**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**КЕМЕРОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**Институт фундаментальных наук**

**ОТЧЕТ**

**О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ**

«Схема компоновки и верстки»

Студента 4 курса, МОА-205 группы

**Чернов Андрей Викторович**

Направление 02.03.03 – «Математическое обеспечение

и администрирование информационных систем»

Руководитель:

Кандидат технических наук,

доцент кафедры ЮНЕСКО по новым информационным технологиям

И.О. Фамилия

С.Ю. Завозкин

Работа защищена

«\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_»

“\_\_\_\_” \_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_г.

Кемерово 2023

**Цель работы:**

Познакомиться со схемами компоновки элементов и их особенностями.

**Задачи**

1. Познакомиться с особенностями работы вложенных схем компоновки.
2. Познакомиться с объектами CheckBox, ProgressBar

**ЗАДАНИЕ**

1. Создать приложение, содержащее в себе вложенные схемы

компоновок;

2. Реализовать обработчик, работающий со вложенными элементами

вложенной схемы компоновки;

3. Реализовать приложение с логически усложненными

обработчиками.

**Описание**

В процессе разработки приложения различные события могут происходить в окружении Android Studio, такие как: запуск приложения, анализ кода, компиляция, сборка, выполнение тестов и другие. Каждое событие может быть записано в лог приложения, что позволяет отслеживать и анализировать действия и результаты работы приложения.

Лабораторная работа по теме "Схемы компоновки и вёрстка" направлена на освоение основных принципов разметки пользовательского интерфейса в приложениях для операционной системы Android.

Целью данной работы является ознакомление с различными схемами компоновки, которые позволяют эффективно организовать расположение элементов пользовательского интерфейса на экране устройства. В ходе выполнения работы, студенты изучают основные понятия и концепции, такие как LinearLayout, ConstraintLayout.

Предлагается создать простое приложение, в котором будет использоваться различные схемы компоновки для размещения кнопок, текстовых полей и других элементов пользовательского интерфейса. В процессе работы предстоит научиться правильно задавать атрибуты компонентов, устанавливать правила и ограничения для их взаимного расположения.

Полученные навыки и знания после выполнения данной лабораторной работы позволят студентам создавать эффективные и привлекательные пользовательские интерфейсы для своих будущих приложений.

**Ход работы**

***Вложенные компоновки***

Создаем новый проект с названием «NestedLayouts» для изучение компоновки в android.

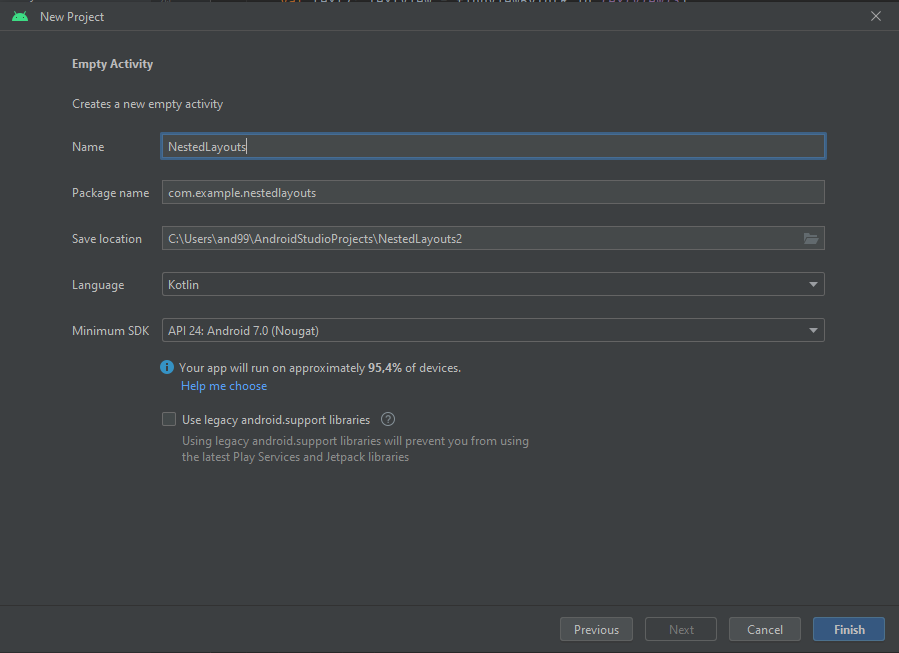


Рис. 1. Создание проекта «Вложенной компоновки»

Добавим три схемы компоновки в соответствующем порядке LinearLayout (вертикальный), LinearLayout (горизонтальный) и ConstraintLayout.



