БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ   
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ   
Кафедра программного обеспечения информационных технологий

Факультет КСиС   
Специальность ПОИТ

Индивидуальная практическая работа по модулю 1   
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 отображения руководства пользователя (подсказок)

Скриншоты работы приложения

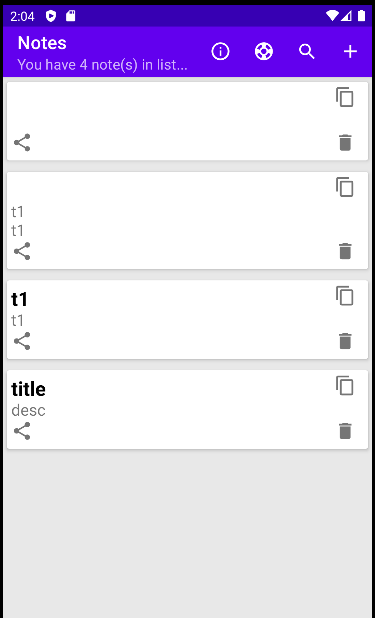


Рис. 1 Стартовый экран со списком заметок

Изображение выглядит как текст

Автоматически созданное описание

Рис. 2 Экран добавления/изминения заметки

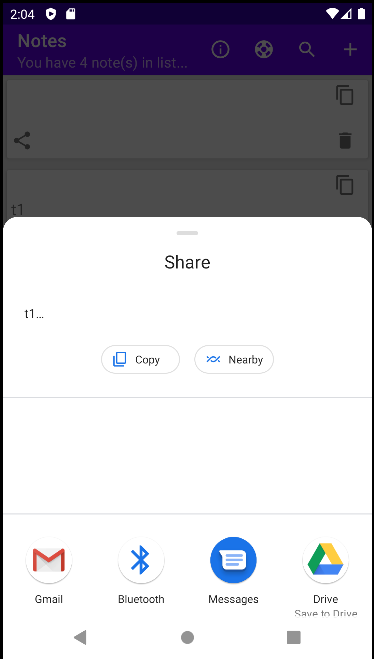


Рис. 3 Меню "Поделиться" заметки

Изображение выглядит как текст

Автоматически созданное описание

Рис. 4 Поиск заметок по названию или тексту

Изображение выглядит как текст

Автоматически созданное описание

Рис. 5 Руководство пользователя на главном экране

Изображение выглядит как текст

Автоматически созданное описание

Рис. 6 Руководство пользователя на экране добавления заметки

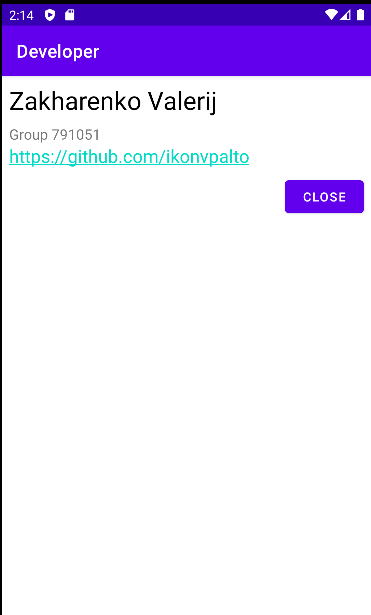


Рис. 7 Страница с информацией о разработчике

Листинг исходного приложения

AddNoteActivity.kt

package org.kvpbldsck.notes.activities  
  
import android.app.SearchManager  
import android.content.ContentValues  
import android.content.Context  
import android.content.Intent  
import androidx.appcompat.app.AppCompatActivity  
import android.os.Bundle  
import android.view.Menu  
import android.view.MenuItem  
import android.view.View  
import android.widget.EditText  
import android.widget.Toast  
import androidx.appcompat.widget.SearchView  
import org.kvpbldsck.notes.R  
import org.kvpbldsck.notes.db.DbManager  
  
class AddNoteActivity : AppCompatActivity() {  
  
 private var id = 0  
  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_add\_note*)  
  
