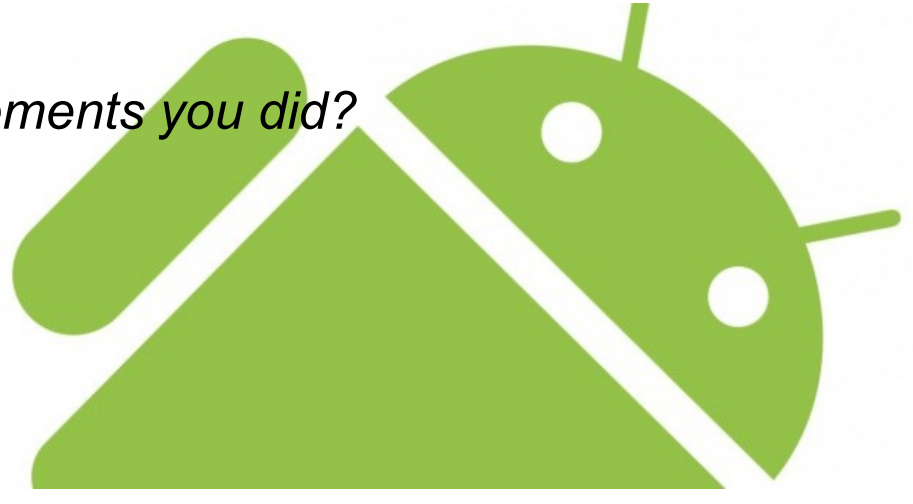
Android Projects Group

Meeting 3:

Designing your app

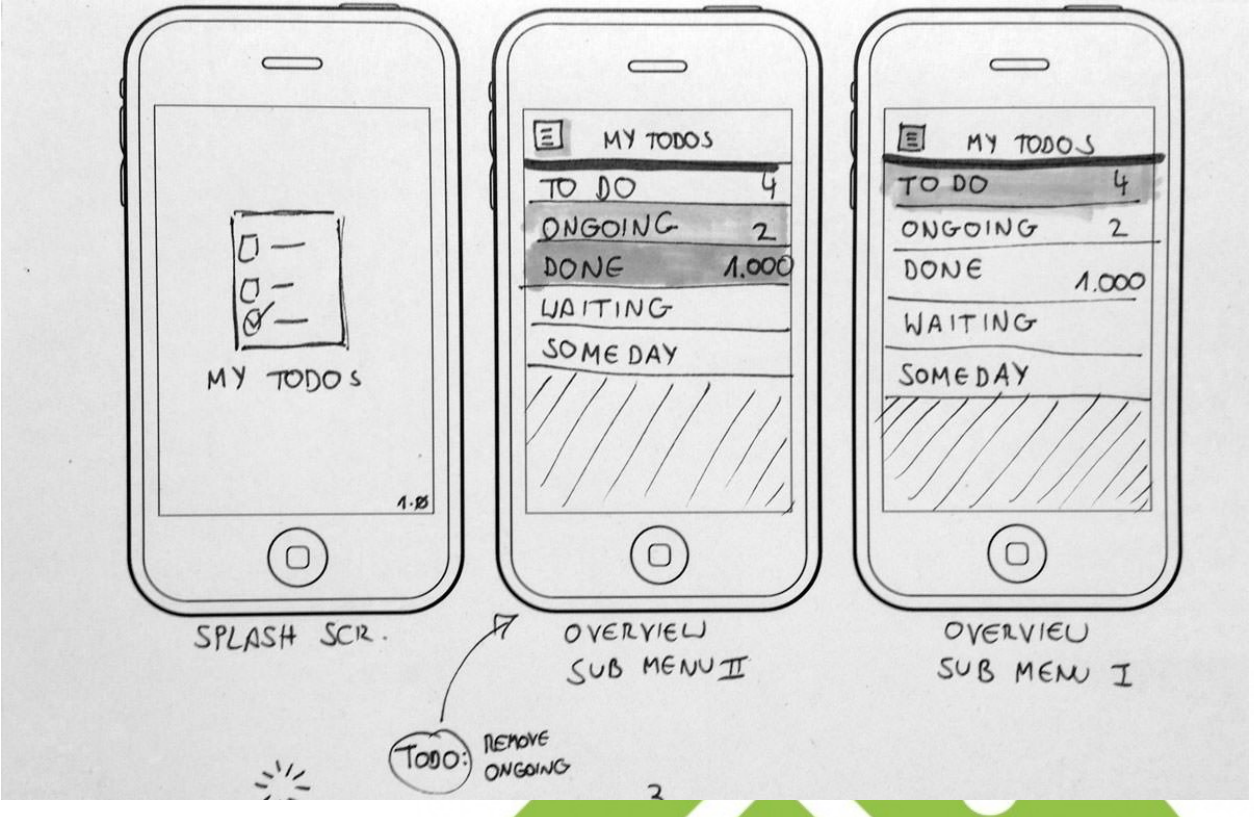
Designing your app

- Your app idea ✓
- Literature review ✓
  - Find out what similar apps are out there
  - What the user needs might be
  - Refines your idea of what your app will do
- Functional Requirements specification ✓
  - Defines your app core functionality
  - Sets out what your app will do
  - *Remember those functional requirements you did?*



