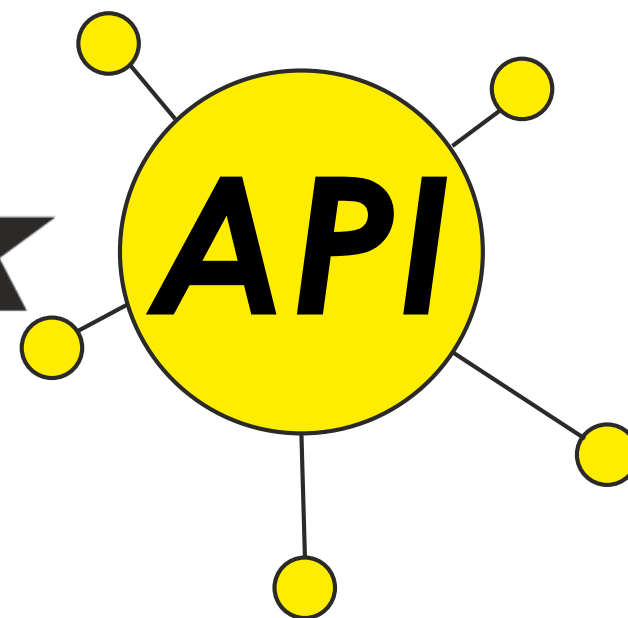
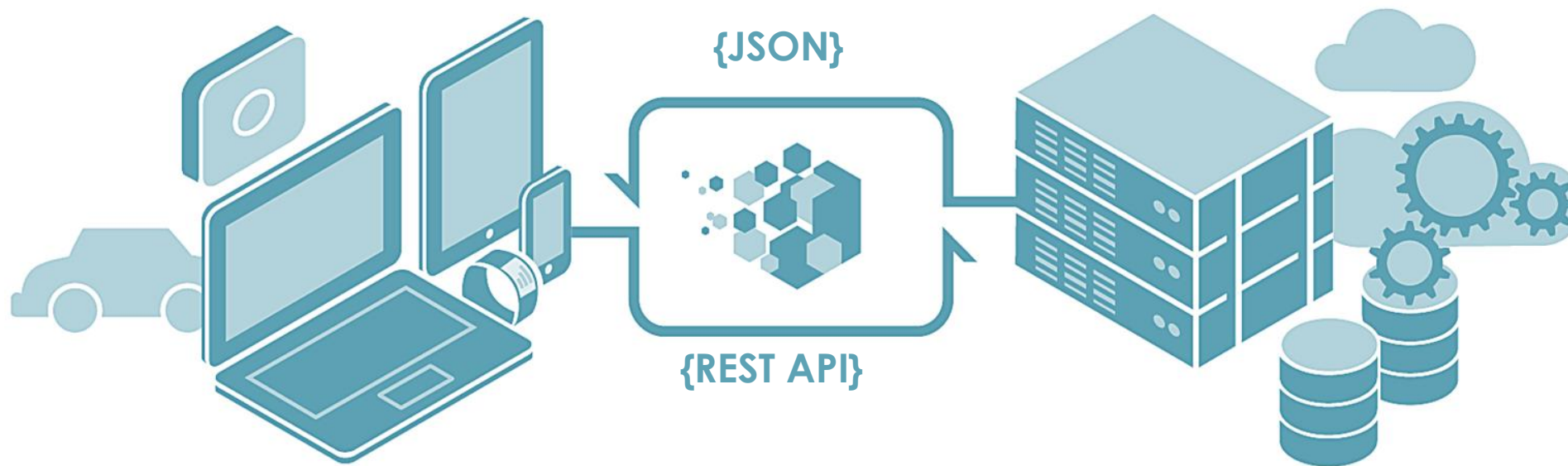




