



Software System Development Project |
COMPS456F

Final Report

**Topic: Scanning Enable Personalized
Calendar**

“TexPic Calendar”

Project Supervisor

Dr. Yalin Liu

Team members

*Li Heng 12895168 *Computer Science*

Cheung Ho Yan 12754335 *Computer Science*

Chan Wai Hei 12896614 *Computer Science*

Wong Kong Lun 12728784 *Computer Science*

Group Declaration

We, Li Heng 12895168, Cheung Ho Yan 12754335, Chan Wai Hei 12896614, Wong Kong Lun 12728784 , certify that the work is original and we have utilized guidance of our supervisor in completing this project, and that the content which is not our own has been attributed and referenced properly. There should be no copyrighted content without permission to use. There should be no confidential data.

We declare that the description and information outlined in the individual team member reports are true reflection of the project status to the best of our knowledge.

Signature and Date (of all team members)

The image shows handwritten signatures and dates for four team members. From left to right:
1. A black ink signature consisting of the numbers '13' and a cursive name.
2. A black ink signature of the name 'Chan'.
3. An orange ink signature consisting of the letters 'WHL' and 'Ho Yan'.
4. A black ink signature consisting of the date '22/4/2023'.
Below the first three signatures, there is a date '22/4/2023' written twice, once in black ink and once in orange ink.
At the bottom center, there is a large, thin-lined black 'X' mark.
Below the 'X' mark, the date '22/4/2023' is written in black ink.

Table of Contents

Chapter 1. INTRODUCTION.....	8
1.1 Overview.....	8
1.2 Aim.....	9
1.3 Objective.....	10
1.3.1 Objective 1. Optical Character Recognition (Text-scanner).....	10
1.3.2 Objective 2. Auto-schedule.....	12
1.3.3 Objective 3. Event management.....	13
1.4 Impact and Value.....	16
1.4.1 Impact & Value 1.....	16
1.4.2 Impact & Value 2.....	16
1.4.3 Impact & Value 3.....	16
1.5 Structure of the Report.....	17
Chapter 2. Literature Review.....	19
Chapter 3. Methodology.....	23
3.1 Overview of Solution.....	23
3.1.1 Programming Language and Framework:.....	23
3.1.2 Integrated Development Environment (IDE).....	24
3.1.3 Idea Structure of Calendar Application.....	24
3.2 Design and Major Components.....	25
3.2.1 Development Process.....	25
3.2.2 Design Diagram.....	26
3.2.3 Calendar Application Architecture.....	27
3.2.4 Optical Character Recognition —Text scanning.....	29
3.2.5 Auto-schedule.....	30
3.3 Design of Prototype System.....	32
3.3.1 Low-Fidelity Prototype.....	32
3.3.2 Mid-Fidelity Prototype.....	34
3.3.3 High-Fidelity Prototype.....	35
3.3.4 Final Product.....	36
3.4 Requirement.....	38
3.4.1 Functional requirements:.....	38
3.4.2 Non-functional requirements:.....	39
3.5 Supporting Technologies.....	40
3.5.1 Flutter Neat and Clean Calendar Package.....	40
3.5.2 OCR Scanning - ML Kit Text Recognition Flutter API.....	40
Chapter 4. Calendar Application Evaluation.....	41

4.1 Firebase Robo Testing.....	41
4.1.1 Testing Result.....	42
4.1.2 Conclusion of Robo Test.....	47
4.2 Survey of TexPic Calendar.....	48
4.2.1 Survey Result - UI Design.....	48
4.2.2 Survey Result - Functionality.....	51
4.2.3 Conclusion of Survey.....	54
Chapter 5. Optical Character Recognition Result.....	56
Chapter 6. Auto-Schedule Result.....	65
Chapter 7. Conclusion.....	73
7.1 Satisfaction of the aim and objectives.....	73
7.2 Achievements Summary.....	73
7.3 Limitation of the TexPic Calendar.....	73
7.4 Solution of the limitations.....	75
7.5 Suggestions of future work.....	76
References.....	77
Appendix A - GROUP INFORMATION.....	79
Appendix B1 Team Member #1 Report.....	80
Appendix B2 Team Member #2 Report.....	81
Appendix B3 Team Member #3 Report.....	82
I, Cheung Ho Yan(12754335), certify that the work described in this document is original and done by me and I have utilized guidance of our supervisor in completing this project, and that the content which is not our own has been attributed and referenced properly. There should be no copyrighted content without permission to use. There should be no confidential data. I also declare that the contributions listed in Self Appraisal of Contribution are truly and correctly attributed to me.....	82
Appendix B4 Team Member #4 Report.....	83
Appendix C - Survey Question.....	84
Appendix D - Survey Result.....	100
Appendix E - Test of Auto-schedule.....	111

List of Table

[Table 1: Functional & Non-function Requirement](#)

Table 2: Member's Role, Responsibilities and Work Percentage

List of Figures

- [Figure 1 & 2: Scan Screen and Scan Result](#)
- [Figure 3: Upload to Add event Page](#)
- [Figure 4: Auto-Schedule Page](#)
- [Figure 5 & 6: Add schedule page and update schedule page](#)
- [Figure 7 & 8: Schedule list and schedule list management](#)
- [Figure 9,10&11: Event analysis page](#)
- [Figure 12. The Dart packages of we used](#)
- [Figure 13. The conceptual layers of the calendar app](#)
- [Figure 14. Development and design process for the mobile application.](#)
- [Figure 15. The design diagram of the calendar application](#)
- [Figure 17. the structure of recognizedText variable](#)
- [Figure 18. Simple flow of the Auto-Schedule](#)
- [Figure 19. Hand draw low-fidelity prototype 1](#)
- [Figure 20. Hand draw low-fidelity prototype 2](#)
- [Figure 21 & 22. Mid-fidelity prototype of the calendar UI](#)
- [Figure 23 & 24. High-fidelity prototype of the calendar UI](#)
- [Figure 25 & 26. event adding page UI](#)
- [Figure 27,28 & 29. home page UI](#)
- [Figure 30 & 31. block UI](#)
- [Figure 32. Robo test result 1](#)
- [Figure 33. Robo test result 2](#)
- [Figure 34. Robo test result 3](#)
- [Figure 35. Robo test result 4](#)
- [Figure 36. Robo test result 5](#)
- [Figure 37. the layout of the input field](#)
- [Figure 38. the survey result of UI design](#)
- [Figure 39 & 40. the score of event adding page UI design](#)
- [Figure 41 & 42. the score of event adding page UI design](#)
- [Figure 43. survey result of OCR function](#)
- [Figure 44. the result of auto-scheduling function](#)
- [Figure 45. the result of data analysis page](#)
- [Figure 46. the score of the UI design](#)
- [Figure 47. ocr_screen.dart source code part 1](#)
- [Figure 48. ocr_screen.dart source code part 2](#)
- [Figure 49 & 50: The text scan camera and the result](#)
- [Figure 51: The text recognize result in console](#)
- [Figure 52 & 53: The testing text and the result without blank](#)

[Figure 54: For loop algorithm for recognizing the text inside “<>”](#)

[Figure 55 & 56: The error pages](#)

[Figure 57,58,59 & 60: Four error types of recognition result](#)

[Figure 61 & 62: The rotate camera and ml-kit scanning method](#)

[Figure 63 & 64: The rotated result and its input](#)

[Figure 65: Auto schedule page](#)

[Figure 66: Setting page of Auto-Schedule](#)

[Figure 67 & 68: Auto-Scheduling Page and Calendar View](#)

[Figure 69: Calendar view that shows the event will not overlap](#)

[Figure 70: result of survey](#)

[Figure 71: Field Format](#)

Abstract

Nowadays, people have a lot of activities to carry on. They will record the events on paper casually and sometimes will forget about it. Concerning this problem, we find that calendars are a standard tool for people to check and record some events. But many E-calendar need help to read the events from the paper and schedule them automatically. Therefore, this project will develop a mobile calendar application that can read the text from the paper and record it on the calendar, which can alleviate some of people's concerns.

Chapter 1. INTRODUCTION

1.1 Overview

Nowadays, whether the worker or student, they also have a lot of trivial problems or events that need to be remembered, such as homework, project, meeting, special festival. To mark down the event, they would like to use an E-calendar to record it, such as google calendar, apple calendar, Jorte calendar, and Todoist, which is convenient and effective.

One of the advantages of using E-calendar is portability, which is a highly portable digital tool that provides convenience to users. They can modify or check their schedule from anywhere, at any time, whether they are working outside or in the comfort of their own home. The accessibility and flexibility of E-calendar allows users to stay on top of their schedule and make adjustments as needed, without being tied to a specific location or device.

The other advantage is that it allows users to set event reminders for important dates or before deadlines. It ensures users never miss an important meeting or work. It not only reduces the risk of forgetfulness, but also enables users to stay on top of their schedule and plan their daily lives more efficiently.

The E-calendar not only has merit but also includes several problems. The big problem is that the e-calendar needs to inform users before the test and deadline. For example, the missing reminder will lead to the late submission of students and No class preparation for the teacher. Such as the Jorte calendar, sometimes it may miss the reminder, although users have set it. The second problem is that it is not user-friendly. Some newbies need to learn how to use it, which is not a good enough guideline. It leads to the concern of users and may reduce their wish of using E-calendar.

1.2 Aim

To counter the inconvenience of the traditional calendar and the drawbacks of the E-calendar, our application will develop a function with Optical Character Recognition (OCR) for digitizing the pictures to text, so we call this application “PicText”.

Other than OCR, the automatic scheduling is also developed for event management. Through automation, it makes a personalized design for users. It can assist users in managing their daily tasks efficiently with the logical conditions- "if-else statements".

1.3 Objective

To improve the calendar's personalization, three objectives are included:

1.3.1 Objective 1. Optical Character Recognition (Text-scanner)

For the first objective, it contains the technology of optical character recognition(OCR) for scanning the text. The application allows the user to use a phone camera to capture the text. Through the digitalization from word to digital by the ML-Kit, it displays the result of the recognized text (See Figure 2).

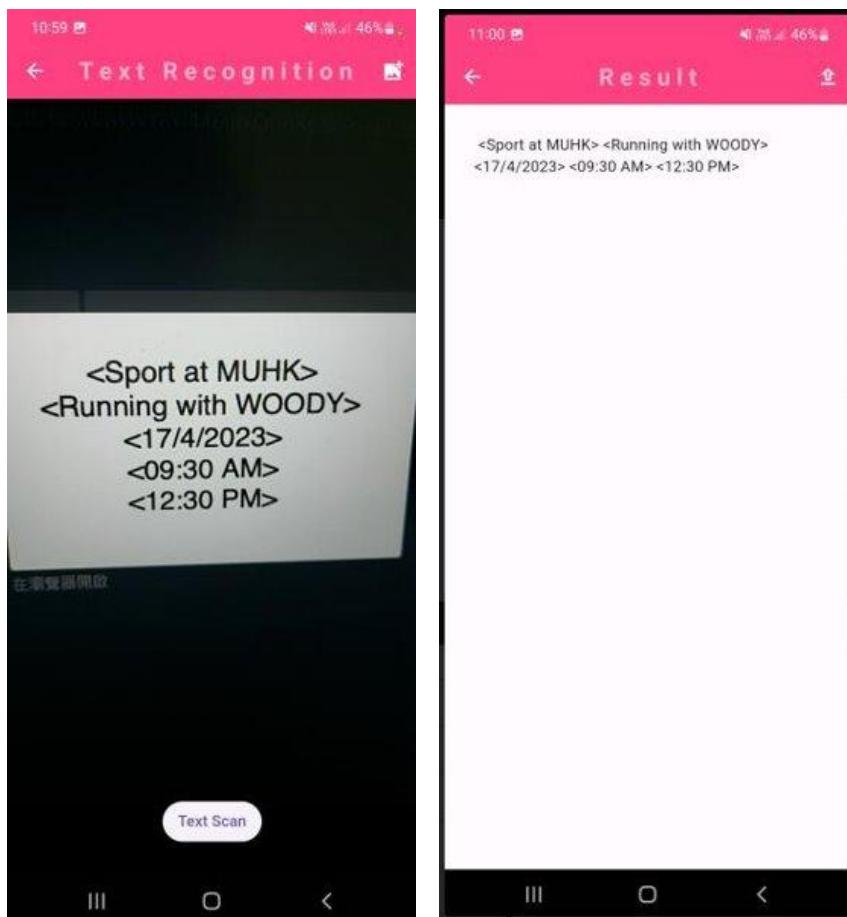


Figure 1 & 2: Scan Screen and Scan Result

The type of scanning includes the event title, event notes, the date, start time, and end time. Using this method, the system can quickly and easily read all the required details within the blank "<>"and put them into the add

event page (See Figure 2). It only passes the text inside the blank “< >”, which means that other text outside the blank will not be passed. Also, users can select their photo from the gallery, which contains the photo inside the phone.

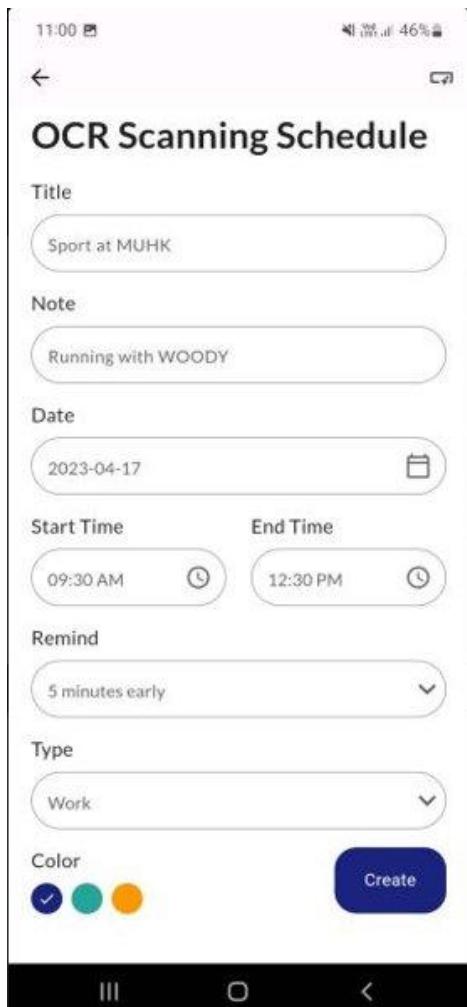


Figure 3: Upload to Add event Page

Finally, the event will be created and the data will also be added into the database.

1.3.2 Objective 2. Auto-schedule

For objective 2, we want to through the Auto-schedule function to help users to manage their daily tasks efficiently. So we have 3 goal to achieve:
Time-saving: Quickly generate schedules based on predetermined criteria.

Personalize:The schedule can be arranged based on their needs and preferences.

Efficiency: Automatically allocate tasks to available time slots, avoiding overlaps and conflicts.

To achieve 3 goals , we design the Auto-schedule page to be clear and easy to use. The User only needs to input the 4 fields:“Title”, “Note”, “Type”, “Duration”.(See Figure 4)

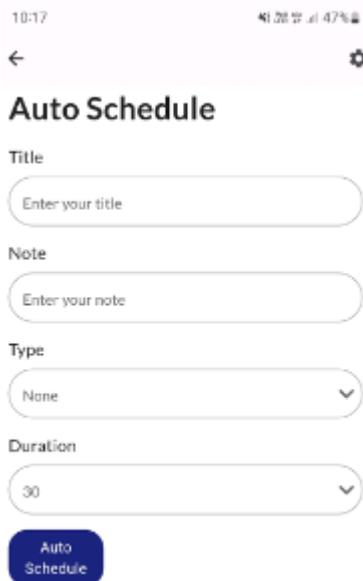


Figure 4: Auto-Schedule Page

Then the user no need to decide the date and time , just hit the function button and the function will automatically allocate the event based on your schedule .

1.3.3 Objective 3. Event management

For objective 3, our project involves the implementation of CRUD functionality. CRUD stands for creating the event, checking the details of the schedule, updating the event and deleting the wrong event. These are the basic requirements of the E-calendar.

The figure consists of two side-by-side screenshots of a mobile application's event management screen.
Left Screenshot (Add Schedule):
- Title: Enter your title
- Note: Enter your note
- Date: 2023-04-13
- Start Time: 11:04 PM
- End Time: 10:00 PM
- Remind: 5 minutes early
- Type: Work
- Color: A row of three colored circles (blue, green, orange) with a 'Create' button to its right.
Right Screenshot (Update Schedule):
- Title: COMP 5456F: Lesson
- Note: At MU 3/F
- Date: 2023-04-13
- Start Time: 2:00 PM
- End Time: 4:00 PM
- Remind: 5 minutes early
- Type: Work
- Color: A row of three colored circles (blue, green, orange) with a 'Update' button to its right.

Figure 5 & 6: Add schedule page and update schedule page

Creation allows users to add new events to their calendar easily such as the event title, date, time, and location.

Also, users can view the details about this month's schedule or upcoming event by checking the list.

For updating the event. Users can modify the details of their scheduled events which contain some wrong input or typo.

Additionally, users can remove the events from their calendar if they are no longer needed or have some error.

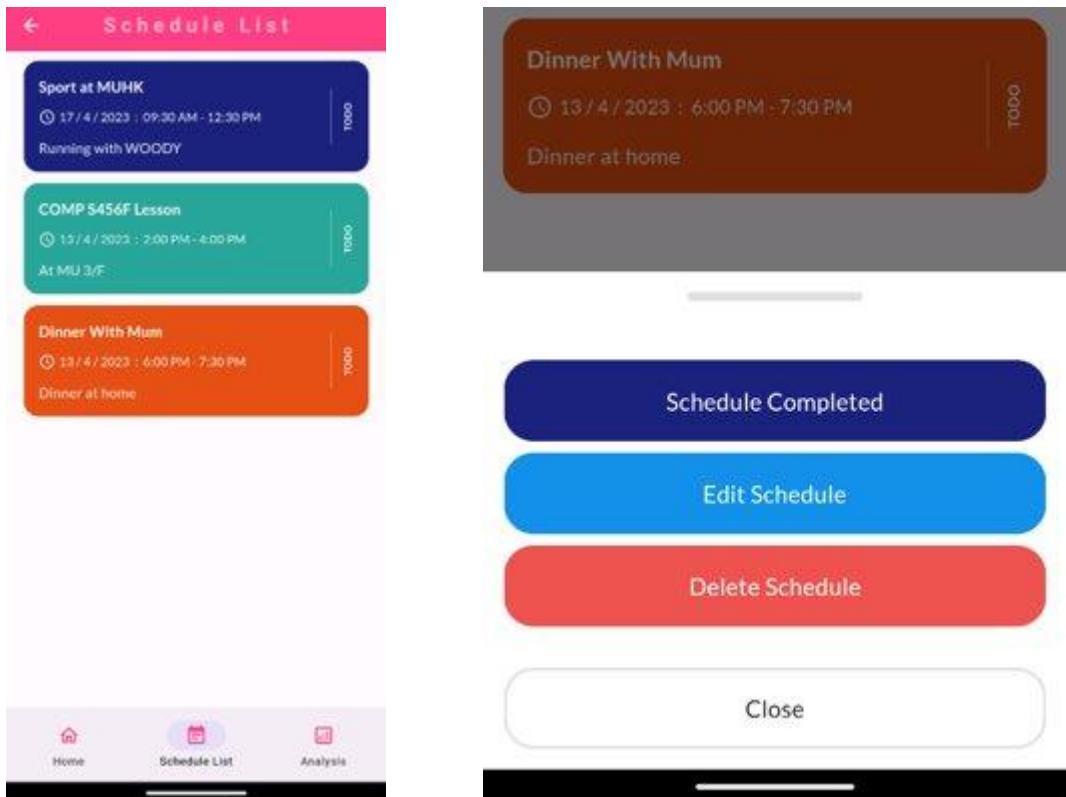


Figure 7 & 8: Schedule list and schedule list management

For the event management, users can review the event in the schedule list (See Figure 7). In this page, users can edit the event status, such as Ensure the schedule is complete or not, modify the schedule and delete the schedule.

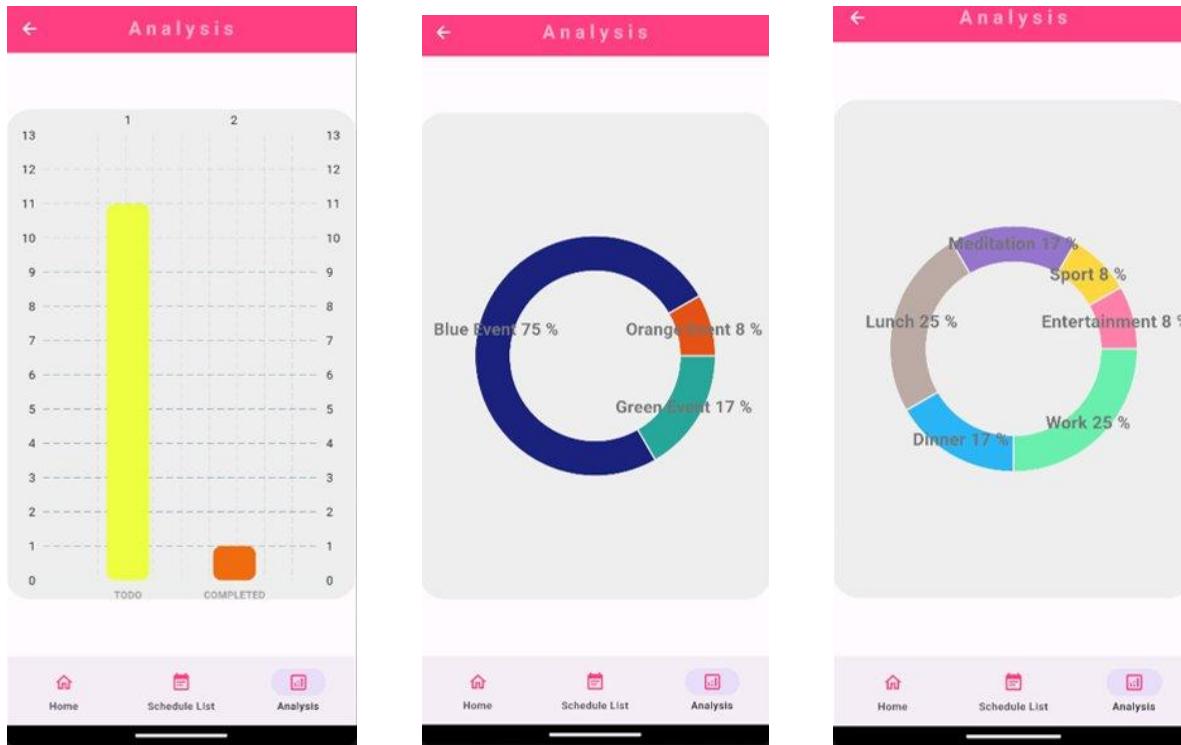


Figure 9,10&11: Event analysis page

Also, the analysis page provides users with insights into their scheduling habits and tendencies. One of the key features of this page is to display the number and types of events that users have added to their calendar (see figure 11). It can help them to understand their own scheduling preferences and habits. Additionally, the analysis page includes a bar chart that tracks the completion status of events, providing users with real-time feedback on their progress and helping to motivate them to stay on track and complete their tasks on time (See Figure 9 & 10). Also, there are notifications functions that remind users before the event starts.

1.4 Impact and Value

1.4.1 Impact & Value 1

One of the biggest challenges when dealing with various repeated records is the need to move the data to other places manually. This is a time-consuming process. It may lead to errors and decreased efficiency. However, using a text-scanner can significantly enhance the convenience of this task. It allows a quick and accurate transfer of repeated data to other locations with fixed format. This technology can increase efficiency and accuracy of coping data, which can reduce the risk of errors. By using a text-scanner, users can improve their operations of coping data. It enhances productivity and minimizes the risk of errors in their data.

1.4.2 Impact & Value 2

The other challenge is the risk of forgetting important or trivial matters. This can lead to missed appointments, deadlines, and other critical events. However, with digitalizing the event data, users can reduce the risk of forgetting important matters. With all records and details stored in a database, it becomes much easier to manage and track events, which can enhance the efficiency of event management. It is because knowing that all events are being managed and tracked.

1.4.3 Impact & Value 3

When we have a busy schedule, it can be challenging to find the time to consider all the details of each event. It may lead to missed appointments or errors. However, the Auto-Schedule feature can manage a busy schedule. It takes into account all the necessary factors when scheduling an event, such as the time required and the time duration. When an event does not require any specific time, it will automatically apply the event to a suitable period, which can make the schedule become more flexible. By

using the auto-scheduling feature, users can manage their busy schedules more efficiently, ensuring that all important events are accounted for and properly scheduled. This not only saves time but also reduces the risk of errors and oversights.

1.5 Structure of the Report

In chapter 1, we describe the current background of the E-calendar. Such as advantages, disadvantages and how the public treat the current calendar. Also, we introduce our project aim and objectives in this chapter. Proof that how do these objectives impact the public. Via analyzing the impact and objectives, the core value of our project is also uncovered.

In chapter 2, we will explore the TexPic technology in addressing various problems and challenges. We will provide a detailed discussion of the specific applications of this technology and how it can be utilized to enhance efficiency, and accuracy in different fields. Additionally, we will present evidence from academic journals and relevant research studies to support our claims.

In chapter 3, we will provide an overview of the methodology, including OCR, Auto-Schedule, and Calendar-View. We will explain each methodology in detail, outlining the major components and design elements that how they can enhance our project effectiveness. Through examination of each technology, we will provide evidence to support our assertion that they are suitable for the project. Additionally, we will talk about the design of prototype systems and those technologies' architecture. Explained how they work together to enhance the overall application functionality.

In chapter 4-6, we will talk about the significant achievements of those three methodologies: Calendar System, OCR, Auto-Schedule. We will examine how each of these methods works, exploring their underlying principles and the benefits they offer. Additionally, we will show the outcomes and results of tests conducted on each methodology which can provide a comprehensive understanding of their effectiveness and usability.

By examining the significant achievements of these methodologies, we can gain a deeper understanding of how they can improve the quality of TexPic.

