|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |
| --- |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** |
| **Институт кибербезопасности и цифровых технологий** |

Кафедра КБ-14 «Цифровые технологии обработки данных»

**ОТЧЕТ**

**Практическая работа 1**

**по дисциплине «Разработка мобильных компонент анализа безопасности информационно-аналитических систем»**

Выполнил:

Студент группы БСБО-07-22

Сладкина Анастасия Андреевна

**Москва, 2025 г**

Для начала были изучены атрибуты макетов «ViewGroup» и для каждого типа созданы свои Layout (см. рис 1-2).

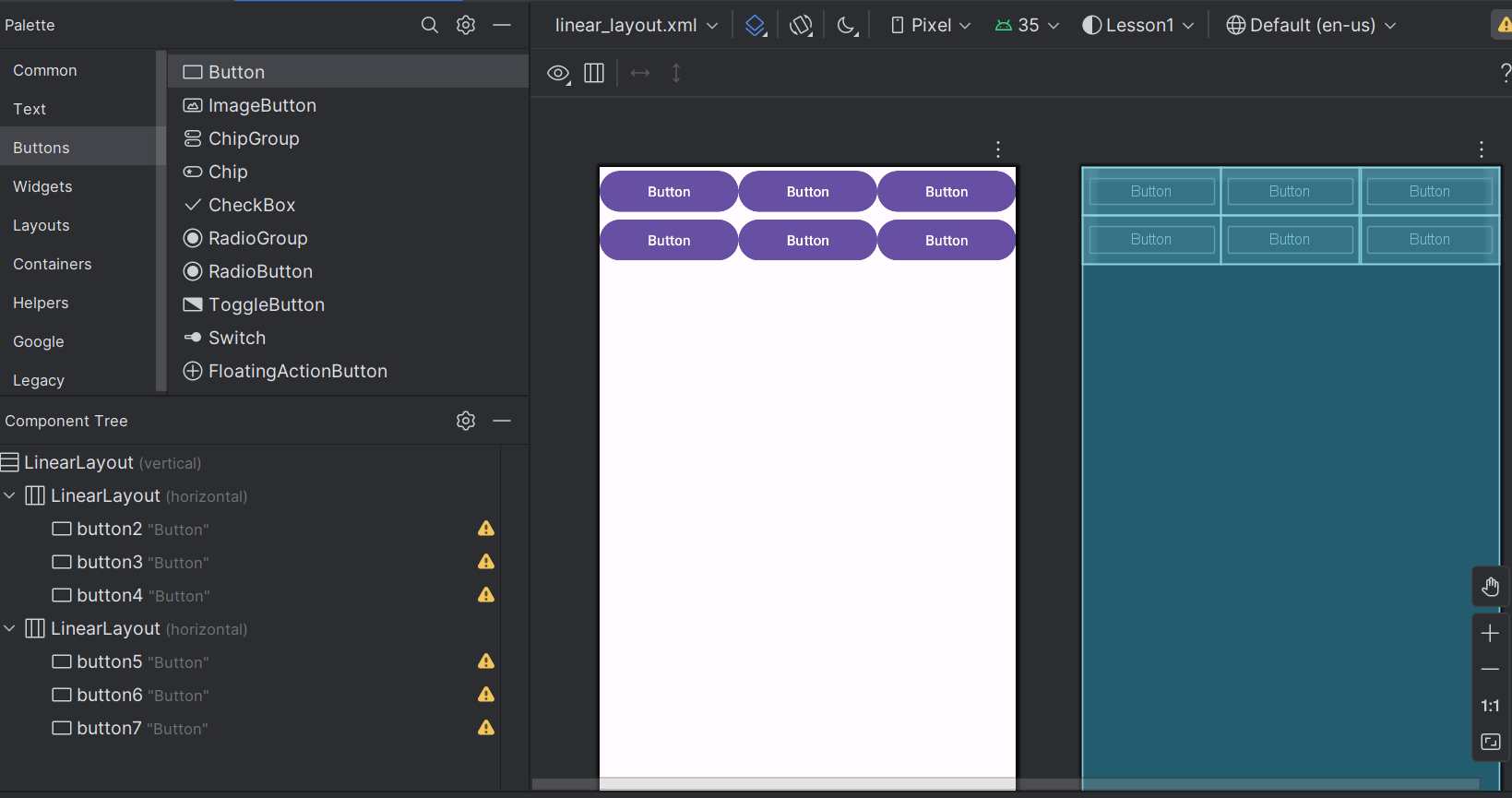


