Trends in Undergraduate Research (2024) 7(1): c16-25
https://doi.org/10.33736/tur.6110.2024

Receipt Image Extraction in Budgeting Mobile Application

WONG ROU YI* & TAN PING PING*

Faculty of Computer Science and Information Technology, Universiti Malaysia Sarawak, 94300 Kota Samarahan, Sarawak, Malaysia
*Corresponding author: 71970@siswa.unimas.my, pptan@unimas.my

ABSTRACT

Technology brings convenience to people and changes the lifestyle of people as well in various aspects such as work, entertainment, education, finance and so on. It has revolutionized the way people approach budgeting, make it become easier and straight forward. In the aspect of finance, budgeting mobile application helps the group of people to record transactions who need to track their money flow. However, the normal manual input of data can be tedious and with high possibility of error. In this paper, 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. 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. Based on user testing, 26 out of 30 respondents (86.7%) agreed and 14 out of them strongly agreed that the receipt scanning feature and its ability to automatically fill in the transaction details are satisfactory. In summary, the OCR feature is important in a budgeting app.

Keywords: budgeting mobile application, Computer Vision, Optical Character Recognition (OCR), Form Recognizer, receipt extraction

INTRODUCTION

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. Besides, receipts are kept for tracking the expenses, proofing the purchases for warranties, and for returns and exchanges of goods. Therefore, most people kept the receipts after purchasing goods. 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). According to the Malaysian Financial Literacy Survey 2023 from RinggitPlus (2023), expenses tracking is one of the proactive measures for the Malaysian to start practicing in order to address their financial situation. At the same time, recording expenses and income can be useful when filing personal income taxes yearly. In this modern era which is full of advanced technology and automated approaches provided to make life easier, one of the methods to record money flow is to make use of the mobile application. For instance, a budgeting application can be used by people to record transactions including expenses and income so they can manage and track their income and spending including those with scanned receipts (Garcia & Claour, 2021 & Horton, 2022).

Different people have different ways to record their money flow. Some people may do so manually by totaling up the prices for daily spending and note it down somewhere such as notebook, excel file, and smart phone. They would also like to keep the evidence of spending, that is receipt, invoice and some other hard copies. A survey was conducted on 14 December 2022 using Google Form to gather the perspective of the user experience on manage their budget and according to the survey, 20 out of 30 respondents mentioned that receipts are kept for money flow tracking, and the following reasons are used as proof of purchase for warranties and return or exchange purpose. However, 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. The receipts are not categorized or summarized. This is also supported in (Kumar et al. 2020). As a result, it would be hard and inconvenient for them to track the spending or balance amount by month. According to the survey again, 83.9% of the respondents believe that the budgeting mobile application is helpful for them to track their expenses. There are various budgeting applications in the market, but most of those budgeting application's user is required to enter every transaction information such as amount, name, category, etc. when he or she records their expense or income. If the user wants to record many transactions, the user needs to repeat the steps by referring to the receipts to get the amount and enter the data into the budgeting application. In fact, the problem of repeating steps to record a transaction record can be solved by using artificial intelligence.

An idea comes out for solving the problems faced by users, which is designing a mobile application that allows users to record their expenses and income (Polak et al. 2020). This project aims to develop and test a mobile application that can extract the data from the image of a receipt implementing computer vision including OCR and record the data into the system. Optical Character Recognition acts like humans recognize the characters including handwritten or machine printed. Image processing, neural network, character positioning and segmentation are the steps to achieve character recognition. (Candra et al., 2020) In the proposed application, OCR is not enough as it only assists to extract the character from image but still need to process form-like document to get the meaningful data. Therefore, Azure Form Recognizer API is called to extract the data of the receipts. It is helpful for processing document images including receipt and extracts the meaningful data on receipt and exports them in JSON format. (Microsoft, 2020) The target user of this budgeting mobile application is any individual who needs to record daily transactions including expenses or income. 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 by inputting spending records one by one.

Rapid Application Development (RAD) is a system development methodology to develop software that requires minimal planning for rapid prototyping (Naz & Khan, 2015). The RAD model was 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.

At the end of the project, a budgeting mobile application that applies computer vision is designed and developed in order to bring convenience to users who want to record all their transactions into a system by taking a picture of the receipt and the data from the receipts will be extracted automatically. The users can record the transaction manually by inserting the data field on their own as well. Computer vision is implemented in the application so the data from receipts can be extracted, and it reduces the time and steps for the user to enter the details of the expenses into the system. A monthly report will be generated at the end of the month for users to track their monthly expenses, income and balance amount. With this application, users can record their expenses and income and track them easily.

