

Kent Crozier, Submission # ____/3

Android Project Self-Evaluation

Instructions for Use

Enter your name and submission number in the header. Complete the following rubric as a self-evaluation of your project for each of your three submissions. Remember that this self/peer/instructor-evaluation document must be submitted with your assignment to receive a grade. The following steps should be completed by each of a) yourself, b) two of your classmates (different people for each of the three submissions), and c) your instructor.

1. Enter a *Score* from zero to three for each row (component) of the rubric, based on the project specifications and point scale descriptions for that component.
2. Multiply the score by the weight for that component and enter the result in the *Score Earned* column. For rubric component rows with a weight greater than one, you may make minor adjustments (i.e. adjustments of less than the weight of that component) to the score earned to reflect your assessment on the degree to which you achieved the component.
3. Enter the sum of the scores earned in the *Total Earned* row at the bottom.
4. Complete the comments section at the bottom.

Your instructor will determine your grade on your third and final submission. Scoring less than 100% on your first two submissions is expected! Use your assessment as well as feedback from your peers and instructor to identify where you should focus your attention when making improvements (correcting defects, adding required components, new features to improve usefulness and UX, code organization, etc.).

Rubric for Android Project

Rubric Component	Point Scale				Score (0-3)		Weight	Score Earned
	3	2	1	0				
<i>General</i>	Exemplary code organization/structure and efficiency	Adequate code organization/ structure and efficiency	Needs improvement in terms of code organization/ structure and efficiency	Inadequate commenting, poor organization/structure and code efficiency	You	2	1	
					P1	3		
					P2	3		3
					Ins.			
<i>User Interface and Event-Handling</i>	All user interface and event-handling requirements are met;	Most requirements are met, but there are a few notable defects	Some requirements are met, but there are numerous defects/errors	Incomplete/does not meet minimum level of performance	You	3	1	
					P1	3		
					P2	3		3

	very minor, or no defects identified				Ins.			
<i>Data Persistence</i>	All data persistence requirements are met; very minor, or no defects identified	Most requirements are met, but there are a few notable defects	Some requirements are met, but there are numerous defects/errors	Incomplete/does not meet minimum level of performance	You	3	2	
					P1	3		
					P2	3		6
					Ins.			
<i>Connectivity</i>	The app performs Internet/web or another acceptable form of connectivity, to a relatively high degree of complexity. The processing is efficient (i.e. a separate thread is used) and application lifecycle is appropriately considered.	The app performs Internet/web or another acceptable form of connectivity, but not to a very high degree of complexity.	The app performs Internet/web or another acceptable form of connectivity at a basic level. There are lifecycle-related issues and/or threading (asynchronous processing) is not properly employed.	The app does not effectively achieve the required connectivity requirements.	You	3	2	
					P1	3		
					P2	3		6
					Ins.			
<i>Usefulness and usability</i>	The app solves a problem, is useful, and the user experience (UX) is compelling. Required techniques (data persistence, connectivity) are used appropriately and are employed usefully within the context of the purpose of the app.	The app solves a problem, though the user experience could use some improvement. Required techniques are mostly used appropriately and usefully.	The usefulness and UX of the app could be improved. Required techniques are not all effectively employed.	The app is incomplete, does not have a well-defined purpose, offers an unintuitive UX, or does not employ the required features effectively.	You	3	1	
					P1	3		
					P2	3		3
					Ins.			
Total Earned (max 21)								

Your Comments

Is there anything you are having trouble with? What do you think you did exceptionally well? What are you aiming to do for your next submission? Is there anything you would like your instructor to give special attention to when reviewing or evaluating your assignment?:

Trouble-wise, there is one potential app-breaking bug in the ViewRouteActivity for the onBackPressed() callback. For runs with numerous points, loading up either the map or data fragment causes a binder exception when clicking the back button to return. It's unclear why this is

occurring - the exception message references only framework code - since reports suggest this happens when sending too large of a parcel and there is no parcel being sent in that activity call.

In any case, that OnBackPressed override is currently disabled in code and no exception is triggered when using the phone's back button. This work-around works great for viewing runs, but will cause the app to go back to the new/edit run screen after editing and existing run or saving a new run. Not the end of the world, but not professional either. I will have to look into this issue going forward.

From an 'okay' standpoint is some of the custom code in classes like MonthlyFragment and JoggrHelper. It was written for functionality and not efficiency, so for that reason I've given myself a 2 when it comes to code organization. I also think I have programmed this app in a very desktop-oriented, resource-intensive way, and have probably run afoul of some mobile paradigms.

On the positive side, the GPS service class works great, and the Parcelable implementation represents a unique solution to one of the problems I stumbled upon - sending a custom data object via an intent. Despite the inefficient code in MonthlyFragment, I was very satisfied with the dynamic tab functionality that will occur as the user accrues runs over different months and years. I also think my approach for calculating warm-up, cool-down, and run times is interesting and unique, though I recognize that it will require some testing to optimize.

I added a lot of little things into the code that aren't really anything on their own, but in the aggregate are a small source of pride. Examples of these sorts of things include ExportType and RunRating enums, disabling the GoogleMap during GPS initialization, and using AsyncTask to write points to the database - things that help round out my programming experience.

Implementing some of the bells and whistles like the heat map and CSV export was a breeze, and I can safely say that I believe I have a fully-functional - albeit small scope - app. For the moment, the only improvements I really want to focus on are some superficial UI elements and design.

Peer Comments

For each submission you need to have *at least* two people – and different people for each of your three submissions -- complete this rubric for the assignment and enter their comments here:

e.g. Reviewer: John B. (submission 1): I think the app is coming along well so far. The layout and screens are well designed already. The connectivity component isn't well defined yet and there are some errors saving data in some cases. I think a settings screen which allows saving of favourite locations would be a big UX improvement in terms of usability.

Reviewer: Flavio M. (Submission 3) - Kent's app is amazing. I would definitely download it and use it if I become an exercise type of guy. The app looks really professional and several coding had to be handled to produce it. He even took the effort to export the data logged through his database is a CSV file which is a nice feature for users who want to work with the data generated through the app. In my personal opinion it was one of the best apps presented in the demos for our section. All features requested in the rubrics were successfully met.

Reviewer: Andrew Xia

Fantastic eye-catching app! This is exactly what I want. UI is well designed and very friendly. The presentation and visualization of data is perfect. This app meets all rubric requirements, and personally, it meets all my needs. Well done.

Instructor Comments