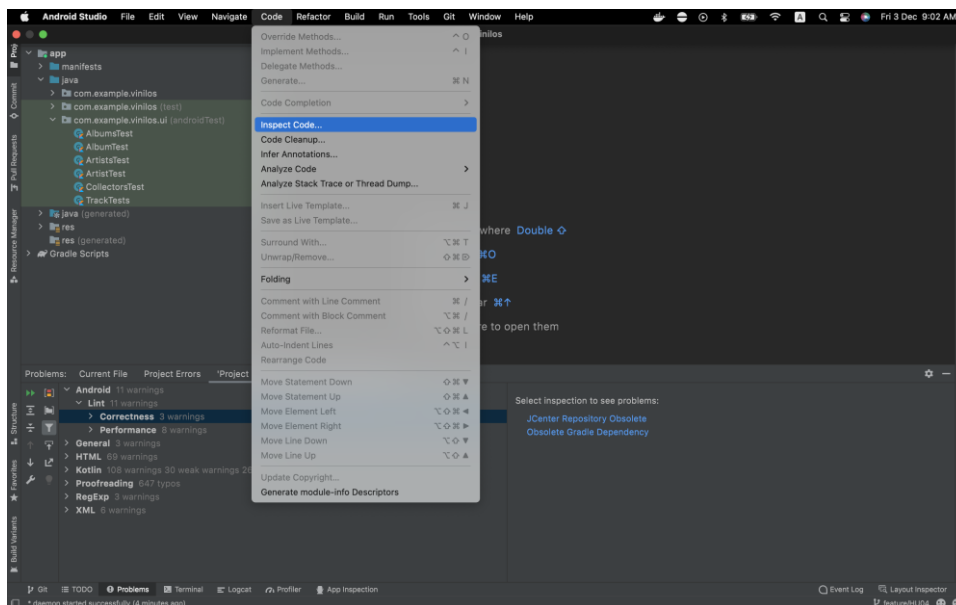


Análisis de accesibilidad

Para mejorar la accesibilidad del aplicativo realizamos diferentes mejoras que proporcionan un mayor acceso a usuarios que puedan tener alguna discapacidad de cualquier tipo, esas modificaciones se describen a continuación junto con las herramientas que usamos para detectar los puntos de mejora.

Lint

Se utilizó la herramienta linter de Android Studio para realizar una inspección de código al proyecto y así detectar puntos de mejora de accesibilidad. La opción se encuentra en la sección Code-> Inspect code.



Al realizar el análisis de accesibilidad se vio que no había mejoras propuestas ya que en iteraciones anteriores se corrió el mismo linter donde nos sugerían utilizar el field XML **`android:contentDescription`** con el cual aseguramos una descripción a los elementos de la pantalla que en caso de utilizar una aplicación de tipo Talkback le leyera al usuario como se puede ver a continuación:

```
<com.google.android.material.floatingactionbutton.FloatingActionButton
    android:id="@+id/fab"
    android:visibility="@{viewModel.isLoading || viewModel.isNetworkErrorShown ? View.GONE : View.VISIBLE}"
    app:backgroundTint="@color/vinilos_green"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="end|bottom"
    android:src="@drawable/ic_add_dark"
    android:layout_margin="16dp"
    android:contentDescription="@string/add_new_artist_button"
/>
```

Así mismo se utilizaron sugerencias para que los elementos gráficos sean mayores a 48dp x 48dp, texto mayor a 18sp y finalmente utilización de colores que no tengan una gran saturación.

