



PROJECT SPECIFICATION

Musical Structure App**App Design**

CRITERIA	MEETS SPECIFICATIONS
Suitability	The app's structure is suitable for a music player app. A similarly structured app which focuses on audiobooks, podcasts, or other audio media is also acceptable.
Clarity	The purpose of each activity is easy for a user to understand through the UI design and feature labeling.
Number of Activities	The app contains 2 to 6 activities

Layout

CRITERIA	MEETS SPECIFICATIONS
Structure	The app contains multiple activities, each labelled, which together make a cohesive music app.
Labelling	Features in the app are clearly defined either by labelling or images. For example, a button to play music could use a universally recognized triangular "Play" symbol or could

CRITERIA	a universally recognized triangular "Play" symbol or could have the text label "Play". MEETS SPECIFICATIONS
Buttons	Each activity contains button(s) which link it to other activities a user should be able to reach from that activity. For instance, a 'Library' activity might contain a button to move to the 'Now Playing' activity.
Layout Best Practices	The code adheres to all of the following best practices: <ul style="list-style-type: none">• Text sizes are defined in sp• Lengths are defined in dp• Padding and margin is used appropriately, such that the views are not crammed up against each other.

Functionality

CRITERIA	MEETS SPECIFICATIONS
Runtime Errors	The code runs without errors
OnClickListeners	Each button's behavior is determined by an OnClickListener in the Java code rather than by the <code>android:onClick</code> attribute in the XML Layout.
Intents	Each button properly opens the intended activity using an explicit Intent.

CRITERIA	MEETS SPECIFICATIONS
Custom Adapter	App uses a custom adapter to populate the layout with views based on instances of the custom class.
Data Storage and Retrieval	<p>Information about instances of the custom class are stored in an appropriate data structure (e.g. ArrayList, Array).</p> <p>When the information needs to be displayed, it is efficiently retrieved (e.g. Looping).</p>
Custom Class	<p>Data about each song (or equivalent audio media such as podcast episode) should be stored in a custom class that contains at least 2 states (e.g. Song Name, Artist Name)</p> <p>If images are included (e.g. Album Art), they are stored as drawables. All drawables are stored at multiple densities. Images are not required.</p>

Code Quality

CRITERIA	MEETS SPECIFICATIONS
Code Formatting	<p>The code is properly formatted:</p> <ul style="list-style-type: none"> • No unnecessary blank lines • No unused variables or methods • No commented out code

CRITERIA	The code also has proper indentation when defining variables and methods.
Readability	Code is easily readable so that a fellow programmer can understand the purpose of the app.
Naming Conventions	All variables, methods, and resource IDs are descriptively named so that another developer reading the code can easily understand their function.