



**IMPLEMENTATION OF IMAGE ENCRYPTION-BASED LOGISTIC CHAOTIC MAP FOR
TRANSACTION PROOF IN WASTE BANK TRANSACTION SYSTEM WEB APPLICATION**

UNDERGRADUATE THESIS

**Submitted as one of the requirements to obtain
Sarjana Komputer (S.Kom.)**

By:

Cynthia Paramita

001201900047

**FACULTY OF COMPUTER SCIENCE
INFORMATION TECHNOLOGY STUDY PROGRAM**

CIKARANG

FEBRUARY 2023

PANEL OF EXAMINER APPROVAL

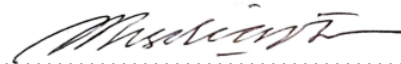
The Panel of Examiners declare that the undergraduate thesis entitled **Implementation of Image Encryption-Based Logistic Chaotic Map for Transaction Proof in Waste Bank Transaction System Web Application** that was submitted by Cynthia Paramita majoring in Information Technology from the Faculty of Computer Science. was assessed and approved to have passed the Oral Examination on 22 February 2023.

Panel of Examiner



.....
Cutifa Safitri B. CS., M.IT., Ph.D

Chair of Panel Examiner



.....
Rusdianto Roestam MSc., PhD.

Examiner I

STATEMENT OF ORIGINALITY

In my capacity as an active student of President University and as the author of the undergraduate thesis/final project/business plan (underline that applies) stated below:

Name : Cynthia Paramita
Student ID number : 001201900047
Study Program : Information Technology
Faculty : Computer Science

I hereby declare that my undergraduate thesis/final project/business plan entitled "Implementation of Image Encryption-Based Logistic Chaotic Map for Transaction Proof in Waste Bank Transaction System Web Application" is, to the best of my knowledge and belief, an original piece of work based on sound academic principles. If there is any plagiarism, including but not limited to Artificial Intelligence plagiarism, is detected in this undergraduate thesis/final project/business plan, I am willing to be personally responsible for the consequences of these acts of plagiarism, and accept the sanctions against these acts in accordance with the rules and policies of President University.

I also declare that this work, either in whole or in part, has not been submitted to another university to obtain a degree.

Cikarang, March 2023



(Cynthia Paramita)

Full name & signature

SCIENTIFIC PUBLICATION APPROVAL FOR ACADEMIC INTEREST

As a student of the President University, I, the undersigned:

Name : Cynthia Paramita
Student ID number : 001201900047
Study program : Information Technology

for the purpose of development of science and technology, certify, and approve to give President University a non-exclusive royalty-free right upon my final report with the title:

‘Implementation of Image Encryption-Based Logistic Chaotic Map for Transaction Proof in Waste Bank Transaction System Web Application’.

With this non-exclusive royalty-free right, President University is entitled to converse, to convert, to manage in a database, to maintain, and to publish my final report. There are to be done with the obligation from President University to mention my name as the copyright owner of my final report.

This statement I made in truth.

Cikarang, March 2023



(Cynthia Paramita)
Full name & signature

ADVISOR'S APPROVAL FOR PUBLICATION

As a lecturer of the President University, I, the undersigned:

Advisor's Name : Nur Hadisukmana, M.Sc
NIDN : 0423076302
Study program : Information Technology
Faculty : Computer Science

declare that following thesis:

Title of thesis : Implementation of Image Encryption-Based Logistic Chaotic Map for
Transaction Proof in Waste Bank Transaction System Web Application
Thesis author : Cynthia Paramita
Student ID number : 001201900047

will be published in **journal** / **institution's repository** / **proceeding** / **unpublish** (underline one that applies)

Cikarang, March 2023



(Nur Hadisukmana, M.Sc)
Advisor Full name & signature

IMPLEMENTATION OF IMAGE ENCRYPTION-BASED LOGISTIC CHAOTIC MAP FOR TRANSACTION PROOF IN WASTE BANK TRANSACTION SYSTEM WEB APPLICATION

