Android Training Project

PROBLEM STATEMENT:

Build a music player with the following functionalities:
1. A Splash screen (gradient background and app logo in center)
2. A Navigation drawer with app logo section at the top along with links to 'All Songs', 'Favorites', 'Settings' and 'About Us'.
3. An 'All songs' screen (where of list all the tracks fetched from offline storage are displayed and user can sort the tracks by name or recently added). This will the home screen of the app.
4. The app should be able to fetch and play .mp3 and .wav files.
5. A 'Favorites' screen (where list of all the favorite songs are displayed)
6. A 'Settings' screen (where the 'Shake to change song' feature can be enabled or disabled)
7. An 'About us' screen (where we will display information about the app developer and the app version)
8. A 'Now playing' screen with following features:
    a. Track title and track artist
    b. Play / Pause button
    c. Next button
    d. Previous button
    e. Shuffle button
    f. Loop button
    g. Seek bar
    h. Mark track as favorite or unfavorite it
    i. Third party visualiser in upper half background
    j. A 'Back to list' button in the header which should take the user to the screen he came from (kind of like back button behaviour).
    k. Shake to change song
9. A 'Now playing' bar at the bottom with name of the track playing and play or pause feature. This would appear if the user has moved from 'Now playing' screen to 'All songs' screen or 'Favorites' screen without pausing the track.
10. Background play. The app will continue playing the track if the app gets closed (not killed) without the music being paused.
11. A notification saying "A track is playing in the background" only if the app gets closed (not killed) without the music being paused.

**The final music player should look like this:** Echo music player app demo

Icons and Images you would need: App Resources

Primary Color scheme: #9b2a58, #00032a

SPECIFICATIONS:

Splash screen

This would be the first screen that gets displayed when a user opens the app. It would have a linear gradient background and the app logo in the center of the screen. The user will see the screen for 1 second and then the home screen will pop up.
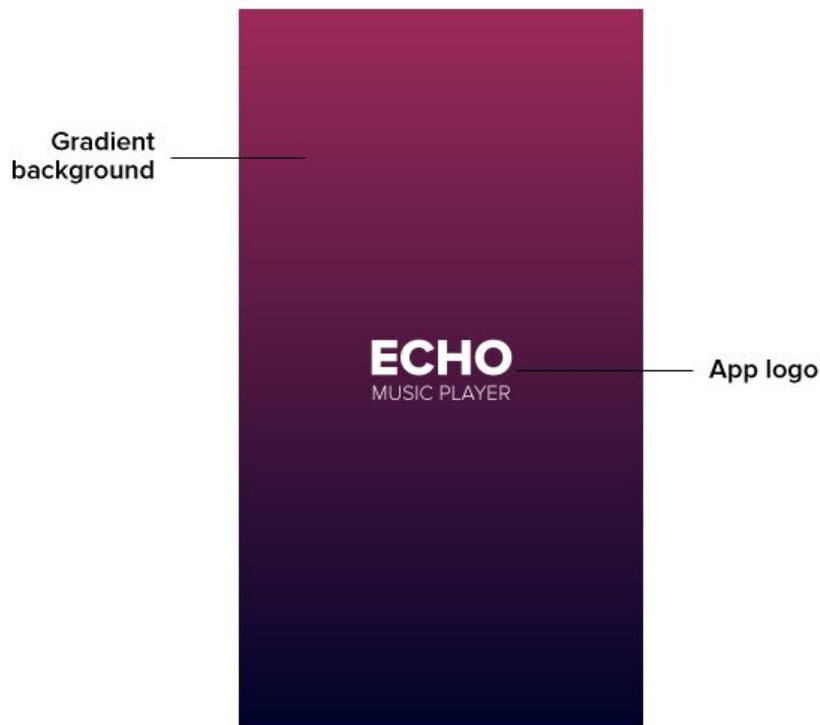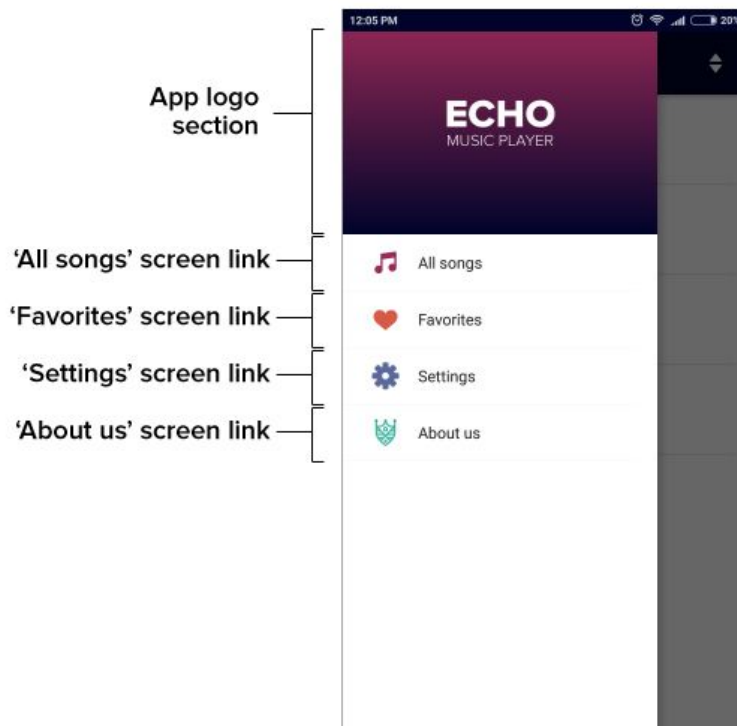

