



Faculty of Computer Science and Information Technology

BUDGETING MOBILE APPLICATION WITH COMPUTER VISION

Wong Rou Yi

Bachelor of Computer Science with Honours

(Software Engineering)

2023

BUDGETING MOBILE APPLICATION WITH COMPUTER VISION

WONG ROU YI

This project is submitted in partial fulfillment of the
requirements for the degree of
Bachelor of Computer Science with Honours (Software Engineering)

Faculty of Computer Science and Information Technology

UNIVERSITI MALAYSIA SARAWAK

2023

APLIKASI MUDAH ALIH BELANJAWAN DENGAN VISI KOMPUTER

WONG ROU YI

Projek ini merupakan salah satu keperluan untuk Ijazah
Sarjana Muda Sains Komputer dengan Kepujian (Kejuruteraan Perisian)

Fakulti Sains Komputer dan Teknologi Maklumat

UNIVERSITI MALAYSIA SARAWAK

2023

UNIVERSITI MALAYSIA SARAWAK

THESIS STATUS ENDORSEMENT FORM

TITLE BUDGETING MOBILE APPLICATION WITH COMPUTER VISION

ACADEMIC SESSION: 2022/2023

WONG ROU YI

(CAPITAL LETTERS)

hereby agree that this Thesis* shall be kept at the Centre for Academic Information Services, Universiti Malaysia Sarawak, subject to the following terms and conditions:

1. The Thesis is solely owned by Universiti Malaysia Sarawak
2. The Centre for Academic Information Services is given full rights to produce copies for educational purposes only
3. The Centre for Academic Information Services is given full rights to do digitization in order to develop local content database
4. The Centre for Academic Information Services is given full rights to produce copies of this Thesis as part of its exchange item program between Higher Learning Institutions [or for the purpose of interlibrary loan between HLI]
5. ** Please tick (✓)

CONFIDENTIAL (Contains classified information bounded by the OFFICIAL SECRETS ACT 1972)

RESTRICTED (Contains restricted information as dictated by the body or organization where the research was conducted)

UNRESTRICTED



(AUTHOR'S SIGNATURE)

Validated by



(SUPERVISOR'S SIGNATURE)

Permanent Address

S/L 27, LORONG 1,
PINE DESA, DESA DAMAI,
97000 BINTULU, SARAWAK

DR TAN PING PING
Senior Lecturer (DS2)
Software Engineering Programme
Faculty of Computer Science & Information Technology
Universiti Malaysia Sarawak

Date: 17 JULY 2023

Date: 18 Jul, 2023

Note * Thesis refers to PhD, Master, and Bachelor Degree

** For Confidential or Restricted materials, please attach relevant documents from relevant organizations / authorities

DECLARATION

I hereby to declare that this work is entirely my own creation. No part of my report was prepared for me by another individual, nor have I plagiarized from any other student's work or other sources unless a reference or recognition is provided in the text.



.....
(WONG ROU YI)

1 JULY 2023

Matric No: 71970

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to Dr Tan Ping Ping who is my supervisor of the final year project for her supervision, support, and encouragement throughout the project. Her valuable knowledge and advice helped me a lot to complete this project and make it a success. Next, I would also like to thank the Final Year Project coordinator, Prof. Wang Yin Chai who provide valuable information and tips during lectures.

I would also like to thank the respondents who provided useful and constructive information and contributed to this project. Lastly, I would like to thank my family and friends for their moral support and encouragement.

