

**DIGITAL-ONLY BANK MOBILE APPLICATION DEVELOPMENT IN  
MALAYSIA: ENHANCING CUSTOMER EXPERIENCE**

BY  
CHUA HUEY MEI

A REPORT  
SUBMITTED TO  
Universiti Tunku Abdul Rahman  
in partial fulfillment of the requirements  
for the degree of  
BACHELOR OF INFORMATION SYSTEMS (HONOURS) BUSINESS  
INFORMATION SYSTEMS  
Faculty of Information and Communication Technology  
(Kampar Campus)

JAN 2022

## REPORT STATUS DECLARATION FORM

Title: Digital-only Bank Mobile Development in Malaysia:  
Enhancing Customer Experience

Academic Session: Jan 2022

I CHUA HUEY MEI  
(CAPITAL LETTER)

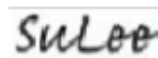
declare that I allow this Final Year Project Report to be kept in  
Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.



(Author's signature)

Verified by,



(Supervisor's signature)

Address:

212, Jalan Berangan  
Kampung Jambu 34000  
Taiping, Perak.

SU LEE SENG

Supervisor's name

Date: 15<sup>th</sup> April 2022

Date: 15<sup>th</sup> April 2022

<b>Universiti Tunku Abdul Rahman</b>			
Form Title : <b>Sample of Submission Sheet for FYP/Dissertation/Thesis</b>			
Form Number: <b>FM-IAD-004</b>	Rev No.: <b>0</b>	Effective Date: <b>21 JUNE 2011</b>	Page No.: <b>1 of 1</b>

**FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY**  
**UNIVERSITI TUNKU ABDUL RAHMAN**

Date: 15<sup>th</sup> April 2022

**SUBMISSION OF FINAL YEAR PROJECT /DISSERTATION/THESIS**

It is hereby certified that Chua Huey Mei (ID No: 18ACB04692) has completed this final year project entitled "Digital-only Bank Mobile Development in Malaysia: Enhancing Customer Experience" under the supervision of Mr. Su Lee Seng (Supervisor) from the Department of Digital Economy Technology, Faculty of Information and Communication Technology.

I understand that University will upload softcopy of my final year project in pdf format into UTAR Institutional Repository, which may be made accessible to UTAR community and public.

Yours truly,



\_\_\_\_\_  
 (Chua Huey Mei)

## DECLARATION OF ORIGINALITY

I declare that this report entitled “**DIGITAL-ONLY BANK MOBILE APPLICATION DEVELOPMENT IN MALAYSIA: ENHANCE CUSTOMER EXPERIENCE**” is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature : 

Name : Chua Huey Mei

Date : 15<sup>th</sup> April 2022

## **ACKNOWLEDGEMENTS**

I would like to express my sincere thanks and appreciation to my supervisor, Mr. Su Lee Seng who has given me this bright opportunity to engage in a digital-only banking mobile application project. A special thanks to him who provides simulation suggestions and encouragement, and helps me to coordinate my project, especially in report parts.

I would also like to acknowledge with much appreciation to my parents and friends for their constant motivation and encouragement; without their assistance and moral support, this project may not have seen the light of the day.

## **ABSTRACT**

Customers' digital lives are being accelerated nowadays, thanks to increased internet penetration, smartphone use, and access to wider networks. In this environment, traditional banks confront a major problem in sustaining and growing their business in the face of digital disruption and the rise of new client expectations. The term "digital-only bank" is introduced which refers to the development of banking services and the delivery of products and services via mobile devices without needing a physical branch to operate. The purpose of this project is to enable customers to obtain financial services through mobile devices more easily to let them have a look of their financial status, introduce the impact of financial inclusion on economic growth to reduce the population with unbanked and underbanked individuals and lift them out of poverty and social inequality, and provide transparency in the transaction. As a result, creating technology in the banking sector is critical for attracting customers, and improving technology from the customer's perspective is also critical. Methodology adopted in this project is prototype model which consists of six phases. By using this methodology, it is easier to improve or remaking the prototype after getting user feedback whenever each phase completed. This is to ensure the satisfaction of customers toward digital banking and to find out problems encountered by customers. This paper discusses a few relevant literature reviews, research analysis, research findings, and conclusion of this study.

# TABLE OF CONTENTS

<b>TITLE PAGE</b>	<b>i</b>
<b>REPORT STATUS DECLARATION FORM</b>	<b>ii</b>
<b>FYP THESIS SUBMISSION FORM</b>	<b>iii</b>
<b>DECLARATION OF ORIGINALITY</b>	<b>iv</b>
<b>ACKNOWLEDGEMENTS</b>	<b>v</b>
<b>ABSTRACT</b>	<b>vi</b>
<b>TABLE OF CONTENTS</b>	<b>vii</b>
<b>LIST OF FIGURES</b>	<b>xi</b>
<b>LIST OF TABLES</b>	<b>xvii</b>
<b>LIST OF SYMBOLS</b>	<b>xix</b>
<b>LIST OF ABBREVIATIONS</b>	<b>xx</b>
<b>CHAPTER 1 INTRODUCTION</b>	<b>1</b>
1.1 Background Information	1
1.2 Problem Statement and Motivation	4
1.3 Objectives	7
1.4 Project Scope	10
1.5 Contributions	11
1.6 Report Organization	13
<b>CHAPTER 2 LITERATURE REVIEW</b>	<b>14</b>
2.1 Review of the Technologies	14
2.1.1 Hardware Platform	15
2.1.2 Software Platform	15
2.1.3 Firmware/OS	15
2.1.4 Database	16
2.1.5 Programming Language	16
2.1.6 Summary of the Technologies Review	16
2.2 Review of the Existing Systems/Applications	17
2.2.1 Monzo Bank (Monzo, 2015)	17

2.2.2	Monese – Mobile Money Account for UK & Europe (Monese, 2015)	20
2.2.3	Revolut – Get more from your money (Revolut, 2015)	23
2.2.4	N26 – The Mobile Bank (N26, 2015)	25
2.2.5	Comparison Table	27
 <b>CHAPTER 3 SYSTEM METHODOLOGY/APPROACH</b>		<b>28</b>
3.1	Methodology	28
3.2	System Design Diagram/Equation	30
3.2.1	System Architecture Diagram	30
3.2.2	Overall Use Case Diagram of Digital-only Banking Mobile Application	31
3.2.3	Login Module	32
3.2.4	Forgot Password Module	34
3.2.5	Register Module	36
3.2.6	Account Module	38
3.2.7	Fund Transfer Module	40
3.2.8	Pay Bills Module	43
3.2.9	Apply Loan Module	47
3.2.10	Budgeting Module	49
3.2.11	Profile Module	51
3.2.12	Manage Customer Module	54
3.2.13	Manage Account Module	56
3.2.14	Manage Loan Module	58
3.2.15	Manage Report Module	62
3.2.16	Contact Customer Module	65
3.3	System Pre-Authoring Analysis Survey Results	67
 <b>CHAPTER 4 SYSTEM DESIGN</b>		<b>74</b>
4.1	System Design Diagram	74
4.1.1	System Flowchart	74
4.1.2	System Block Diagram	78
4.2	System Flow Description	79



<b>CHAPTER 5 SYSTEM IMPLEMENTATION</b>	<b>93</b>
5.1 Hardware Setup	93
5.2 Software Setup	94
5.3 Setting and Configuration	95
5.4 System Operation (with Screenshot)	96
<b>CHAPTER 6 SYSTEM EVALUATION AND DISCUSSION</b>	<b>157</b>
6.1 System Evaluation Survey Results	157
6.2 Testing Setup and Result	162
6.2.1 Unit Testing 1 – Login	162
6.2.2 Unit Testing 2 – Register User	163
6.2.3 Unit Testing 3 – Register Bank Staff	164
6.2.4 Unit Testing 4 – User Main Page (Home Navigation Bar Button)	165
6.2.5 Unit Testing 5 – My Account	166
6.2.6 Unit Testing 6 – Fund Transfer	167
6.2.7 Unit Testing 7 – Fund Transfer Within Banky	168
6.2.8 Unit Testing 8 – Fund Transfer To Other Bank	169
6.2.9 Unit Testing 9 – Request Fund	171
6.2.10 Unit Testing 10 – Pay Bills	172
6.2.11 Unit Testing 11 – Loan	173
6.2.12 Unit Testing 12 – Budgeting	174
6.2.13 Unit Testing 13 – User Main Page (History Navigation Bar Button)	176
6.2.14 Unit Testing 14 – User Main Page (QR Scan Navigation Bar Button)	176
6.2.15 Unit Testing 15 – User Main Page (Notification Navigation Bar Button)	176
6.2.16 Unit Testing 16 – User Main Page (Profile Navigation Bar Button)	177
6.2.17 Unit Testing 17 – Bank Manager Main Page	179
6.2.18 Unit Testing 18 – Manage Customer	180

6.2.19	Unit Testing 19 – Manage Account	181
6.2.20	Unit Testing 20 – Manage Loan	182
6.2.21	Unit Testing 21 – Manage Report	183
6.2.22	Unit Testing 22 – Contact Customer	186
6.3	Project Challenges	187
6.4	Objectives Evaluation	188
<b>CHAPTER 7 CONCLUSION AND RECOMMENDATION</b>		<b>189</b>
7.1	Conclusion	189
7.2	Recommendation	190
<b>REFERENCES</b>		<b>191</b>
<b>APPENDIX</b>		<b>A-1</b>
<b>WEEKLY LOG</b>		<b>C-1</b>
<b>POSTER</b>		<b>D-1</b>
<b>PLAGIARISM CHECK RESULT</b>		<b>E-1</b>
<b>FYP2 CHECKLIST</b>		<b>F-1</b>

## LIST OF FIGURES

<b>Figure Number</b>	<b>Title</b>	<b>Page</b>
Figure 1.1.2	Framework to illustrate FinTech, banks, and governments aim to help in poverty reduction [8]	3
Figure 2.2.1	Monzo Mobile Application Interface [32]	17
Figure 2.2.2	Monese mobile application interface [35]	20
Figure 2.2.3	Revolut Mobile Application Interface [38]	23
Figure 2.2.4	N26 Mobile Application Interface [42]	25
Figure 3.1	Prototype Model [43]	28
Figure 3.2.1	System Architecture Diagram	30
Figure 3.2.2	Overall Use Case Diagram of Digital-only Banking Mobile Application	31
Figure 3.2.3	Activity Diagram – Login Module	33
Figure 3.2.4	Activity Diagram – Forgot Password Module	35
Figure 3.2.5	Activity Diagram – Register Module	37
Figure 3.2.6	Activity Diagram – Account Module	39
Figure 3.2.7	Activity Diagram – Fund Transfer Module	42
Figure 3.2.8	Activity Diagram – Pay Bills Module	45
Figure 3.2.9	Activity Diagram – Apply Loan Module	48
Figure 3.2.10	Activity Diagram – Budgeting Module	50
Figure 3.2.11	Activity Diagram – Profile Module	53
Figure 3.2.12	Activity Diagram – Manage Customer Module	55
Figure 3.2.13	Activity Diagram – Manage Account Module	57
Figure 3.2.14	Activity Diagram – Manage Loan Module	60
Figure 3.2.15	Activity Diagram – Manage Report Module	64
Figure 3.2.16	Activity Diagram – Contact Customer Module	66
Figure 3.3.1	Demographic 1	67
Figure 3.3.2	Demographic 2	67
Figure 3.3.3	Demographic 3	68
Figure 3.3.4	Demographic 4	68
Figure 3.3.5	Demographic 5	68

Figure 3.3.6	Demographic 6	69
Figure 3.3.7	Demographic 7	69
Figure 3.3.8	Demographic 8	69
Figure 3.3.9	Demographic 9	70
Figure 3.3.10	Experience of using another banking service	70
Figure 3.3.11	Perceptions on favorite transactions in digital banking	70
Figure 3.3.12	Perceptions on the benefit of digital banking	71
Figure 3.3.13	Perceptions on the benefit of digital banking to community	71
Figure 3.3.14	Perceptions on the budgeting feature	71
Figure 3.3.15	Perceptions on the standalone budgeting application	72
Figure 4.1.1.1	System Flowchart	74
Figure 4.1.1.2	System Flowchart (cont)	75
Figure 4.1.1.3	System Flowchart (cont)	76
Figure 4.1.1.4	System Flowchart (cont)	77
Figure 4.1.2	System Block Diagram	78
Figure 5.4.1	Onboarding Screen1	96
Figure 5.4.2	Onboarding Screen2	96
Figure 5.4.3	Onboarding Screen3	97
Figure 5.4.4	Splash Screen	97
Figure 5.4.5	Login Page	98
Figure 5.4.6	Forgot Password	98
Figure 5.4.7	Reset Password	98
Figure 5.4.8	Register User Page	100
Figure 5.4.9	Register Bank Staff Page	100
Figure 5.4.10	User Main Page	101
Figure 5.4.11	User Main Page (cont)	101
Figure 5.4.12	Account Page	102
Figure 5.4.13	Account Page (cont)	102
Figure 5.4.14	Cash Deposit History Page	103
Figure 5.4.15	Cash Withdrawal History Page	103
Figure 5.4.16	Fund Transfer Page	104
Figure 5.4.17	Transfer Within Banky Page	105

Figure 5.4.18	No Such Account User Exists Yet	105
Figure 5.4.19	Transfer Within Banky Page2	106
Figure 5.4.20	Insufficient Fund to Transfer	106
Figure 5.4.21	Transfer Within Banky Page3	107
Figure 5.4.22	Transaction Receipt	107
Figure 5.4.23	Selection of Other Bank	108
Figure 5.4.24	Transfer To Other Bank Page	108
Figure 5.4.25	No Such Account User Exists Yet	109
Figure 5.4.26	Wrong Payee Name	109
Figure 5.4.27	Other Bank Dummy Table	110
Figure 5.4.28	Other Bank Account Exists	111
Figure 5.4.29	Insufficient Fund to Transfer	111
Figure 5.4.30	Transfer To Other Bank Page2	112
Figure 5.4.31	Transaction Receipt	112
Figure 5.4.32	Main Page-New Balance	113
Figure 5.4.33	Main Page-New Balance2	113
Figure 5.4.34	Add New Contact	114
Figure 5.4.35	Request Fund Page	114
Figure 5.4.36	Request Fund Detail	115
Figure 5.4.37	Notification (Payer Side)	115
Figure 5.4.38	Pay Bills Page	116
Figure 5.4.39	Pay Bill Details Page	116
Figure 5.4.40	Payment Receipt	117
Figure 5.4.41	Main Page-New Balance3	117
Figure 5.4.42	Loan Page	118
Figure 5.4.43	Apply Loan Page	118
Figure 5.4.44	Selection of Loan Type	119
Figure 5.4.45	Filled out Loan Application	119
Figure 5.4.46	Repayment Plan	120
Figure 5.4.47	Loan Application Pending	120
Figure 5.4.48	Loan Page Updated	121
Figure 5.4.49	Monthly Repayment Details	121
Figure 5.4.50	Monthly Repayment Details2	122

Figure 5.4.51	Repay The Loan	122
Figure 5.4.52	Main Page-New Balance4	123
Figure 5.4.53	Budgeting Page	123
Figure 5.4.54	Set Weekly Budget	124
Figure 5.4.55	Add Weekly Budget	124
Figure 5.4.56	Selection of Item Category	125
Figure 5.4.57	Updated Weekly Budget	125
Figure 5.4.58	Update/Delete Budget Item	126
Figure 5.4.59	Updated Weekly Budget2	126
Figure 5.4.60	Add Today Spending	127
Figure 5.4.61	Today Spending Page	127
Figure 5.4.62	Update/Delete Spending Item	128
Figure 5.4.63	Updated Today Spending2	128
Figure 5.4.64	This Week Spending	129
Figure 5.4.65	All Spending History	129
Figure 5.4.66	Manage Your Budget	130
Figure 5.4.67	Manage Your Budget (cont)	130
Figure 5.4.68	History Transactions (Current User side)	131
Figure 5.4.69	History Transactions (Beneficiary side)	131
Figure 5.4.70	QR Scan	132
Figure 5.4.71	QR Receive	132
Figure 5.4.72	QR Pay	133
Figure 5.4.73	Notification Page	133
Figure 5.4.74	Profile Page	134
Figure 5.4.75	Profile Page (cont)	134
Figure 5.4.76	Profile Settings	135
Figure 5.4.77	Edit Profile	135
Figure 5.4.78	Privacy Page	136
Figure 5.4.79	Help Center	136
Figure 5.4.80	Contact Us Page	137
Figure 5.4.81	Email Banky	137
Figure 5.4.82	Phone Call Banky	138
Figure 5.4.83	WhatsApp Banky	138

Figure 5.4.84	Nearby ATM	139
Figure 5.4.85	Bank Manager Main Page	140
Figure 5.4.86	Manage Customer Page	140
Figure 5.4.87	Customer Details	141
Figure 5.4.88	Update Customer Profile	141
Figure 5.4.89	Manage Account	142
Figure 5.4.90	Disabled Customer Account	142
Figure 5.4.91	Disabled Acc Cannot Login	143
Figure 5.4.92	Enabled Back Customer	143
Figure 5.4.93	Manage Loan	144
Figure 5.4.94	Search by Loan Type	144
Figure 5.4.95	Filter by Loan Status	145
Figure 5.4.96	Filter Result	145
Figure 5.4.97	Loan Application Details	146
Figure 5.4.98	Edit Loan Status	146
Figure 5.4.99	Add Remark	147
Figure 5.4.100	Remark updated	147
Figure 5.4.101	Loan History (Customer Side)	148
Figure 5.4.102	Rejected Reason	148
Figure 5.4.103	Manage Report	149
Figure 5.4.104	Daily Active User	149
Figure 5.4.105	Event Clicked	150
Figure 5.4.106	Event Clicked (cont)	150
Figure 5.4.107	User Account Chart	151
Figure 5.4.108	Select a Date	151
Figure 5.4.109	Specific Date User Account Result	152
Figure 5.4.110	Loan Status Chart	152
Figure 5.4.111	Updated Manage Loan	153
Figure 5.4.112	Specific Date Loan Status Result	153
Figure 5.4.113	Loan Ranking Chart	154
Figure 5.4.114	Specific Date Loan Ranking Result	154
Figure 5.4.115	Daily Transaction Chart	155
Figure 5.4.116	Contact Customer Page	155

Figure 5.4.117	Call Customer	156
Figure 5.4.118	Email Customer	156
Figure 6.1.1	Names of all respondents	157
Figure 6.1.2	Onboarding screen evaluation	157
Figure 6.1.3	Overall satisfaction with experiences at Banky	158
Figure 6.1.4	Screen layout design evaluation	158
Figure 6.1.5	Navigation button placement evaluation	158
Figure 6.1.6	Evaluation at using Banky	159
Figure 6.1.7	Evaluation of top feature selection	159
Figure 6.1.8	Recommendations	160
Figure 6.1.9	Asking opinion whether to recommend this Banky	160



## LIST OF TABLES

<b>Table Number</b>	<b>Title</b>	<b>Page</b>
Table 2.1.5	Summary of the Technologies Review	16
Table 2.2.5	Comparison Between Similar Mobile Application and Proposed Project	27
Table 3.2.3	Use Case Description – Login Module	32
Table 3.2.4	Use Case Description - Forgot Password Module	34
Table 3.2.5	Use Case Description - Register Module	36
Table 3.2.6	Use Case Description - Account Module	38
Table 3.2.7	Use Case Description – Fund Transfer Module	40
Table 3.2.8	Use Case Description – Pay Bills Module	43
Table 3.2.9	Use Case Description – Apply Loan Module	47
Table 3.2.10	Use Case Description - Budgeting Module	49
Table 3.2.11	Use Case Description - Profile Module	51
Table 3.2.12	Use Case Description – Manage Customer Module	54
Table 3.2.13	Use Case Description – Manage Account Module	56
Table 3.2.14	Use Case Description – Manage Loan Module	58
Table 3.2.15	Use Case Description – Manage Report Module	63
Table 3.2.16	Use Case Description – Contact Customer Module	65
Table 5.1.1	Specification of Laptop	93
Table 5.2.1	Hardware Requirement for Android Studio	94
Table 5.2.2	Requirement for the system	94
Table 6.2.1	Unit Testing 1 – Login	162
Table 6.2.2	Unit Testing 2 – Register User	163
Table 6.2.3	Unit Testing 3 – Register Bank Staff	164
Table 6.2.4	Unit Testing 4 – User Main Page (Home Navigation Bar Button)	165
Table 6.2.5	Unit Testing 5 – My Account	166
Table 6.2.6	Unit Testing 6 – Fund Transfer	167
Table 6.2.7	Unit Testing 7 – Fund Transfer Within Banky	168
Table 6.2.8	Unit Testing 8 – Fund Transfer To Other Bank	169
Table 6.2.9	Unit Testing 9 – Request Fund	171

Table 6.2.10	Unit Testing 10 – Pay Bills	172
Table 6.2.11	Unit Testing 11 – Loan	173
Table 6.2.12	Unit Testing 12 – Budgeting	174
Table 6.2.13	Unit Testing 13 – User Main Page (History Navigation Bar Button)	176
Table 6.2.14	Unit Testing 14 – User Main Page (QR Scan Navigation Bar Button)	176
Table 6.2.15	Unit Testing 15 – User Main Page (Notification Navigation Bar Button)	176
Table 6.2.16	Unit Testing 16 – User Main Page (Profile Navigation Bar Button)	177
Table 6.2.17	Unit Testing 17 – Bank Manager Main Page	179
Table 6.2.18	Unit Testing 18 – Manage Customer	180
Table 6.2.19	Unit Testing 19 – Manage Account	181
Table 6.2.20	Unit Testing 20 – Manage Loan	182
Table 6.2.21	Unit Testing 21 – Manage Report	183
Table 6.2.22	Unit Testing 22 – Contact Customer	186

## LIST OF SYMBOLS

% Percentage

## LIST OF ABBREVIATIONS

<i>SOP</i>	Standard Operating Procedures
<i>MCO</i>	Movement Control Order
<i>GDP</i>	Gross Domestic Product
<i>CX</i>	Customer Experience
<i>P2P</i>	Peer-to-Peer
<i>CVP</i>	Customer Value Proposition
<i>PFM</i>	Personal Financial Management
<i>SME</i>	Small to Medium-Sized Enterprise
<i>Fintech</i>	Financial Technology
<i>KYC</i>	Know Your Client
<i>QR</i>	Quick Response
<i>ATM</i>	Automated Teller Machine
<i>API</i>	Application Programming Interface
<i>TPP</i>	Third-Party Provider
<i>IDE</i>	Integrated Development Environment
<i>AVD</i>	Android Virtual Device
<i>TNB</i>	Tenaga Nasional Berhad
<i>PTPTN</i>	Perbadanan Tabung Pendidikan Tinggi Nasional

## CHAPTER 1 INTRODUCTION

### 1.1 Background Information

Financial technology, which is also known as fintech, is an economic field consisting of companies that make financial services more productive by using technology [1]. The use of innovative technology-based technologies for financial services is at the core of Fintech. Fintech refers to technologies designed to provide financial services in competition with traditional financial approaches. Fintech will introduce the impact of inclusive finance on economic growth in order to reduce the population with unbanked and underbanked individuals and lift them out of poverty and social inequality. Besides that, traditional banking companies, alternatively, are hampered by legacy operating systems, innovation capability, agility and experience in technology. The Covid-19 crisis has become an unexpected catalyst to stimulate Malaysia's digital banks, despite the digital disruption to the banking industry. To achieve a cashless society in Malaysia, the digital-only bank can help to encourage all sections of society to use electronic payments more.

Fintech financial services is being transformed the entire banking system from branch-specific processes to multiple digital platforms, such as online, social, and mobile. It also reduces the dependency of banks on physical branches. In particular, Fintech has transformed the revolutionary idea of the banking space into modern standards, paving the way for a new generation of banking, making it a real-time, transparent, and seamless way [2]. There are many types of financial services offered to customers such as digital-only banking, wealth management, mutual funds, insurance, capital restricting and other services [3]. It can see that these financial services provided are not much different from physical branches.

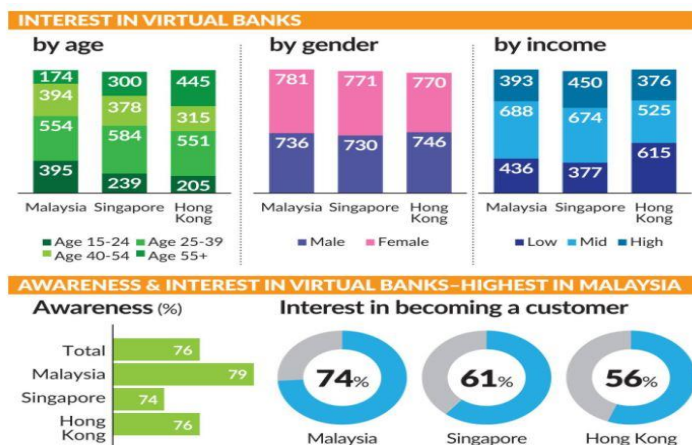


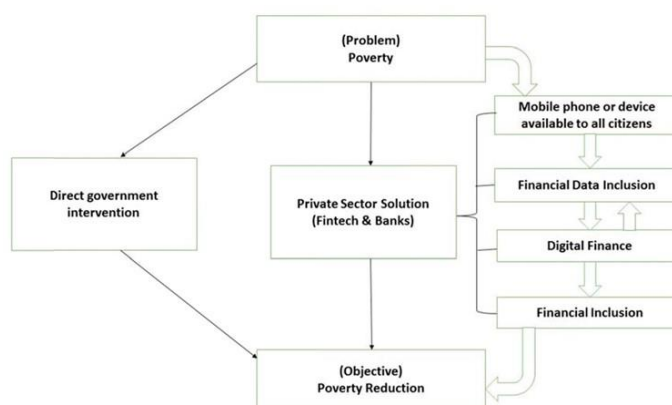
Figure 1.1.1 Statistic of Interesting in Becoming a Virtual Bank Customer [4]

PwC Malaysia reported that around 74% of Malaysians are interested in becoming customers of digital-only banks [5] due to the rise in smartphone use has forced banks to build mobile applications that provide convenient FinTech banking services. In October 2019, PwC conducted a customer survey among bank account owners who 18 years old and above in Malaysia, Singapore, and Hong Kong, and it discovered that 45% of Malaysians are looking for virtual banking service providers who can give a better mobile and digital experience [6]. Given that digital banking service providers are new to Malaysian customers, downtime will harm the customer experience and undermine efforts to gain trust with them. Besides that, PwC also indicated that 77% of Malaysians are interested in additional services offered beyond financial products by their banks [5], thus digital-only banks will be a good choice as it are more sophisticated and elaborate: they include using technology when providing banking services and products in every financial transactions, process and level, with the aim of making the experience of the customer more seamless, efficient and effective. The digital-only banks are different from traditional banks that serve A-Z when meeting customer needs. Its model targets very narrow segments among industries and customers, and it also prefers to choose incredibly unique micro market segments. Digital-only banks are able to address the lack of physical branch presence and offer greater convenience to banking industries and customers through offering online banking services.

In reality, the only reason people still continue to visit physical branches is because the bank requires them to do so for regulatory reasons, for example, physically visit bank to show their identity card or passport and utility bills in order to create a new account, according to Know Your Client (KYC) guidelines [7]. The truth is that, in today's era of digitization, many only visit a physical branch because the regulations require it, similarly, if given the opportunity to remove the rules, then most would avoid visiting physical branch. Now by using digital-only banking mobile application, customers can easily obtain financial services from digital-only bank such as deposits, loans, fund transfers, set budget for spending, overview expenditure and others which it provides more than a traditional bank, so that customers may have greater control and clearer visibility over their personal finances, greater making financial decisions and being able to send and receive payments in a few of seconds. In addition, it can also be used as a budgeting tool, and using it independently to keep closer track on customer expenses. The digital-only banking works exactly the same way customers are used to

Bachelor of Information Systems (Honours) Business Information Systems  
Faculty of Information and Communication Technology (Kampar Campus), UTAR

everyday, for example, the customer has a physical card, an account number, and an authentication code, so the customer can spend and request payment in exactly the same way. It is like a current account from a traditional bank, but with additional flexibility and transparency, the customer can perform several banking functions such as check account balance, deposit, statements, and make payment and others. Thus, customers can enjoy the banking services anytime and anywhere without visiting the physical branches since some of the physical branches close temporarily during the Covid-19 pandemic and also some certain areas do not have any physical branches.



**Figure 1.1.2 Framework to illustrate FinTech, banks, and governments aim to help in poverty reduction [8]**

Based on the figure above, we can see that digital-only bank needs comprehensive financial data inclusiveness in order to fully realize its potential in financial inclusiveness. The way how financial inclusion impacts to individuals and SME businesses is that it allows low-income people to save money aside for the future, therefore ensuring personal financial independent, as well as a high level of utilization of bank deposits, which helps digital-only banks maintain a more solid deposit based during difficult times [9]. Some more with the digital-only banking mobile application, it also can provide low-income households with more possibilities to save, invest, and borrow money. According to Prasad [10], he pointed out that the lack of adequate finance for SME businesses and small-scale entrepreneurs has a negative impact on total employment growth because these businesses are far more labor-intensive in their operations. Furthermore, digital-only banks can help various sectors of the economy participate more fully in the formal financial system which shows a positive impact on economic growth because the case for using interest rates as a key policy tool for macroeconomic stability strengthens as the formal financial sector grows [11].

## **1.2 Problem Statement and Motivation**

Nowadays, digitization not only assists banks in their transformation, but it also forces them to reimagine their businesses. In the face of the Covid-19 pandemic, banks should not only accelerate their transformation program with transactions shifting to digital platforms, but also reconsider the nature of their role in the lives of their customers. Establishing digital-only banks will position them optimally for the digital age. The Malaysian government should encourage people to use digital-only banking mobile applications to better manage their finances without having to visit physical branches, especially during this pandemic period, and for customers who do not have physical branches nearby to visit. That is because it provides them with a lot of convenience in terms of receiving financial services at the right time. Aside from that, the digital-only bank can help to alleviate the impact of financial inclusion on economic growth by lifting people out of poverty and social inequality. Furthermore, it can provide customers with greater transparency when making international transfers and enhance customer experience by leveraging technology.

There are a few problems that will arise when customers want to use traditional methods to access physical branches of traditional banks to obtain financial services.

- **Difficult to access banking services and products**

Recently, traditional banks needed to maintain social distancing practices according to the standard operating procedures (SOP) during the movement control order (MCO) period, so most of the physical branches are forced to temporarily close or some of the physical branches are allowed to operate but with limited counter services and operating hours. This trend and problem are expected to continue even after the Covid-19 pandemic. It is expected to be a normal way of life post Covid-19 pandemic. Besides, not all the regions in Malaysia provide financial services through traditional banks, especially rural areas. For people who live in places where there is no physical branch nearby, it may be inconvenient for them to obtain financial services. Customers who have no option and must visit the physical branch for opening a new bank account, applying loans, and others, may not be able to get the financial services during this period or from rural areas. Customer satisfaction ultimately declined because the conventional banks failed to focus on providing services to customers at the right time with the right service level. Hence, digital-only banks that operates with no bricks-and-



mortar branches and provides digital banking solutions to customers through internet and the mobile application anytime and anywhere to have better control of their financial status which is more convenient than traditional bank [12].

- **Economic growth impact of financial inclusion is still insufficient**

Recently, global economic growth has been adversely impacted by the Covid-19 pandemic, resulting an economic downturn and a decline in gross domestic product (GDP). The effect of financial inclusion on economic growth is still scant. There are still a lot of unbanked and underbanked customers in the world. The term “unbanked” implies that customers do not have any checking or saving accounts in commercial banks while the term called “underbanked” indicates that although customers may have a checking or savings account but still have restricted access to the financial services [13]. According to a study, there is over 70% of consumers among the 668 million population in Southeast Asia, which covers Malaysia, Singapore, Thailand, Indonesia, the Philippines, and Vietnam, are unbanked and underbanked, and millions of Small to Medium-Sized Enterprises (SMEs) in the region are currently facing large funding gaps [14] & [15]. Estimated there is around 55 percent of Malaysia population is currently still unbanked or underbanked [16]. Therefore, digital-only banks may reduce both poverty and social inequality, that is because it will open up blocked opportunities for deprived segments of the population to make progress, it eventually makes their lives better.

- **Lack of transparency in international transfer of funds**

The existing financial services provided by traditional banks are not transparent to customers, especially when it comes to overseas remittances. Given the lack of funds transparency from banks, it is also hard to predict how much the transfer fee will cost the customer in advance. And the exact amount sent by customer to the beneficiary may not always be the same. To ensure that the amount that the recipient will get as planned, the customer will have to pay extra for the charge of transfer fees and the exchange rate. It basically depends on how much the customer sends and where he sends the transfer to and how he pays for the transfer service. Besides, they also apply a markup or fee to the exchange rate they use to convert the money on top of the transfer fee. For example,

the customer will need to pay an additional 5% or RM 50 more as the transfer fee because of the exchange rate markup, if sending RM1,000 to a friend in Singapore [17]. Thus, digital-only banks can benefit customers by omitting the transfer fees which makes the international transactions cheaper than traditional banks, besides it also makes customers more transparent by letting customers know the exchange rate. Customers will have greater transparency and will be able to take advantage of the best deals available.

- **Lack of technology leverage leads to a decline in customer experience (CX)**

Many traditional banks are unable to adopt the digital-only bank strategy since they still use many decades-old legacy systems. Traditional banks spend over 70% of their IT spending on traditional IT services, such as core system maintenance, and only about 30% on non-traditional digital transformation solutions, such as cloud and data analytics [18]. Since the traditional bank is unable to leverage data which means it also cannot meet the needs of the customer because it is trapped in product silos [7]. In other words, incumbent banks have fallen behind Fintech startups which are digital-only banks because of failing to provide personalization to the customer [19]. Hence, it is obviously to see that digital-only banks can enhance customer experience by raising the bar in terms of satisfying customer needs to provide quick, seamless, and personalized services, for example, real-time balances, budgeting tools, transaction alerts, and personal finance solutions.

### 1.3 Project Objectives

- **To develop a digital-only bank that operates through mobile application allowing customers to easily obtain financial services**

According to The Star, the Director General of Health Malaysia stated that the best way to avoid the current wave of Covid-19 affecting the country is to stay at home [20]. In order to allow customers to access financial services without leaving their homes and with just a few clicks, instead of going out to physical outlets, and avoiding the congestion of traditional bank physical outlets, they can use digital-only banking mobile application for digital self-registration, open account remotely in under few minutes, access their checking and saving accounts, instant P2P (Peer-to-Peer) payments, and others. One of the good things is digital-only bank will be able to cut costs by eliminating queues in banking halls and lowering manual paperwork and documentation [21]. So those customers can have better control their finances and provide clearer visibility through the operation of digital-only banking mobile applications because with the rise of digital banking services, a huge number of depositors (customers) may simply transfer banks in short, compelling banks to deliver superior service or risk losing customers to competitors' bank. In addition, digital-only banking mobile applications will allow households and businesses to retain and increase access to financial services during lockdowns and business reopening, including mobile money, given the increasing preference for cashless and contactless transactions to reduce the spread of the virus [22] and reduce the physical cash in circulation, as well as helps reduce of excessive inflation in developing and poor countries.

- **To build a digital-only banking mobile application to positively address the impact of financial inclusion on economic growth**

Financial inclusion can help bridge these gaps and provide households and businesses with greater access to the resources needed for finance consumption and investment, thereby increasing their level of economic activity [23]. There are few examples showing that participating in the digital-only banking will bring various personal benefits, such as the ability to pay for children's tuition fees, which in turn has cultivated a new generation of well-trained and knowledgeable people, and the ability

to start and grow into business that provide opportunities for individuals improve their long-term prospects through micro-financing programs, as well as the ability to deal with uncertainties that require temporary or unexpected payments. It can benefit the poorest and most vulnerable people in society, get rid of poverty, and reduce social inequality. Financial inclusion not only benefits individuals but can also develop the entire community together and help boost economic growth. Thus, it is a trend to offer equal and transparent financial services to more people at affordable rate [24].

- **To construct a digital-only bank that operates via a mobile application and provides transparency in fund transfer transactions**

In general, transaction transparency usually means that the sender knows the details and the total fees, how much net funds the beneficiary will receive, exactly how long it will take, and what remedial steps can be taken if any of these promises are not fulfilled. Even if you transfer money between different banks in different countries, the interbank transactions will be free. Transparency in fund transfer transaction is not only important for senders and recipients, but also for banks, because hidden transfer fees and exchange rate markups can hurt customers. The presence of digital-only banks can challenge traditional banks' traditional business models by charging customers transparent and cheaper fees, and providing faster financial services and a better user experience [25]. Hence, by using a digital-only banking mobile application, it will eventually establish a better relationship with customers if it is able to provide transparency in fund transfer transactions because customers will feel that they have taken control of the transaction.

- **To launch a digital-only banking mobile application enhances customers' experience by leveraging technology**

Customer experience in financial services or in banking field refers to all a digital-only bank's efforts to make each customer feel significant when they contact its bank. It is also the entirety of a customer's perceptions of all encounters throughout the trip. Digital-only banks will place a significant emphasis on a strong digital customer value proposition (CVP), utilizing flexible technology platforms to achieve cheap and innovation acquisition and servicing costs [26]. In order to enhance customer experience, digital-only banks need a deep understanding or 360-degree view of their

customers and leverage the data available, then only relate all the things to customers in ways that they understand. Big data analytics is a critical component in improving financial services and making critical business decisions as well as gaining deep insights from leveraging big data analytics [27]. For example, digital-only banking mobile application will provide Personal Financial Management (PFM) solutions to customers by relying heavily on comparative insights based on customer transaction history and behaviour to give tailored financial advice.

## 1.4 Project Scope

The purpose of this project is to develop a digital-only banking mobile application based on Android platform which enables customers to obtain financial services through mobile devices. The main outcome of this project is to develop a digital-only banking mobile application that aims to make the experience of the customer more seamless, efficient and effective.

The following features and functionalities are financial services that can help a digital-only bank to make their customers seamless, efficient, and effective:

- I. The digital-only banking mobile application will be able to have customer service support.
- II. The digital-only banking mobile application will be able to set budgets limits.
- III. The digital-only banking mobile application will be able to add expense or spending item.
- IV. The digital-only banking mobile application will be able to overview of the spending.
- V. The digital-only banking mobile application will be able to offer loan such as personal, business, car loan, and others.
- VI. The digital-only banking mobile application will be able to request funds from or to other users.
- VII. The digital-only banking mobile application will be able to receive or transfer funds.
- VIII. The digital-only banking mobile application will be able to pay for the bills such as electricity bill, water bill, Wi-Fi bill and others.
- IX. The digital-only banking mobile application will be able to use Quick Response (QR) code to scan and transfer funds.
- X. The digital-only banking mobile application will be able to receive funds from others by letting others scan the generated QR code.
- XI. The digital-only banking mobile application will be able to view for a map displaying branches and Automated Teller Machines (ATMs) nearby.

## 1.5 Contributions

An effective digital-only bank can have a good impact of financial inclusion on economic growth as it will get rid of poverty and reduce social inequality. There are still millions of people intentionally and unjustly excluded from the financial system by traditional banks because of inadequate income levels and consumer discrimination in developing countries, resulting in the possible loss of savings, investment funds and wealth accumulation. Digital-only bank promises to provide a range of formal financial services tailored to the needs of the unbanked and underbanked population that are ethically delivered at a cost that is reasonable for customers and sustainable for providers. Then, a well-designed digital-only banking mobile application could benefit Malaysian citizens to obtain financial services, thereby helping them to make their life wealthier, happier, and so on and without having to go out of their doorstep to access financial services. For example, financial inclusion will impact the lives of people in Malaysia as people will get better off and richer, and children from poor families can eat better food, have money to go to school and get better education. Malaysian households can have money to engage in SME business because it is easier to have loans. They can also cope with income shocks caused by unforeseeable circumstances such as health issue or job loss if they are financially included. Because of the digital-only bank's easy accessibility and affordability, it can assist impoverished customers in transitioning from cash-based transactions to formal digital financial transactions on a secure digital-only bank platform which is its mobile application. Malaysian households and SME can benefit from digital-only bank by having quick access to a varied variety of digital banking services, which can enhance aggregate consumption and hence improve GDP levels. Ultimately, Malaysia will become wealthier, citizens will be happier and healthier as they and their families live can benefit from better economic stability and financial intermediation brought on by digital banking services, more tax revenue will be used to build schools, hospitals, and other facilities, and it will help to reduce poverty levels in developing and emerging economies of Malaysia. In short, financial inclusion benefits digital finance providers, financial services users, governments, as well as Malaysian economy by improving access to finance for the needy and raising government aggregate expenditure.

In addition, the digital-only banking mobile application allows users to have a clearer understanding of their financial situation, so that they can set the budgets and

spending limits to achieve the goal of saving more money or spending these money within their ability and try not to spend more than earn. If Malaysian citizens always keep track of their spending limits and have the habit of saving money, there might be less chances of bankruptcy or not financially prepared if the MCO extended, as according to The Edge Malaysia reported that the survey found that 69.7% of individuals who have worked for less than a year showed that their financial savings can only sustain them for less than a month [28]. By having this digital-only banking mobile application, customers will have a better visualization of their financial situation by having an overview of spending to let customers have deep insights of their financial situation, sending or requesting from or to other users, receiving or transferring money. So that it should provide more transparency to customers when they can check the transfer fee and exchange rate markup whenever customers conduct a fund transfer transaction, whether transfer or receive money. Apart from this, the digital-only bank can help to reduce financial intermediary costs for banks and other financial technology providers such as Monzo, Revolut and others. Additionally, if access to digital banking services provided by digital-only bank is affordable, low-income, and poor citizens will engage in the digital-only banking mobile application, thus resulting in good financial inclusion impacts.



## **1.6 Report Organization**

This report has organized into 7 chapters: Chapter 1 Introduction, Chapter 2 Literature Review, Chapter 3 System Methodology/Approach, Chapter 4 System Design, Chapter 5 System Implementation, System 6 System Evaluation and Discussion, and Chapter 7 Conclusion and Recommendation. The first chapter will discuss about the problem statement and motivation, objectives, project scope and contributions that related to Digital-only Bank. The second chapter will be reviewing of the technologies and the existing systems/applications. The third chapter will give an overall appearance of Digital-only Bank system design diagram such as system architecture diagram, use case diagram with its description, and activity diagram. The fourth chapter discusses top-down system design diagrams in more depth by providing system flowchart and design block diagram, as well as the function of each block in the diagram. The fifth chapter will brief about hardware and software setup, setting and configuration, and system operation with screenshot. The sixth chapter will be the system evaluation and discussion that includes system testing and performance metrics, testing setup and result, project challenges and objectives evaluation. The last chapter will be talked about the conclusion and recommendation for this project.

## CHAPTER 2 LITERATURE REVIEW

### 2.1 Review of the Technologies

Open Banking can assist financial services companies in achieving service offerings, boosting income through new channels and overall consumer involvement, but it requires explicit agreement from them. Open Application Programming Interface (API) such as data API, payment API, consent and identity orchestration API and “ecosystem expansion” API [29], which third-party providers (TPPs) can access consumption stored by current customers’ bank financial information, for example, customers bank accounts and their balances, transaction history, statements, payments, and other financial data from banks and non-bank financial institutions [30].

Open API can pose a risk to banks since they allow fintech startups to leverage financial data held by them. For example, a fintech company may opt to develop a digital banking mobile application using the data API where customers can monitor and manage their money, obtain real-time investing and real-time financial planning advice through chat which most physical banks do not provide such real-time services. This implies that by offering their API, the bank allows the fintech startups to fill the existing gaps and build a barrier between the bank and customers.

Banks, on the other hand, may perceive the aforementioned threat as an opportunity as they sure will not turn a blind eye to these efforts if they want to secure their place in the value chain. Therefore, banks will better manage all issues such as fintech startups developing such services utilizing insecure tactics, for example, screen scraping, exploiting existing APIs, or without using APIs at all, by controlling current connections and upgrading their own goods and services through new collaborations in order to make profit from this potential and improve security.

### 2.1.1 Hardware Platform

Description	Specifications
Model	Asus Vivobook X510UQR
Processor	Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz 1800 Mhz, 4 Core(s), 8 Logical Processor(s)
Operating System	Windows 10
System type	64-bit operating system, x64-based processor
Graphic	Intel(R) UHD Graphics 620, NVIDIA GeForce 940MX
Display Resolution	1920 x 1080 x 60 hertz
Installed RAM	12.0 GB
Storage	476 GB

**Table 2.1.1 Hardware Platform**

### 2.1.2 Software Platform – Android Studio

Android Studio is an integrated development environment (IDE) used to developed the Digital-only Banking android mobile application. By using the Android Emulator provided by Android Studio, it can prototype and test my application on a variety of Android Virtual Device (AVD) types, including phones, tablets and Android TV devices which it will be faster than using a real device to test. Android Studio makes life easier as it is able to multitask, drag-and-drop as well as swipe between emulator screens. The drag-and-drop visual editor for working with XML layout files, making it easier than ever to design a new layout. Not only that, but Android Studio provides an intelligent code editor that enhances code completion, refactoring, and code analysis, helping programmers create better code, work quicker, and be more productive.

### 2.1.3 Firmware/OS – Android

The digital-only banking mobile application will be developed based on the Android platform. The Android version to be used in the software is 4.0 and above. The reason that the Digital-only Bank mobile application chooses Android OS instead of IOS is that Android application development can be done on Windows, Linux, or Mac PCs, but IOS application development requires a Mac. The developer has more flexibility in

customizing the Android application by adding the capabilities and features that the user requires since Android is open-source.

#### 2.1.4 Firebase

Firebase is the backend for Android and is more stable because it is a cloud-based platform. It is a forum for mobile application and web creation built on the framework of Google. The reason to use Firebase as the backend for developing digital-only banking mobile applications is quick setup. It is free to use all the key features of Firebase, it will have pre-made APIs to connect the platform functionality to the mobile application, so that it can build functionalities very quickly such as digital registration.

#### 2.1.5 Java Programming Language

Java is an object-oriented programming language of Android development that is supported by Android Studio. Java is easy to learn and understand because it is one of the most human-readable languages out there. It can be used in many different kinds of projects such as from web applications to desktop applications to mobile applications no matter their scale. Thus, all the features, functionalities and interface in digital-only banking mobile applications will be developed using Java language.

#### 2.1.6 Summary of the Technologies Review

Description	Concept / Technology
-Open financial data source for TPPs -Pose risk to banks, but at the same time, it also can be opportunity for banks to become better	Open Banking and Open API
Software Platform	Android Studio
Firmware/OS	Android 4.0 and above
Database	Firebase
Programming Language	Java Programming Language

**Table 2.1.5 Summary of the Technologies Review**

## 2.2 Review of the Existing Systems/Applications

This part will focus more on benchmarking existing digital-only banking mobile applications such as Monzo Bank, Monese – Mobile Money Account for UK & Europe, Revolut – Get more from your money, and N26 – The Mobile Bank, in order to illustrate the current practices or existing results to solve the problem.

### 2.2.1 Monzo Bank [31]

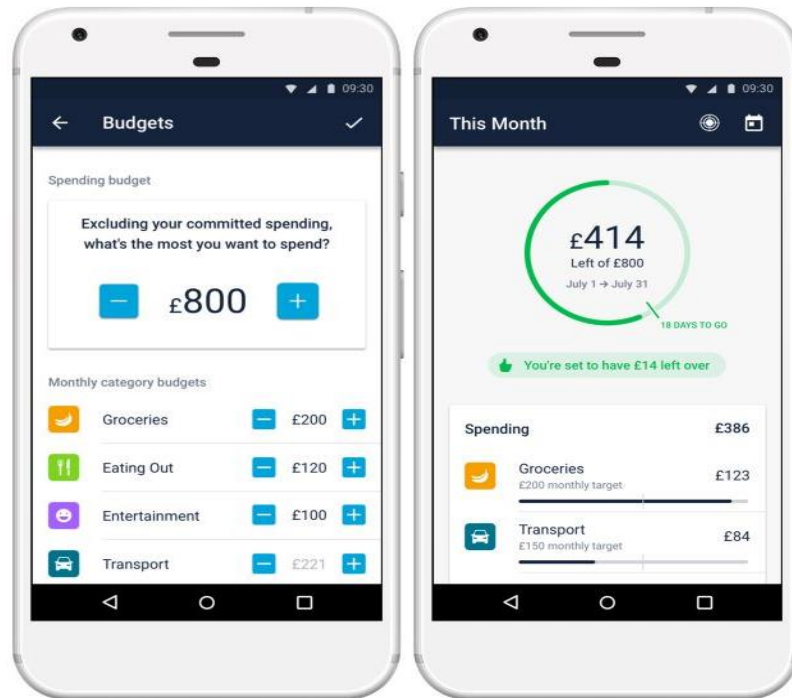


Figure 2.2.1 Monzo Mobile Application Interface [32]

Monzo, which is a digital-only banking mobile application based in the United Kingdom, was founded in 2015 that wins over customers with its intuitive interface, variety of features and attractive visuals. A variety of accounts are currently provided by Monzo, including personal, joint and business accounts. Customers can use Monzo everywhere that accepts a Mastercard in the world and Monzo will not charge any international spending fees and also will not mark up the exchange rate.

### Strengths

- **Provide 24/7 customer support service**

Monzo shows to its customers that it can be assured in getting the customers back at all times by offering customer service on demand. It will be able to solve and fix

customers' problems on the go directly through the mobile phone. If customers are able to get a smooth experience of 24/7 customer support service that will keep the business in mind long after the initial sale [33].

- **Set spending budgets**

Monzo customers are allowed to set a monthly spending budget, so that customers are able to limit themselves to spend how much for each category. When customers spend, it will let customers know how close they are getting for each category to the spending limit.

- **Keep an eye on the day-to-day spending**

If customers spend too quickly, it will be notified immediately to gently remind them. The dial informs the customer how much he/she has already spent and how much it has left. And if Monzo seems like the customer is going to run out of money before the end of month, it will also remind him/her.

- **Offer to turn on loans**

Monzo does offer its customers loans up to 3,000 GBP. Monzo will check if the customer is eligible by checking the customer's credit file to help it to determine whether they can lend to the customer. Customers are able to set the overdraft limit within the amount that Monzo will offer at any time in the mobile application or turn off the overdraft function completely.

- **Request money from/to other users**

Monzo is able to let customers request money from and to friends and family. In order for others to send money to the customer, the customer must provide their username to someone who would like to send money to them.

- **Transfer money locally and internationally**

Monzo links up with Transferwise to allow customers to transfer money abroad at a cheaper rate than other banks. It is free if customers transfer cash to other Monzo

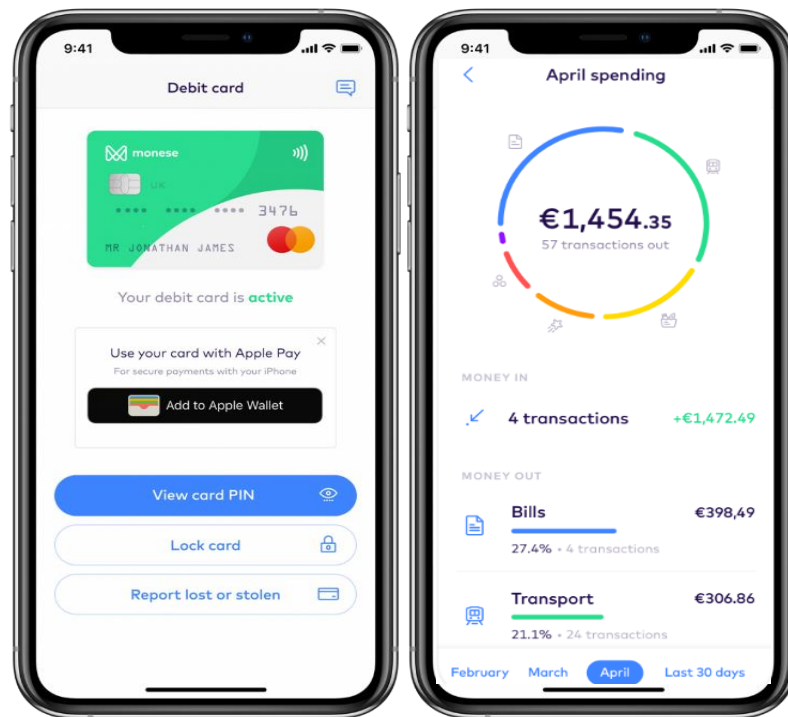
customers using the same currency, while the international transfer fee is around 0.5-3.0 EUR. For foreign transactions, Monzo often charges a currency exchange fee. This, depending on the currency, ranges from 0.35-2.00% of the amount traded. Monzo allows customer to use either pay by bank transfer or pay by scanning QR code.

### **Weaknesses**

- **Do not have a map showing branches and ATMs**

Monzo did not provide a map showing all nearby branches and ATMs. Without these, if customers wish to withdraw cash, then they need to search for the branches and ATMs by themselves to check and see which ATM allows them to retain cash and skip for the bank fee or surcharge.

### 2.2.2 Monese – Mobile Money Account for UK & Europe [34]



**Figure 2.2.2 Monese mobile application interface [35]**

Monese, which is a digital-only banking mobile application released in 2015, is making it one of the most developed digital banks. It was established in the United Kingdom and provided the country's first 100% mobile current account. It provides customers with a fast and simple way to set up a United Kingdom account without having utility bills, proof of residence or credit history. It facilitates transfers via fee-free card and ATM withdrawals overseas, something that people can only dream of with a conventional bank.

#### **Strengths**

- **Set spending limit**

Monese allows customers to set a monthly budget for each category on the account which can visualize it in the Budget screen from the Expenses section. Monese thinks that the monthly budget should be inserted straight into the customer's current account rather than storing it in a spreadsheet, or even a dedicated budgeting application.



- **See the overview of spending**

Monese enables customers to access a new detailed overview of expenses from the entry point, providing insight into everything from the spending on shopping or travel, to household bills. A donut graph allows customers to view their expenditure at a glance, while the transaction list below shows precisely what customers spent, where customers spend it, and at what store.

- **Able to request funds from/to other users**

Monese will provide a handy list of people who are already using Monese in the customer's address book, the customer will be able to pay instantly with a Monese-to-Monese transfer. Besides, customers also can invite the non-Moneser to register and pay the share through the Monese application.

- **Allow to receive/transfer money locally and internationally**

Monese enables customers to transfer money local and overseas; however, Monese can only choose to transfer from the customer's contacts. Able to let customers make an international transfer of funds is important because the money sent by the migrant relatives is nothing less than a lifeline for many families who live in developing countries [36]. For transferring money abroad, Monese usually takes the transfer transaction around 2-4 business days to receive the funds that the customer has transferred.

### **Weaknesses**

- **Lack of customer service support**

Although Monese does provide in-app; however, the in-app chat is open from 8.00am to 5.00pm from Monday to Sunday. Besides that, if after the chatbot was unable to answer the inquiry, the wait was between 8 hours and several days for a personalized response. Monese mentioned that its customer service usually responds within a day.

- **Lack of applying loan facility**

Since Monese does not have a banking license, it cannot provide loans to its customers. Without borrowing loan, it cannot be an emergency fund or short-term credit option that might help customers in an emergency. Then, if there is not enough cash in the account and the mobile application does not provide an overdraft line, the customer may be fined for returning the bill.

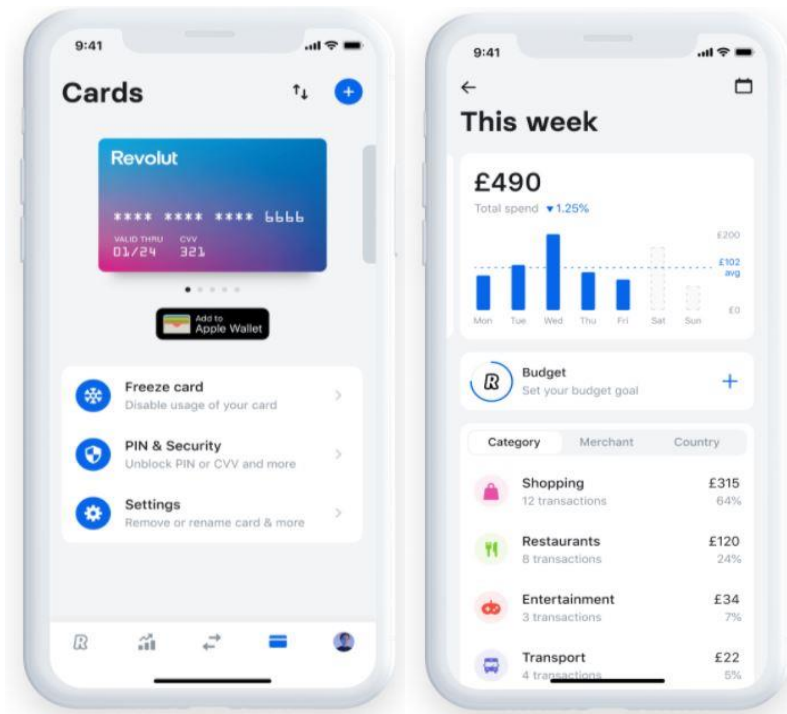
- **Lack of using QR feature to transfer or received funds**

Although Monese does offer customers to transfer money from the contact list, it does not allow customers to use the QR code function to transfer money by scanning other customers' QR codes. In addition, this also means that it cannot generate QR code for the sender to scan it.

- **Lack a map displaying branches and ATMs nearby**

Monese lacks a map displaying all nearby branches and ATMs that accept the customers to withdraw the money there. Since there are quite a lot of places with cash only which means customers can know how to locate the nearest ATM, so that it can benefit customers to withdraw more cash and no need to pay for processing fee.

### 2.2.3 Revolut – Get more from your money [37]



**Figure 2.2.3 Revolut Mobile Application Interface [38]**

Revolut is a creative company for banking and currency trading which is launching in the United Kingdom in 2015 that uses an advanced mobile application and many other innovative methods. It provides personal and business customers with current accounts and a wide variety of money transfer and other services.

#### Strengths

- **Provide 24/7 customer service**

Customer service provided by Revolut is available 24/7, so that customers are able to contact Revolut when they encounter certain situations that require assistance, for example, the customer requested to block his card because the payment card may have been stolen.

- **Set monthly budgets and spending goals**

Revolut can help customers to achieve their spending limit goals by calculating the daily spending limit before customers tell it how much customers will spend each month. In order to monitor customers own spending, they can then set limits in the budgie to help customers to keep on track.

- **Overview the spending amount**

Customers are able to track their consumption patterns through automatic classification. To make useful forecasts that help customers plan ahead, Revolut analyzes their spending. So that customer can get a clearer understanding of where the money goes any time when a payment is made.

- **Make requesting money from/to friends and family**

Revolut does allow customers to send or request money with their friends and family instantly whether they are across the table or the world. If the customer's friends who are already using Revolut will pay the customer from their account with a single tap, but for those who do not use Revolut can choose with a debit or credit card to pay back instantly.

- **Accept local and international transfers and receive**

It is free when transferring in the same currency or using a Revolut account to transfer to any other foreign bank with regular wire transfers. However, it will charge customers if using SWIFT. With the interbank exchange rate, transfer cash abroad in 30+ currencies, with a nominal 0.5% charge on something over 1,000 GBP per month. Revolut allows customers to transfer money to other Revolut users from the friend list, traditional bank transfer [39] or pay by scanning the generated QR code [40].

### **Weaknesses**

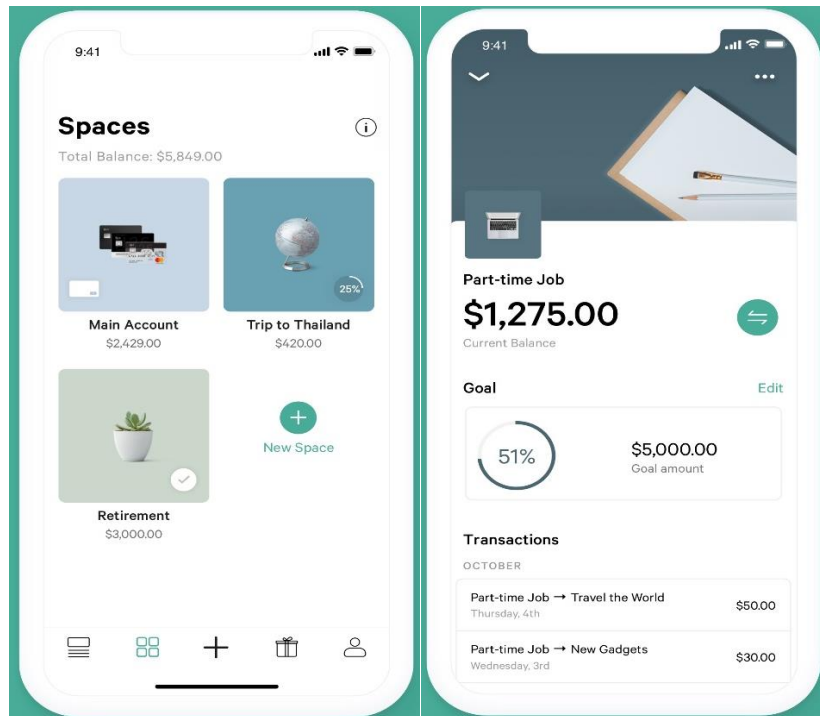
- **Do not provide loan facility**

Customers will not get beneficial when they need a small cash infusion before the next payday due to their budget that cannot afford a larger purchase, then the customer might not consider continuing to use Revolut. That is because Revolut does not offer the borrowing loan facility to customers.

