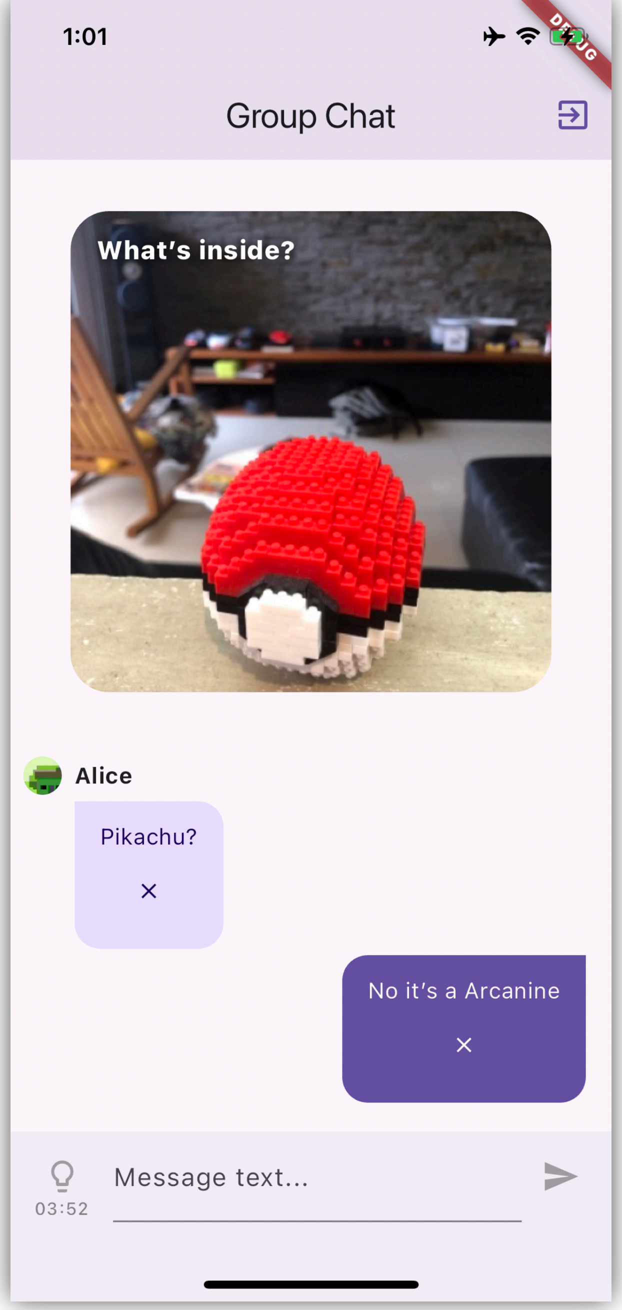
Platform Channels & Home Widgets

Shan-Hung Wu

CS, NTHU

Discovery & Chat

- Mediator can add a "discovery" message
  - Contains a photo taken via ***platform channel***
- Data pushed to all devices in topic: "discoveries"
- Each device updates ***home widgets*** locally based on pushed data
  - No web support!
  - Android and iOS only



