1. Record your reasons for implementing the solution the way you did, the struggles you faced, and the problems you overcame.

* I faced major difficulty during player development because I have never created such a type of app before.
* To implement the player I have searched by so many ways and implemented different types of Third party libraries, Media player etc.
* I hadn’t got perfect library which can fulfill app’s requirement.
* After that I referred Exo Player default android library and get idea from that how to create persistence radio player to fulfill app’s requirement.
* In Exo Player lib I tried hardly to display seek bar on player. But due to .aac extension which is holding by radio stream URL I didn’t get succeed on that because of live stream couldn’t get the play duration. I checked with other format which was working well and shown perfect seek bar on player. So, I knew that I couldn’t show seek bar progress on player because of .aac format and live stream URL which is not able to get the play duration.
* Secondly, I wasn’t able to display song title, artist name, and album art of the currently playing song directly on the player.
* So, I overcome this problem by creating a customize player layout view which displays song title, artist name, and album art of the currently playing song and handled play/pause event by custom play button.

2. What shortcuts did you take that would be a bad practice in production?

* Currently, I have implemented Exo Player with basic functionality directly in the activity class but instead of that it should be inside well maintained class for real world app such as a using background / foreground services to tackle player lifecycle events through that way we can enhance player’s functionality which can play persistence even if app closed by user. Also we can show player on notification bar with play/pause event for real word app.

3. What would you have done with more time? We know you have a life. :-)

* To handle and develop player functionality only take a more time.

4. Where you don't have the time to implement something, how you would have changed or added to your implementation in the "real world".

* Currently, I haven’t implemented caching API data mechanism due to lack of time. But I would like to implement the functionality to cache the API data with Room persistence library in the “real world”.

5. Please include an architecture diagram.