- **Missing a map displaying all nearby branches and ATMs**

Revolut lacks a map displaying all nearby branches and ATMs, which allows customers to view the map and understand where their nearest branch or ATM that accept to withdraw cash from their Revolut account and never pay a bank fee.

### 2.2.4 N26 – The Mobile Bank [41]



**Figure 2.2.4 N26 Mobile Application Interface [42]**

N26 is a digital-only banking mobile application based in the United Kingdom which was released in 2015. N26 promises to make the mobile application easier to save, transfer and control the cash. It offers customers to choose from 4 different types of accounts such as N26 Standard, N26 Smart, N26 You, and N26 Metal.

#### **Strengths**

- **Set budget or spending limit**

N26 provides a feature that is able to let customers set the budget or spending limit. Customers are in control and can decide how much to spend and what to spend the money on. Customers are allowed to set the daily withdrawal limit and also daily payment limit.

- **Allow to keep track of spending**

By looking at the statistical graph, customers can know the exact amount of money spent on each category. Customers will be able to get an up-to-date analysis of current expenditure with automatic spending classification.

- **Provide overdraft and loan facility**

Customers can activate overdraft with their N26 accounts and set the overdraft amount according to their needs. N26 enables customers to get an overview of the cost of overdraft at a transparent cost and keep track of it. However, it only allows customers to get an overdraft with maximum up to 10,000 EUR.

- **Able to request funds from/to other users**

N26 lets customers send or receive up to 3,000 USD per day with other N26 users. For example, customers want to pay for something big like rent or something small like fees of buying groceries, N26 will support customers.

- **Receive/transfer funds locally and internationally**

Customers will be paid as little as possible when transferring money internationally. N26 customers will be beneficial as it follows the real market exchange rate and low fees compared to the conventional banks and other banks that also provide these services. N26 customers can send money by using money beam, which is instant transfer to N26 friends, using bank transfer which is transfer to another bank or scanning the QR code of another N26 customer to perform instant transfer.

### **Weaknesses**

- **Lack of chat feature to support customer service**

Although N26 does provide customers with in-app support, it is only available from Monday to Friday from 8.00am to 8.00pm and Saturday from 8.00am to 7.00pm. If customers need the help outside of working hours, then N26 is not able to provide support and services on demand. Thus, it eventually affects customer satisfaction.

- **Do not provide a map displaying all branches and ATMs**

Without a map displaying all branches and ATMs near customers, Revolut customers will need to find out the branches or ATMs nearby themselves that support the services from N26 in order to withdraw the cash and no need to pay for service charge.

## 2.2.5 Comparison Table

	Monzo	Monese	Revolut	N26	Proposed project
<b>Customer service</b>					
<b>Chat feature</b>	✓	×	✓	×	✓
<b>Features</b>					
<b>Budgeting tools</b>	✓	✓	✓	✓	✓
<b>Overview of spending</b>	✓	✓	✓	✓	✓
<b>Add expense</b>	✓	✓	✓	✓	✓
<b>Loan facility</b>	✓	×	×	✓	✓
<b>Request money from other users</b>	✓	✓	✓	✓	✓
<b>Receive/transfer money locally and internationally using account number</b>	✓	✓	✓	✓	✓
<b>Scan QR code to transfer funds</b>	✓	✓	✓	✓	✓
<b>Generate QR code to receive money</b>	✓	✓	✓	✓	✓
<b>Branches and ATM Map</b>	×	×	×	×	✓

Table 2.2.5 Comparison Between Similar Mobile Application and Proposed Project

## CHAPTER 3 SYSTEM METHOD/APPROACH

### 3.1 Methodology

The methodology that will be applied in this project to developed the digital-only banking mobile application is prototype model as it helps to gain a better understanding of the customers' needs by getting user feedback after each phase completed, so that it can improve or remaking the prototype. The reason to use prototype model because the design of this model is more flexible as it can quickly discover problems and locate any missing functionality when there is a new scope of refinement, which eventually customers can be more satisfied and comfortable because new requirements may be met easily.

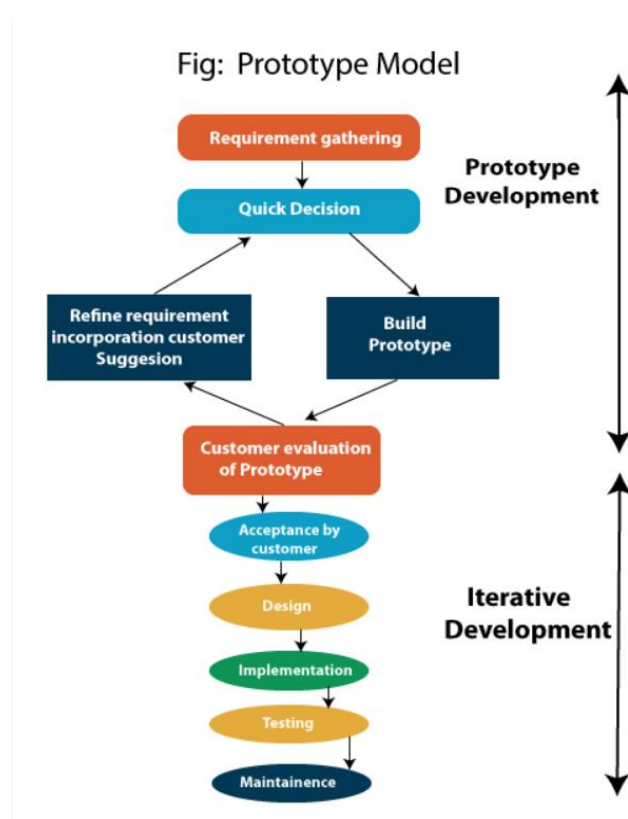


Figure 3.1 Prototype Model [43]

The prototype model has following six phases:

#### Step 1: Requirement gathering and analysis

The first phase of prototyping model is started with planning of the financial system by collecting users' requirements and then gathering the information to analyze it in order to learn what their perceptions of the system are. During the process, the Gantt chart will be used to plan for the schedule of the process of developing this system, so that it



will not go out of schedule. Besides, customers might be happy because they will feel they are also participating in the development process at a very early stage.

**Step 2: Quick decision**

In the second step of prototyping model, it is only a preliminary design or a quick design that only a simple system design will be developed by using the result of the first phase which we will perform an investigation to generate complete details for the implementation of the system; however, it is not complete design yet. The purpose of it is used to give a general idea of the financial system to the users as well as helping in developing the prototype later.

**Step 3: Build prototype**

The third stage, based on the data and results which gathered from preliminary design or quick design, an actual prototype is designed; however, it is only a small working model or part of the financial system required.

**Step 4: Assessment or user evaluation**

In the fourth phrase, the proposed system is introduced to the client for an initial assessment because it can help me to assess the intensity and vulnerability of the working model. We will gather all the comments and feedback received from the client, so that we can know where should improve in the proposed system because it is possible to find missing features in the financial system.

**Step 5: Prototype refinement**

We will need to refine the prototype based on gathered suggestions and feedback, if the client is not satisfied with the current prototype. We would need to refine the prototype based on received suggestions and feedback from client. We need to refine the prototype before it really meets customer needs or else it is not going to be over and still need to continue improvement of the prototype. This fifth stage will be accomplished by the development of a final system based on the approved final prototype, before the client accepts the created prototype.

**Step 6: Engineer product**

In this last stage, if the final system is built on the basis of the final prototype, it is thoroughly tested and deployed for production. Regular maintenance will be carried out on the final system, so it will help minimize downtime and prevent large-scale failures.

### 3.2 System Design Diagram/Equation

following is the format of the scientific formula for calculating the monthly loan payment:

$$p = \frac{a \times r(1 + r)^n}{(1 + r)^n - 1}$$

where

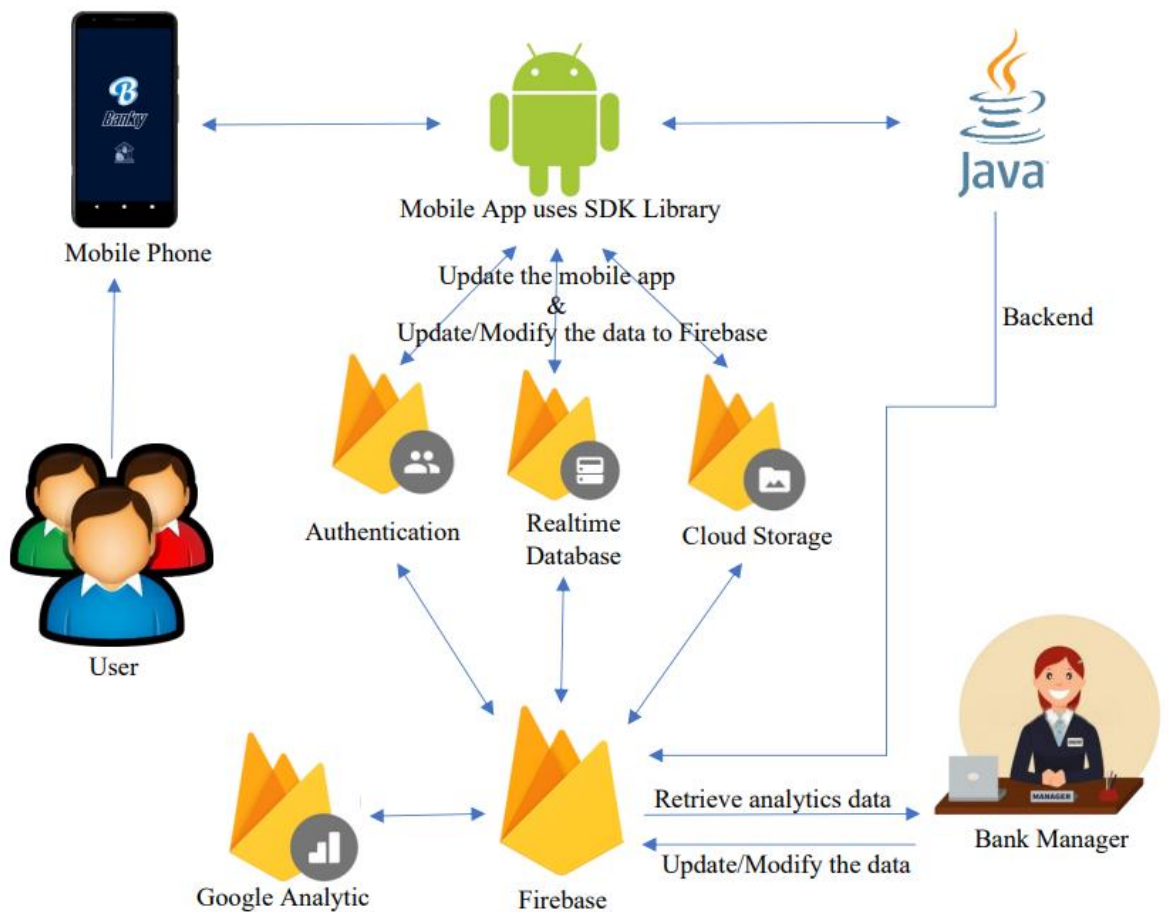
$p$  is monthly payment

$a$  is the amount borrowed

$n$  is number of payments per year (12 for monthly)

$r$  is annual interest rate

#### 3.2.1 System Architecture Diagram



**Figure 3.2.1 System Architecture Diagram**

### 3.2.2 Overall Use Case Diagram of Digital-only Banking Mobile Application

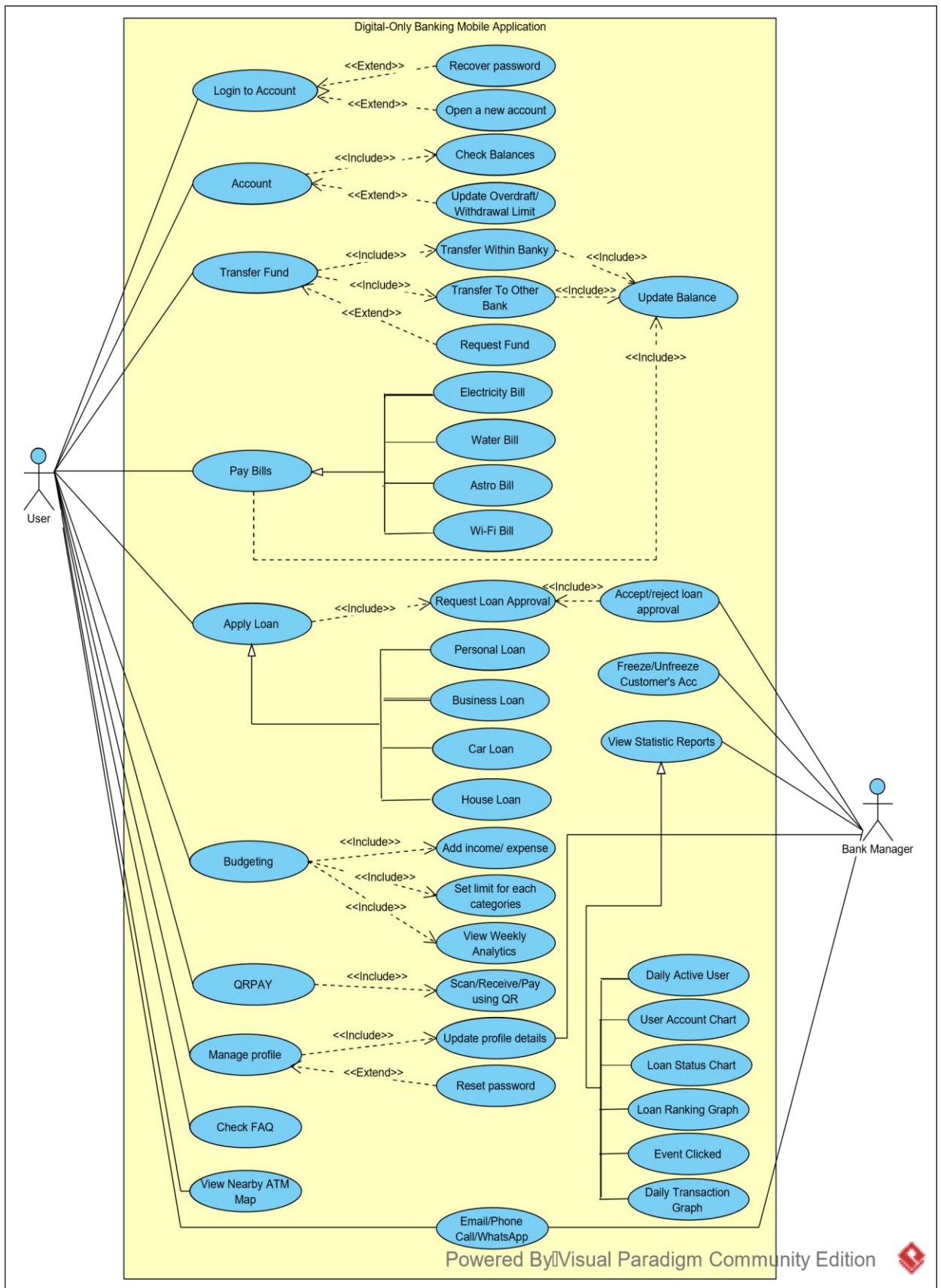
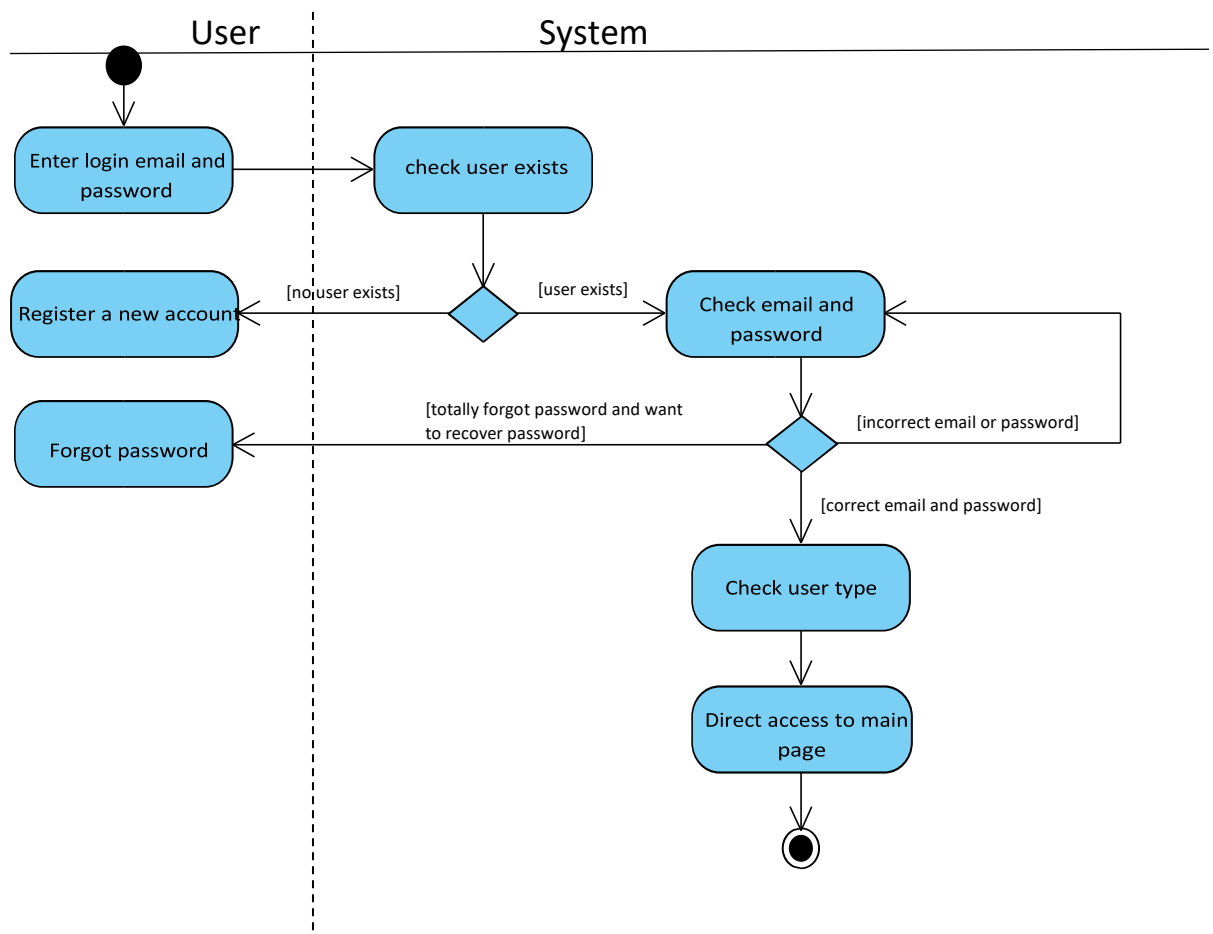


Figure 3.2.2 Overall Use Case Diagram

**3.2.3 Login Module****Use Case Description – Login Module**

Use Case Name: Login to account	ID: <u>  1  </u>	Importance Level: <u>  High  </u>
Primary Actor: User, System	Use Case Type: Detail, Essential	
Stakeholders and Interests:		
<ul style="list-style-type: none"> <li>• User - wants to login to his/her account.</li> </ul>		
Brief Description: This use case describes how we handle the process of login to the users' account.		
Trigger: User wants to view his/her account details.		
Type: External		
Relationships:		
Association: User		
Include: type email and password		
Extend: forgot password, register as a new user		
Generalization:		
Normal Flow of Events:		
<ol style="list-style-type: none"> <li>1. User needs to enter email and password.</li> <li>2. The system will check whether this email/user exists in the system or not.</li> <li>3. The system will check whether the user's email and password correct or not.</li> <li>4. The system will check and validate user type whether he/she is customer or bank manager.</li> <li>5. The system will load all the details of the user's account and direct access to main interface.</li> </ol>		
Sub Flows: Not applicable		
Alternate / Exceptional Flows:		
<ol style="list-style-type: none"> <li>1a. User needs to press "Forgot Password" if he/she forgets the password.</li> <li>2a. User has not registered as digital-only bank customer yet, then he/she needs to press "Not have account? Register" to perform simple registration process.</li> </ol>		

**Table 3.2.3 Use Case Description – Login Module**

**Activity Diagram – Login Module****Figure 3.2.3 Activity Diagram – Login Module**

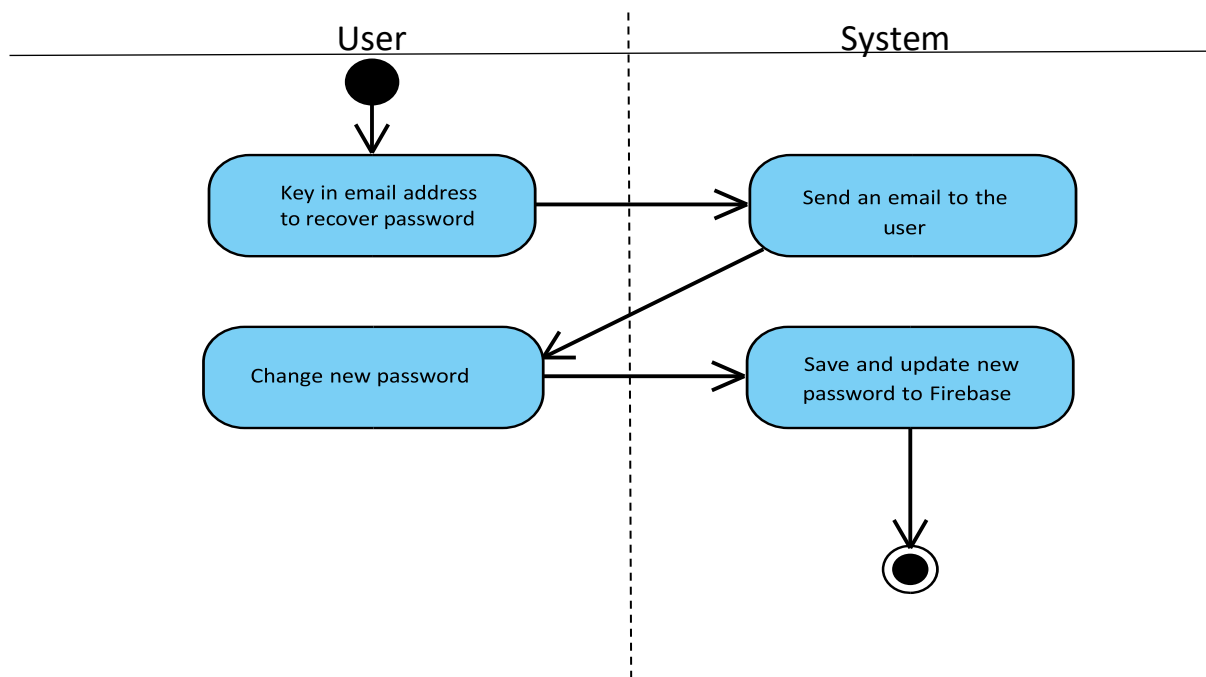
The above figure shows the user's login activity flow. When the user enters the email and password and clicks on "Login" button, the system will then check whether this user exists in the system or not. If no such user exists in the system, then it may require the user to register a new account; else if the user exists in the system, then the system will check whether the email and password correct or not. If the user key in incorrect email or password, the system will prompt a message saying incorrect email or password, and if the user really forgot the password, then he/she can press the "Forgot password" button; else if the email and password are correct, then the system may check for the user type whether the user is customer or bank manager, after that the system will bring the user to access the main page.

### 3.2.4 Forgot Password Module

#### Use Case Description - Forgot Password Module

Use Case Name: Recover password	ID: <u>  2  </u>	Importance Level: <u>  High  </u>
Primary Actor: User, System	Use Case Type: Detail, Essential	
Stakeholders and Interests:		
<ul style="list-style-type: none"> <li>• User – forgot his/her password.</li> </ul>		
Brief Description: This use case describes how we handle the process when user is a user exists in the system and he/she forgot the password.		
Trigger: User wants to recover back his/her password.		
Type: External		
Relationships:		
Association: User		
Include: type email		
Extend:		
Generalization:		
Normal Flow of Events:		
<ol style="list-style-type: none"> <li>1. The user needs to key in his/her email address.</li> <li>2. The system will send an email letter to his/her mailbox.</li> <li>3. The user is required to change the new password over there.</li> <li>4. The system will save and update the new password to Firebase.</li> </ol>		
Sub Flows: Not applicable		
Alternate / Exceptional Flows:		
Not applicable.		

**Table 3.2.4 Use Case Description - Forgot Password Module**

**Activity Diagram – Forgot Password Module**

**Figure 3.2.4 Activity Diagram – Forgot Password Module**

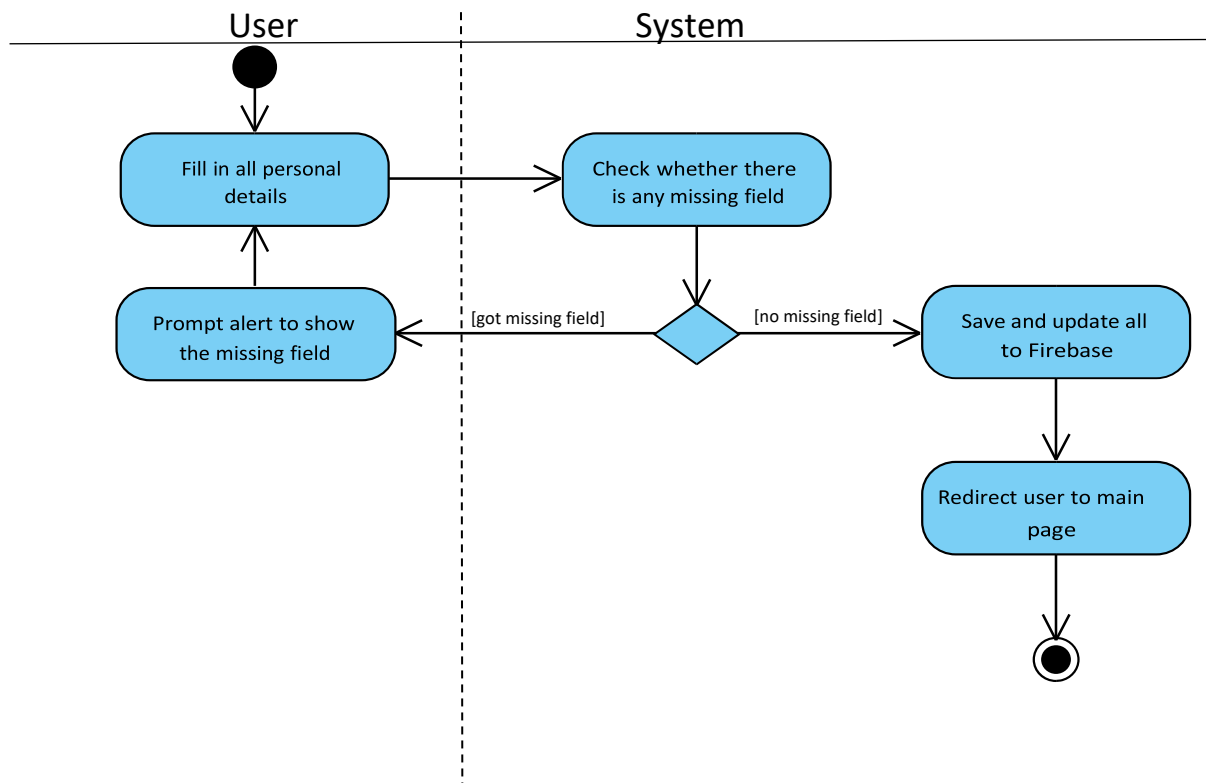
Based on the figure above, it shows that when the user forgot his/her password, he/she needs to click the “Forgot Password” in the Login page, then it will require the user to key in the email address in order to recover password. The system will send an email about reset your password for the project (this digital-only banking mobile application), by clicking on the link that it provided, it will direct the user to reset the password. The user needs to make sure that the new password must have at least 6 characters. Once the password has changed, the system will save and update the new password in Firebase in order to replace the old password. The user can now use the new password to login to main page.

**3.2.5 Register Module****Use Case Description – Register Module**

Use Case Name: Open a new account	ID: <u>3</u>	Importance Level: <u>High</u>
Primary Actor: User, System	Use Case Type: Detail, Essential	
Stakeholders and Interests:		
<ul style="list-style-type: none"> <li>User - wants to register a new digital-only bank account.</li> </ul>		
Brief Description: This use case describes how we handle the process of registering a new user account or setting up simple registration process.		
Trigger: User wants to become digital-only bank user.		
Type: External		
Relationships:		
Association: User		
Include: fill in personal details		
Extend: register as bank manager		
Generalization:		
Normal Flow of Events:		
<ol style="list-style-type: none"> <li>The user registers as a digital-only bank user.</li> <li>The user needs to fill in all the personal details such as full name, identification number, phone number, home address includes country, state, city and complete address, email, password.</li> <li>The system would prompt an alert to tell the user if any field left out to fill in.</li> <li>The system would save and update all the personal details to the firebase.</li> <li>The system would redirect the user to main page.</li> </ol>		
Sub Flows: Not applicable		
Alternate / Exceptional Flows:		
<ol style="list-style-type: none"> <li>If the user is bank manager, then he/she should click the “Register as bank manager”.</li> </ol>		

**Table 3.2.5 Use Case Description - Register Module**



**Activity Diagram – Register Module****Figure 3.2.5 Activity Diagram – Register Module**

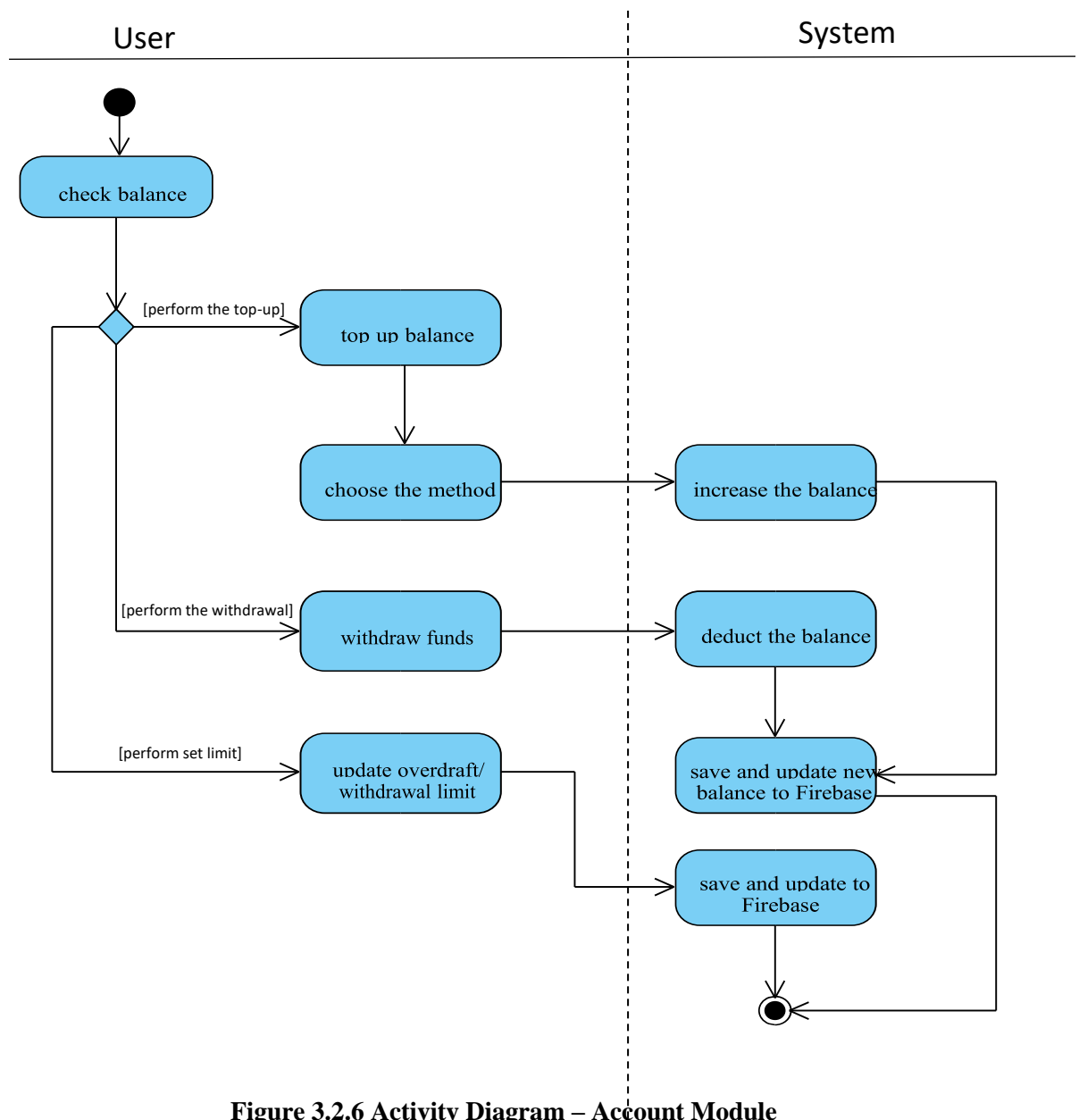
For the register activity diagram, it displays that the user will need to fill in all personal details in particular field such as full name, identification number, phone, home address including country, state, city and complete address, email, password and confirm password. The system will check whether there is any missing field, if there is a missing field(s), then the system will prompt the alert to require the user to fill in that missing field. Else if there is no missing field, then the system will check whether the password and confirm password are exactly the same or not. The system will then save and update all the personal details to Firebase. After that, it will redirect the user to the main page.

### 3.2.6 Account Module

#### Use Case Description – Account Module

Use Case Name: Deposit & Withdraw Funds	ID: <u>4</u>	Importance Level: <u>High</u>
Primary Actor: User, System	Use Case Type: Detail, Essential	
Stakeholders and Interests:		
<ul style="list-style-type: none"> <li>User - wants to check available balance, current balance, fixed deposit and then update overdraft or withdrawal limit.</li> </ul>		
Brief Description: This use case describes how we handle the process of letting the user set new overdraft or withdrawal limit after he/she has been checking the account balances.		
Trigger: User wants to check account balances and update new overdraft or withdrawal limit.		
Type: External		
Relationships:		
Association: User		
Include: check available balance, check current balance, check fixed deposit balance, set withdrawal and overdraft limit		
Extend:		
Generalization:		
Normal Flow of Events:		
<ol style="list-style-type: none"> <li>The user can check his/her account current balance, available balance, and fixed deposit.</li> <li>The user can even set and update the withdrawal and overdraft limit to Firebase.</li> <li>If the user chooses to top up balance, then he/she can choose the method of toping up balance such as using TNG eWallet or going to the nearest 7-eleven to top up/ deposit funds.</li> <li>The system will deposit the balance and update the new balance to Firebase.</li> <li>If the user chooses to withdraw the fund from the digital-only bank account.</li> <li>The system will then deduct the balance and update the new balance to Firebase.</li> </ol>		
Sub Flows: Not applicable		
Alternate / Exceptional Flows: Not applicable.		

**Table 3.2.6 Use Case Description - Account Module**

**Activity Diagram – Account Module****Figure 3.2.6 Activity Diagram – Account Module**

When the user clicks the account icon in the main page, it will display the bank account number, available balance, current balance and also loan balance in order to let the user checks for it. If the user clicks the top up icon, then the user may choose the method of topping up his/her balance, for example, using TNG eWallet or going to the nearest 7-eleven to top up/ deposit funds. Then the system will calculate the latest balance and update it to Firebase. Else if the user clicks the withdrawal icon, then the system will allow the user to withdraw the money from the account, on the other hand, the system will calculate the latest balance and update it to Firebase.

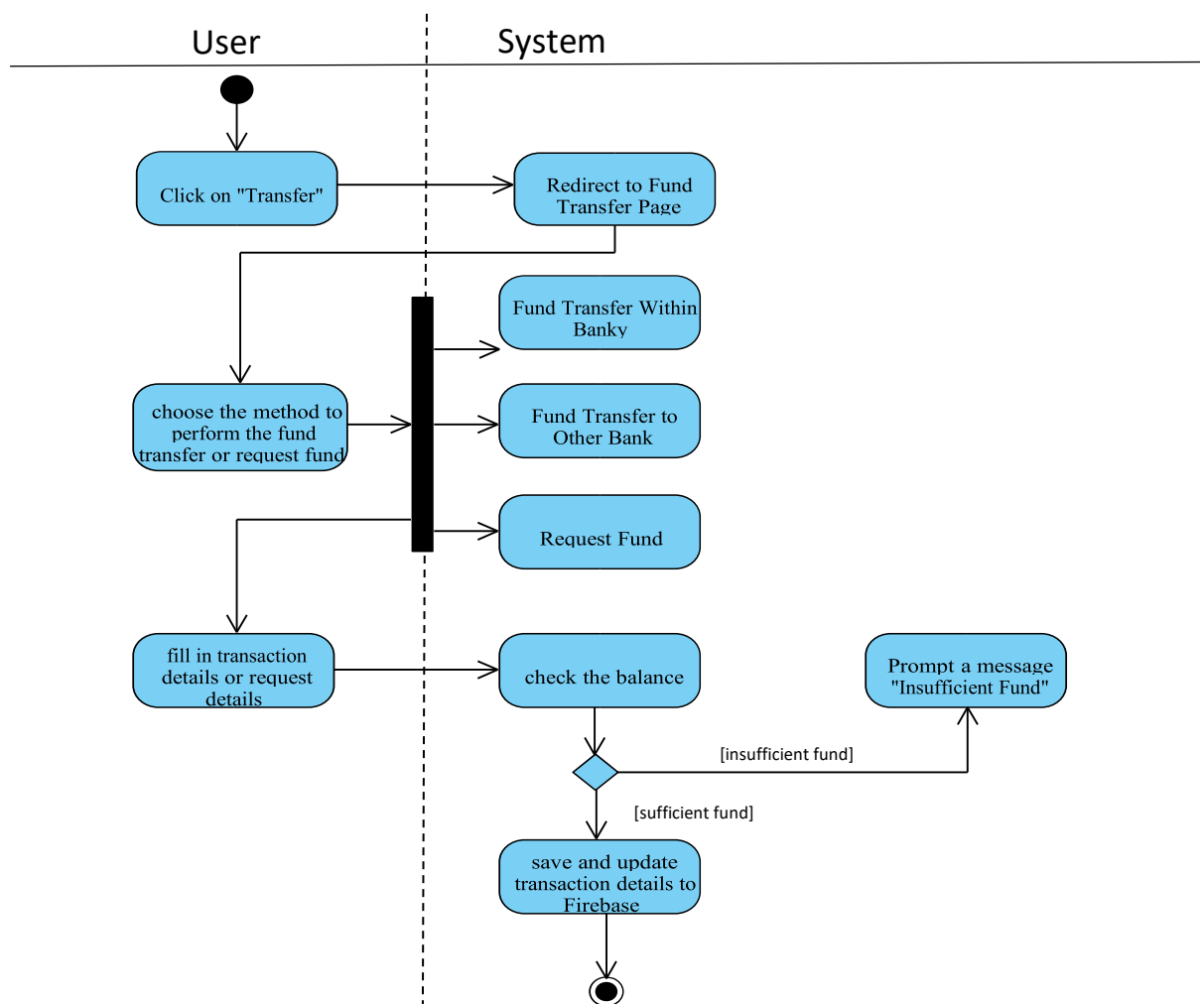
### 3.2.7 Fund Transfer Module

#### Use Case Description – Fund Transfer Module

Use Case Name: Transfer Fund	ID: <u>5</u>	Importance Level: <u>High</u>
Primary Actor: User, System	Use Case Type: Detail, Essential	
Stakeholders and Interests:		
<ul style="list-style-type: none"> <li>• User - wants to transfer fund to others.</li> </ul>		
Brief Description: This use case describes how we handle the process of letting the user to transfer the fund to others whether the other user is from the digital-only bank, or from physical bank.		
Trigger: User wants to transfer funds from his/ her account to others.		
Type: External		
Relationships:		
Association: User		
Include: choose the method to perform the fund transaction, fill in transaction details		
Extend:		
Generalization:		
Normal Flow of Events:		
<ol style="list-style-type: none"> <li>1. The user clicks on the “Transfer”.</li> <li>2. The system will redirect the user to the Fund Transfer page.</li> <li>3. The user can choose different method to perform the fund transfer process.</li> <li>4. The user needs to fill in some transaction details.</li> <li>5. The system will check whether the balance to transfer to the other user enough or not.</li> <li>6. The system will then update the balance and save all the transaction details to Firebase.</li> </ol>		
When the user clicks on “Fund Transfer Within Banky”		
<ol style="list-style-type: none"> <li>4.1 The system will bring the user to the fund transfer within the digital-only banking mobile application interface, then the user needs to fill some transaction details such as account number of the other user, amount to transfer, date to transfer, recipient reference and payment details.</li> </ol>		

<p>When the user clicks on “Fund Transfer to Other Bank”</p> <p>4.2 The system will bring the user to the fund transfer to other bank interface, then the user needs to select the bank and fill in the account number of the other user, recipient name, amount to transfer, date to transfer, recipient reference and payment details.</p> <p>When the user clicks on “Request Fund”</p> <p>4.3 The system will bring the user to the request fund interface, then the user needs to select a contact to enter the requested amount, if there does not have any contact, the user can click the add icon floating button to add a new contact.</p>
<p>Sub Flows: Not applicable.</p>
<p>Alternate / Exceptional Flows: Not applicable.</p>

**Table 3.2.7 Use Case Description – Fund Transfer Module**

**Activity Diagram – Fund Transfer Module****Figure 3.2.7 Activity Diagram – Fund Transfer Module**

When the user clicks on “Transfer” showed in main page, the system will redirect the user to Fund Transfer Page. The user can choose the method to perform the fund transfer transaction, for example, Fund Transfer Within Banky, Fund Transfer to Other Bank, and Request Fund. If the user chooses either of Fund Transfer Within Banky or Fund Transfer to Other Bank, then he/she needs to fill in some transaction details in order to complete the transaction. If the user chooses to Request Fund, then he/she needs to select a contact to enter the requested amount, how if there does not any contact yet, the user can click the add icon floating button to add a new contact. However, before the transaction completed, the system will need to check whether the leftover balance in the account enough to cover it. If there is insufficient fund, then it will stop the transaction and prompt an alert “Insufficient Fund”, else if there is sufficient fund in the account balance, then it will update the balance and save the transaction details to Firebase.

### 3.2.8 Pay Bills Module

#### Use Case Description – Pay Bills Module

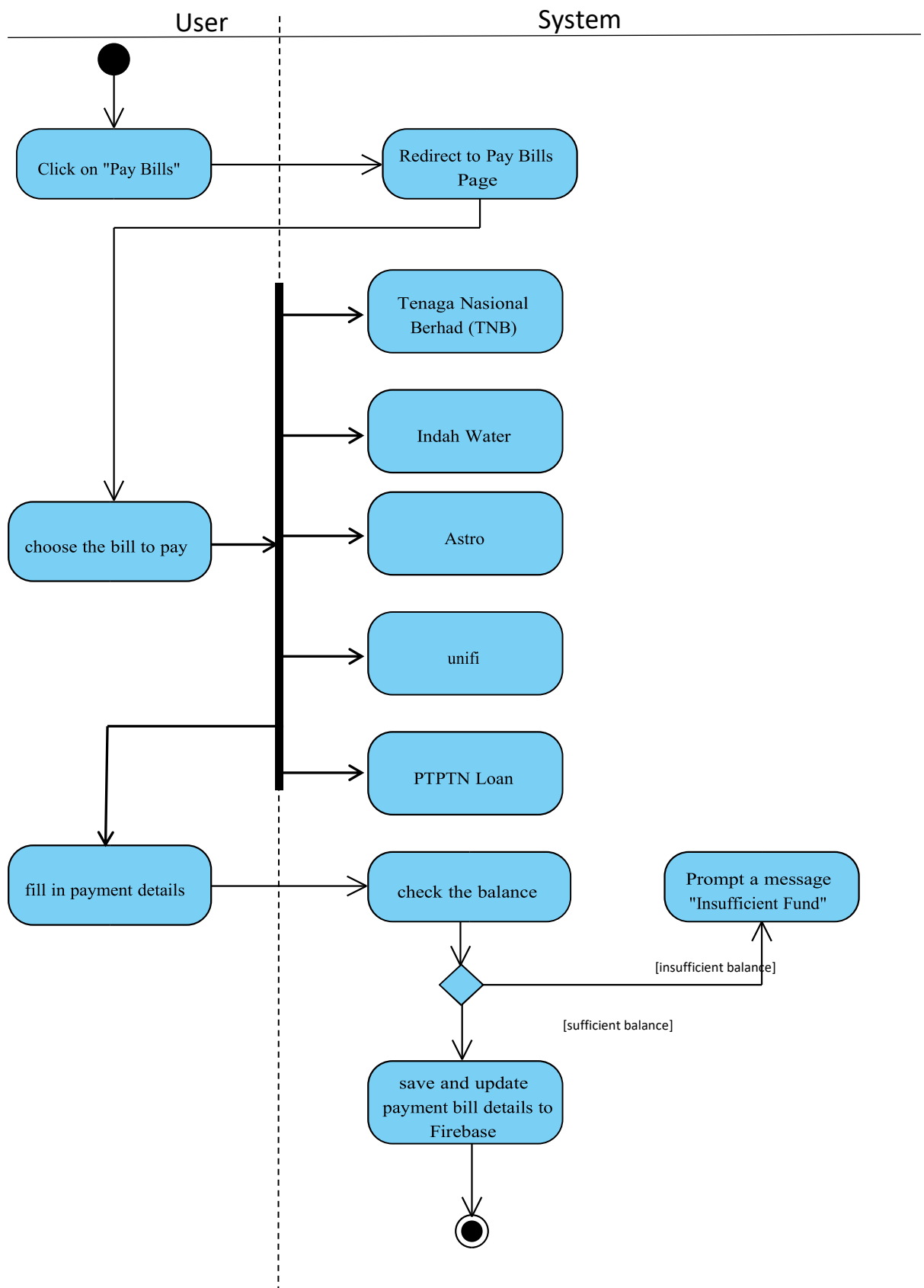
Use Case Name: Pay Bills	ID: <u>6</u>	Importance Level: <u>High</u>
Primary Actor: User, System	Use Case Type: Detail, Essential	
Stakeholders and Interests:		
<ul style="list-style-type: none"> <li>User - wants to pay the bill for either electricity bill, water bill, Astro bill, Wi-Fi bill or others.</li> </ul>		
Brief Description: This use case describes how we handle the process of the user choose the bill to pay for it.		
Trigger: User wants to make payment, for example, pay different utility bills.		
Type: External		
Relationships:		
Association: User		
Include: choose the bill to pay, fill in payment bill details		
Extend:		
Generalization:		
Normal Flow of Events:		
<ol style="list-style-type: none"> <li>The user clicks on the “Pay Bills”.</li> <li>The system will redirect the user to the Pay Bills page.</li> <li>The user can choose different bill to complete the transaction.</li> <li>The user needs to fill in reference number got from the bill, that organization’s account number and amount to pay for it.</li> <li>The system will check whether the balance to transfer to the other user enough or not.</li> <li>The system will then update and save all the bill payment details to Firebase.</li> </ol>		
When the user clicks on “Tenaga Nasional Berhad (TNB)”		
3.1 The system will redirect the page for paying the electricity bill.		
When the user clicks on “Indah Water”		
3.2 The system will redirect the page for paying the water bill.		

<p>When the user clicks on “Astro”</p> <p>3.3 The system will redirect the page for paying the Astro bill.</p> <p>When the user clicks on “unifi”</p> <p>3.4 The system will redirect the page for paying the Wi-Fi bill.</p> <p>When the user clicks on “PTPTN Loan”</p> <p>3.5 The system will redirect the page for paying the Perbadanan Tabung Pendidikan Tinggi Nasional (PTPTN) study loan bill.</p>
<p>Sub Flows: Not applicable</p>
<p>Alternate / Exceptional Flows:</p> <p>Not applicable.</p>

**Table 3.2.8 Use Case Description – Pay Bills Module**



**Activity Diagram – Pay Bills Module**



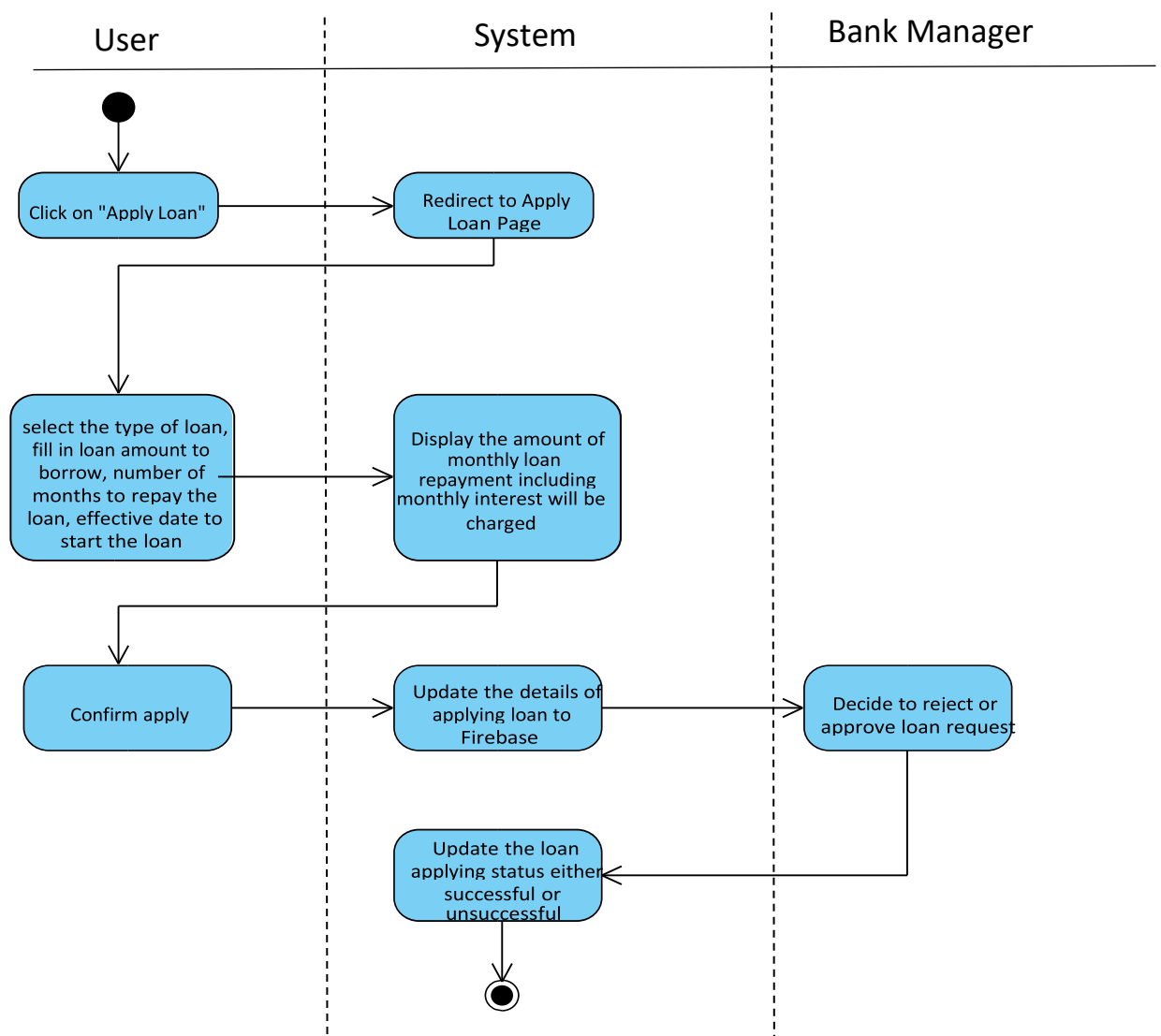
**Figure 3.2.8 Activity Diagram – Pay Bills Module**

When the user clicks on “Pay Bills” showed in main page, the system will redirect the user to Pay Bills Page. The user can pay for different bills, for example, TNB, Indah Water, Astro, unifi or PTPTN Loan. The user will need to fill in the payment bill details such as reference number, the receiver company’s account number and amount to pay. Then the system will check whether the account balance enough to cover it. If there is insufficient fund, then it will stop the payment and prompt an alert “Insufficient Fund”; else if there is sufficient fund in the account balance, then it will update the balance and save the payment bill details to Firebase.

**3.2.9 Apply Loan Module****Use Case Description – Apply Loan Module**

Use Case Name: Apply Loan	ID: <u>7</u>	Importance Level: <u>High</u>
Primary Actor: User, Bank Manager, System	Use Case Type: Detail, Essential	
Stakeholders and Interests:		
<ul style="list-style-type: none"> <li>• User - wants to apply loan from digital-only bank.</li> </ul>		
Brief Description: This use case describes how we handle the process of letting the user to apply loan, such as personal, business, car loan or other loans.		
Trigger: User wants to apply for loan.		
Type: External		
Relationships:		
Association: User		
Include: fill in loan amount to borrow, the number of months to repay the loan, the effective date to start the loan		
Extend:		
Generalization:		
Normal Flow of Events:		
<ol style="list-style-type: none"> <li>1. The user clicks on the “Apply Loan”.</li> <li>2. The system will redirect the user to the Apply Loan page.</li> <li>3. The user will need to select the type of loan to be applied, fill in loan amount to borrow, the number of months to repay the loan and the effective date to start the loan.</li> <li>4. The system will display the amount of monthly loan repayment the user needs to pay, and monthly interest will be charged.</li> <li>5. If the user confirms to apply loan, the system will then update the details of applying loan to Firebase.</li> <li>6. It will take 2-3 working days to proceed the procedures because the bank manager needs to decide whether to accept or reject the loan request by looking at the user’s credit score.</li> <li>7. The system will update the user about the status of whether the process of applying loan successful or unsuccessful.</li> </ol>		
Sub Flows: Not applicable		
Alternate / Exceptional Flows: Not applicable.		

**Table 3.2.9 Use Case Description – Apply Loan Module**

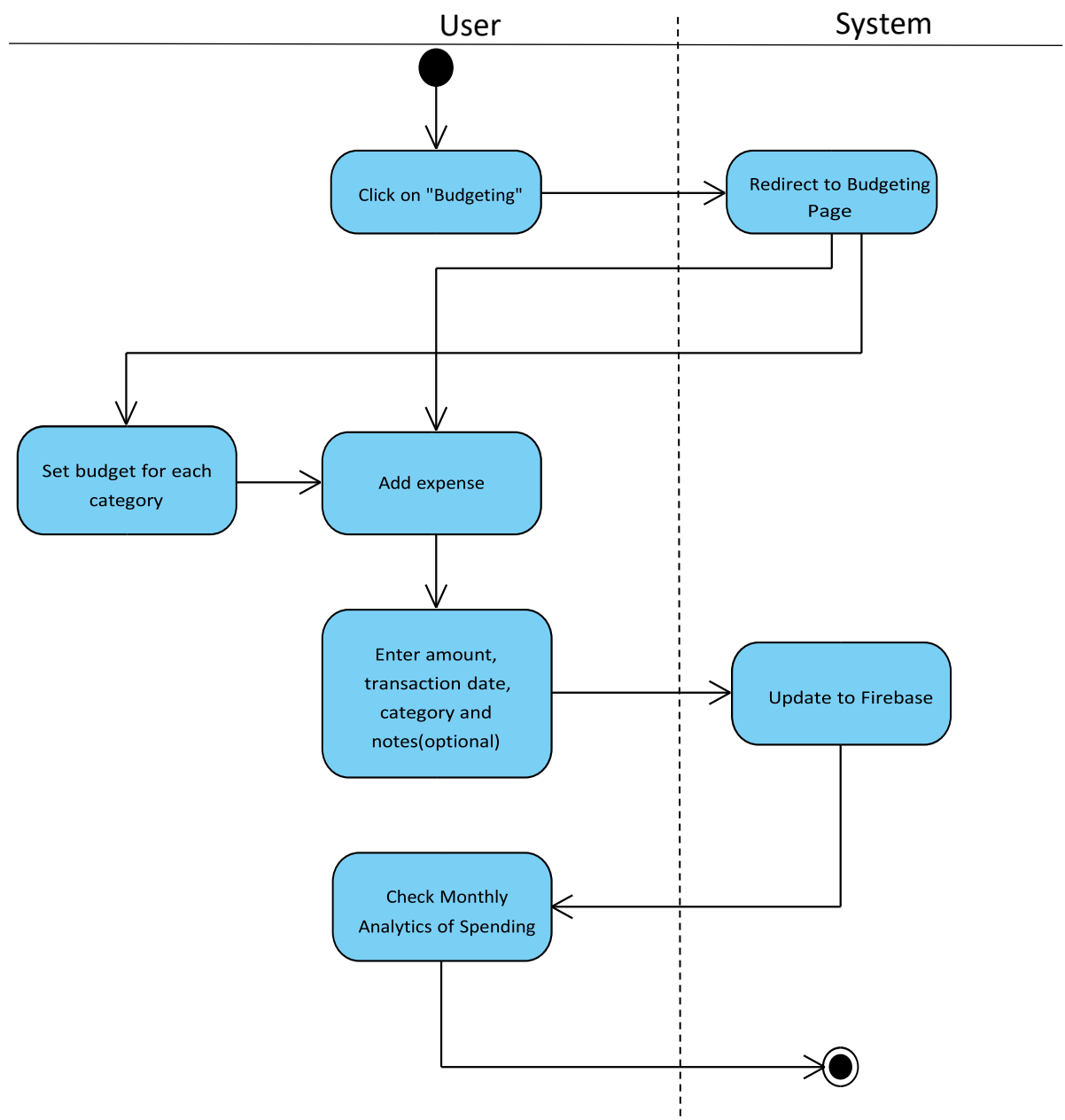
**Activity Diagram – Apply Loan Module****Figure 3.2.9 Activity Diagram – Apply Loan Module**

When the user clicks on “Apply Loan” showed in main page, the system will redirect the user to Apply Loan Page. The user needs to select the type of loan to be applied and fill in all the loan applying details such as the amount of money borrowed on a loan, number of months to repay the loan and effective date to start the loan. The system will then display the amount of monthly loan repayment including the monthly interest will be charged. Once the user confirms to apply the loan, the system will only update the details of applying loan to Firebase. After that, it will be the responsibility for the bank manager exists to decide whether to reject or approve the loan request by looking at the user’s credit score. The system will update the loan applying status to the user whether apply successful or unsuccessful.

**3.2.10 Budgeting Module****Use Case Description – Budgeting Module**

Use Case Name: Use budgeting tool	ID: <u>8</u>	Importance Level: <u>High</u>
Primary Actor: User, System	Use Case Type: Detail, Essential	
Stakeholders and Interests:		
<ul style="list-style-type: none"> <li>User - wants to have clear visibility over their personal finance and to have an overview of their expenditure.</li> </ul>		
Brief Description: This use case describes how we handle the process of letting the user use the budgeting tool to have an overall of their expenditure and also can set limit for each category in order to avoid exceed their spending limit.		
Trigger: User wants to use budgeting tool to overview personal finance.		
Type: External		
Relationships:		
Association: User		
Include: add expense, check weekly analytics of the spending, set budget for each category		
Extend:		
Generalization:		
Normal Flow of Events:		
<ol style="list-style-type: none"> <li>The user clicks on the “Budgeting”.</li> <li>The system will redirect the user to the Budgeting page.</li> <li>The user can add expense, then the system will update it to Firebase.</li> <li>The user can check weekly analytics of the spending.</li> </ol>		
Sub Flows: Not applicable		
Alternate / Exceptional Flows:		
<ol style="list-style-type: none"> <li>3a. The user can set budget for each category, so that the user can try to avoid exceed spending.</li> </ol>		

**Table 3.2.10 Use Case Description - Budgeting Module**

**Activity Diagram – Budgeting Module**

**Figure 3.2.10 Activity Diagram – Budgeting Module**

When the user clicks on “Budgeting” showed in main page, the system will redirect the user to Budgeting Page. The user can set limit for each category, but this is optional. However, the user needs to add expense, then the system will only have the data to display in the weekly analytics of spending. The user will require to select the category from drop down list and enter the amount, transaction date and notes (optional). After that the system will update all the information from that expense to Firebase.

