

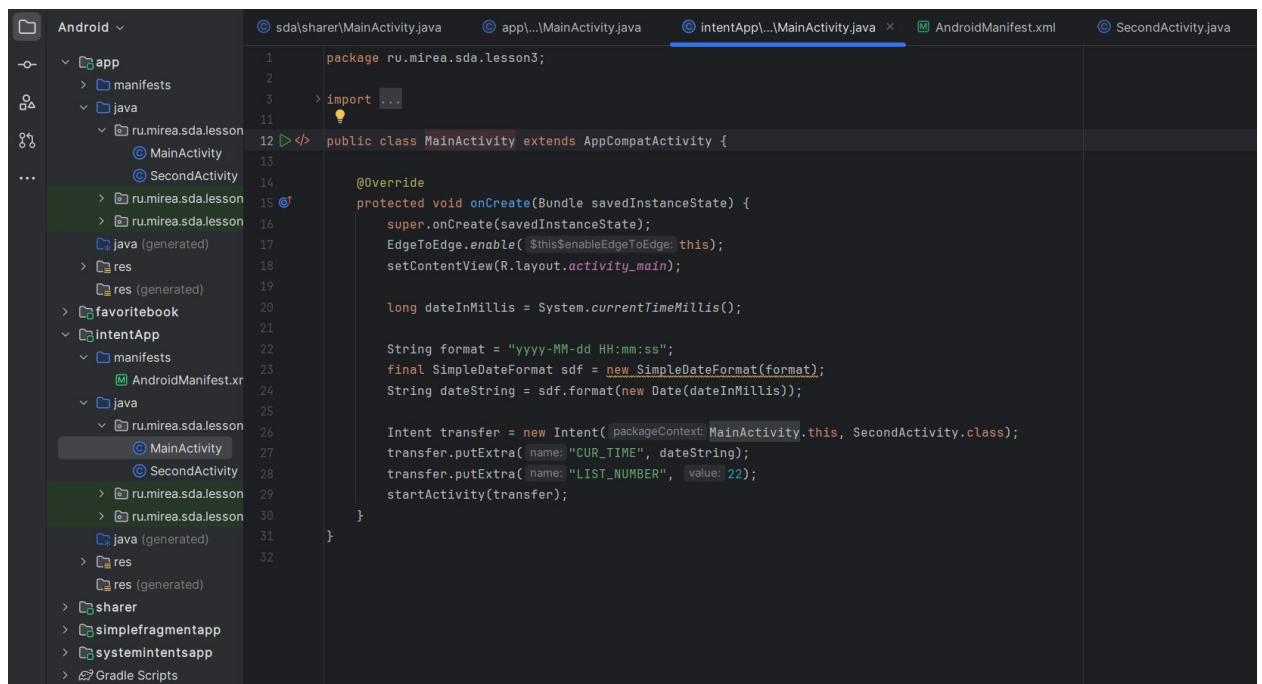
Практическая работа №3

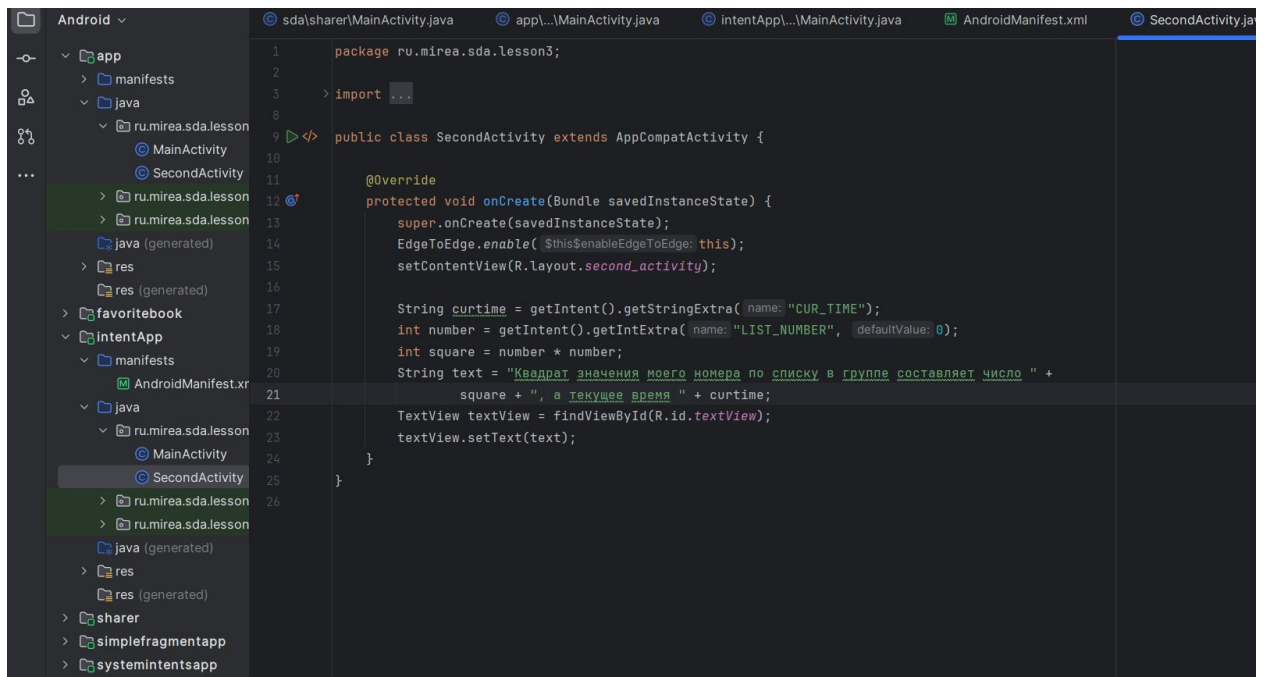
Работа с интенентами, диалогами, фрагментами и системными вызовами

1. Передача данных между активностями

Создан новый модуль IntentApp, включающий две активности: MainActivity и SecondActivity. В первой активности размещены элементы интерфейса для ввода времени и передачи данных. При нажатии на кнопку создаётся Intent, в который передаётся текущее время и рассчитанное значение (квадрат номера по списку).