Рис. 2. Размещение трех схем компоновки

Разместив схемы компоновки, добавим в каждую из них по три объекта TextView так, как на рисунке 3.

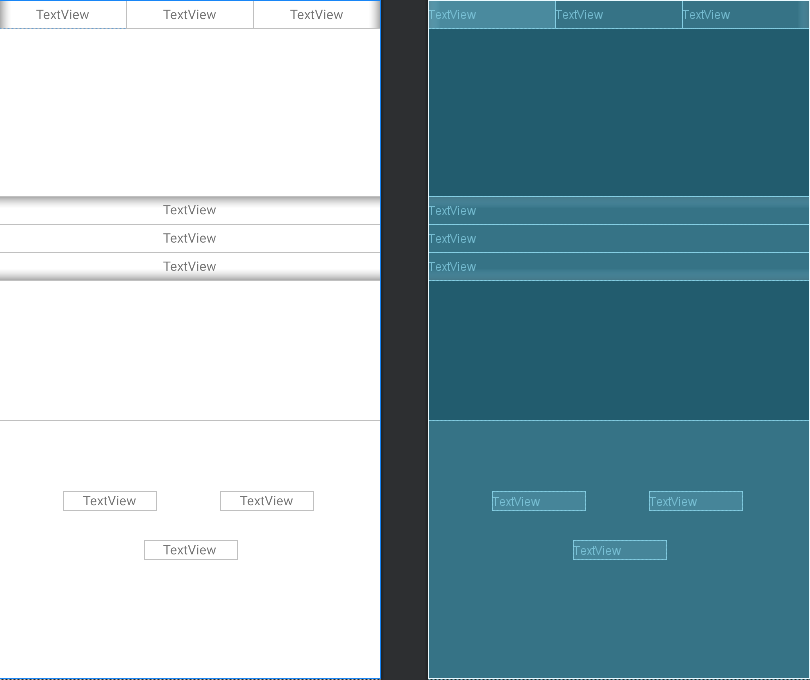


Рис. 3. Размещение объектов по схемам компоновки

**Обработчик для вложенных объектов**

Реализация обработчика событий (Рисунок 1):

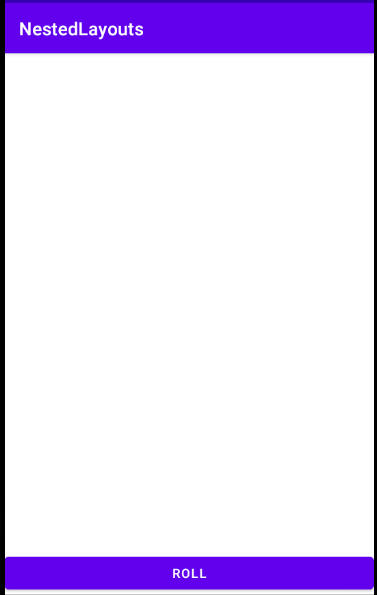
****

Рис. 1. Первый запуск приложения

При нажатии кнопки «ROLL» вызывается обработчик нажатия кнопки, который заполнит TextView цифрами из счетчика (Рисунок 2).

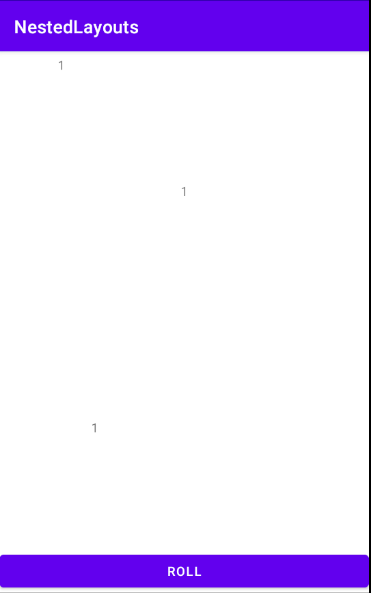
****

Рис. 2. Первое нажатие кнопки

Второе нажатие кнопки «ROLL» произойдет смещение в следующие TextView и счетчик увеличиться на 1 (Рисунок 3).