In Chapter 7, we will provide a comprehensive summary of the project we have undertaken and evaluate whether we have achieved our stated aims and objectives. We will reflect on the challenges that we have encountered during the research and development process. Also, we will point out any limitations of our application and its improvement.

Chapter 2. Literature Review

OCR is a technology to automate the identification of text characters in an image or document (Karthick, 2019). It enables machines to read and interpret text from various sources, including printed documents, handwritten notes, and digital images. To make it easier and faster for people to process large amounts of text data, the text recognition system is the most suitable.

Texttract is one of the similar technologies that extract text and data from scanned documents or images. It is an AWS service that uses OCR utilizes advanced algorithms and machine learning models to recognize and extract text from various types of documents, including PDFs, images, and handwriting.

Other than Texttract, ABBYY FineReader is also an OCR software application that uses OCR technology to convert scanned documents, PDFs, and digital photos into editable formats. It has features such as automatic language detection, image enhancement, and table recognition that make it a powerful tool for digitizing paper documents.

Therefore, based on those OCR technology merits, one of the main functions scanning for record events is created. Tex-pic needs to recognize text and data from scanned documents or images. Since the PDF files are digital-born, we can get pretty printed textline images by converting them into page images and extracting the text lines with their cropped images (Li, Wang, & Shen, 2021). It means that through digitalization, the scanned data which are the event information become easy to manage.

In addition, Refers to the ABBYY FineReader, the formatted text recognition is useful for our application. Using the features such as font type, size, and spacing, as well as layout and formatting elements of tables and columns, our application can more easily recognize and extract text

from scanned documents or images. The use of consistent formatting in documents can also improve the accuracy (Walker, Walker, & Brooks, 2005), including OCR technology. It helps to reduce errors and inconsistencies in the recognition process.

After adding in those technologies, it can increase the recognition accuracy, task management efficiency and effectiveness.

On the other hand, the issue of time management has been widely debated in academia and practice. Various techniques and solutions have been proposed to solve this problem, ranging from traditional time management methods to modern technological solutions.

A popular time management method is the Eisenhower Matrix (Brett & Kate McKay, 2013), which categorizes tasks into four sections based on their urgency and importance. This method has been widely used in the business world and is known to be effective in improving productivity.

The Eisenhower Matrix is a popular productivity framework created by President Dwight D. Eisenhower. It helps individuals prioritize tasks based on their urgency and importance. The matrix is divided into four quadrants:

1. Important and urgent tasks: These tasks require immediate attention and should be completed as soon as possible.
2. Important but not urgent tasks: These tasks are important to achieve long-term goals and should be scheduled for later.
3. Urgent but not important tasks: These tasks are time-sensitive but do not contribute to long-term goals and can be delegated to others.
4. Not important and not urgent tasks: These tasks should be eliminated or minimized as they do not add value to one's goals or well-being.

One of the main benefits of using the Eisenhower Matrix is that it helps individuals focus on important tasks that contribute to their long-term goals. It also helps them avoid wasting time on unimportant or urgent tasks. Another benefit is that it helps individuals become more organized and efficient in managing their time.

However, one of the main disadvantages of using the Eisenhower Matrix is that it may not be suitable for everyone. Some people may find it difficult to categorize their tasks into all four quartiles, or they may have different criteria for important or urgent tasks. In addition, the matrix does not take into account external factors, such as unforeseen events or changing priorities, which may require individuals to adjust their plans.

The Eisenhower Matrix has become a popular technique for individuals looking to manage their time and prioritize their tasks effectively. The one of the productivity apps that has adopted this method is Todoist.

And we found that other productivity apps use different techniques to solve the problem of time management. Such as Trello and Asana.

Trello's automatic scheduling feature is implemented through its "Butler" plugin. Butler can automate actions on Trello cards, including automatically scheduling cards with due dates and moving cards to specific lists. Users can create automation rules through Butler, which can be simple "if/then" statements, such as "If a card is moved to the 'To Do' list, set its due date in three days." Users can write custom rules according to their own needs to meet different automatic scheduling needs. (<https://trello.com/power-ups/category/automation>)

Asana's automatic scheduling function is implemented through its "Rules". Rules allow users to automate tasks, such as automatically changing task status based on certain conditions, moving tasks to different projects or

departments, or adding instructions or attachments to tasks at a specified time. Users can use the preset rules or custom rules in Asana to meet different automatic scheduling needs.(<https://asana.com/zh-tw/guide/help/premium/rules>)

After we did research on the above methods, we already have some ideas about Auto-schedule. The auto-scheduling features on Asana and Trello are great, but those methods are too complicated for the non-computer literate. And also it will take a lot of time during the operation. The purpose of Auto-schedule function should be efficient and time-saving. So we need to improve those method to make the function easy to use .

The automatic scheduling functionality proposed in this project builds on these existing solutions by providing automatic scheduling functionality that considers user preferences and availability. By harnessing the power of computer algorithms, we aim to provide more efficient and personalized solutions for managing your daily tasks.

Chapter 3. Methodology

3.1 Overview of Solution

3.1.1 Programming Language and Framework:

The whole mobile application is developed by Flutter framework and Dart language. Flutter is an open source framework by Google for building multi-platform and natively compiled applications, the application that developed by Flutter framework is fast and easy. It is good for our calendar app development because the Flutter framework and Dart language are easy and clean.

Pub is the package manager for the Dart language and Flutter app, containing reusable libraries and packages for Flutter and general Dart programs. There are many useful packages and libraries for our development. We use the following packages and libraries for building our calendar app

```
37  cupertino_icons: ^1.0.2
38  get: ^4.6.5
39  get_storage:
40  google_fonts: ^4.0.3
41  sqflite: ^2.2.5
42  flutter_neat_and_clean_calendar: ^0.3.13+28
43  intl: ^0.18.0
44  flutter_staggered_animations: ^1.1.1
45  flutter_local_notifications: ^13.0.0
46  flutter_native_timezone: ^2.0.0
47  fl_chart: ^0.61.0
48  camera: ^0.10.0+4
49  google_mlkit_text_recognition: ^0.4.0
50  permission_handler: ^10.2.0
51  image_picker: ^0.8.4+4
52  path_provider: ^2.0.7
53  path: ^1.8.1
54  provider: ^6.0.1
```

Figure 12. The Dart packages of we used

3.1.2 Integrated Development Environment (IDE)

We use Android Studio for the development environment. Android Studio is an official IDE for Google's Android OS. Android Studio is built on JetBrains' IntelliJ IDEA and designed specifically for Android development. It provides an emulator for the smartphone with Android OS and we can test the app in that emulator. It allows us hot reload the code so that we can test the code fastly.

3.1.3 Idea Structure of Calendar Application

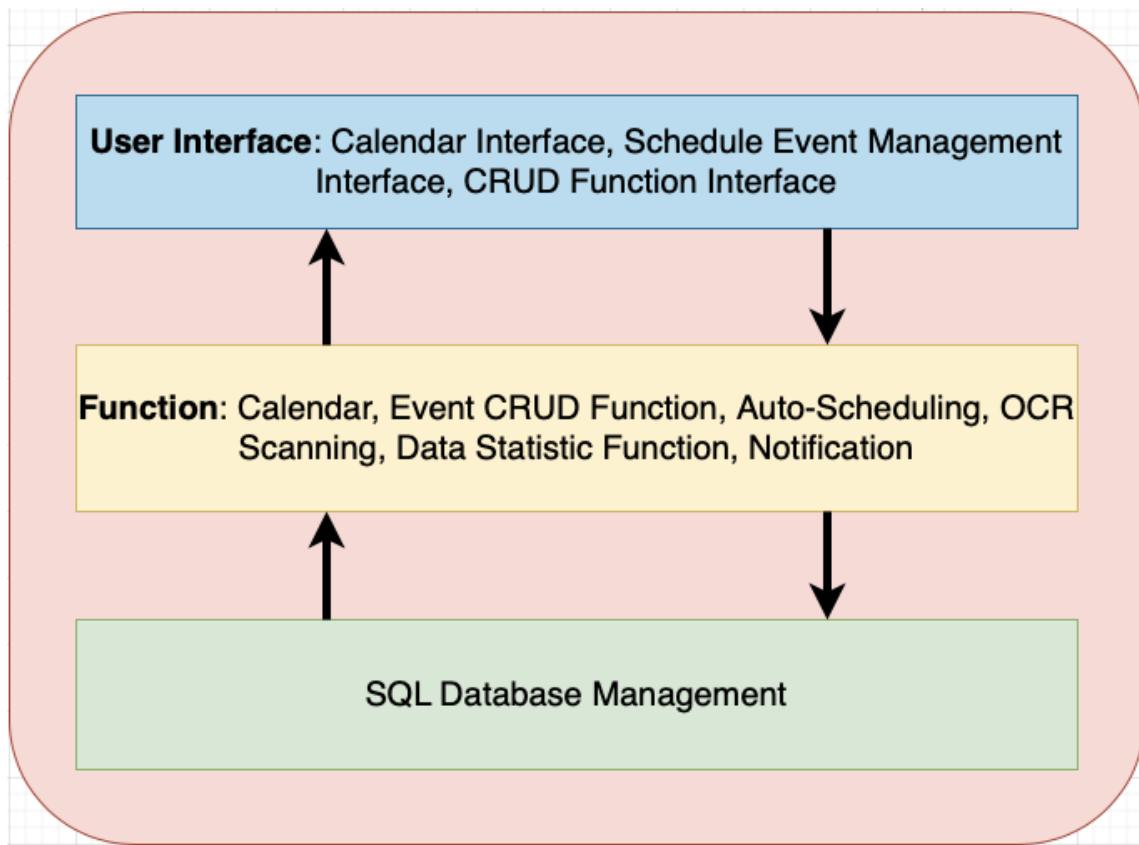


Figure 13. The conceptual layers of the calendar app

Our development process focuses on Figure 13 three layers: User Interface Layer, Function Layer and Database Layer.

User Interfaces contain all the UI elements, for example: calendar view, schedule event part, create, read, update, delete (CRUD) operations UI and so on.

Function is the main part of the calendar application. Those functions need to connect with the UI and Database so that it can perform to the user and store the user data.

Database is the part of storage user data and function parameters. It is useful for the user to use the calendar application for the long term.

Those three layers are the whole application core part.

3.2 Design and Major Components

3.2.1 Development Process

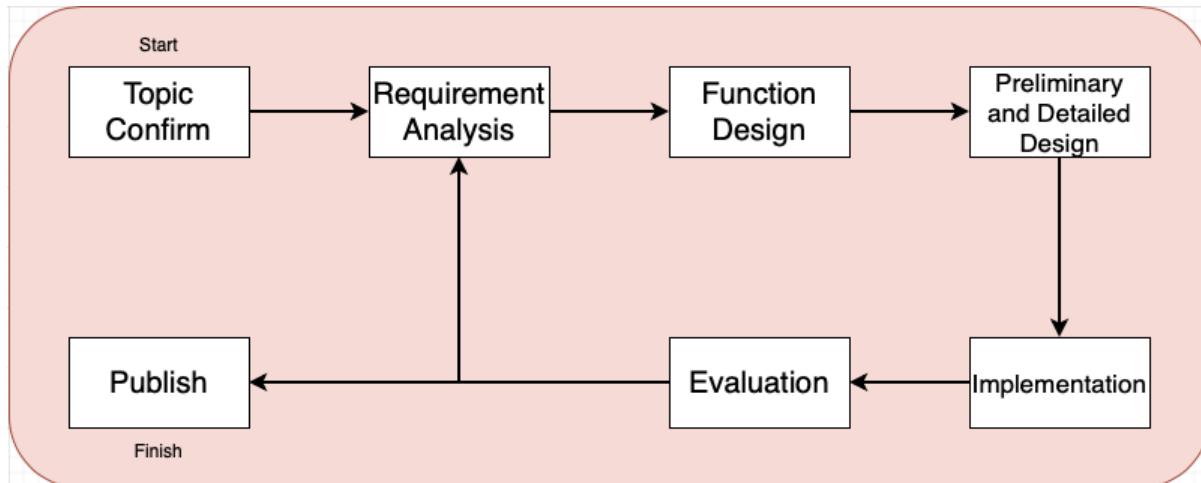


Figure 14. Development and design process for the mobile application.

We follow the steps for the application development and design. Requirement analysis, function design, preliminary and detailed design, implementation and evaluation is a cycle of the development process. There are many iterations of the application development so that the product is more suitable to users and approaches the topic.

3.2.2 Design Diagram

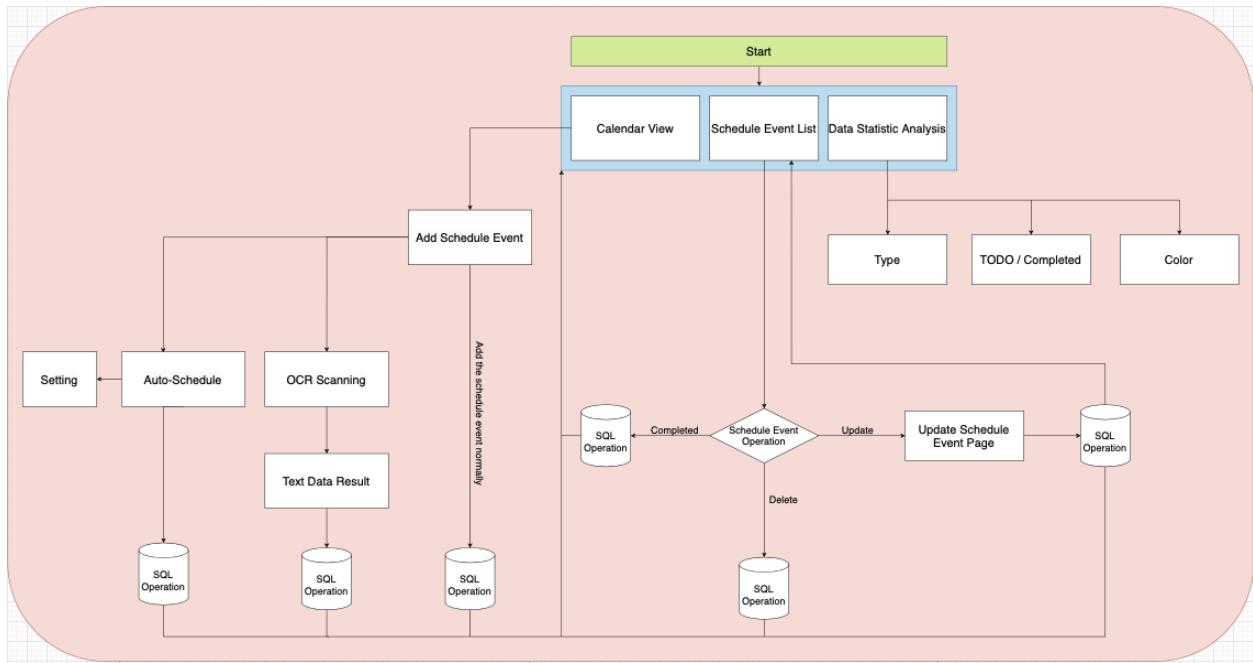


Figure 15. The design diagram of the calendar application

Figure 15 shows the design diagram of the TexPic Calendar App. It is a complete diagram of the application flow.

The UI of the calendar can be divided into three parts: Calendar View, Schedule Event List and Data Statistic Analysis. All the functions are related to the SQL operation.

3.2.3 Calendar Application Architecture

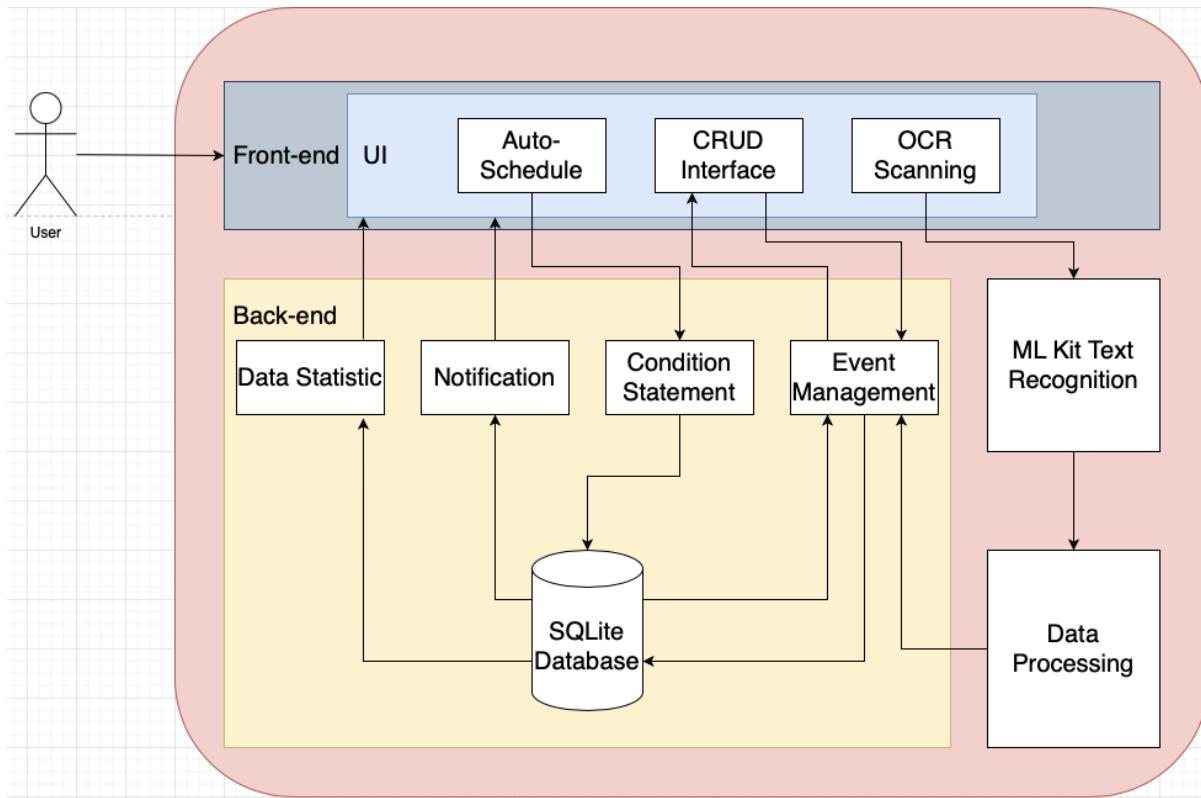


Figure 16. The application architecture

Figure 16 shows the Architecture of the TexPic Application. It has three parts: the front-end, backend and the ML Kit for text recognition.

For the front end, It has the calendar UI, CRUD operation UI, Auto-Schedule and OCR scanning. When a user accesses the application, the UI is shown—the Auto-Schedule and Scanning function provided to the user. The Scanning will send photos to the ML Kit text recognition tool, which is the API from google ML Kit. Then process the data which is the result from text recognition. The UI will finally receive the response from the backend of the event management. Auto-Scheduling sends the request to the backend of the condition statement to handle.

For the backend, the event management receives requests from the front-end, ML Kit text recognition and the database. If there is a list-checking request, event management will request the SQLite database

to get the data and send it to the front-end. Also, the Auto-Schedule function will send a request to the back-end. Via the condition statement (if-else statement), the system will choose the most suitable duration and set it to the database.

For the ML Kit text recognition API, the image will be sent to the ML Kit API and return the text data. Data processing will format the result data and send it to the event management.

3.2.4 Optical Character Recognition —Text scanning

For the technology of OCR, we will use ML-kit to achieve text-recognition. ML-kit is a google API that allows developers to easily integrate machine learning features into our applications. One of the technologies is called text-recognition. It contains several key capabilities. First of all, it recognizes text across various scripts and languages. Regardless if it is Chinese or English or Japanese, the learning machine will distinguish the language. Analyzing structure of text which can detect symbols, elements, lines and paragraphs.

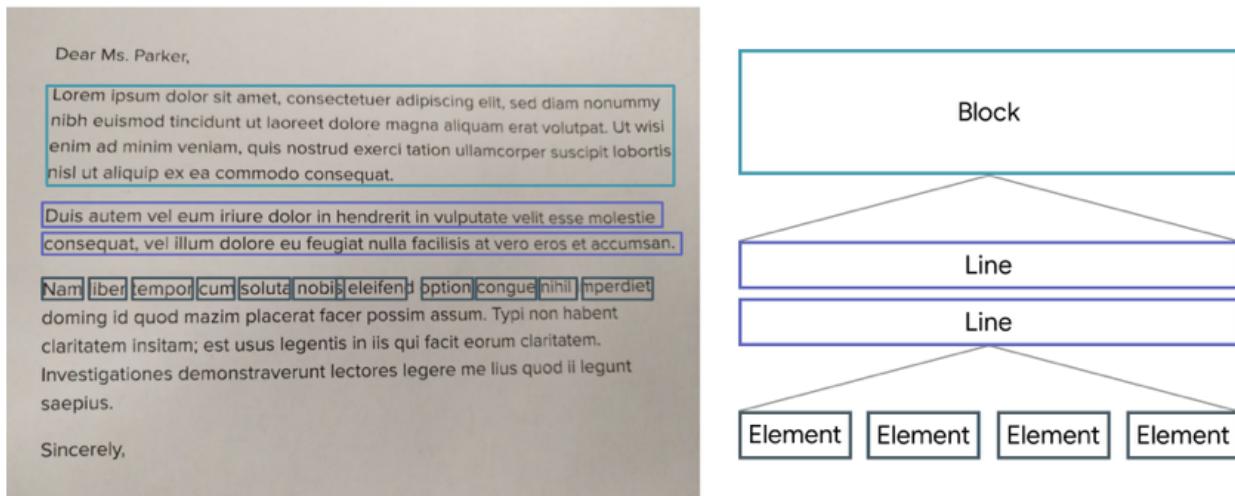


Figure 17. the structure of `recognizedText` variable

The OCR result `recognizedText` is a block of the text, it can be divided to lines, and each line can be divided to elements. The element is a word in the image. In our application, we use the function for-loop to separate the element. In the for-loop statement, it has a simple format process for the input format.

3.2.5 Auto-schedule

The Auto-schedule is the solution to create a function that could efficiently allocate tasks to available time slots while avoiding overlaps and conflicts. The core approach to achieving this was to classify events into different types and allocate them based on availability.

To accomplish this, the Auto-schedule function was designed with several major components: Classify events, Setting event time, Allocate time slots, , Resolve conflicts.

Classify events: The app first classifies the events into different types, such as work, sport , dinner , or other. And then default the setting time for each event .

Setting event time: Users can set the start and end times of different event types (such as lunch, dinner, sport, etc.) through the settings page. These settings will be used by the auto-scheduling algorithm to allocate time slots for events in a personalized way based on the user's preferences and availability.

Allocate time slots: The app then allocates time slots for each event. This involves identifying available time slots in the user's calendar and assigning the event to the most appropriate time slot.

Resolve conflicts: If there are scheduling conflicts between events, the app will try to resolve them by assigning the task to the next available 30-minute time slot. The app will search for available time slots between the start and end times of the conflicting events and allocate the task to the next available slot. If there are no available time slots on that day, it will move on to the next day. If no available time slots are found in the next 7 days, the function will return a message informing the user that there are no available time slots.

These components work together to ensure that tasks are allocated to the most appropriate time slot.

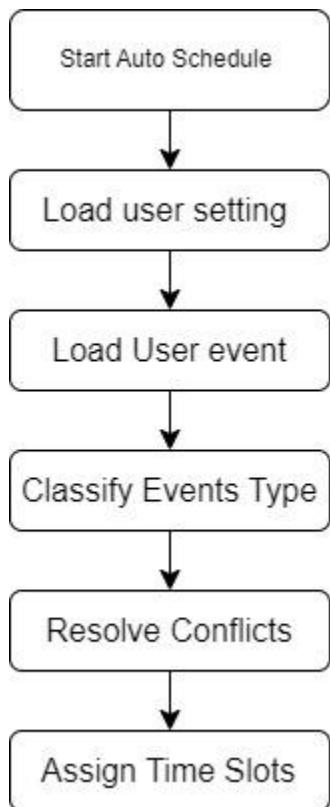


Figure 18. Simple flow of the Auto-Schedule

The Figure 18 Show the Simple flow of the Auto-Schedule,

The app will first classify the event into different types, and each event will have the default start time and end time. Users can pre-set the time by their personal preferences through the settings page. So that the Auto-Schedule function can automate the scheduling of tasks for a given time period while taking into consideration the user's preferences and availability. The ultimate goal is to provide users with a personalized and efficient schedule that meets their needs and helps them achieve their goals

3.3 Design of Prototype System

3.3.1 Low-Fidelity Prototype

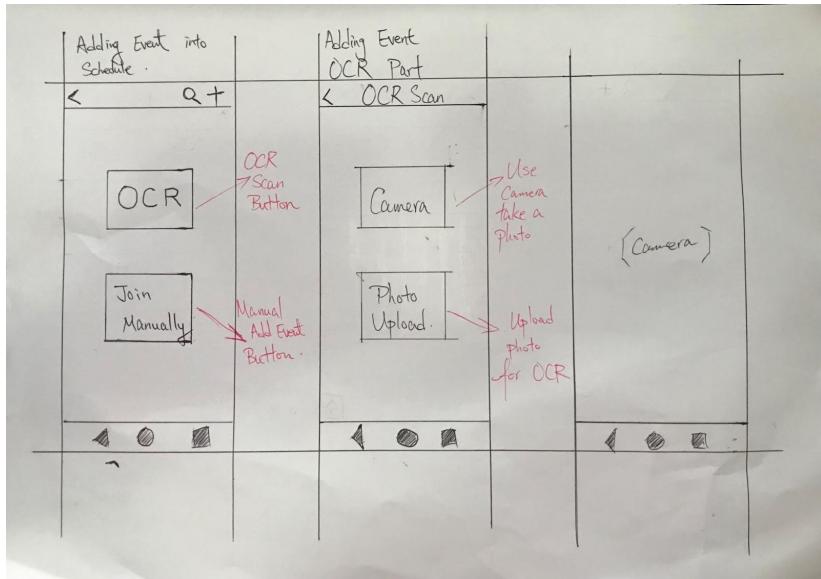


Figure 19. Hand draw low-fidelity prototype 1

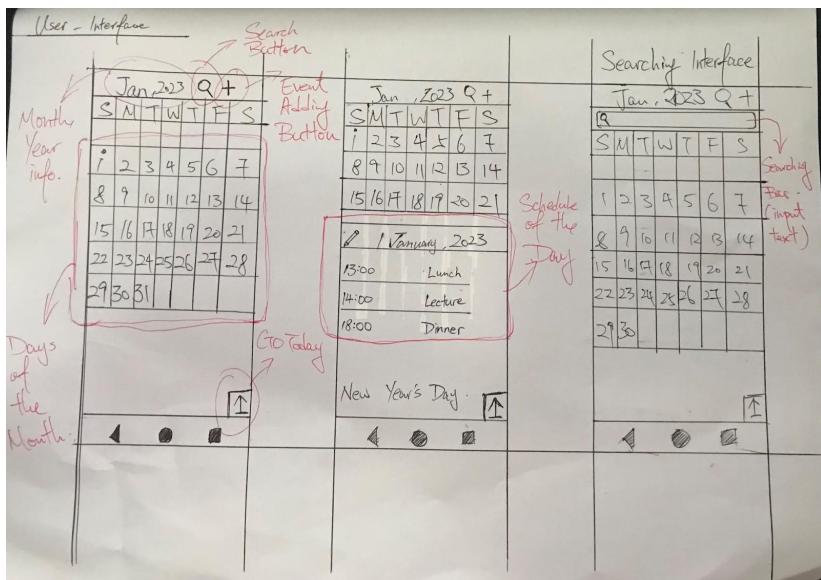


Figure 20. Hand draw low-fidelity prototype 2

Low-fidelity prototypes show the UI design and layout idea. It draws the calendar layout roughly and specifies the use of the layout components.

