Normal	No litigation	90.00	90.00
		C2 - Special Mention		90.00	90.00
		C3 - Sub-standard		62.00	0.00
		C4 - Doubtful		62.00	0.00
		C5 - Doubtful of Loss		62.00	0.00

Copyright by Mahidol University

Table 3.11 Set the PV % for every collateral type (Cont.)

Collateral Type	Subtype	Aging	Legal status	%PV	
				Evaluation cost is less than 3 year	Evaluation cost is more than 3 year
2.Structures	-	C1 - Normal	Litigation is not consider.	90.00	90.00
		C2 - Special Mention		90.00	90.00
		C3 - Sub-standard	In progress litigation Adjudge /Sequester Auction	62.00	0.00
		C4 - Doubtful		66.38	0.00
		C5 - Doubtful of Loss		71.02	0.00
3. Machinery	-	C1 - Normal	No litigation	90.00	90.00
		C2 - Special Mention		90.00	90.00
		C3 - Sub-standard		84.40	0.00
		C4 - Doubtful		84.40	0.00
		C5 - Doubtful of Loss		84.40	0.00
3. Machinery		C1 - Normal	Litigation is not considered.	90.00	90.00
		C2 - Special Mention			

Copyright by Mahidol University

Table 3.12 Set the PV % for every collateral type (Cont.)

Collateral Type	Subtype	Aging	Legal status	%PV	
				Evaluation cost is less than 3 year	Evaluation cost is more than 3 year
		C3 - Sub-standard	In progress litigation Adjudge /Sequester Auction	84.40	0.00
		C4 - Doubtful		84.40	0.00
		C5 - Doubtful of Loss		84.40	0.00
4. Depository Account	-	C1 - Normal	Litigation is not considered.	100.00	100.00
		C2 - Special Mention		100.00	100.00
		C3 - Sub-standard		100.00	100.00
		C4 - Doubtful		100.00	100.00
		C5 - Doubtful of Loss		100.00	100.00
5. Bond	-	C1 - Normal	Litigation is not considered.	100.00	100.00
		C2 - Special Mention		100.00	100.00
		C3 - Sub-standard		100.00	100.00
		C4 - Doubtful		100.00	100.00
		C5 - Doubtful of Loss		100.00	100.00

Copyright by Mahidol University

Table 3.13 Set the PV % for every collateral type (Cont.)

Collateral Type	Subtype	Aging	Legal status	%PV	
				Evaluation cost is less than 3 year	Evaluation cost is more than 3 year
6. Stock Certificate	-	C1 - Normal	Litigation is not considered.	95.00	95.00
		C2 - Special Mention		95.00	95.00
		C3 - Sub-standard		95.00	95.00
		C4 - Doubtful		95.00	95.00
		C5 - Doubtful of Loss		95.00	95.00
7. Guarantor/ Juristic Person Guarantor	-	C1 - Normal	Litigation is not considered.	0.00	0.00
		C2 - Special Mention		0.00	0.00
		C3 - Sub-standard		0.00	0.00
		C4 - Doubtful		0.00	0.00
		C5 - Doubtful of Loss		0.00	0.00

Copyright by Mahidol University

Table 3.14 Set the PV % for every collateral type (Cont.)

Collateral Type	Subtype	Aging	Legal status	%PV	
				Evaluation cost is less than 3 year	Evaluation cost is more than 3 year
8. Condominium	-	C1 - Normal	Litigation is not considered.	90.00	90.00
		C2 - Special Mention		90.00	90.00
		C3 - Sub-standard		62.00	0.00
		C4 - Doubtful		62.00	0.00
		C5 - Doubtful of Loss		62.00	0.00
8. Condominium	-	C1 - Normal	Litigation is not considered. In progress litigation Adjudge /Sequester Auction	90.00	90.00
		C2 - Special Mention		90.00	90.00
		C3 - Sub-standard		62.00	0.00
		C4 - Doubtful		62.00	0.00
		C5 - Doubtful of Loss		66.38	0.00

Copyright by Mahidol University

Table 3.15 Set the PV % for every collateral type (Cont.)

Collateral Type	Subtype	Aging	Legal status	%PV	
				Evaluation cost is less than 3 year	Evaluation cost is more than 3 year
9. Concession Right Transfer	-	C1 - Normal	Litigation is not considered.	0.00	0.00
		C2 - Special Mention		0.00	0.00
		C3 - Sub-standard		0.00	0.00
		C4 - Doubtful		0.00	0.00
		C5 - Doubtful of Loss		0.00	0.00
10. Automobile/Motorcycle	-	C1 - Normal	Litigation is not considered.	90.00	90.00
		C2 - Special Mention		90.00	90.00
		C3 - Sub-standard		93.50	0.00
		C4 - Doubtful		93.50	0.00
		C5 - Doubtful of Loss		0.00	0.00

Copyright by Mahidol University

Table 3.16 Set the PV % for every collateral type (Cont.)

Collateral Type	Subtype	Aging	Legal status	%PV	
				Evaluation cost is less than 3 year	Evaluation cost is more than 3 year
11. Rental Rights	All subtype	C1 - Normal	No litigation	90.00	90.00
		C2 - Special Mention		90.00	90.00
		C3 - Sub-standard		62.00	0.00
		C4 - Doubtful		62.00	0.00
		C5 - Doubtful of Loss		62.00	0.00
11. Rental Rights	-	C1 - Normal	Litigation is not considered. In progress litigation Adjudge /Sequester Auction	90.00	90.00
		C2 - Special Mention		90.00	90.00
		C3 - Sub-standard		62.00	0.00
		C4 - Doubtful		66.38	0.00
		C5 - Doubtful of Loss		71.02	0.00

Copyright by Mahidol University

Table 3.17 Set the PV % for every collateral type (Cont.)

Collateral Type	Subtype	Aging	Legal status	%PV	
				Evaluation cost is less than 3 year	Evaluation cost is more than 3 year
12.Right to Transfer Assets into Capital	All subtype	C1 - Normal	No litigation	90.00	90.00
		C2 - Special Mention		90.00	90.00
		C3 - Sub-standard		90.00	0.00
		C4 - Doubtful		90.00	0.00
		C5 - Doubtful of Loss		90.00	0.00
13. Guarantor Certificate	All subtype	C1 - Normal	No litigation	90.00	90.00
		C2 - Special Mention		90.00	90.00
		C3 - Sub-standard		90.00	90.00
		C4 - Doubtful		90.00	90.00
		C5 - Doubtful of Loss		90.00	90.00

Copyright by Mahidol University

Table 3.18 Set the PV % for every collateral type (Cont.)

Collateral Type	Subtype	Aging	Legal status	%PV	
				Evaluation cost is less than 3 year	Evaluation cost is more than 3 year
14. Lottery Ticket (Lots)	-	C1 - Normal	No litigation	95.00	95.00
		C2 - Special Mention		95.00	95.00
		C3 - Sub-standard		95.00	95.00
		C4 - Doubtful		95.00	95.00
		C5 - Doubtful of Loss		95.00	95.00
15. Pledged crops	All subtype	C1 - Normal	No litigation	80.00	80.00
		C2 - Special Mention		80.00	80.00
		C3 - Sub-standard		80.00	80.00
		C4 - Doubtful		80.00	80.00
		C5 - Doubtful of Loss		80.00	80.00

Copyright by Mahidol University

Table 3.19 Set the PV % for every collateral type (Cont.)

Collateral Type	Subtype	Aging	Legal status	%PV	
				Evaluation cost is less than 3 year	Evaluation cost is more than 3 year
16.Bill	-	C1 - Normal	No litigation	90.00	90.00
		C2 - Special Mention		90.00	90.00
		C3 - Sub-standard		90.00	90.00
		C4 - Doubtful		90.00	90.00
		C5 - Doubtful of Loss		90.00	90.00
17.Other property	All subtype	C1 - Normal	No litigation	0.00	0.00
		C2 - Special Mention		0.00	0.00
		C3 - Sub-standard		0.00	0.00
		C4 - Doubtful		0.00	0.00
		C5 - Doubtful of Loss		0.00	0.00

Copyright by Mahidol University

Table 3.20 Set the PV % for every collateral type (Cont.)

Collateral Type	Subtype	Aging	Legal status	%PV	
				Evaluation cost is less than 3 year	Evaluation cost is more than 3 year
99.Other	-	C1 - Normal	No litigation	0.00	0.00
		C2 - Special Mention		0.00	0.00
		C3 - Sub-standard		0.00	0.00
		C4 - Doubtful		0.00	0.00
		C5 - Doubtful of Loss		0.00	0.00

Copyright by Mahidol University

3.2. Problem of Testing

Table 3.21 Testing manday

	Estimate Manday		**Actual Manday**	
	Senior	Junior	Senior	Junior
Requirements	8		8	
Test Planning	4		4	
Test Analysis	4		4	
Test Design		30		36
Testing - SIT - UAT	 2 2	 8 8	 2 2	 12 12
Total	20	46	20	60
Senior rate(Baht.)				18,000.00
Junior rate(Baht.)				15,000.00
Each rate (Baht.)	360,000.00	690,000.00	360,000.00	900,000.00
Total rate (Baht.)		1,050,000.00		1,260,000.00
Different Cost (Baht.)			1,260,000.00 – 1,050,000.00 = **210,000.00**	

Senior: 10 persons

Junior: 30 persons

***** Each rate = senior or junior x total of manday**

***** Total rate = rate of senior + rate of junior**

For table 3.5 Testing manday can be describe;

3.2.1. Redundant Testing

3.2.2. Business scenarios are not cover scope because Tester separate write testing design.

Copyright by Mahidol University

3.2.3. Time does not relations because run day end does different so testing process are delay

3.2.4. Very hard testing summary

3.2.5. Document deliver is incorrect pattern

3.2.6. Increased cost

3.3. Design Software Testing and Apply Black Box Testing Technique

Figure 3.5 Software Testing by Black Box Testing Technique

3.3.1. Test Planning

Test planning involves making plans for testing software, which includes the scheduling of various activities, allocating necessary resources, clearly setting the scope of the work and preparing necessities for the test environment such as hardware, software (OS, browsers, database systems), various instruments, workplaces and other relevant equipment, etc.

Copyright by Mahidol University

Table 3.22 Test Plan

Details	Month					
	Jun 2014	**Jul 2014**	**Aug 2014**	**Sep 2014**	**Oct 2014**	**Nov 2014**
Test plan	←	→				
Test analysis	←	→				
Test design			←	→		
Testing				←	→	→

3.3.2. Test Analysis

Test analysis involves setting a test scenario for the testing in concurrence with requirement, system specification, functional specification and business process specification for use in designing a test case model and test script, including prioritizing the test cases.

Table 3.23 Test Scenario

No	Business Scenario
C1	Business Scenario: Check the debts reserve a One to One. Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment. Legal status = Standard.
C2	Business Scenario: Check the debts reserve a One to One. Collateral: Property / land and buildings / Subtype = Total deed has not reviewed the appraisal. Legal status = Litigation status
C3	Business Scenario: Check the debts reserve a One to Many. Collateral: Property / land and buildings / Subtype = Certificate utilization [Nor. Sor. 3 Khor.], Subtype = pre-emption certificate Collateral buildings / Subtype = Not review the appraisal. Legal status = Litigation status

Copyright by Mahidol University

3.3.3. Test Design

Test design involves the arrangement of a test case and a test script used in the testing with a checklist containing the descriptions covering the features and functions of the software being tested. When the testing is complete, the outcomes obtained will be stated (pass/fail) and other necessary details will be put on the checklist for use as key evidence in subsequently sending or receiving deliveries.

3.3.3.1. Test case

Table 3.24 Test case example

Req.#	Test Case #		Test condition	Expectation test summary
R_DRS_4_15.1_1	C_DRS_4_15.1_1	001	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed	- Create collateral by condition - Record collateral owner
	C_DRS_4_15.1_1	002	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed by primary key with collateral	- Cannot create collateral because system is check duplicate data.
	C_DRS_4_15.1_1	003	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed have insurance.	- Create collateral by condition - Record collateral owner - Relate collateral and insurance

Copyright by Mahidol University

3.3.3.2. Test script

Table 3.25 Test script example

Action	Simulated Date	Test Case	Action and Data	Expect value	Test results OK / FR	Each case result
	Day 0		CIF number = xxxx, Customer Type = Personal, Customer Code = xxxxx branch XXXX			
	Day 0		Create loan account due end of month (1MAE)			
1	Day 1		**Log in Web CSR**	xxxxxx		
2	Day 1	001	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed			
			Menu Create Collateral			
			Collateral Type	Land/Land with Structures		
			Collateral Sub type	Land Title Deed		
			Create	Click buttom		
			Tab General			

3.3.4. Test Implementation

Test Implementation such as the following:

- Test Execution Schedule
- Prepare the Environment and Data for Use in Testing
- Prepare the Process Used in Testing Follow-up
- Prepare the process used in monitoring for defects occurring as a result of the testing.
- Prepare Various Instruments

Copyright by Mahidol University

3.3.5. Test Execution

- Perform Testing by the Test Execution Schedule
- Record, Correct, Monitor for Defects

- Monitor Testing Progress
- Control Testing According to Plan

3.3.6. Evaluating Exit Criteria and Test Reporting

This involves checking the test results, e.g. whether or not the testing was performed to completion according to the test execution schedule or whether all defects occurring as a result of the testing have been corrected, etc. Importantly, the following testing progress needs to be reported:

- Number of pass/fail test cases or cases in the process of testing.
- Number of Defects Discovered Categorized by Level of Severity and Status
- Number of Change Requests (CR) occurring, etc.

3.3.7. Test Closure Activities

Test Closure Activities such as the following:

- Check whether or not testing in complete according to the test execution schedule and whether or not all defects discovered have been corrected.
- Hand the work over, e.g. test results, data on incidents occurring in testing and report to the persons involved.
- Store important data and documents, e.g. test results, data on incidents occurring in testing and various reports; and enter into the configuration management system.

3.4. Evaluate result

The evaluate result use a questionnaire containing a survey opinion senior and junior. The questionnaire is divided into the following two parts:

Part 1 Background of respondents

- Gender
- Age
- Education
- Occupation
- Position

Copyright by Mahidol University

Part 2 Opinions of users satisfy using black-box testing technique as follows:

Table 3.26 Evaluate result using questionnaire

Question	Excellent (5)	Good (4)	Normal (3)	Few (2)	Poor (1)
Do you think using Black-box testing help to cover function testing?					
Do you think using Black-box testing help to reduce redundancy?					
Do you think using Black-box testing help to correct and accurate testing?					
Do you think using Black-box testing help to reduce time of work?					
Do you think using Black-box testing help to easy coordinate between department?					
Do you think using Black-box testing help to formal report to user?					
Do you think using Black-box testing help to reduce resource?					
Do you think using Black-box testing help to reduce cost?					
Do you think apply Black-box testing help to using efficacy resource?					
Do you think satisfy using Black-box testing technique ?					

Discussion of research findings from questionnaire use of interval scale

Interval scale = (Maximum – Minimum) / Range

= (5-1)/5

= 0.80

4.21 – 5.00 means Excellent

3.41 – 4.20 means Good

2.61 – 3.40 means Normal

1.81 – 2.60 means Few

Copyright by Mahidol University

1.00 – 1.80 means Poor

Interval scale = (Maximum – Minimum) / Range

= (5-1)/3

= 1.33

3.67 – 5.00 means Excellent

2.33 – 3.66 means Normal

0.99 – 2.32 means Poor

Interval scale = (Maximum – Minimum) / Range

= (5-1)/2

= 2

3.00 – 5.00 means Excellent

0.99 – 2.99 means Poor

3.5. Research Methodology

3.5.1. To study and review software testing theory.

3.5.2. Collection of theories to prepare research by using black-box testing for collateral management system

3.5.3. Review collateral management operation and system.

3.5.4. Define testing software plans which include the scheduling of various activities.

3.5.5. Write business scenario relate Requirement, System Specification, Functional Specification and Business Process Specification for design test case and test script.

