

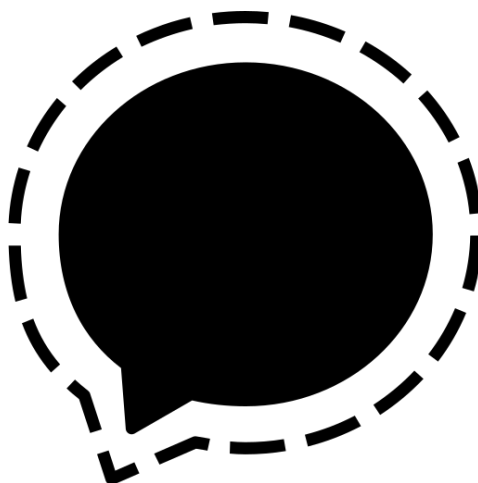
Отчёт по проделанной работе на семинаре по мобильной разработке (25.11.2021).

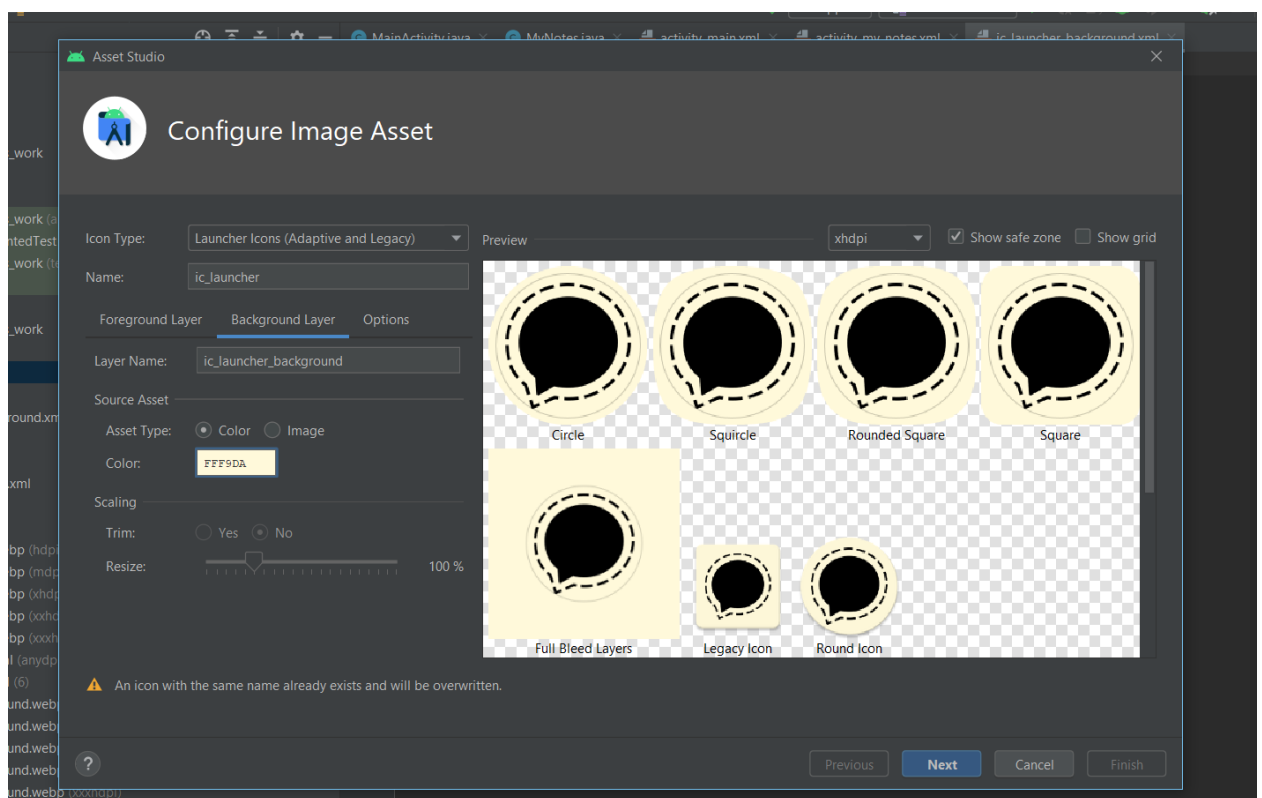
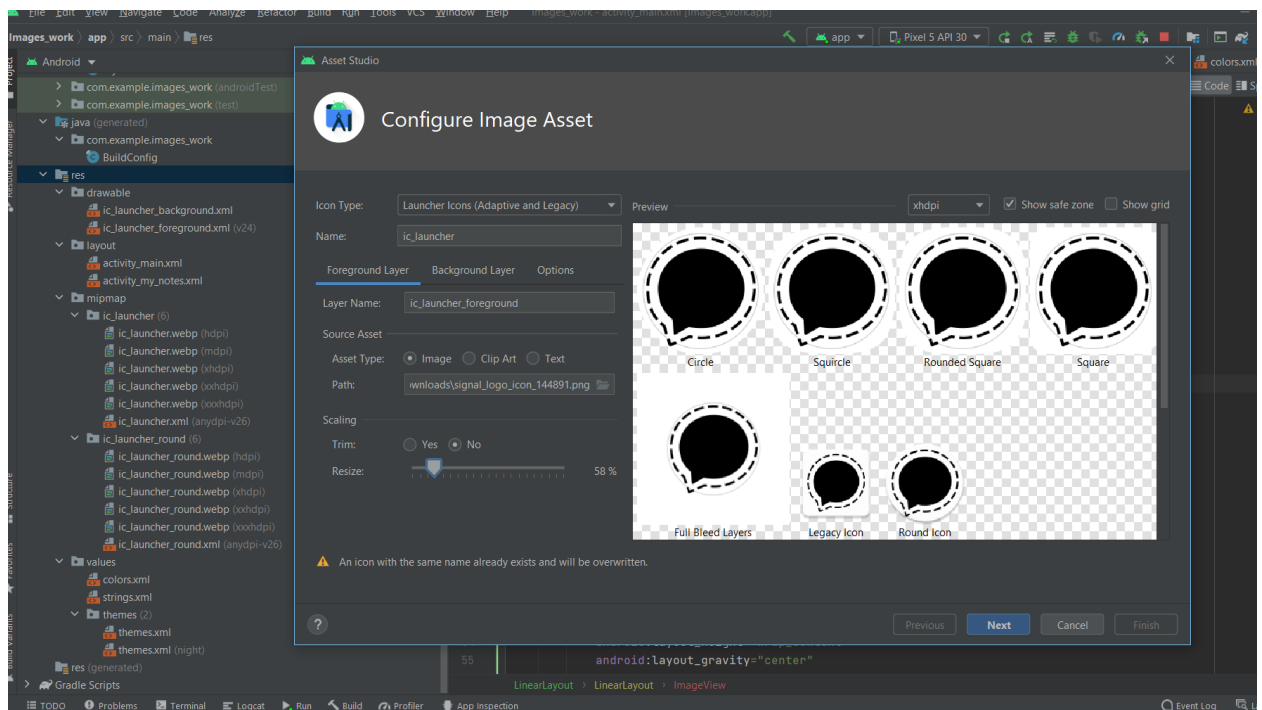
(выполнил Валяев Георгий Анатольевич)

Пожалуй, в наше время сейчас колоссальное внимание активных пользователей привлекает больше дизайн приложения, чем его функционал и возможности. Именно поэтому большая команда разработчиков разрабатывает дизайн мобильного приложения и дополнительно настраивает интересную динамику элементов.

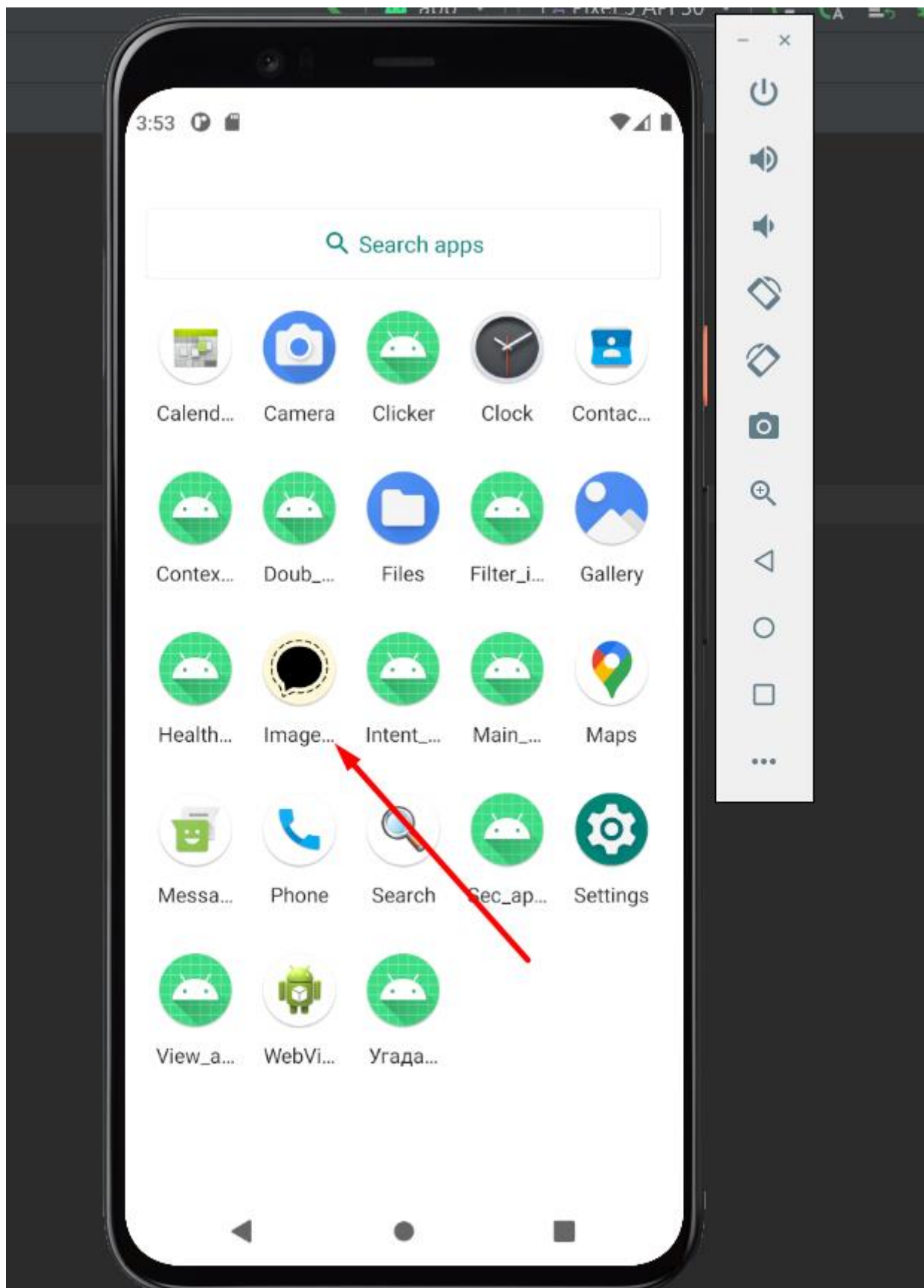
В данной методичке будет разобрана работа по написанию приложения планировщика дел с небольшим дизайном и интересной иконкой как самая базовая и начальная попытка влиться в атмосферу разработчика дизайна и интерфейсов приложения.