Resources:
Gradient hex-color codes: #9b2a58, #00032a
App logo: echo_logo

Navigation drawer

The navigation drawer is needed so that the user can navigate to different screens within the app. The user should be able to access the navigation drawer on all the app screens by clicking hamburger button on the left side of the header or by swiping right from the left edge. It would have an app logo section at the top with gradient in the background and app logo in center (similar to Splash screen but smaller). Below the app logo section, there would links to 'All songs', 'Favorites', 'Settings' and 'About us' screen in a list style. Clicking on any link will open the corresponding screen.

Resources:

Gradient hex-color codes: #9b2a58, #00032a

App logo: echo_logo

'All songs' icon: navigation all songs

'All songs' color: #212121

'Favorites' icon: navigation favorites

'Favorites' color: #212121

'Settings' icon: navigation settings

'Settings' color: #212121

'About us' Icon: navigation about us

'About us' color: #212121

'All songs' screen

'All songs' screen will be the home screen of the app, meaning when the app is launched, user will see the 'All songs' screen after the splash screen.

Once the app is launched, all the tracks (.mp3 and .wav) will be fetched from the offline storage and will be displayed on the 'All songs' screen in a list view. By default, all the tracks would be sorted by name. For each track in the list, the user should see the title of the track and the artist of the track. If there is no title, the track name will be displayed in place of the title. If there is no artist, 'unknown' will be displayed in place of the artist. There would also be a line separating each track.

When the user clicks on any track, the 'Now playing' screen should open and the track should start to play.
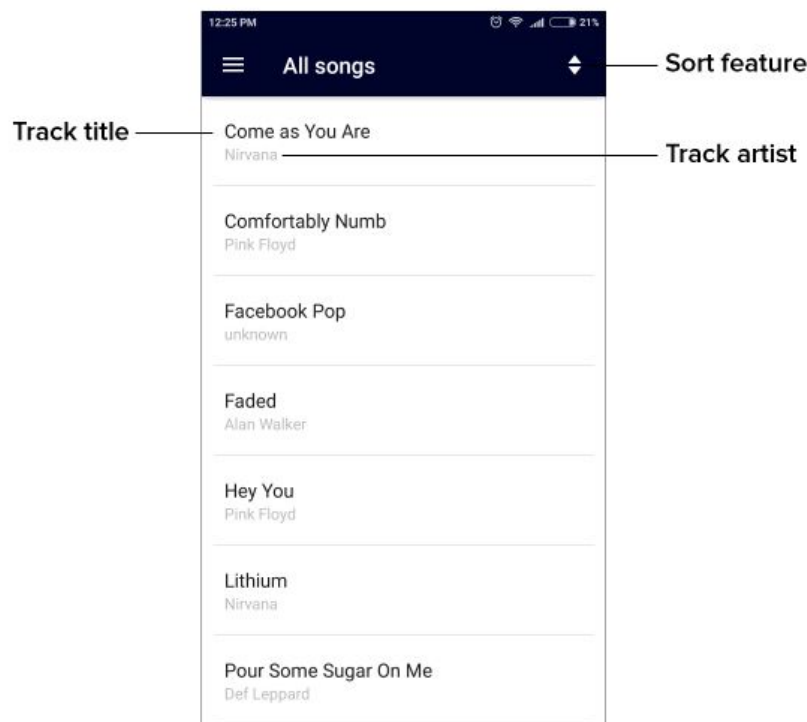
Resources:
Track title hex color code: #212121
Track artist hex color code: #bdbdbd

