АО "НТЦ ИТ РОСА"

СЕРВИС ИДЕНТИФИКАЦИИ И УПРАВЛЕНИЯ АККАУНТАМИ "РОСА ID"

Руководство пользователя

РСЮК.10903-01 34 01

Листов 33

2026

АННОТАЦИЯ

Данное руководство описывает функции личного кабинета (далее – ЛК) и возможности программного продукта "Сервис идентификации и управления аккаунтами РОСА ID" РСЮК.10903-01 (далее – РОСА ID, Сервис).

Перед использованием функционала РОСА ID рекомендуется ознакомиться с настоящим руководством.

Сведения, необходимые для установки, первичной настройки и эксплуатации РОСА ID, приведены в документах "РОСА ID. Руководство администратора " (индекс РСЮК.10903-01 32 01).

При разработке документа использованы положения следующих стандартов:

– ГОСТ Р 2.105-2019 "Единая система конструкторской документации (ЕСКД). Общие требования к текстовым документам";

– ГОСТ 2.601 "Единая система программной документации. Виды программных документов";

– ГОСТ 19.101-77 "Единая система программной документации.

Виды программ и программных документов";

– ГОСТ 19.105-78 "Единая система программной документации. Общие требования к программным документам";

– ГОСТ 19.503-79 "Единая система программной документации. Руководство системного программиста".

Настоящий документ подготовлен в соответствии с технологической инструкцией: "РОСА. Регламент формирования документации к программным продуктам" .

СОДЕРЖАНИЕ

1 Назначение Сервиса **4**
2 Вход в личный кабинет **5**
2.1 Регистрация в POCA ID 6
2.2 Сброс пароля 7
3 Использование личного кабинета **9**
3.1 Интерфейс личного кабинета 9
3.2 Персональная информация 10
3.3 Безопасность 12
3.4 Сервис 13
3.5 Сеансы 17
3.6 Устройства 18
3.7 История 19
3.8 Документы 20
4 Подключение сервиса к POCA ID **22**
4.1 Общие сведения 22
4.2 Авторизация пользователя 22
4.3 Получение токена 24
4.4 Обновление токена 27
4.5 Список запросов 29
4.6 ID-токен 30
4.7 Ошибки 31
Термины и сокращения **32**

1 НАЗНАЧЕНИЕ СЕРВИСА

РОСА ID – это единый и безопасный способ входа (идентификация) в приложения в экосистеме программных продуктов РОСА.

Сервис может быть использован для единого и безопасного способа входа в приложения.

Личный кабинет пользователя РОСА ID предоставляет графический веб-интерфейс для контроля и управления персональными данными пользователя, его учётной информацией, а также устройствами, сеансами и историей подключений.

В ЛК можно настроить подключения сервисов к РОСА ID.

Дополнительно в ЛК размещены соглашения по использованию Сервиса и обработке персональных данных.

2 ВХОД В ЛИЧНЫЙ КАБИНЕТ

Для корректной работы ЛК Сервиса необходимо использование актуальных операционных систем и браузеров, обеспечивающих поддержку современных веб-стандартов (HTML5, CSS3, JavaScript), а также совместимость с аппаратной архитектурой устройств.

Использование ЛК не требует установки дополнительного программного обеспечения.

Следует обратить внимание, что при использовании устаревших версий ОС и браузеров не гарантируется стабильная работа и полная функциональность Сервиса.

Доступ к веб-интерфейсу ЛК POCA ID осуществляется пользователем с рабочей станции или мобильного устройства через один из рекомендуемых браузеров актуальной версии:

- Google Chrome;
- Microsoft Edge;
- Apple Safari;
- Mozilla Firefox;
- Яндекс.Браузер.

Для входа в веб-интерфейс ЛК Сервиса необходимо:

а) ввести в адресной строке браузера доменное имя сервера POCA ID:

```
https://id.rosa.ru
```

б) на экране появится страница авторизации веб-интерфейса (рисунок 1);

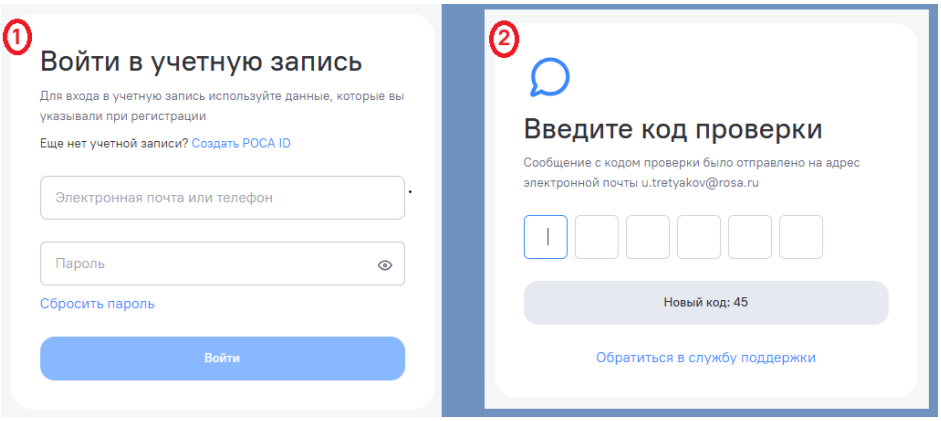

Рисунок 1 – Экран авторизации в POCA ID

в) ввести учётную запись (электронная почта или телефон) и пароль пользователя, после чего нажать кнопку `Войти`;

г) ввести код проверки, направленный на указанные электронную почту или телефон (рисунок 1).

В случае успешной авторизации на экране откроется интерфейс ЛК.

2.1 Регистрация в POCA ID

Если у пользователя нет аккаунта POCA ID, то для его создания необходимо нажать гиперссылку `Создать POCA ID` . Затем в форме регистрации (рисунок 2), ввести данные в поля:

- Имя*;
- Электронная почта*;
- Телефон*;
- Пароль*;
- Подтвердить пароль*.

После этого отметить флажками согласия:

- Я принимаю условия Пользовательского соглашения*;
- Я согласен на Обработку персональных данных*;
- Я согласен получать рекламные материалы НТЦ ИТ РОСА.

Поля, помеченные символом "*" (звёздочка), обязательны для заполнения. При их отсутствии регистрация невозможна.

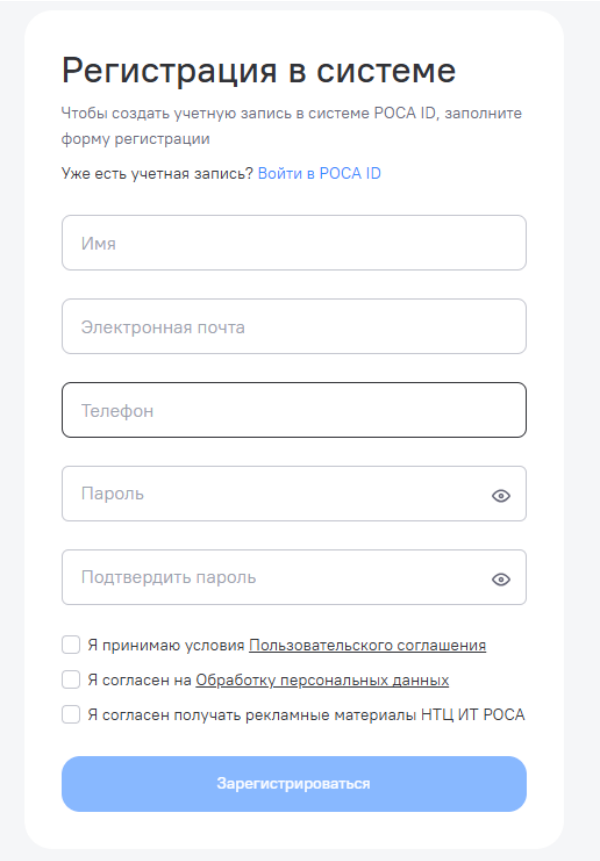

Рисунок 2 – Форма регистрации в РОСА ID

По готовности нужно нажать кнопку `Зарегистрироваться` . На введённую электронную почту будет направлено письмо. После выполнения указанных в нём действий можно приступить к использованию РОСА ID.

