

Student Name	Course	Group	Licence Utility (2)		
			Licence Utility with ArraList and Validation	App class - field added, updated mutators, constructors, setters etc	AppStoreAPI and Driver uses new Licence field
			4	5	4
Luke Blignaut			4	2	0
Arthur Bilyk			3	3	3
Sade Hegazi			3	3	1
Joshua Webster			0	0	0
Dean Sinnott			4	5	4
Michael Ogunnaike			1	2	3
Luke Bury			0	0	0
Kyle Wall			1	2	0
Aaron Flanagan			2	3	2

Ilija Gacov			2	5	4
Mimi Osita			4	3	0
Odinaka Eloji			0.5	0.5	1
Alexandra O'Brien					
Luke Donovan			3	2	2

2%)		Javadoc (5%)	Developer (11%)		
Licence Utility Test	EducationApp Test - licence added and tested	Javadoc on AppStoreAPI class	developerPhoneNumber (String) – must only contain numbers	developerCountry (String) – max 80 chars	developerLatitude (double) – values between -90 and 90
4	5	5	3	2	2
0	0	0	0.5	0.5	0.5
3	2	3	0.5	0.5	0.5
0	0	3	2	1	1
0	0	0	0	0	0
0	0	1	3	2	2
0	0	0	1.5	1.5	1.5
0	0	0	0	0	0
0	0	0	1	1	1
0	3	0	2	1.5	1.5

0	3	5	1	0.5	0.5
0	0	0	0	0	0
0.5	2	0	1	1	1
0.5	1	1	1	1	1

		DeveloperAPI (7 marks)		Driver (4 marks)	
developerLongitude (double): values between -180 and 180 inclusive	getters, setters, toString etc	all fields updated when updating developer	list all Apps from developers in a given country	New fields in add developer	New fields in update developer
2	3	3	4	2	2
0.5	1	0	0	1.5	0
0.5	1	0	0	1.5	0
1	2	0	0	2	0
0	0	0	0	0	0
2	3	3	4	2	2
1.5	1	0	0	0	0
0	0	0	0	0	0
1.5	2	3	0	2	1.5
1.5	3	0	0	2	0

0.5	1.5	0	0	2	2
0	0	0	0	0	0
1	1	1	2	1	0
1	2	0	0	2	0

Existing Code from May Specification (50%)				TOTAL before Interview
App (abstract), Education, Game and Productivity Hierarchy	AppStoreAPI - crud, reporting, validation and sorting methods	Driver - app, developer menus, search, sort, randomApp, recommended app, save/load, exit	JUnit - for hierarchy and AppStoreAPI	
13	13	12	12	100
6	7	4	2	29.5
6	7	8	2	44.5
11	10	3	1	44
8	8	3	12	31
13	6	5	2	63
6	9	6	0	34
4	2	2	0	10
7	1	5	11	40
10	10	6	2.5	50

10	11	12	11	71
12	1	1	6	27
6	2	4	2	27.5
				25
6	6	6	4.5	40

Interview (understanding / authorship)	Final Grade	Checked in Broadsheets
10	100	
AB	AB	Y
3	13.35	Y
10	44	Y
10	31	Y
10	63	Y
3	10.2	y
10	10	Y
10	40	Y
10	50	Y

10	71	Y
10	27	y
10	27.5	Y
9	22.5	Y
10	40	TODO: update to 40

Comments / Feedback

App - licence field added, but not validated and not in constructor. Developer fields added but no validation. Added to equals and toString however. Existing code - hierarchy has good bit of code but not compiling and other issues are in it. Revisit marks for existing code after interview - only ball parks put down here.

didn't understand this., super, foreach loop, += and most things I asked for . Said he got help at sorts. didn't understand boolean functions. Got quite irate when asked questions. I had to leave it that I did not believe it that it was his work and to come back to me if he disagreed with this.

Licence - inefficient validation check. App - licence added but no validation and not in constructor. In toString however. Driver - licence in addApp. Developer fields - validation done at mutator level - none at constructor level. Existing code - junit only two classes and all commented out and appears to be supplied skeleton code. App - hierarchy - bugs in code eg calcualteRating. Revisit marks for existing code after interview - only ball parks put down here. had to create a new project and add sade's files

No items from the repeat specification were included. Existing code wise, driver hasn't much beyond the skeleton code included. app - validation done in getters, appstoreapi - two arraylists. junits - lots of empty classes, not all passing.

