

**ИКОНОМИЧЕСКИ УНИВЕРСИТЕТ – ВАРНА**

**ФАКУЛТЕТ „ИНФОРМАТИКА“**

**КАТЕДРА „ИНФОРМАТИКА“**



**Курсова работа по Мобилни приложения**

**на тема**

**„Пътеводител на Варна“**

**Изготвил:**

**Карина Панчул**

**Курс: 4**

**Група: 45**

**Фак. №: 117971**

**ВАРНА, 2023**

# Съдържание

<b>I.</b>	<b>Описание .....</b>	<b>3</b>
<b>II.</b>	<b>Екрани и код.....</b>	<b>3</b>
1.	Задаване на икона и име на приложението .....	3
2.	Използване на навигация.....	4
3.	Използване на 3 или повече екрана.....	4
4.	Използване на различни видове контроли .....	5
5.	Прихващане на различни събития на два или повече контрола .....	5
6.	Визуализиране на диалогов прозорец на потребителя и прихващане на избора му...	6
7.	Стартиране на друго приложение чрез Intent или работа с услуги.....	7
8.	Изграждане на екран чрез RecyclerView .....	8
9.	Визуализиране на данни от текстови файл или база от данни .....	10
<b>III.</b>	<b>Допълнителни функции.....</b>	<b>10</b>

## I. Описание

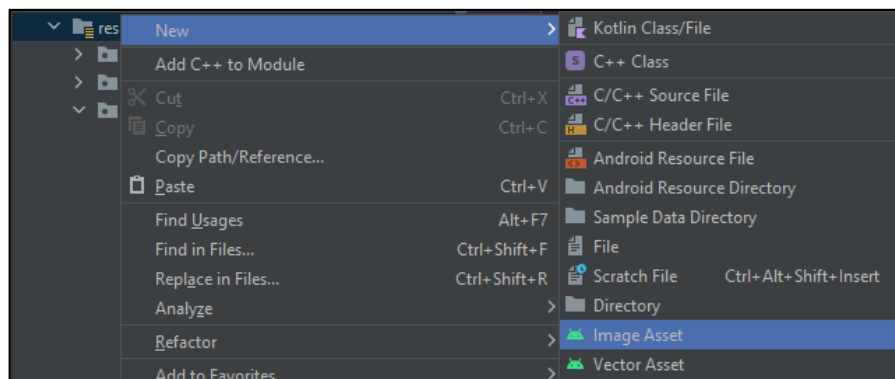
Приложението е създадено с цел да даде информация на потребителя за град Варна и местата, които биха били интересни за посещение. Те са групирани по категории: „Разходки“, „Снимки“ и „Плаж“, като включват снимка и кратко описание за мястото. Също така има отделна страница, която съдържа интересни факти и описание за Варна. Приложението ще бъде полезно както за туристи, така и за местните жители, които биха желали да посетят избраната от тях дестинация.

## II. Екрани и код

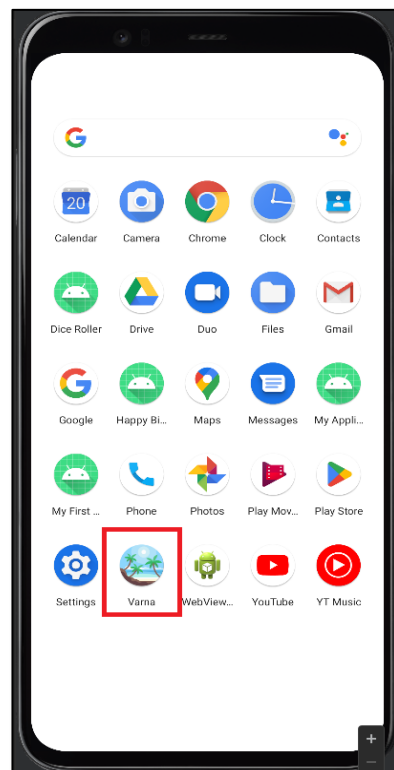
### 1. Задаване на икона и име на приложението

Промяна на иконата: десен бутон върху папката: res ⇒ New ⇒ Image Asset (Фиг. 1)