Первым отличительным признаком нашего будущего проекта будет служить интересная и уникальная тематическая иконка приложения на смартфоне. Научимся правильно её подключать в проект (нажав при этом правой кнопкой мыши по папке res и выбрав пункт New -> Image Asset):





В результате чего имеем супер красивую иконку приложения в меню приложений смартфона:



Теперь перейдём непосредственно к написанию приложения и начнём с главной активности, на которой разместим 4 кнопки с соответствием Матрице Эйзенхауэра.

В файлах activity_main.xml, colors.xml, strings.xml, и themes.xml составим одновременно и стилистику экрана приложения, и сами элементы экрана (кнопки, текстовые окна):

```
MainActivity.java × MyNotes.java × strings.xml × colors.xml × activity_main.xml × themes.xml × activity_my_notes.xml ×
1 <resources xmlns:tools="http://schemas.android.com/tools">
2     <!-- Base application theme. -->
3     <style name="Theme.Images_work" parent="Theme.MaterialComponents.DayNight.DarkActionBar">
4         <!-- Primary brand color. -->
5         <item name="colorPrimary">@color/gold</item>
6         <item name="colorPrimaryVariant">@color/purple_700</item>
7         <item name="colorOnPrimary">@color/white</item>
8         <!-- Secondary brand color. -->
9         <item name="colorSecondary">@color/teal_200</item>
10        <item name="colorSecondaryVariant">@color/teal_700</item>
11        <item name="colorOnSecondary">@color/black</item>
12        <!-- Status bar color. -->
13        <item name="android:statusBarColor" tools:targetApi="l">?attr/colorPrimaryVariant</item>
14        <!-- Customize your theme here. -->
15    </style>
16</resources>
```

```
MainActivity.java × MyNotes.java × activity_main.xml × strings.xml × colors.xml × activity_my_notes.xml ×
Edit translations for all locales in the translations editor.
1 <resources>
2
3     <string name="app_name">Images_work</string>
4     <string name="welcome_to_do_list">Welcome to the «My planer of topics»</string>
5
6     <string name="important_fast_topics">Срочные и важные дела</string>
7     <string name="not_fast_impopor_topics">Несрочные и важные дела</string>
8     <string name="fast_not_impopor_topics">Срочные и неважные дела</string>
9     <string name="impopor_not_fast_topics">Несрочные и неважные дела</string>
10
11    <string name="my_category">Моя категория</string>
12
13    <string name="input_topic">Введите ваш пункт:</string>
14
15    <string name="date_and_time">Дата и время</string>
16    <string name="adding_topic">Добавить дело</string>
17
18</resources>
```

```
MainActivity.java × MyNotes.java × activity_main.xml × strings.xml × colors.xml × activity_my_notes.xml ×
1 <?xml version="1.0" encoding="utf-8"?>
2 <resources>
3     <color name="purple_200">#FFBB86FC</color>
4     <color name="purple_500">#FF6200EE</color>
5     <color name="purple_700">#FF3700B3</color>
6     <color name="gold">#DAA520</color>
7     <color name="teal_200">#FF03DAC5</color>
8     <color name="teal_700">#FF018786</color>
9     <color name="sea_green">#2E8B57</color>
10    <color name="aqua">#00FFFF</color>
11    <color name="bisquit">#FFE4C4</color>
12    <color name="black">#FF000000</color>
13    <color name="white">#FFFFFFFF</color>
14 </resources>
```

```
MainActivity.java × MyNotes.java × activity_main.xml × activity_my_notes.xml ×
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout
3     xmlns:android="http://schemas.android.com/apk/res/android"
4     xmlns:app="http://schemas.android.com/apk/res-auto"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent"
7     android:background="@color/sea_green"
8     android:orientation="vertical">
9
10    <TextView
11        android:layout_width="wrap_content"
12        android:layout_height="wrap_content"
13        android:text="Welcome to the «My planer of topics»"
14
15        android:layout_gravity="center"
16        android:layout_marginTop="25dp"
17        android:textColor="@color/bisquit"
18        android:textSize="22sp">
19    </TextView>
20
21    <ImageView
22        android:id="@+id/imageView2"
23        android:layout_marginTop="50dp"
24
25        android:layout_width="match_parent"
26        android:layout_height="wrap_content"
27        app:srcCompat="@mipmap/ic_launcher_round">
28    </ImageView>
29
30    <LinearLayout
31        android:layout_width="match_parent"
```

```

25         android:layout_width="match_parent"
26         android:layout_height="wrap_content"
27         app:srcCompat="@mipmap/ic_launcher_round">
28     </ImageView>
29
30     <LinearLayout
31         android:layout_width="match_parent"
32         android:layout_height="wrap_content"
33         android:orientation="vertical">
34
35         <Button
36             android:id="@+id/btnMyNotes_1"
37             android:layout_width="wrap_content"
38
39             android:layout_height="wrap_content"
40             android:layout_gravity="center"
41             android:layout_marginTop="40dp"
42             android:layout_marginBottom="10dp"
43
44             android:onClick="onClick"
45             android:text="Срочные и важные дела">
46     </Button>
47
48     <Button
49         android:id="@+id/btnMyNotes_2"
50         android:layout_width="wrap_content"
51
52         android:layout_height="wrap_content"
53         android:layout_gravity="center"
54         android:layout_marginTop="40dp"
55         android:layout_marginBottom="10dp"

```

```
MainActivity.java × MyNotes.java × activity_main.xml × activity_my_notes.xml ×
43
44     android:onClick="onClick"
45     android:text="Срочные и важные дела">
46 </Button>
47
48 <Button
49     android:id="@+id/btnMyNotes_2"
50     android:layout_width="wrap_content"
51
52     android:layout_height="wrap_content"
53     android:layout_gravity="center"
54     android:layout_marginTop="40dp"
55     android:layout_marginBottom="10dp"
56
57     android:onClick="onClick"
58     android:text="Несрочные и важные дела">
59 </Button>
60
61 <Button
62     android:id="@+id/btnMyNotes_3"
63     android:layout_width="wrap_content"
64
65     android:layout_height="wrap_content"
66     android:layout_gravity="center"
67     android:layout_marginTop="40dp"
68     android:layout_marginBottom="10dp"
69
70     android:onClick="onClick"
71     android:text="Срочные и неважные дела">
72 </Button>
```

```
MainActivity.java x MyNotes.java x activity_main.xml x activity_my_notes.xml x
61 <Button
62     android:id="@+id/btnMyNotes_3"
63     android:layout_width="wrap_content"
64
65     android:layout_height="wrap_content"
66     android:layout_gravity="center"
67     android:layout_marginTop="40dp"
68     android:layout_marginBottom="10dp"
69
70     android:onClick="onClick"
71     android:text="Срочные и неважные дела">
72 </Button>
73
74 <Button
75     android:id="@+id/btnMyNotes_4"
76     android:layout_width="wrap_content"
77
78     android:layout_height="wrap_content"
79     android:layout_gravity="center"
80     android:layout_marginTop="40dp"
81     android:layout_marginBottom="10dp"
82
83     android:onClick="onClick"
84     android:text="Несрочные и неважные дела">
85 </Button>
86
87 </LinearLayout>
88
89 </LinearLayout>
```