3.5.6 Write test case and test script.

3.5.7 Evaluate result for using black-box testing technique

3.5.8. Report black-box testing summary

Copyright by Mahidol University

3.6. Data Collection

3.6.1. Study of theories.

3.6.1.1. Software testing by using black box testing

3.6.1.2. Collateral management system requirement and design manual

3.6.1.3. Write test case and test script technique

3.6.2. Collection

3.6.2.1. Foundation of software testing (ISTQB ERTIFICATION)

3.6.2.2. Black box testing technique

Copyright by Mahidol University

CHAPTER IV
RESULTS

4.1. Test Planning of Collateral Data System

Table 4.1 Test Planning of Collateral Data System

Details	Month					
	Jun 2014	Jul 2014	Aug 2014	Sep 2014	Oct 2014	Nov 2014
Test plan	←	→				
Test analysis	←	→				
Test design			←	→		
Testing				←	→	

Test planning involves making plans for Collateral Data System testing, which includes the scheduling of various activities such as online day, batch day, logical date start, logical date end. Although testing is allocating necessary resources on core bank system and clearly setting the scope of the work then preparing necessities for the test environment.

4.2. Test Analysis of Collateral Data System

Test analysis study collateral data system requirement, system specification, functional specification and collateral management process. Then involves setting a test scenario for the testing. Test Scenario might be classified as positive and negative.

Copyright by Mahidol University

Test Scenario was written all business flow ensure application testing can be using each business action. For example, Loan Account Information can be related in a number of ways as follows:

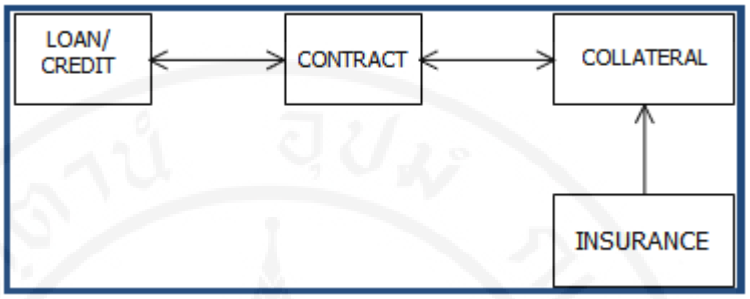


Figure 4.1 Loan Account Information criteria

4.2.1. Test Scenario

Collateral Data System Business Scenario: Check the debts reserve a One to One.

Table 4.2 Test Scenario

No	Business Scenario
C1	Business Scenario: Check the debts reserve a One to One. Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment. Legal status = Standard.

4.3. Test Design

Test design involves the arrangement of a test case and a test script used in the testing with a checklist containing the descriptions covering the features and functions of the software being tested. When the testing is complete, the outcomes obtained will be stated (pass/fail) and other necessary details will be put on the checklist for use as key evidence in subsequently sending or receiving deliveries.

Copyright by Mahidol University

4.3.1. Test case

4.3.1.1. Collateral Management Test Case

Table 4.3 Collateral Management Test Case

Req.#	Test Case #		Test condition	Expectation test summary
R_DRS_01	C_DRS_01	001	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed	- Create collateral by condition - Record collateral owner
	C_DRS_01	002	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed by primary key with collateral	- Cannot create collateral because system is check duplicate data.
R_DRS_02	C_DRS_02	058	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed have insurance.	- Create collateral by condition - Record collateral owner - Relate collateral and insurance
R_DRS_02	C_DRS_02	066	Verify calculate provision rate by asset classification	- Calculate provision rate by asset classification from condition
	C_DRS_04	074	Input Non CRM Flag = Y	- Edit collateral value
	C_DRS_04	075	Input Non CRM Flag = N	- Collateral value is Non CRM Flag = Y cannot calculate provision rate

Copyright by Mahidol University

4.3.1.2. Collateral Management Business Case

Table 4.4 Collateral Management business Case

Scenario	Collateral Type	Subtype	Total appraisal price (Bath)	Pledge (Contract)	Seq.	Loan Type	Credit Limit	Balance	Contract Date
1	Land/ Structures	Land Title Deed	3,000,000	3,000,000	1	LN	3,000,000	3,000,000	DD /MM /YYYY

4.3.1.3. Collateral Management Aging

Table 4.5 Collateral Management Aging

Date	Due Date	Aging	Remain
Day 1 Month 2	End of Month 1	C1	-
Day 1 Month 3	End of Month 2	C1	1M
Day 1 Month 4	End of Month 3	C2	2M
Day 1 Month 5	End of Month 4	C2	3M
Day 1 Month 6	End of Month 5	C3	4M
Day 1 Month 7	End of Month 6	C3	5M
Day 1 Month 8	End of Month 7	C3	6M
Day 1 Month 9	End of Month 8	C3	7M
Day 1 Month 10	End of Month 9	C4	8M
Day 1 Month 11	End of Month 10	C4	9M
Day 1 Month 12	End of Month 11	C4	10M
Day 1 Month 13	End of Month 12	C4	11M
Day 1 Month 14	End of Month 13	C4	12M
Day 1 Month 15	End of Month 14	C5	13M

Copyright by Mahidol University

4.3.1.4. Collateral Management Requirement Number

Table 4.6 Collateral Management Requirement Number

Req.#	Function
R_DRS_01	Create all Collateral data
R_DRS_02	Create contract relate collateral
R_DRS_03	Edit/Change loan account data
R_DRS_04	Edit/Change Collateral data
R_DRS_05	Create Insurance
R_DRS_06	Verify calculate provision rate by asset classification

4.3.2. Test script

Table 4.7 Menu Create Collateral Test script (Cont.)

Scenario Number : R_DRS_01					
Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment. **Legal status = Standard.**					
Version	**Release date**	**Requirement no.**	**Description of Version/ Page (-s)**	**Author**	**Comment**
001	26 Aug 13	R_DRS_01	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed	Chumpol	
Action	**Simulated Date**	**Test Case**	**Action and Data**	**Expect value**	**Test results OK / FR**	**Each case result**
	Day 0		CIF number = xxxx, Customer Type = Personal, Customer Code = xxxxx branch XXXX			

Copyright by Mahidol University

Table 4.8 Menu Create Collateral Test script (Cont.)

Action	Simulated Date	Test Case	Action and Data	Expect value	Test results OK / FR	Each case result
	Day 0		Create loan account due end of month (1MAE)			
1	Day 1		**Log in Web CSR**	xxxxxx		
2	Day 1	001	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed			
			Menu Create Collateral			
			Collateral Type	Land/Land with Structures		
			Collateral Sub type	Land Title Deed		
			Create	Click		

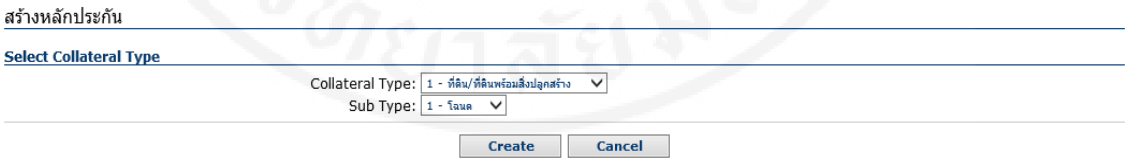


Figure 4.2 Menu Create Collateral on system

Copyright by Mahidol University

Table 4.9 Tab General test script

Scenario Number : R_DRS_01						
Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment. **Legal status = Standard.**						
Version	**Release date**	**Requirement no.**	**Description of Version/ Page (-s)**	**Author**	**Comment**	
001	26 Aug 13	R_DRS_01	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed	Chumpol		
Action	**Simulated Date**	**Test Case**	**Action and Data**	**Expect value**	**Test results OK / FR**	**Each case result**
			Tab General			
			Collateral Type	Yes		
			Branch officer	Show user ID of collateral creator		
			Collateral Branch owner	Show collateral Branch owner		
			Cost centre	Default collateral branch owner value		
			Social care	Default cost centre value		
			Currency	Default - THB		
			Continue	Click		

Copyright by Mahidol University

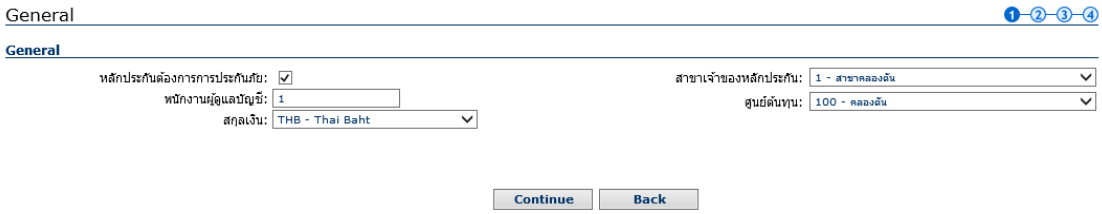


Figure 4.3 Tab General on Collateral management system

Table 4.10 Tab Collateral Information on Collateral management system test script (Cont.)

Scenario Number : R_DRS_01						
Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment. **Legal status = Standard.**						
Version	**Release date**	**Requirement no.**	**Description of Version/ Page (-s)**	**Author**	**Comment**	
001	26 Aug 13	R_DRS_01	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed	Chumpol		
Action	**Simulated Date**	**Test Case**	**Action and Data**	**Expect value**	**Test results OK / FR**	**Each case result**
			Tab Collateral Information			
			Land and Property Information			
			Collateral Type	Land/Land with Structures		
			Collateral Sub type	Land Title Deed		
			Land Title Deed / Nor. Sor. 3 Kor/ Booking/Reservation	10101		

Copyright by Mahidol University

Table 4.11 Tab Collateral Information on Collateral management system test script (Cont.)

Action	Simulated Date	Test Case	Action and Data	Expect value	Test results OK / FR	Each case result
			provincial	Bangkok		
			district	Yannawa		
			sub-district	BangPhongPhang		
			Area	10-0-00-00		
			Appraisal	3,000,000.00		
			Appraisal date	DD/MM/YYYY		
			assessor	Branch		
			Branch assessor	Choose branch		
			Non CRM Flag	N		
			Continue	Click		

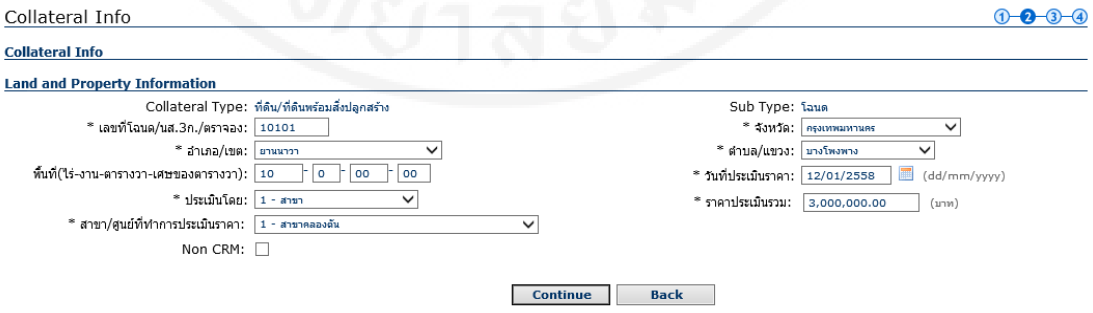


Figure 4.4 Tab Collateral Information on Collateral management system

Copyright by Mahidol University

Table 4.12 Tab Collateral Owners on Collateral management system test script

Scenario Number : R_DRS_01						
Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment. **Legal status = Standard.**						
Version	**Release date**	**Requirement no.**	**Description of Version/ Page (-s)**	**Author**	**Comment**	
001	26 Aug 13	R_DRS_01	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed	Chumpol		
Action	**Simulated Date**	**Test Case**	**Action and Data**	**Expect value**	**Test results OK / FR**	**Each case result**
			Tab Collateral Owners			
			Add			
			Collateral owner	CIF Number xxxx		
			Continue	Click		

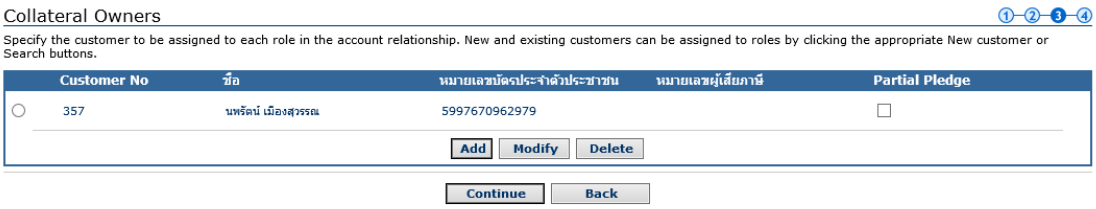


Figure 4.5 Tab Collateral Owners on Collateral management system

Copyright by Mahidol University

Table 4.13 Tab Insurance on Collateral management system test script

Scenario Number : R_DRS_01						
Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment. **Legal status = Standard.**						
Version	**Release date**	**Requirement no.**	**Description of Version/ Page (-s)**	**Author**	**Comment**	
001	26 Aug 13	R_DRS_01	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed	Chumpol		
Action	**Simulated Date**	**Test Case**	**Action and Data**	**Expect value**	**Test results OK / FR**	**Each case result**
			Tab Insurance			
			No			
			Continue	Click		
			Submit			
			Success create Collateral	Collateral Number xxxxxx		

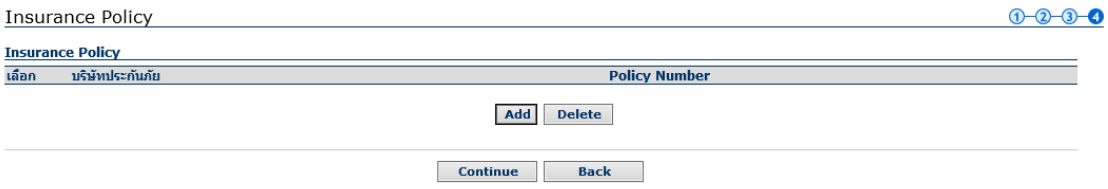


Figure 4.6 Add insurance

Copyright by Mahidol University

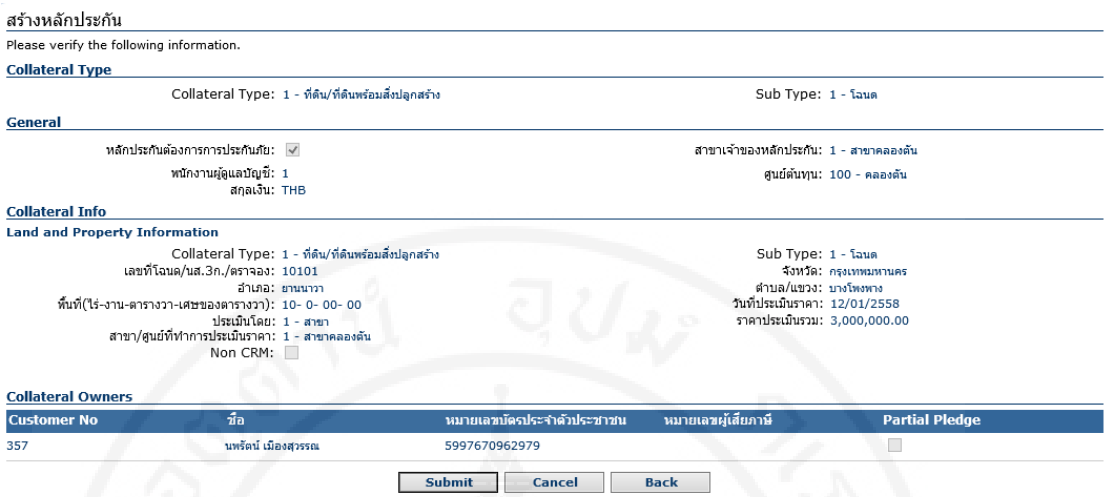


Figure 4.7 Show and submit create collateral

Table 4.14 Tab Insurance and Menu Create Collateral on Collateral management system test script

Action	Simulated Date	Test Case	Action and Data	Expect value	Test results OK / FR	Each case result
3	Day1	002	Create Collateral Type = Land/Land with Structures Subtype = Land Title Deed by double Primary Key with Collateral			
			Menu Create Collateral			
			Collateral Type	Land/Land with Structures		
			Collateral Sub type	Land Title Deed		

Copyright by Mahidol University

Table 4.15 Tab General on Collateral management system test script

Scenario Number : R_DRS_01						
Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment. **Legal status = Standard.**						
Version	**Release date**	**Requirement no.**	**Description of Version/ Page (-s)**	**Author**	**Comment**	
001	26 Aug 13	R_DRS_01	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed	Chumpol		
Action	**Simulated Date**	**Test Case**	**Action and Data**	**Expect value**	**Test results OK / FR**	**Each case result**
			Tab General			
			Collateral Type	Yes		
			Branch officer	Show user ID of collateral creator		
			Collateral Branch owner	Show collateral Branch owner		
			Cost centre	Default collateral branch owner value		
			Social care	Default cost centre value		
			Currency	Default - THB		
			Continue	Click		

Copyright by Mahidol University

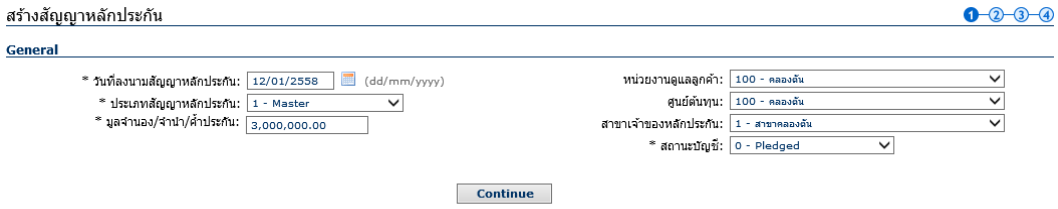


Figure 4.8 Tab General on Collateral management system

Table 4.16 Tab Collateral Information on Collateral management system test script (Cont.)