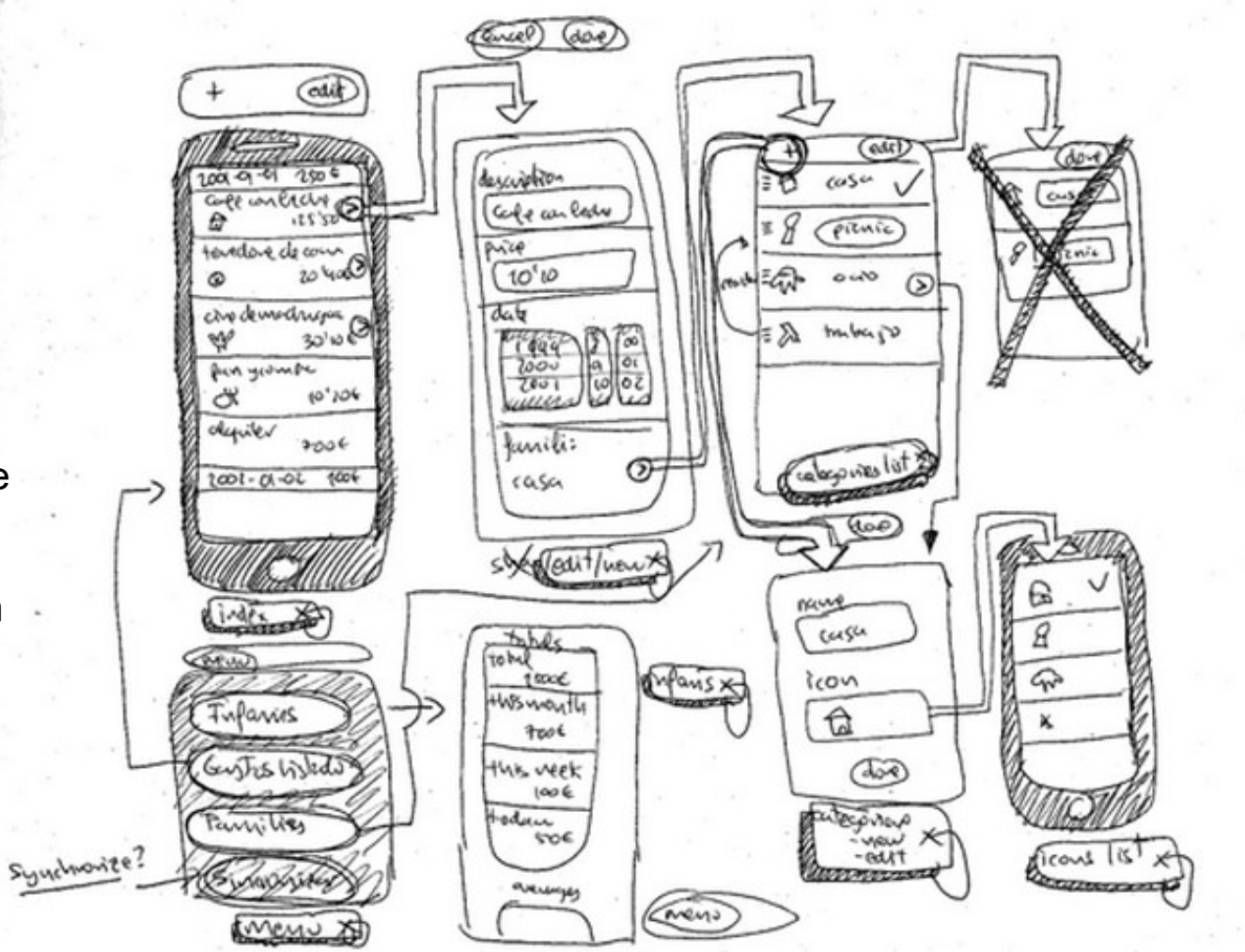
App activities

- Sketch your app ‘Activities’ based on your functional requirements
  - Wireframes and interface sketches or storyboards show what your app screens, or ‘activities’ will look like.
  - Show how users will interact with your app.
  - Remember ‘Material Design’ when sketching.