MATERIALS & METHODS

Literature review

The budgeting applications that are going to do a comparison are Monefy (Monefy, n.d.), Easy Expense (Easy Expense, n.d.) and Foreceipt (Foreceipt, n.d.). These apps are compared in the aspect of main functions that are scanning the receipt and applying computer vision, recording transactions including expenses and income and showing monthly summary reports. The other functions which are categorizing expense and payment methods, scanning digital receipts, saving multiple receipts, restoring deleted records, setting currency, team collaboration and cloud synchronization are compared within those apps as well. Table 1 shows the summary of the existing application and the comparison with the proposed budgeting application.

Table 1. Summary of the existing application

Application Features	Existing Budgeting Mobile Application			Proposed Budgeting application
	Monefy	Easy Expense	Foreceipt	
Scan Paper Receipt	No	Yes	Yes	Yes
Scan Digital Receipt	No	No	Yes	No
Apply Computer Vision	No	Yes	Yes	Yes
Customize Categories of Expense/Income	Yes	Yes	Yes	Yes
Save multiple receipt	No	Yes	Yes	No
Record Expense	Yes	Yes	Yes	Yes
Record Income	Yes	No	Yes	Yes
Cloud Sync	Yes	Yes	Yes	No
Local Backup	Yes	No	No	No

Team Collaboration	No	Yes	Yes	No
Link to Bank Account	No	Yes	No	No
Restore Deleted Record	No	Yes	Yes	Yes
Set Currency	Yes	Yes	Yes	Yes
Monthly Expenses Summary	Yes	Yes	Yes	Yes
Income Tax Receipt Collection	No	No	No	Yes
Paid for more features	Yes	Yes	Yes	No

Based on Table 1, in comparison to existing budgeting apps in the marker, the feature scan receipts is an important feature in mobile budgeting app. This is also supported in the literature works by Garcia & Claour (2021, November). As most receipts are standard, optical character recognition (OCR) technology is suitable (Candra et al. 2020; Kumar et al., 2020, September). Robert and Talbot (2020) had also demonstrated the accuracy of OCR to extract information from receipts.

System design

The methodology used for designing and developing this proposed budgeting mobile application is Rapid Application Development (RAD). RAD consists of four phases which are requirements planning, user design, construction, and implementation (Chien, 2020). This methodology is used to control and plan the process of development of the proposed application, so the whole process is well-performed and systematic, and a completed application is developed at the end. The activities of the user design phase of RAD are listed in the Table 2 below.

Table 2. Activities of user design phase

Phases		Activities
Requirements planning		1. Collect user' s problem and experience on budgeting mobile application. 2. Determine problem statement and objectives of the title. 3. Logical Design - Use Case Diagram - Sequence Diagram - Class Diagram - Database Diagram 4. Physical Design - Graphical User Interface Design
User Design	Prototype	Create prototype based on requirements.
	Testing	Collect feedback on prototype from users.
	Refine	Update prototype based on the collected feedback.
Construction		Develop the proposed budgeting mobile application.
Implementation		Implement the proposed budgeting mobile application.

There is only one main user who is the normal user that uses the proposed budgeting mobile application. Table 3 outlines the functional requirements for the main user. Table 4 provides an overview of the development tools utilized for developing the proposed budgeting mobile application. Other than Visual Studio Code and Android Studio which are used for coding purposes, various Azure services were applied to enable running the application's backend in the cloud, hosted by Azure Services.

Table 3. Functional Requirements for the Main User

	Requirement Description	Priority
FR.MU01	Login to the proposed application.	High
FR.MU02	Register an account for the proposed application.	High
FR.MU03	Record every transaction including income and expenses in the proposed application.	High
FR.MU04	Snap receipt and upload the receipt to the proposed application.	High
FR.MU05	Edit transaction record detail such as amount, category, transaction type, payment method and etc.	High
FR.MU06	Delete transaction record.	Medium
FR.MU07	View monthly summary which show the income, expenses and balance of the month.	High

FR.MU08	View yearly income tax receipt collection that shows the income and expenses record, and the total amount of them.	Medium
FR.MU09	Restore deleted transaction record.	Medium
FR.MU10	Set currency of the transaction amount.	Medium
FR.MU11	Manage profile by changing account username or password.	Medium