Во второй активности реализован приём данных из интента и отображение строки в формате: «КВАДРАТ ЗНАЧЕНИЯ МОЕГО НОМЕРА ПО СПИСКУ В ГРУППЕ СОСТАВЛЯЕТ ЧИСЛО, а текущее время ВРЕМЯ». Скриншоты отражают корректную работу передачи и отображения.

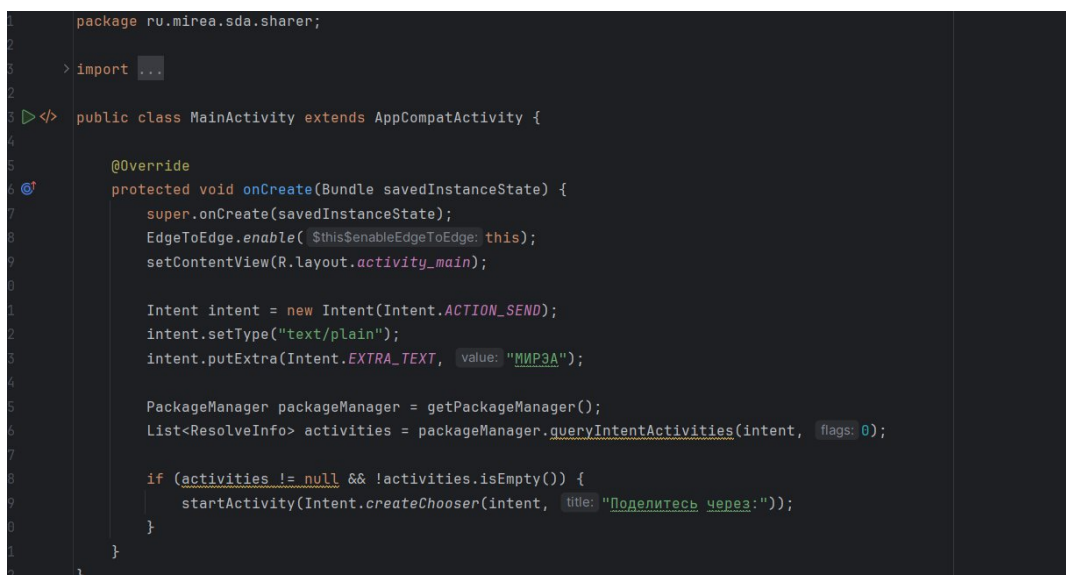




Квадрат значения моего номера по списку
в группе составляет число 529, а текущее
время 2025-06-03 15:34:30

2. Обработка намерений (intent-filter)

Создан модуль Sharer. В AndroidManifest.xml настроен фильтр интента для обработки внешнего запроса типа "text/plain". В MainActivity реализован вывод полученного текста в TextView, если активность была запущена системой через общий доступ.