Sort feature:

The 'All songs' screen will have a Sort feature. Using the Sort feature, the user should be able to sort the tracks by name or by recently added. A sort icon will be there on the right side of the header, when clicked on, then a dropdown will appear with two options:

1. By recently added
2. By name

If the user clicks on 'By recently added', all the tracks would get sorted by recency with most recently added track on the top. 'By name', all the tracks would get sorted in alphabetical order (or in this order: track title starting with symbols, track title starting with numbers and then track title starting with letters).
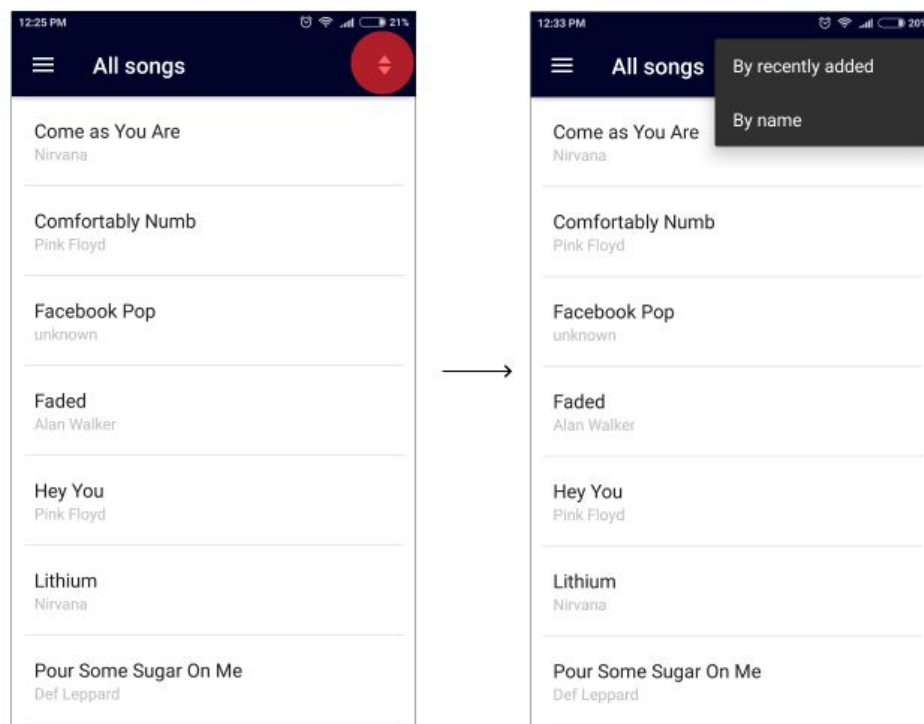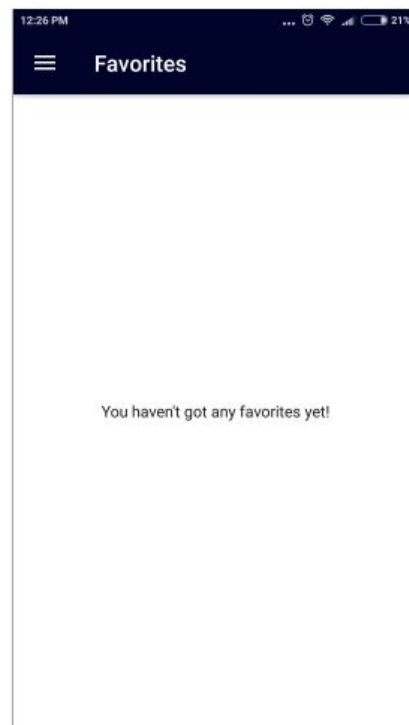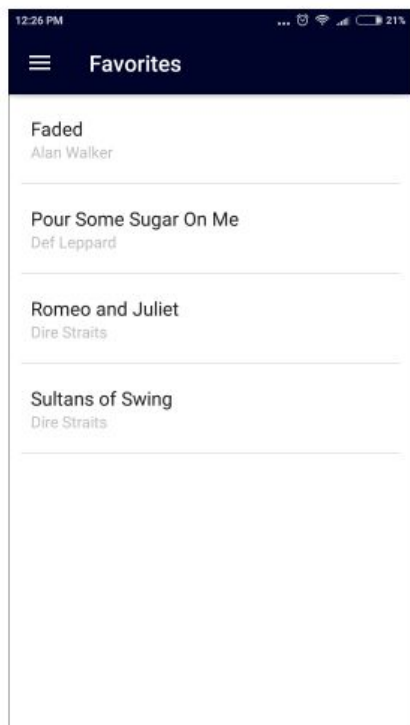