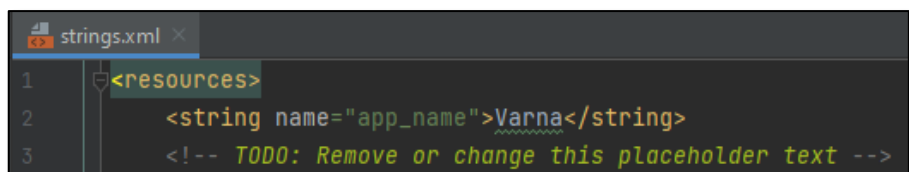
Наименование на приложението се сменя от папката: values ⇒ strings.xml (Фиг. 2)



Фиг. 1



Фиг. 3

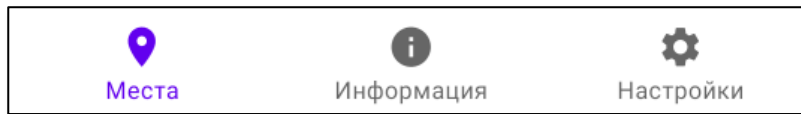


Фиг. 2

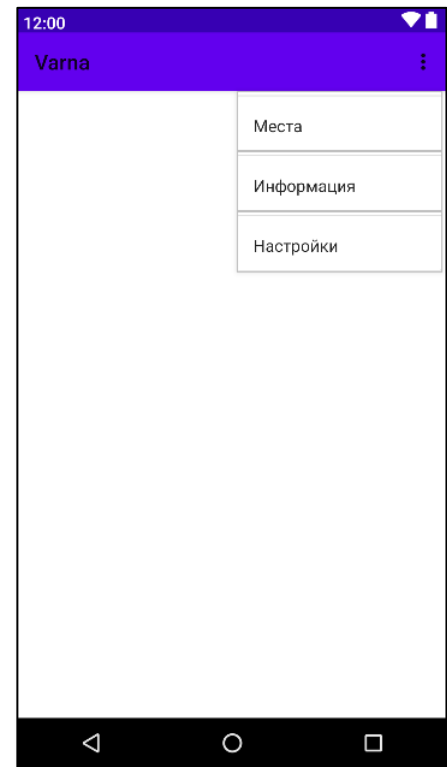
## 2. Използване на навигация

За навигация е използвано: BottomNavigationView.

Създадено е меню: mymenu.xml (Фиг. 5) в activity\_main.xml и menu\_places.xml (Фиг. 6) във fragment\_frag2.xml.



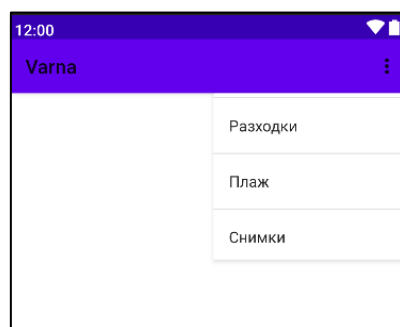
Фиг. 4



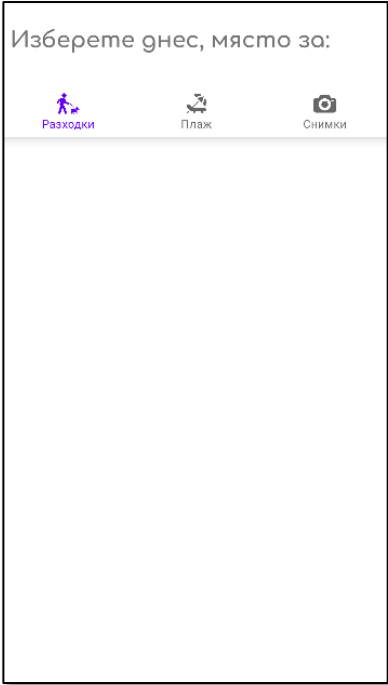

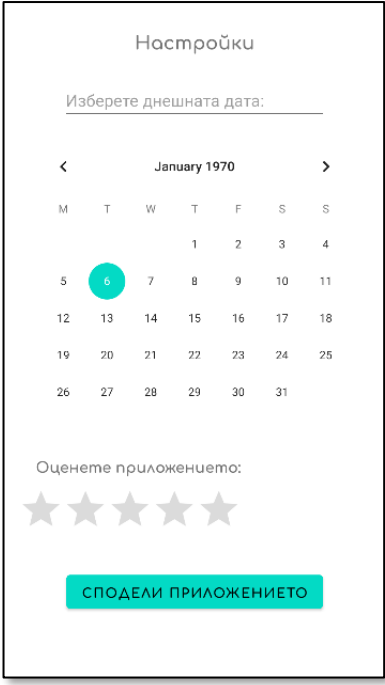
Фиг. 5

## 3. Използване на 3 или повече екрана

Създадени са 3 фрагмента, като на Фиг. 7 са включени още 3 фрагмента от меню: menu\_places.xml (Фиг. 6).