2.2 Сброс пароля

Если пользователь забыл пароль, то для восстановления необходимо нажать ссылку `Сбросить пароль` . Затем в форме "Сброс пароля" (рисунок 3), ввести данные в поля:

- Электронная почта* – указать почтовый адрес, привязанный к ранее созданному аккаунту;
- Пароль* – указать новый пароль;
- Подтвердить пароль* – указать новый пароль.

Поля, помеченные символом "*" (звёздочка), обязательны для заполнения. При их отсутствии сброс пароля невозможен.

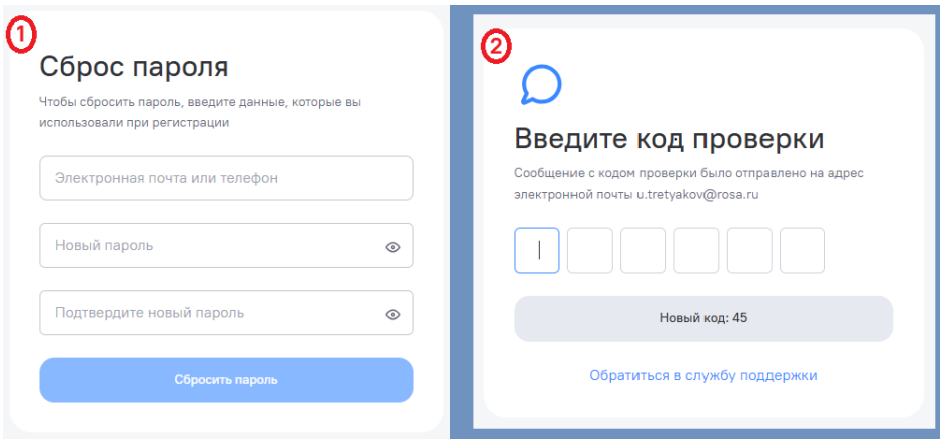

Рисунок 3 – Форма "Сброс пароля"

По готовности нужно нажать кнопку `Сбросить пароль` . Далее ввести код проверки, направленный на указанные электронную почту или телефон (Рисунок 3). После выполнения можно приступить к использованию РОСА ID с новым паролем.

3 ИСПОЛЬЗОВАНИЕ ЛИЧНОГО КАБИНЕТА

3.1 Интерфейс личного кабинета

Интерфейс ЛК РОСА ID состоит из панели навигации с доступными пользователю пунктами меню, информацией о пользователе и функциональными кнопками, а также рабочей области, в которой выводится информация и функционал выбранного раздела (рисунок 4).

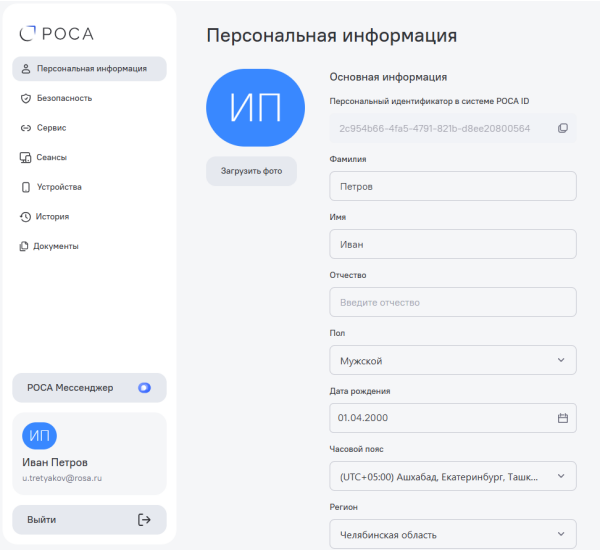

Рисунок 4 – Интерфейс ЛК РОСА ID

Пункты меню панели навигации позволяют переключаться по следующим разделам ЛК:

- Персональная информация;
- Безопасность;
- Сервис;
- Сеансы;
- Устройства;
- История;
- Документы.

Блок информации о пользователе панели навигации содержит:

- Фото, прикреплённое в аккаунте пользователя. Если фото не было прикреплено, то отобразится символьная иконка с инициалами;
- Имя и Фамилия пользователя, указанные в аккаунте;
- Адрес электронной почты пользователя (e-mail).

В навигационной панели размещены две кнопки:

– 'РОСА Мессенджер – запускает РОСА Мессенджер для текущего аккаунта РОСА ID;

– 'Выйти – завершает работу в ЛК.

Навигационная панель постоянно присутствует на экране.

Выбранный пункт меню подсвечивается, а в рабочей области отображается информация соответствующего раздела.

Далее описаны свойства и возможности разделов.

3.2 Персональная информация

Раздел "Персональная информация" предназначен для управления персональными данными пользователя (рисунок 5).

На экране раздела размещаются следующие данные (поля, помеченные символом "*" (звёздочка), обязательны для заполнения):

– Персональный идентификатор в системе РОСА ID;
– Фотография (загрузка с диска; если не загружена, то отображается символьная иконка);
– Фамилия;
– Имя*;
– Отчество;
– Пол* (выбор из списка: мужской, женский);
– Дата рождения* (выбор из календаря);
– Часовой пояс *(выбор из списка);
– Регион (выбор из списка);
– Адрес;
– Язык* (выбор из списка: русский, английский):
– Тема оформления* (выбор из списка: светлая, тёмная, системная);
– Контактные данные:
– Электронная почта* (специальная форма);
– Телефон* (специальная форма).

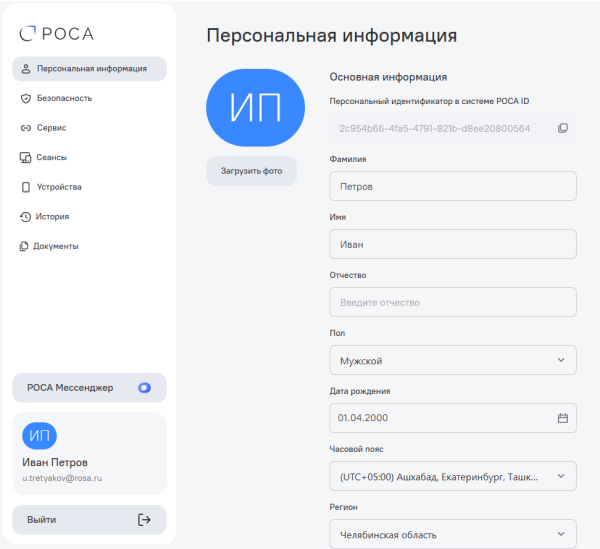

Рисунок 5 – Раздел "Персональная информация"

В разделе есть возможность копировать и изменять информацию.

Поля с пиктограммой (копия) могут быть скопированы в буфер обмена.