ABSTRACT

Making transactions including spending or receiving money is an action that is done by youngsters and adults nowadays. He or she receives receipts after making a transaction. The transaction is recorded in order to track and manage their budget and money flow. Budgeting mobile application helps the group of people to record transactions. The normal budgeting application with normal manual input of data can be improved by adding computer vision to make the record transaction process easier. In this project, a budgeting application that applies computer vision was proposed and designed. It allows the application's user to scan the receipts and extract the data from the receipts. The technology used to extract the data of receipt is Azure Form Recognizer which is a specialized Optical Character Recognizer (OCR) service that helps to extract form-like data. The user is required to take a picture of a receipt then the data of the receipt would be extracted. This would help to save the time for the user to repeat the steps to record the transaction by entering the details such as name, amount, date and so on one by one. The tools used for development are Android Studio and Visual Studio Code. Azure Services including Azure Web App Service, Azure Database for MySQL flexible server and Azure Form Recognizer are utilized in this project to develop a web-hosting budgeting mobile application. After the development of the application is done, the real application was tested by 30 respondents and received positive feedback from the respondents where 86.7% of the respondents agreed that the receipt scanning feature and its ability to automatically fill in the transaction details were satisfactory. The respondents provided some suggestion regarding the improvement of the application as well and some of the suggestion was discussed at the end of this project. At the end of the project, the future work of the project was discussed as well.

ABSTRAK

Melakukan transaksi termasuk membelanja atau menerima wang adalah tindakan yang dilakukan oleh semua golongan muda and dewasa pada masa kini. Resit diterima selepas membuat transaksi. Transaksi-transaksi tersebut direkodkan untuk menguruskan kewangan mereka. Aplikasi mudah alih membantu kumpulan orang yang ingin merekod aliran wang merekodnya bila-bila masa. Dalam projek ini, sebuah aplikasi mudah alih yang menggunakan visi komputer untuk merekodkan belanja dan pendapatan telah dicadangkan dan direka bentuk. Ia membolehkan pengguna aplikasi belanjawan tersebut mengimbas resit dan mengekstrak data daripada resit tersebut supaya dapat menjimatkan masa pengguna mengulangi langkah merekod aliran wang dengan memasukkan data satu demi satu. Teknologi yang digunakan untuk mengekstrak data resit adalah *Azure Form Recognizer* yang merupakan salah satu perkhidmatan Pengesanan Aksara Optik (OCR) yang pakar dalam mengekstrak data dalam bentuk borang. Pengguna diminta untuk mengambil gambar resit kemudian data resit akan diekstrak. Ini akan membantu menjimatkan masa pengguna untuk mengulangi langkah-langkah untuk mencatat butiran seperti nama, jumlah, tarikh, dan sebagainya satu persatu. Alat yang digunakan untuk pembangunan projek ini adalah *Android Studio* dan *Visual Studio Code*. Perkhidmatan Azure termasuk *Azure Web App Service*, *Azure Database for MySQL flexible server*, dan *Azure Form Recognizer* digunakan dalam projek ini untuk membangunkan aplikasi mudah alih dalam projek ini. Selepas pembangunan aplikasi selesai, aplikasi telah diuji oleh 30 responden dan menerima maklum balas positif daripada responden di mana 86.7% responden bersetuju bahawa ciri pengimbasan resit dan keupayaannya untuk mengisi butiran transaksi secara automatik adalah memuaskan. Responden memberikan beberapa cadangan berkaitan penambahbaikan aplikasi dan beberapa cadangan telah dibincangkan di akhir projek ini. Di akhir projek ini, kerja masa depan projek juga dibincangkan.

TABLE OF CONTENTS

DECLARATION	i
ACKNOWLEDGEMENT	ii
ABSTRACT.....	iii
ABSTRAK.....	iv
TABLE OF CONTENTS.....	v
LIST OF FIGURES	ix
LIST OF DIAGRAMS.....	xii
LIST OF TABLES.....	xiii
CHAPTER 1: INTRODUCTION	1
1.1 Background	1
1.2 Problem Statement	2
1.3 Aims and Objectives	3
1.4 Scope	3
1.5 Methodology	3
1.5.1 Requirements planning	4
1.5.2 User Design.....	4
1.5.3 Construction.....	5
1.5.4 Implementation	5
1.6 Significance of Project	6
1.7 Project Schedule.....	7
1.8 Expected Outcome	8
1.9 Project Report Outline.....	8
1.9.1 Chapter 1: Introduction	8
1.9.2 Chapter 2: Literature Review.....	9
1.9.3 Chapter 3: Requirement Analysis and Design.....	9
1.9.4 Chapter 4: Implementation	10
1.9.5 Chapter 5: Testing.....	10

