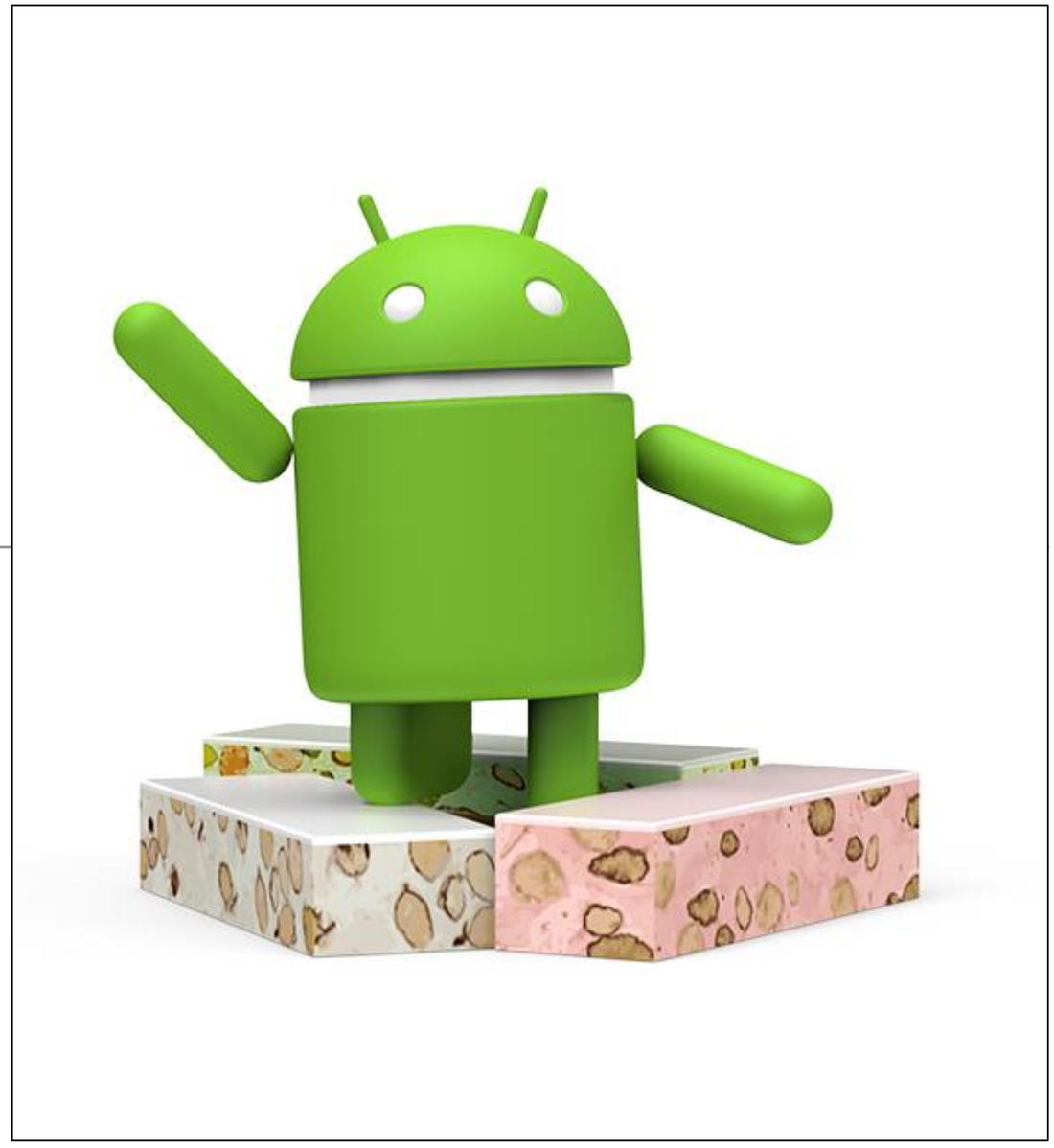
Assignment 2

60% of Overall Grade



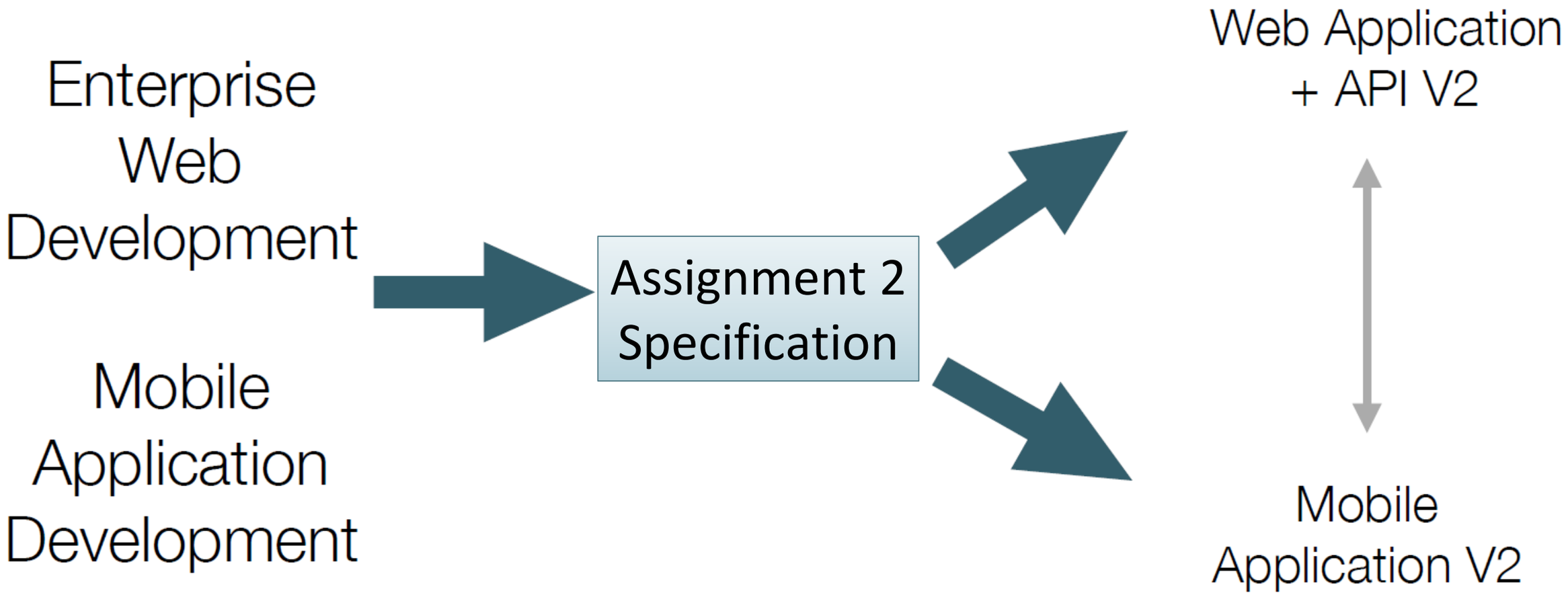
Agenda

- Specification
- Grading Rubric
- Submission Guidelines
- Presentation

Agenda

- Specification
- Grading Rubric
- Submission Guidelines
- Presentation

Assignment 2 - Approach



Assignment 2 - Two options

1. Continue building your MyTweet (twitter) App..

or

2. Continue working on your own app, exhibiting similar level of complexity/feature density.

Assignment 2 - Specification

Two applications:

- A Web Application V2.0.
- A Mobile Application V2.0.

Features:

- Shared Features (present in both apps) e.g. social → following / follower.
- Features unique to each app.

Sample Features (carried over from Assignment 1)

1. Enable User Signup / Registration / Login.
2. Enable user to post 140 character tweets.
3. These tweets are persisted, and will be reloaded when a user logs in.
4. Support viewing all tweets user has posted - the users public timeline.
5. Allow a user to delete a subset of tweets.
6. Allow a user to delete all tweets.
7. Allow a user to edit account settings (email, password, and other details).

Agenda

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

Assignment Rubric for Assignment 2

Standard	Functionality [50%]	Persistence [20%]	Git [10%]	UX/DX [20%]
Baseline	Assignment 1 Functionality with full CRUD	Persistence for duration of app only.	git + commit History	Adherence to Android Best Practices
Good	Use of 1 Google API	JSON	Tagged releases + quality readme	Data validation
Very Good	Use of 2+ google APIs	SQLite	Branch-based workflow. http://bit.ly/1WeHsJu	Advanced navigation, etc.
Excellent/ Outstanding (70%)	Integration with web app	Cloud-based Persistence	Comprehensive branching model http://bit.ly/197szRP	Mobile/Web best practice UX e.g. similar look-and-feel, etc.