3.3.2 Mid-Fidelity Prototype

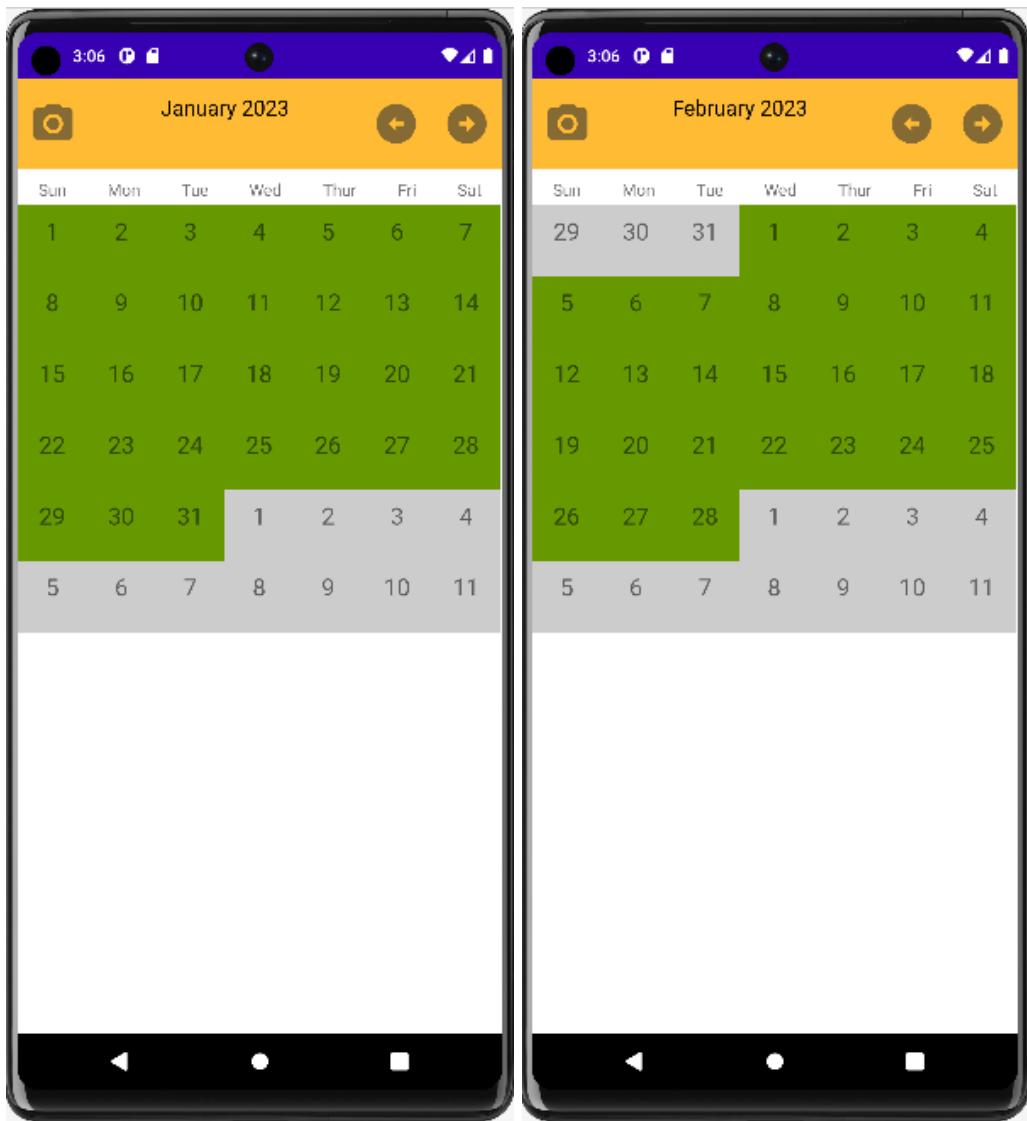


Figure 21 & 22. Mid-fidelity prototype of the calendar UI

Figure 21 & 22 are the mid-fidelity prototypes of the TexPic Calendar App. It can show the UI on the emulator and interact with the user. But it is not the final version of layout because it is still simple and crude. It still has no function like CRUD operation or OCR scanning.

3.3.3 High-Fidelity Prototype

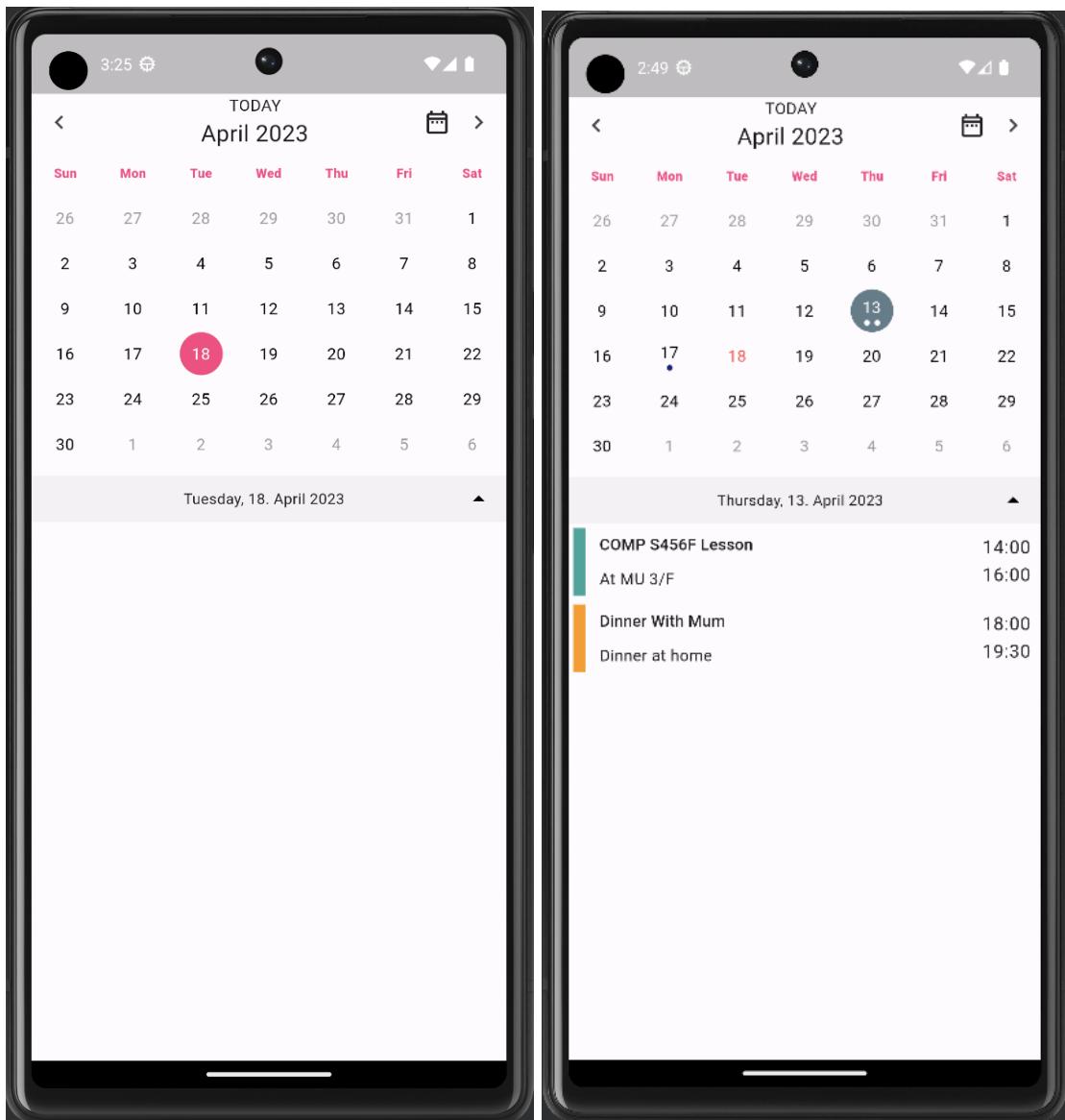


Figure 23 & 24. High-fidelity prototype of the calendar UI

Figure 23 & 24 shows the high-fidelity prototype of the TexPic Calendar App. That is the calendar view's final layout of the app. We will add the functions based on this layout.

3.3.4 Final Product

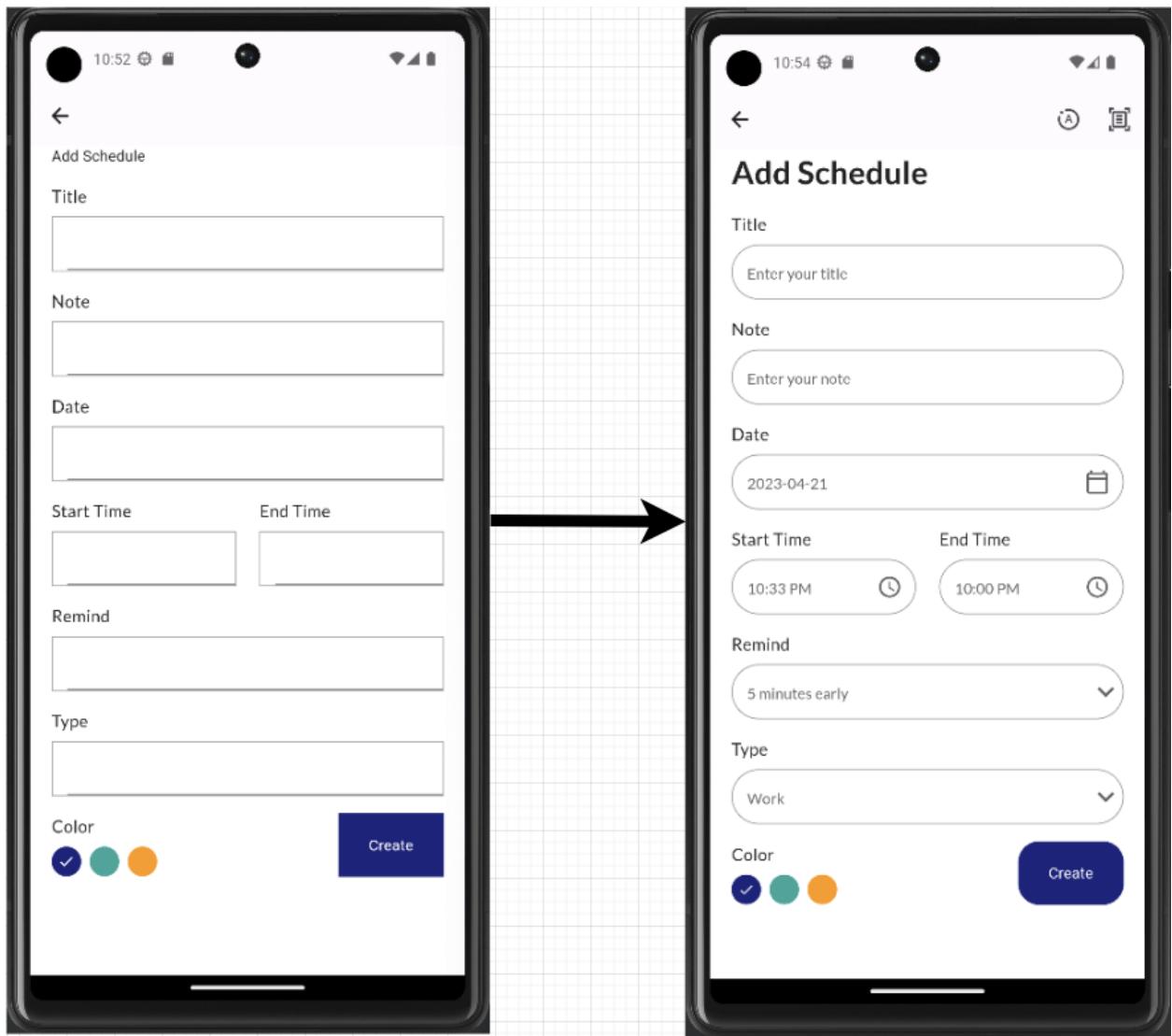


Figure 25 & 26. event adding page UI

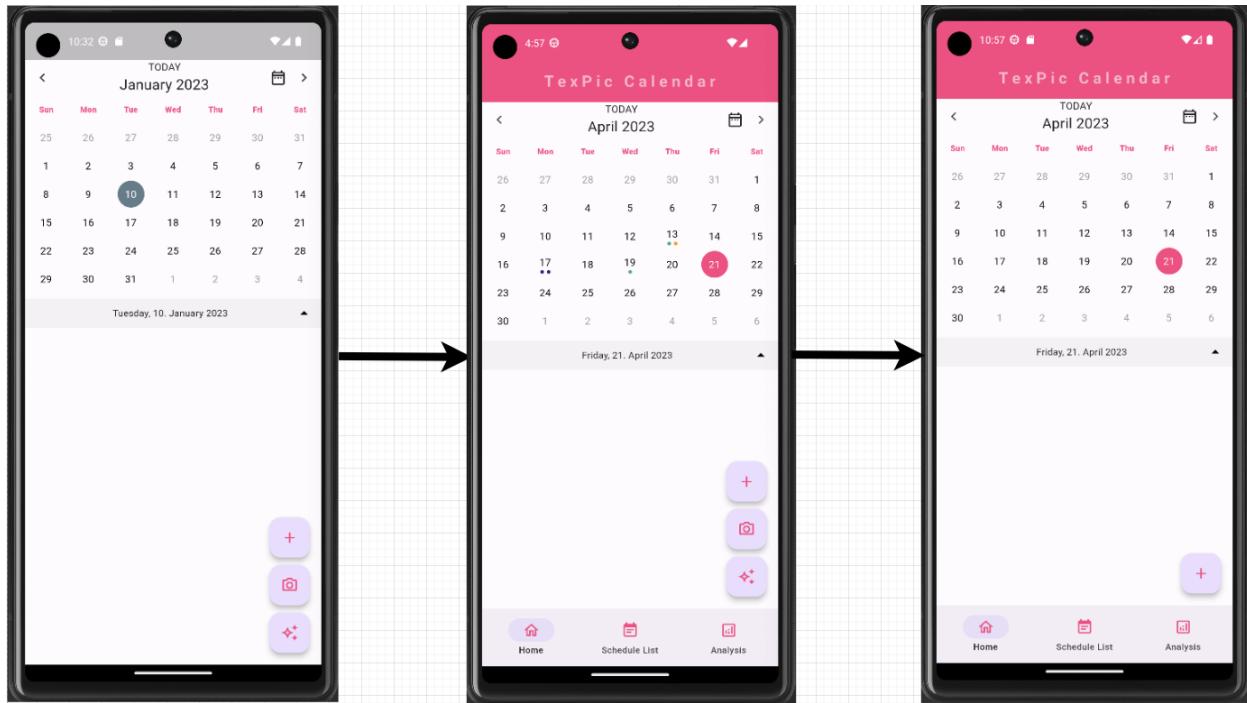


Figure 27,28 & 29. home page UI

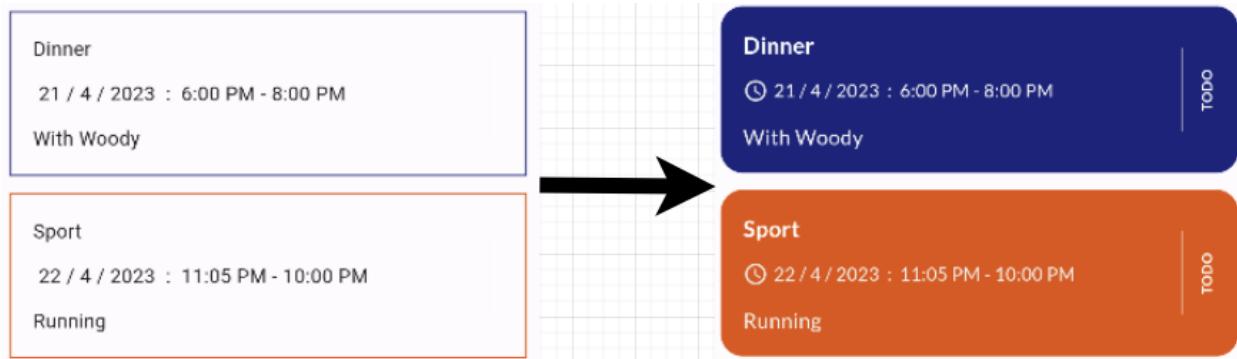


Figure 30 & 31. block UI

Figure 27 to 31 shows the final product UI layout optimization. Based on the high-fidelity prototype, we combine some buttons, add the app bar and navigation bar so that the user can access different functions of the calendar app. We have the theme color which is pink, and we will use Material 3 widgets in the calendar application.

3.4 Requirement

Functional Requirement	Non-functional Requirement
Task management	UI design
OCR text scanning	Performance of the application
Auto-scheduling	
Data analysis	

Table 1: Functional & Non-function Requirement

3.4.1 Functional requirements:

1. Task management is the basic function of the calendar event which includes CRUD operations. CRUD operations are related to the SQL database. Users can modify their own schedule event and the app will show the event to the user correctly.
2. OCR text scanning is one of the input methods for creating events. Users can import a gallery' image or take a photo for the event data. OCR can recognize the text data and autofill into the input field of the event adding page.
3. Auto-scheduling is a function for Automatically scheduling events. Users can set the personalization event during, then the system will automatically apply the event to the schedule.
4. Data analysis is a function of showing the event data to the user. Function will collect the event data from the database. Then calculate the percentage of each event in the total amount. The value will be shown to the user by using pie charts and bar charts.

3.4.2 Non-functional requirements:

1. UI design is an important part of our application. That provides a clean and user-friendly layout and style. The UI should keep clean and provide sufficient information to the user.
2. Performance of the application is mainly part of non-functional requirements. The calendar app performance should be fast, and it can update the information immediately. So we use Flutter framework and GetX for state management.

3.5 Supporting Technologies

3.5.1 Flutter Neat and Clean Calendar Package

It provides a clean and neat UI calendar view. That will be the base of the calendar app. We will design the UI and add the function based on this calendar.

3.5.2 OCR Scanning - ML Kit Text Recognition Flutter API

By using OCR scanning, the user can input the image and process to be data filled into the adding event page automatically. ML Kit provides an API for text recognition. The API receives an image and outputs a text. We will process the text result so that it can fit the format of our function.

Chapter 4. Calendar Application Evaluation

4.1 Firebase Robo Testing

Robo Test is a mobile app test in Firebase platform. Robo Test automatically explores your app on various devices to find defects and report any crashes that occur. Robo Test doesn't require us to write app tests, and it can provide a detailed testing report, testing step and recording to us.

4.1.1 Testing Result

The screenshot shows the Firebase Test Lab results page for a Robo test. The top navigation bar includes the Firebase logo, project name 'matrix-y6sg8ltoya1ha', and user profile icons. The left sidebar has sections for '專案總覽', 'Test Lab' (which is selected), 'Analytics Dashboard', and '所有產品'. The main content area displays a summary card for a 'Robo 測試' run on April 21, 2023, at 4:22 PM. The card shows 0 failures (red triangle icon) and 1 pass (green checkmark icon) across 1 device. A call-to-action banner encourages using ARM virtual devices with a 'View ARM devices' button. Below the summary is a detailed table for the 'Pixel 5, API 層級 30' device, showing 1 pass (green checkmark) and 0 failures (red triangle). The table also includes columns for '英文 (美國)' (language), '語言代碼' (language code), and '螢幕方向' (screen orientation). At the bottom of the page, there's a welcome message for customizing your profile, a 'Spark' plan offer, and a 'View details' link.

Figure 32. Robo test result 1

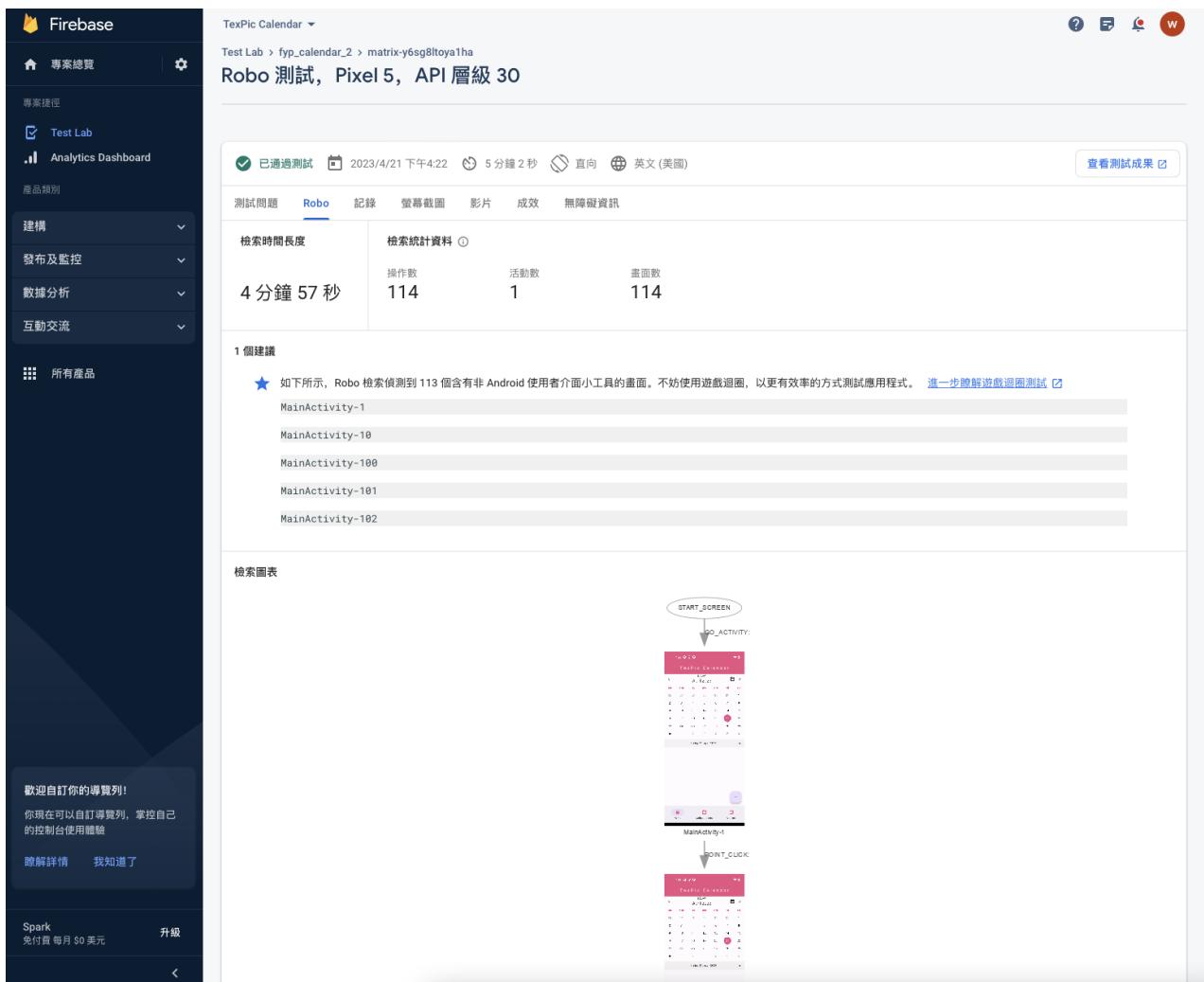


Figure 33. Robo test result 2

Figure 32 and 33 show that TexPic Calendar passes the testing, that means our application has no any big error or bug. All widgets and services work together as expected.