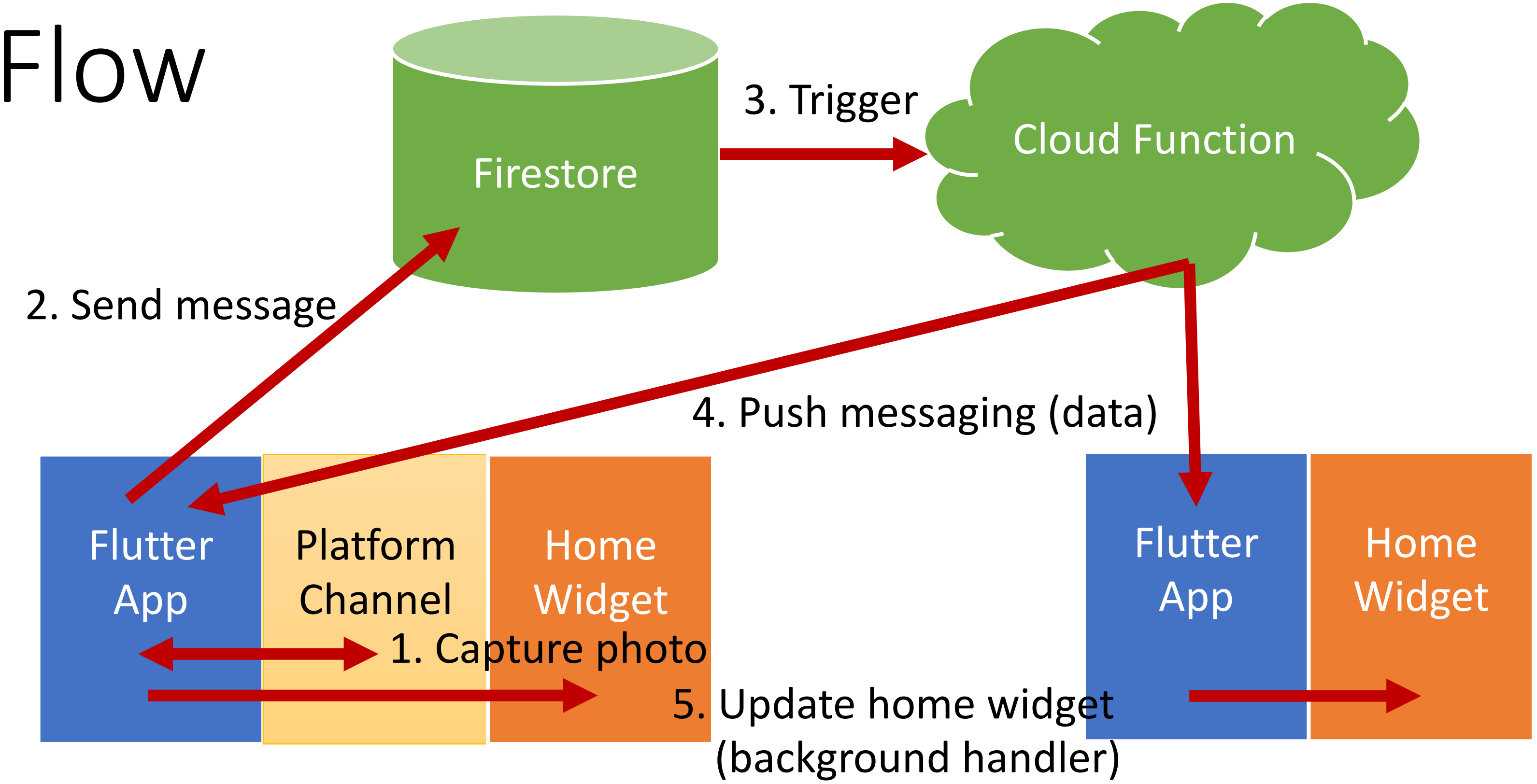
What's inside?

Alice

Pikachu?