1.9.6	Chapter 5: Conclusion and Future Work	10
1.10	Summary	10
CHAPTER 2: LITERATURE REVIEW		11
2.1	Introduction	11
2.2	Reviews on Similar Existing Applications.....	11
2.2.1	Monefy.....	12
2.2.2	Easy Expense	15
2.2.3	Foreceipt	17
2.3	Comparison between the Existing Applications	20
2.4	Brief overview of Proposed Application.....	21
2.5	Review on tools and technologies.....	22
2.5.1	Visual Studio Code	22
2.5.2	Android Studio.....	22
2.5.3	MySQL	23
2.5.4	Computer Vision Read API from Microsoft Azure.....	23
2.5.5	Java	23
2.5.6	PHP	24
2.6	Conclusion.....	24
CHAPTER 3: REQUIREMENT ANALYSIS AND SYSTEM DESIGN.....		25
3.1	Introduction	25
3.2	Requirement Analysis	26
3.2.1	Questionnaire Session.....	26
3.2.1.1	User Background Information.....	27
3.2.1.2	User Experience on recording expenses.....	29
3.2.1.3	Feedback on Prototype	33
3.2.1.4	Conclusion of results	36
3.2.2	Functional Requirements	37

3.3	Design.....	38
3.3.1	Logical Design	38
3.3.1.1	Use Case Diagram.....	38
3.3.1.2	Use Case Descriptions.....	39
3.3.1.3	Sequence Diagram.....	56
3.3.1.4	Activity Diagram.....	67
3.3.1.5	Class Diagram	71
3.3.1.6	Database Diagram	72
3.3.2	Physical Design.....	73
3.3.2.1	Graphical User Interface Design	73
CHAPTER 4: IMPLEMENTATION		85
4.1	Introduction	85
4.2	Environment Setup.....	85
4.2.1	Android Studio.....	85
4.2.2	Microsoft Azure	86
4.2.2.1	App Service	86
4.2.2.2	Azure database for MySQL flexible server.....	88
4.2.2.3	Azure Form Recognizer	91
4.2.3	Visual Studio Code	93
4.3	User Interface Design.....	94
4.3.1	User Login and Register	94
4.3.2	Home Page	95
4.3.3	Add Transaction.....	98
4.3.4	Edit Transaction	103
4.3.5	Delete Transaction	103
4.3.6	Restore Transaction	104
4.3.7	Monthly Report.....	105

4.3.8	Income Tax Receipt Collection	106
4.3.9	User Profile and Setting	107
4.4	Summary	109
CHAPTER 5:	TESTING.....	110
5.1	Introduction	110
5.2	Unit Testing.....	110
5.3	Functional Testing.....	131
5.4	Usability Testing	136
5.4.1	User Background	136
5.4.2	User Experience	139
5.4.3	User Feedback.....	142
5.5	Summary	143
CHAPTER 6:	CONCLUSION AND FUTURE WORK.....	144
6.1	Introduction	144
6.2	Project Achievement	144
6.3	Project Limitation.....	146
6.4	Future Work	146
6.4.1	List Down Every Purchased Item of a Receipt.....	147
6.4.2	Decrease the Time Taken for Waiting for the Analyzed Result.....	147
6.4.3	Improve Receipt Scanning Feature	147
6.4.4	Set Monthly Spending Limit.....	148
6.4.5	Connect to E-wallet and Bank	148
6.5	Conclusion.....	148
REFERENCES	149
APPENDIX A:	QUESTIONNAIRE FOR REQUIREMENT ANALYSIS	151
APPENDIX B:	QUESTIONNAIRE FOR USABILITY TESTING	159