More done for extra - funny bug where iserializer was causing a non-implemented bug. Good work

Attempt made at licence utility, but not correct. App - licence field added, but not in constructor, no validation, get/set added. AppStoreAPI - getAppLicence() method added but code is unusual. No junit and no javadoc. Developer fields added but sout prints included in validation. Fields not in constructor or toString. App hierarchy - bugs in isRecommendedApp() and no validation on fields.

Not compiling, no tests, no new fields in Developer, no licence utility added. App hierachy is modeled on Post hierarchy, not per spec.

Attempt made at licence utility, but added a Licence Class with three vars, one for each licence type. App has an array of licences instead of one licence. Code from May is minimal in places and only skeleton code was included. Developer - new fields added - validation only done in setters.

Very good interview. Developer and Licence done but not all validation present. Tests - only ed app. No javadoc.

naming standards - field starting with capital letter (Licence), category in LicenceUtility - should have been renamed to licence (in multiple places). Licence utility also does the adding in the getCategories method - not cohesive. Licence utility and other comparisons doesn't take case sensitivity into account when comparing. bloated code e.g. see isRecommendedApp throughout. toString has a parameter called information. AppStoreAPI - use of var and inference...ask about this in interview. driver has some comparison issues too. Developer fields added, no validation. JUnit - all tests passing, however, there is only positive tests done on new field(s).

Licence -utility is grand, added to App class but not validated anywhere. AppStoreAPI - minimal code and commented out. Existing code - hierarchy - good work but unnecessary code in EdApp constructor. Developer - no new fields. Driver - commented out and minimal code added. JUnit - 14 tests failed out of 52 - rest passed. Ask questions in interview on tests maybe? AppStoreAPI tests commented out, much more comfortable with pf2 explanations.

No zip file submitted - siobhan sent slack message to query this. Api and driver...little code added/included. Not much done on tests either. Attempt made at app hierarchy but not compiling either. Understood most of what was submitted however, did have issues with list developers by country.

submitting 1st sept evening, interview 9:30 am. Submitted the Employee App for repeat. There were a lot of compile errors in the app and not a huge amount of coding was done in the api or driver. No validation was done on fields in hierarchy. Tests were just the skeleton ones. In interview, Alex was not sure what the super constructor call was doing, the += or the data type of arraylists.

wed 9am submit, zoom interview at 11am

no show - contacted beforehand

good understanding of the code submitted...only attempted sections she understood and completed

Interview - looked like he understood code but did nothing extra 45% with 8/10

good work done on this - briefly looked over it. no tests really though

Interview ok but extra Rating dependency in hierarchy. Rating not written to match

Interview ok but mistake in putting License as class has thrown everything.

mimi repeating three modju pf1 - difficulty explaining for loop for creating clouds, doesn't know

w what a constructor is for

POJO Class (Player) 20%			API (Team) Class 38%			
BASELINE	GOOD	EXC / OUT	BASELINE	GOOD	EXC / OUT	EXC / OUT
Player Boilerplate Generated Code	Player - non boilerplate code - updated toString() etc.	Validation implemented robustly	Team - boilerplate (generated) code)	Team - Listing Methods	Team-Calculation Methods	Team - Other Methods
8	4	8	4	16	12	6

Copy for Players:

POJO Class (Player) 20%		
BASELINE	GOOD	EXC / OUT
Player Boilerplate Generated Code	Player - non boilerplate code - updated toString() etc.	Validation implemented robustly
8	4	8

API (Team) Class 38%			
BASELINE	GOOD	EXC / OUT	EXC / OUT
Team - boilerplate (generated) code)	Team - Listing Methods	Team-Calculation Methods	Team - Other Methods
4	16	12	6

User Experience (UX) (Driver) 22%			
BASELINE	GOOD/EXCELLENT/OUTSTANDING		
Driver - Menu Working and Exiting properly	Driver - Listing Methods	Driver - Calculation Methods	Driver- Other Methods
10	6	4	2

Developer Experience (DX) 20%		
BASELINE	GOOD	EXC / OUT
Standard naming, indentation	Standard naming, indentation	DRY Code - elegance of code
8	6	6

Extra Credit (see spec)

User Experience (UX) (Driver) 22%				Developer Experience (DX) 20%		
BASELINE	GOOD/EXCELLENT/OUTSTANDING			BASELINE	GOOD	EXC / OUT
Driver - Menu Working and Exiting properly	Driver - Listing Methods	Driver - Calculation Methods	Driver- Other Methods	Standard naming, indentation	Standard naming, indentation	DRY Code - elegance of code
10	6	4	2	8	6	6

**Extra Credit
(see spec)**

10