

АВТОРСКИЙ ДОГОВОР

о передаче исключительных прав на использование произведения

г. 2024

«23» сентября 2024 г.

Гражданин Горшенин Андрей Игоревич, именуемый в дальнейшем "**Получатель**" с одной стороны, и гр. Корягин Дмитрий Игоревич, именуемый в дальнейшем «**Автор**», с другой стороны, именуемые в дальнейшем «Стороны», заключили настоящий договор, составленный на 7 (семи) листах, в дальнейшем «**Договор**», о нижеследующем:

1. ПРЕДМЕТ ДОГОВОРА

1.1. Автор предоставляет Получателю исключительные права на использование программного обеспечения - приложения "Wild Politics" для операционной системы Android (далее- Приложение), исходный код которого приведён в Приложении 1, а Получатель за предоставление этих прав уплачивает вознаграждение Автору. Под использованием в настоящем договоре понимается реализация произведения, в качестве товара, или иное его использование в гражданском обороте в пределах, предусмотренных договором.

1.2. Автор гарантирует наличие у него предоставляемых по настоящему договору авторских прав на произведение.

1.3. Автор гарантирует наличие у передаваемого в соответствии с настоящим договором произведения характеристик, указанных в Приложении.

2. ПРАВА И ОБЯЗАННОСТИ СТОРОН

2.1. По настоящему договору Автор предоставляет Получателю следующие права:

- право использовать произведение под фирменным наименованием, производственной маркой и товарным знаком Получателя. При этом каждый экземпляр произведения не должен содержать имя или псевдоним автора.
- право на обнародование произведения, т.е. на сообщение произведения в какой-либо форме или каким-либо способом неопределенному кругу лиц. Не считается обнародованием информирование широкого круга лиц о назначении, функциях, технических и прочих характеристиках произведения, например, в рекламных целях;
- право на воспроизведение произведения (дублирование, тиражирование или иное размножение, т.е. неоднократное придание произведению объективной формы, допускающей его функциональное использование), без ограничения тиража;
- право на распространение произведения любым способом путем реализации размноженных материальных носителей произведения среди конечных пользователей (потребителей, осуществляющих функциональное использование), без ограничений;

- право на переработку произведения (создание на его основе нового, творчески самостоятельного произведения) (или внесение изменений, не представляющих собой его переработку);
- право на перевод произведения;
- право на публичное использование произведения и демонстрацию в информационных, рекламных и прочих целях;
- право переуступить на договорных условиях часть полученных по настоящему договору прав третьим лицам;

2.2. Автор не сохраняет за собой право использовать самостоятельно или предоставлять аналогичные права на его использование третьим лицам.

2.3. Получатель и Автору пришли к соглашению, что вознаграждением за передачу прав на Приложение является единоразовая компенсация в сумме 1000 (одна тысяча) российских рублей.

3. КОНТРОЛЬ ЗА ИСПОЛЬЗОВАНИЕМ ПРОИЗВЕДЕНИЯ

3.1. Автор вправе:

- публиковать в открытом доступе информацию о том, что является автором Приложения, без необходимости указания об утрате исключительных прав на него;

4. ОТВЕТСТВЕННОСТЬ СТОРОН

4.1. Сторона, не исполнившая или ненадлежащим образом исполнившая обязательства по настоящему договору, обязана возместить другой стороне причиненные таким неисполнением убытки. Если сторона, нарушившая договор, получила вследствие этого доходы, сторона, права которой нарушены, вправе требовать возмещения наряду с другими убытками упущенной выгоды в размере не меньшем, чем такие доходы.

4.2. В случае нарушения договора сторона, чье право нарушено, вправе также потребовать признания права, восстановления положения, существовавшего до нарушения права, и прекращения действий, нарушающих право или создающих угрозу его нарушения.

4.3. В случаях, не предусмотренных настоящим договором, имущественная ответственность определяется в соответствии с действующим законодательством РФ.

