Перечень эталонных значений конфигурационных параметров приложения (публикации в App Store и Play Store)

Оглавление

[1. Перечень значений 2](#_Toc92702927)

[1.1. Тестовое окружение 2](#_Toc92702928)

[1.2. Рабочее окружение 2](#_Toc92702929)

[1.3. Настройка подписи 2](#_Toc92702930)

[1.3.1. IOS 2](#_Toc92702931)

[1.3.2. Android 3](#_Toc92702932)

[1.4. Настройка сервиса обмена сообщениями Firebase Cloud 3](#_Toc92702933)

[1.4.1. Создание ключа APN 4](#_Toc92702934)

[1.4.2. Создание и настройка проекта Firebase Cloud 5](#_Toc92702935)

# Перечень значений

Настройка конфигурационных значений представляет следующий процесс:

* Получение исходного кода приложения;
* Редактирование значений конфигурации:
  + относительный путь: lib/constants/urls.dart;
* Редактирование сертификат файлов:
  + относительный путь: assets/certs;
* Редактирование значений firebase конфигурации:
  + относительный путь: lib/providers/firebase\_options.dart;
* Сборка исходных кодов приложения для публикации App Store / Google Store;

# Тестовое окружение

|  |  |  |
| --- | --- | --- |
| **Ключ** | **Значение** | **Описание** |
| natsWssUrl | wss://cloud.jetrabbits.com:9191 | Адрес сервера NATS |
| natsCluster | test-cluster | Наименование NATS |
| natsCertificatePath | assets/certs/cloud.jetrabbits.com.pem | Головной сертификат |
| apiUrl | https://test.ink.vpool.qsoft.ru/api/v1 | Адрес REST API |
| apiCertificatePath | assets/certs/test.ink.vpool.qsoft.ru.pem | API сертификат |

# Рабочее окружение

|  |  |  |
| --- | --- | --- |
| **Ключ** | **Значение** | **Описание** |
| natsWssUrl | wss://messenger.irkutskoil.ru | Адрес сервера NATS |
| natsCluster | test-cluster | Наименование NATS |
| natsCertificatePath | assets/certs/\_.irkutskoil.ru.pem | Головной сертификат |
| apiUrl | https://portal.irkutskoil.ru/api/v1 | Адрес REST API |
| apiCertificatePath | assets/certs/\_.irkutskoil.ru.pem | API сертификат |

# Настройка подписи

Настройка подписи представляет следующий процесс:

* Получение исходного кода приложения;
* Редактирование сценария сборки (относительный путь: android/app/build.gradle);
* Указание параметров подписи и переменных окружения;
* Сборка исходных кодов приложения для публикации App Store / Google Store;

# IOS

Переменные окружения

|  |  |  |
| --- | --- | --- |
| **Ключ** | **Значение** | **Описание** |
| DEVELOPMENT\_TEAM | Строка без пробелов | Apple Team ID:  Xcode -> Signing &… -> Team |
| PRODUCT\_BUNDLE\_IDENTIFIER | ru.irkutskoil.portal | Bundle ID:  Наименование пакета приложения |

Пример редактирования сценария сборки (относительный путь: ios/Runner.xcodeproj/project.pbxproj)

|  |
| --- |
| ...  DEVELOPMENT\_TEAM = 3HN8L2J92Q; ... |

# Android

Переменные окружения

|  |  |  |
| --- | --- | --- |
| **Ключ** | **Значение** | **Описание** |
| KEY\_ALIAS | Строка без пробелов | ALIAS подписи |
| KEY\_PASSWORD | Строка без пробелов | пароль подписи |
| KEY\_STORE | Путь до файла | путь до файла-хранилища подписи |
| STORE\_PASSWORD | Строка без пробелов | пароль хранилища подписи |

Пример редактирования сценария сборки (относительный путь: android/app/build.gradle)

|  |
| --- |
| ...  android {  ...  signingConfigs {  release {  keyAlias System.env.KEY\_ALIAS  keyPassword System.env.KEY\_PASSWORD  storePassword System.env.STORE\_PASSWORD  try {  storeFile file(System.env.KEY\_STORE)   } catch (Exception e) {  println("please specify keystore: ${e}")  }  }  debug {  }  }   buildTypes {  release {signingConfig signingConfigs.release  minifyEnabled true  proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-rules.pro'  }   debug {  signingConfig signingConfigs.debug  minifyEnabled true  proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-rules.pro'  }  }  ... } |