LIST OF FIGURES

Figure 1.1: Phases of Rapid Application Development, by C. Chien, 2020, Retrieved from https://codebots.com/app-development/what-is-rapid-application-development-rad	4
Figure 1.2: Gantt Chart for Final Year Project	7
Figure 2.1: Default Expense Categories in Monefy, by Monefy, n.d., Retrieved from https://monefy.me/	12
Figure 2.2: Dashboard of Monefy, by Monefy, n.d., Retrieved from https://monefy.me/	13
Figure 2.3: Image capturing and outcome of the key fields, by Easy Expense, n.d., Retrieved from https://easy-expense.com/	15
Figure 2.4: Dashboard of Foreceipt, by Foreceipt, n.d., Retrieved from https://easy-expense.com/	18
Figure 3.1: Respondents' gender	27
Figure 3.2: Respondents' age.....	27
Figure 3.3: Respondents' employment status	28
Figure 3.4: Respondents' salary range.....	28
Figure 3.5: Budgeting mobile application experience	29
Figure 3.6: Budgeting mobile applications used by respondents	29
Figure 3.7: Frequency of respondents recorded expenses	30
Figure 3.8: Methods of recording expenses	30
Figure 3.9: Opinion of respondents on the effectiveness of budgeting mobile application used to track their expenses.....	31
Figure 3.10: Respondents who kept the receipt after purchasing	31
Figure 3.11: Reasons of respondents keeping receipts	32
Figure 3.12: Perception of respondents on receipt scanning features in a budgeting mobile application.....	32
Figure 3.13: Respondents who think proposed budgeting application is easy to use.....	33
Figure 3.14: Missing information of expense in proposed budgeting mobile application	33
Figure 3.15: Missing information in monthly report	34
Figure 3.16: Respondents' expected features	34
Figure 3.17: Unnecessary features in proposed budgeting mobile application	35
Figure 3.18: Difficulty level of the proposed budgeting mobile application prototype	35
Figure 3.19: Sign Up Page	73
Figure 3.20: Login Page.....	74

Figure 3.21: Home Page	75
Figure 3.22: Transaction Detail Page.....	76
Figure 3.23: Select Page	77
Figure 3.24: Upload Picture Page	78
Figure 3.25: Check Receipt Detail Page	79
Figure 3.26: Monthly Report Page.....	80
Figure 3.27: Balance Amount Page	81
Figure 3.28: Income Tax Receipt Collection Page	82
Figure 3.29: Manage Profile Page	83
Figure 3.30: Setting Page.....	84
Figure 4.1: Overview of Web App	87
Figure 4.2: GitHub Repository of APIs	87
Figure 4.3: Overview of the deployed database.....	89
Figure 4.4: Set up new connection in MySQL Workbench.....	89
Figure 4.5: Connect to Azure Database	90
Figure 4.6: Overview of Azure Form Recognizer in Azure Portal.....	92
Figure 4.7: Connect to Azure Form Recognizer through REST API	92
Figure 4.8: User Interface of Visual Studio Code.....	93
Figure 4.9: Login and Sign up Page.....	94
Figure 4.10: User Home Page.....	95
Figure 4.11: Date Picker Dialog on Home Page.....	96
Figure 4.12: Transaction Detail Page.....	97
Figure 4.13: Select Page	98
Figure 4.14: Record Transaction Detail.....	99
Figure 4.15: Add Success Message	100
Figure 4.16: Receipt Detail Page	100
Figure 4.17: Transaction Detail Page after Scanning Receipt	101
Figure 4.18: Upload Image Without a Receipt and Error Message.....	102
Figure 4.19: Edit Success Message.....	103
Figure 4.20: Confirm Delete Message	103
Figure 4.21: Delete Success Message	104
Figure 4.22: Restore Transaction Page	104
Figure 4.23: Confirm Restore Message	104
Figure 4.24: Restore Success Message	105

