**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ**

**Факультет кібербезпеки, комп’ютерної та програмної інженерії**

Кафедра інженерії програмного забезпечення



Лабораторна робота №3

З дисципліни «**Системи для малих та моб платформ**»

на тему: «**Розмітки LinearLayout і FrameLayout**»

Виконав студент:

групи ПІ-321

Іванюк Н. О.

Перевірив викладач:

Щерба В. В.

Київ 2021

**Теоретичні відомості**

Розмітка або макет (Layout) використовується для розташування дочірніх компонентів на екрані пристрою. Використовуючи вкладені розмітки, можна створювати інтерфейси користувача будь-якої складності. В Android існують такі основні розмітки:

• RelativeLayout - визначає положення одних View відносно інших.

• LinearLayout - визначає розташування всіх дочірніх елементів по вертикалі або по горизонталі один за одним.

• TableLayout - розглядає дочірні елементи у вигляді рядків і стовпців.

• GridView - елементи представлені в двовимірної сітці з прокруткою. Елементи в GridView потрапляють з призначеного примірника ListAdapter.

• FrameLayout - заповнює собою екран для відображення єдиного View.

• ConstraintLayout - подібно RelativeLayout прив’язує елементи інтерфейсу до контейнера (батьківського елемента), в якому вони знаходяться або до інших елементів цього контейнера і має розширені можливості таких привязок.

• CoordinatorLayout - координує взаємне розташування View-елементів, які знаходяться всередині цього View-group • AppbarLayout - нагадує вертикальний LinearLayout, елементи якого можуть вести себе порізному (в залежності від прапорів) при прокручуванні вмісту.

• CollapsingToolbarLayout - це ViewGroup, який надає багато візуальних характеристик та взаємодій для згортання панелей інструментів. Використовується разом з AppbarLayout та CoordinatorLayout.

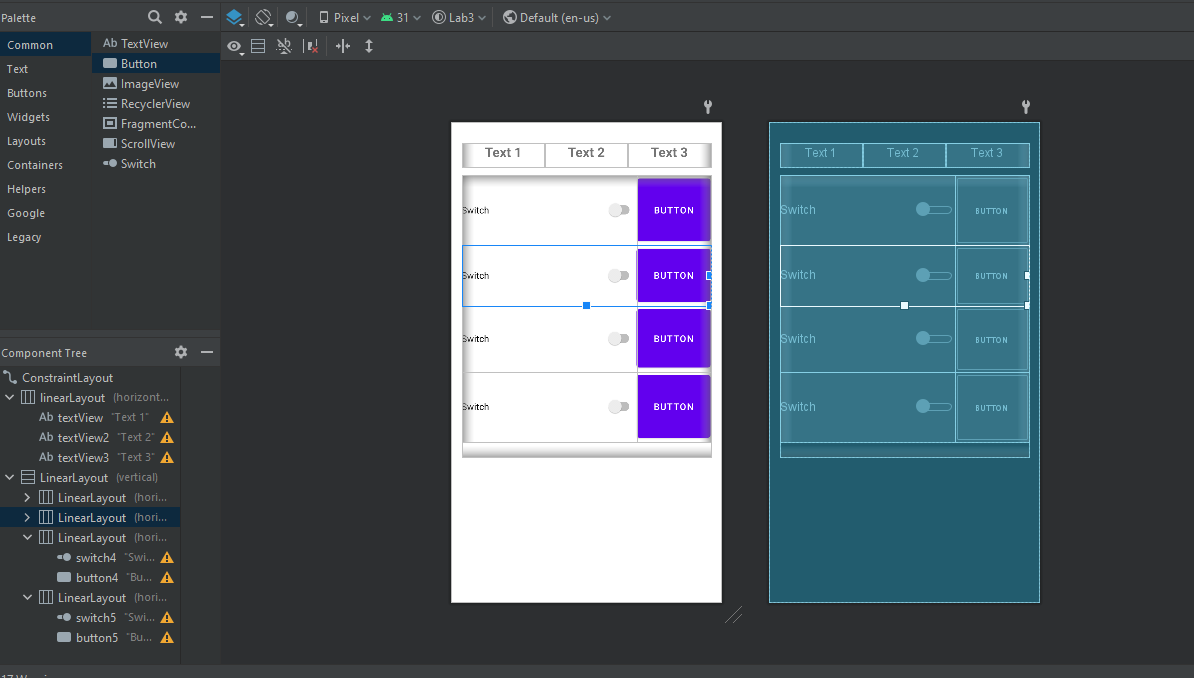
Розмітка LinearLayout дозволяє компонувати різні елементи графічного інтерфейсу один за одним вертикально або горизонтально. LinearLayout може бути кореневим елементом макета або бути вкладеним сам в себе стільки разів, скільки потрібно. Також можна взаємодіяти з іншим розмітками, наприклад, RelativeLayout або FrameLayout, де LinearLayout може бути як вкладеним макетом, так і тим, в який вкладається.LinearLayout дозволяє задавати орієнтацію контенту (за замовчуванням використовується горизонтальна орієнтація), висоту і ширину елементів, встановлювати тип вирівнювання самого елемента або його вмісту, зовнішні та внутрішні відступи з усіх боків і таке інше. Розмітка FrameLayout призначена для виведення на екран одного або кількох візуальних елементів, які по замовчуванню прикріплюються до верхнього лівого кута екрану. Елементи будуть накладатися один на одного, частково або повністю затінюючи попередніх, якщо зверху буде знаходитися непрозорий елемент.