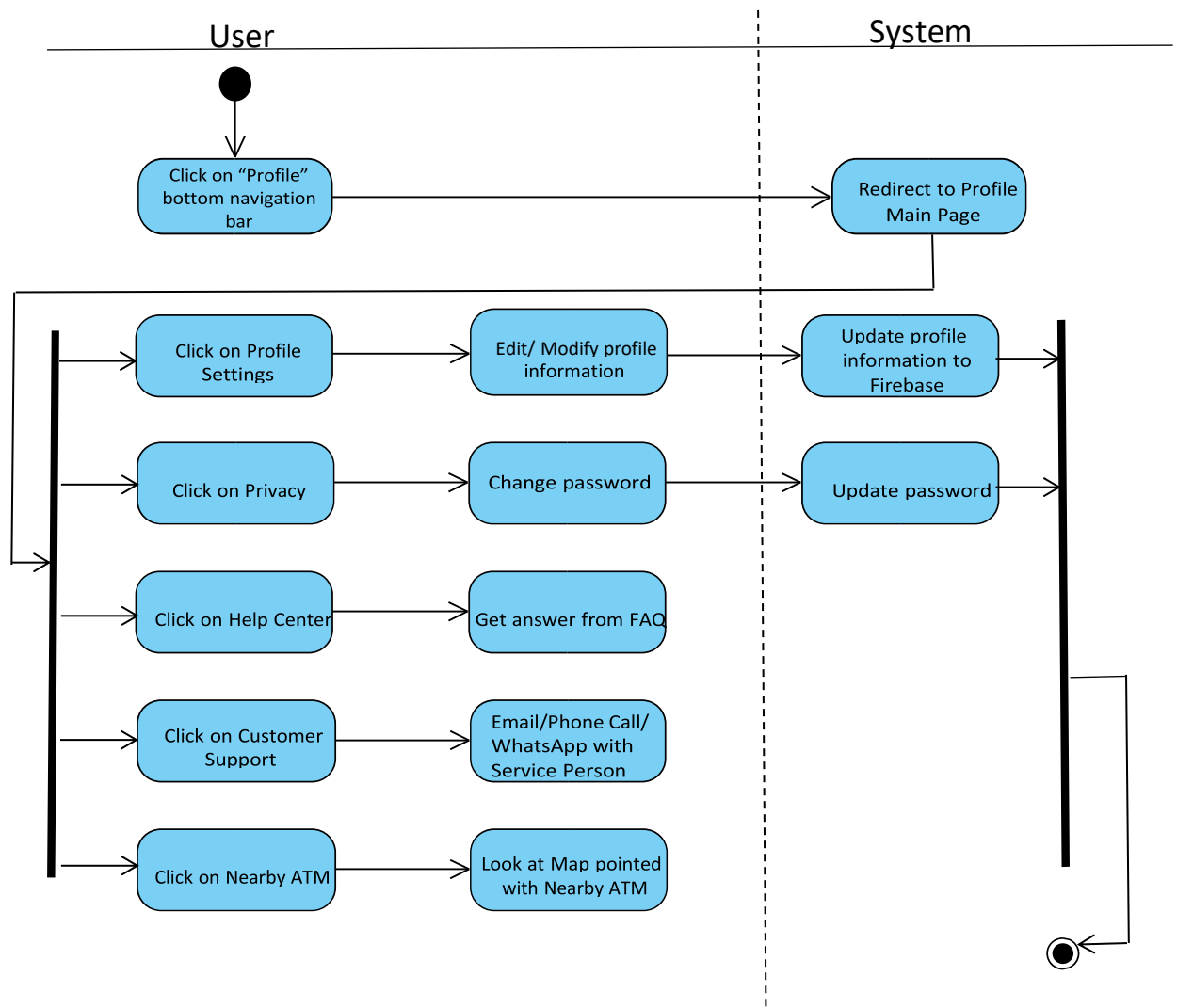
**3.2.11 Profile Module****Use Case Description – Profile Module**

Use Case Name: Profile	ID: <u>  9  </u>	Importance Level: <u>  High  </u>
Primary Actor: User, System	Use Case Type: Detail, Essential	
Stakeholders and Interests:		
<ul style="list-style-type: none"> <li>• User - wants to change his/her profile details or change the account password.</li> </ul>		
Brief Description: This use case describes how we handle the process of letting the user to update the profile details or account password.		
Trigger: User wants to use manage profile to update profile details.		
Type: External		
Relationships:		
Association: User		
Include: update profile details such as name, identification number, phone number, home address or email address, change the account password.		
Extend:		
Generalization:		
Normal Flow of Events:		
<ol style="list-style-type: none"> <li>1. The user clicks on the “Profile” over bottom navigation bar there.</li> <li>2. The system will redirect the user to the Profile Main page.</li> <li>3. The user can click any button to redirect to another page.</li> </ol>		
When the user clicks on “Profile Settings”		
3.1 The system will redirect the page to update and modify profile.		
When the user clicks on “Privacy”		
3.2 The system will redirect the page to change the password.		
When the user clicks on “Help Center”		
3.3 The system will redirect the page to display the frequent ask questions and its supported answers.		
When the user clicks on “Customer Support”		

<p>3.4 The system will redirect the page to let the user chat with the bank manager either by using Email, Phone call or WhatsApp.</p> <p>When the user clicks on “Nearby ATM”</p> <p>The system will redirect the page to display the Map showing nearby ATM Location.</p>
<p>Sub Flows: Not applicable</p>
<p>Alternate / Exceptional Flows:</p> <p>Not applicable.</p>

**Table 3.2.11 Use Case Description - Profile Module**



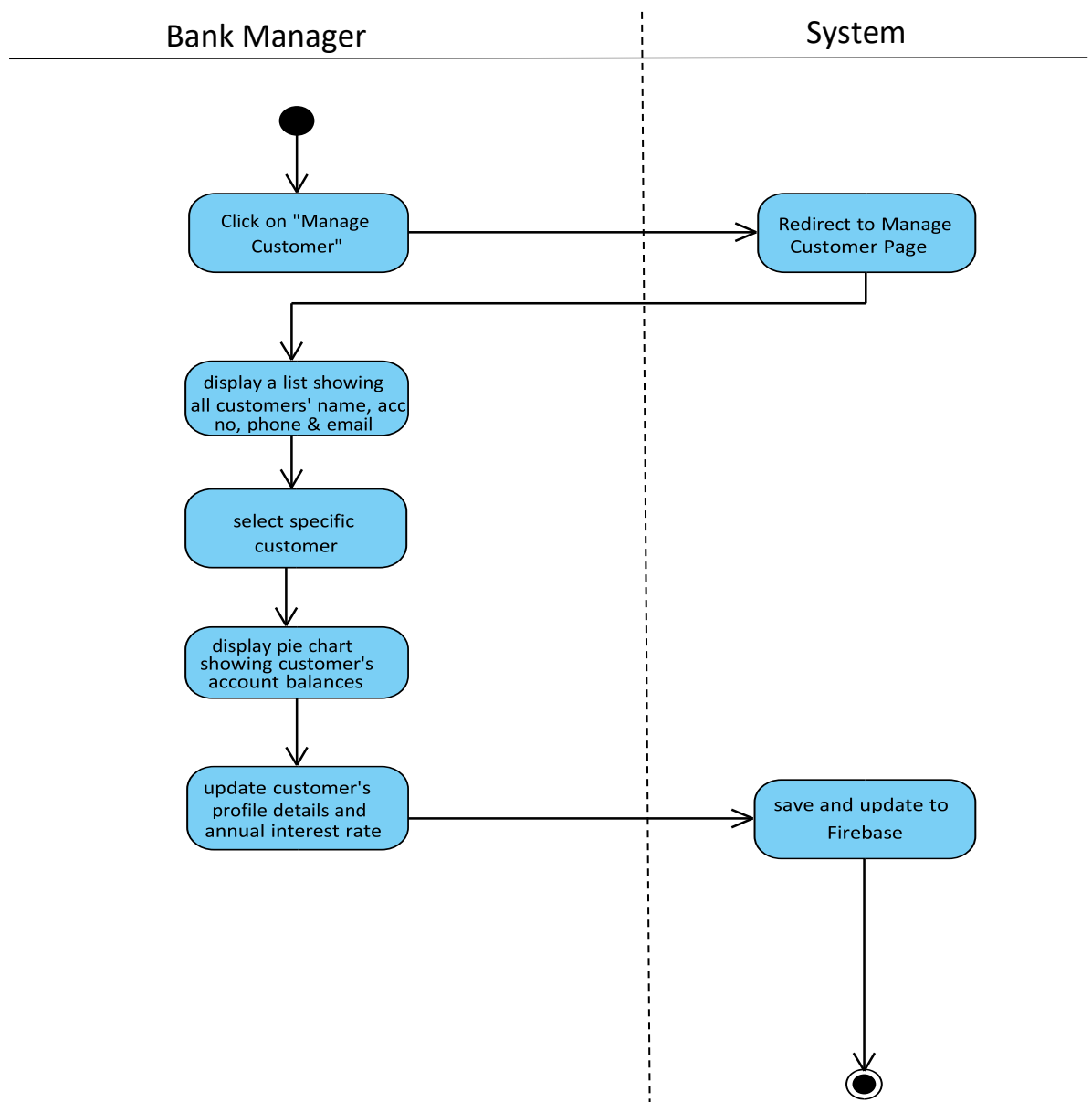
**Activity Diagram – Profile Module****Figure 3.2.11 Activity Diagram – Profile Module**

When the user clicks on “Profile” over the bottom navigation bar there, the system will redirect the user to Profile Main Page. The user can click on the Profile Settings, if he/she wishes to edit or modify the personal profile information, the system will then update the latest profile information to Firebase. If the user wants to change password, then he/she can click on Privacy, after that the system will update the password to newest password. If the user has any enquiry, then he/she can click on Help Center, so basically there will have some Frequent Answer Question normally asked by the other user along with its supported answer. Else if the user still thinks the Help Center cannot solve his/her problem, then he/she can click on Customer Support, there will be a customer service person always there for the user through Email/Phone Call/WhatsApp. The user can view the Nearby ATM on the Map after he/she clicks on the Nearby ATM.

**3.2.12 Manage Customer Module****Use Case Description – Manager Customer Module**

Use Case Name: Manage Customer	ID: <u>10</u>	Importance Level: <u>High</u>
Primary Actor: Bank Manager, System	Use Case Type: Detail, Essential	
Stakeholders and Interests:		
<ul style="list-style-type: none"> <li>Bank Manager - wants to view customer's account balances or update customer's profile details.</li> </ul>		
Brief Description: This use case describes how we handle the process of letting the bank manager to have a look of customers' account balances or to update profile details.		
Trigger: Bank Manager wants to use manage customers to check customers' account balances or update customers' profile details.		
Type: External		
Relationships:		
Association: Bank Manager		
Include: display account balances, update customer's profile		
Extend:		
Generalization:		
Normal Flow of Events:		
<ol style="list-style-type: none"> <li>The bank manager clicks on the "Manage Customer".</li> <li>The system will redirect the bank manager to manage customer page.</li> <li>The system will display a list showing all customers' name, acc no, phone no, email.</li> <li>The bank manager is able to click any one of the customers to view the pie chart showing the customer's account balances.</li> <li>The bank manager allows to edit the customer's profile such as name, identification number, Bank account number, phone number, email and even annual interest rate, the system will then update the profile details to Firebase.</li> </ol>		
Sub Flows: Not applicable		
Alternate / Exceptional Flows:		
Not applicable.		

**Table 3.2.12 Use Case Description – Manage Customer Module**

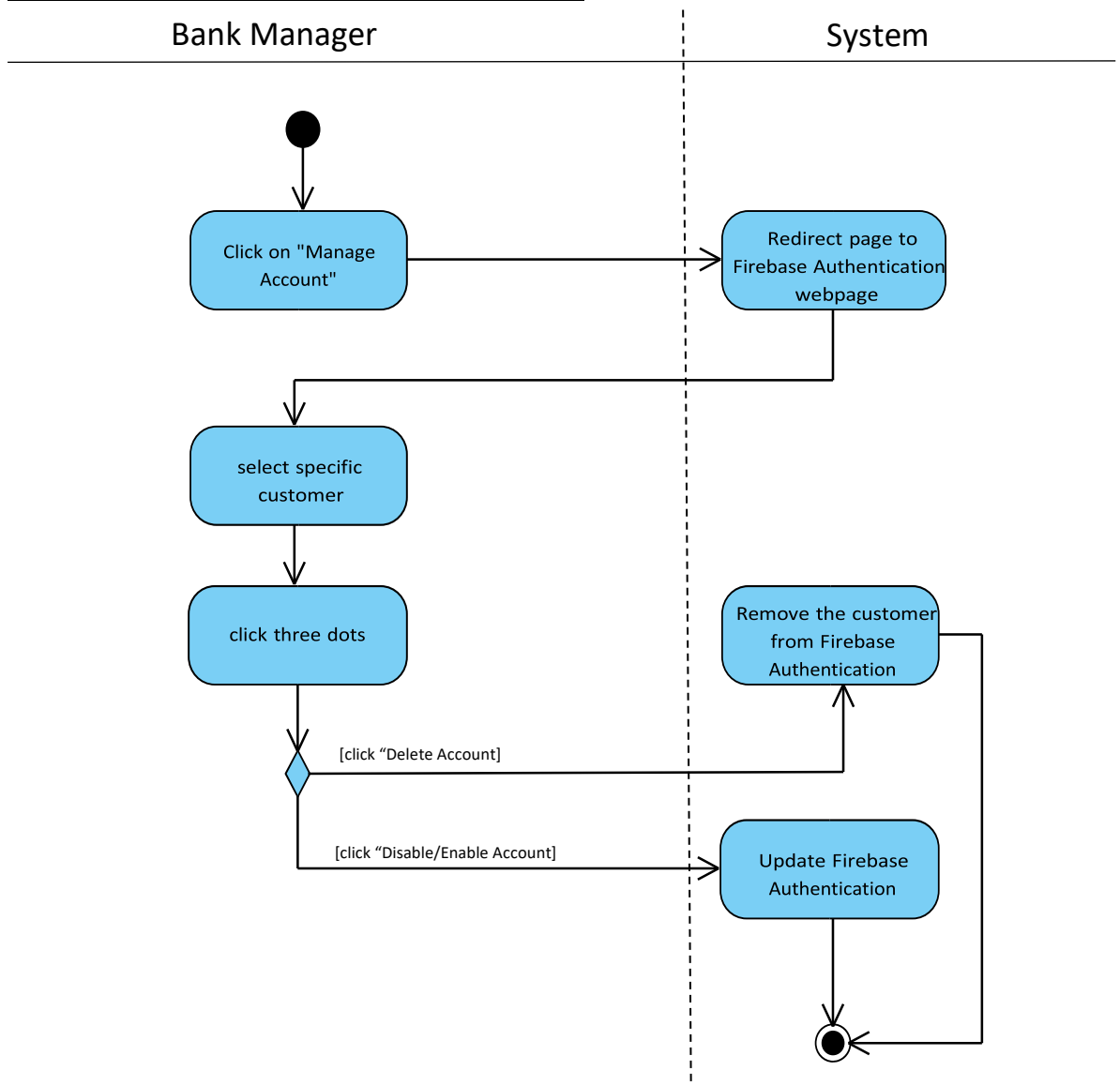
**Activity Diagram – Manage Customer Module****Figure 3.2.12 Activity Diagram – Manage Customer Module**

When the bank manager clicks on “Manage Customer”, it will redirect the bank manager to manage customer page. Inside the manage customer page, it will have a list row-by-row showing all customers’ name, account number, phone number and email address. The bank manager can click any customer that he/she would like to view a pie chart created by using the amount of that customer’s available balance, current balance, fixed deposit, total received, total sent, total payment made. Besides that, the bank manager can turn on the switch, and the system will display the above mentioned customer profile, as well as the identification number and annual interest rate, for the bank manager to edit and update, and the edited data will be updated in Firebase.

**3.2.13 Manage Account Module****Use Case Description – Manager Account Module**

Use Case Name: Manage Account	ID: <u>11</u>	Importance Level: <u>High</u>
Primary Actor: Bank Manager, System	Use Case Type: Detail, Essential	
Stakeholders and Interests:		
<ul style="list-style-type: none"> <li>Bank Manager - wants to freeze or unfreeze the customer's account.</li> </ul>		
Brief Description: This use case describes how we handle the process of letting the bank manager to freeze or unfreeze the customer's account.		
Trigger: Bank Manager wants to use manage account to freeze or unfreeze the customer's account.		
Type: External		
Relationships:		
Association: Bank Manager		
Include: disable account, unable account, delete account		
Extend:		
Generalization:		
Normal Flow of Events:		
<ol style="list-style-type: none"> <li>The bank manager clicks on the "Manage Account".</li> <li>The system will redirect the bank manager to Firebase Authentication webpage.</li> <li>The bank manager can hover over the customer whose account he/she wishes to freeze by clicking the three dots.</li> <li>The Firebase Authentication will then disable or unable the customer.</li> <li>If the bank manager clicks on "Delete Account", the Firebase Authentication will delete the customer immediately, so that the customer will never ever login with the deleted email and password anymore.</li> </ol>		
Sub Flows: Not applicable		
Alternate / Exceptional Flows:		
Not applicable.		

**Table 3.2.13 Use Case Description – Manage Account Module**

**Activity Diagram – Manage Account Module****Figure 3.2.13 Activity Diagram – Manage Account Module**

When the bank manager clicks on “Manage Account”, the system will redirect the bank manager to the Banky Firebase Authentication webpage. This allows the bank manager to choose customers to freeze or unfreeze accounts when unusual transactions or actions are detected by clicking the “Disable Account” or “Enable Account”. When that customer showing “Disable” in Firebase Authentication there, the customer cannot log in and will only be able to log in to the Banky mobile application unless the bank manager re-enables the customer. However, if the bank manager clicks “Delete Account”, the customer will no longer be logged into the Banky mobile application. Therefore, usually the bank manager will not delete or remove a customer easily, he/she will just disable the customer account instead of deleting the customer account directly.

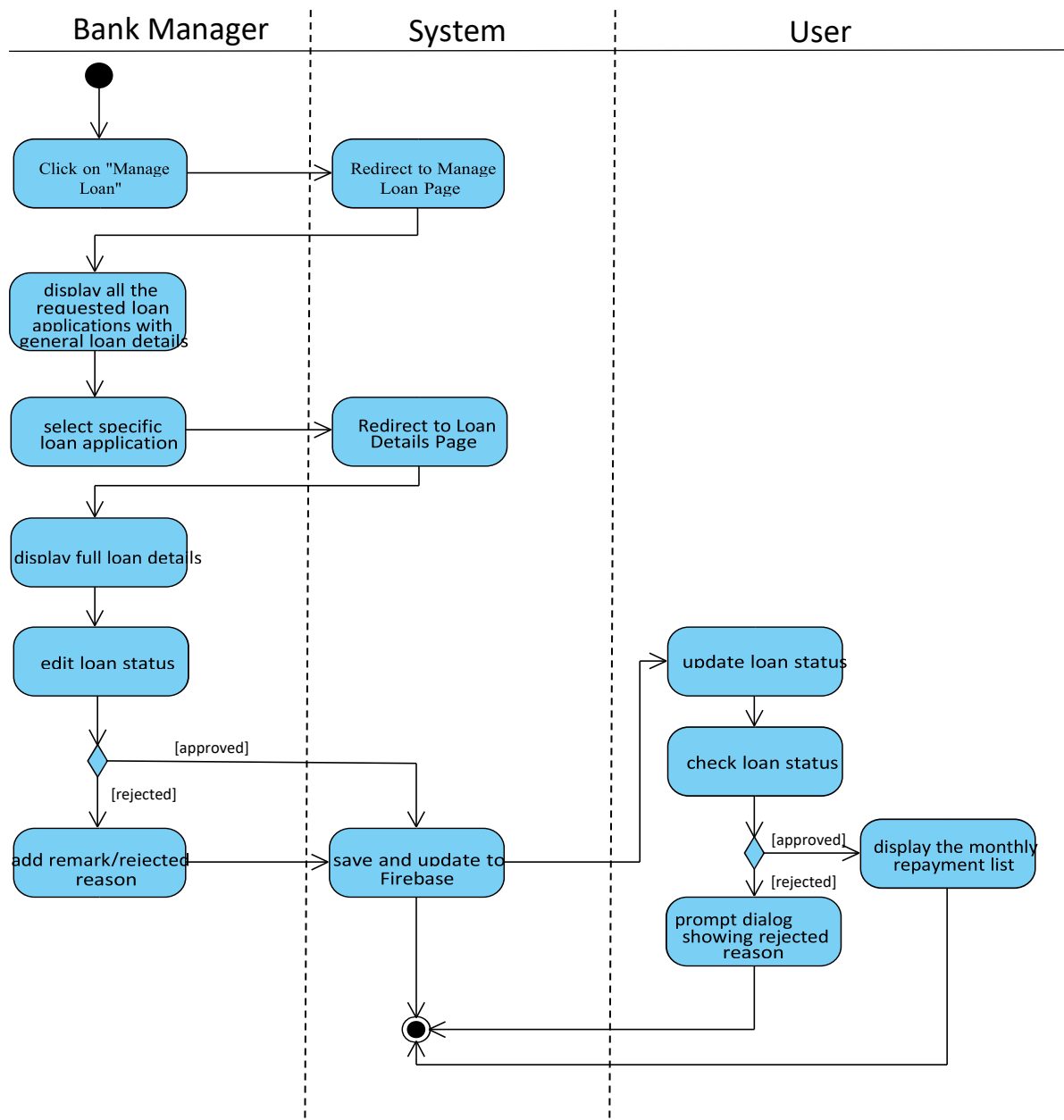
**3.2.14 Manage Loan Module****Use Case Description – Manager Loan Module**

Use Case Name: Manage Loan	ID: <u>12</u>	Importance Level: <u>High</u>
Primary Actor: Bank Manager, System	Use Case Type: Detail, Essential	
Stakeholders and Interests:		
<ul style="list-style-type: none"> <li>Bank Manager - wants to update customers' loan status and add remark for the loan.</li> </ul>		
Brief Description: This use case describes how we handle the process of letting the bank manager to update customer's loan status either by pending, approved, or rejected as well as letting the bank manager to add remark if the loan is rejected.		
Trigger: Bank Manager wants to use manage loan to edit customers' loan status and add remark.		
Type: External		
Relationships:		
Association: Bank Manager		
Include: update loan status, add remark for rejected loan		
Extend:		
Generalization:		
Normal Flow of Events:		
<ol style="list-style-type: none"> <li>The bank manager clicks on the "Manage Loan".</li> <li>The system will redirect the bank manager to Manage Loan Page.</li> <li>The system will display a list of requested loan applications which includes the requested id, loan status, applicant's name, loan name, loan interest rate, loan amount requested, and months to pay back as well as requested date.</li> <li>The bank manager can click on any one of the requested loan applications, then the system will redirect the page to display the full loan application details including monthly repayment, total interest charged and total cost that the applicant needs to pay back.</li> <li>The bank manager can then click on the edit button to update the loan status either by pending, approved or rejected.</li> </ol>		

6. If the bank manager rejected the loan application, then he/she needs to add remark or rejected reason.
Sub Flows: Not applicable
Alternate / Exceptional Flows: Not applicable.

**Table 3.2.14 Use Case Description – Manage Loan Module**

**Activity Diagram – Manage Loan Module**



**Figure 3.2.14 Activity Diagram – Manage Loan Module**

When the bank manager clicks on “Manage Loan”, the system will redirect the bank manager to the Manage Loan Page. The system will display a list of requested loan applications with the requested id, loan status, applicant’s name, loan name, loan interest rate, requested loan amount, months to pay back the loan and requested date & time. The bank manager can click either one of the loan applications, the system will then redirect to Loan Details Page display full loan details includes monthly repayment amount, total interest charged, total cost that the customer needs to pay back. After the bank manager has checking the credit scoring of the applicant, he/she can update the



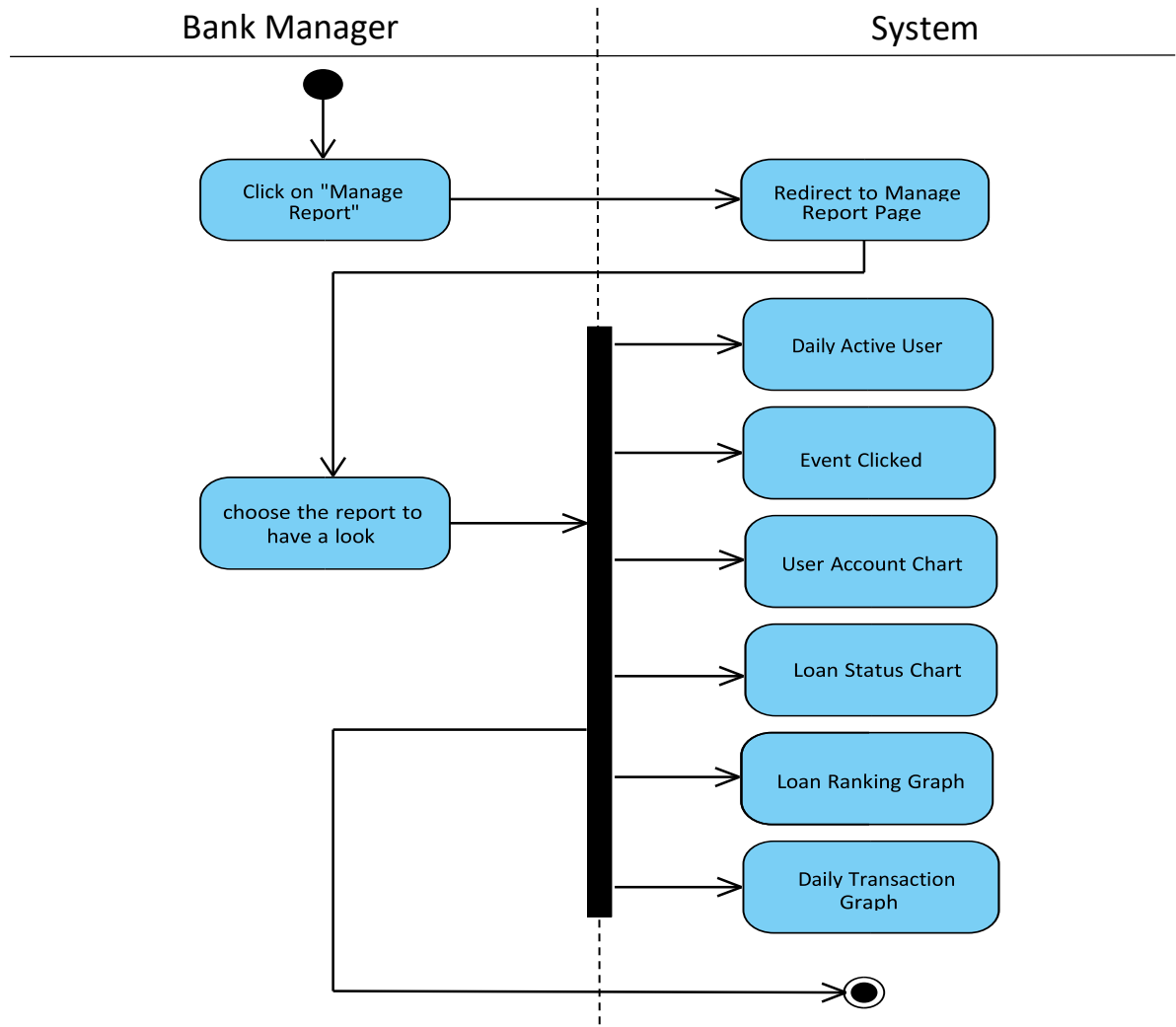
loan status to approved or rejected. If the bank manager rejected the loan application, then he/she must also provide the remark or rejected reason. At the meanwhile, the Firebase will also update the loan status showing in applicant's interface there, so that the applicant can check on the loan status. If the loan application is approved, then the system will be displaying a list of repayments sorted by month. Else if the loan application is rejected, the system will prompt out a dialog box showing the rejected reason.

**3.2.15 Manage Report Module****Use Case Description – Manager Report Module**

Use Case Name: Manage Report	ID: <u>13</u>	Importance Level: <u>High</u>
Primary Actor: Bank Manager, System	Use Case Type: Detail, Essential	
Stakeholders and Interests:		
<ul style="list-style-type: none"> <li>Bank Manager - wants to have a look on Banky statistics reports.</li> </ul>		
Brief Description: This use case describes how we handle the process of letting the bank manager to view the Banky statistics reports.		
Trigger: Bank Manager wants to use manage report to view some specific statistics report.		
Type: External		
Relationships:		
Association: Bank Manager		
Include: view statistic report		
Extend:		
Generalization:		
Normal Flow of Events:		
<ol style="list-style-type: none"> <li>The bank manager clicks on the “Manage Report”.</li> <li>The system will redirect the bank manager to the Manage Report Page.</li> <li>The bank manager can click on either one report showing over there.</li> <li>The report displays statistics in the form of either insight, pie chart or bar graph.</li> <li>Statistics got the data from Firebase, but some are from Google Analytics.</li> </ol>		
When the bank manager clicks on “Daily Active User”		
3.1 The system will redirect the page to the Banky Firebase Overview webpage.		
When the bank manager clicks on “Event Clicked”		
3.2 The system will redirect the page to the Banky Google Analytics webpage.		

<p>When the bank manager clicks on “User Account Chart”</p> <p>3.3 The system will redirect to the User Account Chart Page to display the pie chart of counting the number of open/disable/delete account.</p> <p>When the bank manager clicks on “Loan Status Chart”</p> <p>3.4 The system will redirect to the Loan Status Chart Page to display the pie chart of counting the total loan pending, approval, denial.</p> <p>When the bank manager clicks on “Loan Ranking Graph”</p> <p>3.5 The system will redirect to the Loan Ranking Graph Page to display the bar graph of counting the number of loan application grouped by same loan name.</p> <p>When the bank manager clicks on “Daily Transaction Graph”</p> <p>3.6 The system will redirect to the Daily Transaction Graph Page to display the statistical bar graph of counting daily payments, transactions, and loan applications.</p>
<p>Sub Flows: Not applicable</p>
<p>Alternate / Exceptional Flows: Not applicable.</p>

**Table 3.2.15 Use Case Description – Manage Report Module**

**Activity Diagram – Manage Report Module**

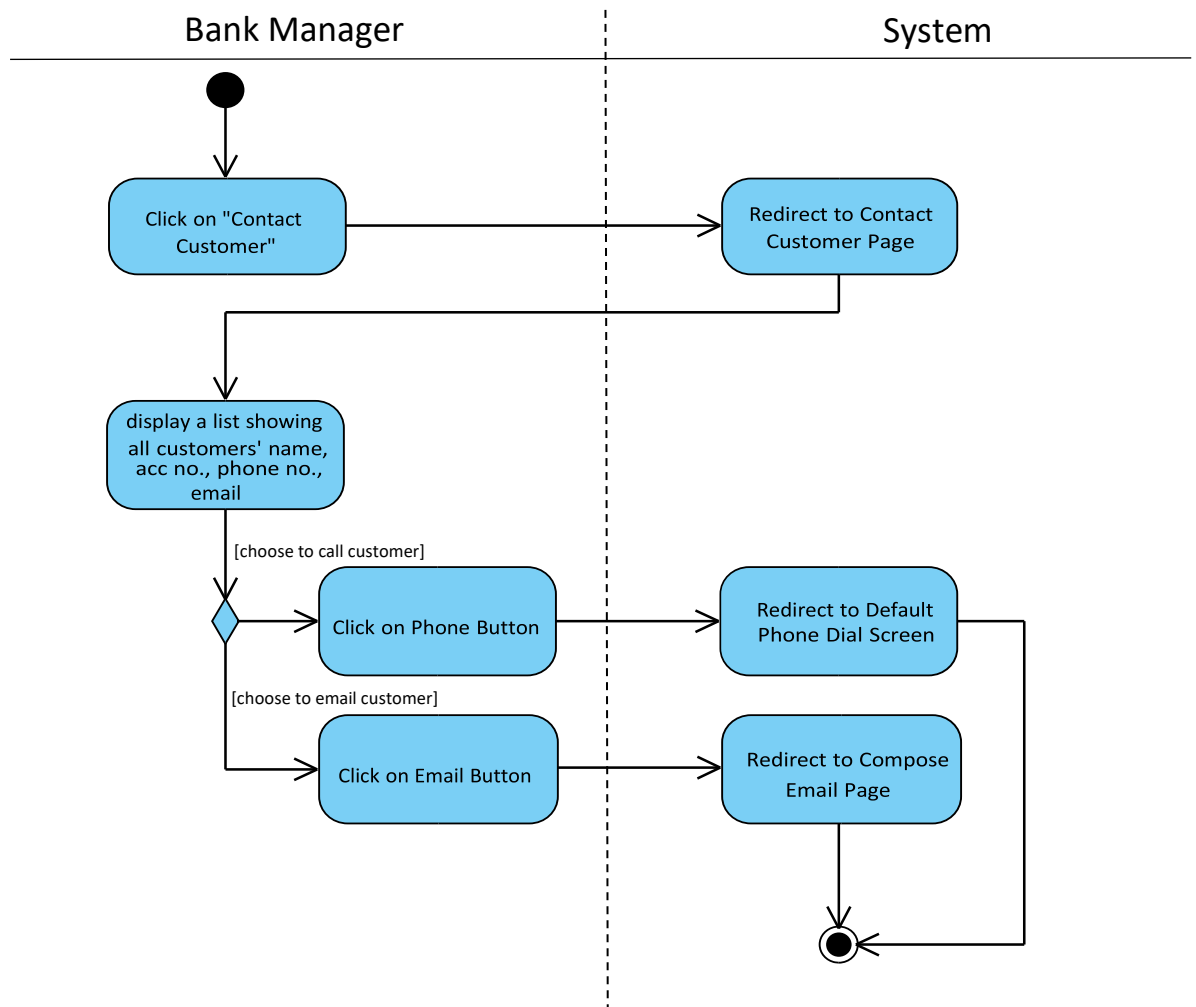
**Figure 3.2.15 Activity Diagram – Manage Report Module**

When the bank manager clicks on “Manage Report”, the system will redirect the bank manager to the Manage Report Page. This allows the bank manager to choose any report to have a look, for example, Daily Active User, Event Clicked, User Account Chart, Loan Status Chart, Loan Ranking Graph and Daily Transaction Graph. If the bank manager clicks on “Daily Active User”, the system will redirect to the Banky Firebase Overview webpage. Else if, the bank manager chooses “Event Clicked”, it will redirect the page to the Banky Google Analytics webpage. User account chart basically will display a pie chart counting the number of open/disable/delete account. Loan status chart will display a pie chart counting the total loan pending, approval, denial. Loan ranking graph will be counting how many numbers of each loan has been applied. Daily transaction graph is showing the total daily payments, transactions, and loan applications.

**3.2.16 Contact Customer Module****Use Case Description – Contact Customer Module**

Use Case Name: Contact Customer	ID: <u>14</u>	Importance Level: <u>High</u>
Primary Actor: Bank Manager, System	Use Case Type: Detail, Essential	
Stakeholders and Interests:		
<ul style="list-style-type: none"> <li>Bank Manager - wants to contact customer either by using phone call or email.</li> </ul>		
Brief Description: This use case describes how we handle the process of letting the bank manager to contact customers.		
Trigger: Bank Manager wants to use contact customer module to contact customers.		
Type: External		
Relationships:		
Association: Bank Manager		
Include: phone call customer, email customer		
Extend:		
Generalization:		
Normal Flow of Events:		
<ol style="list-style-type: none"> <li>The bank manager clicks on the “Contact Customer”.</li> <li>The system will redirect the bank manager to the Contact Customer Page.</li> <li>The system will display all customers’ name, account number, phone number and email address.</li> <li>If the bank manager can click on the phone button on a specific customer, the system will redirect the page to default phone dial screen with the customer’s phone number.</li> <li>If the bank manager clicks on the email button on a specific customer, the system will redirect the page to compose an email and send it to the specific customer.</li> </ol>		
Sub Flows: Not applicable		
Alternate / Exceptional Flows:		
Not applicable.		

**Table 3.2.16 Use Case Description – Contact Customer Module**

**Activity Diagram – Contact Customer Module****Figure 3.2.16 Activity Diagram – Contact Customer Module**

When the bank manager clicks on “Contact Customer”, the system will redirect the bank manager to the Contact Customer Page. It will then display a list row-by-row showing all customers’ name, account number, phone number and email address along with phone button and email button. If the bank manager wants to make a phone call to a specific customer, he/she can click the phone button and the system will redirect the page to the default phone dialing screen and display the customer’s phone number on the dial pad. Else if the bank manager wants to send an email to a specific customer, he/she can click the email button and the system will redirect to compose email page.

### 3.3 System Pre-Authoring Analysis Survey Results

Before starting the system design part, a questionnaire is prepared to understand the basic requirements or expectations from the user's perspective as a bank customer. All results from the questionnaire were filled out by 50 respondents from various demographics and backgrounds.

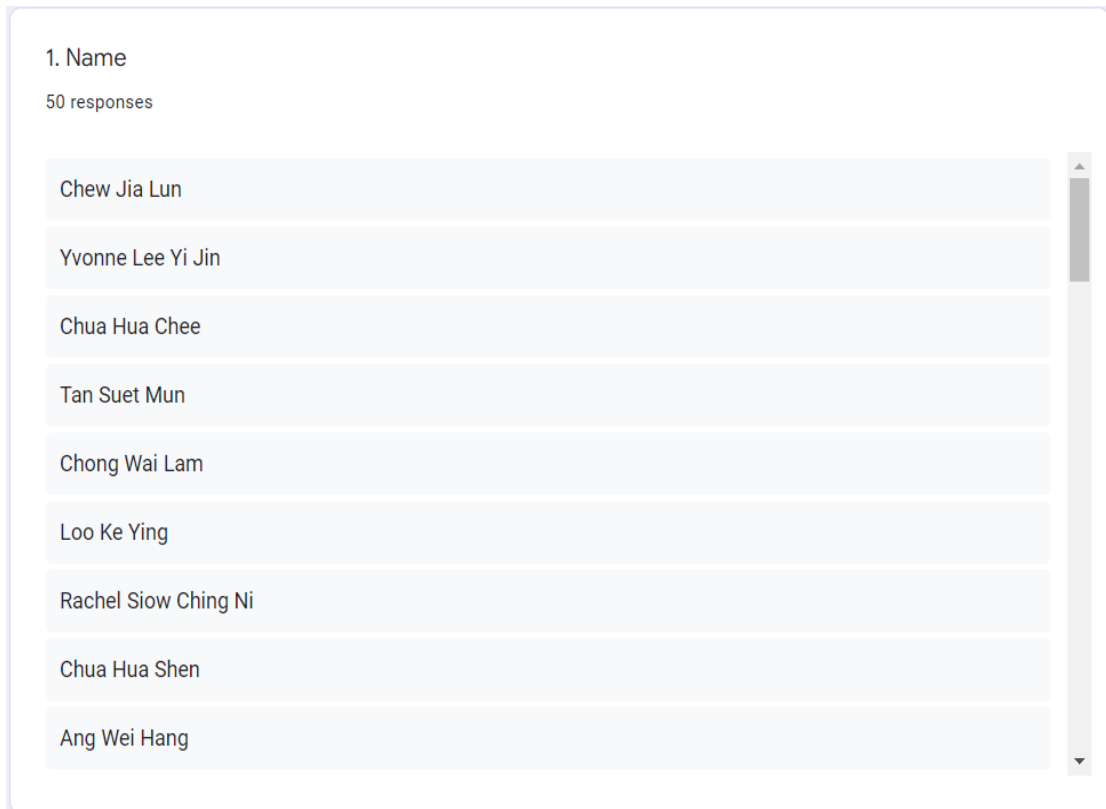


Figure 3.3.1 Demographic 1

2. How old are you?  
50 responses

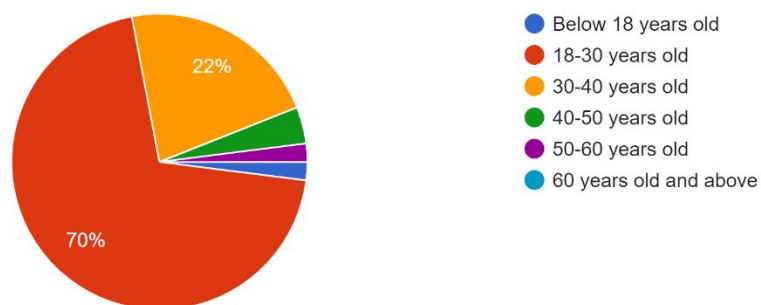
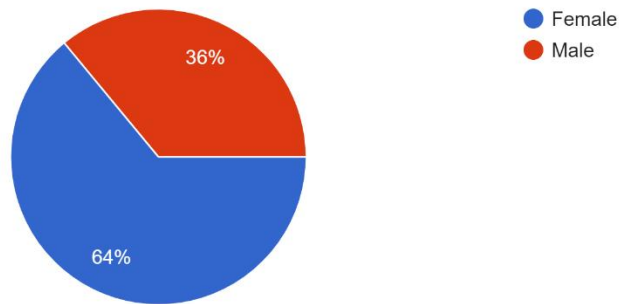


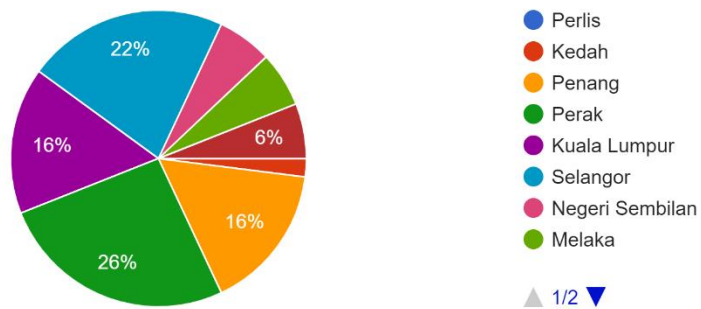
Figure 3.3.2 Demographic 2

3. What is your gender?  
50 responses



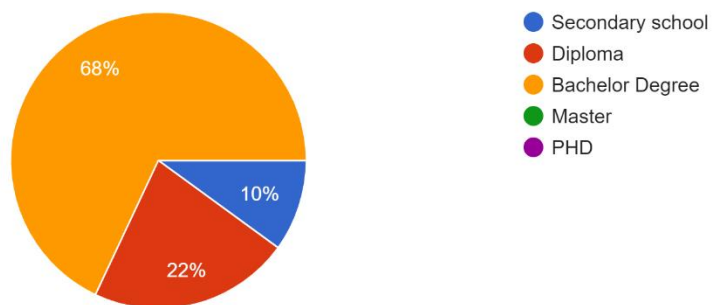
**Figure 3.3.3 Demographic 3**

4. Where are you located?  
50 responses



**Figure 3.3.4 Demographic 4**

5. What is your Highest Education completed?  
50 responses

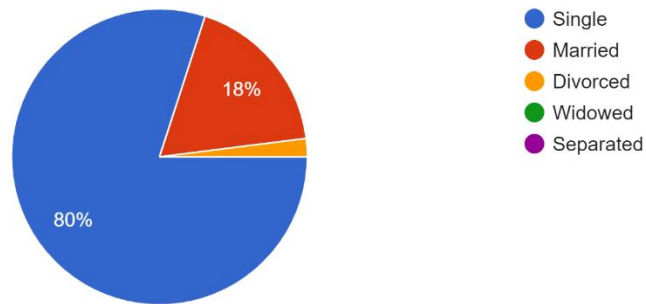


**Figure 3.3.5 Demographic 5**



6. What is your marital status?

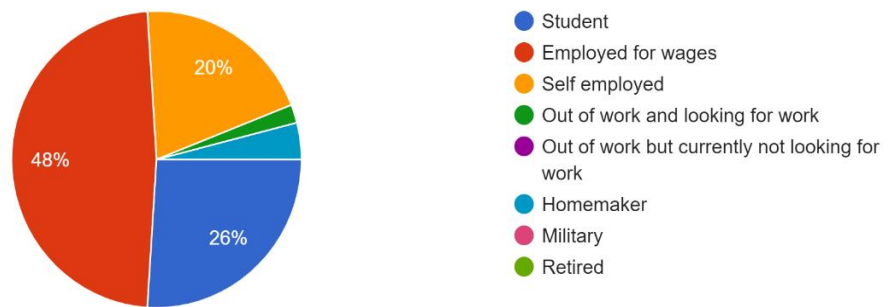
50 responses



**Figure 3.3.6 Demographic 6**

7. What is your occupation?

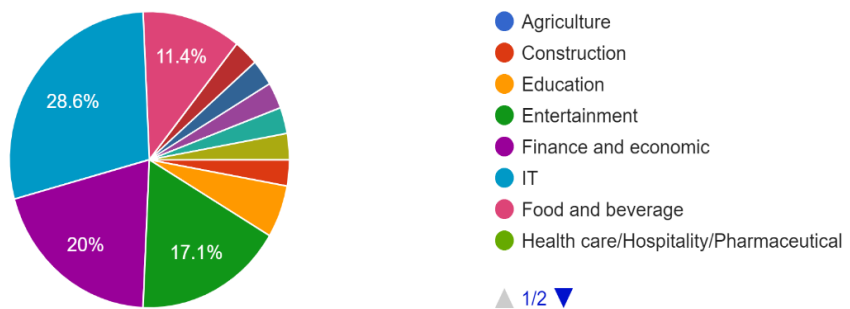
50 responses



**Figure 3.3.7 Demographic 7**

8. Based on the answer to Question 7, what industry sector do/did you work in? [Tips: Student and Homemaker can skip this question]

35 responses



**Figure 3.3.8 Demographic 8**

9. What is your income?  
50 responses

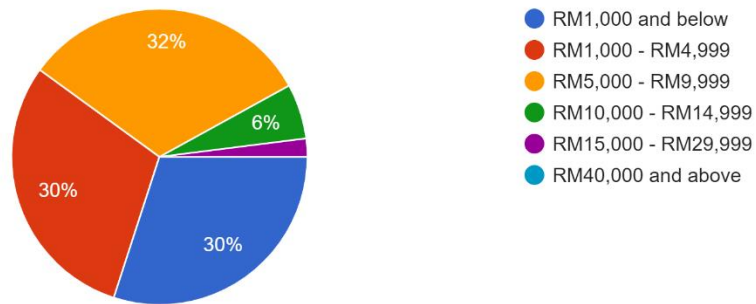


Figure 3.3.9 Demographic 9

10. On a scale of 1-5 (with 1 being 1 out of 5 mark and 5 being 5 out of 5 mark ), please give an overall score for the digital banking services you have visited before based on your satisfaction with speed, experience or usability



Figure 3.3.10 Experience of using another banking service

11. What are your favorite 3 transactions in digital banking? Please select 3 that apply.  
50 responses

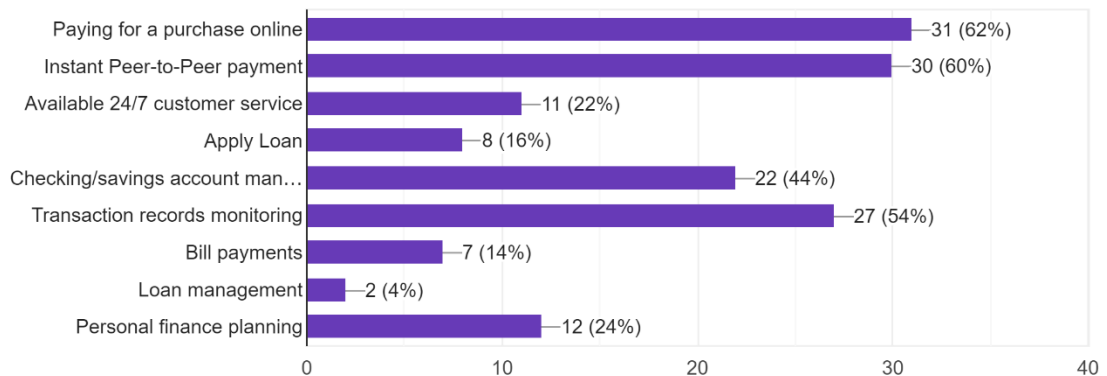
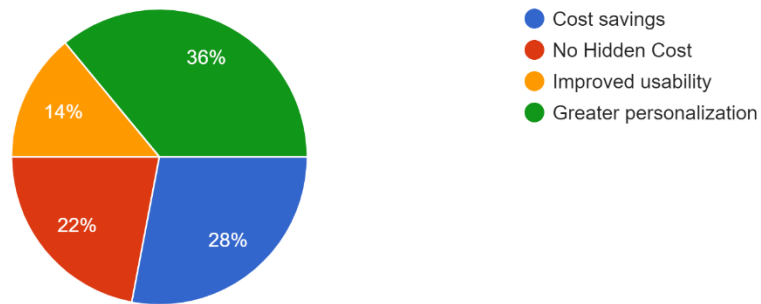


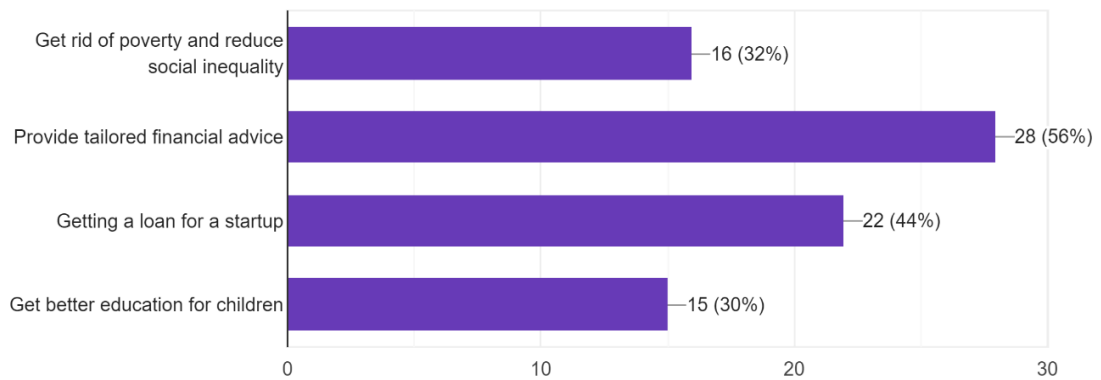
Figure 3.3.11 Perceptions on favorite transactions in digital banking

12. What is the main benefit of digital banking that you think will help you a lot?  
50 responses



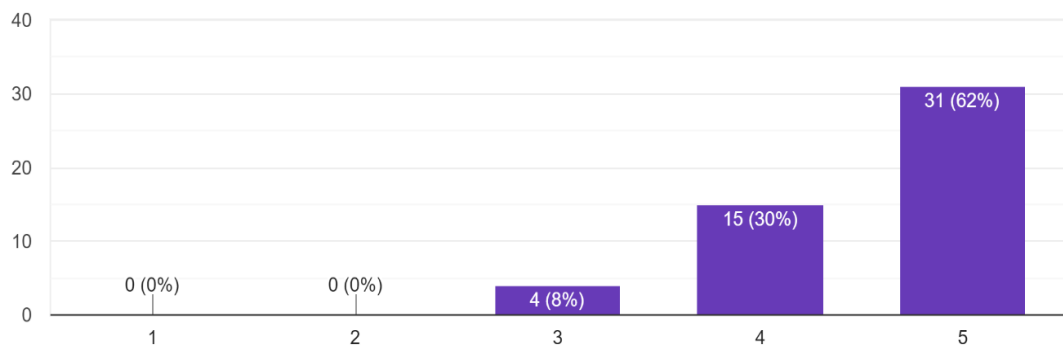
**Figure 3.3.12 Perceptions on the benefit of digital banking**

13. What are the benefits that you think digital banking can give to community?  
50 responses



**Figure 3.3.13 Perceptions on the benefit of digital banking to community**

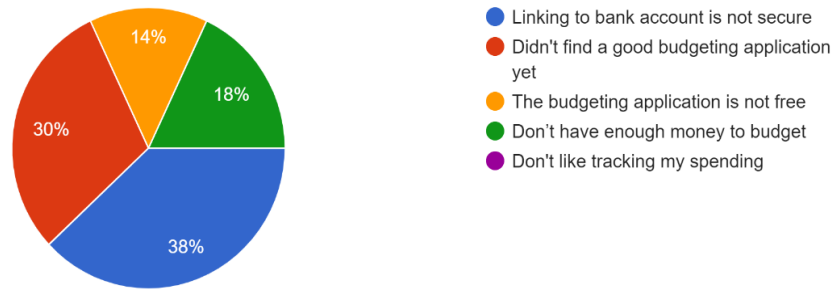
14. On a scale of 1-5, please give a score on how likely are you to try the budgeting feature if it is available in digital banking services?  
50 responses



**Figure 3.3.14 Perceptions on the budgeting feature**

15. What might be preventing you from using a standalone budgeting application?

50 responses



**Figure 3.3.15 Perceptions on the standalone budgeting application**

*Figure 3.3.1* shows that all respondents' names, 70% are in the 18-30 age group, 22% are in the 30-40 age group, 4% are in the 40-50 age group, 2% are in the 50-60 age group and the rest are in below 18 age group (*refer to Figure 3.3.2*). Based on the *Figure 3.3.3*, 64% of respondents are female which occupied 32 out of 50 respondents where only 36% of respondents are male which obtained 18 out of 50 respondents. Most of respondents are from Perak, which occupied 26%, 22% are from Selangor, 16% are from Kuala Lumpur and Penang, 6% are from Negeri Sembilan, Melaka and Johor, only 2% are from Kedah (*refer to Figure 3.3.4*). From *Figure 3.3.5*, majority of respondents, their highest education is until bachelor's degree level, which is 34 out of 50 respondents. 11 out of 50 respondents studied until diploma level, and only 5 of them studied until secondary school level. Based on the *Figure 3.3.6*, 80% of respondents are single status where married and divorced status only occupied 18% and 2% accordingly. As shown in the *Figure 3.3.7*, 24 respondents are employed for wages, 13 respondents are still a student, 10 respondents are self-employed, 2 of them are homemaker and only 1 of them is unemployment and looking for new job. Majority of them work in IT industry, followed by Finance and economic industry, Entertainment industry, Food and beverage industry and other industry (*refer to Figure 3.3.8*). From *Figure 3.3.9*, 32% of them are in RM5000 – RM9999 salary range level, each 30% of them are in the salary range level of RM4999 and below, 6% of them are in RM10000 – RM14999 salary range level and only 2% of them got the salary in the range of RM15000 – RM29999.

Based on the *Figure 3.3.10*, the results showed the satisfaction of different banking services provided by another bank that respondents have visited before. 24 out of 50 respondents give 5 marks for the satisfaction of quick access to digital banking services, followed by 21 out of 50 respondents give 4 marks and 5 out of 50 respondents give 3 marks. For the satisfaction of 24/7 customer service provided by another bank, 12 Bachelor of Information Systems (Honours) Business Information Systems  
Faculty of Information and Communication Technology (Kampar Campus), UTAR

respondents give 5 marks, 22 respondents give 4 marks, 14 respondents give 3 marks and only 2 respondents give 2 marks. For the satisfaction of fund transfer provided another bank, 32 out of 50 respondents give 5 marks, 15 out of 50 respondents give 4 marks and followed by 3 respondents give 3 marks. For the satisfaction of loan management of another bank, 8 of them give 5 marks, 14 of them give 4 marks, 21 of them give 3 marks and only 7 of them give 2 marks. The experience in using digital banking services such as pay bills, based on the result and it found that 25 respondents give 5 marks, 19 of them give 4 marks and 6 out of 50 respondents give 3 marks. The banking service, for example, QR function to scan and receive fund, it can see those 31 respondents like this function, so they give 5 marks, 16 respondents give 4 marks, and only 3 respondents give 3 marks. Based on the *Figure 3.3.11*, the top 3 favorite transactions in digital banking are paying for a purchase online which obtained 62%, followed by instant peer-to-peer payment which occupied 60%, and transaction records monitoring which held 54%. Furthermore, 36% of respondents think that greater personalization is the main benefit that the digital banking can help them a lot, 28% of respondents think cost savings is the main benefit, 22% of them think no hidden cost, and 14% think improved usability (*refer to Figure 3.3.12*). Based on results shown in *Figure 3.3.13*, the top benefit that respondents think the digital banking can give to community is to provide tailored financial advice (56%), the second top benefit is to provide a loan for a startup (44%), the third top benefit is to get rid of poverty and reduce social inequality (32%), and to let children have better education is the least benefit that respondents think (30%). There are 62% respondents who are extremely interested in trying a budgeting feature built-in together with other digital banking services, 30% are just interested, and 8% of them do not care too much and think like if it is built-in together with other services then can give it a try, but still do not think it is a problem if the budgeting feature does not come with other banking services (*refer to Figure 3.3.14*). From results showing in *Figure 3.3.15*, 38% of respondents think that link to bank account is not secure as the reason preventing them from using a standalone budget application, 30% of respondents think that they did not find a good budget application yet as the reason preventing them from using a standalone budget application, 18% think that they do not have enough money to budget as the reason they do not want to use the budget application, and 14% think that since the budget application is not free as the reason prevents them using budget application.

