



**СибГУПИ**

СИБИРСКИЙ  
ГОСУДАРСТВЕННЫЙ  
УНИВЕРСИТЕТ  
ТЕЛЕКОММУНИКАЦИЙ  
И ИНФОРМАТИКИ



Минцифры  
России

Федеральное государственное бюджетное образовательное учреждение  
высшего образования

Сибирский государственный университет телекоммуникаций и информатики



Кафедра прикладной математики и кибернетики института информатики и вычислительной техники

# ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА БАКАЛАВРА

**Разработка мобильного приложения для автоматизации процессов  
обучения и контроля полученных знаний по заказу АНО ДПО  
«Магистраль-Авто»**

**Выполнил**

студент группы ИП-015

Шевелькова Валерия Юрьевна

**Руководитель**

к.т.н. доцент Ракитский Антон Андреевич

Новосибирск – 2024

# ЦЕЛИ И ЗАДАЧИ

- ❑ Провести анализ требований организации и изучить потребности целевой аудитории
- ❑ Разработать макет интерфейса пользователя, обеспечивающий интуитивно понятное взаимодействие
- ❑ Реализовать основной функционал: расписание занятий по вождению, тесты по правилам дорожного движения, раздел с теоретическими материалами
- ❑ Провести тестирование приложения и его отладку для обеспечения стабильной работы

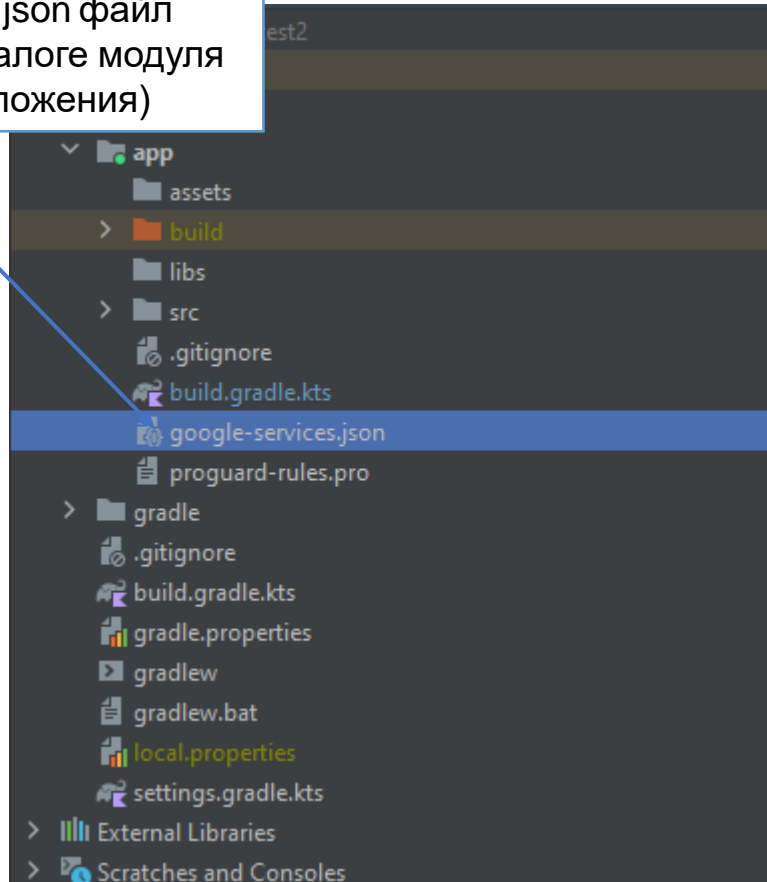
- ☐ База данных FireBase
- ☐ Язык программирования Java
- ☐ Среда разработки Android Studio
- ☐ Figma



# ОПИСАНИЕ РАЗРАБОТКИ

## Подключение базы данных к проекту

google-services.json файл  
в корневом каталоге модуля  
(на уровне приложения)



Добавление плагинов и  
зависимостей для БД

```
1 buildscript { this: ScriptHandlerScope
2     dependencies { this: DependencyHandlerScope
3         classpath("com.google.gms:google-services:4.4.2")
4     }
5 }
6 // Top-level build file where you can add configuration options common to all sub-projects/modules.
7 plugins { this: PluginDependenciesSpecScope
8     id("com.android.application") version "8.2.2" apply false
9     id("com.google.gms.google-services") version "4.4.2" apply false
10 }

1 plugins { this: PluginDependenciesSpecScope
2     id("com.android.application")
3     id("com.google.gms.google-services")
4 }

49 implementation(platform("com.google.firebase:firebase-bom:33.1.0"))
50 implementation("com.google.firebase:firebase-analytics")
```