Для корректировки поля "Электронная почта" необходимо нажать пиктограмму (карандаш). После заполнения данных на форме (рисунок 6) нажать кнопку `Отправить письмо` . Далее последовательно ввести коды проверки, направленные на указанный в аккаунте телефон и новый адрес электронной почты.

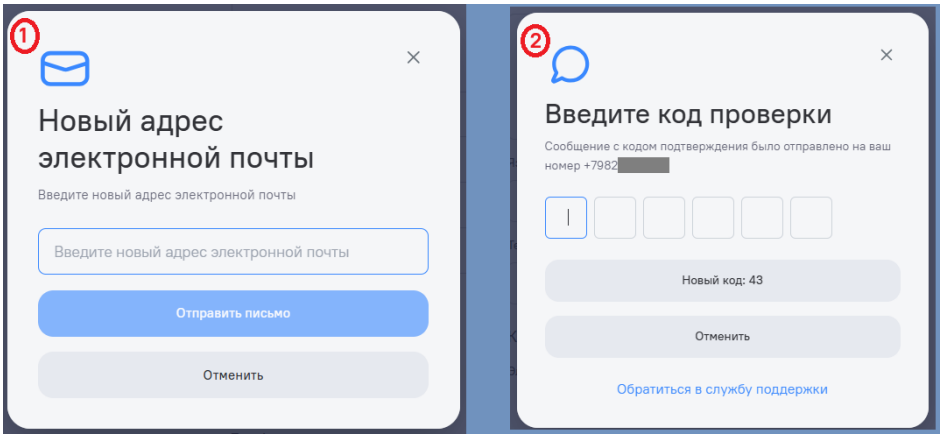

Рисунок 6 – Форма изменения "Электронная почта"

Для корректировки поля "Телефон" необходимо нажать пиктограмму (карандаш). После заполнения данных на форме (рисунок 7) нажать

кнопку `Получить код`. Подтвердить изменение вводом кода проверки, направленного на указанную в аккаунте электронная почту, и далее кодом проверки, направленным на новый номер телефона.

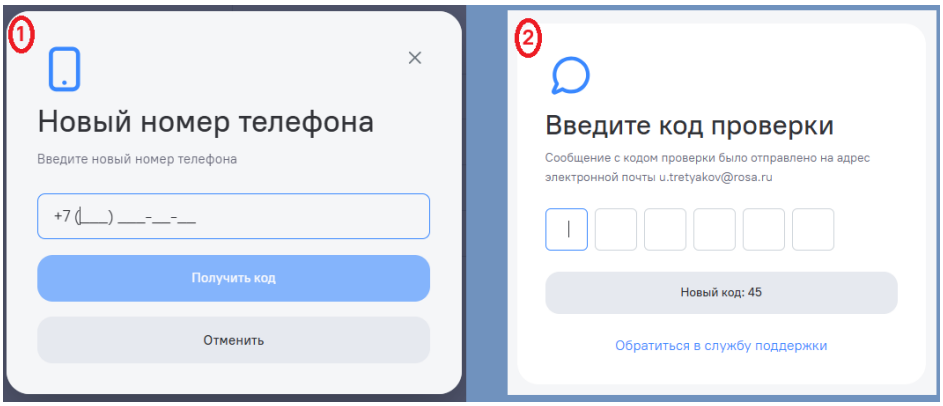

Рисунок 7 – Форма изменения "Телефон"

После корректировки данных необходимо нажать кнопку `Сохранить` для фиксации или `Отменить` – для отказа от изменений.

3.3 Безопасность

Раздел предназначен для управления аккаунтом POCA ID: смены пароля или удаления (рисунок 8).

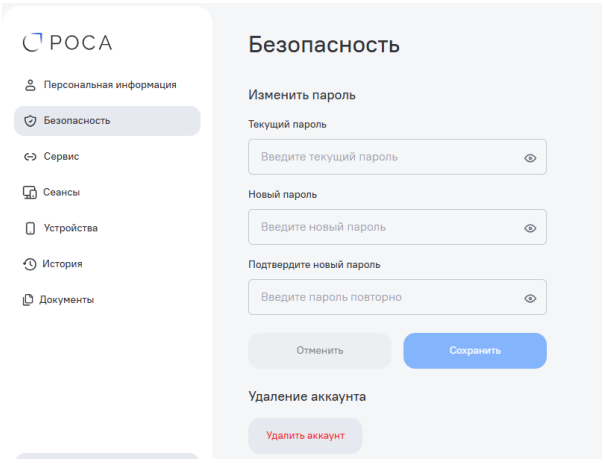

Рисунок 8 – Раздел "Безопасность"

Для смены пароля необходимо ввести текущий пароль.

Затем необходимо дважды ввести новый пароль и нажать на кнопку `Сохранить` .

Набранный пароль по умолчанию не отображается на экране (отображается в виде символов "*"). Для его отображения следует нажать на пиктограмму (изменить вид). Для возвращения пароля к скрытому виду можно повторно нажать на пиктограмму.

Для удаления аккаунта необходимо нажать кнопку `Удалить аккаунт` и подтвердить действие в следующих запросах. После удаления все сессии на других устройствах и текущий сеанс работы в ЛК будут завершены.

3.4 Сервис

Раздел предназначен для регистрации подключения сторонних, в том числе собственной разработки, приложений к POCA ID.

Этот раздел нужен, при наличии своих сайтов, приложений, ботов или сервисов и необходимости входа в них пользователей по кнопке `Войти через POCA ID`, с использованием единых идентификационных данных. По нажатию пиктограммы (стрелка вниз) можно посмотреть расширенное описание механизма использования POCA ID сторонними приложениями.

Для регистрации необходимо нажать кнопку `Подключить сервис` (рисунок 9).

