БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ   
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ   
Кафедра программного обеспечения информационных технологий

Факультет КСиС   
Специальность ПОИТ

Индивидуальная практическая работа по модулю 2   
по дисциплине «Разработка программного обеспечения для мобильных платформ»

Выполнил студент: Захаренко В. В.   
группа 791051   
Зачетная книжка No 5510047

Минск 2021

Индивидуальное задание

Вариант 18: добавить возможность добавлять свой источник информации помимо изначального

Скриншоты работы приложения

Изображение выглядит как текст

Автоматически созданное описание

Рис. 1 Стартовый экран со списком новостей

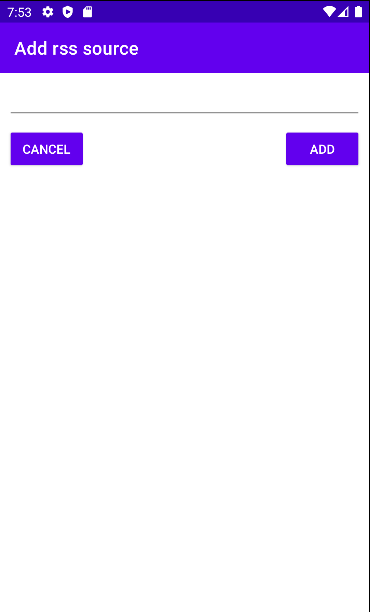


Рис. 2 Экран добавления источника новостей

Изображение выглядит как текст

Автоматически созданное описание

Рис. 3 Меню "Поделиться" новости

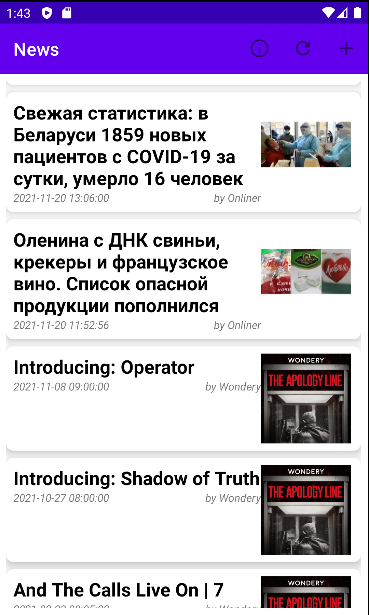


Рис. 4 Список новостей из нескольких источников

Изображение выглядит как текст

Автоматически созданное описание

Рис. 5 Страница с информацией о разработчике

Листинг исходного приложения

AddRssSourceActivity.kt

package org.kvpbldsck.rss.activities  
  
import android.app.Activity  
import android.content.Intent  
import androidx.appcompat.app.AppCompatActivity  
import android.os.Bundle  
import android.view.View  
import androidx.appcompat.widget.Toolbar  
import com.google.android.material.textfield.TextInputEditText  
import org.kvpbldsck.rss.R  
  
class AddRssSourceActivity : AppCompatActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_add\_rss\_source*)  
  
 getToolbar().*title* = "Add rss source"  
 }  
  
 fun addRssSource(view: View) {  
 val data = Intent()  
 data.putExtra("rss\_source", getRssSourceInput().*text*.*toString*())  
  
 setResult(Activity.*RESULT\_OK*, data)  
 finish()  
 }  
  
 fun close(view: View) {  
 val data = Intent()  
 setResult(Activity.*RESULT\_CANCELED*, data)  
 finish()  
 }  
  
 private fun getRssSourceInput() : TextInputEditText = findViewById(R.id.*add\_rss\_source\_input*)  
 private fun getToolbar() : Toolbar = findViewById(R.id.*add\_rss\_source\_toolbar*)  
}

