

浙江大学

本科实验报告

课程名称: 移动平台技术开发

姓 名: 朱雨珂

学 院: 计算机科学与技术

专 业: 软件工程

学 号: 3190103312

指导教师: 章国锋、陶煜波

2022 年 3 月 31 日

浙江大学实验报告

课程名称: 移动平台技术开发 实验类型: 软件开发

实验项目名称: iOS programming assignment

学生姓名: 朱雨珂 专业: 软件工程 学号: 3190103312

同组学生姓名: 单人作业 同组学生学号: 单人作业

指导老师: 章国锋、陶煜波

实验地点: 线上 实验日期: 2022年 3月 29日

目录

1. 项目介绍	1
2. 任务概述	2
3. 总体设计	3
4. 程序详细描述	4
5. 数据结构设计	5
6. 开发结果	6
7. 讨论与总结	7

1. 项目介绍

提醒事项 App 是 Apple 推出的一款用于创建和整理提醒事项的系统软件。可以用它来跟踪所有日常生活中的待办事项，包括购物清单、工作项目或要跟踪和记住的任何其他事项。在建立提醒事项时，用户可以自定义颜色、时间、地点、提醒内容、重复频次等，具有极高的自由度。

本项目主要复现了 reminder 的 UI 界面和一些简单的交互功能。

2. 任务概述

2.1 成员简介与分工、目标

(个人小作业，无组员/分工)

目标：完成 IOS 官方程序 reminder 的界面复现，并完成基本交互和动画

2.2 运行环境

开发环境：



首要适配硬件：

iPhone12 及以上型号

2.3 程序目标分析与概述

程序目标：

- 1、实现 reminder 的基本界面 UI 复现
- 2、实现一定的 list 动态功能——list 的自定义添加（包括命名、颜色修改），list 的顺

序移动、删除（edit 按钮功能）、list 中 reminder 的添加、实时展示 list 中的 reminders

3、实现一定的 reminders 动态功能——reminder 的自定义添加（包括选择要归入的 list、日期 date、时间 time、定位 location、优先级 priority）

缺失：

- 1、reminder 的重复 repeat 属性没有实现
- 2、edit 按钮中移动 schedule、today、all 的功能没有实现
- 3、list 中的一些小功能没有实现，包括——reminders 的 sort by、delete list、select reminder、show completed 功能
- 4、all、today、scheduled 等子页面里只有展示功能，没有完成实际操作功能
- 5、一些排版问题没有得到解决（主要是 vertical 方向的位移容易导致 button 按钮不能点击，因此有些界面没有解决位移偏移标准点的问题）

所使用的组件概述：

Form、Text、Image、Button、Menu、toolbar、TextField、sheet、Toggle、DatePicker、Picker、ScrollView 等。

所使用的布局方式概述：

VStack、HStack、ZStack、LazyVGrid 等。

所使用的跳转方式主要有 NavigationLink、sheet 等。

所使用/自定义的动画主要有：

主页面 edit 模式中的移动、删除效果；

Add List 页面中的颜色按钮点击效果；

对 New Reminder 页面中 Detail 子页面中的 DatePicker 出现、收起添加 `withAnimation(.easeInOut(duration: 1.5))` 动画效果；

对 New Reminder 页面中 Detail 子页面中由于日期、时间选择产生的显示变化添加了 `withAnimation(.spring())` 效果。

2.4 运行条件与限制

由于本程序未连接数据库，因此只能用作前段展示，不可以做真实的数据存储。

3. 总体设计

3.1 需求分析与设计

3.1.1 需求分析

提醒事项 App 是 Apple 推出的一款用于创建和整理提醒事项的系统软件。可以用它来跟踪所有日常生活中的待办事项，包括购物清单、工作项目或要跟踪和记住的任何其他事项。在建立提醒事项时，用户可以自定义颜色、时间、地点、提醒内容、重复频次等，具有极高的自由度。