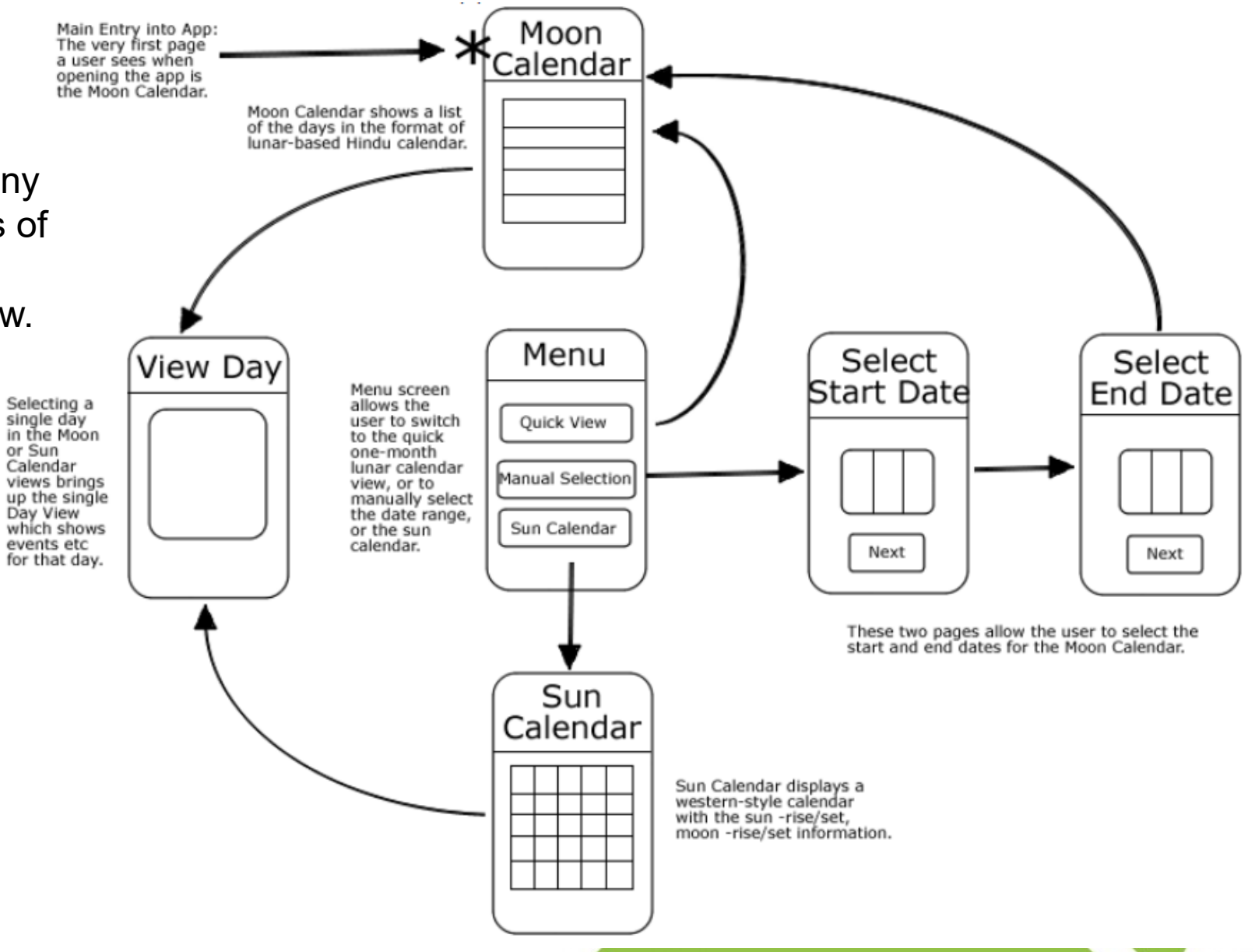
App interaction flow

- Plan the app interaction flow.
  - An overall view showing how it all connects together.
  - Show how the user interacts with your app, from start to finish.
  - Perhaps start with the home activity.
  - Show how interaction moves between activities.



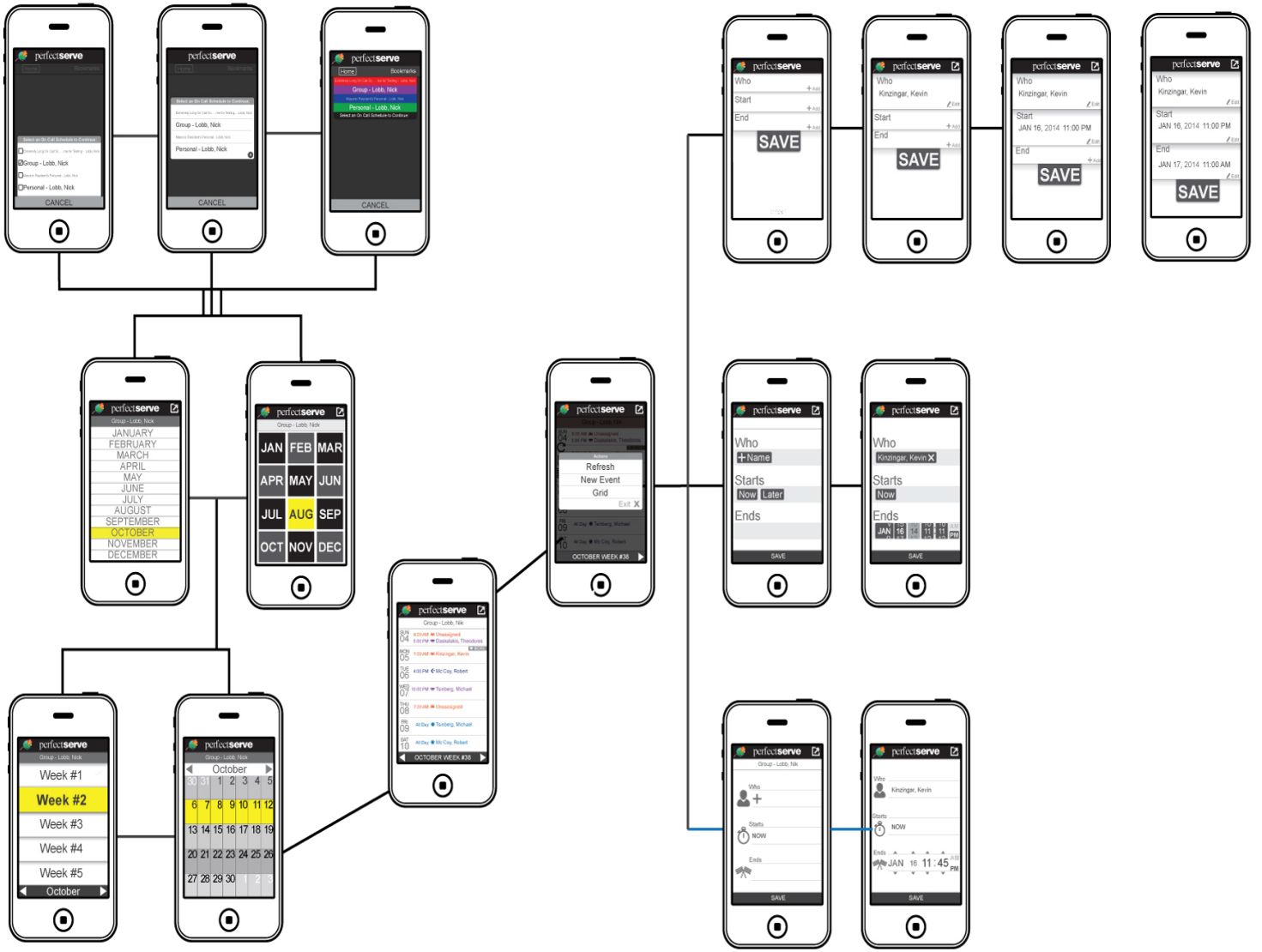
App interaction flow

- There are many different ways of showing your interaction flow.