MainActivity.kt

package org.kvpbldsck.rss.activities  
  
import android.app.Activity  
import android.content.Intent  
import androidx.appcompat.app.AppCompatActivity  
import android.os.Bundle  
import android.util.Log  
import android.view.Menu  
import android.view.MenuItem  
import androidx.activity.result.contract.ActivityResultContracts  
import androidx.appcompat.widget.Toolbar  
import androidx.recyclerview.widget.LinearLayoutManager  
import androidx.recyclerview.widget.RecyclerView  
import org.kvpbldsck.rss.R  
import org.kvpbldsck.rss.adapters.FeedAdapter  
import org.kvpbldsck.rss.common.RetrofitServiceGenerator  
import org.kvpbldsck.rss.models.RssObject  
import retrofit2.Call  
import retrofit2.Callback  
import retrofit2.Response  
  
class MainActivity : AppCompatActivity() {  
  
 private val \_rssSourceAddresses = *mutableListOf*("https://people.onliner.by/feed")  
 private var \_feedAdapter: FeedAdapter? = null  
  
 private val intentLauncher =  
 registerForActivityResult(ActivityResultContracts.StartActivityForResult()) **{** result **->** if (result.*resultCode* == Activity.*RESULT\_OK*) {  
 if (result.*data*?.hasExtra("rss\_source") == true) {  
 \_rssSourceAddresses += result.*data*!!.getStringExtra("rss\_source") ?: ""  
 loadAllRss()  
 }  
 }  
 **}** override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_main*)  
  
 \_feedAdapter = FeedAdapter(*baseContext*)  
 getRecyclerView().*adapter* = \_feedAdapter  
  
 getToolbar().*title* = "News"  
 setSupportActionBar(getToolbar())  
 val linearLayoutManager = LinearLayoutManager(  
 *baseContext*,  
 LinearLayoutManager.*VERTICAL*, false  
 )  
 getRecyclerView().*layoutManager* = linearLayoutManager  
 loadAllRss()  
 }  
  
 override fun onCreateOptionsMenu(menu: Menu): Boolean {  
 *menuInflater*.inflate(R.menu.*main\_menu*, menu)  
 return true  
 }  
  
 override fun onOptionsItemSelected(item: MenuItem): Boolean {  
 when (item.*itemId*) {  
 R.id.*menu\_refresh\_button* -> {  
 loadAllRss()  
 }  
  
 R.id.*menu\_add\_button* -> {  
 addRssSource()  
 }  
  
 R.id.*menu\_developer\_info* -> {  
 startActivity(Intent(this, DeveloperInfoActivity::class.*java*))  
 }  
 }  
 return true  
 }  
  
 private fun addRssSource() {  
 intentLauncher.launch(Intent(this, AddRssSourceActivity::class.*java*))  
 }  
  
 private fun loadAllRss() {  
 \_feedAdapter?.clearNews()  
 \_rssSourceAddresses.*forEach* **{** loadRss(**it**) **}** }  
  
 private fun loadRss(rssSource: String) {  
 val call = RetrofitServiceGenerator.createService().getFeed(rssSource)  
 call.enqueue(object : Callback<RssObject> {  
 override fun onFailure(call: Call<RssObject>?, t: Throwable?) {  
 Log.d("ResponseError", "failed")  
 }  
  
 override fun onResponse(call: Call<RssObject>?, response: Response<RssObject>?) {  
 if (response?.*isSuccessful* == true) {  
 response.body()?.*let* **{** rssObject **->** Log.d("Response", rssObject.toString())  
 \_feedAdapter?.addNews(rssObject.items)  
 **}** }  
 }  
 })  
 }  
  
 private fun getToolbar(): Toolbar = findViewById(R.id.*main\_toolbar*)  
 private fun getRecyclerView(): RecyclerView = findViewById(R.id.*main\_recycler\_view*)  
}

NewsItemActivity.kt

package org.kvpbldsck.rss.activities  
  
import android.content.Intent  
import androidx.appcompat.app.AppCompatActivity  
import android.os.Bundle  
import android.view.Menu  
import android.view.MenuItem  
import android.widget.TextView  
import androidx.appcompat.widget.Toolbar  
import org.kvpbldsck.rss.R  
import org.sufficientlysecure.htmltextview.HtmlHttpImageGetter  
import org.sufficientlysecure.htmltextview.HtmlTextView  
  
class NewsItemActivity : AppCompatActivity() {  
  
 var title = ""  
 var date = ""  
  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_news\_item*)  
  
 val author = *intent*.getStringExtra("author") ?: ""  
 val content = *intent*.getStringExtra("content") ?: ""  
 title = *intent*.getStringExtra("title") ?: ""  
 date = *intent*.getStringExtra("date") ?: ""  
  
 this.getToolbar().*title* = "Author - \n$author"  
 setSupportActionBar(getToolbar())  
  