本产品面向拥有苹果产品的用户。

3.1.2 流程图

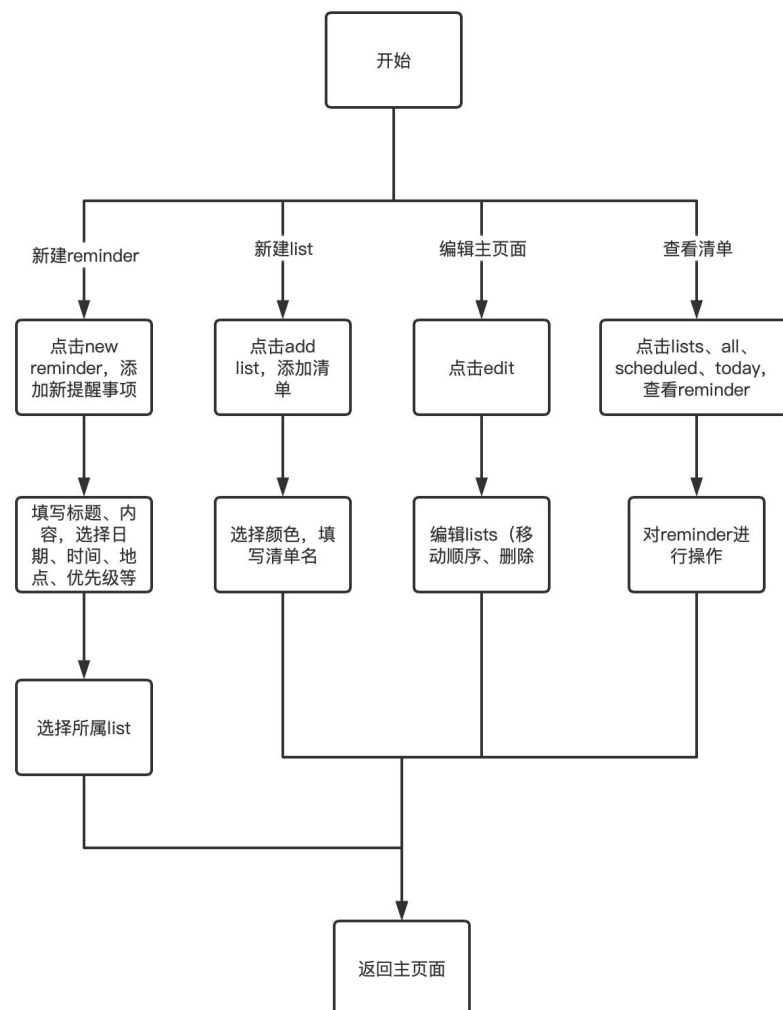


图 1: 程序操作流程图

3.1.3 用例图

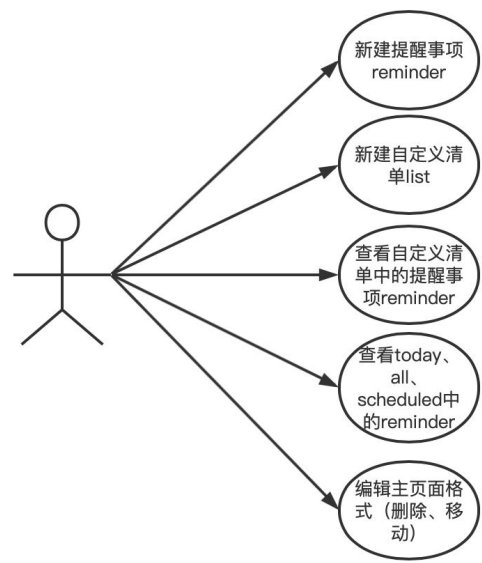


图 2： 用例图

3.2 详细设计

3.2.1 类图

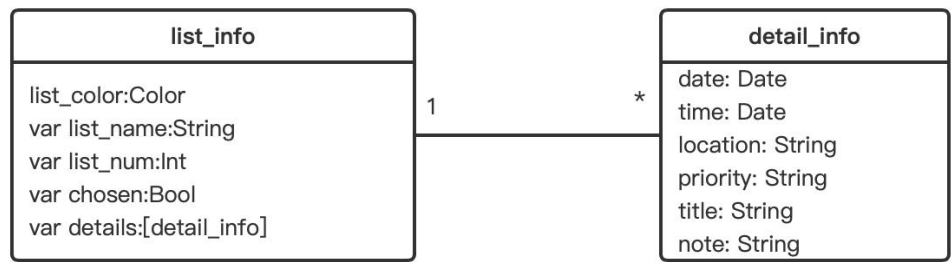
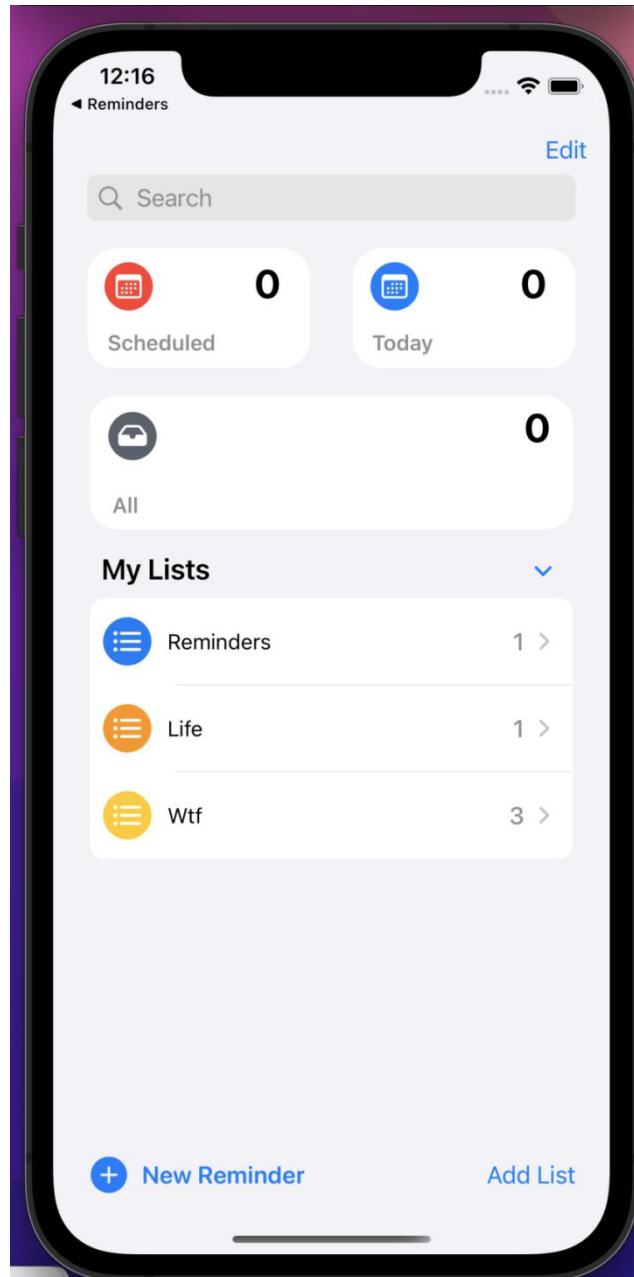


图 3： 类图

4. 程序详细设计

4.1 程序唤醒界面



本程序无唤醒界面，此为程序主界面。

4.2 主程序 UI

这部分详细介绍了各个界面之间的逻辑和所使用的一些组件的名称、结构。

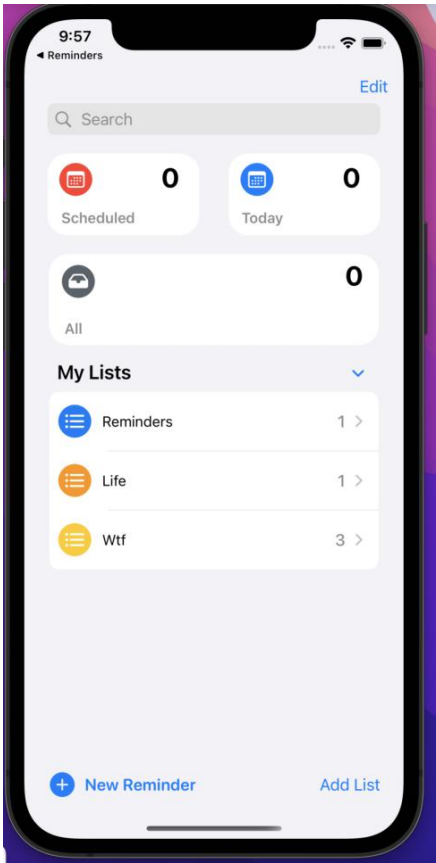
1、界面之间的逻辑：

为方便管理，程序主要分了三个主要的页面——分别是主页面、new

reminder: 添加新的 reminder 页面、add list: 添加 list 页面。页面之间使用 Button、sheet 进行切换。

2、详细界面的组件描述:

1) 主页面



主页面组件主要有——一个搜索框  ,
scheduled  、all  、today  三个矩形，一个
list  , 页面右上角 edit 按钮  以及页面下方两个按钮 new reminder
和 add list  。

其中，all、scheduled、today、list 点击都会进入各自的子页面，new reminder 和 add list 点击会进入 New reminder 页面和 Add list 页面。