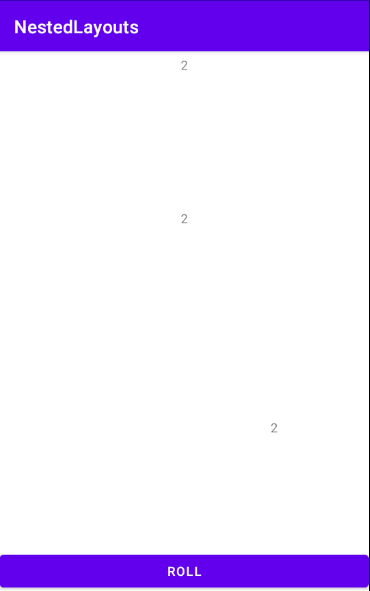
****

Рис. 3 Второе нажатие кнопки

При третьем нажатии кнопки «ROLL» произойдет повторное смещение TextView и счетчик увеличиться снова (Рисунок 4).

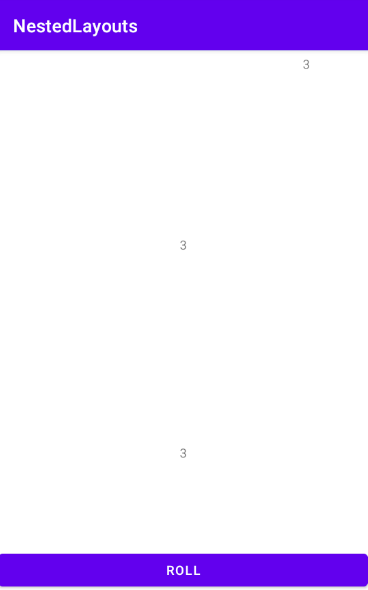
****

Рис. 4. Третье нажатие кнопки

При следующем нажатии к виду, как при первом нажатии (Рисунок 2), кроме счетчика, он увеличиться. И цикл нажатий повториться.

**Атрибуты background, textColor, textSize**

Создаем новый проект с название «ComplexEvent» (Рисунок 1).

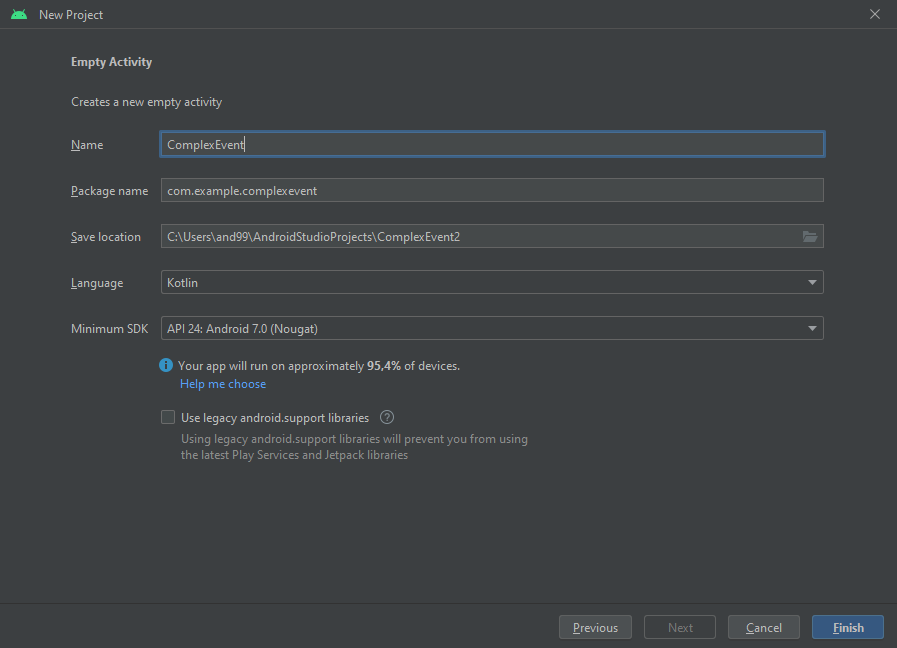


Рис. 1. Создание проекта «ComplexEvent»

Разместим LinearLayout и конвертируем его в вертикальный (Рисунок 2).



Рис. 2. Размещение схемы компоновки

Указываем параметры компоновки (Рисунок 3).

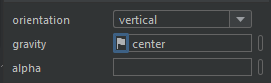


Рис. 3. Указание параметра gravity

Размещаем на схеме объекты EditText, Checkbox, Button, TextView и

ProgressBar (Рисунок 4).