Table 4. Tools used to develop the proposed budgeting application

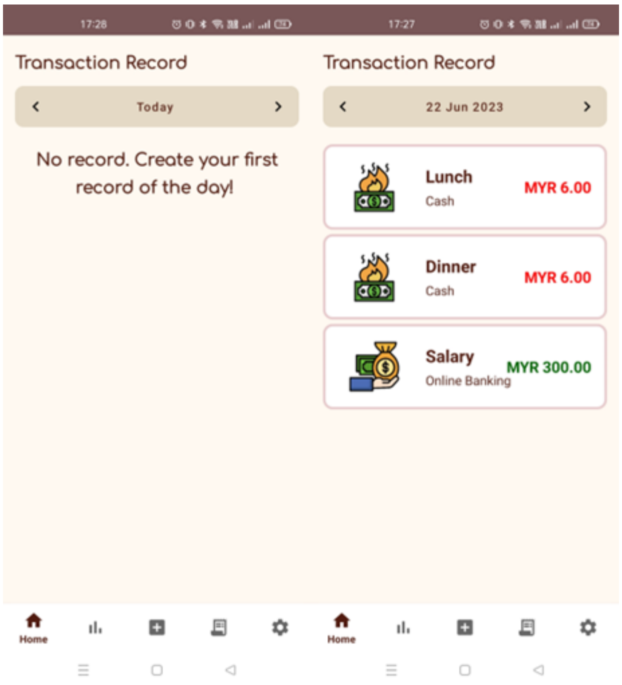
Tools	Applications
Visual Studio Code	Used to code API with the PHP language and JSON format data to achieve the data transferring between database and the application.
Android Studio	Used to develop the proposed budgeting mobile application with Java language and XML.
Azure Form Recognizer	Used to extract the data from receipt.
Azure App Service	Used to host the APIs that are coded in PHP languages for the connection between the frontend and the backend of the proposed application.
Azure database for MySQL Flexible Server	Used to deploy and manage MySQL server of the proposed application.

RESULTS & DISCUSSION

Microsoft Azure provides Computer Vision Read APIs which consist of Computer Vision v4.0 to process non-document images and Form Recognizer to process digital documents with asynchronous API. Form recognizer helps process document images including receipt (Microsoft, 2020). It extracts the key field data on receipt and exports them in JSON format. The Form Recognizer consists of many types of models such as Read OCR model, General Document model, layout model, invoice model, receipt model, etc. The receipt model can be applied in the proposed budgeting application to meet the requirement that is extracting data from receipts. In the development of the proposed application, Azure Form Recognizer API is called to analyze the receipt. The implemented Azure Form Recognizer API version is 2022-08-31. Other than that, Azure Web App is used to host the APIs of the proposed application as well and the application is developed by using Android Studio with Java language.

At the end of the development, a budgeting application with the designed features was developed. It allows users to sign up, login and perform CRUD operations which is adding, viewing, updating, and deleting transactions. The proposed application also provides users with the ability to access monthly summary reports and yearly income tax receipt collection and restore deleted transactions. The users are allowed to manage their profile details and customize the currency used in the application as well. In this section, the feature of receipt extraction is emphasized and discussed.

The focused feature of the proposed budgeting application is adding transaction records to the system with the help of receipt extraction. Figure 1 shows the home page of the system. To add a transaction record, firstly, the user clicks the add button in the bottom navigation bar and the select page is shown for the users to select whether they want to record the transaction manually or take a picture of a receipt to use the receipt scanning function. If the user selects to record a transaction manually, the user clicks on the “Record Transaction” button and the system navigates to the record transaction detail page. The user is required to fill in the form by entering the detail of the transaction record which are date, type of transaction, category, name, amount, payment type, remark and use for income taxes filing as shown in Figure 2.



Transaction Record

Today

No record. Create your first record of the day!

Transaction Record

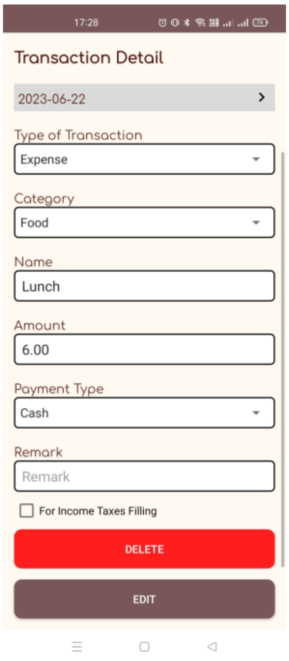
22 Jun 2023

Lunch — Cash — MYR 6.00

Dinner — Cash — MYR 6.00

Salary — Online Banking — MYR 300.00

Figure 1. Home page



Transaction Detail

2023-06-22

Type of Transaction: Expense

Category: Food

Name: Lunch

Amount: 6.00

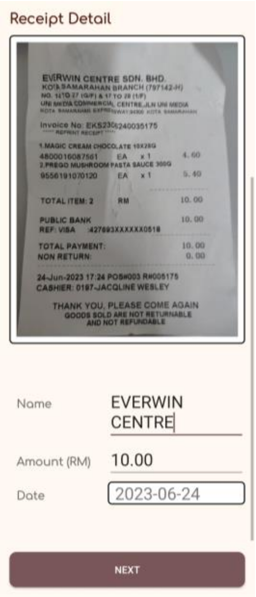
Payment Type: Cash

Remark: Remark

□ For Income Taxes Filling

Figure 2. Transaction detail page

If the user selects to take a picture of the receipt, the camera of the phone is accessed, and the users are asked to take the picture. The taken picture should be clear and the whole receipt is within the picture. After the picture is taken, the image of the receipt is uploaded to Azure Form Recognizer through API to analyze the receipt. If the picture contains a receipt, once the result of the analysis is obtained, the system parses the result that is in JSON string format and gets the required key fields which are the name, total amount and the purchase date. After that, the system navigates to the receipt detail page and shows the name, total amount and date as shown in Figure 3. The user can edit the data if the data are not correct. Once the user clicks on the next button, the record transaction detail form as shown in Figure 4 is displayed, and the detail will be automatically filled in the form. The rest of the key fields which type of transaction, category, payment type and remark are needed to be filled in by the users. The user can also mark it for income tax filing as well for income tax receipt collection reference. To check the analysis result, the picture can be uploaded to Azure Form Recognizer Studio and the result is produced as shown in Figure 5.