Scenario Number : R_DRS_01						
Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment. **Legal status = Standard.**						
Version	**Release date**	**Requirement no.**	**Description of Version/ Page (-s)**	**Author**	**Comment**	
001	26 Aug 13	R_DRS_01	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed	Chumpol		
Action	**Simulated Date**	**Test Case**	**Action and Data**	**Expect value**	**Test results OK / FR**	**Each case result**
4	Day 1	002	**Tab Collateral Information**			
			Land and Property Information			
			Collateral Type	Land/Land with Structures		
			Collateral Sub type	Land Title Deed		
			Land Title Deed / Nor. Sor. 3 Kor/ Booking/Reservation	10101		
			Provincial	Bangkok		

Copyright by Mahidol University

Table 4.17 Tab Collateral Information on Collateral management system test script (Cont.)

Action	Simulated Date	Test Case	Action and Data	Expect value	Test results OK / FR	Each case result
			District	Yannawa		
			Sub-district	BangPhongPhang		
			Area	10-0-00-00		
			Appraisal	3,000,000.00		
			Appraisal date	DD/MM/YYYY		
			Assessor	Branch		
			Branch assessor	Choose branch		
			Non CRM Flag	N		
			Continue	Click		

Figure 4.9 Tab Collateral Information on Collateral management system

Copyright by Mahidol University

Table 4.18 Tab Customer link on Collateral management system test script

Scenario Number : R_DRS_01						
Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment. **Legal status = Standard.**						
Version	**Release date**	**Requirement no.**	**Description of Version/ Page (-s)**	**Author**	**Comment**	
001	26 Aug 13	R_DRS_01	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed	Chumpol		
Action	**Simulated Date**	**Test Case**	**Action and Data**	**Expect value**	**Test results OK / FR**	**Each case result**
			Tab Customer link			
			Add			
			Collateral owner	CIF Number xxxx		
			Continue	Click		

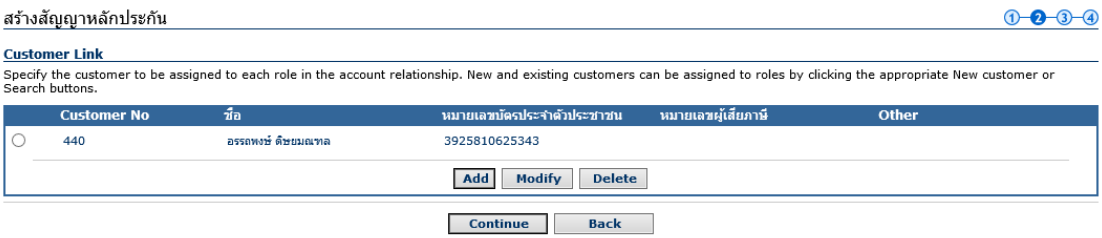


Figure 4.10 Show tab Collateral link on Collateral management system (Cont.)

Copyright by Mahidol University

Table 4.19 Tab Account Link on Collateral management system test script

Scenario Number : R_DRS_01						
Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment. **Legal status = Standard.**						
Version	**Release date**	**Requirement no.**	**Description of Version/ Page (-s)**	**Author**	**Comment**	
001	26 Aug 13	R_DRS_01	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed	Chumpol		
Action	**Simulated Date**	**Test Case**	**Action and Data**	**Expect value**	**Test results OK / FR**	**Each case result**
			Tab Account Link			
			Add			
			Loan account relate contract	Loan Account Number = 8xxxxxxxxxx		
			Continue	Click		

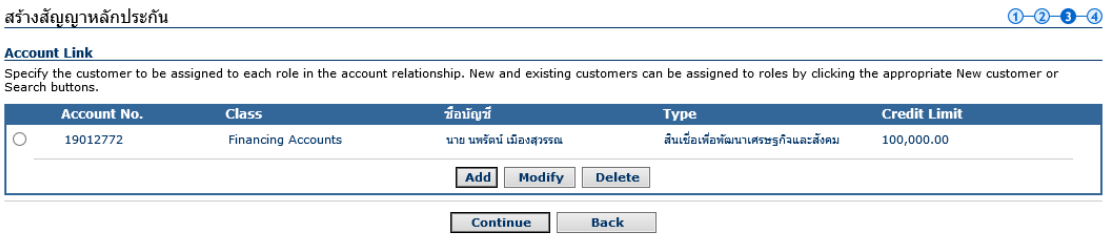


Figure 4.11 Show tab Account Link on Collateral management system

Copyright by Mahidol University

Table 4.20 Tab Collateral Link on Collateral management system test script

Scenario Number : R_DRS_01						
Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment. **Legal status = Standard.**						
Version	**Release date**	**Requirement no.**	**Description of Version/ Page (-s)**	**Author**	**Comment**	
001	26 Aug 13	R_DRS_01	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed	Chumpol		
Action	**Simulated Date**	**Test Case**	**Action and Data**	**Expect value**	**Test results OK / FR**	**Each case result**
			Tab Collateral Link			
			Add			
			Collateral blinding contract	Collateral ID = xxxxx		
			Account status	P - pledge / mortgage		
			Status Date	Collateral contract sign date		
			Submit			
5	Day 1 Month 2		**Success**	Contract Number = xxxx		
6	Day 1 Month 2	066	**Log in Web CSR of branch**	xxxxxx		
			Verify calculate provision rate by asset classification			
			Account Number	Loan Account		

Copyright by Mahidol University

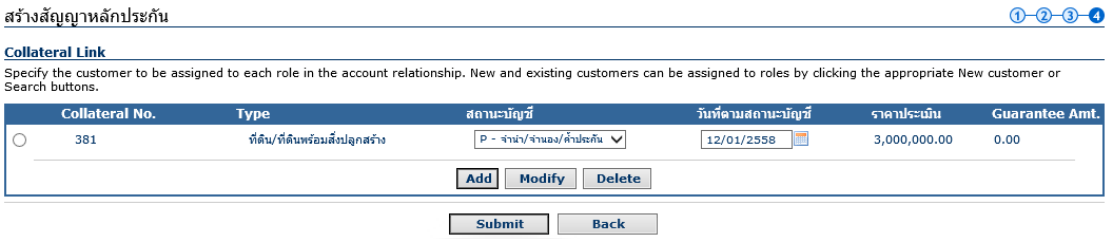


Figure 4.12 Show Collateral Link on Collateral management system test script

Table 4.21 Tab Provision / Reclassification by Provision Percentage = 1.00000 and C Class = 1 on Collateral management system test script (Cont.)

Scenario Number : R_DRS_01						
Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment. **Legal status = Standard.**						
Version	**Release date**	**Requirement no.**	**Description of Version/ Page (-s)**	**Author**	**Comment**	
001	26 Aug 13	R_DRS_01	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed	Chumpol		
Action	**Simulated Date**	**Test Case**	**Action and Data**	**Expect value**	**Test results OK / FR**	**Each case result**
			Tab Delinquency			
			Tab sub provision / reclassification			
			Provision Processing			
			Provision Balance	Show calculate provision rate by asset classification results		

Copyright by Mahidol University

Table 4.22 Tab Provision / Reclassification by Provision Percentage = 1.00000 and C Class = 1 on Collateral management system test script (Cont.)

Action	Simulated Date	Test Case	Action and Data	Expect value	Test results OK / FR	Each case result
			Provision Percentage	1.00000		
			Reclassification Processing			
			C Class - Aging	1		
			Provision Processing			

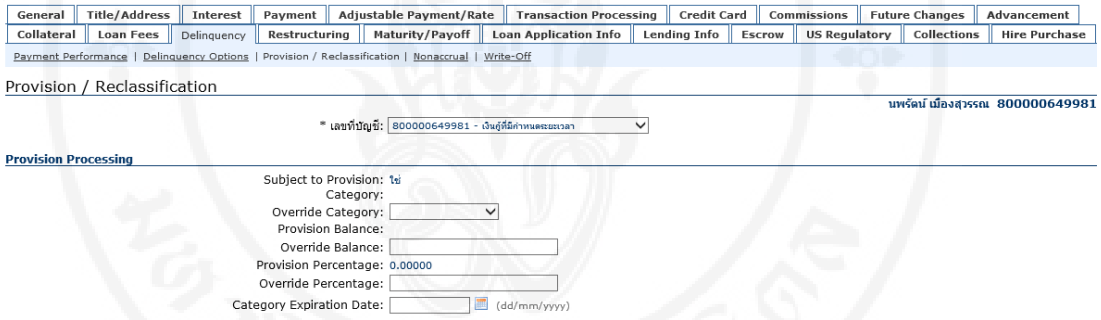


Figure 4.13 Tab Provision / Reclassification by Provision Percentage = 1.00000 and C Class = 1 on Collateral management system

Copyright by Mahidol University

Table 4.23 Log in Web CSR of branch and click Tab Delinquency by Provision Percentage = 1.00000 and C Class = 1 on Collateral management system test script (Cont.)

Scenario Number : R_DRS_01						
Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment. **Legal status = Standard.**						
Version	**Release date**	**Requirement no.**	**Description of Version/ Page (-s)**	**Author**	**Comment**	
001	26 Aug 13	R_DRS_01	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed	Chumpol		
Action	**Simulated Date**	**Test Case**	**Action and Data**	**Expect value**	**Test results OK / FR**	**Each case result**
7	Day 1 Month 3		**Log in Web CSR of branch**	xxxxxx		
8	Day 1 Month 3	066	Verify calculate provision rate by asset classification			
			Account Number	Loan Account		
			Tab Delinquency			
			Tab sub provision / reclassification			
			Provision Processing			
			Provision Balance	Show calculate provision rate by asset classification results		
			Provision Percentage	1.00000		

Copyright by Mahidol University

Table 4.24 Log in Web CSR of branch and click Tab Delinquency by Provision Percentage = 1.00000 and C Class = 1 on Collateral management system test script (Cont.)

Action	Simulated Date	Test Case	Action and Data	Expect value	Test results OK / FR	Each case result
			Reclassification Processing			
			C Class - Aging	1		
			Provision Processing			

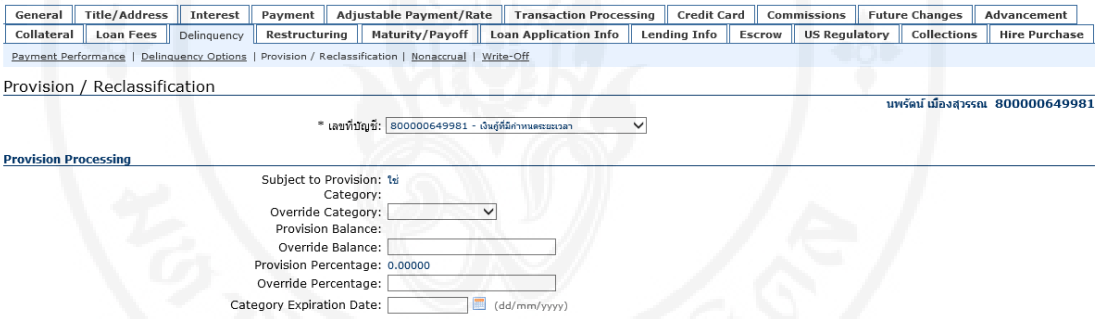


Figure 4.14 Log in Web CSR of branch and click Tab Delinquency by Provision Percentage = 1.00000 and C Class = 1 on Collateral management system

Copyright by Mahidol University

Table 4.25 Tab Provision / Reclassification by Provision Percentage = 2.00000 and C Class = 2 on Collateral management system test script

Scenario Number : R_DRS_01						
Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment. **Legal status = Standard.**						
Version	**Release date**	**Requirement no.**	**Description of Version/ Page (-s)**	**Author**	**Comment**	
001	26 Aug 13	R_DRS_01	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed	Chumpol		
Action	**Simulated Date**	**Test Case**	**Action and Data**	**Expect value**	**Test results OK / FR**	**Each case result**
9	Day 1 Month 4		**Log in Web CSR of branch**	xxxxxx		
10	Day 1 Month 4	066	Verify calculate provision rate by asset classification			
			Account Number	Loan Account		
			Tab Delinquency			
			Tab sub provision / reclassification			
			Provision Processing			
			Provision Balance	Show calculate provision rate by asset classification results		
			Provision Percentage	2.00000		
			Reclassification Processing			
			C Class - Aging	2		
			Provision Processing			

Copyright by Mahidol University

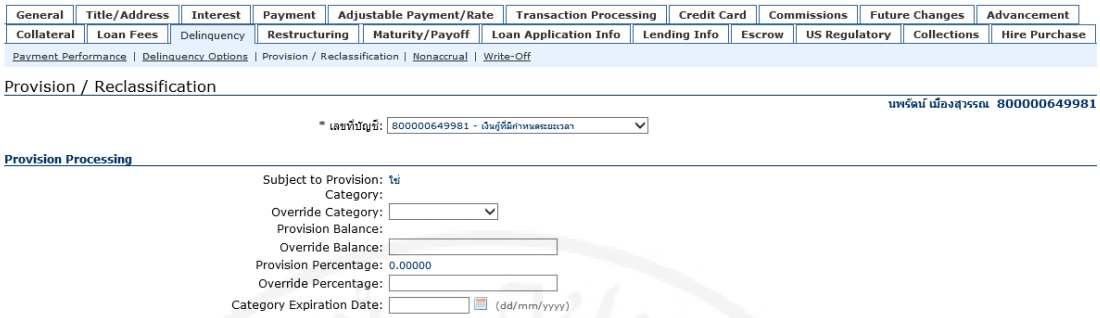


Figure 4.15 Tab Provision / Reclassification by Provision Percentage = 2.00000 and C Class = 2 on Collateral management system

Table 4.26 Log in Web CSR of branch and click Tab Delinquency by Provision Percentage = 2.00000 and C Class = 2on Collateral management system test script (Cont.)

Scenario Number : R_DRS_01						
Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment. **Legal status = Standard.**						
Version	**Release date**	**Requirement no.**	**Description of Version/ Page (-s)**	**Author**	**Comment**	
001	26 Aug 13	R_DRS_01	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed	Chumpol		
Action	**Simulated Date**	**Test Case**	**Action and Data**	**Expect value**	**Test results OK / FR**	**Each case result**
10	Day 1 Month 5		**Log in Web CSR of branch**	xxxxxx		
11	Day 1 Month 5	066	Verify calculate provision rate by asset classification			
			Account Number	Loan Account		

Copyright by Mahidol University

Table 4.27 Log in Web CSR of branch and click Tab Delinquency by Provision Percentage = 2.00000 and C Class = 2on Collateral management system test script (Cont.)