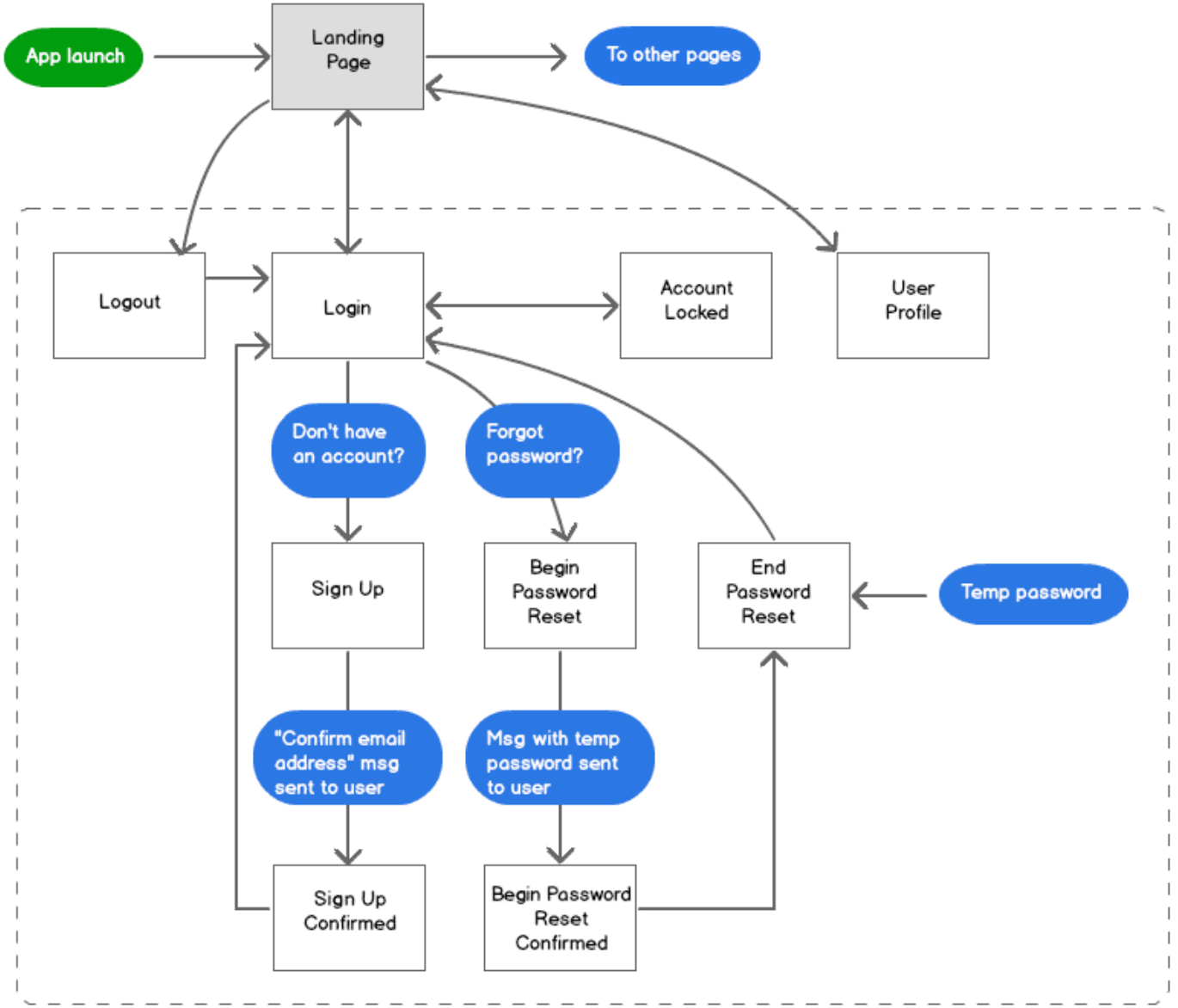
App interaction flow

- There are many different ways of showing your interaction flow.