Priorbank










Архитектурный стиль сервисов: **RESTful API**


Формат передаваемых данных: **JSON**


Среда документации сервисов: **Swagger**


Протокол аутентификации: **OAuth 2.0**

  | Магазин API


 API Все ▾  

 ПРИЛОЖЕНИЯ


 ФОРУМ

 АНАЛИТИКА ▾


API




Authorize
v2
Priorbank JSC
★★★★★



Currency
v2
Priorbank JSC
★★★★★



ReferenceData
v1
Priorbank JSC
★★★★★



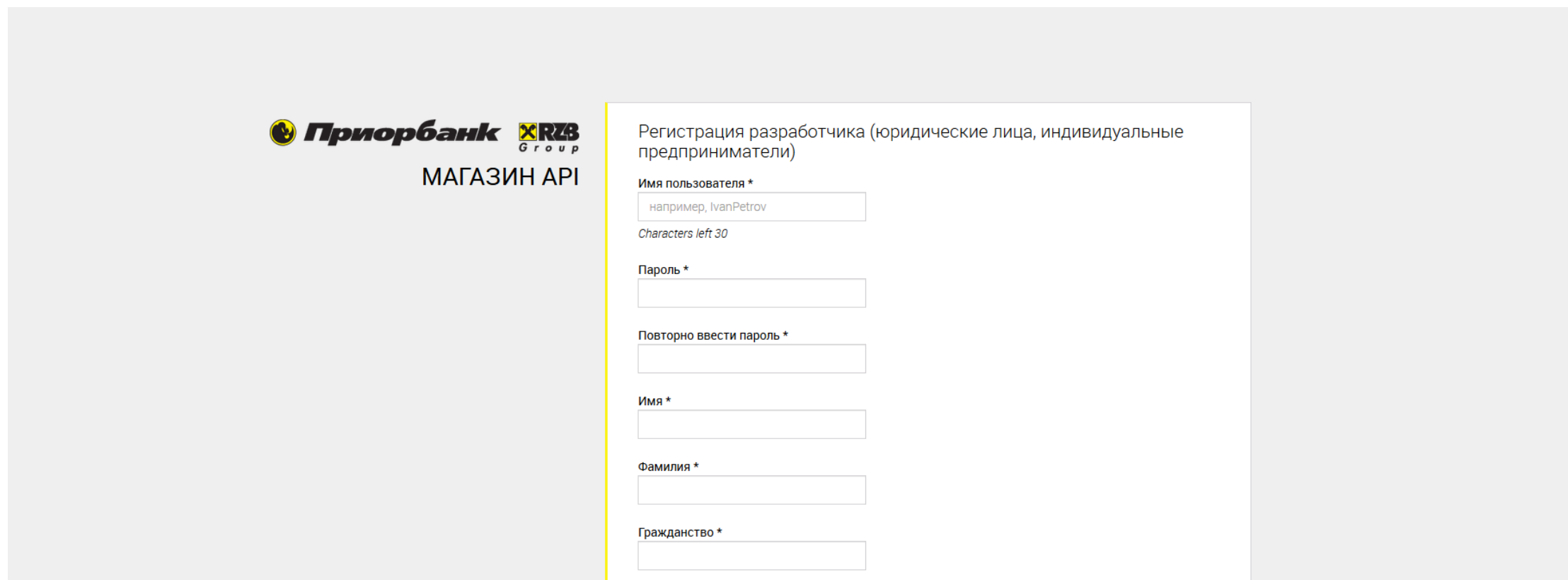
ServicePoint
v1
Priorbank JSC
★★★★★


Для получения доступа к «закрытым» API банка, вам необходимо пройти несколько несложных шагов:

1) Зарегистрироваться на портале API Приорбанка, по ссылке:

<https://api.priorbank.by/store/site/pages/sign-up.jag> (регистрация подтверждается на стороне банка).

Доступ к portalу предоставляется только Юридическим лицам и индивидуальным предпринимателям!