Приложение проверено путём запуска с отправкой данных из другого приложения. Полученные данные корректно отображаются на экране.

3. Приложение с вводом и отображением любимой книги

Создан модуль FavoriteBook, состоящий из двух активностей:

- В MainActivity реализовано отображение текста с местом для вывода информации о книге и цитате, а также кнопка перехода на экран ввода.

```
public class MainActivity extends AppCompatActivity {  
    2 usages  
    private TextView textViewUserBook;  
    2 usages  
    private ActivityResultLauncher<Intent> activityResultLauncher;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        EdgeToEdge.enable( $this$enableEdgeToEdge: this);  
        setContentView(R.layout.activity_main);  
  
        textViewUserBook = findViewById(R.id.textView);  
  
        activityResultLauncher = registerForActivityResult(  
            new ActivityResultContracts.StartActivityForResult(),  
            new ActivityResultCallback<ActivityResult>() {  
                @Override  
                public void onActivityResult(ActivityResult result) {  
                    if (result.getResultCode() == Activity.RESULT_OK) {  
                        Intent data = result.getData();  
                        if (data != null) {  
                            String bookName = data.getStringExtra(USER_MESSAGE);  
                            String quoteText = data.getStringExtra(ShareActivity.QUOTES_KEY);  
                            String display = "Вы указали книгу: \"\" + bookName + "\".\nЦитата: \" + quoteText;  
                            textViewUserBook.setText(display);  
                        }  
                    }  
                }  
            }  
        );  
  
        Button butt = findViewById(R.id.button);  
        butt.setOnClickListener(new View.OnClickListener() {  
            @Override  
            public void onClick(View v) {  
                Intent intent = new Intent( packageContext: MainActivity.this, ShareActivity.class);  
                activityResultLauncher.launch(intent);  
            }  
        });  
    }  
}
```

- В ShareActivity размещены два поля ввода (EditText) и кнопка. Введённые значения передаются обратно и отображаются в первой активности в формате: «Название Вашей любимой книги: КНИГА. Цитата: ЦИТАТА».

```

3 public class ShareActivity extends AppCompatActivity {
5     static final String QUOTES_KEY = "quotes_name";
6
7     2 usages
8     private TextView textView;
9     no usages
10    private EditText editTextMessage;
11
12    @Override
13    protected void onCreate(Bundle savedInstanceState) {
14        super.onCreate(savedInstanceState);
15        setContentView(R.layout.shared_activity);
16
17        textView = findViewById(R.id.textTitle);
18        EditText bookInput = findViewById(R.id.editTitle);
19        EditText quoteInput = findViewById(R.id.editQuote);
20        Button sendButton = findViewById(R.id.button3);
21
22        Bundle data = getIntent().getExtras();
23        if (data != null) {
24            String book = data.getString(MainActivity.BOOK_NAME_KEY);
25            String quote = data.getString(MainActivity.QUOTES_KEY);
26            String formatted = String.format("Вы ранее выбрали: %s\nЦитата: %s", book, quote);
27            textView.setText(formatted);
28        }
29
30        sendButton.setOnClickListener( View v -> {
31            String title = bookInput.getText().toString();
32            String quotes = quoteInput.getText().toString();
33
34            Intent resultIntent = new Intent();
35            resultIntent.putExtra(MainActivity.USER_MESSAGE, title);
36            resultIntent.putExtra(ShareActivity.QUOTES_KEY, quotes);
37
38            setResult(Activity.RESULT_OK, resultIntent);
39            finish();
40        });
41    }
42}

```

<div>Ввести данные</div> <div>Здесь отобразится название книги и цитата</div>	<div>Любимая книга</div> <div>Введите название книги</div> <div>Цитата из книги</div> <div>Введите любимую цитату</div> <div>Отправить данные</div>
---	---

Механизм основан на `startActivityForResult` и `setResult`. Интерфейс оформлен вручную, все элементы протестированы.

