



ELEWARE TECHNOLOGIES

SEPTEMBER 2024

# NOTIFIER MOBILE APP

# TABLE OF CONTENTS

01

**Project Overview**

02

**Android Studio & Flutter Configurations**

03

**API Documentations**

04

**UI/UX Design**

05

**Code Documentation**

06

**Known Issues and Bugs**

1

# PROJECT OVERVIEW

# INTRODUCTION

Notifier app is a project in which we can track the alarms, assets and tasks within an organization.

There are two versions of Notifier.

1. Xamarin based, developed by Edmund
2. Flutter based, developed by Asyraf



2

# ANDROID STUDIO & FLUTTER CONFIGURATIONS

# ANDROID STUDIO

During the development of the Flutter version, Android Studio was chosen for its cleaner user interface and ease of use with the emulator.

The project was developed using the latest version of Android Studio and tested on Android emulators running API 33 (Tiramisu) and API 27 (Oreo).



# FLUTTER

To continue developing in Flutter, install Flutter from links below:

Windows: <https://docs.flutter.dev/get-started/install/windows>

MacOS: <https://docs.flutter.dev/get-started/install/macos>

The configurations can be referred as below

1. Flutter configurations - pubspec.yaml
2. App configurations - build.gradle & settings.gradle



# FLUTTER PACKAGES/DEPENDENCIES

The packages used in the project are listed below:

- **easy\_pie\_chart v1.0.0** – Dashboard's pie chart
- **http v1.2.2** – API Integration
- **flutter\_barcode\_scanner v2.0.0** – To scan QR code on assets
- **get v4.6.6** – State Management
- **video\_player v2.9.1** – Video playback
- **flick\_video\_player v0.9.0** – Control video playback
- **files\_picker v8.1.2** – To pick media from internal storage
- **chatview v2.2.0** – Messaging UI
- **emoji\_picker\_flutter v2.1.1** – To choose emoji (chatview requisite)

```
dependencies:  
  flutter:  
    sdk: flutter  
  
  cupertino_icons: ^1.0.8  
  easy_pie_chart: ^1.0.0  
  http: ^1.2.2  
  flutter_barcode_scanner: ^2.0.0  
  get: ^4.6.6  
  video_player: ^2.9.1  
  flick_video_player: ^0.9.0  
  files_picker: ^8.1.2  
  chatview: ^2.2.0  
  emoji_picker_flutter: ^2.1.1
```

file: pubspec.yaml

# BUILD.GRADLE FILE

To avoid Kotlin issues, code below has been added in build.gradle file:

```
subprojects {  
    // fix for verifyReleaseResources  
    // =====  
    afterEvaluate { project ->  
        if (project.plugins.hasPlugin("com.android.application") ||  
            project.plugins.hasPlugin("com.android.library")) {  
            project.android {  
                compileSdkVersion 34  
                buildToolsVersion "34.0.0"  
            }  
        }  
        if (project.hasProperty("android")) {  
            project.android {  
                if (namespace == null) {  
                    namespace project.group  
                }  
            }  
        }  
    }  
    // =====  
    project.buildDir = "${rootProject.buildDir}/${project.name}"  
    project.evaluationDependsOn(":app")  
}
```

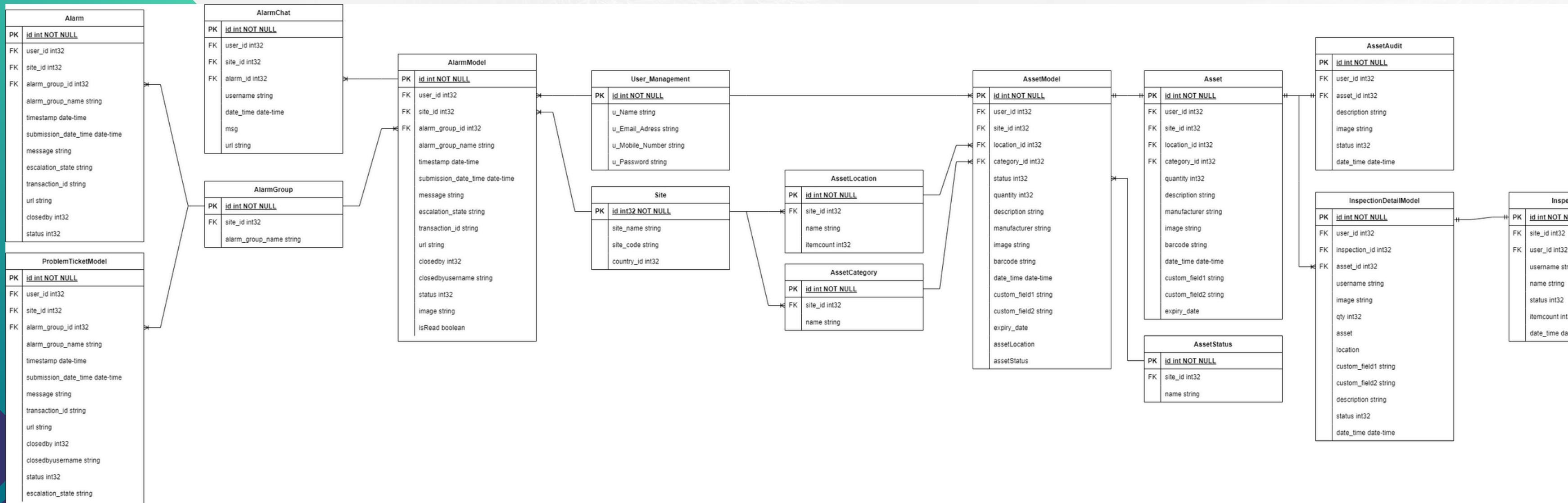
3

## API DOCUMENTATION

# API DOCUMENTATION

API Link: <http://13.126.15.56:180/index.html>

# DATABASE STRUCTURE



# 4

## UI/UX DESIGN

# UI/UX DESIGN

Current Notifier version is based on the UI design linked below.  
This Figma is read-only:

Figma link:

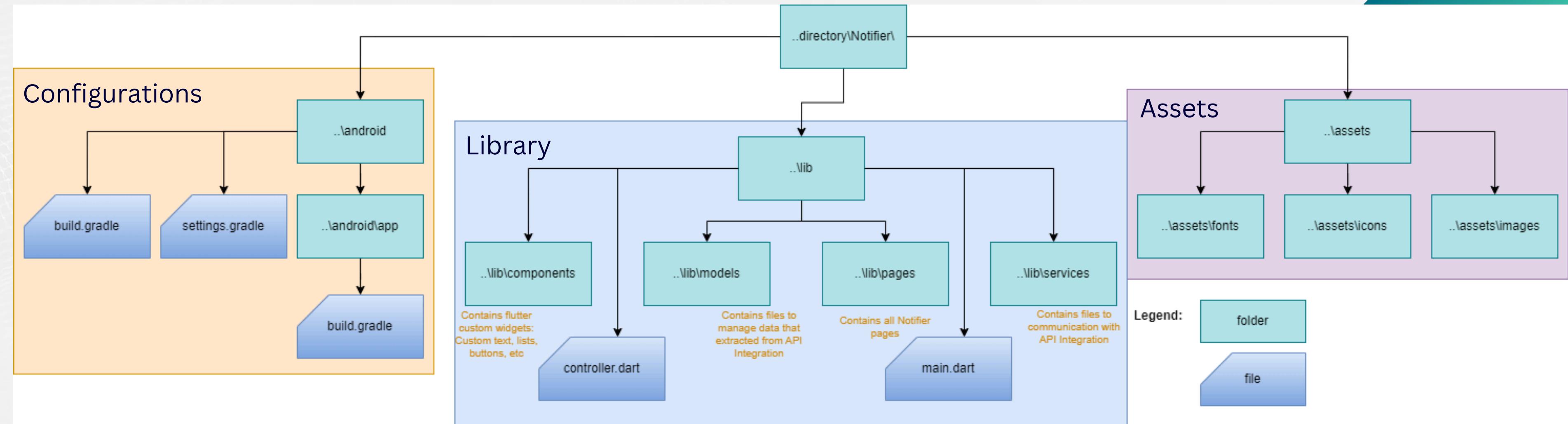
<https://www.figma.com/design/NwQUtoalSXuT5DRPDskKNu/Notifier-Flow-29%2F8?node-id=0-1&node-type=canvas&t=0x7dHOwuKSGnVvxg-0>

However, there will be a new UI design to be developed in the later version by a new UI/UX designer.

5

# CODE DOCUMENTATION

# CODE STRUCTURE



# MAIN.DART

- Turned off debugMode banner
- Set the first page as LoginPage().
- Font theme - Inter.
- Routes

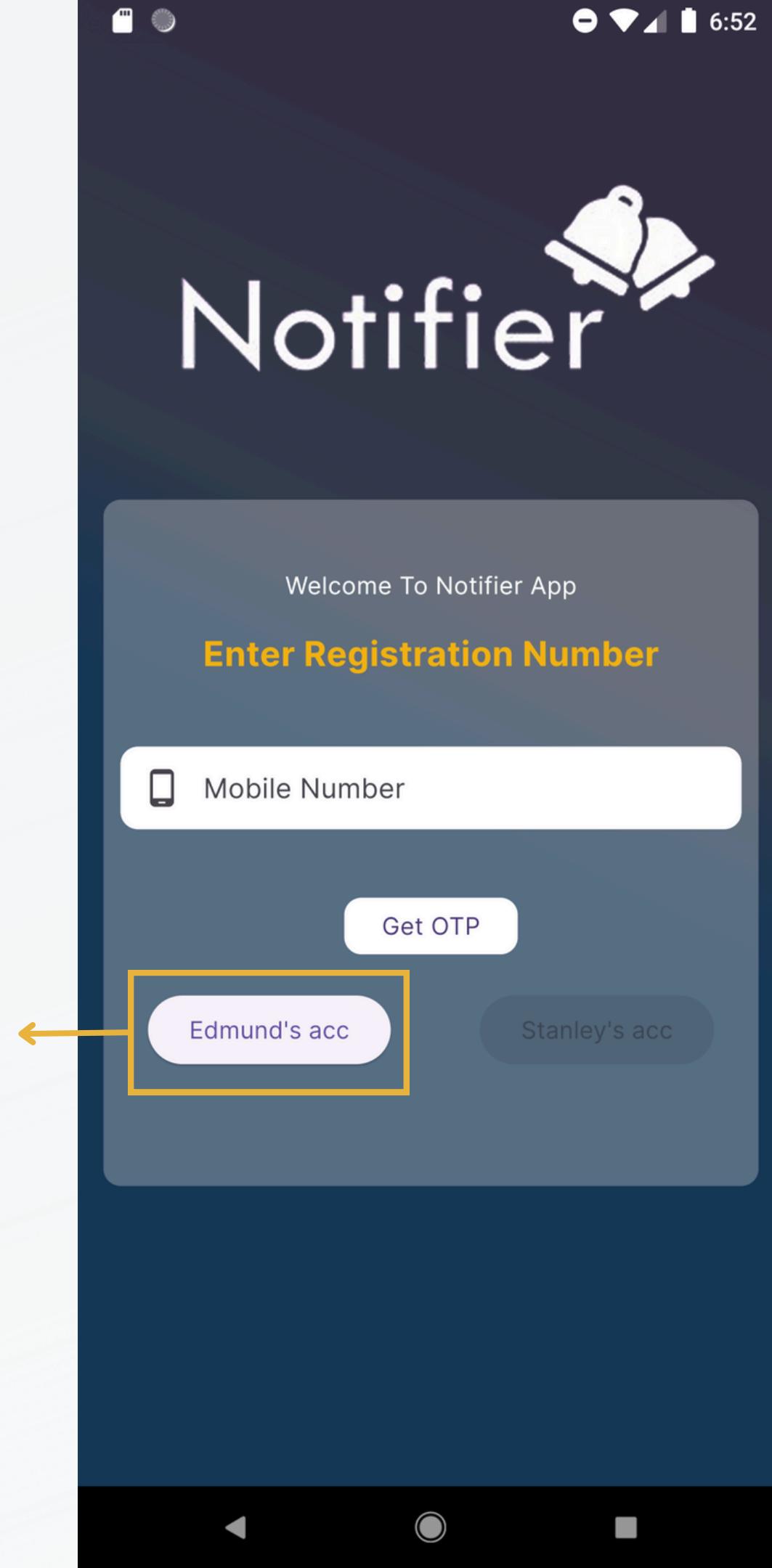
```
Widget build(BuildContext context) {  
  return MaterialApp(  
    debugShowCheckedModeBanner: false,  
    title: 'Notifier App',  
    home: const LoginPage(),  
    theme: ThemeData(fontFamily: 'Inter'),  
    routes: {  
      '/homepage': (context) => const HomePage(),  
      '/loginpage': (context) => const LoginPage(),  
      '/taskdetailspage': (context) => TaskDetailsPage(),  
      '/alarmlistpage': (context) => const AlarmListPage(),  
      '/progresschecklistpage': (context) => const ProgressChecklistPage(),  
      '/dashboardpage': (context) => const DashboardPage(),  
      '/taskchecklistpage': (context) => const TaskCheckListPage(),  
      '/assetlocationpage': (context) => const AssetLocationsPage(),  
      '/assetitemspage': (context) => const AssetItemsPage(),  
      '/edithistorypage': (context) => const EditHistoryPage(),  
      '/assetdetailspage': (context) => const AssetDetailsPage(),  
      '/assetinspectionpage': (context) => const AssetInspectionPage(),  
      '/addinspectiondialog': (context) => const AddInspectionDialog(),  
      '/chathistorypage': (context) => const ChatHistoryPage(),  
      '/chatpage': (context) => const ChatPage(),  
      '/manualalarmpage': (context) => const ManualAlarmPage(),  
      '/taskassignmentpage': (context) => const TaskAssignmentPage(),  
      '/assetoptionpage': (context) => const AssetOptionPage(),  
      '/scanneditemspage': (context) => const ScannedItemsPage(),  
      '/messageoptionpage': (context) => const MessageOptionPage(),  
      '/alarmmedia': (context) => const AlarmMedia(),  
      '/messagepage': (context) => const MessagePage()  
    },  
  ); // MaterialApp
```