Action	Simulated Date	Test Case	Action and Data	Expect value	Test results OK / FR	Each case result
			Tab Delinquency			
			Tab sub provision / reclassification			
			Provision Processing			
			Provision Balance	Show calculate provision rate by asset classification results		
			Provision Percentage	2.00000		
			Reclassification Processing			
			C Class - Aging	2		
			Provision Processing			

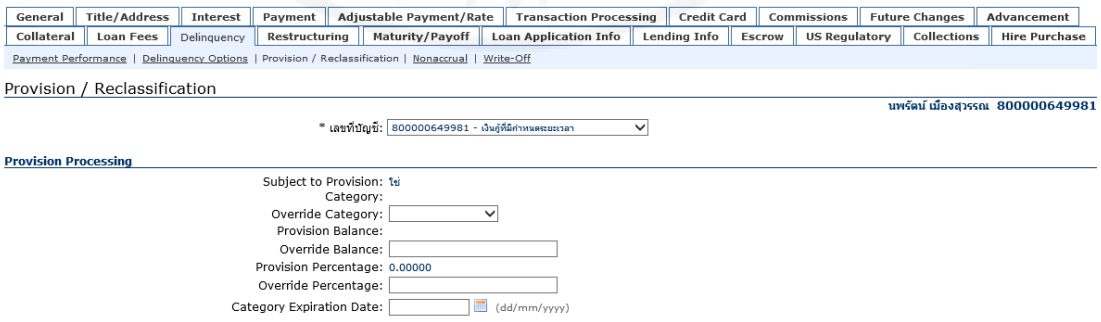


Figure 4.16 Log in Web CSR of branch and click Tab Delinquency by Provision Percentage = 2.00000 and C Class = 2on Collateral management system

Copyright by Mahidol University

Table 4.28 Tab Provision / Reclassification by Provision Percentage = 100.00000 and C Class = 3 on Collateral management system test script

Scenario Number : R_DRS_01						
Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment. **Legal status = Standard.**						
Version	**Release date**	**Requirement no.**	**Description of Version/ Page (-s)**		**Author**	**Comment**
001	26 Aug 13	R_DRS_01	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed		Chumpol	
Action	**Simulated Date**	**Test Case**	**Action and Data**	**Expect value**	**Test results OK / FR**	**Each case result**
12	Day 1 Month 6		**Log in Web CSR of branch**	xxxxxx		
13	Day 1 Month 6	066	Verify calculate provision rate by asset classification			
			Account Number	Loan Account		
			Tab Delinquency			
			Tab sub provision / reclassification			
			Provision Processing			
			Provision Balance	Show calculate provision rate by asset classification results		
			Provision Percentage	100.00000		
			Reclassification Processing			
			C Class - Aging	3		
			Provision Processing			

Copyright by Mahidol University

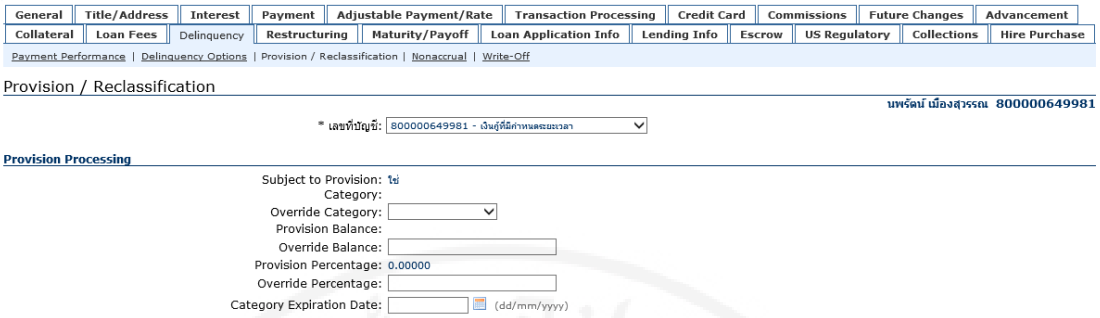


Figure 4.17 Tab Provision / Reclassification by Provision Percentage = 100.00000 and C Class = 3 on Collateral management system

Table 4.29 Tab Provision / Reclassification by Provision Percentage = 100.00000 and C Class = 3 on Collateral management system test script (Cont.)

Scenario Number : R_DRS_01					
Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment. **Legal status = Standard.**					
Version	**Release date**	**Requirement no.**	**Description of Version/ Page (-s)**	**Author**	**Comment**
001	26 Aug 13	R_DRS_01	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed	Chumpol	

Action	**Simulated Date**	**Test Case**	**Action and Data**	**Expect value**	**Test results OK / FR**	**Each case result**
14	Day 1 Month 7		**Log in Web CSR of branch**	xxxxxx		
15	Day 1 Month 7	066	Verify calculate provision rate by asset classification			
			Account Number	Loan Account		

Copyright© by Mahidol University

Table 4.30 Tab Provision / Reclassification by Provision Percentage = 100.00000 and C Class = 3 on Collateral management system test script (Cont.)

Action	Simulated Date	Test Case	Action and Data	Expect value	Test results OK / FR	Each case result
			Tab Delinquency			
			Tab sub provision / reclassification			
			Provision Processing			
			Provision Balance	Show calculate provision rate by asset classification results		
			Provision Percentage	100.00000		
			Reclassification Processing			
			C Class - Aging	3		
			Provision Processing			

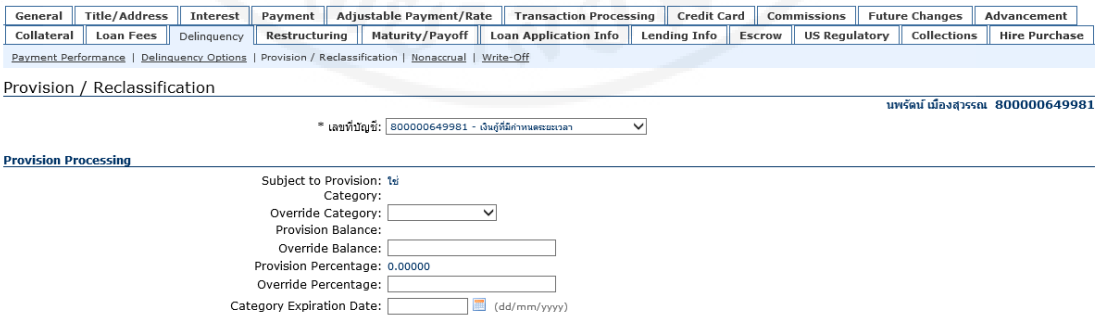


Figure 4.18 Tab Provision / Reclassification by Provision Percentage = 100.00000 and C Class = 3 on Collateral management system

Copyright by Mahidol University

Table 4.31 Log in Web CSR of branch and click Tab Delinquency by Provision Percentage = 100.00000 and C Class = 3 on Collateral management system test script

Scenario Number : R_DRS_01						
Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment. **Legal status = Standard.**						
Version	**Release date**	**Requirement no.**	**Description of Version/ Page (-s)**	**Author**	**Comment**	
001	26 Aug 13	R_DRS_01	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed	Chumpol		
Action	**Simulated Date**	**Test Case**	**Action and Data**	**Expect value**	**Test results OK / FR**	**Each case result**
16	Day 1 Month 8		**Log in Web CSR of branch**	xxxxxx		
17	Day 1 Month 8	066	Verify calculate provision rate by asset classification			
			Account Number	Loan Account		
			Tab Delinquency			
			Tab sub provision / reclassification			
			Provision Processing			
			Provision Balance	Show calculate provision rate by asset classification results		
			Provision Percentage	100.00000		
			Reclassification Processing			
			C Class - Aging	3		
			Provision Processing			

Copyright by Mahidol University

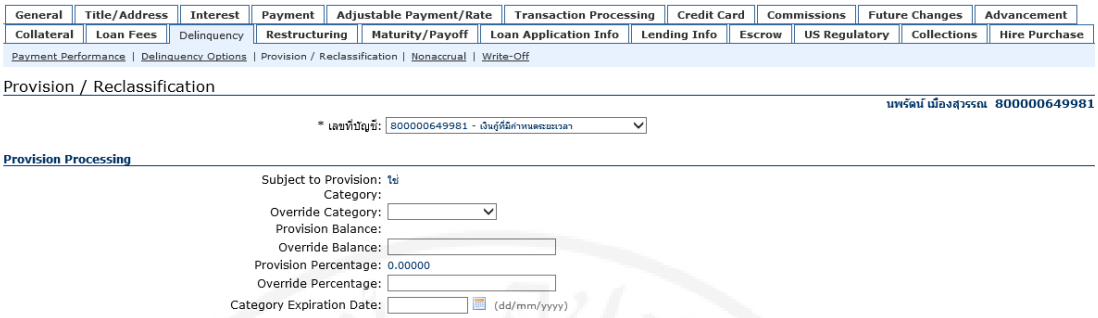


Figure 4.19 Log in Web CSR of branch and click Tab Delinquency by Provision Percentage = 100.00000 and C Class = 3 on Collateral management system

Table 4.32 Tab Provision / Reclassification by Provision Percentage = 100.00000 and C Class = 4 on Collateral management system test script (Cont.)

Scenario Number : R_DRS_01					
Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment. **Legal status = Standard.**					
Version	**Release date**	**Requirement no.**	**Description of Version/ Page (-s)**	**Author**	**Comment**
001	26 Aug 13	R_DRS_01	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed	Chumpol	

Action	**Simulated Date**	**Test Case**	**Action and Data**	**Expect value**	**Test results OK / FR**	**Each case result**
14	Day 1 Month 9		**Log in Web CSR of branch**	xxxxxx		
15	Day 1 Month 9	066	Verify calculate provision rate by asset classification			
			Account Number	Loan Account		

Copyright© by Mahidol University

Table 4.33 Tab Provision / Reclassification by Provision Percentage = 100.00000 and C Class = 4 on Collateral management system test script (Cont.)

Action	Simulated Date	Test Case	Action and Data	Expect value	Test results OK / FR	Each case result
			Tab Delinquency			
			Tab sub provision / reclassification			
			Provision Processing			
			Provision Balance	Show calculate provision rate by asset classification results		
			Provision Percentage	100.00000		
			Reclassification Processing			
			C Class - Aging	3		
			Provision Processing			

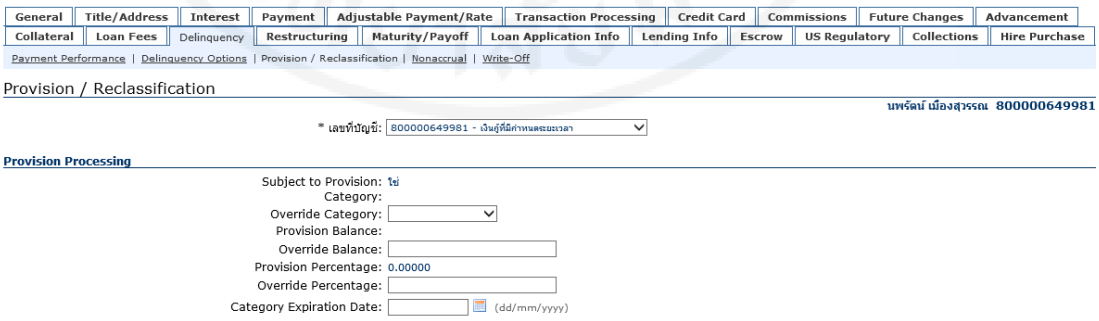


Figure 4.20 Tab Provision / Reclassification by Provision Percentage = 4.00000 and C Class = 4 on Collateral management system (Cont.)

Copyright by Mahidol University

Table 4.34 Log in Web CSR of branch and click Tab Delinquency by Provision Percentage = 100.00000 and C Class = 4 on Collateral management system test script

Scenario Number : R_DRS_01						
Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment. **Legal status = Standard.**						
Version	**Release date**	**Requirement no.**	**Description of Version/ Page (-s)**		**Author**	**Comment**
001	26 Aug 13	R_DRS_01	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed		Chumpol	
Action	**Simulated Date**	**Test Case**	**Action and Data**	**Expect value**	**Test results OK / FR**	**Each case result**
14	Day 1 Month 10		**Log in Web CSR of branch**	xxxxxx		
15	Day 1 Month 10	066	Verify calculate provision rate by asset classification			
			Account Number	Loan Account		
			Tab Delinquency			
			Tab sub provision / reclassification			
			Provision Processing			
			Provision Balance	Show calculate provision rate by asset classification results		
			Provision Percentage	100.00000		
			Reclassification Processing			
			C Class - Aging	3		
			Provision Processing			

Copyright by Mahidol University

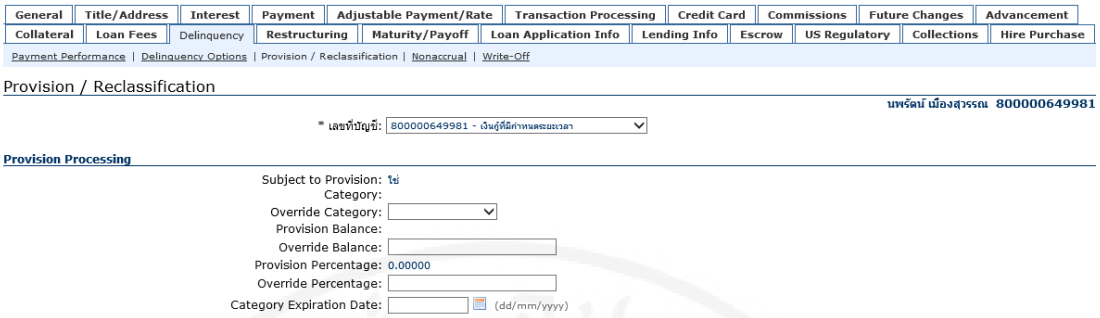


Figure 4.21 Log in Web CSR of branch and click Tab Delinquency by Provision Percentage = 100.00000 and C Class = 4 on Collateral management system

Table 4.35 Log in Web CSR of branch and Input Non CRM Flag = Y on Collateral management system test script

Scenario Number : R_DRS_01						
Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment. **Legal status = Standard.**						
Version	**Release date**	**Requirement no.**	**Description of Version/ Page (-s)**		**Author**	**Comment**
001	26 Aug 13	R_DRS_01	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed		Chumpol	
Action	**Simulated Date**	**Test Case**	**Action and Data**	**Expect value**	**Test results OK / FR**	**Each case result**
16	End of Month 11		**Log in Web CSR of branch**	xxxxxx		
17	End of Month 12	074	Input Non CRM Flag = Y			
			Menu Get Collateral			
			Land and Property Information			
			Non CRM Flag	Y		

Copyright by Mahidol University

Table 4.36 Log in Web CSR of branch and Input Non CRM Flag = N on Collateral management system test script

Scenario Number : R_DRS_01						
Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment. **Legal status = Standard.**						
Version	**Release date**	**Requirement no.**	**Description of Version/ Page (-s)**		**Author**	**Comment**
001	26 Aug 13	R_DRS_01	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed		Chumpol	
Action	**Simulated Date**	**Test Case**	**Action and Data**	**Expect value**	**Test results OK / FR**	**Each case result**
16	End of Month 11		**Log in Web CSR of branch**	xxxxxx		
17	End of Month 13	075	Input Non CRM Flag = N			
			Menu Get Collateral			
			Land and Property Information			
			Non CRM Flag	N		

Copyright by Mahidol University

Table 4.37 Log in Web CSR of branch and click Tab Delinquency by Provision Percentage = 100.00000 and C Class = 5 on Collateral management system test script

Scenario Number : R_DRS_01						
Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment. **Legal status = Standard.**						
Version	**Release date**	**Requirement no.**	**Description of Version/ Page (-s)**		**Author**	**Comment**
001	26 Aug 13	R_DRS_01	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed		Chumpol	
Action	**Simulated Date**	**Test Case**	**Action and Data**	**Expect value**	**Test results OK / FR**	**Each case result**
18	Day 1 Month 12		**Log in Web CSR of branch**	xxxxxx		
19	Day 1 Month 12	066	Verify calculate provision rate by asset classification			
			Account Number	Loan Account		
			Tab Delinquency			
			Tab sub provision / reclassification			
			Provision Processing			
			Provision Balance	Show calculate provision rate by asset classification results		
			Provision Percentage	100.00000		
			Reclassification Processing			
			C Class - Aging	5		
			Provision Processing			

Copyright by Mahidol University

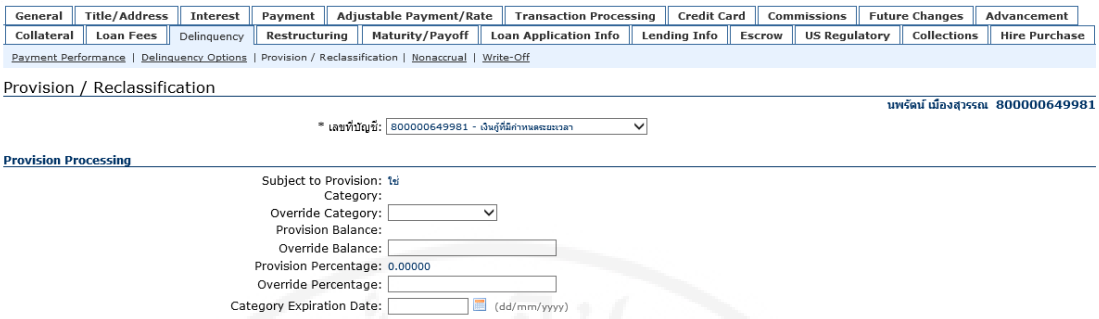


Figure 4.22 Log in Web CSR of branch and click Tab Delinquency by Provision Percentage = 100.00000 and C Class = 4 on Collateral management system

4.3.3. Test Summary

Table 4.38 Test summary (Cont.)