ORIGINALITY REPORT

12%

SIMILARITY INDEX

12%

INTERNET SOURCES

1%

PUBLICATIONS

0%

STUDENT PAPERS

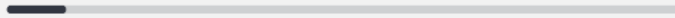
PRIMARY SOURCES

1	repository.president.ac.id Internet Source	11%
2	www.researchgate.net Internet Source	<1%
3	geoffboeing.com Internet Source	<1%
4	Pellicer-Lostao Carmen, Lopez-Ruiz Ricardo. "Chapter 4 Notions of Chaotic Cryptography: Sketch of a Chaos Based Cryptosystem", IntechOpen, 2012 Publication	<1%
5	byjusexamprep.com Internet Source	<1%
6	ir.lib.uwo.ca Internet Source	<1%
7	unsworks.unsw.edu.au Internet Source	<1%

GPT ZERO PLAGIARISM DETECTOR RESULT

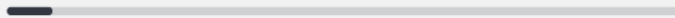
Stats

Average Perplexity Score: 87.778



A document's perplexity is a measurement of the randomness of the text

Burstiness Score: 68.024



A document's burstiness is a measurement of the variation in perplexity

Your sentence with the highest perplexity, "Nowadays proof of transaction has many forms.", has a perplexity of: 406

ABSTRACT

Garbage often refers to residues that are undesirable or useless to humans after the completion of household activities and processes. Garbage can cause many problems, including health, environmental and other issues, so proper waste management is important. Therefore, disposal of household waste is necessary. Waste banks are one sort of waste management solution. A waste bank is a concept that gathers dry, sorted rubbish; it functions similarly to a bank, except it takes in waste rather than cash. A set sum of money is awarded for the saved garbage, which is then sold to cooperative partners after being weighed.

Within the waste bank, different types of transaction processes take place, including depositing waste, requesting withdrawals of savings funds, the sale of collected waste, etc. Each transaction, whether inbound or outbound, must be recorded and accompanied by valid evidence. Proof of transaction is important to ensure that the transaction is valid and genuine. This allows both the payer and the payee to know that a transaction is taking place. Therefore, it must be stored properly and collected efficiently.

There are many forms of proof of transaction today. Can be used for printing, images, soft files, etc. These documents contain personal information about the recipient and the transaction itself. Therefore, it is important to ensure data security so that unauthorized users cannot access it. This thesis is created to design a waste bank web application that helps manage the transactional activities of a waste bank system and improves the security of the proofs of transactions image with image encryption.

ACKNOWLEDGMENTS

I want to express my gratitude to those that support and encourage me during the completion of this final project as one of the requirements for fulfilling the Bachelor of Science degree. I would want to take this chance to thank the following people:

1. The Almighty God, whose mercy and strength enable me to complete this thesis.
2. My family and parents, who are my biggest supporters.
3. My thesis advisor, Mr. Nur Hadisukmana, M.Sc., who offers suggestions and direction for finishing this thesis and studies.
4. Every single computing lecturer who helped me out with information and advice while I was a university student.
5. The entire Waste Bank development team, who constantly encourages and assists one another in creating this program.
6. All of my friends at President University who have shared their knowledge and experience with me.

TABLE OF CONTENTS

DEDICATION	ii
ACKNOWLEDGMENTS	iii
TABLE OF CONTENTS.....	iv
LIST OF TABLES	vii
LIST OF FIGURES	viii
CHAPTER 1	1
1.1 Background	1
1.2 Problem Statement	2
1.3 Thesis Objective.....	2
1.4 Scope and Limitations.....	3
1.5 Thesis Methodology.....	3
1.6 Thesis Outline	5
CHAPTER 2	7
2.1 Cryptography.....	7
2.2 Chaotic Cryptography	8
2.3 Image Encryption	9
2.4 Logistic Map	10
2.5 Related Works	11
2.5.1 A Simple Method for Image Encryption Using Chaotic Logistic Map.	11
2.5.2 A Digital Image Encryption Algorithm Based A Composition of Two Chaotic Logistic Maps.....	12
2.6 Comparison Overview.....	13
CHAPTER 3	14
3.1 System Overview	14