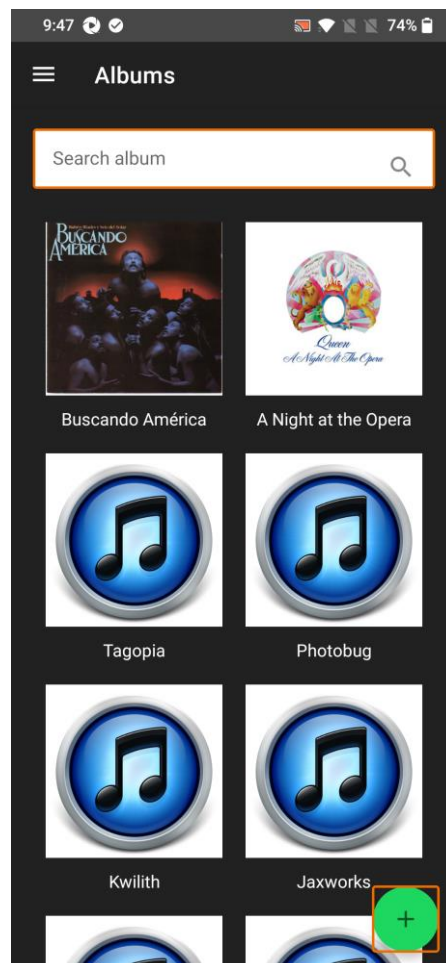
```

<com.google.android.material.textview.MaterialTextView
    android:id="@+id/description_section"
    app:cornerRadius="10dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="We are a platform that provides the amateur collector w..."
    android:textAlignment="center"
    android:textSize="18sp"
    android:padding="20dp"
    app:layout_constraintTop_toBottomOf="@id/quote_section"
    app:layout_constraintBottom_toTopOf="@+id/actions_section"
    android:background="@color/vinilos_light_gray"
    android:textColor="@color/white"
    android:layout_marginHorizontal="10dp"
    android:layout_marginVertical="10dp"
/>

```

Accessibility Scanner

Para mejorar la accesibilidad de la aplicación se ejecutó la aplicación Accessibility Scanner por Google para grabar un flujo completo de todas las pantallas de la aplicación. Una vez finalizada la grabación la app presenta los lugares donde se puede realizar una mejora.

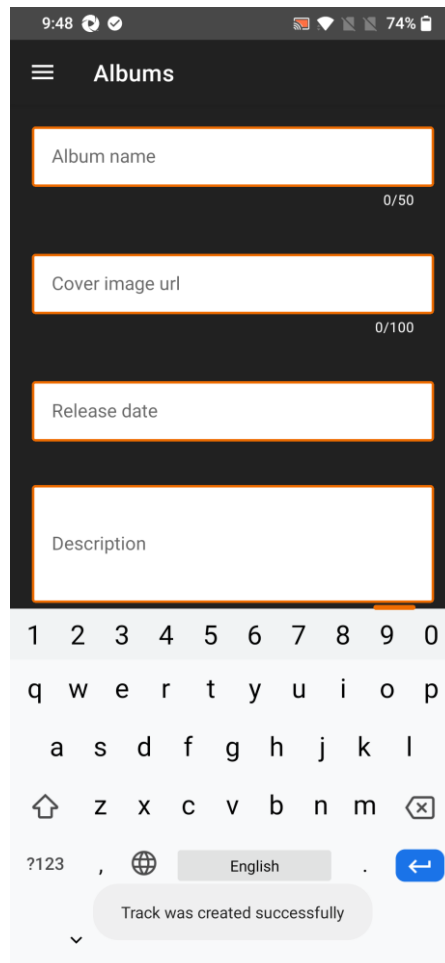


com.example.vinilos:id/floating_add_button: Mejorar texto para que contentDescription no contenga la palabra "button".