# NOTE:

- For pages other than main.dart, explanation using diagrams will be used for better understanding. Please refer comments in code for details

# LOGIN PAGE

lib\pages\login\_page.dart

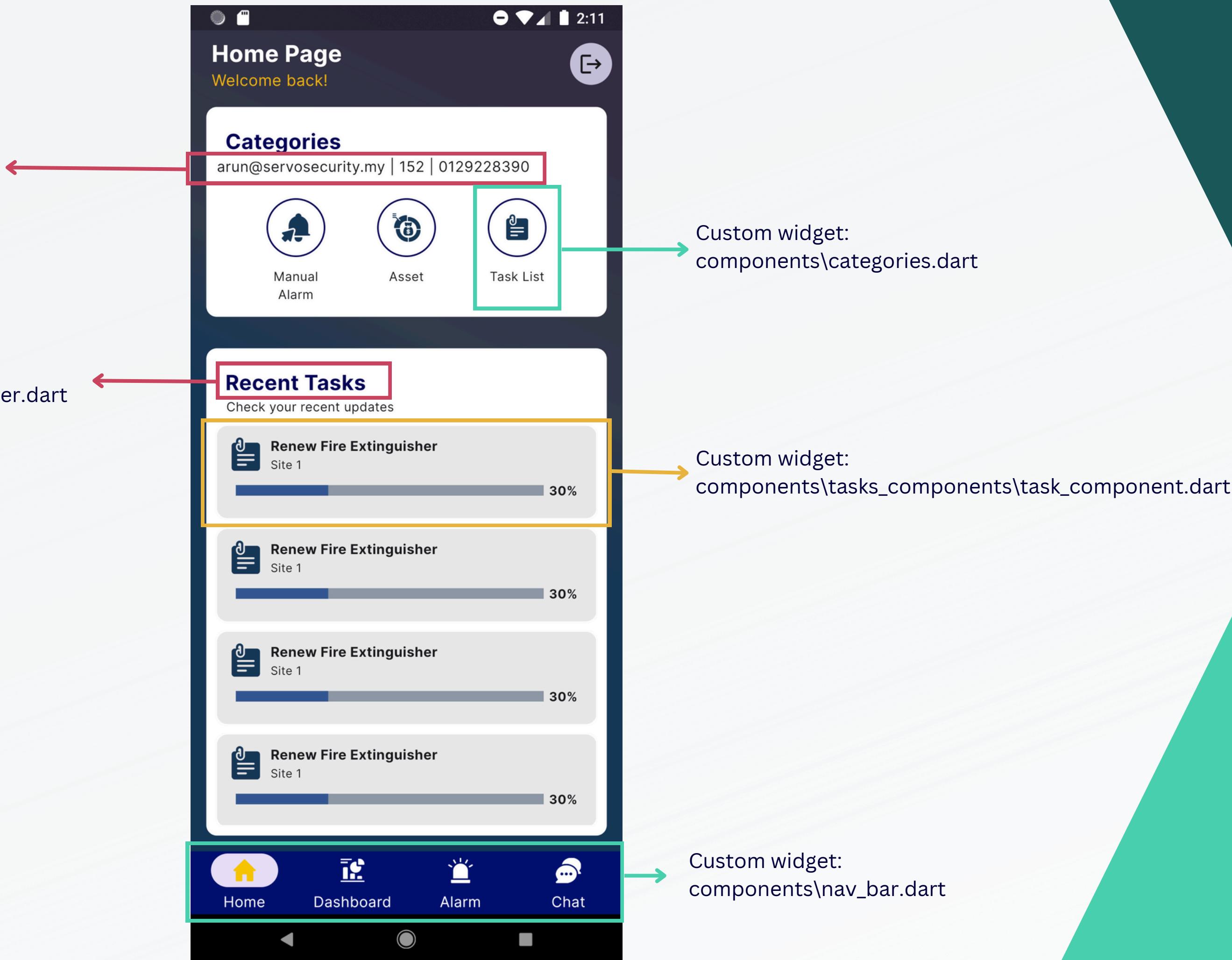


Get into Edmund's account for development purposes

# HOME PAGE

lib\pages\home\_page.dart

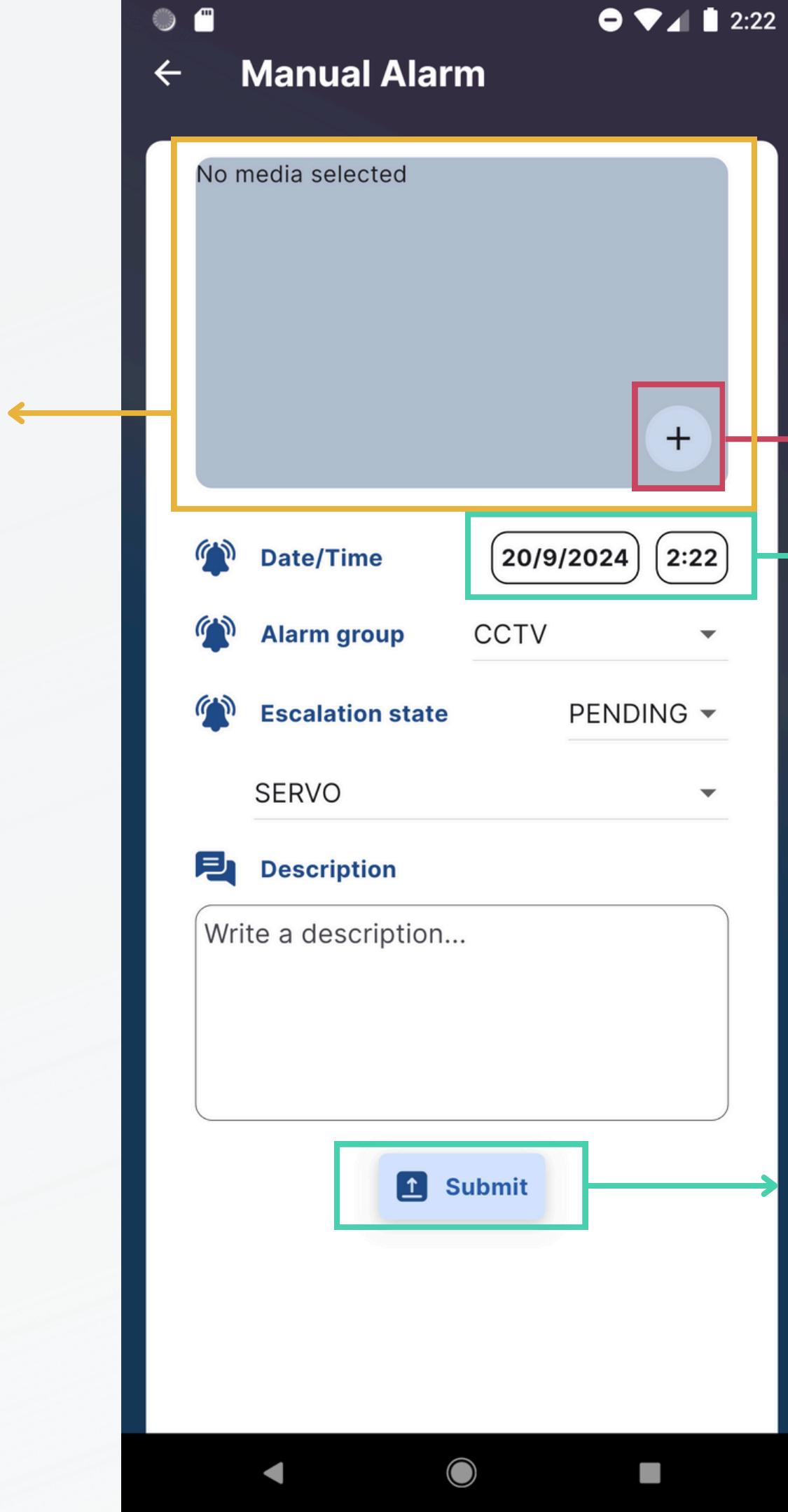
Development purpose: identify current ID used



# MANUAL ALARM

lib\pages\alarm\manual\_alarm\_page.dart

mediaSelected = false: Blank container  
mediaSelected = true: Container filled with selected image



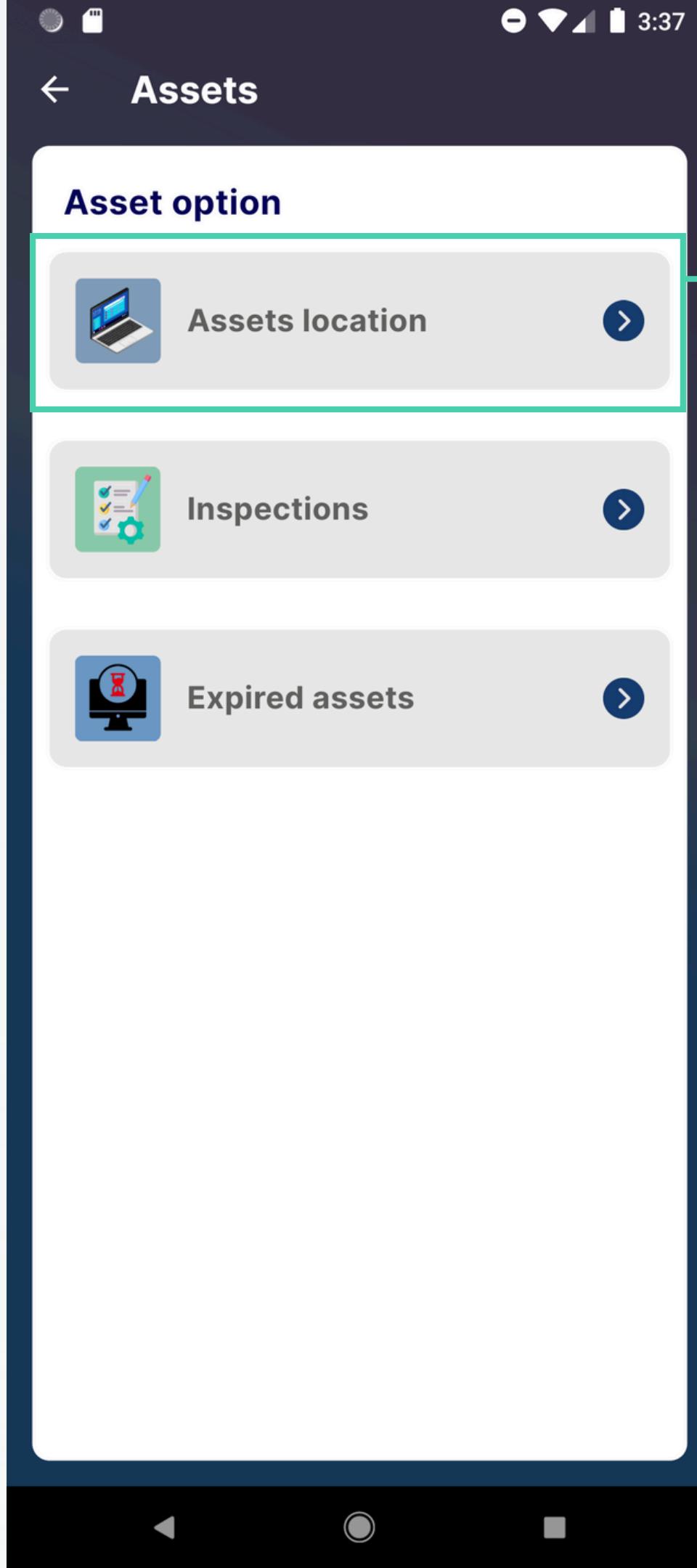
Invokes openDirectoryPicker() to select image or video from phone storage.