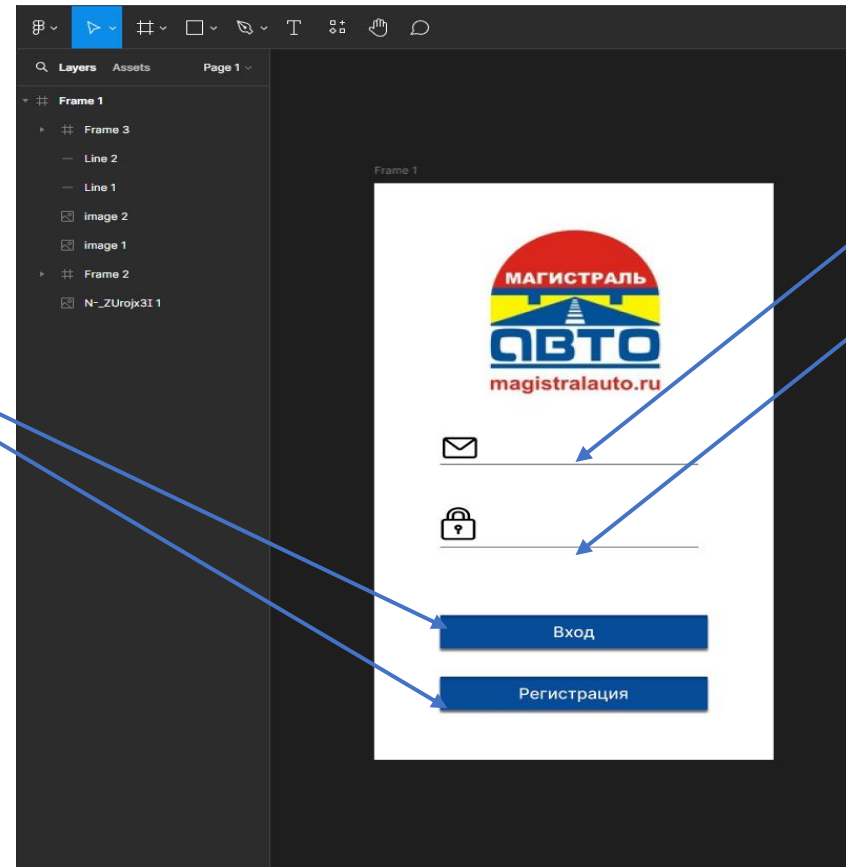
# ОПИСАНИЕ РАЗРАБОТКИ

## Создание макета в Figma

Кнопки вход и  
регистрация

Логин

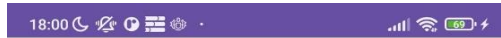
Пароль





# ОПИСАНИЕ РАЗРАБОТКИ

## Реализация окон регистрации и входа



логин

пароль

Вход

Регистрация

```
binding.button1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if (binding.editTextTextEmailAddress.getText().toString().isEmpty() || binding.editTextTextPassword.getText().toString().isEmpty()) {
            Toast.makeText(getApplicationContext(), "Fields cannot be ", Toast.LENGTH_SHORT).show();
        } else {
            FirebaseAuth.getInstance().signInWithEmailAndPassword(binding.editTextTextEmailAddress.getText().toString(), binding.editTextTextPassword.getText().toString());
            FirebaseAuth.getInstance().addOnCompleteListener(new OnCompleteListener<AuthResult>() {
                @Override
                public void onComplete(@NonNull Task<AuthResult> task) {
                    if (task.isSuccessful()) {
                        startActivity(new Intent(getApplicationContext(), LoginActivity.class));
                    }
                }
            });
        }
    }
});
```