Рисунок 1. Изучение LinearLayout

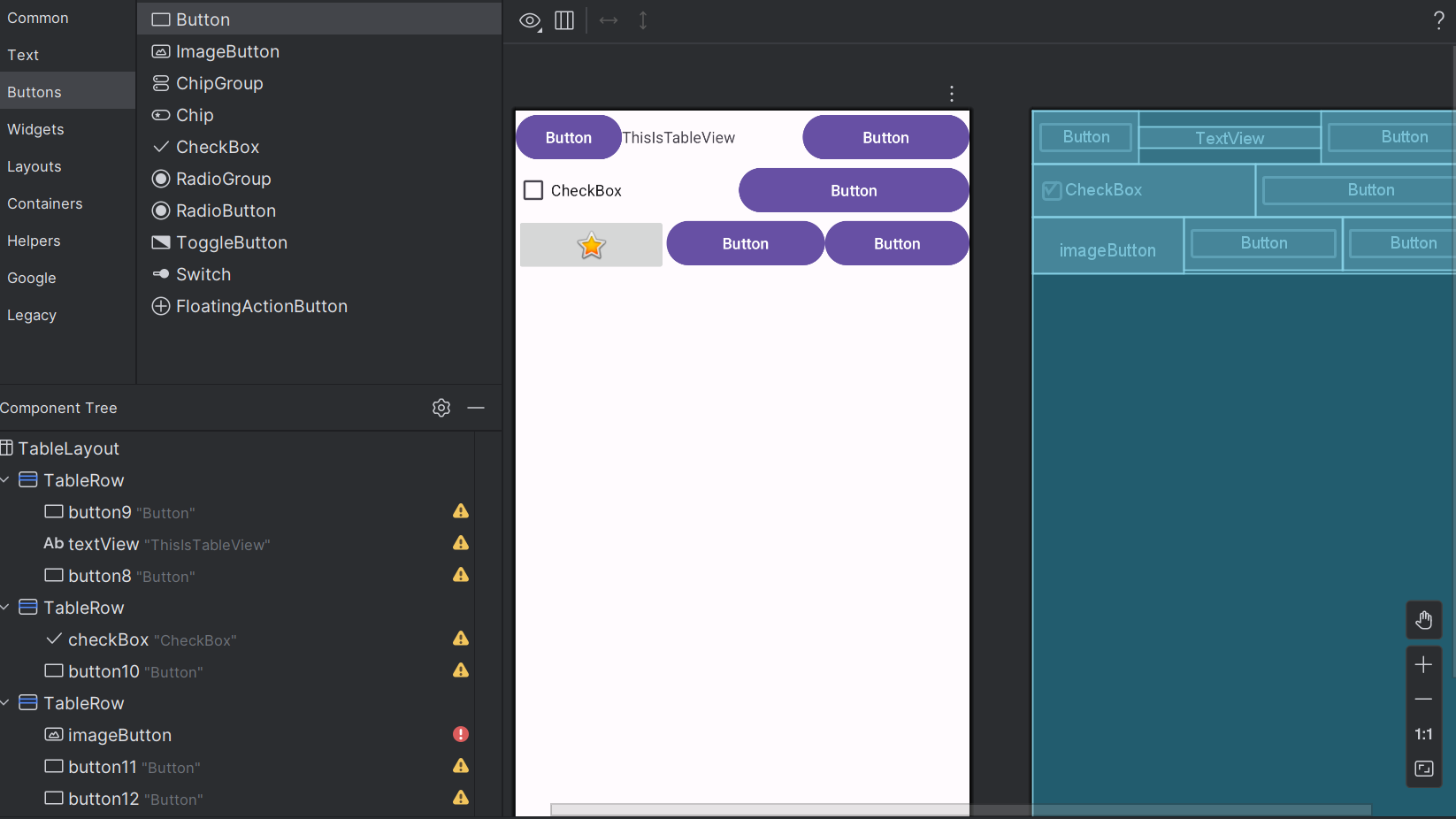


Рисунок 2. Изучение TableLayout

Далее необходимо создать файл разметки «table\_layout.xml» с корневым элементом «TableLayout» и добавить в корневой элемент три «TableRow»-строки (см. рис 3).

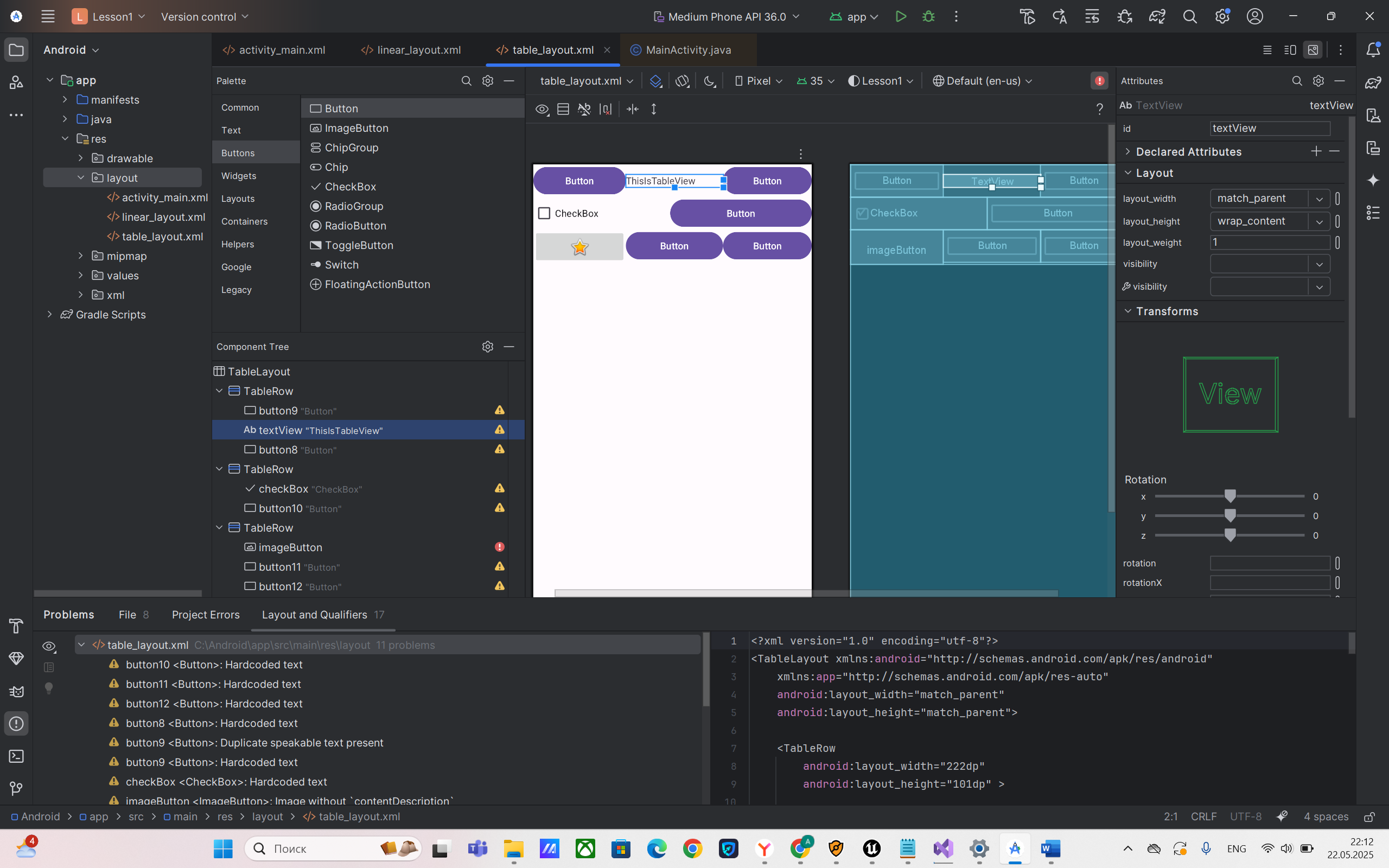


Рисунок 3. Экран TableLayout

Управление компонентами внутри данного контейнера отличается от предыдущих контейнеров взаимодействия. Требуется добавить элемент «TextView» на экран, который отобразится в верхнем левом углу, используя ConstraintLayout (см. рис. 4).