Receipt Detail

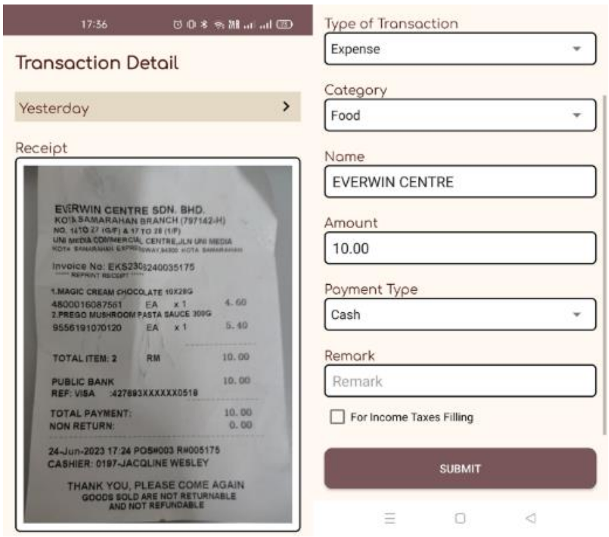
EVERWIN CENTRE SDN. BHD.
KOTA SAMARAHAN BRANCH (797142-H)
Invoice No: EKS2306240035175
1.MAGIC CREAM CHOCOLATE 10X28G
4800016087561 EA x 1 4.60
2.PREGO MUSHROOM PASTA SAUCE 300G
9556191070120 EA x 1 5.40
TOTAL ITEM: 2 RM 10.00
PUBLIC BANK 10.00
REF: VISA 427693XXXXXX0518
TOTAL PAYMENT: 10.00
NON RETURN: 0.00
24-Jun-2023 17:24 POS#003 R#005175
CASHIER: 0197-JACQLINE WESLEY
THANK YOU, PLEASE COME AGAIN
GOODS SOLD ARE NOT RETURNABLE
AND NOT REFUNDABLE

Name: EVERWIN CENTRE

Amount (RM): 10.00

Date: 2023-06-24

Figure 3. Receipt detail page



Transaction Detail

Yesterday

Receipt

Type of Transaction: Expense

Category: Food

Name: EVERWIN CENTRE

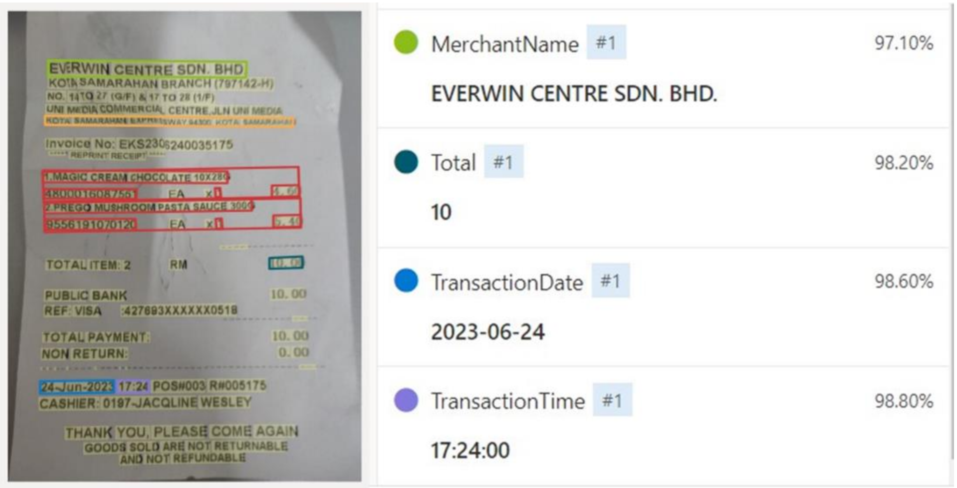
Amount: 10.00

Payment Type: Cash

Remark: Remark

□ For Income Taxes Filling

Figure 4. Transaction detail page after Scanning Receipt



EVERWIN CENTRE SDN. BHD
KOTA SAMARAHAN BRANCH (797142-H)
NO. 14 TO 27 (G/F) & 17 TO 28 (1/F)
UNI MEDIA COMMERCIAL CENTRE,JLN UNI MEDIA
KOTA SAMARAHAN EXPRESSWAY 94300 KOTA SAMARAHAN