TEST SUMMARY					
Project Name : Core Banking System					
Application : Profile Direct					
System Name : Collateral management system					
Product Name : Loan Processing - Collateral and Provision					
Version	Release date	Description of Version/Page (-s)	Requirement no.	Author	Comment
Scenario Number : R_DRS_01					
Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment. Legal status = Standard.					
Prerequisite Test Case Number :					

Test Script Day	Test Case#	No .	Description
Day 0			CIF number = xxxx, Customer Type = Personal, Customer Code = xxxxx branch XXXX
Day 0			Create loan account due end of month (1MAE)

Copyright by Mahidol University

Table 4.39 Test summary (Cont.)

Test Script Day	Test Case#	No .	Description
Day 1	R_DRS_01_15	001	Create collateral Type = Land/Land with Structures Subtype = Land Title Deed
Day 1	R_DRS_01_15	002	Create Collateral Type = Land/Land with Structures Subtype = Land Title Deed by double Primary Key with Collateral
Day 1	R_DRS_01_15		Land and Property Information
Day 1 Month 2	R_DRS_01_15	066	Verify calculate provision rate by asset classification
Day 1 Month 3	R_DRS_01_15	066	Verify calculate provision rate by asset classification
Day 1 Month 4	R_DRS_01_15	066	Verify calculate provision rate by asset classification
Day 1 Month 5	R_DRS_01_15	066	Verify calculate provision rate by asset classification
Day 1 Month 6	R_DRS_01_15	066	Verify calculate provision rate by asset classification
Day 1 Month 7	R_DRS_01_15	066	Verify calculate provision rate by asset classification
Day 1 Month 8	R_DRS_01_15	066	Verify calculate provision rate by asset classification
Day 1 Month 9	R_DRS_01_15	066	Verify calculate provision rate by asset classification
Day 1 Month 10	R_DRS_01_15	066	Verify calculate provision rate by asset classification
End of Month 12	R_DRS_01_15	074	Input Non CRM Flag = Y
End of Month 13	R_DRS_01_15	075	Input Non CRM Flag = N
Day 1 Month 12	R_DRS_01_15	066	Verify calculate provision rate by asset classification

Copyright by Mahidol University

CHAPTER V
DISCUSSION AND CONCLUSION

5.1. Discussion summary

Table 5.1 Testing Manday summary

	Estimate Manday		**Actual Manday (Before improvement)**		**Actual Manday (After improvement)**	
	Senior	**Junior**	**Senior**	**Junior**	**Senior**	**Junior**
Requirements	8		8		8	
Test Planning	4		4		4	
Test Analysis	4		4		4	
Test Design		30		36		28
Testing -SIT -UAT	 2 2	 8 8	 2 2	 12 12	 1 1	 6 6
Total	20	46	20	60	18	40
Senior rate(Baht.)	18,000.00					
Junior rate(Baht.)	15,000.00					
Each rate (Baht.)	360,000	690,000	360,000	900,000	324,000	600,000
Total rate (Baht.)	1,050,000		1,260,000		924,000	
Different Cost (Baht.)			1,050,000 -1,260,000 = **(210,000)**		1,050,000 -924,000 = **126,000**	

Senior: 10 persons, Junior: 30 persons

Copyright by Mahidol University

Result can be described testing manday concern test design and testing. If testing design was good design, It reduce time, cost and people. Table 5.1 should be improving process to testing as follow;

Project leader plan testing manday that senior was 20 manday and junior was 46 manday. All cost is about 1,050,000 baths. Testing process was not using black-box testing. Therefore testing is redundant testing and no cover scope effect to time is increased and over cost. Then time does not relations because run day end does different so testing process is delay. Actual date was over such as testing design increase 36 manday from 30 manday and testing process increase 28 manday from 20 manday affect on over cost. From this problem project leader improve testing process by using black-box testing technique. Black-box testing technique begin analyze business scenario and write test scenario. Then write test case and test script
Test case relates to business scenario such as check the debts reserve a one to one.
Collateral: Property / land and buildings / Subtype = Total deed has reviewed the valuation of up to 3 years from the date of the last assessment and legal status is standard. Then test case set condition from business scenario relates collateral management system. After test both user acceptance test and system integration test.
If testing is finish, tester is test summary and report to project leader. When using black-box testing technique can be reducing time and cost as follow;

Table 5.2 Actual Manday (After improvement)

	Actual Manday (After improvement)	
Test Design		28
Testing -SIT -UAT	 1 1	 6 6

Copyright by Mahidol University

Table 5.2 Actual Manday after improve testing process demonstrate Test Design manday reduce 30 manday to 28 manday. Testing process manday reduce 20 manday to 14 manday. So cost is lower.

Table 5.3 Compare manday after using black-box testing technique

Process	**Estimate Manday**	**Cost (Bath)**	**Actual Manday**	**Cost**
Test Design	30	450,000	28	420,000
Testing	20	Senior= 72,000 Junior= 240,000	14	Senior= 36,000 Junior= 180,000
Total	50	762,000	42	636,000

Calculate method

Process cost = Manday x Rate

Test design estimate cost = 30 x 15,000 = 450,000

Test design actual cost = 28 x 15,000 = 420,000

Testing estimate cost = (18,000 x 4) + (15,000 x 16)

= 72,000 + 240,000 = 312,000

Testing actual cost = (18,000 x 2) + (15,000 x 12)

= 36,000+ 180,000 = 216,000

Table 5.3 Compare manday after using black-box testing technique describes actual cost was 636,000 bath reduce 126,000 bath from estimate cost.

5.2. Assessment

Table 5.4 Evaluate results by senior (Cont.)

Question	**s1**	**s2**	**s3**	**s4**	**s5**	**s6**	**s7**	**s8**	**s9**	**s10**	**Total**
Do you think using Black-box testing help to cover function testing?	5	5	5	5	5	4	5	5	5	5	**4.90**

Copyright by Mahidol University

Table 5.5 Evaluate results by senior (Cont.)

Question	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10	Total
Do you think using Black-box testing help to reduce redundancy?	5	4	5	4	5	5	5	5	5	5	**4.80**
Do you think using Black-box testing help to correct and accurate testing?	4	5	5	5	5	5	5	5	5	5	**4.90**
Do you think using Black-box testing help to reduce time of work?	5	5	3	4	5	5	5	4	5	5	**4.60**
Do you think using Black-box testing help to easy coordinate between departments?	4	5	4	4	5	5	4	4	5	5	**4.50**
Do you think using Black-box testing help to formal report to user?	4	5	5	5	5	5	4	5	4	4	**4.60**
Do you think using Black-box testing help to reduce resource?	5	5	4	4	5	4	4	5	5	5	**4.60**
Do you think using Black-box testing help to reduce cost?	5	4	5	5	5	5	5	5	4	5	**4.80**
Do you think apply Black-box testing help to using efficacy resource?	4	4	5	5	5	5	5	5	5	5	**4.80**
Do you think satisfy using Black-box testing technique?	5	5	5	4	5	5	5	5	5	5	**4.90**
											4.74

Table 5.4 and Table 5.5 based on the senior in the collateral management system testing has using Black-box testing can be describe;

Table 5.6 Describe result assessment by senior (Cont.)

Factor	score	Discussion result
Cover function in collateral management system	4.90	Excellent
Reduce redundancy	4.80	Excellent
Correct and accurate testing	4.90	Excellent
Reduce time of work	4.60	Excellent

Copyright by Mahidol University

Table 5.7 Describe result assessment by senior (Cont.)

Factor	score	Discussion result
Easy coordinate between department	4.50	Excellent
Formal report to user	4.60	Excellent
Reduce resource	4.60	Excellent
Reduce cost	4.80	Excellent
Using efficacy resource	4.80	Excellent
Satisfy using Black-box testing technique	4.90	Excellent

Table 5.6 and Table 5.7 can be describe senior satisfy using Black-box testing technique. Seniors are excellent satisfaction. There is most benefit such as cover function in collateral management system, correct and accurate testing average 4.90 while reduce redundancy, reduce cost and using efficacy resource average 4.80.

Table 5.8 Evaluate results by junior (Cont.)

Question	J1	J2	J3	J4	J5	J6	J7	J8	J9	J10
Do you think using Black-box testing help to cover function testing?	5	5	5	5	5	5	5	5	5	5
Do you think using Black-box testing help to reduce data redundancy?	4	5	5	5	5	5	4	4	4	4
Do you think using Black-box testing help to correct and accurate testing?	5	5	5	5	5	5	5	5	4	4
Do you think using Black-box testing help to reduce time of work?	5	5	5	5	5	5	5	5	5	5
Do you think using Black-box testing help to easy coordinate between departments?	4	4	4	4	4	4	4	4	4	4
Do you think using Black-box testing help to formal report to user?	4	4	4	4	4	4	4	5	4	5
Do you think using Black-box testing help to reduce resource?	5	5	5	5	5	4	5	5	5	4
Do you think using Black-box testing help to reduce cost?	5	5	5	5	5	5	5	5	4	4
Do you think apply Black-box testing help to using efficacy resource?	5	5	5	5	5	5	5	4	4	5
Do you think satisfy using Black-box testing?	5	5	5	5	5	5	5	5	5	5

Copyright by Mahidol University

Table 5.9 Evaluate results by junior (Cont.)

Question	J 11	J 12	J 13	J 14	J 15	J 16	J 17	J 18	J 19	J 20
Do you think using Black-box testing help to cover function testing?	5	5	5	5	5	5	5	5	5	5
Do you think using Black-box testing help to reduce data redundancy?	4	4	4	5	5	5	5	5	5	5
Do you think using Black-box testing help to correct and accurate testing?	5	5	5	5	5	5	5	5	5	5
Do you think using Black-box testing help to reduce time of work?	5	5	5	5	4	5	4	5	5	5
Do you think using Black-box testing help to easy coordinate between departments?	5	5	5	5	5	5	5	5	5	5
Do you think using Black-box testing help to formal report to user?	4	5	5	5	4	4	4	4	4	4
Do you think using Black-box testing help to reduce resource?	5	5	4	5	5	4	5	5	4	5
Do you think using Black-box testing help to reduce cost?	4	4	4	4	4	4	4	5	4	5
Do you think apply Black-box testing help to using efficacy resource?	5	5	5	5	5	5	5	5	5	5
Do you think satisfy using Black-box testing?	4	4	5	5	5	5	5	5	5	5

Table 5.10 Evaluate results by junior (Cont.)

Question	J 21	J 22	J 23	J 24	J 25	J 26	J 27	J 28	J 29	J 30	Total
Do you think using Black-box testing help to cover function testing?	5	5	4	5	5	5	5	5	5	5	**4.97**
Do you think using Black-box testing help to reduce data redundancy?	5	5	5	5	4	5	4	4	4	4	**4.57**
Do you think using Black-box testing help to correct and accurate testing?	5	5	5	5	5	5	5	5	5	5	**4.93**
Do you think using Black-box testing help to reduce time of work?	4	5	5	5	5	5	5	5	5	5	**4.90**
Do you think using Black-box testing help to easy coordinate between departments?	5	4	4	4	4	4	4	4	4	4	**4.37**
Do you think using Black-box testing help to formal report to user?	4	5	5	5	5	5	4	4	4	4	**4.33**
Do you think using Black-box testing help to reduce resource?	5	5	5	4	5	5	5	5	5	5	**4.80**
Do you think using Black-box testing help to reduce cost?	5	4	5	5	5	4	5	5	4	5	**4.57**

Copyright by Mahidol University

Table 5.11 Evaluate results by junior (Cont.)

Question	J 21	J 22	J 23	J 24	J 25	J 26	J 27	J 28	J 29	J 30	Total
Do you think apply Black-box testing help to using efficacy resource?	5	5	5	4	5	4	5	5	5	5	4.87
Do you think satisfy using Black-box testing?	5	5	5	5	5	5	5	5	5	5	4.93
											4.72

Table 5.8-Table 5.11 based on the junior in the collateral management system testing has using Black-box testing can be describe;

Table 5.12 Describe result assessment by junior

Factor	score	Discussion result
Cover function in collateral management system	4.97	Excellent
Reduce redundancy	4.57	Excellent
Correct and accurate testing	4.93	Excellent
Reduce time of work	4.90	Excellent
Easy coordinate between department	4.37	Excellent
Formal report to user	4.33	Excellent
Reduce resource	4.80	Excellent
Reduce cost	4.57	Excellent
Using efficacy resource	4.87	Excellent
Satisfy using Black-box testing technique	4.93	Excellent

Table 5.12 can be describe junior satisfy using Black-box testing technique. Juniors are excellent satisfaction. There is most benefit such as cover function in collateral management system average 4.97 while correct and accurate testing average 4.93.

Copyright by Mahidol University

Table 5.13 Average satisfy summary

Factor	score	Discussion result
Senior	4.74	Excellent
Junior	4.72	Excellent
	4.73	**Excellent**

Table 5.13 can be describing senior and junior satisfy using Black-box testing technique. There are excellent satisfactions averages 4.73. Black-box testing technique opinion can be applied another software development or develop collateral management system with other bank because collateral management system is same concept but improve business condition and parameter value. This research using practice for organization.

Copyright by Mahidol University

REFERENCES

E. Steegmans, P. Bekaert, F. Devos, G. Delanote, N. Smeets, M. van Dooren and J. Boydens. Black & White Testing. Bridging Black Box Testing and White Box Testing. 2003; 1(1), 1-12.

Harsh Bhasin, Esha Khanna, Sudha. Black Box Testing based on Requirement Analysis and Design Specifications. International Journal of Computer Applications. 2014; 87(18), 36-41.

Kaushik Rana. (2013). Blackbox testing of web service. Blackbox testing of web service,1 ,48-57.

Kwang Ik Seo, Eun Man Choi. Comparison of Five Black-box Testing Methods for Object-Oriented Software. the Fourth International Conference on Software Engineering Research. 2006; 1(6), 1-8.

Mohd. Ehmer Khan, Farmeena Khan. A Comparative Study of White Box, Black Box and Grey Box Testing Techniques. International Journal of Advanced Computer Science and Applications. 2012; 3(6), 12-15.

OWASP. OWASP Testing guide, 2002-2008. USA. OWASP Foundation; 2008.

Pak-Lok Poon. Black Box Testing: Its Fundamental Concepts and Problems. ISACA. 2008; 1(6), 1-8.

P.B.Selvapriya, Different Software Testing Strategies and Techniques. International Journal of Science and Modern Engineering (IJISME). 2013; 2(1):21-24.

Shivani Acharya, Vidhi Pandya. 2008; Bridge between Black Box and White Box – Gray Box Testing Technique,2 ,175-184.

Tsuyoshi Yumoto, Toru Matsuodani, Kazuhiko Tsuda. A Test Analysis Method for Black Box Testing Using AUT and Fault Knowledge.Tokyo: University of Tsukuba; 2013.