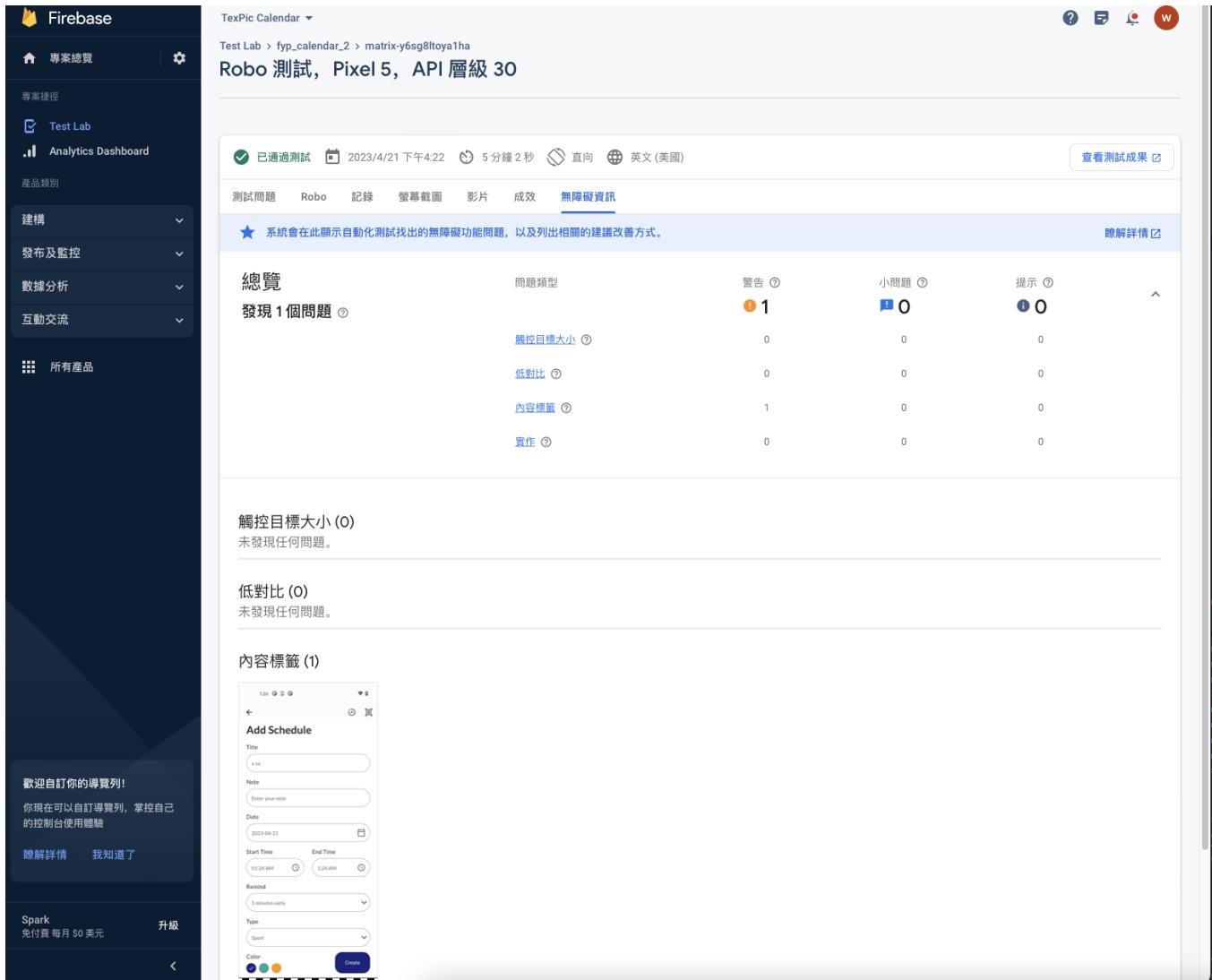


Figure 34. Robo test result 3

Figure 34 shows the accessibility information report. Robo Test warning the application has no content tag for the user, it will affect the accessibility for all users. Users of accessibility services, such as screen readers, must refer to content tags to understand the meaning of interface elements. If elements of the user interface do not provide content labels, some users may find it difficult to understand the information conveyed by these elements or to know what to do through the interface.

That is our application problem and we need to fix this problem.

TexPic Calendar has no major issues at accessibility performance other than the content tag.

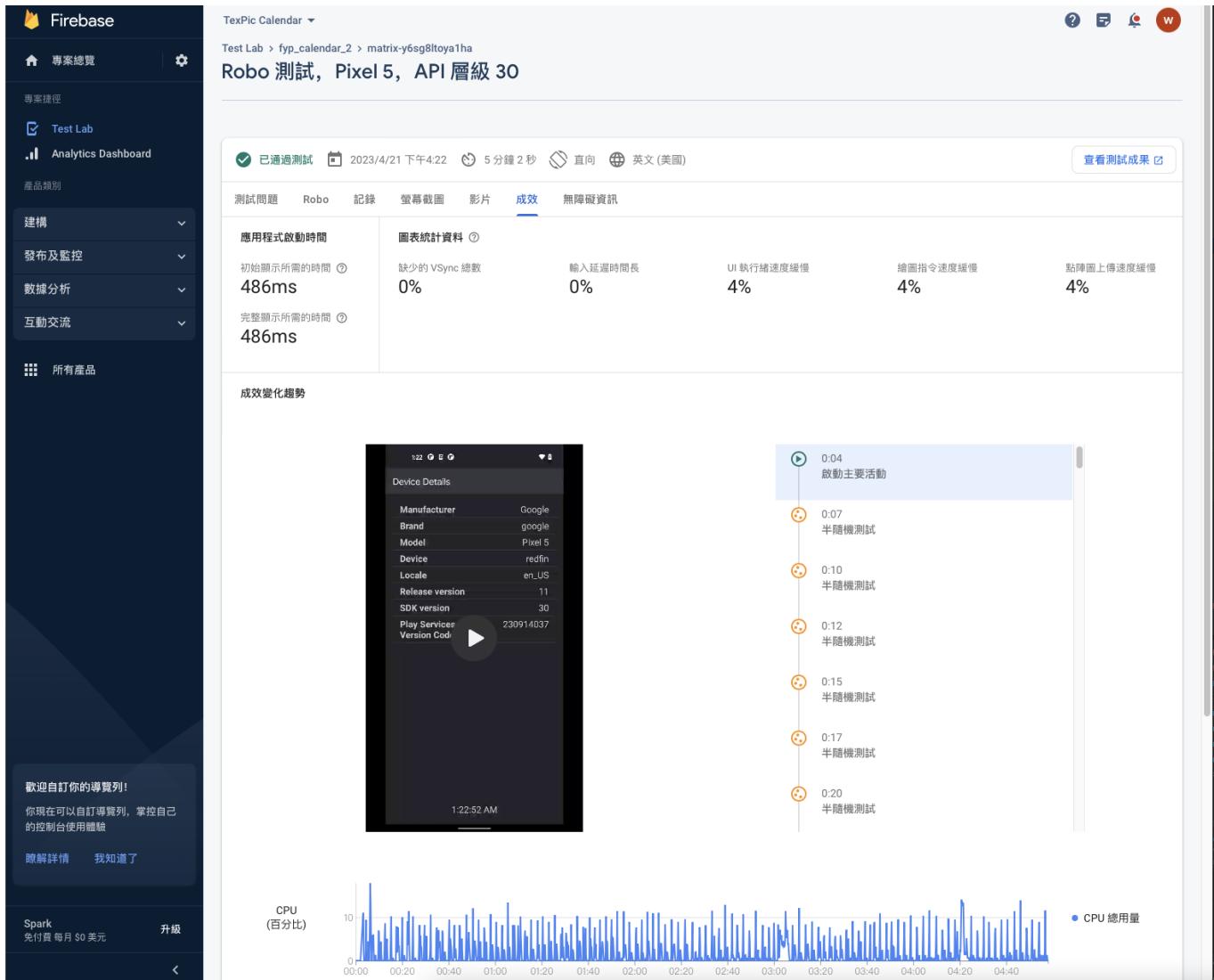


Figure 35. Robo test result 4

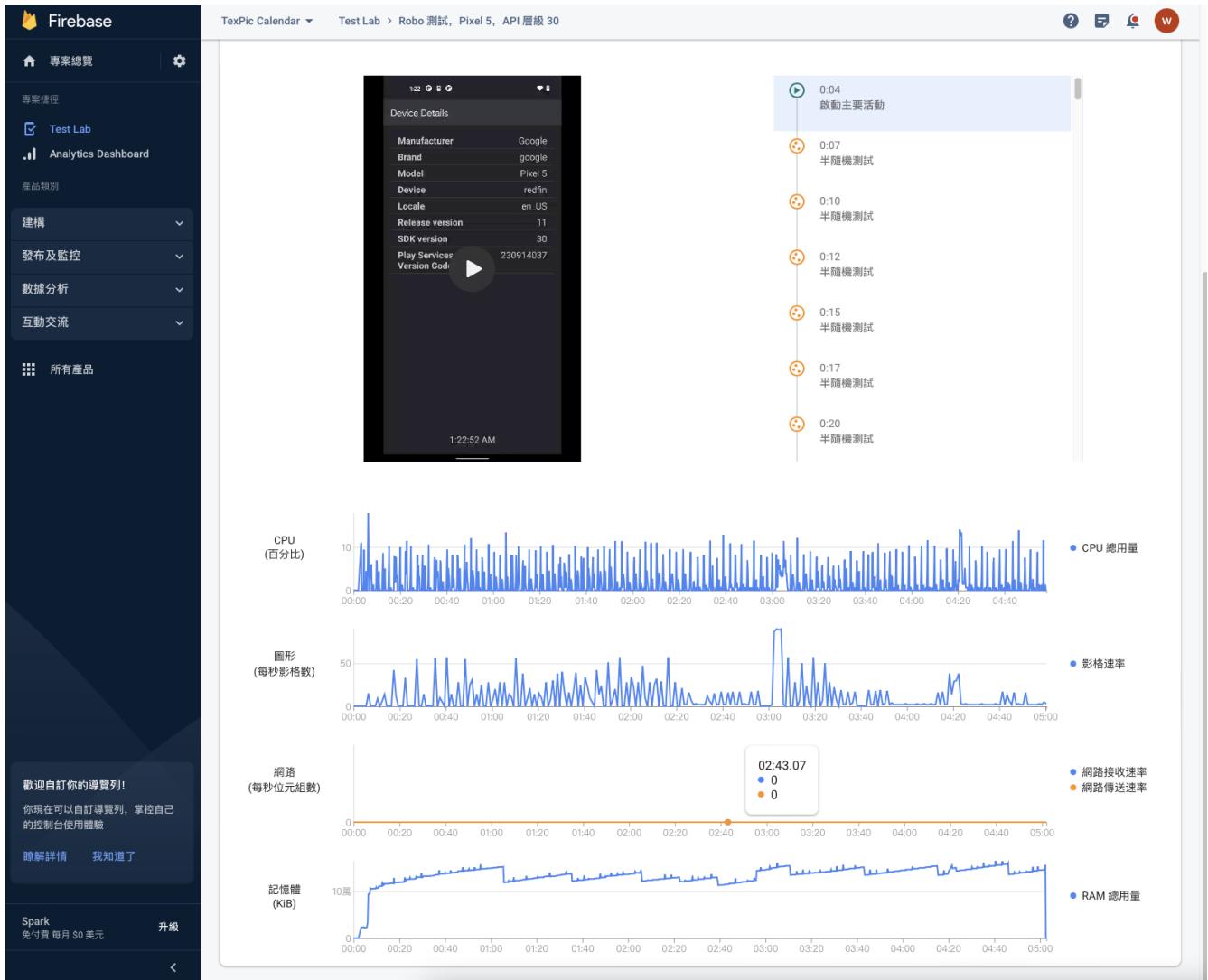


Figure 36. Robo test result

Figure 35 and 37 show the program performance in Robo Test. Time required for full display of TexPic Calendar is 486 ms. UI thread latency, drawing command speed and bitmap upload latency are 4% only. That results in the acceptable range.

4.1.2 Conclusion of Robo Test



Figure 37. the layout of the input field

The program performance , widget, service and program logic are passed in the Robo Test. For the accessibility information, we need to improve the content tag so that TexPic Calendar is suitable for all users. For example: we added the hint into the input field so that the user can know what information they need to write. But we still need to add the content tag for the widget so that all input fields have more accessibility.

4.2 Survey of TexPic Calendar

We have a survey of the TexPic Calendar. In this survey, we show the TexPic Calendar final product to the respondents so that they can rate our final product by UI design and functionality.

4.2.1 Survey Result - UI Design

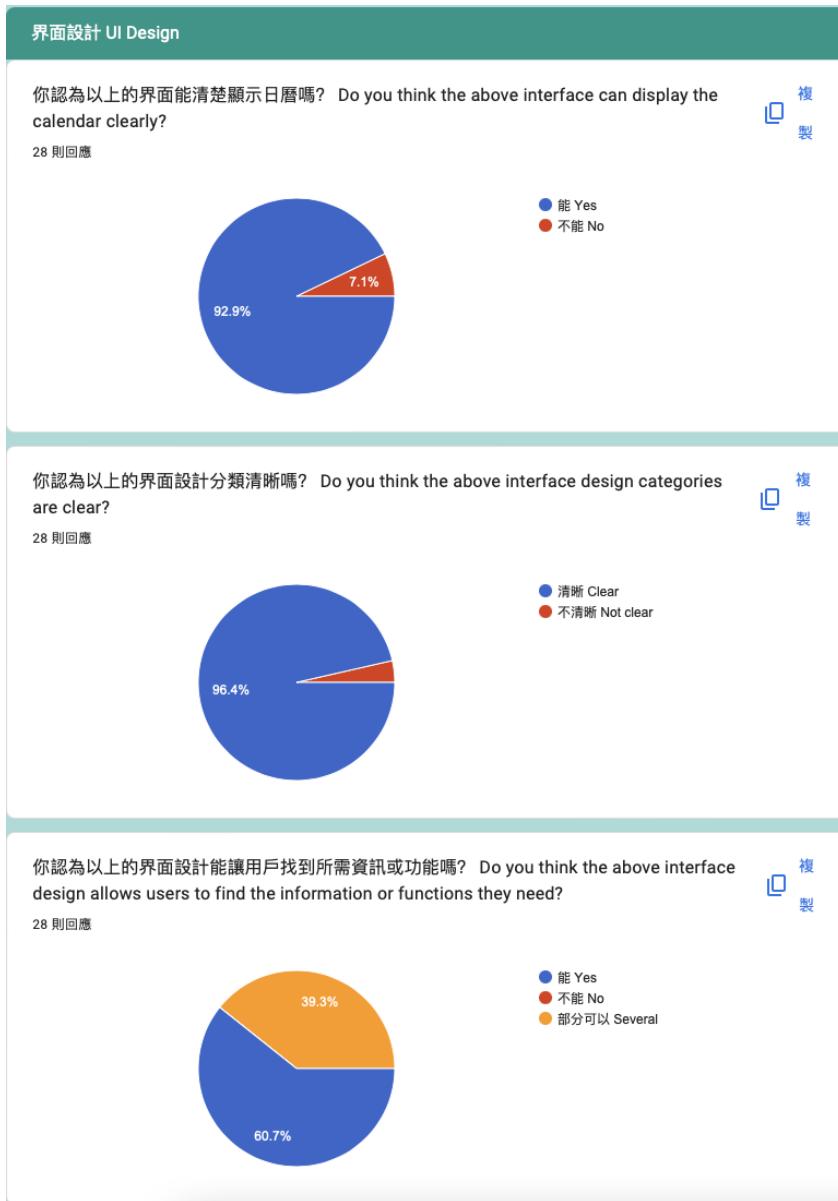


Figure 38. the survey result of UI design

Figure 38 shows the result of the survey. Over 90% respondents agree TexPic Calendar UI design is clean and provides a clean and neat style calendar. Most of the respondents think that our UI provides enough hints for finding out target functions.

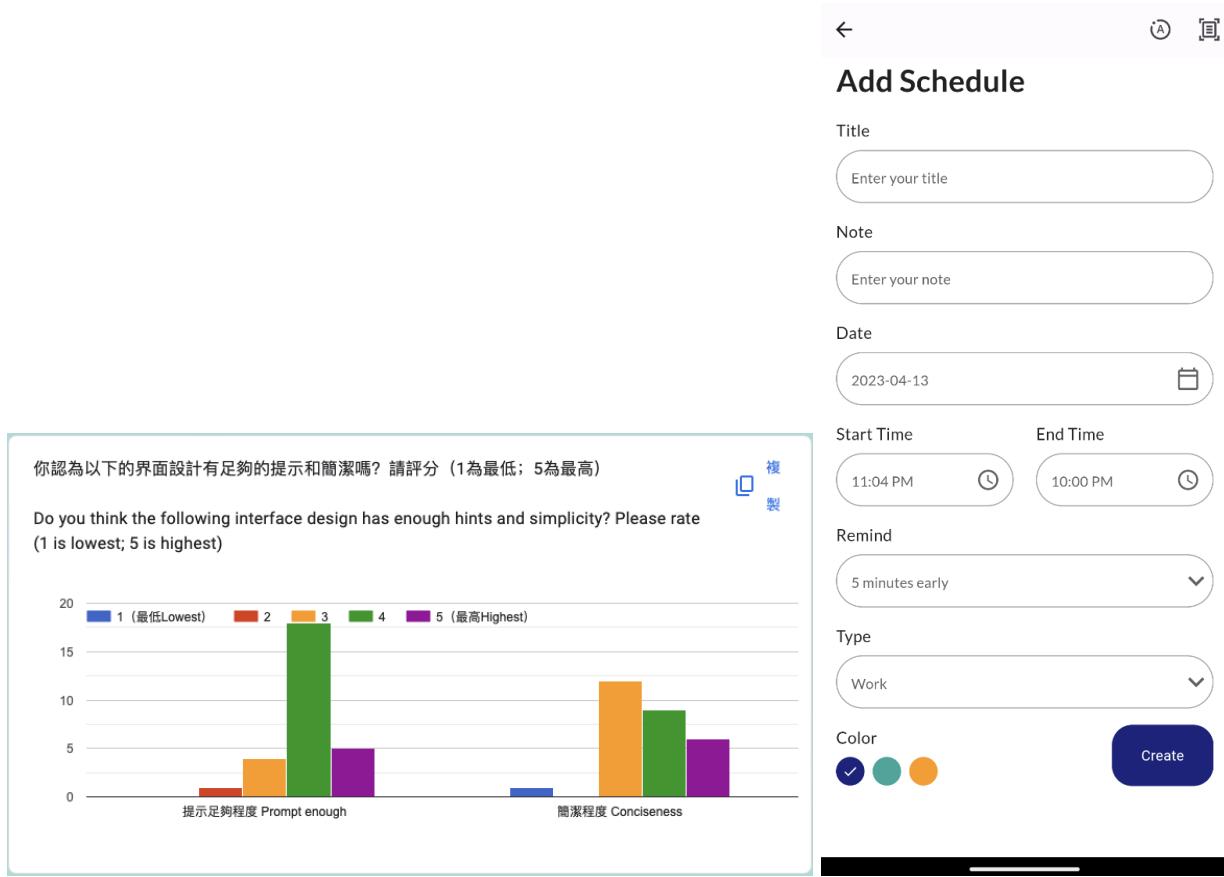
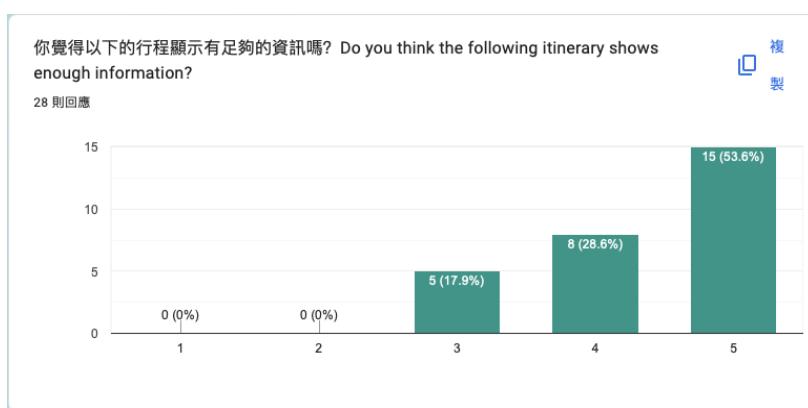


Figure 39 & 40. the score of event adding page UI design

Figure 39 & 40 shows the score of the event adding page. Respondents think our UI design has enough prompt, that means it can lead the user input correct information of the event.

For the conciseness of UI design, respondents think our design is qualified. That means our design can show the widget neatly but not clean.



Sport at MUHK

⌚ 17 / 4 / 2023 : 09:30 AM - 12:30 PM

Running with WOODY

TODO

COMP S456F Lesson

⌚ 13 / 4 / 2023 : 2:00 PM - 4:00 PM

At MU 3/F

TODO

Dinner With Mum

⌚ 13 / 4 / 2023 : 6:00 PM - 7:30 PM

Dinner at home

TODO

Figure 41 & 42. the score of event adding page UI design

Figure 41 & 42 shows the score of the event block, which is showing the event information to the user. It should provide sufficient information to the user clearly. All respondents think that our UI design is qualified, and over 50% respondents think our event block can show the information perfectly.

4.2.2 Survey Result - Functionality

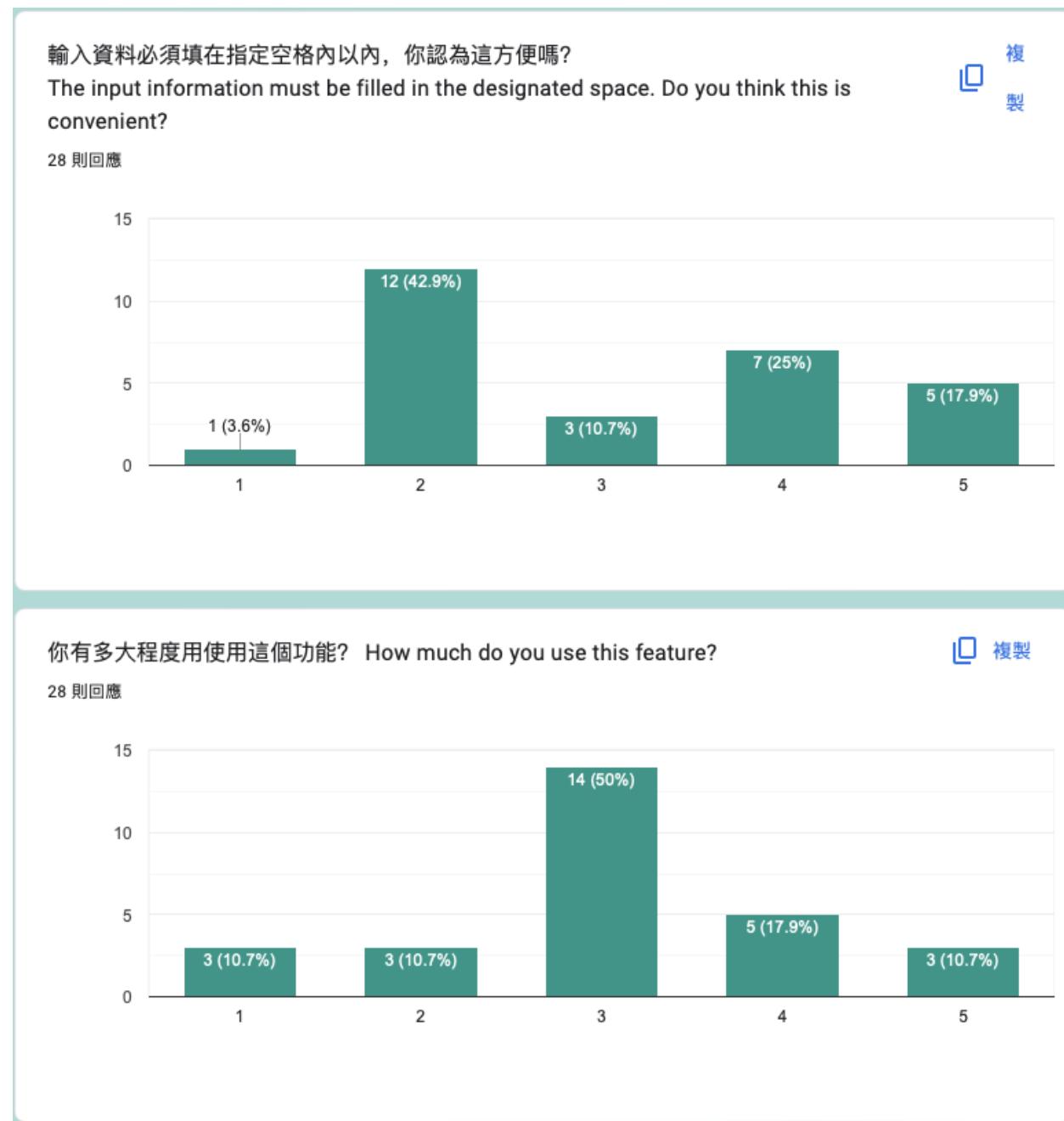


Figure 43. survey result of OCR function

Figure 43 shows the result of OCR function. Over 50% respondents agree that OCR scanning for input methods is convenient. And nearly 80% of respondents will try to use this function.