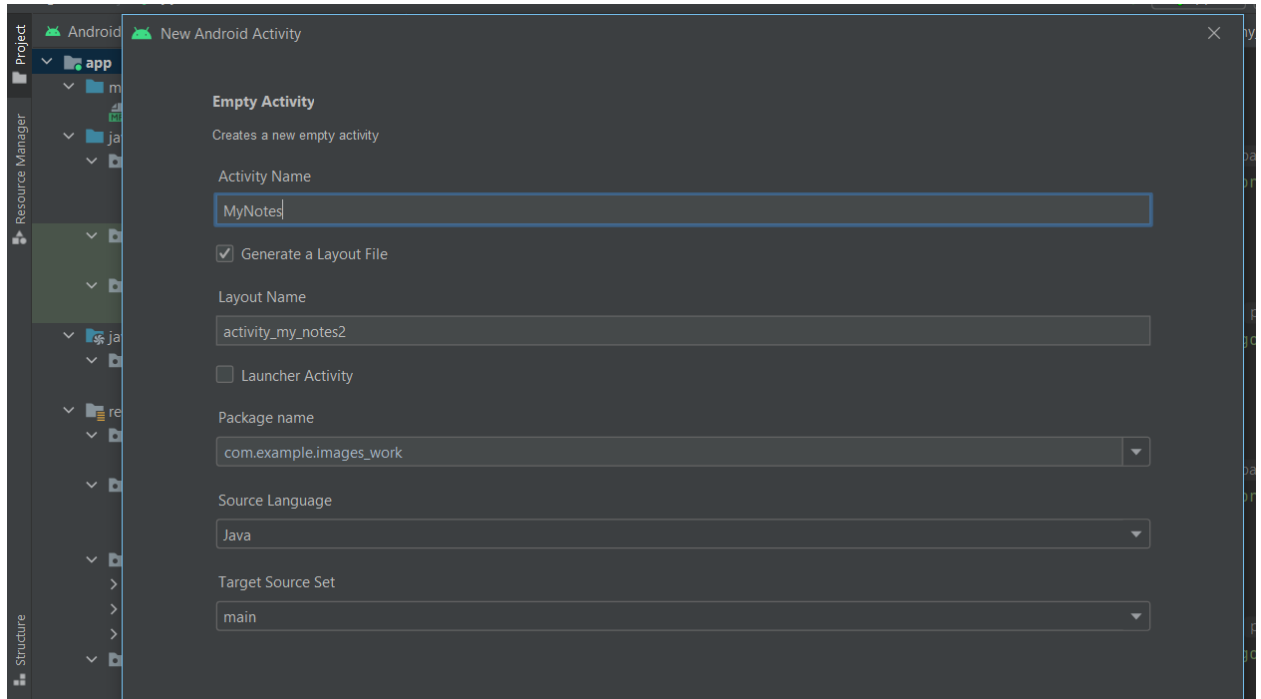
Далее в MainActivity.java пропишем обыкновенный обработчик тапа с интентами, в которые мы будем передавать название выбранной пользователем категории дел через метод .putExtra():


```
MainActivity.java x MyNotes.java x activity_main.xml x activity_my_notes.xml x
1 package com.example.images_work;
2
3 import ...
4
5
6
7
8
9 public class MainActivity extends AppCompatActivity {
10
11     @Override
12     protected void onCreate(Bundle savedInstanceState) {
13         super.onCreate(savedInstanceState);
14         setContentView(R.layout.activity_main);
15     }
16
17     @
18     public void onClick(View v) {
19         switch (v.getId()) {
20             case (R.id.btnMyNotes_1):
21                 Intent first_category = new Intent( packageContext: this, MyNotes.class);
22                 first_category.putExtra( name: "category", value: "Срочные и важные дела");
23                 startActivity(first_category);
24                 break;
25
26             case (R.id.btnMyNotes_2):
27                 Intent second_category = new Intent( packageContext: this, MyNotes.class);
28                 second_category.putExtra( name: "category", value: "Несрочные и важные дела");
29                 startActivity(second_category);
30                 break;
31
32             case (R.id.btnMyNotes_3):
33                 Intent third_category = new Intent( packageContext: this, MyNotes.class);
34                 third_category.putExtra( name: "category", value: "Срочные и неважные дела");
35                 startActivity(third_category);
36                 break;
37
38         }
39     }
40 }
```

```
MainActivity.java x MyNotes.java x activity_main.xml x activity_my_notes.xml x
13 super.onCreate(savedInstanceState);
14 setContentView(R.layout.activity_main);
15 }
16
17 @
18 public void onClick(View v) {
19     switch (v.getId()) {
20         case (R.id.btnMyNotes_1):
21             Intent first_category = new Intent( packageContext: this, MyNotes.class);
22             first_category.putExtra( name: "category", value: "Срочные и важные дела");
23             startActivity(first_category);
24             break;
25
26         case (R.id.btnMyNotes_2):
27             Intent second_category = new Intent( packageContext: this, MyNotes.class);
28             second_category.putExtra( name: "category", value: "Несрочные и важные дела");
29             startActivity(second_category);
30             break;
31
32         case (R.id.btnMyNotes_3):
33             Intent third_category = new Intent( packageContext: this, MyNotes.class);
34             third_category.putExtra( name: "category", value: "Срочные и неважные дела");
35             startActivity(third_category);
36             break;
37
38         case (R.id.btnMyNotes_4):
39             Intent fourth_category = new Intent( packageContext: this, MyNotes.class);
40             fourth_category.putExtra( name: "category", value: "Несрочные и неважные дела");
41             startActivity(fourth_category);
42             break;
43     }
44 }
```

При этом все 4 кнопки будут перенаправлять пользователя на одну и ту же активность, но с разными заголовками по категориям (поэтому нам важно отправить при помощи интента название категории).

Создаём эту активность с подходящим названием – MyNotes:



В `activity_my_notes.xml` пропишем структуру, состоящую из текстового окна с категорией дел, выбранной пользователем, поля для ввода информации о деле и 2 кнопок с выбором даты и добавлением дела с датой в массив, который будет выводиться за созданным элементом `view` (чёрточка) для лучшей визуализации дел в категории:

```
MainActivity.java × MyNotes.java × activity_main.xml × activity_my_notes.xml ×
1      <?xml version="1.0" encoding="utf-8"?>
2      <LinearLayout
3          xmlns:android="http://schemas.android.com/apk/res/android"
4          android:layout_width="match_parent"
5          android:layout_height="match_parent"
6          android:background="@color/sea_green"
7          android:orientation="vertical">
8
9          <TextView
10             android:id="@+id/tvTopic"
11
12             android:layout_width="wrap_content"
13             android:layout_height="wrap_content"
14
15             android:layout_marginTop="40dp"
16             android:layout_marginBottom="40dp"
17
18             android:text="Моя категория"
19             android:textSize="24sp"
20
21             android:textColor="@color/bisquit"
22             android:layout_gravity="center_horizontal">
23      </TextView>
24
25      <LinearLayout
26          android:layout_width="match_parent"
27          android:layout_height="wrap_content"
28          android:orientation="horizontal"
29          android:layout_marginHorizontal="20dp">
30
```

```
MainActivity.java × MyNotes.java × activity_main.xml × activity_my_notes.xml ×
23 </TextView>
24
25 <LinearLayout
26     android:layout_width="match_parent"
27     android:layout_height="wrap_content"
28     android:orientation="horizontal"
29     android:layout_marginHorizontal="20dp">
30
31     <TextView
32         android:layout_width="match_parent"
33         android:layout_height="wrap_content"
34
35         android:textAlignment="center"
36         android:textColor="@color/bisquit"
37
38         android:textSize="18sp"
39         android:text="Введите ваш пункт:"
40
41         android:layout_weight="2">
42     </TextView>
43
44     <EditText
45         android:id="@+id/etTopic"
46
47         android:layout_width="match_parent"
48         android:layout_height="wrap_content"
49         android:layout_weight="2"
50         android:inputType="text">
51     </EditText>
52
53 </LinearLayout>
```

```
MainActivity.java × MyNotes.java × activity_main.xml × activity_my_notes.xml ×
50         android:inputType="text">
51     </EditText>
52
53 </LinearLayout>
54
55 <Button
56     android:id="@+id/btnDateTime"
57     android:layout_width="match_parent"
58     android:layout_height="wrap_content"
59
60     android:layout_marginTop="20dp"
61     android:layout_marginHorizontal="20dp"
62     android:layout_marginBottom="20dp"
63
64     android:layout_gravity="center"
65     android:onClick="onClick"
66     android:text="@string/date_and_time">
67 </Button>
68
69 <Button
70     android:id="@+id/btnAddTopic"
71     android:layout_width="match_parent"
72     android:layout_height="wrap_content"
73     android:layout_marginHorizontal="20dp"
74     android:layout_marginBottom="20dp"
75     android:onClick="onClick"
76     android:text="Добавить дело">
77 </Button>
```

```
MainActivity.java × MyNotes.java × activity_main.xml × activity_my_notes.xml ×
68         android:onClick="onClick"
69         android:text="@string/date_and_time">
70     </Button>
71
72     <Button
73         android:id="@+id/btnAddTopic"
74         android:layout_width="match_parent"
75         android:layout_height="wrap_content"
76         android:layout_marginHorizontal="20dp"
77         android:layout_marginBottom="20dp"
78         android:onClick="onClick"
79         android:text="Добавить дело">
80     </Button>
81
82     <View
83         android:layout_width="match_parent"
84         android:layout_height="1dp"
85         android:background="@color/bisquit">
86     </View>
87
88     <ListView
89         android:id="@+id/lvFirst"
90         android:layout_width="match_parent"
91         android:layout_height="wrap_content">
92     </ListView>
93
94 </LinearLayout>
```

Уже в MyNotes.java мы наследуем для начала методы классов времени и даты для вывода интерактивных окон для пользователя с возможностью определения даты и времени намеченного на будущее дела под категорию.

Также в этом файле мы занимаемся организацией динамического массива дел в категории для его дальнейшего отображения на экране приложения.

Объявляем переменные элементов экрана приложения, целочисленные переменные для работы со временем и датой и динамический строковый массив дел.

В главном методе связываем переменные с ресурсами через класс R, получаем интент из главной активности с содержимым переменной category.