 getTitleView().*text* = title  
 getPublishDateView().*text* = date  
  
 val contentView = getContentView()  
 contentView.setHtml(content, HtmlHttpImageGetter(contentView))  
 }  
  
 override fun onCreateOptionsMenu(menu: Menu): Boolean {  
 *menuInflater*.inflate(R.menu. *news\_item\_menu*, menu)  
 return true  
 }  
  
 override fun onOptionsItemSelected(item: MenuItem): Boolean {  
 if (item.*itemId* == R.id.*menu\_share\_button*) {  
 shareData()  
 }  
 return true  
 }  
  
 private fun shareData() {  
 val s = title + "\n" + date  
  
 val shareIntent = Intent()  
 shareIntent.*action* = Intent.*ACTION\_SEND* shareIntent.*type* = "text/plain"  
 shareIntent.putExtra(Intent.*EXTRA\_TEXT*, s)  
  
 startActivity(Intent.createChooser(shareIntent, s))  
 }  
  
 private fun getToolbar(): Toolbar = findViewById(R.id.*news\_item\_toolbar*)  
 private fun getTitleView(): TextView = findViewById(R.id.*news\_item\_title*)  
 private fun getPublishDateView(): TextView = findViewById(R.id.*news\_item\_publish\_date*)  
 private fun getContentView(): HtmlTextView = findViewById(R.id.*news\_item\_content*)  
  
}

DeveloperInfoActivity.kt

package org.kvpbldsck.rss.activities  
  
import androidx.appcompat.app.AppCompatActivity  
import android.os.Bundle  
import android.view.View  
import androidx.appcompat.widget.Toolbar  
import org.kvpbldsck.rss.R  
  
class DeveloperInfoActivity : AppCompatActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_developer\_info*)  
  
 getToolbar().*title* = "Developer"  
 }  
  
 fun close(view: View) {  
 finish()  
 }  
  
 private fun getToolbar() : Toolbar = findViewById(R.id.*developer\_info\_toolbar*)  
}

FeedAdapter.kt

package org.kvpbldsck.rss.adapters  
  
import android.content.Context  
import android.content.Intent  
import android.view.LayoutInflater  
import android.view.ViewGroup  
import androidx.recyclerview.widget.RecyclerView  
import com.bumptech.glide.Glide  
import org.kvpbldsck.rss.R  
import org.kvpbldsck.rss.activities.NewsItemActivity  
import org.kvpbldsck.rss.interfaces.ItemClickListener  
import org.kvpbldsck.rss.models.NewsItem  
import org.kvpbldsck.rss.models.RssObject  
  
class FeedAdapter(  
 private val mContext: Context  
) : RecyclerView.Adapter<FeedViewHolder>() {  
  
 private val news = *mutableListOf*<NewsItem>()  
 private val inflater = LayoutInflater.from(mContext)  
  
 override fun onBindViewHolder(holder: FeedViewHolder, position: Int) {  
 holder.newsRowTitle.*text* = news[position].title  
 holder.newsRowPublishDate.*text* = news[position].pubDate  
 holder.newsRowAuthor.*text* = "by ${news[position].author}"  
  
 Glide.with(mContext)  
 .load(news[position].thumbnail)  
 .into(holder.newsRowImage)  
  
 holder.setItemClickListener **{** \_, pos, isLongClick **->** if (!isLongClick) {  
 val intent = Intent(mContext, NewsItemActivity::class.*java*)  
 intent.*flags* = Intent.*FLAG\_ACTIVITY\_NEW\_TASK* intent.putExtra("title", news[pos].title)  
 intent.putExtra("date", news[pos].pubDate)  
 intent.putExtra("content", news[pos].content)  
 intent.putExtra("author", news[pos].author)  
 mContext.startActivity(intent)  
 }  
 **}** }  
  
 fun addNews(news : List<NewsItem>) {  
 val oldCount = this.news.*count*()  
 this.news += news  
 notifyItemRangeInserted(oldCount, news.*count*())  
 }  
  
 override fun getItemCount(): Int = news.size  
  
 override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): FeedViewHolder {  
 val itemView = inflater.inflate(R.layout.*row*, parent, false)  
 return FeedViewHolder(itemView)  
 }  
  
 fun clearNews() {  
 val oldCount = this.news.*count*()  
 news.clear()  
 notifyItemRangeRemoved(0, oldCount)  
 }  
}