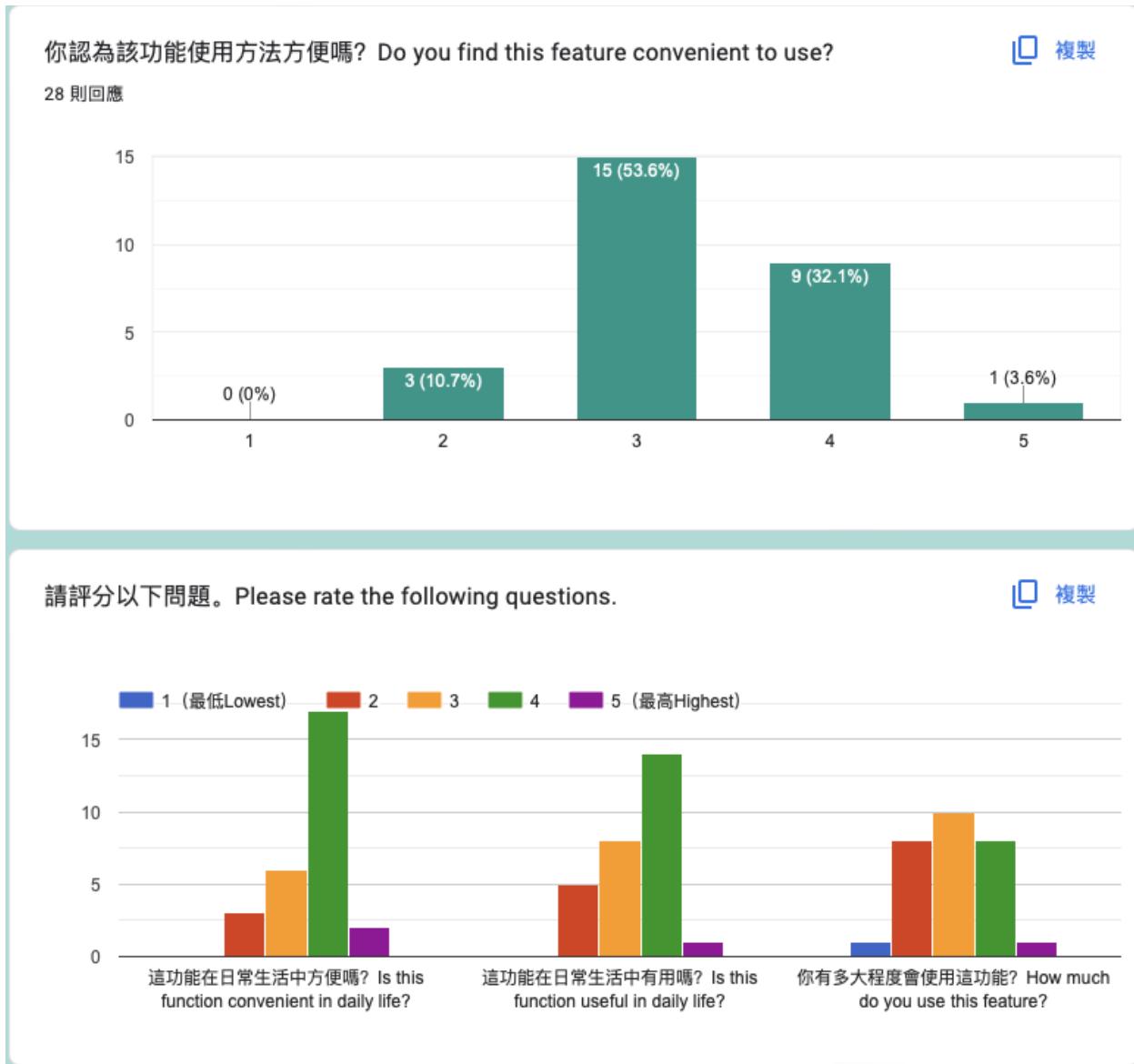


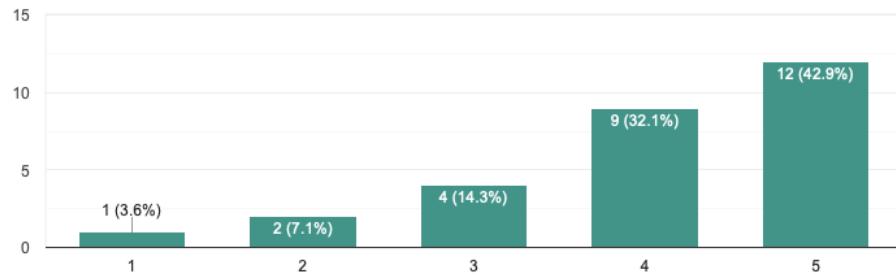
Figure 44. the result of auto-scheduling function

Figure 44 shows the result and opinion of the auto-scheduling function. Over 80% of respondents think the function is convenient to use. Over 50% of respondents think that is convenient and useful in daily life, they will try to use this function.

你認為以下數據圖所反映的結果清晰嗎? Do you think the results reflected in the data graph below are clear?

複製

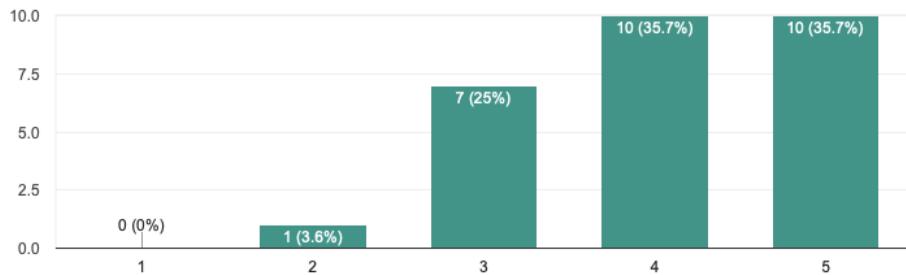
28 則回應



你認為此功能能夠反映用戶的日常數據嗎? Do you think this feature can reflect the user's daily data?

複製

28 則回應



你會經常檢視該數據嗎? How often do you review this data?

複製

28 則回應

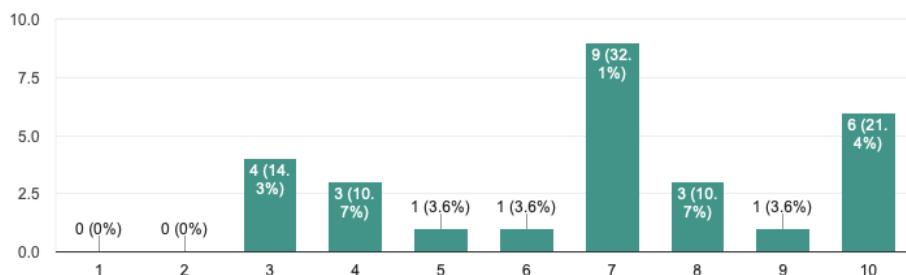


Figure 45. the result of data analysis page

Figure 45 shows the result of the data analysis page. Nearly 90% of the respondents the pie chart and bar chart show the data result clearly, and it

can reflect the user's daily data. Nearly 70% of the respondents will review the data oftenly.

4.2.3 Conclusion of Survey

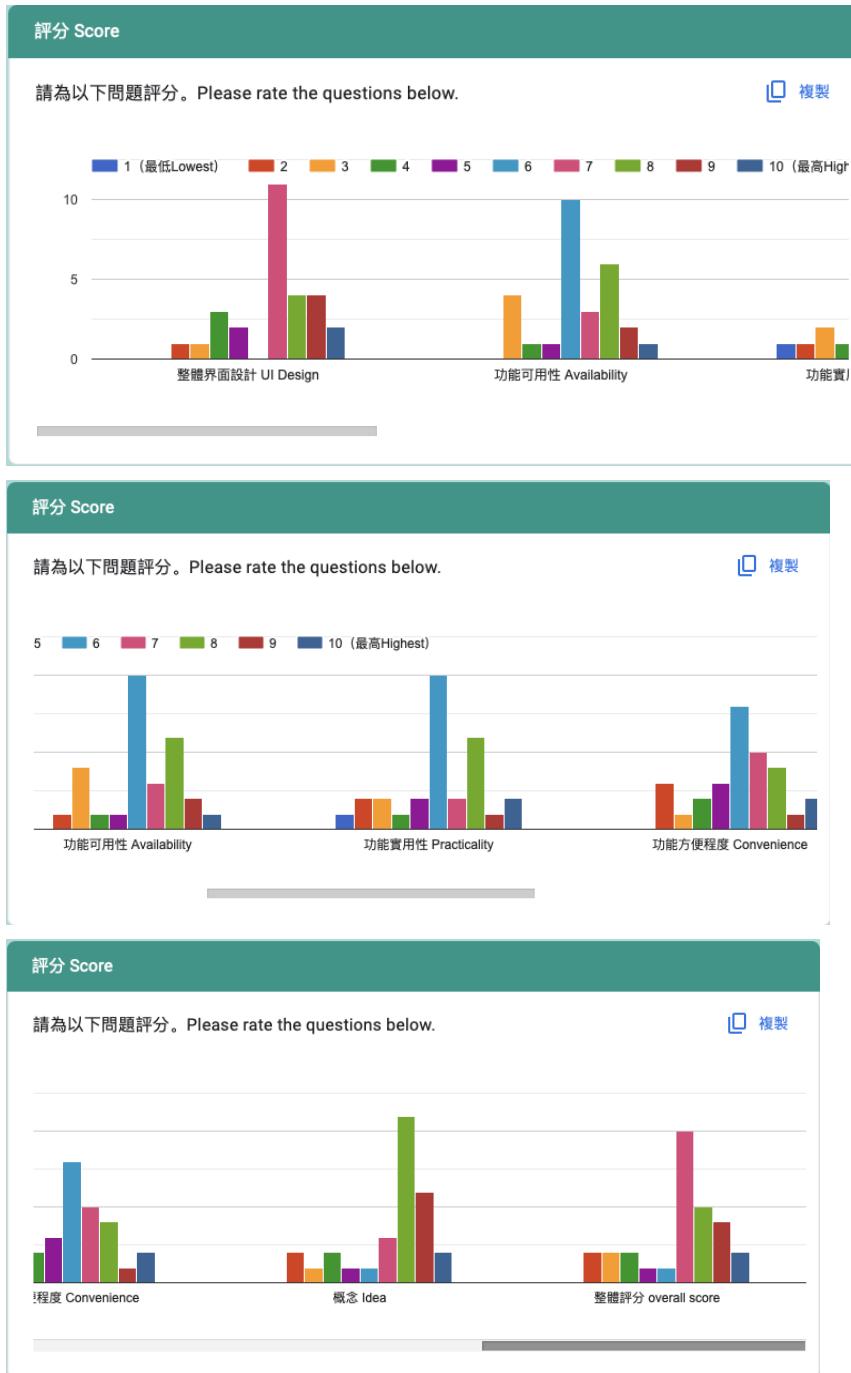


Figure 46. the score of the UI design

Figure 46 shows the score of overall UI design, most of the respondents give our UI design is qualified. We can conclude that our UI design is clean and neat enough, and it also can provide sufficient hints to the users.

For the functionality, availability, practicality and convenience are qualified in this survey. Over a half of respondents think our function can afford daily life, and it is useful in daily life. Most of the respondents think our concept or idea is great. That means our direction is correct.

All in all, our TexPic Calendar is qualified in their eyes. Their response shows that the calendar is personalized to the user.

Chapter 5. Optical Character Recognition Result

In this Chapter, we focus on the significant achievements of the Optical Character Recognition (OCR) technology, which is a critical component of our E-calendar application. OCR technology allows users to extract text from images, such as scanned documents and photographs, and convert it into a digital format that can be easily edited and managed.

Within this chapter, we will explore the various applications and benefits of OCR technology, including its ability to enhance efficiency, accuracy, and productivity in different aspects. We will also talk about the underlying principles and design elements of OCR , and provide evidence to support our claims.

Additionally, we will present the results of tests and experiments conducted on our OCR technology. By examining the significant achievements of OCR technology, we can gain a deeper understanding of its potential and limitations, and identify opportunities for further research and development.

```
194     Future<void> _scanImage() async {
195         if (_cameraController == null) return;
196
197         try {
198             final pictureFile = await _cameraController!.takePicture();
199             final file = File(pictureFile.path);
200             final inputImage = InputImage.fromFile(file);
201             final recognizedText = await _textRecognizer.processImage(inputImage);
202
203             String text = " ";
204
205             for (TextBlock block in recognizedText.blocks) {
206                 for (TextLine line in block.lines) {
207                     for (TextElement word in line.elements) {
208                         if(word.text == "<"){
209                             text = text + "<";
210                         } else if (word.text == ">"){
211                             text = text + ">";
212                         } else {
213                             text = text + word.text + " ";
214                         }
215                     }
216                 }
217             }
218
219             _stopCamera();
220             _textRecognizer.close();
221
222             print("Scan Image function call $text");
223
224             await navigator?.push(
225                 MaterialPageRoute(
226                     builder: (BuildContext context) => ResultScreen(text: text),
227                 ), // MaterialPageRoute
228             );
229         } catch (e) {
230             ScaffoldMessenger.of(context).showSnackBar(
231                 const SnackBar(
232                     content: Text('scanning text occur error'),
233                 ), // SnackBar
234             );
235         }
236     }
}
```

Figure 47. *ocr_screen.dart* source code part 1

```

232     Future<void> _pickImage() async {
233         final picker = ImagePicker();
234         final pickedFile = await picker.pickImage(source: ImageSource.gallery);
235         if (pickedFile == null) return;
236
237         try {
238             final file = File(pickedFile.path);
239             final inputImage = InputImage.fromFile(file);
240             final recognizedText = await _textRecognizer.processImage(inputImage);
241
242             String text = " ";
243
244             for (TextBlock block in recognizedText.blocks) {
245                 for (TextLine line in block.lines) {
246                     for (TextElement word in line.elements) {
247                         if(word.text == "<"){
248                             text = text + "<";
249                         } else if (word.text == ">"){
250                             text = text + ">";
251                         } else {
252                             text = text + word.text + " ";
253                         }
254                     }
255                 }
256             }
257
258             _stopCamera();
259             _textRecognizer.close();
260
261             print("Pick Image function call -----:$text");
262
263             await navigator?.push(
264                 MaterialPageRoute(
265                     builder: (BuildContext context) => ResultScreen(text: text),
266                 ), // MaterialPageRoute
267             );
268         } catch (e) {
269             ScaffoldMessenger.of(context).showSnackBar(
270                 const SnackBar(
271                     content: Text('scanning text occur error'),
272                 ), // SnackBar
273             );
274         }
275     }
276 }
```

Figure 48. *ocr_screen.dart* source code part 2

Figure 47 and 48 are the code of OCR text recognition functions. There are two methods for image input: camera or gallery import. The input image will pass to the `_textRecognizer.processImage()` function. That function is from the ML-Kit library and it can recognize the text data in the image. The result is stored in the `recognizedText` variable.

For example, we can select a photo from the gallery. According to the trained database of the machine, ml-kit will automatically recognize the image and convert them to digital.

Now the result page has appeared. It can recognize the the text inside the blank <> correctly(see Figure 50)

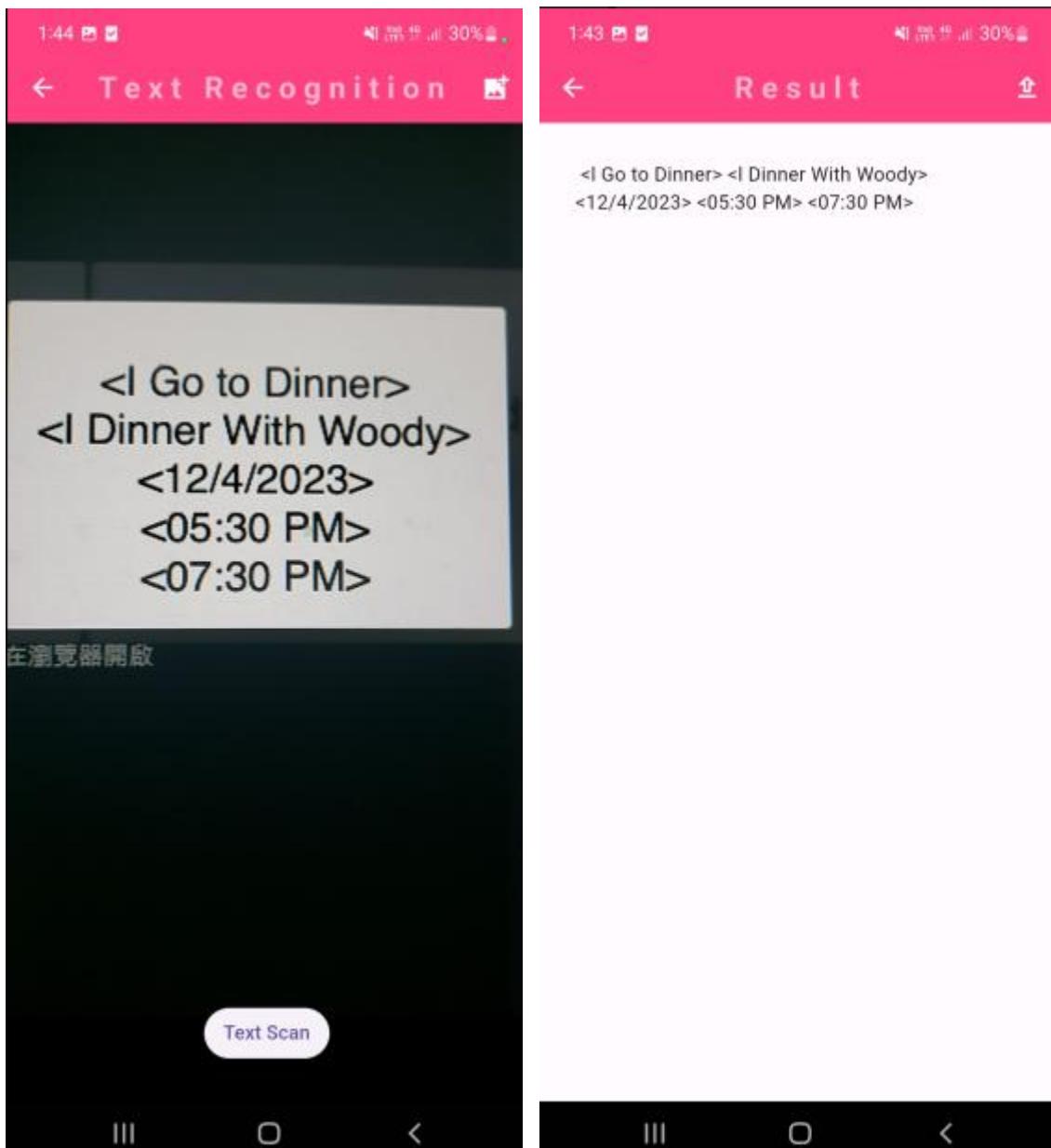


Figure 49 & 50: The text scan camera and the result

```
<I Go to Dinner> <I Dinner With Woody> <12/4/2023> <05:30 PM> <07:30 PM>
```

Figure 51: The text recognize result in console

For this scanning, the ml-kit machine can recognize the character from the photo, such as the alphabet, integer, punctuation mark. The recognition record shows on the console (See Figure 51). It means that the recognition format is fixed. The contents should be within the blank when user scanning. It is one of the limitations of our application text scanning.

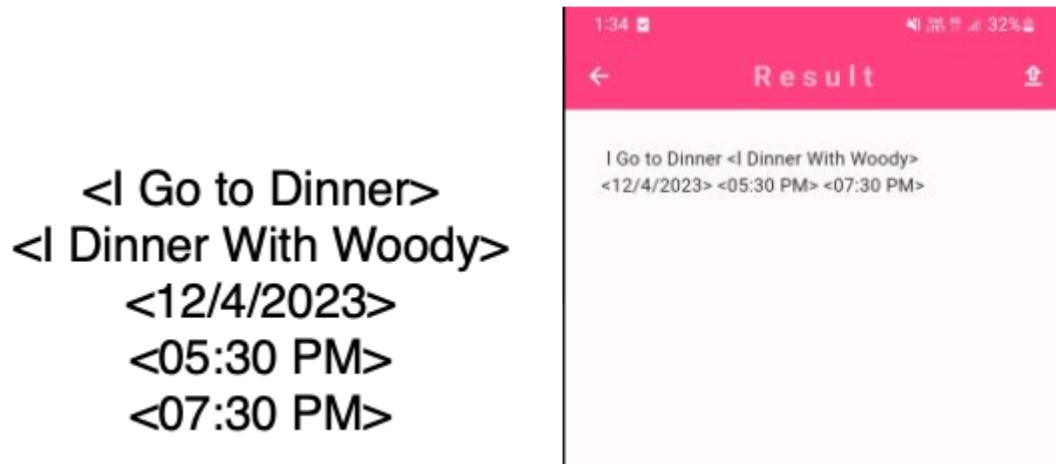


Figure 52 & 53: The testing text and the result without blank

Sometimes when users scan the text within the blank, there still appear some errors, such as cannot detect the blank “<>”(Figure 53). In this figure, all the fields contain “<>” but some of the results cannot read it like “I Go to Dinner”. The result is that without a blank screen the screen will become gray. It is because the condition of transferring the text is within the blank.

```
for (TextBlock block in recognizedText.blocks) {  
    for (TextLine line in block.lines) {  
        for (TextElement word in line.elements) {  
            if(word.text == "<"){  
                text = text + "<";  
            } else if (word.text == ">"){  
                text = text + ">";  
            } else {  
                text = text + word.text + " ";  
            }  
        }  
    }  
}
```

Figure 54: For loop algorithm for recognizing the text inside “<>”.

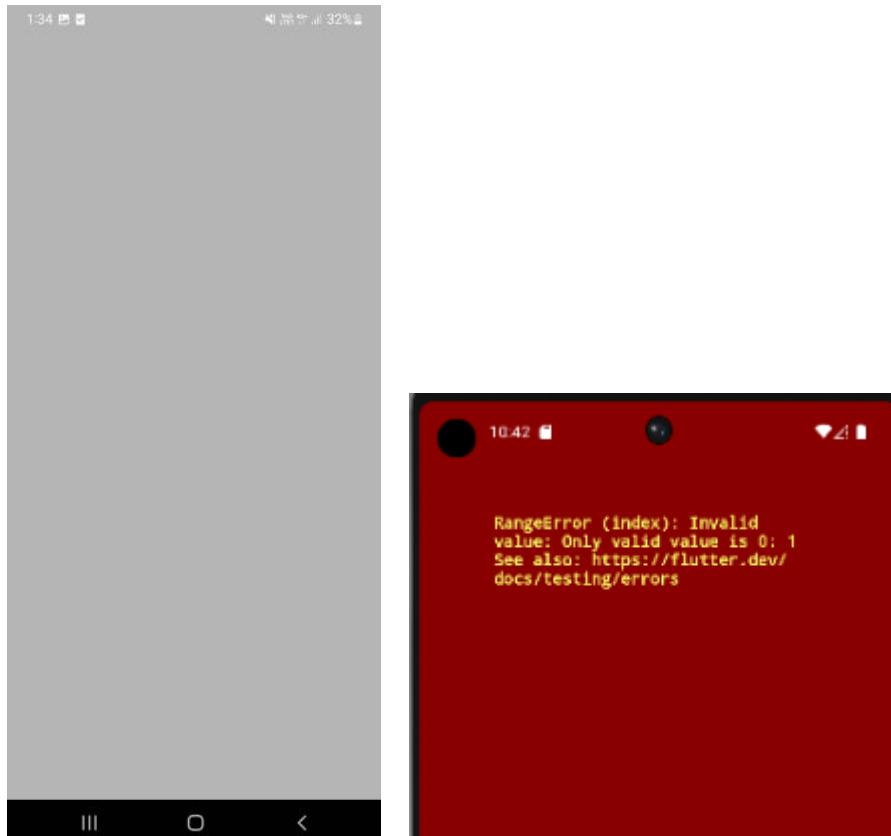


Figure 55 & 56: The error pages

There may be a problem that the ml-kit tool cannot recognize the image accurately. Sometimes the recognition has some deviation like cannot get the correct text, the missed required data fields and cannot read the blank.

Users may need to try to get the word with 1-2 times

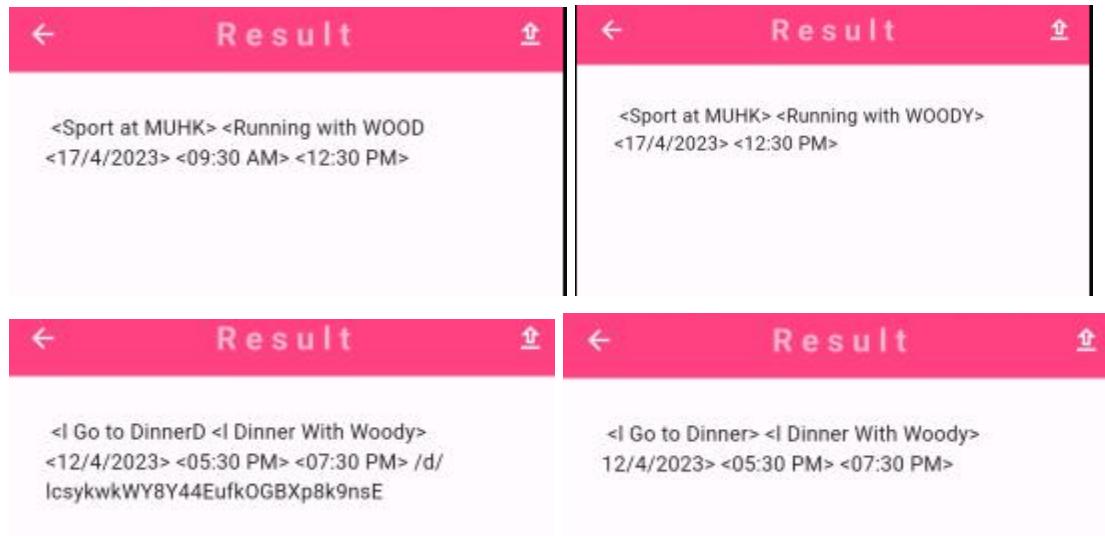


Figure 57,58,59 & 60: Four error types of recognition result

One of the reasons causing those problems is the image may contain unwanted background or noise which can affect the recognition process. Users need to preprocess the image by removing the unwanted background or noise.

Also, since the ml-kit is a machine learning model. There is too little training data in the database and we have to train it to increase the accuracy of text recognition.

Another problem is that the user should take the photo or select the picture with a vertical display method. It means that there are some bugs with the horizontal photo. Since the text recognition method is from the upper left corner to the lower right corner (see figure 62), when the photo is rotated 90 angle (See figure 61), it will scan the other fields first, of which the format priority should be the title first.

