

Мобильная медиация

Интеграция

10.07.2024

Мобильная медиация. Интеграция. Версия 2.0

Дата подготовки документа: 10.07.2024

Этот документ является составной частью технической документации Яндекс.

© 2008—2024 ООО «ЯНДЕКС». Все права защищены.

Предупреждение об исключительных правах и конфиденциальной информации

Исключительные права на все результаты интеллектуальной деятельности и приравненные к ним средства индивидуализации юридических лиц, товаров, работ, услуг и предприятий, которым предоставляется правовая охрана (интеллектуальную собственность), используемые при разработке, поддержке и эксплуатации службы Мобильная медиация, включая, но не ограничиваясь, программы для ЭВМ, базы данных, изображения, тексты, другие произведения, а также изобретения, полезные модели, товарные знаки, знаки обслуживания, коммерческие обозначения и фирменные наименования, принадлежат ООО «ЯНДЕКС» либо его лицензиарам.

Использование результатов интеллектуальной деятельности и приравненных к ним средств индивидуализации в целях, не связанных с разработкой, поддержкой и эксплуатацией службы Мобильная медиация, не допускается без получения предварительного согласия правообладателя. Настоящий документ содержит конфиденциальную информацию ООО «ЯНДЕКС». Использование конфиденциальной информации в целях, не связанных с разработкой, поддержкой и эксплуатацией службы Мобильная медиация, а равно как и разглашение таковой, не допускается. При этом под разглашением понимается любое действие или бездействие, в результате которых конфиденциальная информация в любой возможной форме (устной, письменной, иной форме, в том числе с использованием технических средств) становится известной третьим лицам без согласия обладателя такой информации либо вопреки трудовому или гражданско-правовому договору.

Отношения ООО «ЯНДЕКС» с лицами, привлекаемыми для разработки, поддержки и эксплуатации службы Мобильная медиация, регулируются законодательством Российской Федерации и заключаемыми в соответствии с ним трудовыми и/или гражданско-правовыми договорами (соглашениями). Нарушение требований об охране результатов интеллектуальной деятельности и приравненных к ним средств индивидуализации, а равно как и конфиденциальной информации, влечет за собой дисциплинарную, гражданско-правовую, административную или уголовную ответственность в соответствии с законодательством Российской Федерации.

Контактная информация

ООО «ЯНДЕКС»

<https://www.yandex.ru>

Тел.: +7 495 739 7000

Email: pr@yandex-team.ru

Главный офис: 119021, Россия, г. Москва, ул. Льва Толстого, д. 16

Содержание

Поддержка SKAdNetwork.....	4
Примечание.....	4
Подключение Mobile Ads SDK.....	4
Инициализация Mobile Ads SDK.....	6
Форматы рекламы.....	6
Баннерная реклама.....	6
Типы баннера.....	6
Подключение баннера.....	8
Пример работы с баннерной рекламой.....	9
Полноэкранная реклама.....	12
Создание InterstitialAd.....	12
Загрузка рекламы.....	12
Отображение рекламы.....	12
Пример работы с полноэкранной рекламой.....	13
Реклама с вознаграждением.....	14
Создание RewardedAd.....	15
Загрузка рекламы.....	15
Отображение рекламы.....	15
Пример работы с рекламой с вознаграждением.....	15
Нативная реклама.....	16
Загрузка и показ рекламных объявлений.....	17
Адаптеры мобильной медиации.....	21
AdMob.....	21
myTarget.....	22
Start.io.....	23
UnityAds.....	24
AppLovin.....	25
IronSource.....	25
AdColony.....	26
ChartBoost.....	27
Pangle.....	27
Tapjoy.....	28
Vungle.....	29
Mintegral.....	29

Ресурсы

Примечание:

Все вызовы Mobile Ads SDK следует выполнять с главного потока.

[Настройте медиацию](#) в Партнерском интерфейсе и в кабинете рекламной сети.

Поддержка SKAdNetwork

Примечание:

SKAdNetwork поддерживается для версии SDK 4.1.2 и выше.

Mobile Ads SDK поддерживает трекинг установок приложений с помощью фреймворка [SKAdNetwork](#). Трекинг установок работает для всех устройств, даже если доступ к IDFA отсутствует.

Чтобы включить функциональность, добавьте идентификаторы поддерживаемых рекламных сетей в файл `Info.plist` приложения.

```
<key>SKAdNetworkItems</key>
  <array>
    <dict>
      <key>SKAdNetworkIdentifier</key>
      <string>zq492l623r.skadnetwork</string>
    </dict>
  </array>
```

Для дополнительной информации ознакомьтесь с разделом [Configuring a Source App](#) документации Apple.

Примечание

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Подключение Mobile Ads SDK



Внимание:

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Примечание:

1. Для загрузки любого вида рекламы необходима версия Android 4.1 и выше.
2. Видеореклама подбирается только на устройства с версией Android 5.0 и выше.

Есть несколько способов подключения:

Подключение SDK с адаптерами

Подключить все доступные адаптеры можно автоматически с помощью общей библиотеки медиации `YandexMobileAdsMediation`.

1. Добавьте зависимость от `YandexMobileAdsMediation` в `build.gradle` файл модуля вашего приложения:

```
dependencies {
    ...
    implementation 'com.yandex.android:mobileads-mediation:5.10.0.0'
}
```

Для каждого адаптера самая свежая совместимая версия будет выбрана автоматически.

2. Добавьте поддержку Java 8 в `build.gradle` файл модуля вашего приложения:

```
android {
```

```

compileOptions {
    sourceCompatibility JavaVersion.VERSION_1_8
    targetCompatibility JavaVersion.VERSION_1_8
}
}

```

3. Добавьте код в `build.gradle` файл модуля вашего приложения:

```

// IronSource
maven {
    url 'https://android-sdk.is.com/'
}

// Pangle
maven {
    url 'https://artifact.bytedance.com/repository/pangle'
}

// Tapjoy
maven {
    url "https://sdk.tapjoy.com/"
}

// Mintegral
maven {
    url "https://dl-maven-android.mintegral.com/repository/mbridge_android_sdk_oversea"
}

```

4. Настройте разрешение на использование рекламного идентификатора.

В Yandex Mobile Ads SDK, начиная с версии 4.5.0, появилось новое разрешение `com.google.android.gms.permission.AD_ID`. Оно прописано в файле `AndroidManifest.xml` библиотеки. Поэтому указывать его дополнительно в основном манифесте приложения не нужно. Разрешение позволяет использовать рекламный идентификатор для подбора релевантной рекламы от рекламных сетей.

При необходимости вы можете удалить разрешение. Например, если какие-то политики, как Families Policy, не позволяют использовать идентификатор для подбора рекламы.

Чтобы разрешение не попало в основной манифест приложения, добавьте в файл `AndroidManifest.xml` строку:

```
<uses-permission android:name="com.google.android.gms.permission.AD_ID" tools:node="remove"/>
```

Подключение SDK

1. Если вам не нужно автоматически подключать все доступные адаптеры, добавьте зависимость от `YandexMobileAds` в `build.gradle` файл модуля вашего приложения:

```

dependencies {
    ...
    implementation 'com.yandex.android:mobileads:5.10.0'
}

```

Чтобы подключить необходимые адаптеры воспользуйтесь отдельными инструкциями по подключению.

2. Добавьте поддержку Java 8 в `build.gradle` файл модуля вашего приложения:

```

android {
    compileOptions {
        sourceCompatibility JavaVersion.VERSION_1_8
        targetCompatibility JavaVersion.VERSION_1_8
    }
}

```

3. Настройте разрешение на использование рекламного идентификатора.

В Yandex Mobile Ads SDK, начиная с версии 4.5.0, появилось новое разрешение `com.google.android.gms.permission.AD_ID`. Оно прописано в файле `AndroidManifest.xml` библиотеки. Поэтому указывать его дополнительно в основном манифесте приложения не нужно. Разрешение позволяет использовать рекламный идентификатор для подбора релевантной рекламы от рекламных сетей.