В обработчике нажатия отслеживаем через switch(), какая кнопка была нажата, и (в 1 случае) заносим в переменные данные о реальном календарном времени и вызываем интерактивное окно даты для взаимодействия с пользователем, (во 2 случае) добавляем в элемент ListView полученные данные о времени и деле и отправляем сигнал через setAdapter() о выводе динамического строкового массива на экран приложения:

```
MainActivity.java x MyNotes.java x activity_main.xml x activity_my_notes.xml x
1 package com.example.images_work;
2
3 import ...
22
23 public class MyNotes extends AppCompatActivity implements DatePickerDialog.OnDateSetListener,
24     TimePickerDialog.OnTimeSetListener {
25
26     ListView lvNotes;
27     EditText etCase;
28     TextView category_number;
29
30     int year, month, day, hour, minute;
31     int myYear, myDay, myMonth, myHour, myMinute;
32
33     ArrayList<String> lvList = new ArrayList<>();
34
35     @Override
36     protected void onCreate(Bundle savedInstanceState) {
37         super.onCreate(savedInstanceState);
38         setContentView(R.layout.activity_my_notes);
39
40         lvNotes = findViewById(R.id.lvFirst);
41         etCase = findViewById(R.id.etTopic);
42
43         category_number = findViewById(R.id.tvTopic);
44         Intent intent = getIntent();
45
46         category_number.setText(intent.getStringExtra( name: "category"));
47     }
48
```

```

46 category_number.setText(intent.getStringExtra( name: "category"));
47 }
48
49 @Override
50 public void onClick(View v) {
51     switch (v.getId()) {
52         case (R.id.btnDateTime):
53             Calendar calendar = Calendar.getInstance();
54
55             year = calendar.get(Calendar.YEAR);
56             month = calendar.get(Calendar.MONTH);
57             day = calendar.get(Calendar.DAY_OF_MONTH);
58
59             DatePickerDialog datePickerDialog = new DatePickerDialog( context: MyNotes.this, listener: MyNotes.this, year, month, day);
60             datePickerDialog.show();
61             break;
62
63             case (R.id.btnAddTopic):
64
65                 lvList.add(myHour + ":" + myMinute + " " + myDay + "-" + (myMonth + 1) + "-" + myYear + " " + etCase.getText().toString());
66
67                 ArrayAdapter<String> listviewAdapter = new ArrayAdapter<>( context: this, android.R.layout.simple_list_item_1, lvList);
68
69                 lvNotes.setAdapter(listviewAdapter);
70             }
71     }
72 }
73

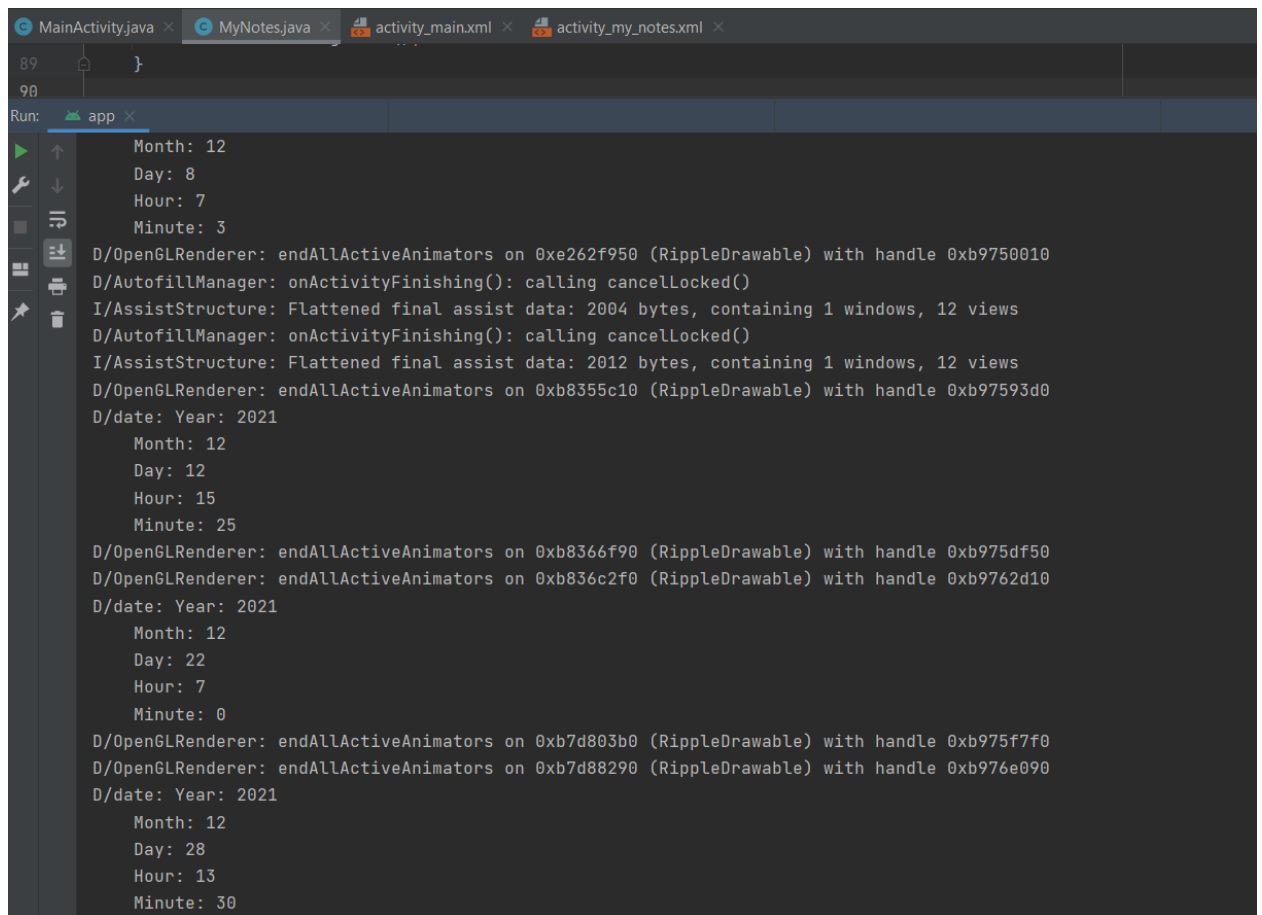
```

В оставшихся 2 методах мы делаем аналогичные действия для выведения окна со временем (PM, AM в английском варианте) и выводом выбранного пользователем времени и дела в консоль через Log.d():

```

73
74
75 @Override
76 public void onDateSet(DatePicker view, int year, int month, int dayOfMonth) {
77     myYear = year;
78     myDay = dayOfMonth;
79     myMonth = month;
80
81     Calendar c = Calendar.getInstance();
82
83     hour = c.get(Calendar.HOUR);
84     minute = c.get(Calendar.MINUTE);
85
86     TimePickerDialog timePickerDialog = new TimePickerDialog( context: MyNotes.this, listener: MyNotes.this, hour, minute, DateFormat.is24HourFormat( context: this));
87     timePickerDialog.show();
88 }
89
90
91 @Override
92 public void onTimeSet(TimePicker view, int hourOfDay, int minute) {
93     myHour = hourOfDay;
94     myMinute = minute;
95
96     Log.d( tag: "date", msg: "Year: " + myYear + "\n" +
97         "Month: " + (myMonth + 1) + "\n" +
98         "Day: " + myDay + "\n" +
99         "Hour: " + myHour + "\n" +
100         "Minute: " + myMinute);
101 }
102
103 }
104

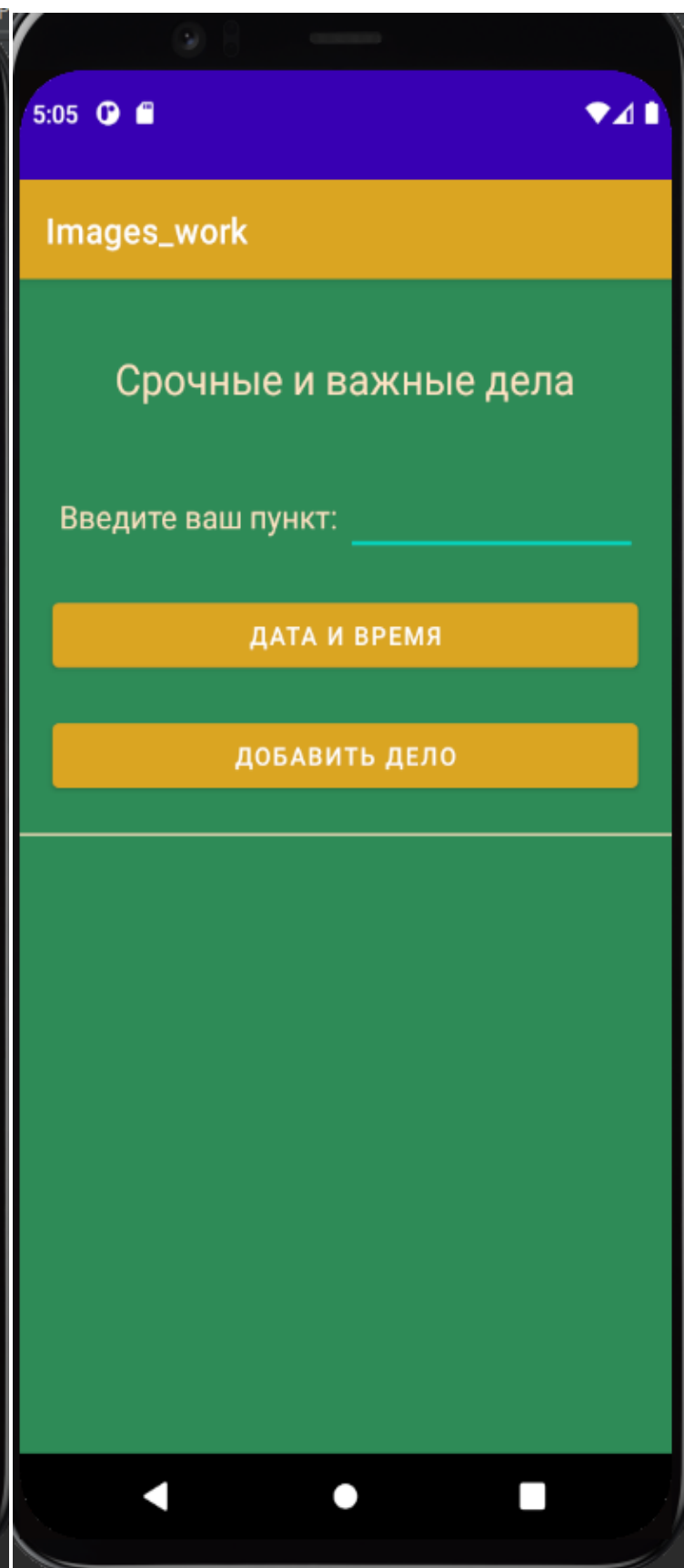
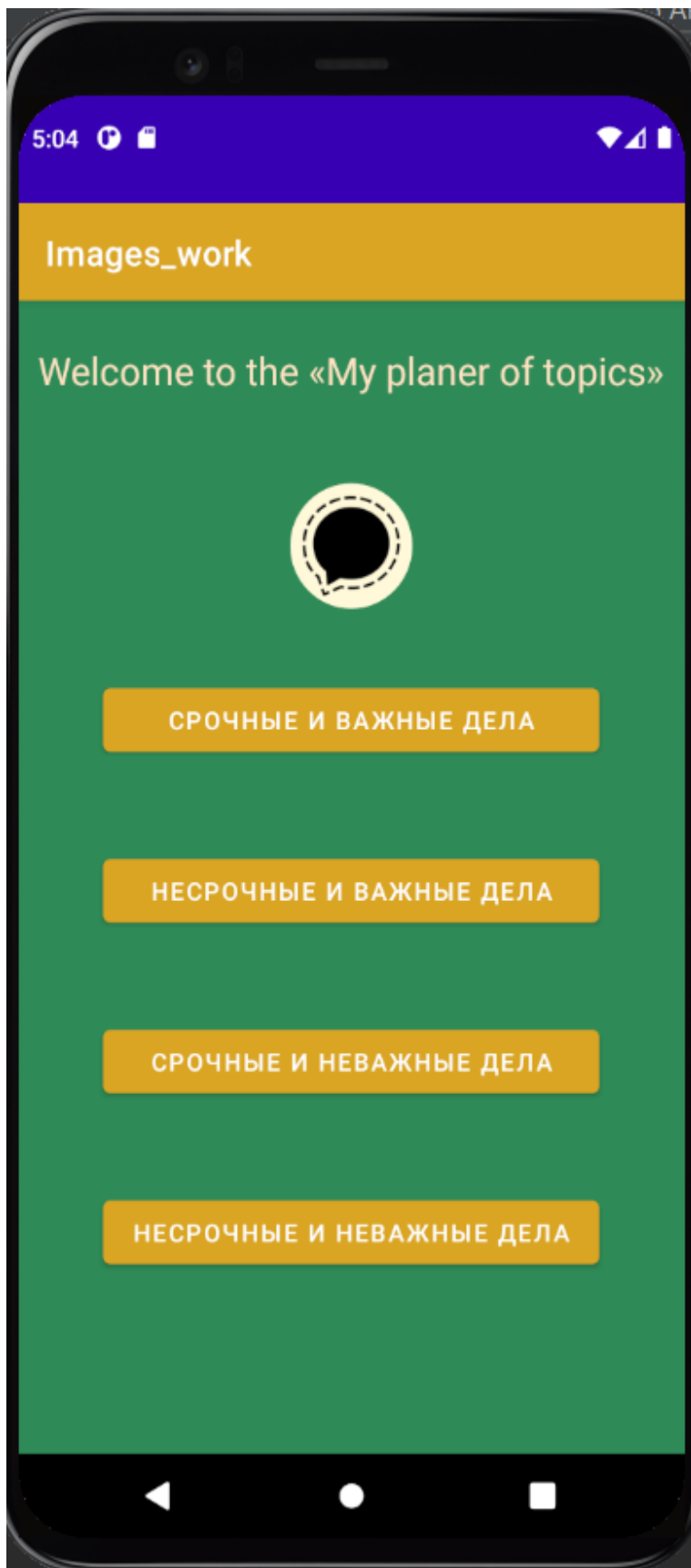
```

