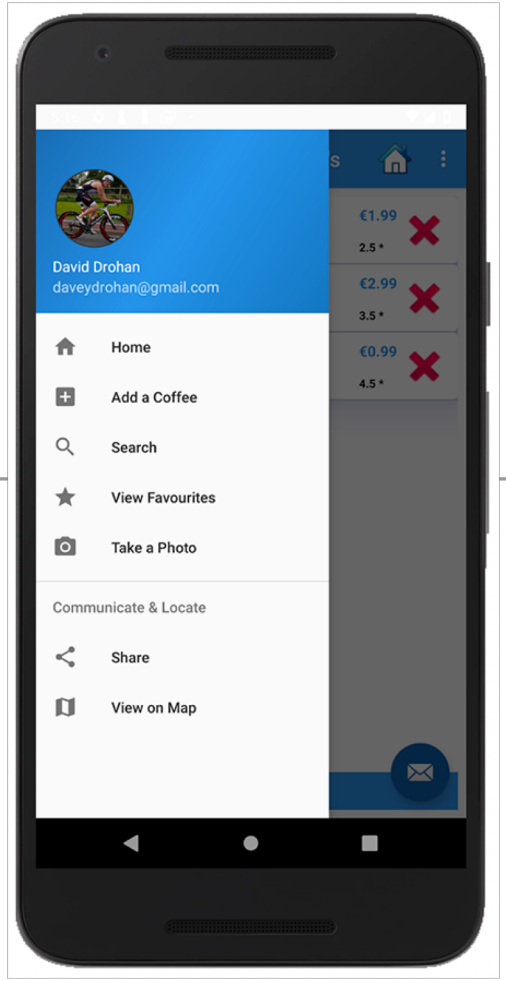
Assignment 2

70% of Overall Grade



Agenda

- Specification
- Grading Rubric
- Submission Guidelines
- Presentation

Agenda

- Specification
- Grading Rubric
- Submission Guidelines
- Presentation

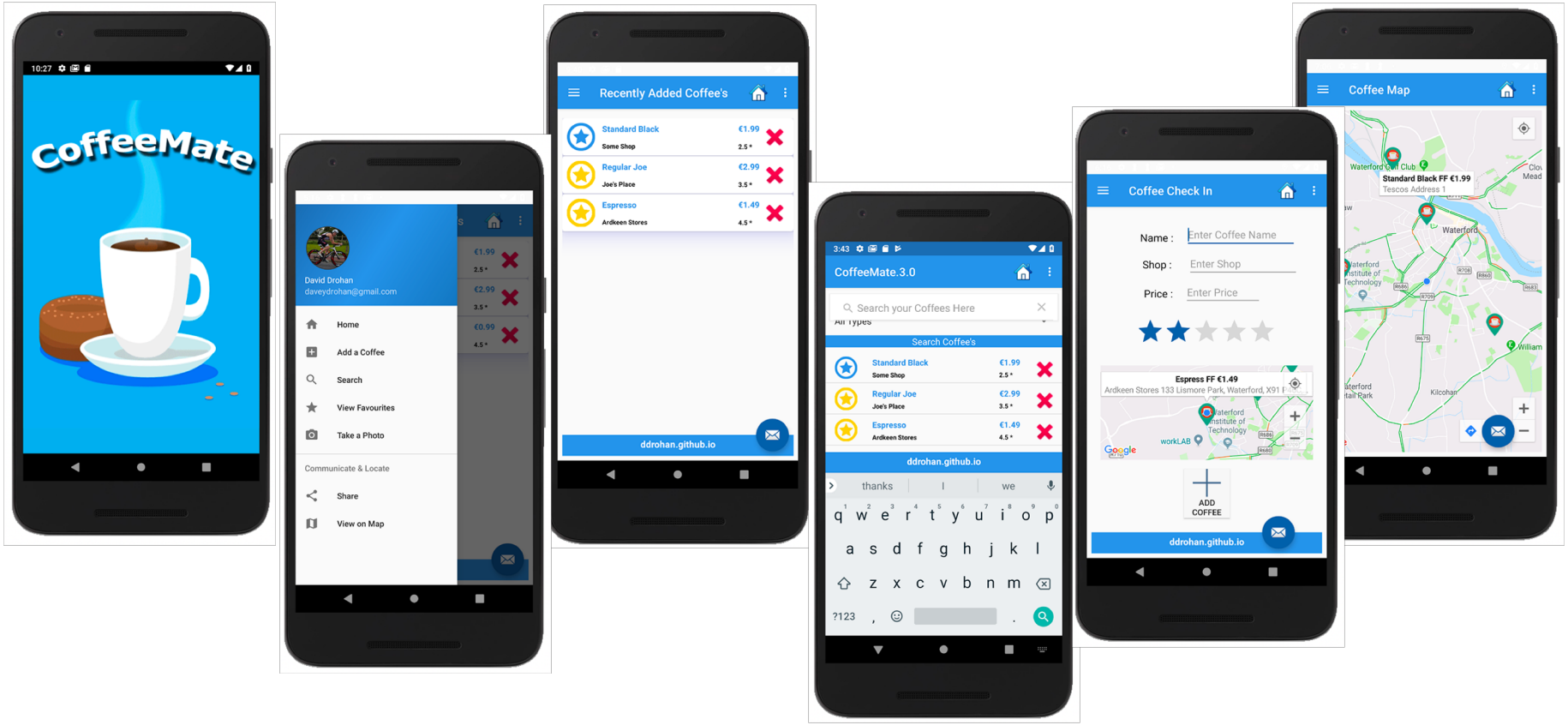
Assignment 2

Continue working on your own app, exhibiting similar level of complexity / feature density as covered in the Case Study throughout the remainder of the Semester

Sample Features (as covered in Lectures / Labs)

1. A ‘Splash’ Screen (via multithreading)
2. Enable User Signup / Registration / Login, via Google Sign In
3. The coffees are persisted (in an SQLite database), and will be reloaded when a user logs in.
4. Support viewing individual coffees.
5. Allow a user to delete all coffees from the database.
6. Individual coffee can be updated/deleted/filtered
7. All coffees can be view on a map via Google Maps
8. Current user location stored when coffee added

Sample Features (as covered in Lectures / Labs)



Agenda

- ~~Specification~~
- Grading Rubric
- Submission Guidelines
- Presentation

Assignment 2 Rubric (70% - [90% App / 10% Doc])

Standard	Functionality [55%]	Persistence [25%]	UX [10%]	DX [10%]
Baseline	Basic Functionality with full CRUD	Persistence for duration of app only.	Conditional App Navigation (via Menus)	Data Validation
Good	Additional Functionality as part of CRUD eg searching/filtering	SQLite / Realm	Use of UI elements to complement UX eg NumberPicker Vs EditText	Adherence to Android Best Practices
Very Good	Use of >1 3rd Party API	Cloud-based Persistence	Alternative Navigation	Repo Usage, git etc.