При необходимости вы можете удалить разрешение. Например, если какие-то политики, как Families Policy, не позволяют использовать идентификатор для подбора рекламы.

Чтобы разрешение не попало в основной манифест приложения, добавьте в файл `AndroidManifest.xml` строку:

```
<uses-permission android:name="com.google.android.gms.permission.AD_ID" tools:node="remove"/>
```

Инициализация Mobile Ads SDK

Перед загрузкой рекламы инициализируйте библиотеку с помощью метода `initialize()`. Инициализация ускорит загрузку объявлений.

Примечание:

Это необходимо делать во время каждого нового запуска приложения. Поэтому рекомендуется добавить код инициализации в метод `onCreate` класса `Application`.

Пример инициализации:

```
public class YandexApplication extends Application {
    private static final String YANDEX_MOBILE_ADS_TAG = "YandexMobileAds";

    @Override
    public void onCreate() {
        super.onCreate();

        MobileAds.initialize(this, new InitializationListener() {
            @Override
            public void onInitializationCompleted() {
                Log.d(YANDEX_MOBILE_ADS_TAG, "SDK initialized");
            }
        });
    }
}
```

Ознакомьтесь с [примерами использования SDK](#).

Форматы рекламы

Баннерная реклама



Внимание:

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Баннер — это настраиваемое объявление, которое занимает часть экрана и реагирует на нажатие.

Типы баннера

Sticky баннер

Особенности:

1. Используется заданная ширина баннера. Высота подбирается автоматически.
2. Ширина баннера задается с помощью метода [stickySize](#).
3. Высота баннера не должна превышать 15% высоты устройства и не должна быть меньше 50 dp.

Примеры отображения баннера:

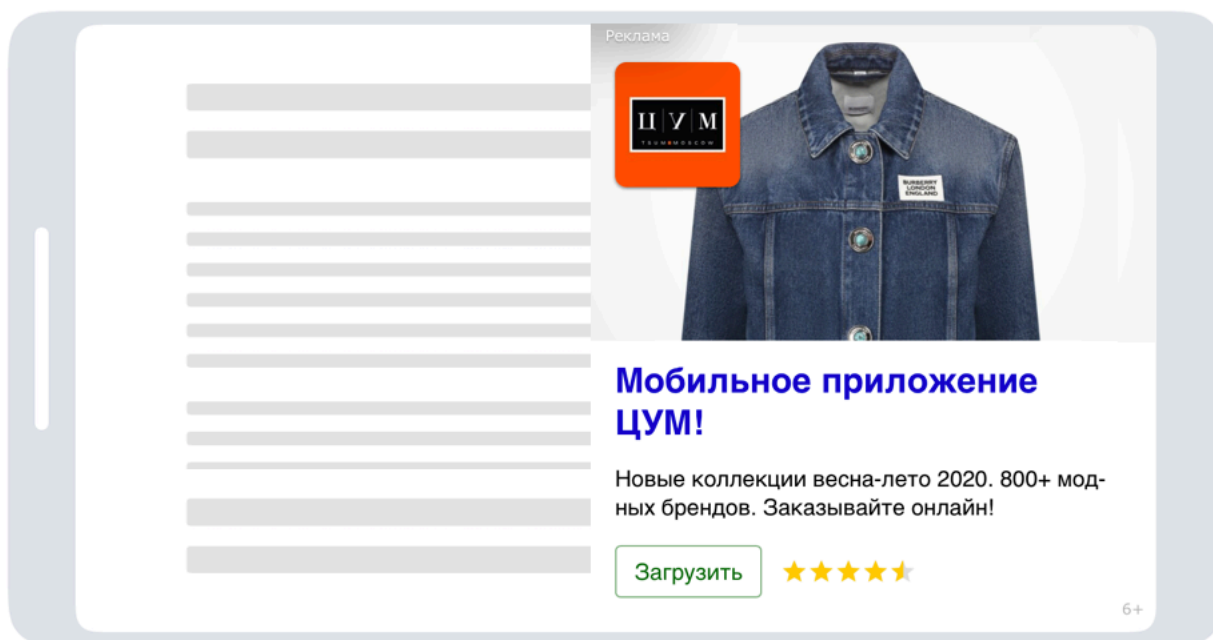


Flex баннер

Особенности:

1. Баннер заполняет весь блок, используя заданные максимальные размеры.
2. Ширина и высота баннера задается с помощью метода `flexibleSize(int width, int height)`.

Примеры отображения баннера:



Подключение баннера

Создание Adview

1. Добавьте объект класса BannerAdView в проект с помощью XML-файла или программно.

```
// Создание экземпляра mBannerAdView с помощью XML-файла.
mBannerAdView = (BannerAdView) findViewById(R.id.banner_view);

// Создание экземпляра mBannerAdView программно.
mBannerAdView = new BannerAdView(this);
```

2. Установите AdUnitId, используя метод [setAdUnitId](#).

```
mBannerAdView.setAdUnitId(<AdUnitId>)
```

AdUnitId — уникальный идентификатор рекламного места, который выдается в Партнерском интерфейсе и имеет вид: R-M-XXXXXX-Y.

- Установите размеры баннера, используя метод [setAdSize](#).

Sticky баннер

Чтобы задать ширину баннера, вызовите метод [stickySize\(int width\)](#), где `width` — максимальная ширина баннера.

```
mBannerAdView.setAdSize(AdSize.stickySize(width));
```

Flex баннер

Чтобы задать ширину и высоту баннера, вызовите метод [flexibleSize\(int width, int height\)](#).

```
mBannerAdView.setAdSize(AdSize.flexibleSize(width, height));
```

Ограничение: Требования к размерам баннера при отображении видеорекламы

Минимальный размер баннера, в котором поддерживается воспроизведение видео: 300x160 или 160x300 в dp (density-independent pixels).

- После создания и настройки объекта класса `BannerAdView` для отслеживания событий (открытие или закрытие рекламы, выход из приложения, успешная и неудачная загрузка рекламы) на объект рекламы можно установить слушатель [AdEventListener](#).

Загрузка рекламы

Примечание:

Все вызовы Mobile Ads SDK следует выполнять с главного потока.

После создания и настройки объекта класса `BannerAdView`, рекламу необходимо загрузить. Для загрузки рекламы, используйте метод [loadAd](#), принимающий в качестве параметра объект [AdRequest](#) ([Builder](#), опционально принимающий данные для таргетирования рекламы).

Пример работы с баннерной рекламой

Следующий код демонстрирует создание и настройку объекта `BannerAdView`, регистрацию слушателя, а также загрузку баннера:

```
...
<LinearLayout>
    ...
    <com.yandex.mobile.ads.banner.BannerAdView
        android:id="@+id/banner_ad_view"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>
</LinearLayout>

...
private BannerAdView mBannerAdView;
@Override
public void onCreate(Bundle savedInstanceState) {
    ...
    // Создание экземпляра mAdView.
    mBannerAdView = (BannerAdView) findViewById(R.id.banner_ad_view);
    mBannerAdView.setAdUnitId(AdUnitId);
    mBannerAdView.setAdSize(AdSize.stickySize(maxBannerWidthDp));

    // Создание объекта таргетирования рекламы.
    final AdRequest adRequest = new AdRequest.Builder().build();

    // Регистрация слушателя для отслеживания событий, происходящих в баннерной рекламе.
    mBannerAdView.setAdEventListener(new BannerAdEventListener() {
        @Override
        public void onAdLoaded() {
            ...
        }

        @Override
        public void onAdFailedToLoad(AdRequestError adRequestError) {
            ...
        }

        @Override
        public void onLeftApplication() {
            ...
        }

        @Override
        public void onReturnedToApplication() {
            ...
        }
    });
}
```

```
        ...  
    });  
    // Загрузка объявления.  
    mBannerAdView.loadAd(adRequest);  
} }
```