主界面结构代码如下:

```

var body: some View{
    VStack{
        HStack{//searchbar 模块，使用 hstack{image text}，加上
            roundedrectangle 形状的 background
            Image()
            Text()
        }
        today_schedule//并排的 today、schedule 两个模块
        all//all 模块
        list//下方的 lists 模块
    }.toolbar{//底部工具栏
        Button("new reminder")//左下方 new reminder 的按钮，使用
        hstack{image text}的排版
        Button("add list")//右下方的 add list 按钮
    }
    .sheet(newreminder_page)//开启新建 reminder 的页面
    .sheet(addlist_page)//开启新建 list 的页面
}

```

以 list 组件为例，结构代码如下（all、today、scheduled 组件都是使用 ZStack 叠加 Text、Rectangle、Image 组件，使用 navigationlink 进行跳转）：

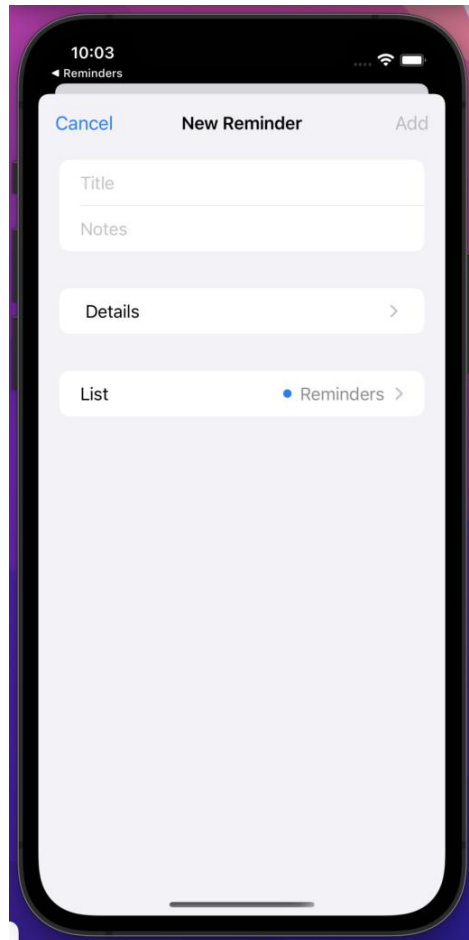
```

var list:some View{//主页面 list 组件
    List{
        Foreach(list){//遍历所有的 list
            navigation(destination:reminders_page){//每个 list
            都跳转到 reminders_page
                HStack{//横排
                    Image
                    Text(name)//list 的 name
                    Text(number)//list 中 reminder 的个数
                }
            }
        }
        .onDelete { deletelist(at :$0) }//edit 按钮的功能——删除
        该 list
        .onMove { movelist(from: $0, to: $1) }//edit 按钮的功能
        ——移动 list
    }.toolbar{editbutton}//右上方的 edit 按钮
}

```

2) New reminder 页面

- 主页面



这个页面主要采用 Form+Section 格式, 将 title、note、detail 和 list 都放在 section 中。其中 details 和 list 另有子页面。

```
struct newreminder_page:View{//这是添加 reminder 的页面
    var body: some View{//主页面
        VStack{
            Form{//使用 Form-section 模式
                Section{
                    TextField()//title
                    TextField()//note
                }
                Section{//details
                    NavigationLink(destination:details){
                        //跳转到 details 页面
                        Text("Details")//如果已经选择了
                        details 的详细内容, 会显示详细的内容小字
                    }
                }
            }
        }
    }
}
```

- detail 子页面

detail 子页面如下:



date 组件和 time 组件的结构代码如下：

```
Button{
  Toggle{
    if(!editingDate){//如果不是在编辑时间，只显示“Date”
      HStack{
        Image()
        Text("Date")
      }
    }
    else{//如果编辑了时间，就显示“Date”+date
      HStack{
        Image()
        VStack{
          Text("Date")
          Text(date)
        }
      }
    }
  }
}
if((editingDate&&firstDate)||(!firstDate&&showingDate&&editingDate)) {
//这里指如果是第一次点击 toggle, 或者后面点击整个 date 组件, 都可以开、关 DatePicker
  DatePicker()
}
```

location 组件的结构代码如下：

```
Section{
  VStack{
    Toggle{//toggle 组件，选择则显示一系列 location 图标
      HStack{
        Image()
        Text("Location")
      }
    }
    if showingLocation {
      HStack{//横排 4 个 location 选择图标，这里只展示一个
        VStack{//一个可以点击的 location 图标
```

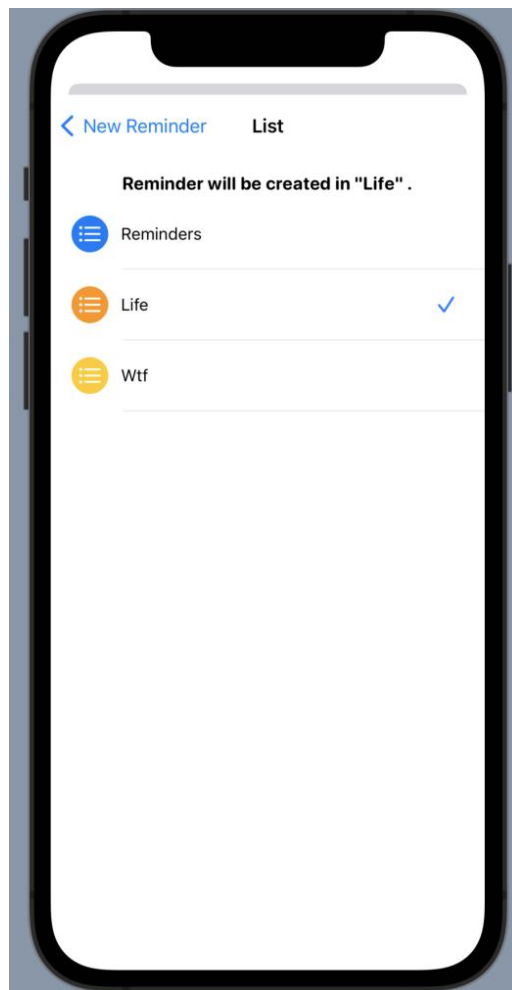
```

        Image()
        Text()
      }.onTapGesture(perform: {}))
    }
  }
  if showingLocation{
    HStack{//所选择的 location 会显示在第二行
      Text(location)
      Image()
    }
  }
}
}
}