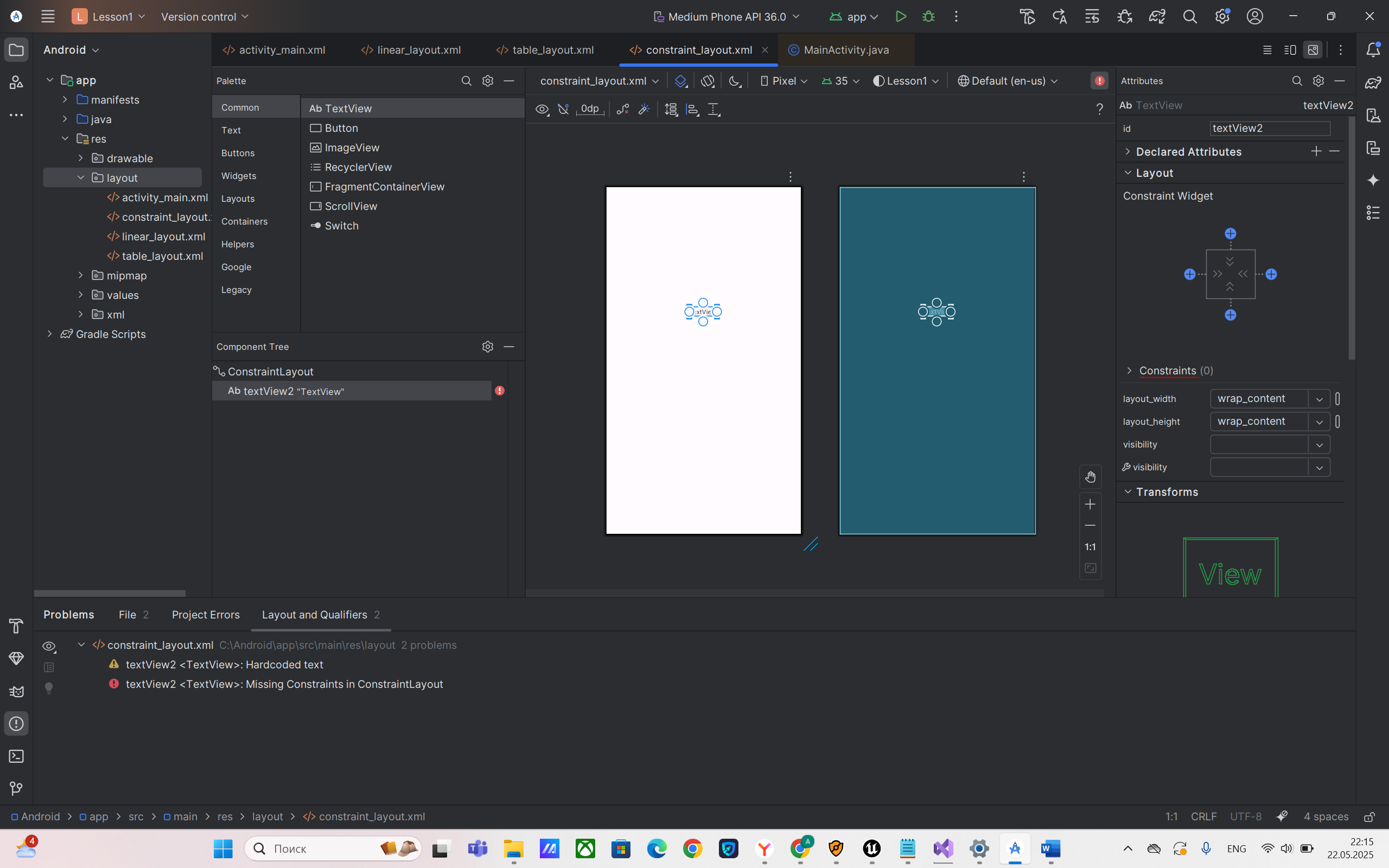


Рисунок 4. Изучение ConstraintLayout

Далее в текстовом представлении экрана (вкладка «Code») элемент «TextView» подчеркивается красной линией и отображается ошибка: Этим сообщением среда разработки сообщает, что «View» не привязано к элементам экрана (см. рис. 5).

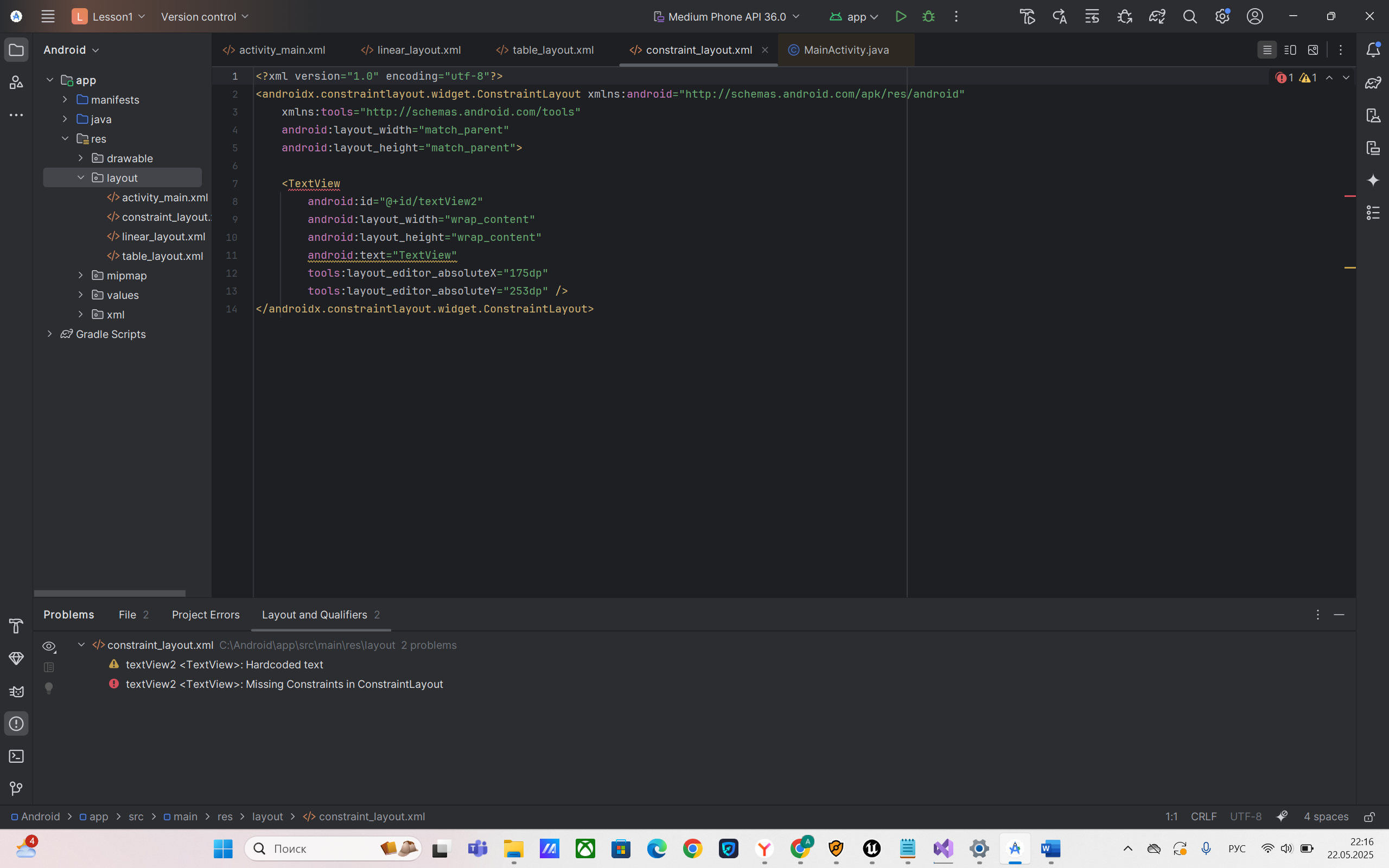


Рисунок 5. Изучение ошибки «TextView»