No it's a Arcanine

Flow



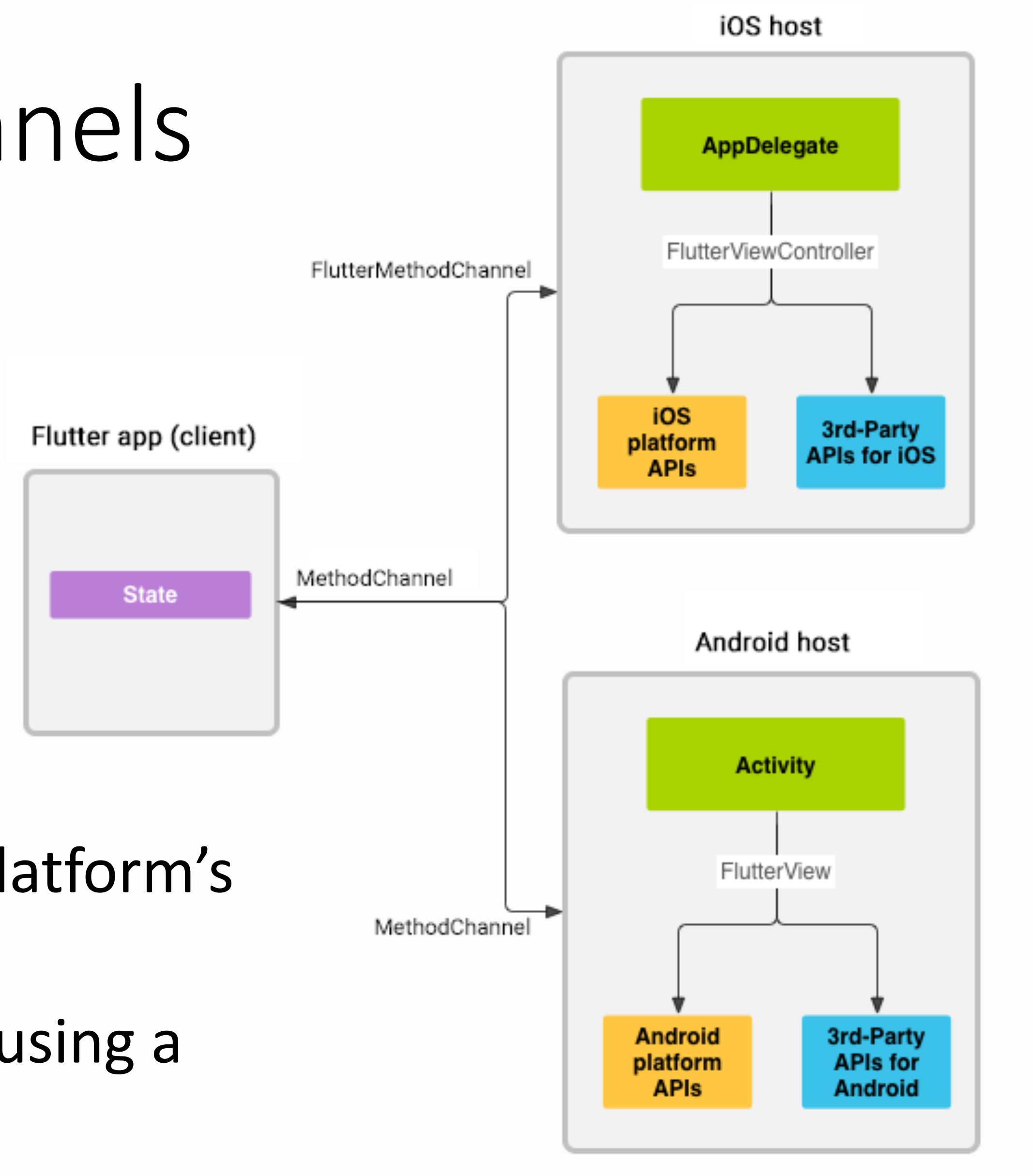
- Why updating home widgets in background push message handler?
  - Users may not have their apps opened when mediator sends a discovery

Push Message Types & Handling

App \ Message	Notification	Data
Foreground	Not shown by OS; call onMessage()	onMessage()
Background	Shown by OS	onBackgroundMessage()
Terminated	Shown by OS	No handler

- See “lib/services/push_messaging.dart”

Platform Channels



- Must be invoked on platform’s main (UI) thread, or
- Background thread if using a Task Queue