 try {  
 val bundle = *intent*.*extras*!!  
 id = bundle.getInt("ID", 0)  
 if (id != 0) {  
 getTitleEditor().setText(bundle.getString("name"))  
 getDescriptionEditor().setText(bundle.getString("des"))  
 }  
 } catch (ex: Exception) {  
 }  
 }  
  
 override fun onCreateOptionsMenu(menu: Menu?): Boolean {  
 *menuInflater*.inflate(R.menu.*add\_note\_menu*, menu)  
  
 return super.onCreateOptionsMenu(menu)  
 }  
  
 fun addFunc(view: View) {  
 val dbManager = DbManager(this)  
  
 val values = ContentValues()  
 values.put("Title", getTitleEditor().*text*.toString())  
 values.put("Description", getDescriptionEditor().*text*.toString())  
  
 val isNoteAddedSuccessfully =  
 if (id == 0) {  
 dbManager.insert(values) > 0  
 } else {  
 val selectionArgs = *arrayOf*(id.toString())  
 dbManager.update(values, "ID=?", selectionArgs) > 0  
 }  
  
 if (isNoteAddedSuccessfully) {  
 Toast.makeText(this, "Note is added", Toast.*LENGTH\_SHORT*).show()  
 finish()  
 } else {  
 Toast.makeText(  
 this, "Error adding note...",  
 Toast.*LENGTH\_SHORT* ).show()  
 }  
 }  
  
 override fun onOptionsItemSelected(item: MenuItem): Boolean {  
 when (item.*itemId*) {  
 R.id.*showHelp* -> {  
 val intent = Intent(this, HelpActivity::class.*java*)  
 intent.putExtra("helpMessageTitle", getString(R.string.*help\_add\_note\_title*))  
 intent.putExtra("helpMessage", getString(R.string.*help\_add\_note*))  
 startActivity(intent)  
 }  
 }  
 return super.onOptionsItemSelected(item)  
 }  
  
 fun closeNote(view: View) {  
 finish()  
 }  
  
 private fun getTitleEditor() = findViewById<EditText>(R.id.*titleEditor*)  
 private fun getDescriptionEditor() = findViewById<EditText>(R.id.*descriptionEditor*)  
}

HelpActivity.kt

package org.kvpbldsck.notes.activities  
  
import androidx.appcompat.app.AppCompatActivity  
import android.os.Bundle  
import android.view.View  
import android.widget.TextView  
import org.kvpbldsck.notes.R  
  
class HelpActivity : AppCompatActivity() {  
 private var helpMessage = ""  
 private var helpMessageTitle = ""  
  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_help*)  
  
 try {  
 val bundle = *intent*.*extras*!!  
 helpMessage = bundle.getString("helpMessage", "")  
 helpMessage = helpMessage.*replace*("\\\n", System.getProperty("line.separator")!!)  
 helpMessageTitle = bundle.getString("helpMessageTitle", "")  
 getHelpMessageView().*text* = helpMessage  
 getHelpMessageHeaderView().*text* = helpMessageTitle  
 } catch (ex: Exception) {  
 }  
 }  
  
 private fun getHelpMessageHeaderView() = findViewById<TextView>(R.id.*helpMessageHeaderTextView*)  
 private fun getHelpMessageView() = findViewById<TextView>(R.id.*helpMessageTextView*)  
  
 fun closeHelp(view: View) {  
 finish()  
 }  
}

MainActivity.kt

package org.kvpbldsck.notes.activities  
  
import android.app.SearchManager  
import android.content.ClipData  
import android.content.ClipboardManager  
import android.content.Context  
import android.content.Intent  
import androidx.appcompat.app.AppCompatActivity  
import android.os.Bundle  
import android.view.Menu  
import android.view.MenuItem  
import android.view.View  
import android.view.ViewGroup  
import android.widget.\*  
import androidx.appcompat.widget.SearchView  
import org.kvpbldsck.notes.R  
import org.kvpbldsck.notes.db.DbManager  
import org.kvpbldsck.notes.models.Note  
  