3.2	Functional Analysis.....	14
3.3	Use Case Diagram.....	15
3.4	Use Case Narrative.....	16
3.5	Activity Diagram.....	33
CHAPTER 4		41
4.1	User Interface Design.....	41
4.1.1	Login Page	41
4.1.2	Waste Collection Page	42
4.1.3	Waste Collection Detail Page	43
4.1.4	Withdrawal Request Page	44
4.1.5	Customer Credit Page	45
4.1.6	Customer Debit Page	46
4.1.7	Vendor Transaction Page.....	47
4.2	Physical Design.....	48
4.3	Database Design.....	49
CHAPTER 5		50
5.1	User Interface Development.....	50
5.1.1	Login Page	50
5.1.2	Waste Collection Page	51
5.1.3	Waste Collection Detail Page	53
5.1.4	Customer Withdrawal Request Page	55
5.1.5	Customer Credit Page	58
5.1.6	Customer Debit Page	59
5.1.7	Vendor Transaction Page.....	60
5.2	Application Details.....	64
5.2.1	Configure Database Connection	64

5.2.2	Image Encryption and Decryption with Logistic Map	65
5.2.3	Login.....	71
5.2.4	Manage Waste Collection Data	72
5.2.5	Manage Collection Detail Data.....	79
5.2.6	Manage Customer Withdrawal Request	88
5.2.7	View Customer Credit	100
5.2.8	View Customer Debit	102
5.2.9	Manage Vendor Transaction.....	107
5.2.10	Logout.....	121
CHAPTER 6		123
6.1	Testing Environment.....	123
6.2	Testing Scenario.....	123
6.2.1	Login Testing Scenario	123
6.2.2	Waste Collection Testing Scenario	125
6.2.3	Waste Collection Detail Testing Scenario	128
6.2.4	Customer Withdrawal Request Testing Scenario	132
6.2.5	Customer Credit Testing Scenario	136
6.2.6	Customer Debit Testing Scenario	138
6.2.7	Vendor Transaction Testing Scenario.....	140
6.2.8	Logout Testing Scenario	146
CHAPTER 7		148
7.1	Conclusion.....	148
7.2	Future Work	148
REFERENCE		

LIST OF TABLES

Table	Page
Table 3.1 Functional Description Table.....	14
Table 3.2 Use Case Narrative - Login	16
Table 3.3 Use Case Narrative - Manage Waste Collection	17
Table 3.4 Use Case Narrative - Manage Waste Collection Detail.....	20
Table 3.5 Use Case Narrative - Manage Vendor Transaction	23
Table 3.6 Use Case Narrative - View Customer Credit.....	27
Table 3.7 Use Case Narrative - View Customer Debit.....	28
Table 3.8 Use Case Narrative - Manage Withdrawal Request	29
Table 3.9 Use Case Narrative - Logout	32
Table 4.1 Login Page Description	41
Table 4.2 Waste Collection Page Description	42
Table 4.3 Waste Collection Detail Page Description.....	43
Table 4.4 Customer Withdrawal Request Page Description.....	44
Table 4.5 Customer Credit Page Description	45
Table 4.6 Customer Debit Page Description.....	46
Table 4.7 Vendor Transaction Page Description	47
Table 4.8 Software Requirements.....	48
Table 4.9 Hardware Requirements	49
Table 6.1 Login Page Test Scenario	124
Table 6.2 Waste Collection Page Test Scenario	126
Table 6.3 Waste Collection Detail Testing Scenario	129
Table 6.4 Customer Withdrawal Request Testing Scenario	132
Table 6.5 Customer Credit Testing Scenario	137
Table 6.6 Customer Debit Testing Scenario	138
Table 6.7 Vendor Transaction Testing Scenario.....	140
Table 6.8 Logout Testing Scenario.....	146