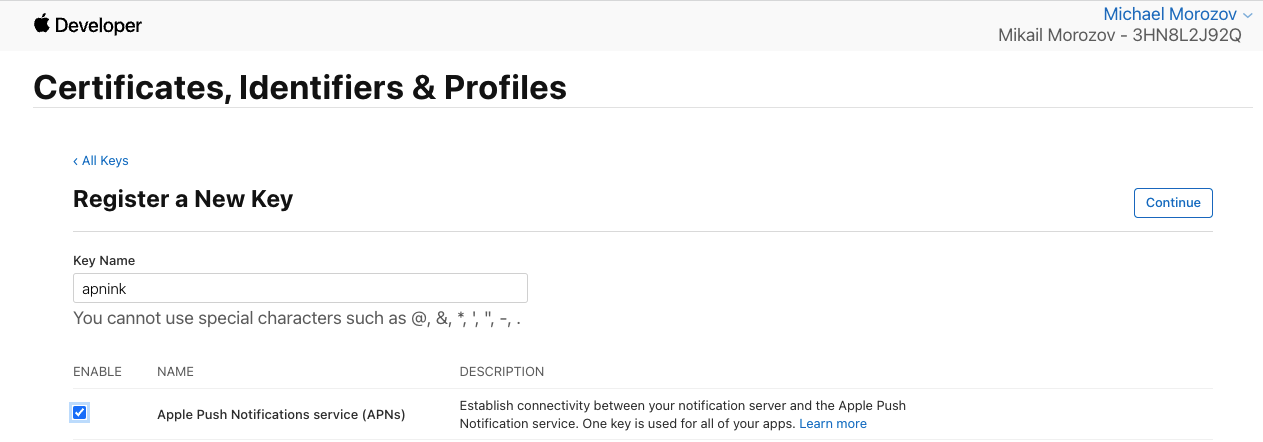
# Настройка сервиса обмена сообщениями Firebase Cloud

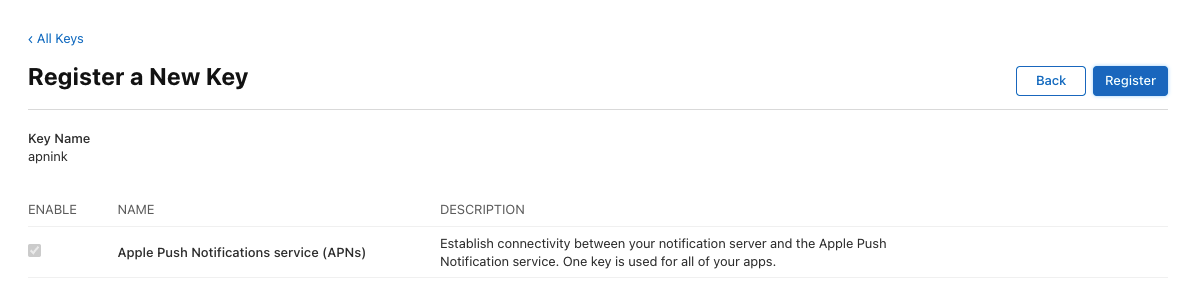
Настройка нотификации представляет следующий процесс:

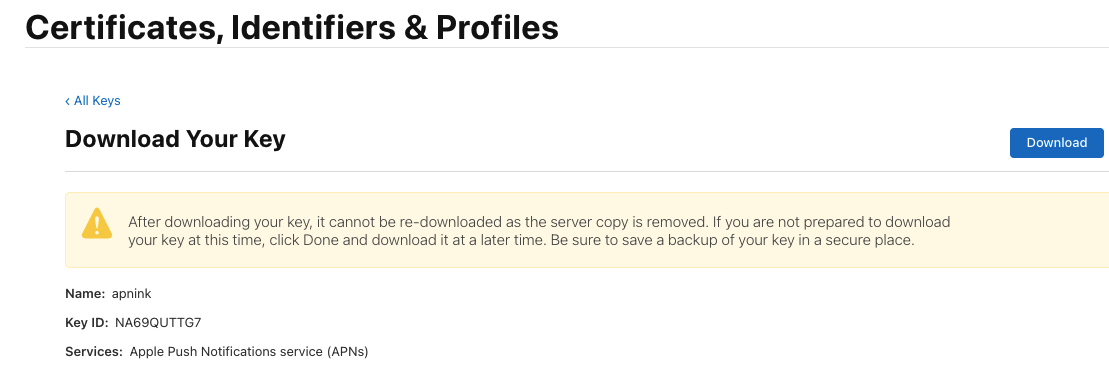
* Получение исходного кода приложения;
* Пре-реквизиты (активация рабочей среды приложения):
  + Регистрация аккаунта разработчика [Google Play console](https://play.google.com/console/u/0/developers/app/pre-registration?pli=1);
  + Регистрация аккаунта разработчика [Apple Developer Program](https://developer.apple.com/programs/enroll/);
* Создание ключа [Apple Push Notifications service](https://developer.apple.com/library/archive/documentation/NetworkingInternet/Conceptual/RemoteNotificationsPG/APNSOverview.html#//apple_ref/doc/uid/TP40008194-CH8-SW1);
* Создание и настройка проекта [Firebase Cloud](https://firebase.google.com/);
* Указание параметров нотификации на уровне исходных кодов приложения;
* Указание параметров нотификации на уровне исходных кодов сервиса NATS;
* Сборка исходных кодов приложения для публикации App Store / Google Store;

# Создание ключа APN

Активация сервиса обмена сообщениями Firebase Cloud для IOS устройств включает интеграцию [FCM via APNs](https://firebase.flutter.dev/docs/messaging/apple-integration/). Поставщиком сообщений выступает Apple Push Notifications service. Для активации сервиса необходимо создать ключ. **Результат**: скачать ключ и скопировать значение: Key ID.







# Создание и настройка проекта Firebase Cloud

Шаг представляет следующий процесс:

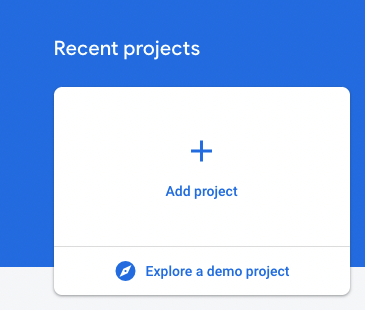
1. Создание аккаунта Firebase Cloud;
2. Создание проекта Firebase Cloud;
3. Настройка проекта Firebase Cloud для платформ IOS и Android;

**Создание аккаунта Firebase Cloud**

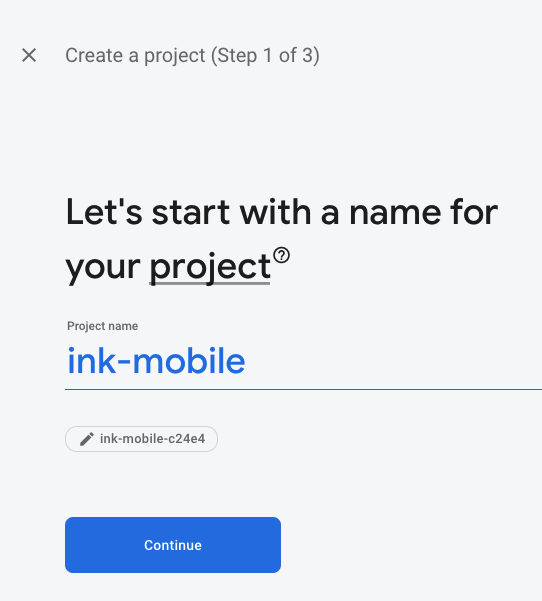
1. Перейти по ссылке: <https://firebase.google.com>;
2. Зарегистрировать аккаунт Firebase Cloud, используя google аккаунт разработчика;
3. Перейти в консоль аккаунта (страница проектов): <https://console.firebase.google.com>;

**Создание проекта Firebase Cloud**

1. Нажать кнопку Add Project (Добавить проект);



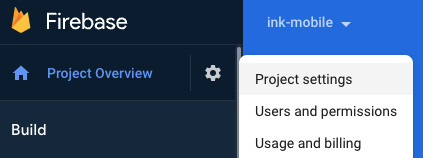
1. Ввести наименование проекта:



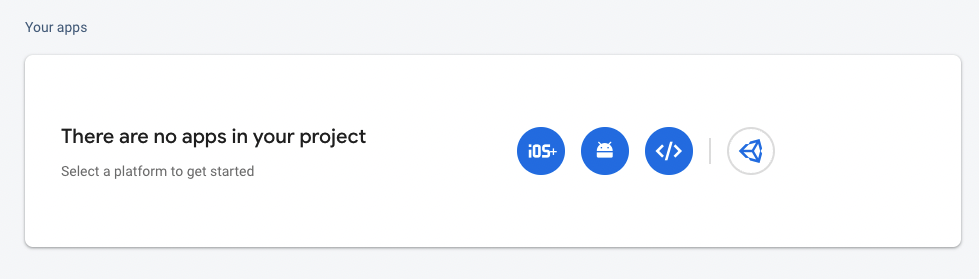
1. Выключить опцию **Enable Google Analytics for this project**;
2. Нажать кнопку **Create Project** (Создать проект) и перейти в консоль проекта;

**Настройка проекта Firebase Cloud для платформ IOS и Android**

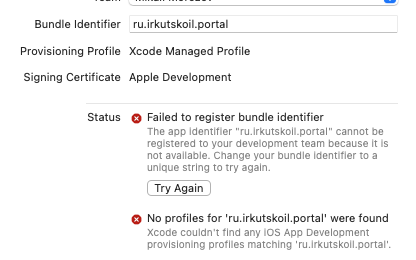
1. Выбрать пункт меню Project settings (Настройки проекта):



1. Перейти на вкладку General (Основное):

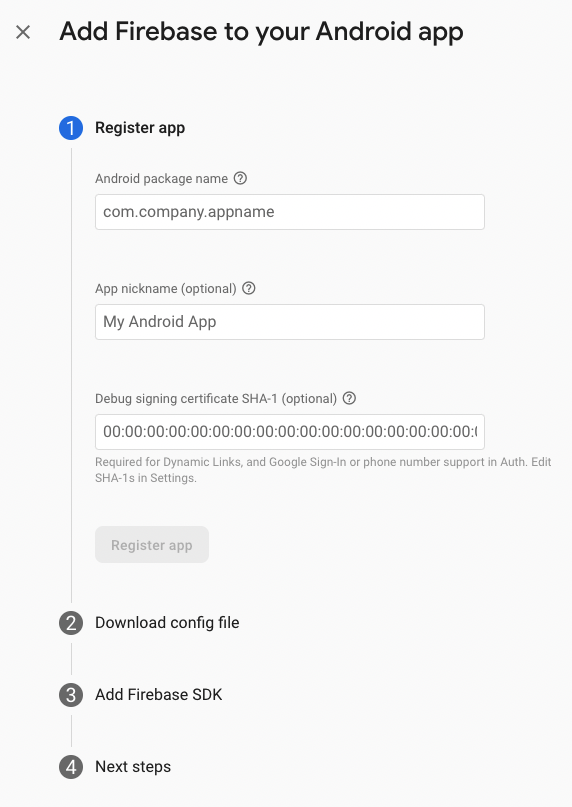


1. Определить наименование пакета для обоих платформ: IOS и Android
   * Пример наименования: ru.irkutskoil.portal
   * Среда Xcode подтверждает создание provisioning profile по введённому bundle ID
   * Рекомендуется использовать оригинальное наименование пакета, исключив служебные сочетания символов, такие как: example, test, debug, release;
   * Ниже представлен пример отрицательного запроса с причиной отказа в наименовании



1. Добавить конфигурацию Android, заполнив Android package name
   * Наименование пакета берётся из Google Play console аккаунта разработчика
   * Наименование должно соответствовать значению package из исходного кода приложения, относительный путь файла:

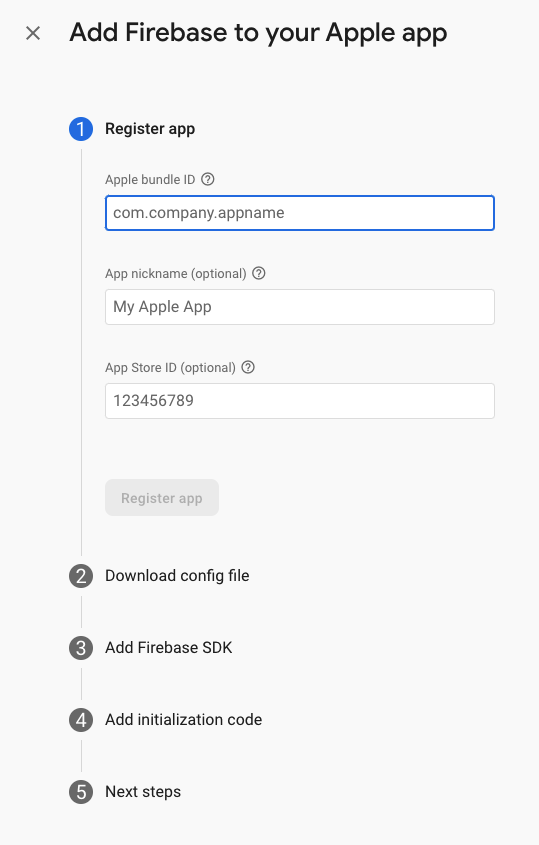
**android/app/src/main/AndroidManifest.xml**