Invokes showTimePicker() and showDatePicker()

Invokes  
ManualAlarmService().createManualAlarm()

# ASSETS

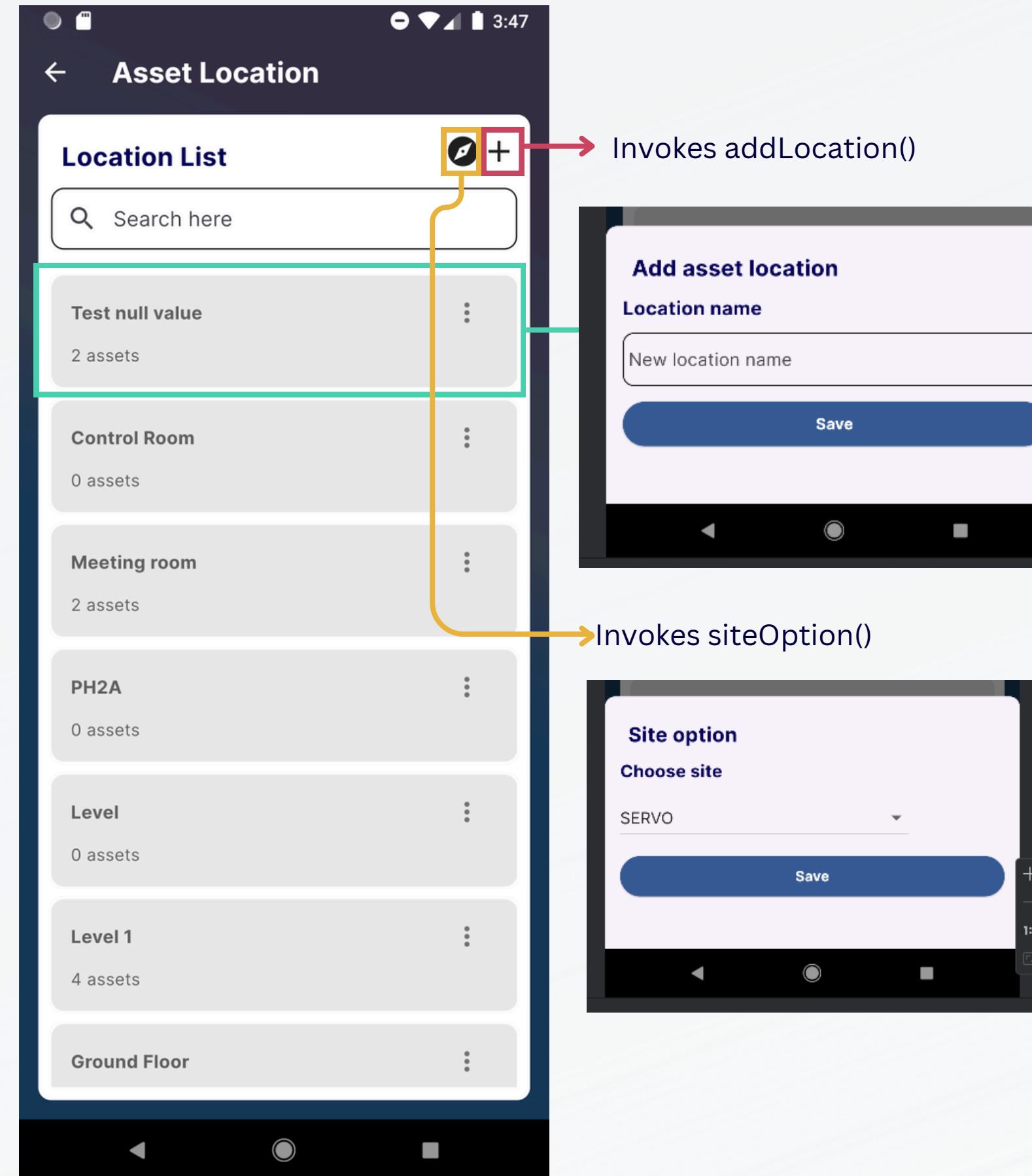
lib/pages/assets/asset\_option\_page.dart



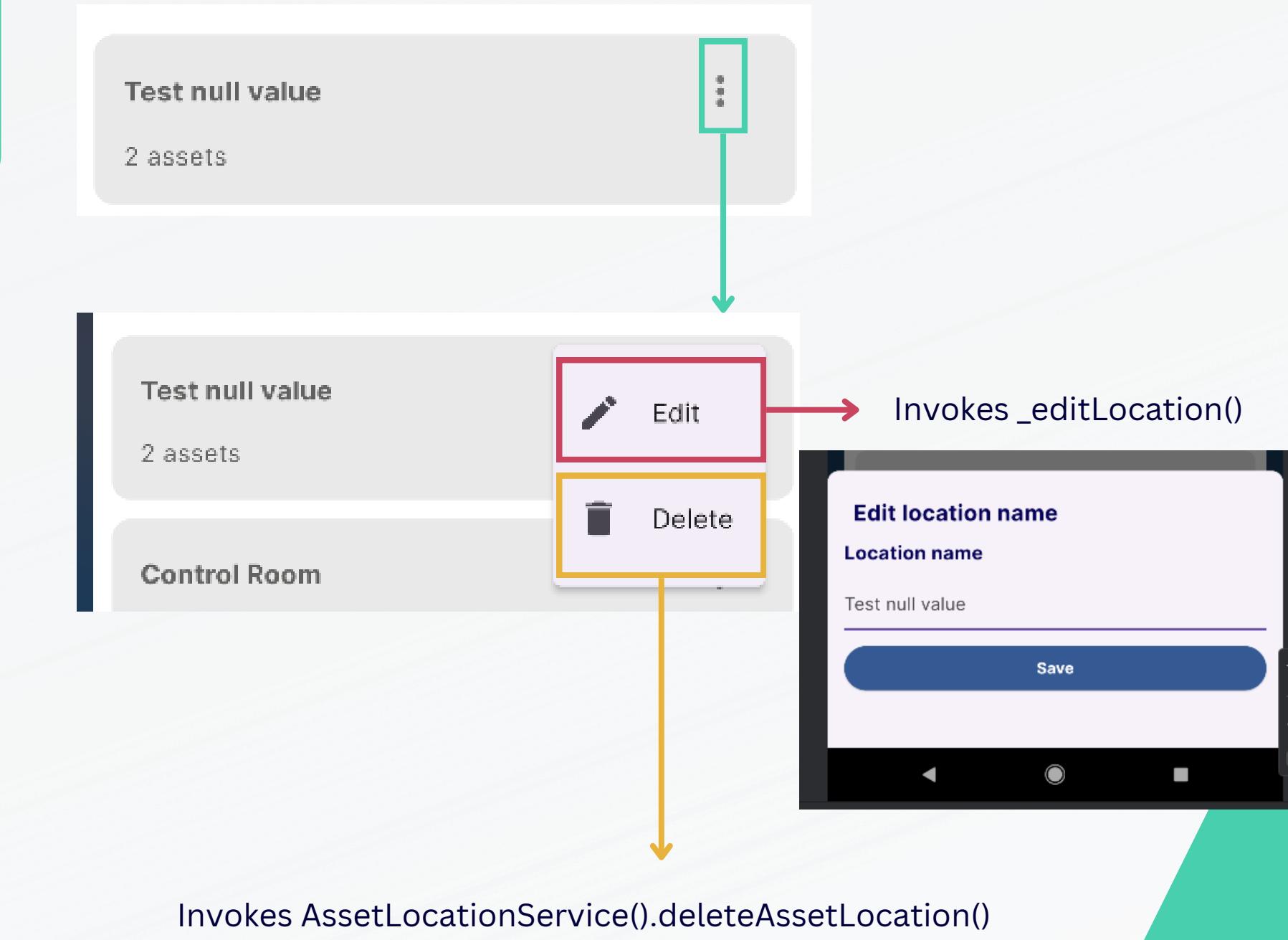
Custom widget:  
components/assets\_components/  
options\_component.dart

# ASSET LOCATION

lib/pages/assets/asset\_locations\_page.dart



Custom widget:  
components/assets\_components/  
asset\_locations\_component.dart

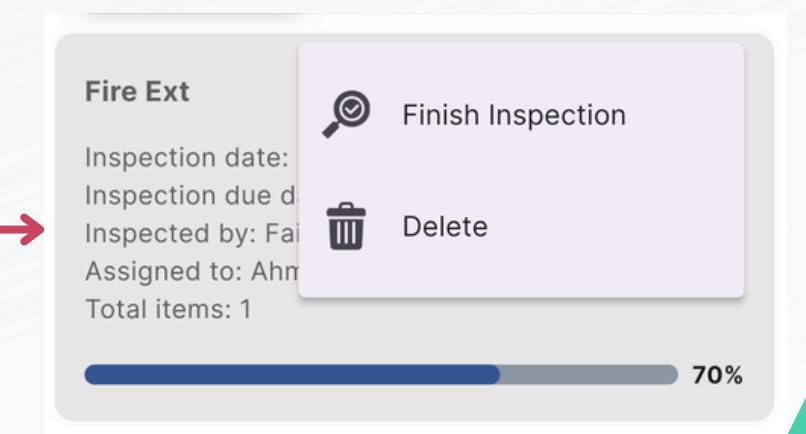
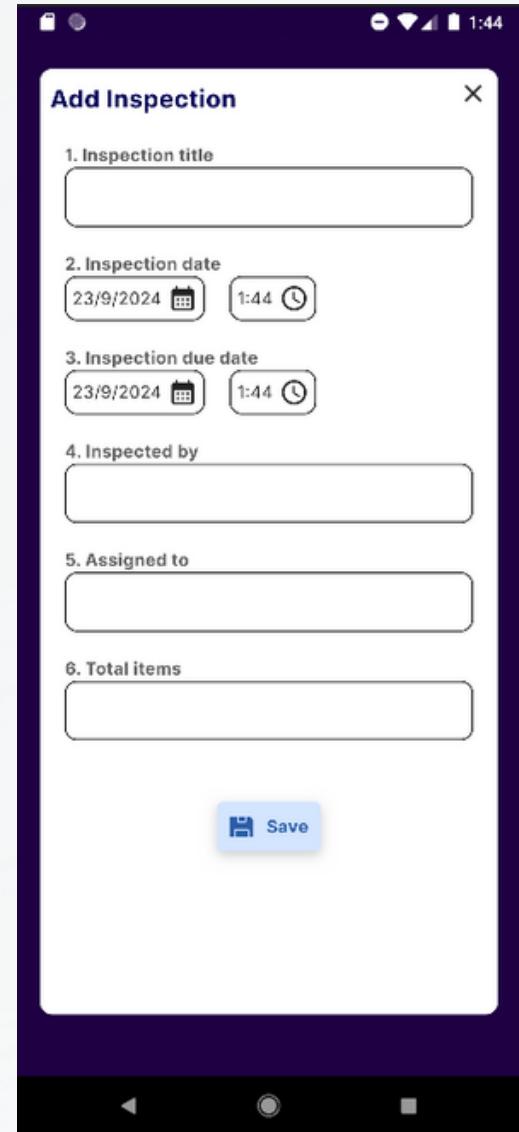
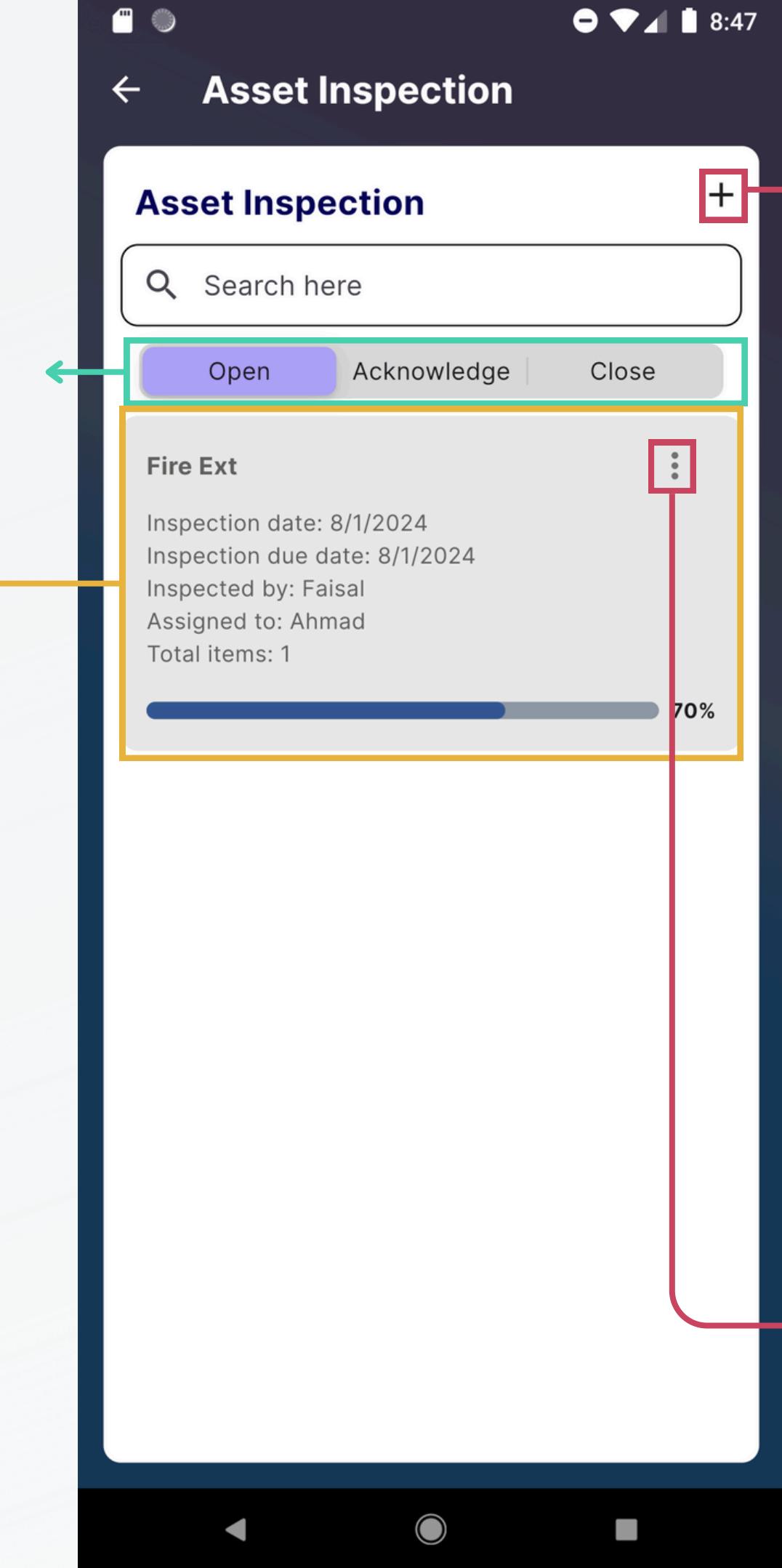
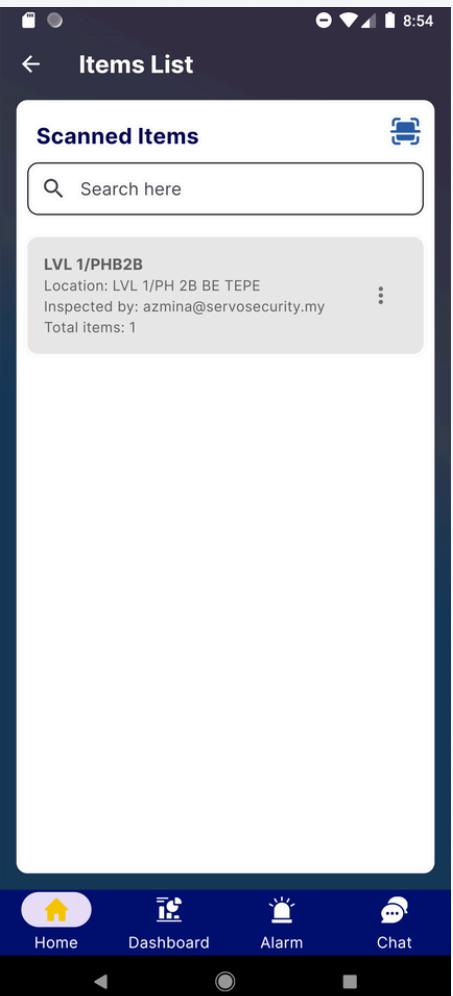