Figure 4.25: Monthly Report Pages	105
Figure 4.26: Yearly Income Tax Record Overview.....	106
Figure 4.27: Setting Page.....	107
Figure 4.28: Profile Page	108
Figure 4.29: Edit Currency Dialog.....	108
Figure 5.1: Gender of Respondents	136
Figure 5.2: Age of Respondents.....	137
Figure 5.3: Employment Status of Respondents.....	137
Figure 5.4: Experience of Mobile Budgeting Application.....	138
Figure 5.5: Scale of Each Statement Part 1.....	140
Figure 5.6: Scale of Each Statement Part 2.....	140

LIST OF DIAGRAMS

Diagram 3.1: Use Case Diagram of Proposed Budgeting Mobile Application	38
Diagram 3.2: Sequence Diagram for Register New User	56
Diagram 3.3: Sequence Diagram for Login	57
Diagram 3.4: Sequence Diagram for Record Transaction	58
Diagram 3.5: Sequence Diagram for Upload and Scan Receipt	59
Diagram 3.6: Sequence Diagram for Edit Transaction	60
Diagram 3.7: Sequence Diagram for Delete Transaction	61
Diagram 3.8: Sequence Diagram for View Report	62
Diagram 3.9: Sequence Diagram for View Income Tax Receipt Collection.....	63
Diagram 3.10: Sequence Diagram for Restore Deleted Transaction	64
Diagram 3.11: Sequence Diagram for Set Currency	65
Diagram 3.12: Sequence Diagram for Manage Profile.....	66
Diagram 3.13: Activity Diagram of Save Transaction	67
Diagram 3.14: Activity Diagram of View Report	68
Diagram 3.15: Activity Diagram of Restore Deleted Transaction	68
Diagram 3.16: Activity Diagram for Income Tax Receipt Collection.....	69
Diagram 3.17: Activity Diagram for Manage Transaction	70
Diagram 3.18: Class Diagram of Proposed Budgeting Mobile Application	71
Diagram 3.19: Database Diagram of Proposed Budgeting Mobile Application	72
Diagram 5.1: Overall Impression of the Proposed Application.....	142

LIST OF TABLES

Table 2.1: Summary of comparison between similar existing applications and proposed budgeting mobile application.....	20
Table 3.1: Functional Requirements for Main User	37
Table 3.2: Use Case Description for Register New User.....	39
Table 3.3: Use Case Description for Login	41
Table 3.4: Use Case Description for Verify Email and Password.....	42
Table 3.5: Use Case Description for Record Transaction.....	44
Table 3.6: Use Case Description for Upload Receipt and Scan	46
Table 3.7: Use Case Description for Edit Transaction	48
Table 3.8: Use Case Description for Delete Transaction.....	49
Table 3.9: Use Case Description for View Monthly Report.....	50
Table 3.10: Use Case Description for View Yearly Income Tax Receipt Collection	51
Table 3.11: Use Case Description for Restored Deleted Transaction	52
Table 3.12: Use Case Description for Set Currency	53
Table 3.13: Use Case Description for Manage Profile	54
Table 5.1: Unit Test for Sign Up Page.....	111
Table 5.2: Unit Test for Login Page	114
Table 5.3: Unit Test for Home Page	116
Table 5.4: Unit Test for Monthly Summary	118
Table 5.5: Unit Test for Adding Transaction.....	122
Table 5.6: Unit Test for Transaction Detail Page	124
Table 5.7: Unit Test for Income Tax Receipt Collection Page.....	126
Table 5.8: Unit Test for Setting Page.....	128
Table 5.9: Unit Test for Profile Page	129
Table 5.10: Unit Test for Exit Dialog	130
Table 5.11: Test on Functions that are done by the User.....	131
Table 5.12: Number of Respondents for each Scale of each Statement	139
Table 6.1: Summary of objectives and achievements.....	144