LIST OF FIGURES

Figure	Page
Figure 1.1 Rapid Application Development Diagram	4
Figure 2.1 Image Encryption Algorithm.....	10
Figure 2.2 Logistic Map Equation	10
Figure 2.3 Bifurcation Diagram of Logistic Map	10
Figure 2.4 Histogram analysis: Both for the plain image and encrypted image from R, G, and B channels[1]	11
Figure 2.5 Histogram of the plain image and cipher image [5]	12
Figure 3.1 Use Case Diagram	15
Figure 3.2 Activity Diagram – Login	34
Figure 3.3 Activity Diagram - Manage Waste Collection: (a) Add new waste collection, (b) Delete waste collection data, and (c) Edit waste collection data ..	35
Figure 3.4 Activity Diagram - Manage Waste Collection Detail: (a) Add new waste collection detail, (b) Edit waste collection detail data, and (c) Delete waste collection detail data.....	36
Figure 3.5 Activity Diagram - Manage Vendor Transaction: (a) Add new vendor transaction, (b) Edit vendor transaction data, (c) Delete vendor transaction data, and (d) View transaction proof.....	37
Figure 3.6 Activity Diagram - View Customer Credit	38
Figure 3.7 Activity Diagram - View Customer Debit.....	38
Figure 3.8 Activity Diagram – Manage Withdrawal Request: (a) Add new customer withdrawal request, (b) Accept customer withdrawal request, (c) Decline customer withdrawal request	39
Figure 3.9 Activity Diagram - Logout	40
Figure 4.1 User Interface – Login Page	41
Figure 4.2 User Interface – Waste Collection Page	42
Figure 4.3 User Interface – Waste Collection Detail Page	43
Figure 4.4 User Interface – Customer Withdrawal Request	44
Figure 4.5 User Interface – Customer Credit Page	45
Figure 4.6 User Interface – Customer Debit Page	46

Figure 4.7 User Interface – Vendor Transaction	47
Figure 4.8 Database Scheme Design - Waste Bank System.....	49
Figure 5.1 Login Page.....	50
Figure 5.2 Waste Collection Page.....	51
Figure 5.3 Waste Collection Page - Add Waste Collection Modal	52
Figure 5.4 Waste Collection Page - Delete Confirmation Modal.....	52
Figure 5.5 Waste Collection Detail Page.....	53
Figure 5.6 Waste Collection Detail Page - Add Waste Collection Detail Modal.....	54
Figure 5.7 Waste Collection Detail Page - Edit Waste Collection Detail Modal	54
Figure 5.8 Waste Collection Detail Page - Deletion Confirmation Modal.....	55
Figure 5.9 Customer Withdrawal Request.....	56
Figure 5.10 Customer Withdrawal Request Page - Add Request Modal	56
Figure 5.11 Customer Withdrawal Request Page - Paid Request Modal	57
Figure 5.12 Customer Withdrawal Request Page - Decline Request Confirmation....	58
Figure 5.13 Customer Credit Page.....	58
Figure 5.14 Customer Debit Page.....	59
Figure 5.15 Customer Debit Page - View Transaction Proof Modal.....	60
Figure 5.16 Vendor Transaction Page	61
Figure 5.17 Vendor Transaction Page - Add Vendor Transaction Modal.....	62
Figure 5.18 Vendor Transaction Page - Edit Vendor Transaction Modal	62
Figure 5.19 Vendor Transaction Page - Delete Confirmation Modal.....	63
Figure 5.20 Vendor Transaction Page - View Transaction Proof Modal	64
Figure 5.21 init.js – Database Connection.....	65
Figure 5.22 generateKey() Function - Paid Customer Withdrawal Request Server Side	66
Figure 5.23 LogisticEncryption() Function (1) - Paid Customer Withdrawal Request Server Side.....	66
Figure 5.24 LogisticEncryption() Function (2) - Paid Customer Withdrawal Request Server Side.....	67
Figure 5.25 LogisticEncryption() Function (3) - Paid Customer Withdrawal Request Server Side.....	67
Figure 5.26 LogisticEncryption() Function (4) - Paid Customer Withdrawal Request	