## CHAPTER 4 SYSTEM DESIGN

### 4.1 System Design Diagram

#### 4.1.1 System Flowchart

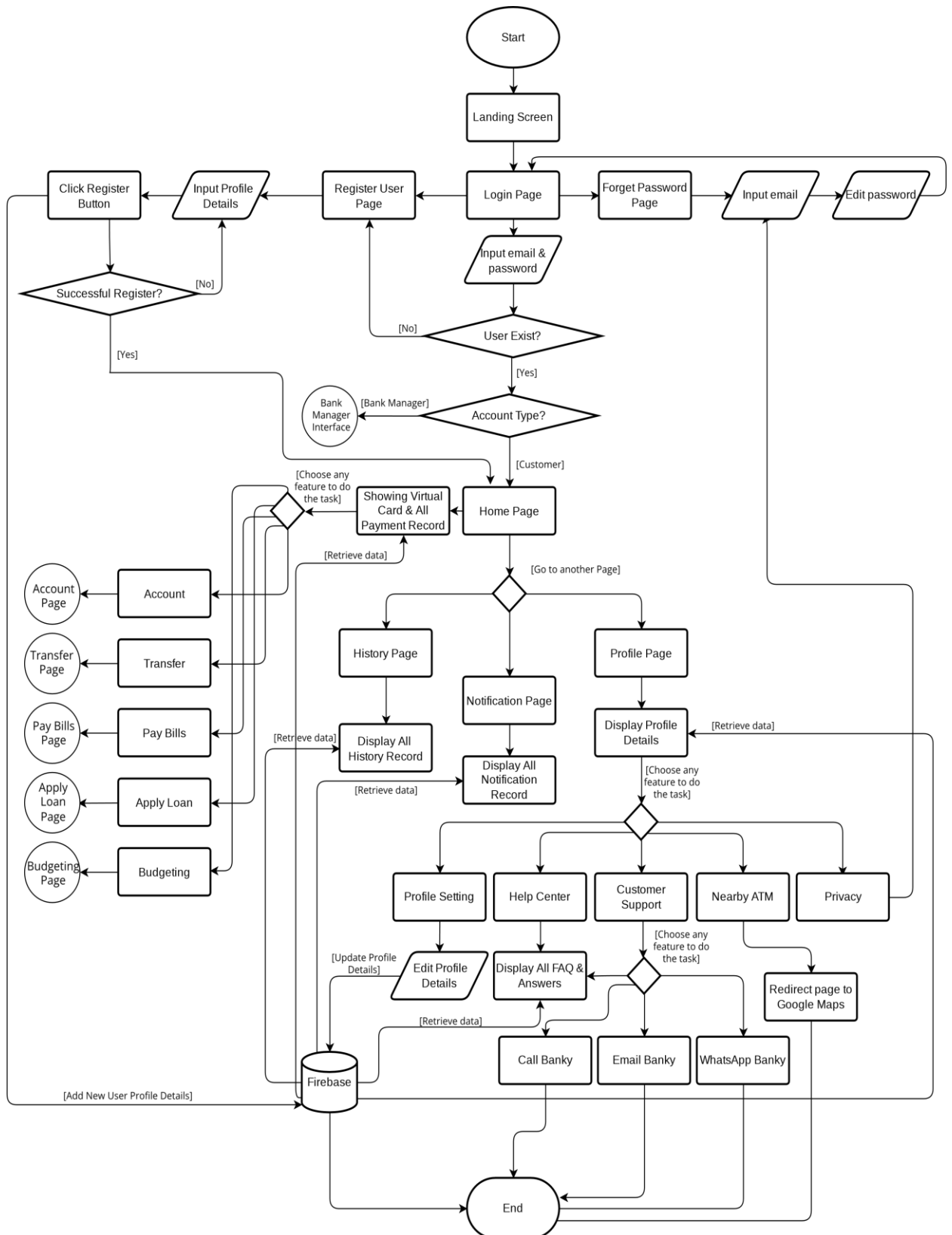
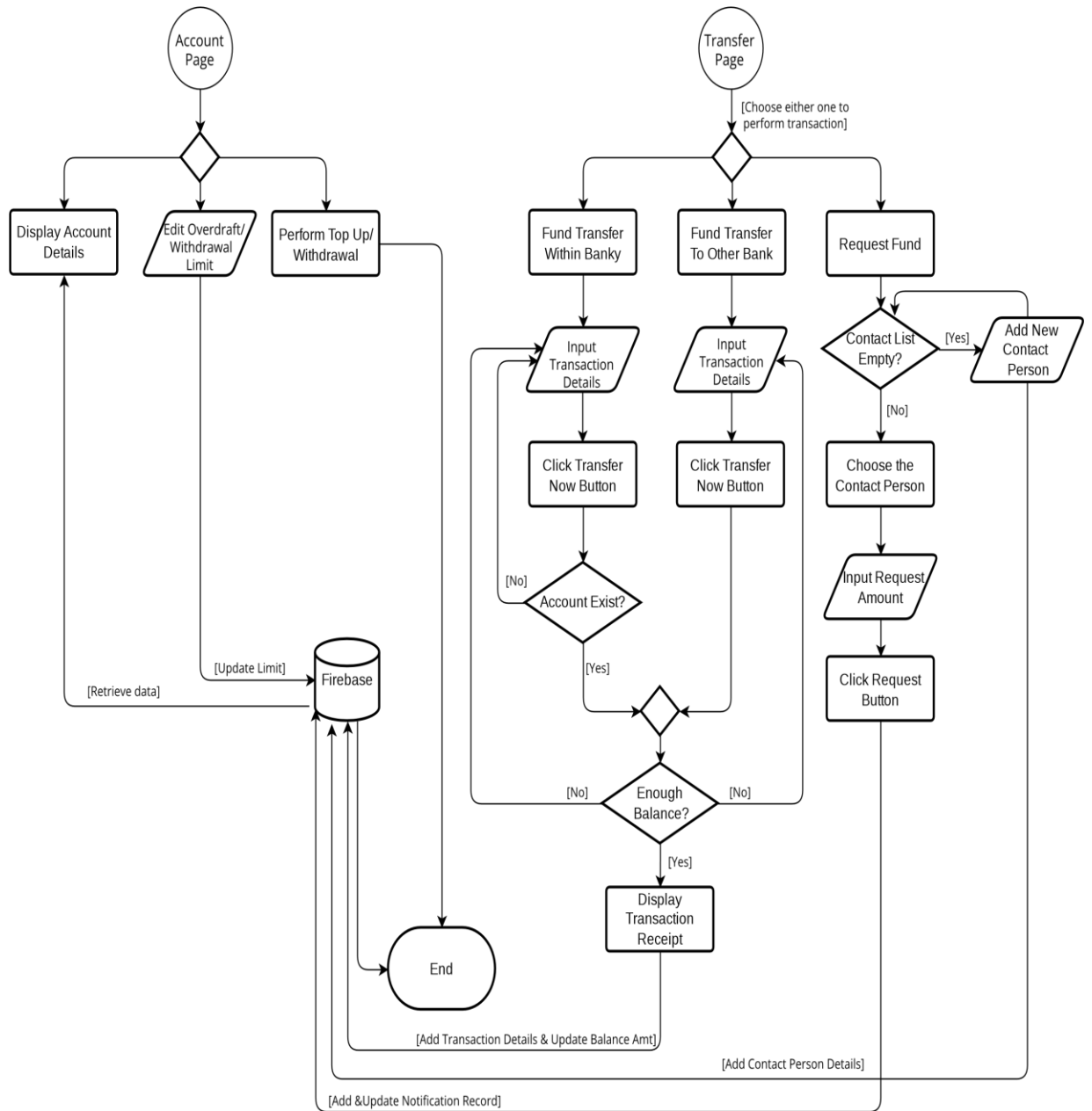


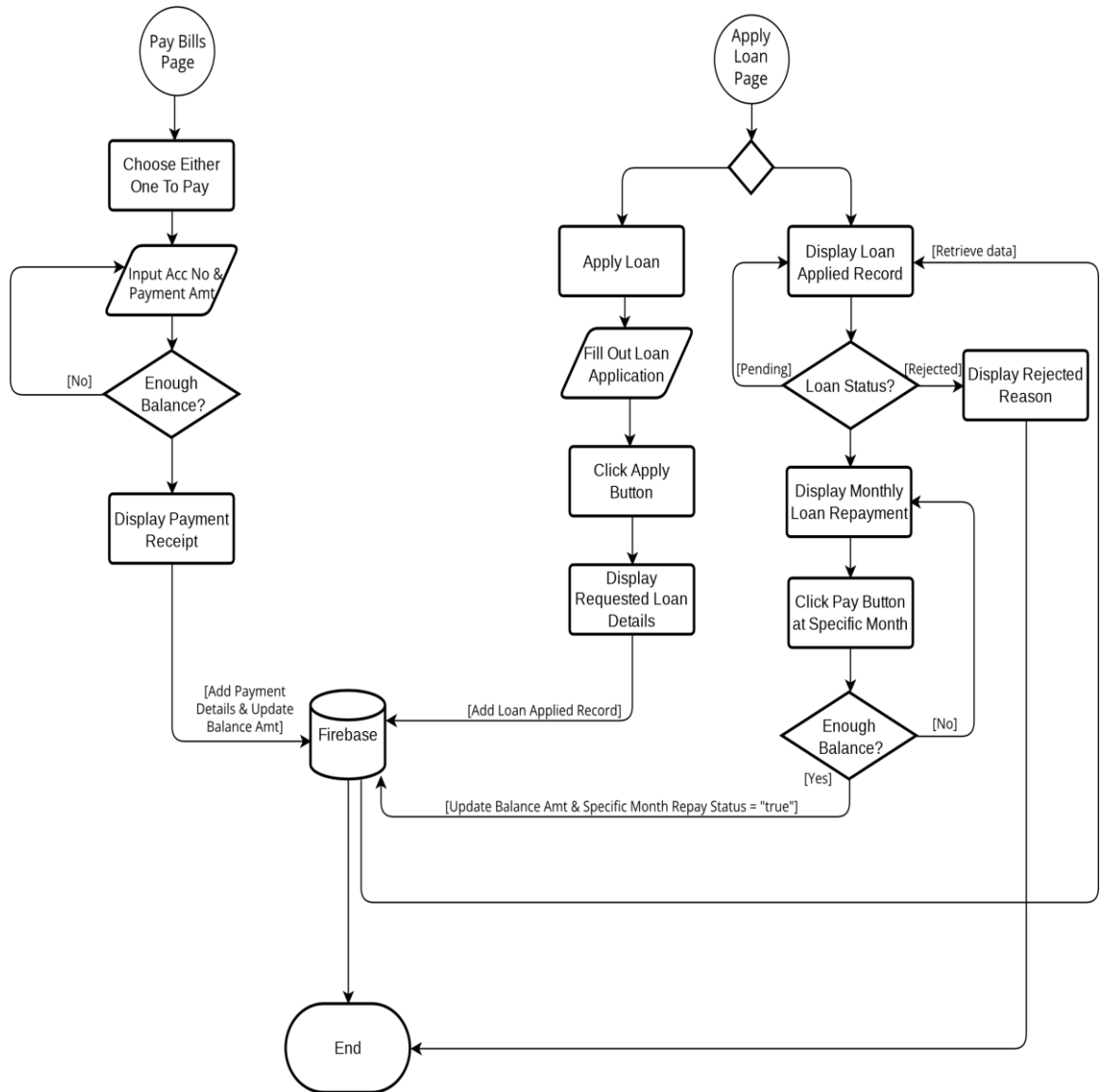
Figure 4.1.1.1 System Flowchart

**System Flowchart (Cont)**



**Figure 4.1.1.2 System Flowchart (cont)**

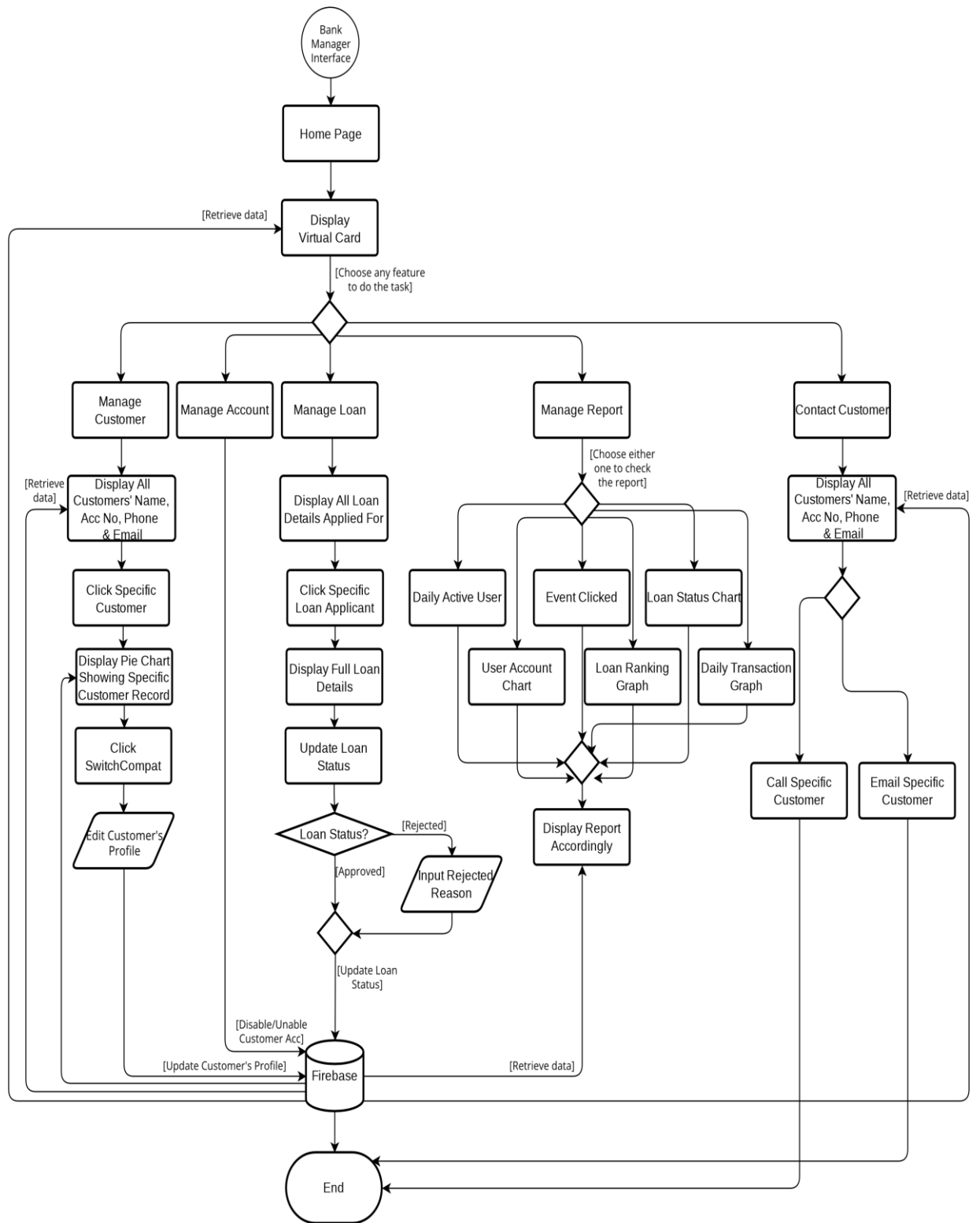
**System Flowchart (Cont)**



**Figure 4.1.1.3 System Flowchart (cont)**



**System Flowchart (Cont)**



**Figure 4.1.1.4 System Flowchart (cont)**

4.1.2 System Block Diagram

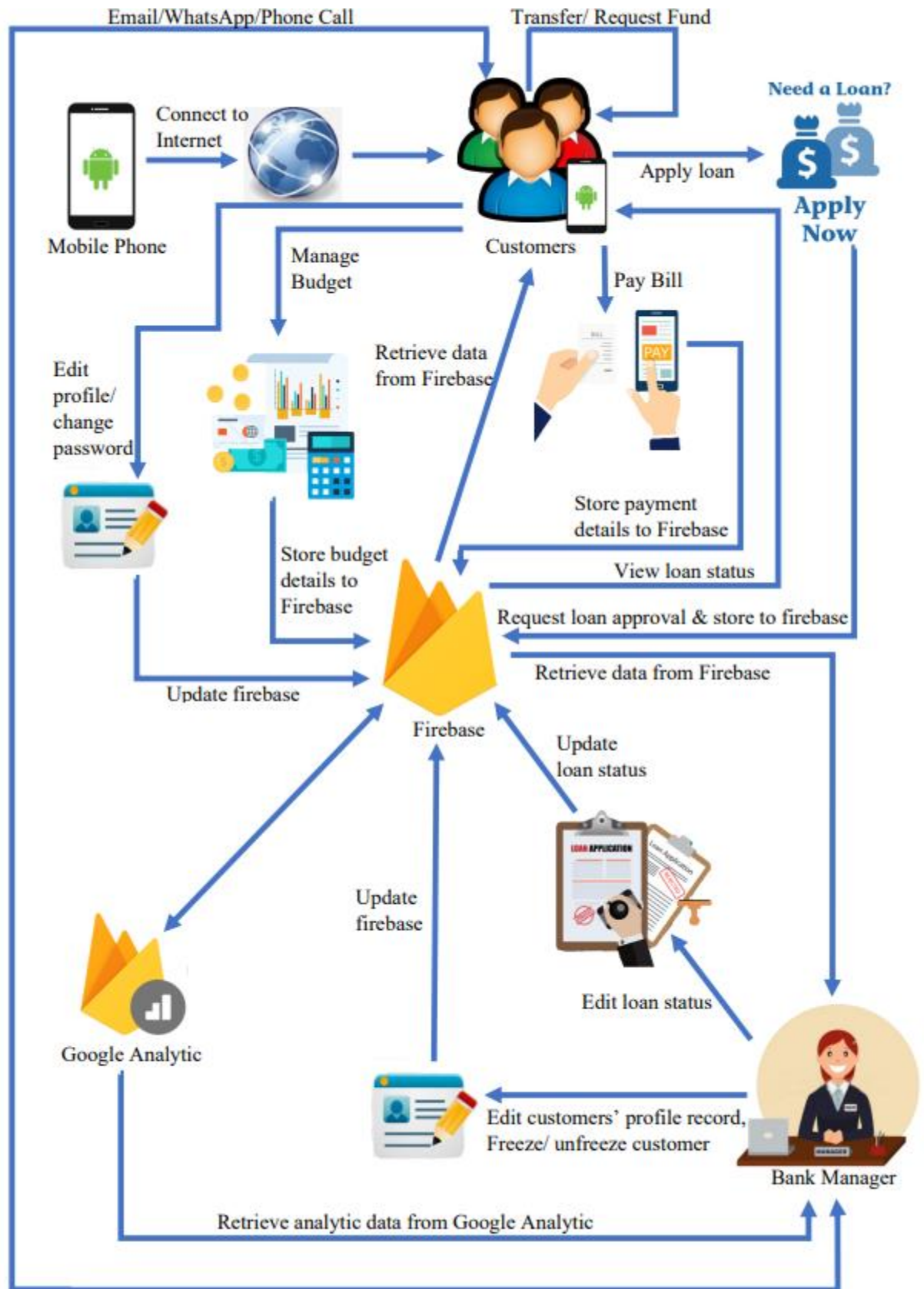


Figure 4.1.2 System Block Diagram

## 4.2 System Flow Description

First of all, the user has to use his/her Android phone to connect to the Internet, then only can access Banky mobile application. Before logging into Banky, the user must have a user UID existed in the Firebase Authentication or another word we called that the user has been registered in our system.

### 1. RegisterUserActivity class

If the user is a new user, he/she can click “Not have account? Register” and must fill in all fields of profile details such as name, identification number, phone number, complete home address, email and password, while uploading a profile picture is optional. The `checkLocationPermission()`, `checkCameraPermission()`, `checkStoragePermission()` methods will be called to check all these permissions, and if any of them are found to be disallowed, the user will be asked for permission using the `requestLocationPermission()`, `requestCameraPermission()` and `requestStoragePermission()` methods. If the user chooses to upload a profile image, they can use the camera to take a new photo to upload, or use the `pickFromCamera()` or `pickFromGallery()` methods to upload a photo from gallery storage, while auto-detecting the GPS icon will use the `detectLocation()` and `findAddress()` methods to detect the user's latitude and longitude to find the address. The `inputData()` method will check to make sure none of the fields are not empty, then it will use the method called `createAccount()` to handle creating the account. If the creation is successful, then it only will save and upload the user's profile details to Firebase Authentication and Realtime Database, which the `saverFirebaseData()` method will handle it, but the profile image will be stored to Firebase Storage. Each user's account number is automatically generated by the system using the formula “2400000000000000L + (Math.random() \* 100000000000000L)”, this way each user will have an account number starting with 24 and the remaining 14 digits will be random. If the user is an employee of Banky, then clicking "Are you a bank manager or staff? Register here" will redirect the employee to the registration management page handled by the supported class `RegisterAdminActivity`.

## **2. LoginActivity class**

If the user has registered before, he/she can enter the email address and password to log in to Banky. But before that, the `loginUser()` method will check if the pattern of the entered email address is valid, and if the password is empty. If the email and password match with the Firebase Authentication, it will use the method `makeMeOnline()` to bring the user online. At the meanwhile, the `checkUserType()` method will participate in checking the `accountType` in Firebase, so if the `accountType` is Admin, the user will be redirected to the `MainAdminActivity` class, otherwise to `MainUserActivity` class. If it is assumed that the user has forgotten the password, he/she can click “Forgot Password” below the text field and the system will redirect the user to the `ForgotPasswordActivity` class.

### **2a. ForgotPassowrdActivity class**

The `recoverPassword()` will handle the process of checking whether the pattern of the entered email address is valid. If it is valid email pattern, the Firebase backend will send a password reset email to the user, so the user can go to email there to update the new password.

## **3. MainUserActivity class**

In this class, it contains the bottom navigation bar or what we call a fragment, so the system will redirect to that fragment based on the selected navigation item. For example, `HomeUserActivity()` fragment when the user clicks on the home icon displayed in the bottom navigation bar, `HistoryUserActivity()` fragment when the user clicks on the history icon displayed in the bottom navigation bar, `NotificationUserActivity()` fragment when the user clicks the notification icon displayed in the bottom navigation bar, and `ProfileUserActivity()` fragment when the user clicks the person icon displayed in the bottom navigation bar. The user will be redirected to the `QRScanUserActivity` class if he/she clicks on the bottom center with the Scan icon.

### **3a. QRScanUserActivity class**

It contains 3 tabs such as Scan, Receive or Pay for the user to choose from. If the Scan tab is clicked, it will hide the Receive and Pay contents using `showScanUI()` method, and it will check if camera and storage permission is allowed using `checkCameraPermission()` and `checkStoragePermission()` methods. Otherwise, it will

use `requestCameraPermission()` and `requestStoragePermission()` methods to ask the user for permission, so that it can only scan the QR code with the camera or scan the QR code uploaded from gallery storage. Else if, the Receive tab is clicked, it will hide the Scan and Pay contents using `showReceiveUI()` method and also using the `BarcodeEncoder()` to generate a QR code. Else if, the Pay tab is clicked, it will hide the Scan and Receive contents using `showPayUI()` method and also using the `BarcodeEncoder()` to generate a QR code and a bar code.

### **3b. HomeUserActivity() fragment**

In this fragment, the `loadMyInfo()` method loads and displays the username, available balance and account number retrieved from Firebase Realtime Database. It will also display all payment history using the `loadPaymentItems()` method. The `loadPaymentItems()` method uses `ModelPayment` and `AdapterPaymentUser` as a way to retrieve data and display it in a list. The Model is used to retrieve the data by matching variable names in Firebase Realtime Database whereas the Adapter is used to convert data items into a list of views, such as `RecyclerView` in this case. The 5 functions of Account, Transfer, Pay Bills, Apply Loan and Budgeting are shown in this fragment. If the user clicks the Account icon, the user is redirected to the `AccountUserActivity` class. Else if, the user clicks the Transfer icon, the user will be redirected to the `TransferUserActivity` class. Else if, the user clicks the Pay Bills icon, the user will be redirected to the `PayBillUserActivity` class. Else if, the user clicks the Apply Loan icon, the user will be redirected to the `LoanUserActivity` class. Otherwise, the user will be redirected to the `BudgetUserActivity` class.

#### **3b-1. AccountUserActivity class**

The system will be using the `loanMyInfo()` method to display the user's account number, available balance, current balance, fixed deposit balance, default overdraft limit and default withdrawal limit retrieved from Firebase Realtime Database. If the user wishes to update the overdraft limit, he/she can simply adjust the seek bar and it will use `updateOverdraftLimit()` method to handle the process of updating overdraft limit in Firebase Realtime Database. Otherwise, if the user wants to update the withdrawal limit, he/she can direct adjust the seek bar and it will use `updateWithdrawalLimit()` method to handle the process of updating withdrawal limit in Firebase Realtime Database.

**3b-2. TransferUserActivity class**

In this page, the user can choose to transfer money within Banky using the supported class `TransferWithinBankActivity`, or transfer money to other banks using the supported class `TransferToOtherBankActivity`, or request funds with using the supported class `RequestFundUserActivity`.

**3b-2i. TransferWithinBankActivity class**

The user does not need to enter his/her account number because the `loadMyInfo()` method has retrieved the account number, name, available balance from Firebase Realtime Database. However, the user will have to enter the recipient's account number, transfer amount, recipient reference, payment details and select the transfer date from calendar using the `datePickDialog()` method, where the `inputData()` method will check that all fields except the payment details are not empty as the text field of payment details is optional. When the user clicks the "Transfer Now" button, the `loadOtherInfo()` method will handle checking whether the recipient's account number exists. If the recipient exists in the Firebase Realtime Database, then use `addTransactionToDb()` and `addTransactionToBeneficiary()` methods to add the transaction details to Firebase Realtime Database under the Transaction child of both users. But before adding a transaction to Firebase, the system will compare the available balance with the transfer amount, and only updates both sender and recipient Firebase if the available balance is greater than or equal to the transfer amount and the transfer amount is not zero, otherwise the system will toast an error message. At the same time, `addNotificationToBeneficiary()` method will also update the recipient's child of the Notification in Firebase Realtime Database. Since the transaction has already been completed, the system of course needs to update the available balances of the sender and receiver using the `updateBalance()` and `updateTransferToBalance()` methods. The `updateBalance()` method is used on the sender side as it will deduct the transfer amount from the available balance, and `updateTransferToBalance()` method is used on the recipient side as it needs to increase the available balance based on the amount received. After a successful transfer, a transaction receipt is displayed using the `loadTransactionInfo()` method from the supported class `TransferToOtherDetails` to retrieve data from Firebase Realtime Database.

**3b-2ii. TransferToOtherBankActivity class**

Actually, for this class, it is not much different from TransferWithinBankActivity class as it also uses loadMyInfo(), inputData(), addTransactionToDb() and updateBalance() methods. One of the differences is that the user is required to select the other bank name from the spinner, and it will check if the account number is valid, and the beneficiary's name is exactly the same from the Firebase Realtime Database under the "zOtherBankUsers" by using the loadOtherBankUserInfo() after the user clicks the "Transfer Now" button. If both conditions are correct, it then will use the addTransactionToDb() method to compare the available balance and the transfer amount, and update the user own available balance and the beneficiary's available balance using updateBalance() and updateOtherBankUserBalance() methods only if the available balance is greater or equal to transfer amount. A transaction receipt is also displayed using the loadTransactionDetails() method from the supported class TransferToOtherDetails which shares the same class as TransferWithinBankActivity if it is transferred successfully.

### **3b-3. RequestFundUserActivity class**

The user can click on any of the contact persons to request funds from them, which this action is handled by RequestFundDetailsUserActivity class. But before selecting a contact, it uses the loadContactListItems() method to load contact list data retrieved using ModelContactList and AdapterContactListUser to display it in a list. If there is no contact list record, the page will prompt "You do not have any contact list at this moment. Please create a new contact.", so the user can click the add icon floating button to add a new contact which the AddContactListUserActivity class will handle this operation.

### **3b-3i. AddContactListUserActivity class**

The user needs to enter the Banky account number, name, phone and email address of the contact, while uploading the picture of the contact is optional. After the inputData() method checks to make sure none of the text fields are empty, it only processes calling addContactToDb() method to handle the action of adding a new contact to the Firebase Realtime Database.

### **3b-3ii. RequestFundDetailsUserActivity class**

It will use the loadContactDetails() method to display the selected contact's name and profile image and ask the user to enter the requested fund amount. Whenever the user clicks the "Request Now" button, the inputData() method handles to make sure the

requested fund amount is not empty, then it only will continue checking if the contact account number exists in Banky Firebase, and if so, just adds it using the `addNotificationToSenderDb()` method add the request fund data to the notification record for this contact.

### **3b-4. PayBillUserActivity class**

This page only shows icons and names of miscellaneous items like ASTRO, unifi, Indah Water, Air Perak, Air Selangor, TNB and PTPTN, so the user can click on the one he/she wants to make a payment and the system redirects the user to which is handled by the `PayBillDetailsUserActivity` class on the payment bill details page.

#### **3b-4i. PayBillDetailsUser Activity class**

The user needs to enter the payment details such as the account number of the payment company and the payment amount. When the user clicks the "Pay Now" button, the `inputData()` method handles checking if the text field is empty, then it continues to compare if the available balance is greater than or equal to the payment amount, only it adds the payment details to Firebase Realtime Database using the `addPaymentToDb()` method, otherwise the user will get an error message that the funds are insufficient, or the payment amount is zero. After successfully making the payment, a payment receipt is displayed using the `loadPaymentDetails()` method from the supported class `TransferToOtherDetails` to retrieve data from Firebase Realtime Database.

### **3b-5. LoanUserActivity class**

In this class, the user can click "Apply Loan" and it will redirect the user to the loan application page, which is handled by the `ApplyLoanUserActivity` class. It uses the `loadLoanItems()` method to load past loan applications retrieved using `ModelLoan` and `AdapterLoanUser` to display them in a list. If there is no past loan application or the user has not applied for any loan, the poster will show the process of applying for a loan. Otherwise, the user can see the loan status of each loan application previously applied for, and if the loan status is approved or rejected, a "Details" button is displayed so that the user can view a list of monthly repayments handled by the `LoanRepayDetailsUserActivity` class or view the reason for the loan rejection using a dialog box.

#### **3b-5i. ApplyLoanUserActivity class**



The `inputData()` method will be processed to ensure that the user has selected the loan type from the spinner, entered the loan amount to borrow, the loan term (in months) and selected the loan start date from the calendar, which forced the start date to be a date two weeks from today onwards using the `datePickDialog()` method. It also forces the user to turn on the repayment plan switch, and calls the `loadRepaymentPlan()` method to calculate and display the final monthly repayment amount, the total interest charged, and the total cost to be paid. The `addLoanToDb()` and `addLoanToBankManager()` methods are used to handle adding loan application details to the Firebase Realtime Database when the user clicks "Promptly Apply". The reason for also adding the loan application details under the child of "Loans" of the bank manager in Firebase, so that later the bank manager can edit or update the loan status to Pending, Approved, or Rejected. Meanwhile, it will use `addLoanRepaymentToDb()` to throw all monthly repayments for each month into the Firebase Realtime Database, and only those loan applications with approved loan status will see loan repayment details. A requested loan application is displayed using the `loadLoanDetails()` method in the supporting class `ApplyLoanDetailsUserActivity` to retrieve data from Firebase Realtime Database.

### **3b-5ii. LoanRepayDetailsUserActivity class**

This class can be accessed only with those loan applications with approved status. It will show the month needed to repay the loan along with the repayment amount using `loadLoanRepaymentItems()` method to load monthly repayment data retrieved using `ModelLoanRepayment` and `AdapterLoanRepaymentUser` to display it in a list. If the user clicks on the "Pay" button in one of the monthly repayment lists, `updateTransactionDateTime (modelLoanRepayment, holder)` method in the `AdapterLoanRepaymentUser` will be checking if the available balance enough to cover repayment amount, then update the available balance using the `updateBalance(modelLoanRepayment, holder)` method, else it will toast an error message showing insufficient funds.

### **3b-6. BudgetUserActivity class**

In this budget page, the user can choose to set a weekly budgeting with the supported class `SetBudgetUserActivity`, or add today spending using the supported class `SpendingTodayUserActivity`, or track spending for this week with the supported class

SpendingWeekUserActivity, or manage the budget by using the supported class BudgetAnalyticUserActivity or view all the spending history using SpendingWeekUserActivity that shared class with the track spending for this week.

### **3b-6i. SetBudgetUserActivity class**

The user can click the add icon floating button to add the budget amount for each category by using addBudgetItem() method to prompt a dialog to select category from the spinner and type the budget amount, it will add the budget item to Firebase Realtime Database using addBudgetItemToDb() method only when the user clicks the save button. It calculates and updates the new total budget amount using the updateTotalBudget() method, and uses loadMyInfo() method to display it out. When there are any budget items, it will use loadBudgetItems() method to load all budget items retrieved using ModelBudget and AdapterBudgetUser to display them in a list.

### **3b-6ii. SpendingTodayUserActivity class**

It uses the loadSpendItems() method to load all today's spending items retrieved using ModelBudget and AdapterTodaySpendingUser to display them in a list. The total today spend amount will be calculated when the loadSpendItems() method is called. If the user does not have any spending item today or wants to add a new spending item, he/she can click on the add icon floating button, which prompts a dialog to select an item category, and enter spending amount and note using the addTodaySpendingItem() method, which it will only add it to Firebase Realtime Database when the user clicks the save button. The total today spending amount will be calculated automatically when the time the loadSpendItems() method is called.

### **3b-6iii. SpendingWeekUserActivity class**

This class is used loadWeekSpendItems() method to load all spending items within this week retrieved using ModelBudget and AdapterWeekSpendingUser to display them in a list. But, if the user is clicking the "View All Spending History" in Budget Page to come to this class, then loadAllSpendItems() method is called to load all spendings items and display them in a list using ModelBudget and AdapterWeekSpendingUser as well.

### **3b-6iv. BudgetAnalyticUserActivity class**

In this class, the loadMyInfo() method is called to retrieve the budget amount for all category from Firebase Realtime Database that were set up earlier. After that, it uses loadTotalWeekApparel(), loadTotalWeekEducation(), loadTotalWeekEntertainment(),

loadTotalWeekFood(), loadTotalWeekHealth(), loadTotalWeekHouse(), loadTotalWeekPersonal(), loadTotalWeekTransport(), loadTotalWeekOther(), loadTotalWeekSpending() methods are called to load and automatically calculate the total spend amount for each category in this week, and calculate the spend percentage versus budget amount. So, if the percentage is less than 50%, it will display a green color status, indicating that it is still healthy. Otherwise, if the percentage is greater than or equal to 50% but less than or equal to 100%, a yellow color status is displayed, indicating a warning. Else, if the percentage is greater than 100%, it will display a red color status to indicate that the user has exceeded the budget for that category. The total spend percentage status is also same as the spending percentage for each category. After that, if none of the spend amount is empty, then it will call setupPieChart to display spent items in pie chart.

### **3c. HistoryUserActivity fragment**

In this fragment, the loadTransactionItems() method loads and displays the transaction date, transaction ID, sender/recipient name, transfer details, payment details and amount received or sent retrieved from Firebase Realtime Database. The loadTransactionItems() method uses ModelTransaction and AdapterTransactionUser as a way to retrieve data and display it in a list.

### **3d. NotificationUserActivity fragment**

In this fragment, the loadNotificationItems() method loads and displays all the records of receiving funds from another Banky user and all the records of requesting funds from another Banky user retrieved from Firebase Realtime Database. The loadNotificationItems() method uses ModelNotification and AdapterNotificationUser as methods to retrieve data and display it in a list. If it is the request funds notification record, it will show the “Pay” button if the user has not paid back, and it will hide the “Pay” button if the user has already paid. When the user clicks the “Pay” button, the user is redirected to Fund Transfer Within Banky page which is handled by the supported class TransferWithinBankActivity. Instead of filling out all the text fields as before in the TransferWithinBankActivity class, in this case the user just needs to select the transfer date, and enter the recipient reference and payment details.

### 3e. ProfileUserActivity fragment

In this fragment, it will display the user's profile image, name and email using the `loadMyInfo()` method to retrieve the data from Firebase Realtime Database. There are 5 functions in total of this fragment, such as Profile Settings, Privacy, Help Center, Customer Support and Nearby Atm. If the user wishes to update or modify his/her profile, he/she can click on "Privacy Setting" and the user will be redirected to profile settings page handled by the `ProfileSettingUser` class. If the user wants to change the account password, he/she can click "Privacy", and the system will redirect the user to the forgot password page processed by the `ForgotPasswordActivity` class. If the user wants to see more frequently asked questions, he/she can click on "Help Center" and the system will redirect the user to the help center page handled by the `FAQUserActivity` class. If the user finds that the FAQ doesn't answer his/her question, he/she can look to Banky's customer support which is handled by the `CSUserActivity` class. Or if the user wants to view nearby ATM locations, he/she can click on "Nearby Atm", the user will redirect to open Google Maps using the Google Maps Intents and display all the nearby ATM locations using the user's latitude and longitude.

#### 3e-1. ProfileSettingUser class

The `loadMyInfo()` method will retrieve and display the user's profile image, name, identification number, phone number, complete address along with city, state and country, and email address from Firebase Realtime Database, so that the user can check if the profile details are present correct after the user successfully registered to Banky. If suppose the user has changed a new phone number or maybe moved to a new house and thus may need to update a new home address or any other profile details, he/she can click the edit icon in the upper right corner to update or make some changes, which is handled by the `ProfileEditUserActivity` class.

#### 3e-1i. ProfileEditUserActivity class

It will also use the `loadMyInfo()` method to display all profile details. If the user wants to change a new profile image, the `showImagePickDialog()` method will be called to let the user update the profile image by taking a new photo with the camera or uploading a photo from the gallery storage using `pickFromCamera()` or `pickFromGallery()` methods, but these two operation needs to check the camera and storage permissions handled by the `checkCameraPermission()` and `checkStoragePermission()` methods, if not, it will ask the user for permission using the `requestCameraPermission()`,

requestStoragePermission() methods. If the user wants to change a new full address, he can directly click the GPS icon in the upper right corner, but the system will use checkLocationPermission() and requestLocationPermission() to check and request the location permission from the user, so that detectLocation() and findAddress() can be used to detect and find the user's location or address. After the user has made changes, the inputData() method will use the updateProfile() method to get all the edited data and update the profile details to Firebase Realtime Database, but the profile image will be uploaded to Firebase Storage.

### **3e-2. FAQUserActivity class**

This page just shows all frequently asked questions and its supported answers without using any method as all questions and answers are written in plain text using TextView.

### **3e-3. CSUserActivity class**

The user can choose to contact Banky using phone, WhatsApp or email. If the user chooses to make a phone call, the dialPhone() method will be called to redirect the user to the default phone dial screen and display the Banky phone number on the dial pad. If the user wishes to contact Banky using WhatsApp, the appInstalledOrNot() boolean method will check if WhatsApp is installed on the user's phone, otherwise an error message will be displayed saying "WhatsApp is not installed on your device". If the user would like to send an email to Banky, the system will redirect the user to Email page handled by EmailBankUserActivity class.

### **3e-3i. EmailBankUserActivity class**

The inputData() method is used to check and make sure the email subject and body entered by the user are not empty after completing the email composition, after that it just opens the default email application on the user's phone and displays the Banky email in the email recipient line, and the subject and body of the email that the user just typed.

### **1a. RegisterAdminActivity class**

This registration management page is not much different from the RegisterUserActivity class as it also uses the inputData(), createAccount() and saverFirebaseData() methods,

but the text field for filling in the user's identification number becomes the job position text field.

#### **4. MainAdminActivity class**

It will use loadMyInfo() to load and display the bank manager's name, title and bank account number. The bank manager has 5 main functions such as Manage Customers, which is handled by the ManageCustomerAdminActivity class, Manage Accounts that redirects the page to display the Firebase Authentication webpage, Manage Loan, which is handled by the supporting class ManageLoanAdminActivity, and the Manage Reporting, which is handled by the supporting class ManageReportAdminActivity and contact clients handled by the ManageCSAdminActivity class.

#### **4a. ManageCustomerAdminActivity class**

It uses the loadCustomerDetails() method to load customer details such as customer name, profile image, Banky account number, phone number, and email address retrieved using ModelCustomer and AdapterCustomerAdmin to display them in a list. It will also display an online or offline status mode next to each customer's picture, with a green circle only if the customer is currently online. The bank manager can click on any customer in the customer list to view a pie chart of that customer's account balances, or edit and modify that customer's profile details as handled by the CustomerDetailsAdminActivity class.

#### **4a-1. CustomerDetailsAdminActivity class**

The setupPieChart() method is used to display the available balance, current balance, term deposit balance, total received, total sent, and total paid amount retrieved from the Firebase Realtime Database in a pie chart. If the bank manager wants to edit or modify the customer profile details, he/she needs to turn on the Update Customer Profile switch. In this way, the loadCustomerInfo() method can only load the customer information such as customer's profile picture, name, identification number, Banky account number, phone number, email address and even annual interest rate, so that the bank manager can use updateCusProfile() method to update the customer information in Firebase Realtime Database only after the inputData() method has checked none of the text fields are empty.

#### **4b. ManageLoanAdminActivity class**

It uses the `loadLoanAdminItems()` method to load loan application details such as requested ID, applicant name, loan type along with its interest rate, loan amount, months to repay the loan, loan application date time dan loan status retrieved using `ModelLoan` and `AdapterLoanAdmin` to display them in a list. The bank manager can click on any loan application in the list to view the full loan application details, including the monthly repayment amount, total interest charged, and total cost processed by the `EditLoanStatusAdminActivity` class.

##### **4b-1. EditLoanStatusAdminActivity class**

This class is used the `loadLoanInfo()` method to retrieve the selected loan application details, such as applicant name, account number, loan type, loan term, start date, monthly repayment amount, loan amount, interest rate, total interest charged, and total cost. The bank manager can click the edit icon button in the upper right corner, and a dialog box will pop up to select "Pending", "Approved" or "Rejected", which is handled by the `editLoanStatusDialog()` method. In the `editLoanStatusDialog()` method, it calls the `editLoanStatus()` method to update the loan status based on the options selected by the bank manager. At the same time, the `updateApplicantLoanStatus()` method will also be used to update the applicant's loan status based on the options selected by the bank manager. If the bank manager rejects the loan application, then he/she should click the add comment icon button, leave some message in the dialog, provide the reason for rejection using the `addCommentDialog()` method. It will then use the `updateCommentToDb()` and `updateApplicantComment()` methods to update the comments for the bank manager and applicant Firebase.

#### **4c. ManageReportUserActivity class**

The bank manager can click "Daily Active User" to see the number of daily active users from the Banky Firebase Overview webpage. If a bank manager wants to know which feature is more popular or used by more people, he/she can click the "Event Clicked" and the user will redirect to the Banky Google Analytics webpage. The User Account Chart is meant to give the bank manager an idea of how many people have opened, disabled or deleted accounts until now, which handled by the `ReportUserAccAdminActivity` class. The Loan Status Chart will count the total number of loans pending, approved, and rejected so far, which this operation is processed by

the ReportLoanStatusAdminActivity class. The Loan Ranking Graph will count total number of each loan that has been applied using the ReportLoanRankAdminActivity class. The Daily Transaction Graph will sum the selected date of payments, transactions, and loan applications using ReportDailyTransAdminActivity class.

#### **4c-1. ReportUserAccAdminActivity class**

By default, without choosing any date to do the filtering, the setupPieChart() method will be called to load and retrieve the overall total number of accounts opened, enabled, and disabled from Firebase Realtime Database and displayed as a pie chart. After the bank manager selects a date from the calendar using datePickDialog() method and clicks the search button, the setupPieChartWithSelectedDate() method is called to display the total number of accounts opened, enabled, and disabled that occurred on the selected date.

#### **4c-2. ReportLoanStatusAdminActivity class**

By default, no dates are selected for filtering, which also uses the setupPieChart() method to load and retrieve all the total pending, approved, and rejected loans from the Firebase Realtime Database and displays as a pie chart. After the bank manager selects a date from the calendar using datePickDialog() method and clicks the search button, the setupPieChartWithSelectedDate() method is called to display the total number of pending, approved, and rejected loans which the actions took place on the selected date.

#### **4c-3. ReportLoanRankAdminActivity class**

This class is used the setupBarChart() method to load and retrieve the total count of each loan type from Firebase Realtime Database and displayed as a bar graph if the bank does not do any date filtering. If the bank manager wants to view the count of each loan type that applied on the selected, he/she can select a date from the calendar using the datePickDialog() method and clicks the search button, the setupPieChartWithSelectedDate() method is called to meet his/her requirement.

#### **4c-4. ReportDailyTransAdminActivity class**

This page is used to summarize all payments, transactions and loan application amounts that occurred on the selected date. The bank manager needs to use the datePickDialog() method to pick a date from the calendar, so sumDailyPayment(), sumDailyTransaction(), sumDailyLoanApplication() will be called at the same time to sum the amount retrieved from the Firebase Realtime Database. After getting the data, setupBarChart is used to display in a bar chart.



#### **4c-5. ManageCSAdminActivity class**

It uses the loadCustomerDetails() method to load customer details such as customer name, account number, phone number and email number retrieved from the Firebase Realtime Database by using the ModelCustomer to match variable names in Firebase to the system, and using the AdapterContactCustomerAdmin to display the list of view. It will also display phone and email icons for each customer in the customers list. When the bank manger clicks the phone icon button for a specific customer, he/she will be redirected to default phone dialing screen with the customer's phone number displayed on the dial pad. Suppose the bank manager wants to write an email to a specific customer, he/she just clicks the email icon button on the specific customer and the system redirects the user to the email page which is handled by the EmailBankUserActivity class.

**CHAPTER 5 SYSTEM IMPLEMENTATION****5.1 Hardware Setup**

## Specification of Laptop

<b>Description</b>	<b>Specifications</b>
Model	Asus Vivobook X510UQR
Processor	Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz 1800 Mhz, 4 Core(s), 8 Logical Processor(s)
Operating System	Windows 10
System type	64-bit operating system, x64-based processor
Graphic	Intel(R) UHD Graphics 620, NVIDIA GeForce 940MX
Display Resolution	1920 x 1080 x 60 hertz
Installed RAM	12.0 GB
Storage	476 GB

**Table 5.1.1 Specification of Laptop**

## 5.2 Software Setup

### Software Requirement for Android Studio

Description	Specifications
Operating System	Android 4.1.1
Processor	X86_64 CPU architecture; 2nd generation Intel Core or newer, or AMD CPU with support for a Windows Hypervisor
Storage	Minimum 8 GB of available disk space
RAM	Minimum of 4 GB, 8 GB is recommended
Display Resolution	Minimum 1280 x 800

**Table 5.2.1 Hardware Requirement for Android Studio**

### Requirement for Banky Mobile Application

Description	Specifications
Operating System	Android 8.0 and above
Storage	Minimum 2 GB of available storage is recommended for improved the performance, even the Banky application just requires 25 MB
RAM	Minimum of 4 GB, 8 GB is recommended
Screen Resolution	Minimum 1080 x 2220

**Table 5.2.2 Requirement for the system**

### **5.3 Setting and Configuration**

After each activity of development is completed, the developer can run the emulator by simply clicking on the “Run” or “Debug” icon. But before starting to run or debug the application, if this is the first time running the application, the developer must first create a virtual device to configure it. This proposed system is running an emulator that its phone model called Google Pixel 3a with a resolution of 1080 x 2220: 440 dpi for results testing, which this emulator is a built-in virtual device in Android Studio. The Android version of this Google Pixel 3a emulator is Android 8.0.0. The database used to store all the records of transactions, payments, loan applications, budget history, and profile information is the Firebase Realtime Database, which is a cloud database. All photos will be stored using Firebase Storage. Firebase Authentication is used to store email addresses and passwords, so whenever a user wants to log into Banky application, Firebase Authentication calls to check if the email and password match the ones already saved in the list. This proposed system also uses the Google Analytics for the Firebase to get to know some analytics such as the number of views per activity (screen/page), average engagement time per activity, and event counts by event name over time.

## 5.4 System Operation

The digital-only banking mobile application is named “Banky” because it carries the meaning of “This Digital-Only Bank is Trusty”, since it is a digital-only bank, and it just took the “y” from the “Trusty” word and renamed it to “Banky”.

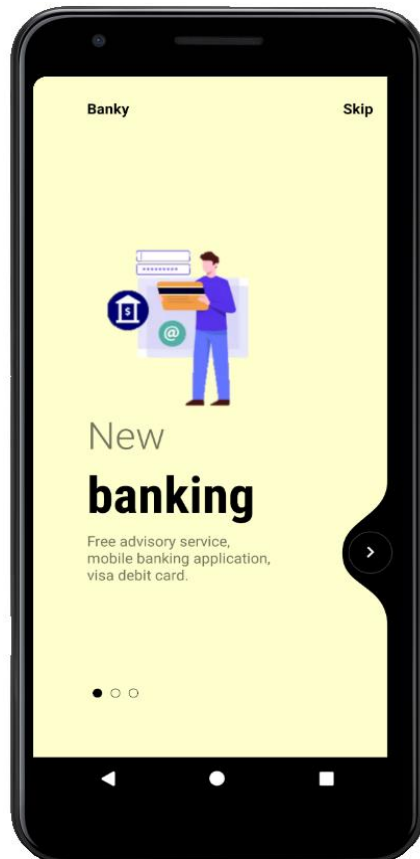


Figure 5.4.1 Onboarding Screen1

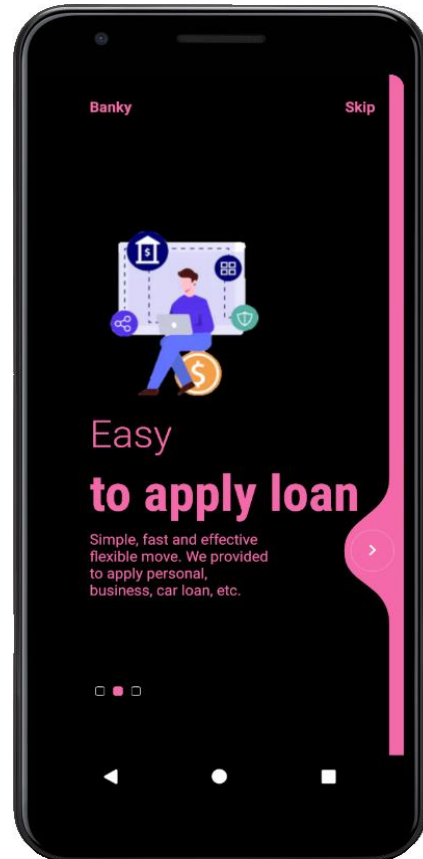
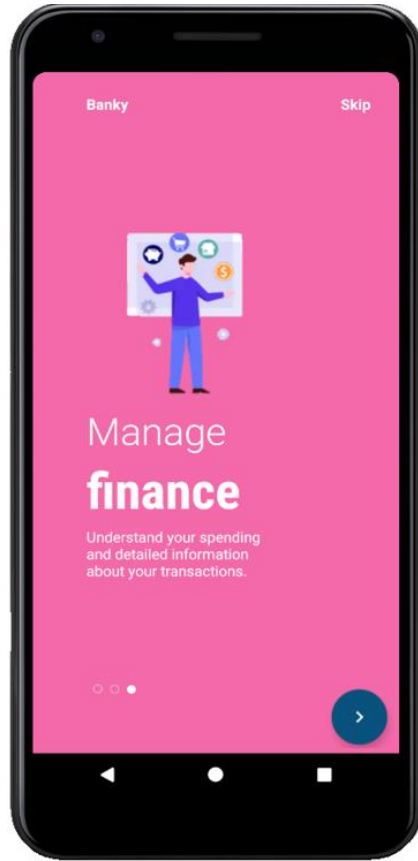
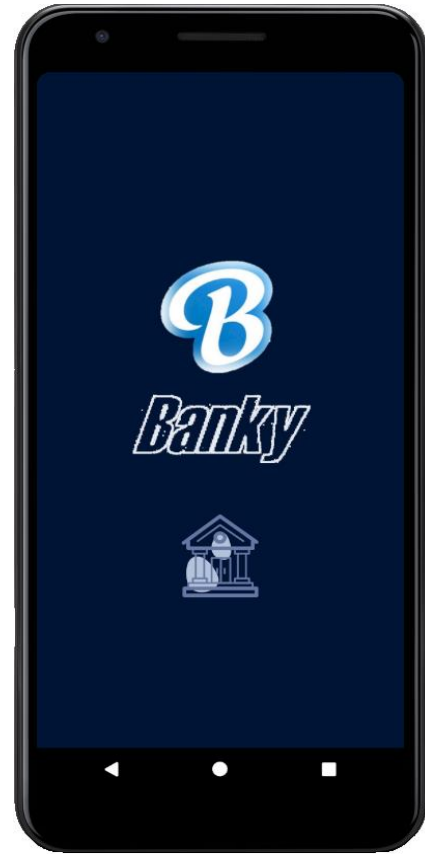


Figure 5.4.2 Onboarding Screen2

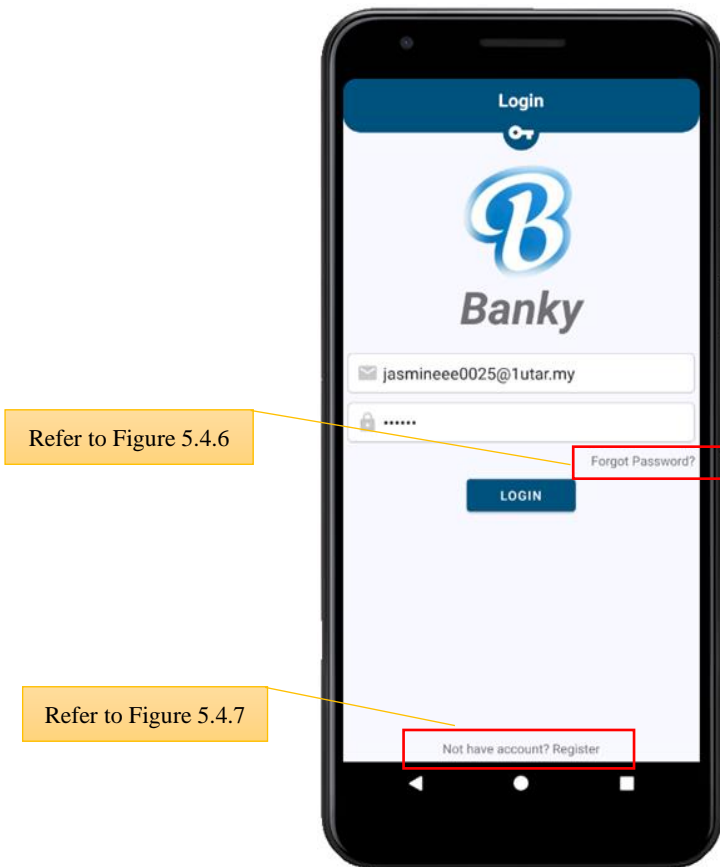


**Figure 5.4.3** Onboarding Screen3

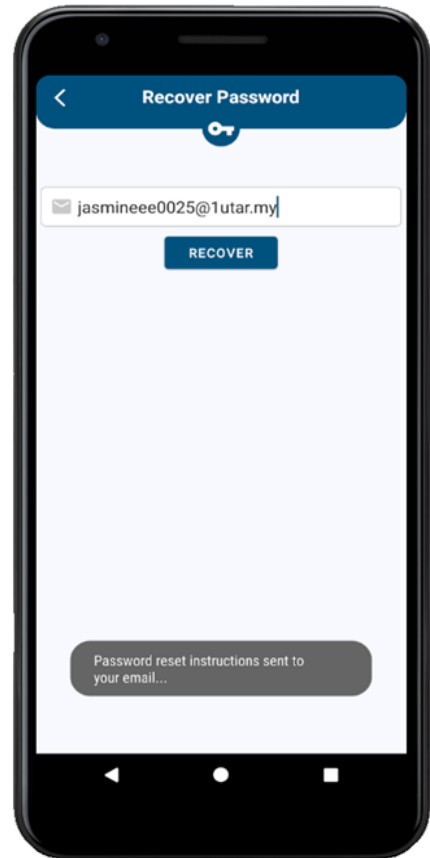


**Figure 5.4.4** Splash Screen

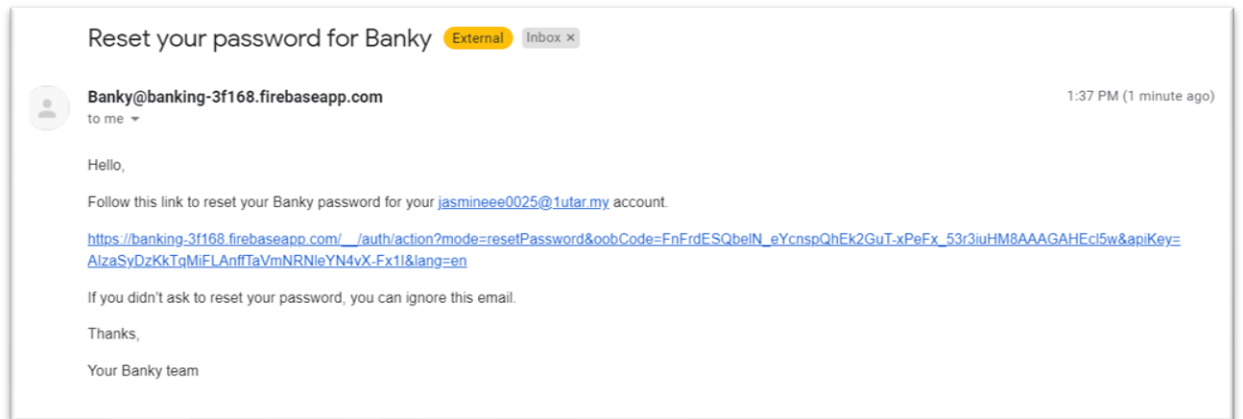
When the first-time user is launching the digital-only banking mobile application, it will display the onboarding screen (*refer from Figure 5.4.1 to Figure 5.4.3*) to welcoming the user. Basically, it is liked a walkthrough of the mobile application, and its purpose is to let the user knows what mobile application it is. For the Figure 5.4.4, no matter the user is existing user or first-time user, whenever they opened the digital-only banking mobile application, it will display the splash screen with its bank animation.



**Figure 5.4.5** Login Page



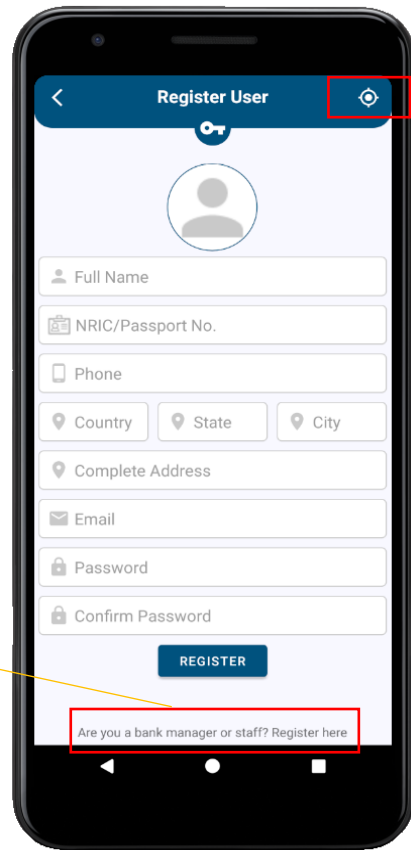
**Figure 5.4.6** Forgot Password



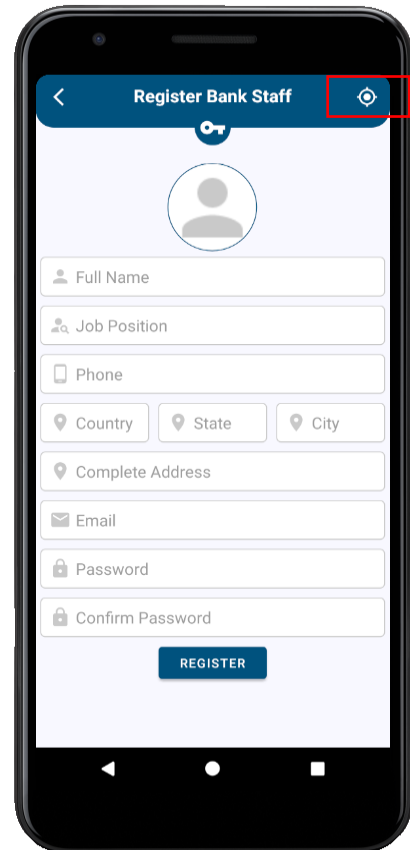
**Figure 5.4.7** Reset Password

The user is required to enter the email and password (*refer to Figure 5.4.5*), then the Firebase Authentication in this case will be used to validate the email and password, and also check for whether this user exists in the system or not. Once the system checked, the user only can access to the main page of the digital-only banking mobile application. But before that, the system will also check for the user type, there will be two types of user accounts in my digital-only bank such as normal customer of digital-only bank and the bank manager for this digital-only bank. These two user account will have different layout design and different information and functionalities. If let's say the user forgot his/her password, then he/she can go to click on the "Forgot Password" in *Figure 5.4.6*, it will require the user to type his/her email address. The system will then send an email to him/her in order to let them reset the password over there (*refer to Figure 5.4.7*). The new password also must consists of at least six characters and above.