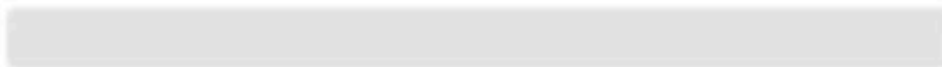
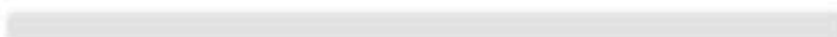
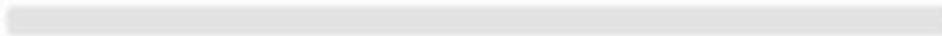
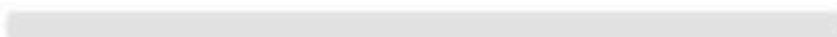
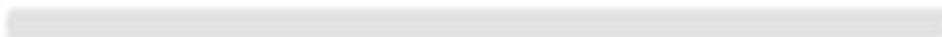
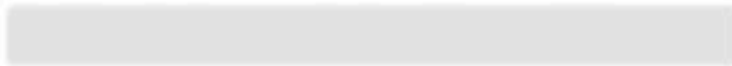
Если реклама подключена данным образом, после запуска приложения появится баннер:

10:45

◀ BannerExample



Yandex Mobile Ads



Чтобы посмотреть, как баннерная реклама будет отображаться в приложении, используйте демонстрационный AdUnitId:

```
YANDEX_AD_UNIT_ID = "demo-banner-yandex"  
ADCOLONY_AD_UNIT_ID = "demo-banner-adcolony"  
ADMOB_AD_UNIT_ID = "demo-banner-admob"  
APPROVIN_AD_UNIT_ID = "demo-banner-applovin"  
CHARTBOOST_AD_UNIT_ID = "demo-banner-chartboost"  
MINTTEGRAL_AD_UNIT_ID = "demo-banner-mintegral"  
MYTARGET_AD_UNIT_ID = "demo-banner-mytarget"  
STARTAPP_AD_UNIT_ID = "demo-banner-startapp"  
VUNGLE_AD_UNIT_ID = "demo-banner-vungle"
```

Понятия, связанные с данным

[Классы и интерфейсы для работы с баннерной рекламой](#)

Узнайте больше

[Пример рекламы](#)

Полноэкранная реклама



Внимание:

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Полноэкранная реклама (Interstitial) — это настраиваемое объявление, отображаемое на весь экран и реагирующее на нажатие.

Чтобы подключить рекламу, необходимо:

[Создать InterstitialAd](#)

[Загрузить рекламу](#)

[Отобразить рекламу](#)

Создание InterstitialAd

1. Создайте объект класса [InterstitialAd](#). Объект может быть создан только программно.

```
mInterstitialAd = new InterstitialAd(this);
```

2. Установите AdUnitId, используя метод [setAdUnitId](#).

```
mInterstitialAd.setAdUnitId(AdUnitId);
```

AdUnitId — уникальный идентификатор рекламного места, который выдается в Партнерском интерфейсе и имеет вид: R-M-XXXXXX-Y.

3. После создания и настройки объекта класса [InterstitialAd](#), для отслеживания событий (открытие или закрытие рекламы, выход из приложения, успешная и не успешная загрузка рекламы), на объект рекламы можно установить слушатель [InterstitialEventListener](#) интерфейса.

Загрузка рекламы

Примечание:

Все вызовы Mobile Ads SDK следует выполнять с главного потока.

После создания и настройки объекта класса [InterstitialAd](#), рекламу необходимо загрузить. Для загрузки рекламы, используйте метод [loadAd](#), принимающий в качестве параметра объект [AdRequest \(Builder\)](#), опционально принимающий данные для таргетирования рекламы).

Отображение рекламы

Загрузка полноэкранной рекламы происходит в фоновом потоке сразу после вызова метода [loadAd](#). Чтобы показать полноэкранную рекламу необходимо вызвать метод [show](#).

Рекомендуется предварительно проверить, что реклама действительно загружена. Для этого вызовите метод `isLoaded`.

Примечание:

Такая проверка не требуется, если метод `show` вызывается после срабатывания callback об окончании загрузки `onAdLoaded`.

Пример работы с полноэкранный рекламой

Следующий код демонстрирует создание и настройку объекта `InterstitialAd`, регистрацию слушателя, а также загрузку и отображение полноэкранный рекламы:

```
...
public class InterstitialExample extends Activity {
    ...
    private static final String AdUnitId = "YOUR_AdUnitId";
    private InterstitialAd mInterstitialAd;
    @Override
    public void onCreate(Bundle savedInstanceState) {
        ...
        // Создание экземпляра InterstitialAd.
        mInterstitialAd = new InterstitialAd(this);
        mInterstitialAd.setAdUnitId(AdUnitId);

        // Создание объекта таргетирования рекламы.
        final AdRequest adRequest = new AdRequest.Builder().build();

        // Регистрация слушателя для отслеживания событий, происходящих в рекламе.
        mInterstitialAd.setInterstitialAdEventListener(new InterstitialAdEventListener() {
            @Override
            public void onAdLoaded() {
                mInterstitialAd.show();
            }

            @Override
            public void onAdFailedToLoad(AdRequestError adRequestError) {
                ...
            }

            @Override
            public void onAdShown() {
                ...
            }

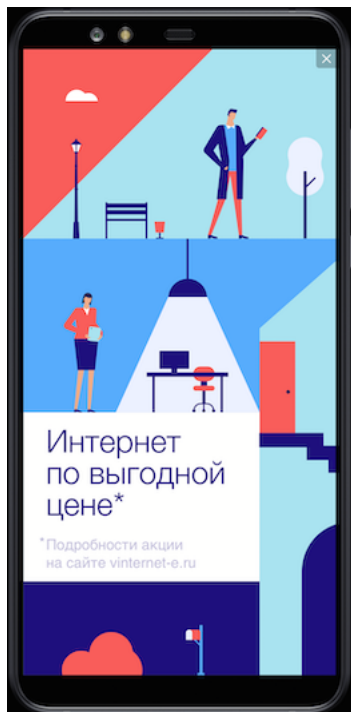
            @Override
            public void onAdDismissed() {
                ...
            }

            @Override
            public void onLeftApplication() {
                ...
            }

            @Override
            public void onReturnedToApplication() {
                ...
            }
        });

        // Загрузка объявления.
        mInterstitialAd.loadAd(adRequest);
    }
}
```

Если реклама подключена данным образом, после запуска приложения появится рекламный блок:



Чтобы посмотреть, как реклама будет отображаться в приложении, используйте демонстрационный AdUnitId:

```
YANDEX_AD_UNIT_ID = "demo-interstitial-yandex"  
ADCOLONY_AD_UNIT_ID = "demo-interstitial-adcolony"  
ADMOB_AD_UNIT_ID = "demo-interstitial-admob"  
APPROVIN_AD_UNIT_ID = "demo-interstitial-applovin"  
CHARTBOOST_AD_UNIT_ID = "demo-interstitial-chartboost"  
IRONSOURCE_AD_UNIT_ID = "demo-interstitial-ironsource"  
MINTEGRAL_AD_UNIT_ID = "demo-interstitial-mintegral"  
MYTARGET_AD_UNIT_ID = "demo-interstitial-mytargеt"  
PANGLE_AD_UNIT_ID = "demo-interstitial-pangle"  
STARTAPP_AD_UNIT_ID = "demo-interstitial-startapp"  
TAPJOY_AD_UNIT_ID = "demo-interstitial-tapjoy"  
UNITYADS_AD_UNIT_ID = "demo-interstitial-unityads"  
VUNGLE_AD_UNIT_ID = "demo-interstitial-vungle"
```

Понятия, связанные с данным

[Классы и интерфейсы для работы с полноэкранной рекламой](#)

[Узнайте больше](#)

[Пример рекламы](#)

Процесс подключения рекламы с вознаграждением



Внимание:

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Реклама с вознаграждением (Rewarded Ad) — настраиваемое объявление, отображаемое на весь экран. За просмотр такой рекламы пользователь получает вознаграждение.