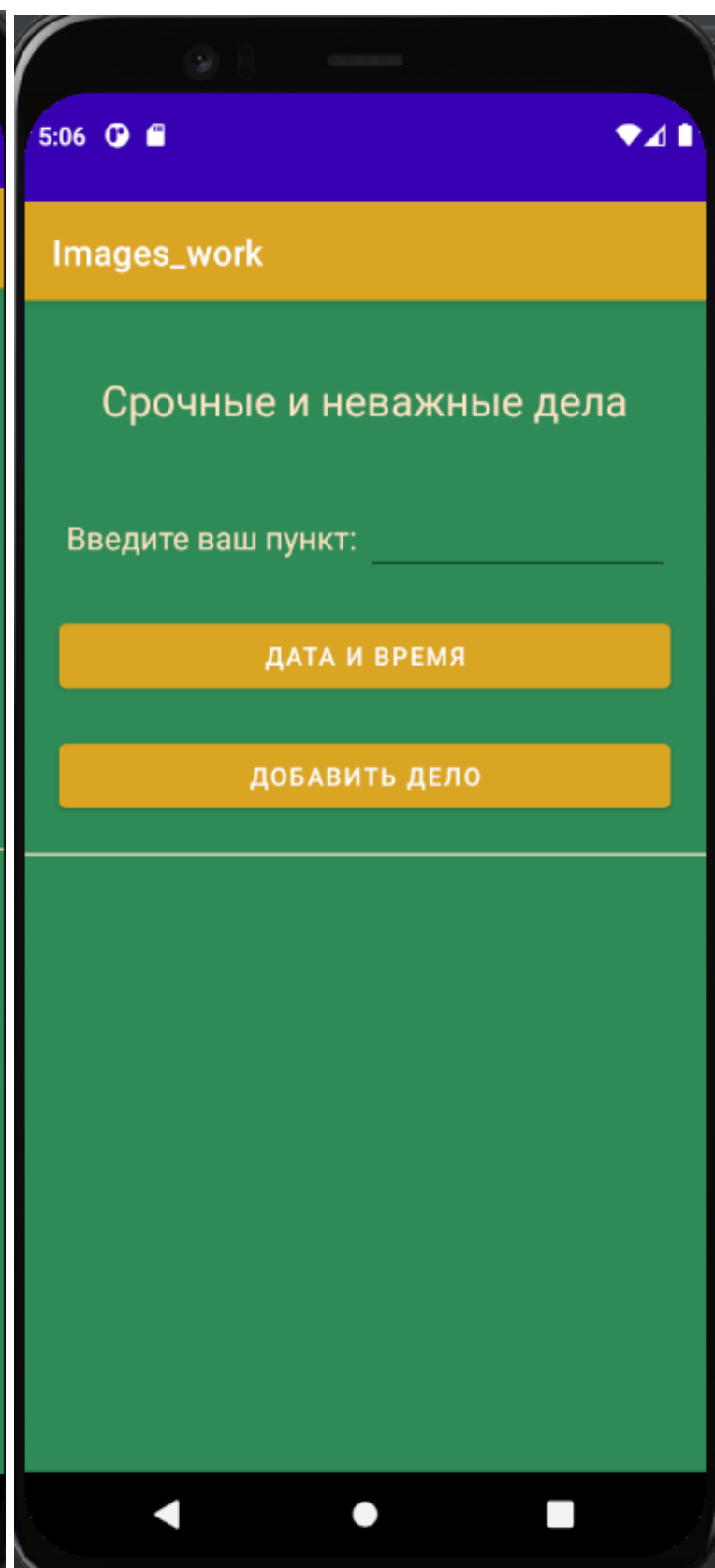
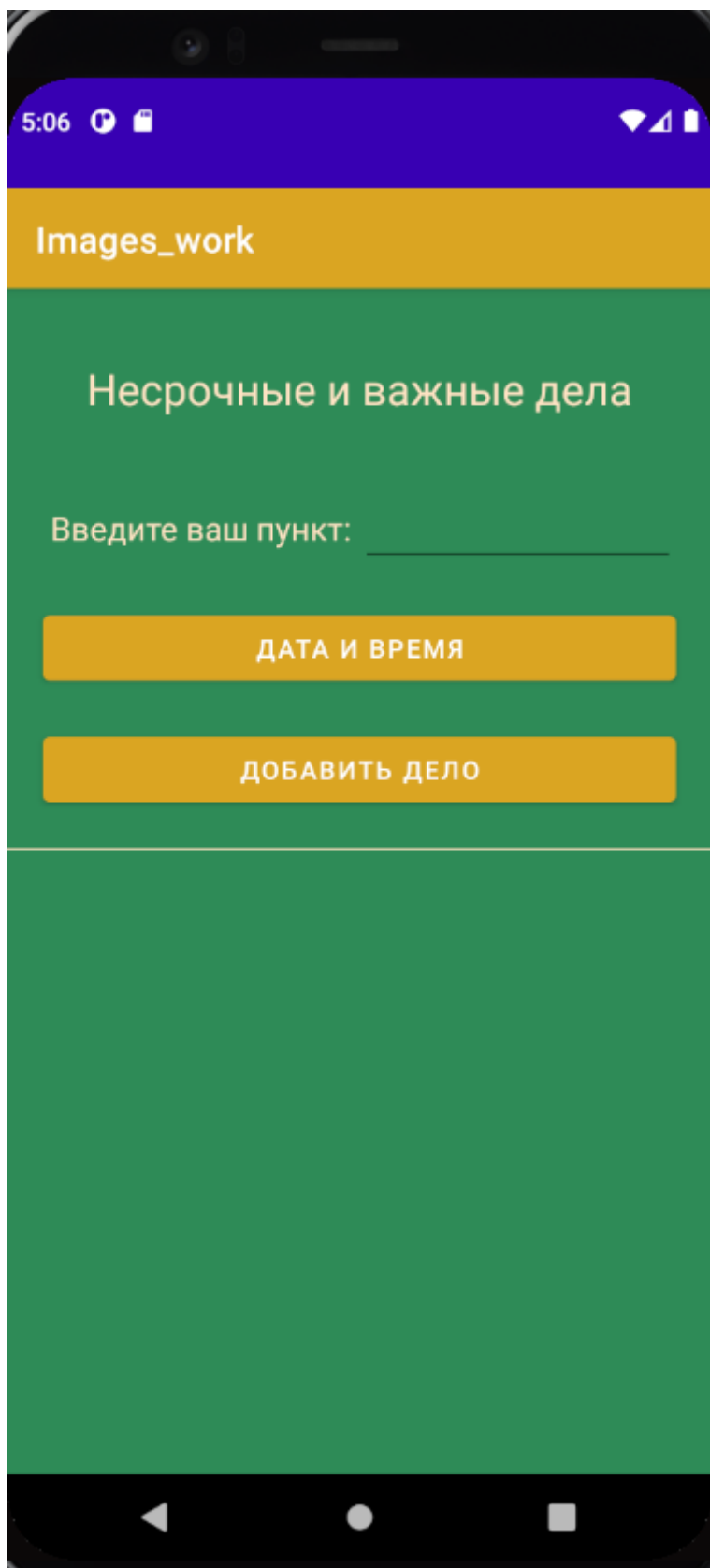