Copyright by Mahidol University

APPENDICES

Copyright by Mahidol University

APPENDIX A

Business Scenario

No	Business Scenario
C1	Business Scenario : ตรวจสอบการคำนวณสำรองตามชั้นหนี้ แบบ One to One หลักประกันประเภทที่ดิน/ที่ดินพร้อมสิ่งปลูกสร้าง/Subtype = โฉนด มีการทบทวนการประเมินราคา ไม่เกิน 3 ปี นับจากวันที่ประเมินครั้งล่าสุด สถานะทางกฎหมาย = ปกติ

TEST CASES
Test Cases : Loan Processing - Collateral and Provision **1. การสร้างข้อมูลหลักประกันทุกประเภท** **2. การสร้างข้อมูลสัญญาและผูกความสัมพันธ์กับหลักประกัน** **3. ปรับปรุง/เปลี่ยนแปลงข้อมูลบัญชีสินเชื่อ** **4. ปรับปรุง/เปลี่ยนแปลงข้อมูลหลักประกัน** **5. การสร้างกรมธรรม์ประกันภัย** **6. ตรวจสอบการคำนวณการกันสำรองตามชั้นหนี้**

Version			Requirement no.	Author	Comment

Req.#	Test Case #		เงื่อนไขในการทดสอบ	ผลการทดสอบที่คาดหวัง
R_DRS_4_15.1_1	C_DRS_4_15.1_1	001	สร้างหลักประกันประเภท Collateral Type = ที่ดิน/ที่ดินพร้อมสิ่งปลูกสร้าง Subtype = โฉนด ประเภทที่ดินเปล่า	- สามารถสร้างหลักประกันได้ตามเงื่อนไข - สามารถบันทึกเจ้าของกรรมสิทธิ์หลักประกันได้

Copyright by Mahidol University

Req.#	Test Case #		เงื่อนไขในการทดสอบ	ผลการทดสอบที่คาดหวัง
	C_DRS_4_15.1_1	002	สร้างหลักประกันประเภท Collateral Type = ที่ดิน/ที่ดินพร้อมสิ่งปลูกสร้าง Subtype = โฉนด ประเภทที่ดินเปล่า โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้ เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ
	C_DRS_4_15.1_1	003	สร้างหลักประกันประเภท Collateral Type = ที่ดิน/ที่ดินพร้อมสิ่งปลูกสร้าง Subtype = โฉนด ประเภทที่ดินพร้อมสิ่งปลูกสร้าง มีกรมธรรม์ประกันภัย	- สามารถสร้างหลักประกันได้ตามเงื่อนไข - สามารถบันทึกเจ้าของกรรมสิทธิ์หลักประกันได้ - สามารถผูกความสัมพันธ์กับกรมธรรม์ประกันภัยได้
	C_DRS_4_15.1_1	004	สร้างหลักประกันประเภท Collateral Type = ที่ดิน/ที่ดินพร้อมสิ่งปลูกสร้าง Subtype = นส.3ก	- สามารถสร้างหลักประกันได้ตามเงื่อนไข - สามารถบันทึกเจ้าของกรรมสิทธิ์หลักประกันได้
	C_DRS_4_15.1_1	005	สร้างหลักประกันประเภท Collateral Type = ที่ดิน/ที่ดินพร้อมสิ่งปลูกสร้าง Subtype = นส.3ก โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้ เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ

Copyright by Mahidol University

Req.#	Test Case #		เงื่อนไขในการทดสอบ	ผลการทดสอบที่คาดหวัง
	C_DRS_4_15.1_1	006	สร้างหลักประกันประเภท Collateral Type = ที่ดิน/ที่ดินพร้อมสิ่งปลูกสร้าง Subtype = ตราจอง	- สามารถสร้างหลักประกันได้ตามเงื่อนไข - สามารถบันทึกเจ้าของกรรมสิทธิ์หลักประกันได้
	C_DRS_4_15.1_1	007	สร้างหลักประกันประเภท Collateral Type = ที่ดิน/ที่ดินพร้อมสิ่งปลูกสร้าง Subtype = ตราจอง โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ
	C_DRS_4_15.1_1	008	สร้างหลักประกันประเภท Collateral Type = ที่ดิน/ที่ดินพร้อมสิ่งปลูกสร้าง Subtype = นส.3	- สามารถสร้างหลักประกันได้ตามเงื่อนไข - สามารถบันทึกเจ้าของกรรมสิทธิ์หลักประกันได้
	C_DRS_4_15.1_1	009	สร้างหลักประกันประเภท Collateral Type = ที่ดิน/ที่ดินพร้อมสิ่งปลูกสร้าง Subtype = นส.3 โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ

Copyright by Mahidol University

Req.#	Test Case #		เงื่อนไขในการทดสอบ	ผลการทดสอบที่คาดหวัง
	C_DRS_4_15.1_1	010	สร้างหลักประกันประเภท Collateral Type = ที่ดิน/ที่ดินพร้อมสิ่งปลูกสร้าง Subtype = นส.3 ข	- สามารถสร้างหลักประกันได้ตามเงื่อนไข - สามารถบันทึกเจ้าของกรรมสิทธิ์หลักประกันได้
	C_DRS_4_15.1_1	011	สร้างหลักประกันประเภท Collateral Type = ที่ดิน/ที่ดินพร้อมสิ่งปลูกสร้าง Subtype = นส.3 ข โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ
	C_DRS_4_15.1_1	012	สร้างหลักประกันประเภท Collateral Type = สิ่งปลูกสร้าง Subtype = ไม่มีประเภทย่อย มีกรมธรรม์ประกันภัย	- สามารถสร้างหลักประกันได้ตามเงื่อนไข - สามารถบันทึกเจ้าของกรรมสิทธิ์หลักประกันได้ - สามารถผูกความสัมพันธ์กับกรมธรรม์ประกันภัยได้
	C_DRS_4_15.1_1	013	สร้างหลักประกันประเภท Collateral Type = สิ่งปลูกสร้าง Subtype = ไม่มีประเภทย่อย โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ

Copyright by Mahidol University

Req.#	Test Case #		เงื่อนไขในการทดสอบ	ผลการทดสอบที่คาดหวัง
	C_DRS_4_15.1_1	014	สร้างหลักประกันประเภท Collateral Type = เครื่องจักร Subtype = ไม่มีประเภทย่อย อยู่ในความต้องการของตลาด	- สามารถสร้างหลักประกันได้ตามเงื่อนไข
	C_DRS_4_15.1_1	015	สร้างหลักประกันประเภท Collateral Type = เครื่องจักร Subtype = ไม่มีประเภทย่อย ไม่อยู่ในความต้องการของตลาด	- สามารถสร้างหลักประกันได้ตามเงื่อนไข
	C_DRS_4_15.1_1	016	สร้างหลักประกันประเภท Collateral Type = เครื่องจักร Subtype = ไม่มีประเภทย่อย โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้ เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ
	C_DRS_4_15.1_1	017	สร้างหลักประกันประเภท Collateral Type = เงินฝาก Subtype = ไม่มีประเภทย่อย	- สามารถสร้างหลักประกันได้ตามเงื่อนไข - ระบบ Hold บัญชีเงินฝากตามมูลค่าที่บันทึกให้โดยอัตโนมัติ

Copyright by Mahidol University

Req.#	Test Case #		เงื่อนไขในการทดสอบ	ผลการทดสอบที่คาดหวัง
	C_DRS_4_15.1_1	018	สร้างหลักประกันประเภท Collateral Type = เงินฝาก Subtype = ไม่มีประเภทย่อย โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้ เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ
	C_DRS_4_15.1_1	019	สร้างหลักประกันประเภท Collateral Type = พันธบัตร Subtype = ไม่มีประเภทย่อย	- สามารถสร้างหลักประกันได้ตามเงื่อนไข
	C_DRS_4_15.1_1	020	สร้างหลักประกันประเภท Collateral Type = พันธบัตร Subtype = ไม่มีประเภทย่อย โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้ เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ
	C_DRS_4_15.1_1	021	สร้างหลักประกันประเภท Collateral Type = ใบหุ้น Subtype = ไม่มีประเภทย่อย	- สามารถสร้างหลักประกันได้ตามเงื่อนไข
	C_DRS_4_15.1_1	022	สร้างหลักประกันประเภท Collateral Type = ใบหุ้น Subtype = ไม่มีประเภทย่อย โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้ เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ

Copyright by Mahidol University

Req.#	Test Case #		เงื่อนไขในการทดสอบ	ผลการทดสอบที่คาดหวัง
	C_DRS_4_15.1_1	023	สร้างหลักประกันประเภท Collateral Type = บุคคล/นิติบุคคล ค้ำประกัน Subtype = บุคคล/นิติบุคคล แบบบุคคลธรรมดา	- สามารถสร้างหลักประกันได้ตามเงื่อนไข
	C_DRS_4_15.1_1	024	สร้างหลักประกันประเภท Collateral Type = บุคคล/นิติบุคคล ค้ำประกัน Subtype = บุคคล/นิติบุคคล แบบบุคคลธรรมดา โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้ เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ
	C_DRS_4_15.1_1	025	สร้างหลักประกันประเภท Collateral Type = บุคคล/นิติบุคคล ค้ำประกัน Subtype = บุคคล/นิติบุคคล แบบนิติบุคคล	- สามารถสร้างหลักประกันได้ตามเงื่อนไข
	C_DRS_4_15.1_1	026	สร้างหลักประกันประเภท Collateral Type = บุคคล/นิติบุคคล ค้ำประกัน Subtype = บุคคล/นิติบุคคล แบบนิติบุคคล โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้ เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ

Copyright by Mahidol University

Req.#	Test Case #		เงื่อนไขในการทดสอบ	ผลการทดสอบที่คาดหวัง
	C_DRS_4_15.1_1	027	สร้างหลักประกันประเภท Collateral Type = บุคคล/นิติบุคคล ค้ำประกัน Subtype = หนังสือรับรอง	- สามารถสร้างหลักประกันได้ตามเงื่อนไข
	C_DRS_4_15.1_1	028	สร้างหลักประกันประเภท Collateral Type = บุคคล/นิติบุคคล ค้ำประกัน Subtype = หนังสือรับรอง โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ
	C_DRS_4_15.1_1	029	สร้างหลักประกันประเภท Collateral Type = คอนโดมีเนียม/อาคารชุด Subtype = ไม่มีประเภทย่อย มีกรมธรรม์ประกันภัย	- สามารถสร้างหลักประกันได้ตามเงื่อนไข
	C_DRS_4_15.1_1	030	สร้างหลักประกันประเภท Collateral Type = คอนโดมีเนียม/อาคารชุด Subtype = ไม่มีประเภทย่อย โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ

Copyright by Mahidol University

Req.#	Test Case #		เงื่อนไขในการทดสอบ	ผลการทดสอบที่คาดหวัง
	C_DRS_4_15.1_1	031	สร้างหลักประกันประเภท Collateral Type = โอนสิทธิเรียกร้อง/มอบอำนาจรับเงิน Subtype = ไม่มีประเภทย่อย	- สามารถสร้างหลักประกันได้ตามเงื่อนไข
	C_DRS_4_15.1_1	032	สร้างหลักประกันประเภท Collateral Type = โอนสิทธิเรียกร้อง/มอบอำนาจรับเงิน Subtype = ไม่มี ประเภทย่อย โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ
	C_DRS_4_15.1_1	033	สร้างหลักประกันประเภท Collateral Type = รถยนต์/รถจักรยานยนต์ Subtype = รถยนต์ มีกรมธรรม์ประเภทชั้น 1	- สามารถสร้างหลักประกันได้ตามเงื่อนไข
	C_DRS_4_15.1_1	034	สร้างหลักประกันประเภท Collateral Type = รถยนต์/รถจักรยานยนต์ Subtype = รถยนต์ ไม่มีกรมธรรม์ประกันภัย	- สามารถสร้างหลักประกันได้ตามเงื่อนไข
	C_DRS_4_15.1_1	035	สร้างหลักประกันประเภท Collateral Type = รถยนต์/รถจักรยานยนต์ Subtype = รถยนต์ โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ

Copyright by Mahidol University

Req.#	Test Case #		เงื่อนไขในการทดสอบ	ผลการทดสอบที่คาดหวัง
	C_DRS_4_15.1_1	036	สร้างหลักประกันประเภท Collateral Type = รถยนต์/รถจักรยานยนต์ Subtype = รถจักรยานยนต์ มีกรมธรรม์ประกันภัย	- สามารถสร้างหลักประกันได้ตามเงื่อนไข
	C_DRS_4_15.1_1	037	สร้างหลักประกันประเภท Collateral Type = รถยนต์/รถจักรยานยนต์ Subtype = รถจักรยานยนต์ โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ
	C_DRS_4_15.1_1	038	สร้างหลักประกันประเภท Collateral Type = สิทธิการเช่า Subtype =ไม่มีประเภทย่อย	- สามารถสร้างหลักประกันได้ตามเงื่อนไข
	C_DRS_4_15.1_1	039	สร้างหลักประกันประเภท Collateral Type = สิทธิการเช่า Subtype =ไม่มีประเภทย่อย โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ
	C_DRS_4_15.1_1	040	สร้างหลักประกันประเภท Collateral Type = สิทธิการแปลงสินทรัพย์เป็นทุน Subtype =ใบอนุญาต	- สามารถสร้างหลักประกันได้ตามเงื่อนไข

Copyright by Mahidol University

Req.#	Test Case #		เงื่อนไขในการทดสอบ	ผลการทดสอบที่คาดหวัง
	C_DRS_4_15.1_1	041	สร้างหลักประกันประเภท Collateral Type = สิทธิการแปลงสินทรัพย์เป็นทุน Subtype =ใบอนุญาต โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้ เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ
	C_DRS_4_15.1_1	042	สร้างหลักประกันประเภท Collateral Type = สิทธิการแปลงสินทรัพย์เป็นทุน Subtype =สิทธิบัตร	- สามารถสร้างหลักประกันได้ตามเงื่อนไข
	C_DRS_4_15.1_1	043	สร้างหลักประกันประเภท Collateral Type = สิทธิการแปลงสินทรัพย์เป็นทุน Subtype =สิทธิบัตร โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้ เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ
	C_DRS_4_15.1_1	044	สร้างหลักประกันประเภท Collateral Type = สิทธิการแปลงสินทรัพย์เป็นทุน Subtype =อนุสิทธิบัตร	- สามารถสร้างหลักประกันได้ตามเงื่อนไข
	C_DRS_4_15.1_1	045	สร้างหลักประกันประเภท Collateral Type = สิทธิการแปลงสินทรัพย์เป็นทุน Subtype =อนุสิทธิบัตร โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้ เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ

Copyright by Mahidol University

Req.#	Test Case #		เงื่อนไขในการทดสอบ	ผลการทดสอบที่คาดหวัง
	C_DRS_4_15.1_1	046	สร้างหลักประกันประเภท Collateral Type = สิทธิการแปลงสินทรัพย์เป็นทุน Subtype =เครื่องหมายการค้า	- สามารถสร้างหลักประกันได้ตามเงื่อนไข
	C_DRS_4_15.1_1	047	สร้างหลักประกันประเภท Collateral Type = สิทธิการแปลงสินทรัพย์เป็นทุน Subtype =เครื่องหมายการค้า โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ
	C_DRS_4_15.1_1	048	สร้างหลักประกันประเภท Collateral Type = สิทธิการแปลงสินทรัพย์เป็นทุน Subtype = หนังสือรับรองเอกสารสิทธิ์	- สามารถสร้างหลักประกันได้ตามเงื่อนไข
	C_DRS_4_15.1_1	049	สร้างหลักประกันประเภท Collateral Type = สิทธิการแปลงสินทรัพย์เป็นทุน Subtype =หนังสือรับรองเอกสารสิทธิ์ โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ
	C_DRS_4_15.1_1	050	สร้างหลักประกันประเภท Collateral Type = หนังสือค้ำประกัน Subtype =บสย.	- สามารถสร้างหลักประกันได้ตามเงื่อนไข

Copyright by Mahidol University

Req.#	Test Case #		เงื่อนไขในการทดสอบ	ผลการทดสอบที่คาดหวัง
	C_DRS_4_15.1_1	051	สร้างหลักประกันประเภท Collateral Type = หนังสือค้ำประกัน Subtype =บสย. โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้ เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ
	C_DRS_4_15.1_1	052	สร้างหลักประกันประเภท Collateral Type = สลาก Subtype = ไม่มีประเภทย่อย ประเภทสลากออมสินพิเศษ	- สามารถสร้างหลักประกันได้ตามเงื่อนไข - ระบบ Hold สลากตาม Sequence ให้โดยอัตโนมัติ
	C_DRS_4_15.1_1	053	สร้างหลักประกันประเภท Collateral Type = สลาก Subtype = ไม่มีประเภทย่อย ประเภทสลาก โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่	- ไม่สามารถสร้างหลักประกันได้ เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ
	C_DRS_4_15.1_1	054	สร้างหลักประกันประเภท Collateral Type = สลาก Subtype = ไม่มีประเภทย่อย ประเภทสลากออมสินธนโชค	- สามารถสร้างหลักประกันได้ตามเงื่อนไข - ระบบ Hold สลากตาม Sequence ให้โดยอัตโนมัติ
	C_DRS_4_15.1_1	055	สร้างหลักประกันประเภท Collateral Type = สลาก Subtype = ไม่มีประเภทย่อย ประเภทสลากออมสินธนโชค โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้ เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ
	C_DRS_4_15.1_1	056	สร้างหลักประกันประเภท Collateral Type = อื่นๆ Subtype = ไม่มีประเภทย่อย	- สามารถสร้างหลักประกันได้ตามเงื่อนไข

