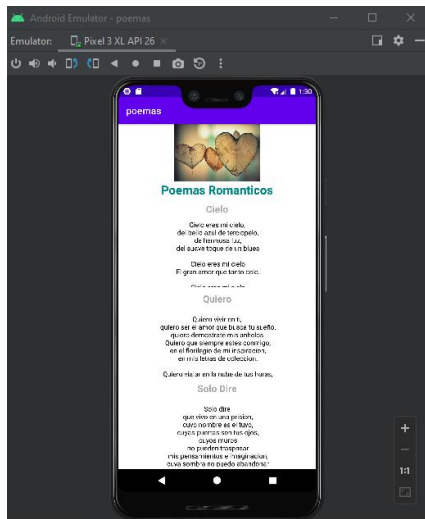
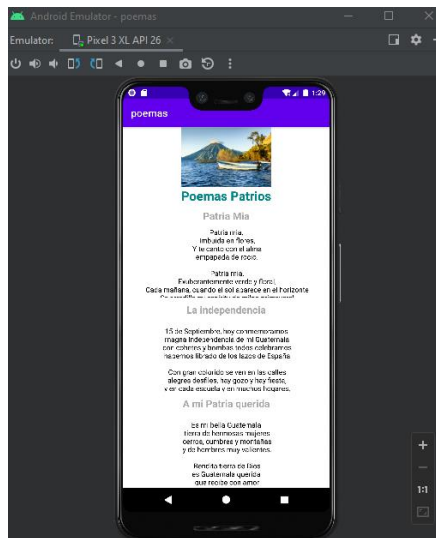
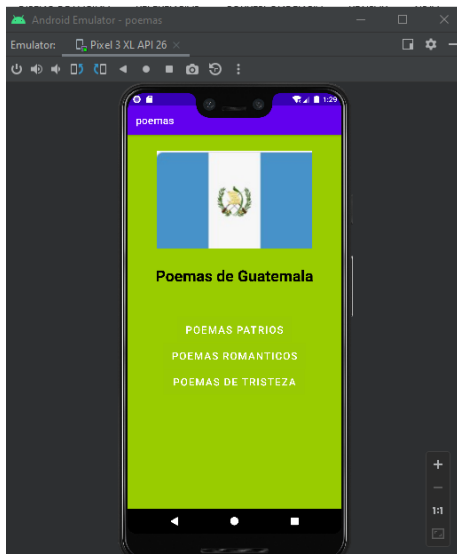


## Aplicacion Poemas



### Diseño

En el primer activity (Activity\_main.xml), creo un contenedor con RelativeLayout, tanto el ancho como la altura, les configuro la propiedad “match parent” para que tanto la altura como el ancho se ajusten a la pantalla. Creo un textView para “Poemas de Guatemala”. Creo un ImageView para la imagen de la bandera de Guatemala. Creo tres Button con la propiedad: “@color/cardview\_shadow\_end\_color” para que tome el color del background del contenedor para que el mismo no se vea como un botón tradicional si no como, un enlace que lo direcciona a otra pantalla (activity).

En los siguientes tres activities (activity\_main\_patrios.xml, activity\_main\_romanticos.xml y activity\_main\_tristeza.xml) creo un contenedor principal con LinearLayout que aloja 3 contenedores secuenciales hacia abajo con LinearLayout, dentro de estos últimos 3 contenedores, creo un scrollbar en cada contenedor, esto, con el fin de que puedan leerse los poemas sin ningún problema.

### String

En el apartado de “values” en el archivo “strings.xml”, creo los recursos “string” respectivos para ser llamados en la propiedad “@string/txt...” de los textView que contienen los contenedores del contenido del poema en cada activity\_main\_x.

### Logica

En la actividad “MainActivity.java”, se desarrolla la lógica de la app en relación al evento del clic para cada botón, y con dicho evento, el direccionamiento a cada activity. Declaro las variables button de tipo Button como variables privadas

globales. Creo el método “protected void Oncreate (Bundle savedInstanceState)” en el que mando a llamar a los elementos del activity\_main con la función findViewById() como una forma de interaccion con el método onClick. Por ultimo creo el método void Click , el cual manda a llamar al método que apertura la activity: startActivity(Intent).