Server Side.....	67
Figure 5.27 getImageMatrixAndResize() Function - Paid Customer Withdrawal	
Request Server Side.....	68
Figure 5.28LogisticDecryption() Function (1) - View Customer Debit Transaction	
Proof Server Side.....	69
Figure 5.29 LogisticDecryption() Function (2) - View Customer Debit Transaction .	69
Figure 5.30 LogisticDecryption() Function (3) - View Customer Debit Transaction .	70
Figure 5.31 LogisticDecryption() Function (4) - View Customer Debit Transaction .	70
Figure 5.32 getImageMatrix() Function - View Customer Debit Transaction	70
Figure 5.33 submitLogin() Function – Login Client Side	71
Figure 5.34 userLoginService() Function – Server Side	72
Figure 5.35 findUserbyUsernamePwd() Function - Login SQL Server Side.....	72
Figure 5.36 fetchData() Function (1)- View Waste Collection Client Side	73
Figure 5.37 fetchData() Function (2)- View Waste Collection Client Side	73
Figure 5.38 getWasteCollection() Function - View Waste Collection Server Side	74
Figure 5.39 getWasteCollectionData() Function (1) – View Waste Collection SQL	
Server Side.....	74
Figure 5.40 getWasteCollectionData() Function (2) – View Waste Collection SQL	
Server Side.....	75
Figure 5.41 submit() Function - Add Waste Collection Client Side.....	75
Figure 5.42 addWasteCollection() Function - Add Waste Collection Client Side	76
Figure 5.43 addWasteCollection() Function (1) - Add Waste Collection SQL Server	
Side	76
Figure 5.44 addWasteCollection() Function (2) - Add Waste Collection SQL Server	
Side	77
Figure 5.45 addWasteCollection() Function – Add Waste Collection SQL Server Side	
.....	77
Figure 5.46 handleDelete() Function – Delete Waste Collection Client Side	78
Figure 5.47 deleteWasteCollection() Function – Delete Waste Collection Client Side	
.....	78
Figure 5.48 deleteWasteCollection() Function – Delete Waste Collection Server Side	
.....	79

Figure 5.49 deleteWasteCollection() Function – Delete Waste Collection SQL Server Side	79
Figure 5.50 getWasteCollectionDetail() Function – View Waste Collection Detail Client Side	80
Figure 5.51 getDetailWasteCollection() Function – View Waste Collection Detail Server Side.....	81
Figure 5.52 getDetailWasteCollection() Function – ViewWaste Collection Detail SQL Server Side.....	81
Figure 5.53 submit() Function – Add Waste Collection Detail Client Side	82
Figure 5.54 addWasteCollectionDetail() Function – Add Waste Collection Detail Client Side	82
Figure 5.55 addWasteCollectionDetail() Function (1) – Add Waste Collection Detail Server Side.....	83
Figure 5.56 addWasteCollectionDetail() Function (2) – Add Waste Collection Detail Server Side.....	83
Figure 5.57 addWasteCollectionDetail() Function – Add Waste Collection Detail SQL Server Side.....	84
Figure 5.58 submit() Function – Edit Waste Collection Detail Client Side	84
Figure 5.59 editWasteCollectionDetail() Function – Edit Waste Collection Detail Client Side	85
Figure 5.60 updateWasteCollectionDetail() Function – Edit Waste Collection Detail Server Side.....	85
Figure 5.61 updateWasteCollectionDetail() Function – Edit Waste Collection Detail SQL Server Side	86
Figure 5.62 handleDelete() Function – Delete Waste Collection Detail Client Side ..	87
Figure 5.63 deleteWasteCollectionDetail() Function – Delete Waste Collection Detail Client Side	87
Figure 5.64 deleteWasteCollectionDetail() Function – Delete Waste Collection Detail Server Side.....	88
Figure 5.65 deleteWasteCollectionDetail() Function – Delete Waste Collection Detail SQL Server Side	88
Figure 5.66 fetchData() Function (1)- View Customer Withdrawal Request Client	

Side	89
Figure 5.67 fetchData() Function (2) - View Customer Withdrawal Request Client	
Side	90
Figure 5.68 getDebitData() Function - View Customer Withdrawal Request Server	
Side	91
Figure 5.69 getDebitData() Function (1) - View Customer Withdrawal Request SQL	
Server Side.....	91
Figure 5.70 getDebitData() Function (2) - View Customer Withdrawal Request SQL	
Server Side.....	92
Figure 5.71 submit() Function - Add Customer Withdrawal Request Client Side.....	93
Figure 5.72 validateRequestAmount() Function - Add Customer Withdrawal Request	
Client Side	93
Figure 5.73 addRequest() Function - Add Customer Withdrawal Request Client Side	
.....	94
Figure 5.74 addDebitData() Function - Add Customer Withdrawal Request Server	
Side	94
Figure 5.75 addDebit() Function - Add Customer Withdrawal Request SQL Server	
Side	95
Figure 5.76 submit() Function - Paid Customer Withdrawal Request Client Side.....	95
Figure 5.77 approveRequest() Function - Paid Customer Withdrawal Request Client	
Side	96
Figure 5.78 acceptRequestService() Function (1) - Paid Customer Withdrawal	
Request Server Side.....	96
Figure 5.79 acceptRequestService() Function (2) - Paid Customer Withdrawal	
Request Server Side.....	97
Figure 5.80 updateDebit() Function - Paid Customer Withdrawal Request SQL Server	
Side	97
Figure 5.81 addImageDebitKey() Function - Paid Customer Withdrawal Request SQL	
Server Side.....	98
Figure 5.82 handleDecline() Function - Decline Customer Withdrawal Request Client	
Side	98
Figure 5.83 declineRequest() Function - Decline Customer Withdrawal Request	