# ASSET INSPECTION

lib/pages/assets/asset\_inspection\_page.dart

Custom widget:  
components/assets\_components/  
asset\_inspection\_component.dart

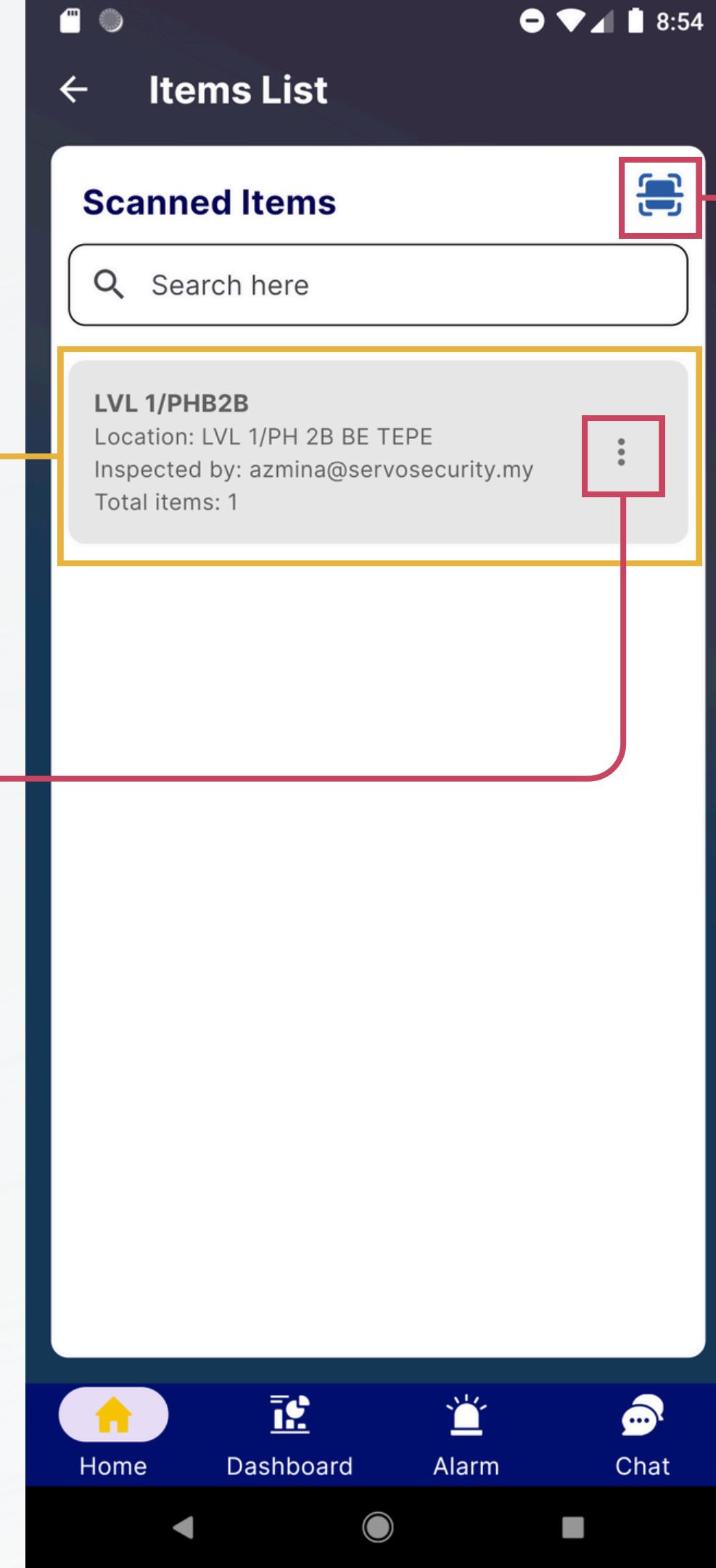
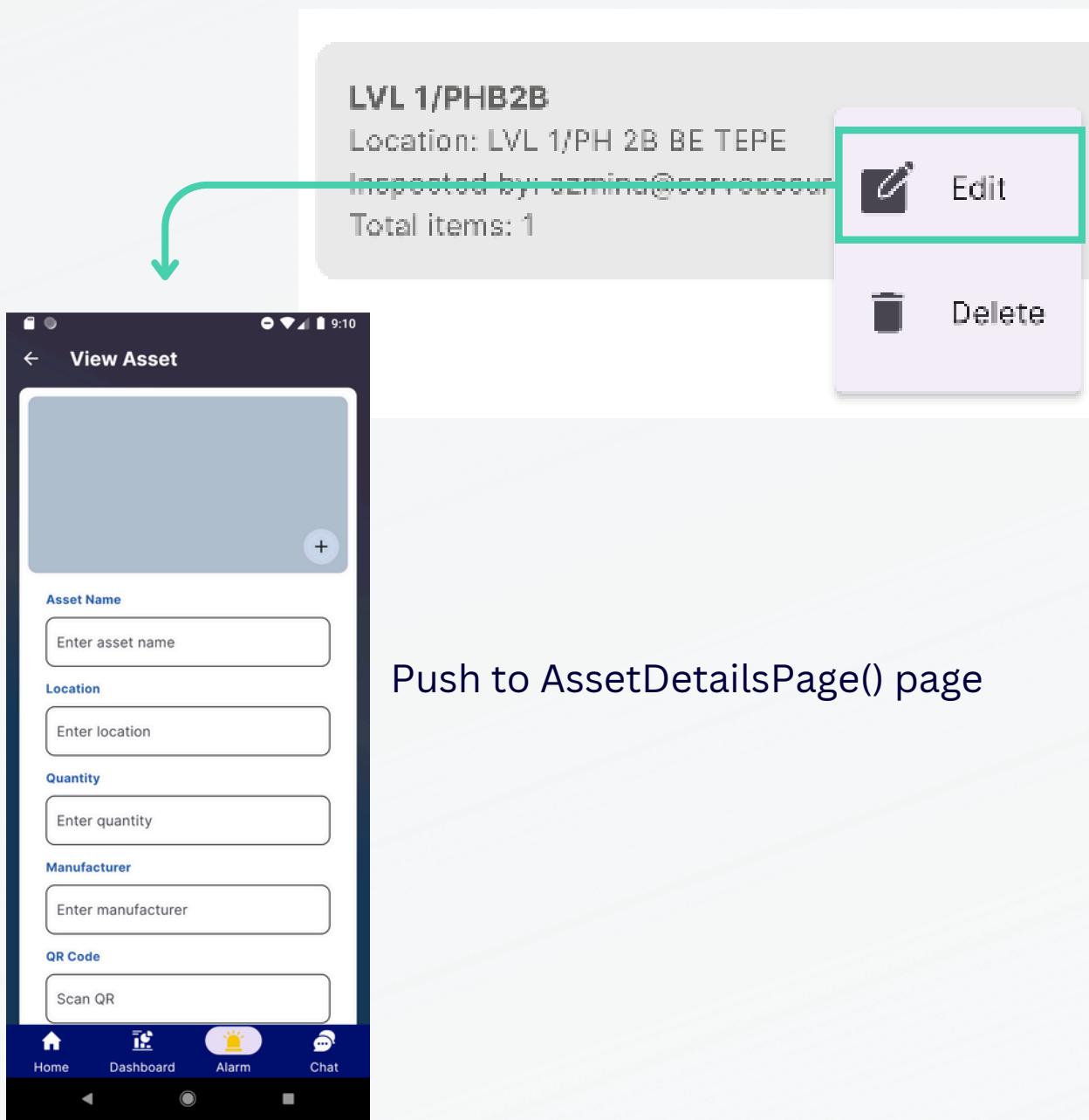
Push to ScannedItemsPage()