 МАГАЗИН API

Регистрация разработчика (юридические лица, индивидуальные предприниматели)

Имя пользователя *

Characters left 30

Пароль *

Повторно ввести пароль *

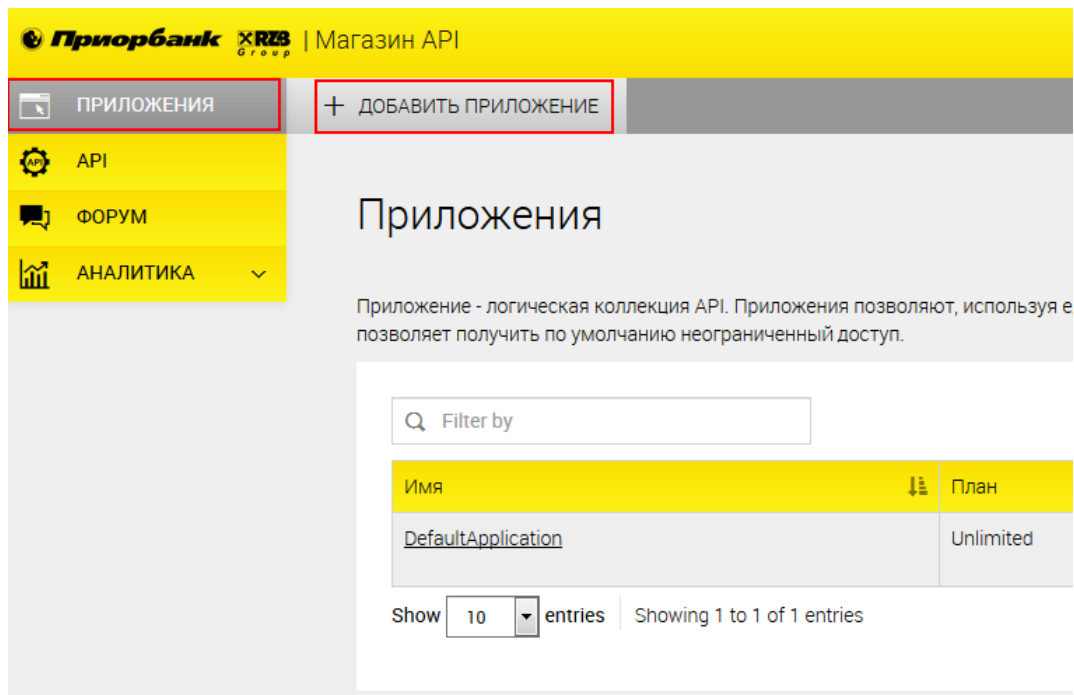
Имя *

Фамилия *

Гражданство *

2) После успешного подтверждения регистрации со стороны банка, необходимо войти на портал API Приорбанк и создать приложение. Для этого:

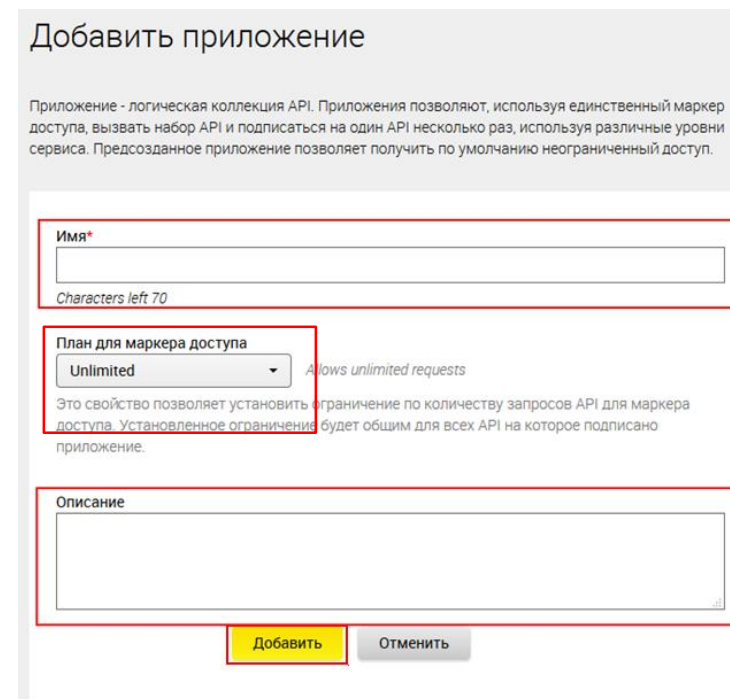
- перейти в раздел **«Приложения»**;
- нажать кнопку **«Добавить приложение»**;
- ввести **Имя** приложения (вводится латинскими буквами, без пробелов);
- выбрать **План для маркера доступа** из списка;
- добавить **Описание** приложения (описание функционала приложения, на русском языке);
- нажать кнопку **«Добавить»**.