Чтобы подключить рекламу:

[Создать RewardedAd](#)

[Загрузить рекламу](#)

[Отобразить рекламу](#)

Создание RewardedAd

1. Добавьте объект класса [RewardedAd](#).

```
mRewardedAd = new RewardedAd(this);
```

2. Установите AdUnitId, используя метод [setAdUnitId](#).

```
mRewardedAd.setAdUnitId(AdUnitId);
```

AdUnitId — уникальный идентификатор рекламного места, который выдается в Партнерском интерфейсе и имеет вид: R-M-XXXXXX-Y.

3. Если вы используете механизм награждения на стороне приложения («client-side reward»), реализуйте метод интерфейса [onRewarded](#). Он вызывается, когда показ засчитан и пользователь может быть награжден за просмотр рекламы. Используйте этот момент, чтобы выдать награду пользователю приложения.

После создания и настройки объекта класса [RewardedAd](#), для отслеживания событий (открытие или закрытие рекламы, выход из приложения, успешная и не успешная загрузка рекламы), на объект рекламы необходимо установить слушатель [RewardedAdEventListener](#) интерфейса.

Загрузка рекламы

Примечание:

Все вызовы Mobile Ads SDK следует выполнять с главного потока.

После создания и настройки объекта класса [RewardedAd](#), рекламу необходимо загрузить. Для загрузки рекламы, используйте метод [loadAd](#), принимающий в качестве параметра объект [AdRequest](#) ([Builder](#), опционально принимающий данные для таргетирования рекламы).

Отображение рекламы

Загрузка рекламы с вознаграждением происходит в фоновом потоке сразу после вызова метода [loadAd](#). Чтобы показать рекламу с вознаграждением необходимо вызвать метод [show](#).

Рекомендуется предварительно проверить, что реклама действительно загружена. Для этого вызовите метод [isLoading](#).

Примечание:

Такая проверка не требуется, если метод [show](#) вызывается после срабатывания callback об окончании загрузки [onAdLoaded](#).

Пример работы с рекламой с вознаграждением

Следующий код демонстрирует создание и настройку объекта [RewardedAd](#), регистрацию слушателя, а также загрузку и отображение рекламы с вознаграждением:

```
...
public class RewardedAdExample extends Activity {
    ...
    private static final String AdUnitId = "YOUR_AdUnitId";
    private RewardedAd mRewardedAd;
    @Override
    public void onCreate(Bundle savedInstanceState) {
        ...
        // Создание экземпляра RewardedAd.
        mRewardedAd = new RewardedAd(this);
        mRewardedAd.setAdUnitId(AdUnitId);

        // Создание объекта таргетирования рекламы.
        final AdRequest adRequest = new AdRequest.Builder().build();

        // Регистрация слушателя для отслеживания событий, происходящих в рекламе.
        mRewardedAd.setRewardedAdEventListener(new RewardedAdEventListener() {
            @Override
            public void onLoaded() {
                mRewardedAd.show();
            }
        });

        @Override
        public void onRewarded(final Reward reward) {
            ...
        }
    }
}
```

```

        }
        ...
        @Override
        public void onAdFailedToLoad(final AdRequestError adRequestError) {
            }
            ...
        }

        @Override
        public void onAdShown() {
            }
            ...
        }

        @Override
        public void onAdDismissed() {
            }
            ...
        }

        @Override
        public void onLeftApplication() {
            }
            ...
        }

        @Override
        public void onReturnedToApplication() {
            }
            ...
        }
    });

    // Загрузка объявления.
    mRewardedAd.loadAd(adRequest);
}
}

```

Если реклама подключена данным образом, после запуска приложения появится рекламный блок.

Чтобы посмотреть, как реклама будет отображаться в приложении, используйте демонстрационный AdUnitId:

```

YANDEX_AD_UNIT_ID = "demo-rewarded-yandex"
ADCOLONY_AD_UNIT_ID = "demo-rewarded-adcolony"
ADMOB_AD_UNIT_ID = "demo-rewarded-admob"
APPROVIN_AD_UNIT_ID = "demo-rewarded-applovin"
CHARTBOOST_AD_UNIT_ID = "demo-rewarded-chartboost"
IRONSOURCE_AD_UNIT_ID = "demo-rewarded-ironsource"
MINTTEGRAL_AD_UNIT_ID = "demo-rewarded-mintegral"
MYTARGET_AD_UNIT_ID = "demo-rewarded-mytargat"
PANGLE_AD_UNIT_ID = "demo-rewarded-pangle"
STARTAPP_AD_UNIT_ID = "demo-rewarded-startapp"
TAPJOY_AD_UNIT_ID = "demo-rewarded-tapjoy"
UNITYADS_AD_UNIT_ID = "demo-rewarded-unityads"
VUNGLE_AD_UNIT_ID = "demo-rewarded-vungle"

```

Нативная реклама



Внимание:

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Yandex Mobile Ads SDK предоставляет возможность показывать рекламу с использованием собственных визуальных компонентов.

Нативная реклама может изменяться в зависимости от функций и дизайна приложения, в котором она отображается. Оформление нативной рекламы соответствует среде, в которую она встраивается. Такая реклама выглядит органично и дополняет приложение полезной информацией.

SDK также предоставляет набор готовых настраиваемых визуальных компонентов (шаблонов), которые позволяют использовать все преимущества отрисовки нативными средствами платформы и не требуют создания собственного дизайна.

Есть два типа загрузки рекламы:

- [Загрузка одного объявления](#)
- [Загрузка нескольких объявлений](#)

Понятия, связанные с данным

[Требования к размещению рекламы и рекламных компонентов](#)

Загрузка и показ рекламных объявлений



Внимание:

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Подготовка к работе

Для корректной работы Yandex Mobile Ads SDK выполните все шаги [подключения рекламной библиотеки](#).

Загрузка рекламных объявлений

1. Создайте экземпляр класса `NativeAdLoader` для получения нативных рекламных объявлений.
2. Создайте конфигурацию запроса `nativeAdRequestConfiguration` с помощью класса [NativeAdRequestConfiguration.Builder](#). В качестве параметров запроса можно передать идентификатор рекламного блока, способ загрузки изображений, возраст, гендерные признаки и другие данные, способные улучшить качество подбора рекламы.
3. Для получения уведомлений (успешная загрузка рекламы или ошибка при загрузке рекламы), создайте экземпляр [NativeAdLoadListener](#) и установите его в качестве слушателя событий загрузчика рекламных объявлений.
4. Запустите процесс загрузки рекламы.

Если на блоке включены медиа форматы рекламы, следует запрашивать рекламу с указанием размеров контейнера.

Общий запрос за рекламой

```
final NativeAdLoader nativeAdLoader = new NativeAdLoader(this);
nativeAdLoader.setNativeAdLoadListener(new NativeAdLoadListener() {
    @Override
    public void onAdLoaded(@NonNull final NativeAd nativeAd) {
        //bind nativeAd
    }

    @Override
    public void onAdFailedToLoad(@NonNull final AdRequestError error) {
        //log error
    }
});

final NativeAdRequestConfiguration nativeAdRequestConfiguration =
    new NativeAdRequestConfiguration.Builder(AdUnitId).build();
nativeAdLoader.loadAd(nativeAdRequestConfiguration);
```

Совет:

Рекомендуется держать сильную ссылку на рекламу и ее загрузчик на всем протяжении жизни экрана, в рамках которого происходит работа с данными компонентами.