Copyright by Mahidol University

Req.#	Test Case #		เงื่อนไขในการทดสอบ	ผลการทดสอบที่คาดหวัง
	C_DRS_4_15.1_1	057	สร้างหลักประกันประเภท Collateral Type = อื่นๆ Subtype =ไม่มีประเภทย่อย โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว	- ไม่สามารถสร้างหลักประกันได้ เนื่องจากระบบมีการตรวจสอบว่ามีข้อมูลซ้ำ
R_DRS_4_15.1_2	C_DRS_4_15.1_2	058	สร้างสัญญาหลักประกันแบบหลักประกัน 1 หลักประกัน ค้ำประกัน 1 บัญชีเงินกู้ และผูกความสัมพันธ์ระหว่างหลักประกัน บัญชีเงินกู้ และสัญญาหลักประกัน	- สามารถผูกความสัมพันธ์ระหว่างหลักประกัน 1 ชิ้น กับบัญชีเงินกู้ 1 บัญชีได้
	C_DRS_4_15.1_2	059	สร้างสัญญาหลักประกันแบบหลักประกันหลาย หลักประกัน ค้ำประกัน 1 บัญชีเงินกู้ และผูกความสัมพันธ์ระหว่างหลักประกัน บัญชีเงินกู้ และสัญญาหลักประกัน	- สามารถผูกความสัมพันธ์ระหว่างหลักประกันหลายชิ้น กับบัญชีเงินกู้ 1 บัญชีได้
	C_DRS_4_15.1_2	060	สร้างสัญญาหลักประกันแบบหลักประกันหลาย หลักประกัน ค้ำประกันหลายบัญชีเงินกู้ และผูกความสัมพันธ์ระหว่างหลักประกัน บัญชีเงินกู้ และสัญญาหลักประกัน	- สามารถผูกความสัมพันธ์ระหว่างหลักประกันหลายชิ้น กับบัญชีเงินกู้หลายบัญชีได้

Copyright by Mahidol University

Req.#	Test Case #		เงื่อนไขในการทดสอบ	ผลการทดสอบที่คาดหวัง
	C_DRS_4_15.1_2	061	สร้างสัญญาหลักประกันแบบหลักประกัน 1 หลักประกัน ค้ำประกันหลายบัญชีเงินกู้ และผูกความสัมพันธ์ระหว่างหลักประกัน บัญชีเงินกู้ และสัญญาหลักประกัน	- สามารถผูกความสัมพันธ์ระหว่างหลักประกัน 1 ชิ้น กับบัญชีเงินกู้หลายบัญชีได้
	C_DRS_4_15.1_2	062	สร้างสัญญาหลักประกันแบบจำนองเพิ่มหลักทรัพย์โดยไม่เพิ่มวงเงิน และผูกความสัมพันธ์กับสัญญาหลัก	- สามารถผูกความสัมพันธ์แบบแบบจำนองเพิ่มหลักทรัพย์โดยไม่เพิ่มวงเงิน
	C_DRS_4_15.1_2	063	สร้างสัญญาหลักประกันแบบเพิ่มวงเงินจำนอง เนื่องจากหลักประกันมีมูลค่าเพิ่มขึ้น และผูกความสัมพันธ์กับสัญญาหลัก	- สามารถผูกความสัมพันธ์แบบเพิ่มวงเงินจำนอง - วงเงินรวมของทั้งสัญญาถูกต้อง
	C_DRS_4_15.1_2	064	สร้างสัญญาหลักประกันแบบลดวงเงินจำนอง เนื่องจากมีการไถ่ถอนหลักประกันบางส่วนออกไป และผูกความสัมพันธ์กับสัญญาหลัก	- สามารถผูกความสัมพันธ์แบบลดวงเงินจำนอง - วงเงินรวมของทั้งสัญญาถูกต้อง
	C_DRS_4_15.1_2	065	สร้างสัญญาหลักประกันแบบจำนองลำดับสอง	- สามารถผูกความสัมพันธ์แบบจำนองลำดับสอง
R_DRS_4_15.4_1	C_DRS_4_15.4_1	066	ตรวจสอบการคำนวณการกันสำรองตามชั้นหนี้	- คำนวณสำรองตามชั้นหนี้ได้ถูกต้องตามเกณฑ์ที่กำหนด

Copyright by Mahidol University

Req.#	Test Case #		เงื่อนไขในการทดสอบ	ผลการทดสอบที่คาดหวัง
R_DRS_4_15.1_3	C_DRS_4_15.1_3	067	เปลี่ยนสถานะบัญชีเป็นอยู่ระหว่างการดำเนินคดี	- สามารถเปลี่ยนสถานะทางกฎหมาย เป็น - อยู่ระหว่างดำเนินคดีได้
	C_DRS_4_15.1_3	068	เปลี่ยนสถานะบัญชีเป็นศาลพิพากษาแล้ว	- สามารถเปลี่ยนสถานะทางกฎหมาย เป็น - ศาลพิพากษาได้ - Field "Court Max Bal Cal" เป็น Mandatory - ระบบคำนวณดอกเบี้ยจากค่า "Last Balance to Accrue"
	C_DRS_4_15.1_3	069	เปลี่ยนสถานะบัญชีเป็นยึดทรัพย์/บังคับคดี	- สามารถเปลี่ยนสถานะทางกฎหมาย เป็น - บังคับคดี/ยึดทรัพย์ได้ - Field "Court Max Bal Cal" ยังแสดงค่า - ระบบคำนวณดอกเบี้ยจากค่า "Last Balance to Accrue"
	C_DRS_4_15.1_3	070	เปลี่ยนสถานะบัญชีเป็นขายทอดตลาดได้แล้ว	- สามารถเปลี่ยนสถานะทางกฎหมาย เป็น - ขายทอดตลาดได้แล้ว - ระบบคำนวณดอกเบี้ยจากค่าเงินต้นคงค้าง (Ledger Balance) - เปลี่ยนลำดับการตัดชำระหนี้ (Payment Grid) เป็นตัดชำระเงินต้นก่อน
	C_DRS_4_15.1_3	071	Override ชั้นหนี้	- สามารถทำการ Override ชั้นหนี้ได้ - ระบบคำนวณการกันสำรองตามชั้นหนี้ Override

Copyright by Mahidol University

Req.#	**Test Case #**		**เงื่อนไขในการทดสอบ**	**ผลการทดสอบที่คาดหวัง**
	C_DRS_4_15.1_3	072	เปลี่ยนสถานะบัญชีเป็นประนอมหนี้ตามคำพิพากษา	- สามารถเปลี่ยนสถานะทางกฎหมาย เป็น - ประนอมหนี้ตามคำพิพากษา - ระบบคำนวณดอกเบี้ยจากค่าเงินต้นคงค้าง (Ledger Balance) - เปลี่ยนลำดับการตัดชำระหนี้ (Payment Grid) เป็นตัดชำระเงินต้นก่อน
R_DRS_4_15.1_4	C_DRS_4_15.1_4	073	ทบทวนการประเมินราคาหลักประกัน	- สามารถแก้ไขมูลค่าหลักประกันได้
	C_DRS_4_15.1_4	074	ใส่ค่า Non CRM Flag = Y	- กรณีหลักประกันมีค่า Non CRM Flag = Y จะไม่นำมาคำนวณการกันสำรอง
	C_DRS_4_15.1_4	075	ใส่ค่า Non CRM Flag = N	- กรณีหลักประกันมีค่า Non CRM Flag N จะนำมาคำนวณการกันสำรอง

Copyright by Mahidol University

Req.#	Test Case #		เงื่อนไขในการทดสอบ	ผลการทดสอบที่คาดหวัง
	C_DRS_4_15.1_4	076	ไถ่ถอนหลักประกันบางส่วน	- สามารถไถ่ถอนหลักประกันบางส่วนได้ - ระบบนำมูลค่าหลักประกันไปคำนวณการกันสำรองได้ถูกต้อง
R_DRS_4_15.1_5	C_DRS_4_15.1_5	077	สร้างกรมธรรม์ ประเภทการประกันอัคคีภัย ลูกค้าชำระเบี้ยประกันเอง	- สามารถสร้างกรมธรรม์ประกันภัยได้ตามเงื่อนไข
	C_DRS_4_15.1_5	078	สร้างกรมธรรม์ ประเภทการประกันอัคคีภัย ธนาคารเป็นผู้ชำระค่าเบี้ยประกัน	- สามารถสร้างกรมธรรม์ประกันภัยได้ตามเงื่อนไข
	C_DRS_4_15.1_5	079	สร้างกรมธรรม์ ประเภทการประกันอัคคีภัย ลูกค้าชำระค่าเบี้ยประกันบางส่วน	- สามารถสร้างกรมธรรม์ประกันภัยได้ตามเงื่อนไข

Copyright by Mahidol University

TEST SCRIPT						
Scenario Number : R_DRS_4_15_001						
Business Scenario : ตรวจสอบการคำนวณสำรองตามชั้นหนี้ แบบ One to One **หลักประกันประเภทที่ดิน/ที่ดินพร้อมสิ่งปลูกสร้าง/Subtype = โฉนด มีการทบทวนการประเมินราคา ไม่เกิน 3 ปี นับจากวันที่ประเมินครั้งล่าสุด สถานะทางกฎหมาย = ปกติ**						
Version	**Release date**	**Description of Version/Page (-s)**	**Requirement no.**	**Author**	**Comment**	
				ใส่ชื่อผู้ทดสอบด้วย		
(ให้กรอกด้านข้างแทน)						
Action #	**Simulated Date**	**Test Case #**	**Action and Data**	**ผลการทดสอบที่คาดหวัง**	**ผลการทดสอบ OK / FR**	**ผลสรุปของแต่ละ Case**
	Day 0		CIF number = xxxx, Customer Type = Personal, Customer Code = xxxxx สาขา XXXX			

Copyright by Mahidol University

Action #	Simulated Date	Test Case #	Action and Data	ผลการทดสอบที่คาดหวัง	ผลการทดสอบ OK / FR	ผลสรุปของแต่ละ Case
	Day 0		สร้างบัญชีเงินกู้ Due ทุกสิ้นเดือน 1MAE			
1	Day 1		**เข้าสู่ระบบ Web CSR สาขา**	xxxxxx		
2	Day 1	001	สร้างหลักประกัน ประเภท Collateral Type = ที่ดิน/ที่ดินพร้อมสิ่งปลูกสร้าง Subtype = โฉนด ประเภทที่ดิน เปล่า			
			Menu Create Collateral			
			Collateral Type	ที่ดิน/ที่ดินพร้อมสิ่งปลูกสร้าง		
			Collateral Sub type	โฉนด		
			Create	คลิกปุ่ม		
			Tab General			
			หลักประกันต้องการการประกันภัย	Yes		

Copyright by Mahidol University

Action #	Simulated Date	Test Case #	Action and Data	ผลการทดสอบที่คาดหวัง	ผลการทดสอบ OK / FR	ผลสรุปของแต่ละ Case
			พนักงานผู้ดูแลบัญชี	แสดง ID ของผู้สร้างหลักประกัน		
			สาขาเจ้าของหลักประกัน	แสดงสาขาเจ้าของหลักประกัน		
			ศูนย์ต้นทุน	Default ค่าตาม สาขาเจ้าของหลักประกัน		
			หน่วยงานดูแลลูกค้า	Default ค่าตาม ศูนย์ต้นทุน		
			สกุลเงิน	Default - THB		
			Continue	คลิกปุ่ม		
			Tab Collateral Information			
			Land and Property Information			
			Collateral Type	ที่ดิน/ที่ดินพร้อมสิ่งปลูกสร้าง		
			Collateral Sub type	โฉนด		
			เลขที่โฉนด/นส.3ก./ตราจอง	10101		
			จังหวัด	กรุงเทพมหานคร		
			อำเภอ	ยานนาวา		
			ตำบล	บางโพงพาง		

Copyright by Mahidol University

Action #	Simulated Date	Test Case #	Action and Data	ผลการทดสอบที่คาดหวัง	ผลการทดสอบ OK / FR	ผลสรุปของแต่ละ Case
			พื้นที่(ไร่-งาน-ตารางวา-เศษของตารางวา)	10-0-00-00		
			ราคาประเมินรวม	3,000,000.00		
			วันที่ประเมินราคา	DD/MM/YYYY		
			ประเมินโดย	สาขา		
			สาขา/ศูนย์ที่ทำการประเมินราคา	เลือกสาขาที่ประเมินราคา		
			Non CRM Flag	N		
			Continue	คลิกปุ่ม		
			Tab Collateral Owners			
			Add			
			Continue	คลิกปุ่ม		
			เจ้าของกรรมสิทธิ์หลักประกัน	CIF Number xxxx		
			Tab Insurance			
			ไม่มี			
			Continue	คลิกปุ่ม		
			Submit			
			สร้างหลักประกันสำเร็จ	Collateral Number xxxxxx		

Copyright by Mahidol University

Action #	Simulated Date	Test Case #	Action and Data	ผลการทดสอบที่คาดหวัง	ผลการทดสอบ OK / FR	ผลสรุปของแต่ละ Case
3	Day 1	002	สร้างหลักประกัน ประเภท Collateral Type = ที่ดิน/ที่ดินพร้อมสิ่งปลูกสร้าง Subtype = โฉนด ประเภทที่ดินเปล่า โดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว			
			Menu Create Collateral			
			Collateral Type	ที่ดิน/ที่ดินพร้อมสิ่งปลูกสร้าง		
			Collateral Sub type	โฉนด		
			Tab General			
			หลักประกันต้องการการประกันภัย	Yes		
			พนักงานผู้ดูแลบัญชี	แสดง ID ของผู้สร้างหลักประกัน		
			สาขาเจ้าของหลักประกัน	แสดงสาขาเจ้าของหลักประกัน		
			ศูนย์ต้นทุน	Default ค่าตาม สาขาเจ้าของหลักประกัน		
			หน่วยงานดูแลลูกค้า	Default ค่าตาม ศูนย์ต้นทุน		
			สกุลเงิน	Default - THB		
			Continue	คลิกปุ่ม		

Copyright by Mahidol University

Action #	Simulated Date	Test Case #	Action and Data	ผลการทดสอบที่คาดหวัง	ผลการทดสอบ OK / FR	ผลสรุปของแต่ละ Case
			Tab Collateral Information			
			Land and Property Information			
			Collateral Type	ที่ดิน/ที่ดินพร้อมสิ่งปลูกสร้าง		
			Collateral Sub type	โฉนด		
			เลขที่โฉนด/นส.3ก./ตราจอง	10101		
			จังหวัด	กรุงเทพมหานคร		
			อำเภอ	ยานนาวา		
			ตำบล	บางโพงพาง		
			พื้นที่(ไร่-งาน-ตารางวา-เศษของตารางวา)	10-0-00-00		
			ราคาประเมินรวม	3,000,000.00		
			วันที่ประเมินราคา	DD/MM/YYYY		
			ประเมินโดย	สาขา		
			สาขา/ศูนย์ที่ทำการประเมินราคา	เลือกสาขาที่ประเมินราคา		
			Non CRM Flag	N		
			Continue	คลิกปุ่ม		

Copyright by Mahidol University

Action #	Simulated Date	Test Case #	Action and Data	ผลการทดสอบที่คาดหวัง	ผลการทดสอบ OK / FR	ผลสรุปของแต่ละ Case
			Tab Collateral Owners			
			Add			
			เจ้าของกรรมสิทธิ์หลักประกัน	CIF Number xxxx		
			Continue	คลิกปุ่ม		
			Tab Insurance			
			ไม่มี			
			Continue	คลิกปุ่ม		
			Add			
			เจ้าของกรรมสิทธิ์หลักประกัน	CIF Number xxxx		
			Submit			
			ไม่สามารถสร้างหลักประกันได้	ระบบแสดงข้อความเตือนเนื่องจากสร้างหลักประกันโดยมี Primary Key ซ้ำกับหลักประกันที่มีอยู่แล้ว		

Copyright by Mahidol University

Action #	Simulated Date	Test Case #	Action and Data	ผลการทดสอบที่คาดหวัง	ผลการทดสอบ OK / FR	ผลสรุปของแต่ละ Case
4	Day 1	058	สร้างสัญญาหลักประกัน แบบหลักประกัน 1 หลักประกัน ค้ำประกัน 1 บัญชีเงินกู้ และผูกความสัมพันธ์ระหว่างหลักประกัน บัญชีเงินกู้ และสัญญาหลักประกัน			
			สร้าง Contract และผูกกับหลักประกันและบัญชีเงินกู้ - Menu Create Contract			
			Tab General			
			วันที่ลงนามสัญญาหลักประกัน	DD/MM/YYYY		
			สาขาเจ้าของหลักประกัน	แสดงสาขาเจ้าของหลักประกัน		
			ศูนย์ต้นทุน	Default ค่าตาม สาขาเจ้าของหลักประกัน		
			หน่วยงานดูแลลูกค้า	Default ค่าตาม ศูนย์ต้นทุน		
			ประเภทสัญญาหลักประกัน	Master		