Interface

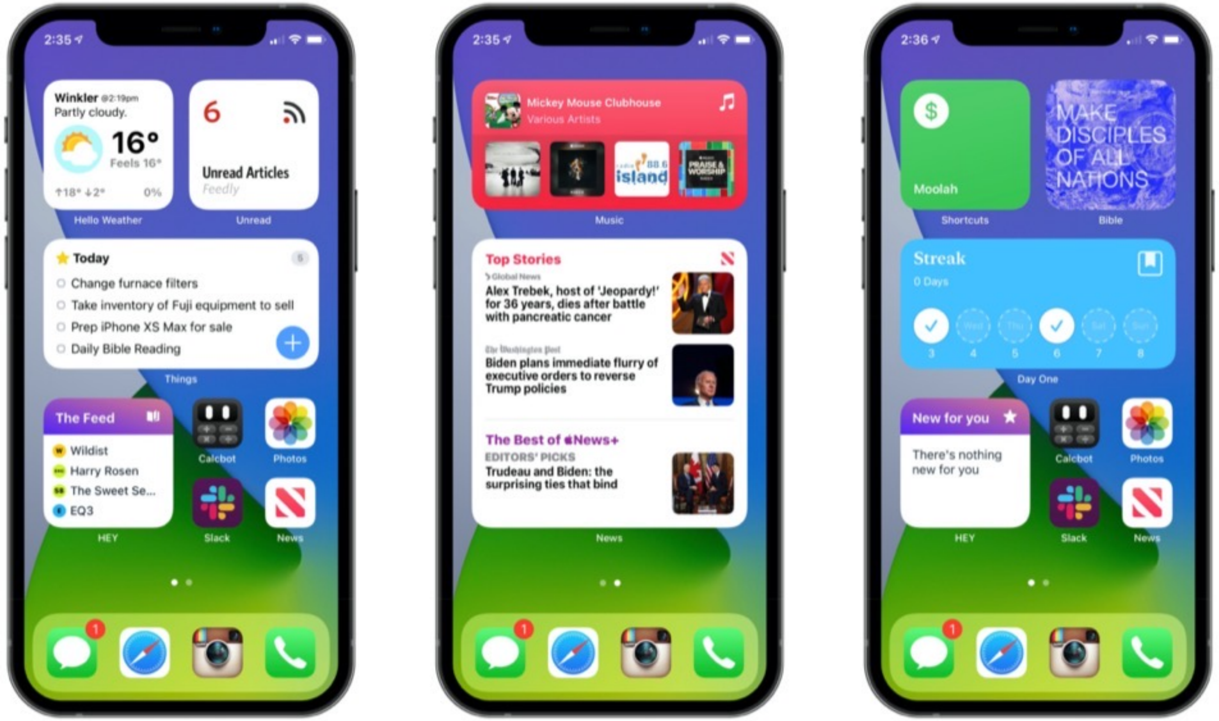
```
// in Flutter
import 'package:flutter/services.dart';
...
const channel = MethodChannel('com.app/battery');
final result = await channel.invokeMethod('getBatteryLevel');

// in Android
import io.flutter.plugin.common.MethodChannel

class MainActivity: FlutterActivity() {
  override fun configureFlutterEngine(flutterEngine: engine) {
    MethodChannel(engine.dartExecutor.binaryMessenger,
        "com.app/battery").setMethodCallHandler { call, result ->
      if (call.method == "getBatteryLevel") {
        ...
      }
      ...
    }
  }
}
```

- See iOS example in project

Home Widgets



Alex Trebek, host of 'Jeopardy!' for 36 years, dies after battle with pancreatic cancer

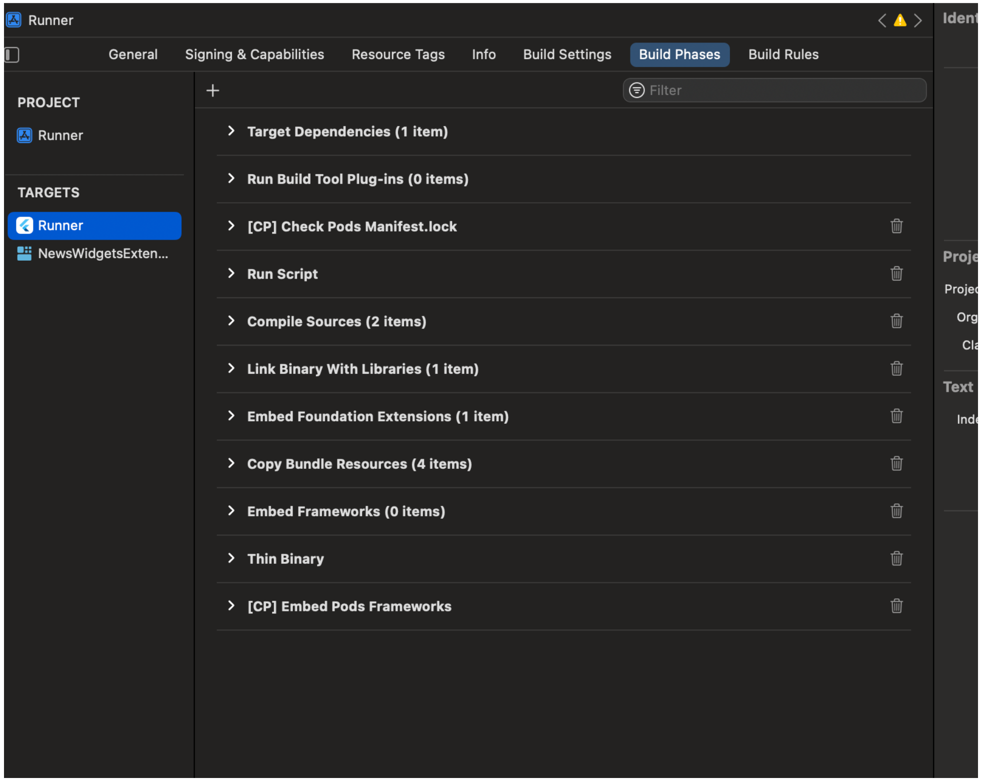
Biden plans immediate flurry of executive orders to reverse Trump policies

