

Лабораторная работа №18

Настройка кнопок

1 Цель работы

1.1 Изучить процесс настройки и использования элементов управления кнопок в приложениях Android.

2 Литература

2.1 Jetpack Compose | Руководство. metanit.com – Текст : электронный // metanit.com, 2023. – URL: <https://metanit.com/kotlin/jetpack/> – гл.5.

3 Подготовка к работе

3.1 Повторить теоретический материал (см.п.2).

3.2 Изучить описание лабораторной работы.

4 Основное оборудование

4.1 Персональный компьютер.

5 Задание

5.1 Обработка нажатия кнопки

5.1.1 Создать экран авторизации. Разместить на нем: TextField для ввода логина и пароля, кнопку «Авторизоваться». Поля ввода должны быть однострочными.

5.1.2 При вводе значений они должны отображаться в полях ввода.

5.1.3 При нажатии на кнопку отобразить в метке под кнопкой текст «Логин, вы авторизованы» (вместо логина должно отображаться значение из поля ввода логина).

5.1.4 Настроить скругление у кнопки.

5.1.5 Сравнить внешний вид интерфейса при использовании Button, OutlinedButton, TextButton.

5.2 Изменение внешнего вида кнопки после нажатия

5.2.1 Создать экран отображения товара и разместить на нем:

- метки с названием и ценой товара

- кнопку с текстом «Добавить в корзину»

При нажатии на кнопку ее подпись должна меняться на «Перейти в корзину», а цвет – становиться другим (например, после нажатия цвет стал белым, контур остался исходного цвета).

5.3 Настройка изображения на кнопке

5.3.1 Создать экран отображения товара и разместить на нем:

- метки с названием и ценой товара

- кнопку с изображением корзины (картинку с корзиной загрузить в ресурсы) и текстом «Добавить»

- метку для вывода количества товаров в корзине (по умолчанию – 0, при нажатии на кнопку должно увеличиваться на 1).

5.4 Использование кнопок-иконок IconButton

5.4.1 Создать экран отображения заказанного товара и разместить на нем:

- метки с названием и ценой товара
- метку для отображения заказанного количества (по умолчанию – 1)
- метку с итоговой стоимостью
- IconButton с пиктограммой + (Icons.Filled.Add) для увеличения

заказанного количества

- IconButton с пиктограммой – (Icons.Filled.Delete) для уменьшения заказанного количества.

5.4.2 Реализовать изменение значений в метках с заказанным количеством и итоговой стоимостью при нажатии на кнопки IconButton +/-.

5.4.3 Кнопка + должна быть недоступна, если заказанное количество 10, кнопка – должна быть недоступна, если заказанное количество 1.

5.5 Использование всплывающих кнопок FloatingActionButton

5.5.1 Создать экран отображения данных о приложении, разместить на нем:

- FloatingActionButton с пиктограммой звонка (Icons.Filled.Call) для звонка

- FloatingActionButton с пиктограммой email (Icons.Filled.Email) для отправки email

5.6 Использование всплывающих кнопок ExtendedFloatingActionButton

5.6.1 Создать экран отображения товара и разместить на нем:

- метки с названием и ценой товара

- ExtendedFloatingActionButton с пиктограммой + (Icons.Filled.Add) и текстом «Добавить» для добавления товара в корзину.

5.6.2 При нажатии на кнопку должны меняться ее пиктограмма и подпись на Icons.Filled.Delete и «Удалить» соответственно. Для реализации создать и использовать логическое значение:

```
val isAdded = remember{ mutableStateOf(false) }.
```

6 Порядок выполнения работы

6.1 Выполнить все задания из п.5.

6.2 Ответить на контрольные вопросы.

7 Содержание отчета

7.1 Титульный лист

7.2 Цель работы

7.3 Ответы на контрольные вопросы

7.4 Вывод

8 Контрольные вопросы

8.1 Как объявить button?

8.2 Что такое «IconButton»?

8.3 Что такое «IconButtonToggle»?

8.4 Что такое «FloatingActionButton»?

8.5 Что такое «ExtendedFloatingActionButton»?

8.6 Какие параметры нужно указать при создании объекта Icon?