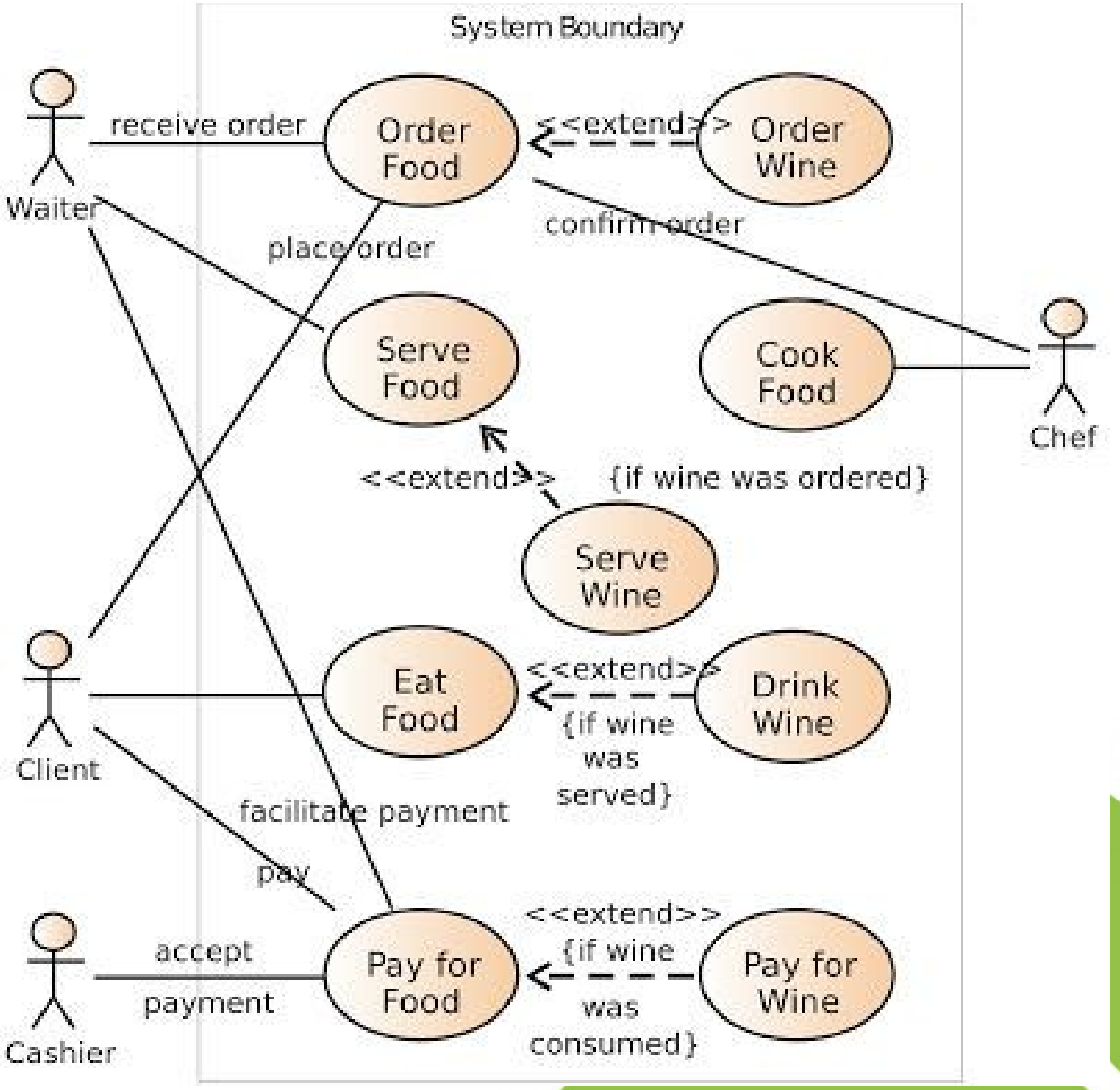
App interaction flow

- You can use flowcharts to show more detail when designing your app.
- These diagrams will help you design better structure, and save time later.



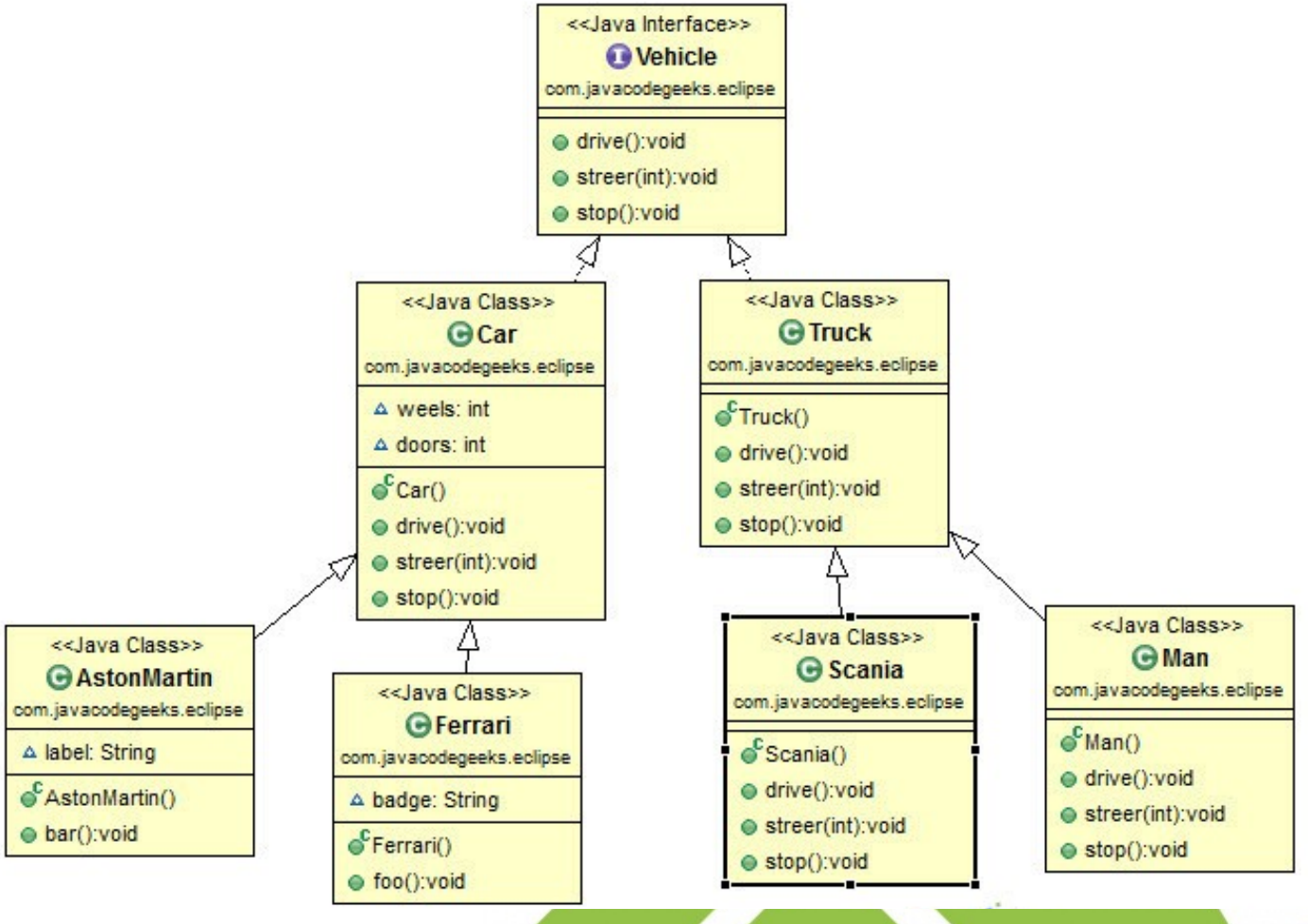
Use Case diagrams

- These are valuable for visualizing the functional requirements of a system.
- They show how the functional requirements join together when the app is being used.
- They work to define your Classes, Functions etc and how they talk to each other.