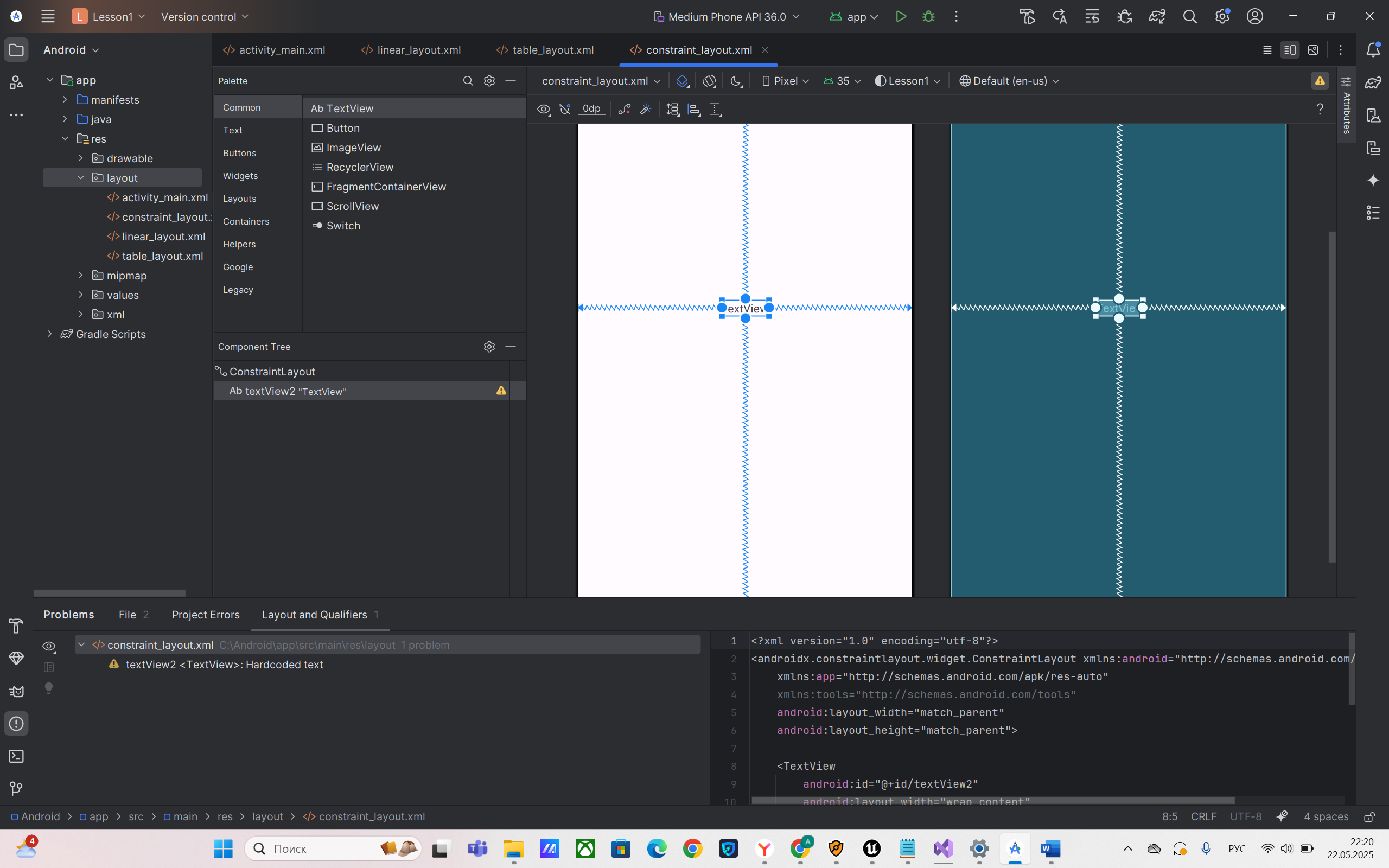
Элемент «TextView» располагается в «ConstraintLayout», который в приведенном примере занимает весь экран. Поэтому края «ConstraintLayout» совпадают с краями экрана (см. рис. 6).

Рисунок 6. «TextView» в «ConstraintLayout»

Далее был создан новый модуль «control\_lesson1». После открытия разметки «activity\_main.xml» (res>layout) из меню palette были перенесены следующие элементы и изучены их свойства (область «attributes»): - Text>textView, plainText (editText) и т.д.; - Buttons>button, imageButton, checkBox и т.д.; - Widgets>imageview (установить изображение) и т.д.. Далее добавлены на экран несколько элементов и привязаны между собой (см. рис. 7).

