

U13-Soundmachine

Zuletzt bearbeitet von Lukas Schauhuber

U13-Soundmachine



Abbildung 1: Cover für die dreizehnte Übungsaufgabe

Aufgabe

Das Ziel dieser Aufgabe ist die Implementierung einer simplen Soundmachine-App. Die Anwendung soll durch Betätigen von Buttons kurze Soundeffekte abspielen können, die in der Projektstruktur gespeichert sind.

Hinweise

- Ihr könnt euch an dem Screenshot unten orientieren, müsst euch aber nicht daran halten. Das Design ist von euch frei wählbar. Das `GridLayout` ist für das Layout auf dem Screenshot eine gute Lösung
- Das Aussehen der Buttons innerhalb der Anwendung ist bereits über ein Theme in der `styles.xml` angepasst

- Es sind bereits einige Dateien im MP3-Format im *raw*-Verzeichnis des Starterpakets hinterlegt (Achtung Cringe), die verwendet werden können. Zu finden sind diese in der **AppConfig**-Klasse. Ihr könnt allerdings auch eigene Audioclips einbauen.
- In Android gibt es zwei grundlegende Möglichkeiten um Sounds abzuspielen: Den **SoundPool**, der für kurze (<5 Sekunden), untermalende Soundeffekte z.B. in Spielen verwendet werden kann, und den **MediaPlayer**, der für Musik und Video, sowie Streaming benutzt wird. Beides hat Vor- und Nachteile, in dieser Aufgabe wird der **MediaPlayer** verwendet, da Clips abgespielt werden sollen, die zum Teil das Dauerlimit des Soundpools überschreiten.
 - **SoundPool**: <https://developer.android.com/reference/android/media/SoundPool>
 - **MediaPlayer**: <https://developer.android.com/reference/android/media/MediaPlayer>

Tipps zum Vorgehen

- Entwerft ein *User Interface* das über Elemente verfügt, mit denen die Soundeffekte später ausgelöst werden können
- Referenziert alles Notwendige in der MainActivity und belegt die Elemente mit **OnClickListenern**. Versucht, falls Ihr das **GridLayout** verwendet, nicht jeden Button einzeln zu speichern, sondern mit dem **GridLayout** und dessen Kindern zu arbeiten
- Vervollständigt die Klasse **MediaPlayerHelper**, die die **MediaPlayer**-Interaktion, also das Abspielen von Sounds, steuert und mit der die **MainActivity** interagieren soll
- Implementiert die **MediaPlayer**-Funktionalität. Ein **MediaPlayer** benötigt die Referenz einer Datei, die über eine **resId** (Integer) übergeben werden kann. Anschließend kann das Playback gestartet (**start()**) werden. Wichtig ist dabei, dass der definierte Lifecycle des **MediaPlayer**-Objekts eingehalten wird, so muss eine Instanz nach dem Abspielende entweder zurückgesetzt (**reset()**) und/oder geschlossen (**release()**) werden. Um den Ablauf zu steuern, gibt die **MediaPlayer**-Klasse Interfaces, wie z.B. **OnCompletionListener** vor.
- Ein **MediaPlayer** kann nur einen Sound gleichzeitig abspielen und muss nach dem **reset** erneut vorbereitet werden, um ein anderes File anzugeben. Allerdings sollten auch nicht übermäßig viele Instanzen zur selben Zeit aktiv sein, da dies sehr viele Ressourcen benötigt und zu Problemen führen kann. Die Schwierigkeit liegt hier im Detail. Überlegt euch eine Lösung, um sicher zu stellen, dass nur eine begrenzte Anzahl von **MediaPlayern** existiert, oder sorgt dafür, dass der Nutzer beispielsweise Buttons nicht betätigen kann, solange das aktuelle Playback nicht beendet ist.

Screenshots

