Notetaker Email: jial8@uw.edu

Professor: Lauren Bricker

CSE 340, Spring 2020

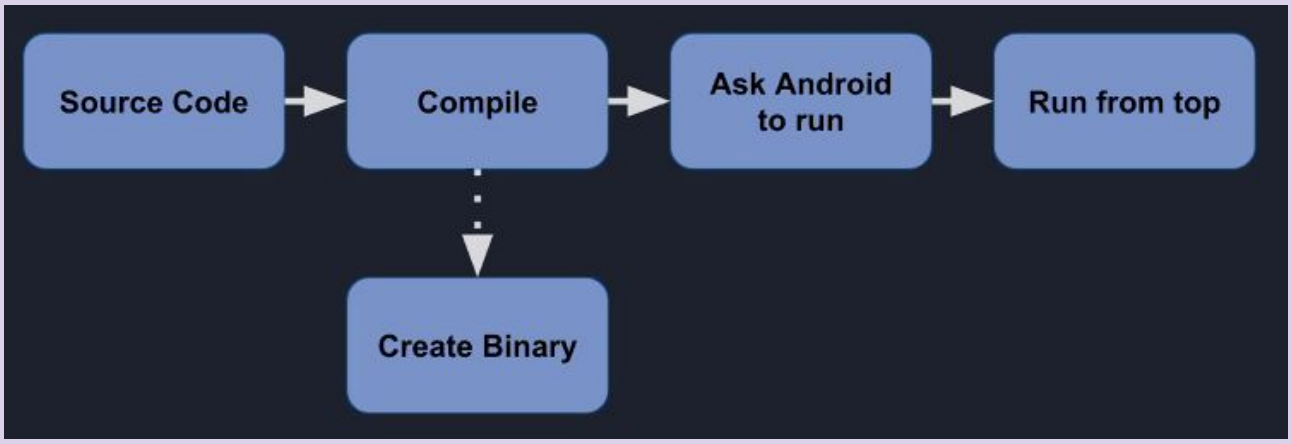
Link to Google Docs for better viewing: https://tinyurl.com/ydz45sdf

Guest Lecturer: Andrew Fitts, Advocate at Google for Flutter, CSE 154 instructor

Lecture Topic: **Interaction Programming with Flutter**

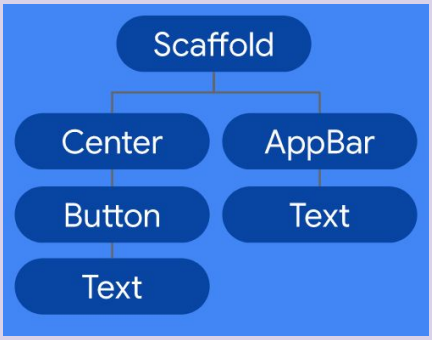
App Development in General

1. Define Layout + Structure (the "view")
2. Define the data we need to work with (ex: String, the "model")
3. Functionality → the "controller" (ex: Activites)
4. Write it up → Compile (takes a lot of time) → GO!



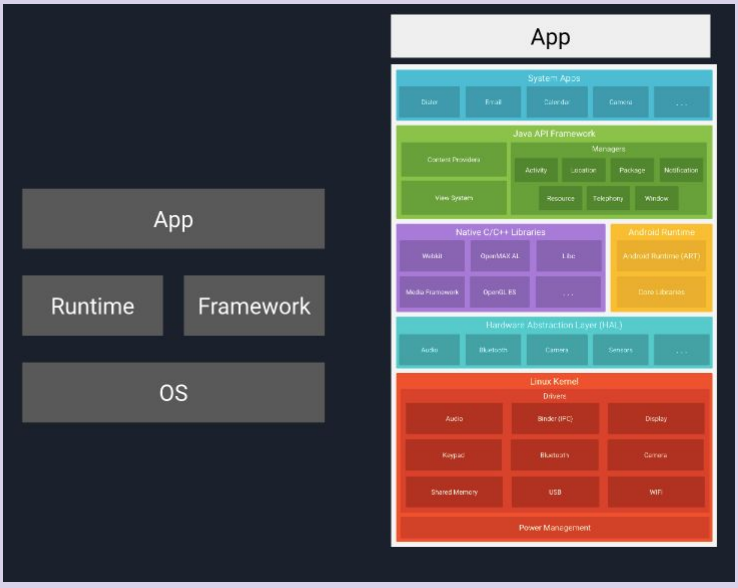
Reactive Pattern in Flutter

- *Everything* is a **widget** in Flutter → widgets are like *components*, images, interactive elements, tools, buttons... etc. Can be:
  - Stateless
  - Stateful
- Explicit state mutation
- Rebuilding widgets is *cheap*



