

# SUBCAL

不會再忘記訂閱了什麼奇怪的服務

B11323213 @ 黃宥睿



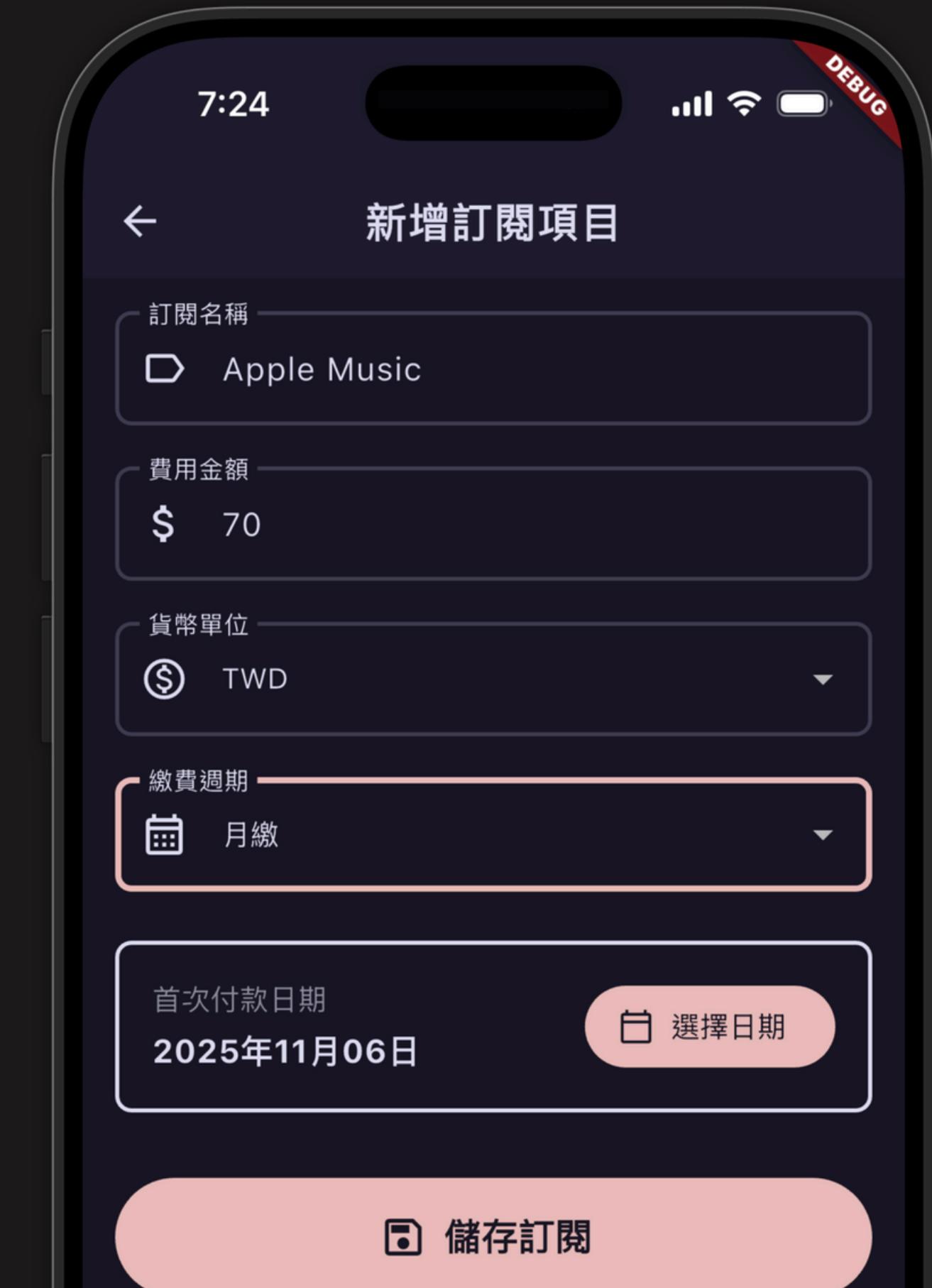
# 專案介紹



SUBCAL 是一款用來記錄網路訂閱服務的 APP  
使用者可以將有訂閱的服務紀錄在 APP 內  
APP 會記錄每月、總年花費，和日曆付費  
讓使用者不再發生突然信用卡被扣款  
卻不知道是訂閱了什麼服務