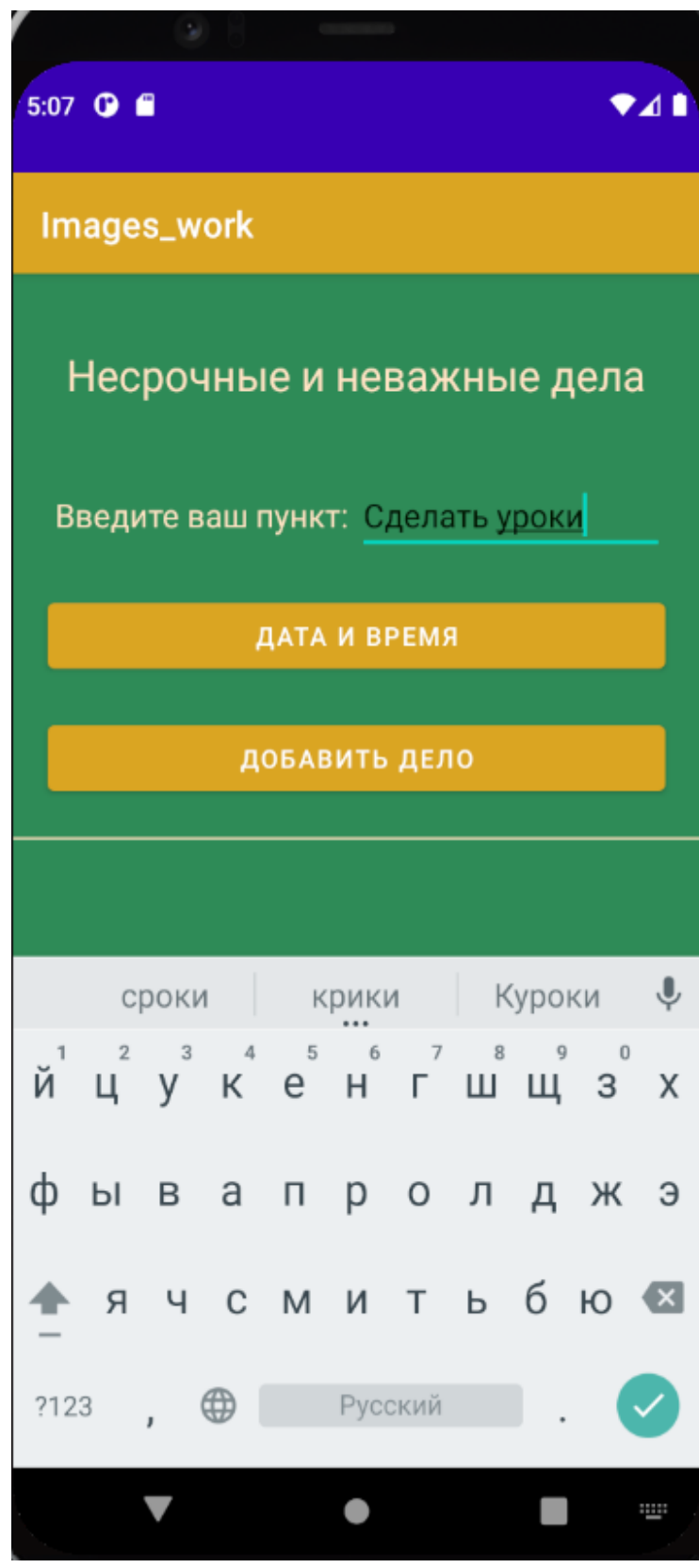
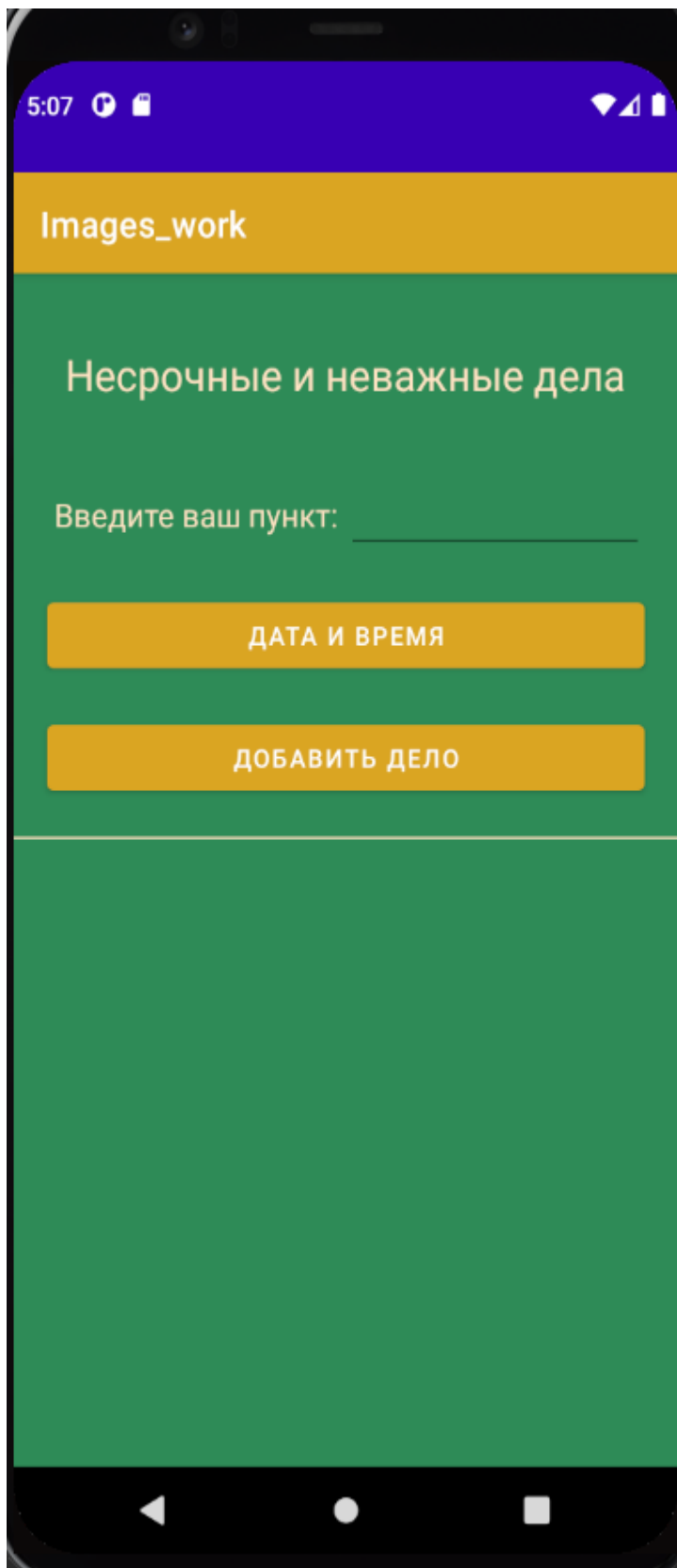
The screenshot shows the Android Studio IDE with the Logcat window open. The Logcat window displays a series of log messages from the application. The messages include system logs from OpenGL and AssistStructure, as well as custom logs from the application. The custom logs show the date and time in a specific format.