# ОПИСАНИЕ РАЗРАБОТКИ

## Реализация окон регистрации и входа

2:34



valeriashh@mail.ru

Валерия

\*\*\*\*\*

Регистрация

```
public void onClick(View v){
    if (binding.editTextTextEmailAddress2.getText().toString().isEmpty() || binding.editTextName.getText().toString().isEmpty() || binding.editTextTextPassword2.getText().toString().isEmpty()){
        Toast.makeText(getApplicationContext(), "Fields cannot be empty", Toast.LENGTH_SHORT).show();
    } else {
        FirebaseAuth.getInstance().createUserWithEmailAndPassword(binding.editTextTextEmailAddress2.getText().toString(), binding.editTextTextPassword2.getText().toString())
        .addOnCompleteListener(new OnCompleteListener<AuthResult>() {
            @Override
            public void onComplete(@NonNull Task<AuthResult> task) {
                if (task.isSuccessful()){
                    String userId = FirebaseAuth.getInstance().getCurrentUser().getUid();
                    HashMap<String, Object> userInfo = new HashMap<>();
                    userInfo.put("email", binding.editTextTextEmailAddress2.getText().toString());
                    userInfo.put("userName", binding.editTextName.getText().toString());

                    // Schedule data
                    HashMap<String, HashMap<String, String>> schedule = new HashMap<>();
                    schedule.put("Пн", createScheduleEntry( date: "-", time: "-"));
                    schedule.put("Вт", createScheduleEntry( date: "-", time: "-"));
                    schedule.put("Ср", createScheduleEntry( date: "-", time: "-"));
                    schedule.put("Чт", createScheduleEntry( date: "-", time: "-"));
                    schedule.put("Пт", createScheduleEntry( date: "-", time: "-"));
                    schedule.put("Сб", createScheduleEntry( date: "-", time: "-"));
                    schedule.put("Вс", createScheduleEntry( date: "-", time: "-"));

                    userInfo.put("schedule", schedule);

                    FirebaseDatabase.getInstance().getReference().child( pathString: "Users").child(userId)
                        .setValue(userInfo);

                    Intent intent = new Intent( packageContext: RegisterActivity.this, MainActivity.class);
                    intent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TASK | Intent.FLAG_ACTIVITY_NEW_TASK);
                    startActivity(intent);
                }
            }
        });
    }
}
```





# ОПИСАНИЕ РАЗРАБОТКИ

## Реализация окна расписания

17:57 69%

Username: Лепа  
Email: djdh@mail.ru

Пн : 17.06	17:00-20:00
Вт : 18.06	17:00-20:00
Ср : -	-
Чт : 20.06	17:00-20:00
Пт : -	-
Сб : -	-
Вс : -	-

➞

Расписание    Билеты ПДД    Учебные материалы

```
public void onDataChange(@NonNull DataSnapshot snapshot) {  
    String username = snapshot.child( path: "userName").getValue(String.class);  
    String email = snapshot.child( path: "email").getValue(String.class);  
  
    Log.d( tag: "ScheduleFragment", msg: "Username: " + username);  
    Log.d( tag: "ScheduleFragment", msg: "Email: " + email);  
}
```

```
for (DataSnapshot daySnapshot : snapshot.child( path: "schedule").getChildren()) {  
    String day = daySnapshot.getKey();  
    String date = daySnapshot.child( path: "date").getValue(String.class);  
    String time = daySnapshot.child( path: "time").getValue(String.class);  
  
    // Add to schedule list  
    ScheduleItem item = new ScheduleItem(day, date, time);  
    scheduleList.add(item);  
}  
  
ScheduleItem item1 = scheduleList.get(0);  
ScheduleItem item2 = scheduleList.get(1);  
ScheduleItem item3 = scheduleList.get(2);  
ScheduleItem item4 = scheduleList.get(3);  
ScheduleItem item5 = scheduleList.get(4);  
ScheduleItem item6 = scheduleList.get(5);  
ScheduleItem item7 = scheduleList.get(6);  
vsDate.setText(item1.getDay() + " : " + item1.getDate());  
vsClock.setText(item1.getTime());  
  
vtDate.setText(item2.getDay() + " : " + item2.getDate());  
vtClock.setText(item2.getTime());  
  
pnDate.setText(item3.getDay() + " : " + item3.getDate());  
pnClock.setText(item3.getTime());  
  
ptDate.setText(item4.getDay() + " : " + item4.getDate());  
ptClock.setText(item4.getTime());  
  
sbDate.setText(item5.getDay() + " : " + item5.getDate());  
sbClock.setText(item5.getTime());  
  
srDate.setText(item6.getDay() + " : " + item6.getDate());  
srClock.setText(item6.getTime());  
  
chtDate.setText(item7.getDay() + " : " + item7.getDate());  
chtClock.setText(item7.getTime());
```

```
5Nma8mjszChA9LLchmjcd2H0PE83  
├── email: "valeriashhh@gmail.com"  
├── progress  
└── schedule  
    ├── Вс  
    ├── Вт  
    │   ├── date: "11.06"  
    │   └── time: "16:00-18:00"  
    ├── Пн  
    ├── Пт  
    ├── Сб  
    ├── Ср  
    └── Чт  
    └── userName: "Валерия"
```