Resources:

Sort icon: sort icon

'Favorites' screen

The 'Favorites' screen resembles the 'All songs' screen. It will display all the tracks that have been marked favorite by the user. The user would be able to mark a track favorite or unfavorite it only on the 'Now playing' screen. Ensure that if the user marks a track favorite and then later deletes it from the offline storage, the track shouldn't appear on the 'Favorites' screen.

If the user has no favorite tracks, there would be a message in the center on the screen saying, "You haven't got any favorites yet!".
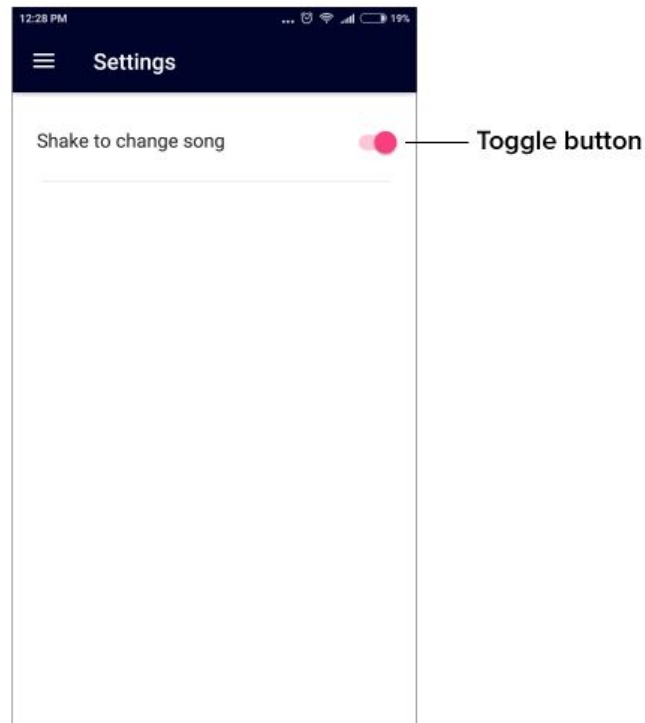

Resources:
Track title hex-color code: #212121
Track artist hex-color code: #bdbdbd

'Settings' screen

The 'Settings' screen will have the option to enable or disable 'Shake to change song' feature using the toggle button. The default state of 'Shake to change song' feature would be 'disabled'. If the user enables the 'Shake to change song' feature, the app should remember it, so when the user kills the app and launches it again, the 'Shake to change song' should stay enabled.
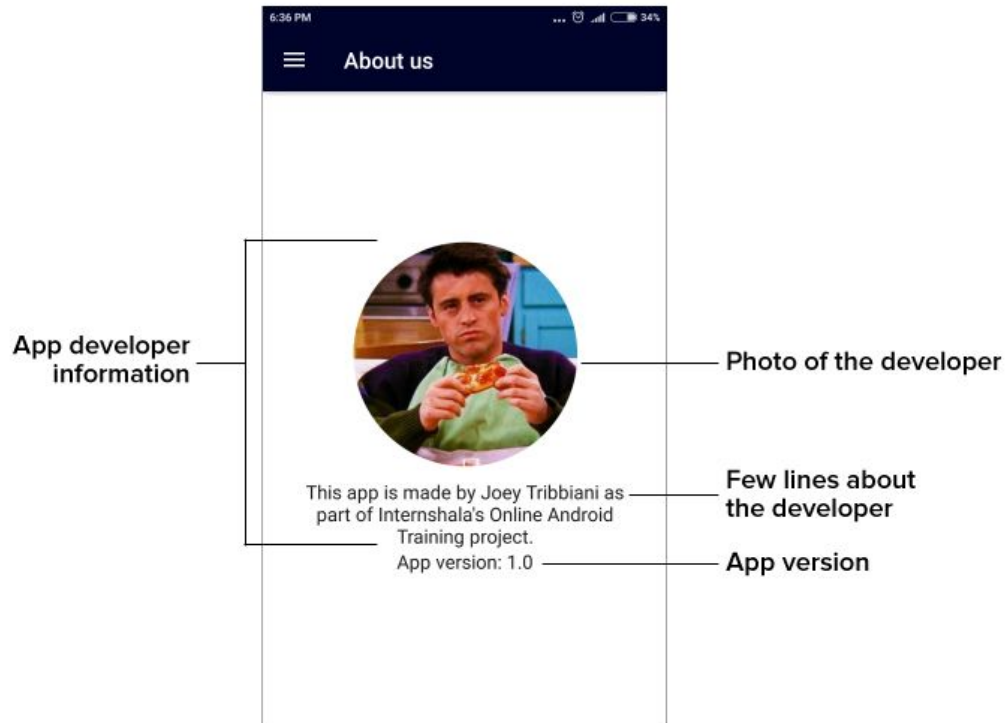