```

- lists 子页面

list 页面功能为选择所属的 list，所使用的组件主要有 Text、Image、Divider 等，使用 HStack 和 VStack 排版。

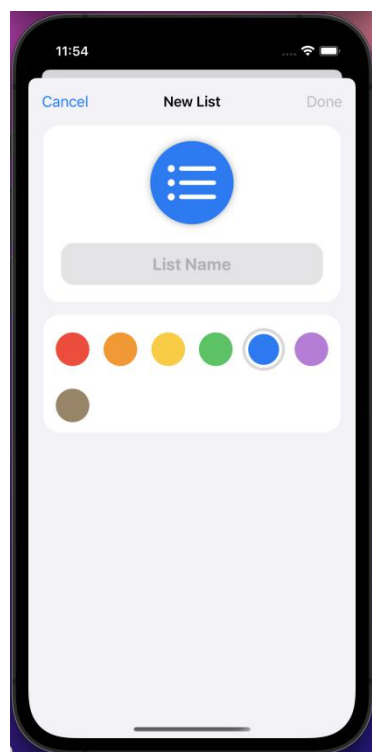


其结构代码如下：

```
VStack{
  Text()//实时显示选中的 list
  ForEach(lists){//这里循环 lists
    Button(){
      VStack{
        HStack{
          Image()
          Text()
          if(lists[index].chosen==true){//如果选中
            Image()//显示打勾
          }
        }
        Divider()
      }
    }
  }
}
```

3) Add list 页面

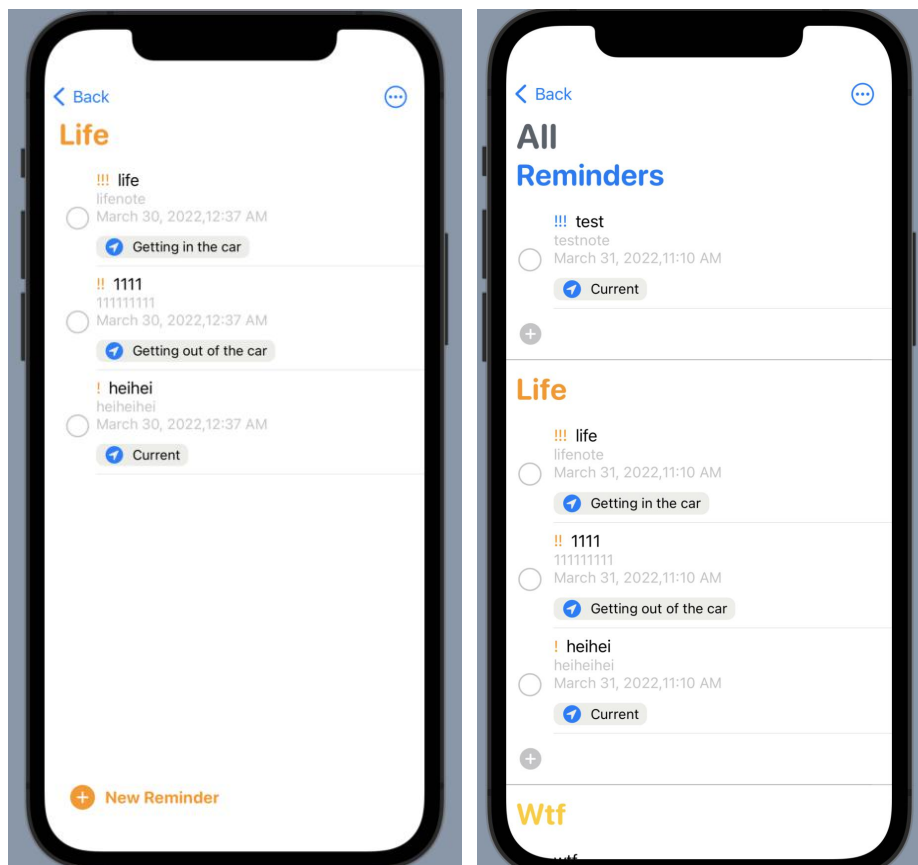
这个界面为添加新 list 的界面，主要使用的组件有 Image、TextField、Button、Rectangle 等，使用 VStack、ZStack、HStack 和 LazyVGrid 进行排版，具体结构代码不细说啦。同时，此处制作了颜色按钮的点击动画，点击后会有相应的效果。



4) 子页面们（以 reminders_page、All_page 为例）、具体信息页面

以 reminders_page 页面为例，主要使用 Button、Image、Text、toolbar、Menu 等组件，使用 ScrollView、HStack、ZStack、VStack 等布局方式，其中细节比较琐碎，因此不一一列出结构代码。

All_page 页面则可以顺序显示全部 lists 的 reminders。



4.3 核心算法

本程序主要以前端为主，因此没有核心算法。

4.4 数据库设计

由于第一个作业要求中提到使用静态数据即可，因此没有连接数据库。

4.5 其他设计

暂时无

5. 数据结构设计

5.1 list_info 数据结构

list_info 存储的是一个 list 的详细信息，包括颜色、名称、list 中 reminder 的数量、以及用于存储 reminder 的 detail 数组。

```
struct list_info:Identifiable{
    var id=UUID()

    var list_color:Color
    var list_name:String
    var list_num:Int
    var chosen:Bool

    var details:[detail_info]
}
```

5.2 detail_info 数据结构

detail_info 中存储的是 reminder 的具体信息，包括日期、时间、定位、优先级、标题和内容。

```
struct detail_info:Identifiable{
    var id=UUID()

    var date:Date
    var time:Date
    var location:String
    //var repeating: String
    var priority:String
    var title:String
    var note:String
}
```