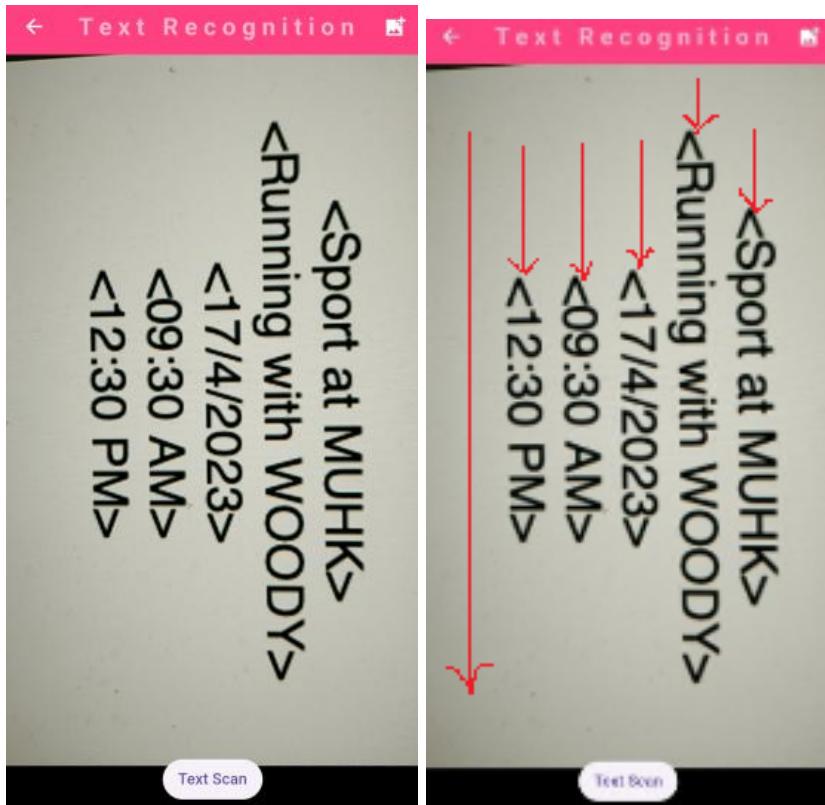


Figure 61 & 62: The rotate camera and ml-kit scanning method

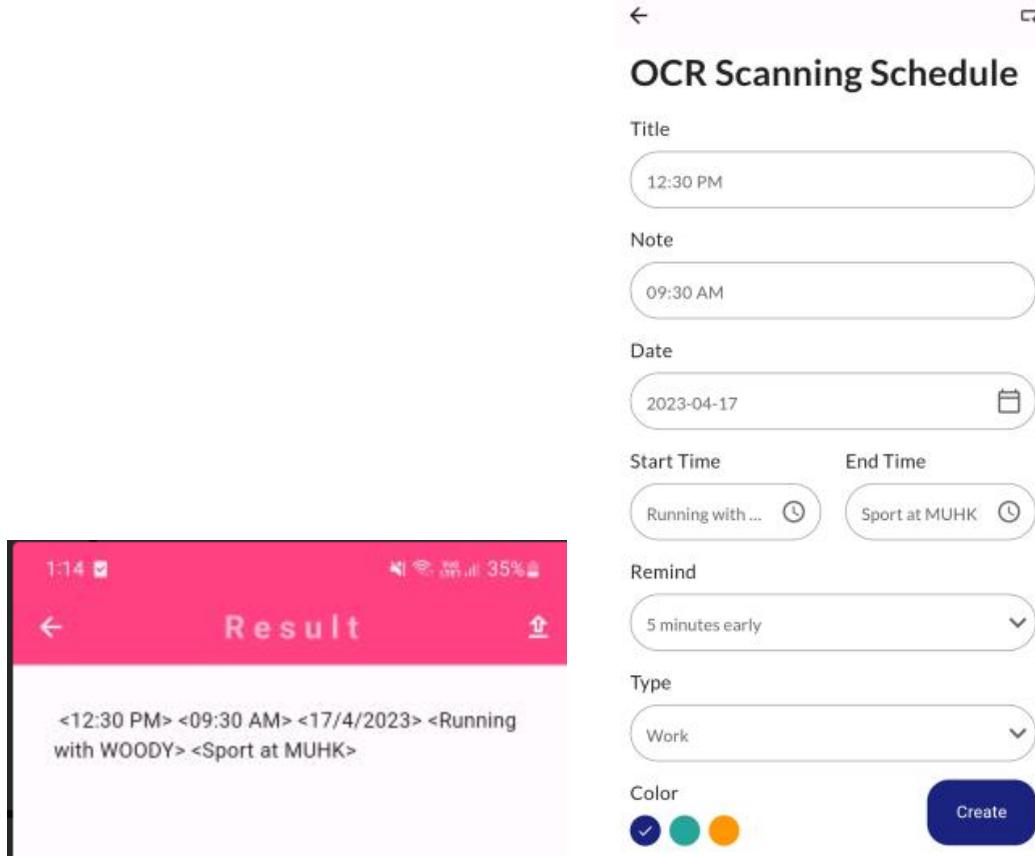


Figure 63 & 64: The rotated result and its input

In this testing, it rotates the camera to scan, which lets the phone become horizontal. The result becomes the time <12:30PM> in the first input and <09:30> in the second input. After passing the event to the add event page, all the input fields are incorrect because of the fixed format.

In conclusion, although the OCR text scanning technology of ml-kit is completed and executed in this project, there are some restrictions about using this technology such as fixed format, cannot rotate the phone or picture and need a clear background or noise photo to read the text.

Chapter 6. Auto-Schedule Result

In this chapter, we will discuss the prototype system design and implementation of Auto-Schedule. We will provide an overview of the system's design, as well as its major components and functionalities.

6.1 Overview of the Solution

Auto-Schedule is a function to assist users in efficiently managing their daily tasks by automatically generating personalized schedules based on predetermined criteria, resolving conflicts, and allocating tasks to available time slots. The main process includes:

1. Collecting tasks and their details from the user
2. Classifying tasks into different types
3. Setting event times based on user preferences
4. Resolving conflicts by allocating tasks to the next available 30-minute time slot
5. Allocating tasks to available time slots based on their type
6. Generating a personalized schedule for the user

The system's core approach involves several key steps, including event classification, time slot allocation, and conflict resolution. Auto-Schedule's design is centered around a user-friendly interface that allows users to input their tasks and preferences easily.

6.2 Design and Major Components

Auto-Schedule comprises several major components, including a task management module, a scheduling engine, a user interface module and data storage. The task management module is responsible for capturing and storing user tasks, while the scheduling engine generates schedules based on predefined criteria. The user interface module provides a user-friendly interface for inputting tasks and preferences and visualizing

schedules. The data storage component uses GetStorage to store task and event time data on the user's device.

6.3 Functionality

The Auto-Schedule function is designed to automatically generate a schedule based on the tasks and event times inputted by the user. The function uses an algorithm to allocate tasks to available time slots, avoiding overlaps and conflicts. If there are conflicts between events, the function will attempt to resolve them by allocating the task to the next available 30-minute time slot.

The function also allows users to adjust their schedule by dragging and dropping tasks to different event types or changing the duration of tasks.

6.4 Implementation

The Auto-Schedule function was implemented using the Flutter framework. The following tools and technologies were used during the implementation phase:

- Flutter: A mobile app development framework created by Google.
- Dart: The programming language used to develop the Auto-Schedule function.
- Get Storage: A lightweight key-value storage library used to store user settings and preferences.

The implementation of the Auto-Schedule function can be divided into two main parts: the user interface and the backend logic.

6.4.1 User Interface

The user interface (UI) of the Auto-Schedule function consists of two main pages: the task input page and the settings page.

The task input page is where users can input their tasks for the day. The page is designed to be user-friendly and easy to navigate. Users can enter the title , note ,type and duration For each task. Once all tasks have been entered, users can click on the "Auto Schedule" button to create their schedule. Shown in Figure 65.

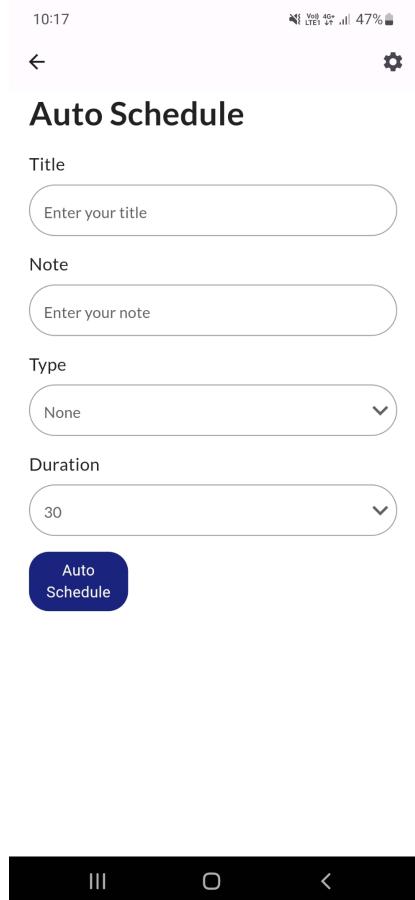


Figure 65: Auto schedule page

Let's talk about the setting page ,The settings page is where users can set the time for different types of events. Users can set the start and end times for breakfast, lunch, and dinner, as well as the start and end times for sport

time. The page allows users to customize their schedule to fit their individual needs and preferences. Shown in Figure 66.

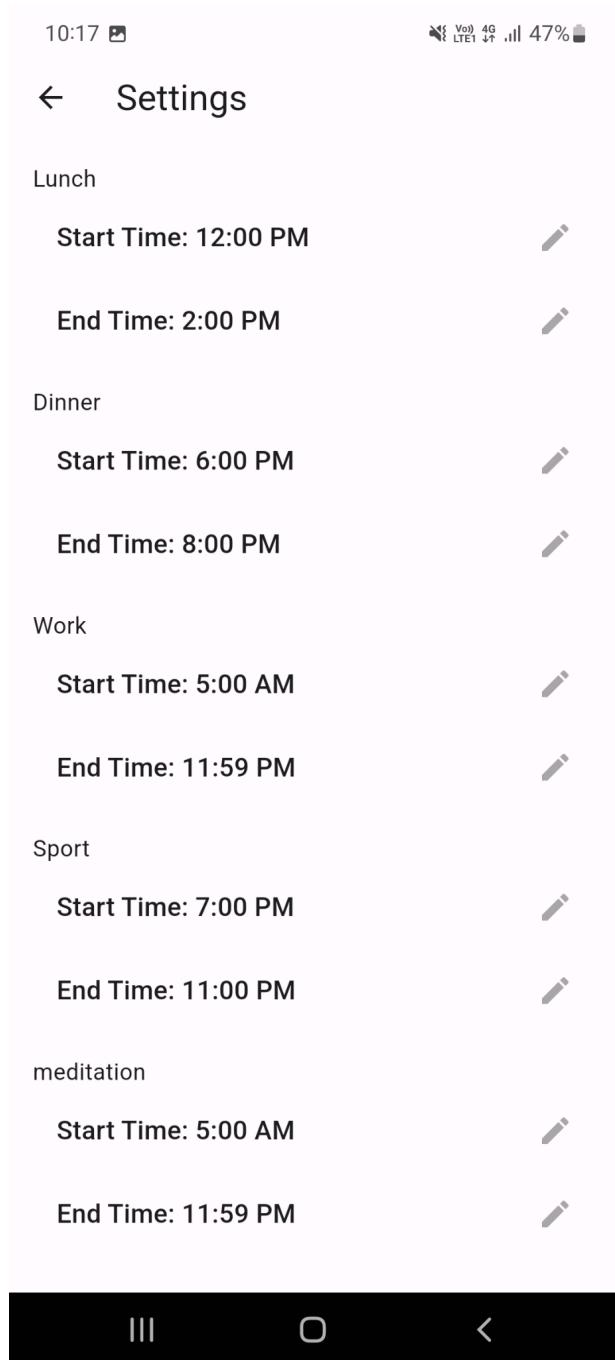


Figure 66: Setting page of Auto-Schedule

6.4.2 Backend Logic

The backend logic of the Auto-Schedule function was designed to base on their event type , duration , and to allocate them to available time slots in the user's schedule. The following steps were taken during the implementation phase:

1. .For each day, divide the available time period into 30-minute time chunks
2. Checks if the task's start and end times are within available time blocks
3. If yes, assign the task to that time block
4. If not, check the next available time block until an available timeblock is found
5. If no available time blocks are found within one day, assign the task to the next day
6. Returns an error message if no available timeblock is found within seven days

6.5 Testing

We try to conducted various tests to evaluate the effectiveness of the Auto-Schedule function in solving the problem of managing daily tasks efficiently

First, we tested the accuracy of the conflict resolution algorithm by deliberately creating conflicting events in the input and observing how the Auto-Schedule function handles them. Shown in Figure 67 & 68.

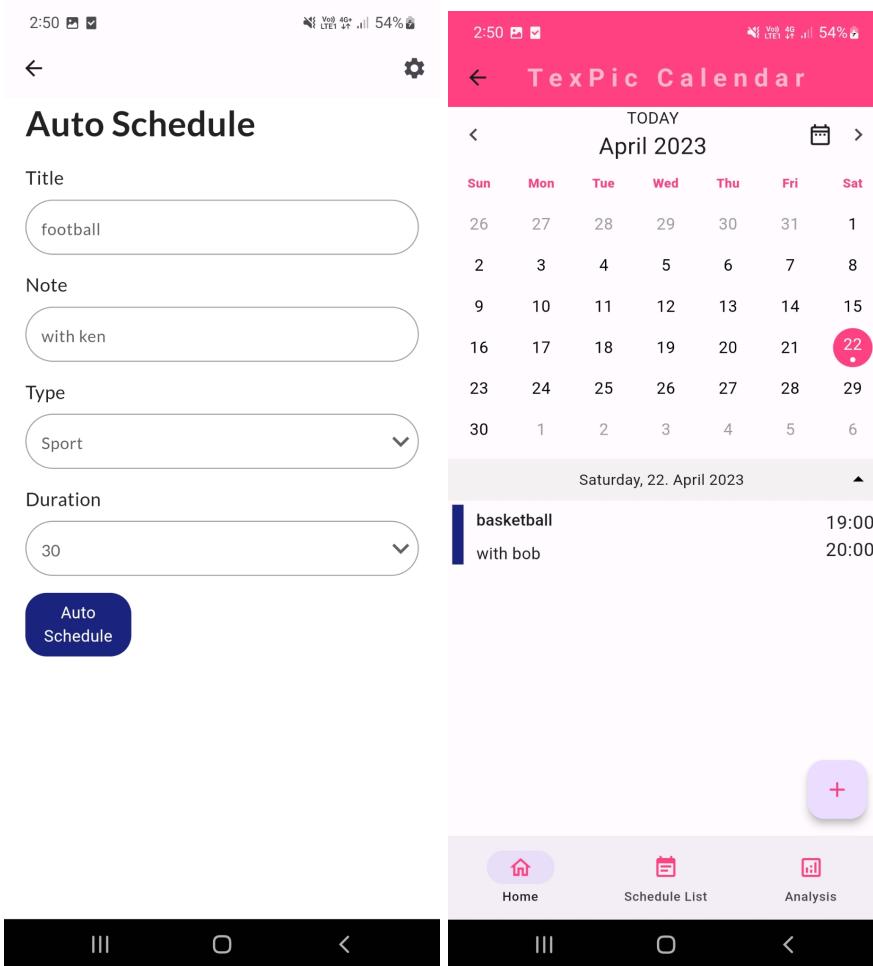


Figure 67 & 68: Auto-Scheduling Page and Calendar View

We found that the function was able to resolve conflicts by allocating the task to the next available time slot and that it was able to accurately identify the availability of time slots between the start and end times of the conflicting events. Shown in Figure 67 & 68.

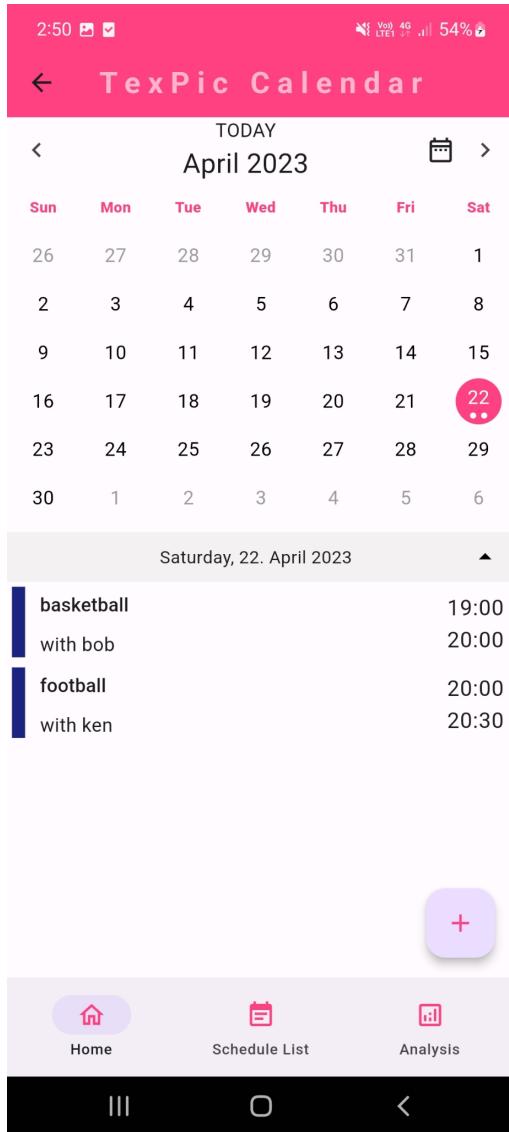


Figure 69: Calendar view that shows the event will not overlap

Next , we tested the efficiency of the Auto-Schedule function by comparing the time taken to manually schedule tasks versus using the Auto-Schedule function. The video can be found in the appendix of this report.The results showed that the Auto-Schedule function only took 19 seconds to complete the task, while manual scheduling took 39 seconds.This indicates that Auto-Schedule is more efficient and time-saving compared to manual scheduling.

One reason for the longer time taken during manual scheduling is the need to check the schedule first to identify available time slots. Additionally, manual scheduling requires inputting more information compared to the Auto-Schedule function which only requires 4 inputs. Therefore, the time taken for manual scheduling is longer compared to using the Auto-Schedule function.

6.6 Evaluation

Our survey interviewed 29 people about their opinions on Auto-schedule function. Based on the survey results, the majority of respondents rated the convenience of the Auto-Schedule function in daily life as a 4 out of 5. However, some respondents gave a lower rating of 2 or 3. In terms of usefulness, the majority of respondents rated the function as a 4 out of 5, with a few giving a lower rating of 2 or 3. As for the likelihood of using the function, the majority of respondents rated it as a 3 out of 5 or higher, with the highest rating being 4 out of 5. Overall, the survey results suggest that the Auto-Schedule function is generally perceived as useful and convenient, but there is room for improvement to better meet the needs and preferences of all users. The result shown in Figure 70.

請評分以下問題。Please rate the following questions.

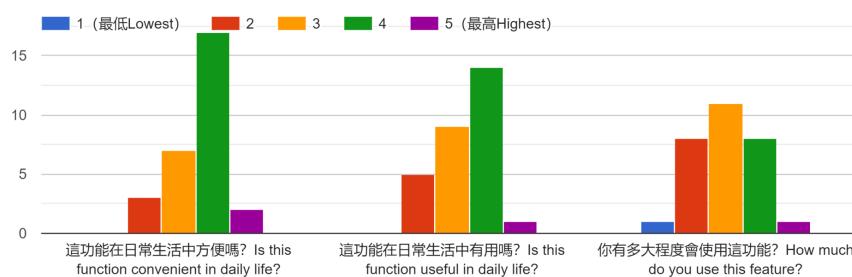


Figure 70: result of survey

Chapter 7. Conclusion

7.1 Satisfaction of the aim and objectives

To conclude, we have designed a personalized calendar which can “CRUD”, Auto-Schedule and Text-scanning successfully. Those functions can provide a certain degree of convenience, such as easy to manage the event, clear UI which is easy to use, reduce the confusion to think about the event duration and digitalization.

7.2 Achievements Summary

In the final product of TexPic Calendar. We build the main functions that we mention in the proposal: OCR text scanning and auto-scheduling. We built an android mobile application successfully and it has a good UI design and layout, it also contains the calendar basic functions like task management and data analysis. We use ML Kit API for OCR function and the data process so that the data can fill into the input field automatically. And we build an auto-scheduling logic function, which can search the database data time slot and allocate the suitable time slot to the user. Data analysis can show the user their own information in daily life, it can help the user to review their daily life exhibition and record. According to those functions, we think that the TexPic Calendar application is more approaching the topic - personalization.

7.3 Limitation of the TexPic Calendar

TexPic Calendar has enough functions for calendar missions, but still has some limitations in the calendar functions.

In OCR text scanning function. We use the ML Kit Text Recognition tool for the OCR function. It has a high accuracy for recognizing the computer font text. That means the input image text must be regular, neat and consistent in size. Some handwritten text may not be recognized and misleading the output result. For the data format, it is another limitation of the OCR. We want the data to fill into the input field automatically. The input data format must be separated by correct data. That is the limitation in OCR text scanning function.

In auto-scheduling function. We will search and compare the time slot in the database and allocate the available time slot to the user. But users need to set the preference time slot by event type. And the user can not create a new event type. Moreover, many of the time segments assigned to events are not humanized. Therefore, this function may not be practical in daily life.

7.4 Solution of the limitations

<Event title>
<Note>
<dd/mm/yyyy>
<Start Time AM/PM>
<End Time AM/PM>

Figure 71: Field Format

In OCR text scanning function. Handwritten and data format are the limitations of that function. We design a format paper for the user. Users need to follow the format of the table so that helps our data processing. For the text font, we provided the symbol in the table so that it would avoid the error of symbol input.

In auto-scheduling function. We have pre-designed common event types and time slots for users to choose, and users can customize different types of time slots in the settings page.

7.5 Suggestions of future work

In the future, we hope that we can improve the accuracy of OCR handwritten features and add the natural language processing into the calendar program or using API. We hope that can help the user fill into the input field without formatting the table. But it is a big challenge of the technical problem. Natural language processing needs a complex model and algorithm to implement. It needs a lot of data to train the model and test the model. We can use the pre-trained model for natural language processing, but we still need a lot of data to train the model and make it fit our mission.

Then, we want an auto-scheduling function that can apply an AI model so that the function can predict and allocate the time slot for the user like a human. It is also complex work and needs a lot of knowledge in the AI aspect. Beside that, we may need to redesign the logic flow of that function and test to find out the limitations of it.

In conclusion, we hope our function can be more flexible and diversified. We want TexPic Calendar to be more personalized and humanized. It can really help the user in their daily life.

References

- Celayix. (2023, January 3). *Using artificial intelligence as a scheduling tool*. Celayix. Retrieved January 29, 2023, from <https://www.celayix.com/blog/using-artificial-intelligence-as-a-scheduling-tool/>
- Copeland, M. (2021, July 17). *The difference between AI, Machine Learning, and deep learning?* NVIDIA Blog. Retrieved January 29, 2023, from [https://blogs.nvidia.com/blog/2016/07/29/whats-difference-artificial-intelligence-machine-learning-deep-lea
rning-ai](https://blogs.nvidia.com/blog/2016/07/29/whats-difference-artificial-intelligence-machine-learning-deep-learning-ai)
- Christine M. Kincaid, Pierre B. Dupont, and A. R. Kaye. 1985. Electronic calendars in the office: an assessment of user needs and current technology. ACM Trans. Inf. Syst. 3, 1 (Jan. 1985), 89–102. <https://doi.org/10.1145/3864.3868>
- J. Memon, M. Sami, R. A. Khan and M. Uddin, "Handwritten Optical Character Recognition (OCR): A Comprehensive Systematic Literature Review (SLR)," in IEEE Access, vol. 8, pp. 142642-142668, 2020, doi: 10.1109/ACCESS.2020.3012542.
- Karthick, K., Ravindrakumar, K. B., Francis, R., & Ilankannan, S. (2019). *Steps Involved in Text Recognition and Recent Research in OCR: A Study*. International Journal of Recent Technology and Engineering (IJRTE), 8(1), 3095-3099. Retrieved from <https://www.ijrte.org/wp-content/uploads/papers/v8i1s/F05811981S19.pdf>
- Li, X., Wang, C., & Shen, C. (2021). TrOCR: *Transformer-based Optical Character Recognition with Pre-trained Models*. arXiv. <https://arxiv.org/abs/2105.14757>
- Michael Copeland, "The Difference between AI, Machine Learning, and Deep Learning?," NVIDIA Blog, July 17, 2021,
- R. Smith. "An Overview of the Tesseract OCR Engine." *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*. 2007. pp 629-633.
- Ravina Mithe, Supriya Indalkar, Nilam Divekar. 2013. Optical Character Recognition. International Journal of Recent Technology and Engineering (IJRTE).
- Stone, D. L. (2005). *User interface design and evaluation*. Morgan Kaufmann.
- Thi Tuyet Hai Nguyen, Adam Jatowt, Mickael Coustaty, and Antoine Doucet. 2021. Survey of Post-OCR Processing Approaches. ACM Comput. Surv. 54, 6, Article 124 (July 2022), 37 pages. <https://doi.org/10.1145/3453476>
- Uturnsoftware. (2018, November 9). *Uturnsoftware*. YouTube. Retrieved January 29, 2023, from <https://www.youtube.com/@UturnSoftware>
- Walker, S. D., Walker, R. C., & Brooks, K. (2005). *Visual-syntactic text formatting: A new method to enhance online reading*. Journal of Educational Psychology, 97(3), 440-450. <https://doi.org/10.1037/0022-0663.97.3.440>

Todoist. (n.d.). Eisenhower Matrix: A guide to prioritizing tasks. Retrieved March 30, 2023, from <https://todoist.com/zh-TW/productivity-methods/eisenhower-matrix>

McKay, B. & K. (2021, September 26). *The Eisenhower Decision Matrix: How to distinguish between urgent and important tasks and make real progress in your life*. The Art of Manliness. Retrieved April 22, 2023, from <https://www.artofmanliness.com/character/knowledge-of-men/eisenhower-decision-matrix/>

Trello. (n.d.). Retrieved April 22, 2023, from <https://trello.com/power-ups/5e4cd4d5b5c35b5a5d8c5e60/butler>.

Appendix A - GROUP INFORMATION.