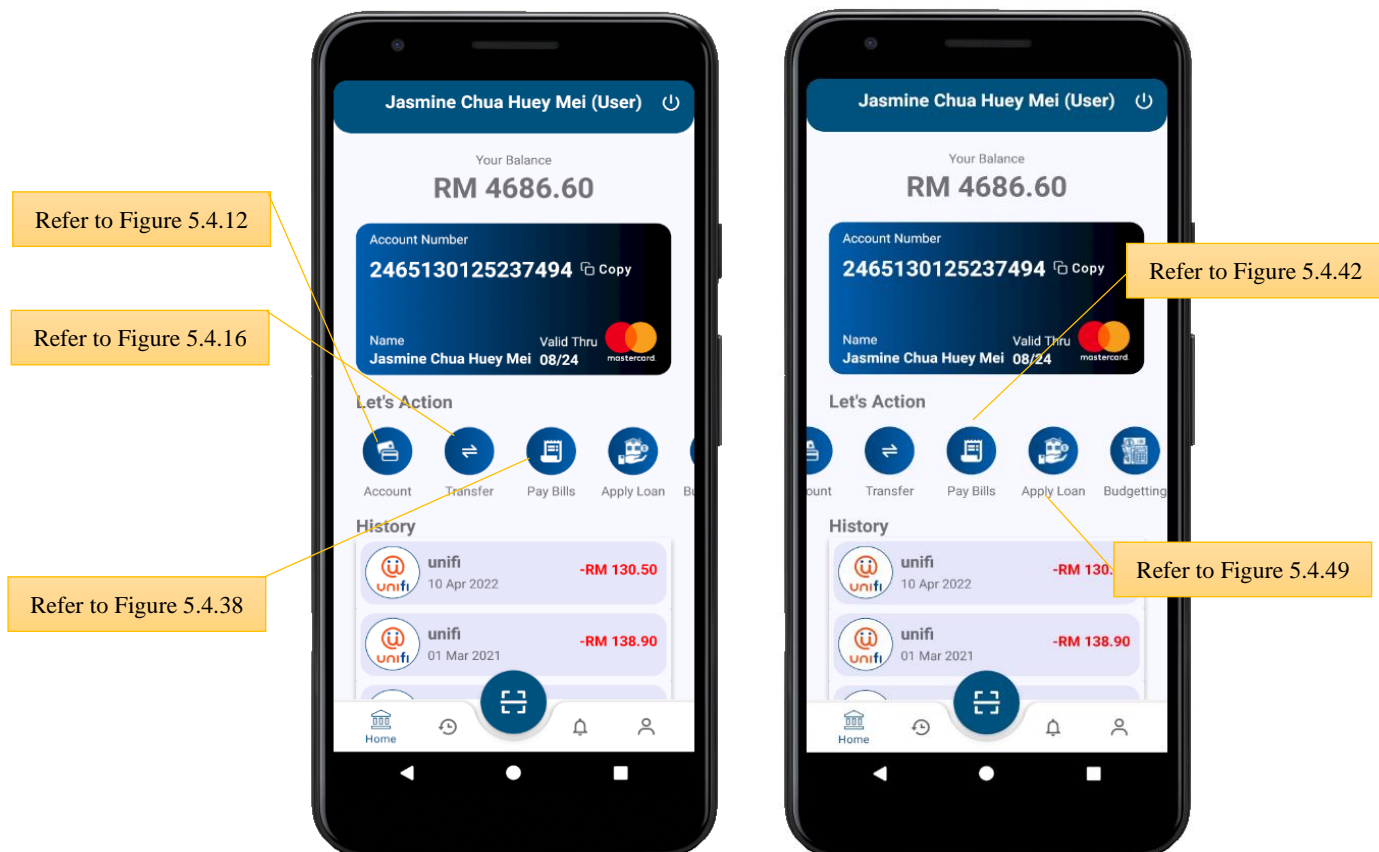
Refer to Figure 5.4.9



**Figure 5.4.8** Register User Page

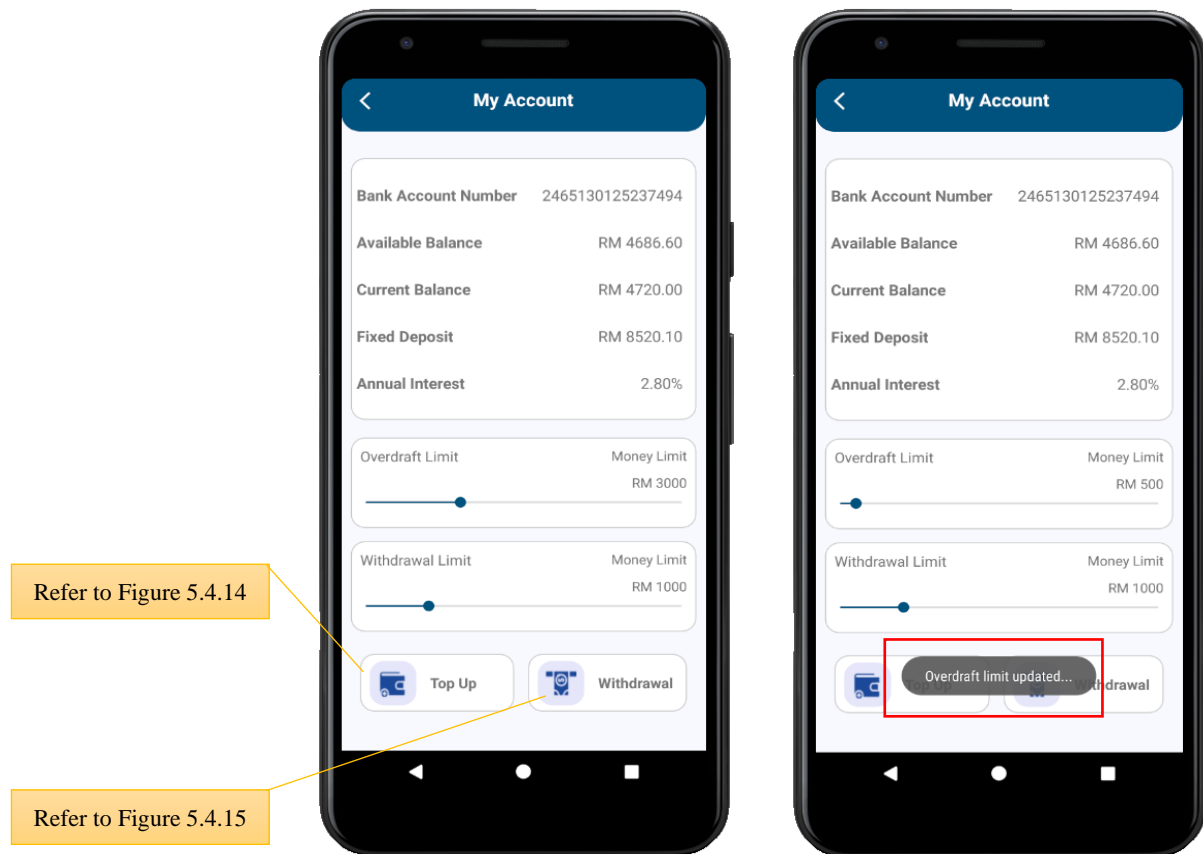
**Figure 5.4.9** Register Bank Staff Page

In order to become Bank user, the user can perform digital self-registration open account remotely in under few minutes. The system will toast a message whenever there is any missing value in the field except for the profile picture. Whether to upload the profile picture, it is optional, if the user chooses to upload the profile picture, then the profile picture can be picked from camera or gallery and it will be stored under Firebase Storage. The icon at the right-hand top corner is used to automatically detect the current location of user, so that the user not needs to type the country, state, city and complete address by themselves. All the personal details will be stored in Firebase Realtime Database.



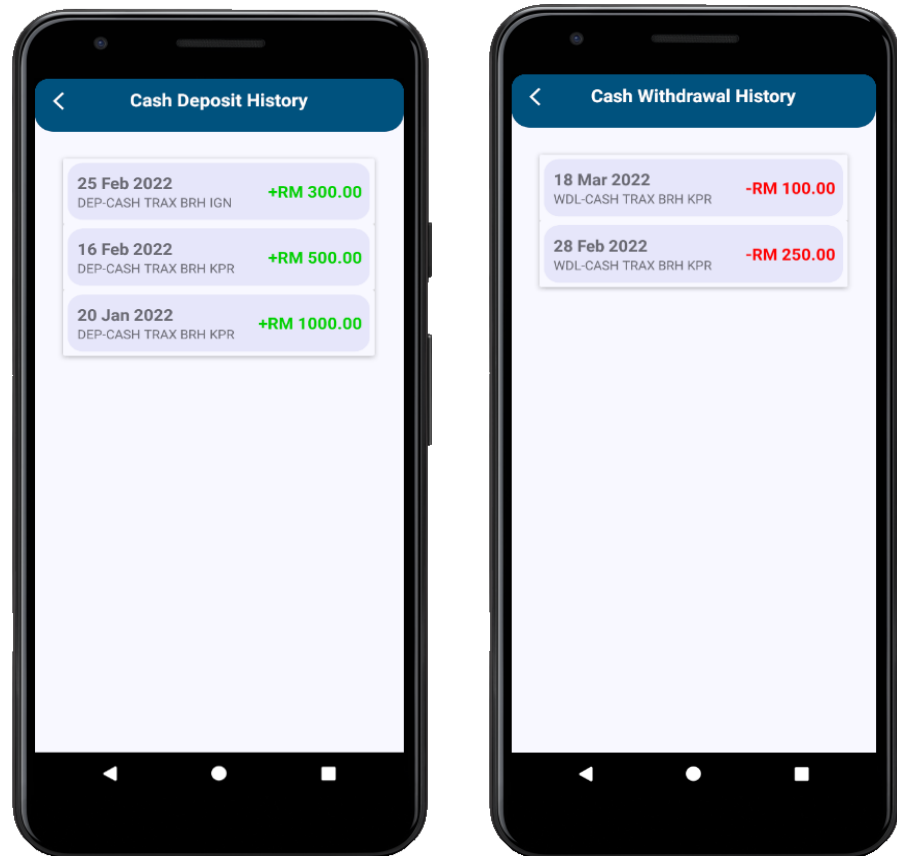
**Figure 5.4.10** User Main Page **Figure 5.4.11** User Main Page (cont)

The toolbar will display the name that typed by the user when the time he/her registers to become Banky user. The account number is automatically generated by the system with the formula of  $\text{Math.random}() * 100000000000000L$  concenate with 2400000000000000L. The reason for doing this because I wish that everyone in my digital-only bank users with the account number starting with “24....”. It does provide the copy function for the user as well, so whenever they want to copy their account number to their friends or families will be easier. This digital virtual visa card can be used anywhere that Visa is accepted and also be used for online purchases. The digital virtual visa card will also display the name of the user along with the valid thru. This digital-only banking mobile application has few functionalities such as account, transfer, pay bills, apply loan and budgetting which display all of them in a horizontal scroll view. The bottom part of the main page is the history of paymets which displays in recycler view form.



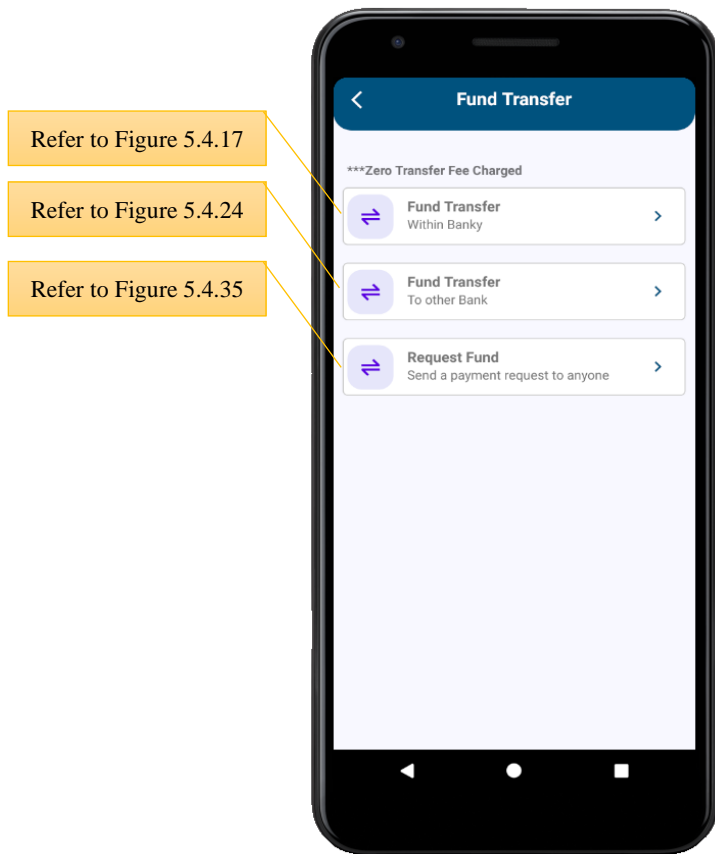
**Figure 5.4.12** Account Page **Figure 5.4.13** Account Page (cont)

When clicking the account icon in the Main Page (refer to Figure 5.4.10), the user redirects to my account page (refer to Figure 5.4.12) along with displaying the account number, the available balance, the current balance, the fixed deposit balance, annual interest rate, the overdraft limit and the withdrawal limit. New users will not show any interest rate, only for those users has been joining and activating our Banky for half a year can get annual interest rate. The user can update his/her overdraft limit and withdrawal limit to Firebase by just adjusting the seek bar left or right, once it is done it will toast a message showing the overdraft limit or withdrawal limit updated (refer to Figure 5.4.13). The default overdraft limit set by the Banky system is RM3000 whereas the default withdrawal limit set by the Banky system is RM1000. If the user wishes to top up his/her account balance, then he/she can use the Touch N Go eWallet or go to physical 7-eleven or go to ATM machine for performing the top up balance. If the user wishes to withdraw the money from his/her account balance, then the user can go to visit ATM from Maybank, Public Bank, OCBC and other banks to perform Cardless Cash Withdrawal without charging any transaction fee between interbank because I did fight for my digital-only bank to negotiate to all those banks mentioned above.



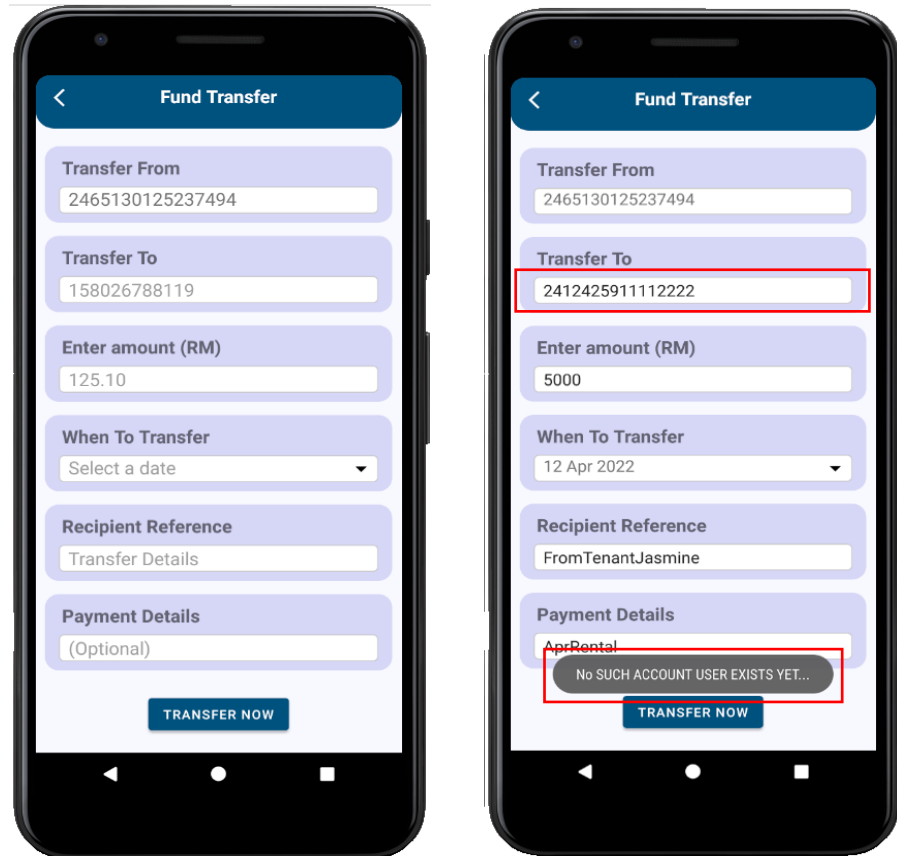
**Figure 5.4.14** Cash Deposit History Page **Figure 5.4.15** Cash Withdrawal History Page

When the top up icon in the Account Page is clicked (*refer to Figure 5.4.12*), it redirects the user to the Cash Deposit History Page (*refer to Figure 5.4.14*). If the user clicks the withdrawal icon from the Account Page (*refer to Figure 5.4.12*), the user will be redirected to the Cash Withdrawal History Page (*refer to Figure 5.4.15*). “DEP” in *Figure 5.1.14* stands “Deposit”, while “WDL” in *Figure 5.4.15* stands for “Withdrawal”. The transaction reference will show to the user where the transaction location took place, the user may refer the branch code from this website [44]. For example, BRH KPR stands for Branch Kampar, BRH IGN stands for Ipoh Garden and so on.



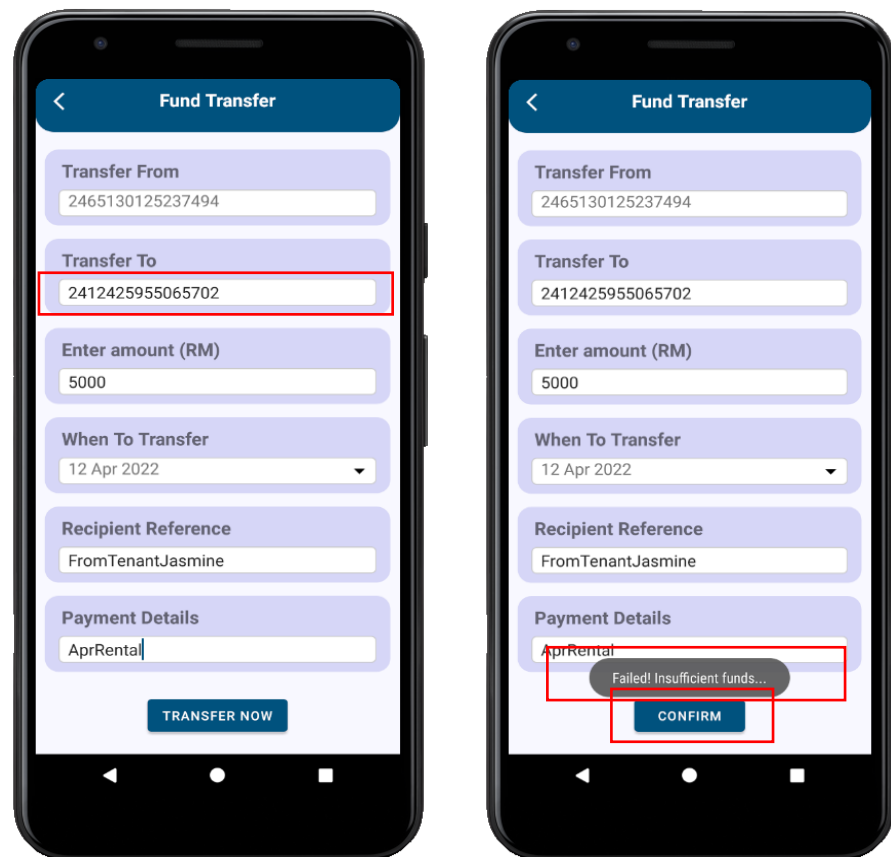
**Figure 5.4.16** Fund Transfer Page

When the user clicks the transfer icon from the Main Page (*refer to Figure 5.4.10*), it redirects the user to the Fund Transfer Page (*refer to Figure 5.4.16*). There are few types of fund transfer way such as transfer within this digital-only bank, transfer to other banks, and request the fund from other within this digital-only bank.



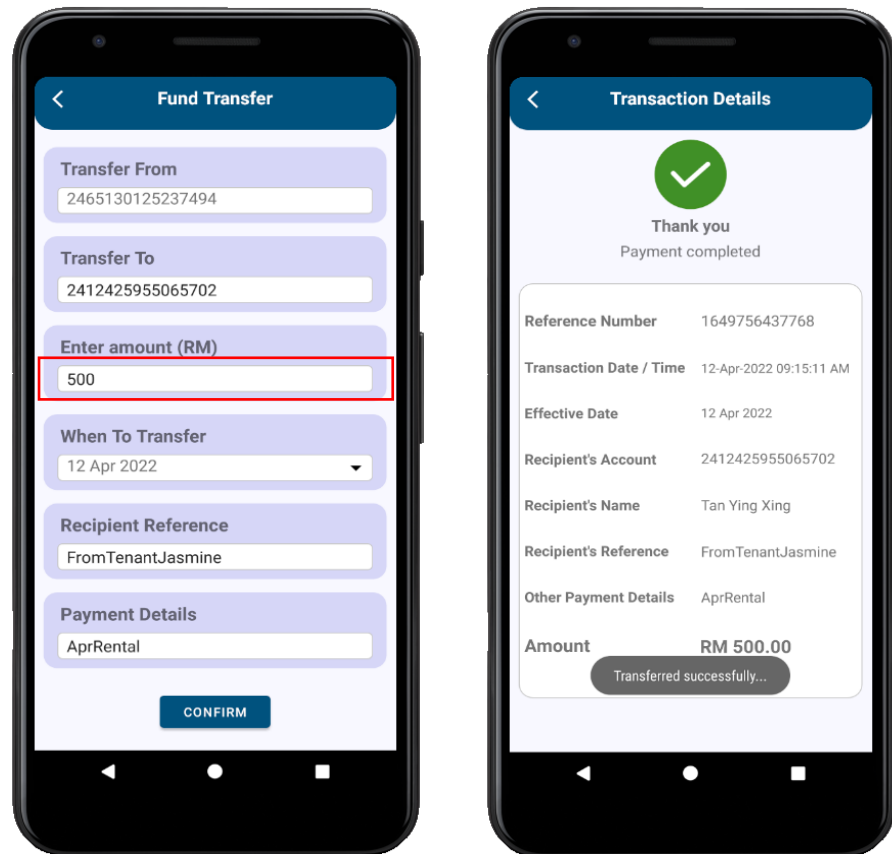
**Figure 5.4.17** Transfer Within Banky Page **Figure 5.4.18** No Such Account User Exists Yet

The *Figure 5.4.17* shows that basic transaction details to be filled whenever making a transaction, the system will then check any missing value in the field expect for payment details, then it will toast the message to point out which field has not filled in yet. Once the user clicks the transfer now button, then the system will need checking for some conditions before letting the transaction done such as if let's say the user just simply to type an account number that is not existing user, then it will toast a message to the current user saying that "no such account exists yet..." and stop continue to process the transaction (*refer to Figure 5.4.18*).



**Figure 5.4.19** Transfer Within Bank Page2 **Figure 5.4.20** Insufficient Fund to Transfer

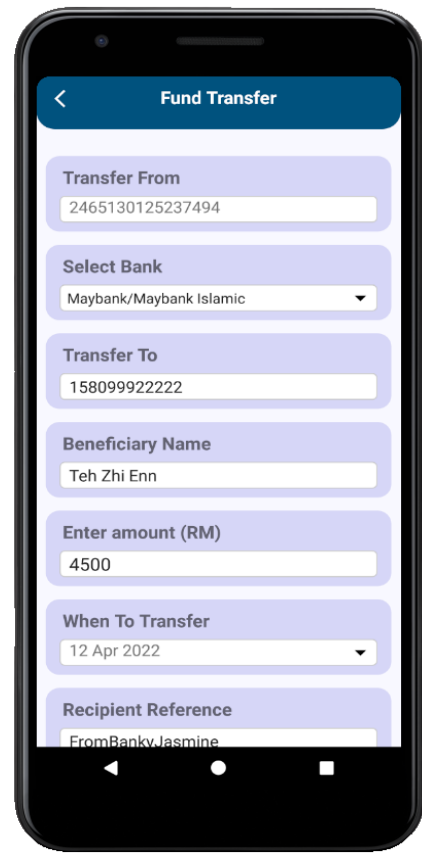
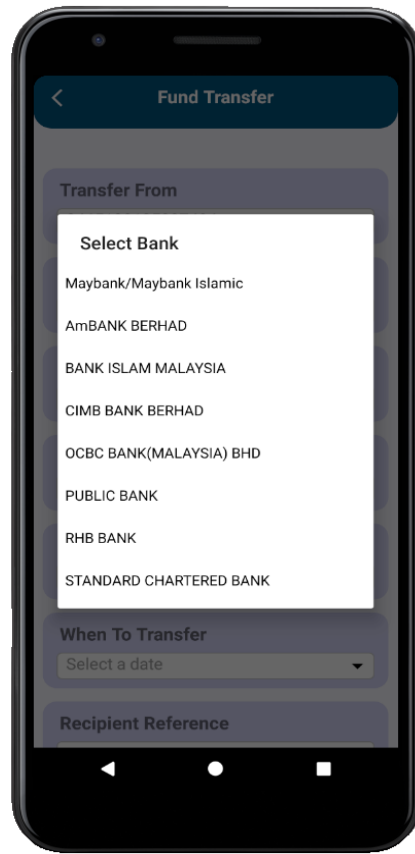
To check whether the account number that the user typed is under one of our digital-only bank user or not, the current user can notice the changes in the word of the button from “Transfer Now” to “Confirm”. The second scenario will be checking for whether the amount to transfer is less than or more than the available balance of the user. For example, the user accidentally typed wrong the amount, he/she supposedly wants to transfer only RM 500.00, but he/she enters RM5000.00 and some more the available balance of the current user is only RM 4686.60 (refer to Figure 5.4.10) which is less than the amount of RM 5000.00 that he/she wishes to transfer (refer to Figure 5.4.19 and Figure 5.4.20), so in the cases, it will toast a message saying that “Failed! Insufficient funds...” and stop continue to process the transaction.



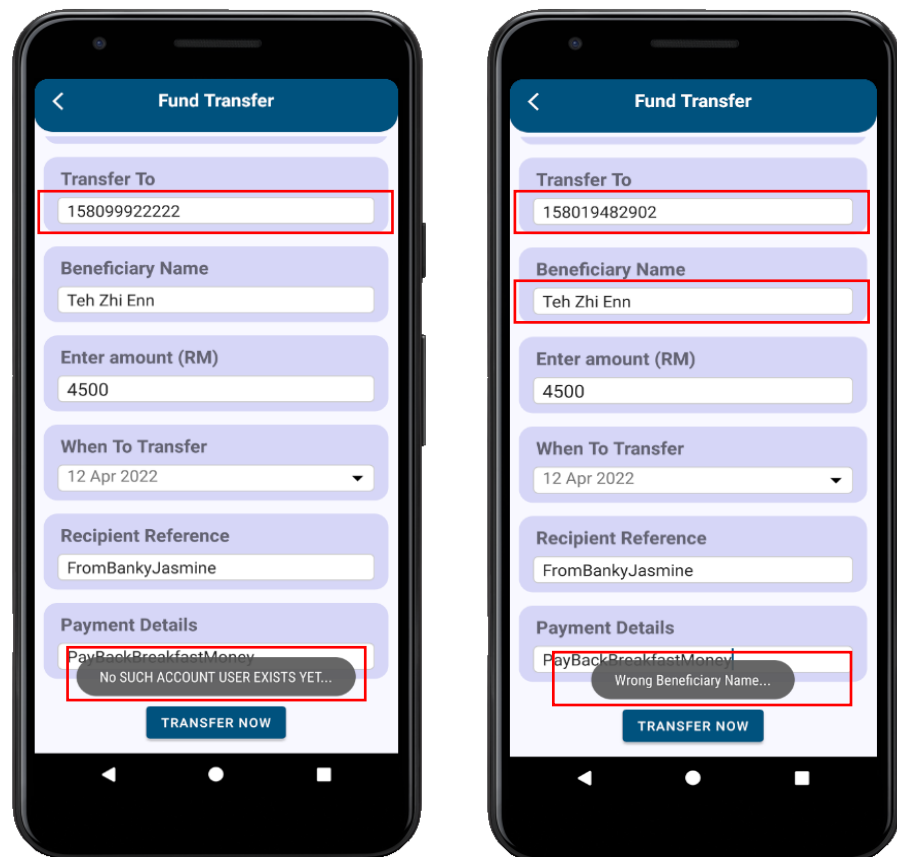
**Figure 5.4.21** Transfer Within Banky Page3 **Figure 5.4.22** Transaction Receipt

When the current user is fulfilling the conditions such as enter the account number of beneficiary is under one of our Banky users and some more the transfer amount is less than or equal to his/her available balance, then it only will be able to transfer successfully to the beneficiary and print the receipt (*refer to Figure 5.4.22*). Once the transaction is done, it will update all the transaction details to Firebase Realtime Database of both current user and beneficiary user and also it will display the transaction details under the history navigational bottom bar there (*refer to Figure 5.4.68 & Figure 5.4.69*). Apart from that, the system will also deduct the transfer amount from the current user's available balance (*refer to Figure 5.4.32*) and increase the available balance of beneficiary user account by the amount that has been transferred from the current user.



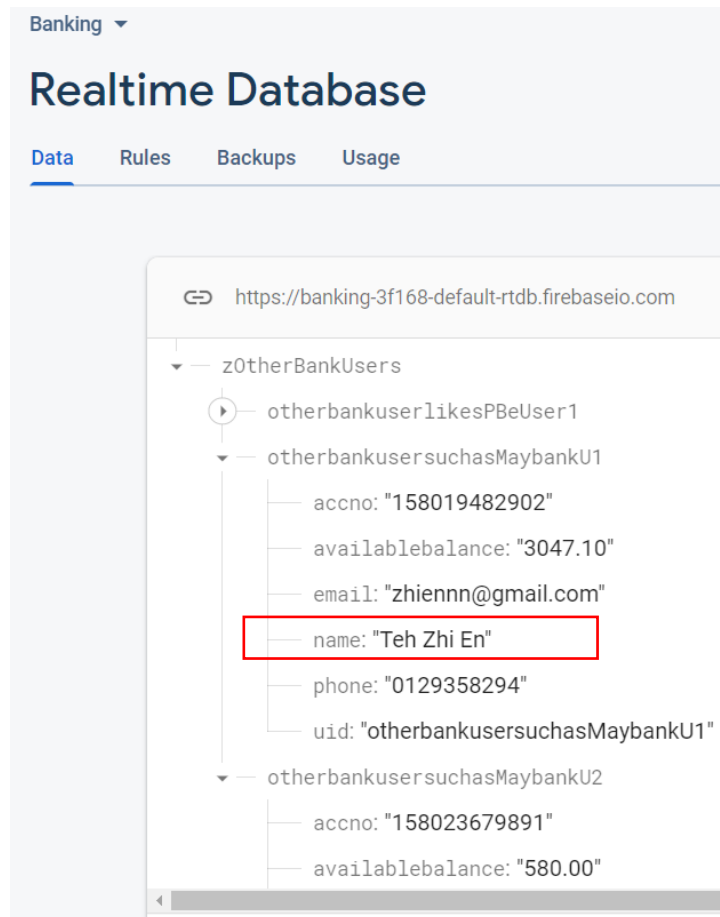


**Figure 5.4.23** Selection of Other Bank **Figure 5.4.24** Transfer To Other Bank Page



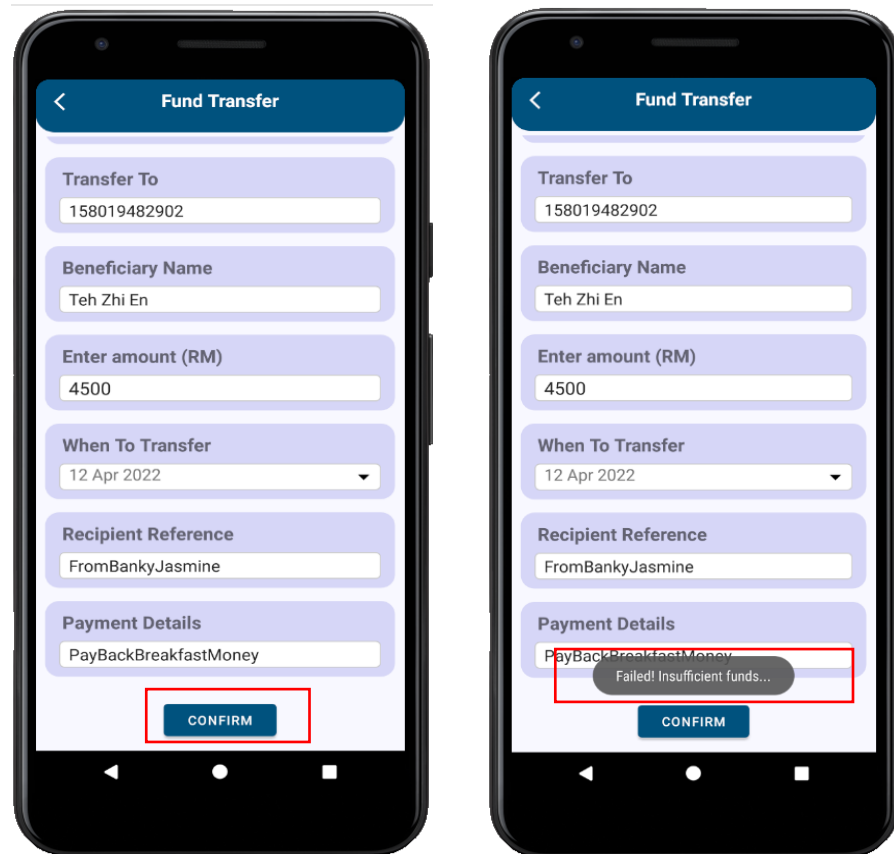
**Figure 5.4.25** No Such Account User Exists Yet **Figure 5.4.26** Wrong Payee Name

Once the user clicks the transfer now button, then the system will need checking for some conditions before letting the transaction done such as if let's say the user just simply to type an account number that is not existing user in the other bank dummy table called "zOtherBankUsers" in the Firebase Realtime Database (refer to Figure 5.4.27), then it will toast a message to the current user saying that "no such account exists yet..." and stop continue to process the transaction (refer to Figure 5.4.26). After the user enters the correct another bank's user account number, it will then compare the entered beneficiary name with the exact beneficiary name of that account number from the other bank dummy table. In this scenario, the user accidentally added an alphabet "n" in the end of the beneficiary name, it will toast a message saying "Wrong Beneficiary Name" and stop processing to transfer fund.



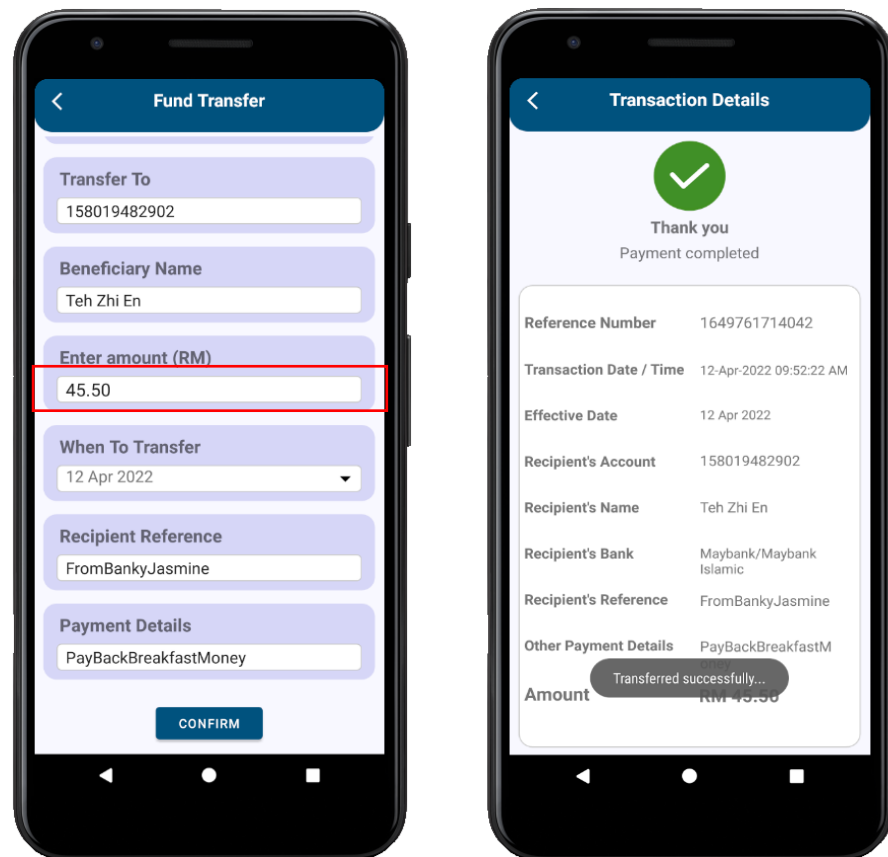
**Figure 5.4.27** Other Bank Dummy Table

I created a dummy table to store other bank information in Firebase Realtime Database (refer the link to <https://console.firebase.google.com/u/0/project/banking-3f168/database/banking-3f168-default-rtdb/data/~2FzOtherBankUsers>). In this way, when the users want to transfer funds to other banks, the system will use the data in this dummy table to check and compare account numbers and account holder names.



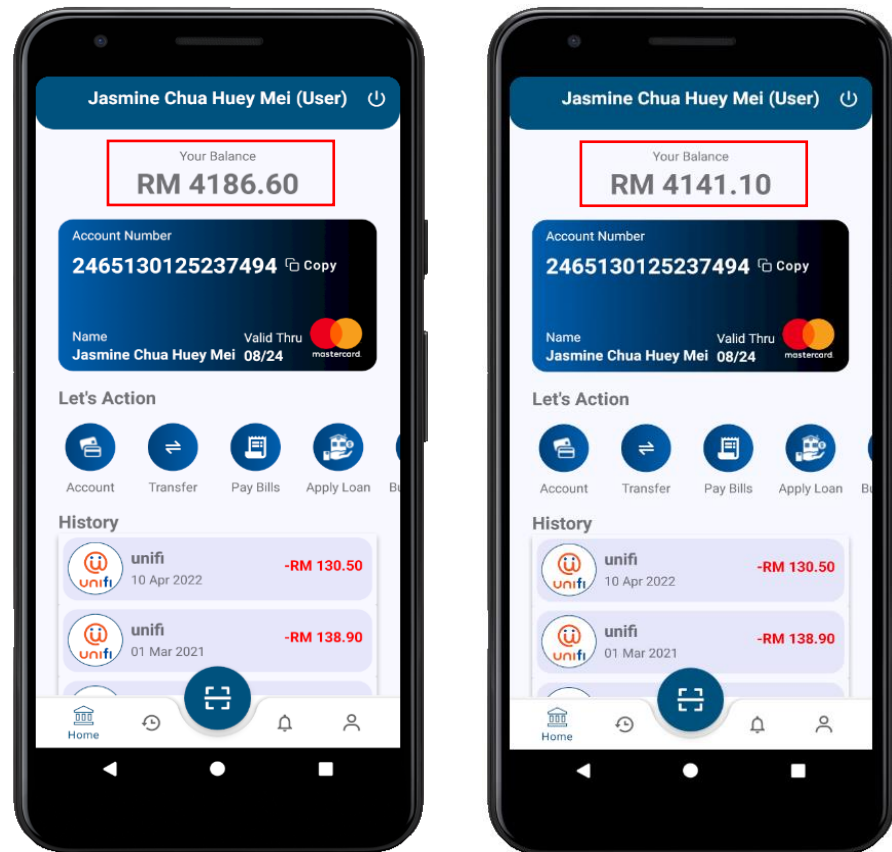
**Figure 5.4.28** Other Bank Account Exists **Figure 5.4.29** Insufficient Fund to Transfer

After the current has met the condition of entering correctly the another bank account number and the another bank account holder name, the word inside the button will change from “Transfer Now” to “Confirm” (refer to *Figure 5.4.28*). The system will then be checking to make sure that the available balance is enough to cover the transfer amount, else it will toast a message displaying “Failed Insufficient funds...” (refer to *Figure 5.4.29*). As currently the user only has RM4186.60 (refer to *Figure 5.4.32*) that the available balance of RM4686.60 (refer to *Figure 5.4.10*) has been deducted from RM500.00 (refer to *Figure 5.4.22*), which the available balance is less than the transfer amount, so it will not continue to proceed the process.



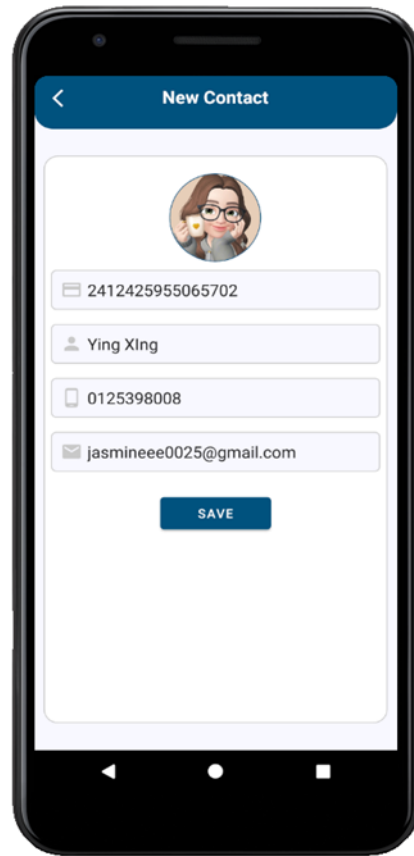
**Figure 5.4.30** Transfer To Other Bank Page2 **Figure 5.4.31** Transaction Receipt

Once all conditions are currently met, such as the existence of the other bank account, the name of the other bank account holder matches the name under the other bank account, and the transfer amount is less than or equal to the current user's available balance. Then it only will be able to transfer successfully to the beneficiary and print the receipt (*refer to Figure 5.4.31*). After the transaction is done, it will update all the transaction details to Firebase Realtime Database of both current user and beneficiary user and also it will display the transaction details under the history navigational bottom bar there (*refer to Figure 5.4.68 & refer to Figure 5.4.69*). Apart from that, the system will also deduct the transfer amount from the current user's available balance (*refer to Figure 5.4.33*) and increase the available balance of beneficiary user account by the amount that has been transferred from the current user.

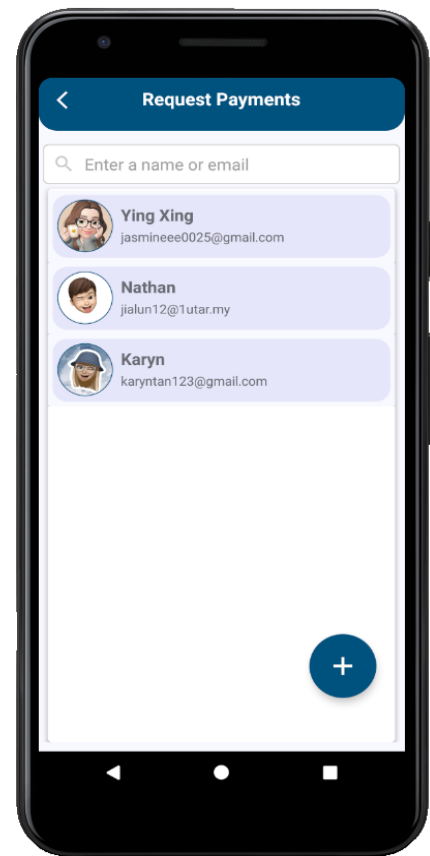


**Figure 5.4.32** Main Page-New Balance **Figure 5.4.33** Main Page-New Balance2

After the user transferred funds to other Bank user, the available balance has been deducted from RM4686.00 (refer to Figure 5.4.10) to RM500.00 (refer to Figure 5.4.22) and become RM4186.60 now (refer to Figure 5.4.32). Once after user made a transfer to other bank user, the available balance has been deducted from RM4186.60 (refer to Figure 5.4.32) to RM45.50 (refer to Figure 5.4.31) and become RM4141.10 now (refer to Figure 5.4.33).

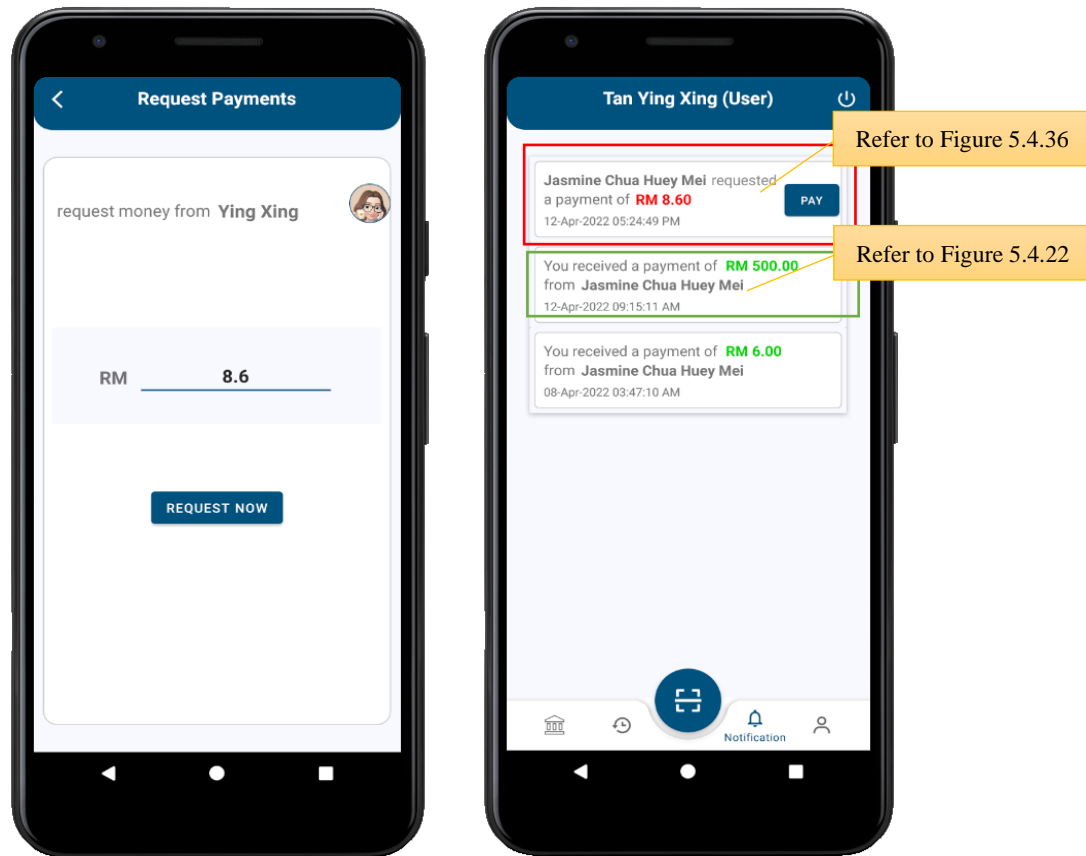


**Figure 5.4.34** Add New Contact



**Figure 5.4.35** Request Fund Page

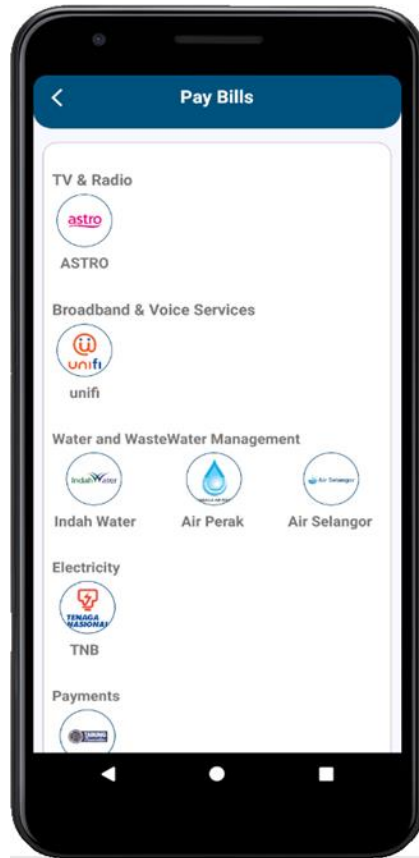
When the user clicks the request fund button in the Fund Transfer Page (*refer to Figure 5.4.16*), the system will redirect him/her to the Request Fund Page that showing all the contact (*refer to Figure 5.4.35*), so the user can click either one to request the fund. But if the user does not have any contact or wishes to add on more new contact, he/she can click the add icon floating button shown in *Figure 5.4.35*, then the user is redirected to the Add New Contact Page (*refer to Figure 5.4.34*), so when the user saved this contact, it will upload the contact details to Firebase Realtime Database.



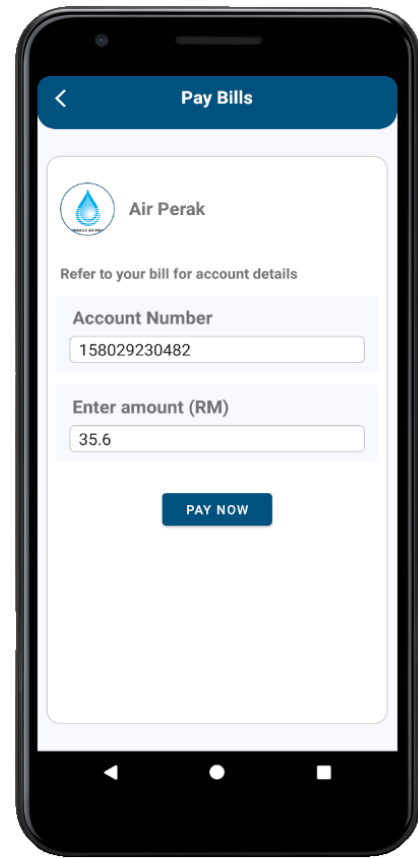
**Figure 5.4.36** Request Fund Detail    **Figure 5.4.37** Notification (Payer Side)

When the user chooses a contact from the contact list, then he/she needs to enter the request amount (*refer to Figure 5.4.36*). So the system will update this fund request to the payer's notification side (*refer to Figure 5.4.37*), the payer just clicks the pay button, and it will redirect the payer to the Pay Request Fund Page.



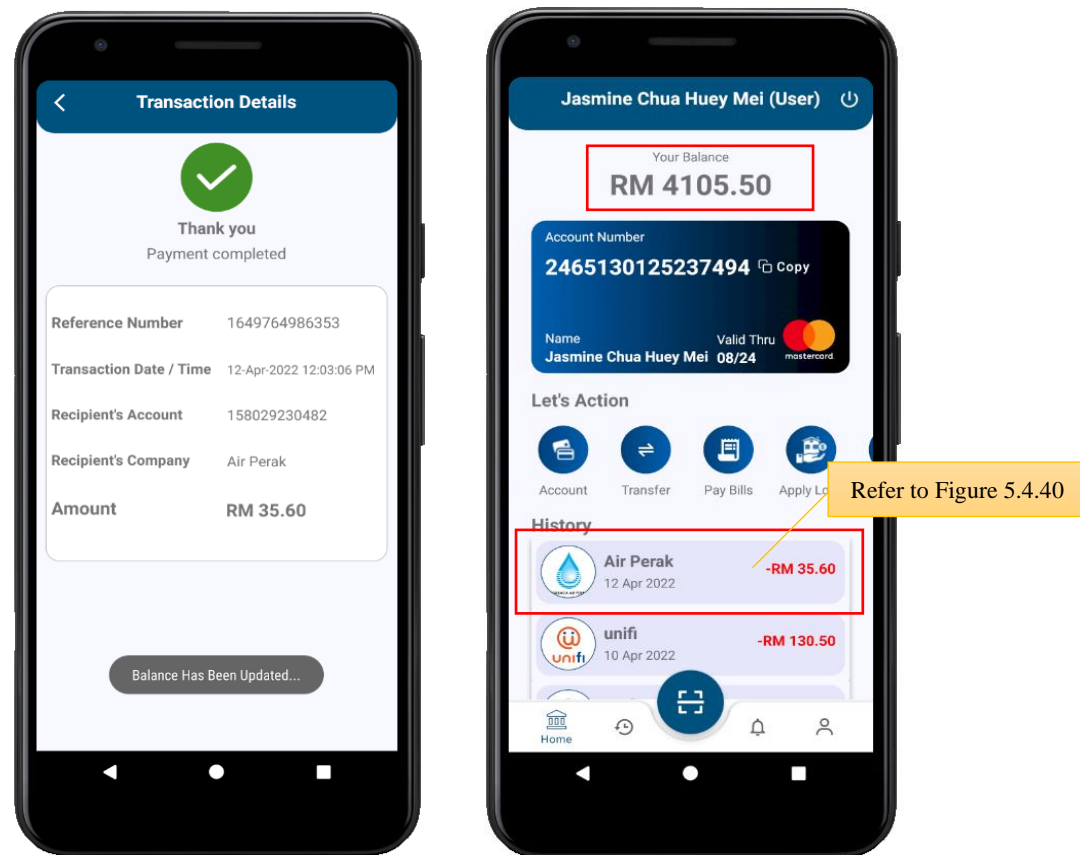


**Figure 5.4.38** Pay Bills Page



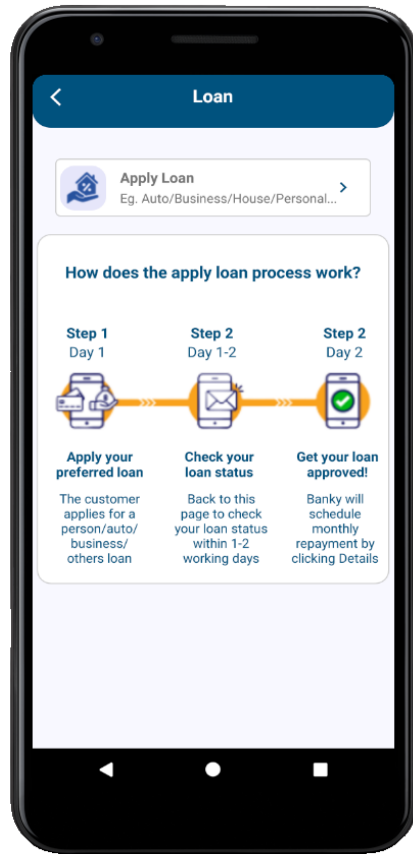
**Figure 5.4.39** Pay Bill Details Page

There are variety of types of bill to let the user chooses to make payments such as ASTRO bill, unifi bill, Indah Water bill and others (*refer to Figure 5.4.38*). For example, the user clicks on the Air Perak icon, then it will redirect to the Pay Bill Details Page there (*refer to Figure 5.4.39*) for letting the user to key in the Air Perak company's account number and amount to pay for the water bill. The account number can be gotten from the original bill. However, the system will also make sure that the amount to pay is less than the current user's available balance, else it will stop the payment and toast a message "Failed! Insufficient funds....".

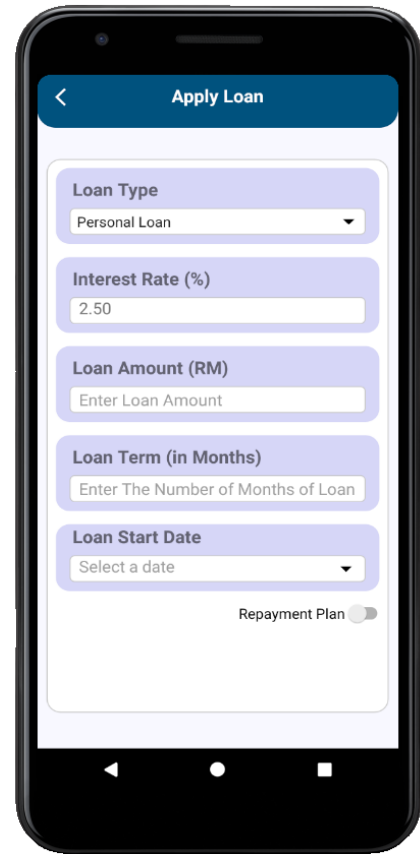


**Figure 5.4.40** Payment Receipt **Figure 5.4.41** Main Page-New Balance3

After clicking the pay now button in *Figure 5.4.39* and everything is already checked by the system and also follow the conditions or rules, then it will print out the payment receipt to the user (*refer to Figure 5.4.40*). Later on, the user can go to check the history of payments showed in main page has updated and the available balance has updated also. We can compare the *Figure 5.4.33* with *Figure 5.4.41*, which its available balance originally is RM4141.10, but after done pay the bill with RM35.60 (*refer to Figure 5.4.40*) and its available balance became RM4105.50 which is true because  $RM4141.10 - RM35.60 = RM4105.50$ . Apart from that, we also can notice that initially there is no Air Perak bill details under “History” word (*refer to Figure 5.4.10*), but now it showed over there (*refer to Figure 5.4.41*).

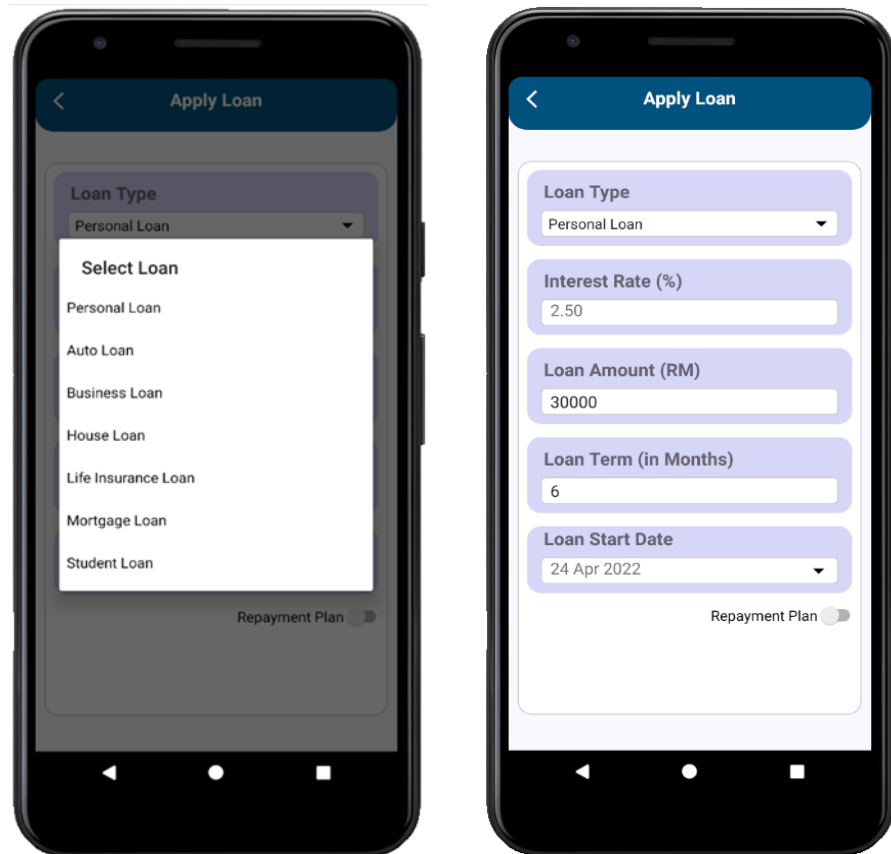


**Figure 5.4.42** Loan Page



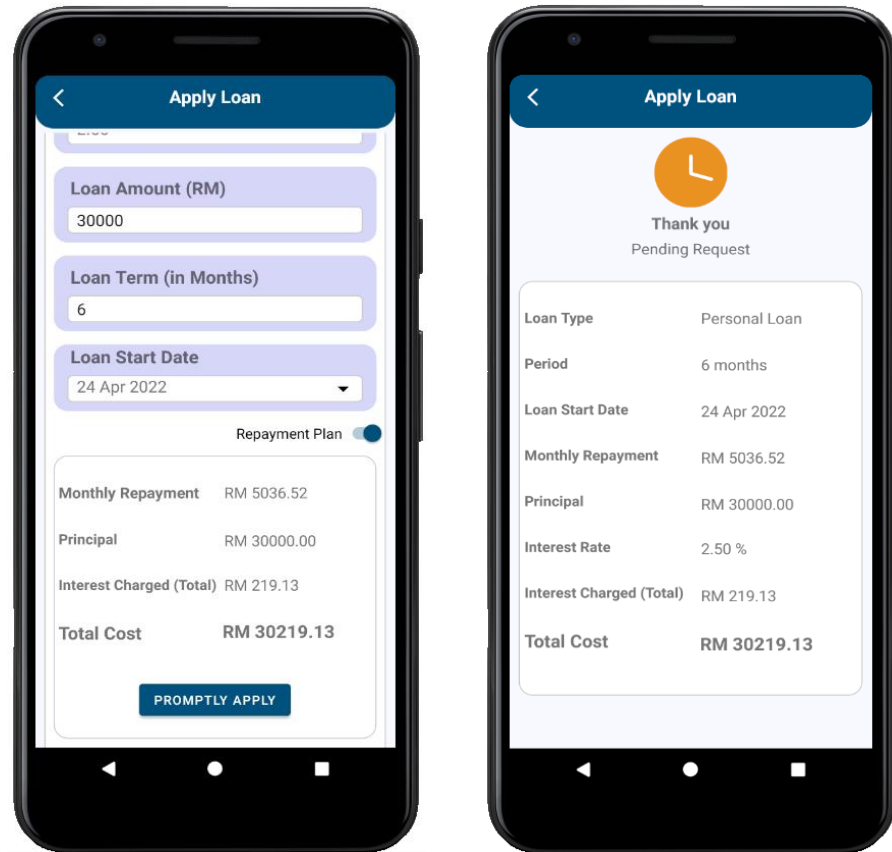
**Figure 5.4.43** Apply Loan Page

Before the user applied any loan application yet, the Loan Page will display a poster to let the user have an idea how many days the loan application taken to process (*refer to Figure 5.4.42*). So, the user can click the apply loan button to apply a loan, the system will then redirect the user to the Apply Loan Page (*refer to Figure 5.4.43*).



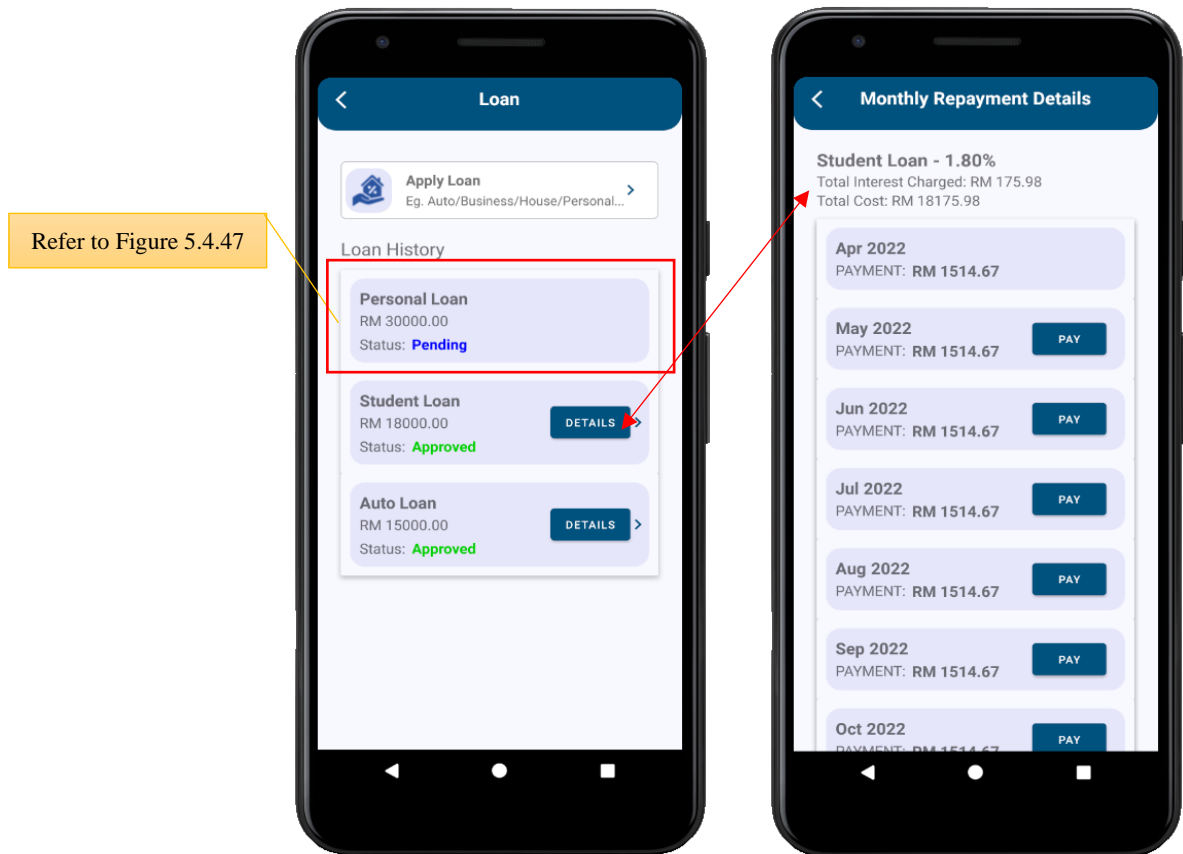
**Figure 5.4.44** Selection of Loan Type **Figure 5.4.45** Filled out Loan Application

Banky offers few types of loans for Banky users such as personal loan with 2.5% interest rate, auto loan with 2.3% interest rate, business loan with 2.8% interest rate, house loan with 2.3% interest rate, life insurance loan with 3.5% interest rate, mortgage loan with 2.1% interest rate and student loan with 1.8% interest rate, different loan types will have different loan interest rates (*refer to Figure 5.4.44*). The user must fill in all the required details before clicking on the repayment plan switch to view the repayment plan, else the system will toast a message displaying which field is empty. The reason to do so because the system wants to force the user to clear about the repayment plan first before applying a loan.