# ОПИСАНИЕ РАЗРАБОТКИ

## Реализация окна расписания

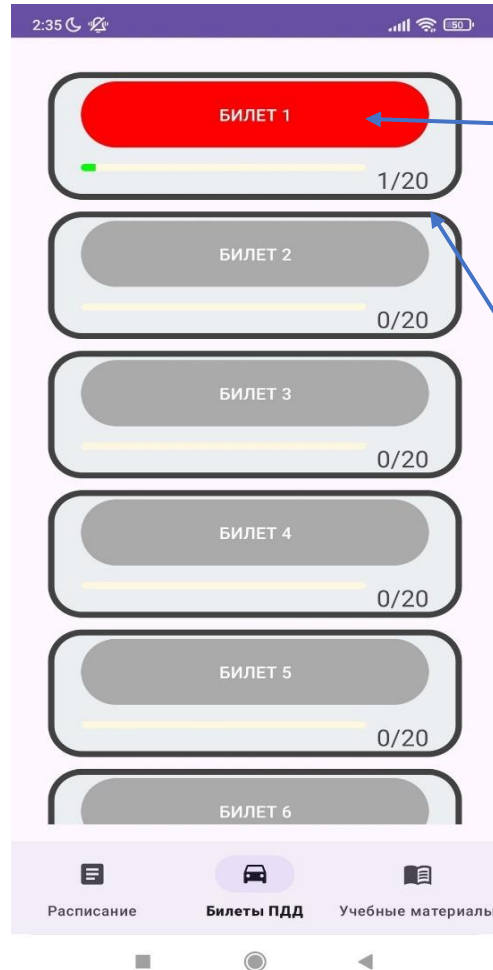


```
logoutButton = view.findViewById(R.id.logout_btn);  
new *  
logoutButton.setOnClickListener(new View.OnClickListener() {  
    new *  
    @Override  
    public void onClick(View v) {  
        FirebaseAuth.getInstance().signOut();  
        Intent intent = new Intent(getActivity(), LoginActivity.class);  
        intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK | Intent.FLAG_ACTIVITY_CLEAR_TASK);  
        startActivity(intent);  
    }  
});
```



# ОПИСАНИЕ РАЗРАБОТКИ

## Реализация окна билетов



```
private void openTicket1Fragment() {  
    Ticket1 ticket1Fragment = new Ticket1();  
    getParentFragmentManager().beginTransaction()  
        .replace(R.id.fragment_container, ticket1Fragment)  
        .addToBackStack( name: null)  
        .commit();  
}
```

```
@Override  
public void onDataChange(@NonNull DataSnapshot snapshot) {  
  
    DataSnapshot progressSnapshot = snapshot.child( path: "progress");  
    String[] progressArray = new String[(int) progressSnapshot.getChildrenCount()];  
    int index = 0;  
  
    for (DataSnapshot itemSnapshot : progressSnapshot.getChildren()) {  
        String progressValue = itemSnapshot.getValue(String.class);  
        progressArray[index] = progressValue;  
        index++;  
    }  
  
    for (int i = 0; i < 40; i++) { |  
        String progressValue = progressArray[i];  
  
        // Получаем идентификаторы ресурсов для ProgressBar и TextView  
        int progressBarId = getResources().getIdentifier( name: "ticket" + (i + 1) + "progressBar", defType: "id", requireContext().getPackageName());  
        int textViewId = getResources().getIdentifier( name: "ticket" + (i + 1) + "progressText", defType: "id", requireContext().getPackageName());  
        int buttonId = getResources().getIdentifier( name: "ticket" + (i + 1), defType: "id", requireContext().getPackageName());  
  
        // Находим соответствующие элементы в rootView  
        ProgressBar progressBar = rootView.findViewById(progressBarId);  
        TextView progressTextView = rootView.findViewById(textViewId);  
        Button ticketButton = rootView.findViewById(buttonId);
```

# ОПИСАНИЕ РАЗРАБОТКИ

## Реализация окна билетов

В каком случае водитель совершит вынужденную остановку?