4. Системные вызовы с помощью интенгов

Создан модуль `SystemIntentsApp`, реализующий следующие функции:

- Открытие веб-страницы через браузер (`Intent.ACTION_VIEW` с URL).
- Открытие координат на карте с передачей гео-URI.
- Вызов номера телефона через `Intent.ACTION_DIAL`.

```

1 package ru.mirea.sda.systemintentsapp;
2
3 > import ...
4
5 public class MainActivity extends AppCompatActivity {
6
7     @Override
8     protected void onCreate(Bundle savedInstanceState) {
9         super.onCreate(savedInstanceState);
10        setContentView(R.layout.activity_main);
11    }
12
13    1 usage
14    public void onClickCall(View view) {
15        Intent callIntent = new Intent(Intent.ACTION_DIAL);
16        callIntent.setData(Uri.parse(uriString: "tel:88005553535"));
17        startActivity(callIntent);
18    }
19
20    1 usage
21    public void onClickOpenBrowser(View view) {
22        Intent browserIntent = new Intent(Intent.ACTION_VIEW);
23        browserIntent.setData(Uri.parse(uriString: "https://online-edu.mirea.ru/login/index.php"));
24        startActivity(browserIntent);
25    }
26
27    1 usage
28    public void onClickOpenMaps(View view) {
29        Uri mapLocation = Uri.parse(uriString: "geo:55.749479,37.613944");
30        Intent mapIntent = new Intent(Intent.ACTION_VIEW, mapLocation);
31        startActivity(mapIntent);
32    }
33 }

```

Позвонить

Открыть браузер

Открыть карту

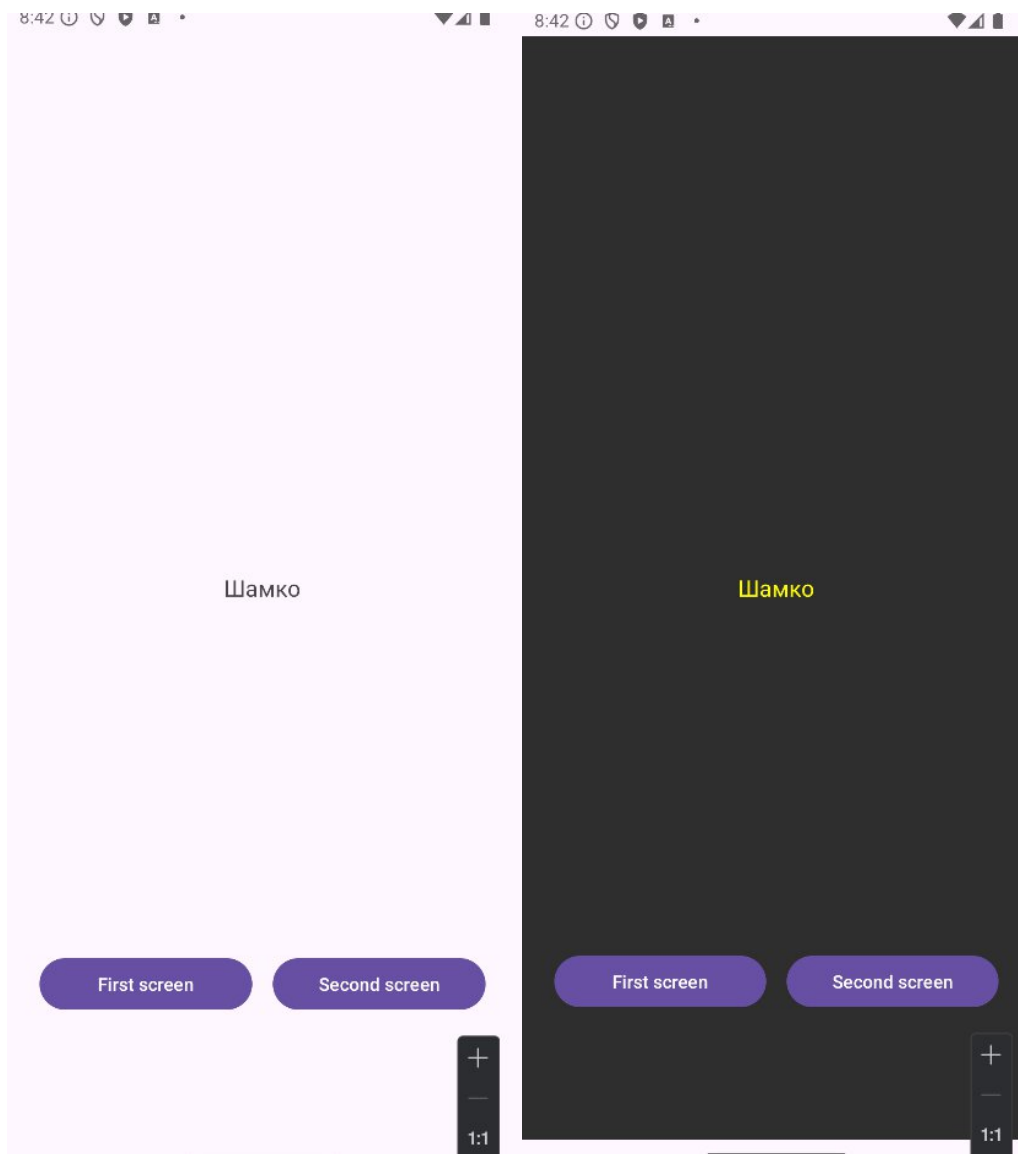
Каждая функция доступна через кнопку на экране. Приложение протестировано.