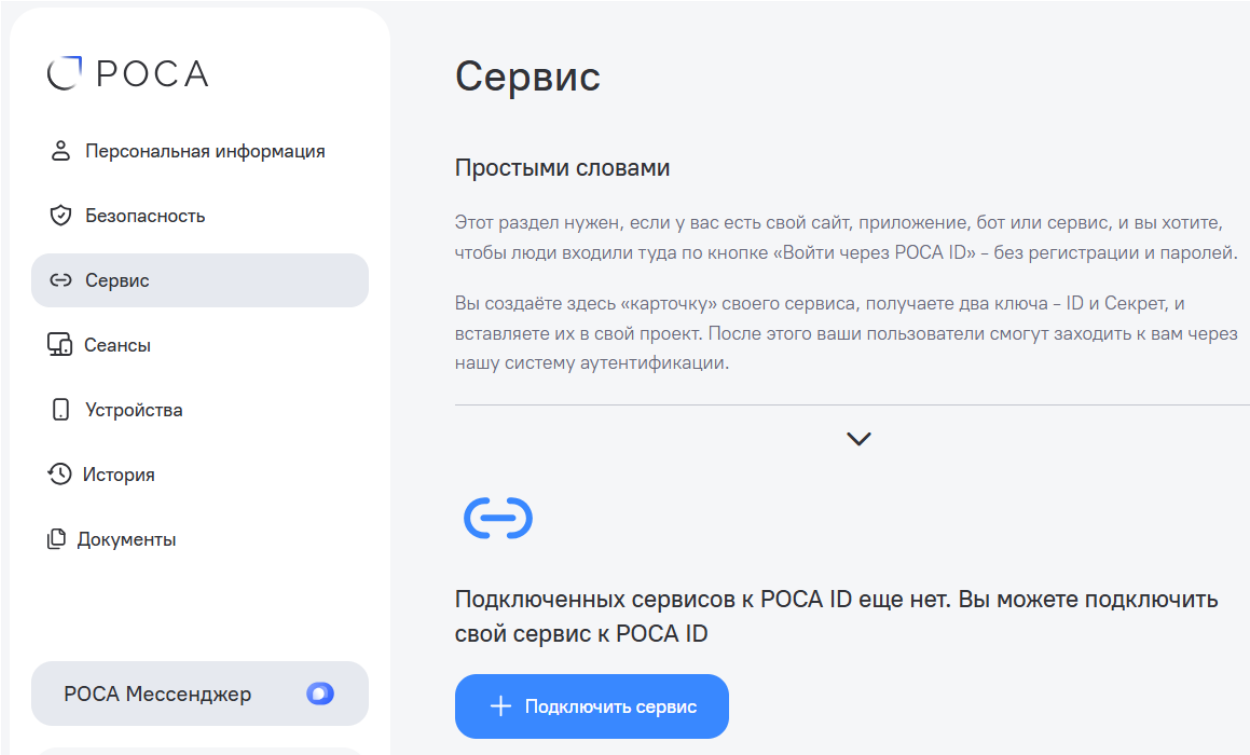

Рисунок 9 – Раздел "Сервис"

В форме подключения (рисунок 10) указать:

– Наименование – используется для идентификации в интерфейсе приложения;

– Используемые платформы – платформы, на которых планируется использование клиента. Выбор производится из списка с возможными вариантами: Веб-сервис, Мобильное приложение;

– Redirect URI – адрес, на который будет происходить переадресация при выполнении авторизации.

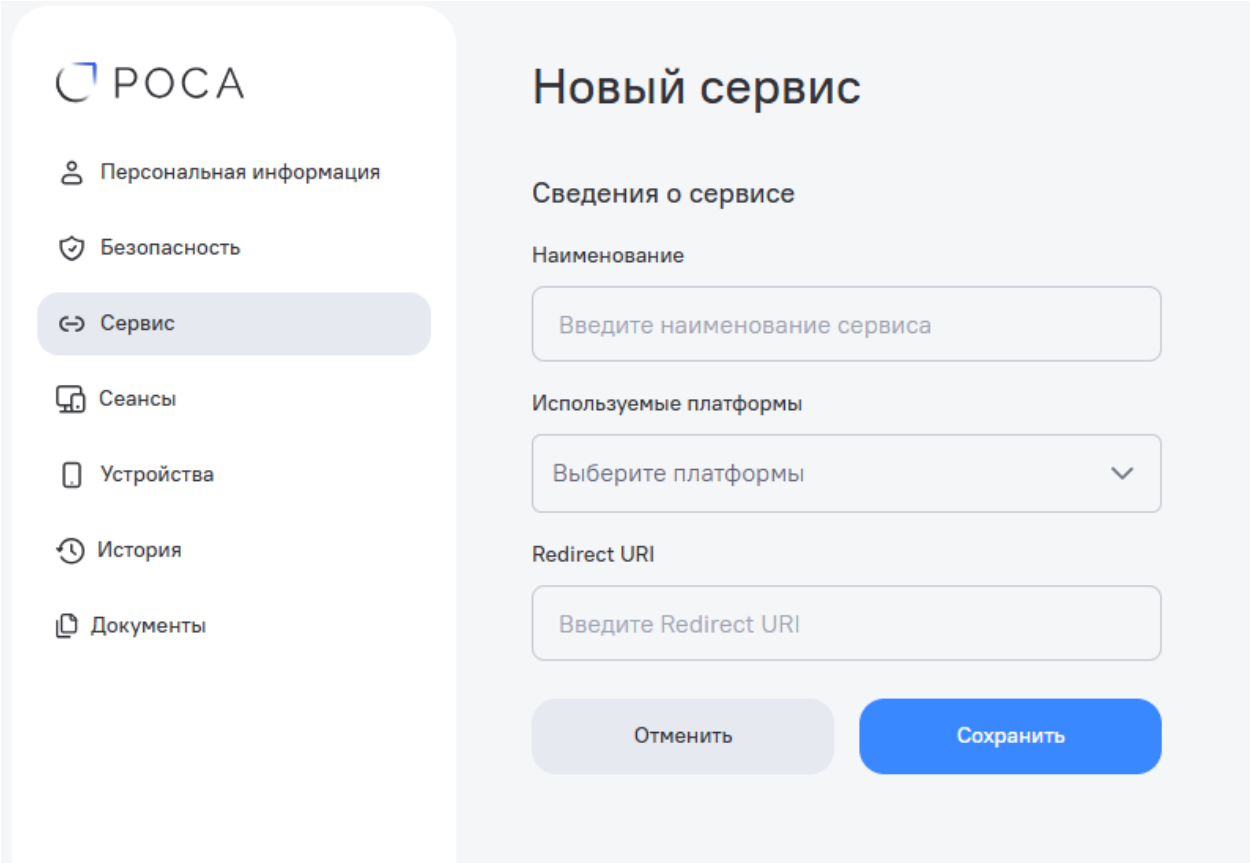

Рисунок 10 – Форма "Новый сервис"

После успешной регистрации клиенту будут присвоены Client ID и Client Secret, которые можно посмотреть на странице "Сведения о сервисе". Значение Client ID является публичным и может передаваться пользователям; значение Client Secret необходимо сохранять в секрете, т. к. оно используется для идентификации клиента при запросах к приложению.

Для обеспечения взаимодействия с Сервисом приложение должно реализовывать соответствующий набор операций (см. Подключение сервиса к POCA ID).

Сервисов может быть добавлено несколько. В разделе они будут отображаться списком (рисунок 11).