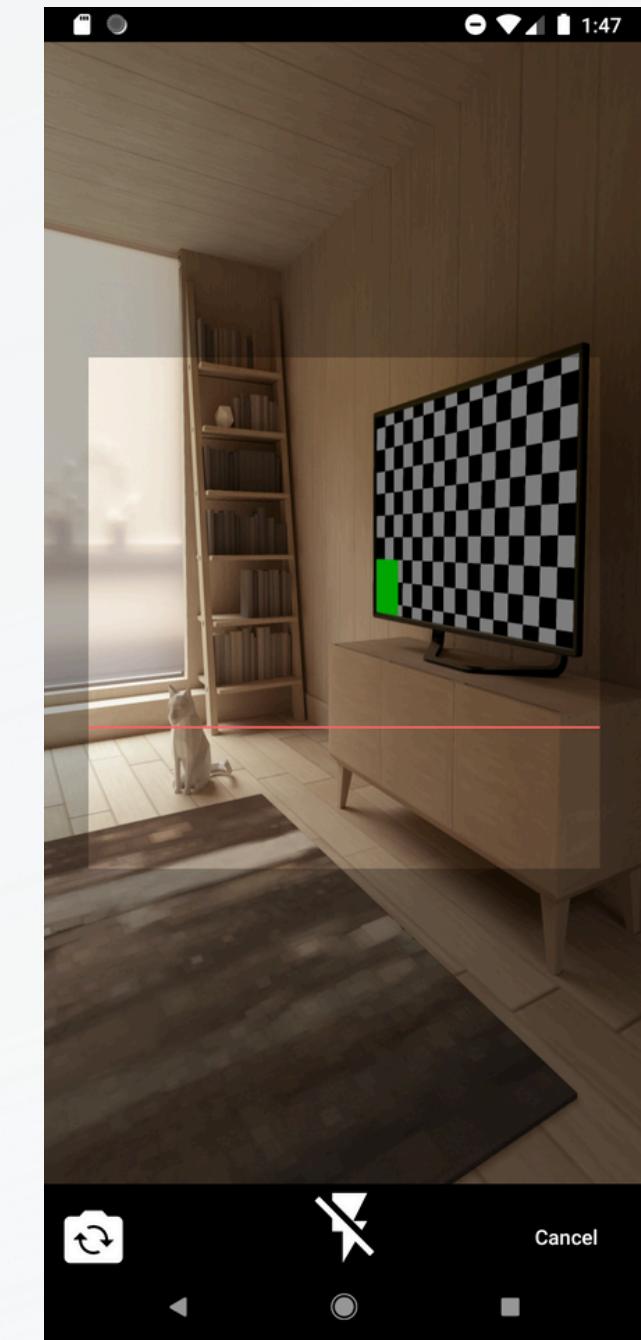
# SCANNED ITEMS

lib/pages/assets/asset\_inspection\_page.dart

Custom widget:  
components/tasks\_components/  
scanned\_items\_component.dart

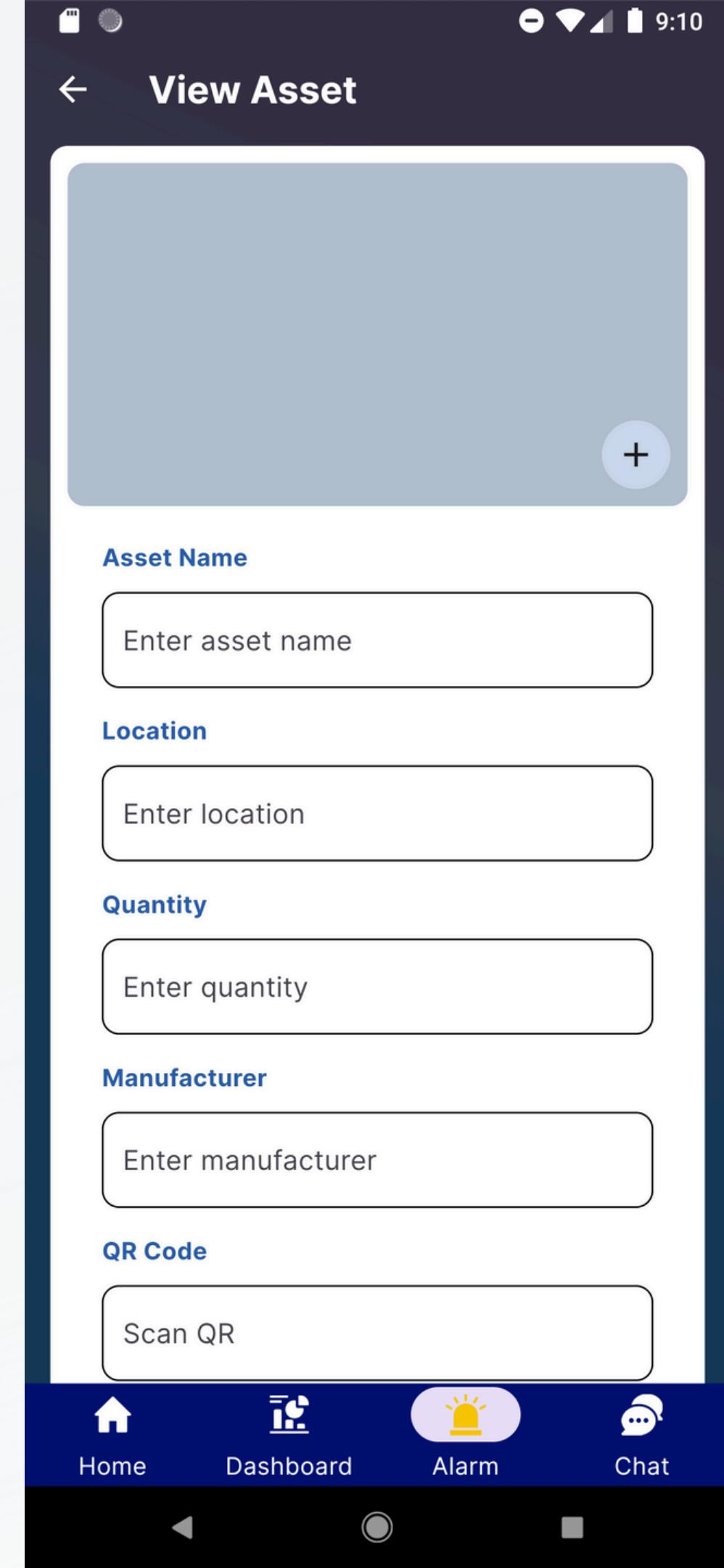


Invoke scanQR()