class MainActivity : AppCompatActivity() {  
  
 private var listNotes = ArrayList<Note>()  
  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_main*)  
  
 loadQuery("%")  
 }  
  
 override fun onResume() {  
 super.onResume()  
 loadQuery("%")  
 }  
  
 private fun loadQuery(query: String) {  
 val dbManager = DbManager(this)  
 val projections = *arrayOf*("ID", "Title", "Description")  
 val selectionArgs = *arrayOf*(query, query)  
 val cursor = dbManager.query(projections, "Title like ? or Description like ?", selectionArgs, "Title")  
 listNotes.clear()  
 if (cursor.moveToFirst()) {  
 do {  
 val idColumnIndex = cursor.getColumnIndex("ID")  
 val id = cursor.getInt(idColumnIndex)  
  
 val titleColumnIndex = cursor.getColumnIndex("Title")  
 val titleFromDb = cursor.getString(titleColumnIndex)  
  
 val descriptionColumnIndex = cursor.getColumnIndex("Description")  
 val description = cursor.getString(descriptionColumnIndex)  
  
 listNotes.add(Note(id, titleFromDb, description))  
  
 } while (cursor.moveToNext())  
 }  
  
 //adapter  
 val myNotesAdapter = MyNotesAdapter(this, listNotes)  
 //set adapter  
 getNotesListView().*adapter* = myNotesAdapter  
  
 //get total number of tasks from ListView  
 val total = getNotesListView().*count* //actionbar  
 val mActionBar = *supportActionBar* if (mActionBar != null) {  
 //set to actionbar as subtitle of actionbar  
 mActionBar.*subtitle* = "You have $total note(s) in list..."  
 }  
 }  
  
 override fun onCreateOptionsMenu(menu: Menu?): Boolean {  
 *menuInflater*.inflate(R.menu.*main\_menu*, menu)  
  
 //searchView  
 val sv: SearchView = menu!!.findItem(R.id.*app\_bar\_search*).*actionView* as  
 SearchView  
  
 val sm = getSystemService(Context.*SEARCH\_SERVICE*) as SearchManager  
 sv.setSearchableInfo(sm.getSearchableInfo(*componentName*))  
 sv.setOnQueryTextListener(object : SearchView.OnQueryTextListener {  
 override fun onQueryTextSubmit(query: String?): Boolean {  
 loadQuery("%$query%")  
 return false  
 }  
  
 override fun onQueryTextChange(newText: String?): Boolean {  
 loadQuery("%$newText%")  
 return false  
 }  
 })  
  
 return super.onCreateOptionsMenu(menu)  
 }  
  
 override fun onOptionsItemSelected(item: MenuItem): Boolean {  
 when (item.*itemId*) {  
 R.id.*menu\_developer\_info* -> {  
 startActivity(Intent(this, DeveloperInfoActivity::class.*java*))  
 }  
 R.id.*addNote* -> {  
 startActivity(Intent(this, AddNoteActivity::class.*java*))  
 }  
 R.id.*showHelp* -> {  
 val intent = Intent(this, HelpActivity::class.*java*)  
 intent.putExtra("helpMessageTitle", getString(R.string.*help\_main\_title*))  
 intent.putExtra("helpMessage", getString(R.string.*help\_main*))  
 startActivity(intent)  
 }  
 }  
 return super.onOptionsItemSelected(item)  
 }  
  
 inner class MyNotesAdapter(context: Context, private var listNotesAdapter: ArrayList<Note>) : BaseAdapter() {  
  
 private var context: Context? = context  
  
 override fun getView(position: Int, convertView: View?, parent: ViewGroup?): View {  
 //inflate layout row.xml  
 val myView = *layoutInflater*.inflate(R.layout.*row*, null)  
 val myNote = listNotesAdapter[position]  
 myView.*getTitleTextView*().*text* = myNote.nodeName  
 myView.*getDescriptionTextView*().*text* = myNote.nodeDes  
  