Copyright by Mahidol University

Action #	Simulated Date	Test Case #	Action and Data	ผลการทดสอบที่คาดหวัง	ผลการทดสอบ OK / FR	ผลสรุปของแต่ละ Case
			มูลจำนอง/จำนำ/ค้ำประกัน	3,000,000.00		
			สถานะบัญชี	Pledge		
			Continue	คลิกปุ่ม		
			Tab Customer Link			
			Add			
			เจ้าของสัญญา/คนที่ลงนามในสัญญา	CIF Number xxxx		
			Continue	คลิกปุ่ม		
			Tab Account Link			
			Add			
			บัญชีเงินกู้ที่ผูกกับสัญญานี้	Loan Account Number = 8xxxxxxxxxx		
			Continue	คลิกปุ่ม		
			Tab Collateral Link			
			Add			
			หลักประกันที่ผูกกับสัญญานี้	เลขทะเบียนหลักประกัน = xxxxx		
			สถานะบัญชี	P - จำนำ/จำนอง		
			Status Date	วันที่ลงนามในสัญญาหลักประกัน		

Copyright by Mahidol University

Action #	Simulated Date	Test Case #	Action and Data	ผลการทดสอบที่คาดหวัง	ผลการทดสอบ OK / FR	ผลสรุปของแต่ละ Case
			Submit			
			ผูกความสัมพันธ์สำเร็จ	Contract Number = xxxx		
5	วันที่ 1 เดือน 2		**เข้าสู่ระบบ Web CSR สาขา**	xxxxxx		
6	วันที่ 1 เดือน 2	066	ตรวจสอบการคำนวณการกันสำรองตามชั้นหนี้			
			Account Number	เลขที่บัญชีเงินกู้		
			Tab Delinquency			
			Tab ย่อย Provision / Reclassification			
			Provision Processing			
			Provision Balance	แสดงผลการคำนวณการกันสำรอง		
			Provision Percentage	1.00000		
			Reclassification Processing			
			C Class - Aging	1		
7	วันที่ 1 เดือน 3		**เข้าสู่ระบบ Web CSR สาขา**	xxxxxx		
8	วันที่ 1 เดือน 3	066	ตรวจสอบการคำนวณการกันสำรองตามชั้นหนี้			

Copyright by Mahidol University

Action #	Simulated Date	Test Case #	Action and Data	ผลการทดสอบที่คาดหวัง	ผลการทดสอบ OK / FR	ผลสรุปของแต่ละ Case
			Account Number	เลขที่บัญชีเงินกู้		
			Tab Delinquency			
			Tab ย่อย Provision / Reclassification			
			Provision Processing			
			Provision Balance	แสดงผลการคำนวณการกันสำรอง		
			Provision Percentage	1.00000		
			Reclassification Processing			
			C Class - Aging	1		
9	วันที่ 1 เดือน 4		**เข้าสู่ระบบ Web CSR สาขา**	xxxxxx		
10	วันที่ 1 เดือน 4	066	ตรวจสอบการคำนวณการกันสำรองตามชั้นหนี้			
			Account Number	เลขที่บัญชีเงินกู้		
			Tab Delinquency			
			Tab ย่อย Provision / Reclassification			
			Provision Processing			
			Provision Balance	แสดงผลการคำนวณการกันสำรอง		

Copyright by Mahidol University

Action #	Simulated Date	Test Case #	Action and Data	ผลการทดสอบที่คาดหวัง	ผลการทดสอบ OK / FR	ผลสรุปของแต่ละ Case
			Provision Percentage	2.00000		
			Reclassification Processing			
			C Class - Aging	2		
11	วันที่ 1 เดือน 5		**เข้าสู่ระบบ Web CSR สาขา**	xxxxxx		
12	วันที่ 1 เดือน 5	066	ตรวจสอบการคำนวณการกันสำรองตามชั้นหนี้			
			Account Number	เลขที่บัญชีเงินกู้		
			Tab Delinquency			
			Tab ย่อย Provision / Reclassification			
			Provision Processing			
			Provision Balance	แสดงผลการคำนวณการกันสำรอง		
			Provision Percentage	2.00000		
			Reclassification Processing			
			C Class - Aging	2		

Copyright by Mahidol University

Action #	Simulated Date	Test Case #	Action and Data	ผลการทดสอบที่คาดหวัง	ผลการทดสอบ OK / FR	ผลสรุปของแต่ละ Case
13	วันที่ 1 เดือน 6		**เข้าสู่ระบบ Web CSR สาขา**	xxxxxx		
14	วันที่ 1 เดือน 6	066	ตรวจสอบการคำนวณการกันสำรองตามชั้นหนี้			
			Account Number	เลขที่บัญชีเงินกู้		
			Tab Delinquency			
			Tab ย่อย Provision / Reclassification			
			Provision Processing			
			Provision Balance	แสดงผลการคำนวณการกันสำรอง		
			Provision Percentage	100.00000		
			Reclassification Processing			
			C Class - Aging	3		
15	วันที่ 1 เดือน 7		**เข้าสู่ระบบ Web CSR สาขา**	xxxxxx		
16	วันที่ 1 เดือน 7	066	ตรวจสอบการคำนวณการกันสำรองตามชั้นหนี้			
			Account Number	เลขที่บัญชีเงินกู้		
			Tab Delinquency			

Copyright by Mahidol University

Action #	Simulated Date	Test Case #	Action and Data	ผลการทดสอบที่คาดหวัง	ผลการทดสอบ OK / FR	ผลสรุปของแต่ละ Case
			Tab ย่อย Provision / Reclassification			
			Provision Processing			
			Provision Balance	แสดงผลการคำนวณการกันสำรอง		
			Provision Percentage	100.00000		
			Reclassification Processing			
			C Class - Aging	3		
17	วันที่ 1 เดือน 8		**เข้าสู่ระบบ Web CSR สาขา**	xxxxxx		
18	วันที่ 1 เดือน 8	066	ตรวจสอบการคำนวณการกันสำรองตามชั้นหนี้			
			Account Number	เลขที่บัญชีเงินกู้		
			Tab Delinquency			
			Tab ย่อย Provision / Reclassification			
			Provision Processing			
			Provision Balance	แสดงผลการคำนวณการกันสำรอง		
			Provision Percentage	100.00000		

Copyright by Mahidol University

Action #	Simulated Date	Test Case #	Action and Data	ผลการทดสอบที่คาดหวัง	ผลการทดสอบ OK / FR	ผลสรุปของแต่ละ Case
			Reclassification Processing			
			C Class - Aging	3		
19	วันที่ 1 เดือน 9		**เข้าสู่ระบบ Web CSR สาขา**	xxxxxx		
20	วันที่ 1 เดือน 9	066	ตรวจสอบการคำนวณการกันสำรองตามชั้นหนี้			
			Account Number	เลขที่บัญชีเงินกู้		
			Tab Delinquency			
			Tab ย่อย Provision / Reclassification			
			Provision Processing			
			Provision Balance	แสดงผลการคำนวณการกันสำรอง		
			Provision Percentage	100.00000		
			Reclassification Processing			
			C Class - Aging	3		
21	วันที่ 1 เดือน 10		**เข้าสู่ระบบ Web CSR สาขา**	xxxxxx		
22	วันที่ 1 เดือน 10	066	ตรวจสอบการคำนวณการกันสำรองตามชั้นหนี้			

Copyright by Mahidol University

Action #	Simulated Date	Test Case #	Action and Data	ผลการทดสอบที่คาดหวัง	ผลการทดสอบ OK / FR	ผลสรุปของแต่ละ Case
			Account Number	เลขที่บัญชีเงินกู้		
			Tab Delinquency			
			Tab ย่อย Provision / Reclassification			
			Provision Processing			
			Provision Balance	แสดงผลการคำนวณการกันสำรอง		
			Provision Percentage	100.00000		
			Reclassification Processing			
			C Class - Aging	4		
22	วันที่ 1 เดือน 11		**เข้าสู่ระบบ Web CSR สาขา**	xxxxxx		
23	วันที่ 1 เดือน 11	066	ตรวจสอบการคำนวณการกันสำรองตามชั้นหนี้			
			Account Number	เลขที่บัญชีเงินกู้		
			Tab Delinquency			
			Tab ย่อย Provision / Reclassification			
			Provision Processing			
			Provision Balance	แสดงผลการคำนวณการกันสำรอง		

Copyright by Mahidol University

Action #	Simulated Date	Test Case #	Action and Data	ผลการทดสอบที่คาดหวัง	ผลการทดสอบ OK / FR	ผลสรุปของแต่ละ Case
			Provision Percentage	100.00000		
			Reclassification Processing			
			C Class - Aging	4		
22	สิ้นเดือน 12		**เข้าสู่ระบบ Web CSR สาขา**			
23	สิ้นเดือน 12	072	ใส่ค่า Non CRM Flag = Y			
			Menu Get Collateral			
			Tab Collateral Information			
			Land and Property Information			
			Non CRM Flag	Y		
24	วันที่ 1 เดือน 12		**เข้าสู่ระบบ Web CSR สาขา**	xxxxxx		
25	วันที่ 1 เดือน 12	066	ตรวจสอบการคำนวณการกันสำรองตามชั้นหนี้			
			Account Number	เลขที่บัญชีเงินกู้		
			Tab Delinquency			

Copyright by Mahidol University

Action #	Simulated Date	Test Case #	Action and Data	ผลการทดสอบที่คาดหวัง	ผลการทดสอบ OK / FR	ผลสรุปของแต่ละ Case
			<u>Tab ย่อย Provision / Reclassification</u>			
			Provision Processing			
			Provision Balance	แสดงผลการคำนวณการกันสำรอง		
			Provision Percentage	100.00000		
			Reclassification Processing			
			C Class - Aging	4		
26	สิ้นเดือน 12		**เข้าสู่ระบบ Web CSR สาขา**			
27	สิ้นเดือน 12	073	ใส่ค่า Non CRM Flag = Y			
			Menu Get Collateral			
			Tab Collateral Information			
			Land and Property Information			
			Non CRM Flag	N		
28	วันที่ 1 เดือน 13		**เข้าสู่ระบบ Web CSR สาขา**	xxxxxx		
29	วันที่ 1 เดือน 13	066	ตรวจสอบการคำนวณการกันสำรองตามชั้นหนี้			

Copyright by Mahidol University

Action #	Simulated Date	Test Case #	Action and Data	ผลการทดสอบที่คาดหวัง	ผลการทดสอบ OK / FR	ผลสรุปของแต่ละ Case
			Account Number	เลขที่บัญชีเงินกู้		
			Tab Delinquency			
			Tab ย่อย Provision / Reclassification			
			Provision Processing			
			Provision Balance	แสดงผลการคำนวณการกันสำรอง		
			Provision Percentage	100.00000		
			Reclassification Processing			
			C Class - Aging	4		
30	วันที่ 1 เดือน 14		**เข้าสู่ระบบ Web CSR สาขา**	xxxxxx		
31	วันที่ 1 เดือน 14	066	ตรวจสอบการคำนวณการกันสำรองตามชั้นหนี้			
			Account Number	เลขที่บัญชีเงินกู้		
			Tab Delinquency			
			Tab ย่อย Provision / Reclassification			
			Provision Processing			
			Provision Balance	แสดงผลการคำนวณการกันสำรอง		
			Provision Percentage	100.00000		

Copyright by Mahidol University

Action #	Simulated Date	Test Case #	Action and Data	ผลการทดสอบที่คาดหวัง	ผลการทดสอบ OK / FR	ผลสรุปของแต่ละ Case
			Reclassification Processing			
			C Class - Aging	4		
32	วันที่ 1 เดือน 15		**เข้าสู่ระบบ Web CSR สาขา**	xxxxxx		
33	วันที่ 1 เดือน 15	066	ตรวจสอบการคำนวณการกันสำรองตามชั้นหนี้			
			Account Number	เลขที่บัญชีเงินกู้		
			Tab Delinquency			
			Tab ย่อย Provision / Reclassification			
			Provision Processing			
			Provision Balance	แสดงผลการคำนวณการกันสำรอง		
			Provision Percentage	100.00000		
			Reclassification Processing			
			C Class - Aging	5		

Copyright by Mahidol University

APPENDIX B

แบบประเมินผลหลังการปรับปรุงกระบวนทดสอบระบบโดยนำเทคนิค Black-box testing มาใช้งาน กรณีศึกษาระบบบริหารหลักประกัน

ส่วนที่ 1 : ข้อมูลทั่วไปของผู้ตอบแบบสอบถาม

1.1 เพศ ❑ ชาย ❑ หญิง

1.2 อายุ ❑ น้อยกว่าหรือเท่ากับ 25 ปี ❑ 26-35 ปี ❑ 36-45 ปี
❑ 46-55 ปี ❑ 56 ปีขึ้นไป

1.3 การศึกษา ❑ ปริญญาตรี ❑ ปริญญาโท ❑ ปริญญาเอก

1.4 อายุการทำงาน ❑ น้อยกว่าหรือเท่ากับ 5 ปี ❑ 6-10 ปี ❑ 11-15 ปี ❑ 16-20 ปี
❑ 21-25 ปี ❑ 26-30 ปี ❑ 30 ปีขึ้นไป

1.5 ตำแหน่ง
❑ พนักงานปฏิบัติงาน
❑ ผู้บริหาร

ส่วนที่ 2 : ข้อมูลการแสดงความคิดเห็นของผู้ใช้งานหลังการปรับปรุงกระบวนทดสอบระบบโดยนำเทคนิค Black-box testing กรณีศึกษาระบบบริหารหลักประกัน

กรุณาทำเครื่องหมาย ✓ ในช่องที่ตรงกับความคิดเห็นของท่านมากที่สุด

คำถาม	น้อยที่สุด (1)	น้อย (2)	ปานกลาง (3)	มาก (4)	มากที่สุด (5)
1.ท่านคิดว่าการนำเทคนิค Black-box testing มาใช้งานช่วยให้การทดสอบมีความครอบคลุมฟังก์ชั่นงานเพียงใด					
2.ท่านคิดว่าการนำเทคนิค Black-box testing มาใช้งานช่วยลดความซ้ำซ้อนในการทดสอบเพียงใด					
3.ท่านคิดว่าการนำเทคนิค Black-box testing มาใช้งานการทดสอบระบบมีความถูกต้องและแม่นยำมากขึ้น					
4.ท่านคิดว่าการนำเทคนิค Black-box testing มาใช้งานช่วยลดเวลาการทำงานเพียงใด					
5.ท่านคิดว่าการนำเทคนิค Black-box testing มาใช้งานช่วยให้การทำงานมีระเบียบแบบแผนมากขึ้นและง่ายต่อการประสานงานภายในหน่วยงาน					

Copyright by Mahidol University

คำถาม	น้อยที่สุด (1)	น้อย (2)	ปานกลาง (3)	มาก (4)	มากที่สุด (5)
6.ท่านคิดว่าการนำเทคนิค Black-box testing มาใช้งานช่วยให้การรายงานผลต่อผู้ใช้งานมีความชัดเจนและไปในแนวทางเดียวกัน					
7.ท่านคิดว่าการนำเทคนิค Black-box testing มาใช้งานช่วยให้ใช้ทรัพยากรในการทำงานลดลงเพียงใด					
8.ท่านคิดว่าการนำเทคนิค Black-box testing มาใช้งานช่วยลดต้นทุนในการทดสอบระบบเพียงใด					
9.ท่านคิดว่าการนำเทคนิค Black-box testing มาใช้งานทำให้สามารถใช้ทรัพยากรได้อย่างมีประสิทธิภาพมากขึ้นเพียงใด					
10.ท่านมีความพึงพอใจในการนำเทคนิค Black-box testing มาใช้งานเพียงใด					

Copyright by Mahidol University

BIOGRAPHY

NAME	Mr. Chumpol Ponrod
DATE OF BIRTH	8 April 1986
PLACE OF BIRTH	Bangkok, Thailand
INSTITUTIONS ATTENDED	Mahidol University, 2006-2009 Bachelor of Management (Management Information System) Mahidol University, 2010-2014 Master of Science (Technology of Information System Management)
HOME ADDRESS	121 M.1 Ratchaphrek Rd. Bang Khun Kong, Bang Kruai, Nonthaburi 11130 Tel: 0869760931 E-mail: chumpol.po@gmail.com

Copyright by Mahidol University