Примеры с демонстрационным AdUnit ID

Чтобы посмотреть, как реклама будет отображаться в приложении, используйте демонстрационный AdUnit ID:

```
YANDEX_AD_UNIT_ID = "demo-native-content-yandex"
ADMOB_AD_UNIT_ID = "demo-native-content-admob"
MYTARGET_AD_UNIT_ID = "demo-native-content-mytarget"
STARTAPP_AD_UNIT_ID = "demo-native-content-startapp"
```

Показ рекламных объявлений

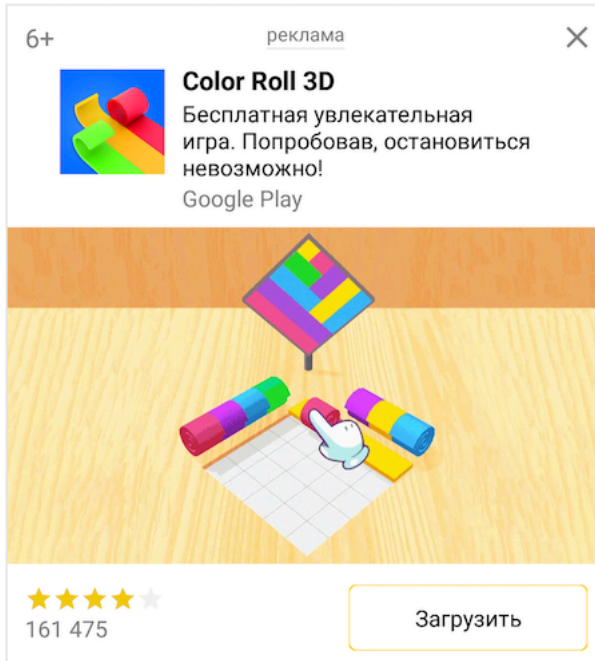
После окончания загрузки рекламы, все ее компоненты необходимо отобразить. Список доступных в объявлении компонентов можно получить из объекта рекламы [NativeAd](#).

Существует два способа настройки внешнего оформления рекламы:

1. [Оформление с помощью шаблона](#)
2. [Оформление без использования шаблона](#)

Оформление с помощью шаблона

Пример шаблона нативной рекламы



Использование стандартного шаблона оформления — самый простой способ работы с нативной рекламой, который требует всего нескольких строк кода в базовом варианте.

В шаблоне уже добавлен весь набор необходимых рекламных компонентов и настроено их расположение относительно друг друга. Шаблон работает с любым поддерживаемым типом нативной рекламы.

```
final NativeBannerView nativeBannerView = new NativeBannerView(this);  
nativeBannerView.setAd(nativeAd);
```

Шаблон нативной рекламы можно кастомизировать. Подробнее об этом в статье [Пример оформления с помощью шаблона](#).

Оформление без использования шаблона

Ручная настройка внешнего оформления нативной рекламы используется в тех случаях, когда возможностей настройки шаблона недостаточно для получения желаемого результата.

Данный способ позволяет самостоятельно сверстать макет нативной рекламы, определить расположение элементов рекламы относительно друг друга. В объявлении могут присутствовать как обязательные, так и опциональные для показа компоненты. Полный их перечень можно найти в разделе [Компоненты нативной рекламы](#).

Совет:

Рекомендуется использовать макет, который включает весь набор возможных компонентов. Как показывает практика, макеты, включающие весь набор компонентов, более кликабельные.

Пример оформления без использования шаблона

Реклама приложе

Фавиконка

Домен

Реклам
и возра
метка

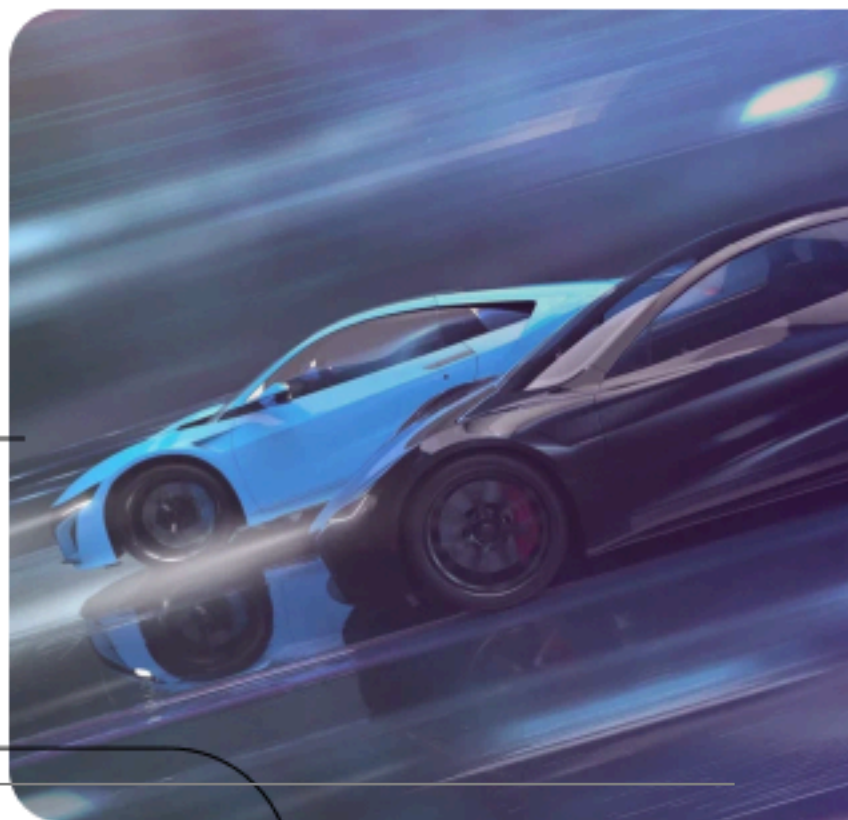


play.google.com

Реклама • 18+

Медиа

Кол-во
оценок



Для каждого компонента рекламного объявления предоставьте View через экземпляр класса [NativeAdViewBinder.Builder](#). Класс принимает в качестве аргумента контейнер [NativeAdView](#). Все компоненты рекламы должны быть определены как subview данного контейнера.

Сформированный макет рекламного объявления необходимо связать с объектом нативной рекламы [NativeAd](#).

```
final NativeAdViewBinder nativeAdViewBinder = new NativeAdViewBinder.Builder(mNativeAdView)
    .setAgeView((TextView) findViewById(R.id.age))
    .setBodyView((TextView) findViewById(R.id.body))
    .setCallToActionView((TextView) findViewById(R.id.call_to_action))
    .setDomainView((TextView) findViewById(R.id.domain))
    .setFaviconView((ImageView) findViewById(R.id.favicon))
    .setFeedbackView((TextView) findViewById(R.id.feedback))
    .setIconView((ImageView) findViewById(R.id.icon))
    .setMediaView((MediaView) findViewById(R.id.media))
    .setPriceView((TextView) findViewById(R.id.price))
    .setRatingView((MyRatingView) findViewById(R.id.rating))
    .setReviewCountView((TextView) findViewById(R.id.review_count))
    .setSponsoredView((TextView) findViewById(R.id.sponsored))
    .setTitleView((TextView) findViewById(R.id.title))
    .setWarningView((TextView) findViewById(R.id.warning))
    .build();

try {
    nativeAd.bindNativeAd(nativeAdViewBinder);
    mNativeAdView.setVisibility(View.VISIBLE);
} catch (final NativeAdException exception) {
    //log exception
}
```

Примечание: Требования к размерам mediaView при отображении видеорекламы

Минимальный размер экземпляра класса `MediaView`, в котором поддерживается воспроизведение видео: 300x160 или 160x300 в dp (density-independent pixels).