****

* + Скачать файл google-services.json
  + Заполнить секцию значений **android** в исходном коде приложения на основе значений из файла google-services.json, относительный путь: lib/providers/firebase\_options.dart:43;

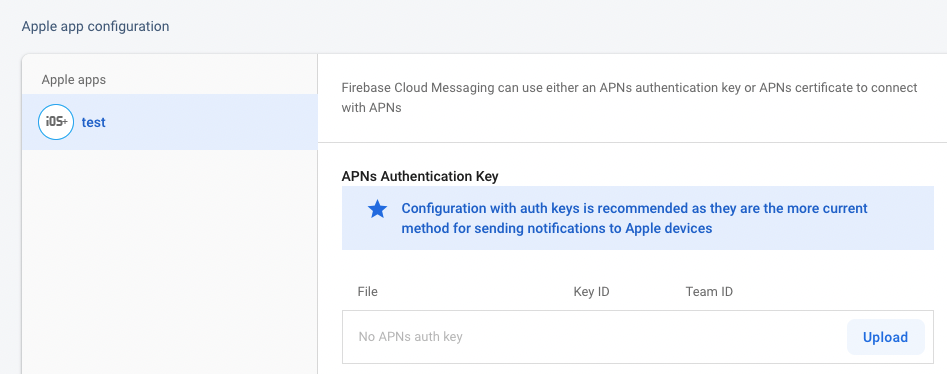
1. Добавить конфигурацию IOS, заполнив Apple Bundle ID
   * Наименование пакета берётся из Apple Developer program аккаунта разработчика
   * Наименование должно соответствовать значению PRODUCT\_BUNDLE\_IDENTIFIER из исходного кода приложения, относительный путь файла:

**ios/Runner.xcodeproj/project.pbxproj**

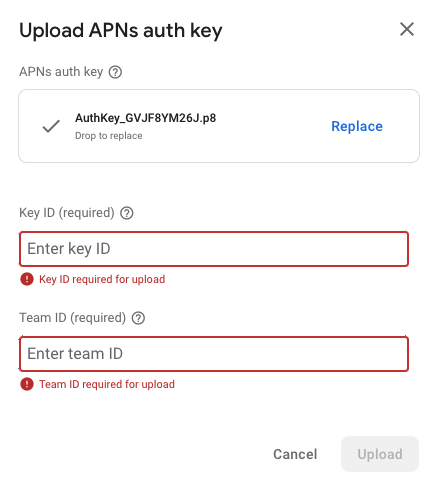
****

* + Скачать файл GoogleService-Info.plist
  + Заполнить секцию значений **ios** в исходном коде приложения на основе значений из файла GoogleService-Info.plist, относительный путь: lib/providers/firebase\_options.dart:51;
  + Сохранить файл в исходный код приложения, относительный путь **ios/Runner/GoogleService-Info.plist**

1. Перейти на вкладку Cloud Messaging и настроить интеграцию APNs:



* + Нажать кнопку Upload напротив секции iOS+
  + Загрузить файл из шага **1.4.1**
  + Заполнить поля формы "Upload APNs auth key": Key ID, Team ID



* + Нажать кнопку Upload для подтверждения

1. Перейти на вкладку Service Accounts (Сервисные учётные записи):
   * Нажать кнопку Generate new private key (Сгенерировать закрытую пару ключей)
   * Скачать JSON файл конфигурации
   * Сохранить файл в исходный код сервиса NATS, относительный путь **credentials/firebase.json**

Необходимо обновить сервис NATS после изменения конфигурации firebase

См. Руководство администратора ИНК (шаг 4.2)