5. Приложение с фрагментами и поддержкой ориентации

Создан модуль SimpleFragmentApp. Созданы два фрагмента FirstFragment и SecondFragment. Реализована логика смены фона по нажатию кнопки во фрагменте.

В портретной ориентации отображается один фрагмент. Для альбомной ориентации добавлен макет в папке layout-land, в котором одновременно размещаются оба фрагмента.

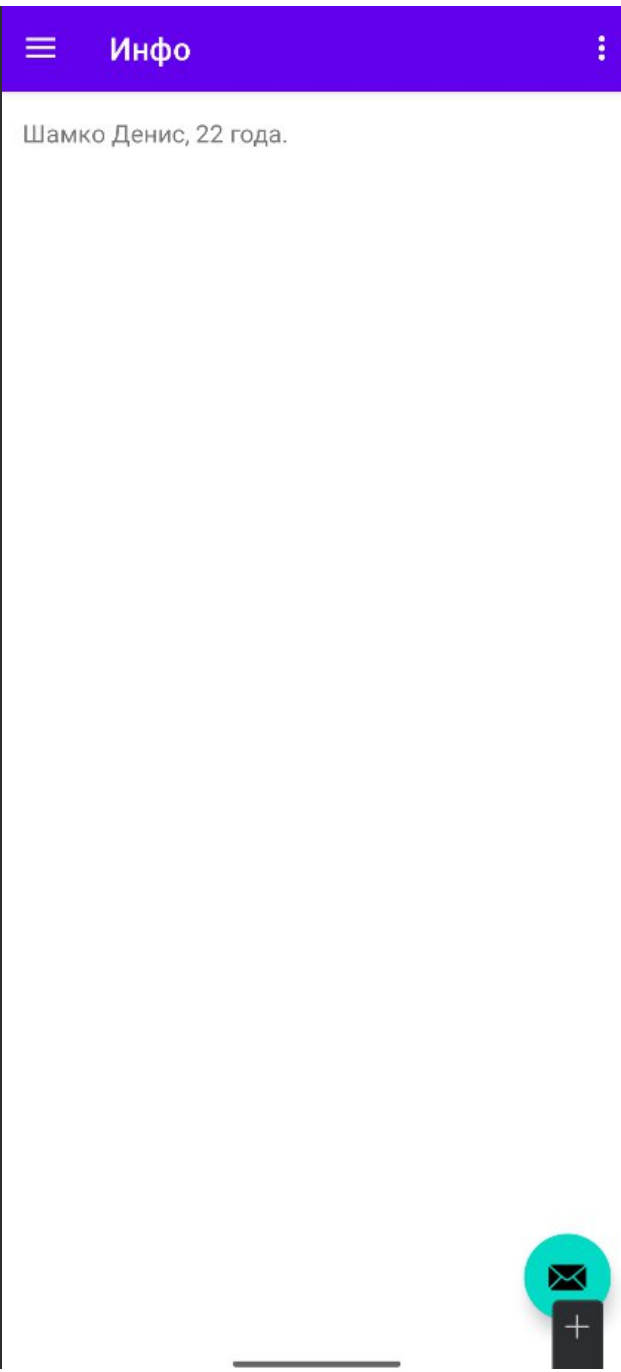
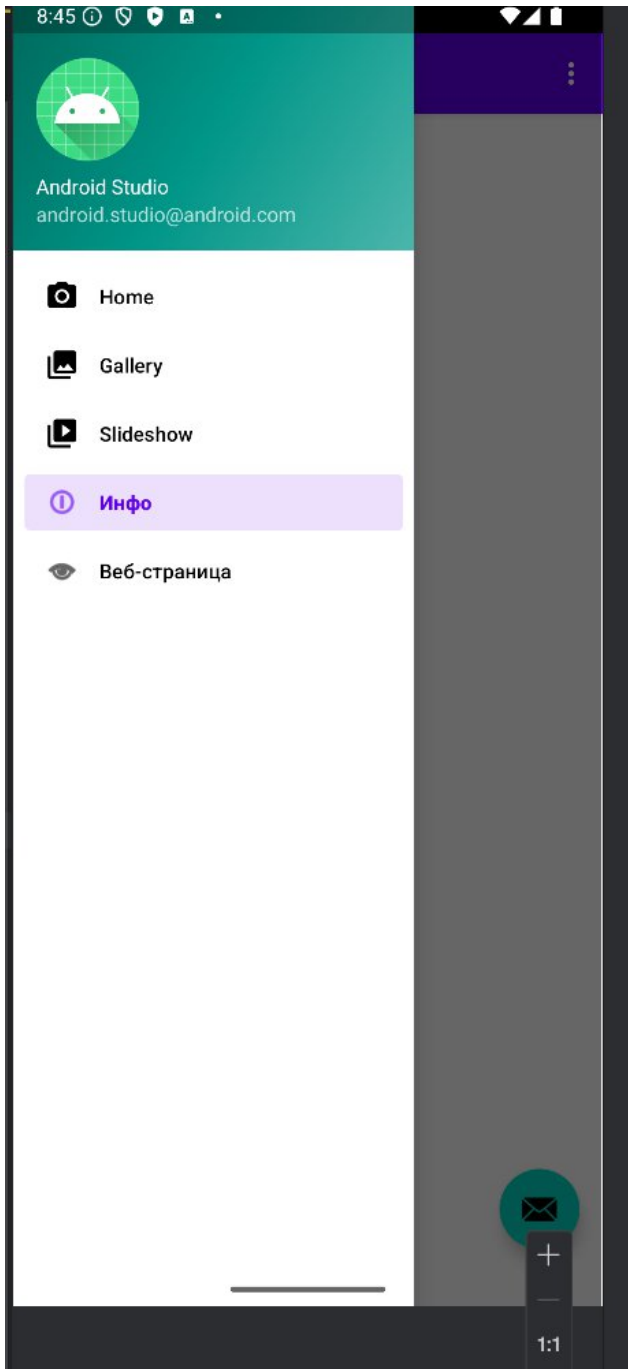
```
public class MainActivity extends AppCompatActivity {  
    2 usages  
    private Fragment fragment1;  
    2 usages  
    private Fragment fragment2;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        EdgeToEdge.enable(this);  
        setContentView(R.layout.activity_main);  
  
        ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (View view, WindowInsetsCompat insets) -> {  
            Insets bars = insets.getInsets(WindowInsetsCompat.Type.systemBars());  
            view.setPadding(bars.left, bars.top, bars.right, bars.bottom);  
            return insets;  
        });  
  
        fragment1 = new BlankFragment();  
        fragment2 = new BlankFragment2();  
  
        FragmentManager fragmentManager = getSupportFragmentManager();  
  
        Button btnFirstFragment = findViewById(R.id.btnFirstFragment);  
        Button btnSecondFragment = findViewById(R.id.btnSecondFragment);  
  
        btnFirstFragment.setOnClickListener(new View.OnClickListener() {  
            fragmentManager  
                .beginTransaction()  
                .replace(R.id.fragmentContainerView, fragment1)  
                .commit();  
        });  
  
        btnSecondFragment.setOnClickListener(new View.OnClickListener() {  
            fragmentManager  
                .beginTransaction()  
                .replace(R.id.fragmentContainerView, fragment2)  
                .commit();  
        });  
    }  
}
```