Приложение - логическая коллекция API. Приложения позволяют, используя единственный маркер доступа, вызвать набор API и подписаться на один API несколько раз, используя различные уровни сервиса. Предсозданное приложение позволяет получить по умолчанию неограниченный доступ.

Имя	План
DefaultApplication	Unlimited

Show 10 entries Showing 1 to 1 of 1 entries



Добавить приложение

Приложение - логическая коллекция API. Приложения позволяют, используя единственный маркер доступа, вызвать набор API и подписаться на один API несколько раз, используя различные уровни сервиса. Предсозданное приложение позволяет получить по умолчанию неограниченный доступ.

Имя*

Characters left 70

План для маркера доступа

Unlimited Allows unlimited requests

Это свойство позволяет установить ограничение по количеству запросов API для маркера доступа. Установленное ограничение будет общим для всех API на которое подписано приложение.

Описание

Добавить Отменить

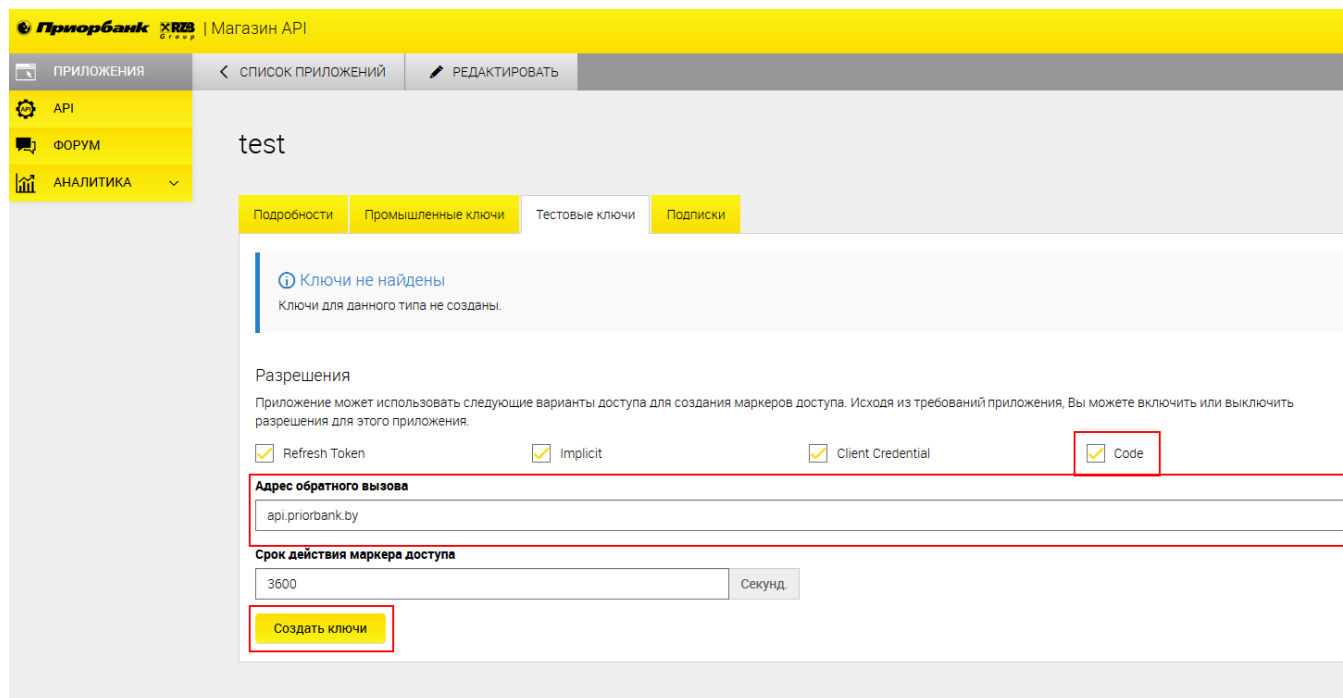
Интеграция с API для тестирования

Настройка добавленного приложения

3) Настроить приложение и создать тестовые ключи. Для этого:

- перейти во вкладку **«Тестовые ключи»**;
- ввести **Адрес обратного вызова** (URL, который будет вызван для передачи параметра **code** после авторизации Пользователя);
- установить галку на параметр **«Code»** (станет активным, после ввода адреса обратного вызова);
- нажать на кнопку **«Создать ключи»**.