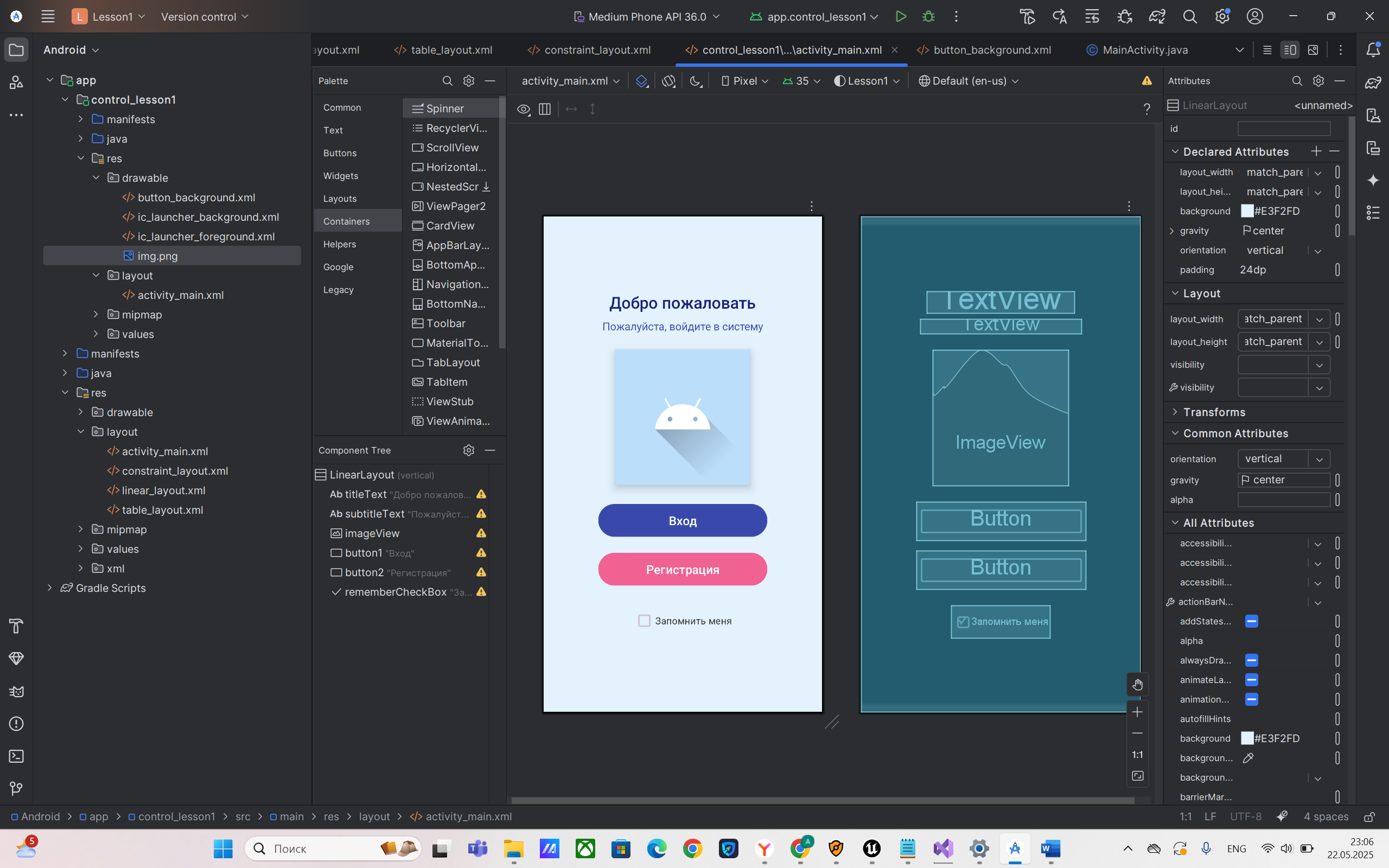


Рисунок 7. Создание красивой кнопки и своего экрана

Требуется вызвать контекстное меню директории «res/layout» в модуле и открыть генератор ресурсов: New> XML> Layout XML File> Layout Resource File. Для нового файла разметки было установлено имя «activity\_second.xml». В созданном layout-файле добавлен на экран элемент «PlainText» и изменен его текст на: «New life for mirea activity!» и 6 кнопок button (см. рис. 8).

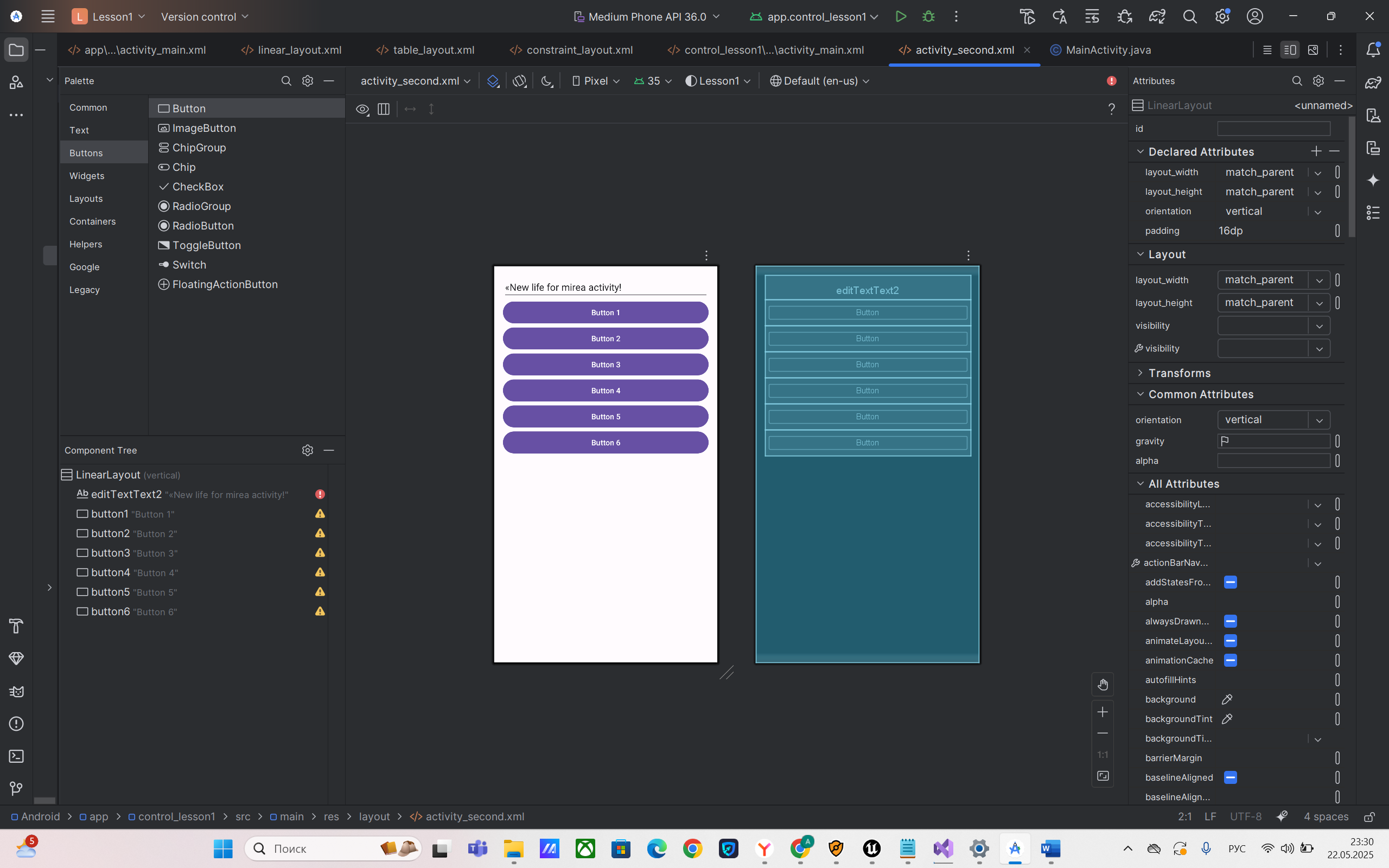


Рисунок 8. Создание шести кнопок и текста

При создании нового layout-файла «activity\_second», среда добавляет в r.java новую константу – «R.layout.activity\_second». После этого в коде класса MainActivity возможно через данную константу указать на новый layout-файл. Для того, чтобы activity использовало новый файл «activity\_second.xml», а не «activity\_main.xml», был изменен аргумент метода setContentView(см. рис. 9).