Excellent/ Outstanding (70%+)	Use of Google APIs (or equilavent) for authentication, location etc.	Integration with Web via REST APi	UI & Material Design Guidelines adhered to	Automated Testing

Pass line

Agenda

- ~~Specification~~
- ~~Grading Rubric~~
- Submission Guidelines
- Presentation

Technical Report (10% of Final Grade)

Include a Technical Report, comprising of:

- In depth discussion of all functionality, including, if any, 3rd party and/or Google APIs used.
- UML & Class Diagrams.
- UX / DX Approach Adopted
- Git approach adopted and link to git project / access, if any
- Personal Statement
- References

Submitting Project Code and APK

Submit archive via Moodle dropbox. This should also include:

- The Technical Report
- an APK & full source of your project (excluding temporary build files)
- Video LINK to App Demo (Max 10 mins)

Give read access to your lecturer to your GitHub / BitBucket repos (if applicable). GitHub and BitBucket id is: **ddrohan.**

NOTE : IF I HAVE TO CONTACT YOU ABOUT YOUR SUBMISSION YOU WILL BE AUTOMATICALLY DEDUCTED 10% FROM YOUR FINAL GRADE

Agenda

- ~~Specification~~
- ~~Grading Rubric~~
- ~~Submission Guidelines~~
- Presentation

Presentation

You will be allocated a 10 - 15 minute slot in the final week of lectures and practical labs to present your project.

- Attended by Tuition team only (me!)
- 10 - 15 Minutes to include demo + Q&A

Note: I will be strict on time allocation, so please be ready to demo *at least 10 minutes before your scheduled time*, with your Laptop/PC powered on with your app / code walkthrough running.

Questions?