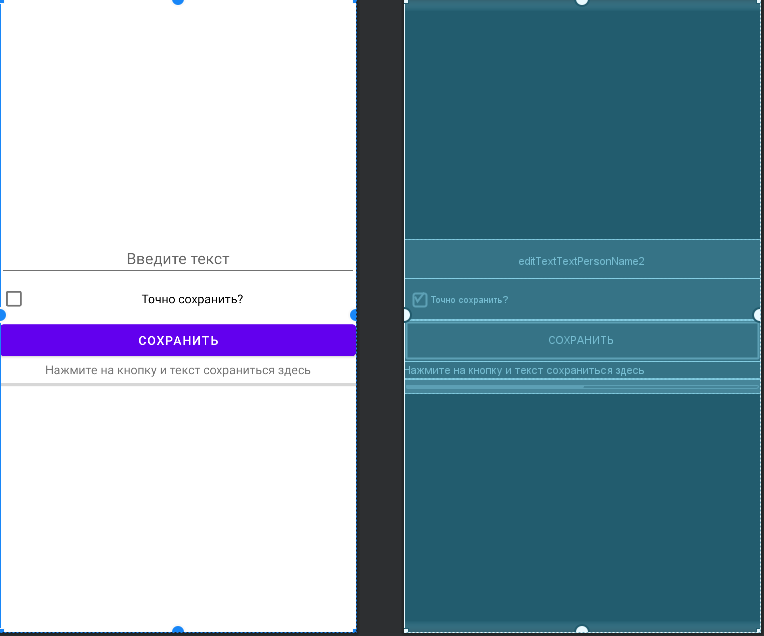


Рис. 4. Размещение объектов в LinearLayout

Запуск приложения (Рисунок 5).

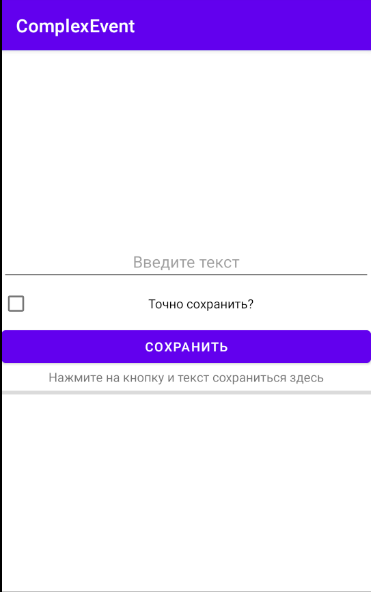


Рис. 5. Первый запуск приложения

При вводе текста «Save», без установки флага «Точно сохранить?», нажимая кнопку сохранить текст не сохраняется и ProgressBar не заполняется (Рисунок 6).

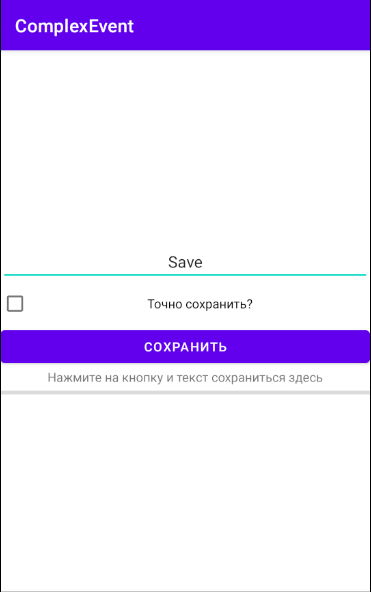


Рис. 6. Нажатие на кнопку, без установки флага

Однако при установке флага сохранения, все сохраняется и текст выводиться, ProgressBar заполняется (Рисунок 7).

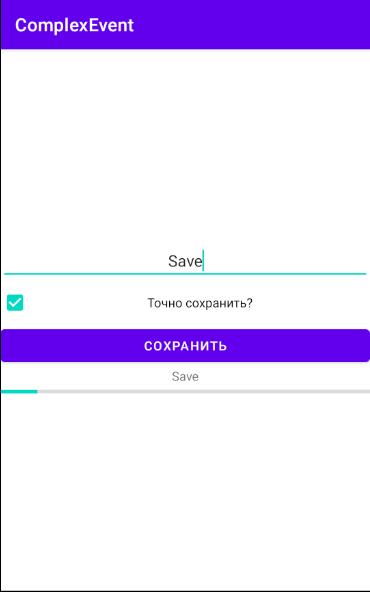


Рис. 7. Сохранение с флагом

**Вывод:**

В данной работе были рассмотрены схемы компоновки и вёрстка в Android. Было изучено несколько основных типов схем компоновки, таких как LinearLayout, ConstraintLayout. Каждый из них обладает своими особенностями и применяется в зависимости от требований проекта.

В целом, изучение схем компоновки и вёрстки в Android позволяет разработчику создавать эффективные и пользовательский дружелюбные интерфейсы, которые будут отлично работать на различных устройствах. Это важный навык для любого Android-разработчика, и знание основных принципов и методов поможет создать качественные приложения.