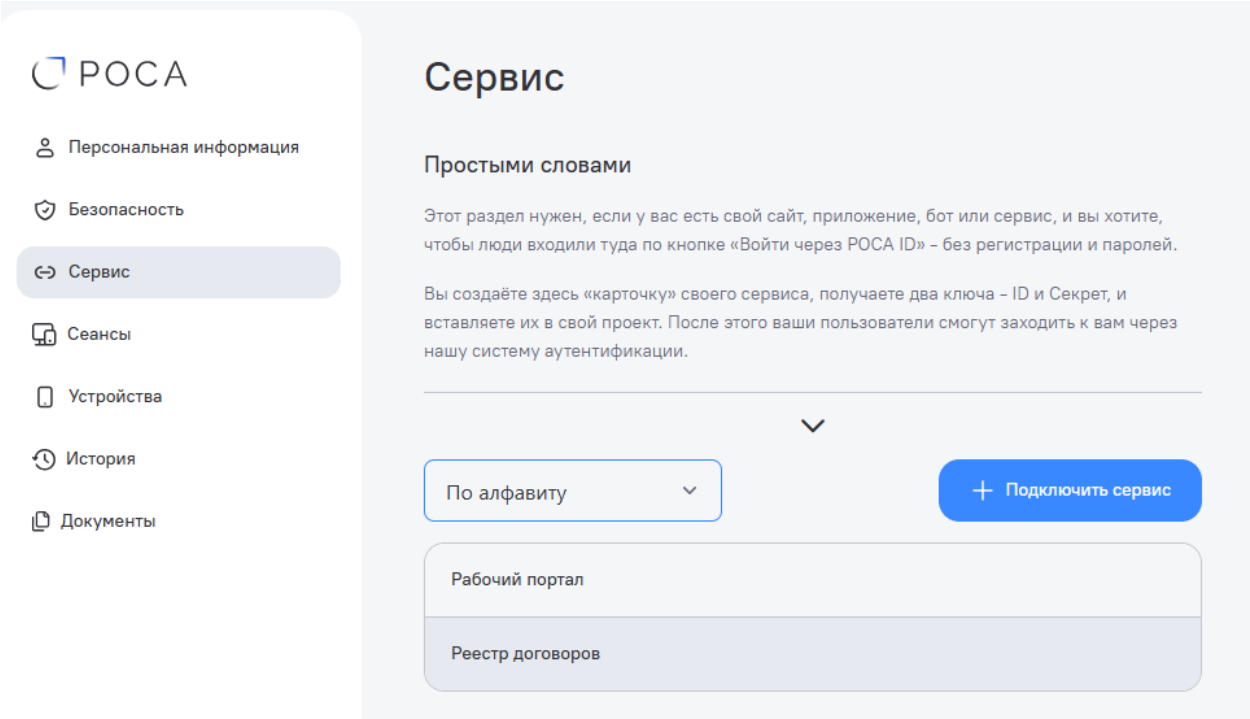

Рисунок 11 – Раздел "Сервис" с добавленными сервисами

При этом возможны изменение порядка в списке и дополнительные действия по изменению и удалению, при нажатии на сервис (рисунок 12).

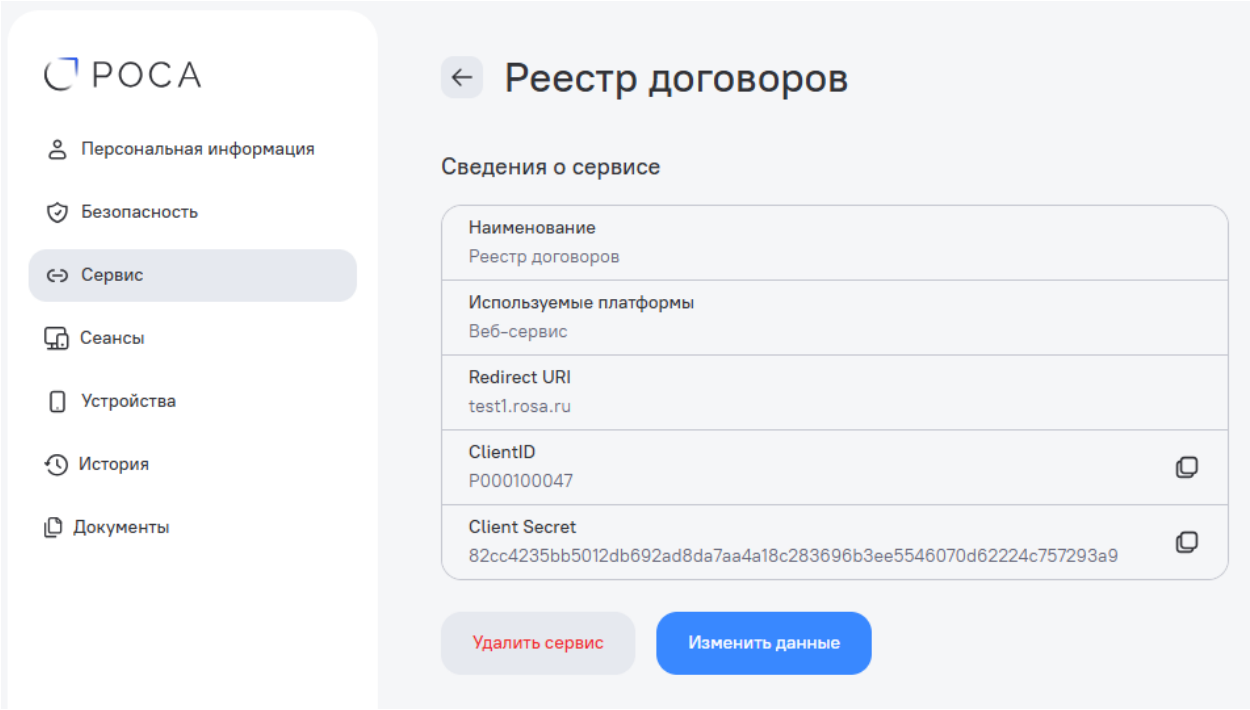

Рисунок 12 – Страница "Сведения о сервисе"

3.5 Сеансы

Раздел предназначен для работы с сеансами POCA ID: просмотра и завершения (рисунок 13).

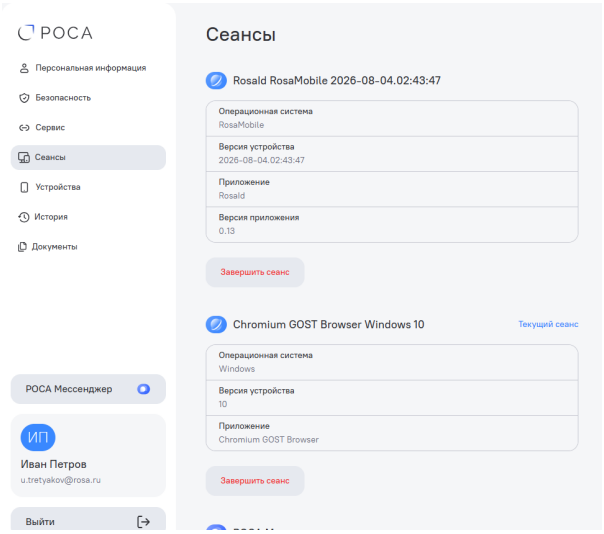

Рисунок 13 – Раздел "Сеансы"

На рабочем экране раздела выводится список активных сеансов с указанием операционной системы, версии устройства, приложения и версии приложения.

Для принудительного завершения сеанса необходимо нажать кнопку `Завершить сеанс` и подтвердить завершение.

Для отключения всех сеансов необходимо нажать кнопку `Завершить все сеансы` и подтвердить завершение.

3.6 Устройства

Раздел предназначен для управления устройствами, подключёнными к POCA ID: просмотр подключений и выход на устройстве из Сервиса (рисунок 14).

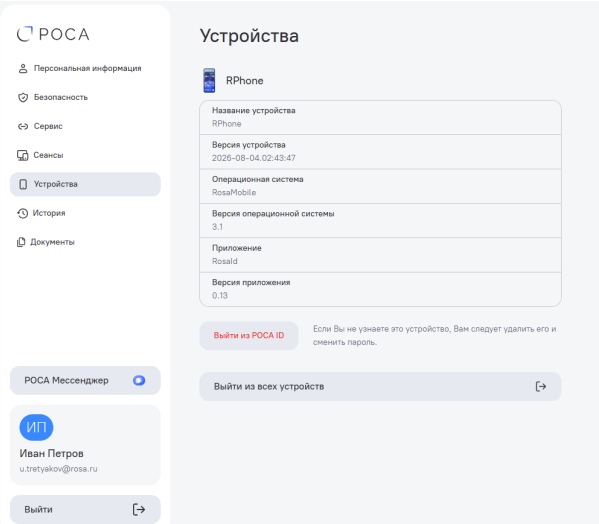

Рисунок 14 – Раздел "Устройства"

На рабочем экране раздела выводится список активных устройств, на которых выполнена идентификация в POCA ID, с указанием названия устройства, версии устройства, операционной системы, версии операционной системы, приложения и версии приложения.

Для принудительного выхода из POCA ID на устройстве, необходимо нажать кнопку `Выйти из POCA ID` и подтвердить выход.

Для выхода на всех устройствах следует нажать кнопку `Выйти из всех устройств` и подтвердить выполнение.