Name & SID	Team Role	Work %
Cheung Ho Yan 12754355	Research Helper, Auto-schedule developer, poster, video editing.	15
Li Heng 12895168	Leader, Final Checker, Android Application Developer, OCR Developer	30
Chan Wei Hei 12896641	Programmer, Research Helper, Auto-Schedule Developer	20
Wong Kong Lun 12728784	Programmer, Source Controller, Calendar developer	35

Table 2: Member's Role, Responsibilities and Work Percentage

Appendix B1 Team Member #1 Report

WONG Kong Lun (12728784)

[Individual Submit]

Declaration statement

I, Wong Kong Lun(12728784), certify that the work described in this document is original and done by me and I have utilized guidance of our supervisor in completing this project, and that the content which is not our own has been attributed and referenced properly. There should be no copyrighted content without permission to use. There should be no confidential data. I also declare that the contributions listed in Self Appraisal of Contribution are truly and correctly attributed to me.

A handwritten signature in orange ink, appearing to read "Wong Kong Lun".

22/4/2023

Appendix B2 Team Member #2 Report

Li Heng (12895168)

[Individual Submit]

Declaration statement

I, Li Heng (12895168), certify that the work described in this document is original and done by me and I have utilized guidance of our supervisor in completing this project, and that the content which is not our own has been attributed and referenced properly. There should be no copyrighted content without permission to use. There should be no confidential data. I also declare that the contributions listed in Self Appraisal of Contribution are truly and correctly attributed to me.

A handwritten signature in black ink, appearing to read "Li Heng".

21/4/2023

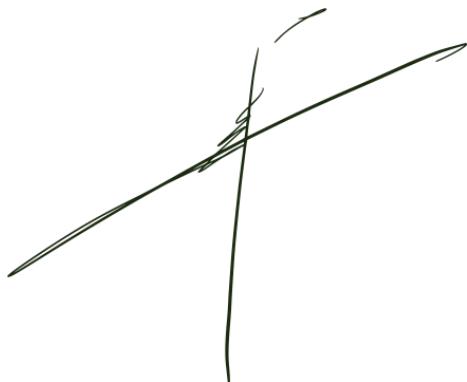
Appendix B3 Team Member #3 Report

Cheung Ho Yan (12754335)

[Individual Submit]

Declaration statement

I, Cheung Ho Yan(12754335), certify that the work described in this document is original and done by me and I have utilized guidance of our supervisor in completing this project, and that the content which is not our own has been attributed and referenced properly. There should be no copyrighted content without permission to use. There should be no confidential data. I also declare that the contributions listed in Self Appraisal of Contribution are truly and correctly attributed to me.

A handwritten signature in black ink, appearing to read "Cheung Ho Yan".

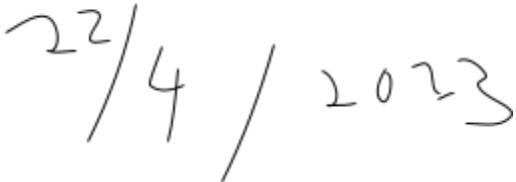
22/4/2023

Appendix B4 Team Member #4 Report

[Individual Submit]

Declaration statement

I, Chan Wai Hei (12896614), certify that the work is original and I have utilized guidance of my supervisor in completing this project, and that the content which is not our own has been attributed and referenced properly. There should be no copyrighted content without permission to use. There should be no confidential data.

Appendix C - Survey Question

第 1 個區段，共 8 個

TexPic Calendar Evaluation Survey



你好，我們是香港都會大學Computer Science Year 4的學生，此問卷是希望閣下可以填寫和提供對於我們Final Year Project: TexPic Calendar的看法，由此我們可以從大眾得知TexPic Calendar的不足之處及如何改善，感謝你的參與！

Hello, we are students of Computer Science Year 4 at Hong Kong Metropolitan University. This questionnaire is for you to fill in and provide your opinions on our Final Year Project: TexPic Calendar, so that we can learn from the public about the shortcomings of TexPic Calendar and How to improve, thank you for your participation!

請問你的性別? What is your gender? *

- 男 Male
- 女 Female

請問你的年齡組別? What is your age group? *

- 0 - 10
- 11 - 18
- 19 - 25
- 26 - 35
- 35或以上 / 35 or above

請問你的工作種類是? What is your job type? *

- 學生 / Student
- 教育 / Education
- 行政人員 / Administrator
- 科技 / Technology
- 工程 / Engineering
- 設計 / Design
- 金融 / Financial services
- 其他 / Others

第 2 個區段，共 8 個

電子日曆使用程度統計 Electronic calendar usage statistics



TexPic Calendar介紹： TexPic Calendar是一個以個人化為目標的電子日曆，當中包含一些基本日曆應有的功能，例如：檢視日期、建立行程事件、修改行程事件、事件提醒等。另外亦有特色功能，例如：光學字元辨識輸入行程、自動編配行程及行程數據分析等。

Introduction to TexPic Calendar: TexPic Calendar is a personalised E-calendar, which includes some functions that a basic calendar should have, such as: viewing dates, creating events, modifying events, event reminders, etc. In addition, it also has special functions, such as: optical character recognition input itinerary, automatic arrangement of itinerary and itinerary data analysis, etc.

請問你日常生活中有使用電子日曆嗎? Do you use electronic calendars in your daily life? *

- 有 Yes
- 沒有 No

第 2 區段後 前往下一個區段



第 3 個區段，共 8 個

區段標題 (選填)



描述 (選填)

如有，請問使用哪款日曆？ If so, which calendar do you use?

- Apple Calendar
- Google Calendar
- ClickUp
- Microsoft Outlook Calendar
- Calendly
- SavvyCal
- Others

第 3 區段後 前往區段 4 (界面設計 UI Design)

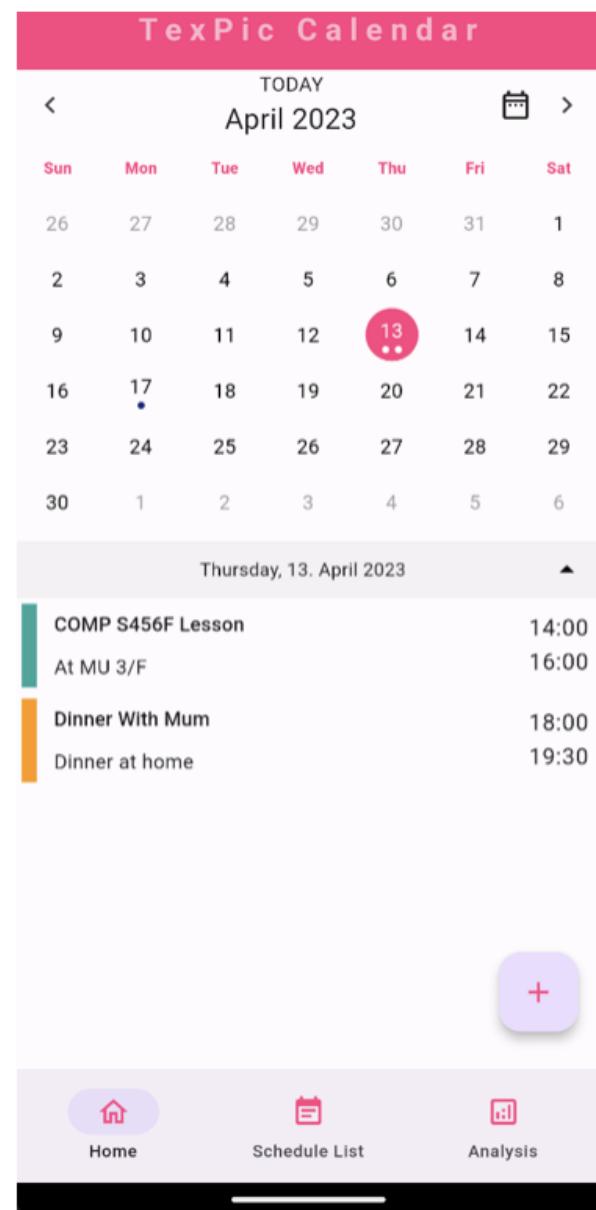


第4個區段，共8個

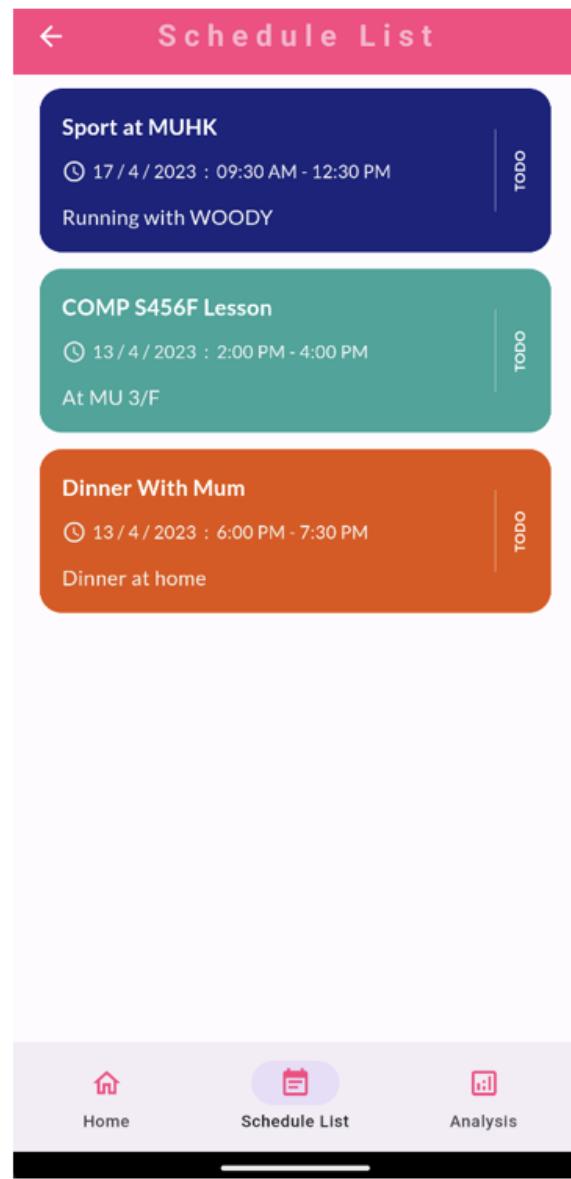
界面設計 UI Design

描述(選填)

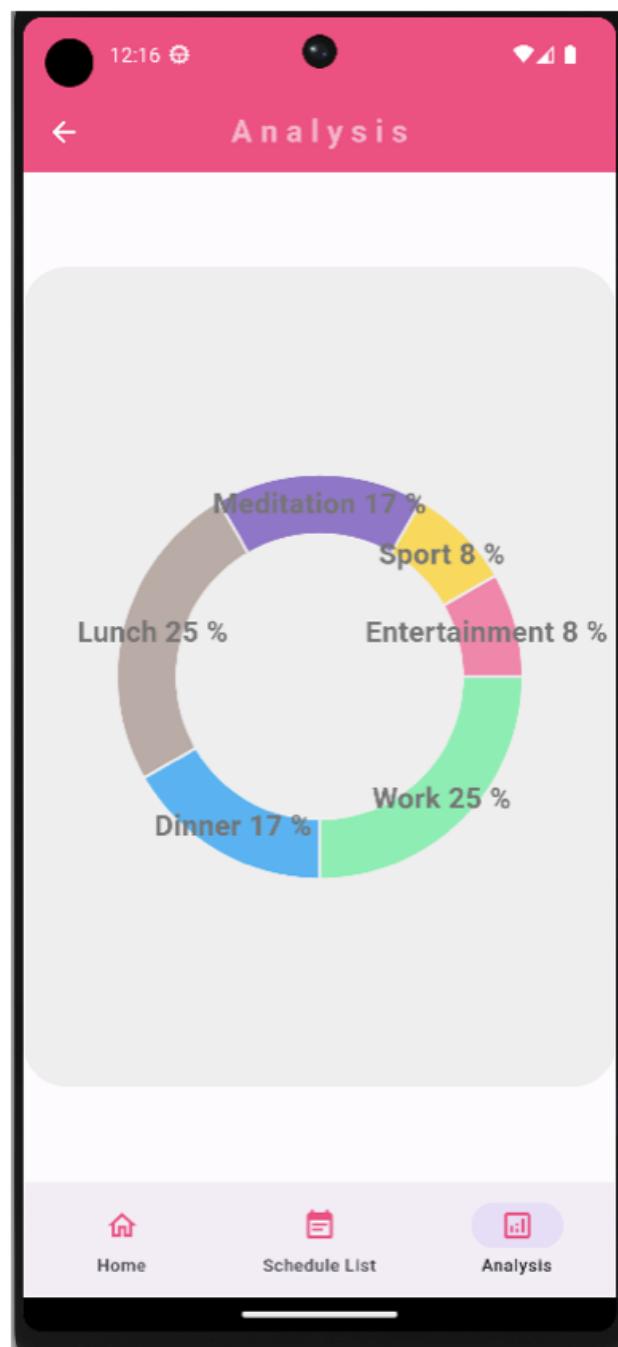
Home Page UI Design



Task Management UI Design



Data Analysis Page UI Design



你認為以上的界面能清楚顯示日曆嗎？ Do you think the above interface can display the calendar clearly? *

能 Yes

不能 No

你認為以上的界面設計分類清晰嗎？ Do you think the above interface design categories are clear? *

清晰 Clear

不清晰 Not clear

你認為以上的界面設計能讓用戶找到所需資訊或功能嗎？ Do you think the above interface design allows users to find the information or functions they need? *

能 Yes

不能 No

部分可以 Several

你認為以下的界面設計有足夠的提示和簡潔嗎？請評分（1為最低；5為最高）

*

Do you think the following interface design has enough hints and simplicity? Please rate (1 is lowest; 5 is highest)



Add Schedule

Title

Enter your title

Note

Enter your note

Date

2023-04-13



Start Time

11:04 PM



End Time

10:00 PM



Remind

5 minutes early



Type

Work



Color



Create



1 (最低Lowest) 2 3 4 5 (最高Highest)

提示足夠程度 ...

簡潔程度 Concise程度

你覺得以下的行程顯示有足夠的資訊嗎? Do you think the following itinerary shows enough information? *

Sport at MUHK

⌚ 17 / 4 / 2023 : 09:30 AM - 12:30 PM

Running with WOODY

TODO

COMP S456F Lesson

⌚ 13 / 4 / 2023 : 2:00 PM - 4:00 PM

At MU 3/F

TODO

Dinner With Mum

⌚ 13 / 4 / 2023 : 6:00 PM - 7:30 PM

Dinner at home

TODO

1

2

3

4

5

不足夠 Not enough

足夠 Enough

第 4 區段後 前往區段 5 (電子日曆功能調查 Electronic calendar function survey) ▼

第 5 個區段，共 8 個

電子日曆功能調查 Electronic calendar function survey



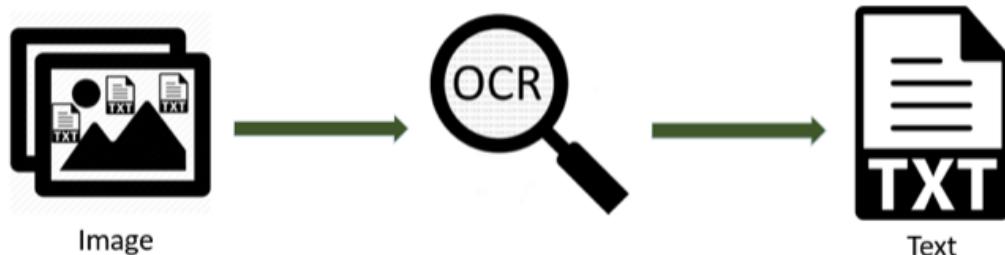
以下將會就本日曆的功能作訪問

The following will access the functions of this calendar

光學字元辨識技術是指對文字資料的圖像檔案進行分析辨識處理，取得文字及版面資訊的過程。因此用戶可以手寫資料再進行輸入程序。

Optical character recognition (OCR) technology refers to the process of analysing and identifying image files of text data to obtain text and layout information. Therefore, the user can write the data by hand and then enter the program.

OCR Technique

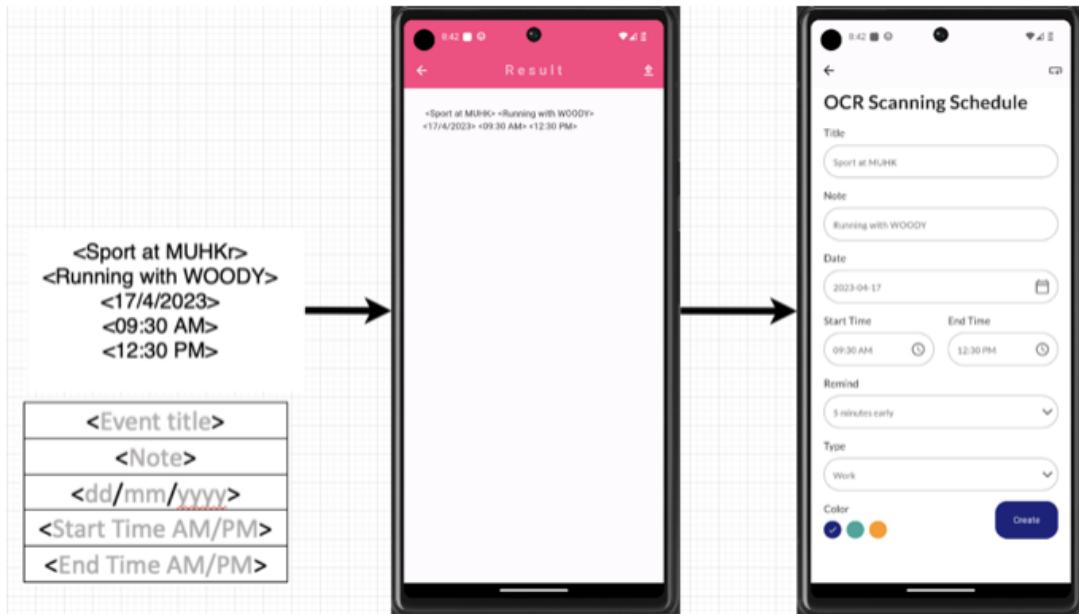


請問你認識或曾經聽過光學字元辨識技術嗎？ Do you know or have heard of optical character * recognition (OCR) technology?

是 Yes

否 No

你認為使用光學字元辨識技術作為輸入行程的方式有趣和有用嗎? Do you think it's interesting * and useful to use OCR as a way to input itineraries?



1 (最低Lowest) 2 3 4 5 (最高Highest)

有趣? Interesting?

有用? Useful?

輸入資料必須填在指定空格內以內，你認為這方便嗎? *

The input information must be filled in the designated space. Do you think this is convenient?

1 2 3 4 5

不方便 Inconvenient 方便 Convenient

你有多大程度用使用這個功能? How much do you use this feature? *

1 2 3 4 5

不會使用 Definitely not 一定會使用 Definitely

第6個區段，共8個

電子日曆功能調查2 Electronic calendar function survey 2



以下將會就本日曆的功能作訪問

The following will access the functions of this calendar

自動編配行程是指用戶只需要輸入標題、內容和事件種類，日曆便會為你在空餘的時間自動安排該事件，而用戶亦可以自定義每個事件種類的時間段。

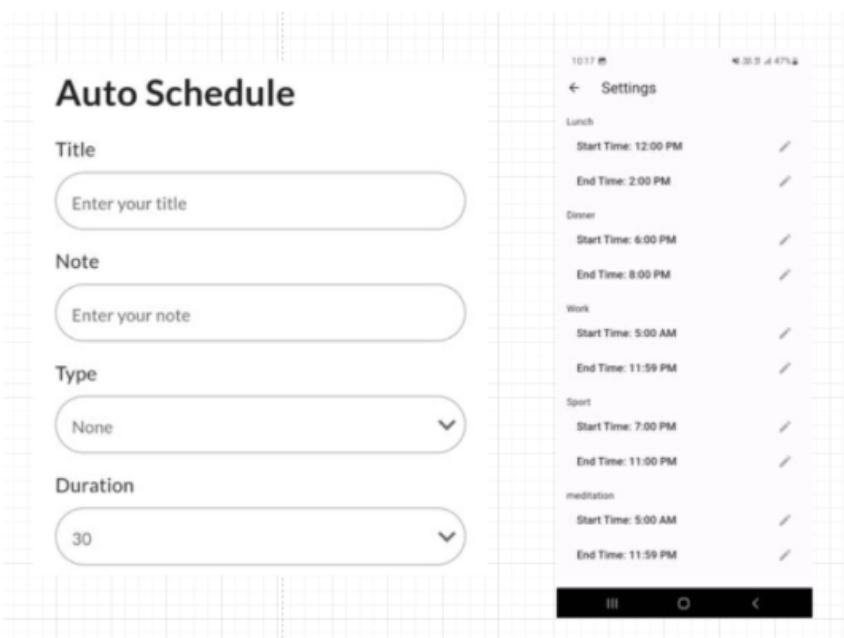
Auto-scheduling means that the user only needs to enter the title, note and event type, and the calendar will automatically arrange the event for you in your free time, and the user can also customise the time period for each event type.

請問你曾經聽過或使用過會自動安排行程的日曆嗎？Have you ever heard of or used a calendar * that automatically arranges your schedule?

有 Yes

沒有 No

你認為該功能使用方法方便嗎? Do you find this feature convenient to use? *



1 2 3 4 5

不方便 Inconvenient



方便 Convenient

請評分以下問題。Please rate the following questions. *

1 (最低Lowest...) 2 3 4 5 (最高Highest...)

這功能在日常...



這功能在日常...



你有多大程度...



第 6 區段後 前往下一個區段

第 7 個區段，共 8 個

電子日曆功能調查3 Electronic calendar function survey 3



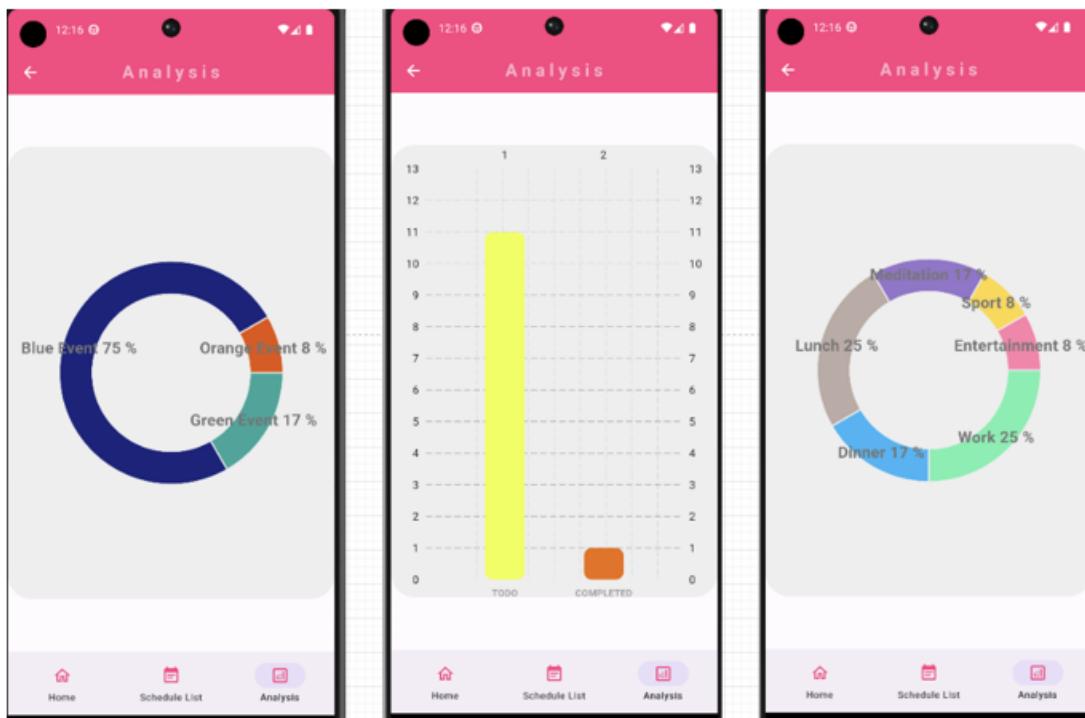
以下將會就本日曆的功能作訪問

The following will access the functions of this calendar

數據統計是根據用戶所建立的行程，透過分析而顯示的結果，它可以反映用戶的日常數據用行程數量等。

Data statistics are the results displayed through analysis based on the itinerary created by the user. It can reflect the user's daily data and the number of trips.

你認為以下數據圖所反映的結果清晰嗎？Do you think the results reflected in the data graph below are clear? *



1

2

3

4

5

不清晰 Not clear



清晰 Clear

你認為此功能能夠反映用戶的日常數據嗎？ Do you think this feature can reflect the user's daily * data?

1	2	3	4	5	
不能 Can not	<input type="radio"/> 能 Can				

你會經常檢視該數據嗎？ How often do you review this data? *

1	2	3	4	5	6	7	8	9	10
不會 Never	<input type="radio"/> 總是 Always								

你認為該功能實用嗎？ Do you think this feature is useful? *

1	2	3	4	5	
不實用 Not practical	<input type="radio"/> 實用 Practical				

第 7 區段後 前往區段 8 (評分 Score)

第 8 個區段，共 8 個

評分 Score

▼ □

請為我們的 TexPic Calendar 評分

Please rate our TexPic Calendar

請為以下問題評分。Please rate the questions below.*

1 (最... 2 3 4 5 6 7 8 9 10 (...)

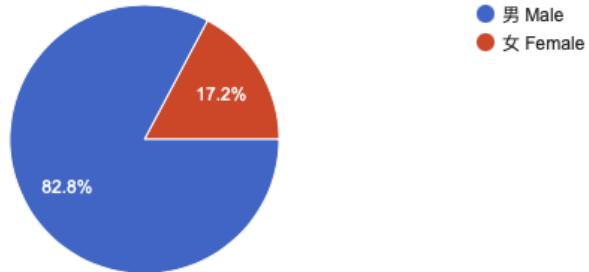
整體界...	<input type="radio"/>									
功能可...	<input type="radio"/>									
功能實...	<input type="radio"/>									
功能方...	<input type="radio"/>									
概念 I...	<input type="radio"/>									
整體評...	<input type="radio"/>									

Appendix D - Survey Result

請問你的性別? What is your gender?