FeedViewHolder.kt

package org.kvpbldsck.rss.adapters  
  
import android.view.View  
import android.widget.ImageView  
import android.widget.TextView  
import androidx.recyclerview.widget.RecyclerView  
import org.kvpbldsck.rss.interfaces.ItemClickListener  
import org.kvpbldsck.rss.R  
  
class FeedViewHolder(itemView: View) :  
 RecyclerView.ViewHolder(itemView),  
 View.OnClickListener,  
 View.OnLongClickListener {  
  
 var newsRowTitle = itemView.findViewById(R.id.*row\_title*) as TextView  
 var newsRowPublishDate = itemView.findViewById(R.id.*row\_publish\_date*) as TextView  
 var newsRowImage = itemView.findViewById(R.id.*row\_image*) as ImageView  
 var newsRowAuthor = itemView.findViewById(R.id.*row\_author*) as TextView  
 private var itemClickListener: ItemClickListener? = null  
  
 init {  
 itemView.setOnClickListener(this)  
 itemView.setOnLongClickListener(this)  
 }  
  
 fun setItemClickListener(itemClickListener: ItemClickListener) {  
 this.itemClickListener = itemClickListener  
 }  
  
 override fun onClick(v: View?) {  
 itemClickListener!!.onClick(v, *adapterPosition*, false)  
 }  
  
 override fun onLongClick(v: View?): Boolean {  
 itemClickListener!!.onClick(v, *adapterPosition*, true)  
 return true  
 }  
}

RetrofitServiceGenerator.kt

package org.kvpbldsck.rss.common  
  
import com.bumptech.glide.BuildConfig  
import org.kvpbldsck.rss.interfaces.FeedService  
import okhttp3.OkHttpClient  
import okhttp3.logging.HttpLoggingInterceptor  
import retrofit2.Retrofit  
import retrofit2.converter.gson.GsonConverterFactory  
  
class RetrofitServiceGenerator {  
 companion object {  
 fun createService(): FeedService {  
 val apiUrl = "https://api.rss2json.com/v1/"  
 val retrofit = Retrofit.Builder()  
 .baseUrl(apiUrl)  
 .client(builderHttpClient())  
 .addConverterFactory(GsonConverterFactory.create())  
 .build()  
 return retrofit.create(FeedService::class.*java*)  
 }  
  
 private fun builderHttpClient(): OkHttpClient {  
 val client = OkHttpClient.Builder()  
 if (BuildConfig.*DEBUG*) {  
 val logging = HttpLoggingInterceptor()  
 logging.*level* = HttpLoggingInterceptor.Level.*BODY* client.addInterceptor(logging)  
 }  
 return client.build()  
 }  
 }  
}

FeedService.kt

package org.kvpbldsck.rss.interfaces  
  
import org.kvpbldsck.rss.models.RssObject  
import retrofit2.Call  
import retrofit2.http.GET  
import retrofit2.http.Query  
  
interface FeedService {  
 @GET("api.json")  
 fun getFeed(@Query("rss\_url") url: String): Call<RssObject>  
}

ItemClickListener.kt

package org.kvpbldsck.rss.interfaces;  
  
import android.view.View;  
  
public interface ItemClickListener {  
 void onClick(View view, int position, boolean isLongClick);  
}

Feed.kt

package org.kvpbldsck.rss.models  
  
data class Feed(  
 val url: String,  
 val title: String,  
 val link: String,  
 val author: String,  
 val description: String,  
 val image: String  
)

NewsItem.kt

package org.kvpbldsck.rss.models  
  
data class NewsItem(  
 val title: String,  
 val pubDate: String,  
 val link: String,  
 val guid: String,  
 val author: String,  
 val thumbnail: String,  
 val description: String,  
 val content: String,  
 val enclosure: Any,  
 val categories: List<String>  
)

RssObject.kt

package org.kvpbldsck.rss.models  
  
data class RssObject(  
 val status: String,  
 val feed: Feed,  
 val items: List<NewsItem>  
)

activity\_add\_rss\_source.xml

<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".activities.AddRssSourceActivity">  
  
 <androidx.appcompat.widget.Toolbar  
 android:id="@+id/add\_rss\_source\_toolbar"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:background="?attr/colorPrimary"  
 android:minHeight="?attr/actionBarSize"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:titleTextColor="@android:color/white" />  
  