# 功能介紹

新增訂閱項目，可自訂名稱、費用、貨幣單位等  
也可以設定繳費週期，讓記帳變得更加容易！





# 功能介紹

所有訂閱項目一目瞭然！



## 我的訂閱項目

- |  |             |                |
|--|-------------|----------------|
|  | Apple Music | TWD70          |
|  | 週期: 月繳      | 下次: 2025/12/06 |
|  | 威威孟孟親筆信     | TWD67          |
|  | 週期: 月繳      | 下次: 2025/12/18 |
|  | ChatGPT     | TWD2,400       |
|  | 週期: 年繳      | 下次: 2026/07/22 |

+ 新增訂閱



列表



日曆

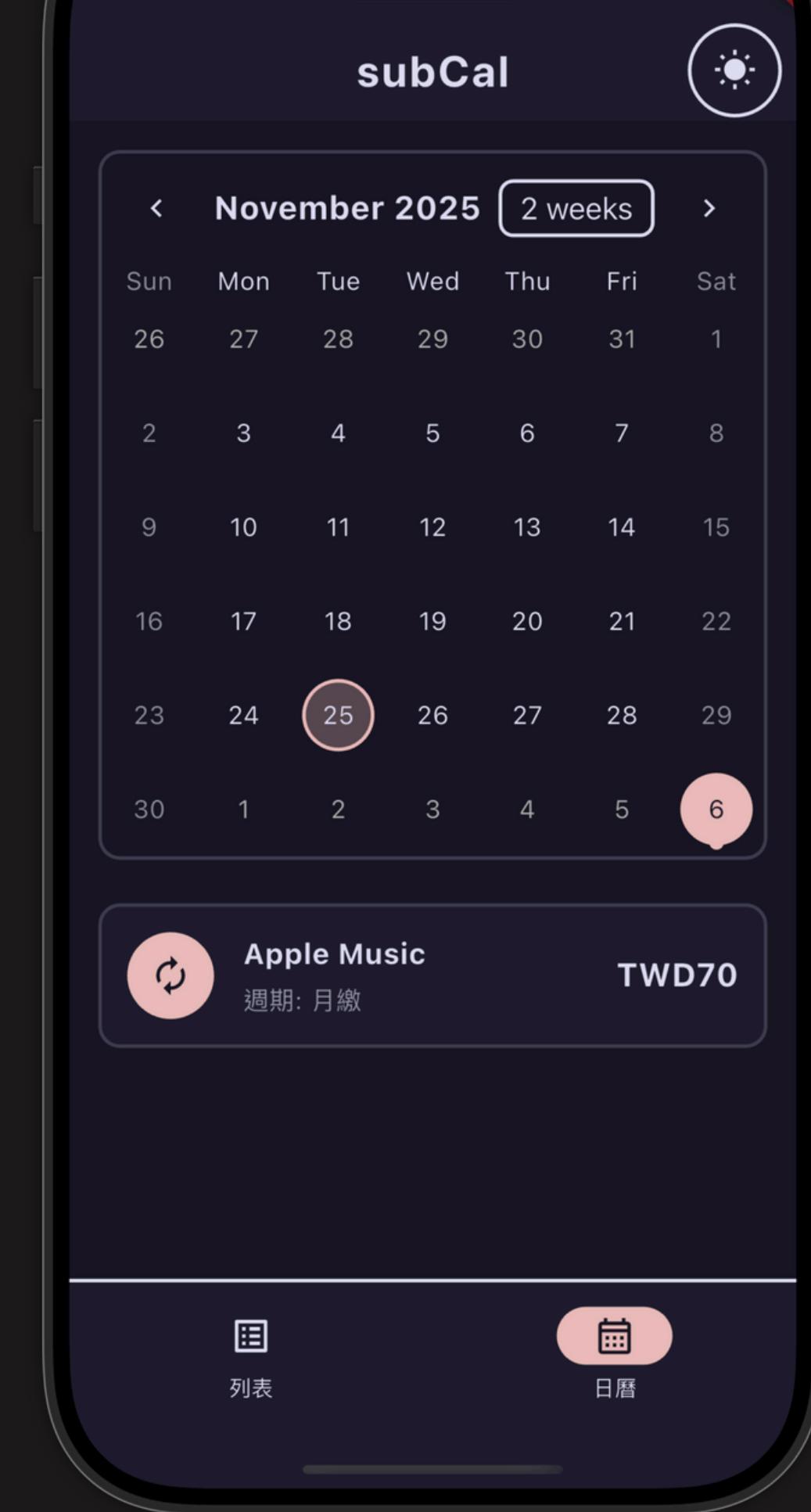
# 功能介紹

輕鬆刪除訂閱項目  
二次確認讓使用者不會誤觸



# 功能介紹

透過日曆查看下次付費時間

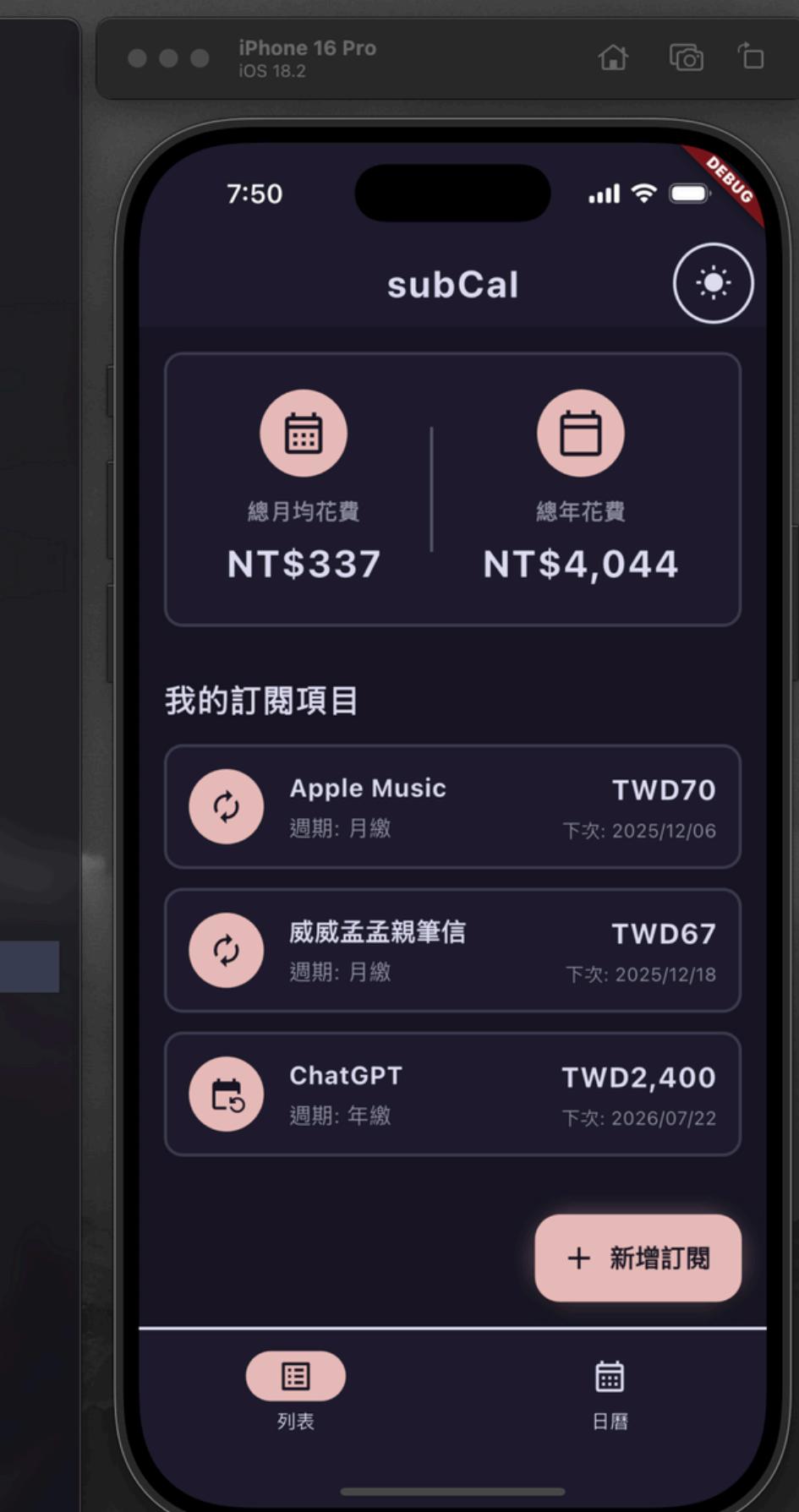


# 功能介紹

**LIGHT MODE & DARK MODE!!!!**



```
17 import 'package:flutter/material.dart';
16 import 'package:intl/intl.dart';
15 import 'package:table_calendar/table_calendar.dart';
14 import 'package:shared_preferences/shared_preferences.dart';
13
12 // --- 資料模型 (Data Model) ---
11
10 /// 訂閱項目的資料結構
9 class Subscription {
8   final String id;
7   final String name;
6   final double cost;
5   final String currency;
4   final BillingCycle cycle;
3   final DateTime startDate; // 第一筆付款的日期
2
1
Subscription({
18   required this.id,
1   required this.name,
2   required this.cost,
3   required this.currency,
4   required this.cycle,
5   required this.startDate,
6 });
7
8   /// 根據當前日期計算下一次付款日期
9   DateTime get nextPaymentDate {
10     DateTime now = DateTime.now();
11     DateTime nextDate = startDate;
12 }
```