Trudeau and Biden: the surprising ties that bind

There's nothing new for you

- The actual “home pages” of your app

iOS Setup



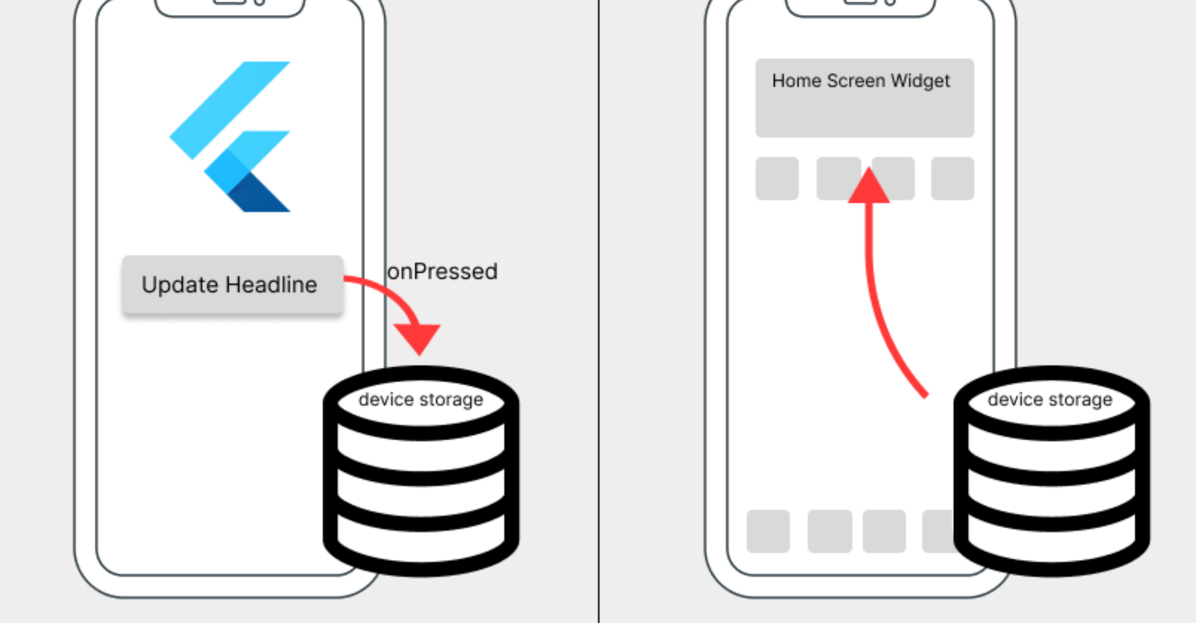
- Target Dependencies (1 item)
- Run Build Tool Plug-ins (0 items)
- [CP] Check Pods Manifest.lock
- Run Script
- Compile Sources (2 items)
- Link Binary With Libraries (1 item)
- Embed Foundation Extensions (1 item)
- Copy Bundle Resources (4 items)
- Embed Frameworks (0 items)
- Thin Binary
- [CP] Embed Pods Frameworks

- Open “ios” folder in Xcode
- File / New/ Target… / Widget Extension
- Build Phases, move “Embed Foundation Extensions” before “Copy Bundle Resources”

iOS Setup

- Set “min deployments” (14+) of extension target to the same version as Runner target
- Prefix of bundle ID with Runner’s bundle ID
- Add same group ID for both Runner and extension targets
- On devices before iOS 17: comment out the code:
  `#Preview() {}`
  in “ios/YourWidgets/YourWidgets.swift”