 複製

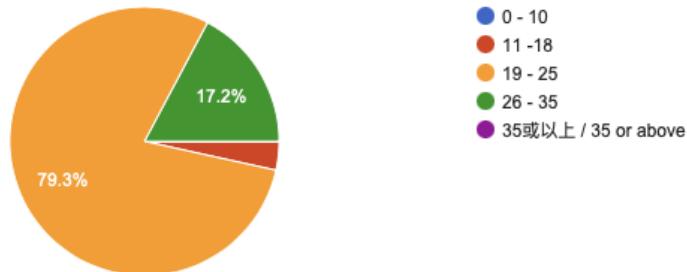
29 則回應



請問你的年齡組別? What is your age group?

 複製

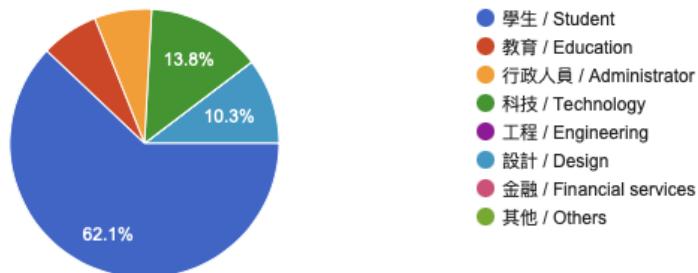
29 則回應



請問你的工作種類是? What is your job type?

 複製

29 則回應

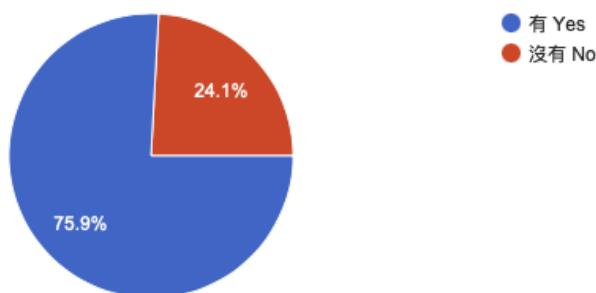


電子日曆使用程度統計 Electronic calendar usage statistics

請問你日常生活中有使用電子日曆嗎? Do you use electronic calendars in your daily life?

 複製

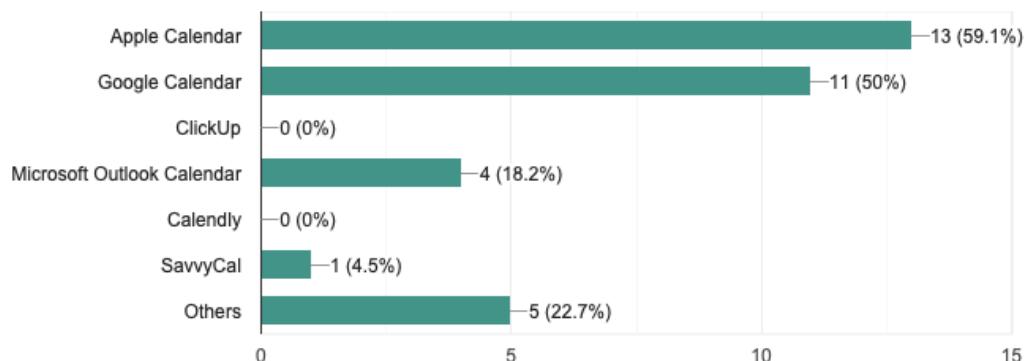
29 則回應



如有，請問使用哪款日曆？ If so, which calendar do you use?

 複製

22 則回應

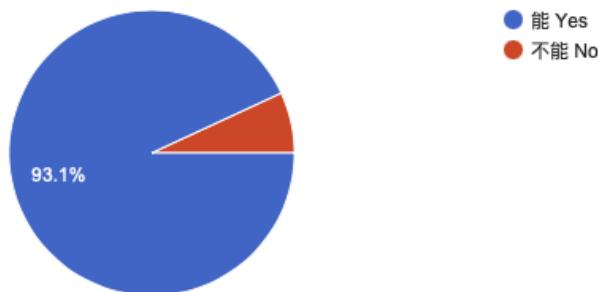


界面設計 UI Design

你認為以上的界面能清楚顯示日曆嗎？ Do you think the above interface can display the calendar clearly?

29 則回應

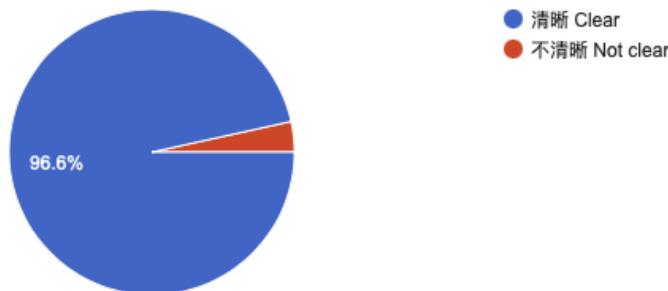
複製



你認為以上的界面設計分類清晰嗎？ Do you think the above interface design categories are clear?

29 則回應

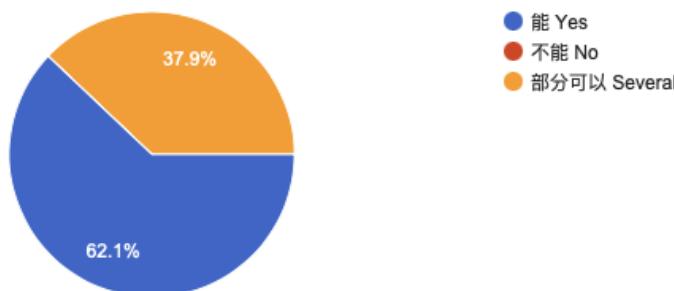
複製



你認為以上的界面設計能讓用戶找到所需資訊或功能嗎？ Do you think the above interface design allows users to find the information or functions they need?

29 則回應

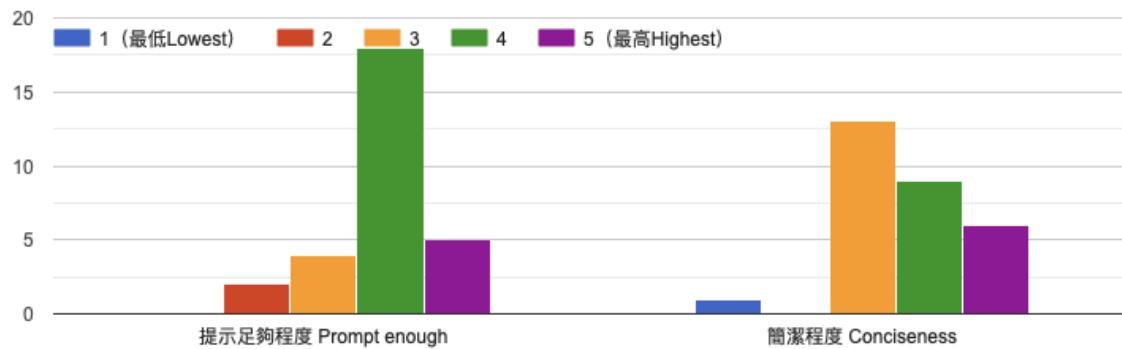
複製



你認為以下的界面設計有足夠的提示和簡潔嗎？請評分（1為最低；5為最高）

複
製

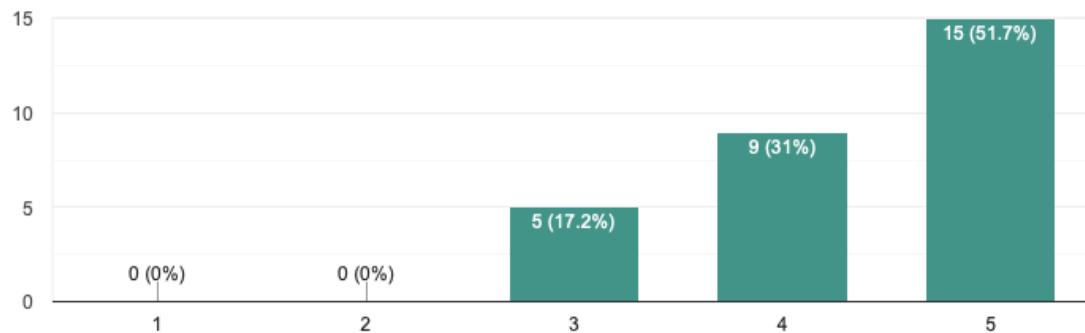
Do you think the following interface design has enough hints and simplicity? Please rate (1 is lowest; 5 is highest)



你覺得以下的行程顯示有足夠的資訊嗎？Do you think the following itinerary shows enough information?

複
製

29 則回應

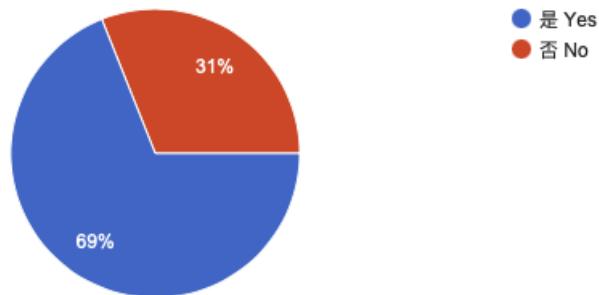


電子日曆功能調查 Electronic calendar function survey

請問你認識或曾經聽過光學字元辨識技術嗎? Do you know or have heard of optical character recognition (OCR) technology?

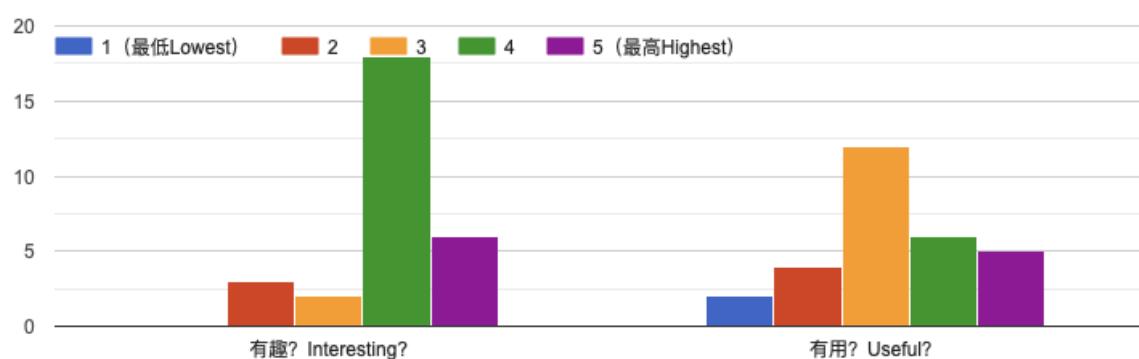
複
製

29 則回應



你認為使用光學字元辨識技術作為輸入行程的方式有趣和有用嗎? Do you think it's interesting and useful to use OCR as a way to input itineraries?

複
製

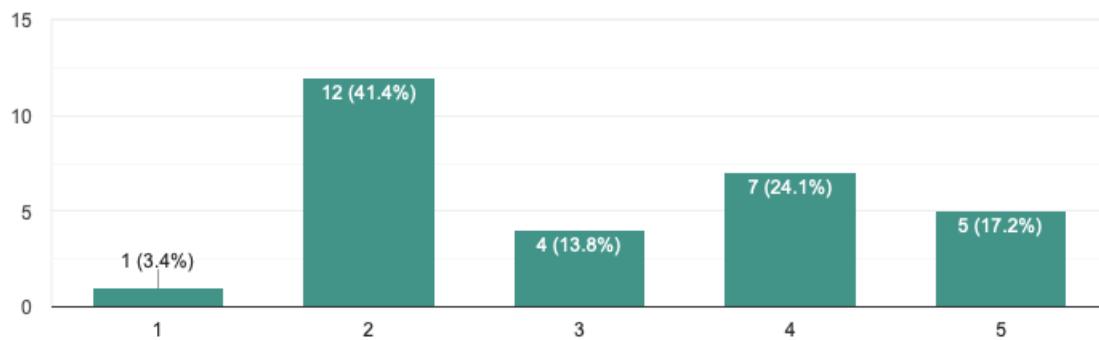


輸入資料必須填在指定空格內以內，你認為這方便嗎？

The input information must be filled in the designated space. Do you think this is convenient?

複製

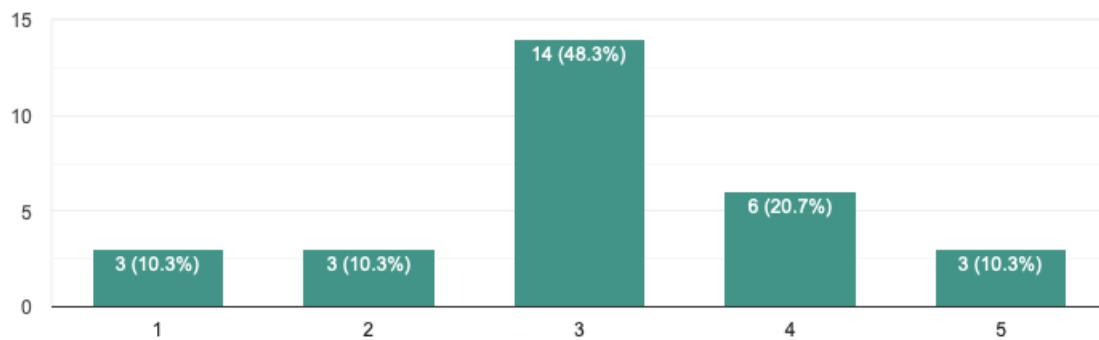
29 則回應



你有多大程度用使用這個功能？ How much do you use this feature?

複製

29 則回應

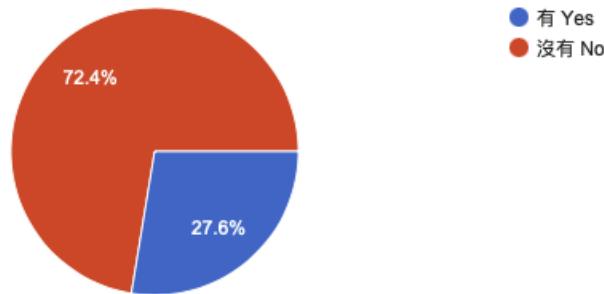


電子日曆功能調查2 Electronic calendar function survey 2

請問你曾經聽過或使用過會自動安排行程的日曆嗎? Have you ever heard of or used a calendar that automatically arranges your schedule?

複製

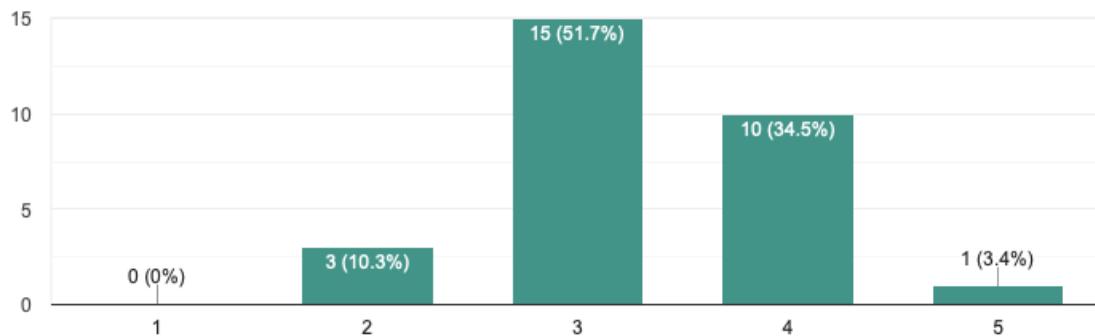
29 則回應



你認為該功能使用方法方便嗎? Do you find this feature convenient to use?

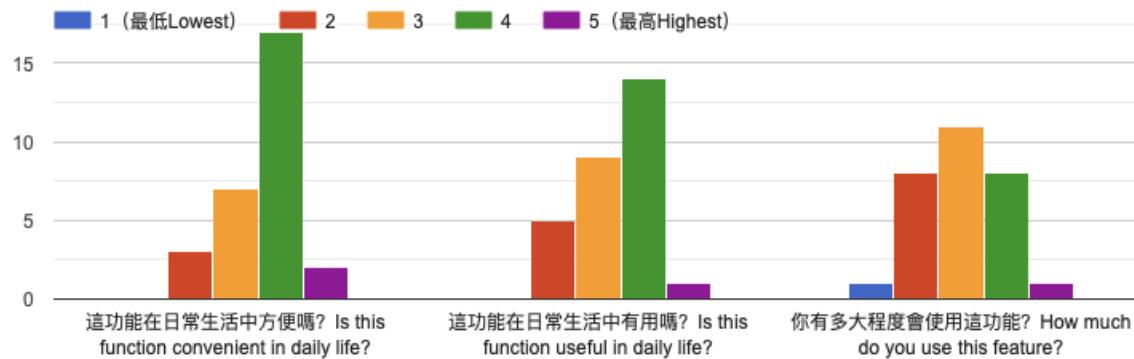
複製

29 則回應



請評分以下問題。Please rate the following questions.

複製

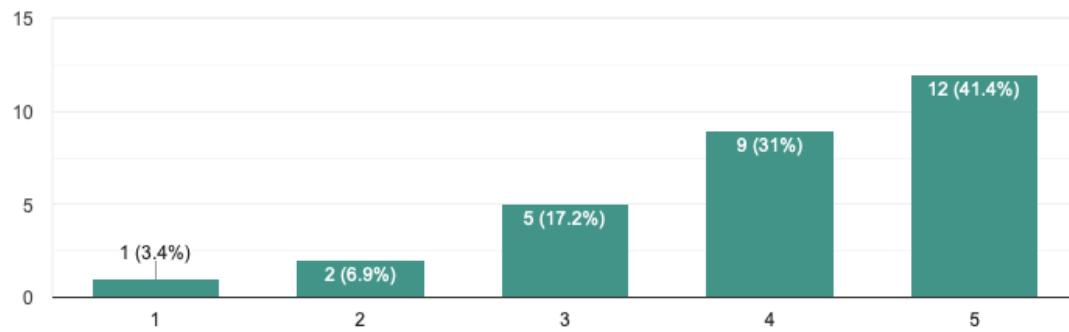


電子日曆功能調查3 Electronic calendar function survey 3

你認為以下數據圖所反映的結果清晰嗎? Do you think the results reflected in the data graph below are clear?

複
製

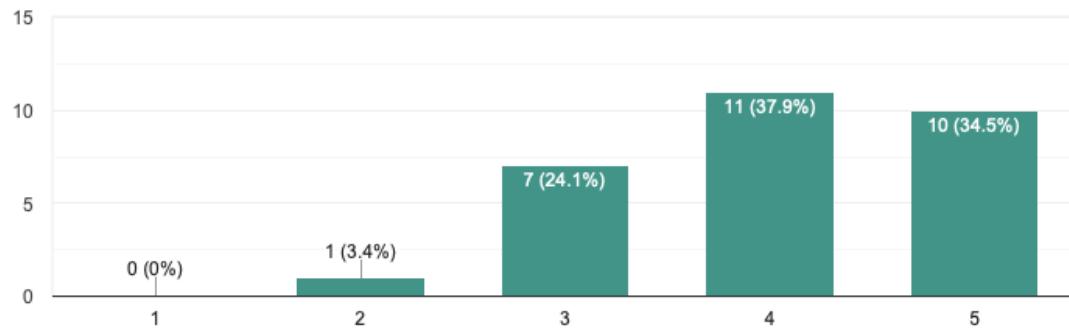
29 則回應



你認為此功能能夠反映用戶的日常數據嗎? Do you think this feature can reflect the user's daily data?

複
製

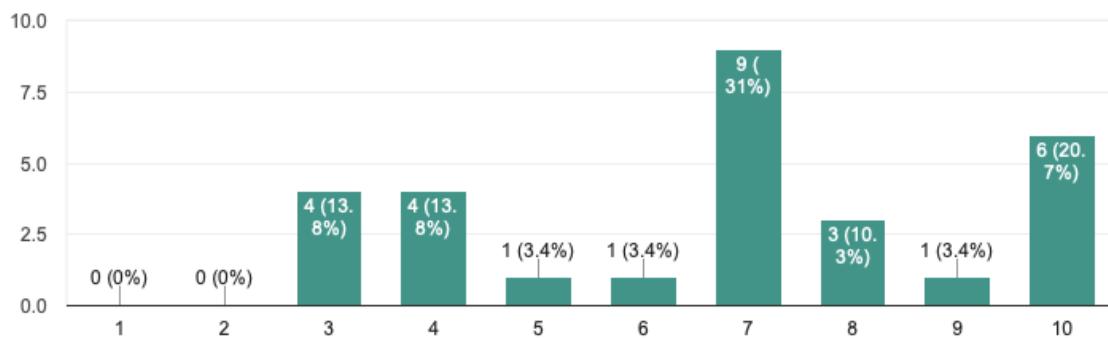
29 則回應



你會經常檢視該數據嗎? How often do you review this data?

複製

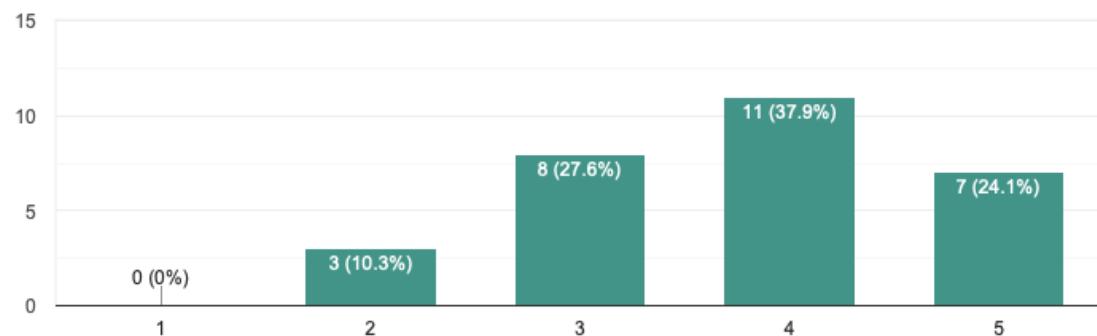
29 則回應



你認為該功能實用嗎? Do you think this feature is useful?

複製

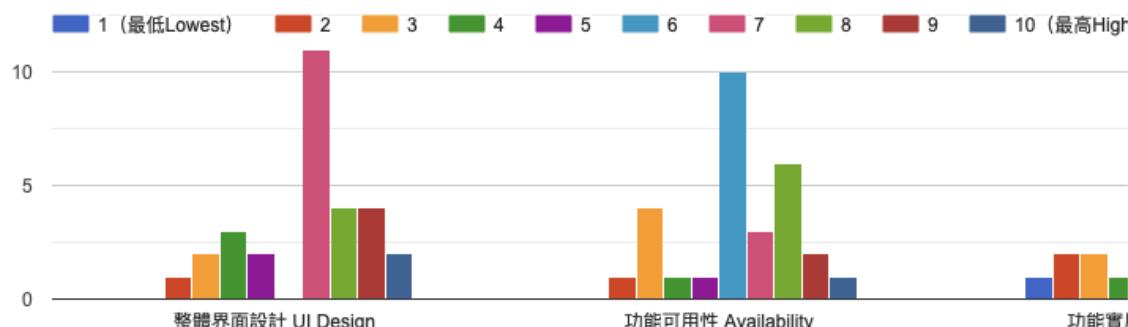
29 則回應



評分 Score

請為以下問題評分。Please rate the questions below.

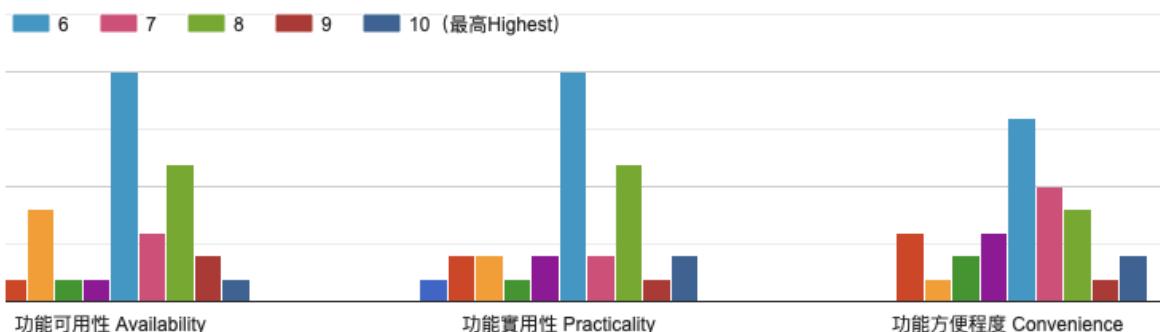
 複製



評分 Score

請為以下問題評分。Please rate the questions below.

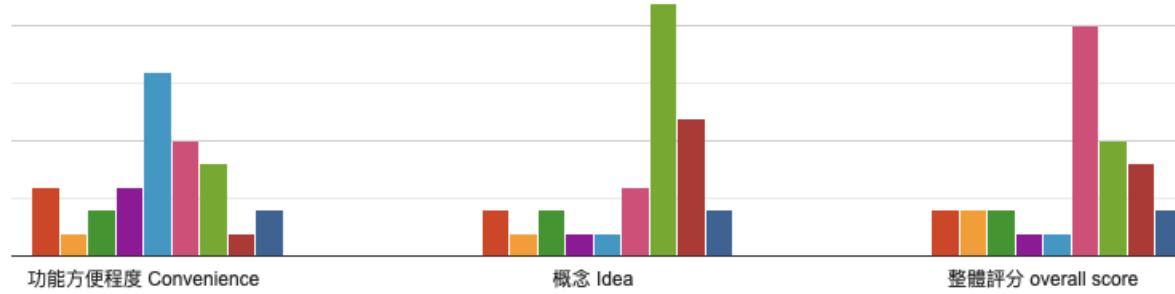
 複製



評分 Score

請為以下問題評分。Please rate the questions below.

複製



Appendix E - Test of Auto-schedule

Time taken of manually schedule:<https://youtu.be/h38yYAA6Fqo>

Time taken of auto-schedule:<https://youtu.be/GHnSjvCouvY>