Client Side	99
Figure 5.84 updateDebitData() Function - Decline Customer Withdrawal Request Server Side.....	99
Figure 5.85 fetchData() Function (1) - View Customer Credit Client Side.....	100
Figure 5.86 fetchData() Function (2) - View Customer Credit Client Side.....	101
Figure 5.87 getCreditData() Function – View Customer Credit Server Side	101
Figure 5.88 getCreditData() Function (1) – View Customer Credit SQL Server Side	102
Figure 5.89 getCreditData() Function (2) – View Customer Credit SQL Server Side	102
Figure 5.90 fetchData() Function (1)- View Customer Debit Client Side.....	103
Figure 5.91 fetchData() Function (2)- View Customer Debit Client Side.....	104
Figure 5.92 viewTransactionProof() Function - View Customer Debit Transaction Proof Client Side	105
Figure 5.93 viewTransactionProof() Function (1) - View Customer Debit Transaction Proof Server Side.....	106
Figure 5.94 viewTransactionProof() Function (2) - View Customer Debit Transaction Proof Server Side.....	106
Figure 5.95 getImageDebitKey() Function - View Customer Debit Transaction Proof Server Side.....	107
Figure 5.96 fetchData() Function (1) - View Vendor Transaction Client Side	108
Figure 5.97 fetchData() Function (2) - View Vendor Transaction Client Side	108
Figure 5.98 getVendorTransactionData() Function – View Vendor Transaction Server Side	109
Figure 5.99 getVendorTransaction() Function (1) – View Vendor Transaction SQL Server Side.....	109
Figure 5.100 getVendorTransaction() Function (2) – View Vendor Transaction SQL Server Side.....	110
Figure 5.101 submit() Function – Add Vendor Transaction Client Side.....	111
Figure 5.102 addVendorTransaction() Function – Add Vendor Transaction Client Side	111
Figure 5.103 addVendorTransaction() Function (1) – Add Vendor Transaction Server	

Side	112
Figure 5.104 addVendorTransaction() Function (2) – Add Vendor Transaction Server Side	112
Figure 5.105 addVendorTransaction() Function – Add Vendor Transaction SQL Server Side.....	112
Figure 5.106 addImageVendorKey() Function – Add Vendor Transaction SQL Server Side	113
Figure 5.107 submit() Function – Edit Vendor Transaction Client Side.....	113
Figure 5.108 editVendorTransaction() Function – Edit Vendor Transaction Client Side	114
Figure 5.109 updateVendorTransaction() Function (1) – Edit Vendor Transaction Server Side.....	114
Figure 5.110 updateVendorTransaction() Function (2) – Edit Vendor Transaction Server Side.....	115
Figure 5.111 updateVendorTransaction() Function (3) – Edit Vendor Transaction Server Side.....	115
Figure 5.112 updateVendorTransaction() Function (4) – Edit Vendor Transaction Server Side.....	115
Figure 5.113 updateVendorTransaction() Function – Edit Vendor Transaction SQL Server Side.....	116
Figure 5.114 updateImageVendorKey() Function – Edit Vendor Transaction SQL Server Side.....	116
Figure 5.115 handleDelete() Function – Delete Vendor Transaction Client Side.....	117
Figure 5.116 deleteVendorTransaction() Function – Delete Vendor Transaction Client Side	117
Figure 5.117 deleteVendorTransaction() Function (1) – Delete Vendor Transaction Server Side.....	118
Figure 5.118 deleteVendorTransaction() Function (2) – Delete Vendor Transaction Server Side.....	118
Figure 5.119 deleteVendorTransaction() Function – Delete Vendor Transaction SQL Client Side	118
Figure 5.120 deleteImageVendorKey() Function – Delete Vendor Transaction SQL	