 <com.google.android.material.textfield.TextInputEditText  
 android:id="@+id/add\_rss\_source\_input"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="8dp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/add\_rss\_source\_toolbar" />  
  
 <androidx.constraintlayout.widget.ConstraintLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="8dp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/add\_rss\_source\_input">  
  
 <Button  
 android:id="@+id/addButton"  
 style="@style/Base.Widget.AppCompat.Button.Colored"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="end"  
 android:onClick="addRssSource"  
 android:text="@string/add"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <Button  
 android:id="@+id/closeButton"  
 style="@style/Base.Widget.AppCompat.Button.Colored"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="end"  
 android:onClick="close"  
 android:text="@string/cancel"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toStartOf="@+id/addButton"  
 app:layout\_constraintHorizontal\_bias="0.0"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.0" />  
  
 </androidx.constraintlayout.widget.ConstraintLayout>  
  
</androidx.constraintlayout.widget.ConstraintLayout>

activity\_main.xml

<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".activities.MainActivity">  
  
 <androidx.appcompat.widget.Toolbar  
 android:id="@+id/main\_toolbar"  
 android:minHeight="?attr/actionBarSize"  
 android:background="?attr/colorPrimary"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 app:titleTextColor="@android:color/white"/>  
  
 <androidx.recyclerview.widget.RecyclerView  
 android:id="@+id/main\_recycler\_view"  
 android:layout\_below="@+id/main\_toolbar"  
 android:padding="8dp"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"/>  
  
</RelativeLayout>

activity\_news\_item.xml

<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".activities.NewsItemActivity">  
  
 <androidx.appcompat.widget.Toolbar  
 android:id="@+id/news\_item\_toolbar"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:background="?attr/colorPrimary"  
 android:minHeight="?attr/actionBarSize"  
 app:titleTextColor="@android:color/white" />  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:layout\_below="@+id/news\_item\_toolbar"  
 android:layout\_margin="8dp"  
 android:orientation="vertical">  
  
 <TextView  
 android:id="@+id/news\_item\_title"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="@string/default\_title"  
 android:textColor="@color/black"  
 android:textSize="25sp"  
 android:textStyle="bold" />  
  
 <TextView  
 android:id="@+id/news\_item\_publish\_date"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="@string/default\_publish\_date"  
 android:textSize="14sp"  
 android:textStyle="italic" />  
  
 <org.sufficientlysecure.htmltextview.HtmlTextView  
 android:id="@+id/news\_item\_content"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:layout\_gravity="bottom"  
 android:text="@string/default\_content"  
 android:textAppearance="@android:style/TextAppearance.Medium"  
 android:textSize="21sp" />  
 </LinearLayout>  
  
  
</RelativeLayout>

activity\_developer\_info.xml

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".activities.DeveloperInfoActivity"  
 android:orientation="vertical">  
  
 <androidx.appcompat.widget.Toolbar  
 android:id="@+id/developer\_info\_toolbar"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:background="?attr/colorPrimary"  
 android:minHeight="?attr/actionBarSize"  
 app:titleTextColor="@android:color/white"/>  
  
 <TextView  
 android:id="@+id/developer\_info\_title"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="8dp"  
 android:text="@string/developer\_info\_title"  
 android:textColor="@color/black"  
 android:textSize="28sp"/>  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginHorizontal="8dp"  
 android:text="@string/developer\_info\_university"  
 android:textSize="16sp"/>  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:autoLink="web"  
 android:linksClickable="true"  
 android:layout\_marginHorizontal="8dp"  
 android:text="@string/developer\_info\_links"  
 android:textSize="20sp"/>  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/close"  
 android:layout\_margin="8dp"  
 android:layout\_gravity="end"  
 android:onClick="close"/>  
  
</LinearLayout>

row.xml

<?xml version="1.0" encoding="utf-8"?>  
<androidx.cardview.widget.CardView  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:cardView="http://schemas.android.com/apk/res-auto"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 android:padding="8dp"  
 android:layout\_marginBottom="8dp"  
 app:cardCornerRadius="8dp"  
 app:cardElevation="8dp">  
  
 <LinearLayout  
 android:layout\_margin="8dp"  
 android:orientation="vertical"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content">  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal"  
 android:weightSum="100">  
 <LinearLayout  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"  
 android:layout\_weight="99">  
 <TextView  
 android:id="@+id/row\_title"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="@string/default\_title"  
 android:textColor="@color/black"  
 android:textSize="21sp"  
 android:textStyle="bold"/>  
 <androidx.constraintlayout.widget.ConstraintLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content">  
  