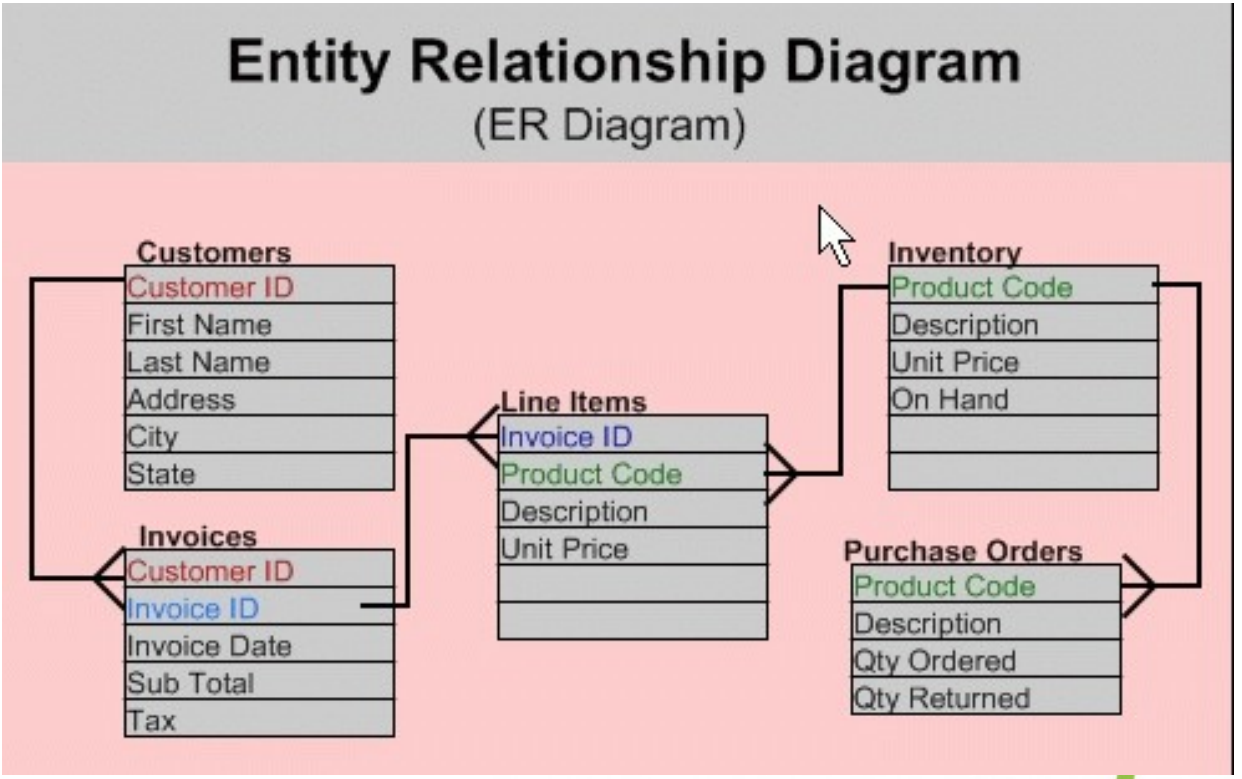
Class Diagrams

- Show the main classes in your app.
- These visualise the relationships and source code dependencies among your classes.
- They define your Classes and how they talk to each other.
- Helps you to decide how the code is structured.
- Helps decide variables etc!



Data structures

- If you are using a database and SQL:
  - Do an entity relationship diagram.
  - Keep the database simple!
- Simple data structures:
  - Arrays, lists, files etc work out what you are storing and how you want to store it.
  - Activities are non-persistent
  - Store on device or cloud?
  - Try to keep things simple.