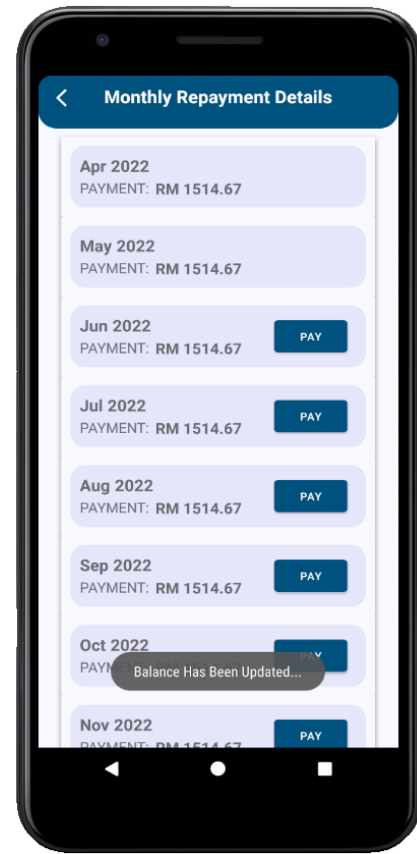
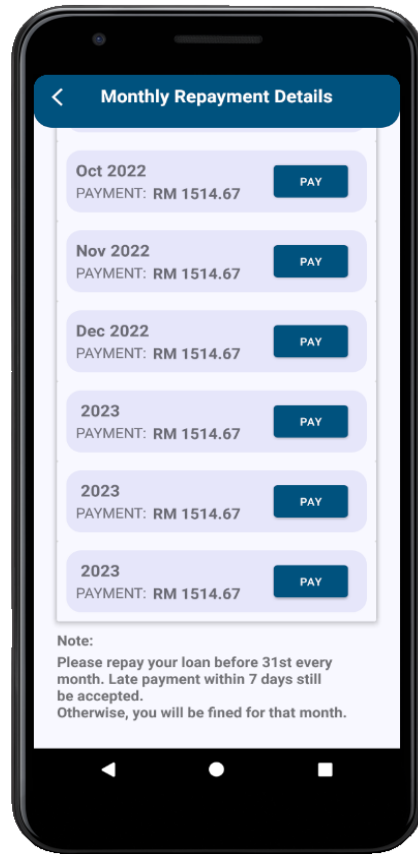
**Figure 5.4.46** Repayment Plan    **Figure 5.4.47** Loan Application Pending

When the user turns on the repayment plan switch, he/she is able to view the monthly repayment amount, total interest charged and total cost to pay back in the end (*refer to Figure 5.4.46*). After the user clicks on the promptly apply button, the system will show a pending loan application (*refer to Figure 5.4.47*). All the loan application details will be update to both the user and the bank manager in Firebase Realtime Database. So that, the bank manager is only able to update the loan status.



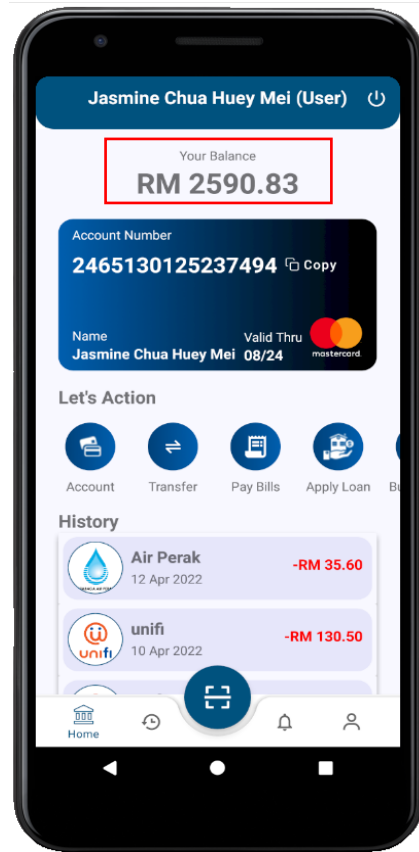
**Figure 5.4.48** Loan Page Updated **Figure 5.4.49** Monthly Repayment Details

Once the loan apply process is done, it will update the loan history with displaying the loan name, loan amount and loan status (*refer to Figure 5.4.48*), so that the user can come here to check the loan status whether it is still pending, or has been approved or rejected. Only that loan status with approved can showed the monthly repayment details(*refer to Figure 5.4.48*). Let's take the example of student loan that has been approved, so when the user clicks the details button, the user is redirected to the monthly repayment details page (*refer to Figure 5.4.49*).



**Figure 5.4.50** Monthly Repayment Details2 **Figure 5.4.51** Repay The Loan

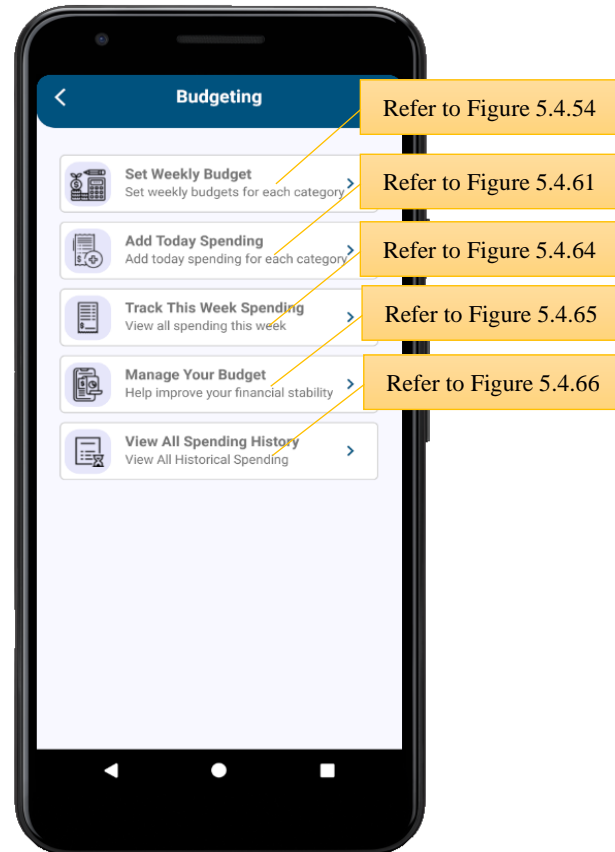
It will display a short note text for the user to let them remember to repay the loan every months, else the user will be fined for that month. Assuming that the current month is May, and the user wants to repay the loan, he/she can click the pay button for the that month, so the system needs to check whether the available balance enough to pay the loan repayment. If the available balance is greater than or equal to the repayment amount, the pay button disappears to indicate that the user has paid for the month.



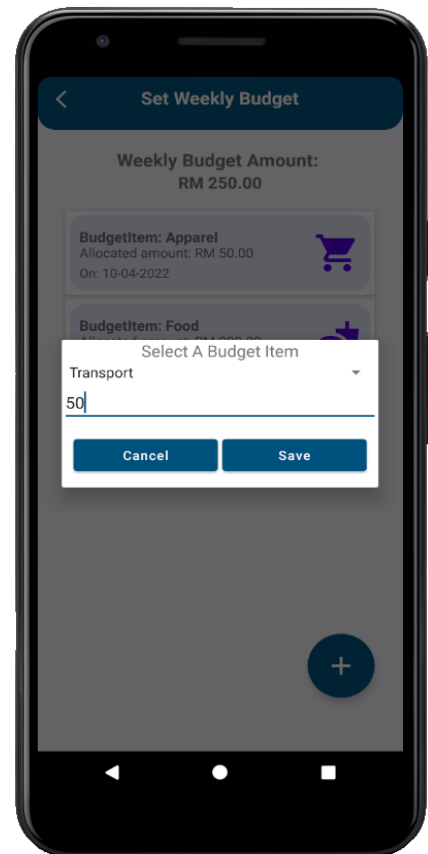
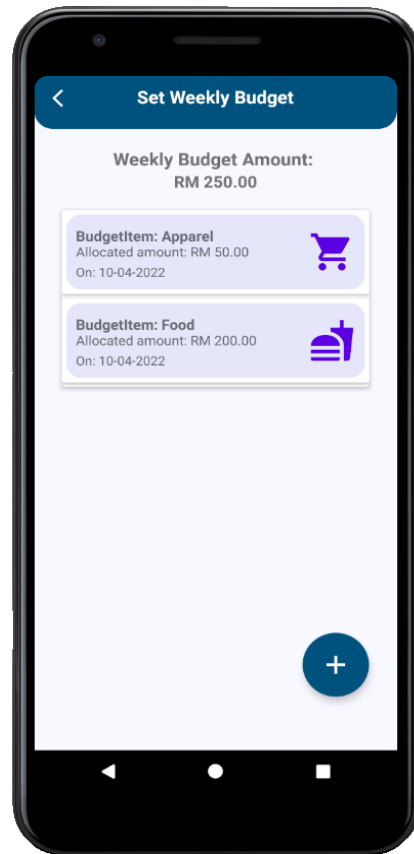
**Figure 5.4.52** Main Page-New Balance4 **Figure 5.4.53** Budgeting Page

When the loan repayment for the specific month is completed, the available balance is also updated, the available balance was RM4105.50 before (*refer to Figure 5.4.41*), and now becomes RM2590.83 (*refer to Figure 5.4.52*) which deducted RM1514.67 from *Figure 5.4.51*.

In the budgeting page, the user is able to set weekly budget, add today spending, track this week spending, manage the budget, and view all spending history.

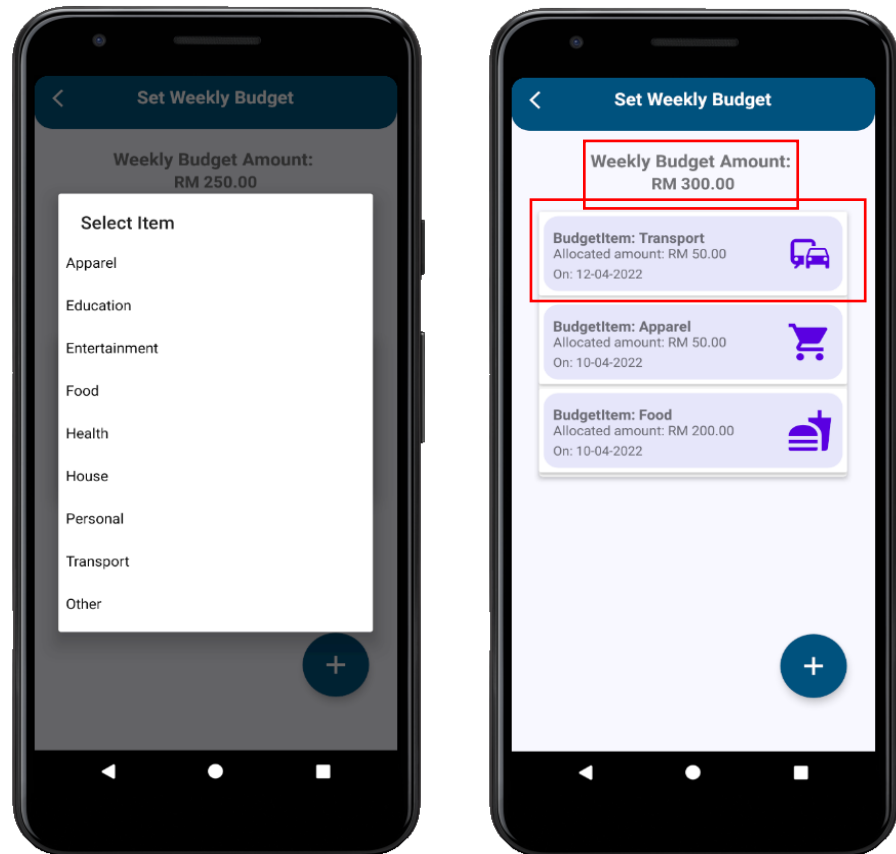




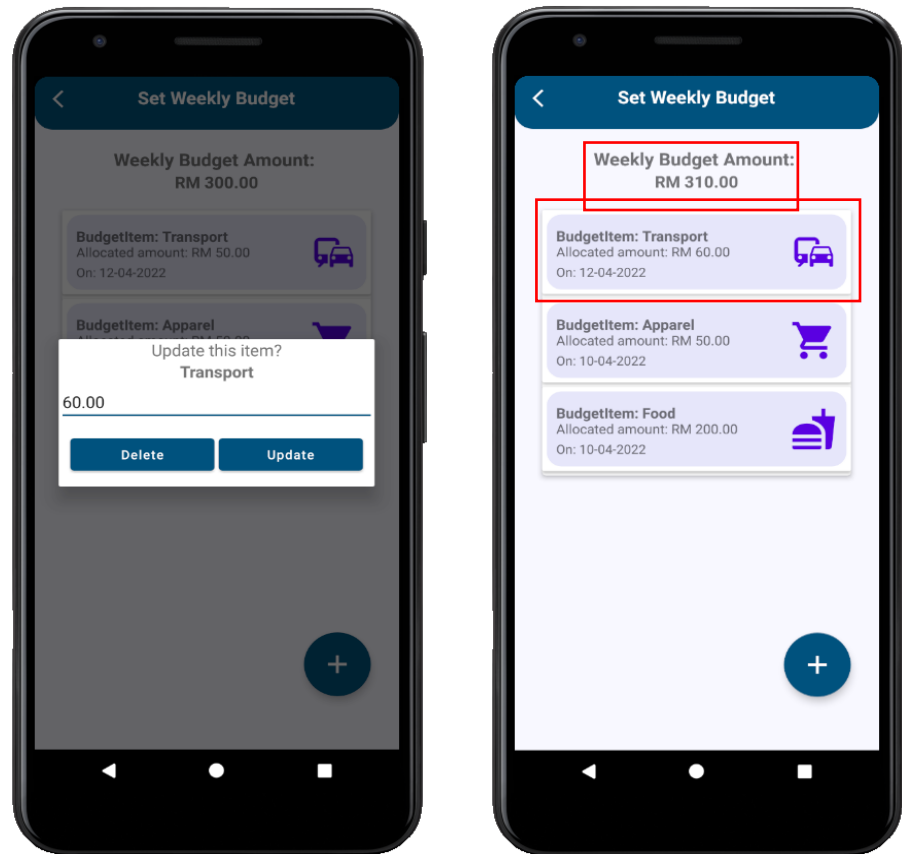


**Figure 5.4.54** Set Weekly Budget    **Figure 5.4.55** Add Weekly Budget

It will display the budget item and its weekly budget amount that has set by the user before (*refer to Figure 5.4.54*). If there does not have any budget item or the user wants to add a new budget item, he/she can click the add icon floating button, so that the system will prompt a dialog box to let the user chooses the item category and enters the budget item for that item. After the user clicks the save button, it only will save the budget item to Firebase Realtime Database.

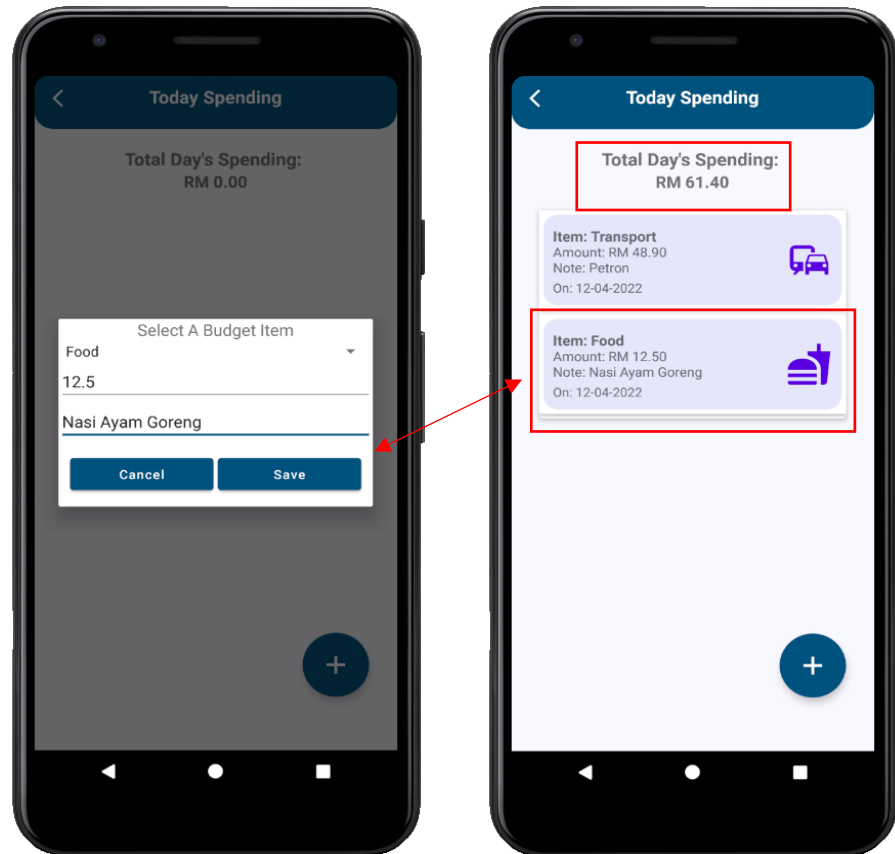


**Figure 5.4.56** Selection of Item Category **Figure 5.4.57** Updated Weekly Budget  
*Figure 5.4.56* shows the item categories that the user can select when the user clicks the drop-down button in *Figure 5.4.55*. Once the user has successfully added a new budget item, it will be displayed there, and it will simultaneously update the total weekly budget amount displayed there as well (*refer to Figure 5.4.57*).



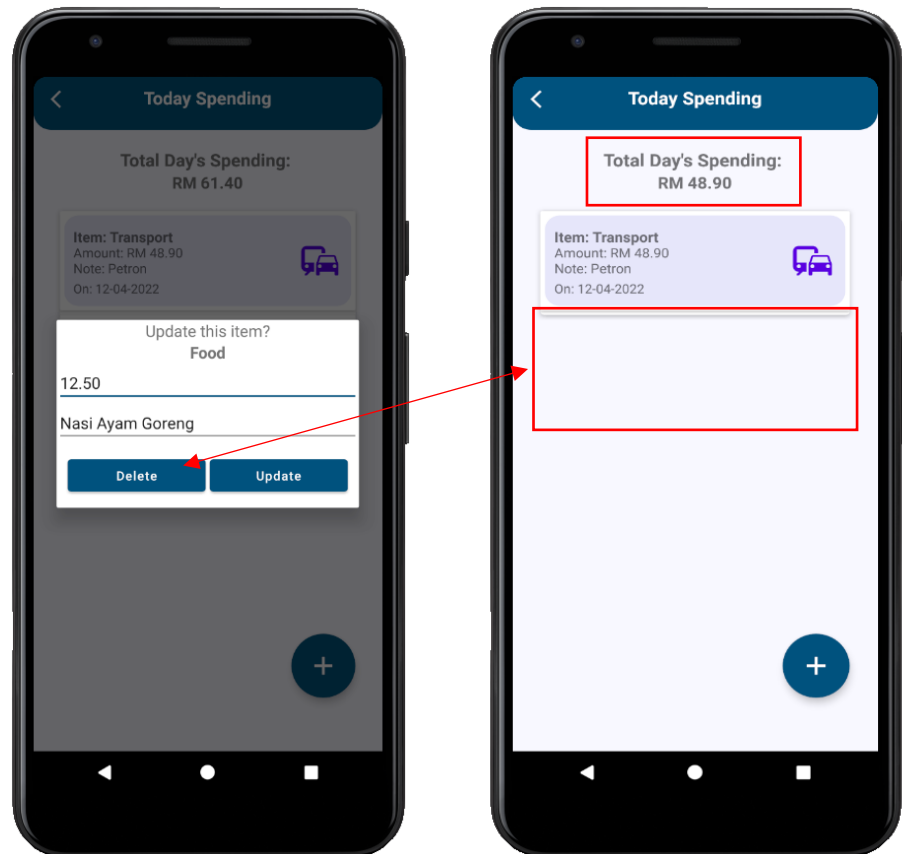
**Figure 5.4.58** Update/Delete Budget Item **Figure 5.4.59** Updated Weekly Budget2

When the set budget button is clicked from the Budgeting Page (refer to Figure 5.4.53), the system will redirect the user to the Set Weekly Budget Page (refer to Figure 5.4.58). The user is able to update or delete any budget item over there. For example, if the user wants to update the transportation budget amount, just click on the transportation, and a dialog box will pop up for the user to edit the transportation budget amount. Once the user clicks the update button in Figure 5.4.58, it will be updated along with the total weekly budget amount. If the user wants to delete a budget item from the list, just click the delete button in Figure 5.4.58, the item will be removed from the list, and the weekly budget amount will also be deducted the item's budget amount (refer to Figure 5.4.59).



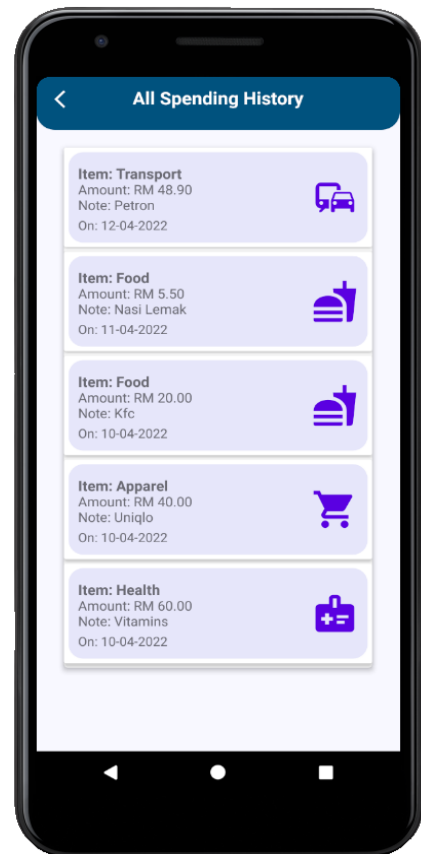
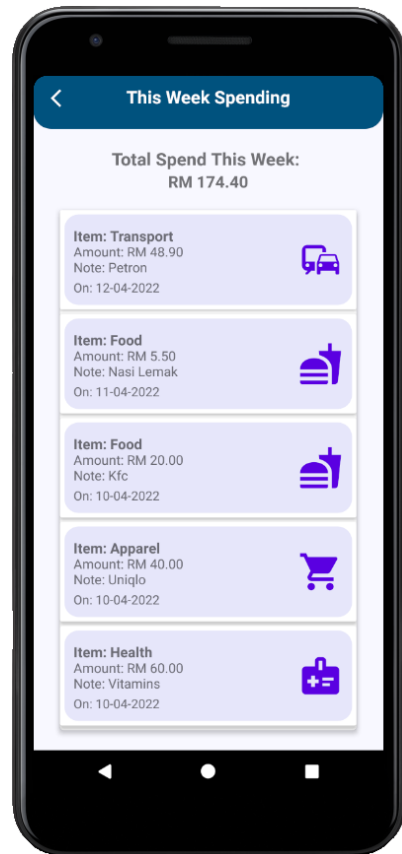
**Figure 5.4.60** Add Today Spending **Figure 5.4.61** Today Spending Page

When the add today spending button is clicked from the Budgeting Page (refer to Figure 5.4.53), the system will redirect the user to the Add Today Spending Page (refer to Figure 5.4.60). If the user does not have any spending today or wants to add on a new spending today, he/she can click the add icon floating button, and the system will pop up a dialog box as shown in Figure 5.4.60 to allow the user to select an item category from the spinner (refer to Figure 5.4.56). When the user clicks the save button, the details such as the item category, amount and note are added to the Firebase Realtime Database. After that, it will show up in the list and update the total spend for today.



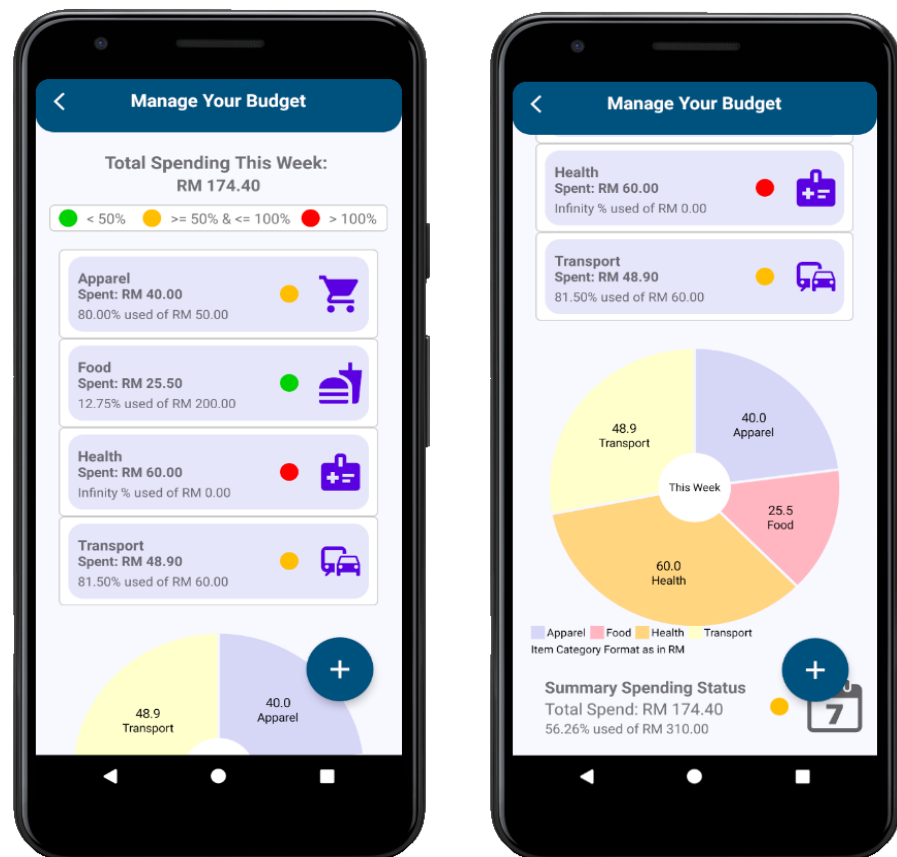
**Figure 5.4.62** Update/Delete Spending Item **Figure 5.4.63** Updated Today Spending2

The today spending page is also same as the set weekly budget item that it allows the user to update or delete the item in the list. When the user deleted the food item spending today, it will delete and update the today spending item in Firebase Realtime Database. At the same time, it will be removed from the list and be updated the total of today's spending.



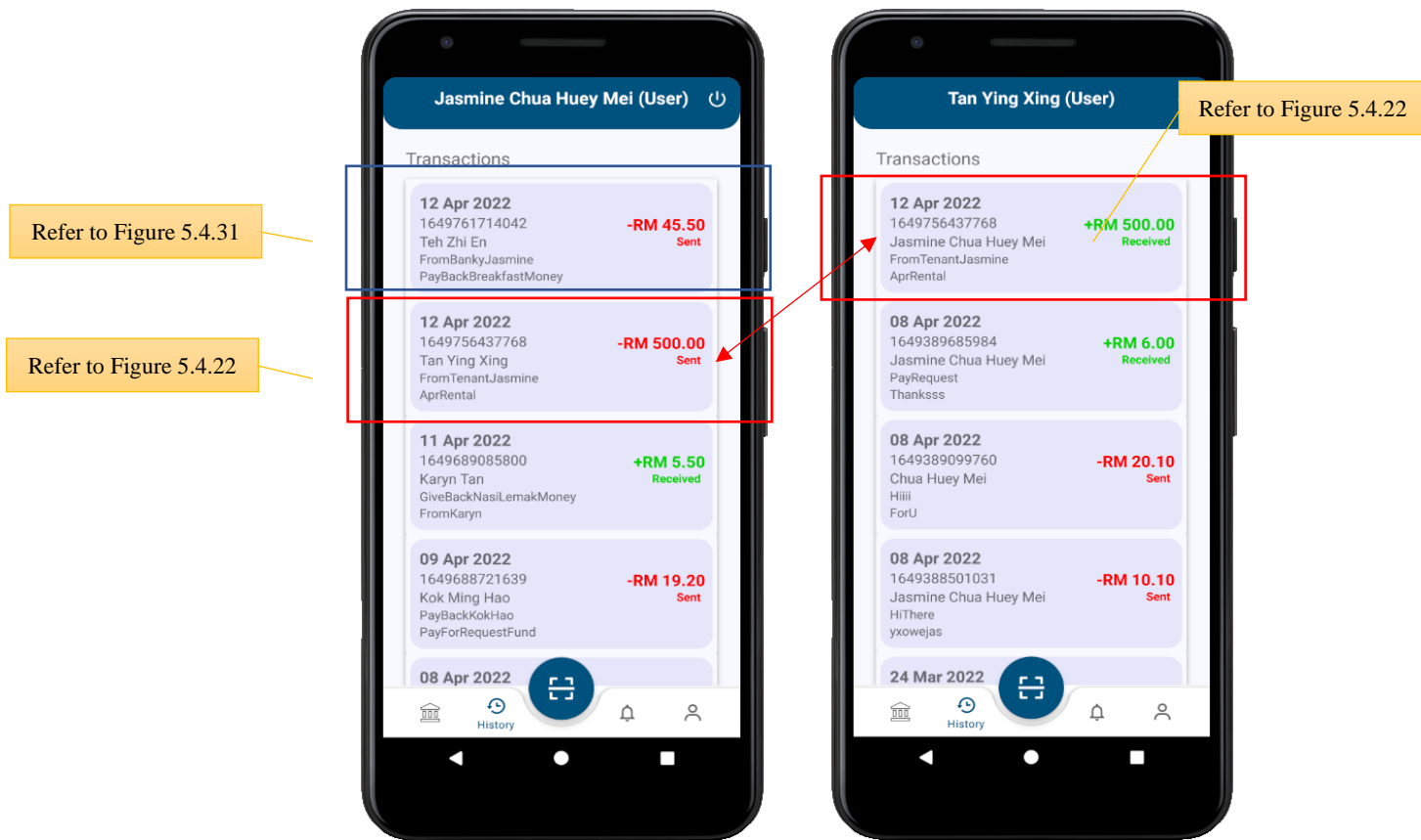
**Figure 5.4.64** This Week Spending      **Figure 5.4.65** All Spending History

When the track this week spending button is clicked from the Budgeting Page (refer to Figure 5.4.53), the system will redirect the user to the Track This Week Spending Page (refer to Figure 5.4.64). Figure 5.4.64 will display all the spending item that happened within this week and also the total amount spent this week. If the user clicks the View All Spending History Page from the Budgeting Page (refer to Figure 5.4.53), the user is redirected to the View All Spending History (refer to Figure 5.4.65). So basically, Figure 5.4.65 will display all the spending history.



**Figure 5.4.66** Manage Your Budget **Figure 5.4.67** Manage Your Budget (cont)

When the manage your budget button is clicked from the Budgeting Page (refer to Figure 5.4.53), the system will redirect the user to the Manage Your Budget Page (refer to Figure 5.4.66 & Figure 5.4.67). All the spending item are grouped together, and it will use the total spending amount for an item to compare with the item's budget amount. So, if the percentage is lower than 50%, it will display a green circle, which indicating it is still healthy. Otherwise, if the spending percentage versus budget amount is greater than or equal to 50% but less than or equal to 100%, it will display a yellow circle to indicate a warning. Else, if the percentage is greater than 100% or totally the user does not set the budget amount for the item, it will show a red circle that indicates the user has exceeded the budget amount. It will put all the spending amount for each category into the pie chart to display for the user to have a look. In the end, it will also display the summary spending status for this week. The user can rotate and even click on the pie chart, so the system can toast a message showing the item selected with its amount. If the user clicks on the add icon floating button, it will redirect the user to the Set Weekly Budget Page to let the user sets a weekly budget.



**Figure 5.4.68** History Transactions (Current User side) **Figure 5.4.69** History Transactions (Beneficiary side)

Once the fund transfer has been successfully transferred to the beneficiary within Banky (refer to Figure 5.4.22), it will update the history transactions at both the current user and beneficiary side. Besides that, the transaction of transferring fund to other bank (refer to Figure 5.4.31) will be also updated and displayed it over the history transactions there. In the history trasactions there, if the fund is transferred by the other, then it will be showing the amout with green color text along with the “received” word, else if it is the current user transfers the fund to other, then it will be showing the amount with red color text along with the “sent” word.





**Figure 5.4.70** QR Scan

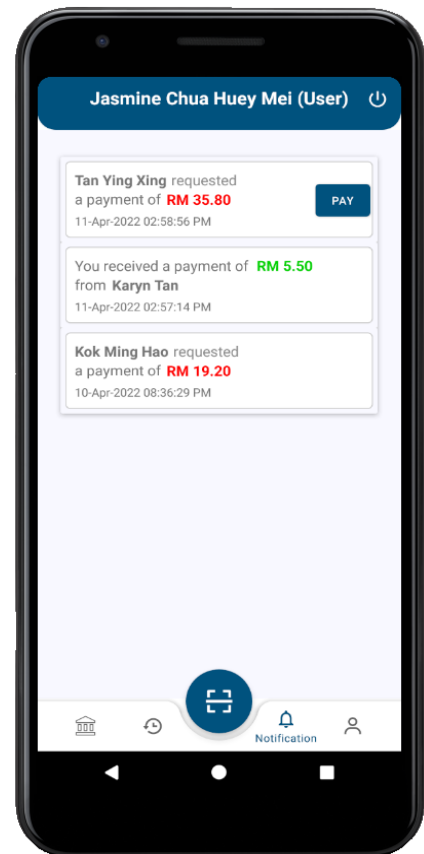


**Figure 5.4.71** QR Receive

When clicking the scan icon on the Main Page under the bottom navigation bar (*refer to Figure 5.4.10*), the system will redirect the user to the QR code scanning Page (*refer to Figure 5.4.70*). can scan the QR code for payment, or scan the QR code of other users for remittance. If the user clicks on the “Receive” tab, it will display a unique QR code which encoded by using the username and account number, which the user can show this QR code to his/her family or friends to receive the money (*refer to Figure 5.4.71*).



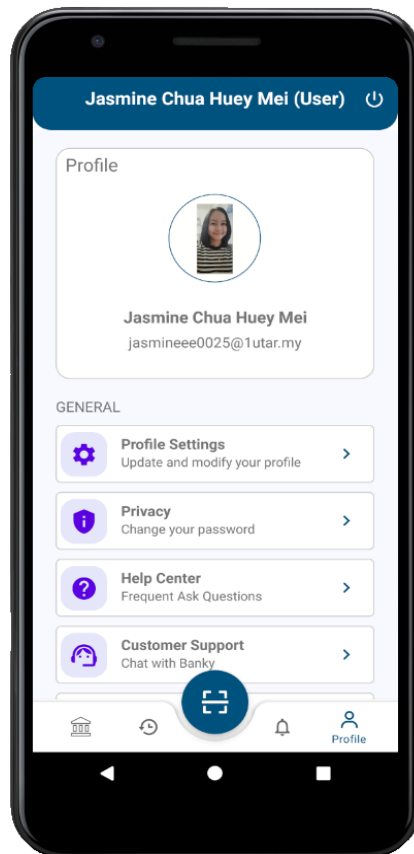
**Figure 5.4.72** QR Pay



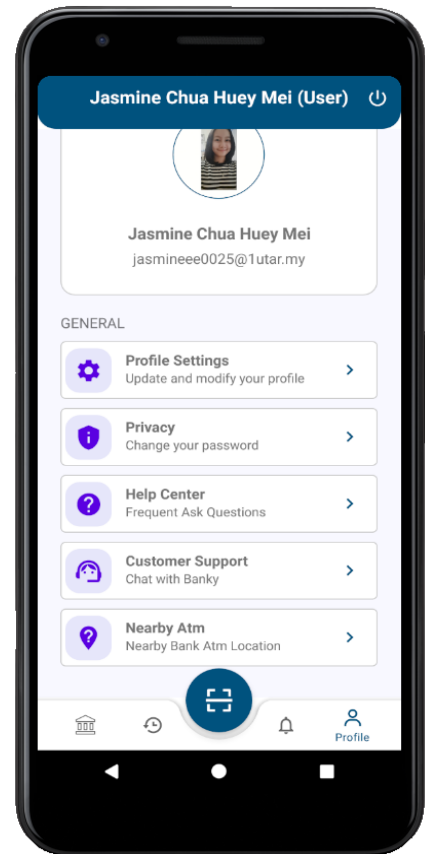
**Figure 5.4.73** Notification Page

Other than QR Scan and QR Receive (refer to *Figure 5.4.70* & *Figure 5.4.71*), there is also a QR Pay (refer to *Figure 5.4.72*) which the user can show this QR code or barcode to the counter or the cashier to make payment.

When the user clicks the Notification icon in the Main Page below the bottom navigation bar, it redirects the user to the Notification Page which will all requested funds history and received funds transactions. If the user did pay for the request, then the pay button will disappear. The user can click on the pay button, it will redirect the user to the Pay For Request Funds Page. The Pay For Request Funds Page is exactly same layout like *Figure 5.4.17*, just that the user not needs to enter the beneficiary's account number and amount to transfer back.



**Figure 5.4.74** Profile Page

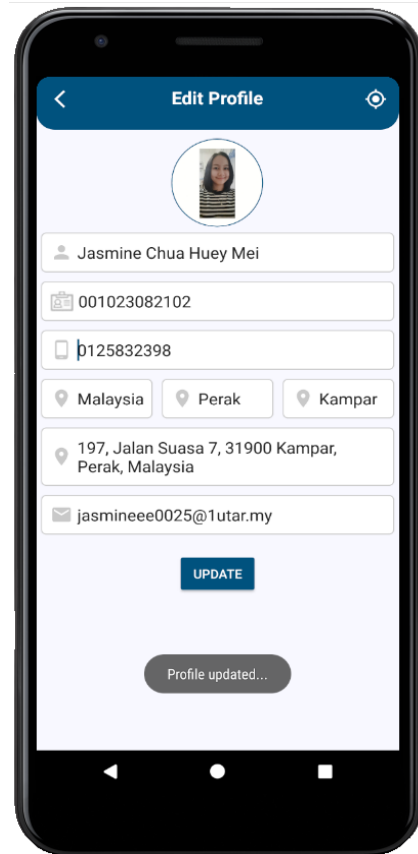


**Figure 5.4.75** Profile Page (cont)

When the user clicks on the profile in the bottom navigational bar, it will redirect the user to *Figure 5.4.74* and *Figure 5.4.75*. The user is able to see his/her profile picture, name and email address. The user can click on the Profile Settings to update or modify his/her personal details. The user can press the Privacy to change his/her password. The user can touch the Help Center to check for the questions they might want to ask because there will display all the questions that are frequently asked by users along with its supported answers. If Help Center still cannot solve the user's problem or the question they want to ask is not over there then he/she can consider to click on the Customer Support in order to let customer support person answers their question either by email, phone call or WhatsApp. The user can check out the location of the nearby ATMs to perform cardless withdrawal by which the user needs to click on the Nearby ATM.

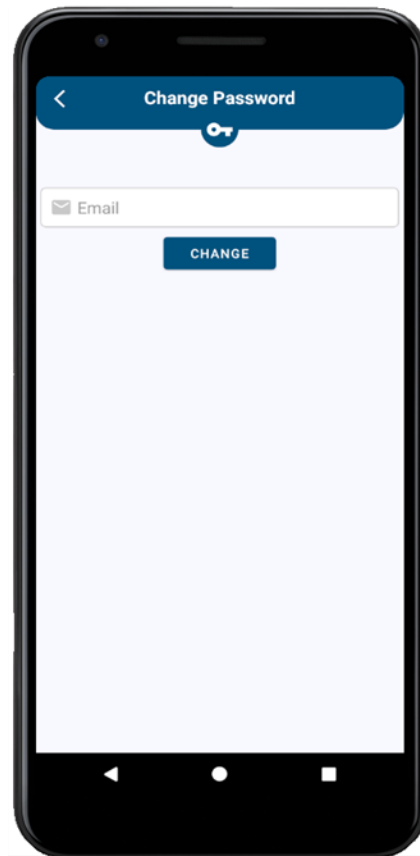


**Figure 5.4.76** Profile Settings

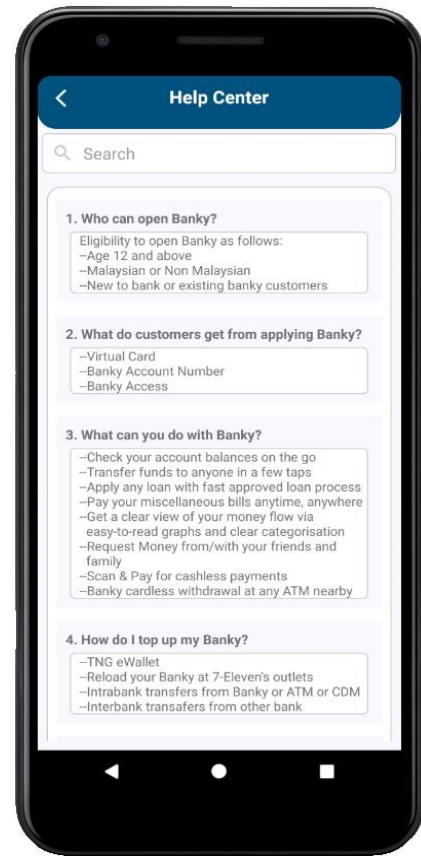


**Figure 5.4.77** Edit Profile

When the user clicks on the Profile Settings from the *Figure 5.4.74*, then the system will redirect the user to *Figure 5.4.76* to let them have a look of their personal details, if they really wants to update or modify it, then they can click on the pencil icon. Then it will redirect the user to *Figure 5.4.77* to edit the personal details. Once the user clicks on the “Update” button, then it will toast a message to the user saying that “Profile updated...”.



**Figure 5.4.78** Privacy Page



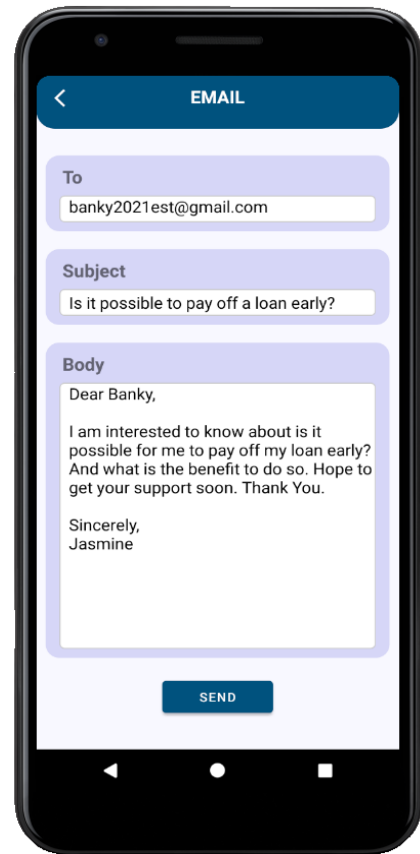
**Figure 5.4.79** Help Center

When the user clicks on the Privacy from the Profile Page (*refer to Figure 5.4.74*), then the system will redirect the user to the Privacy Page (*refer to Figure 5.4.78*) to let the user changes his/her password, it just same like what *Figure 4.6* will do. The user will need to key in their email address, then the system will a reset password instruction email to him/her, after that the user can change his/her password by clicking the link provided in the email.

For new users, it sure that they have a lot of things interested to know more about Banky, then the user can click the Help Center button from the Profile Page (*refer to Figure 5.4.74*), the user will be redirected to the Help Center Page (*refer to Figure 5.4.79*).

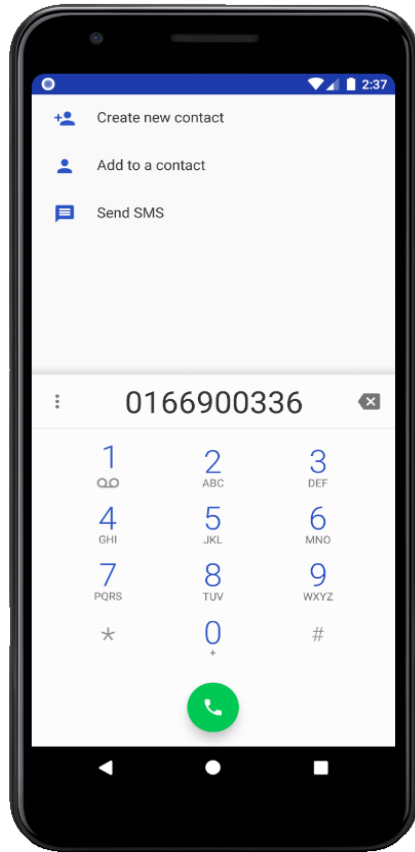


**Figure 5.4.80** Contact Us Page



**Figure 5.4.81** Email Banky

When clicking the customer support button in the Profile Page (*refer to Figure 5.4.74*), the system redirects the user to the Contact Us Page (*refer to Figure 5.4.80*), so the user can choose a channel that they wants to contact Banky more comfortably maybe because the question they wish to ask is not showed over the Help Center there. If the user clicks on the refer FAQ button, it will redirect the user to the Help Center Page (*refer to Figure 5.4.79*). For example, the user wants to send an email to Banky, so the user needs to provide the email subject and enter something in the email body. After the user clicks the send button, it will direct open the user's default mailbox in the phone, so the user can continue to send the email.

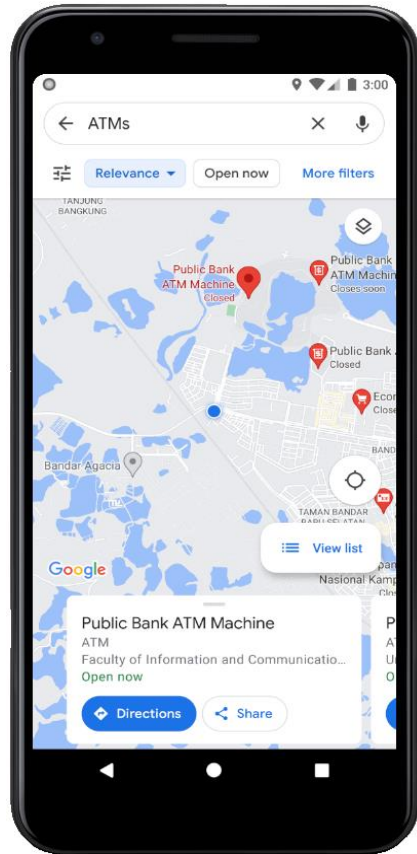


**Figure 5.4.82** Phone Call Banky



**Figure 5.4.83** WhatsApp Banky

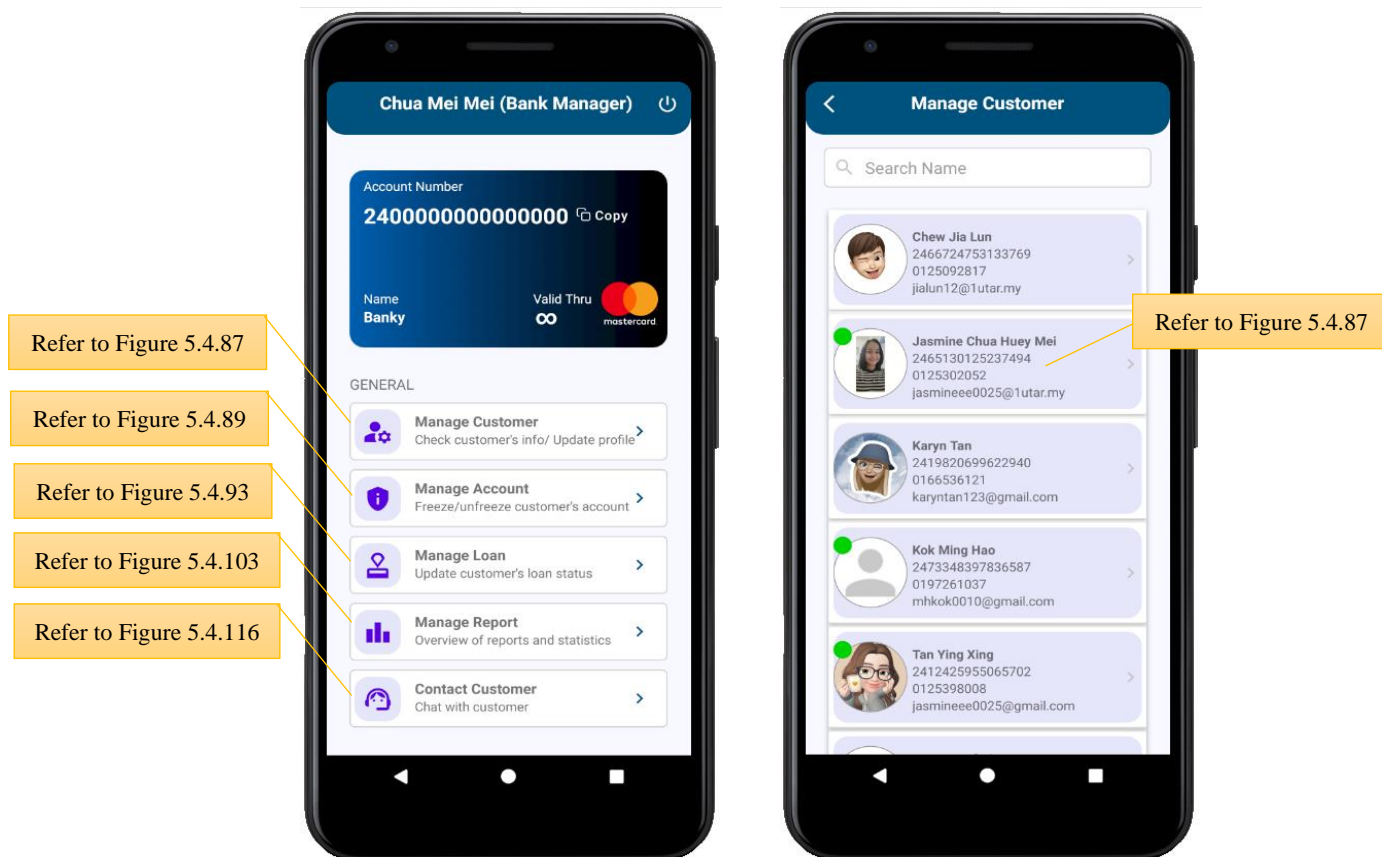
If there is a very urgent situation, such as the user detects his/her money has been withdrawn from his/her account, but he/she does not do the action, then the user should contact Banky immediately and inform them by phone call or WhatsApp to Banky. So that Banky only can take action to protect the user's bank account.



**Figure 5.4.84** Nearby ATM

When clicking the nearby ATM button in the Profile Page (*refer to 5.4.74*), the system will directly call the Google Maps Intents for Android and open up the Google Maps in the phone, search for the “ATMs” keyword, and display the nearby ATM locations using the user’s latitude and longitude.

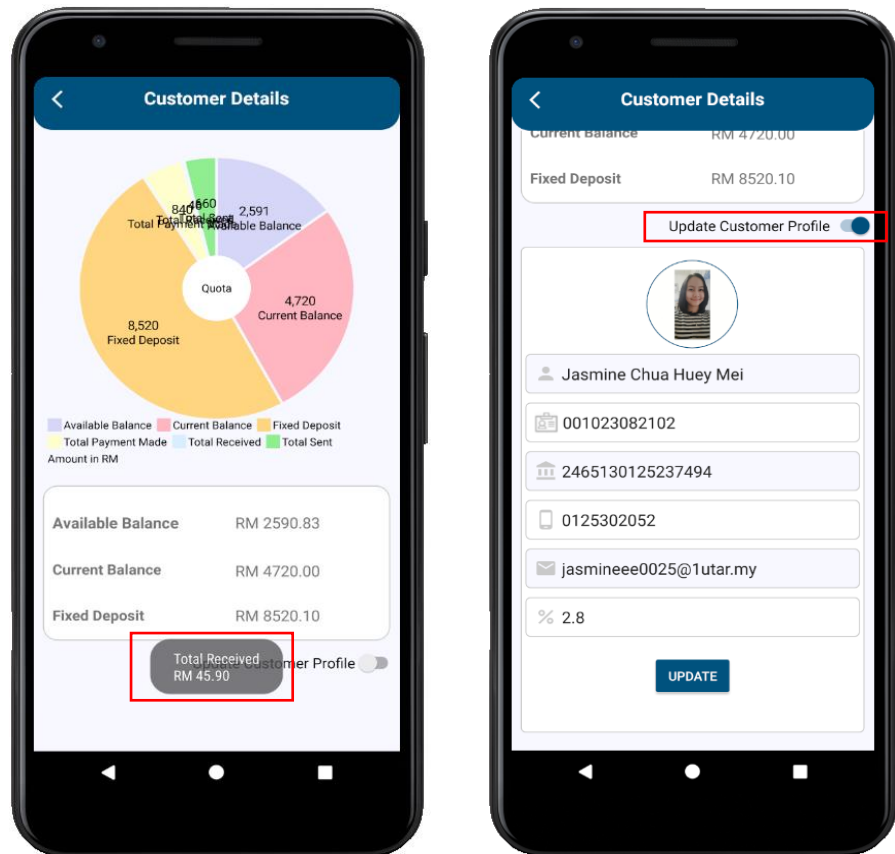




**Figure 5.4.85** Bank Manager Main Page **Figure 5.4.86** Manage Customer Page

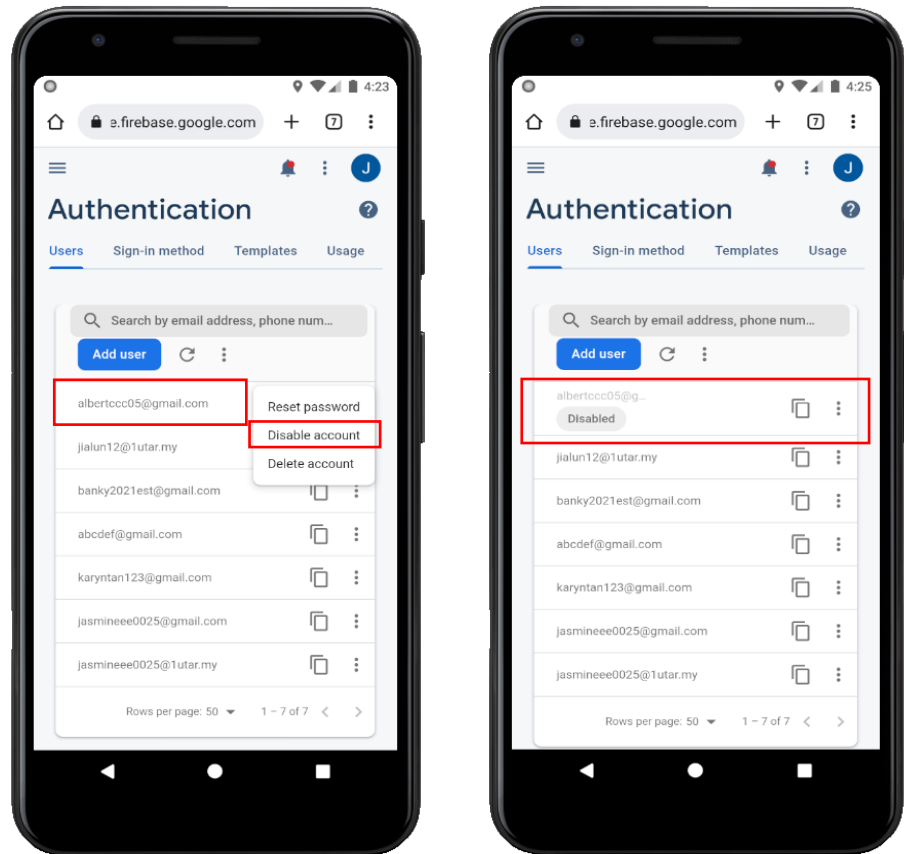
The toolbar will display the username with the job position that typed by the bank staff when the time he/she registers as a Banky staff. The account number for all the Banky staff is 2400000000000000. Banky provides few functionalities for the bank manager, such as manage customer, manage account, manage loan, manage report, and contact customer. The Manage Customer is to have a look of the customer's balances and it can be used to update the customer profile as well. The Manage Account is to freeze or unfreeze the customer's account whenever detect some fraud happening. The Manage Loan is to update the loan application status to either pending, approved or rejected. The Contact Customer is used to contact the customer either using phone call or email to them.

When clicking the manage customer button in the Bank Manager Main Page (*refer to Figure 5.4.85*), the bank manager will be redirected to the Manage Customer Page (*refer to Figure 5.4.86*). The bank can click either one customer to check their balances and profile. The bank manager can utilize the search function to search by customer name. The green circle next to the profile image of each users is to indicate that the user is active now.



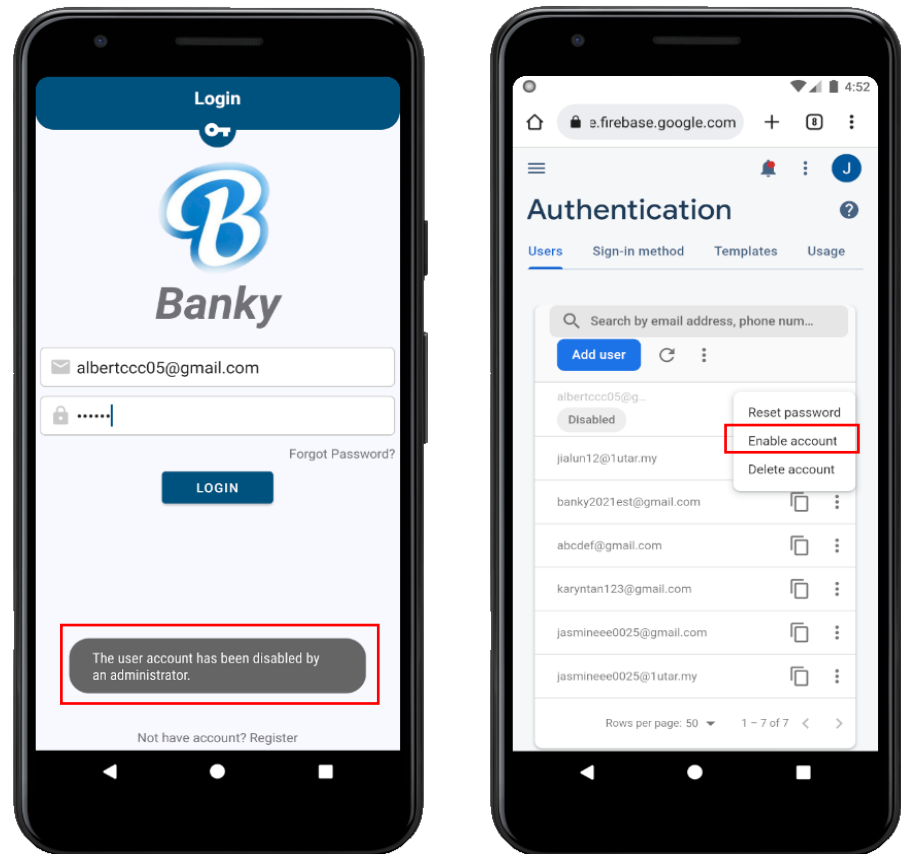
**Figure 5.4.87** Customer Details      **Figure 5.4.88** Update Customer Profile

For example, the bank manager clicks on the customer called Jasmine Chua Huey Mei from the Manage Customer Page (refer to Figure 5.4.86), then the system will display the available balance, current balance, fixed deposit balance, total payment made, total received and total sent from this customer as in pie chart form (refer to Figure 5.4.87). When the pie chart entry is too small for the bank manager to see, the bank manager can click on any field shown in the pie chart, so the system will toast a message to the bank manager telling the bank manager the name of the entry and its amount, for example, the bank manager cannot see clearly how much is the total received, so he/she just clicks on the blue color gap in the pie chart, the system will toast a message displaying the details (refer to Figure 5.4.87) and even the pie chart can be rotated. If the bank manager wishes to edit and update this customer's profile or even annual interest rate, just switch on the Update Customer Profile to make the changes. Once the bank manager clicks the update button, the updated details will be updated to Firebase Realtime Database as well.