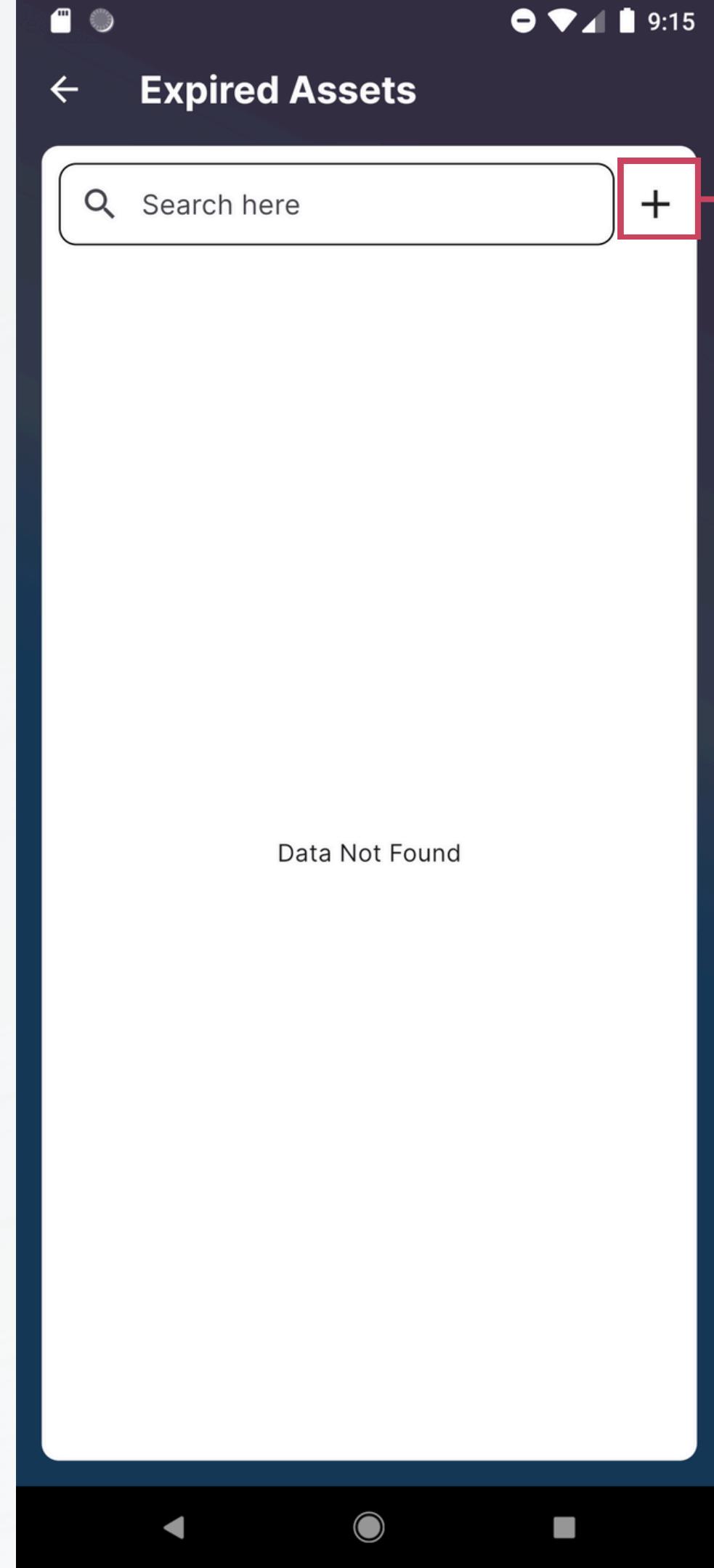
# VIEW ASSET

lib/pages/assets/asset\_details\_page.dart

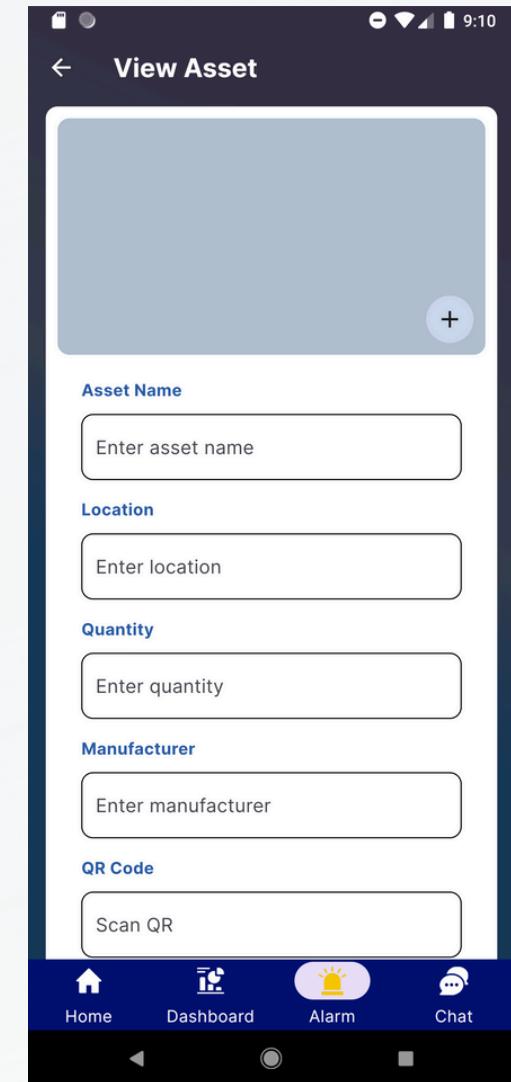


# EXPIRED ASSETS

lib/pages/assets/asset\_items\_page.dart



Push to AssetDetailsPage() page

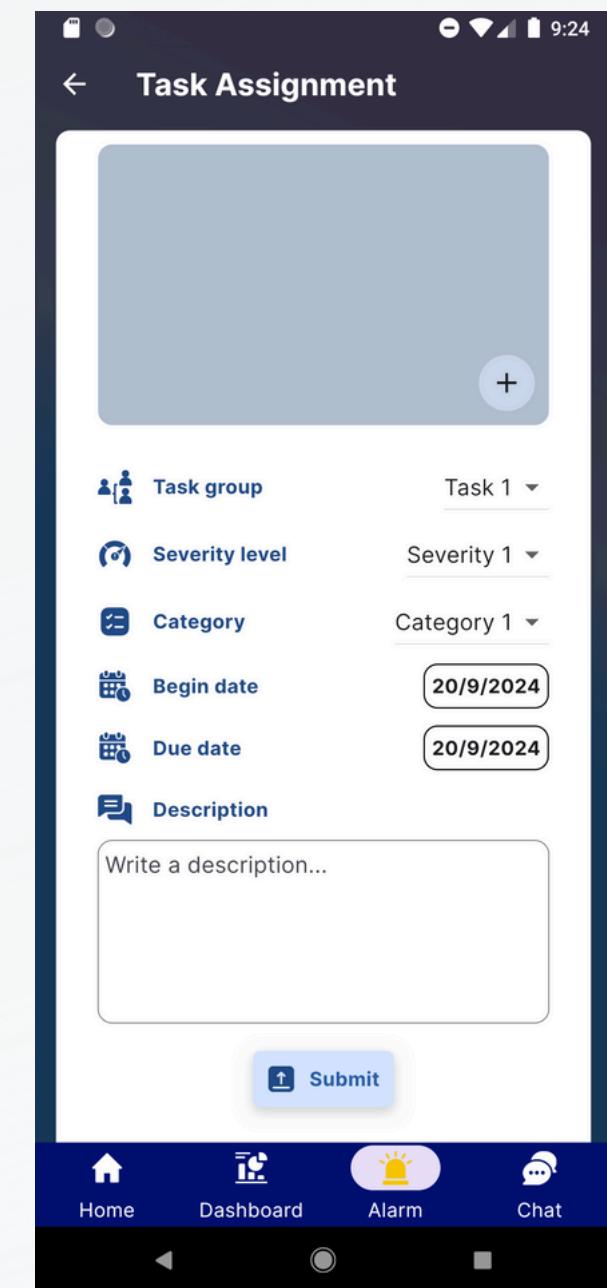
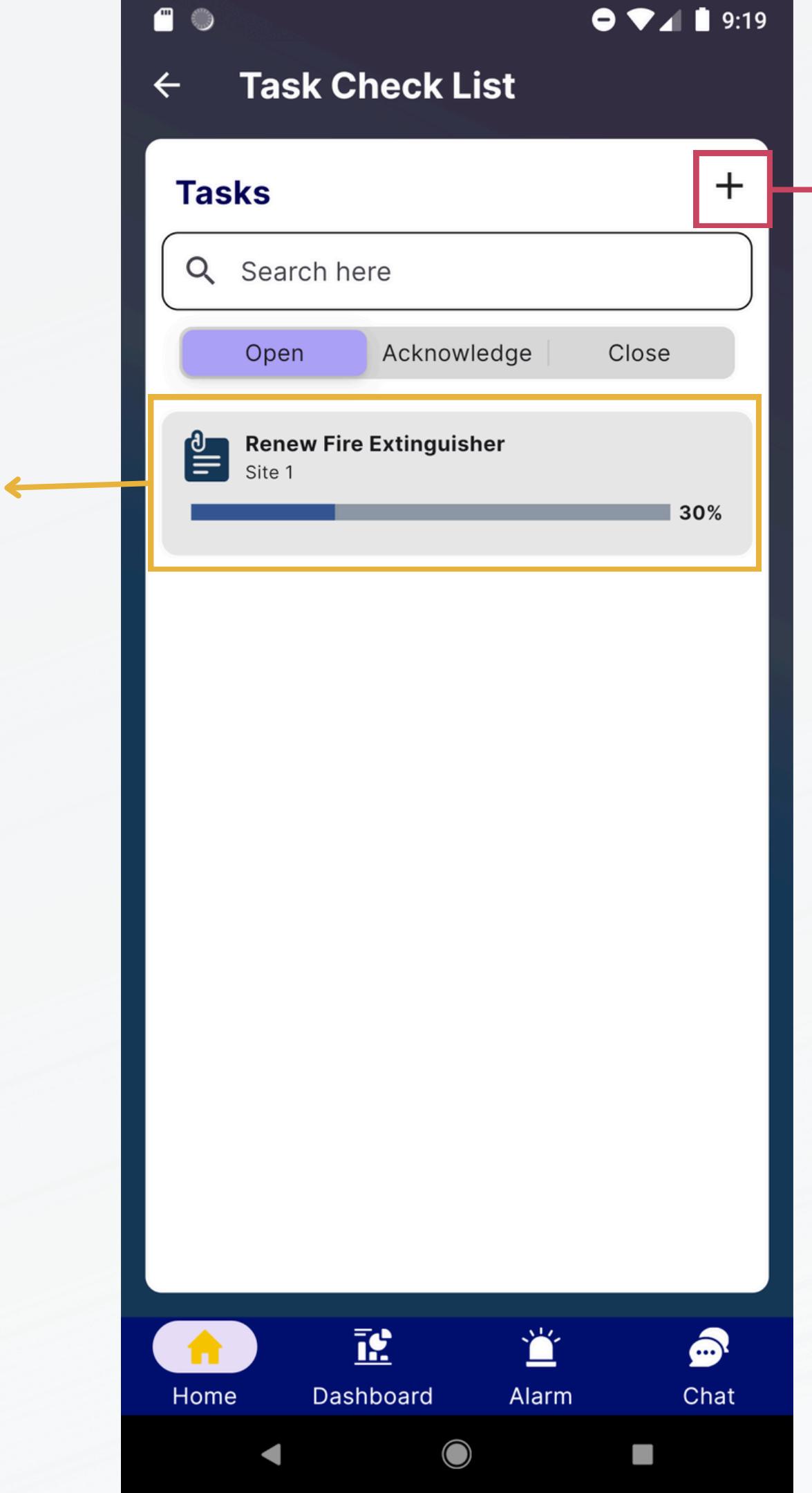
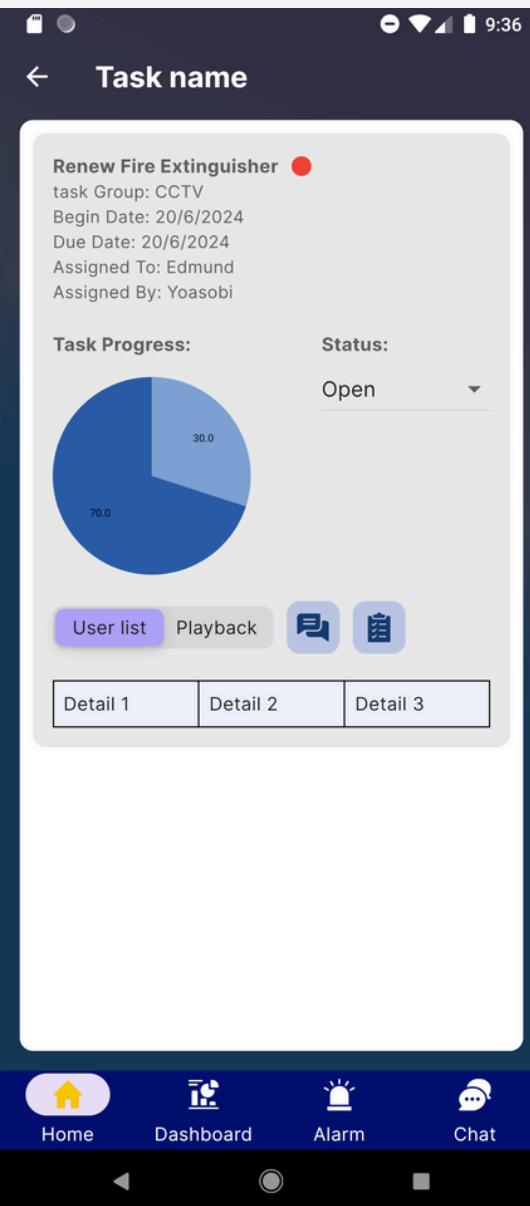


# TASKS

lib/pages/tasks/task\_check\_list\_page.dart

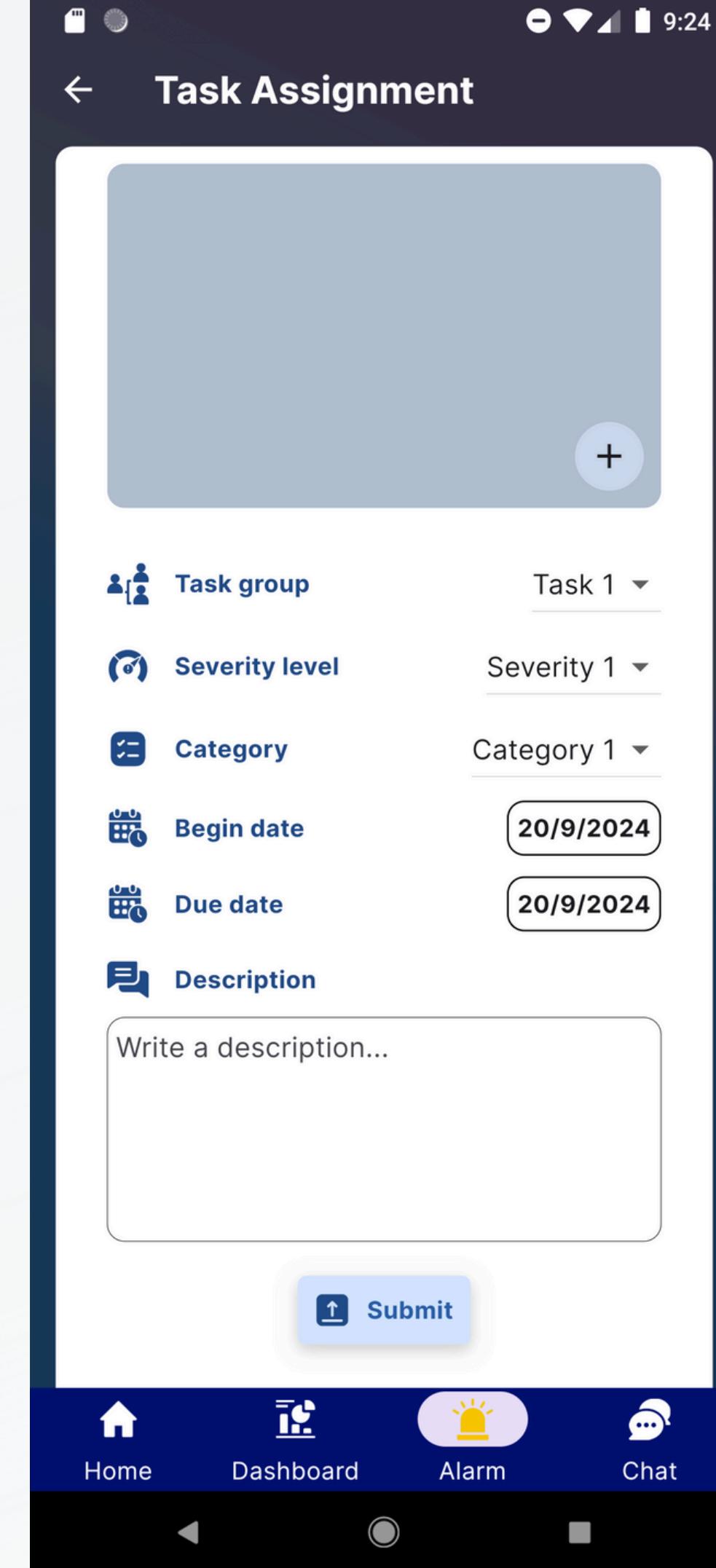
Custom widget:  
components/tasks\_components/  
task\_component.dart

Push to TaskDetailsPage() page



# TASK ASSIGNMENT

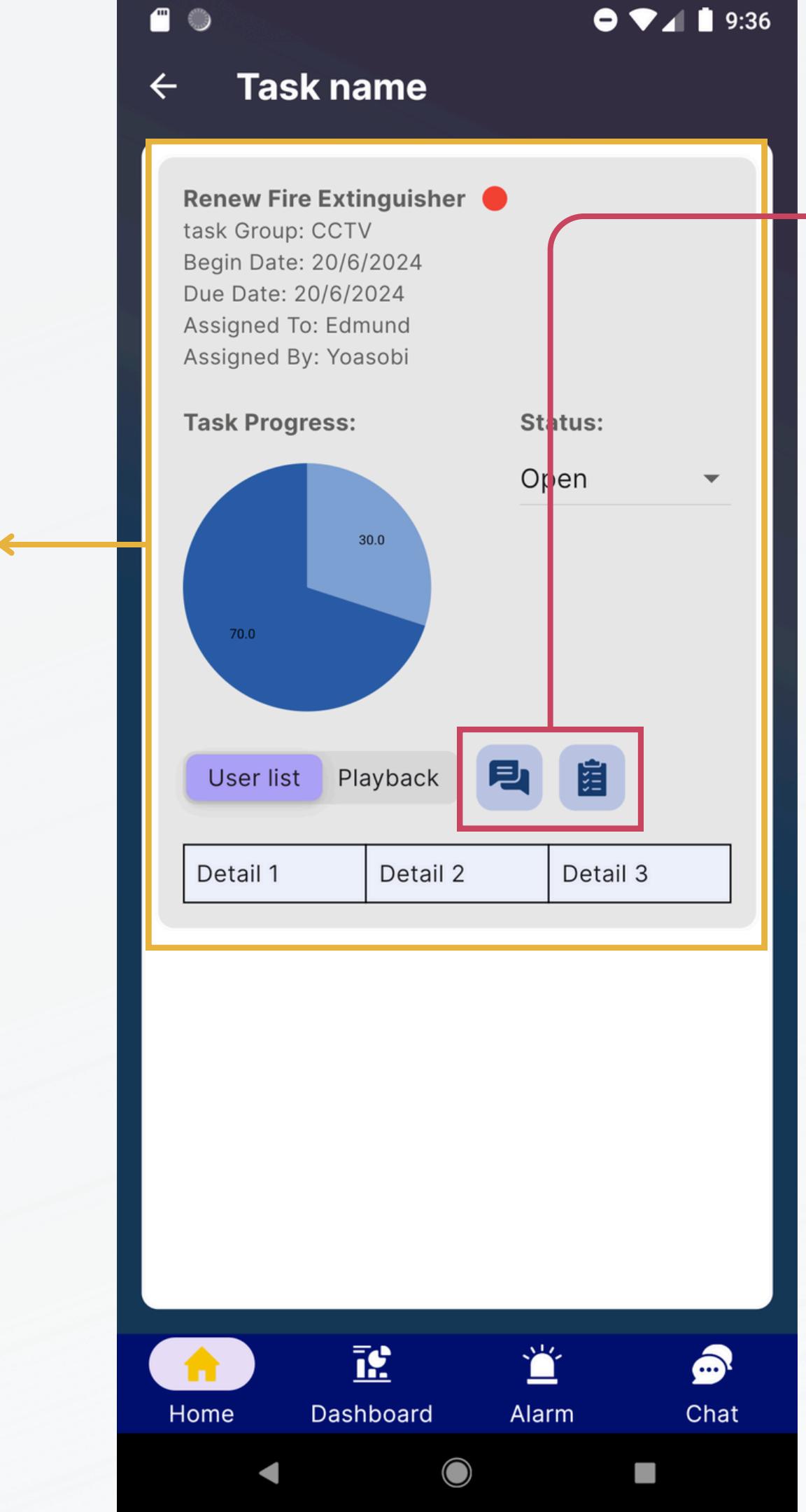
lib/pages/tasks/task\_assignment\_page.dart