Для поддержки воспроизведения видео в шаблонах нативной рекламы рекомендуется выставить ширину для `NativeBannerView` не менее 300 dp. Корректная высота для `mediaView` будет вычислена автоматически, с учетом соотношения ширины и высоты.

Загрузка нескольких рекламных объявлений

Примечание:

Все вызовы Mobile Ads SDK следует выполнять с главного потока.

Yandex Mobile Ads SDK предоставляет возможность загрузки нескольких рекламных объявлений одним запросом (до девяти объявлений).

Для балкового запроса за рекламой воспользуйтесь экземпляром класса [NativeBulkAdLoader](#). Экземпляр класса принимает в методе `loadAds` аргумент `COUNT`, определяющий желаемое количество объявлений за один запрос.

```
final NativeBulkAdLoader nativeBulkAdLoader = new NativeBulkAdLoader(this);
nativeBulkAdLoader.setNativeBulkAdLoadListener(new NativeBulkAdLoadListener() {
    @Override
    public void onAdsLoaded(@NonNull final List<NativeAd> nativeAds) {
        for (final NativeAd nativeAd : nativeAds) {
            //bind native ad
        }
    }

    @Override
    public void onAdsFailedToLoad(@NonNull final AdRequestError error) {
        //log error
    }
});

final NativeAdRequestConfiguration nativeAdRequestConfiguration =
    new NativeAdRequestConfiguration.Builder(AdUnitId).build();
nativeBulkAdLoader.loadAds(nativeAdRequestConfiguration, COUNT);
```

Примечание:

Yandex Mobile Ads SDK не гарантирует, что будет загружено запрошенное количество объявлений. Полученный массив будет содержать от 0 до `COUNT` объектов `NativeAd`. Все полученные объекты рекламы можно показывать отдельно друг от друга, используя описанные выше способы внешнего оформления нативных объявлений.

Адаптеры мобильной медиации



Внимание:

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Мобильная медиация — это платформа автоматического подбора рекламы из нескольких рекламных сетей. Каждая рекламная сеть предложит к показу объявление, а платформа-медиатор выберет из них наиболее выгодное.

Платформа мобильной медиации поддерживает интеграцию адаптеров большинства крупных рекламных сетей, представленных ниже.

Список поддерживаемых рекламных сетей

Рекламная сеть	Версия адаптера	Версия SDK рекламной сети	Баннерная реклама (Banner Ad)	Полноэкранная реклама (Interstitial Ad)	Реклама с вознаграждением (Rewarded Ad)	Нативная реклама (Native Ad)
AdMob	22.1.0.0	22.1.0	✓	✓	✓	✓
myTarget	5.17.0.0	5.17.0	✓	✓	✓	✓
Start.io	4.11.0.1	4.11.0	✓	✓	✓	✓
UnityAds	4.7.1.2	4.7.1	✓	✓	✓	✗
AppLovin	11.10.1.1	11.10.1	✓	✓	✓	✗
IronSource	7.3.1.1.0	7.3.1.1	✗	✓	✓	✗
AdColony	4.8.0.5	4.8.0	✓	✓	✓	✗
ChartBoost	9.3.0.2	9.3.0	✓	✓	✓	✗
Pangle	5.3.0.4.0	5.3.0.4	✗	✓	✓	✗
Tapjoy	13.0.1.1	13.0.1	✗	✓	✓	✗
Vungle	6.12.1.2	6.12.1	✓	✓	✓	✗
Mintegral	16.4.71.0	16.4.71	✓	✓	✓	✓

Подключение AdMob



Внимание:

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Поддерживаемые версии библиотек и платформ:

YandexMobileAdsSDK

Минимальная поддерживаемая версия: 5.10.0.

Максимальная поддерживаемая версия: 6.0.0 (не включая).

GoogleMobileAdsSDK

Минимальная поддерживаемая версия: 22.1.0.

Максимальная поддерживаемая версия: 22.2.0 (не включая).

Поддерживаемые форматы рекламы

- [Баннерная реклама](#) (Banner)
- [Полноэкранная реклама](#) (Interstitial)
- [Реклама с вознаграждением](#) (Rewarded)
- [Нативная реклама](#) (Native)

Подключение

Подключение с использованием Gradle

1. [Настройте медиацию](#) в Партнерском интерфейсе и в кабинете рекламной сети.
2. В файл `build.gradle` добавьте следующие зависимости на уровне приложения:

```
implementation 'com.yandex.android:mobileads:5.10.0'  
implementation 'com.yandex.ads.mediation:mobileads-admob:22.1.0.0'
```

3. Добавьте свой AdMob ID в файл `AndroidManifest.xml` приложения с помощью тега `<meta-data>` с именем `com.google.android.gms.ads.APPLICATION_ID` (подробнее о том, где [найти AdMob ID](#)).

```
<manifest>  
  <application>  
    ...  
    <meta-data  
      android:name="com.google.android.gms.ads.APPLICATION_ID"  
      android:value="ca-app-pub-xxxxxxxxxxxxxxxx~yyyyyyyyyy" />  
    ...  
  </application>  
</manifest>
```

Примеры интеграции

- [Баннерная реклама](#)
- [Полноэкранная реклама](#)
- [Реклама с вознаграждением](#)
- [Нативная реклама](#)

Подключение myTarget



Внимание:

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Поддерживаемые версии библиотек и платформ:

YandexMobileAdsSDK

Минимальная поддерживаемая версия: 5.10.0.

Максимальная поддерживаемая версия: 6.0.0 (не включая).

myTargetSDK

Минимальная поддерживаемая версия: 5.17.0.

Максимальная поддерживаемая версия: 5.18.0 (не включая).

Поддерживаемые форматы рекламы

- [Баннерная реклама](#) (Banner)
- [Полноэкранная реклама](#) (Interstitial)
- [Реклама с вознаграждением](#) (Rewarded)
- [Нативная реклама](#) (Native)

Подключение

Подключение с использованием Gradle

1. [Настройте медиацию](#) в Партнерском интерфейсе и в кабинете рекламной сети.
2. В файл `build.gradle` добавьте следующие зависимости на уровне приложения:

```
implementation 'com.yandex.android:mobileads:5.10.0'  
implementation 'com.yandex.ads.mediation:mobileads-mytarget:5.17.0.0'
```

Подключение вручную

1. [Настройте медиацию](#) в Партнерском интерфейсе и в кабинете рекламной сети.
2. Загрузите версию [myTarget Android SDK 5.17.0](#) и добавьте в ваш проект.
3. Загрузите последнюю версию адаптера для myTarget (файл в формате `.aar`) из [репозитория](#).

Примеры интеграции

- [Баннерная реклама](#)
- [Полноэкранная реклама](#)
- [Реклама с вознаграждением](#)
- [Нативная реклама](#)

Подключение Start.io



Внимание:

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Поддерживаемые версии библиотек и платформ:

YandexMobileAdsSDK

Минимальная поддерживаемая версия: 5.10.0.

Максимальная поддерживаемая версия: 6.0.0 (не включая).

Start.io SDK

Минимальная поддерживаемая версия: 4.11.0.

Максимальная поддерживаемая версия: 4.12.0 (не включая).

Поддерживаемые форматы рекламы

- [Баннерная реклама](#) (Banner)
- [Полноэкранная реклама](#) (Interstitial)
- [Реклама с вознаграждением](#) (Rewarded)
- [Нативная реклама](#) (Native)

Подключение

Примечание:

Перед началом работы медиационная сеть Start.io отображает всплывающее окно с запросом согласия пользователя на персонализацию рекламы (подробнее в [документации сети Start.io](#)). Если вы хотите скрывать всплывающее окно, передайте согласие пользователя на персонализацию рекламы с помощью метода [setUserConsent](#).

