Министерство образования Республики Беларусь

Учреждение образования  
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

Лабораторная работа №3

«Управление жестами в Android»

Студент Д. А. Снитко

Проверил О. М. Внук

МИНСК 2024

**1 ЦЕЛЬ РАБОТЫ**

1. Определить не менее 2-3 сложных действий для управления интерфейсом приложения и добавить их в реализуемое мобильное приложение.

2. Продемонстрировать работоспособность жестов.

**2 ИСХОДНЫЕ ДАННЫЕ**

Среда разработки для Android;

Язык программирования Kotlin;

Источник исходного кода: https://github.com/Luflexia/Currency-Converter

**3 ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ**

Жесты в Android — это механизмы взаимодействия пользователя с приложением через касания, перетаскивания, свайпы и другие виды сенсорного ввода. Система жестов в Android позволяет обрабатывать различные пользовательские действия для создания интуитивного и удобного интерфейса. Основные типы жестов включают:

Касание (Tap) — самое простое действие, используется для выбора элементов интерфейса.

Долгое нажатие (Long Press) — применяется для активации контекстного меню или дополнительного функционала.

Свайп (Swipe) — жест горизонтального или вертикального движения пальцем, часто используется для удаления элементов или переключения между экранами.

Перетаскивание (Drag) — позволяет пользователю перемещать объекты, что удобно для реорганизации элементов списка.

Щипок (Pinch) — жест двух пальцев, используется для увеличения или уменьшения масштаба.

Для обработки жестов в Android можно использовать классы и интерфейсы, такие как GestureDetector, OnGestureListener, View.OnTouchListener, а также специализированные помощники для обработки сложных жестов в списках, такие как ItemTouchHelper. В данной лабораторной работе использован ItemTouchHelper для реализации двух типов жестов в списке валют: перетаскивания и свайпа.

ItemTouchHelper — это утилита, которая позволяет добавлять интерактивные жесты в списки, управляемые RecyclerView. С его помощью можно реализовать, например, перетаскивание элементов для изменения порядка или свайп для удаления.

RecyclerView — это мощный компонент для отображения больших наборов данных, поддерживающий адаптивные списки и работу с жестами.

Преимущество использования RecyclerView и ItemTouchHelper заключается в высокой производительности и гибкости при работе с динамическими списками, а также возможности легко настраивать поведение списка под конкретные задачи приложения.

**4 КОД ПРОГРАММЫ**

Файл GestureHelper.kt

package com.example.currencyconverterv2.utils

import androidx.recyclerview.widget.ItemTouchHelper

import androidx.recyclerview.widget.RecyclerView

import com.example.currencyconverterv2.adapters.CurrencyAdapter

class CurrencyGestureHelper(private val adapter: CurrencyAdapter) : ItemTouchHelper.Callback() {

override fun getMovementFlags(

recyclerView: RecyclerView,

viewHolder: RecyclerView.ViewHolder

): Int {

val dragFlags = ItemTouchHelper.UP or ItemTouchHelper.DOWN // Перетаскивание по вертикали

val swipeFlags = ItemTouchHelper.LEFT // Свайп влево для удаления

return makeMovementFlags(dragFlags, swipeFlags)

}

override fun onMove(

recyclerView: RecyclerView,

viewHolder: RecyclerView.ViewHolder,

target: RecyclerView.ViewHolder

): Boolean {

val fromPosition = viewHolder.adapterPosition

val toPosition = target.adapterPosition

adapter.moveCurrency(fromPosition, toPosition) // Перемещение валют

return true

}

override fun onSwiped(viewHolder: RecyclerView.ViewHolder, direction: Int) {

val position = viewHolder.adapterPosition

adapter.removeCurrency(position) // Удаление валюты из списка

}

}

Файл CurrencyAdapter.kt

// Перемещение валюты

fun moveCurrency(fromPosition: Int, toPosition: Int) {

val fromCurrency = currencies.removeAt(fromPosition)

currencies.add(toPosition, fromCurrency)

notifyItemMoved(fromPosition, toPosition)

}

// Удаление валюты

fun removeCurrency(position: Int) {

currencies.removeAt(position)

notifyItemRemoved(position)

}

**5 ДЕМОНСТРАЦИЯ РАБОТЫ ПРИЛОЖЕНИЯ**

Слева жест перетаскивания курсов, а справа жест удаления свайпом влево на

рисунке 1

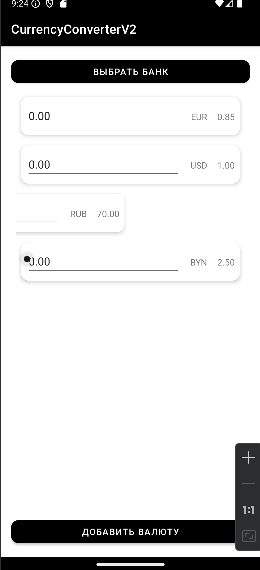
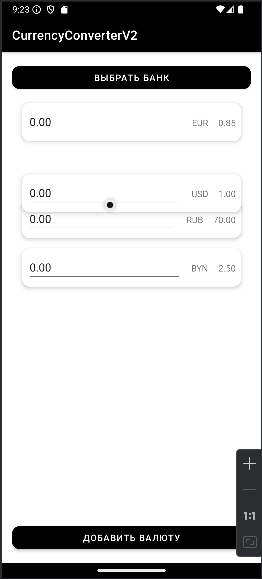


Рисунок 1

**6 ВЫВОД**

В ходе выполнения данной лабораторной работы был изучен и реализован механизм управления жестами в Android с использованием ItemTouchHelper. Были добавлены такие сложные жесты, как перетаскивание элементов списка и их удаление с помощью свайпа влево.