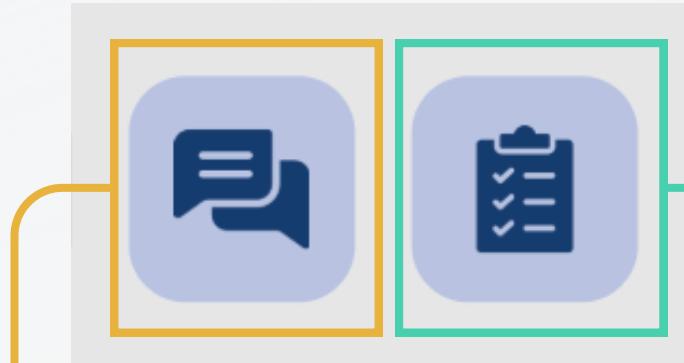
# TASK DETAILS

lib/pages/tasks/task\_details\_page.dart

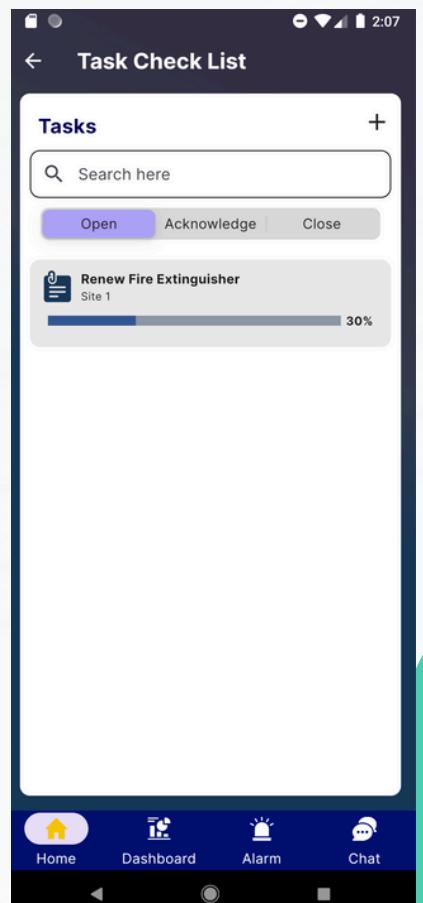
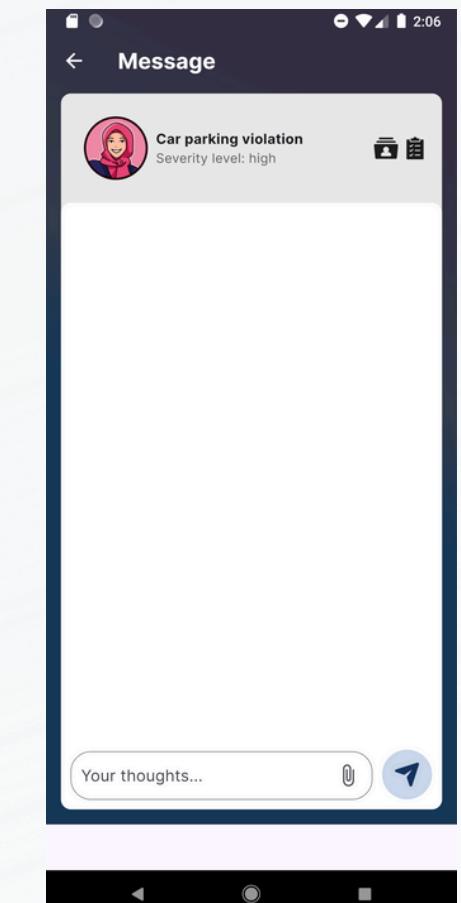
Custom widget:  
components/tasks\_components/  
task\_details.dart



Custom widget:  
components/tasks\_components/  
task\_details\_button.dart



Navigate to ChatPage()

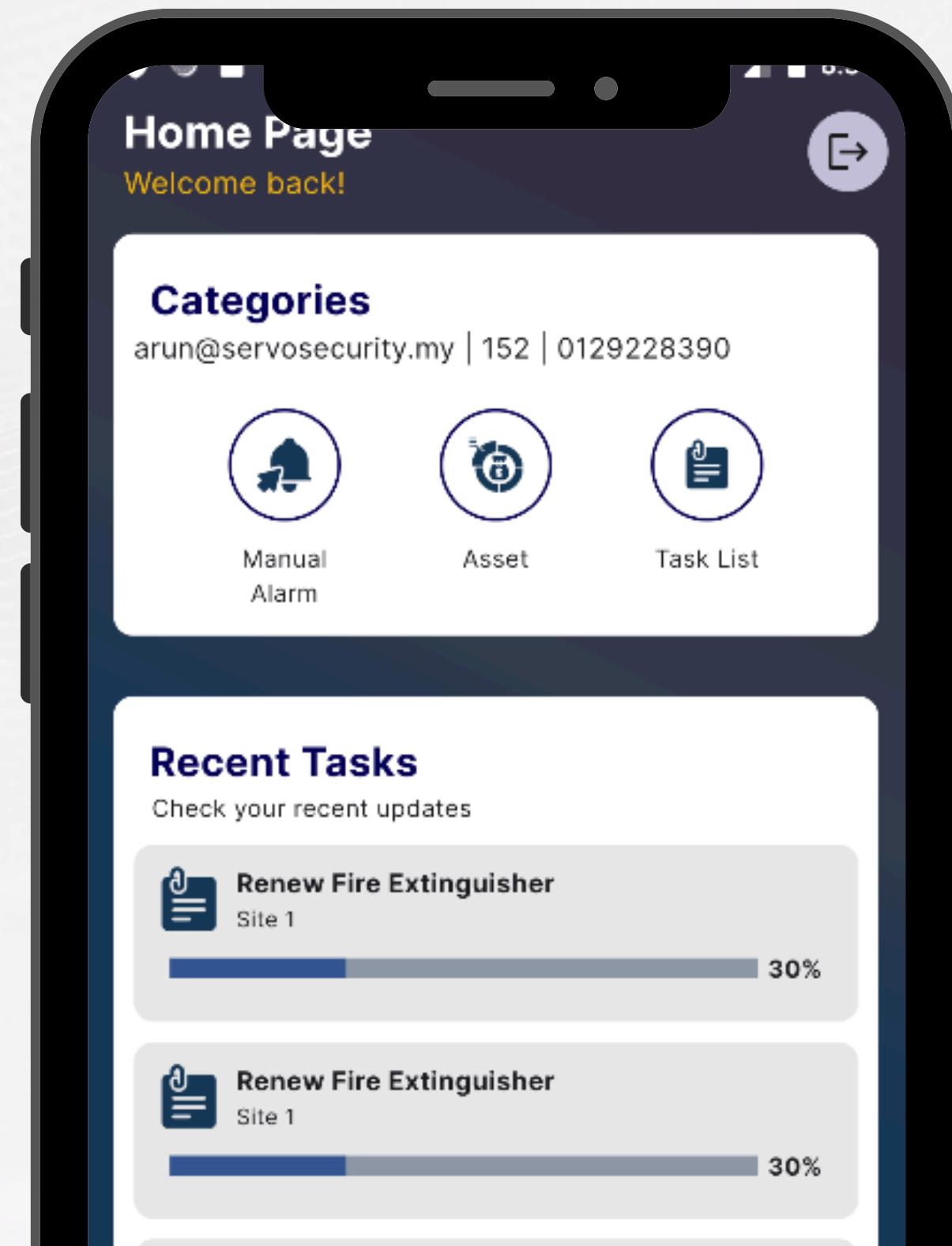


Navigate to  
TaskChecklistPage()

6

## KNOWN ISSUES AND BUGS

# KNOWN ISSUES



- 1
- 2
- 3
- 4
- 5

## Asset > Inspection

There are no inspections data in the database. Development cannot proceed without data to work with.

## Asset > Expired Assets

There are no assets data in the database. Development cannot proceed without data to work with.

## Tasks

New feature. No information and UI design.

## Chat

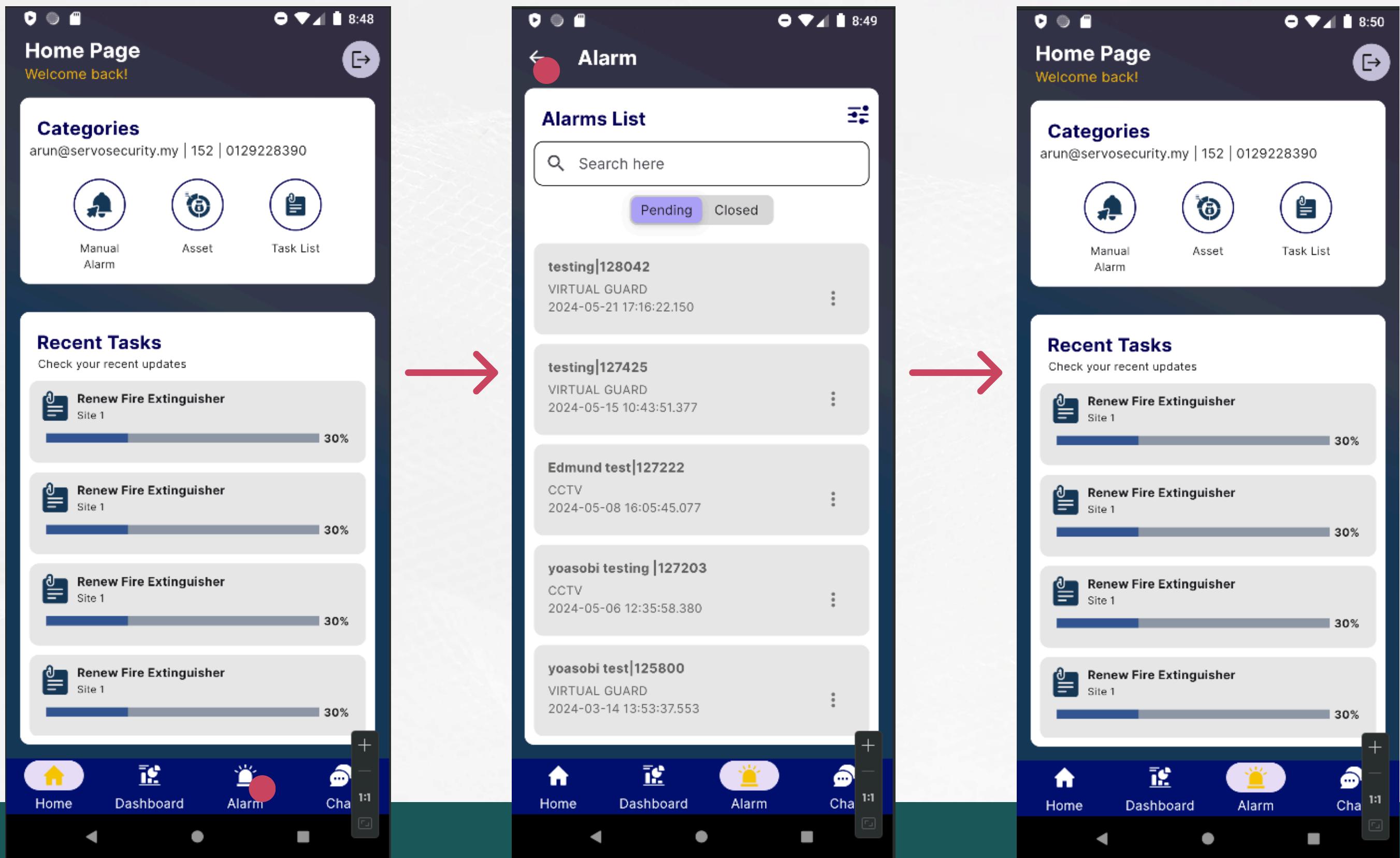
To use a pub.dev packages, “ChatView”. Currently having issues in converting JSON to List<Message> while utilizing Flutter Stream. Refer [ChatView documentation](#)