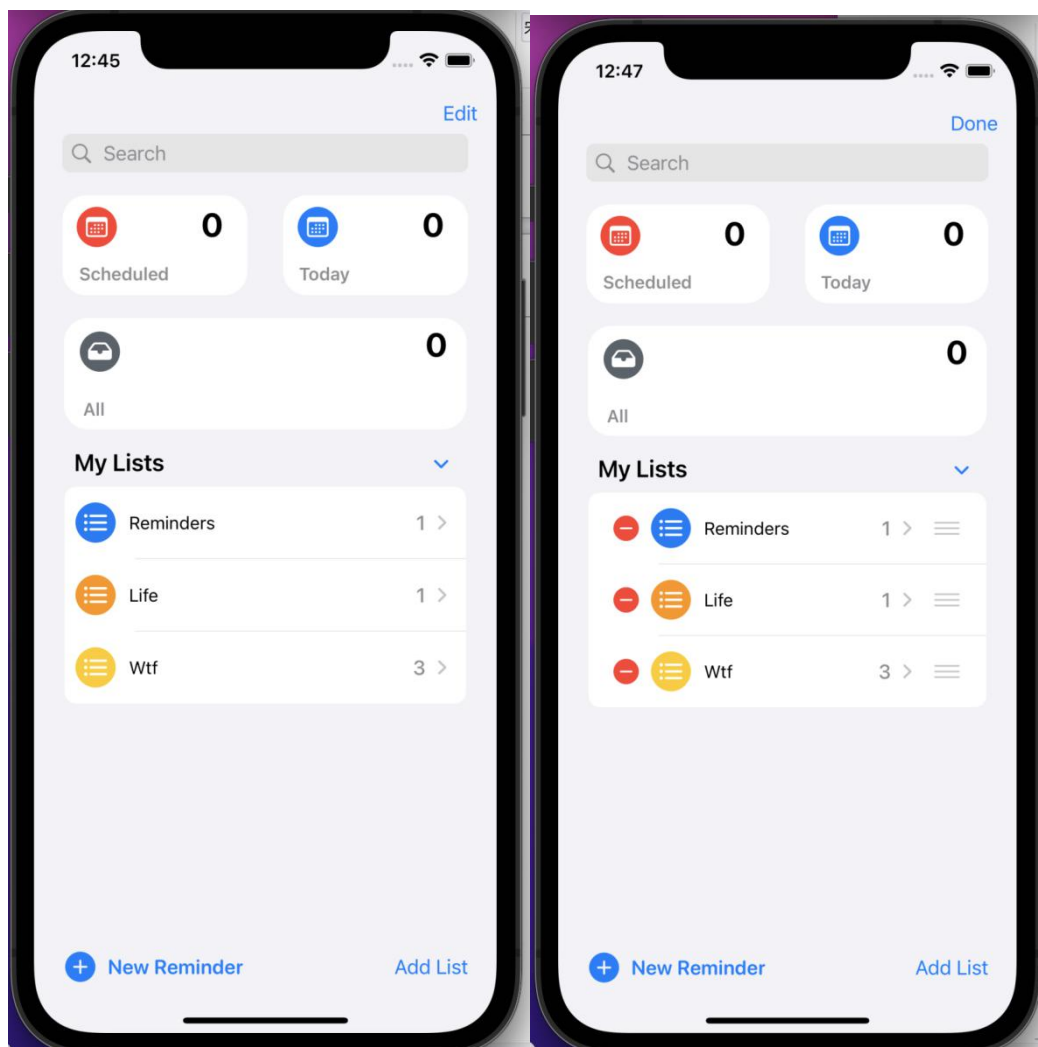
6. 开发结果


6.1 软件使用说明

当前程序只可以展现在 iPhone 设备上，并没有制作适应 iPad 的模式。同时，iPhone 以 iPhone12 及以上版本效果为佳。

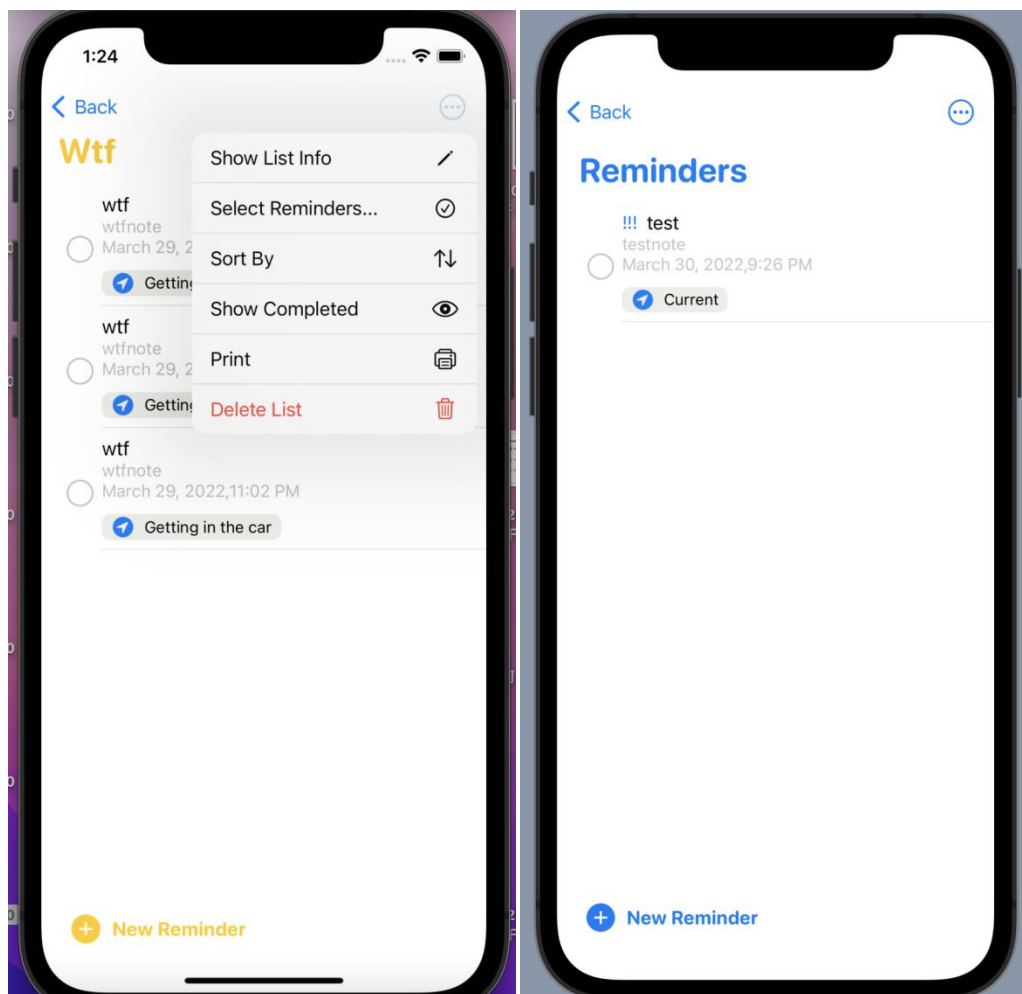
6.2 软件预览

主界面，及主界面 edit 模式（点击 edit 按钮，可以实现 my lists 的移动、删除功能）（My Lists 中皆为实时数据，可以自行删减）：



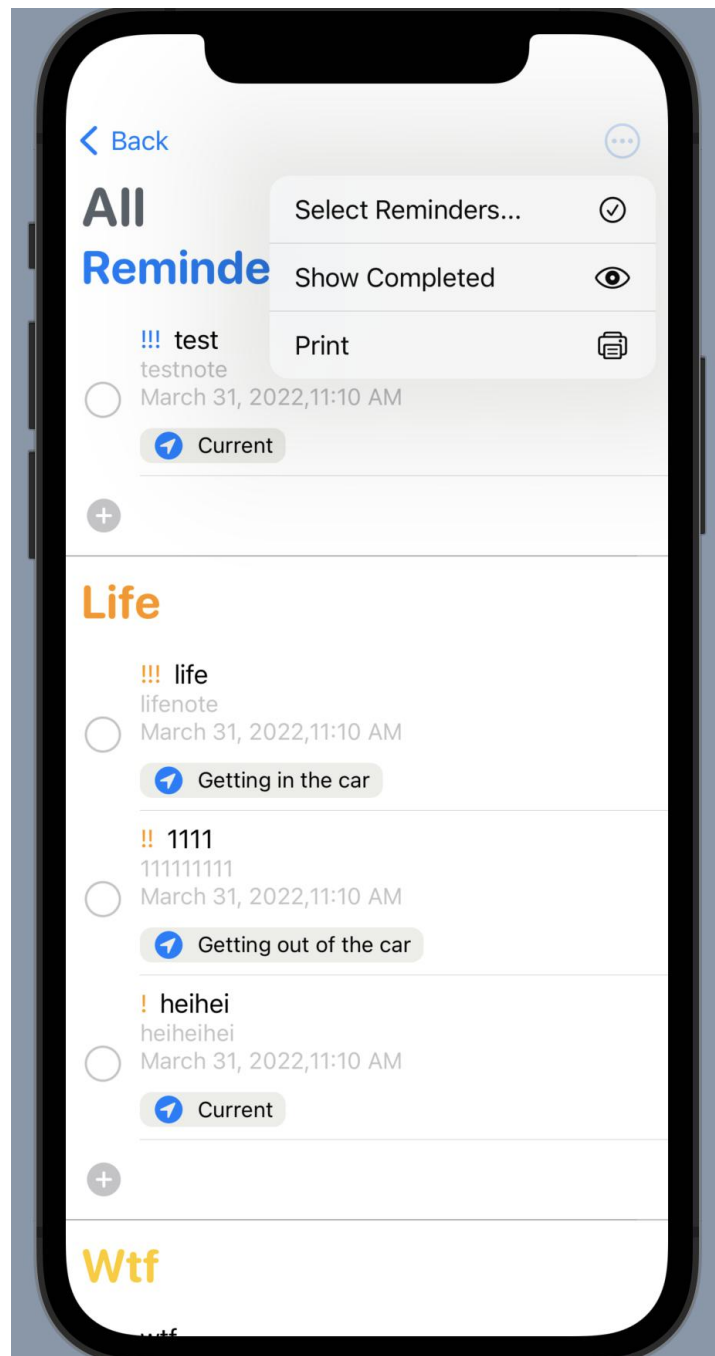
主界面中的 reminders_page，通过点击 My List 中相应 list 到达（有小 bug，需先点击在图标上 ，再点击才可以切换 list），可以实时显示相应 list 中的 reminders 详细信息（包括标题 title、内容 note、日期 date、时间 time、定位 location、优先级 priority），以及点击右上角可以弹出 menu，点击左下角 new reminder 可以新建 reminder（同主页面 new reminder）。