**Завдання**

Розташувати компоненти графічного інтерфейсу, застосовуючи вкладення розміток LinearLayout. Доповнити попереднє розташування додатковими компонентами з використанням розмітки FrameLayout.

**Виконання:**

Створюю розмітку:



Тут ми бачимо компонування елементів у LinearLayout та FrameLayout. LinearLayout використовується як горизонтально так і вертикально.

Елементи в ConstraintLayout можна закріплювати відносно інших елементів та границь. При переносі головного елементу – закріплений будк переміщуватись також.

**Код activity\_main.xml**

<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity">  
  
 <LinearLayout  
 android:id="@+id/linearLayout"  
 android:layout\_width="0dp"  
 android:layout\_height="36dp"  
 android:layout\_marginStart="16dp"  
 android:layout\_marginTop="32dp"  
 android:layout\_marginEnd="16dp"  
 android:orientation="horizontal"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.0"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent">  
  
 <TextView  
 android:id="@+id/textView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:text="Text 1"  
 android:textAlignment="center"  
 android:textSize="20sp"  
 android:textStyle="bold" />  
  
 <TextView  
 android:id="@+id/textView2"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:text="Text 2"  
 android:textAlignment="center"  
 android:textSize="20sp"  
 android:textStyle="bold" />  
  
 <TextView  
 android:id="@+id/textView3"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:text="Text 3"  
 android:textAlignment="center"  
 android:textSize="20sp"  
 android:textStyle="bold" />  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="0dp"  
 android:layout\_height="430dp"  
 android:layout\_marginStart="16dp"  
 android:layout\_marginTop="12dp"  
 android:layout\_marginEnd="16dp"  
 android:orientation="vertical"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="1.0"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/linearLayout">  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="107dp"  
 android:orientation="horizontal">  
  
 <Switch  
 android:id="@+id/switch2"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:text="Switch" />  
  
 <Button  
 android:id="@+id/button2"  
 android:layout\_width="224dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:text="Button" />  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="92dp"  
 android:orientation="horizontal">  
  
 <Switch  
 android:id="@+id/switch3"  
 android:layout\_width="298dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:text="Switch" />  
  
 <Button  
 android:id="@+id/button3"  
 android:layout\_width="144dp"  
 android:layout\_height="92dp"  
 android:layout\_weight="1"  
 android:text="Button" />  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="100dp"  
 android:orientation="horizontal">  
  
 <Switch  
 android:id="@+id/switch4"  
 android:layout\_width="352dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:text="Switch" />  
  
 <Button  
 android:id="@+id/button4"  
 android:layout\_width="198dp"  
 android:layout\_height="100dp"  
 android:layout\_weight="1"  
 android:text="Button" />  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="107dp"  
 android:orientation="horizontal">  
  
 <Switch  
 android:id="@+id/switch5"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:text="Switch" />  
  
 <Button  
 android:id="@+id/button5"  
 android:layout\_width="225dp"  
 android:layout\_height="107dp"  
 android:layout\_weight="1"  
 android:text="Button" />  
 </LinearLayout>  
  
 </LinearLayout>  
  
</androidx.constraintlayout.widget.ConstraintLayout>

**Висновок:**

Виконавши лабораторну роботу, я дізнався про способи розміщення елементів розмітки та її основні типи. Покращив свої знання у створенні графічних інтерфейсів та їх компонування.