Firebase for databases and messaging

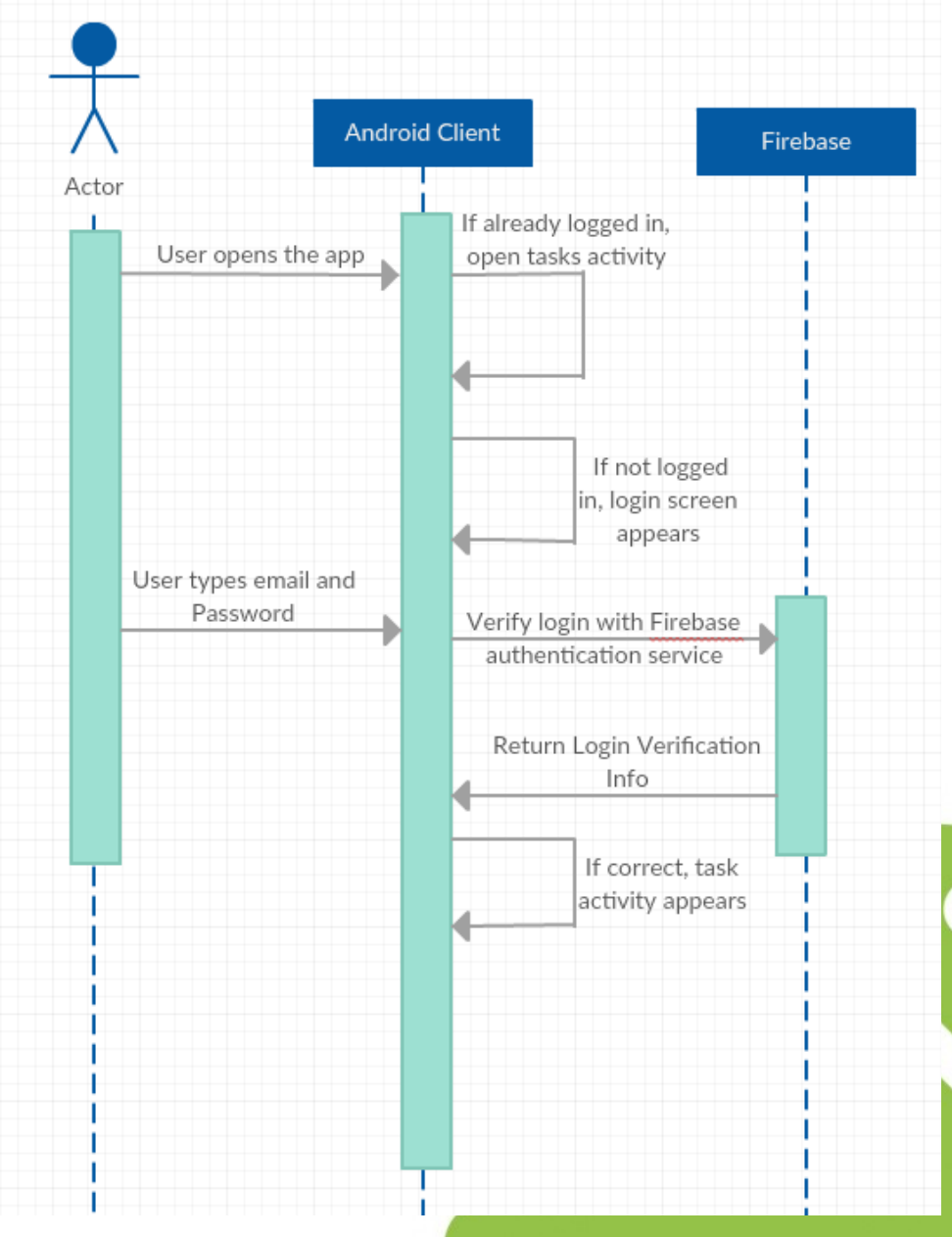
- You may be using Google's Firebase
- This uses a Non-SQL format.
- Firebase Realtime Database data is stored as JSON objects.
- You can think of the database as a cloud-hosted JSON tree.
- Firebase allows data to be instantly shared across devices.
- Firebase sends data updates to all connected devices.
- Here's a sample JSON Firebase data structure, note the keys link to the actual data.

https://friendly-chat-12345.firebaseio.com/

```
friendly-chat-12345
  messages
    -K2ib4H77rj0LYewF7dP
      name: "anonymous"
      text: "Hello"
    -K2ib5JHRbbL0NrztUfO
      name: "anonymous"
      text: "How are you"
    -K2ib62mjHh34CAUbide
      name: "anonymous"
      text: "I am fine"
```

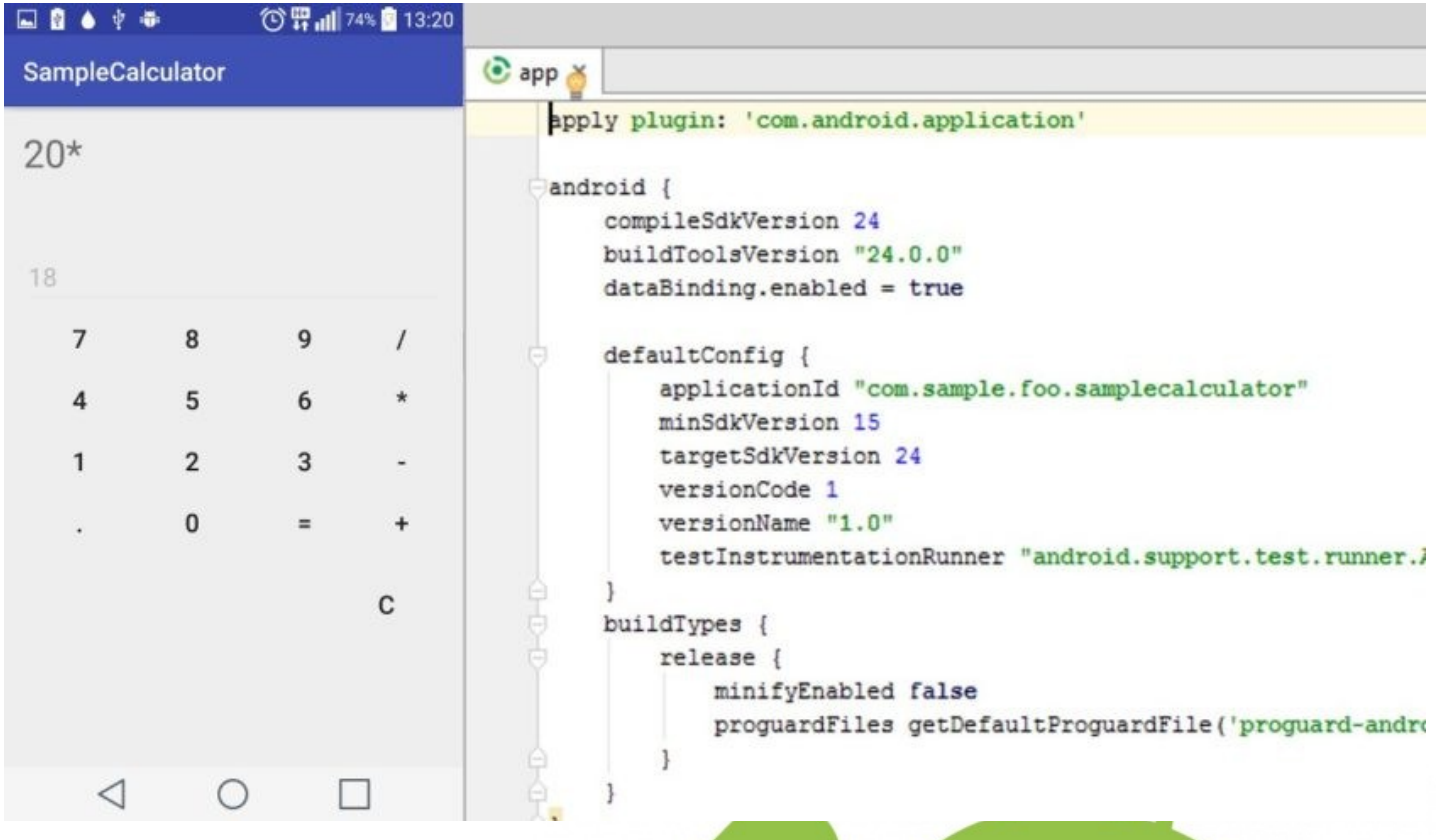
Sequence Diagrams

- If you are connecting to Firebase or another web server.
- You may need a sequence diagram.
- This shows how the app(s) and the server talk to each other.
- Shows timing of messages.
- Useful if you have a messaging app.
- Useful if you are synchronising data between apps.
- Help you to understand how data is moving around.
- Keep it simple!