CHAPTER 1: INTRODUCTION

1.1 Background

Expenses are the money that we spend on something ("Expense", 2022). There are many types of personal expenses, such as housing or rent, transportation, food and groceries, utility bills, school fees, insurance, entertainment, and so on (Sky, 2022) A receipt is a piece of paper or an electronic document that shows the goods or services have been paid for ("Receipt", 2022). When someone makes a payment or transaction, he or she will receive an electronic or paper receipt that records the details of the payment, including the dates, the company's name, the paid items or receiver's name, the amount, the total amount, and some other optional details. Receipts are kept for some reasons including tracking the expenses, proofing the purchases for warranties, and for return and exchanges of goods. Therefore, most of the people kept the receipts after purchasing goods.

The individuals are recommended to keep track of expenses as it would help to meet their financial objectives and reveal spending issues such as negative spending behaviors (Caldwell,2022). Recording daily and monthly transaction including expenses and income helps people manage or track their expenses and make a better budget. It is because they will have an idea of where their money is going and can make smart and sustainable cuts in certain areas so the overspending issue can be avoided. Not only that, record expenses and income can be useful when filing personal income taxes yearly.

There are many methods for people to record his or her spending. For instance, some of the people would simply record it in a notebook by summing up the total for the daily purchase amount and keeping the receipts, while some of them would record it by using an expense tracking worksheet template in Excel. Other than those methods, many people record their spending and income by using budgeting application such as MoneySave, MyMoney and Smoney as well in order to track the total spending and balance amount of a month. Budgeting

application is an application that used by people to record transaction including expenses and income so they can manage and track their income and spending (Horton, 2022). There are various budgeting applications which developed for different groups of people with different requirements in the market.

Computer vision is a field of artificial intelligence in which programs attempt to identify objects represented in digitized images provided by cameras, thus enabling computer to “see” (“Computer Vision”, 2022). Optical character recognition (OCR) is a technique that identifies printed text or numerical data, and it avoids the need to retype already printed material for data entry (“OCR”, 2022). OCR is encompassed in computer vision, and it is used to transform printed data into digital. A budgeting mobile application with computer vision including OCR can help users record their expenses in an efficient and time-saving way by extracting the data from receipts.

1.2 Problem Statement

There are many ways for people to record their spending. For instance, someone who wants to record his or her expenses may do so manually by totaling up the prices for daily spending. After that, the total amount is written down in a notebook and the receipts are kept in the notebook. They may also fill up the expense tracking worksheet template to record the expenses. It is time-consuming and inefficient because they cannot record the expenses anytime and anywhere, so they may not record every single purchase based on the receipts. Some people may record their spending by taking a picture of the receipt and saving it on their smartphones for further reference. The receipts are not categorized and summarized. As a result, it would be hard and inconvenient for them to track the spending or balance amount by month.

Besides, a budgeting application’s user is required to enter transaction information such as amount, name, category and etc. when he or she record their expense or income. If the user

wants to record many spending, the user needs to repeat the steps by referring to the receipts to get the amount and enter the data into budgeting application. The user faces the problem of repeating step to create a transaction record.

1.3 Aims and Objectives

- To analyze and design a mobile application that allows users to record their expenses and income.
- To develop and test a mobile application that can extract the data from the image of a receipt by implementing computer vision including OCR and record the data into the system.

1.4 Scope

The target user of this budgeting mobile application is any individual who needs to record daily transaction including expenses or income. This application would be an Android application, which can be used only on Android phones. Other than the basic functions of a budgeting application such as recording expenses and tracking spending, the project is focusing on implementing computer vision into the application, which can extract the data from receipts to help users save time from inputting spending records one by one.

1.5 Methodology

Rapid Application Development (RAD) is a system development methodology to develop software that requires minimum planning for rapid prototyping. (Naz & Khan, 2015) The RAD model is chosen for this project as it promotes fast, efficient and accurate program

or system development and delivery. This methodology consists of four phases which are requirements planning, user design, construction, and implementation.

