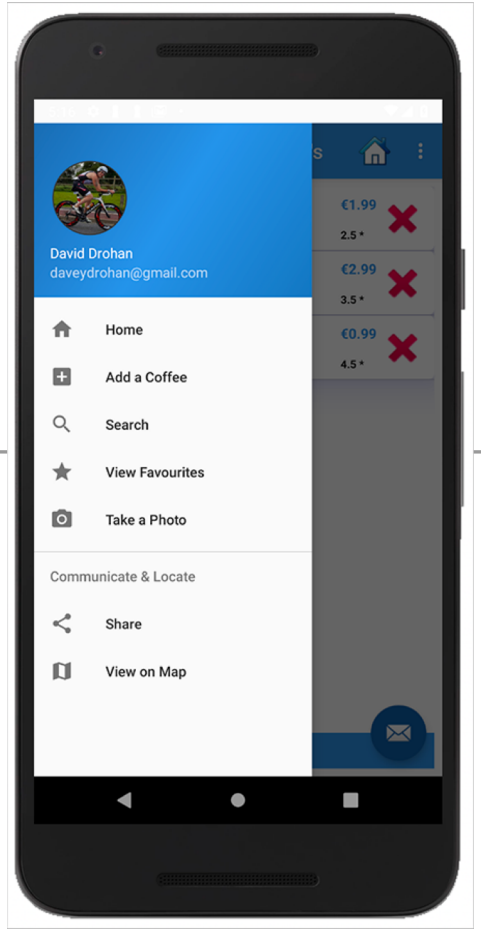
Assignment

100% of Overall Grade



Agenda

- Specification
- Grading Rubric
- Submission Guidelines
- Presentation

Agenda

- Specification
- Grading Rubric
- Submission Guidelines
- Presentation

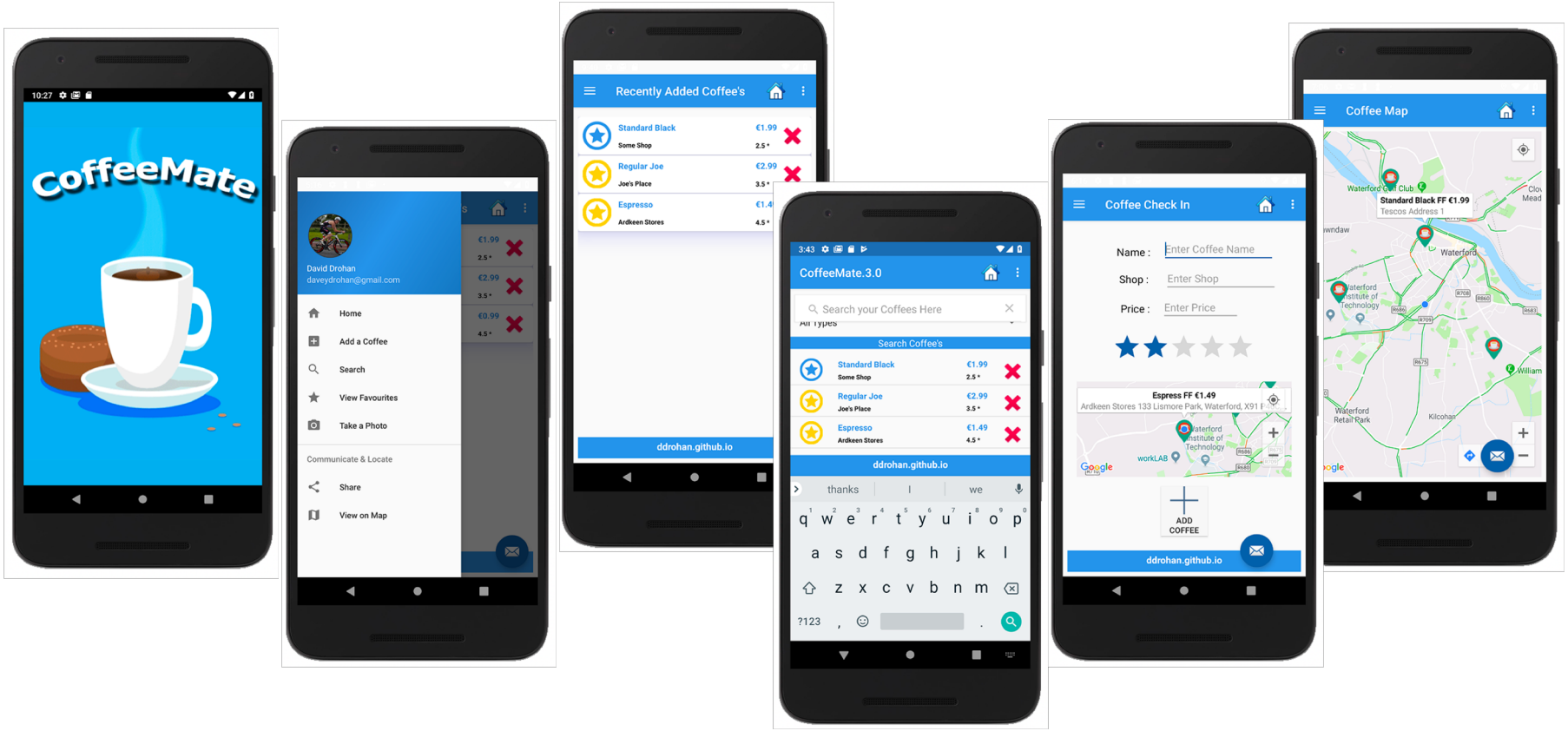
Assignment

Develop your own app, exhibiting similar level of complexity/feature density as covered in the Case Study throughout the Semester.

Sample Features (as covered in Case Study)

1. A ‘Splash’ Screen (via multithreading)
2. Enable User Signup / Registration / Login, via Google Sign In
3. The coffees are persisted (in an SQLite / Realm database), and will be reloaded when a user logs in.
4. Support viewing individual coffees.
5. Allow a user to delete single / multiple 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 Case Study)



Agenda

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

Assignment Rubric (80% of Final Grade)

Standard	Functionality [65%]	Persistence [15%]	UX [10%]	DX [10%]
Baseline	Basic Functionality with full CRUD	Persistence for duration of app only	Conditional App Navigation (via Menus)	Data Validation
Good Pass line	Additional Functionality as part of CRUD eg searching/filtering	Shared Preferences	Use of UI elements to complement UX eg NumberPicker Vs EditText	Adherence to Android Best Practices
Very Good	Use of >1 3rd Party API	SQLite / Realm	UI Guidelines adhered to	Repo Usage, git etc.
Excellent/ Outstanding (70%+)	Use of Google APIs (or equilavent) for authentication, location etc.	Cloud-based Persistence	Material Design Guidelines adhered to	Automated Testing (models)

Agenda

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

Technical Report (20% 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.
- Your App's future development and possibilities.
- The Business Case for the success of your App
- Personal Statement.
- Git approach adopted and link to git project / access, if any.
- References

Submitting Project Code and APK

Submit archive of code via Moodle dropbox. This archive should also include:

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

Give read access to your lecturer to your GitHub / BitBucket repos (if applicable).

GitHub and BitBucket ids are:

- ddrohan.

Agenda

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

Presentation

You will be allocated a 30 minute slot in the final week of lectures and practical labs to present your project.

- Attended by Tuition team only.
- 30 Minutes to include demo + Q&A.

Note: I will be strict on the 30 minute allocation, so please arrive outside the room *at least 10 minutes before your scheduled time*, with your Laptop ready to go with your app / code walkthrough.

Questions?