5. КОНФИДЕНЦИАЛЬНОСТЬ

5.1. Условия настоящего договора и дополнительных соглашений к нему конфиденциальны и не подлежат разглашению.

6. РАЗРЕШЕНИЕ СПОРОВ

6.1. Все споры и разногласия, которые могут возникнуть между сторонами по вопросам, не нашедшим своего разрешения в тексте данного договора, будут разрешаться путем переговоров на основе действующего законодательства и обычаев делового оборота.

6.2. При неурегулировании в процессе переговоров спорных вопросов, споры разрешаются в суде в порядке, установленном действующим законодательством.

7. СРОК ДЕЙСТВИЯ ДОГОВОРА

7.1. Настоящий договор вступает в силу с момента заключения и действует бессрочно.

8. РАСТОРЖЕНИЕ ДОГОВОРА

8.1. Стороны вправе досрочно расторгнуть договор по взаимному письменному соглашению.

8.2. Получатель вправе расторгнуть договор в случае, если на момент заключения Автор не обладает авторским правом на предмет договора. При расторжении договора по указанному основанию Автор обязан возратить всю сумму вознаграждения, полученного по договору.

8.3. Автор вправе расторгнуть договор в случае повторного нарушения Получателем обязанности выплачивать вознаграждение Автору.

9. ДОПОЛНИТЕЛЬНЫЕ УСЛОВИЯ И ЗАКЛЮЧИТЕЛЬНЫЕ ПОЛОЖЕНИЯ

9.1. Дополнительные условия по настоящему договору: отсутствуют.

9.2. Во всем остальном, что не предусмотрено настоящим договором, стороны руководствуются действующим законодательством.

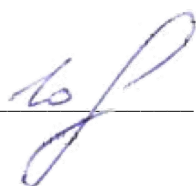
9.3. Любые изменения и дополнения к настоящему договору действительны, при условии, если они совершены в письменной форме и подписаны сторонами или надлежаще уполномоченными на то представителями сторон.

9.4. Все уведомления и сообщения должны направляться в письменной форме.

9.5. Договор составлен в двух экземплярах в формате электронного документа, из которых один находится у Автора, второй – у Получателя.