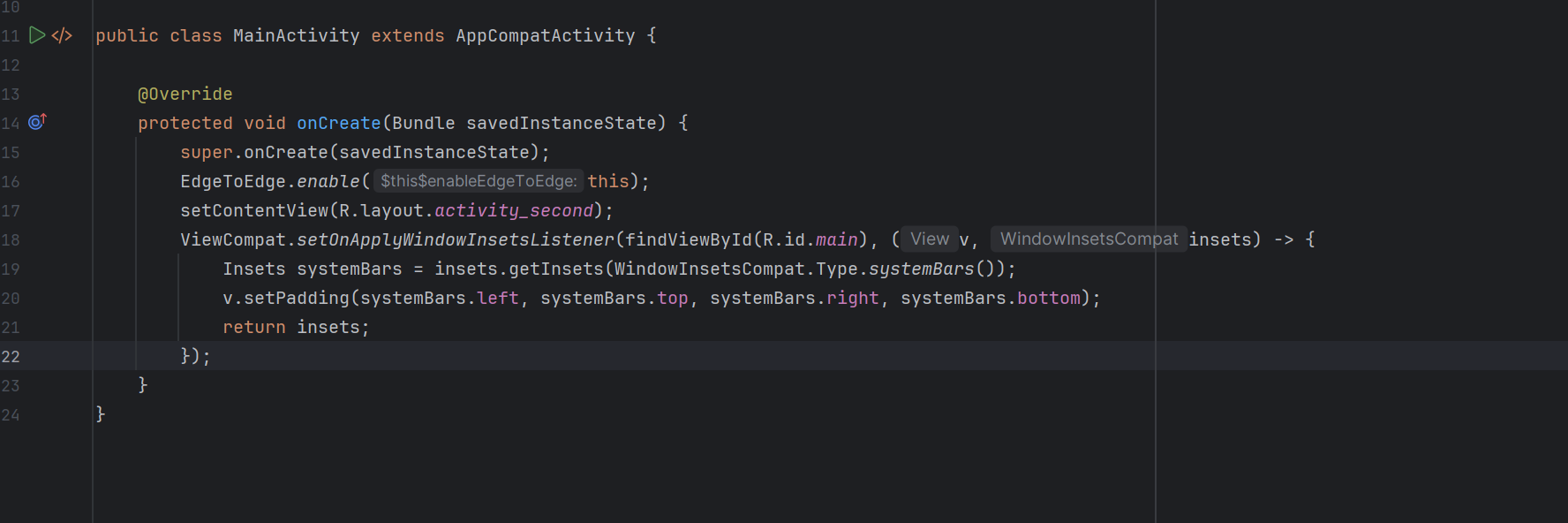


Рисунок 9. Изменение аргумента на секонд

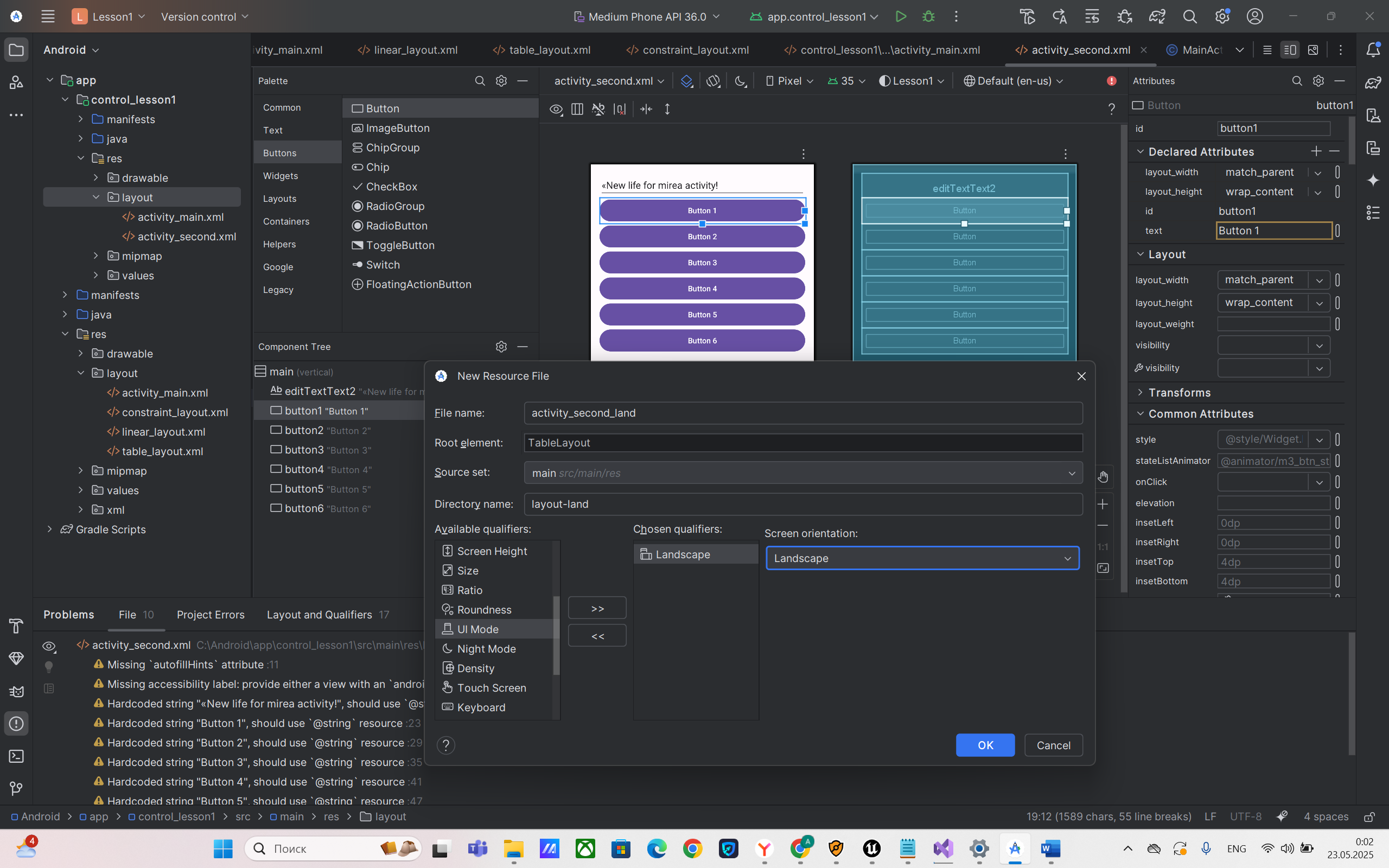


Рисунок 10. Вертикальная ориентация

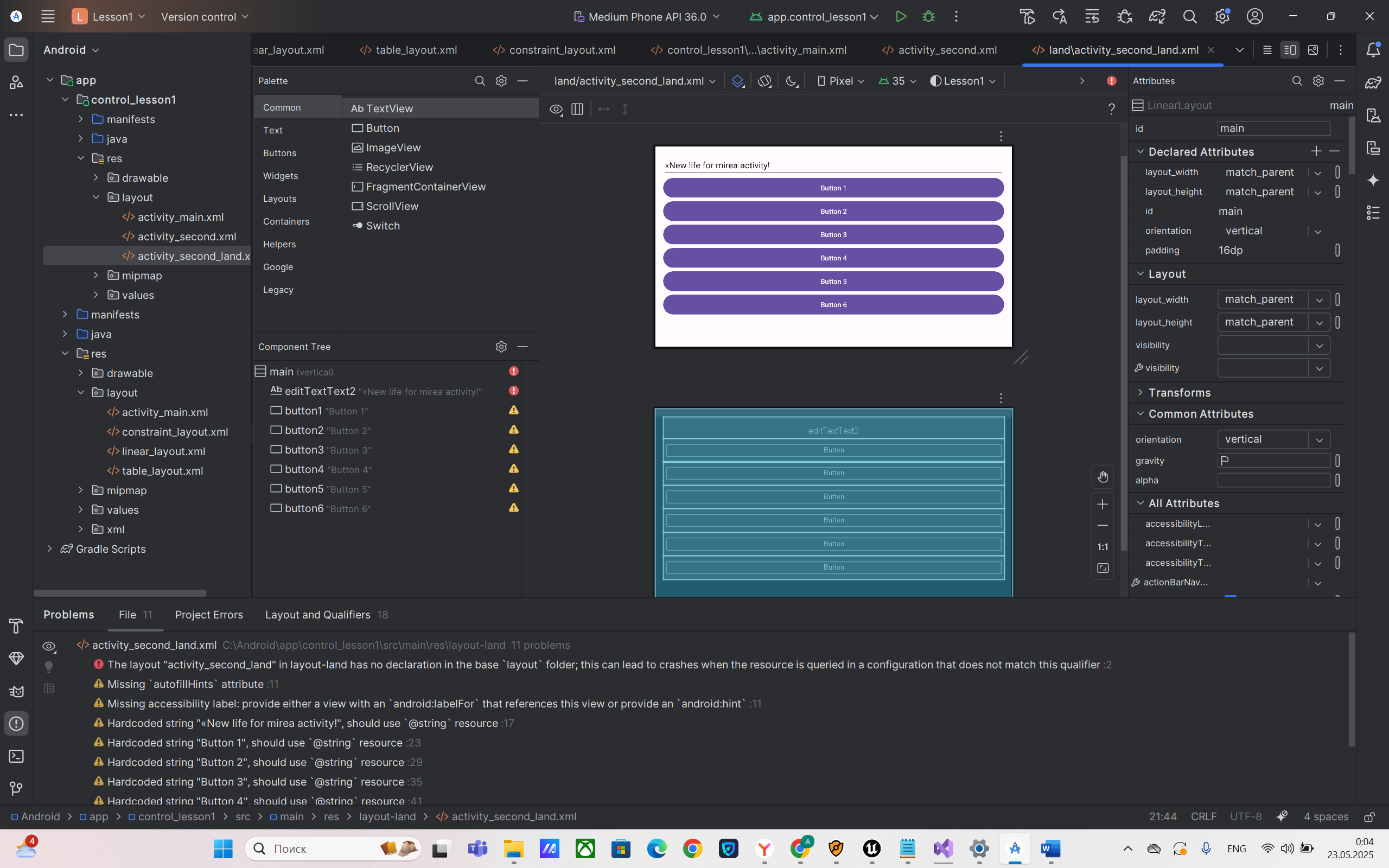


Рисунок 11. Горизонтальная ориентация

Далее требуется открыть разметку «activity\_main.xml» и добавить «TextView» с идентификатором «id="@+id/tvOut"» и две кнопки: «WhoAmI» (id="@+id/btnWhoAmI") и «ItIsNotMe» (id="@+id/btnItIsNotMe ") (см. рис. 12).