# 程式介紹

```

29
28     /// 根據當前日期計算下一次付款日期
27     DateTime get nextPaymentDate {
26         DateTime now = DateTime.now();
25         DateTime nextDate = startDate;
24
23         // 讓 nextDate 往後推移，直到它超過或等於今天
22         while (nextDate.isBefore(other: now)) {
21             if (cycle == BillingCycle.monthly) {
20                 // 月繳：加一個月
19                 nextDate = DateTime(year: nextDate.year, month: nextDate.month + 1, day: nextDate.
day);
18             } else {
17                 // 年繳：加一年
16                 nextDate = DateTime(year: nextDate.year + 1, month: nextDate.month, day: nextDate.
day);
15             }
14         }
13         return nextDate;
12     }
11
10     /// 獲取未來12個月的所有付款日期
9         List<DateTime> getAllUpcomingPaymentDates() {
8             List<DateTime> dates = <DateTime>[];
7             DateTime current = nextPaymentDate;
6             DateTime limit = DateTime.now().add(duration: const Duration(days: 365));
5
4             while (current.isBefore(other: limit)) {
3                 dates.add(value: current);
2                 if (cycle == BillingCycle.monthly) {
1                     current = DateTime(year: current.year, month: current.month + 1, day: current.day);
54                 } else {
1                     current = DateTime(year: current.year + 1, month: current.month, day: current.day);
2                 }
3             }
4         }
        return dates;
    }

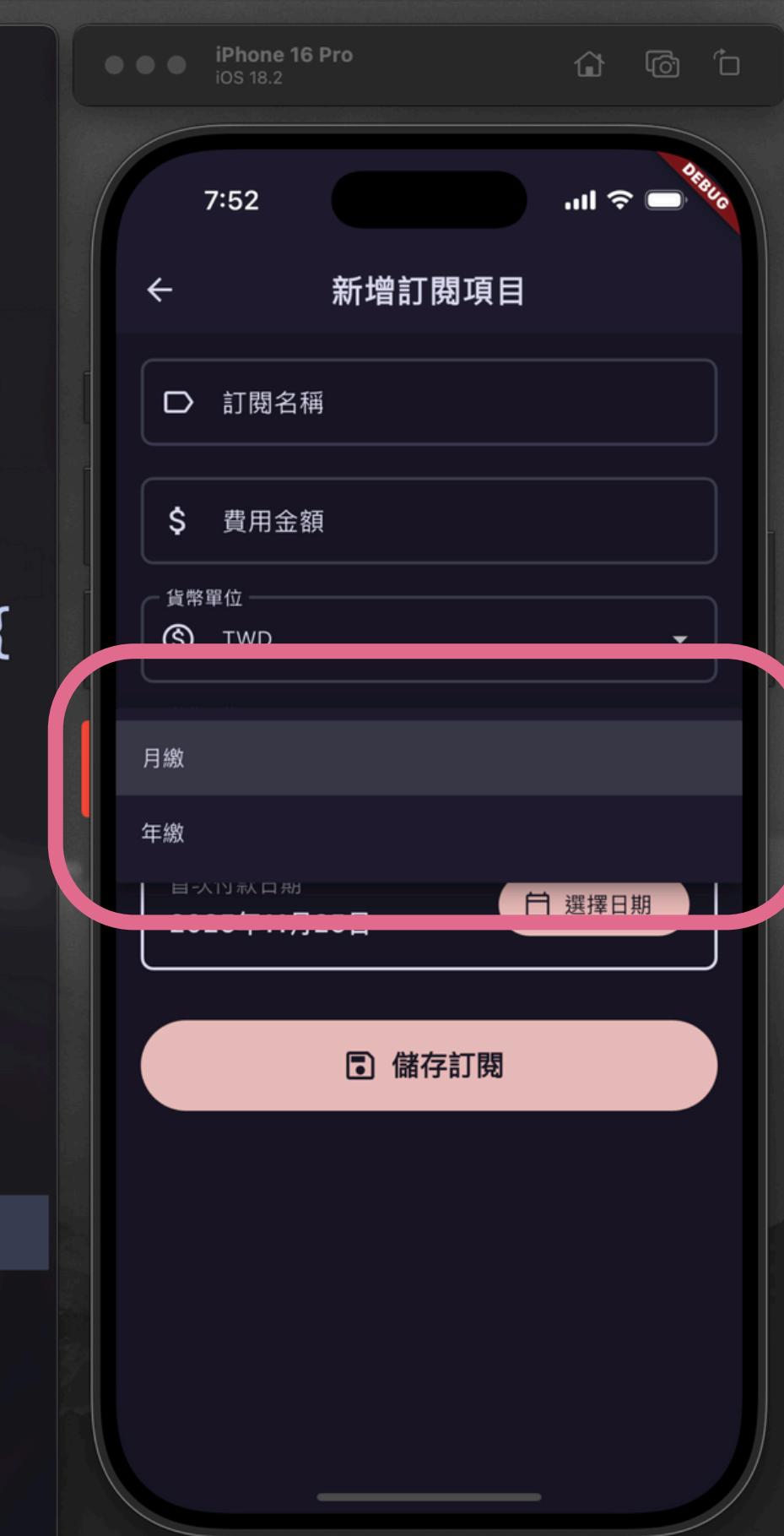
NORMAL <main.dart > Subscription f getAllUpcomingPaymentDates gj ⏺ 23 3% 54:14 ⏺ 19:51
osga@OsGa-mbp app 0:nvim*M 1:dart- ⏺ 19:51:24 / 2025-11-25

```



# 程式介紹

```
15    }
14 }
13
12 /// 帳單週期
11 enum BillingCycle { monthly, annually }
10
9 // 擴充 Enum 來提供更友好的顯示名稱
8 extension BillingCycleExtension on BillingCycle {
7     String get displayName {
6         switch (this) {
5             case BillingCycle.monthly:
4                 return '月繳';
3             case BillingCycle.annually:
2                 return '年繳';
1             }
79     }
1 }
2
3 // --- Theme Provider ---
4
```