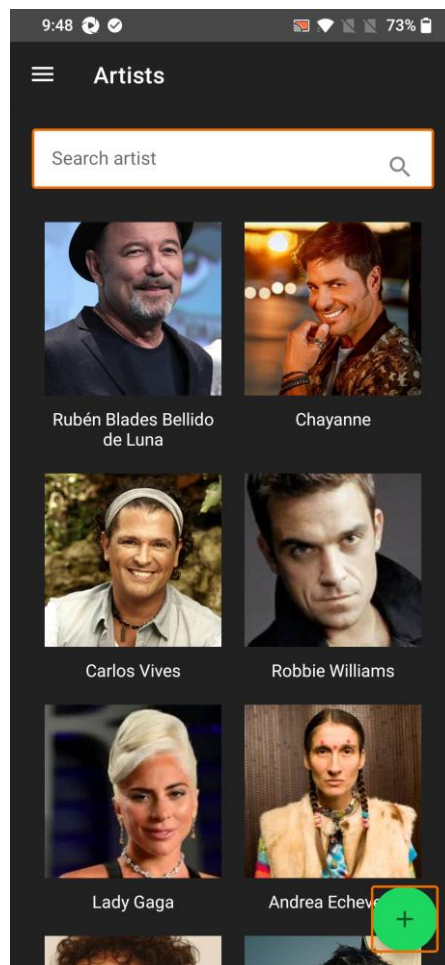
com.example.vinilos:id/search_box_field: Mejorar contraste del color

The screenshot shows a mobile application interface with a dark background. At the top, there is a status bar with the time 9:47, a refresh icon, and a battery level of 74%. Below the status bar is a header with a back arrow and the text "Kwilith". The main content area contains two text input fields. The first field is labeled "Track name" and has a character count of "0/50". The second field is labeled "Track duration" and has a character count of "0/100". Below these fields is a green button with the text "ADD TRACK".

- com.example.vinilos:id/album_detail_description: Mejorar contraste del color
- com.example.vinilos:id/track_name: Mejorar contraste del color
- com.example.vinilos:id/track_duration: Mejorar contraste del color
- com.example.vinilos:id/btn_add_track: Mejorar contraste del color



- `com.example.vinilos:id/record_label`: Elemento que pueda que solo sea legible para personas con vision



- `com.example.vinilos:id/floating_add_button`: Remover la palabra "button" del content description.



- El título del navigation bar es igual a otro título que aparece en el XML inferior

En total se encontraron 20 incidencias sin embargo la mayoría eran elementos que ya habían sido mencionados en otras pantallas o estados de la aplicación. En general la mayoría de correcciones realizadas fueron cambios al contraste de la aplicación y content descriptions con la palabra “button”.

Buenas prácticas de Accesibilidad

De acuerdo a las mejoras propuestas por el Accessibility Scanner se procedió a remover el cambio de título del action bar en la actividad de detalle de un artista. Anteriormente al cargar el artista en el observador se actualizaba el título para que fuera el nombre del artista.

```
viewModel.artist.observe( owner: this, Observer<Artist> { it: Artist!  
    it.apply { this: Artist!  
        title = this.name  
    }  
})
```

Se realiza el cambio para que ya no actualice como se puede ver a continuación.

```
viewModel.artist.observe( owner: this, Observer<Artist> {})
```

Así mismo se procede se notó que no es buena práctica utilizar la palabra “button” en el content descripción de un botón, así se veía el String de los botones crear álbum y artista previamente.

```
<string name="add_new_album_button">Add new album button</string>
<string name="add_new_artist_button">Add new artist button</string>
```

Y se procedió a hacer el cambio para que ya no tuviera la palabra "button".

```
<string name="add_new_album_button">Add new album</string>
<string name="add_new_artist_button">Add new artist</string>
```

Finalmente cabe resaltar que para los componentes de los XML se utilizan los campos content description y así facilitar la identificación de componentes.

```
<com.google.android.material.floatingactionbutton.FloatingActionButton
    android:id="@+id/floating_add_button"
    android:visibility="@{viewModel.isUser || viewModel.isLoading || vie
    app:backgroundTint="@color/vinilos_green"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="end|bottom"
    android:src="@drawable/ic_add_dark"
    android:layout_margin="16dp"
    android:contentDescription="@string/add_new_album_button"
/>
```