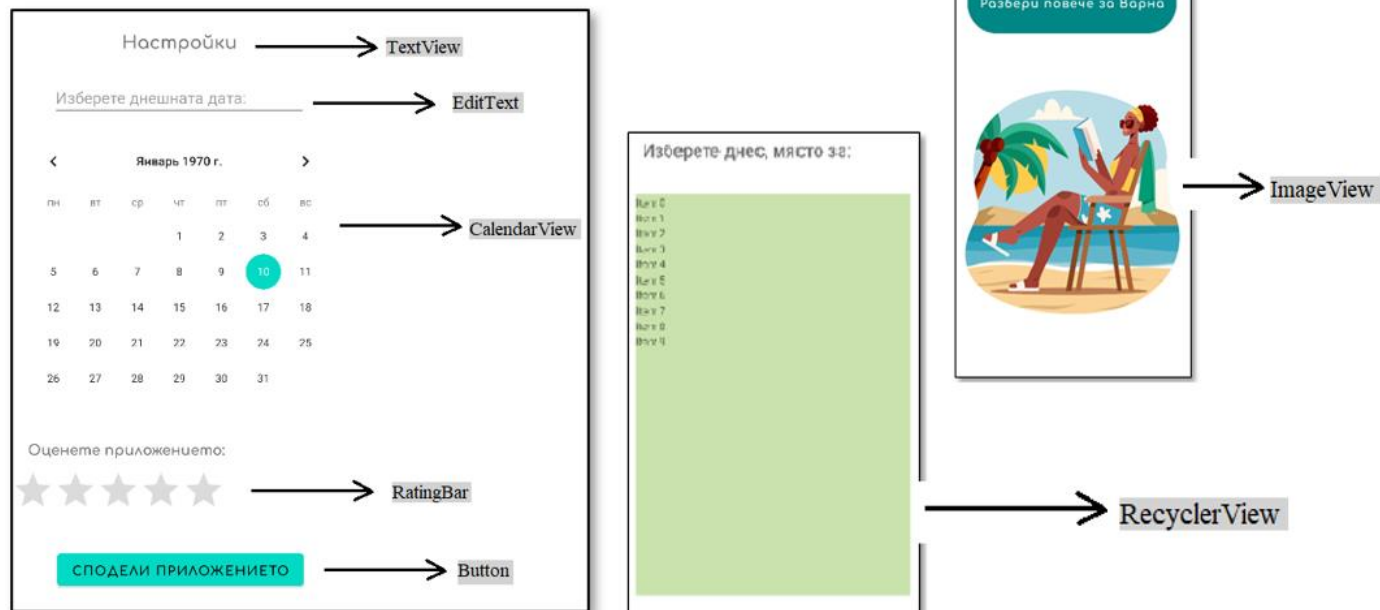


Фиг. 6

fragment_frag2	fragment_frag3	fragment_frag4
 <p>Изберете днес, място за:</p> <p>Разходки Плаж Снимки</p>	 <p>Разбери повече за Варна</p>	 <p>Настройки</p> <p>Изберете днешната дата:</p> <p>January 1970</p> <p>Оценете приложението:</p> <p>СПОДЕЛИ ПРИЛОЖЕНИЕТО</p>
Фиг. 7	Фиг. 8	Фиг. 9

#### 4. Използване на различни видове контроли

Използвани са контролите: TextView, **CalendarView**, EditText, **RatingBar**, Button, ImageView, RecyclerView. (Фиг. 10)