После этого будут созданы ключи приложения, а также его маркер доступа (этот маркер доступа не подходит для вызова закрытых API, для получения маркера доступа необходимо использовать сервис **Authorize - v2**).



Приорбанк | Магазин API

ПРИЛОЖЕНИЯ < СПИСОК ПРИЛОЖЕНИЙ РЕДАКТИРОВАТЬ

API
ФОРУМ
АНАЛИТИКА

test

Подробности Промышленные ключи Тестовые ключи Подписки

Ключи не найдены
Ключи для данного типа не созданы.

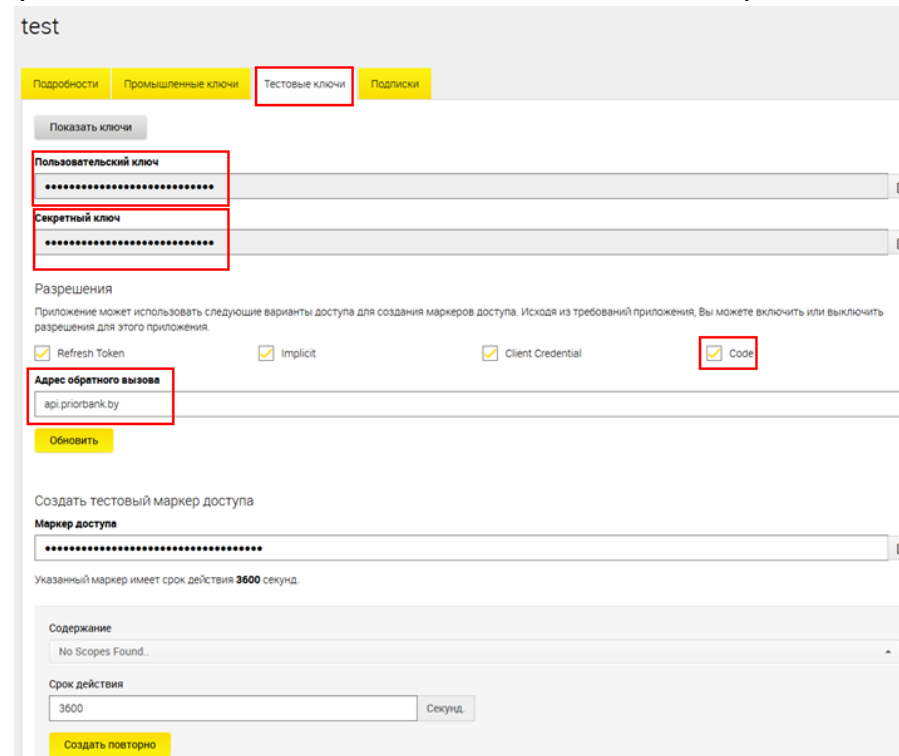
Разрешения
Приложение может использовать следующие варианты доступа для создания маркеров доступа. Исходя из требований приложения, Вы можете включить или выключить разрешения для этого приложения.

Refresh Token Implicit Client Credential Code

Адрес обратного вызова
api.priorbank.by

Срок действия маркера доступа
3600 Секунд

Создать ключи



test

Подробности Промышленные ключи Тестовые ключи Подписки

Показать ключи

Пользовательский ключ
.....

Секретный ключ
.....

Разрешения
Приложение может использовать следующие варианты доступа для создания маркеров доступа. Исходя из требований приложения, Вы можете включить или выключить разрешения для этого приложения.

Refresh Token Implicit Client Credential Code

Адрес обратного вызова
api.priorbank.by

Обновить

Создать тестовый маркер доступа

Маркер доступа
.....