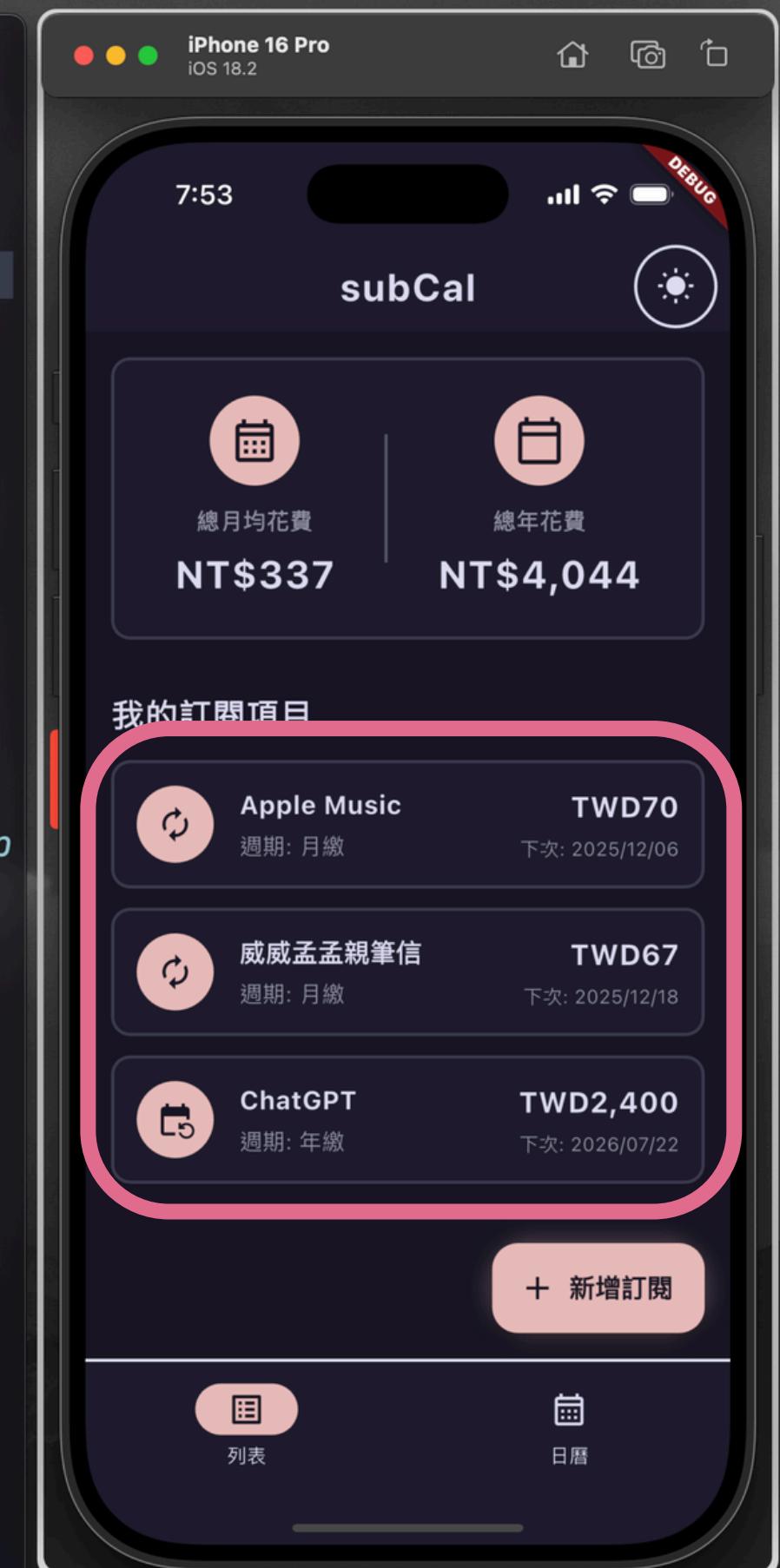
# 程式介紹

```

4
3 // --- 應用程式主頁面 (Home Page) ---
2
1 class SubscriptionHomePage extends StatefulWidget {
307    final ThemeProvider themeProvider;
1
2     const SubscriptionHomePage({super.key, required this.themeProvider});
3
4     @override
5     State<SubscriptionHomePage> createState() => _SubscriptionHomePageState();
6
7
8     class _SubscriptionHomePageState extends State<SubscriptionHomePage> {
9         int _selectedIndex = 0;
10
11        // 示範用的初始數據
12        List<Subscription> _subscriptions = <Subscription>[      ● The private field _subscriptions
13            Subscription(
14                id: '1',
15                name: 'Netflix',
16                cost: 330.0,
17                currency: 'TWD',
18                cycle: BillingCycle.monthly,
19                startDate: DateTime(year: 2024, month: 10, day: 15),
20            ), // Subscription
21            Subscription(
22                id: '2',
23                name: 'Google One (2TB)',
24                cost: 3300.0,
25                currency: 'TWD',
26                cycle: BillingCycle.annually,
27                startDate: DateTime(year: 2024, month: 7, day: 1),
28            ), // Subscription
29        ];

