

Laboratorio #2 - Explicación

Como primer punto, se utiliza el paquete denominado para este laboratorio, mismo que fue `com.example.lab2`, para posteriormente importar todas las funciones y librerías necesarias. Enfocándonos en la clase de `MainActivity`, que es lo principal en nuestra app. Se hace uso de la función `onCreate` que hace que se configure la interfaz en esa parte del código, para después sea pasado al `setContent` y que el `enableEdgeToEdge` y es lo que habilita el diseño de la habitación y finalmente el `LemonadeApp()`, que vendría siendo todo lo que conlleva la aplicación (pantallas y funciones).

Lo que hace la función `LemonadeApp` es un composible y lo que se hace más que todo es hacer un `case` en el que se desarrolla toda la aplicación. Luego, la sección de `currentStep` es que cada vez que se presione un botón, el estado se configura de 1 a 0 y el `selectedIndex` se colocaba como 0, ya que era la parte de presionar varias veces. Así mismo se agregó un `Scaffold` que es para agregar la barra que aparece en la interfaz para que salga lemonade y después se pasa al contador de `currentStep` que se definió arriba en variable y que lo recuerde (el estado). Al llegar a la parte de `SqueezeRandom`, es un random de 2 a 4 botones como se indicaba para esa pantalla en el `codeLab`. Por lo que al ir recorriendo las pantallas, el `currentStep` va cambiando para cambiar de pantalla cada vez que sea `onClick`, agregando así números de estado correspondientes a cada pantalla hasta llegar a la final (pantalla #4), desarrollando el contenido respectivo de cada pantalla. Al finalizar con la pantalla #4, se reinicia todo para empezar nuevamente en la pantalla #1. Cabe destacar que se agregó una parte en el `preview` en la parte de abajo para ir visualizando los cambios más rápido que corriendo el emulador, así mismo, se agregaron las respectivas imágenes de cada pantalla en la sección de `drawable` y los `strings` en `resources` para no dejarlo en `""`.