Указанный маркер имеет срок действия 3600 секунд.

Содержание
No Scores Found.

Срок действия
3600 Секунд

Создать повторно

- 4) После создания ключей можно переходить к процедуре аутентификации и получения токенов доступа, для этого необходимо:
- вызвать метод **/oauth2/authentication** сервиса **Authorize - v2** передав в него следующие параметры:
 - **clientId** – в данном случае это **Пользовательский ключ** (Customer Key), получаемый при создании ключей в добавленном приложении (слайд №6).
 - **scope** – список прав доступа (указывается через пробел) необходимых приложению для доступа к тому или иному сервису. Доступно в описании сервисов API **Account** и **Customer** (вкладка «Консоль API»), параметр **Required Scopes**).
 - **redirectUri** – адрес, который будет вызван для передачи параметра code после успешной аутентификации. Соответствует параметру **Адрес обратного вызова** (Callback URL) из описания приложения.
 - **codeChallenge** – Последовательность символов полученная преобразованием параметра codeVerifier. CodeVerifier - случайная числовая последовательность от 43 до 128 символов, генерируемая отправителем запроса. Для получения параметра codeChallenge необходимо выполнить следующие преобразования
 $codeChallenge = \text{BASE64URL-ENCODE}(\text{SHA256}(\text{ASCII}(\text{codeVerifier})))$.
<https://www.rfc-editor.org/rfc/rfc7636.txt>
 - **codeChallengeMethod** – Код метода преобразования параметра codeVerifier должен быть установлен на **S256**.

Интеграция с API для тестирования

Аутентификация и получение токенов доступа

- пройти процедуру аутентификации, путем ввода логина и пароля конечного пользователя от системы Интернет-банк 2.0 (в данном случае **логин и пароль** тестового пользователя, переданного вам для тестирования) в окне, появившемся после вызова метода **/oauth2/authentication** сервиса **Authorize - v2**;
- После успешной аутентификации и согласия на предоставления доступа посредством нажатия на кнопку **Разрешить** будет вызван URL, который будет состоять, в том числе, из:
 - **Адреса обратного вызова**, введенного при настройке добавленного приложения (слайд №5);
 - параметра **code**, необходимого для вызова метода **/oauth2/authCode/token** сервиса **Authorize - v2** для получения токена доступа.


Например:

```
oauth2/api.priorbank.by?code=8abce0c9-9850-3fd6-93a7-dd523125c846
```

Вход в систему

InternetBank user

.....

Я не робот 
reCAPTCHA
Конфиденциальность · Условия использования

Войти

Авторизация

Приложение (тестовая версия), разработанное запрашивает права доступа на получение следующей информации:

- Сведения о текущих (расчетных) счетах, в том числе о номере и других реквизитах счетов, о размере средств, находящихся на счетах, об операциях по счетам.

Разрешить

Запретить

- Далее следует вызвать метод **/oauth2/authCode/token** сервиса **Authorize - v2** в котором следует передать следующие параметры:
 - **redirectUri** – адрес обратного вызова, который был внесен при настройке добавленного приложения (слайд №5);
 - **code** – код доступа, который был сформирован после успешной аутентификации пользователя и передан в ПО Разработчика по средствам вызова URL (слайд №8)
 - **clientId** – в данном случае это **Пользовательский ключ** (Customer Key), получаемый при создании ключей в добавленном приложении (слайд №6);
 - **clientSecret** – **Секретный ключ** (Customer Secret), получаемый при создании ключей в добавленном приложении (слайд №6);
 - **codeVerifier** – случайная последовательность символов от 43 до 128 символов, которая использовалась для генерации параметра **codeChallenge** в методе **/oauth2/authentication**.

Если все переданные параметры корректны, должен прийти ответ содержащий:

- **accessToken** – токен доступа с запрошенными правами;
- **refreshToken** – токен для обновления **accessToken**;
- **scope** – список прав доступа, необходимых приложению;
- **tokenType** – тип выданного токена. Всегда принимает значение «Bearer»;
- **expiresIn** – время жизни токена в секундах.

