**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** |  | | |  | |  | |
| *General* | 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 | | 2 | |
| P1 | | 2 | 2 | |
| P2 | |  |  | |
| Ins. | |  |  | |
| *User Interface and Event-Handling* | All user interface and event-handling 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 | | 2 | 1 | | 2 | |
| P1 | | 1 | 1 | |
| P2 | |  |  | |
| Ins. | |  |  | |
| *Data Persistence* | 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 | | 0 | 2 | | 0 | |
| P1 | | 1 | 2 | |
| P2 | |  |  | |
| Ins. | |  |  | |
| *Connectivity* | 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 (asynchrounous processing) is not properly employed. | The app does not effectively achieve the required connectivity requirements. | You | | 1 | 2 | | 1 | |
| P1 | | 1 | 2 | |
| P2 | |  |  | |
| Ins. | |  |  | |
| *Usefulness and usability* | 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 | | 2 | 1 | | 2 | |
| P1 | | 1 | 1 | |
| P2 | |  |  | |
| Ins. | |  |  | |
| **Total Earned (max 21)** | | | | | | | | | | 7 | |

**App Description**

I have, since it’s inception, changed the fundamental way that my app will work. I still have all of the functionality there, and I will be re-implementing it into this new app, but as of right now it’s not working properly.

A user will load into (on the first launch) a SignIn Activity, where they can now sign in using Google Sign-in (soon will be using Email Authentication as well), after they sign in, if they have not created an account, they will be asked to do so, entering their name, birthday, an initial first “Long Term Goal” for the family, it’s required point value, any additional family members who they would like to “add” and the ability to add “rewards” that the family can redeem into the app.

When a family members name is clicked on, it will go to their “Chore To-Do List” here, they will see a list of all of the available chores that they have to them for completion. If they click that a chore has been completed, they will need a password from the account owner for the points to be added to their profile.

When a chore has been completed, the point value for the chore will be added to the individuals profile, the family’s running total, as well as progress toward the long term goal.

**Your Comments**

After spending far too much time thinking about it, I wanted to make the app more “people-first” as opposed to “chore-first” like my first inception of the app. I thought about additional functionality and features that I could add to the app, but unfortunately it was not as easy of an undertaking as I had originally imagined. I redid everything, from the implementation of fragments, to a completely new UI design. I will still be implementing the old features that I had, but I will be simplifying things as I feel like right now what I’ve been trying to implement has taken way too much time for very little benefit.

**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: John Distefano-Bell. (submission 2): “There is not much functionality besides signing in, but it is obvious a lot of care had gone into creating an attractive UI with obvious intuitive UX design. It is not far from being an app one would see in an app store. I feel like the majority of time went into designing the interface, and I expect the functionality will remain the focus of the project from here on.”

**Instructor Comments**