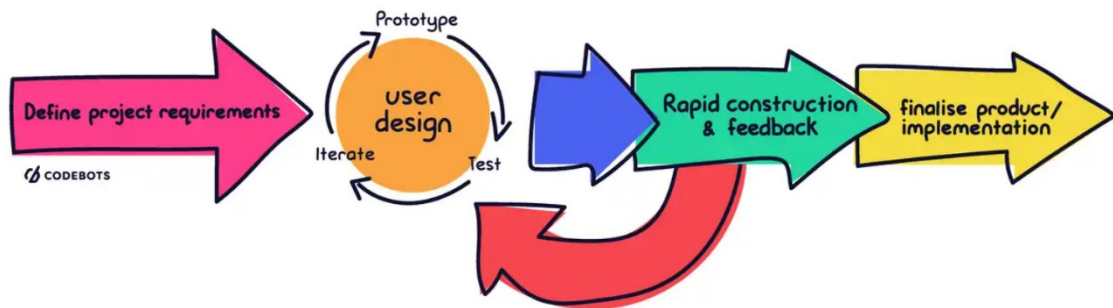


Figure 1.1: Phases of Rapid Application Development, by C. Chien, 2020, Retrieved from <https://codebots.com/app-development/what-is-rapid-application-development-rad>.

1.5.1 Requirements planning

Requirements planning is the first phase of the RAD, where the objectives of the proposed title are identified, and the topic is studied. The proposed title is budgeting mobile application with computer vision. The questionnaire is created and randomly filled out by at least thirty people in order to gather the information about the title such as the ways of budgeting, the problem faced when recording transaction and managing budget, users' requirements on the budgeting application and etc. Three existing similar budgeting mobile applications are compared, and the pros and cons of those applications are identified. In this phase, the methodology of the project is planned as well so that the project flow is smooth and well-planned.

1.5.2 User Design

The second phase is the user design phase, which can also be considered the prototyping phase. The prototype is not necessary a working system with all the functions,

but a simple prototype such as a paper prototype. The prototype will be created using a prototyping application such as Figma. The requirements of the system are figured out, and a quick prototype consisting of the features based on the requirements is created and shown to the users. The users try the application, and the feedback from the users is collected. The feedback will be collected by using Google Form, where the prototype is attached in the Google Form so the feedback will be collected at any time. The prototype will be updated again according to the feedback. After refining the prototype, it will be tested by the user again. This route is iterated until the prototype is satisfied by the user and developer.

1.5.3 Construction

The final prototype is refined based on the feedback collected from the previous phase. Therefore, the budgeting application is developed and tested in the construction phase. Android studio will be used to develop the application with Java programming language. For the backend, PHP will be used to write API to get the data from database while the database management system, MySQL will be used. Azure Form Recognizer will be applied in this budgeting application to achieve receipt scanning and extract the data from receipts. The quality, scalability, maintainability, etc. are focused on during the application development, and eventually a complete budgeting application is developed. The application can still be tested by the user to gather feedback or suggestions.

1.5.4 Implementation

After the construction phase is complete, the necessary documentation is completed, and the budgeting mobile application is installed or launched.

1.6 Significance of Project

The purpose of developing this budgeting mobile application is to make it more convenient for people who want to record transaction including expenses and income by allowing them to get the data directly from receipts instead of manually entering the data. The users are able to record the transaction manually by filling in the data field in a form. Other than that, the users can just take a picture of the receipt as well, and the application will help to extract the data from the receipts and insert into the data field. With the computer vision, this budgeting mobile application can become more attractive to users, and the AI feature that is OCR help to increase the effectiveness of recording transaction by reducing the time taken for them to insert the data.

