Team #7

Team Member Name	PID	UCSD Email ID
Mikhail Boulgakov	A13543506	mboulgak@ucsd.edu
Simar Chhabra	A13891316	sichhabr@ucsd.edu
Eldon Tay	A12963052	ektay@ucsd.edu
Amritansh Gupta	A13831681	amg080@ucsd.edu
Nirha Patel	A13695370	nrp001@ucsd.edu
Val Gonzalez	A13188505	vgd001@ucsd.edu

Milestone 1 - Planning Phase

Risk Analysis

1.
Risk: Not having standup meetings with everyone present
Description: Our team hasn't decided on when to have standup meetings. An easy way to do them would be after every lecture, but not everyone is able to attend every lecture.
Severity: High
Resolution: We will have meetings immediately after every class.
Status: Resolved

2.
Risk: Not having a responsive communication channel
Description: People do not respond to messages regarding group.
Severity: High
Resolution: Moved communication to Slack. We now use @everyone to make sure everyone is notified. In case that doesn't work, we got the phone numbers of our group members.
Status: Resolved

3.
Risk: Not having enough in-person work sessions
Description: Our team hasn't established ongoing all-hands meeting times to work on the project.
Severity: Medium
Resolution: (?) Have everyone fill out a poll to identify common times, and have everyone reserve those times to meet as a group and work.
Status: In Progress

4.
Risk: Not splitting software development work equally
Description: Having people with more software dev. experience carry the bulk of the software dev. work.
Severity: Low
Resolution: Identify the experience level of everyone in the team. If possible, pair-off people with different levels of experience. Or, for those of us who have less experience, give tasks that are lower in priority (not in the MVP) or run for a longer time to account for learning time. Access this regularly, follow suggestions from Week 4 Monday lecture.
Status: Unresolved

5.
Risk: Not enough experience working on Android
Description: As a team we all have zero to very little experience working in Android
Severity: Medium
Resolution: Allocating time to learn how to code with Android at a proficient level. Taking this time into account when estimating time for stories.
Status: In Progress

6.
Risk: General lethargy and lack of will to work
Description: As a team there will be times where we do not want to complete any work when we need to
Severity: Low
Resolution: We will try to encourage each other and be responsible, but it will ultimately come down to each individual to find their own motivation to complete their own work
Status: Unresolved

Velocity Calculation

Our starting **velocity is 0.5**. We do not have a lot of experience working in a large group on the same project. We also do not have experience in Android app development aside from what is gone over in labs. At the same time, we have all completed CSE100 and the other CSE classes (such as 15L which taught us to use git) so we have experience working on relatively large projects and consider our team resourceful in learning skills we are unfamiliar with.

Planning Poker

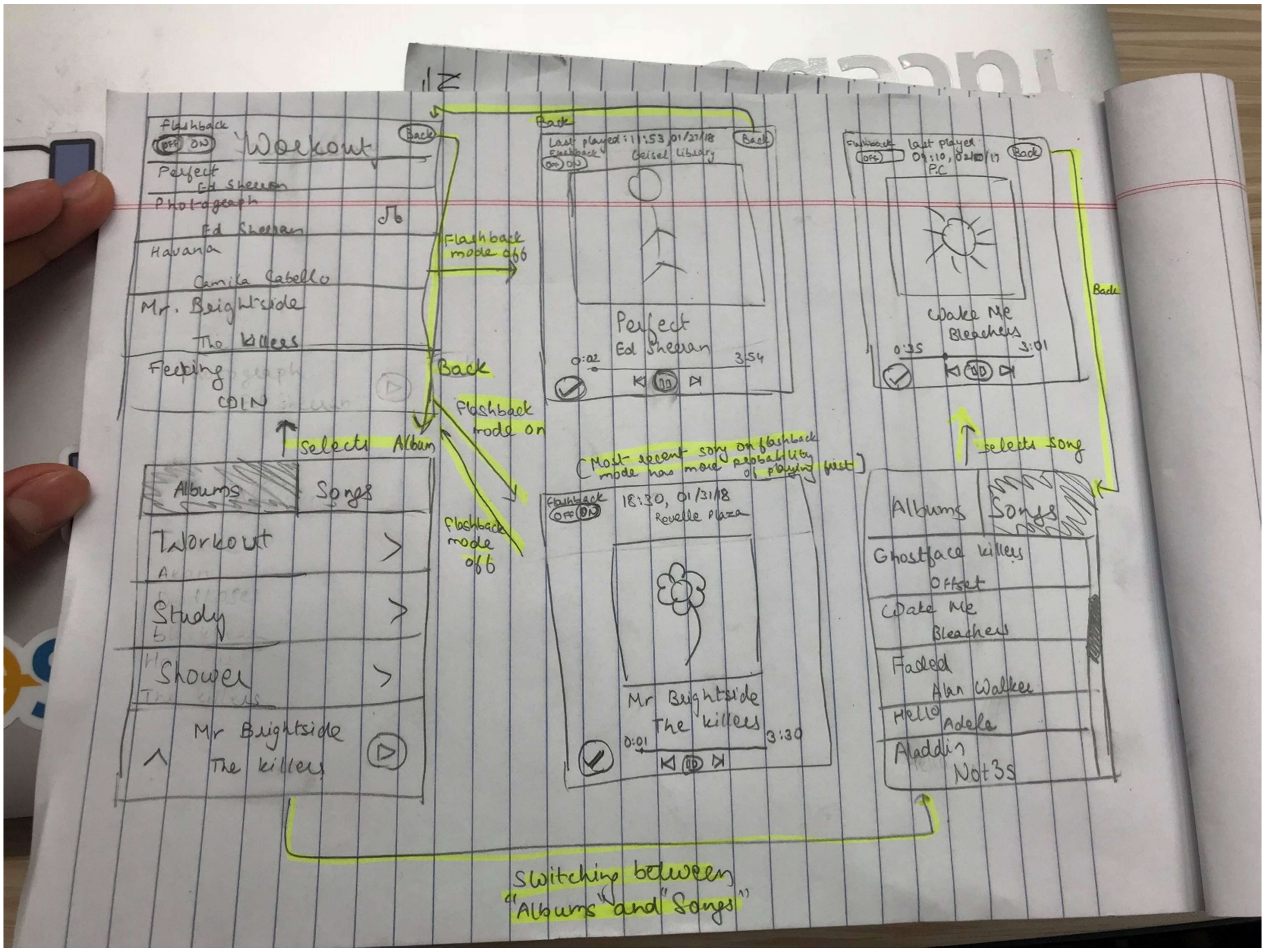
Task	Val	Michael	Simar	Eldon	Amrintash	Nirha	Agreed	False Assumptions Uncovered
Record location	4hr	1hr	1	1.5	1	1hr	1hr	Android has API that gives you location in one call
Store the time a song was played	2hr	1hr	0.5	1	0.5	1hr	1hr	NO assumptions uncovered
Create a song class to store songs' info	2hr	3hr	2	3	3	2hr	3hr	NO assumptions uncovered
Display songs in album	1hr	2hr	2.5	2	3	3	3hr	Had different ideas of how UI worked. Drew up a picture
Toggle between albums display and songs display	1hr	1hr	1	1	1	1	1	NO assumptions uncovered
Navigate back to music library from current song display	2hr	1hr	0.5	2	0.25	1	0.5hr	NO assumptions uncovered
Database to store songs and their information	7hr	4hr	7	7	8	6	7hr	Need a way to traverse the list of songs based on the data stored in them (location they were played in, day of the week). Can be written in SQL, or use a ordered data structure
Display albums	1hr	1	1	1	1	2	1hr	NO assumptions uncovered
Display songs	1hr	1	1	0.5	1	2	1	Once "Display Albums" is done, display songs will be easy
Play selected song	1hr	2	1	0.5	1	1	1	NO assumptions uncovered
Navigate back to music library from current song display	2hr	1	1	0.5	1	0.5	0.5	NO assumptions uncovered
Design a sprite for a favorited track	1hr	1	0.5	0.2	0.2	0.25	0.25	We can keep it simple, use clipart
Flag showing that a track is disliked	1hr	1	0.5	0.75	0.2	0.25	0.25	NO assumptions uncovered
Icon for neutral track	0.5	0.25	0.5	0.25	0.2	0.25	0.25	NO assumptions uncovered
Keeping track of most recent music	0.5	0.25	0.5	2	2	1	0.25	You do not need to keep a list. We will just update a song when it is played.
Calculate chances of playing most recent song	0.5	2	0.25	0.2	2	0.5	0.25	It is just a lookup
Generate list of songs ordered by	3	3	2	3	2	2	3	NO assumptions uncovered