Client Side	119
Figure 5.121 viewTransactionProof() Function - View Vendor Transaction Proof	
Client Side	119
Figure 5.122 viewTransactionProof() Function (1) - View Vendor Transaction Proof	
Server Side.....	120
Figure 5.123 viewTransactionProof() Function (2) - View Vendor Transaction Proof	
Server Side.....	120
Figure 5.124 getImageVendorKey() Function - View Vendor Transaction Proof	
Server Side.....	121
Figure 5.125 handleLogout() Function – Logout Client Side.....	121
Figure 5.126 userLogoutService() Function – Logout Client Side.....	122
Figure 6.1 Login Page appear when accessing localhost:3000 – Login Testing	
Scenario	124
Figure 6.2 Redirected to dashboard page when login with correct credentials – Login	
Testing Scenario	125
Figure 6.3 Alert will show if login with incorrect credentials – Login Testing Scenario	
.....	125
Figure 6.4 Waste Collection Page appear when waste collection menu is click on the	
side bar – Waste Collection Testing Scenario.....	127
Figure 6.5 Waste Collection data shown based on the filter - Waste Collection Testing	
Scenario	127
Figure 6.6 Add Form appear when Add Waste Collection Button clicked - Waste	
Collection Testing Scenario	128
Figure 6.7 Conformation appears when Delete Button clicked - Waste Collection	
Testing Scenario	128
Figure 6.8 Detail Page appear when the Detail Button clicked - Waste Collection	
Detail Testing Scenario.....	130
Figure 6.9 Add Form appear when Add Detail Button clicked - Waste Collection	
Detail Testing Scenario.....	131
Figure 6.10 Edit Form appear when Edit Button clicked - Waste Collection Detail	
Testing Scenario	131
Figure 6.11 Confirmation appears when Delete Button clicked - Waste Collection	

Detail Testing Scenario.....	132
Figure 6.12 Customer Withdrawal Request Page appear when the Withdrawal Request menu is clicked from the side bar - Customer Withdrawal Request Testing Scenario	134
Figure 6.13 Data appear based on the filter - Customer Withdrawal Request Testing Scenario	135
Figure 6.14 Add Form appear when Add Withdrawal Request Button clicked - Customer Withdrawal Request Testing Scenario.....	135
Figure 6.15 Paid Form appear when the Paid Button clicked - Customer Withdrawal Request Testing Scenario	136
Figure 6.16 Confirmation appear when the Decline Button clicked - Customer Withdrawal Request Testing Scenario	136
Figure 6.17 Customer Credit Page appear when the Credit menu is clicked from the side bar - Customer Credit Testing Scenario.....	137
Figure 6.18 Data shown based on the filter - Customer Credit Testing Scenario	138
Figure 6.19 Customer Debit Page appear when the Debit menu is clicked from the side bar - Customer Debit Testing Scenario	139
Figure 6.20 Data shown based on the filter - Customer Debit Testing Scenario.....	139
Figure 6.21 Transaction Proof Image has been encrypted and uploaded - Customer Debit Testing Scenario	140
Figure 6.22 Transaction Proof image is decrypted and shown when the View Button is clicked - Customer Debit Testing Scenario.....	140
Figure 6.23 Customer Vendor Transaction Page appear when the Vendor transaction menu is clicked from the side bar – Vendor Transaction Testing Scenario.....	143
Figure 6.24 Data shown based on the filter – Vendor Transaction Testing Scenario.....	143
Figure 6.25 Add Form appear when Add Vendor Transaction Button – Vendor Transaction Testing Scenario	144
Figure 6.26 Transaction Proof Image has been encrypted and uploaded – Vendor Transaction Testing Scenario	144
Figure 6.27 Transaction Proof Image is decrypted and shown when View Button clicked – Vendor Transaction Testing Scenario.....	145
Figure 6.28 Edit Form appear when Edit Button – Vendor Transaction Testing	

Scenario 145

Figure 6.29 Confirmation Appear when Delete Button clicked– Vendor Transaction
Testing Scenario 146

Figure 6.30 Redirected to Login Page when Logout Button clicked - Logout Testing
Scenario 147