Pass line

Some Notes on the Rubric

Functionality

- You decide which additional functionality you wish to include e.g. social (following/followers) etc.
- Use the Grading Rubric as a guide for choosing new features.

Persistence:

- To get top marks in this area, the app should be able to store data using all three persistence approaches, for example, SQLite for personal mobile app settings, cloud based storage for data that will be shared with the web app etc.

Agenda

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

README file

Include a VERY brief README file (max two pages):

- Name and Student ID.
- Brief description of new functionality, including the Google APIs used.
- Brief overview of Web App integration and link to deployed app.
- Persistence approach adopted i.e. what's persisted and where.
- Git approach adopted and link to git project / access.
- UX/DX approach adopted.

Submitting Project Code and APK

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

- the README file and
- an APK of your project.

Give read access to your lecturers to your GitHub / BitBucket repos. Our GitHub and BitBucket ids are:

- **sdrohan** and **ddrohan**.

Agenda

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

Presentation

You will be allocated a 20 minute slot early January to present your project.

- Attended by Tuition team only.
- 10 Minute presentation (with slides).
- 10 Minute demo + Q&A.
- Slides to consist of walk though of demo via screen shots + supporting explanatory text.
- Slides to be submitted on Moodle in early Jan, prior to your presentation taken place (date/time to be announced).
- 7-12 slides approx.

Note: We will be strict on the 20 minute allocation, so please arrive into the room with your Laptop ready to go with your presentation / code walkthrough and adhere to the above structure.

Questions?