Data Sharing

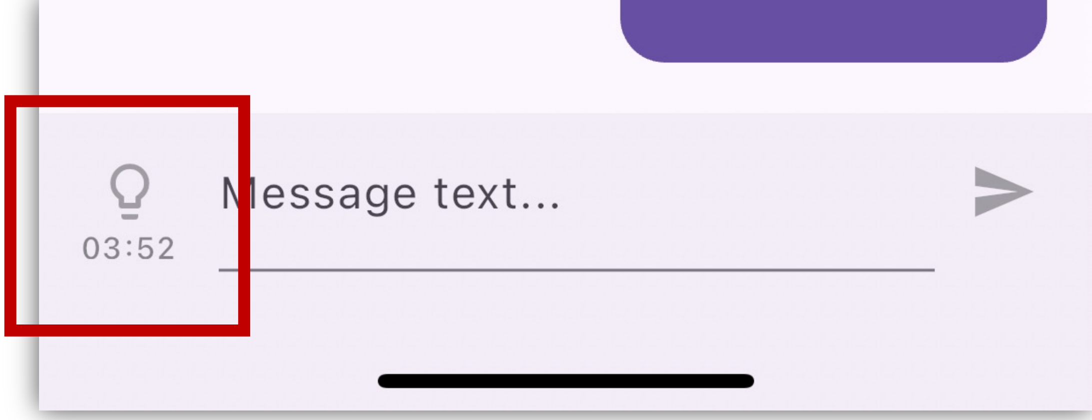


```
import 'package:home_widget/home_widget.dart';

await HomeWidget.saveWidgetData<String>('key1', value1);
await HomeWidget.saveWidgetData<String>('key2', value2);
...
await HomeWidget.updateWidget(
  iOSName: ...,
  androidName: ...,
);
```

- iOS local storage: UserDefaults
- Android local storage: SharedPreferences

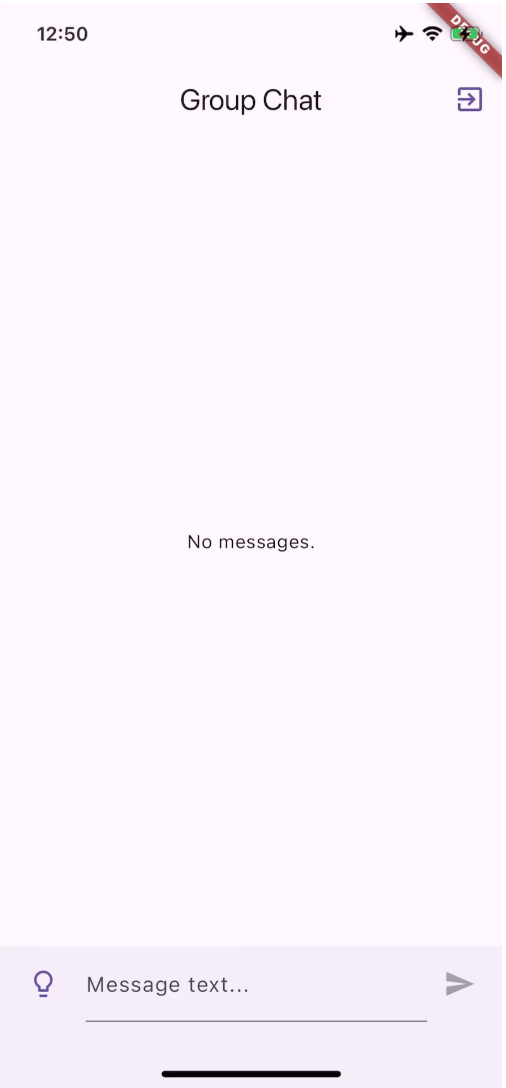
Limitations



- On iOS, you may only update home widgets every 15 minutes
  - Disable sync features in your UI
- OS may kills your home widget if it uses too much resources
  - E.g., >30 MB memory in iOS
  - Avoid showing large images

Bonus: Android Home Widget

- Follow this tutorial first
- Learn how to capture photos using Kotlin
- +5 scores if you can demo this to TA on Android:



12:50

Group Chat

No messages.

Message text...

Your Final Project

- Completeness: ***40%***
- Complexity: ***30%***
- UI & animation: ***30%***

- with ***peer-review***

Presentation

- 5 min each group
- 4 min demo
- 1 min QA

Good luck!