```

NORMAL </main.dart > SubscriptionHomePage themeProvider gk ⚡ 23 22% 307:6 ① 19:53  
osga@OsGa-mbp app 0:nvim\*M 1:dart- ② 19:53:37 2025-11-25



# 程式介紹

```
26     ];
25
24     /// 計算所有訂閱的總月費 (已轉換成年繳 / 12)
23     double get totalMonthlyCost {
22         double total = 0.0;
21         for (var Subscription sub in _subscriptions) {
20             if (sub.cycle == BillingCycle.monthly) {
29                 total += sub.cost;
28             } else {
27                 // 年繳平均到每月
26                 total += sub.cost / 12.0;
25             }
24         }
23         return total;
22     }

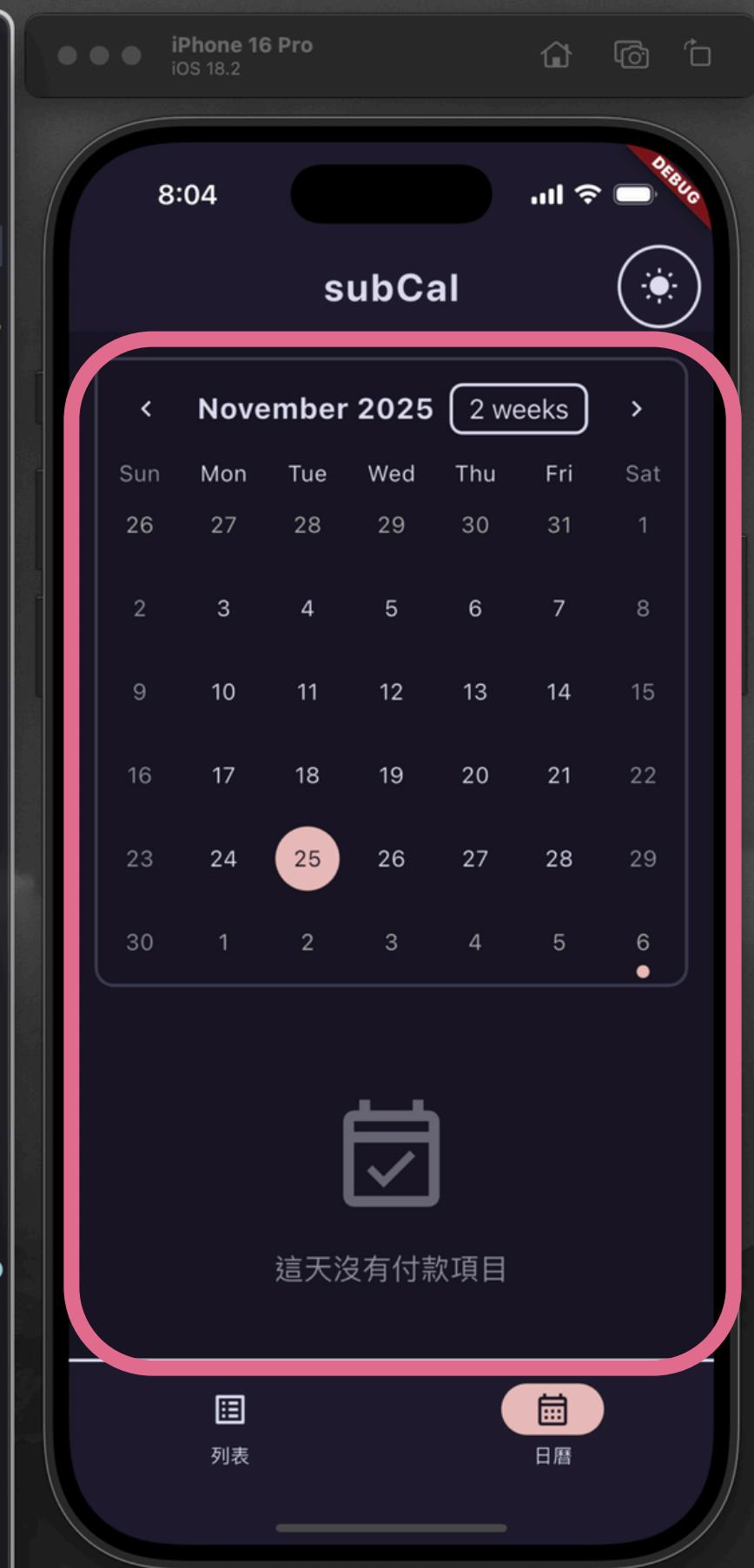
21     /// 計算所有訂閱的總年費
20     double get totalAnnualCost {
19         double total = 0.0;
18         for (var Subscription sub in _subscriptions) {
17             if (sub.cycle == BillingCycle.annually) {
16                 total += sub.cost;
15             } else {
14                 // 月繳乘以 12
13                 total += sub.cost * 12.0;
12             }
11         }
10     }
9         return total;
8     }

7         /// 導航到新增訂閱頁面
6 
```