Фиг. 10

## 5. Прихващане на различни събития на два или повече контрола

Във фрагмент 4 (frag4.kt) са прихванати съответните събития:

```
override fun onCreateView(view: View, savedInstanceState: Bundle?) {
    super.onCreateView(view, savedInstanceState)

    val tvText = view.findViewById<TextView>(R.id.textView3)
    val tvOcenka = view.findViewById<TextView>(R.id.textView6)
    val rating = view.findViewById<RatingBar>(R.id.ratingBar)
    val date = view.findViewById<EditText>(R.id.editTextDate)

    rating.setOnRatingBarChangeListener { ratingBar, fl, b ->
        val oценка = rating.rating
        tvOcenka.text = oценка.toString()
    }

    val cv=view.findViewById<CalendarView>(R.id.calendarView)
    cv.setVisibility(View.INVISIBLE)

    view.findViewById<EditText>(R.id.editTextDate).setOnFocusChangeListener { view, b ->
        cv.isVisible=true
    }

    cv.setOnDateChangeListener { calendarView, year, month, day ->
        val izbor=String.format("%d.%02d.%d",day,month+1,year)
        // На активният контрол му задаваме стойност
        date.setText(izbor)
    }
}
```

## 6. Визуализиране на диалогов прозорец на потребителя и прихващане на избора му

Създаден е диалогов прозорец, в който се визуализира описанието на град Варна и има възможност да се смени цвета на бутона, който е показал диалога. (Фиг. 11)

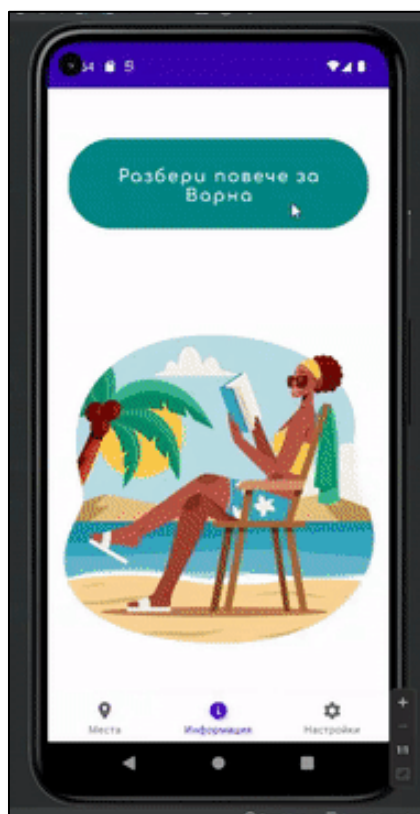
```
fun btn_razberi(view: View)
{
    val btn = findViewById<Button>(R.id.button3)
    val dlg= AlertDialog.Builder(this)
    dlg.setTitle("Варна - морската столица")
    dlg.setMessage("    Варна (Varna) е третият по големина град в България - Морската столица. Разположен е в "+
        "Североизточна България. Варна е туристически център, отправна точка за много курорти по Северното Черноморие. "+
        "Варна е център на много културни прояви като фестивалът „Варненско лято“, кинофестивалът „Любовта е лудост“.\n"+
        "    Историята на града е хилядолетна. Варна е носила името Одесос и според ранни писмени сведения градът е основан " +
        "от изселници от малоазийския град Милет през VI в. пр.н.е. Това е периодът на Великата гръцка колонизация.\n" +
        "    Интересни забележителности в града са: Морската градина, Делфинариума, Астрономическа обсерватория и планетариум " +
```

```

"Николай Коперник, Варненският некропол – най-старото златно съкровище в света,
Варненски археологически музей, " +
"Военноморски музей, Музей за нова история, Етнографски музей, Музей на
Възраждането.\n" +
"    В околностите на Варна се намират: Аладжа манастир – скален манастир, Побитите
камъни, Резерват Камчия.")

dlg.setPositiveButton("Задай син цвят на бутона") { dialog, which ->
    btn.setBackgroundColor(Color.BLUE)
}
dlg.setNegativeButton("Задай червен цвят на бутона") { dialog, which ->
    btn.setBackgroundColor(Color.RED)
}
dlg.show()
}