Подключение с использованием Gradle

1. [Настройте медиацию](#) в Партнерском интерфейсе и в кабинете рекламной сети.
2. В файл `build.gradle` добавьте следующие зависимости на уровне приложения:

```
implementation 'com.yandex.android:mobileads:5.10.0'  
implementation 'com.yandex.ads.mediation:mobileads-startapp:4.11.0.1'
```

Подключение вручную

1. [Настройте медиацию](#) в Партнерском интерфейсе и в кабинете рекламной сети.
2. Загрузите версию [Start.io SDK 4.11.0](#) и добавьте в ваш проект.
3. Загрузите последнюю версию адаптера для Start.io (файл в формате `.aar`) из [репозитория](#).

Примеры интеграции

- [Баннерная реклама](#)
- [Полноэкранная реклама](#)
- [Реклама с вознаграждением](#)
- [Нативная реклама](#)

Подключение UnityAds



Внимание:

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Поддерживаемые версии библиотек и платформ:

YandexMobileAdsSDK

Минимальная поддерживаемая версия: 5.10.0.

Максимальная поддерживаемая версия: 6.0.0 (не включая).

UnityAdsSDK

Минимальная поддерживаемая версия: 4.7.1.

Максимальная поддерживаемая версия: 4.8.0 (не включая).

Поддерживаемые форматы рекламы

- [Баннерная реклама](#) (Banner)
- [Полноэкранная реклама](#) (Interstitial)
- [Реклама с вознаграждением](#) (Rewarded)

Подключение

Подключение с использованием Gradle

1. [Настройте медиацию](#) в Партнерском интерфейсе и в кабинете рекламной сети.
2. В файл `build.gradle` добавьте следующие зависимости на уровне приложения:

```
implementation 'com.yandex.android:mobileads:5.10.0'  
implementation 'com.yandex.ads.mediation:mobileads-unityads:4.7.1.2'
```

Подключение вручную

1. [Настройте медиацию](#) в Партнерском интерфейсе и в кабинете рекламной сети.
2. Загрузите версию [UnityAds 4.7.1](#) и добавьте в ваш проект.
3. Загрузите последнюю версию адаптера для UnityAds (файл в формате `.aar`) из [репозитория](#).

Примеры интеграции

- [Полноэкранная реклама](#)
- [Реклама с вознаграждением](#)

Подключение AppLovin

**Внимание:**

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Поддерживаемые версии библиотек и платформ:

YandexMobileAdsSDK

Минимальная поддерживаемая версия: 5.10.0.

Максимальная поддерживаемая версия: 6.0.0 (не включая).

AppLovinSDK

Минимальная поддерживаемая версия: 11.10.1.

Максимальная поддерживаемая версия: 11.11.0 (не включая).

Поддерживаемые форматы рекламы

- [Баннерная реклама](#) (Banner)
- [Полноэкранная реклама](#) (Interstitial)
- [Реклама с вознаграждением](#) (Rewarded)

Подключение**Подключение с использованием Gradle**

1. [Настройте медиацию](#) в Партнерском интерфейсе и в кабинете рекламной сети.
2. В файл `build.gradle` добавьте следующие зависимости на уровне приложения:

```
implementation 'com.yandex.android:mobileads:5.10.0'  
implementation 'com.yandex.ads.mediation:mobileads-applovin:11.10.1.1'
```

Примеры интеграции

- [Баннерная реклама](#)
- [Полноэкранная реклама](#)
- [Реклама с вознаграждением](#)

Подключение IronSource

**Внимание:**

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Поддерживаемые версии библиотек и платформ:

YandexMobileAdsSDK

Минимальная поддерживаемая версия: 5.10.0.

Максимальная поддерживаемая версия: 6.0.0 (не включая).

IronSourceSDK

Минимальная поддерживаемая версия: 7.3.1.1.

Максимальная поддерживаемая версия: 7.4.0 (не включая).

Поддерживаемые форматы рекламы

- [Полноэкранная реклама](#) (Interstitial)
- [Реклама с вознаграждением](#) (Rewarded)

Подключение

Подключение с использованием Gradle

1. [Настройте медиацию](#) в Партнерском интерфейсе и в кабинете рекламной сети.
2. В файл `build.gradle` добавьте следующие зависимости на уровне приложения:

```
implementation 'com.yandex.android:mobileads:5.10.0'  
implementation 'com.yandex.ads.mediation:mobileads-ironsource:7.3.1.1.0'
```

Подключение вручную

1. [Настройте медиацию](#) в Партнерском интерфейсе и в кабинете рекламной сети.
2. Загрузите версию [IronSource SDK 7.3.1.1](#) и добавьте в ваш проект.
3. Загрузите последнюю версию адаптера для IronSource (файл в формате `.aar`) из [репозитория](#).

Примеры интеграции

- [Полноэкранная реклама](#)
- [Реклама с вознаграждением](#)

Подключение AdColony



Внимание:

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Поддерживаемые версии библиотек и платформ:

YandexMobileAdsSDK

Минимальная поддерживаемая версия: 5.10.0.

Максимальная поддерживаемая версия: 6.0.0 (не включая).

AdColonySDK

Минимальная поддерживаемая версия: 4.8.0.

Максимальная поддерживаемая версия: 4.9.0 (не включая).

Поддерживаемые форматы рекламы

- [Баннерная реклама](#) (Banner)
- [Полноэкранная реклама](#) (Interstitial)
- [Реклама с вознаграждением](#) (Rewarded)

Подключение

Подключение с использованием Gradle

1. [Настройте медиацию](#) в Партнерском интерфейсе и в кабинете рекламной сети.
2. В файл `build.gradle` добавьте следующие зависимости на уровне приложения:

```
implementation 'com.yandex.android:mobileads:5.10.0'  
implementation 'com.yandex.ads.mediation:mobileads-adcolony:4.8.0.5'
```

Подключение вручную

1. [Настройте медиацию](#) в Партнерском интерфейсе и в кабинете рекламной сети.
2. Загрузите версию [AdColony SDK 4.8.0](#) и добавьте в ваш проект.
3. Загрузите последнюю версию адаптера для AdColony (файл в формате `.aar`) из [репозитория](#).

Примеры интеграции

- [Баннерная реклама](#)
- [Полноэкранная реклама](#)
- [Реклама с вознаграждением](#)

Подключение ChartBoost



Внимание:

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Поддерживаемые версии библиотек и платформ:

YandexMobileAdsSDK

Минимальная поддерживаемая версия: 5.10.0.

Максимальная поддерживаемая версия: 6.0.0 (не включая).

ChartBoostSDK

Минимальная поддерживаемая версия: 9.3.0.

Максимальная поддерживаемая версия: 9.4.0 (не включая).

Поддерживаемые форматы рекламы

- [Баннерная реклама](#) (Banner)
- [Полноэкранная реклама](#) (Interstitial)
- [Реклама с вознаграждением](#) (Rewarded)

Подключение

Подключение с использованием Gradle

1. [Настройте медиацию](#) в Партнерском интерфейсе и в кабинете рекламной сети.
2. В файл `build.gradle` добавьте следующие зависимости на уровне приложения:

```
implementation 'com.yandex.android:mobileads:5.10.0'  
implementation 'com.yandex.ads.mediation:mobileads-chartboost:9.3.0.2'
```

Подключение вручную

1. [Настройте медиацию](#) в Партнерском интерфейсе и в кабинете рекламной сети.
2. Загрузите версию [ChartBoost SDK 9.3.0](#) и добавьте в ваш проект.
3. Загрузите последнюю версию адаптера для ChartBoost (файл в формате `.aar`) из [репозитория](#).

Примеры интеграции

- [Баннерная реклама](#)
- [Полноэкранная реклама](#)
- [Реклама с вознаграждением](#)

Подключение Pangle



Внимание:

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Поддерживаемые версии библиотек и платформ:

YandexMobileAdsSDK

Минимальная поддерживаемая версия: 5.10.0.

