Санкт-Петербургский политехнический университет Петра Великого

Институт компьютерных наук и технологий

Высшая школа интеллектуальных систем и суперкомпьютерных технологий

**Отчёт по лабораторной работе №2**

**Дисциплина**: Проектирование мобильных приложений

**Тема**: “ Activity Lifecycle. Alternative resources. ”

Выполнил студент гр. 3530901/80202 Смирнов Н. В.

(подпись)

Преподаватель Алексюк А. О. (подпись)

“\_\_\_”\_\_\_\_\_\_\_\_\_\_\_\_\_ 2020 г.

Санкт-Петербург

2020

# Цели

* Ознакомиться с жизненным циклом Activity
* Изучить основные возможности и свойства alternative resources

# Задачи

## Задача 1. Activity

### Задание

Продемонстрируйте жизненный цикл Activity на любом нетривиальном примере.

### Выполнение задания

Для демонстрации были выбраны два примера - выключение телефона и звонок вовремя работы приложения.

1.Выключение телефона

При включении приложения сработал метод **onCreate()**, **onStart()**, **onResume()**. При выключении телефона сработал метод **onPause()**, **onSaveInstanceState()**, **onStop()**

2. Звонок вовремя работы приложения

При запуске программы вызвался метод **onCreate()**, **onStart()**, **onResume()**. При звонке на верху экрана всплыло оповещение со звонком и никаких методов не вызвалось. При ответе на звонок сработал метод **onPause()**, **onSaveInstanceState()** и **onStop()**. Когда звонок завершился сработал метод **onRestart()**, **onStart()**, **onResume()**.

3. Вызов Google Assistant

При включении приложения сработал метод **onCreate()**, **onStart()**, **onResume()**. Когда пользователь вызывает Google Assistant, происходит вызов метода **onPause()**, **onSaveInstanceState()**, **onStop()**. При возращении обратно в приложение вызывается метод **onRestart**, **onStart()**, **onResume()**.

## Задача 2. Alternative Resources

## Задание

### Задание

Продемонстрируйте работу альтернативного ресурса (тип ресурса согласно [варианту](https://github.com/andrei-kuznetsov/android-lectures/blob/master/labs/02/VARIANTS02.md)) на каком-либо примере.

Вариант 13: Navigation key availability.

### Выполнение задания

## Задача 3. Best-matching resourceе

### Задание

Для заданного набора альтернативных ресурсов, предоставляемых приложением, и заданной конфигурации устройства (оба параметра согласно [варианту](https://github.com/andrei-kuznetsov/android-lectures/blob/master/labs/02/VARIANTS03.txt)) объясните, какой ресурс будет выбран в конечном итоге. Ответ докажите.

Вариант: 13.

### Выполнение задания

|  |  |
| --- | --- |
| Конфигурация устройства: | Конфигурация ресурсов: |
| LOCALE\_LANG: fr | **(default)** |
| LOCALE\_REGION: rUS | fr-small-port-finger-nokeys-trackball-v25 |
| SCREEN\_SIZE: small | ~~notlong-port-night-finger-v27~~ |
| SCREEN\_ASPECT: long | ~~long-vrheadset-12key~~ |
| ROUND\_SCREEN: round | ~~vrheadset-night-notouch-v27~~ |
| ORIENTATION: land | fr |
| UI\_MODE: vrheadse | ~~en-notlong-round-dpad~~ |
| NIGHT\_MODE: notnight | ~~large-wheel-v26~~ |
| PIXEL\_DENSITY: mdpi | ~~fr-rFR-xlarge-notround-vrheadset-night-qwerty-dpad~~ |
| TOUCH: finger | ~~rUS-notlong-round-car~~ |
| PRIMARY\_INPUT: 12key | ~~notlong-round-port-car-nodpi-finger~~ |
| NAV\_KEYS: wheel |  |
| PLATFORM\_VER: v25 |  |

## Задача 4. Сохранение состояния Activity.

### Задание

Студент написал приложение: [continuewatch](https://github.com/andrei-kuznetsov/android-lectures/blob/master/labs/02/continuewatch). Это приложение [по заданию](https://github.com/andrei-kuznetsov/android-lectures/blob/master/labs/02/continuewatch/README.md) должно считать, сколько секунд пользователь провел в этом приложении.