```



Фиг. 11

## 7. Стартиране на друго приложение чрез Intent или работа с услуги

В последния фрагмент има включена функция за работа с Intent, като се създава нов прозорец, в който имаме възможност да споделим приложението. (Фиг. 12)

```

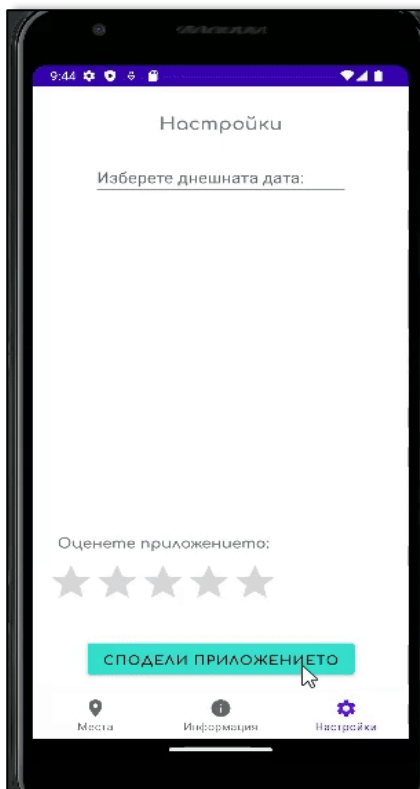
fun btn_spodeli(view: View)
{
    val btn = findViewById<Button>(R.id.button)
    val sendIntent: Intent = Intent().apply {
        action = Intent.ACTION_SEND
    }
}

```

```

        putExtra(Intent.EXTRA_TEXT, "Искам да споделя това приложение!")
        type = "text/plain"
    }
    val shareIntent = Intent.createChooser(sendIntent, null)
    startActivity(shareIntent)
}

```

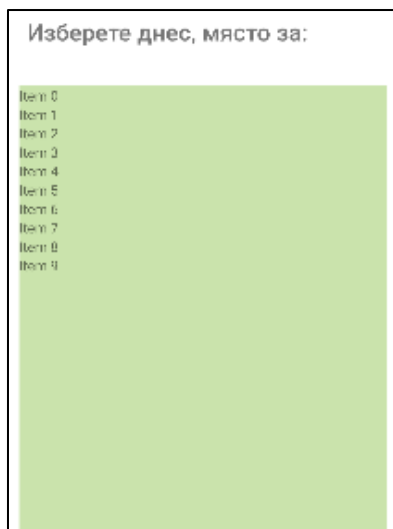


Фиг. 12

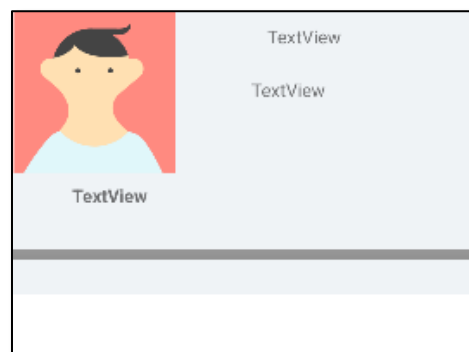
## 8. Изграждане на екран чрез RecyclerView

Във `fragment_frag2.xml`, който се показва при кликване на бутона „Места“, имаме меню, което ни навигира по категории за възможните дестинации за посещение. Менюто включва 3 фрагмента и във всеки един от тях е включено `RecyclerView`, което зарежда съответните данни. (Фиг. 13)

Създадени са класовете: `mymanager` и `places` (data class). `layout_red.xml` – по този начин ще се визуализират данните във `RecyclerView`. (Фиг. 14)