10. ПОДПИСИ СТОРОН

Получатель _____



Автор _____

Приложение 1. Исходный код Приложения

файл MainActivity.java:

```
package com.fogonrog.wildpolitics;

import android.annotation.TargetApi;
import android.app.Activity;
import android.content.Intent;
import android.net.Uri;
import android.os.Build;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.view.WindowManager;
import android.webkit.JsResult;
import android.webkit.ValueCallback;
import android.webkit.WebChromeClient;
import android.webkit.WebResourceRequest;
import android.webkit.WebSettings;
import android.webkit.WebView;
import android.webkit.WebViewClient;
import android.webkit.WebViewRenderProcess;
import android.widget.Toast;

import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    private ValueCallback<Uri[]> mFilePathCallback;
    private static final int FILECHOOSER_RESULTCODE = 1;

    private WebView webView;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,
            WindowManager.LayoutParams.FLAG_FULLSCREEN);
        getWindow().getDecorView().setSystemUiVisibility(
            View.SYSTEM_UI_FLAG_HIDE_NAVIGATION);

        setContentView(R.layout.activity_main);

        webView = findViewById(R.id.webView);

        // Включаем поддержку JavaScript
        WebSettings webSettings = webView.getSettings();
        webSettings.setJavaScriptEnabled(true);
        webSettings.setDomStorageEnabled(true); // Включить DOM storage

        // Устанавливаем пользовательский агент
        webView.getSettings().setUserAgentString(System.getProperty("http.agent"));

        webView.setBackgroundColor(getResources().getColor(R.color.dark_blue));

        // Включаем логи консоли JavaScript для отладки
        webView.setWebChromeClient(new WebChromeClient() {
            @Override
```

```

        public void onConsoleMessage(String message, int lineNumber,
String sourceID) {
            Log.d("WebView", message + " -- From line " + lineNumber +
" of " + sourceID);
        }

        @Override
        public boolean onJsAlert(WebView view, String url, String
message, JsResult result) {
            Toast.makeText(MainActivity.this, message,
Toast.LENGTH_LONG).show();
            result.confirm();
            return true;
        }

        // Поддержка открытия новых окон (для модальных окон)
        @Override
        public boolean onCreateWindow(WebView view, boolean isDialog,
boolean isUserGesture, android.os.Message resultMsg) {
            WebView newWebView = new WebView(MainActivity.this);
            webView.addView(newWebView);

            WebView.WebViewTransport transport =
(WebView.WebViewTransport) resultMsg.obj;
            transport.setWebView(newWebView);
            resultMsg.sendToTarget();
            return true;
        }

        // Обработка выбора файла
        @Override
        public boolean onShowFileChooser(WebView webView,
ValueCallback<Uri[]> filePathCallback, FileChooserParams fileChooserParams)
{
            if (mFilePathCallback != null) {
                mFilePathCallback.onReceiveValue(null);
            }
            mFilePathCallback = filePathCallback;

            // Создаем Intent для выбора файла
            Intent intent = new Intent(Intent.ACTION_GET_CONTENT);
            intent.addCategory(Intent.CATEGORY_OPENABLE);
            intent.setType("image/*"); // Устанавливаем тип данных
(картинки)

            // Открываем диалог выбора файла
            startActivityForResult(Intent.createChooser(intent,
"Выберите изображение"), FILECHOOSER_RESULTCODE);
            return true;
        }
    });

    // Устанавливаем WebViewClient для открытия ссылок внутри WebView
    webView.setWebViewClient(new MyWebViewClient());

    // Загружаем URL
    webView.loadUrl("https://wildpolitics.ru");
}

private static class MyWebViewClient extends WebViewClient {

```

```

        @TargetApi(Build.VERSION_CODES.N)
        @Override
        public boolean shouldOverrideUrlLoading(WebView view,
        WebResourceRequest request) {
            if (!(request.getUrl().toString().startsWith("tg")) &&
            !(request.getUrl().toString().startsWith("https://wiki")) &&
            !(request.getUrl().toString().endsWith("politics"))) {
                view.loadUrl(request.getUrl().toString());
                return false;
            }

            Intent intent = new Intent(Intent.ACTION_VIEW,
            Uri.parse(request.getUrl().toString()));
            view.getContext().startActivity(intent);
            return true;
        }

        // Для старых устройств
        @Override
        public boolean shouldOverrideUrlLoading(WebView view, String url) {
            view.loadUrl(url);
            return true;
        }

        @Override
        protected void onActivityResult(int requestCode, int resultCode, Intent
        data) {
            if (requestCode == FILECHOOSER_RESULTCODE) {
                if (mFilePathCallback != null) {
                    Uri[] result = null;
                    if (resultCode == Activity.RESULT_OK && data != null) {
                        result = new Uri[]{data.getData()};
                    }
                    mFilePathCallback.onReceiveValue(result);
                    mFilePathCallback = null;
                }
            } else {
                super.onActivityResult(requestCode, resultCode, data);
            }
        }

        @Override
        public void onBackPressed() {
            if (webView.canGoBack()) {
                webView.goBack();
            } else {
                super.onBackPressed();
            }
        }
    }
}

```

файл AndroidManifest.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">
    <uses-permission android:name="android.permission.INTERNET" />
    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@drawable/logo_apk"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/Theme.MyApplication"
        android:usesCleartextTraffic="true"
        tools:targetApi="31">
        <activity
            android:name=".MainActivity"
            android:theme="@style/Theme.AppCompat.NoActionBar"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER"
            />
            </intent-filter>
        </activity>
    </application>
</manifest>
```