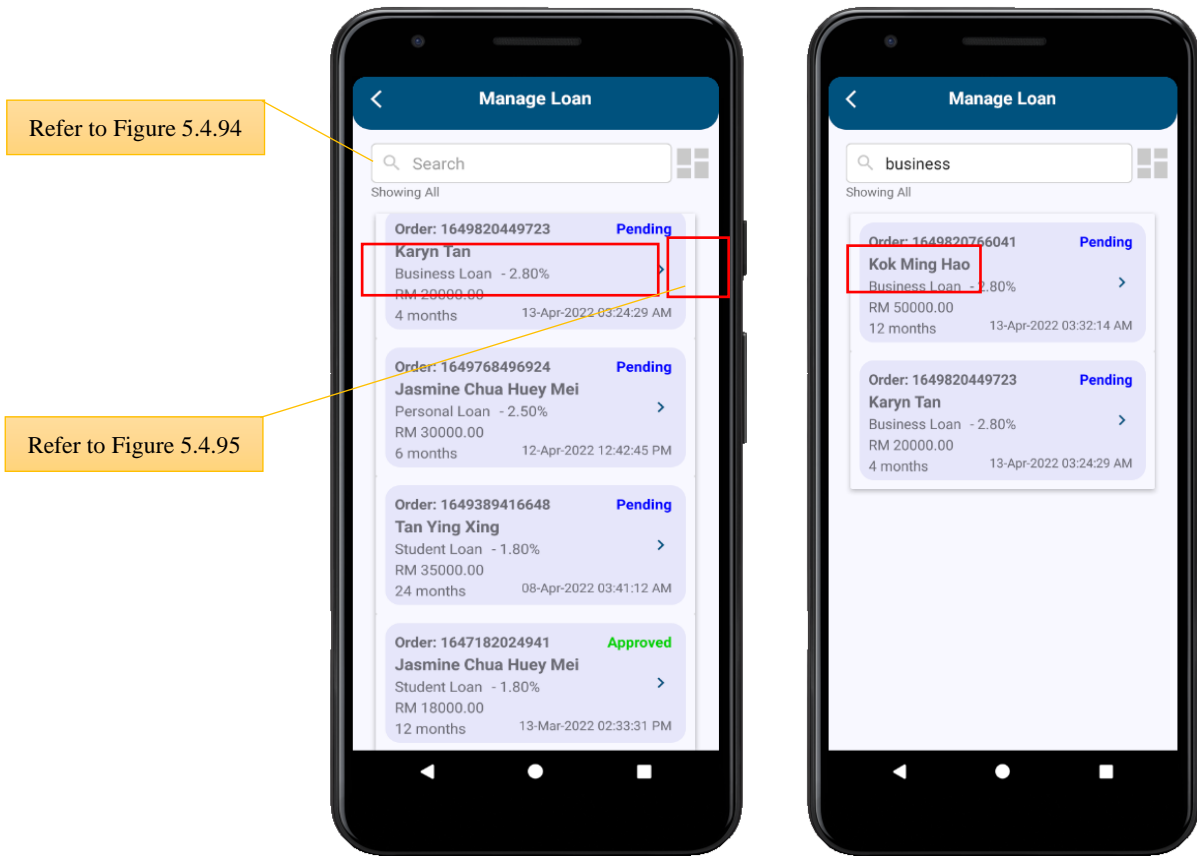
**Figure 5.4.89** Manage Account **Figure 5.4.90** Disabled Customer Account

When the manage account button is clicked on the Bank Manager Main Page (refer to Figure 5.4.58), the system redirects the bank manager to Banky Firebase Authentication webpage. The bank manager can select the customer whose account that the customer has requested to freeze, by clicking the three dots next to that customer, and then clicking the “Disable account” (refer to Figure 5.4.89).



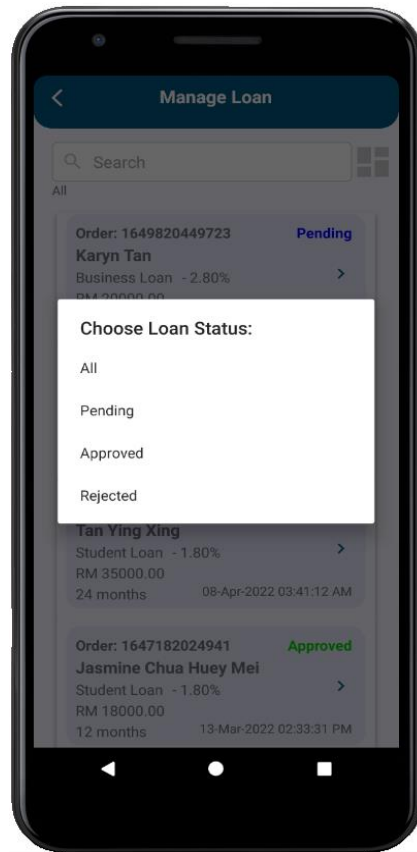
**Figure 5.4.91** Disabled Acc Cannot Login **Figure 5.4.92** Enabled Back Customer

Therefore, when a customer’s account is disabled (*refer to Figure 5.4.90*), that customer cannot log into Banky application (*refer to Figure 5.4.91*) until the bank manager enables the customer again by clicking the “Enable account” shown in *Figure 5.4.92*, then the customer only can log into Banky. Even if a customer cannot log into the Banky app to use its features due to the account is disabled, it does not restrict others from transferring money or requesting funds to that customer.

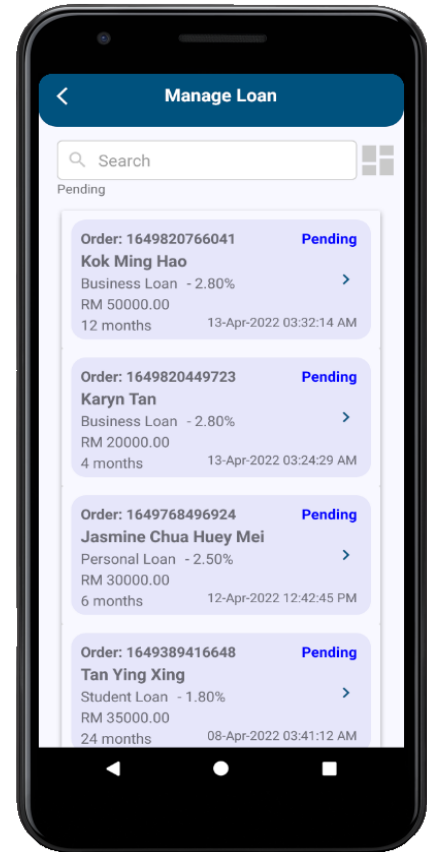


**Figure 5.4.93** Manage Loan      **Figure 5.4.94** Search by Loan Type

When clicking the Manage Loan in the Bank Manager Main Page (*refer to Figure 5.4.85*), the system redirects the bank manager to Manage Loan Page and displays all the loan applications (*refer to Figure 5.4.93*). If let's say the bank manager just wants to display all the business loan application, he/she can utilize the search function to search by the loan type (*refer to Figure 5.4.94*).

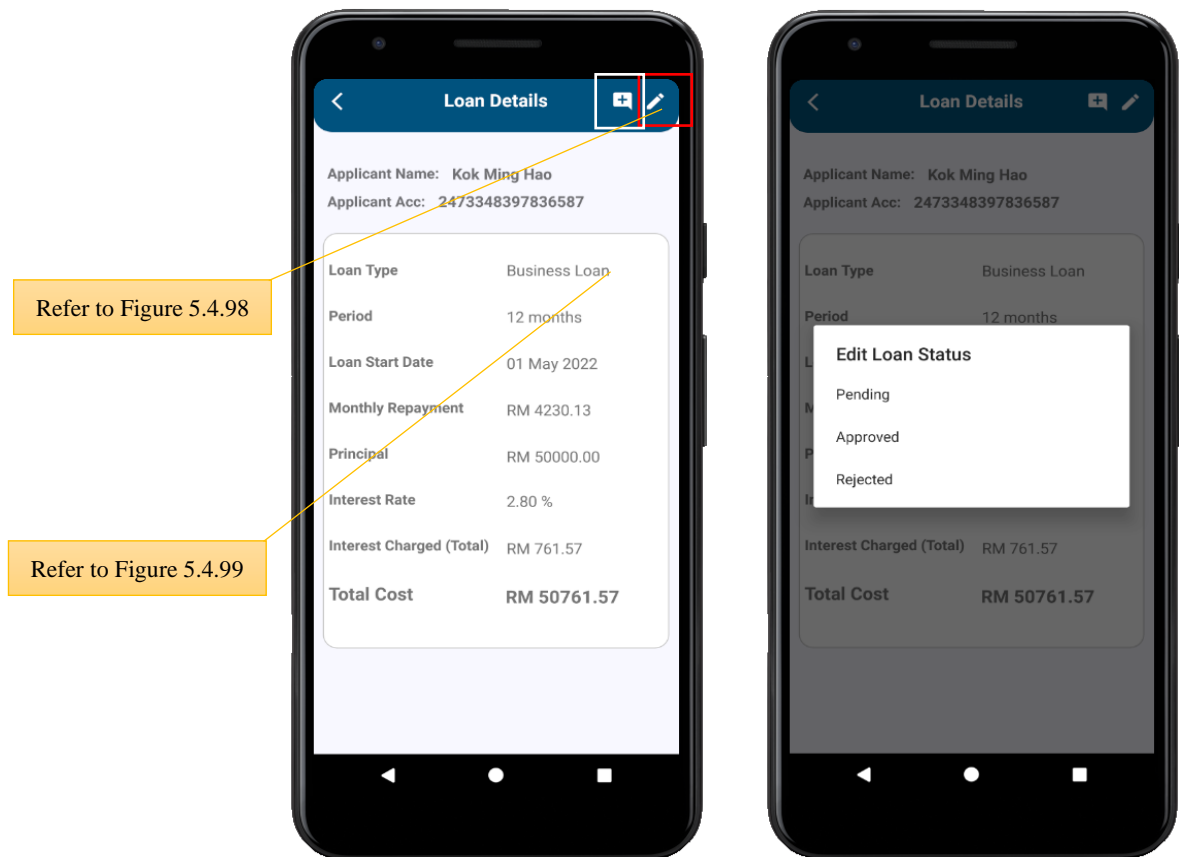


**Figure 5.4.95** Filter by Loan Status



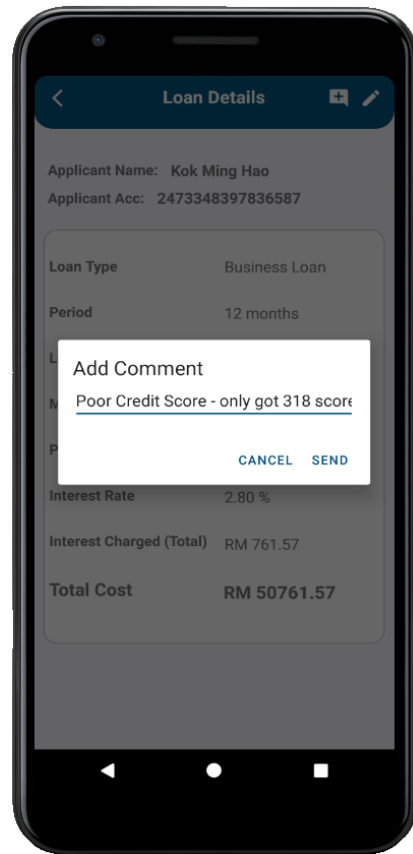
**Figure 5.4.96** Filter Result

Assuming that the bank manager only wants to view all pending loan applications and not all loan applications, he/she can click the filter button displayed in the upper right corner of the Manage Loans page (refer to Figure 5.4.93). So the system will pop up a dialog box to let the bank manager select the loan status that he/she wants, for example pending. The system will then display the results of loan applications based on the bank manager's selection (refer to Figure 5.4.96).

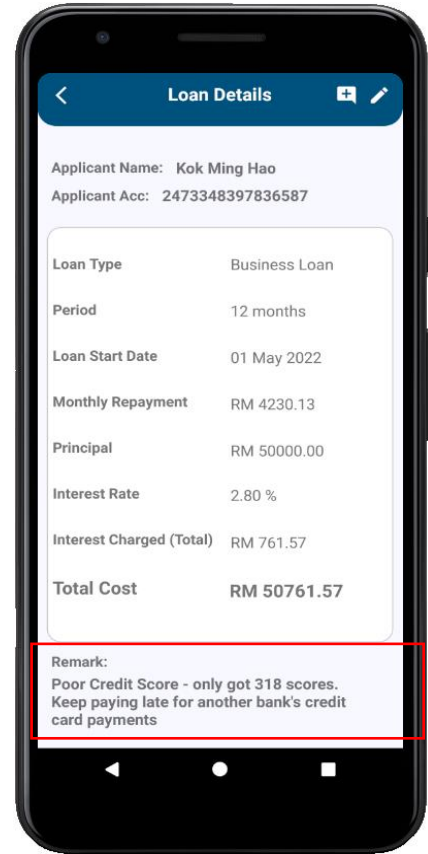


**Figure 5.4.97** Loan Application Details    **Figure 5.4.98** Edit Loan Status

The bank manager can click either one of the loan applications shown in the Manage Loan Page (*refer to Figure 5.4.93*) to check the loan application details and even to update the loan status or leave a remark. For example, if let's say the bank manager has checked that this customer has a poor credit score after looking at the CTOS website's credit score and can see that this customer is not creditworthiness because this customer keeps paying late for another bank's credit card payment. So the bank manager will reject his loan application in this case.



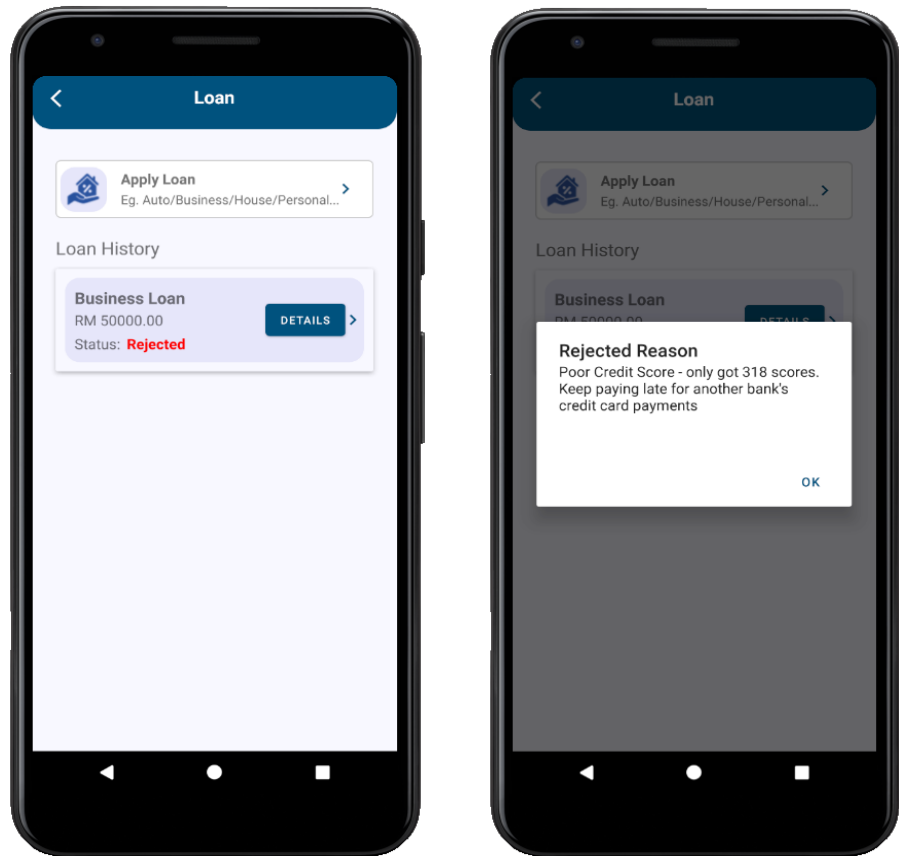
**Figure 5.4.99** Add Remark



**Figure 5.4.100** Remark updated

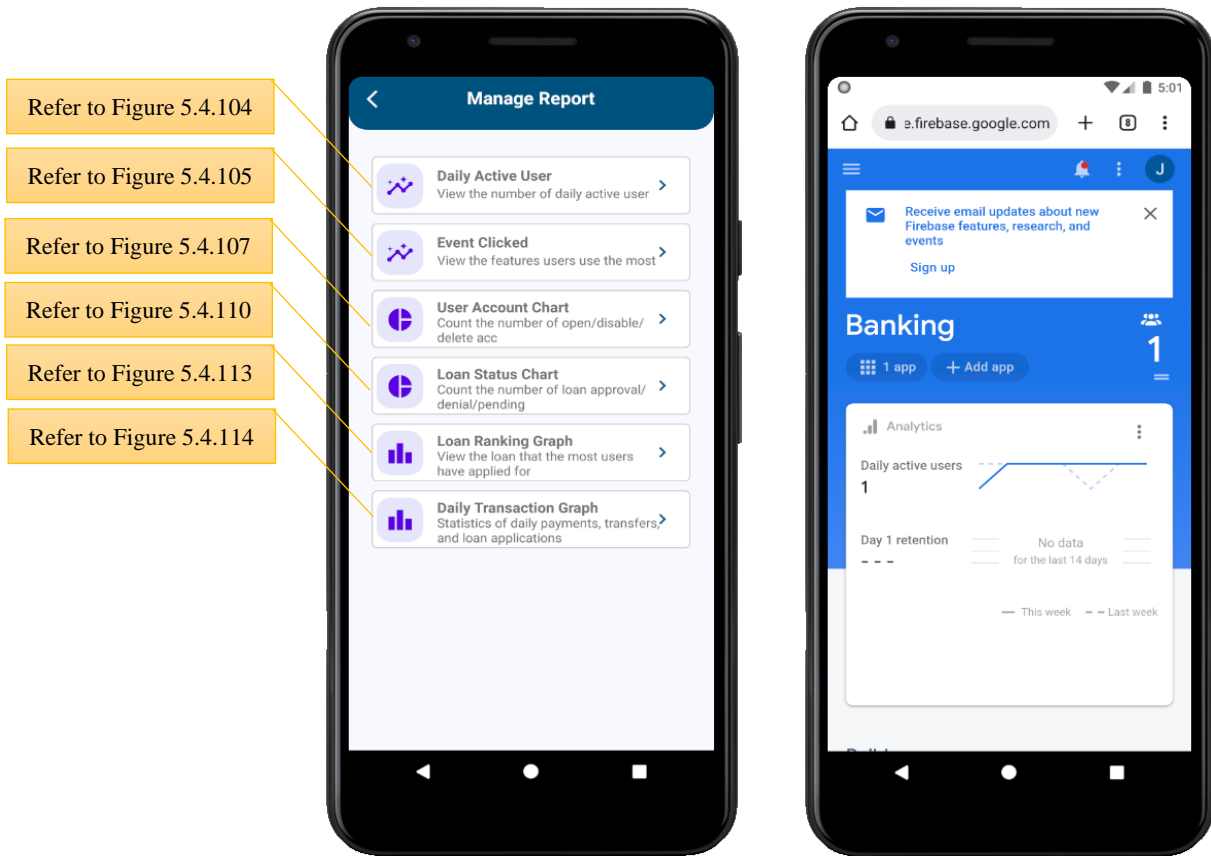
Since the scenario just now is that the bank manager rejected the loan application from this customer, he/she needs to add a note explaining the reason for his/her rejection (*refer to Figure 5.4.99*). After the bank manager clicks the send button, the notes appear at the bottom of the loan application (*refer to Figure 5.4.100*). The updated loan status and remark will be updated in Firebase of both the bank manager and the customer side as well.





**Figure 5.4.101** Loan History (Customer Side) **Figure 5.4.102** Rejected Reason

After the bank manager has updated the loan status (*refer to Figure 5.4.98*), the customer can view the updated loan status on his/her loan history side here. Since the loan application was rejected, when the customer clicks the details button, a dialog box will pop up showing the reason for the rejection.

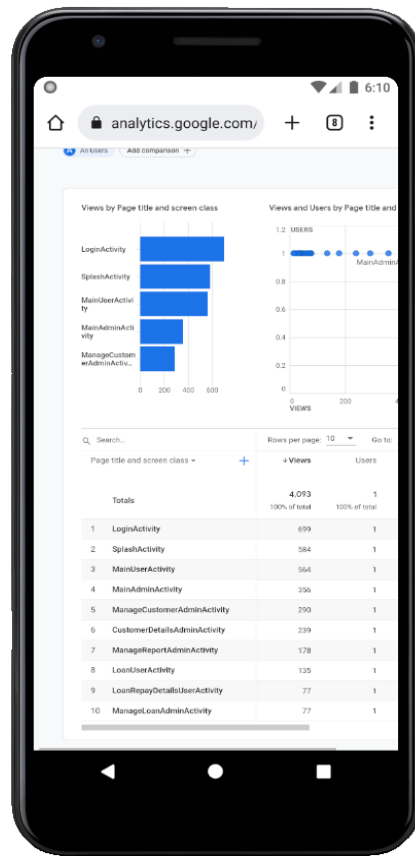


**Figure 5.4.103** Manage Report

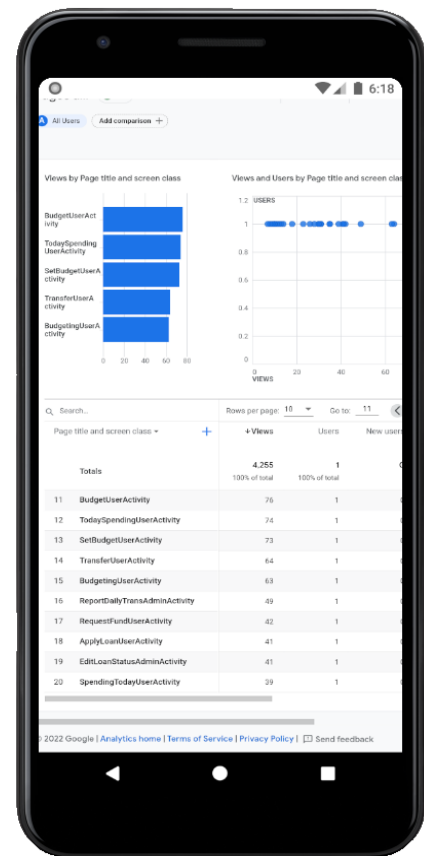
**Figure 5.4.104** Daily Active User

When clicking the manage report in the Bank Manage Main Page (refer to Figure 5.4.85), the bank manager is redirect to the Manage Report Page (refer to Figure 5.4.103). In the manage report page, there are few reports such as daily active user, event clicked, user account chart, loan status chart, loan ranking graph and daily transaction graph which different reports will display in different types of charts.

If the bank manager wants to view daily active user, he/she can click the daily active user button in the Manage Report (refer to Figure 5.4.103), the system redirects the bank manager to the Banky Firebase Overview webpage to see the statistic chart.

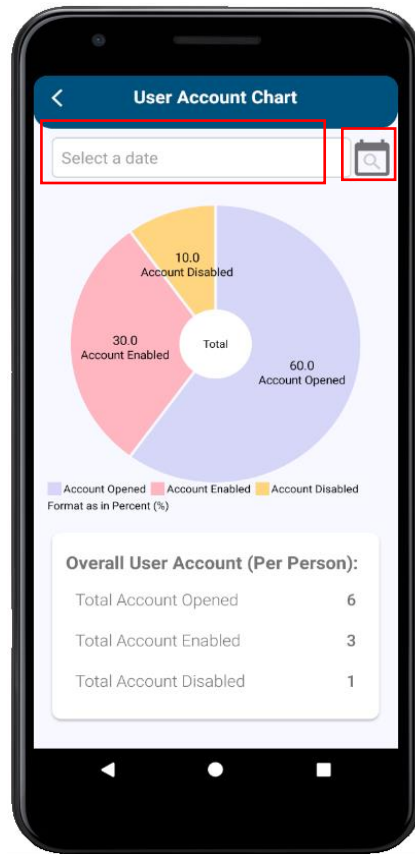


**Figure 5.4.105** Event Clicked

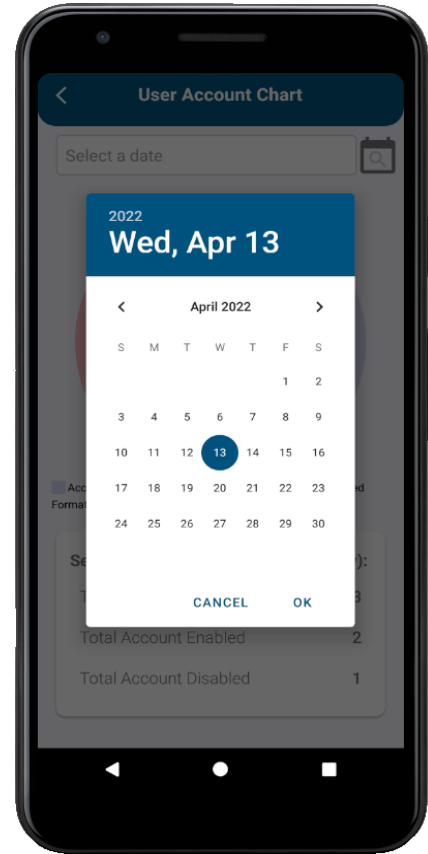


**Figure 5.4.106** Event Clicked (cont)

If the bank manager wants to know which function users used the most, he/she can click the event clicked button in the Manage Report Page (refer to Figure 5.4.103), the system will then redirect the bank manager to the Banky Google Analytics webpage. Basically, it will show all the activity along with its view, which the bank manager can zoom in to see more clearly as the text size showed by this webpage is quite small. So that the bank manager can know which function users used the least, and try to investigate whether it is because the function is not user-friendly enough or poor design layout for that function, then the bank manager only can improve the function.

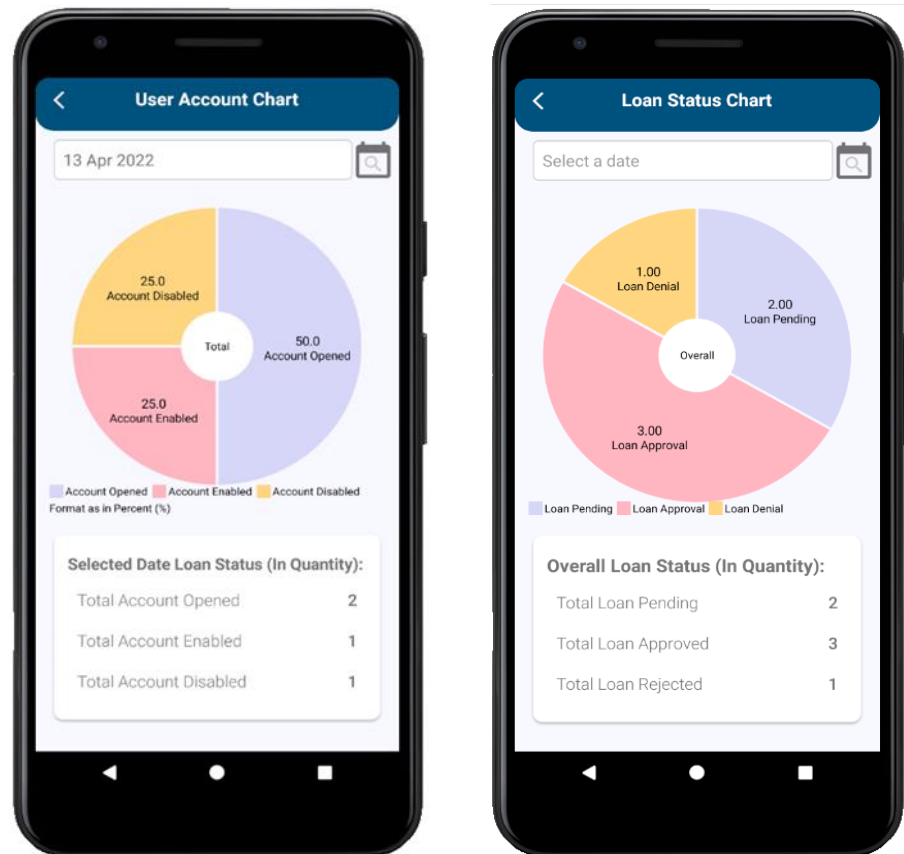


**Figure 5.4.107** User Account Chart



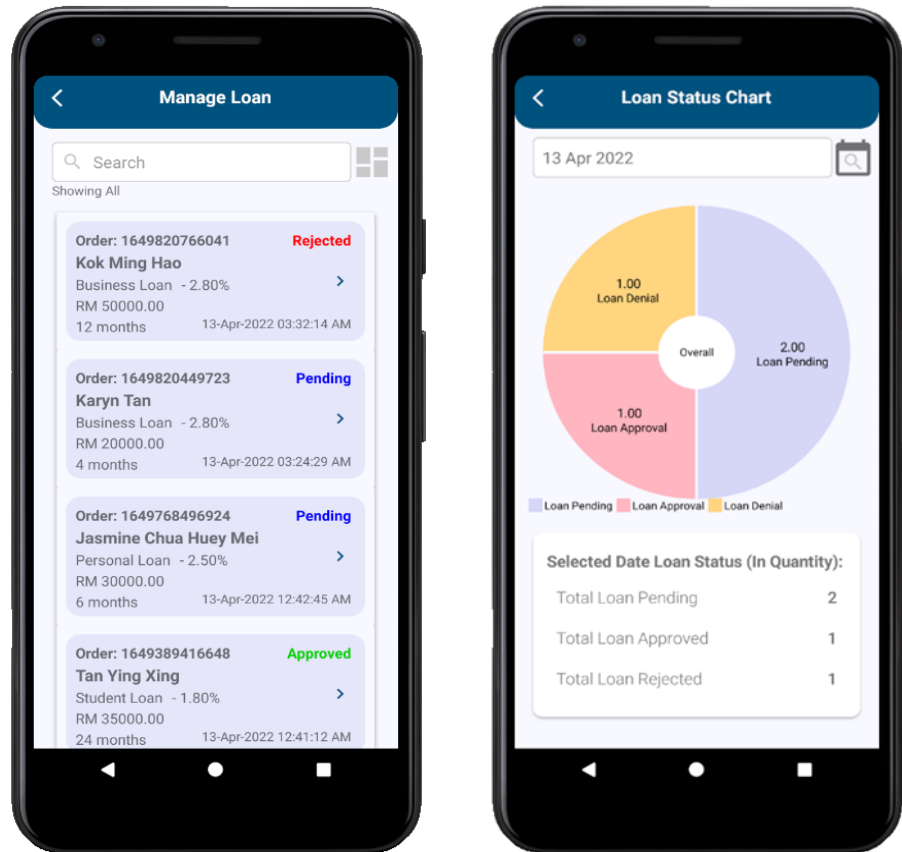
**Figure 5.4.108** Select a Date

When clicking the user account chart in the Manage Report Page (refer to Figure 5.4.103), the user is redirected to the User Account Chart page (refer to Figure 5.4.107) that displaying all the total account opened, enabled and disabled so far. If let's say the bank manager just wants to specific the total account opened, enabled and disabled on the specific date. He/she just needs to click the “select a date” box to select a date from a calendar (refer to Figure 5.4.108).



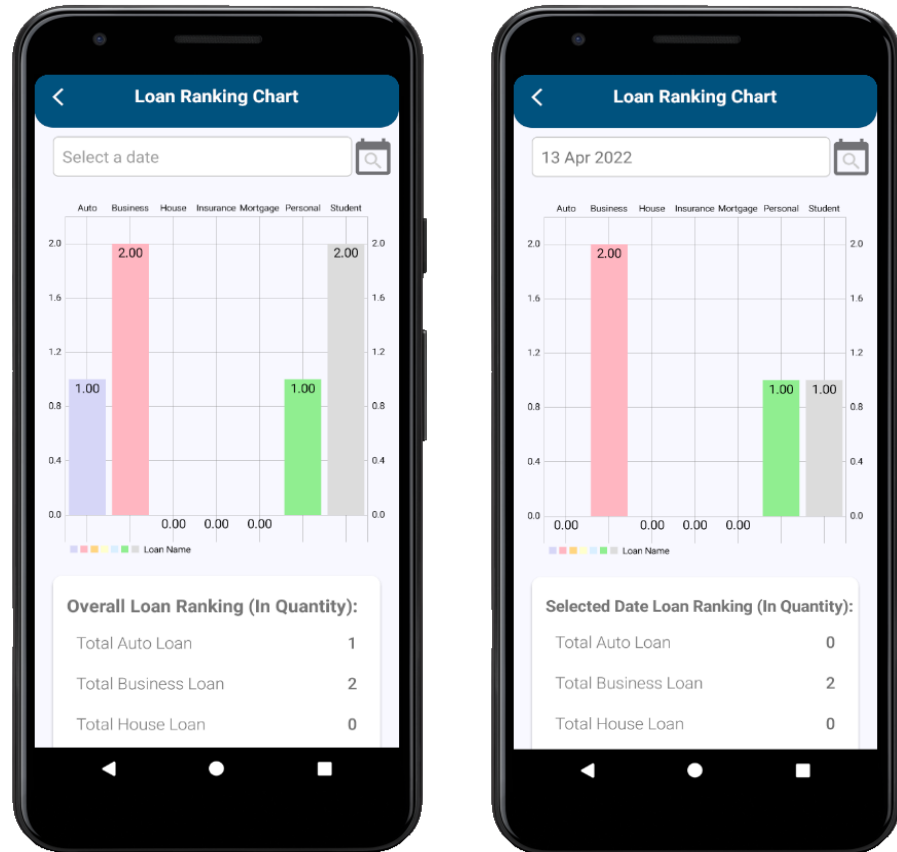
**Figure 5.4.109** Specific Date User Account Result **Figure 5.4.110** Loan Status Chart After the bank manager selects a date, he needs to click the search button next to the box (refer to Figure 5.4.107), and the system will display the results of the total number of accounts opened, enabled and disabled based on that date that selected by the bank manager. The bank manager can rotate the user account pie chart and even click on it, then it will toast a message displaying the selected item with its quantity.

When clicking the loan status chart button in the Manage Report Page (refer to Figure 5.4.103), the system redirects the bank manager to the Loan Status Chart Page and displays the overall loan status pie chart not filtered by any date (refer to Figure 5.4.110).



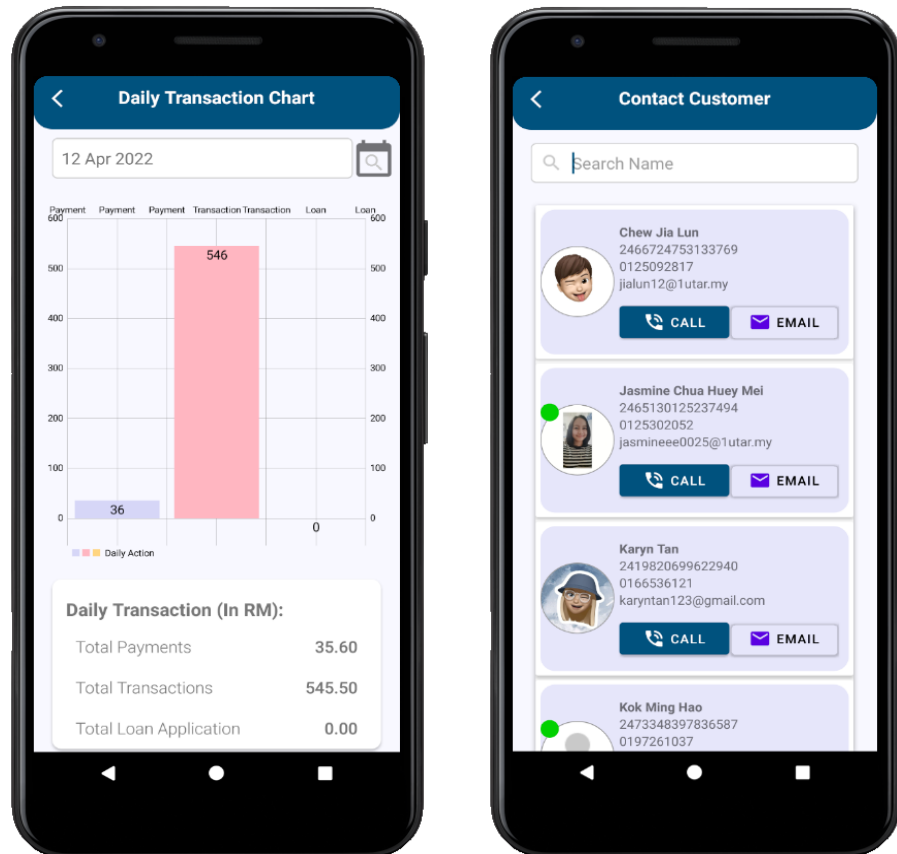
**Figure 5.4.111** Updated Manage Loan **Figure 5.4.112** Specific Date Loan Status Result

For example, the bank manager decides to view all loan applications matching April 13, 2022, so he/she simply selects April 13, 2022 from the calendar (*refer to Figure 5.4.108*). Therefore, we can see that the loan status result for the selected date correctly counts the number of each loan status from *Figure 5.4.111*, as shown in the *Figure 5.4.112*. The bank manager can rotate the loan status pie chart and even click on it, then it will toast a message displaying the selected item with its quantity.



**Figure 5.4.113** Loan Ranking Chart **Figure 5.4.114** Specific Date Loan Ranking Result

When the bank manager clicks the loan ranking chart button in the Manage Report Page (refer to Figure 5.4.107), the system redirects the bank manager to the Loan Ranking Chart Page (refer to Figure 5.4.113) that showing the overall loan ranking without do any filtering by date. If let's say the bank manager wants to view all loan applications that applied on April 13, 2022, he/she clicks the "select a date" box to pick and search a date from the calendar (refer to Figure 5.4.108). The system will display the loan ranking on the selected date (refer to Figure 5.4.114), so the bank manager can get to know which loan type that people applied the most on that date.

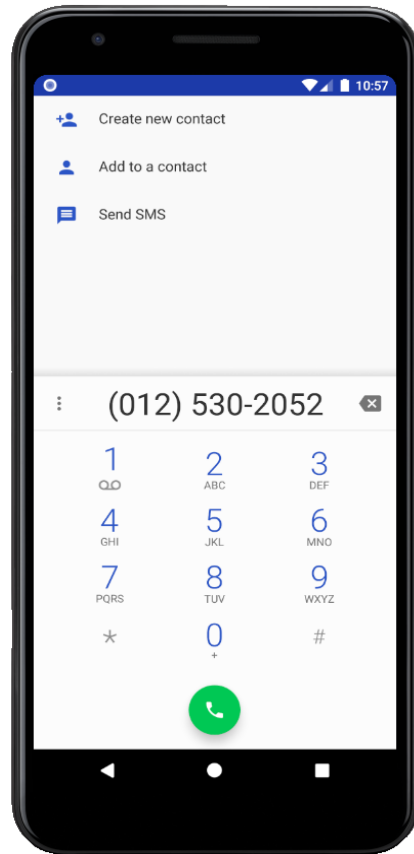


**Figure 5.4.115** Daily Transaction Chart **Figure 5.4.116** Contact Customer Page

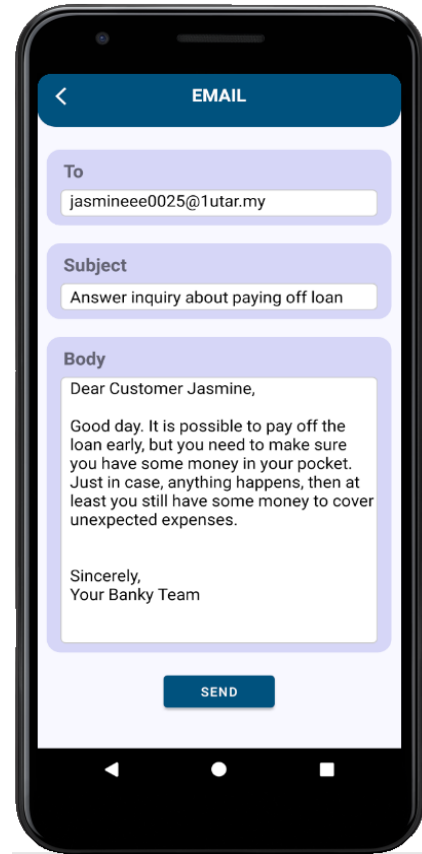
When the bank manager clicks the daily transaction chart button in the Manage Report Page (refer to Figure 5.4.107), the system redirects the bank manager to the Daily Transaction Chart Page (refer to Figure 5.4.115) that showing the daily total payment amount, daily total transaction amount and daily total loan application amount without do any filtering by date. If let's say the bank manager wants to view the data for April 12, 2022, he/she clicks the "select a date" box to pick and search a date from the calendar (refer to Figure 5.4.108). The system will display the total payments, total transactions and total loan applications on the selected date (refer to Figure 5.4.115), so the bank manager can get to know how much these transactions cost each day.

When clicking the contact customer button in the Manage Report Page (refer to Figure 5.4.107), the bank manager is redirected to the Contact Customer Page (refer to Figure 5.4.116). The bank manager can either one of the customers to contact them by either using phone call or email.





**Figure 5.4.117** Call Customer



**Figure 5.4.118** Email Customer

The bank manager can select the customer he/she wants to contact by simply clicking the call button below that specific customer, so the system will redirect the bank manager to the default phone dialing screen with the customer's phone number displayed on the keypad (*refer to Figure 5.4.117*), or click the email button below that particular customer and the bank manager will be redirected to the email page with the customer's email address displayed in the recipient column (*refer to Figure 5.4.118*).

## CHAPTER 6 SYSTEM EVALUATION AND DISCUSSION

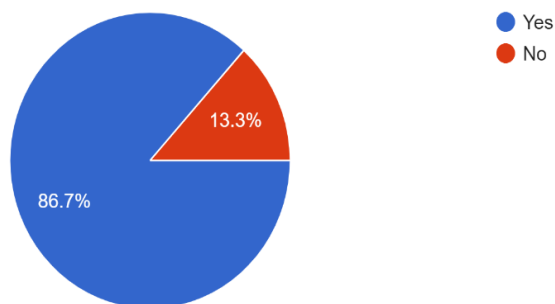
### 6.1 System Evaluation Survey Results

1. Name  
15 responses

Chua Hua Shen
Lee Hui Shan
Rachel Siow Ching Ni
Yvonne Lee Yi Jin
kelvin goh
Ang Wei Hang
Kwan Khai Yang
Ting Zhe Wei
Chang Je See
Joey Chia Mi Qi
chua tian hong
Goh YiTong
Chong Wai Lam
chua hua Chee
Chew Jia Lun

**Figure 6.1.1 Names of all respondents**

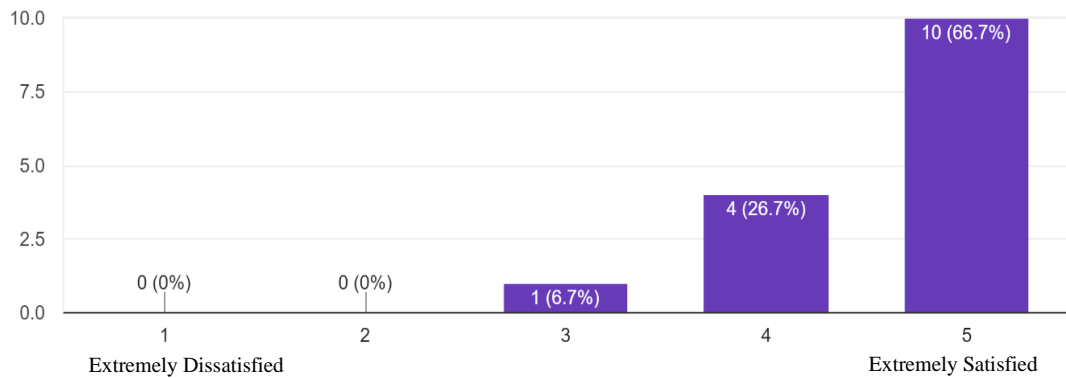
2. Did the onboarding screen (guide to using the Banky mobile application) catch your attention?  
15 responses



**Figure 6.1.2 Onboarding screen evaluation**

3. Using the scale below, please indicate your level of overall satisfaction with your experiences at this Banky application.

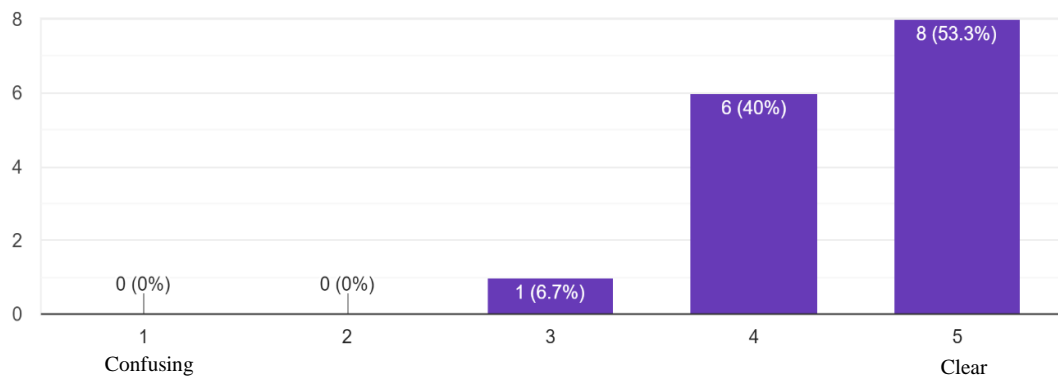
15 responses



**Figure 6.1.3 Overall satisfaction with experiences at Banky**

4. Consider your experience with SCREEN LAYOUTS when using the Banky application. Please rate that most appropriately reflects your impression of using the application.

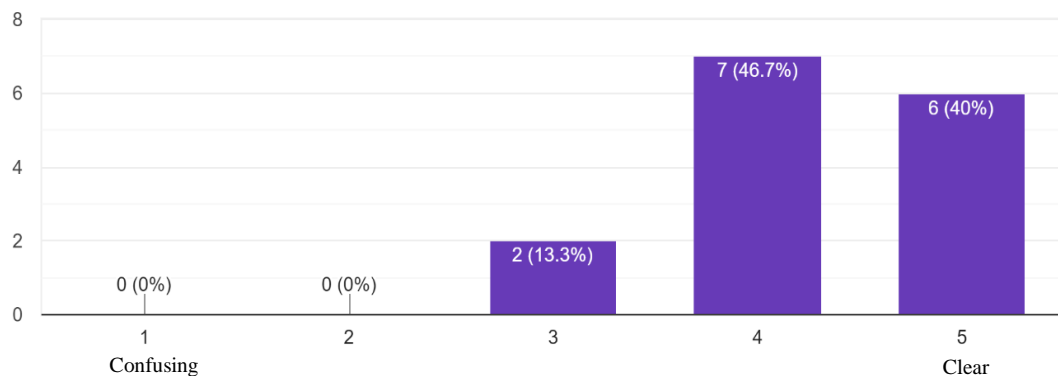
15 responses



**Figure 6.1.4 Screen layout design evaluation**

5. Consider your experience with NAVIGATION BUTTON when using the Banky application. Please rate that most appropriately reflects your impression of using the application.

15 responses



**Figure 6.1.5 Navigation button placement evaluation**

6. Consider your experience with the Banky application after using the Banky application. Please rate your agreement with the following statements about this application.

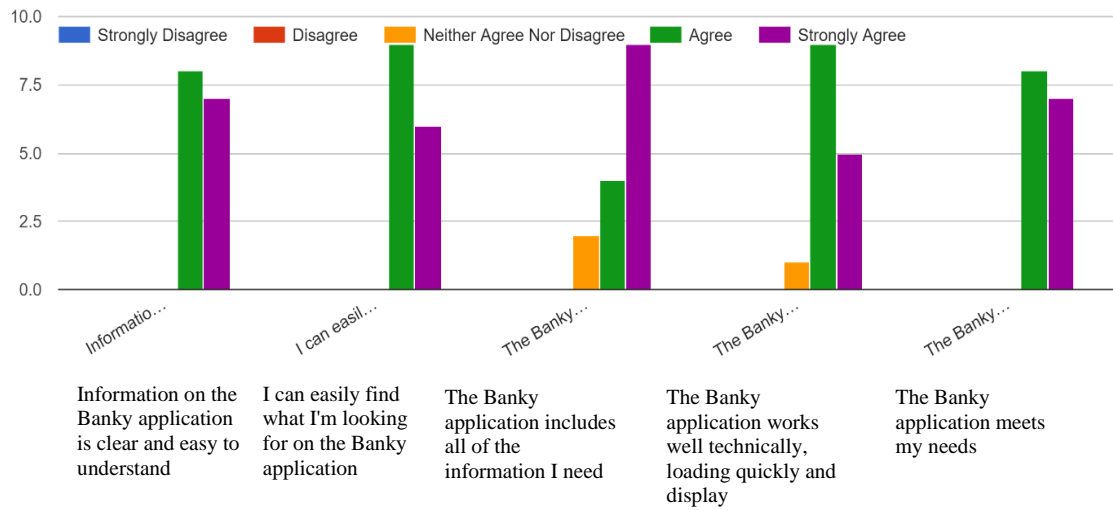


Figure 6.1.6 Evaluation at using Banky

7. Which of the following best describes your primary role in using the application?  
15 responses

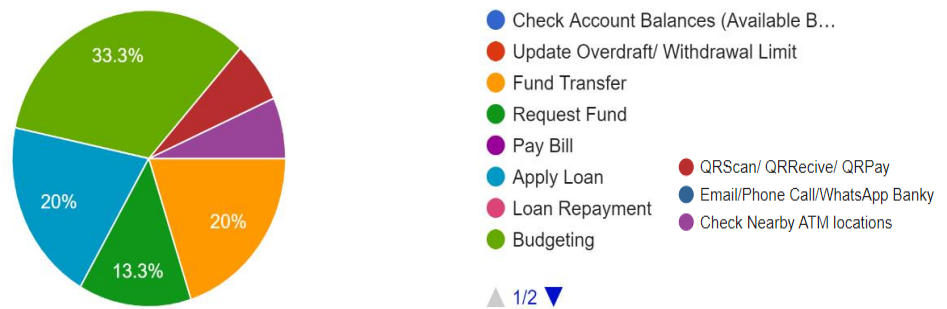
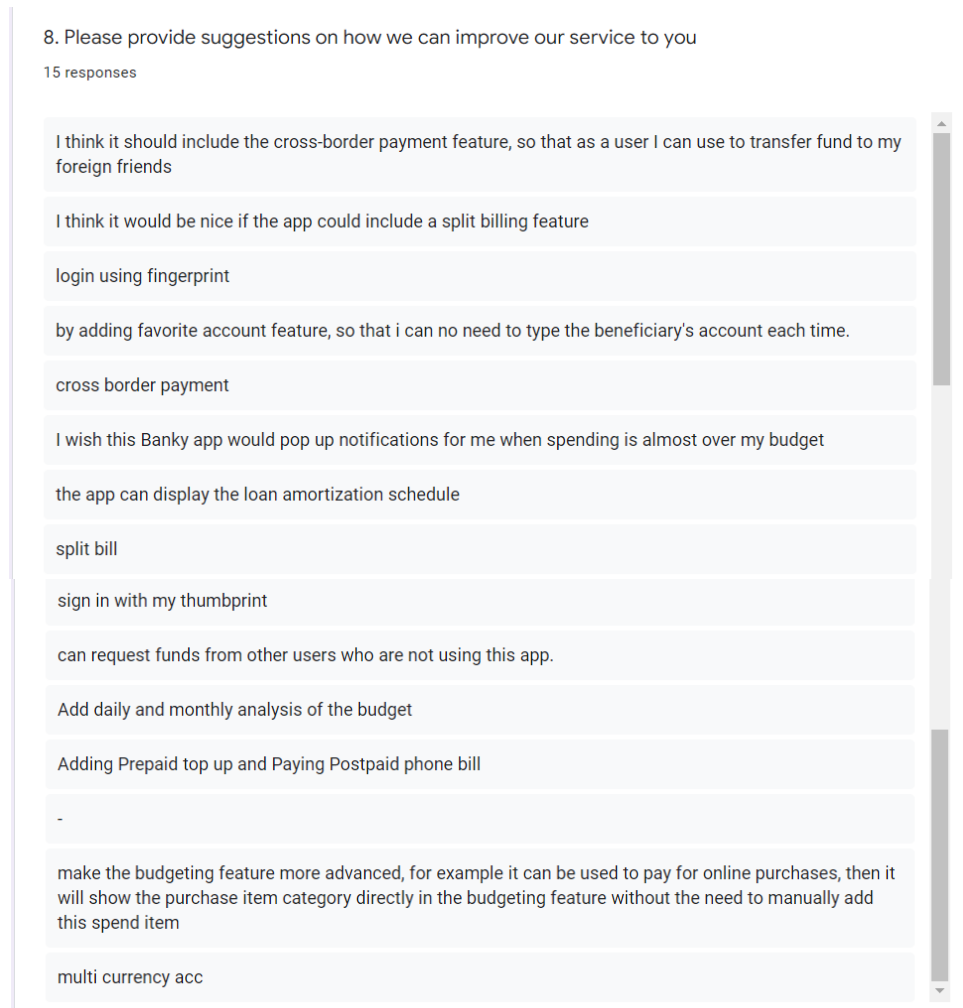


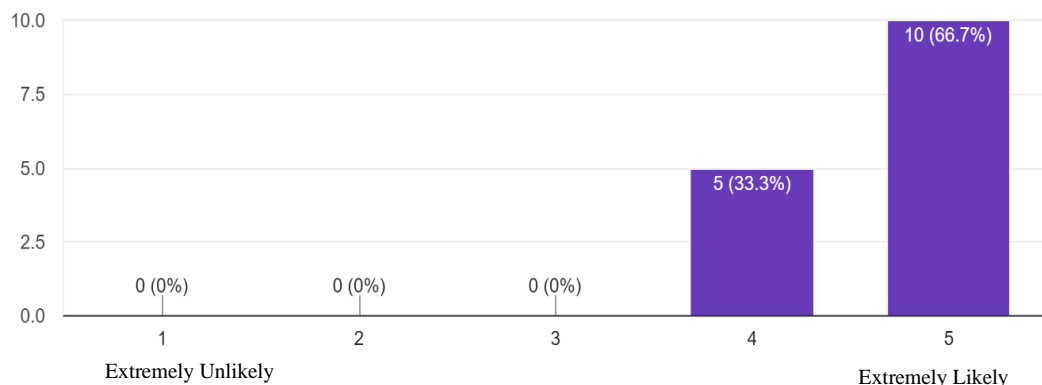
Figure 6.1.7 Evaluation of top feature selection



**Figure 6.1.8 Recommendations**

9. Considering your complete experience with our banking services, how likely is that you would recommend this Banky application to your family, friends or colleague?

15 responses



**Figure 6.1.9 Asking opinion whether to recommend this Banky**

There are 15 participants involve in this system evaluation, whose names are shown in *Figure 6.1.1*. Based on *Figure 6.1.2*, 86.7% of respondents feel that the onboarding screen are catching their attention, and only few respondents think the onboarding screen is not attracting their eyes. The purpose is to know that whether the onboarding screen catches their attention as the onboarding screen is a crucial step that guides users through the application and helps users to familiarize the application. 10 respondents give 5 marks, 4 respondents give 4 marks and only 1 respondent gives 3 marks for his/her overall satisfaction (*refer to Figure 6.1.3*). From the results shown in *Figure 6.1.4*, 53.3% of respondents said that the application provides a very clear screen layout interface, 40% of respondents thought that its screen layout interface is clear, and only 6.7% of respondents told that the screen layout interface neither clear nor confusing. 6 respondents said that the application offers very clear all navigation buttons, 7 respondents thought that all navigation buttons in the application is clear, and 2 out of 15 said that all navigation buttons in the application is neither clear nor confusing (*refer to Figure 6.1.5*). Basically, all the questions asked in *Figure 6.1.6* are to understand how users perceive the ease of use, completeness, speed, and user expectations of the application. 8 respondents agreed and 7 respondents strongly agreed that the information on the Banky is clear and easy to understand. 9 respondents agreed and 6 respondents strongly agreed that they can easily find out what they are looking for on the Banky application. 9 respondents strongly agreed, 4 agreed and 2 neither agreed nor disagreed that the Banky application did include all the information that they need. 9 respondents agreed, followed by 5 respondents strongly agreed and only 1 neither agreed nor disagreed that the Banky application works well technically, loading quickly and displaying appropriately. 8 respondents agreed and 7 respondents strongly agreed that the Banky application meets their needs. Based on the result displayed in *Figure 6.1.7*, the ranking of all features from highest to lowest are budgeting, apply loan, fund transfer, request fund, pay bill and QR Scan/ QR Receive /QR Pay, which followed by 33.3%, 20%, 20%, 13.3%, 6.7% and 6.7% correspondingly. *Figure 6.1.8* showed all the suggestions provided by all respondents, so that the system can be better like adding some features such as cross-border payment, split bill, sign in with fingerprint, and so on. 66.7% of respondents extremely likely to recommend this Banky application to their friends, family or colleagues, while 33.3% of respondents are likely to recommend this as well (*refer to Figure 6.1.9*).

## 6.2 Testing Setup and Result

### 6.2.1 Unit Testing 1 - Login

Test Objective: To ensure the user is able to login with valid email and password.

Input	Expected Output	Actual Output
Login by entering the correct email address and password.	The system allows the user login.	The user login successfully.
Login by entering the wrong email address or password.	The system does not allow the user to login	The user cannot log into the system.
Login by without entering any value.	The system does not allow the user to login.	The user cannot log into the system.
Click the “Forgot Password”	The system sends a reset password email instruction.	The user is able to reset new password.
Click the “Not have account? Register”	The system redirects the user to register user page.	The user is able to go to register user page.

**Table 6.2.1 Unit Testing 1 – Login**

**6.2.2 Unit Testing 2 – Register User**

Test Objective: To ensure the user is able to register as a new user account.

<b>Input</b>	<b>Expected Output</b>	<b>Actual Output</b>
Filled in all the text fields without uploading a profile picture.	The registration process was successfully.	The user has been registered successfully as a new user account.
Filled in all the text fields including uploading a profile picture.	The registration process was successfully.	The user has been registered successfully as a new user account.
Enter invalid email address pattern.	The registration process was unsuccessfully.	The user failed to register as a new user account.
Enter password which is less than 6 characters.	The registration process was unsuccessfully.	The user failed to register as a new user account.
Click the register button without entering any fields	The registration process was unsuccessfully.	The user failed to register as a new user account.
Click the “Are you a bank manager or staff? Register here”	The system redirects the user to register bank staff page.	The user is able to go to register bank staff page.

**Table 6.2.2 Unit Testing 2 – Register User**



**6.2.3 Unit Testing 3 – Register Bank Staff**

Test Objective: To ensure the user is able to register as a new bank staff account.

<b>Input</b>	<b>Expected Output</b>	<b>Actual Output</b>
Fill all text fields without uploading a profile picture.	The registration process was successfully.	The user has been registered successfully as a new user account.
Fill all text fields including uploading a profile picture.	The registration process was successfully.	The user has been registered successfully as a new user account.
Enter invalid email address pattern.	The registration process was unsuccessfully.	The user failed to register as a new user account.
Enter password which is less than 6 characters.	The registration process was unsuccessfully.	The user failed to register as a new user account.
Click the register button without entering any fields	The registration process was unsuccessfully.	The user failed to register as a new user account.

**Table 6.2.3 Unit Testing 3 – Register Bank Staff**

**6.2.4 Unit Testing 4 – User Main Page (Home Navigation Bar Button)**

Test Objective: To ensure the user is able to use the functionalities provided.

<b>Input</b>	<b>Expected Output</b>	<b>Actual Output</b>
Click the copy icon that shown in the digital virtual visa card	The system allows the user to copy the account number.	The user can copy the account number.
Click the logout button	The system logs the user out.	The user is able to logout successfully.
Click the Account icon.	The system redirects the user to the My Account Page.	The user is able to go to the My Account Page.
Click the Transfer icon.	The system redirects the user to the Fund Transfer Page.	The user is able to go to the Fund Transfer Page.
Click the Pay Bills icon.	The system redirects the user to the Pay Bills Page.	The user is able to go to the Pay Bill Page.
Click the Apply Loan icon.	The system redirects the user to the Loan Page.	The user is able to go to the Loan Page.
Click the Budgeting icon.	The system redirects the user to the Budget Page.	The user is able to go to the Budget Page.