 myView.*getDeleteButton*().setOnClickListener **{** val dbManager = DbManager(this.context!!)  
 val selectionArgs = *arrayOf*(myNote.nodeID.toString())  
 dbManager.delete("ID=?", selectionArgs)  
 loadQuery("%")  
 **}** myView.setOnClickListener **{** goToUpdateFun(myNote) **}** myView.*getCopyButton*().setOnClickListener **{** val title = myView.*getTitleTextView*().*text*.toString()  
 val desc = myView.*getDescriptionTextView*().*text*.toString()  
  
 val s = title + "\n" + desc  
 copyTextToClipboard(s)  
 Toast.makeText(this@MainActivity, "Copied...", Toast.*LENGTH\_SHORT*).show()  
 **}** myView.*getShareButton*().setOnClickListener **{** //get title  
 val title = myView.*getTitleTextView*().*text*.toString()  
 //get description  
 val desc = myView.*getDescriptionTextView*().*text*.toString()  
 //concatenate  
 val s = title + "\n" + desc  
 //share intent  
 val shareIntent = Intent()  
 shareIntent.*action* = Intent.*ACTION\_SEND* shareIntent.*type* = "text/plain"  
 shareIntent.putExtra(Intent.*EXTRA\_TEXT*, s)  
 startActivity(Intent.createChooser(shareIntent, s))  
 **}** return myView  
 }  
  
 override fun getItem(position: Int): Any {  
 return listNotesAdapter[position]  
 }  
  
 override fun getItemId(position: Int): Long {  
 return position.toLong()  
 }  
  
 override fun getCount(): Int {  
 return listNotesAdapter.size  
 }  
  
 private fun copyTextToClipboard(text: String) {  
 val clipboardManager = getSystemService(Context.*CLIPBOARD\_SERVICE*) as ClipboardManager  
 val clipData = ClipData.newPlainText("text", text)  
 clipboardManager.setPrimaryClip(clipData)  
 }  
  
 }  
  
 private fun goToUpdateFun(myNote: Note) {  
 val intent = Intent(this, AddNoteActivity::class.*java*)  
 intent.putExtra("ID", myNote.nodeID) //put id  
 intent.putExtra("name", myNote.nodeName) //ut name  
 intent.putExtra("des", myNote.nodeDes) //put description  
 startActivity(intent) //start activity  
 }  
  
 private fun getNotesListView() = findViewById<ListView>(R.id.*notesListView*)  
 private fun View.getTitleTextView() = this.findViewById<TextView>(R.id.*titleTextView*)  
 private fun View.getDescriptionTextView() = this.findViewById<TextView>(R.id.*descriptionTextView*)  
 private fun View.getDeleteButton() = this.findViewById<ImageButton>(R.id.*deleteButton*)  
 private fun View.getCopyButton() = this.findViewById<ImageButton>(R.id.*copyButton*)  
 private fun View.getShareButton() = this.findViewById<ImageButton>(R.id.*shareButton*)  
}

DbManager.kt

package org.kvpbldsck.notes.db  
  
import android.content.ContentValues  
import android.content.Context  
import android.database.Cursor  
import android.database.sqlite.SQLiteDatabase  
import android.database.sqlite.SQLiteOpenHelper  
import android.database.sqlite.SQLiteQueryBuilder  
import android.widget.Toast  
  
class DbManager(context: Context) {  
  
 var dbName = "Notes"  
 var dbTable = "Notes"  
  
 private val columnId = "ID"  
 private val columnTitle = "Title"  
 private val columnDescription = "Description"  
  
 var dbVersion = 1  
  
 //CREATE TABLE IF NOT EXISTS MyNotes (ID INTEGER PRIMARY KEY,title TEXT, Description TEXT);"  
 val sqlCreateTable =  
 "CREATE TABLE IF NOT EXISTS $dbTable($columnId INTEGER PRIMARY KEY, $columnTitle TEXT, $columnDescription TEXT);"  
  
 private var sqlDb: SQLiteDatabase? = null  
  
 init {  
 sqlDb = DatabaseHelperNotes(context).*writableDatabase* }  
  
 inner class DatabaseHelperNotes(context: Context) :  
 SQLiteOpenHelper(context, dbName, null, dbVersion) {  
  