3.7 История

В разделе можно посмотреть историю подключений к POCA ID (рисунок 15).

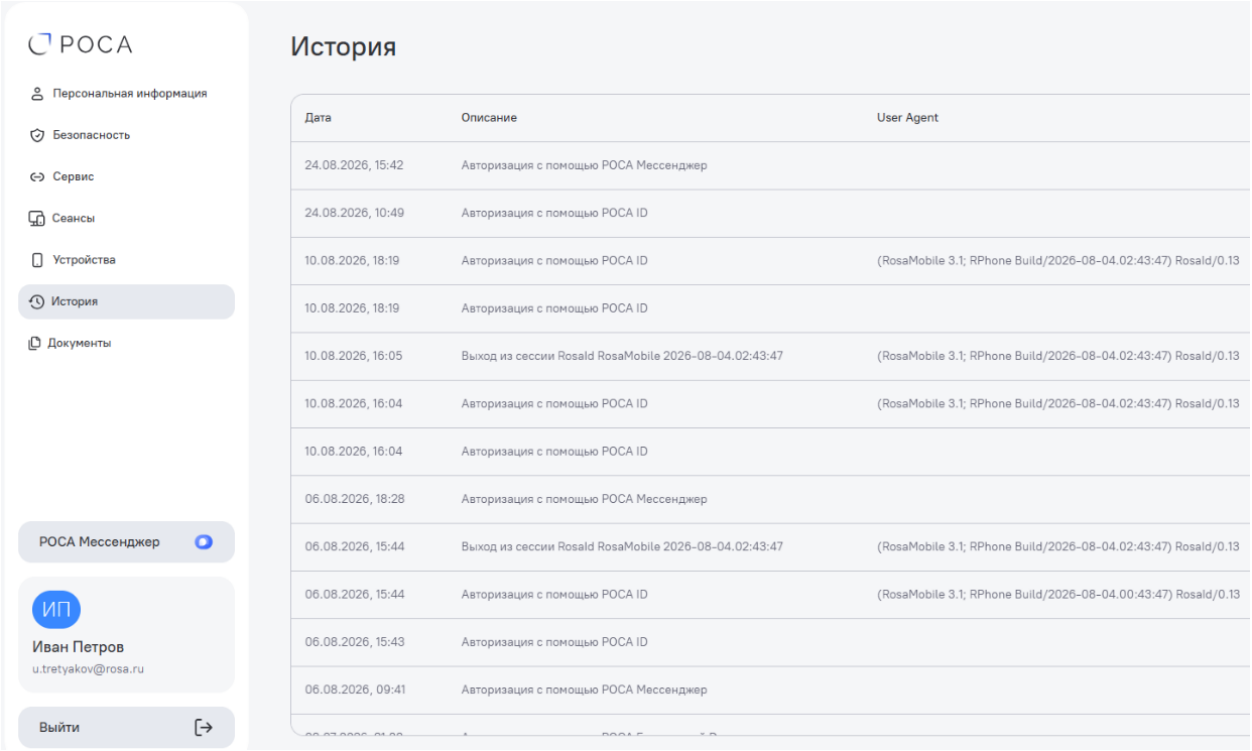

Рисунок 15 – Раздел "История"

На экран выводится список операций авторизации и выхода с указанием даты, описания и устройства (User Agent).

3.8 Документы

В разделе представлены для ознакомления документы АО "НТЦ ИТ РОСА" по использованию сервиса POCA ID (рисунок 16):

- Пользовательское соглашение сервиса POCA ID;
- Политика конфиденциальности POCA ID;
- Соглашение об обработке персональных данных POCA ID.

Для просмотра документа необходимо нажать на соответствующую строку.

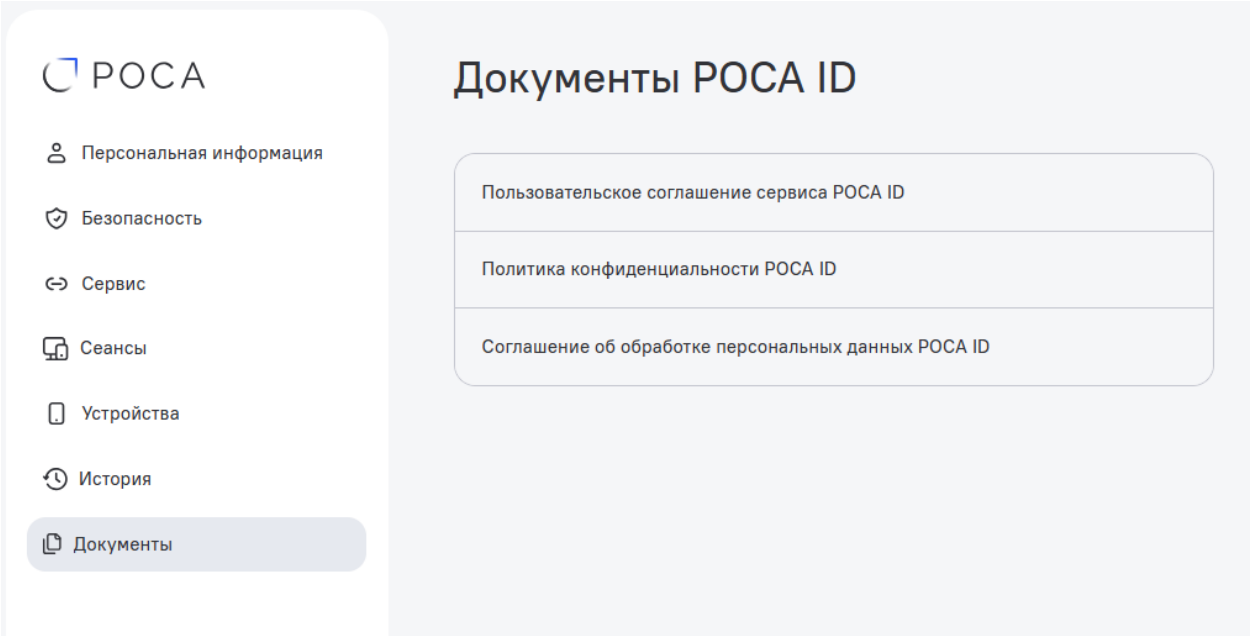

Рисунок 16 – Раздел "Документы"

4 ПОДКЛЮЧЕНИЕ СЕРВИСА К POCA ID

4.1 Общие сведения

Клиентом в POCA ID называется подключённый внешний сервис, взаимодействующий с зарегистрированным пользователем. Авторизация пользователя во внешнем сервисе осуществляется в соответствии со стандартом *OpenID*.

4.2 Авторизация пользователя

После успешной регистрации клиента появляется возможность авторизовать пользователя от имени данного клиента. Для этого необходимо направить пользователя по сформированному URL-адресу.

Ссылка для формирования, по которой нужно направить пользователя, желающего авторизоваться с использованием POCA ID, состоит из следующих частей:

- хост – базовый URL приложения POCA ID (например, `https://id.rosa.ru`);
- путь – путь до эндпойнта авторизации (всегда /authorize);
- query-параметры – параметры для авторизации, состоящие из пар "ключ-значение".

Query-параметры и их возможные значения приведены в таблице 1.

Таблица 1 – Query-параметры

Название параметра	Возможные значения	Описание	Пример
scope	openid name email phone_number birthdate gender address	Список значений, определяющий границы списка возвращаемых полей пользователя, а также возможных действий, совершаемых над пользователем. Значения списка разделяются пробелом. Описание значений описаны далее	openid email name
response_type	Code		code

Название параметра	Возможные значения	Описание	Пример
client_id		Client ID клиента, полученный после регистрации	P000001000
redirect_uri		Redirect URI клиента, указанный при регистрации	https://google.com

Для авторизации включение значения openid в параметре scope является обязательным.