Максимальная поддерживаемая версия: 6.0.0 (не включая).

PangleSDK

Минимальная поддерживаемая версия: 5.3.0.4.

Максимальная поддерживаемая версия: 5.4.0.0 (не включая).

Поддерживаемые форматы рекламы

- [Полноэкранный рекламный баннер](#) (Interstitial)
- [Реклама с вознаграждением](#) (Rewarded)

Подключение

Примечание:

Во время настройки ориентации рекламы в рекламной сети [Pangle](#), учитывайте, что заданная ориентация должна совпадать с ориентацией вашего приложения. Иначе блок не будет монетизироваться.

Подключение с использованием Gradle

1. [Настройте медиацию](#) в Партнерском интерфейсе и в кабинете рекламной сети.
2. В файл `build.gradle` добавьте следующие зависимости на уровне приложения:

```
implementation 'com.yandex.android:mobileads:5.10.0'  
implementation 'com.yandex.ads.mediation:mobileads-pangle:5.3.0.4.0'
```

Подключение вручную

1. [Настройте медиацию](#) в Партнерском интерфейсе и в кабинете рекламной сети.
2. Загрузите версию [Pangle SDK 5.3.0.4](#) и добавьте в ваш проект.
3. Загрузите последнюю версию адаптера для Pangle (файл в формате `.aar`) из [репозитория](#).

Примеры интеграции

- [Полноэкранный рекламный баннер](#)
- [Реклама с вознаграждением](#)

Подключение Tapjoy



Внимание:

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Поддерживаемые версии библиотек и платформ:

YandexMobileAdsSDK

Минимальная поддерживаемая версия: 5.10.0.

Максимальная поддерживаемая версия: 6.0.0 (не включая).

TapjoySDK

Минимальная поддерживаемая версия: 13.0.1.

Максимальная поддерживаемая версия: 13.1.0 (не включая).

Поддерживаемые форматы рекламы

- [Полноэкранный рекламный баннер](#) (Interstitial)
- [Реклама с вознаграждением](#) (Rewarded)

Подключение

Подключение с использованием Gradle

1. [Настройте медиацию](#) в Партнерском интерфейсе и в кабинете рекламной сети.
2. В файл `build.gradle` добавьте следующие зависимости на уровне приложения:

```
implementation 'com.yandex.android:mobileads:5.10.0'  
implementation 'com.yandex.ads.mediation:mobileads-tapjoy:13.0.1.1'
```

Подключение вручную

1. [Настройте медиацию](#) в Партнерском интерфейсе и в кабинете рекламной сети.

2. Загрузите версию [Tapjoy SDK](#) 13.0.1 и добавьте в ваш проект.
3. Загрузите последнюю версию адаптера для Tapjoy (файл в формате .aar) из [репозитория](#).

Примеры интеграции

- [Полноэкранная реклама](#)
- [Реклама с вознаграждением](#)

Подключение Vungle



Внимание:

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Поддерживаемые версии библиотек и платформ:

YandexMobileAdsSDK

Минимальная поддерживаемая версия: 5.10.0.

Максимальная поддерживаемая версия: 6.0.0 (не включая).

VungleSDK

Минимальная поддерживаемая версия: 6.12.1.

Максимальная поддерживаемая версия: 6.13.0 (не включая).

Поддерживаемые форматы рекламы

- [Баннерная реклама](#) (Banner)
- [Полноэкранная реклама](#) (Interstitial)
- [Реклама с вознаграждением](#) (Rewarded)

Подключение

Подключение с использованием Gradle

1. [Настройте медиацию](#) в Партнерском интерфейсе и в кабинете рекламной сети.
2. В файл build.gradle добавьте следующие зависимости на уровне приложения:

```
implementation 'com.yandex.android:mobileads:5.10.0'  
implementation 'com.yandex.ads.mediation:mobileads-vungle:6.12.1.2'
```

Подключение вручную

1. [Настройте медиацию](#) в Партнерском интерфейсе и в кабинете рекламной сети.
2. Загрузите версию [Vungle SDK](#) 6.12.1 и добавьте в ваш проект.
3. Загрузите последнюю версию адаптера для Vungle (файл в формате .aar) из [репозитория](#).

Примеры интеграции

- [Баннерная реклама](#)
- [Полноэкранная реклама](#)
- [Реклама с вознаграждением](#)

Подключение Mintegral



Внимание:

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Поддерживаемые версии библиотек и платформ:

YandexMobileAdsSDK

Минимальная поддерживаемая версия: 5.10.0.

Максимальная поддерживаемая версия: 6.0.0 (не включая).

MintegralSDK

Минимальная поддерживаемая версия: 16.4.71.

Максимальная поддерживаемая версия: 16.5.0 (не включая).

Поддерживаемые форматы рекламы

- [Баннерная реклама](#) (Banner)
- [Полноэкранная реклама](#) (Interstitial)
- [Реклама с вознаграждением](#) (Rewarded)
- [Нативная реклама](#) (Native)

Подключение

Подключение с использованием Gradle

1. [Настройте медиацию](#) в Партнерском интерфейсе и в кабинете рекламной сети.
2. В файл `build.gradle` добавьте следующие зависимости на уровне приложения:

```
implementation 'com.yandex.android:mobileads:5.10.0'  
implementation 'com.yandex.ads.mediation:mobileads-mintegral:16.4.71.0'
```

3. При наличии проблем с загрузкой библиотеки добавьте код в файл `build.gradle` в соответствии с [документацией Mintegral](#):

```
//Non-listed GP market applications# Android X Version  
maven {  
    url "https://dl-maven-android.mintegral.com/repository/mbridge_android_sdk_china"  
}  
  
//Launch GP market application# Android X Version  
maven {  
    url "https://dl-maven-android.mintegral.com/repository/mbridge_android_sdk_oversea"  
}  
  
//Non-listed GP market applications, not the Android X version  
maven {  
    url "https://dl-maven-android.mintegral.com/repository/mbridge_android_sdk_support/"  
}
```

Подключение вручную

1. [Настройте медиацию](#) в Партнерском интерфейсе и в кабинете рекламной сети.
2. Загрузите версию [Mintegral SDK 16.4.71](#) и добавьте в ваш проект.
3. Загрузите последнюю версию адаптера для Mintegral (файл в формате `.aar`) из [репозитория](#).

Примеры интеграции

- [Баннерная реклама](#)
- [Полноэкранная реклама](#)
- [Реклама с вознаграждением](#)
- [Нативная реклама](#)

Подключение BigoAds



Внимание:

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Поддерживаемые версии библиотек и платформ:

YandexMobileAdsSDK

Минимальная поддерживаемая версия: 5.10.0.

Максимальная поддерживаемая версия: 6.0.0 (не включая).

BigoAdsSDK

Минимальная поддерживаемая версия: 2.9.0.

Максимальная поддерживаемая версия: 2.10.0 (не включая).

Поддерживаемые форматы рекламы

- [Баннерная реклама](#) (Banner)
- [Полноэкранная реклама](#) (Interstitial)
- [Реклама с вознаграждением](#) (Rewarded)

Подключение

Подключение с использованием Gradle

1. [Настройте медиацию](#) в Партнерском интерфейсе и в кабинете рекламной сети.
2. В файл `build.gradle` добавьте следующие зависимости на уровне приложения:

```
implementation 'com.yandex.android:mobileads:5.10.0'  
implementation 'com.yandex.ads.mediation:mobileads-bigoads:2.9.0.2'
```

Подключение вручную

1. [Настройте медиацию](#) в Партнерском интерфейсе и в кабинете рекламной сети.
2. Загрузите версию [BigoAds SDK 2.9.0](#) и добавьте в ваш проект.
3. Загрузите последнюю версию адаптера для BigoAds (файл в формате `.aar`) из [репозитория](#).