Invoice No: EKS2306240035175
***** REPRINT RECEIPT *****

1.MAGIC CREAM CHOCOLATE 10X28G
4800016087561 EA x1 4.60
2.PREGO MUSHROOM PASTA SAUCE 300G
9556191070120 EA x1 5.40

TOTAL ITEM: 2 RM 10.00

PUBLIC BANK 10.00
REF: VISA :427693XXXXXX0518

TOTAL PAYMENT: 10.00
NON RETURN: 0.00

24-Jun-2023 17:24 POS#003 R#005175
CASHIER: 0197-JACQLINE WESLEY

THANK YOU, PLEASE COME AGAIN
GOODS SOLD ARE NOT RETURNABLE
AND NOT REFUNDABLE

MerchantName #1 97.10%
EVERWIN CENTRE SDN. BHD.

Total #1 98.20%
10

TransactionDate #1 98.60%
2023-06-24

TransactionTime #1 98.80%
17:24:00

Figure 5. Analysis result on Azure Form Recognizer Studio

Figure 6 illustrates a picture that has been taken or uploaded without any associated receipt. Once the picture is uploaded, the system analyzes the JSON string result that is get from Form Recognizer API. If the picture does not contain any text or it contains text but no key fields such as name, total amount and date are extracted, it is considered that there is no receipt was analyzed. The proposed application would prompt the message to notify the users the picture does not contain any receipt as shown in Figure 7. The users can close the prompt-out message and the application will navigate to the previous page and ask the users to retake a picture of the receipt for receipt extraction.

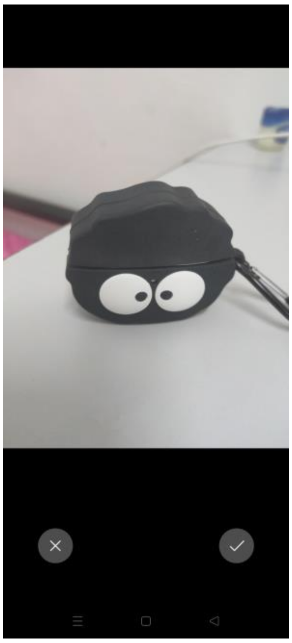
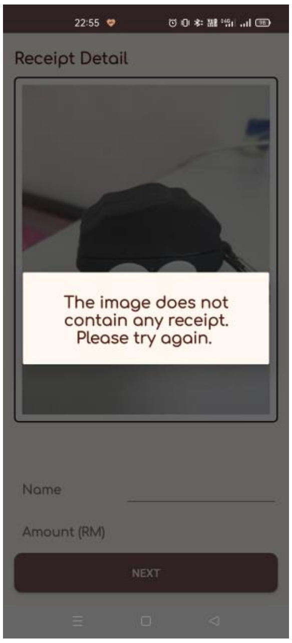


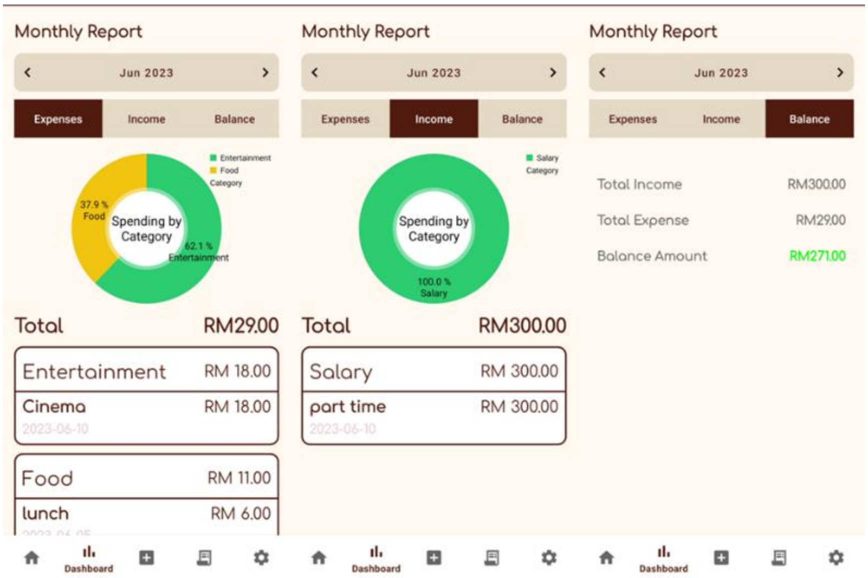
Figure 6. Uploaded picture without receipt



The image does not contain any receipt. Please try again.

Figure 7. Analysis result with error message

The proposed application also allows users to view the monthly report that shows the monthly expense, monthly income and monthly balance. A pie chart is displayed to show the expense or income by category visually as shown in Figure 8. Every transaction record of a category is displayed below the pie chart as well. The user can click the transaction record in the list and the system will navigate to the transaction detail page and shows the detail of the selected record. Besides, the balance of the month is calculated and shown to the user. Other than viewing the monthly report, the proposed budgeting application also shows the yearly income tax overview as shown in Figure 9. The user can view the total expense and income records that are tagged for income tax filing of the year.