1.7 Project Schedule

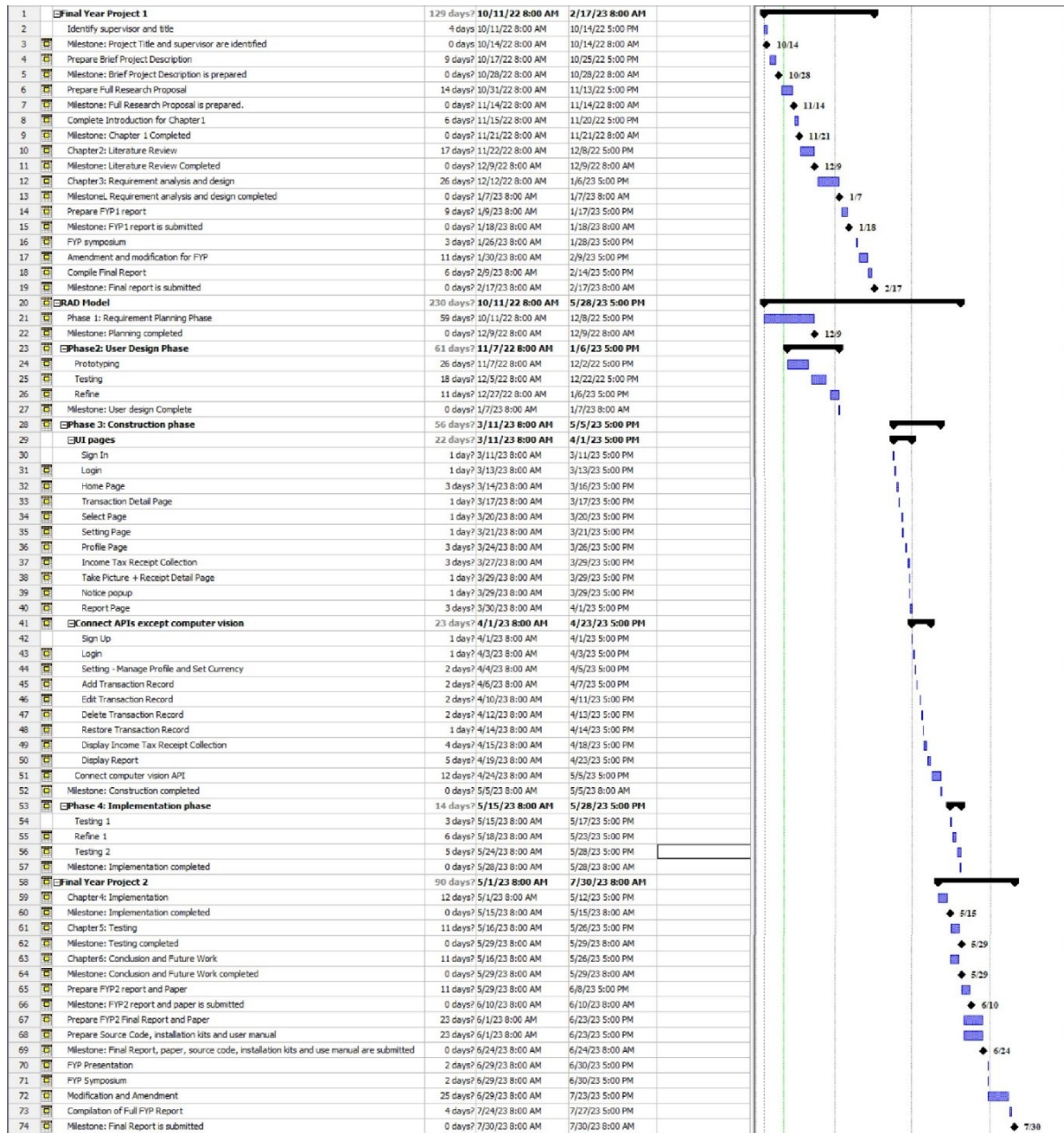


Figure 1.2: Gantt Chart for Final Year Project