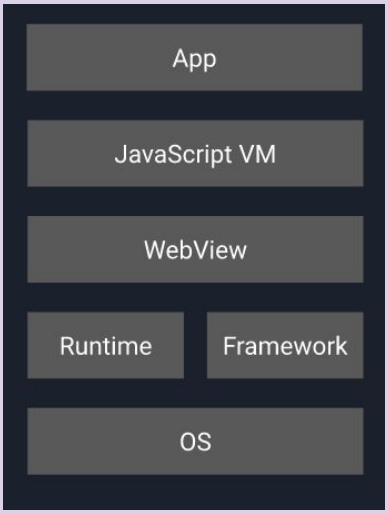
Software Development Kits (SDKs)

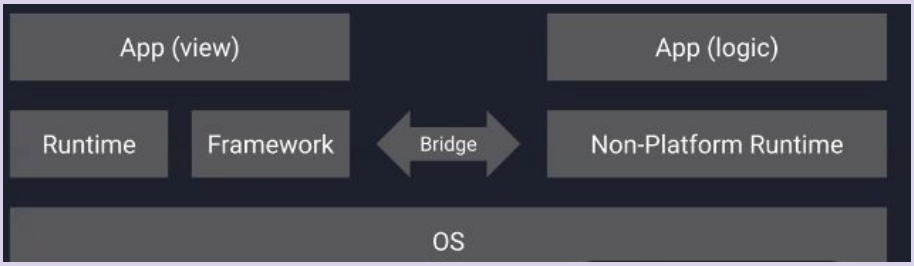
- “collection of software development tools in one installable package”
- Includes compiler + debugger + software framework, eases creation of apps
- Usually specific to hardware platform + OS system combination



WebViews (in Android)

- Use this like a Browser
  - App loads a webpage
- WebView-based SDKs → Has many layers (see above right diagram)
- “**Bridge**”: *links between Framework → Non-Platform Runtime*, connection to other kinds of toolkits





Back to Flutter Stuff…

In Flutter, .dart files → **Dart** language was inspired by Java and JavaScript!

- Uses *main() => runApp(MyApp())* → starting point for all programs
    - Equivalency in Android → onCreate in the MainActivity
- The "App" itself is the most general widget, most high-up/abstract (and stateless!)

```
// In Flutter: no separation of concerns… seems like everything nested inside each other, so if something breaks, everything else breaks

Widget Build(BuildContext context) {
        return Scaffold(
                appBar ….
        ),
        body: Center(
                child: Column(.....) // Column is a widget → vertical contain children
                        // does the container itself show up on screen? no?
        // everything else
        floatingActionButton: FloatingActionButton(
                onPressed: _incrementFunction,
                tooltip: 'Increment'
                …
        ),
        ...
}

// When you save the code after changing a small thing → Hot Reload
   - displays changes → new code changes without restarting the app from start
     and it effects only on the changed code
```

Note: *Separation of Concerns… what benefits? → localization of languages (no separation = formatting will be messed up)*

- Flutter: mostly still separated of concerns, except when it comes to Layout (everything is squished together!

****Hot Reload****

- *Does NOT rerun main() or initState()*
- How to trigger this? Just click “Save All” in the Flutter IDE
    - Preserves the app state
    - Re-builds the widget tree

Note: *Alignment and Spacer and TextDirection (TD includes specifying right to left or vice versa, is part of intl or “international” class) are also widgets!*

- ***Ex Code: Align(alignment: Alignment.bottomLeft…)***

Questions:

Is Flutter similar to React (Native) What are the differences?

- Yes, the model is similar
- You can run the exact *same code* on desktop app, web, Android, iOS with both frameworks.
- Short answer: there are so many similarities, differences include: Flutter uses the Dart runtime instead of platform’s runtime.
    - Dart has a Virtual Machine, can also be compiled into machine code (native)
    - Android only compiles down to Android VM code, not native

Why is there only 1 FAB allowed in a Scaffold widget?

- Material Theming/Design = UI guideline for how to design things, generic design system for user interfaces
- Short answer: Scaffold is following a specific design spec → has limitations

Is this lecture content part of CSE 154?

- Sorry, no :(
- Some general concepts are applied tho