Total RM29.00

Entertainment RM 18.00
Cinema RM 18.00
2023-06-10

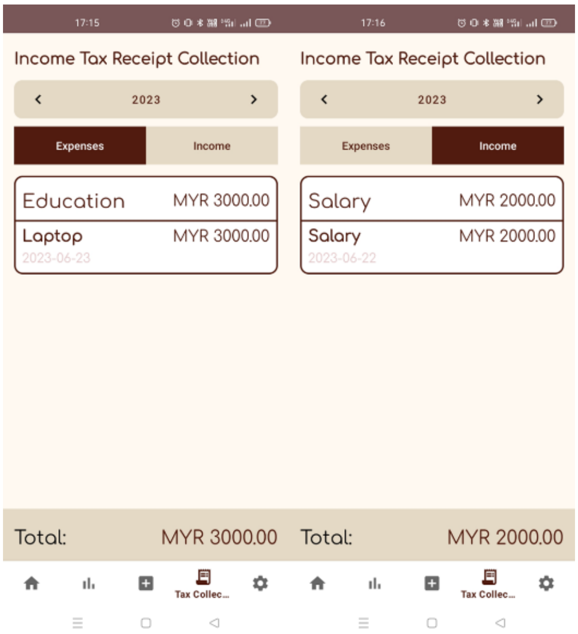
Food RM 11.00
lunch RM 6.00

Total RM300.00

Salary RM 300.00
part time RM 300.00
2023-06-10

Total Income RM300.00
Total Expense RM29.00
Balance Amount RM271.00

Figure 8. Monthly report page



Education MYR 3000.00
Laptop MYR 3000.00
2023-06-23

Total: MYR 3000.00

Salary MYR 2000.00
Salary MYR 2000.00
2023-06-22

Total: MYR 2000.00

Figure 9. Yearly income tax record overview

Once the proposed budgeting application was fully developed, the unit testing, functionality testing, and usability testing was done as well to test the performance, functionality and usability of it. Besides, software testing can help to detect the vulnerabilities and defects of the software before launching the application. All components of each page of the proposed application are tested to ensure all of them operate well. To facilitate usability testing, a questionnaire was prepared to collect feedback from 30 respondents and the respondents are required to test the real application before answering the questionnaire. Figure 10 and Figure 11 present the scale for each statement. Scale 1 represents strongly disagree while scale 5 indicates strongly agree with the statements.

Based on the result, a majority of the respondents, specifically 63.3% of them, strongly agreed that the application's interface layout is clear, consistent, and effectively presents features and information. In addition, statements such as "The application provides clear and visually appealing charts to summarize monthly transactions," "It is straightforward to record transactions," and "It is straightforward to scan a receipt and extract data" received strong agreement from 18 respondents. Besides, there are 26 out of 30 respondents (86.7%) agreed and 14 out of them strongly agreed that the receipt scanning feature and its ability to automatically fill in the transaction details are satisfactory. Next, there are 96.7% of the respondents agreed that the application is user-friendly. In summary, the respondents are satisfied with the user interface design and record transaction feature. At the same time, the receipt scanning feature received positive feedback from the respondents.