**Table 6.2.4 Unit Testing 4 – User Main Page (Home Navigation Bar Button)**

**6.2.5 Unit Testing 5 – My Account**

Test Objective: To ensure the user is able to check and update my account.

<b>Input</b>	<b>Expected Output</b>	<b>Actual Output</b>
Swipe left or right for the overdraft limit seek bar.	The system updates and displays the newest overdraft limit.	The user is able to update the overdraft limit.
Swipe left or right for the withdrawal limit seek bar.	The system updates and displays the newest withdrawal limit.	The user is able to update the withdrawal limit.
Click the Top Up icon.	The system displays all cash deposit history	The user is able to view all cash deposit history
Click the Withdrawal icon.	The system displays all cash withdrawal history	The user is able to view all cash withdrawal history

**Table 6.2.5 Unit Testing 5 – My Account**

**6.2.6 Unit Testing 6 – Fund Transfer**

Test Objective: To ensure the user is able to choose a way to transfer or request funds.

<b>Input</b>	<b>Expected Output</b>	<b>Actual Output</b>
Click the Fund Transfer Within Banky button.	The system redirects the user to the Fund Transfer Within Banky Page.	The user is able to go to the Fund Transfer Within Banky Page.
Click the Fund Transfer To Other Bank button.	The system redirects the user to the Fund Transfer To Other Bank Page.	The user is able to go to the Fund Transfer To Other Bank Page.
Click the Request Fund button.	The system redirects the user to the Request Fund Page.	The user is able to go to the Request Fund Page.

**Table 6.2.6 Unit Testing 6 – Fund Transfer**

**6.2.7 Unit Testing 7 – Fund Transfer Within Banky**

Test Objective: To ensure the user is able to fund transfer within Banky.

<b>Input</b>	<b>Expected Output</b>	<b>Actual Output</b>
Enter all text fields & the account number existed & the available balance $\geq$ transfer amount	The transaction within Banky process was successfully.	The user transferred funds successfully to another Banky user.
Enter all text fields except payment details & the account number existed & the available balance $\geq$ transfer amount	The transaction within Banky process was successfully.	The user transferred funds successfully to another Banky user.
Enter all text fields & the account number not existed & the available balance $\geq$ transfer amount	The transaction within Banky process was unsuccessfully.	The user is not able to transfer funds to another Banky user.
Enter all text fields & the account number not existed & the available balance $<$ transfer amount	The transaction within Banky process was unsuccessfully.	The user is not able to transfer funds to another Banky user.
Enter all text fields & the account number existed & the available balance $<$ transfer amount	The transaction within Banky process was unsuccessfully.	The user is not able to transfer funds to another Banky user.
Click the transfer now button without entering any text fields	The transaction within Banky process was unsuccessfully.	The user is not able to transfer funds to another Banky user.

**Table 6.2.7 Unit Testing 7 – Fund Transfer Within Banky**

**6.2.8 Unit Testing 8 – Fund Transfer To Other Bank**

Test Objective: To ensure the user is able to fund transfer to other bank.

<b>Input</b>	<b>Expected Output</b>	<b>Actual Output</b>
Enter all text fields & the account number existed & the beneficiary's name is correct & the available balance $\geq$ transfer amount	The transaction to other bank process was successfully.	The user transferred funds successfully to other bank user.
Enter all text fields except payment details & the beneficiary's name is correct & the account number existed & the available balance $\geq$ transfer amount	The transaction to other bank process was successfully.	The user transferred funds successfully to other bank user.
Enter all text fields & the account number not existed & the beneficiary's name is correct & the available balance $\geq$ transfer amount	The transaction to other bank process was unsuccessfully.	The user is not able to transfer funds to other bank user.
Enter all text fields & the account number existed & the beneficiary's name is wrong & the available balance $\geq$ transfer amount	The transaction to other bank process was unsuccessfully.	The user is not able to transfer funds to other bank user.
Enter all text fields & the account number existed & the beneficiary's name is correct & the available balance $<$ transfer amount	The transaction to other bank process was unsuccessfully.	The user is not able to transfer funds to other bank user.
Enter all text fields & the account number not existed & the beneficiary's name is	The transaction to other bank process was unsuccessfully.	The user is not able to transfer funds to other bank user.

correct & the available balance < transfer amount		
Enter all text fields & the account number not existed & the beneficiary's name is wrong & the available balance >= transfer amount	The transaction to other bank process was unsuccessfully.	The user is not able to transfer funds to other bank user.
Enter all text fields & the account number not existed & the beneficiary's name is wrong & the available balance < transfer amount	The transaction to other bank process was unsuccessfully.	The user is not able to transfer funds to other bank user.
Click the transfer now button without entering any text fields	The transaction to other bank process was unsuccessfully.	The user is not able to transfer funds to other bank user.

**Table 6.2.8 Unit Testing 8 – Fund Transfer To Other Bank**

**6.2.9 Unit Testing 9 – Request Fund**

Test Objective: To ensure the user is able to request funds from another Banky user.

<b>Input</b>	<b>Expected Output</b>	<b>Actual Output</b>
Click the add icon floating button & enter all contact information	Added a new contact process was successfully.	The user is able to choose the contact from the list.
Click the add icon floating button & enter all contact information except uploading a photo	Added a new contact process was successfully.	The user is able to choose the contact from the list.
Click on any of the contacts & enter the request amount	The request fund process has been successfully sent to the payer.	The user is able to request funds from another Banky user.
Click the add icon floating button & click the save button without entering any contact information	Added a new contact process was unsuccessfully.	The user is not able to choose the contact from the list.
Click on any of the contacts & click the request now button without entering the requested amount	The request fund process was unsuccessfully.	The user is not able to request funds from another Banky user.

**Table 6.2.9 Unit Testing 9 – Request Fund**



**6.2.10 Unit Testing 10 – Pay Bills**

Test Objective: To ensure the user is able to pay the bill.

<b>Input</b>	<b>Expected Output</b>	<b>Actual Output</b>
Choose the type & enter account number & enter payment amount & the available balance $\geq$ payment amount	The payment process was successfully.	The user pays the bill successfully.
Choose the type & enter account number & enter payment amount & the available balance $<$ payment amount	The payment process was unsuccessfully.	The user is not able to pay the bill.
Choose the type & enter account number & the available balance $\geq$ payment amount	The payment process was unsuccessfully.	The user is not able to pay the bill.
Choose the type & enter account number & the available balance $<$ payment amount	The payment process was unsuccessfully.	The user is not able to pay the bill.
Choose the type & enter payment amount & the available balance $\geq$ payment amount	The payment process was unsuccessfully.	The user is not able to pay the bill.
Choose the type & enter payment amount & the available balance $<$ payment amount	The payment process was unsuccessfully.	The user is not able to pay the bill.
Choose the type & click the pay now button without entering the account number and payment amount	The payment process was unsuccessfully.	The user is not able to pay the bill.

**Table 6.2.10 Unit Testing 10 – Pay Bills**

**6.2.11 Unit Testing 11 – Loan**

Test Objective: To ensure the user is able to apply loan, check loan status and repay loan.

<b>Input</b>	<b>Expected Output</b>	<b>Actual Output</b>
Click the apply loan button & enter all required fields & click promptly apply button	The loan application process was successfully.	The user has applied the loan successfully.
Click the apply loan button & switch on the repayment plan without entering any required fields	The loan application process was unsuccessfully.	The user is failed to apply the loan.
Click the details button on any loan application & click the pay button to repay for a specific month & the available balance $\geq$ repayment amount	The repayment loan process was successfully.	The user made repayment for that specific month successfully.
Click the details button on any loan application & click the pay button to repay for a specific month & the available balance $<$ repayment amount	The repayment loan process was unsuccessfully.	The user is failed to make repayment for that specific month.

**Table 6.2.11 Unit Testing 11 – Loan**

**6.2.12 Unit Testing 12 – Budgeting**

Test Objective: To ensure the user is able to set weekly budget and add today spending.

<b>Input</b>	<b>Expected Output</b>	<b>Actual Output</b>
Click the set weekly budget button & click the add icon floating button to add a weekly budget item and its amount.	Added a weekly budget item process was successfully.	The user is able to a new weekly budget item successfully.
Click the set weekly budget button & click any of the budget items & update the budget amount.	Updated a budget item process was successfully.	The user is able to update the budget item successfully.
Click the set weekly budget button & click any of the budget items & click the delete button.	Removed a weekly budget item process was successfully.	The user is able to delete the budget item successfully.
Click the add today spending button & click the add icon floating button & select a spending item & enter amount & enter note.	Added a spending item process was successfully.	The user is able to a new spending item successfully.
Click the add today spending button & click the add icon floating button & select a spending item.	Added a spending item process was unsuccessfully.	The user is not able to a new spending item successfully.
Click the add today spending button & click the add icon floating button & select a spending item & enter amount.	Added a spending item process was unsuccessfully.	The user is not able to a new spending item successfully.

Click the add today spending button & click the add icon floating button & select a spending item & enter note.	Added a spending item process was unsuccessfully.	The user is not able to a new spending item successfully.
Click the add today spending button & click the add icon floating button & click the save button without entering the amount and note.	Added a spending item process was unsuccessfully.	The user is not able to a new spending item successfully.
Click the track this week spending button.	The system displayed all spending records that happened this week.	The user is able to view this week spending.
Click the manage your budget button.	The system displayed the percentage of spending versus budget, a pie chart and summary spending status.	The user is able to view its spending status grouped by each item category and also overall spending status.
Click the view all spending history.	The system displayed all spending records.	The user is able to view all spending history.

**Table 6.2.12 Unit Testing 12 – Budgeting**

**6.2.13 Unit Testing 13 – User Main Page (History Navigation Bar Button)**

Test Objective: To ensure the user is able to view all transaction history.

Input	Expected Output	Actual Output
Click the History icon from the navigation bar.	The system displayed all transaction history.	The user is able to view all transaction history.

**Table 6.2.13 Unit Testing 13 – User Main Page (History Navigation Bar Button)**

**6.2.14 Unit Testing 14 – User Main Page (QR Scan Navigation Bar Button)**

Test Objective: To ensure the user is able to scan, receive or pay using QR code.

Input	Expected Output	Actual Output
Click the Scan icon from the navigation bar.	The system displayed scan, receive and pay tabs.	The user is able to scan or receive or pay by using QR code.
Click the scan button.	The system displayed a scanner for the user to scan QR code.	The user is able to use it to scan QR code.
Click the receive button.	The system displayed a unique QR code.	The user is able to show this QR code to others.
Click the pay button.	The system displayed the QR code and bar code.	The user is able to show this QR code or bar code to the cashier.

**Table 6.2.14 Unit Testing 14 – User Main Page (QR Scan Navigation Bar Button)**

**6.2.15 Unit Testing 15 – User Main Page (Notification Navigation Bar Button)**

Test Objective: To ensure the user is able to view notification records.

Input	Expected Output	Actual Output
Click the Notification icon from the bottom navigation bar.	The system displayed all notification history.	The user is able to view all notification history.
Click the pay button.	The system redirects the user to the Pay For Request Fund Page.	The user is able to pay for the request fund by another Banky user.

**Table 6.2.15 Unit Testing 15 – User Main Page (Notification Navigation Bar Button)**

**6.2.16 Unit Testing 16 – User Main Page (Profile Navigation Bar Button)**

Test Objective: To ensure the user is able to view update profile, change password, check FAQ, contact Banky, and look for nearby ATM.

Input	Expected Output	Actual Output
Click the Profile icon from the bottom navigation bar.	The system displayed the profile picture, name and email address.	The user is able to view all his/her own profile picture, name and email address.
Click the profile setting button & click the edit button & click update button.	Updated profile details process was successfully.	The user is able to update the profile details successfully.
Click the change password & enter the email address.	A reset password email instruction has been sent to the user.	The user is able to change the password.
Click the help center button.	The system displayed a list of FAQs along with its answers.	The user is able to view the FAQ and its answer.
Click the customer support button & click the email button & enter the email subject & enter the email body.	Sending an email to Banky process was successfully.	The user is able to send an email to Banky.
Click the customer support button & click the email button & enter the email subject.	Sending an email to Banky process was unsuccessfully.	The user is not able to send an email to Banky.
Click the customer support button & click the email button & enter the email body.	Sending an email to Banky process was unsuccessfully.	The user is not able to send an email to Banky.

Click the customer support button & click the email button & click send without entering the email subject and body.	Sending an email to Banky process was unsuccessfully.	The user is not able to send an email to Banky.
Click the customer support button & click the phone call button.	The system redirects the user to default phone dialing screen.	The user is able to call Banky now.
Click the customer support button & click the WhatsApp button & WhatsApp is installed in the device.	The system redirects the user to open up the WhatsApp application.	The user is able to use WhatsApp to communicate with Banky.
Click the customer support button & click the WhatsApp button & WhatsApp is not installed in the device.	Opening WhatsApp application process was unsuccessfully.	The user is not able to use WhatsApp to communicate with Banky.
Click the nearby ATM button.	The system redirects the user to open up the Google Maps that displaying all nearby ATMs.	The user is able to check nearby ATM locations from the Google Maps.

**Table 6.2.16 Unit Testing 16 – User Main Page (Profile Navigation Bar Button)**

**6.2.17 Unit Testing 17 – Bank Manager Main Page**

Test Objective: To ensure the bank manager is able to manage customer, account, loan, report and contact customer.

<b>Input</b>	<b>Expected Output</b>	<b>Actual Output</b>
Click the copy icon that shown in the digital virtual visa card.	The system allows the bank manager to copy the account number.	The bank manager can copy the account number.
Click the logout button.	The system logs the bank manager out.	The bank manager is able to logout successfully.
Click the Manage Customer button.	The system redirects the user to the Manage Customer Page.	The user is able to go to the Manage Customer Page.
Click the Manage Account button.	The system redirects the user to the Firebase Authentication webpage.	The user is able to go to the Firebase Authentication webpage.
Click the Manage Loan button.	The system redirects the user to the Manage Loan Page.	The user is able to go to the Manage Loan Page.
Click the Manage Report button.	The system redirects the user to the Manage Report Page.	The user is able to go to the Manage Report Page.
Click the Contact Customer button.	The system redirects the user to the Contact Customer Page.	The user is able to go to the Contact Customer Page.

**Table 6.2.17 Unit Testing 17 – Bank Manager Main Page**



**6.2.18 Unit Testing 18 – Manage Customer**

Test Objective: To ensure the bank manager is able to view customers' balances and profile information.

<b>Input</b>	<b>Expected Output</b>	<b>Actual Output</b>
Enter any keyword into the search bar.	The system allows results matching the keyword to be displayed.	The bank manager can view results showing customer names that contain keywords.
Click any of the customers.	The system displayed the customer balances such as the available balance, current balance, fixed deposit balance, total payment made, total received, and total sent in pie chart form.	The bank manager is able to view the pie chart showing the balances of the customer.
Click any of the customers & click on the pie chart.	The system displayed the item selected and its amount.	The bank manager is able to see a toast message showing it.
Click any of the customers & switch on the update customer profile & click the update button.	The system displayed the customer profile details such as name, ic no, bank account number, phone no, email address and annual interest rate for the bank manager to edit/modify it.	The bank manager is able to update the customer profile details.

**Table 6.2.18 Unit Testing 18 – Manage Customer**

**6.2.19 Unit Testing 19 – Manage Account**

Test Objective: To ensure the bank manager is able to manage customers' account.

<b>Input</b>	<b>Expected Output</b>	<b>Actual Output</b>
Click the three dots next to the customer & click disable account.	The user's account has been successfully frozen, and the user is not allowed to log in to Banky.	Users whose accounts have been disabled cannot log in to Banky.
Click the three dots next to the customer & click enable account.	The user's account has been successfully unfrozen, and the user is allowed to log in to Banky again.	Users with previously disabled accounts can now log into Banky again.

**Table 6.2.19 Unit Testing 19 – Manage Account**

**6.2.20 Unit Testing 20 – Manage Loan**

Test Objective: To ensure the bank manager is able to manage loan applications.

<b>Input</b>	<b>Expected Output</b>	<b>Actual Output</b>
Enter any keyword into the search bar.	The system allows results matching the keyword to be displayed.	The bank manager can view results showing loan types that contain keywords.
Click the filter icon to filter by either pending, approved, or rejected.	The system allowed for the bank manager to filter the loan application by loan status.	The bank manager is able to view the loan applications based on the types of loan status he/she wants.
Click any of the customers.	The system displayed the full loan application details.	The bank manager is able to view the customer's loan application details.
Click any of the customers & click the edit button & edit the loan status.	Updated loan status process was successfully.	The bank manager is able to edit the loan status.
Click any of the customers & click the add comment button & click to send.	Added a comment process was successfully and is displayed at the bottom of the loan application. Also displayed the comment to the customer side as well.	The bank manager is able to add a comment for loan rejections. The customer is able to view the rejected reason.
Click on the pie chart.	The system displayed the item selected and its amount.	The bank manager is able to see a toast message showing it.

**Table 6.2.20 Unit Testing 20 – Manage Loan**

**6.2.21 Unit Testing 21 – Manage Report**

Test Objective: To ensure the bank manager is able to view different types of reports.

<b>Input</b>	<b>Expected Output</b>	<b>Actual Output</b>
Click the daily active user button.	The system redirects the bank manager to Firebase Overview to check to see the number of daily active users.	The bank manager is able to view the number of daily active users.
Click the event clicked button.	The system redirects the bank manager to Google Analytics for Firebase to see the number of views per screen or page.	The bank manager is able to view the number of views per screen or page.
Click the user account chart.	The system displayed an overall user account chart.	The bank manager is able to view the overall user account chart.
Click the user account chart & select a date from the “select a date” box & click the search button.	The system displayed a user account chart with the total number of accounts opened, enabled and disabled happened on the selected date.	The bank manager can decide which date the pie chart data is displayed.
Click the user account chart & select a date from the “select a date” box & without clicking the search button.	The system displayed out the pie chart immediately, right after the user selected a date from the calendar.	The bank manager needs to click the search function, then it only can display out the pie chart with data.
Click on the user account pie chart.	The system displayed the item selected and its amount.	The bank manager is able to see a toast message showing it.

Click the loan status chart.	The system displayed an overall loan status chart.	The bank manager is able to view the overall loan status chart.
Click the loan status chart & select a date from the “select a date” box & click the search button.	The system displayed a loan status chart with the total number of loan pending, approved and rejected happened on the selected date.	The bank manager can decide which date the pie chart data is displayed.
Click the loan status chart & select a date from the “select a date” box & without clicking the search button.	The system displayed out the pie chart immediately, right after the user selected a date from the calendar.	The bank manager needs to click the search function, then it only can display out the pie chart with data.
Click on the loan status pie chart.	The system displayed the item selected and its amount.	The bank manager is able to see a toast message showing it.
Click the loan ranking chart.	The system displayed an overall loan ranking chart.	The bank manager is able to view the overall loan ranking chart.
Click the loan ranking chart & select a date from the “select a date” box & click the search button.	The system displayed a loan ranking chart with the total number of each loan type on the selected date.	The bank manager can decide which date the bar chart data is displayed.
Click the loan ranking chart & select a date from the “select a date” box & without clicking the search button.	The system displayed out the bar chart immediately, right after the user selected a date from the calendar.	The bank manager needs to click the search function, then it only can display out the pie chart with data.
Click on the loan ranking chart.	The system displayed the item selected and its amount.	The bank manager is able to see a toast message showing it.

Click the daily transaction chart & select a date from the “select a date” box & click the search button.	The system displayed a daily transaction chart with the total number of payments, transactions, loan applications on the selected date.	The bank manager can decide which date the bar chart data is displayed.
Click the daily transaction chart & select a date from the “select a date” box & without clicking the search button.	The system displayed out the bar chart immediately, right after the user selected a date from the calendar.	The bank manager needs to click the search function, then it only can display out the pie chart with data.
Click on the daily transaction chart.	The system displayed the item selected and its amount.	The bank manager is able to see a toast message showing it.

**Table 6.2.21 Unit Testing 21 – Manage Report**

**6.2.22 Unit Testing 22 – Contact Customer**

Test Objective: To ensure the bank manager is able to call or email customers.

<b>Input</b>	<b>Expected Output</b>	<b>Actual Output</b>
Enter any keyword into the search bar.	The system allows results matching the keyword to be displayed.	The bank manager can view results showing customer names that contain keywords.
Click the call button under any of the customers.	The system redirects the bank manager to the default phone dialing screen.	The bank manager is able to call the customer.
Click the email button under any of the customers.	The system redirects the bank manager to email page.	The bank manager is able to go to email page.
Click the email button under any of the customers & enter the email subject & enter the email body.	Sending an email to the customer process was successfully.	The bank manager is able to send an email to the customer.
Click the email button under any of the customers & enter the email subject.	Sending an email to the customer process was unsuccessfully.	The bank manager is not able to send an email to the customer.
Click the email button under any of the customers & enter the email body.	Sending an email to the customer process was unsuccessfully.	The bank manager is not able to send an email to the customer.

**Table 6.2.22 Unit Testing 22 – Contact Customer**

### 6.3 Project Challenges

The following are some of the challenges that have been identified. The first challenge when developing the system is that my hostel in Kampar does not have a strong Wi-Fi connection, which make it taking longer time to wait for the debugging to process as the proposed system uses a cloud database, so that the system has to wait until the connection is stable then it only can start debugging or running the system. The reason that a stable Wi-Fi connection is important is because the system has to retrieve all data from the Firebase Authentication, Firebase Realtime Database and Firebase Storage. The second challenge is that API is not freely available. Although there are a lot of open sources out there, but one of the APIs called Google Places API is what it originally planned to implement in the system, but this Places API is not free, even the developer purchased it, he/she might need to pay for extra if the developer has exceeded the call Places API limitations. That's why the system changed to use Google Maps Intents for Android to open the Google Maps application instead of showing the system's built-in Maps. The third challenge is that always needed to check and update the code to the latest. The Firebase Realtime Database structure changed suddenly on April 12, 2022, causing a function named "getChildrenCount()" cannot correctly count the number of children in the Firebase. For example, there are 2 loan applications in pending status, 1 loan application in approved status, and 1 loan application in rejected status, so supposedly pending result count should be 2, approved result count should be 1 and rejected results count should also be 1, but the Firebase returns results like pending result count is 0, approved result count is also 0, and rejected result count is 4 which are wrong. Therefore, I need to update and rethink some of the code for few activities.



#### **6.4 Objective Evaluation**

The proposed system achieves its objectives by successfully allowing users to easily access banking services even if they have not opened any bank accounts, but users can use the proposed system to register as a new user without filling out all physical documents and queuing up in front of the incumbent bank. After done registering as a new user, the user can directly enjoy all the banking services provided by this proposed system regardless of time and place. The proposed system successfully allows users to apply for any loan they want, and the bank manager will update the loan application within 1-2 business days, sometimes even within a few hours the loan application will be approved. The proposed system allows users to use loans to start a new business, or use loans to get better education. The proposed system does explicitly mention on the funds transfer interface there that it will not charge any fees for intrabank transactions and even interbank transactions. Besides that, the proposed system will let users know the loan interest rate and total interest charged amount before they apply for a loan. In order to provide financial advice or enhance customers experience, the proposed system provided a feature called “Budgeting” to let users have a guidance what percentage of budget set they have used.

## **CHAPTER 7 CONCLUSION AND RECOMMENDATION**

### **7.1 Conclusion**

The proposed project is a digital-only bank mobile application. The purpose of this proposed project is to allow more and more customers easily obtain financial services through mobile application. That is because for the traditional bank, the customer without going to physical branch is unable to open an account in order to access the financial services provided by traditional bank. As we all know that not all the area in Malaysia does provide traditional bank especially for the rural area. It eventually leads to there are a lot of unbanked and underbanked exists in Malaysia. By using the proposed project, financial inclusion can help to bridge the gaps by allowing households and businesses easily to have their loan with a low interest rate, so that their children can get better education. Furthermore, they can have some money to start to operate SME business which eventually it will help to positively address the impact of financial on economic growth. Not only that, but the financial inclusion can also help to reduce the social inequality. The difference between the proposed project is the proposed project does provide transparent in fund transfer transactions because we all know that hidden transfer fee can hurt the relationship between the customer and the bank. The most importing thing is emphasizing a strong customer value proposition (CVP) by allowing the customer to keep track of their spending, so that they can start to control in order to avoid overdraft happening. And also, the proposed project can let the customer to set budget limit for each category.

I have analyzed further on the other digital-only banking mobile application such as Monzo, Monese, Revolut and N26. I did highlight the functionalities, features, strengths, and weaknesses of the digital-only banking mobile applications together with my proposed project. After analyzing all these digital-only banking mobile applications, I got an idea what types of functionalities and features other digital-only banking mobile applications have and do not have, so that I can include features those others do not have in my proposed system. It eventually can help me to become more competitive in the market. Furthermore, before I started developing my proposed system, I prepared a questionnaire to understand respondents' user experience with banking services from other banks, as well as user expectations for using the digital-only banking mobile application. After the development of my proposed system was completed, I sent another questionnaire to the 15 respondents to find out whether their

expectations were met, whether the application was easy to use, areas for improvement, and so on. The purpose to do so to understand which aspects of the system can be improved or better.

## **7.2 Recommendation**

The system evaluation survey is used to identify room for improvement in the proposed system. Even though the proposed system provides basic functionality, different users will have different viewpoints, and different viewpoints can help me understand what functionality is missing from the proposed system they think it is important to include it, or that any part of the proposed system does not work properly. According to the system evaluation survey results, cross-board payments, split billing and fingerprint login are the features that most respondents recommend to include it into the system. Another recommendation, such as when a user makes an online purchase using the proposed system, the system can directly classify the item from which category in the application, and also deduct the available balance of the account at the same time. More recommended examples, such as favorite accounts, pop-up notifications, displaying loan amortization schedule, multi-currency accounts, adding prepaid top-ups and postpaid phone bill in the Pay Bills list, adding daily and monthly analysis of budget. All of these suggestions will be carefully considered in future updated versions of the application.

**References**

- [1] V. Sharma, "What is FinTech and where does it Live?," 2016. [Online]. Available: <https://www.finextra.com/blogposting/12890/what-is-fintech-and-where-does-it-live>. [Accessed 6 June 2021].
- [2] R. Crook, "ARE DIGITAL-ONLY BRANDS THE FUTURE OF FINANCIAL SERVICES?," Somo, p. 4, 2019.
- [3] A. Mankotia, "10 Types of Financial Services Offered in India," 2019. [Online]. Available: <https://www.investindia.gov.in/team-india-blogs/10-types-financial-services-offered-india>. [Accessed 6 June 2021].
- [4] G. Kaur, "Are consumers ready to switch to virtual banks?," The Star, 7 December 2019.
- [5] C. Ong, M. v. Engel and K. Lee, "Virtual Banking: Malaysian customers take charge," PwC, Malaysia, 2019.
- [6] K. H. Lum, "COVID-19: Opportunities and Challenges for Digital Banks," PwC, Malaysia, 2020.
- [7] C. Skinner, "Digital Bank : Strategies to Launch or Become a Digital Bank," Digital Bank, p. 316, 2014.
- [8] P. Ozili, "Impact of digital finance on financial inclusion and stability," Borsa Istanbul Review, vol. 18, no. 4, pp. 329-340, 2018.
- [9] R. Han and M. Melecky, "Financial inclusion for financial stability : access to bank deposits and the growth of deposits in the Global Financial Crisis," Policy Research Working Paper Series, 2013.
- [10] E. Prasad, "Financial Sector Regulation and Reforms in Emerging Markets: An Overview," IZA Discussion Papers, No. 5233, 2010.

## REFERNCES

- [11] S. Cecchetti and E. Kharroubi, "Reassessing the impact of finance on growth," Bank for International Settlements, p. 17, 2012.
- [12] A. V. Komarov and V. M. Martyukova, "Neobanking as a direction of development of modern financial technologies," *Vestnik Universiteta*, pp. 134-142, 2020.
- [13] C. Ratcliffe, S. McKernan, E. Kalish and S. Martin, "Where Are the Unbanked and Underbanked in New York City," *URBAN INSTITUTE*, vol. 25, p. 1, 2015.
- [14] T. a. B. & C. Google, "Fulfilling its Promise—The future of Southeast Asia's digital financial services," Google, Temasek and Bain & Company, Indonesia, Malaysia, the Philippines, Singapore, Thailand and Vietnam, 2019.
- [15] malaysiakini, "SAP: Digital banking poised to revolutionise financial services sector in Southeast Asia, Malaysia," *malaysiakini*, 22 April 2021.
- [16] A. Hilao, "Bank Negara Malaysia Charting the Course for Digital Banking," *InsideSAP Asia*, 26 April 2021.
- [17] BigPay, "All you need to know about international money transfers," 2019. [Online]. Available: <https://www.bigpayme.com/post/international-money-transfers>. [Accessed 6 June 2021].
- [18] S. Cocheo, "Digital Transformation in Banking Requires Complete Rewiring, Not Just Fresh Paint," 2018. [Online]. Available: <https://thefinancialbrand.com/76621/digital-banking-data-analytics-gafa/>. [Accessed 6 June 2021].
- [19] M. Spangenberg, "Why the future belongs to banks, and not Fintechs," 2021. [Online]. Available: <https://www.finextra.com/blogposting/20083/why-the-future-belongs-to-banks-and-not-fintechs>. [Accessed 6 June 2021].

## REFERENCES

- [20] J. J. Kaos and T. Tan, "Best way to stop Covid-19 wave: Stay at home, says Health DG," *TheStar*, 19 October 2020.
- [21] J. Manyika, S. Lund, J. Bughin, J. Woetzel, K. Stamenov and D. Dhingra, "Digital globalization: The new era of global flows," McKinsey Global Institute, New York, 2016.
- [22] R. Sahay, U. E. Allmen, A. Lahreche, P. Khera, S. Ogawa, M. Bazarbash and K. Beaton, "The Promise of Fintech: Financial Inclusion in the Post COVID-19 Era," Monetary and Capital Markets Department, pp. 1-8, 2020.
- [23] M. A. Omar and K. Inaba, "Does financial inclusion reduce poverty and income inequality in developing countries? A panel data analysis," *Economic Structures*, no. 37, p. 9, 2020.
- [24] A. Casey, "With Alternative Data, Global Financial Inclusion Is More Attainable," 2019. [Online]. Available: <https://urjanet.com/blog/alternative-data-global-financial-inclusion/>. [Accessed 6 June 2021].
- [25] N. Popper, "The 'Neo-Banks' Are Finally Having Their Moment," *The New York Times*, 20 November 2018.
- [26] P. Francis, "Digital Banking in Malaysia: Are incumbent banks the real winners of the COVID-19 digital revolution?," PwC, Malaysia, 2020.
- [27] S. Patel, "8 Trends Transforming Digital Customer Experience in Banking," 2021. [Online]. Available: <https://www.revechat.com/blog/customer-experience-in-banking/>. [Accessed 6 June 2021].
- [28] Syahirah Syed Jaafar, "Many Malaysians say not financially prepared for an extended MCO — survey," *The Edge Malaysia*, 9 April 2020.
- [29] I. M. F. a. W. Bank, *Fintech: The Experience So Far*, International Monetary Fund, 2019.

## REFERNCES

- [30] E. Eric, "Investopedia," Investopedia, 4 April 2022. [Online]. Available: <https://www.investopedia.com/terms/o/open-banking.asp#citation-2>. [Accessed 5 April 2022].
- [31] Tom Blomfield, Jonas Huckestein, Jason Bates, Paul Rippon and Gary Dolman, *Monzo* (Version 3.60.0) [Mobile App]. Monzo Bank Ltd, 13 May 2016. [Online]. [Accessed 18 November 2020].
- [32] R. Cook, "Monzo," 2018. [Online]. Available: <https://monzo.com/blog/2018/08/21/budget-with-monzo>. [Accessed 8 June 2021].
- [33] M. Ali, K. M. Iraqi, A. S. Rawat and S. Mohammad, "Role of Customer Service Skills on Customer Satisfaction and Its Effects on Customer Loyalty in Pakistan Banking Industry," *South Asian Journal of Management Sciences*, vol. 12, pp. 210-223, 2018.
- [34] N. Koppel, *Monese* (Version 6.35.0.21485) [Mobile App]. Monese Ltd, 21 September 2015. [Online]. Available: <https://monese.com/gb/en>. [Accessed 16 November 2020].
- [35] Monica, "Monese," 2019. [Online]. Available: <https://monese.com/gb/en/blog/simple-spending-overview>. [Accessed 9 June 2021].
- [36] N. Gundaniya, "Digipay.guru," 2019. [Online]. Available: <https://www.digipay.guru/blog/fintech-impact-on-international-remittance/>. [Accessed 9 June 2021].
- [37] Nikolay Storonsky and Vlad Yatsenko, *Revolut* (Version 7.27) [Mobile app]. Revolut Ltd, 1 July 2015. [Online]. Available: <https://www.revolut.com/>. [Accessed 17 November 2020].
- [38] Revolut.com, "Revolut," [Online]. Available: <https://www.revolut.com/>. [Accessed 9 June 2021].

## REFERNCES

- [39] HardReset.info, "HardReset.info," 2021. [Online]. Available: <https://www.hardreset.info/devices/apps/apps-revolut/send-money/>. [Accessed 9 June 2021].
- [40] A. Smith, "Revolut rolls out QR code-based payments for business customers," Finextra, 18 February 2021.
- [41] Valentin Stalf and Maximilian Tayenthal, *N26* (Version 3.53.1) [Mobile app]. N26 Bank GmbH, April 2015. [Online]. Available: <https://n26.com/en-eu>. [Accessed 15 November 2020].
- [42] T. Groenfeldt, "Forbes," 2019. [Online]. Available: <https://www.forbes.com/sites/tomgroenfeldt/2019/02/05/n26-will-bring-successful-european-digital-banking-platform-to-the-u-s/?sh=27dcfa2d6f8b>. [Accessed 22 November 2020].
- [43] javatpoint, "Prototype model," JavaTpoint, [Online]. Available: <https://www.javatpoint.com/software-engineering-prototype-model>. [Accessed 25 November 2020].
- [44] E. Lam, "Public Bank Branch Codes," Blogger, 5 January 2015. [Online]. Available: <http://alleyesonally.blogspot.com/2015/01/public-bank-branch-codes.html>. [Accessed 28 March 2022].



## APPENDIX

## System Pre-Authoring Analysis Survey Form

# Mobile Banking Customer Experience Survey

Read before proceeding!

Hello, I'm Chua Huey Mei who is an undergraduate student from Universiti Tunku Abdul Rahman (UTAR). This questionnaire is prepared to let me study more about customer experience on using mobile banking services as it is a part of my Final Year Project. This questionnaire consists of 15 questions and it will only take you about 15 minutes. Your effort and time to answer the questionnaire will be much appreciated.

\*Required

1. Name \*

Your answer

---

2. How old are you? \*

- Below 18 years old
- 18-30 years old
- 30-40 years old
- 40-50 years old
- 50-60 years old
- 60 years old and above

3. What is your gender? \*

Female

Male

4. Where are you located? \*

Perlis

Kedah

Penang

Perak

Kuala Lumpur

Selangor

Negeri Sembilan

Melaka

Johor

Sabah

Sarawak

Kelantan

Terengganu

Other: \_\_\_\_\_

5. What is your Highest Education completed? \*

- Secondary school
- Diploma
- Bachelor Degree
- Master
- PHD

6. What is your marital status? \*

- Single
- Married
- Divorced
- Widowed
- Separated

7. What is your occupation? \*

- Student
- Employed for wages
- Self employed
- Out of work and looking for work
- Out of work but currently not looking for work
- Homemaker

Military

Retired

8. Based on the answer to Question 7, what industry sector do/did you work in?  
[Tips: Student and Homemaker can skip this question]

Agriculture

Construction

Education

Entertainment

Finance and economic

IT

Food and beverage

Health care/Hospitality/Pharmaceutical

Other: \_\_\_\_\_

9. What is your income? \*

RM1,000 and below

RM1,000 - RM4,999

RM5,000 - RM9,999

RM10,000 - RM14,999

RM15,000 - RM29,999

RM40,000 and above

10. On a scale of 1-5 (with 1 being 1 out of 5 mark and 5 being 5 out of 5 mark ), please give an overall score for the digital banking services you have visited before based on your satisfaction with speed, experience or usability \*

	1	2	3	4	5
Quick access to digital banking services	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
24/7 Customer service	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Fund transfer	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Loan management	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Bill payments	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
QR function to scan and receive fund	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

11. What are your favorite 3 transactions in digital banking? Please select 3 that apply. \*

- Paying for a purchase online
- Instant Peer-to-Peer payment
- Available 24/7 customer service
- Apply Loan
- Checking/savings account management
- Transaction records monitoring

- Bill payments
- Loan management
- Personal finance planning

12. What is the main benefit of digital banking that you think will help you a lot? \*

- Cost savings
- No Hidden Cost
- Improved usability
- Greater personalization
- Other: \_\_\_\_\_

13. What are the benefits that you think digital banking can give to community? \*

- Get rid of poverty and reduce social inequality
- Provide tailored financial advice
- Getting a loan for a startup
- Get better education for children

14. On a scale of 1-5, please give a score on how likely are you to try the budgeting feature if it is available in digital banking services? \*

- |               |                       |                       |                       |                       |                       |             |
|---------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-------------|
|               | 1                     | 2                     | 3                     | 4                     | 5                     |             |
| Very Unlikely | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | Very Likely |

15. What might be preventing you from using a standalone budgeting application? \*

- Linking to bank account is not secure
- Didn't find a good budgeting application yet
- The budgeting application is not free
- Don't have enough money to budget
- Don't like tracking my spending
- Other: \_\_\_\_\_

**Submit** Clear form

Never submit passwords through Google Forms.

This content is neither created nor endorsed by Google. [Report Abuse](#) - [Terms of Service](#) - [Privacy Policy](#)

**Google Forms**

## System Evaluation Survey Form

# Banky Customer Satisfaction Survey

Read before proceeding!

Good day, I'm Chua Huey Mei who is an undergraduate student from Universiti Tunku Abdul Rahman (UTAR). By filling out this 10-minute system evaluation survey, you will help us identify issues and provide you with the best quality product and service. Your effort and time to answer the questionnaire will be much appreciated.

\*Required

1. Name \*

Your answer

2. Did the onboarding screen (guide to using the Banky mobile application) catch your attention? \*

Yes

No

3. Using the scale below, please indicate your level of overall satisfaction with your experiences at this Banky application. \*

Extremely Dissatisfied      1      2      3      4      5      Extremely Satisfied



4. Consider your experience with SCREEN LAYOUTS when using the Banky application. Please rate that most appropriately reflects your impression of using the application. \*

	1	2	3	4	5	
Confusing	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Clear

5. Consider your experience with NAVIGATION BUTTON when using the Banky application. Please rate that most appropriately reflects your impression of using the application. \*

	1	2	3	4	5	
Confusing	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Clear

6. Consider your experience with the Banky application after using the Banky application. Please rate your agreement with the following statements about this application. \*

	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
Information on the Banky application is clear and easy to understand	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
I can easily find what I'm looking for on the Banky application	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

The Banky application includes all of the information I need.

The Banky application works well technically, loading quickly and display appropriately.

The Banky application meets my needs

7. Which of the following best describes your primary role in using the application? \*

- Check Account Balances (Available Balance, Current Balance, Fixed Deposit, Balance)
- Update Overdraft/ Withdrawal Limit
- Fund Transfer
- Request Fund
- Pay Bill
- Apply Loan
- Loan Repayment
- Budgeting
- QRScan/ QRRecive/ QRPay
- Email/Phone Call/WhatsApp Reply

- Email/Phone Call/WhatsApp Banky
- Check Nearby ATM locations

8. Please provide suggestions on how we can improve our service to you \*

Your answer

---

9. Considering your complete experience with our banking services, how likely is that you would recommend this Banky application to your family, friends or colleague? \*

	1	2	3	4	5	
Extremely Unlikely	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Extremely Likely

**Submit**

Clear form

Never submit passwords through Google Forms.

This form was created inside Universiti Tunku Abdul Rahman. [Report Abuse](#)

Google Forms

**WEEKLY LOG**  
**FINAL YEAR PROJECT WEEKLY REPORT**  
*(Project II)*

<b>Trimester, Year: T3, Y3</b>	<b>Study week no.: 4</b>
<b>Student Name &amp; ID: Chua Huey Mei 18ACB04692</b>	
<b>Supervisor: Mr. Su Lee Seng</b>	
<b>Project Title: Digital-only Bank Mobile Application Development in Malaysia: Enhancing Customer Experience</b>	

**1. WORK DONE**

[Please write the details of the work done in the last fortnight.]

-I have completed Chapter 1 Introduction and Chapter 2 Literature Review in the report.

-Done the development of the onboarding screen, splash screen, sign in screen, sign out screen, main customer screen, profile edit screen, change password screen, my account screen, transfer funds within bank or to other bank screen, pay bills screen, display payments and transactions history screen in my digital-only banking mobile application for customer interface.

-Done the development of the main screen of the bank manager interface without any function.

**2. WORK TO BE DONE**

-Continue to complete the rest of the Chapter 3 System Methodology/Approach in the report.

-Continue to code the customer interface QR code scanning, receiving and payment interface, and design the bank manager interface function.

**3. PROBLEMS ENCOUNTERED**

-No problems encountered in Week 4 yet.

**4. SELF EVALUATION OF THE PROGRESS**

- Still need to read more open source to inspect, modify and enhance the functions to be included in my digital-only banking mobile application.



Supervisor's signature



Student's signature

# FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

<b>Trimester, Year: T3, Y3</b>	<b>Study week no.: 6</b>
<b>Student Name &amp; ID: Chua Huey Mei 18ACB04692</b>	
<b>Supervisor: Mr. Su Lee Seng</b>	
<b>Project Title: Digital-only Bank Mobile Application Development in Malaysia: Enhancing Customer Experience</b>	

## 1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- I have completed Chapter 3 System Methodology/Approach in the report.
- Done the development of using QR code to scan, receive or pay, adjusting the “My Account” screen by adding able to let customer to edit or update the overdraft or withdrawal limit in my digital-only banking mobile application for customer interface.
- Done the development of the manage customer screen and the contact customer screen in my digital-only banking mobile application for the bank manager interface.

## 2. WORK TO BE DONE

- Continue to complete the Chapter 4 System Design in the report.
- Continue to code the apply loan screen, the FAQ screen, the contact bank screen and the nearby ATM screen in the customer interface and the manage loan in the bank manager interface.

## 3. PROBLEMS ENCOUNTERED

- No problems encountered in Week 6 yet.

## 4. SELF EVALUATION OF THE PROGRESS

- Still need to read more open source to inspect, modify and enhance the functions to be included in my digital-only banking mobile application.



Supervisor's signature



Student's signature

# FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

<b>Trimester, Year: T3, Y3</b>	<b>Study week no.: 8</b>
<b>Student Name &amp; ID: Chua Huey Mei 18ACB04692</b>	
<b>Supervisor: Mr. Su Lee Seng</b>	
<b>Project Title: Digital-only Bank Mobile Application Development in Malaysia: Enhancing Customer Experience</b>	

## 1. WORK DONE

[Please write the details of the work done in the last fortnight.]

-I have completed Chapter 4 System Design in the report.

-Done the development of the apply loan screen, the FAQ screen, the contact bank screen and the nearby ATM screen in my digital-only banking mobile application for customer interface.

-Done the development of the manage loan screen in my digital-only banking mobile application for the bank manager interface.

## 2. WORK TO BE DONE

-Continue to complete the Chapter 5 System Implementation & Chapter 6 System Evaluation and Discussion in the report.

-Continue to code the request funds screen in the customer interface and the manage account and manage report in the bank manager interface.

## 3. PROBLEMS ENCOUNTERED

-No problems encountered in Week 8 yet.

## 4. SELF EVALUATION OF THE PROGRESS

- Still need to read more open source to inspect, modify and enhance the functions to be included in my digital-only banking mobile application.



Supervisor's signature



Student's signature

# FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

<b>Trimester, Year: T3, Y3</b>	<b>Study week no.: 10</b>
<b>Student Name &amp; ID: Chua Huey Mei 18ACB04692</b>	
<b>Supervisor: Mr. Su Lee Seng</b>	
<b>Project Title: Digital-only Bank Mobile Application Development in Malaysia: Enhancing Customer Experience</b>	

## 1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- I have completed Chapter 5 System Implementation in the report.
- Done the development of the request funds screen in my digital-only banking mobile application for customer interface.
- Done the development of the manage account screen and half of the manage report screen in my digital-only banking mobile application for the bank manager interface.

## 2. WORK TO BE DONE

- Continue to complete the rest of the Chapter 6 System Evaluation and Discussion and Chapter 7 Conclusion and Recommendation in the report.
- Continue to code the budgeting function in the customer interface and the rest of the manage report in the bank manager interface.

## 3. PROBLEMS ENCOUNTERED

- No problems encountered in Week 10 yet.

## 4. SELF EVALUATION OF THE PROGRESS

- Still need to read more open source to inspect, modify and enhance the functions to be included in my digital-only banking mobile application.



\_\_\_\_\_  
Supervisor's signature



\_\_\_\_\_  
Student's signature

## FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

<b>Trimester, Year: T3, Y3</b>	<b>Study week no.: 12</b>
<b>Student Name &amp; ID: Chua Huey Mei 18ACB04692</b>	
<b>Supervisor: Mr. Su Lee Seng</b>	
<b>Project Title: Digital-only Bank Mobile Application Development in Malaysia: Enhancing Customer Experience</b>	

### 1. WORK DONE

[Please write the details of the work done in the last fortnight.]

-I have completed Chapter 6 System Evaluation And Discussion and Chapter 7 Conclusion and Recommendation in the report.

-Done the development of the budgeting screen includes adding budget limit, adding spending item and overview of spending in my digital-only banking mobile application for customer interface.

-Done the development of the manage report screen in my digital-only banking mobile application for the bank manager interface.

### 2. WORK TO BE DONE

-Continue debugging all smart parts & reformat codes in the Android Studio.

### 3. PROBLEMS ENCOUNTERED

-No problems encountered in Week 12 yet.

### 4. SELF EVALUATION OF THE PROGRESS

- Still need to read more open source to inspect, modify and enhance the functions to be included in my digital-only banking mobile application.



Supervisor's signature



Student's signature



POSTER

# Digital-only Bank Mobile Application Development in Malaysia: Enhancing Customer Experience

In this environment, traditional banks confront a major problem in sustaining and growing their business in the face of digital disruption and the rise of new client expectations. The term "digital-only bank" is introduced which refers to the development of banking services and the delivery of products and services via mobile devices without needing a physical branch to operate.

## Objectives

- To develop a digital-only bank that operates through a mobile application allowing customers to easily obtain financial services
- To build a digital-only banking mobile application to positively address the impact of financial inclusion on economic growth
- To construct a digital-only bank that operates via a mobile application and provides transparency in fund transfer transactions
- To launch a digital-only banking mobile application enhances customers' experience by leveraging technology

## Conclusion

By using the Banky digital-only banking mobile application, more and more Malaysians can easily access financial services, even if they are from rural areas, and ultimately financial inclusion can help to bridge the gaps by allowing households and businesses to easily obtain their loan with a low-interest rate, and positively address the impact of finance on economic growth as will reduce the social inequality.

## Project Discussion

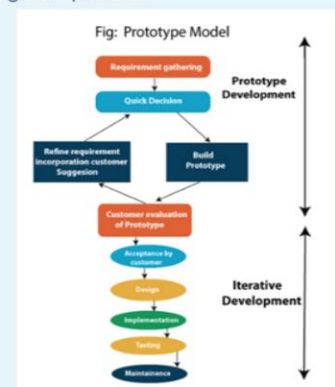
In this project, a new digital-only banking mobile application is named "Banky" because it carries the meaning of "This Digital-Only Bank is Trusty", since it is a digital-only bank, and it just took the "y" from the "Trusty" word and renamed it to "Banky".

Banky's main functions for customers are **Account, Transfer, Pay Bills, Apply Loan, and Budgeting.**

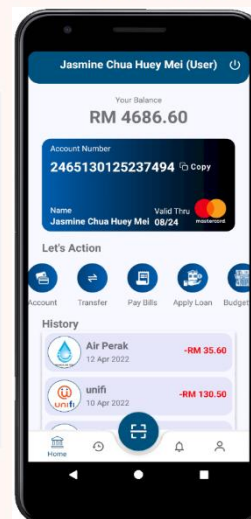
Banky's main functions for the bank manager are **Manage Customer, Manage Account, Manage Loan, Manage Report, and Contact Customer.**

## Methodology - Prototype Model

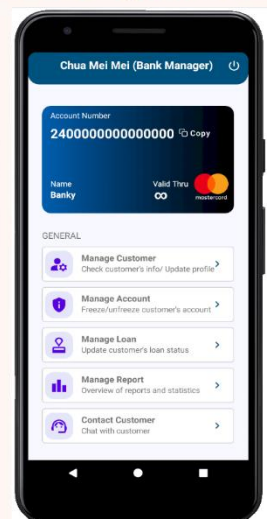
- Step 1: Requirement gathering and analysis
- Step 2: Quick decision
- Step 3: Build prototype
- Step 4: Assessment or user evaluation
- Step 5: Prototype refinement
- Step 6: Engineer product



### Customer Interface



### Bank Manager Interface



Universiti Tunku Abdul Rahman

BACHELOR OF INFORMATION SYSTEMS (HONOURS)  
BUSINESS INFORMATION SYSTEM

Prepared by Chua Huey Mei (18ACB04692)

## PLAGIARISM CHECK RESULT

Feedback Studio - Google Chrome  
 ev.turnitin.com/app/carta/en\_us/?s=8student\_user=1&lang=en\_us&o=1815281103&u=1119674818

feedback studio Huey Mei Chua | FYP2

Match Overview

8%

Match	Source	Similarity
1	eprints.utar.edu.my Internet Source	2%
2	Submitted to Asia Paci... Student Paper	1%
3	semsub.epa.gov Internet Source	1%
4	Mohamed Farouk. "Infr... Publication	<1%
5	www.theseus.fi Internet Source	<1%
6	lo-motion.de Internet Source	<1%
7	Submitted to Nottingha... Student Paper	<1%
8	Submitted to University... Student Paper	<1%
9	Submitted to University... Student Paper	<1%

Page: 1 of 190 Word Count: 33887 Text-Only Report High Resolution On

**CHAPTER 1 INTRODUCTION**

**1.1 Background Information**

Financial technology, which is also known as fintech, is an economic field consisting of companies that make financial services more productive by using technology [1]. The use of innovative technology-based technologies for financial services is at the core of Fintech. Fintech refers to technologies designed to provide financial services in competition with traditional financial approaches. Fintech will introduce the impact of inclusive finance on economic growth in order to reduce the population with unbanked and underbanked individuals and lift them out of poverty and social inequality. Besides that, traditional banking companies, alternatively, are hampered by legacy operating systems, innovation capability, agility and experience in technology. The Covid-19 crisis has become an unexpected catalyst to stimulate Malaysia's digital banks, despite the digital disruption to the banking industry. To achieve a cashless society in Malaysia, the digital-only bank can help to encourage all sections of society to use electronic payments more.

# PLAGIARISM CHECK RESULT

FYP2

---

ORIGINALITY REPORT

---

8%

SIMILARITY INDEX

6%

INTERNET SOURCES

2%

PUBLICATIONS

5%

STUDENT PAPERS

---

PRIMARY SOURCES

---

1	<a href="http://eprints.utar.edu.my">eprints.utar.edu.my</a> Internet Source	2%
2	Submitted to Asia Pacific University College of Technology and Innovation (UCTI) Student Paper	1%
3	<a href="http://semspub.epa.gov">semspub.epa.gov</a> Internet Source	1%
4	Mohamed Farouk. "Infrastructure Software Modules for Enterprises", Springer Science and Business Media LLC, 2017 Publication	<1%
5	<a href="http://www.theseus.fi">www.theseus.fi</a> Internet Source	<1%
6	<a href="http://lo-motion.de">lo-motion.de</a> Internet Source	<1%
7	Submitted to Nottingham Trent University Student Paper	<1%
8	Submitted to University of Northumbria at Newcastle Student Paper	<1%

---

PLAGIARISM CHECK RESULT

9	Submitted to University of Greenwich Student Paper	<1 %
10	Submitted to University of Hertfordshire Student Paper	<1 %
11	Submitted to Liverpool John Moores University Student Paper	<1 %
12	Submitted to Queen Mary and Westfield College Student Paper	<1 %
13	Submitted to National College of Ireland Student Paper	<1 %
14	Derick Ndimbwa, Abubaker Qutieshat. "A Synthesis on Enablers of Digital Finance in Emerging Economies. Lessons from Zambia.", Research Square Platform LLC, 2022 Publication	<1 %
15	Submitted to Edith Cowan University Student Paper	<1 %
16	Submitted to Zambia Centre for Accountancy Studies Student Paper	<1 %
17	<a href="http://www.coursehero.com">www.coursehero.com</a> Internet Source	<1 %
18	Submitted to Universiti Teknologi MARA Student Paper	<1 %

PLAGIARISM CHECK RESULT

19	forums.tomshardware.com Internet Source	<1 %
20	fict.utar.edu.my Internet Source	<1 %
21	Peterson K. Ozili. "Impact of digital finance on financial inclusion and stability", Borsa Istanbul Review, 2018 Publication	<1 %
22	Submitted to City University of Hong Kong Student Paper	<1 %
23	Submitted to HELP UNIVERSITY Student Paper	<1 %
24	Submitted to UNITEC Institute of Technology Student Paper	<1 %
25	Submitted to University of Technology, Sydney Student Paper	<1 %
26	www.kashifali.ca Internet Source	<1 %
27	digitalcommons.lasalle.edu Internet Source	<1 %
28	eprints.utm.my Internet Source	<1 %
29	Submitted to Middle East College of Information Technology	<1 %

## PLAGIARISM CHECK RESULT

Student Paper		
30	Submitted to University of Sussex Student Paper	<1 %
31	Submitted to First City University College Student Paper	<1 %
32	Reto Meier, Ian Lake. "Professional Android®", Wiley, 2018 Publication	<1 %
33	openaccess.mef.edu.tr Internet Source	<1 %
34	Submitted to Sunway Education Group Student Paper	<1 %
35	Submitted to Segi University College Student Paper	<1 %
36	Sibeesh Venu. "Asp.Net Core and Azure with Raspberry Pi 4", Springer Science and Business Media LLC, 2020 Publication	<1 %
37	Siti Nor Katijah Mohd Jamil, Nur Fitri Shareena Abdul Rahim. "Mobile Educational Game Application for Middle School Student in Learning Bahasa Melayu (Peribahasa) Using Interactive Method", 2021 IEEE 9th Conference on Systems, Process and Control (ICSPC 2021), 2021 Publication	<1 %

## PLAGIARISM CHECK RESULT

38	<a href="http://brianschrank.com">brianschrank.com</a> Internet Source	<1 %
39	<a href="http://cse.iku.edu.tr">cse.iku.edu.tr</a> Internet Source	<1 %
40	Submitted to Lingnan University Student Paper	<1 %
41	Submitted to Prague College Student Paper	<1 %
42	<a href="http://staff.utar.edu.my">staff.utar.edu.my</a> Internet Source	<1 %
43	Submitted to Swinburne University of Technology Student Paper	<1 %
44	Submitted to Writtle Agricultural College Student Paper	<1 %
45	<a href="http://orca.cf.ac.uk">orca.cf.ac.uk</a> Internet Source	<1 %
46	Submitted to Cavite State University Student Paper	<1 %
47	Submitted to Hong Kong Baptist University Student Paper	<1 %
48	Submitted to Manchester Metropolitan University Student Paper	<1 %

## PLAGIARISM CHECK RESULT

49	Submitted to University Tun Hussein Onn Malaysia Student Paper	<1 %
50	e-theses.imtlucca.it Internet Source	<1 %
51	intractables.netlify.app Internet Source	<1 %
52	umpir.ump.edu.my Internet Source	<1 %
53	Submitted to Napier University Student Paper	<1 %
54	Submitted to University of New South Wales Student Paper	<1 %
55	Submitted to University of Reading Student Paper	<1 %
56	mafiadoc.com Internet Source	<1 %
57	www.bernama.com Internet Source	<1 %
58	infocouncil.aucklandcouncil.govt.nz Internet Source	<1 %
59	library.nhsggc.org.uk Internet Source	<1 %
60	play.google.com	



PLAGIARISM CHECK RESULT

	Internet Source	<1 %
61	<a href="http://www.hch.org.cn">www.hch.org.cn</a> Internet Source	<1 %
62	<a href="http://www.readbag.com">www.readbag.com</a> Internet Source	<1 %
63	Submitted to University of West London Student Paper	<1 %
64	<a href="http://www.packtpub.com">www.packtpub.com</a> Internet Source	<1 %
65	<a href="http://www.slideshare.net">www.slideshare.net</a> Internet Source	<1 %
66	Tobias Marktscheffel, Wolfram Gottschlich, Wolfgang Popp, Philemon Werli, Simon Dominik Fink, Arne Bilzhause, Hermann de Meer. "QR code based mutual authentication protocol for Internet of Things", 2016 IEEE 17th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2016 Publication	<1 %
67	<a href="http://developer.android.com">developer.android.com</a> Internet Source	<1 %
68	<a href="http://www.drugsandalcohol.ie">www.drugsandalcohol.ie</a> Internet Source	<1 %
69	<a href="http://www.lawyerment.com">www.lawyerment.com</a> Internet Source	<1 %
70	Nadezhda Kunicina, Anatolijs Zabasta, Antons Patlins, Ivana Bilic, Janis Peksa. "Prototyping process in education and science", 2020 IEEE 61th International Scientific Conference on Power and Electrical Engineering of Riga Technical University (RTUCON), 2020 Publication	<1 %

<b>Universiti Tunku Abdul Rahman</b>			
<b>Form Title : Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)</b>			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 01/10/2013	Page No.: 1 of 1



**FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY**

<b>Full Name(s) of Candidate(s)</b>	Chua Huey Mei
<b>ID Number(s)</b>	18ACB04692
<b>Programme / Course</b>	BACHELOR OF INFORMATION SYSTEMS (HONOURS) BUSINESS INFORMATION SYSTEMS
<b>Title of Final Year Project</b>	Digital-only Bank Mobile Application Development in Malaysia: Enhancing Customer Experience

<b>Similarity</b>	<b>Supervisor's Comments (Compulsory if parameters of originality exceeds the limits approved by UTAR)</b>
<b>Overall similarity index: <u>8</u> %</b>  <b>Similarity by source</b> Internet Sources: <u>6</u> % Publications: <u>2</u> % Student Papers: <u>5</u> %	
<b>Number of individual sources listed of more than 3% similarity: <u>0</u></b>	
<b>Parameters of originality required and limits approved by UTAR are as Follows:</b> (i) Overall similarity index is 20% and below, and (ii) Matching of individual sources listed must be less than 3% each, and (iii) Matching texts in continuous block must not exceed 8 words <i>Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.</i>	

Note Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

*Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.*

SuLee

\_\_\_\_\_  
Signature of Supervisor

Name: Su Lee Seng \_\_\_\_\_

Date: 15<sup>th</sup> April 2022 \_\_\_\_\_

\_\_\_\_\_  
Signature of Co-Supervisor

Name: \_\_\_\_\_

Date: \_\_\_\_\_

## FYP2 CHECKLIST



## UNIVERSITI TUNKU ABDUL RAHMAN

 FACULTY OF INFORMATION & COMMUNICATION  
 TECHNOLOGY (KAMPAR CAMPUS)  
 CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	18ACB04692
Student Name	Chua Huey Mei
Supervisor Name	Su Lee Seng

TICK (√)	DOCUMENT ITEMS
	Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
	Front Plastic Cover (for hardcopy)
√	Title Page
√	Signed Report Status Declaration Form
√	Signed FYP Thesis Submission Form
√	Signed form of the Declaration of Originality
√	Acknowledgement
√	Abstract
√	Table of Contents
√	List of Figures (if applicable)
√	List of Tables (if applicable)
√	List of Symbols (if applicable)
√	List of Abbreviations (if applicable)
√	Chapters / Content
√	Bibliography (or References)
√	All references in bibliography are cited in the thesis, especially in the chapter of literature review
√	Appendices (if applicable)
√	Weekly Log
√	Poster
√	Signed Turnitin Report (Plagiarism Check Result - Form Number: FM-IAD-005)
√	I agree 5 marks will be deducted due to incorrect format, declare wrongly the ticked of these items, and/or any dispute happening for these items in this report.

\*Include this form (checklist) in the thesis (Bind together as the last page)

I, the author, have checked and confirmed all the items listed in the table are included in my report.

(Signature of Student)

Date: 15<sup>th</sup> April 2022