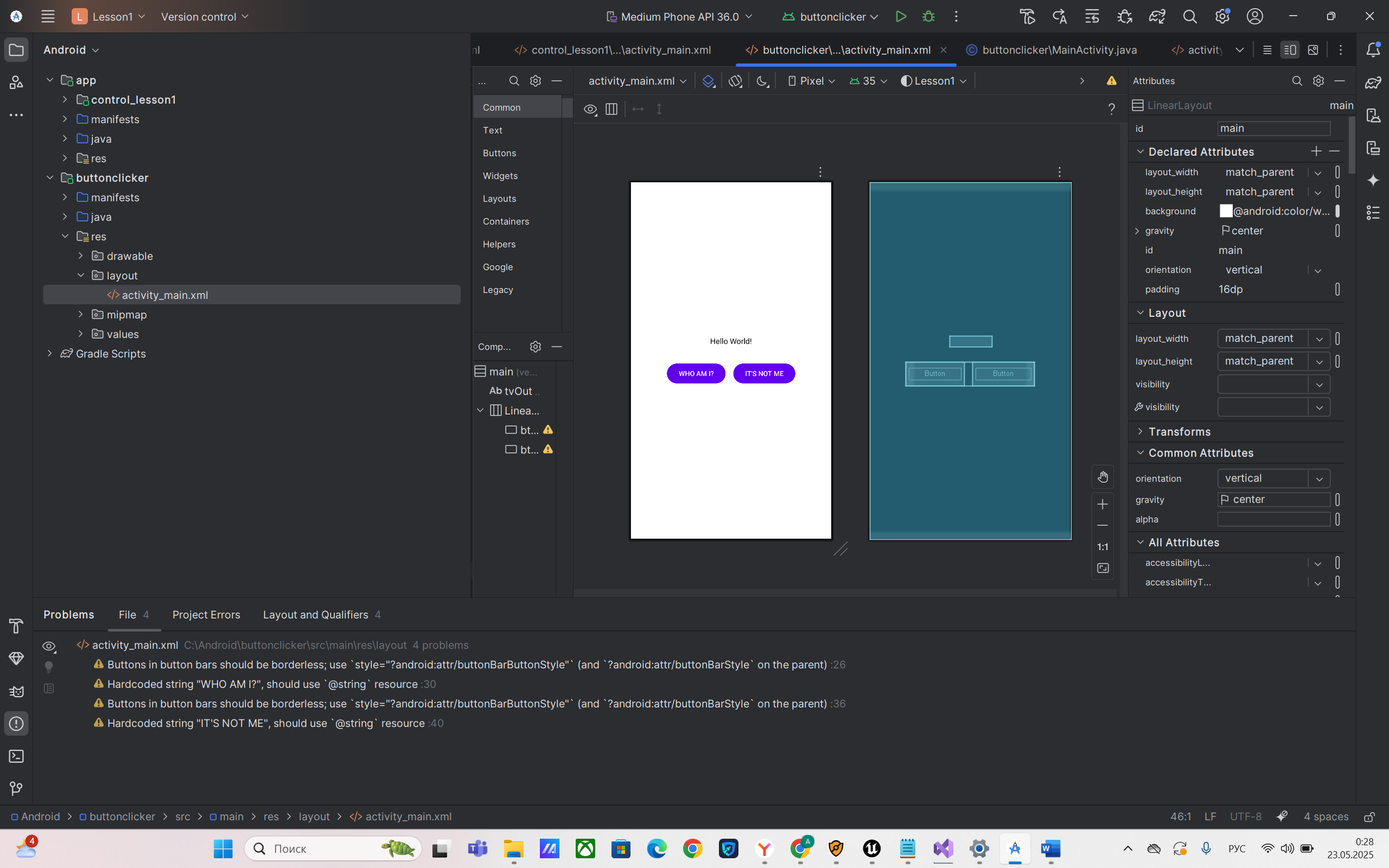


Рисунок 12. Добавление двух кнопок

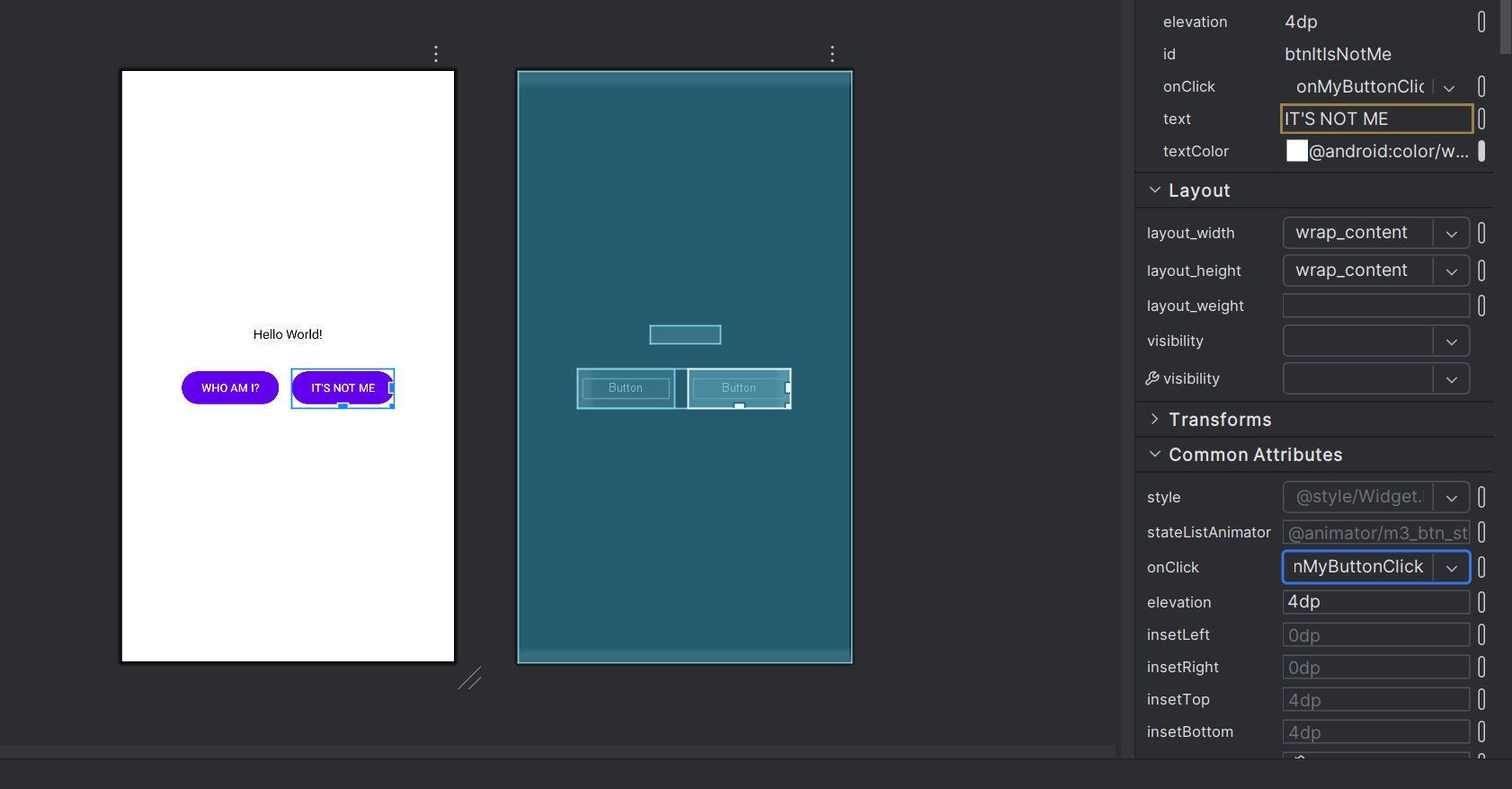
Воспользуемся альтернативным способом обработки событий нажатий - атрибутом «onClick» в xml (на панели свойств отображается как On Click) (см. рис. 13).

Рисунок 13. Обработка событий нажатий

В этом фрагменте кода представлен метод onlyButtonClick, который выполняет несколько действий при нажатии на кнопку . Переключает состояние CheckBox, если checkBox не null, его состояние инвертируется (с true на false и наоборот). Также обновляет TextView — находит TextView с id tvOut и выводит в него текущее состояние CheckBox (см. рис. 14).



Рисунок 14. Метод onlyButtonClick