Найдите ошибки в этом приложении и исправьте их.

### Выполнение задания

Первая ошибка – при сворачивании приложения секунды продолжают идти.

Вторая ошибка - при перевороте экрана секунды сбрасываются.

Третья ошибка – при перевороте экрана показывается надпись “Hello World!” или “TextView”.

Исправленный код можно посмотреть ниже в приложении 2.

# Вывод

Затраченное время на выполнение работы и составление отчета - 10 часов.

Первое задание оказалось простым и помогло разобраться с понимание работы жизненного цикла Activity.

Возникла проблема с Navigation key availability. Оказалось, всё довольно просто, достаточно было добавить новую папку в res.

Task1 – 2 часа – задание простое, но долго осознавал, как работает жизненный цикл.

Task2 – 4 часа – долго пытался продемонстрировать пример в эмуляторе, но так как это довольно редкий и неиспользуемый ресурс, то я не смог найти нужного ответа.

Task3 – 1 час – задание довольно простое и больше включает в себя внимательность.

Task4 – 3 часа – долго разбирался с проблемой переворота экрана и с проблемой когда при перевороте экрана вместо продолжения таймера показывался текст на секунду, а потом продолжался таймер.

# Приложения

### Приложение 1

package com.smirnov.lab2android  
  
import android.os.Bundle  
import android.util.Log  
import androidx.appcompat.app.AppCompatActivity  
  
class MainActivityForTask1 : AppCompatActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_main*)  
 Log.d("myLogs", "onCreate()")  
 }  
  
 override fun onDestroy() {  
 super.onDestroy()  
 Log.d("myLogs", "onDestroy()")  
 }  
  
 override fun onPause() {  
 super.onPause()  
 Log.d("myLogs", "onPause()")  
 }  
  
 override fun onRestart() {  
 super.onRestart()  
 Log.d("myLogs", "onRestart()")  
 }  
  
 override fun onRestoreInstanceState(savedInstanceState: Bundle) {  
 super.onRestoreInstanceState(savedInstanceState)  
 Log.d("myLogs", "onRestoreInstanceState()")  
 }  
  
 override fun onResume() {  
 super.onResume()  
 Log.d("myLogs", "onResume()")  
 }  
  
 override fun onSaveInstanceState(outState: Bundle) {  
 super.onSaveInstanceState(outState)  
 Log.d("myLogs", "onSaveInstanceState()")  
 }  
  
 override fun onStart() {  
 super.onStart()  
 Log.d("myLogs", "onStart()")  
 }  
  
 override fun onStop() {  
 super.onStop()  
 Log.d("myLogs", "onStop()")  
 }  
}

### Приложение 2

package com.smirnov.lab2android  
  
import android.annotation.SuppressLint  
import android.os.Bundle  
import androidx.appcompat.app.AppCompatActivity  
import kotlinx.android.synthetic.main.activity\_main.\*  
  
class MainActivityForTask4 : AppCompatActivity() {  
 var secondsElapsed: Int = 0  
 var flag = false  
  
 @SuppressLint("SetTextI18n")  
 var backgroundThread = Thread **{** while (true) {  
 Thread.sleep(1000)  
 if (flag) {  
 textSecondsElapsed.post **{** textSecondsElapsed.*text* = "Seconds elapsed: " + secondsElapsed++  
 **}** }  
 }  
 **}** override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_main*)  
 flag = true  
 backgroundThread.start()  
  
 }  
  
 override fun onResume() {  
 super.onResume()  
 flag = true  
 }  
  
 override fun onPause() {  
 super.onPause()  
 flag = false  
 }  
  
 override fun onSaveInstanceState(outState: Bundle) {  
 super.onSaveInstanceState(outState)  
 outState.putInt("count", secondsElapsed)  
 }  
  
 @SuppressLint("SetTextI18n")  
 override fun onRestoreInstanceState(savedInstanceState: Bundle) {  
 super.onRestoreInstanceState(savedInstanceState)  
 secondsElapsed = savedInstanceState.getInt("count")  
 textSecondsElapsed.*text* = "Seconds elapsed: " + secondsElapsed++  
 }  
}