Team #29

Team Member Name	PID	UCSD Email ID
Matthias Smyrl	A13014876	msmyrl@ucsd.edu
Stephanie Mitchener	A12861347	smitchen@ucsd.edu
Joel Loo	A12623303	jcloo@ucsd.edu
Amanda Moffitt	A14210917	amoffitt@ucsd.edu
Inga Klassy	A13534486	iklassy@ucsd.edu
Muyao Wu	A11735974	muw006@ucsd.edu

Milestone 1 - Delivery Phase

Refer the grading rubric (link: https://csemoodle3.ucsd.edu/mod/page/view.php?id=1326) for more details.

Here we are following a sample repository (github.com/CSE-110-Winter-2018/Default-Repository) to provide sample links for code snippets, branch etc. as per requirements.

Remember to push this document in the *docs* folder of your project in the branch "milestone1_delivery", like: github.com/CSE-110-Winter-2018/Default-Repository/blob/milestone1_delivery/docs/Milestone1_DeliveryPhase.pdf

Software design

1. Checkout a new branch "milestone1_delivery" , push the final code to it and provide a link to the code folder of your team project repository on Github *https://github.com/CSE-110-Winter-2018/cse-110-team-project-team-29/tree/milestone1_delivery*

2. Provide an example or two from your code (best if you insert relevant links to code snippets) where SRP & DRY were demonstrated.

The Song class' single responsibility is to store data concerning a song's title, and where/when it got played. Thus, the song only stores that data, and has accessors and modifiers for retrieving and changing that data. In addition, there are no redundancies in the getters or setters so this class also demonstrates DRY.
https://github.com/CSE-110-Winter-2018/cse-110-team-project-team-29/blob/milestone1_delivery/app/src/main/java/com/example/stephanie/flashback_music/Song.java

The Album class is similar, in that its only responsibility is to hold a list of songs. So the Album itself holds none of the data for the songs, only the songs themselves, which is reflected in its accessors and modifiers. It demonstrates DRY as well because there is only what is necessary in those methods to complete their function, and nothing else, thus no repetition.
https://github.com/CSE-110-Winter-2018/cse-110-team-project-team-29/blob/milestone1_delivery/app/src/main/java/com/example/stephanie/flashback_music/Album.java

Testing

Link(s) to all the JUnit and/or Espresso tests and logs of untestable non-trivial methods.

https://github.com/CSE-110-Winter-2018/cse-110-team-project-team-29/tree/milestone1_delivery/app/src/androidTest/java

https://github.com/CSE-110-Winter-2018/cse-110-team-project-team-29/blob/milestone1_delivery/app/src/androidTest/java/com/example/stephanie/flashback_music/MainActivityTest_displaytest.java

https://github.com/CSE-110-Winter-2018/cse-110-team-project-team-29/blob/milestone1_delivery/app/src/androidTest/java/com/example/stephanie/flashback_music/MainActivity_PopulationTest.java

ZenHub

1. Insert a valid link to your Zenhub board covering all the required points from the rubric

https://app.zenhub.com/workspace/o/cse-110-winter-2018/cse-110-team-project-team-29/boards?milestones=Iteration%202%232018-02-18,Iteration%201%232018-02-12&repos=119205445

2. Insert a valid link to the burndown chart

Iteration 1:

https://app.zenhub.com/workspace/o/cse-110-winter-2018/cse-110-team-project-team-29/reports?report=burndown&milestoneId=3077091&showPRs=false

Iteration 2:

https://app.zenhub.com/workspace/o/cse-110-winter-2018/cse-110-team-project-team-29/reports?report=burndown&milestoneId=3077102&showPRs=false

GitHub

1. Insert a valid link to the contribution chart of all the contributors

https://github.com/CSE-110-Winter-2018/cse-110-team-project-team-29/graphs/contributors

2. Checkout a new branch “milestone1_delivery” , push the final code to it and provide a link to it for final code review

https://github.com/CSE-110-Winter-2018/cse-110-team-project-team-29/tree/milestone1_delivery