Время «жизни» **accessToken** (токен доступа) – 60 минут;

Время «жизни» **refreshToken** (токен для обновления токена доступа) – 23 часа 30 минут;

Существует отдельный метод **/oauth2/refreshToken/token** сервиса **Authorize - v2** для обновления **access-токена**.

При обращении к данному методу следует передать следующие параметры:

- **clientId** – **Пользовательский ключ** (Customer Key), получаемый при создании ключей в добавленном приложении (слайд №6);
- **clientSecret** – **Секретный ключ** (Customer Secret), получаемый при создании ключей в добавленном приложении (слайд №6);
- **refreshToken** – текущий валидный **refresh-токен**;

В ответе возвращаются следующие параметры:

- **accessToken** – новый **access-токен** (время жизни 1 час);
- **refreshToken** – новый **refresh-токен** (время жизни 23,5 часа);
- **scope** - список прав доступа, необходимых приложению;
- **tokenType** – тип выданного токена. Всегда принимает значение «Bearer»;
- **expiresIn** – время жизни токена в секундах.

В случае прекращения работы конечных пользователей с данным приложением, следует отозвать токен доступа (access token).

Существует метод `/oauth2/ revoke` сервиса **Authorize - v2** для отзыва **access-токена**.

При обращении к данному методу следует передать следующие параметры:

- **token** – действующий **access-токен**, который следует отозвать. При отзыве **access-токена** соответствующий **refresh-токен** отзывается автоматически;
- **clientID** – **Пользовательский ключ** (Customer Key), получаемый при создании ключей в добавленном приложении (слайд №6);
- **clientSecret** – **Секретный ключ** (Customer Secret), получаемый при создании ключей в добавленном приложении (слайд №6);

В случае корректного выполнения данного метода токены будут отозваны.

Для получения новых токенов доступа потребуется вновь пройти процедуру аутентификации (слайд №7).

5) После получения токенов доступа, можно переходить к запросу необходимой «закрытой» информации. Рассмотрим на примере метода **/transactions** сервиса **Account - v1**.

- При вызове метода **/transactions** сервиса **Account - v1**, в запросе следует передать следующие параметры:
 - **accounts** – уникальные идентификаторы счетов в банке. Если параметр не передан – информация будет предоставлена по всем счетам клиента банка, доступные по правам. Например: R-5491, C-4729, C-48, R-8457482.
 - **dateFrom** – дата, с которой следует получить информацию по счетам клиента. Указывается в формате YYYY-MM-DD. Параметр обязателен к заполнению. Задаваемый период не более 3 месяцев. Например: **2016-12-01**.
 - **dateTo** – дата, по которую следует получить информацию по счетам клиента. Указывается в формате YYYY-MM-DD. Параметр обязателен к заполнению. Задаваемый период не более 3 месяцев. Например: **2016-12-01**.
- Также в **header** запроса вызова данного метода следует передать **accessToken** – действующий токен доступа (слайд №9). Например: **«Authorization: Bearer 7a622378-eb95-382f-9178-58ei9a061dc8»**.
- Если все шаги были выполнены правильно, в ответе вы получите информацию о движении по вашим счетам за выбранный период в формате JSON.

Описание структуры полученного файла находится на портале магазина API (<https://api.priorbank.by>), сервис **Account-v1** – вкладка «**Консоль API**» - метод **/transactions** – вкладка **Model**.

Завершение интеграции с API

Переход на промышленное использование

6) После успешного тестирования, следует перенастроить свое приложение для использования реальными конечными пользователями, для этого необходимо:

- перейти в настройки добавленного приложения – вкладка «**Промышленные ключи**»;
- нажать кнопку «**Создать ключи**»;
- дождаться подтверждения создания промышленных ключей;
- заменить все тестовые ключи (**пользовательский ключ** и **секретный ключ**) использовавшиеся для вызова методов на промышленные;
- после этого, при аутентификации (слайд №8) конечный пользователь будет вводить свои **ЛОГИН** и **ПАРОЛЬ** от системы Интернет-банк 2.0 и получать данные о движениях по счетам организации.