Фиг. 14



Фиг. 13



## Клас mymanager:

```
class mymanager(val myplaces:ArrayList<places>):
RecyclerView.Adapter<mymanager.ViewHolder>() {

    override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): ViewHolder {
        return
ViewHolder(LayoutInflater.from(parent.context).inflate(R.layout.layout_red, parent,
false))
    }

    override fun onBindViewHolder(holder: ViewHolder, position: Int) {
        val elem:places = myplaces[position]
        holder.bind(elem)
    }

    override fun getItemCount(): Int {
        return myplaces.size
    }

    class ViewHolder(itemView: View) : RecyclerView.ViewHolder(itemView) {
        fun bind(elem: places) {
            itemView.findViewById<TextView>(R.id.textView).text = elem.title
            itemView.findViewById<TextView>(R.id.textView2).text = elem.desc
            itemView.findViewById<TextView>(R.id.textView7).text = "Рейтинг:
"+elem.reiting.toString()

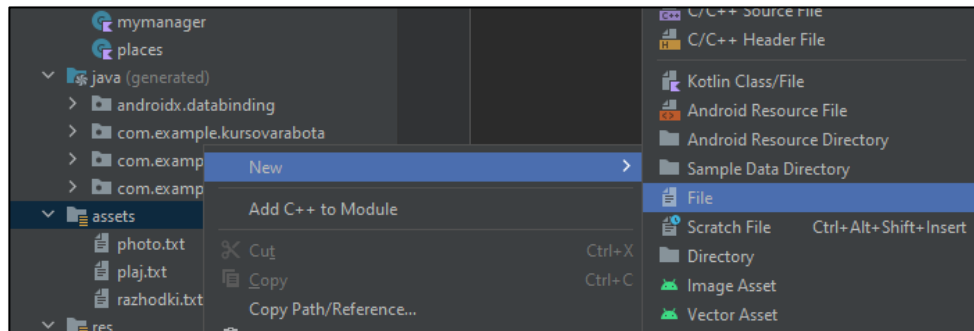
            when (elem.place) {
                1 ->
itemView.findViewById<ImageView>(R.id.imageView3).setImageResource(R.drawable.azalia)
                2 ->
itemView.findViewById<ImageView>(R.id.imageView3).setImageResource(R.drawable.drujba)
                3 ->
itemView.findViewById<ImageView>(R.id.imageView3).setImageResource(R.drawable.kolelo_mors
ka_gara)
                4 ->
itemView.findViewById<ImageView>(R.id.imageView3).setImageResource(R.drawable.morska_grad
ina)
                5 ->
itemView.findViewById<ImageView>(R.id.imageView3).setImageResource(R.drawable.gledka_gala
ta)
                6 ->
itemView.findViewById<ImageView>(R.id.imageView3).setImageResource(R.drawable.fara)
            }
        }
    }
}
```

## Клас places:

```
data class places(val title:String, val desc:String, val reiting:Double, val place:Int)
```

## 9. Визуализиране на данни от текстови файл или база от данни

Създадена е папката assets, а в нея 3 текстови файла (Фиг. 15), информацията, от които се визуализира чрез RecyclerView.



Фиг. 15

Във фрагментите се изпълнява този код за визуализация на данните от файловете:

```
override fun onCreateView(view: View, savedInstanceState: Bundle?) {
    super.onCreateView(view, savedInstanceState)

    val book: ArrayList<places> = ArrayList()
    resources.assets.open("razhodki.txt").use { fp ->
        fp.bufferedReader().lines().forEach() {
            val elem = it.split(';')
            book.add(places(elem[0], elem[1], elem[2].toDouble(), elem[3].toInt()))
        }
    }
    val recycleAdapter = mymanager(book)
    val rv = view.findViewById<RecyclerView>(R.id.my_rec_view)
    rv.adapter = recycleAdapter
    rv.layoutManager = LinearLayoutManager(requireContext())
}
```

## III. Допълнителни функции

При стартиране на приложението се зарежда LoginActivity, което чрез Intent се визуализира за 2 секунди, след което ни препраща към главната страница с местата. (Фиг. 16)

```
class LoginActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_login)
        Handler().postDelayed({
            val intent=Intent(this, MainActivity::class.java)
            startActivity(intent)
            finish()
        },2000)
    }
}
```



Фиг. 16

За преминаване между фрагментите е използван следният код (във frag2.kt):

```
override fun onCreateView(view: View, savedInstanceState: Bundle?) {
    super.onCreateView(view, savedInstanceState)

    val book: ArrayList<places> = ArrayList()
    val page1 = frag_place_1()
    val page2 = frag_place_2()
    val page3 = frag_place_3()
    Smqna_Stranici(page1)

    val lenta = view.findViewById<BottomNavigationView>(R.id.menu_places)

    lenta.setOnNavigationItemSelectedListener {
        when (it.itemId) {
            R.id.razhodki -> Smqna_Stranici(page1)
            R.id.play -> Smqna_Stranici(page2)
            R.id.snimki -> Smqna_Stranici(page3)
        }
        true
    }
}

private fun Smqna_Stranici(tekushta: Fragment) {
    childFragmentManager.beginTransaction().apply {
        replace(R.id.ramka, tekushta)
        commit()
    }
}
```