此页面只做展示用，menu 中的一些功能并未实现：

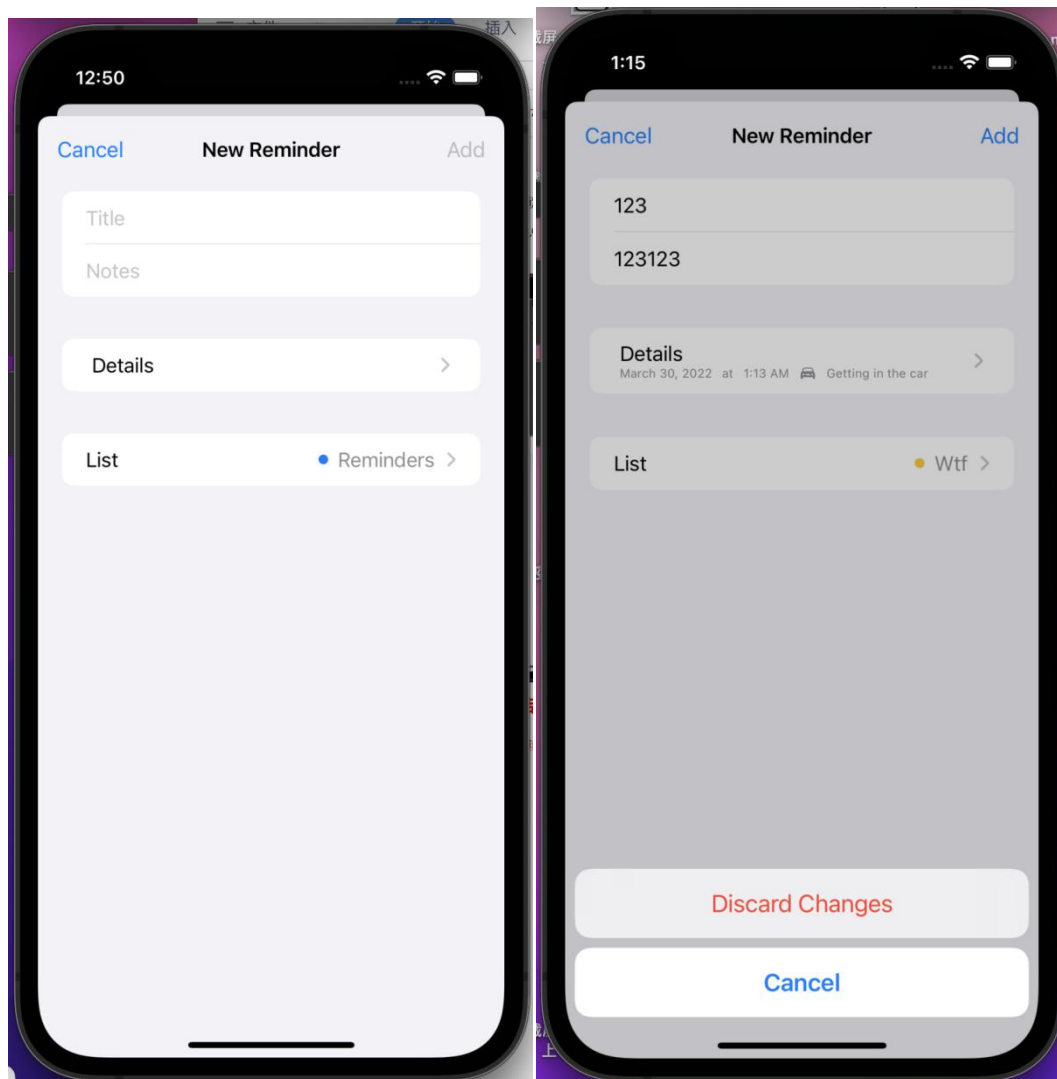


主页面中的 All_page（以此为例，其余和这个都差不多）。其中罗列了所有清单 list 中的所有 reminder。此页面仅作展示用。

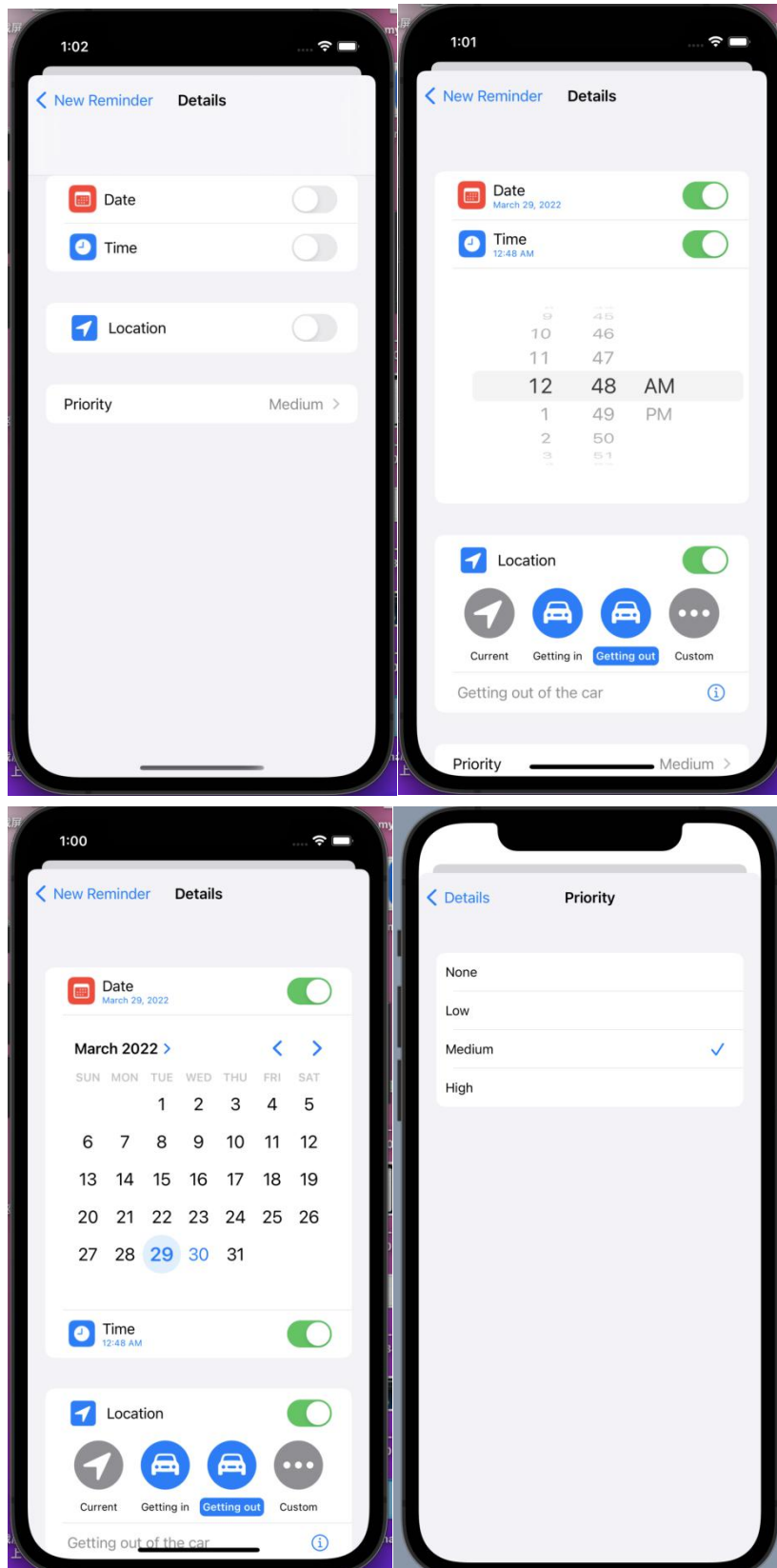
同时，today、scheduled 页面由于一些不知名原因，无法点击（怀疑是 xcode 本身问题，当代码比较复杂时无法做出及时反应）。因此将 today、scheduled 页面暂时只留下一个标题。