Resources:

'Shake to change song' hex-color code: #212121

'About us' screen

The 'About us' screen will display the information about the app developer and the app version. As part of the app developer information, there would developer's photograph and a few lines about him/her.

Resources:

Joey's photo: joey

'Now playing' screen

The 'Now playing' screen is the actual music player in the app. It houses all the elements which let the user interact with the current playing track or the next/previous track. It will open when the user clicks on a track on the 'All songs' screen or the 'Favorites' screen. It would have following features and functionalities:

1. **Track title and track artist**
   The track title and track artist would be displayed in the center of bottom half of the screen. If the track has no title, track name will be displayed. If the track has no artist, 'unknown' will be displayed in its place. If the track title or name is too long, an ellipsis will be added to the title or name.

2. **Play/Pause button**
   As the names suggest, this button will be used to play or pause a track. When a track is playing in the app, this button will become Pause button. When a track is paused in the app, this button will become Play button. When a user clicks on the Pause button, the track will pause and when the Play button is clicked, the track will start playing from the same place where it was paused.

3. **Next button**
   This button will allow the user to play the next track. When the Next button is clicked, the consecutive track in the list (from which the 'Now playing' screen was triggered) will start playing.

4. **Previous button**
   This button will allow the user to play the previous track. When the Previous button is clicked, the consecutive track in the list (from which the 'Now playing' screen was triggered) will start playing.

5. **Shuffle button**
   The default state of Shuffle button would be 'switched off' (white). When the Shuffle button is toggled (switched on) by the user, the button would turn yellow indicating that Shuffle feature has been turned on. When the Shuffle feature is on, the player would randomly choose a track (different from the one currently playing) from the list (from which the 'Now playing' screen was triggered) and play it when the next button is clicked or when the current track ends. When the Shuffle button is toggled again (switched off), the button would again turn white indicating that Shuffle feature has been turned off. When Shuffle feature is white or switched off, the player would play the consecutive track in the list when the next button is clicked or when the current track ends.
   If the user switches the Shuffle feature on, the app should remember it, so when the user kills the app and launches it again, the Shuffle feature stays on.
   If the Shuffle feature is on and the Loop feature is turned on, Shuffle feature would move back to its default state.
   The Shuffle feature and the Loop feature can't be in 'switched on' (yellow) state simultaneously.

6. **Loop button**
   The default state of the Loop button would be 'switched off' (white). When the Loop button is toggled (switched on) by the user, the button would turn yellow indicating that Loop feature has been turned on. When the loop feature is on, the player would play the same track again when the track ends. The loop button won't affect the

behaviour of Next button. When the Loop button is toggled again (switched off), the button would again turn white indicating that Loop feature has been turned off.

If the user switches the Loop feature on, the app should remember it, so when the user kills the app and launches it again, the Loop feature stays on.

If the Loop feature is on and the Shuffle feature is turned on, Loop feature would move back to its default state.

The Loop feature and the Shuffle feature can't be in 'switched on' (yellow) state simultaneously.

7. **Seek bar**

This screen features a *seek-bar* which displays the track progress throughout the track's lifetime.A user can click on the seekbar to skip in between the track or simply to drag the controller to reach a certain part of the track.

8. **'Mark as favourite' button**

Clicking this button adds the current track to the favorites list, the button then turns red indicating that the track has been added to the favorites list. A toast message is displayed on the screen saying "Added to favorites".