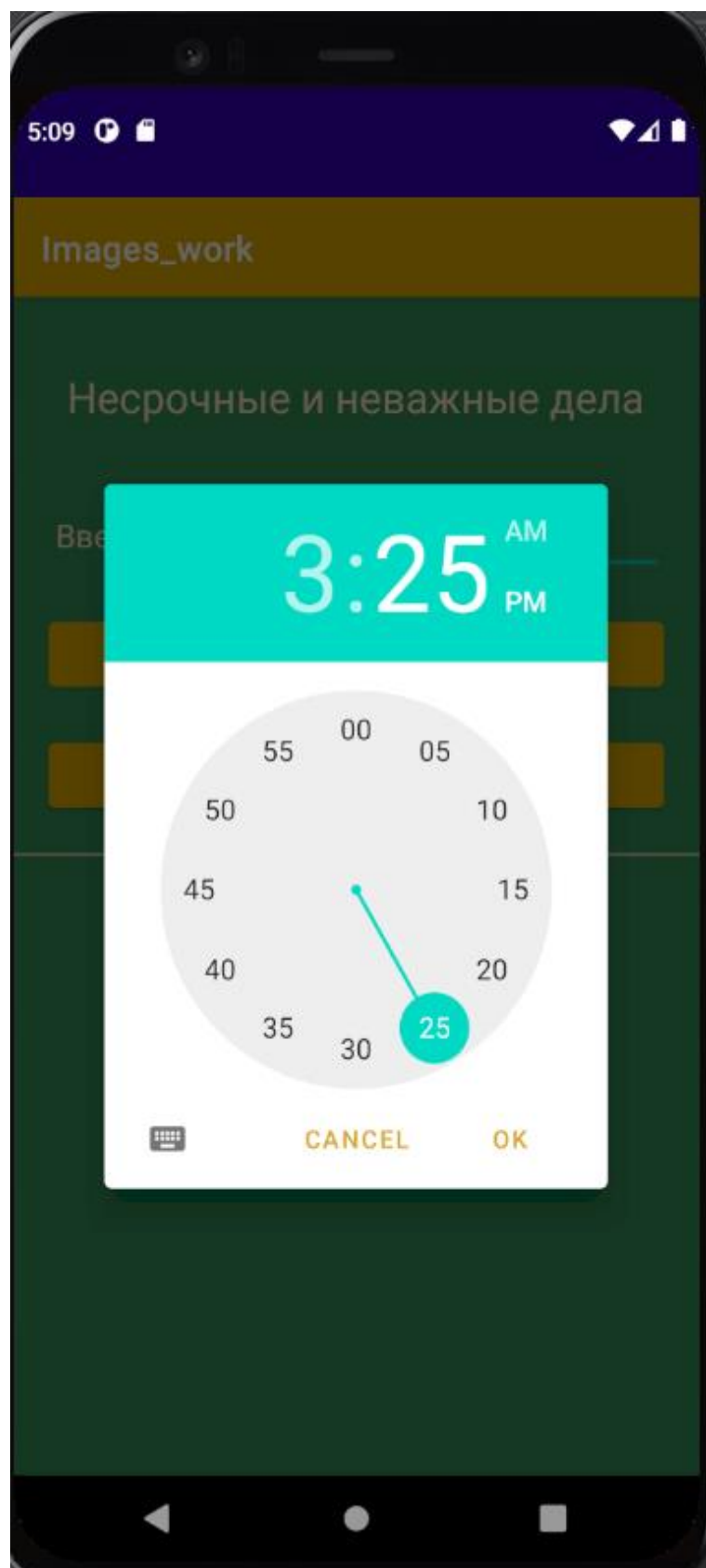
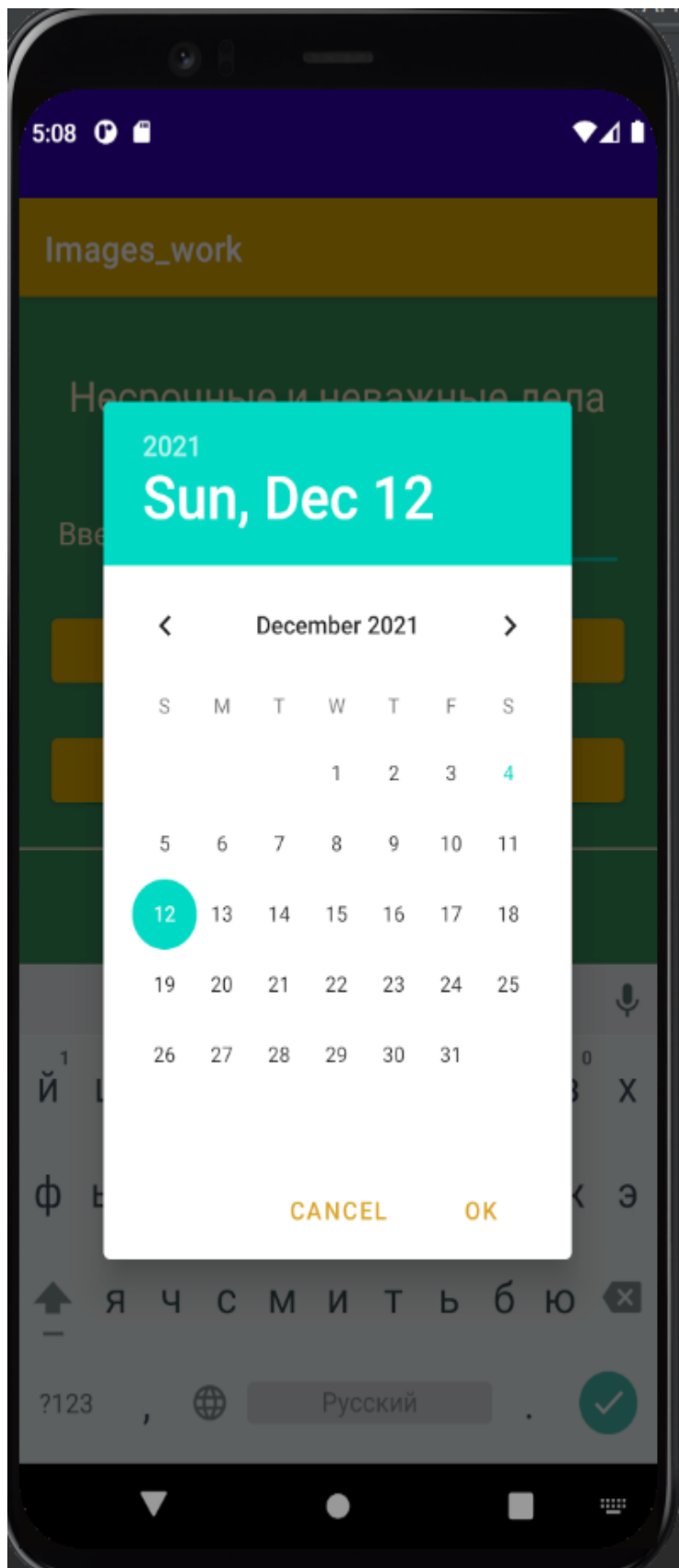
```
Month: 12
Day: 8
Hour: 7
Minute: 3
D/OpenGLRenderer: endAllActiveAnimators on 0xe262f950 (RippleDrawable) with handle 0xb9750010
D/AutoFillManager: onActivityFinishing(): calling cancelLocked()
I/AssistStructure: Flattened final assist data: 2004 bytes, containing 1 windows, 12 views
D/AutoFillManager: onActivityFinishing(): calling cancelLocked()
I/AssistStructure: Flattened final assist data: 2012 bytes, containing 1 windows, 12 views
D/OpenGLRenderer: endAllActiveAnimators on 0xb8355c10 (RippleDrawable) with handle 0xb97593d0
D/date: Year: 2021
      Month: 12
      Day: 12
      Hour: 15
      Minute: 25
D/OpenGLRenderer: endAllActiveAnimators on 0xb8366f90 (RippleDrawable) with handle 0xb975df50
D/OpenGLRenderer: endAllActiveAnimators on 0xb836c2f0 (RippleDrawable) with handle 0xb9762d10
D/date: Year: 2021
      Month: 12
      Day: 22
      Hour: 7
      Minute: 0
D/OpenGLRenderer: endAllActiveAnimators on 0xb7d803b0 (RippleDrawable) with handle 0xb975f7f0
D/OpenGLRenderer: endAllActiveAnimators on 0xb7d88290 (RippleDrawable) with handle 0xb976e090
D/date: Year: 2021
      Month: 12
      Day: 28
      Hour: 13
      Minute: 30
```

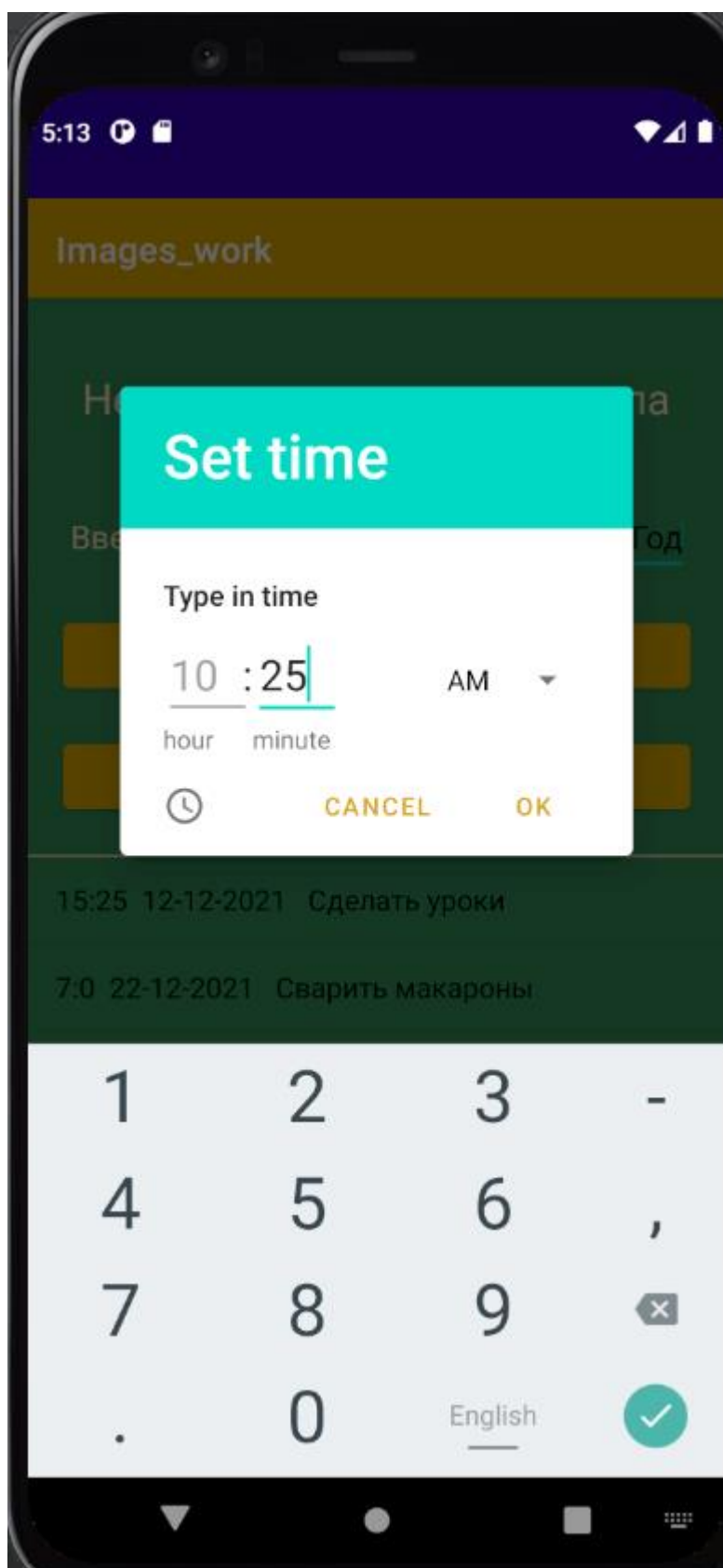
Таким образом, мы получаем следующие результаты работы приложения:

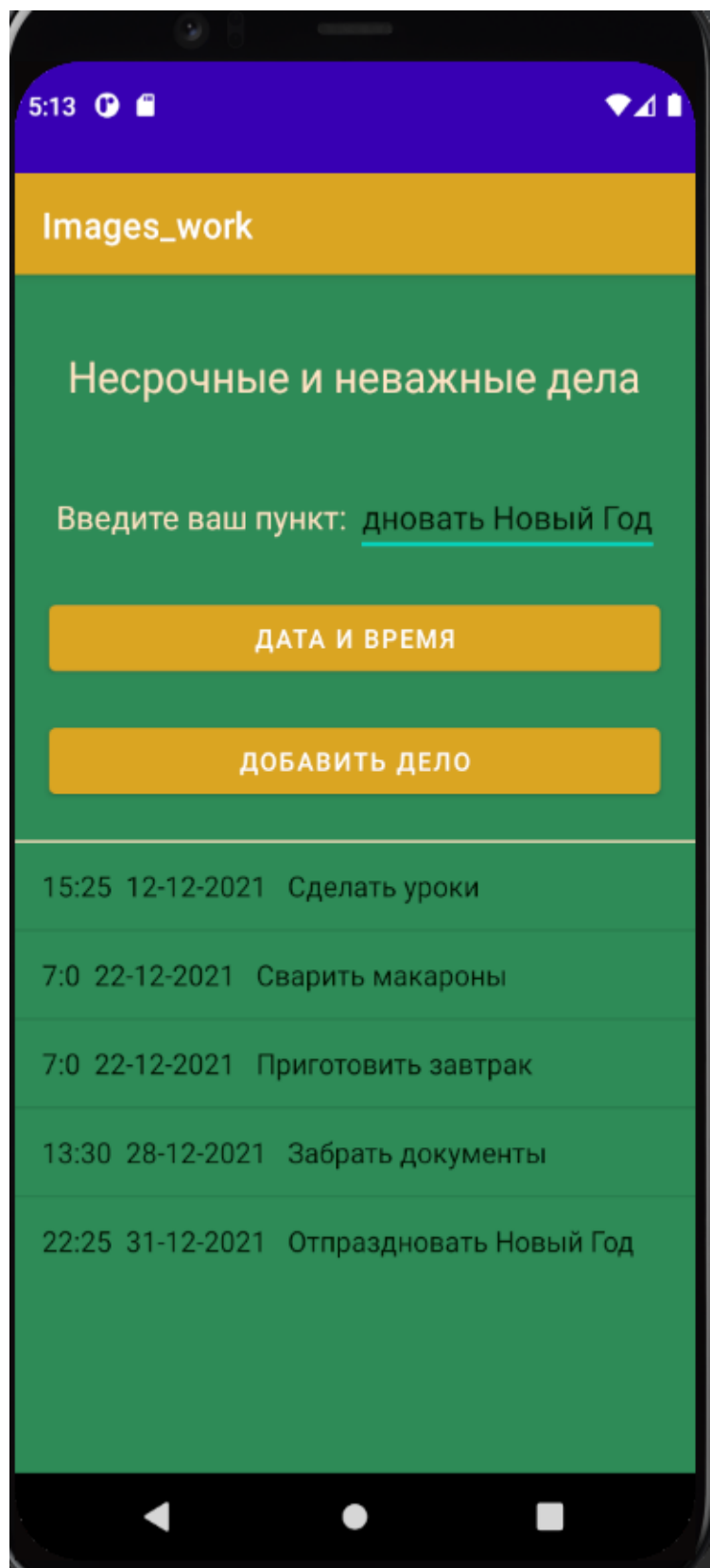
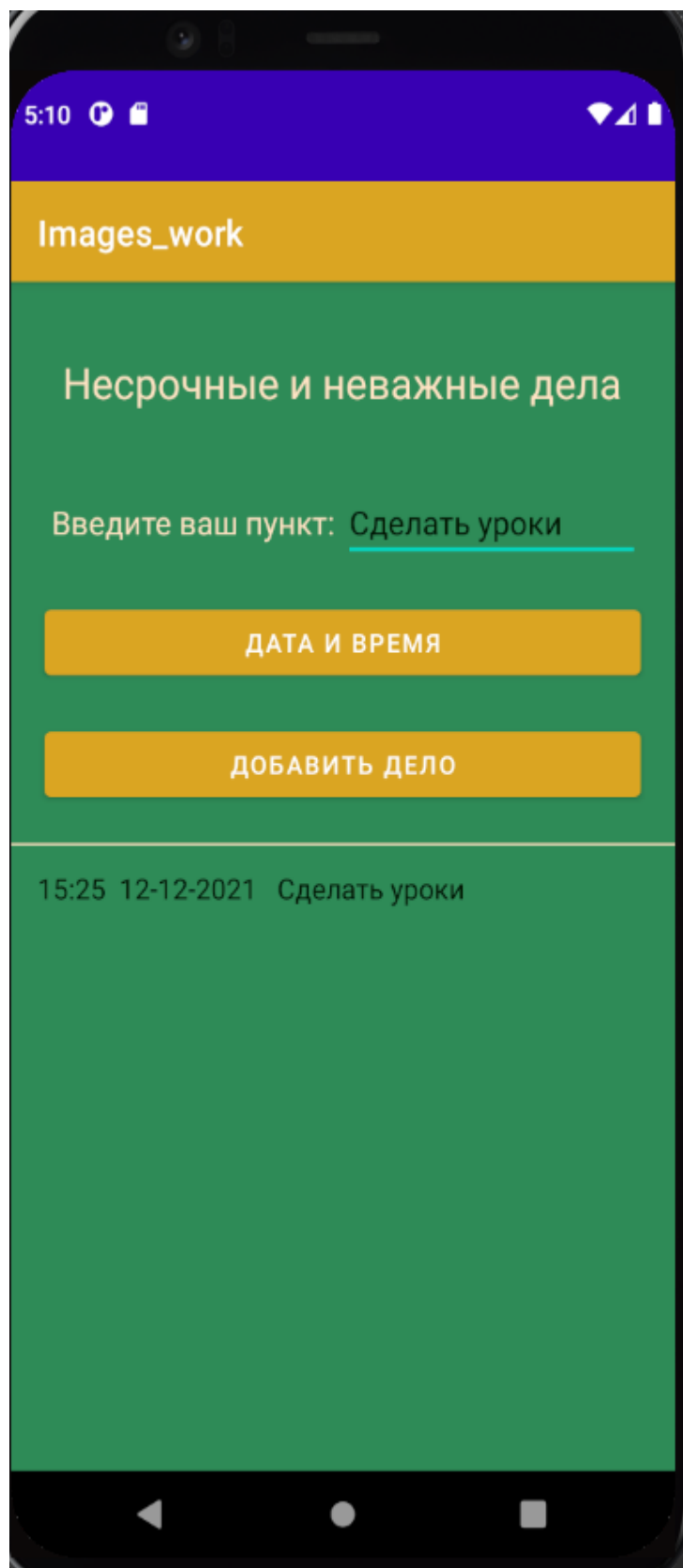












Работа приложения отлично себя продемонстрировала!

Для полноценного ознакомления прикладываю ссылки на свои репозитории GitHub с основной и дополнительной частями:

Основная часть (https://github.com/Valyaevgeorgiy/Android_projects/tree/main/Images_work):

Java-файлы проекта –

https://github.com/Valyaevgeorgiy/Android_projects/tree/main/Images_work/app/src/main/java/com/example/images_work

Изображение-иконка приложения –

https://github.com/Valyaevgeorgiy/Android_projects/blob/main/Images_work/app/src/main/res/drawable/ic_launcher.png

XML-файлы активностей проекта –

https://github.com/Valyaevgeorgiy/Android_projects/tree/main/Images_work/app/src/main/res/layout

Цвета, темы и строковые ресурсы проекта (XML) –

https://github.com/Valyaevgeorgiy/Android_projects/tree/main/Images_work/app/src/main/res/values

Манифест приложения (AndroidManifest.xml) –

https://github.com/Valyaevgeorgiy/Android_projects/blob/main/Images_work/app/src/main/AndroidManifest.xml

Дополнительная часть

(https://github.com/Valyaevgeorgiy/Android_projects/tree/main/Intent_filt/Filter_intent_additional_tasks):

Java-код активностей –

https://github.com/Valyaevgeorgiy/Android_projects/tree/main/Intent_filt/Filter_in_tent_additional_tasks/app/src/main/java/com/example/filter_intent_additional_task_s

XML-код активностей –

https://github.com/Valyaevgeorgiy/Android_projects/tree/main/Intent_filt/Filter_in_tent_additional_tasks/app/src/main/res/layout

Строковые ресурсы проекта –

https://github.com/Valyaevgeorgiy/Android_projects/blob/main/Intent_filt/Filter_in_tent_additional_tasks/app/src/main/res/values/strings.xml