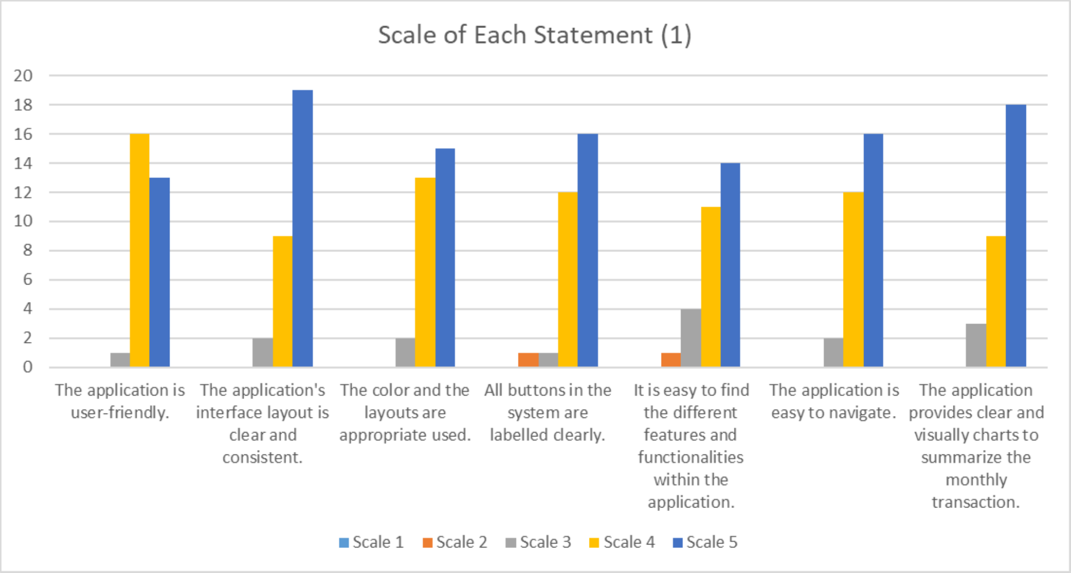


Figure 10. Scale of each statement part 1

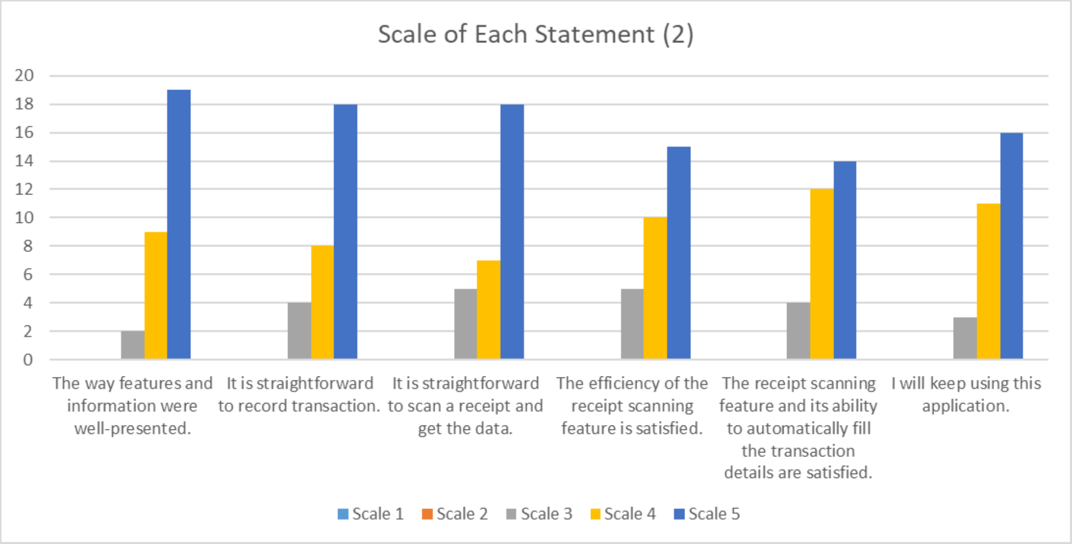


Figure 11. Scale of each statement part 2

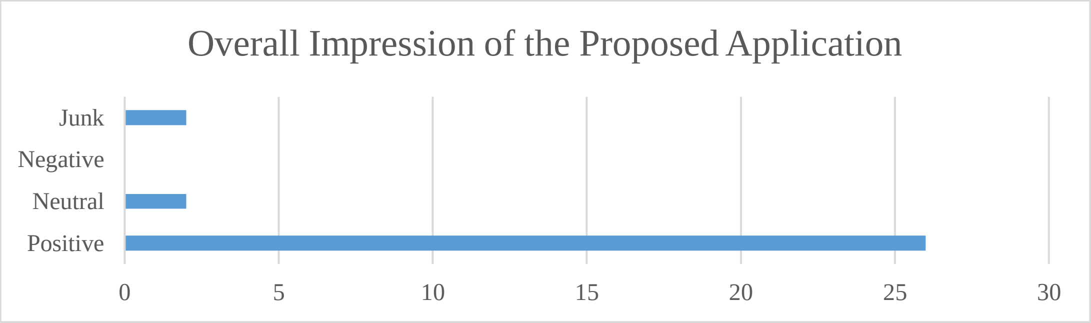


Figure 12. Overall impression of proposed application

Figure 12 shows the overall impression of the proposed application. Based on the feedback collected through the questionnaire, the majority of the respondents hold a positive impression of the application. They expressed satisfaction with the receipt scanning feature, particularly its ability to automatically fill in the transaction details in the form. Besides, some respondents found the application is helpful and user-friendly and possesses a good user interface design. The scanning receipt feature was highlighted as particularly beneficial and time-saving. However, a significant number of users provided feedback regarding the slow loading or waiting time for obtaining outcomes on each page. Additionally, users recommended recording more transaction detail and extracting more information from receipts.

CONCLUSION & FUTURE WORK

The project is aimed to achieve all the objectives and to solve the problems that are stated before. At the end of the project, a budgeting mobile application with the implementation of computer vision is developed to bring convenience to the users. It is successfully connected to the Azure Form Recognizer API to analyze the receipt and extract key fields of the receipt. After that, the data is automatically filled in the form and saved into the system. A budgeting mobile application that allows users to add transaction records is created. Lastly, the budgeting application is tested to make sure the functions are working well.