☐ Остановившись непосредственно перед пешеходным переходом, чтобы уступить дорогу пешеходу

☐ Остановившись на проезжей части из-за технической неисправности транспортного средства

☐ В обоих перечисленных случаях

Ответить

Далее

Расписание Билеты ПДД Учебные материалы

В каком случае водитель совершит вынужденную остановку?

☐ Остановившись непосредственно перед пешеходным переходом, чтобы уступить дорогу пешеходу

☒ Остановившись на проезжей части из-за технической неисправности транспортного средства

☐ В обоих перечисленных случаях

Ответить

«Вынужденная остановка» - прекращение движения транспортного средства, связанное с его технической неисправностью, опасностью, создаваемой перевозимым грузом, состоянием водителя (пассажира) или появления препятствия на дороге. (Пункт 1.2 ПДД, термин «Вынужденная остановка»)

Далее

Расписание Билеты ПДД Учебные материалы

Разрешен ли Вам поворот на дорогу с грунтовым покрытием?

☐ Разрешен

☒ Разрешен только при технической неисправности транспортного средства

☐ Запрещен

Ответить

Неправильный ответ. Попробуйте еще раз.

Далее

Расписание Билеты ПДД Учебные материалы

```
private void checkAnswer() {
    int selectedId = answersRadioGroup.getCheckedRadioButtonId();
    if (selectedId != -1) {
        RadioButton selectedRadioButton = getView().findViewById(selectedId);
        int answerNumber = answersRadioGroup.indexOfChild(selectedRadioButton) + 1;

        if (String.valueOf(answerNumber).equals(correctAnswers[currentQuestionIndex])) {
            explanationTextView.setText(explanations[currentQuestionIndex]);
            explanationTextView.setVisibility(View.VISIBLE);
            explanationTextView.setTextColor(Color.GREEN);
            correctAnswerCount++;
        } else {
            explanationTextView.setText("Неправильный ответ. Попробуйте еще раз.");
            explanationTextView.setVisibility(View.VISIBLE);
            explanationTextView.setTextColor(Color.RED);
        }
    }

    submitBtn.setEnabled(false);
    for (int i = 0; i < answersRadioGroup.getChildCount(); i++) {
        answersRadioGroup.getChildAt(i).setEnabled(false);
    }

    buttonNext.setEnabled(true);
}
```

Тест завершен

Вы ответили на все вопросы.  
Правильных ответов: 2. Что хотите сделать дальше?

ПЕРЕЙТИ К ВЫБОРУ БИЛЕТОВ

ПРОЙТИ ЕЩЁ РАЗ

# ОПИСАНИЕ РАЗРАБОТКИ

## Реализация окна учебных материалов



```
@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
    Bundle savedInstanceState) {
    View rootView = inflater.inflate(R.layout.fragment_study_materials, container, attachToRoot: false);
    Button downloadButton = rootView.findViewById(R.id.downloadButton);

    new *
    downloadButton.setOnClickListener(new View.OnClickListener() {
        new *
        @Override
        public void onClick(View v) { downloadPDF(); }
    });

    return rootView;
}

private void downloadPDF() {
    try {
        // Открываем поток для чтения из файла в папке res/raw
        InputStream inputStream = getResources().openRawResource(R.raw.ustroistvo);

        // Создаем файл для записи во внутреннее хранилище
        File outputFile = new File(getActivity().getFilesDir(), child: "ustroistvo.pdf");
        OutputStream outputStream = new FileOutputStream(outputFile);

        // Читаем данные из файла и записываем их в новый файл
        byte[] buffer = new byte[1024];
        int length;
        while ((length = inputStream.read(buffer)) > 0) {
            outputStream.write(buffer, off: 0, length);
        }

        // Закрываем потоки
        outputStream.flush();
        outputStream.close();
        inputStream.close();
    } catch (IOException e) {
        e.printStackTrace();
    }
}
```

# ЗАКЛЮЧЕНИЕ

- ✓ В ходе выполнения данного проекта было реализовано мобильное приложение.
- ✓ Проект был успешно разработан и протестирован, что подтвердило его работоспособность.
- ✓ Приложение подготовлено и передано к дальнейшему внедрению.



**СибГУТИ**

СИБИРСКИЙ  
ГОСУДАРСТВЕННЫЙ  
УНИВЕРСИТЕТ  
ТЕЛЕКОММУНИКАЦИЙ  
И ИНФОРМАТИКИ



Минцифры  
России

**Спасибо за внимание**