greatest priority								
UI element for flashback mode	5	4	3	4	3	2	3	UI element is just a static window (basically), so it should not be hard too implement.
Flag for current mode	0.5	0.25	0.25	0.25	0.1	1	0.25	NO assumptions uncovered
Have a flag that signifies if a track is favorited	0.5	0.25	1	0.5	0.25	0.25	0.25	No assumptions uncovered
Play song with highest priority in FBM	1.5	1	1	2	1.5	1	1	No assumptions uncovered
Play an album	1	1.5	0.5	0.5	1	0.5	0.5	No assumptions uncovered
Test select and play songs	1.5	2	2	4	0.5	1	2	Tough to come up with an assortment of edge cases
Test flashback toggle	0.5	1	1	0.75	0.5	0.5	0.5	No assumptions uncovered
Test favorite functionality	0.5	0.5	1	0.5	1	0.5	0.5	""
Test display of song information	1	1	2	1.5	0.75	1	1	""
Test prioritize recently played song	0.5	2	2	2.5	2	3	2	Tough to come up with an assortment of edge cases
Test dislike functionality	0.25	0.5	1	0.75	0.75	0.5	0.5	""
Test neutral functionality	0.25	0.5	1	0.75	0.75	0.5	0.5	""
Test progress being saved	2	1.5	1	1.5	2	1	1	""
Scenario Based System Tests	10	5	6	3	5	9	5	A lot of the scenarios are actually very similar, but we still need to do each of them independently.

URL of ZenHub Project:

https://app.zenhub.com/workspace/o/cse-110-winter-2018/cse-110-team-project-team-7/boards?repos=119468919

User Interface Progressions/Screens (Wireframes)


flashback
OFF ON
Workout
Back
Perfect
Ed Sheeran
Photograph
Ed Sheeran
Havana
Camila Cabello
Mr. Brightside
The Killers
Feeling
COIN
selects Album
Albums
Songs
Workout
Study
Shower
Mr Brightside
The killers
Flashback mode off
Back
Flashback mode on
Flashback mode off
Back
Last played: 11:53 01/27/18
Geisel Library
Flashback
OFF ON
Perfect
Ed Sheeran
0:02
3:54
[Most recent song on flashback mode has more probability of playing first]
Flashback
OFF ON
18:30, 01/31/18
Revelle Plaza
Mr Brightside
The killers
0:01
3:30
Flashback
OFF
last played: 09:10, 02/10/17
PC
Back
Wake Me
Bleachers
0:35
3:01
Back
selects song
Albums
Songs
Ghostface killers
Offset
Wake Me
Bleachers
Faded
Alan Walker
Hello
Adele
Aladdin
Not3s
Switching between "Albums" and "Songs"