Возможные значения параметра scope приведены в таблице 2.

Таблица 2 – Значения параметра scope

Значение	Поля пользователя в приложении, к которым предоставляется доступ
openid	-
name	name, given_name, family_name, middle_name
email	email, email_verified
phone_number	phone_number, phone_number_verified
birthdate	birthdate
gender	gender
address	address

Пример полной ссылки (URL-encoded):

```
https://id.rosa.ru/authorize? scope=openid+email+name&response_type=code&client_id=P000001000&redire
ct_uri=https%3A%2F%2Fgoogle.com
```

После успешной (или неуспешной) авторизации пользователь будет перенаправлен по установленному Redirect URI. При этом в query-параметрах будет находиться необходимая информация для дальнейшей обработки клиентом.

Список возвращаемых query-параметров:

– При успешной авторизации:

– code – код авторизации, используемый для дальнейшего получения авторизационного токена, для использования клиентом (процедура описана в следующих разделах);

Пример: `https://google.com?code=AAAAAAAAAAAAAAAAAAAA` .

– При возникновении ошибки:

– unsupported_response_type – значение response_type отличается от единственного поддерживаемого code;

– invalid_scope – запрошено несуществующее значение scope либо отсутствует обязательное для авторизации значение openid;

– access_denied – пользователь отклонил запрос на авторизацию.

Формат возвращаемой ошибки описан в п. 4.7.

Пример:

```
https://google.com? error=invalid_scope&error_description=%22ope
nid%22%20scope%20is%20requ
ired
```

4.3 Получение токена

В случае успешного получения кода авторизации клиент может получить авторизационный токен, привязанный к авторизуемому пользователю. В дальнейшем этот токен будет использоваться для всех запросов, касаемых этого пользователя.

Для получения токена клиенту необходимо отправить следующий запрос:

– URL – api/oauth/client/token;
– Метод – POST;
– Тип данных – x-www-form-urlencoded.

Параметры тела запроса приведены в таблице 3.

Таблица 3 – Параметры теля запроса

Параметр	Возможные значения	Описание
client_id	--	Client ID клиента, полученный после регистрации
client_secret	--	Client Secret клиента, полученный после регистрации
grant_type	authorization_code	Всегда authorization_code
redirect_uri	--	Redirect URI клиента, указанный при регистрации
code	--	Код авторизации, полученный на предыдущем шаге

Пример в формате curl:

```
curl \
--location 'https://id.rosa.ru/api/oauth/client/token' \
--header 'Content-Type: application/x-www-form-urlencoded' \
--data-urlencode 'client_id=P000000002' \
--data-urlencode 'client_secret=9b821de045bf3b55555535\
e84c011cc96217c11b19139cab8786704f9c3ca8b9' \
--data-urlencode 'grant_type=authorization_code' \
--data-urlencode 'code=AAAAAAAAAAAAAAAAAAAAA' \
--data-urlencode 'redirect_uri=https://google.com'
```

Ответ может быть возвращен в формате JSON (application/json).

Список полей при успешном запросе приведет в таблице 4.

Таблица 4 – Поля при успешном запросе

Имя поля	Описание
access_token	Авторизационный токен
token_type	Тип полученного токена (всегда Bearer)
refresh_token	Refresh-токен
expires_in	Срок действия авторизационного токена (в секундах)
scope	Список значений scope. Значения разделяются пробелами. Должен совпадать с запрошенным списком.
id_token	ID-токен, содержащий запрошенные данные пользователя. Представляет из себя зашифрованную JWT-строку.

Формат данных ID-токена описан в п. 4.6.

Пример:

```
{
"access_token":
"e697d038899ccd0204ec41f215e964d63b0fb423d3ff238a1b7632924976804
d",
"token_type": "Bearer",     "refresh_token":
"a093eab561a7278548c857b5699ff463f3180c506cec1b4791cf84ff1aaed4a
4",
```

```
"expires_in": 86399,
"scope": "email openid",
"id_token":
"eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJyb3NhaWQiLCJzdW
IiOiJh
ZDNlZTI3NS04MzE5LTRhMTMtYmNiMC1iMzI4ZTJjNTZmMDAiLCJhdWQiOiJQMDAw
MDAwMD
cwIiwiZXhwIjoxNzE2ODM0NTgzLCJpYXQiOjE3MTY4MzA5ODMsImVtYWlsIjoidX
NlckBk
dW1teS5tYWlsIiwiZW1haWxfdmVyaWZpZWQiOnRydWV9.hqH_9xYPGK09bKEyYRr
FFmrEc
ZD3Xu-
h6T8M9ffIHJgTwe1nH4XMzgpk0Nax3OUS_XkqNtBSDj1VNWGAgfWrDR8gtCnjFZ3
w50HF3
Qz6w2TljsbrfiA6JeW7_Z7-pz-JNY1-YqpuSs0ir-
cChvJAolmVyYa1pSi4oUyPlvDXq2AhCWXMNm0tSrRlR5iuP3XGjy5aZzD2rymFxB
x4kzD9
Y63hYr6bPSm7GFtZS_gTAguI1VkUt79widOsSlQVE4pqgGekcqHEHn61tECiFN4V
z0S86U
rqgmaHspIPM1tH6Eji49b_A_8rF8Dm9IYB-m0ul-
wuB5g5h4cpPiYf7c4WpJxwzG5wBWKWZv060Ake4sKkW9zBNwdcXrUabYvBL4i1Ng
156j3f
xN1O1Z3yqF6rfy7Ud7Y0mdXQh65X3s31s3TYC3WNoZkfyIqL-
wHSbQWxxXrdlEoX1JoWEkdTraNZlRORpUFYgruNVgqCBAwXmx7IY6_mWD8Fb4Fk0
JIOwQg
dxBoA5EVy4X71woeEd-TY0NU4DQK9qfrksQCKZr9R4AaXe-7B-
SjMkFPbRHlSDpDyOyUZMWMnT9ta6CG615TyAos3388ZZqbD3GU_xQMJufu7tIYfi
uXlpbw
-226OvBFI6UxIwnlOMKEGkHWDtvZokj0jD9rlLq_Dwq2fxTOUGlU" }
```


Ответ также может возвращать ошибки.

Формат возвращаемых ошибок описан в п. 4.7.

Список ошибок, которые могут возникнуть, приведен в таблице 5.

Таблица 5 – Список ошибок

Название ошибки	Условия возникновения
invalid_request	Неверно сформирован запрос
invalid_grant	Переданный авторизационный код не найден (или не принадлежит этому клиенту)

Название ошибки	Условия возникновения
unsupported_grant_type	Передано неподдерживаемое значение grant_type
server_error	Ошибка на стороне сервера

На основании полученного в ID-токене значения sub клиент может авторизовывать пользователя.

4.4 Обновление токена

По истечении срока полученный токен можно обновить (получить новую пару).

Для этого необходимо отправить следующий запрос:

- URL – api/oauth/client/token;
- Метод – POST;
- Тип данных – x-www-form-urlencoded.

Параметры тела запроса приведены в таблице 6.