 private var context: Context? = context  
  
 override fun onCreate(db: SQLiteDatabase?) {  
 db!!.execSQL(sqlCreateTable)  
 Toast  
 .makeText(this.context, "database created...", Toast.*LENGTH\_SHORT*)  
 .show()  
 }  
  
 override fun onUpgrade(db: SQLiteDatabase?, oldVersion: Int, newVersion: Int) {  
 db!!.execSQL("DROP TABLE IF EXISTS $dbTable")  
 }  
 }  
  
 fun insert(values: ContentValues): Long {  
 return sqlDb!!.insert(dbTable, "", values)  
 }  
  
 fun query(  
 projection: Array<String>,  
 selection: String,  
 selectionArgs: Array<String>,  
 sorOrder: String  
 ): Cursor {  
 val qb = SQLiteQueryBuilder()  
 qb.*tables* = dbTable  
 return qb.query(sqlDb, projection, selection, selectionArgs, null, null, sorOrder)  
 }  
  
 fun delete(selection: String, selectionArgs: Array<String>): Int {  
 return sqlDb!!.delete(dbTable, selection, selectionArgs)  
 }  
  
 fun update(values: ContentValues, selection: String, selectionArgs: Array<String>): Int {  
 return sqlDb!!.update(dbTable, values, selection, selectionArgs)  
 }  
}

Note.kt

package org.kvpbldsck.notes.models  
  
data class Note(var nodeID: Int, var nodeName: String, var nodeDes: String)

activity\_add\_note.xml

<?xml version="1.0" encoding="utf-8"?>  
<ScrollView 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.AddNoteActivity">  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="vertical">  
  
 <androidx.cardview.widget.CardView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="5dp"  
 app:cardBackgroundColor="@color/white"  
 app:cardCornerRadius="3dp"  
 app:cardElevation="3dp">  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="vertical">  
  
 <EditText  
 android:id="@+id/titleEditor"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:background="@null"  
 android:hint="Enter Title"  
 android:padding="10dp"  
 android:singleLine="true"  
 android:textStyle="bold" />  
  
 <EditText  
 android:id="@+id/descriptionEditor"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:background="@null"  
 android:gravity="top"  
 android:hint="Enter description..."  
 android:minHeight="100dp"  
 android:padding="10dp" />  
 </LinearLayout>  
 </androidx.cardview.widget.CardView>  
  
 <androidx.constraintlayout.widget.ConstraintLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="5dp">  
  
 <Button  
 android:id="@+id/addBtn"  
 style="@style/Base.Widget.AppCompat.Button.Colored"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="end"  
 android:onClick="addFunc"  
 android:text="Add"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintEnd\_toEndOf="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="closeNote"  
 android:text="Close"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toStartOf="@+id/addBtn"  
 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>  
  
  
 </LinearLayout>  
  
</ScrollView>

activity\_help.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.HelpActivity">  
  
 <TextView  
 android:id="@+id/helpMessageHeaderTextView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Help message title"  
 android:textColor="@color/black"  
 android:textSize="24sp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 android:layout\_margin="5dp" />  
  
 <TextView  
 android:id="@+id/helpMessageTextView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:maxLines="10"  
 android:text="Help message"  
 android:textSize="20sp"  
 app:layout\_constraintBottom\_toTopOf="@+id/closeButton"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.0"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/helpMessageHeaderTextView"  
 app:layout\_constraintVertical\_bias="0.00"  
 android:layout\_margin="5dp"/>  
  
 <Button  
 android:id="@+id/closeButton"  
 style="@style/Base.Widget.AppCompat.Button.Colored"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:onClick="closeHelp"  
 android:text="Close"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 android:layout\_marginEnd="5dp"  
 android:layout\_marginBottom="5dp"/>  
  
</androidx.constraintlayout.widget.ConstraintLayout>

activity\_main.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"  
 android:background="@color/gray"  
 android:orientation="vertical"  
 tools:context=".activities.MainActivity">  
  