# 程式介紹

```
4           Expanded(child: _buildEventList()),
3         ],
2       ); // Column
1     }
681
1   Widget _buildEventList() {
2     final bool isDark = Theme.of(context: context).brightness == Brightness.dark; • The value of the
3
4     if (_selectedDay == null) {
5       return const Center(child: Text(data: '請選擇日期'));
6     }
7
8     final List<Subscription> events = _getEventsForDay(day: _selectedDay!);
9
10    if (events.isEmpty) {
11      return Center(
12        child: Column(
13          mainAxisAlignment: MainAxisAlignment.center,
14          children: <Widget>[
15            Icon(
16              icon: Icons.event_available_outlined,
17              size: 80,
18              color: Theme.of(context: context).colorScheme.onSurface.withOpacity(opacity: 0.4),
19            ), // Icon
20            const SizedBox(height: 16),
21            Text(
22              data: '這天沒有付款項目',
23              style: TextStyle(
24                fontSize: 18,
25                color: Theme.of(context: context).colorScheme.onSurface.withOpacity(opacity: 0.6),
26                fontWeight: FontWeight.w500,
27              ), // TextStyle
28            ), // Text
29          ],
30        ), // Column
31      ); // Center
32    }
33
34    return ListView.builder(
35      padding: const EdgeInsets.symmetric(horizontal: 16),
```



程式介紹

# 未來期望

---

第一次做完整的 **APP** 滿有趣的，從生活上的問題點下手開發實用的工具

透過這次的製作讓我對開發 **APP** 更感興趣，希望之後可以把這 **APP** 做得更完善

並且真的在日常生活中使用

目前想到可以添加的功能：

- 扣款通知提醒
- 共享訂閱
- 編輯已有訂閱項目
- 更多的自定義 **ICON** 等...

# THANKS FOR LISTENING

---

**osga24/subCal**



1 Contributor    0 Issues    0 Stars    0 Forks



**osga24/subCal**

Contribute to osga24/subCal development by creating an account on GitHub.

[GitHub](#)

專案連結



DEMO 影片