Clicking this button again will remove the track from the favourites list, the button then turns back to white indicating that the track has been removed from the favorites list. A toast message is displayed on the screen saying "Removed from favorites".

The default state of the 'Mark as favorite' button is white, that means, initially there would be no tracks in the favorites list.

9. **Third party visualiser**

The 'Now playing' screen would have a 4 bar visualiser in the upper half of the screen. As expected, the visualiser would move in the rhythm of the music. The visualiser's motion would be volume sensitive meaning that if you turn down the volume, the visualiser will also tone down its motion and vice-versa. The visualiser should start moving once a track is played and should stop moving when a track is paused.

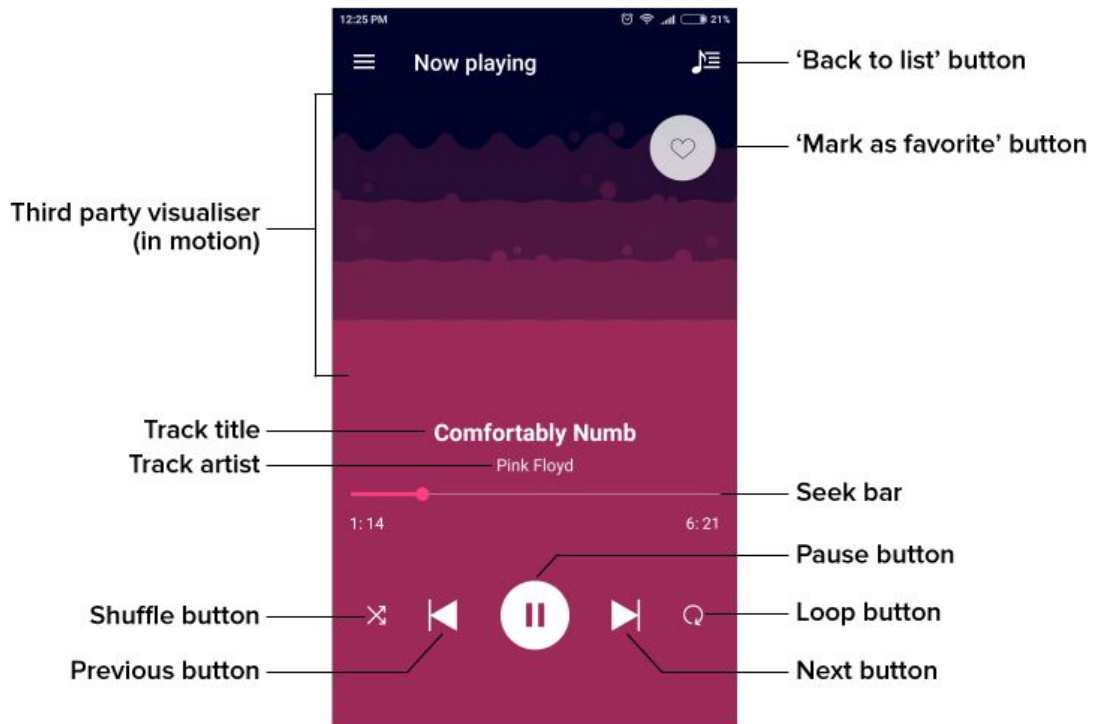10. **'Back to list' button**

The 'Back to list' button would take the user to the screen he came from. For ex: if a user clicked on a track on the 'All songs' screen and lands on the 'Now playing' screen, the 'Back to list' button should take the user back to the 'All songs' screen. This button would be placed in the right side of the header.
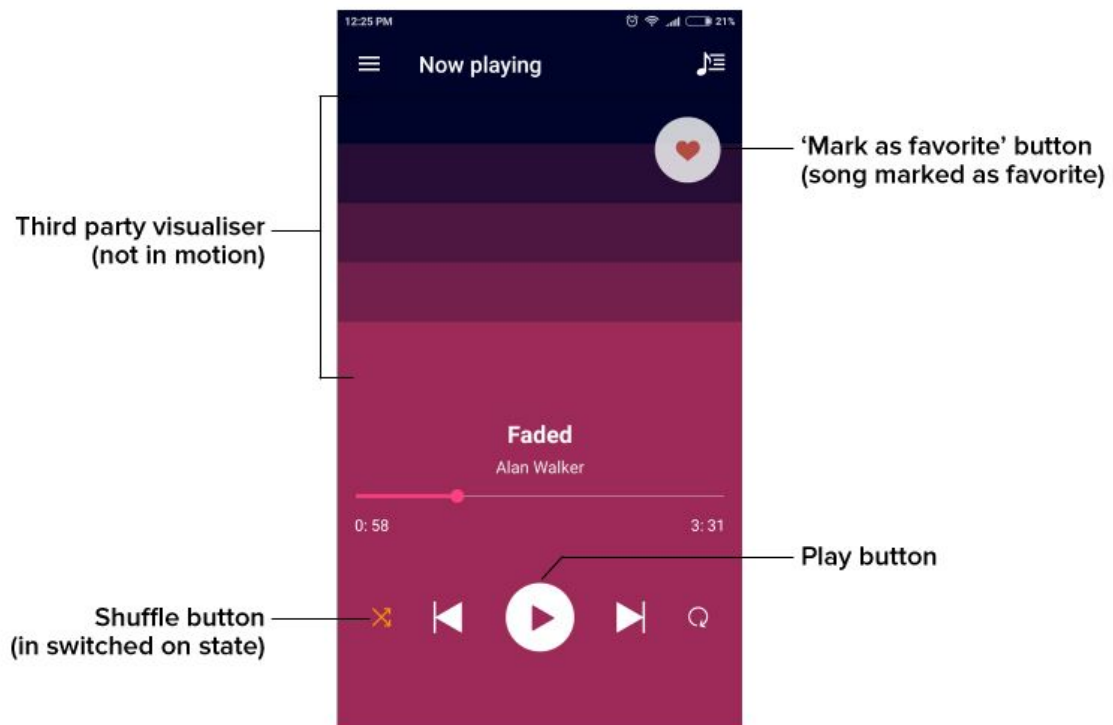
11. **Shake to change song**

As the name suggests, this feature would allow the user to change the track just by shaking his/her phone. We would use the accelerometer on the mobile phones to make this feature work.

Now playing screen (track playing)

Third party visualiser (in motion)

'Back to list' button

'Mark as favorite' button