Таблица 6 – Параметры тела запроса

Параметр	Возможные значения	Описание
client_id	--	Client ID клиента, полученный после регистрации
client_secret	--	Client Secret клиента, полученный после регистрации
grant_type	refresh_token	Всегда refresh_token
refresh_token	--	Refresh-токен, полученный вместе с авторизационным токеном

Пример в формате curl:

```
curl \
--location 'https://id.rosa.ru/api/oauth/client/token' \
--header 'Content-Type: application/x-www-form-urlencoded' \
--data-urlencode 'client_id=P000000002' \
--data-urlencode 'client_secret=9b821de045bf3b33d83135e84\
c011cc96217c11b19139cab8786704f9c3ca8b9' \
--data-urlencode 'grant_type=refresh_token' \
```

```
--data-urlencode 'refresh_token=2de7c7966583ff3a9cf7ae581240\
caf769d1384a0c7cc6a0f7034bacb4d619c3'
```

Ответ возвращается в формате JSON (application/json).

Список полей при успешном запросе приведен в таблице 7.

Таблица 7 – Поля при успешном запросе

Имя поля	Описание
access_token	Авторизационный токен
token_type	Тип полученного токена (всегда Bearer)
refresh_token	Refresh-токен
expires_in	Срок действия авторизационного токена (в секундах)
scope	Список значений scope. Значения разделяются пробелами. Должен совпадать с запрошенным списком.

Пример:

```
{
"access_token":
"e697d038899ccd0204ec41f215e964d63b0fb423d3ff238a1b7632924976804
d",
"token_type": "Bearer",     "refresh_token":
"a093eab561a7278548c857b5699ff463f3180c506cec1b4791cf84ff1aaed4a
4",
"expires_in": 86399,
"scope": "email openid" }
```

В случае возникновения ошибок используется список, приведенный в таблице 8.

Формат возвращаемых ошибок описан в п. 4.7.

Таблица 8 – Список ошибок

Название ошибки	Условия возникновения
invalid_request	Неверно сформирован запрос
invalid_grant	Переданный refresh-токен не найден (или не принадлежит этому клиенту)

Название ошибки	Условия возникновения
unsupported_grant_type	Передано неподдерживаемое значение grant_type
server_error	Ошибка на стороне сервера

4.5 Список запросов

Запросы, которые можно выполнить с применением авторизационного токена:

UserInfo – запрос возвращает информацию о пользователе в соответствии с выданными правами:

- URL – api/oauth/client/userinfo;
- Метод – GET;
- Тип данных – application/json.

Заголовок запроса описан в таблице 9.

Таблица 9 – Заголовок запроса

Имя	Значение	Описание
Authorization	Bearer {{access_token}}	Авторизация с использованием Bearer-токена

Тело запроса отсутствует.

Пример в формате curl:

```
curl \
--location 'https://id.rosa.ru/api/oauth/client/userinfo' \
--header 'Authorization: Bearer e12669547d6ee73e4a7f57c80\
ec97e7dfa1a98b1303e8e9ff4aa8228de9236ea'
```

Ответ возвращается в формате JSON (application/json).

Список полей при успешном запросе приведен в таблице 10. Наличие полей определяется выданными разрешениями.

Таблица 10 – Список полей

Имя поля	Тип	Описание
sub	String (UUID)	Уникальный идентификатор пользователя
name	String	Полное имя

Имя поля	Тип	Описание
given_name	String	Имя
family_name	String	Фамилия
middle_name	String	Отчество
email	String	Email-адрес
email_verified	Boolean	Email-адрес подтвержден
gender	String	Пол
birthdate	String	Дата рождения
phone_number	String	Номер телефона
phone_number_verified	Boolean	Номер телефона подтвержден
address	String	Адрес
avatar	Map	Аватар

Пример:

```
{
"sub": "ad3ee275-8319-4a13-bcb0-b328e2c56f00",
"email": "user@dummy.mail",
"email_verified": true }
```

4.6 ID-токен

ID-токен представляет собой зашифрованную JWT-строку, подписанную соответствующим ключом. При расшифровке обязательна проверка на соответствие подписи и публичного ключа данного сервиса.

Содержание JWT приведено в таблицах 11 и 12.

Таблица 11 – Заголовок (header)

Название	Описание	Возможные значения
typ	Тип (всегда JWT)	JWT
alg	Алгоритм шифрования	RS256

Таблица 12 – Данные (data/payload)

Название	Описание	Возможные значения
iss	Эмитент	rosaid
sub	Тема (уникальный идентификатор пользователя)	--

Название	Описание	Возможные значения
aud	Получатель (ID клиента, запрашивающего данные)	--
exp	Время истечения (в секундах)	--
iat	Время выдачи (в секундах)	--

4.7 Ошибки

OAuth-ошибки состоят из словаря (ключ-значение), содержащего информацию и подробности конкретной ошибки. Ошибка может возвращаться в формате query-значений либо в виде JSON-объекта. В обоих случаях формат остаётся неизменным.

Ошибка содержит поля, приведенные в таблице 13.

Таблица 13 – Поля ошибки

Имя поля	Возможные значения	Описание	Обязательность
error	invalid_request unauthorized_client access_denied unsupported_response_type invalid_scopeserver_error temporarily_unavailable invalid_client invalid_grant unsupported_grant_type	Одно из фиксированных значений для определения ошибки	Да
error_description	--	Текстовое описание возникшей ошибки	Нет
error_uri	--	URL-адрес, содержащий описание ошибки	

ТЕРМИНЫ И СОКРАЩЕНИЯ

Термин/Сокращение	Определение
БД	База данных
ВМ	Виртуальная машина
ИТ	Информационные технологии
ОС	Операционная система
ПО	Программное обеспечение
СИПА	Система идентификации, политик и аудита
ЦС	Центр сертификации
API	Application programming interface – программный интерфейс приложения
BMC	Baseboard management controller – контроллер управления материнской платой
CA	Certification authority – центр сертификации (удостоверяющий центр)
CDN	Content Delivery Network – сеть доставки содержимого
CSV	Comma separated values – текстовый формат, предназначенный для представления табличных данных
DHCP	Dynamic host configuration protocol – протокол динамической настройки сетевой конфигурации узла
DNS	Domain name system – система доменных имен
ESR	Extended support release – релиз с расширенной (долговременной) поддержкой
HTTP	Hypertext transfer protocol – протокол передачи гипертекста
HTTPS	Hypertext transfer protocol secure – защищенная версия протокола передачи гипертекста
IP	Internet protocol – протокол межсетевого взаимодействия
IPA	Identity, policy and audit – система идентификации, политик и аудита (СИПА)
IPMI	Intelligent Platform Management Interface - интеллектуальный интерфейс для управления и администрирования серверов
LDAP	Lightweight directory access protocol – протокол доступа к каталогам
LDAPS	Lightweight directory access protocol secure – защищенная версия протокола доступа к каталогам
MAC	Media access control – уникальный идентификатор сетевого оборудования
MTA	Mail transfer agent – агент передачи почтовых сообщений

Термин/Сокращение	Определение
NTP	Network time protocol – протокол сетевого времени
RHUI	Red Hat Upgrade Infrastructure – инфраструктура обновления Red Hat
SMTP	Simple mail transfer protocol – протокол передачи почтовых сообщений
SSH	Secure shell – защищенная оболочка
SSL	Secure sockets layer – уровень защищенных сокетов
TCP	Transmission control protocol – протокол управления передачей данных
TFTP	Trivial file transfer protocol – протокол передачи файлов
UDP	User datagram protocol – протокол пользовательских датаграмм
URL	Uniform resource locator – сетевой адрес ресурса
YAML	Yet another markup language – язык разметки