Even though the objectives of the project have been achieved and other designed features of the project are developed successfully, there are some limitations to the project. Firstly, it takes a long time to wait for analyzing the receipt which is around 5 to 10 seconds due to the slow internet speed or API connection performance. The user has a bad experience since he or she needs to wait a few seconds to get the result. Other than that, the loading time for getting outcomes on each page is long due to the free version of Azure Web App Service that is used to host the APIs. In addition, the application still requires user manual input or checking the detail of the transaction after the receipt is analyzed since the data extracted may be not accurate. Lastly, the application does not list every item on a receipt. It only shows the name of the merchant, the total amount and the purchase date of the receipt.

Based on the limitation of the project that is discussed above, there are some functions or improvements that can be implemented in the budgeting mobile application in the future. Firstly, it can list down every item of a receipt and record it in the system. Next, the time taken to wait for the analyzed receipt result should be shortened in order to bring a good user experience to the user while using the application. Besides, the application should increase the accuracy of the scanning result and it should extract more other key fields such as payment type, category, tax and so on for user to record as many data as their needs for future expenses tracking. To achieve this function in the future, processing the JSON output from Azure Form Recognizer API to get the item list and display on the screen can be work on. Therefore, the user can use the scanning feature to add transactions automatically into the system without any manual input and save the information they want. To increase the accuracy, the model used to analyze receipts can be trained to increase the accuracy of the result. Besides, increase the image contrast, and provide real-time user feedback such as guiding the user during picture taking process would be the methods to increase the accuracy and user experience. To address the low-resolution of the receipt image, Single image super-resolution (SISR) can be utilized to produce a higher-resolution images of the actual receipt that taken by the user and increase the accuracy of the scanning feature. Robert and Talbot (2020) shown a noteworthy enhancement in OCR performance when the recognition was poor in low-resolution, where the text recognition had been increased by up to 15%. On the other hand, SISR model could be applied to handle the noisy image and increase the OCR performance. (Robert & Talbot, 2020) Other than the improvement based on the limitation, the spending limit feature can be designed and developed as well. Lastly, it is possible to connect with an e-wallet or bank to retrieve the transaction record and save the record into the system automatically. The end goal to of the budgeting application is to provide a digital way for user to track their money flow as simple as they can.

ACKNOWLEDGEMENTS

We would also like to thank the Final Year Project coordinator, Prof. Wang Yin Chai who provide valuable information and tips during lectures. We would also like to thank the reviewers and respondents who provided useful and constructive information and contributed to this project. Lastly, we would like to thank our family and friends for their moral support and encouragement.

REFERENCES

Caldwell, M. (2022). Why You Need to Track Expenses to Become Aware of Your Spending. The Balance. Retrieved from https://www.thebalancemoney.com/is-it-important-to-track-my-expenses-2385679#toc-reasons-to-keep-track-of-expenditures

Candra, S., Sisodia, S., & Gupta, P. (2020). Optical Character Recognition – A Review. *International Research Journal of Engineering and Technology, 07(04),* 3037-3041.

Chien, C. (2020). What is Rapid Application Development (RAD)? Retrieved from https://codebots.com/app-development/what-is-rapid-application-development-rad

Easy Expense. (n.d.). Easy Expense. Retrieved from https://easy-expense.com/

Garcia, M. B., & Claour, J. P. (2021, November). Mobile bookkeeper: personal financial management application with receipt scanner using optical character recognition. In *2021 1st Conference on Online Teaching for Mobile Education (OT4ME)* (pp. 15-20). IEEE.

Horton, C. (2022, April). What is a Budget App? The Balance. Retrieved from https://www.thebalancemoney.com/what-is-a-budget-app-5216548

Kumar, V., Kaware, P., Singh, P., Sonkusare, R., & Kumar, S. (2020, September). Extraction of information from bill receipts using optical character recognition. In *2020 international conference on smart electronics and communication (ICOSEC)* (pp. 72-77). IEEE.

Microsoft Azure. (2020). How to call the Read API. Microsoft Learn. Retrieved from https://learn.microsoft.com/en-us/azure/cognitive-services/computer-vision/how-to/call-read-api

Monefy. (n.d.). Monefy. Retrieved from https://monefy.me/

Naz, R. & Khan, M. N. A. (2015). Rapid Applications Development Techniques: A Critical Review, 9(11), 163-176.

Polak, P., Nelischer, C., Guo, H., & Robertson, D. C. (2020). “Intelligent” finance and treasury management: what we can expect. *Ai & Society*, *35*(3), 715-726.

RinggitPlus. (2023). Malaysian Financial Literacy Survey 2023. Retrieved from https://ringgitplus.com/en/blog/wp-content/uploads/2023/10/RMFLS-2023-Survey-Report-FINAL.pdf

Robert,V. & Talbot, H. (2020). Does Super-Resolution Improve OCR Performance In The Real World? *A Case Study On Images Of Receipts. 2020 IEEE International Conference on Image Processing (ICIP),* 548-552.