## Manual Alarm

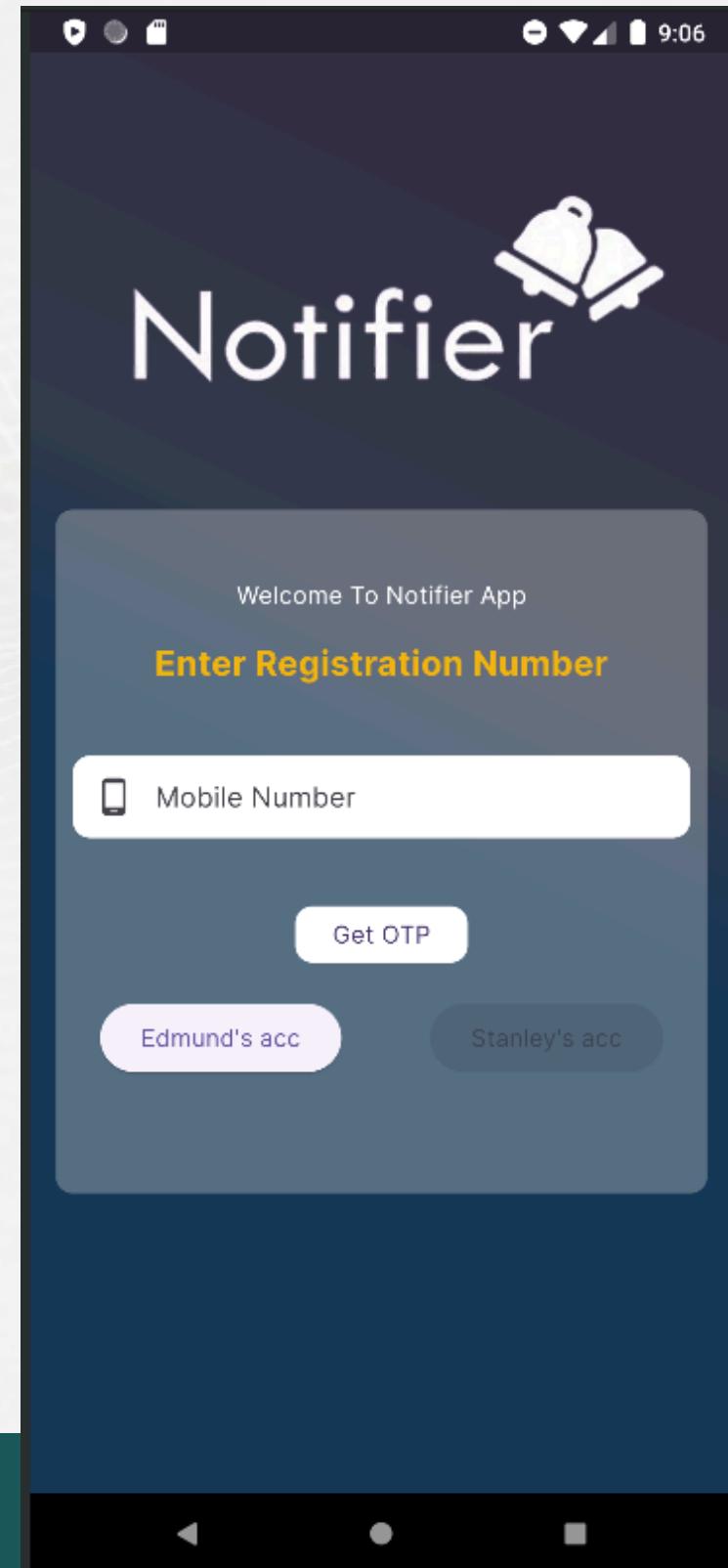
Media can be uploaded, but unsure if it reaches database or not because unable to access database. Not

# KNOWN BUG NAVBAR

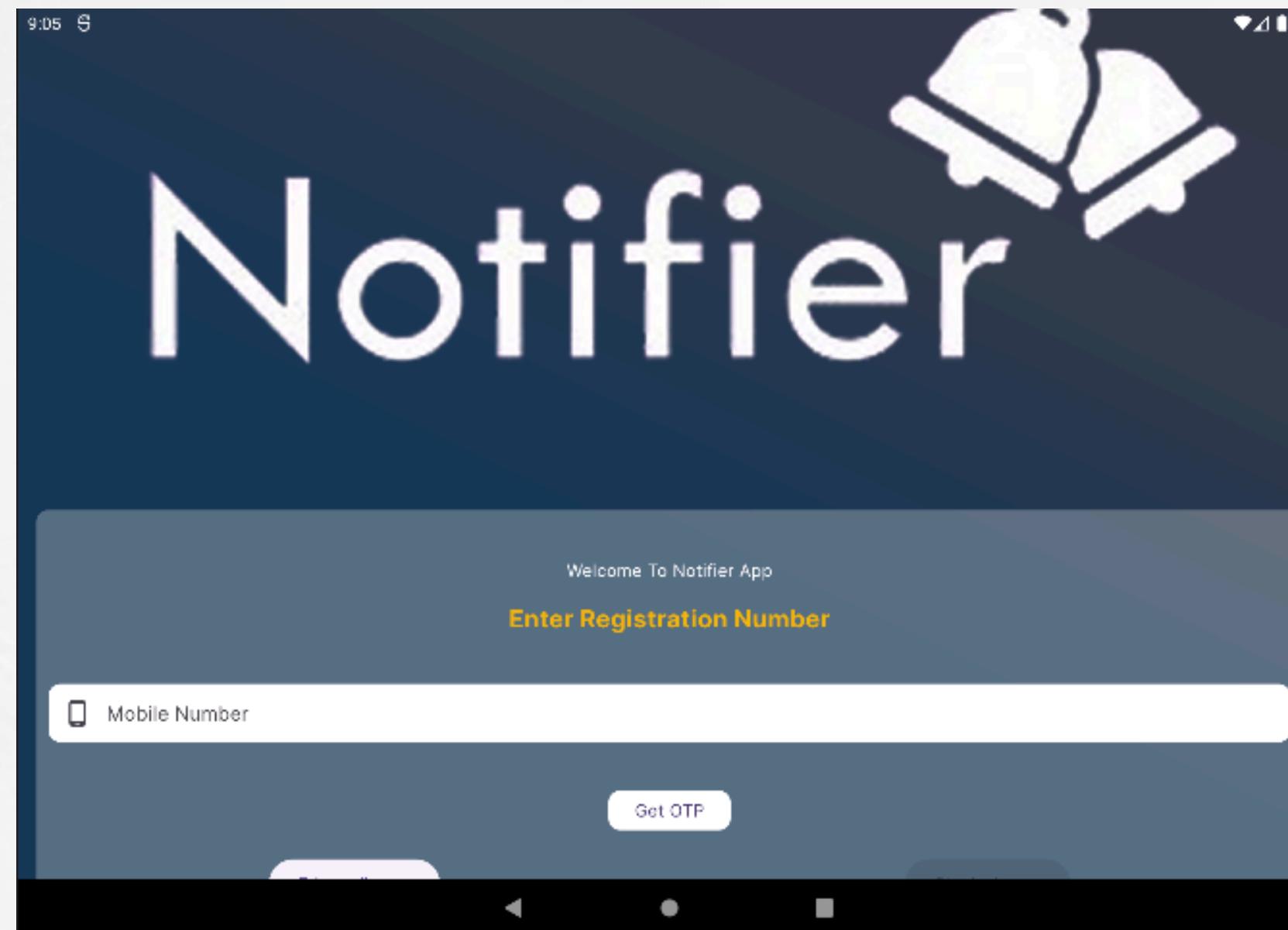
On the Home page, if navigate to any of the other three pages in the NavBar and then click back, the screen returns to Home, but the NavBar still shows the previously selected page instead of the Home page.



# KNOWN BUG LAYOUT



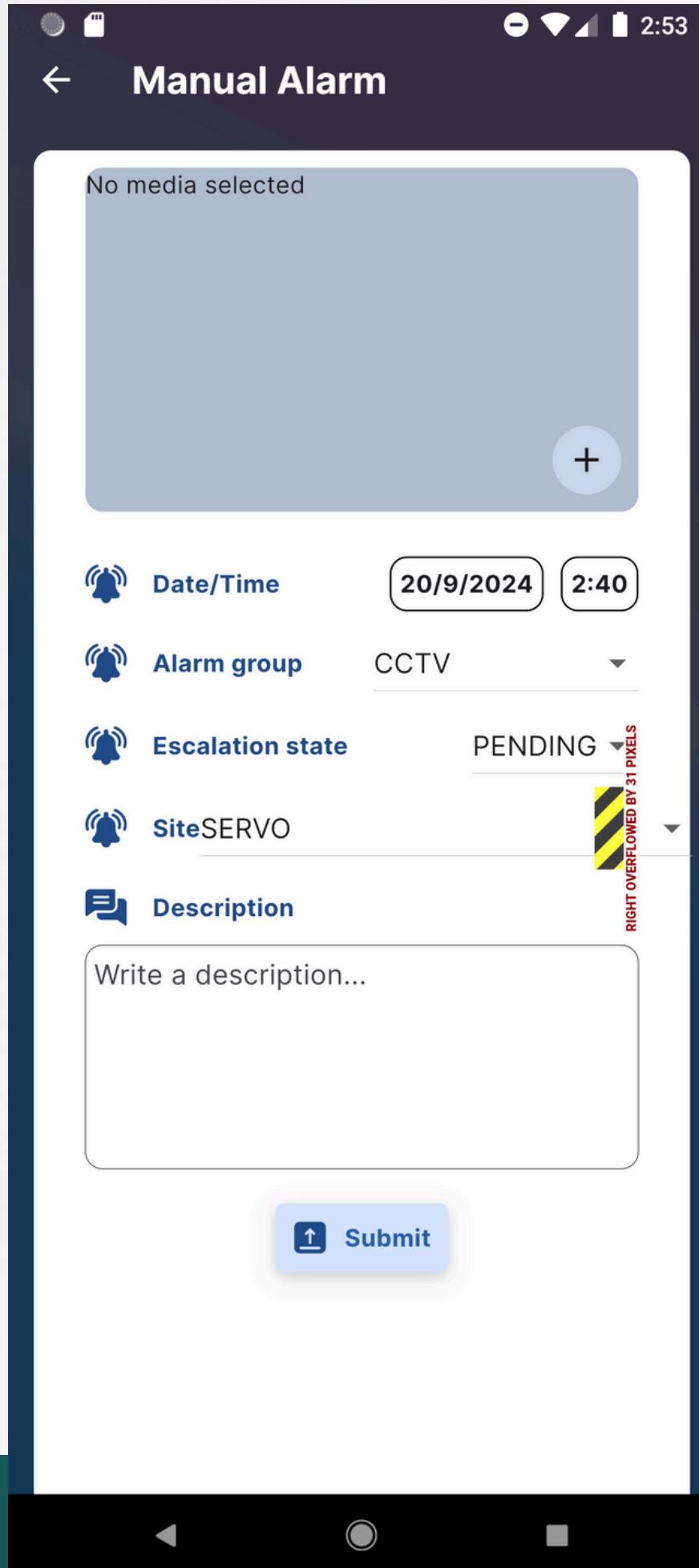
**Mobile phone**  
API 27  
1080x2400px  
412x915dp



**Tablet**  
API 31  
2152x1536px  
1076x768dp

The layout displays correctly on a specific screen size, but does not look well on other screen sizes. For example:

# KNOWN BUG WHITE SPACE



For the Site Dropdown, the white space for “SERVO” in the database text takes many spaces, it breaks the screen width limit



ELEWARE TECHNOLOGIES

SEPTEMBER 2024

**END**