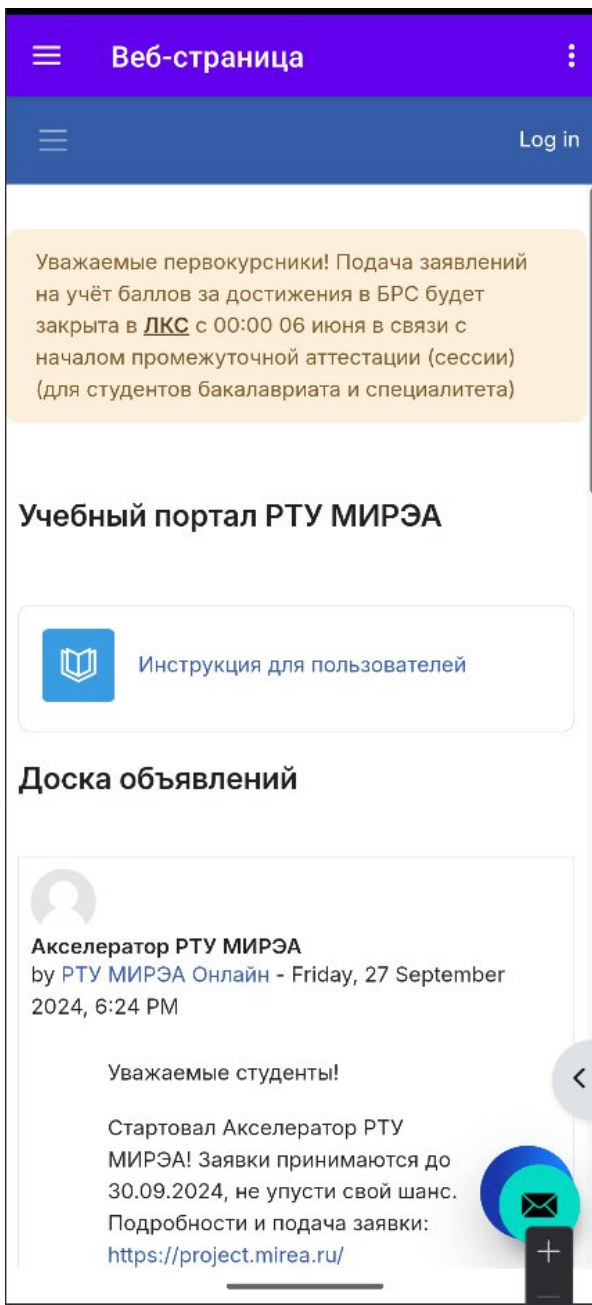


При переключении ориентации состояние сохраняется, а визуальные изменения (например, смена цвета) отображаются корректно. Приложение соответствует требованиям задания.

Контрольное задание

Создано новое приложение MireaProject с шаблоном Navigation Drawer Activity. Добавлены пользовательские фрагменты, текстовые элементы и пункты меню. Интерфейс оформлен вручную, адаптирован под боковое меню.





```

public class MainActivity extends AppCompatActivity {
    protected void onCreate(Bundle savedInstanceState) {

        binding = ActivityMainBinding.inflate(getLayoutInflater());
        setContentView(binding.getRoot());

        setSupportActionBar(binding.appBarMain.toolbar);
        binding.appBarMain.fab.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Snackbar.make(view, text: "Replace with your own action", Snackbar.LENGTH_LONG)
                    .setAction(text: "Action", listener: null)
                    .setAnchorView(R.id.fab).show();
            }
        });
        DrawerLayout drawer = binding.drawerLayout;
        NavigationView navigationView = binding.navView;
        mAppBarConfiguration = new AppBarConfiguration.Builder(
            R.id.nav_home, R.id.nav_gallery, R.id.nav_slideshow,
            R.id.nav_data, R.id.nav_web)
            .setOpenableLayout(drawer)
            .build();
        NavController navController = Navigation.findNavController(activity: this, R.id.nav_host_fragment_content_main);
        NavigationUI.setupActionBarWithNavController(activity: this, navController, mAppBarConfiguration);
        NavigationUI.setupWithNavController(navigationView, navController);
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.main, menu);
        return true;
    }

    5 usages
    @Override
    public boolean onSupportNavigateUp() {
        NavController navController = Navigation.findNavController(activity: this, R.id.nav_host_fragment_content_main);
        return NavigationUI.navigateUp(navController, mAppBarConfiguration)
            || super.onSupportNavigateUp();
    }
}

```