如果想尝试完整页面，可以将代码中的 today_page、scheduled_page 中的注释去除，但可能产生无法点击、卡顿等后果。



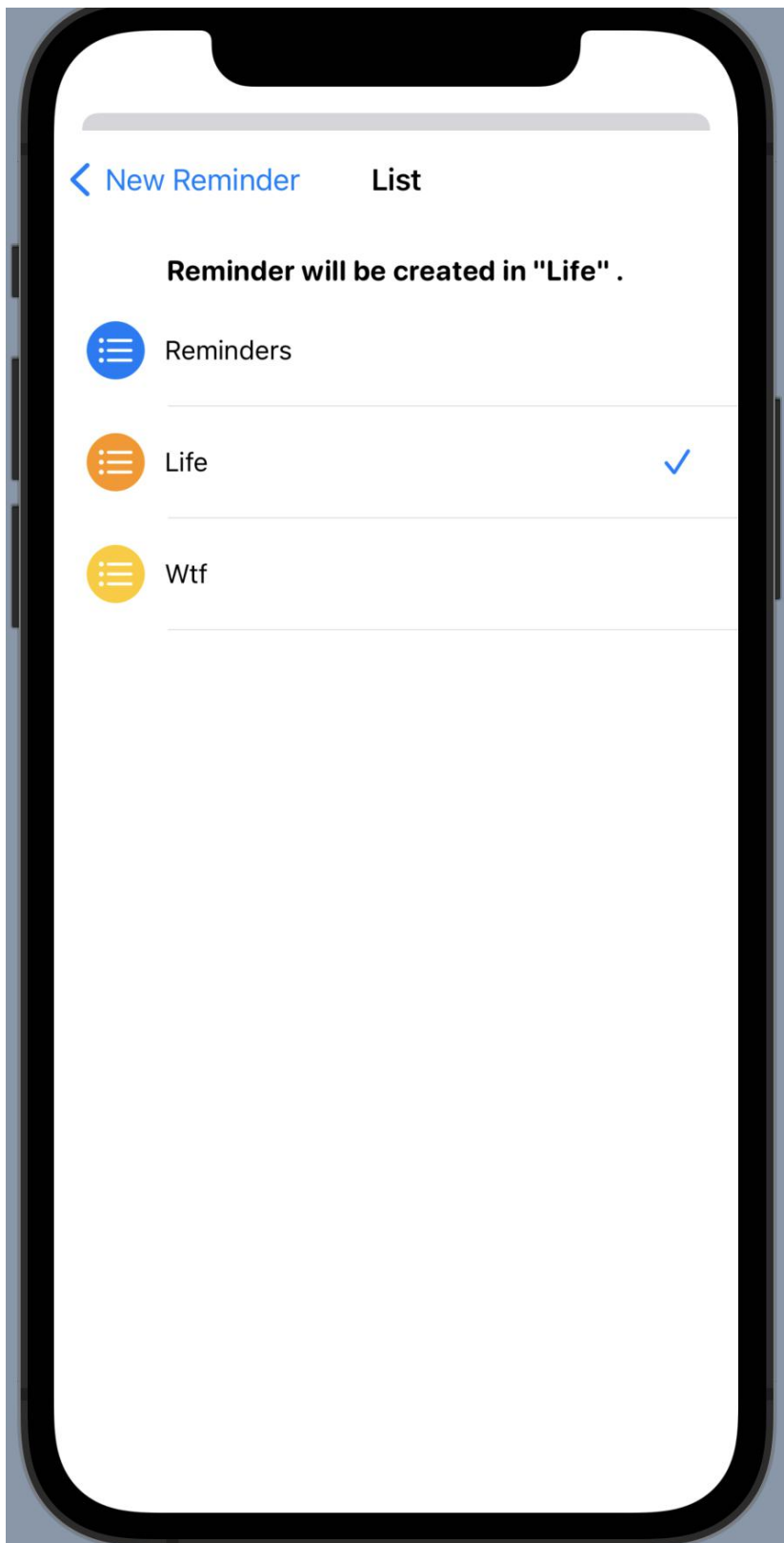
New reminder 页面 (在未输入 title、note 之前, 右上方 add 按钮不可以点击, 在输入之后才允许点击 add), 以及左上方 cancel 的效果 (如果有输入 title、note, 会提示 “diregard change”, 如果为输入则直接回到主界面), 以及如果选择了 detail 中的细节, 会将细节 (包括日期、时间、地点) 实时显示在 “Details” 的下方:



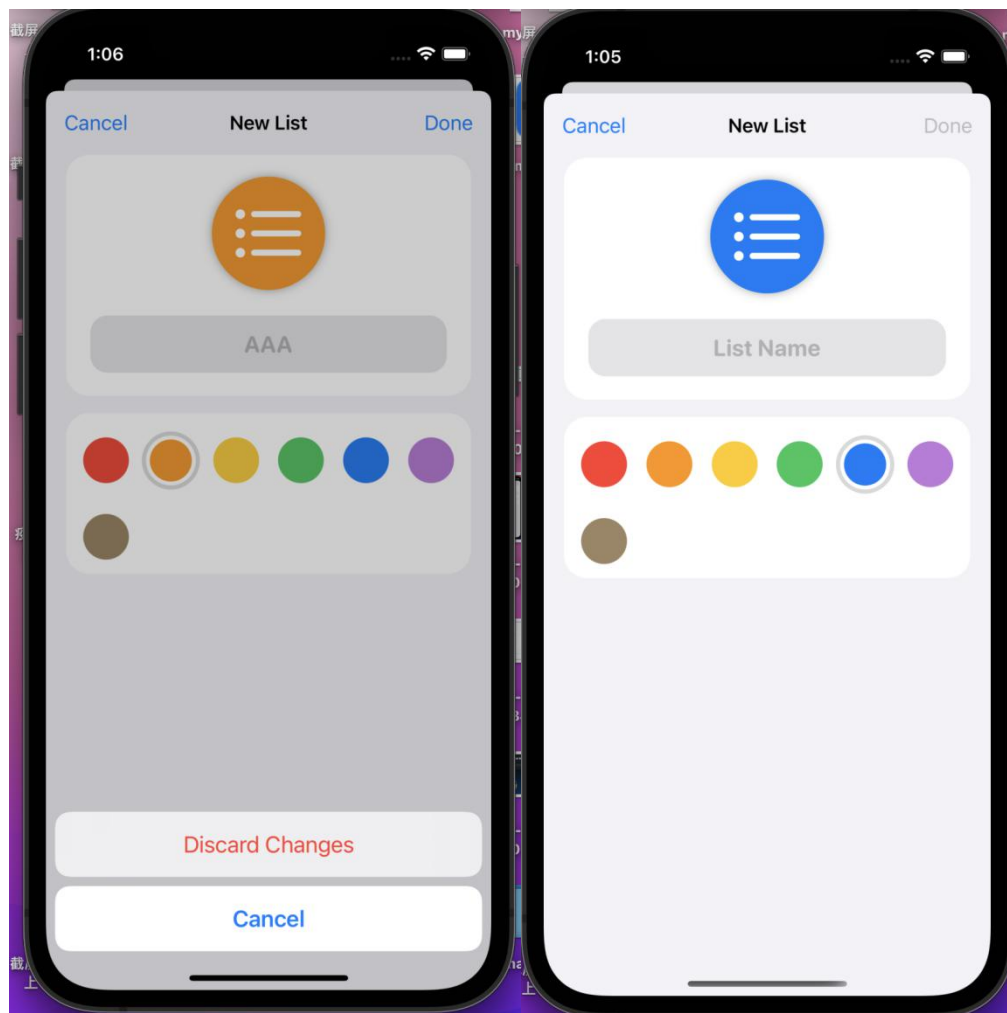
New reminder 页面中的 details 子页面，可以实现选择、下拉 date、time 的功能，选择之后会实时显示蓝色小字；可以实现 location 点击、选择、实时显示在下方的功能；可以实现 priority 选择的功能：



New reminder 页面中的 List 子页面，可以实现所属 list 的选择，并实时显示在上方 text 中：



Add List 页面，实现 list 的自定义添加——包括颜色选择、命名（如果没有输入，则 Done 按钮不可点击），以及 cancel 功能同 new remindew 页面：



7. 讨论与总结

本次作业主要难度在于组件的组和使用、一些细节功能的逻辑（例如在选择 detail 中的 date 和 time 时，首次点击 toggle 可以下滑 timepicker，之后点击整个 time/date 可以实现 timepicker 的显示/隐藏；以及关于 new reminder 和 add list 页面，如果还没有输入任何内容信息，直接点击 cancel 按钮会直接退出，而如果已经输入了一定的信息例如 title，点击 cancel 按钮则会提示“disgard change” or “cancel”）。

还有一个难点是页面排版布局，有很多时候不知道如何处理排版的问题，后

面发现有些排版问题是因为主页面-子页面嵌套了好几层 navigationview。

还有一个问题——在使用 xcode 编程时，我发现 xcode 似乎非常脆弱，不能够支持更复杂的运算和页面布置，经常因此卡顿、报错，例如如果使用 foreach 循环就会卡顿，无法点击页面，但直接使用枚举就没有卡顿问题，因此项目文件中的代码比较冗长，这点很不好 QAQ。

本次项目中的 UI 布局尽可能地按照官方 reminder 进行了复现，同时保留了一定的功能和交互方式，请老师指正。