 <TextView  
 android:id="@+id/row\_publish\_date"  
 android:textSize="12sp"  
 android:textStyle="italic"  
 android:text="@string/default\_publish\_date"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 cardView:layout\_constraintStart\_toStartOf="parent"  
 cardView:layout\_constraintTop\_toTopOf="parent"  
 cardView:layout\_constraintBottom\_toBottomOf="parent"/>  
  
 <TextView  
 android:id="@+id/row\_author"  
 android:textSize="12sp"  
 android:textStyle="italic"  
 android:text="@string/default\_author"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 cardView:layout\_constraintStart\_toStartOf="@+id/row\_publish\_date"  
 cardView:layout\_constraintTop\_toTopOf="parent"  
 cardView:layout\_constraintBottom\_toBottomOf="parent"  
 cardView:layout\_constraintEnd\_toEndOf="parent"  
 cardView:layout\_constraintHorizontal\_bias="1"/>  
  
 </androidx.constraintlayout.widget.ConstraintLayout>  
 </LinearLayout>  
 <LinearLayout  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:gravity="end">  
 <ImageView  
 android:id="@+id/row\_image"  
 android:layout\_width="100dp"  
 android:layout\_height="100dp"/>  
 </LinearLayout>  
 </LinearLayout>  
 </LinearLayout>  
  
  
</androidx.cardview.widget.CardView>

main\_menu.xml

<?xml version="1.0" encoding="utf-8"?>  
<menu xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto">  
  
 <item  
 android:id="@+id/menu\_developer\_info"  
 android:icon="@drawable/ic\_outline\_info\_24"  
 android:title="@string/developer"  
 app:showAsAction="ifRoom"/>  
  
 <item  
 android:id="@+id/menu\_refresh\_button"  
 android:icon="@drawable/ic\_baseline\_refresh\_24"  
 android:title="@string/refresh"  
 app:showAsAction="ifRoom"/>  
  
 <item  
 android:id="@+id/menu\_add\_button"  
 android:icon="@drawable/ic\_baseline\_add\_24"  
 android:title="@string/add"  
 app:showAsAction="ifRoom"/>  
</menu>

news\_item\_menu.xml

<?xml version="1.0" encoding="utf-8"?>  
<menu xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto">  
  
 <item  
 android:id="@+id/menu\_share\_button"  
 android:icon="@drawable/ic\_baseline\_share\_24"  
 android:title="@string/refresh"  
 app:showAsAction="ifRoom"/>  
  
</menu>

strings.xml

<resources>  
 <string name="app\_name">Rss</string>  
 <string name="refresh">Refresh</string>  
 <string name="add">Add</string>  
 <string name="default\_title">Title</string>  
 <string name="default\_publish\_date">2017-05-13 12:30:00</string>  
 <string name="default\_content">Content</string>  
 <string name="cancel">Cancel</string>  
 <string name="default\_author">by Author</string>  
 <string name="developer\_info\_title">Zakharenko Valerij</string>  
 <string name="developer\_info\_university">Group 791051</string>  
 <string name="developer\_info\_links">https://github.com/ikonvpalto</string>  
 <string name="close">Close</string>  
 <string name="developer">Developer</string>  
</resources>

themes.xml

<resources xmlns:tools="http://schemas.android.com/tools">  
 <!-- Base application theme. -->  
 <style name="Theme.Rss" parent="Theme.MaterialComponents.DayNight.NoActionBar">  
 <!-- Primary brand color. -->  
 <item name="colorPrimary">@color/purple\_500</item>  
 <item name="colorPrimaryVariant">@color/purple\_700</item>  
 <item name="colorOnPrimary">@color/white</item>  
 <!-- Secondary brand color. -->  
 <item name="colorSecondary">@color/teal\_200</item>  
 <item name="colorSecondaryVariant">@color/teal\_700</item>  
 <item name="colorOnSecondary">@color/black</item>  
 <!-- Status bar color. -->  
 <item name="android:statusBarColor" tools:targetApi="l">?attr/colorPrimaryVariant</item>  
 <!-- Customize your theme here. -->  
 </style>  
  
</resources>