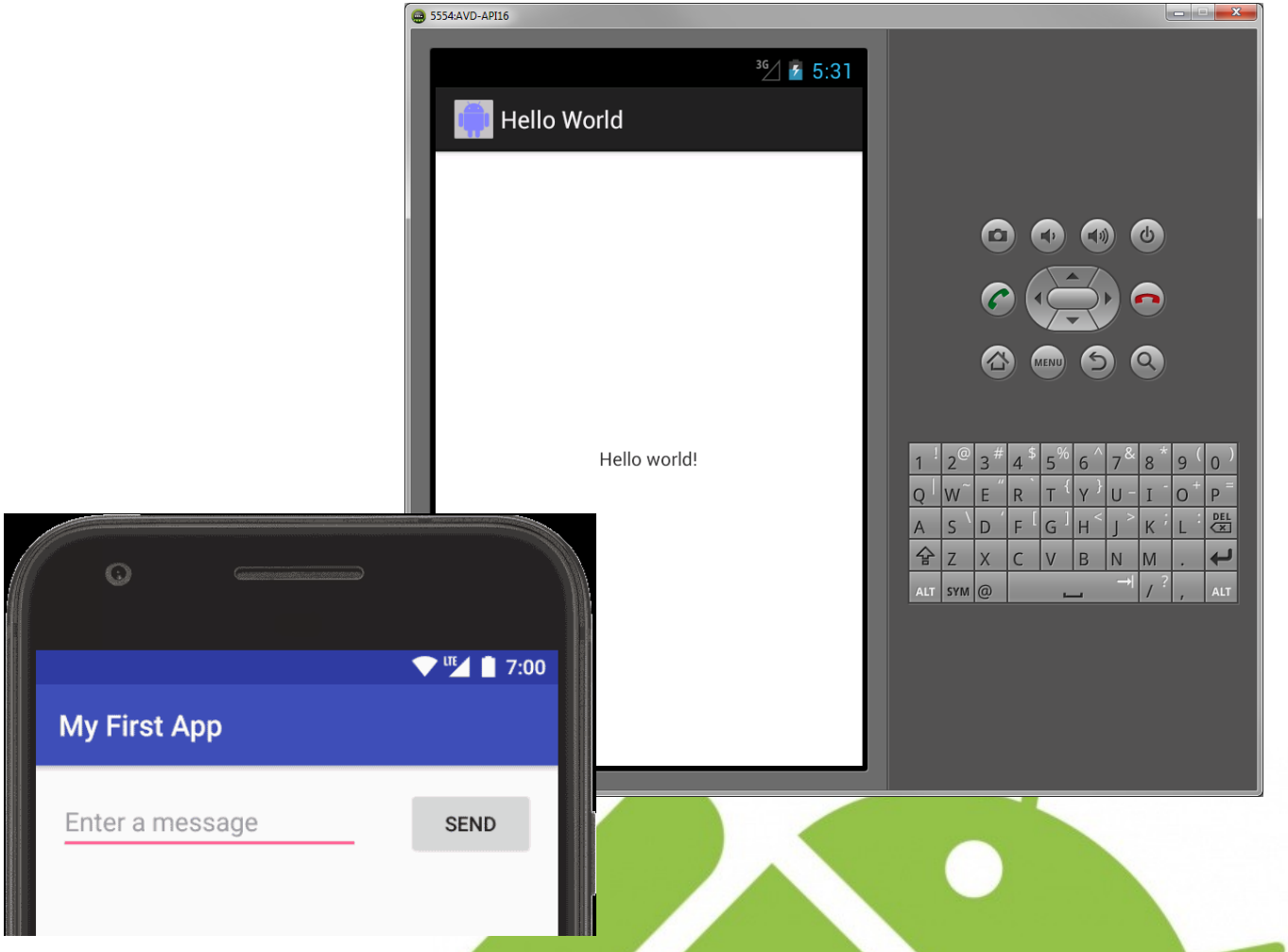
Find sample code

- Now your app is designed you know what it needs to do.
- Find code samples for individual functions.
- Use existing solutions.
- Try them out.
- Understand them.
- Make comments in the code.
- Build a library of code samples that do things you need.
- Keep things simple if you can.



Start Coding Test Modules

- Build one function at a time. Test it.
- Add more functions from your functional requirements. Keep testing as you build it.
- Incrementally build your app, testing as you go.
- Maybe build seperate test apps for the different functions, test them, then put the tested building blocks together later to make the full app.
- Don't build a monster.
- Keep it simple.



Summary

- Work on your app design:
  - Find samples.
  - Look at guides and examples.
  - Good design means easier coding!
  - All your designs will be in your final report.
  - You get marks for good design – it's not all about coding!
  - Use the tools and methods that best describe your app.
  - Research how to do the things your app needs to do.
- What happens next:
  - Individual meetings.
  - More detailed individual help with your app.
  - Send me your designs for feedback.
  - Next meeting Wednesday 6th December GH4.81/82
- Week 7 deadline - Friday 17th November midnight - TurnItIn