 <ListView  
 android:id="@+id/notesListView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:divider="@null"  
 android:dividerHeight="1dp" />  
  
</LinearLayout>

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">  
  
 <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:app="http://schemas.android.com/apk/res-auto"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 app:cardBackgroundColor="@color/white"  
 app:cardCornerRadius="3dp"  
 app:cardElevation="3dp"  
 app:cardUseCompatPadding="true"  
 app:layout\_constraintTop\_toTopOf="parent">  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="5dp"  
 android:gravity="end|bottom"  
 android:orientation="vertical">  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="0dp"  
 android:layout\_weight="15"  
 android:orientation="horizontal"  
 android:weightSum="100">  
  
 <LinearLayout  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="99">  
  
 <TextView  
 android:id="@+id/titleTextView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Title"  
 android:textColor="@color/black"  
 android:textSize="22sp"  
 android:textStyle="bold" />  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:gravity="end">  
  
 <ImageButton  
 android:id="@+id/copyButton"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginEnd="5dp"  
 android:layout\_marginRight="5dp"  
 android:background="@null"  
 android:gravity="end"  
 android:src="@drawable/ic\_baseline\_content\_copy\_24" />  
  
 </LinearLayout>  
  
 </LinearLayout>  
  
  
 <TextView  
 android:id="@+id/descriptionTextView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="There may be a very long description of the note"  
 android:textSize="18sp" />  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="0dp"  
 android:layout\_weight="15"  
 android:orientation="horizontal"  
 android:weightSum="100">  
  
 <LinearLayout  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="99">  
  
 <ImageButton  
 android:id="@+id/shareButton"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginEnd="5dp"  
 android:layout\_marginRight="5dp"  
 android:background="@null"  
 android:src="@drawable/ic\_baseline\_share\_24" />  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:gravity="end">  
  
 <ImageButton  
 android:id="@+id/deleteButton"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginEnd="5dp"  
 android:layout\_marginRight="5dp"  
 android:background="@null"  
 android:src="@drawable/ic\_baseline\_delete\_24" />  
  
 </LinearLayout>  
 </LinearLayout>  
 </LinearLayout>  
  
</androidx.cardview.widget.CardView>

add\_note\_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/showHelp"  
 android:icon="@drawable/ic\_baseline\_support\_24"  
 android:title="Help"  
 app:showAsAction="always" />  
</menu>

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="always"/>  
 <item  
 android:id="@+id/showHelp"  
 android:icon="@drawable/ic\_baseline\_support\_24"  
 android:title="@string/help"  
 app:showAsAction="always" />  
 <item  
 android:id="@+id/app\_bar\_search"  
 android:icon="@drawable/ic\_baseline\_search\_24"  
 android:title="@string/search"  
 app:actionViewClass="androidx.appcompat.widget.SearchView"  
 app:showAsAction="always" />  
 <item  
 android:id="@+id/addNote"  
 android:icon="@drawable/ic\_baseline\_add\_24"  
 android:title="@string/add\_note"  
 app:showAsAction="always"/>  
</menu>

strings.xml

<resources>  
 <string name="app\_name">Notes</string>  
 <string name="help\_main\_title">  
 Приложение для заметок  
 </string>  
 <string name="help\_main">  
 Вы можете нажать кнопку со значком + для добавления заметки  
 \nВы можете нажать кнопку со значком лупы для поиска по заголовку или описанию заметки  
 \nЗаметку можно удалить, нажав на ней кнопку с урной, копировать в буфер обмена кнопкой с двумя листками, изменить, нажав на саму заметку, или поделиться ею, нажав на кнопку с соответствующим значком  
 </string>  
 <string name="help\_add\_note\_title">  
 Здесь вы можете добавить заметку  
 </string>  
 <string name="help\_add\_note">  
 Введите ее заголовок в верхнее поле "title", а описание - в нижнее "description", после чего нажмите кнопку "Add" для добавления ее к списку  
 </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>  
 <string name="add\_note">Add Note</string>  
 <string name="search">Search</string>  
 <string name="help">Help</string>  
</resources>