Track title — **Comfortably Numb**
Track artist — Pink Floyd

Seek bar

Pause button

Shuffle button
Previous button

Loop button
Next button

Now playing screen (track paused)

Third party visualiser (not in motion)

'Mark as favorite' button (song marked as favorite)

Faded
Alan Walker

Play button

Shuffle button (in switched on state)

Resources:

Visualiser hex color codes (top to bottom): #270d36, #4e1741, #74204d, #9d2a58

Background hex color code: #00032a

'Back to List' icon: [back to list](#)

Circle for 'Mark as Favorite': [white circle](#)

Filled heart for 'Mark as Favorite': [favorite on](#)

Unfilled heart for 'Mark as Favorite': [favorite off](#)

Play button: [play icon](#)

Pause button: [pause icon](#)

Next button: [play next](#)

Previous button: [play previous](#)

Shuffle white: [shuffle white](#)

Shuffle yellow: [shuffle](#)

Loop white: [loop white](#)

Loop yellow: [loop](#)

'Now playing' bar

The 'All songs' screen and the 'Favorites' screen will have a 'Now playing' bar at the bottom if the app is playing a track. This bar would display the title of the track playing and play or pause feature. It would also have an image of some equaliser bars in the left side and a static text above the track title name saying "Now playing". If the track title is too long to be displayed in one line, an ellipsis would appear in the track title. The Play/Pause button would work same as it did on the 'Now playing' screen.
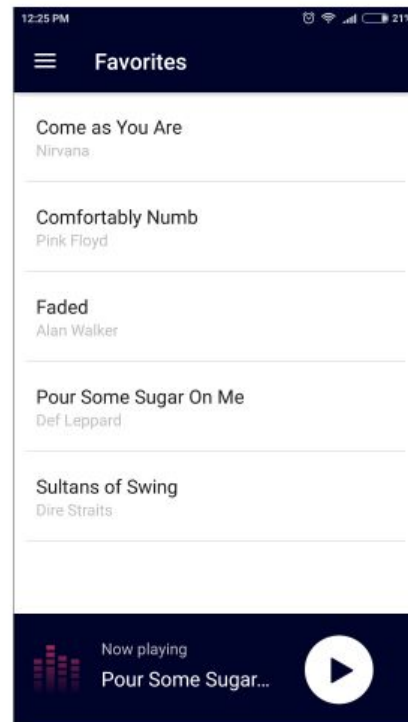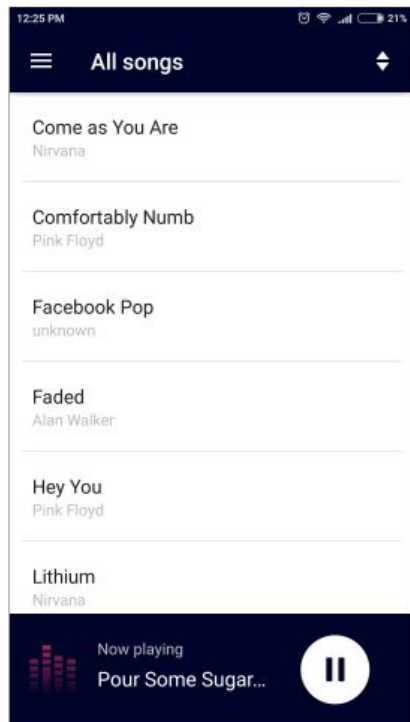
Resources:
Equaliser bar image: now_playing_bar_eq_image

Background play feature

Just like any other music player app, this app will allow the user to listen to music even when the app is running in the background (when a user plays a track in the app and switches to a different app from notification bar or recent apps section or simply goes to the home screen, the app will keep running in the background and keep playing the track.)

If there is an incoming/outgoing call while the app is playing a track, the app should pause the track. As it is a basic music player, the app wouldn't pause a track for any other case. It means that if user starts another music player or video player or any such app while our app is playing a track in the background, our app wouldn't pause the track and the other app can play music over our app.

Notification

With the background play feature in the app, the user can play tracks in the background, but he would have no way to figure out which app is playing the track (except for opening and checking all the recent apps). So the app needs to have a notification indicating that this particular app is playing music in the background.

The notification would say "A track is playing in the background". This notification would pop up in the notifications drawer every time the app plays any track in the background. Ensure that this notification shouldn't in the notifications drawer when the app is not playing a track in the background.

Also, the notification would be there only when the app is playing a track in the background and it won't be there when the user is interacting with the app (i.e. if he/she is on any of the screens of the app).
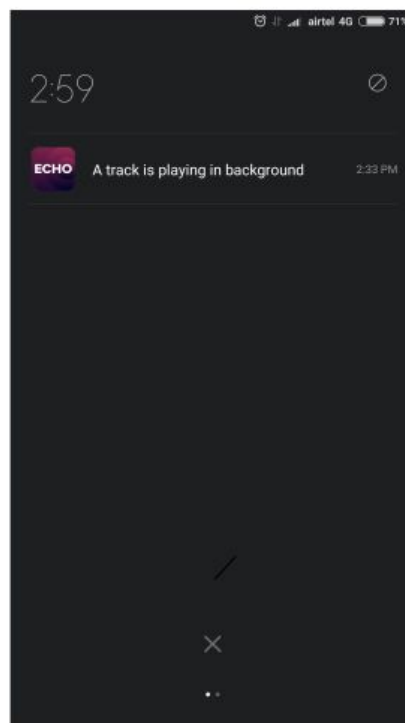
Submission

Please submit the zip file of the complete project folder. Before submitting the assignment make sure you have all the things in place.

How to create a zip file: **Go to Android Studio and click File -> Export to Zip File**. Now upload the created zip file. If the size of the zip file reaches to more than 20Mb, then please upload the zip file to Google Drive and then paste the sharing code into a text document and upload the zip file of this document to progress tracker.

Evaluation

Your project will be evaluated on three grounds:

1. **Application's User Interface(40% weightage)**: Make sure all the UI components taught and asked to make are present and the user is able to view all of them.

2. **Application's Functionality(50% weightage)**: In accordance with the above, you will be marked for the functionality of each UI component present in the application. All the functions in the app should be working properly. You'll lose some percentage of marks if the functionality is faulty. (For e.g. if loop is enabled and music is changed to next and the same song does not play, then this is erroneous and marks will be deducted).

3. **Code(10% weightage)**: The code should be properly formatted(*Hint: Use Ctrl + Alt + L (Windows) and Command + Option + L (Mac) for formatting the code*). All the packages should be made and all files placed in proper packages and please name the files with sensible names relating to their functionality.

Note: Partial marks will be awarded if a component is coded but not attached to the UI properly.

All the best!