

# Sparda Sandbox-API

---

Entwicklerportal

---

Version 3.0.15.1, 18.04.2024

## Inhalt

1	Einleitung und erste Schritte .....	3
2	Konfigurationseinstellungen .....	4
2.1	Aktuelle XS2A-Konfiguration (Bank Profil) .....	4
2.2	Technische Konfiguration der XS2A-API .....	4
3	Anwendungsfälle mit Nutzerdaten .....	7
3.1	Simulation der Nutzerauthentifizierung (SCA).....	7
3.2	Zahlungsinittierung .....	7
3.3	Zahlungsinittierung ohne IBAN .....	11
3.4	Löschen eines Payments (Terminüberweisung oder Dauerauftrag).....	12
3.5	Payment abfragen .....	16
3.6	Erstellung eines Global/Dedicated Consent.....	16
3.7	Consent Löschung.....	21
3.8	Abfragen von Kontodaten.....	21

# 1 Einleitung und erste Schritte

---

Die [Payment Service Directive 2 \(PSD2\)](#) schreibt Banken einen einheitlichen Standard für eine Account-Schnittstelle, erreichbar für Drittanbieter, vor. Im weiteren Text wird diese Account-Schnittstelle als "XS2A" bezeichnet, was für Access-to-Account steht. Der Produktivgang dieser Live-Schnittstelle war für September 2019 vorgeschrieben. XS2A setzt sich aus folgenden Banking-Funktionalitäten zusammen: Zahlungsinittierung (PIS = Payment Initiation Service), Kontodatenabfrage (AIS = Account Information Service) und Abfrage von Zahlungsfähigkeit (FCS = Funds Confirmation Service). Um die Einhaltung der Frist zu gewährleisten und Fehlerbehebungen und Anpassungen berücksichtigen zu können, verpflichtete PSD2 die Banken eine Testversion der Schnittstelle bereits im März 2019 zur Verfügung zu stellen.

Zentrale Komponente der PSD2 Sandbox ist die XS2A-Schnittstelle, die gemäß der Spezifikation der [Berlin Group](#) (Version 1.3) implementiert wurde und auf Testdaten basiert. Neben der Schnittstelle ist es für Banken ebenfalls vorgeschrieben, eine kostenfreie technische Dokumentation bereitzustellen, die über unterstützte Produkte und Zahlungsservices informiert.

Für Drittanbieter ist es notwendig, sich bei der BAFIN zu registrieren und sich von einem Treuhanddienstleister (TSP = Trust Service Provider) ein [eIDAS](#) Zertifikat (ein sog. QWAC) ausstellen zu lassen, um XS2A-Funktionalitäten bei Banken nutzen zu können. Ein QWAC ist Bestandteil von eIDAS und ist möglicherweise besser bekannt als [X.509-Zertifikat](#). Für PSD2-Zwecke wird das Zertifikat anhand des QcStatement um Inhalte wie die Rolle des Drittanbieters erweitert (Siehe auch [ETSI](#)).

## Erste Schritte:

Um XS2A-Funktionalitäten testen zu können, führen Sie die folgenden Schritte aus:

- Beantragen Sie das benötigte QWAC Zertifikat bei einem TSP und fügen Sie die .pem und .key Dateien im Header Ihres Requests ein.
- In den folgenden Kapiteln finden Sie eine genaue Beschreibung der relevantesten Anwendungsfälle.

## Übersicht der unterstützten Institute:

- Sparda-Bank Baden-Württemberg eG (BIC: GENODEF1S02)
- Sparda-Bank Hessen eG (BIC: GENODEF1S12)
- Sparda-Bank München (BIC: GENODEF1S04)
- Sparda-Bank Nürnberg (BIC: GENODEF1S06)
- Sparda-Bank West (BLZ: GENODED1SPW, GENODED1SPE, GENODED1SPK, GENODEF1S08)

## 2 Konfigurationseinstellungen

---

### 2.1 Aktuelle XS2A-Konfiguration (Bank Profil)

- SCA-Approach: Redirect
- Payment-Types
  - Single (sepa-credit-transfers)
  - Future-Dated (sepa-credit-transfers)
  - Periodic (sepa-credit-transfers)
- Confirmation of Funds: Ja
- Unterstützte Consents
  - Dedicated: Ja
  - Bank-Offered: Nein
  - Global: Ja
  - Available Accounts: Nein
- Deaktivierte Features:
  - Signing Basket
  - Bulk Payments
  - Delta-Reports
  - Multi-Level SCA

### 2.2 Technische Konfiguration der XS2A-API

setting:

ais:

consentTypes:

bankOfferedConsentSupported: false

globalConsentSupported: true

availableAccountsConsentSupported: false

accountAccessFrequencyPerDay: 5

notConfirmedConsentExpirationTimeMs: 270000

maxConsentValidityDays: 90

accountOwnerInformationSupported: false

redirectLinkToOnlineBanking:

aisRedirectUriToAspsp: '{host-uri}/oauth2/authorize?bic={bic}&client\_id={client-id}&redirect\_uri={redirect-uri}&response\_type={response-type}&scope={scope}&code\_challenge\_method={code-challenge-method}&code\_challenge={code-

```
challenge}'
  transactionParameters:
    availableBookingStatuses:
      - BOOKED
      - PENDING
      - INFORMATION
      - BOTH
      - ALL
    transactionsWithoutBalancesSupported: true
    supportedTransactionApplicationTypes:
      - application/json
    deltaReportSettings:
      entryReferenceFromSupported: false
      deltaListSupported: false
    scaRequirementsForOneTimeConsents:
      scaByOneTimeAvailableAccountsConsentRequired: true
      scaByOneTimeGlobalConsentRequired: true
  pis:
    supportedPaymentTypeAndProductMatrix:
      SINGLE:
        - sepa-credit-transfers
      PERIODIC:
        - sepa-credit-transfers
    maxTransactionValidityDays: 0
    notConfirmedPaymentExpirationTimeMs: 270000
    paymentCancellationAuthorisationMandated: true
    redirectLinkToOnlineBanking:
      pisRedirectUriToAspsp: '{host-url}/oauth2/authorize?bic={bic}&client_id={client-
id}&redirect_uri={redirect-uri}&response_type={response-
type}&scope={scope}&code_challenge_method={code-challenge-method}&code_challenge={code-
challenge}'
      pisPaymentCancellationRedirectUriToAspsp: >-
        {host-url}/oauth2/authorize?bic={bic}&client_id={client-id}&redirect_uri={redirect-
uri}&response_type={response-type}&scope={scope}&code_challenge_method={code-challenge-
method}&code_challenge={code-challenge}
```

paymentCancellationRedirectUrlExpirationTimeMs: 270000  
countryValidationSupported: DE  
supportedTransactionStatusFormats:  
- application/json  
piis:  
piisConsentSupported: ASPSP\_CONSENT\_SUPPORTED  
sb:  
signingBasketSupported: false  
common:  
scaApproachesSupported:  
- REDIRECT  
scaRedirectFlow: OAUTH  
oauthConfigurationUrl: '{host-url}/idp/'  
startAuthorisationMode: auto  
tppSignatureRequired: false  
psuInInitialRequestMandated: false  
redirectUrlExpirationTimeMs: 270000  
authorisationExpirationTimeMs: 270000  
forceXs2aBaseLinksUrl: true  
xs2aBaseLinksUrl: 'https://api.sparda.de/xs2a/3.0.0'  
supportedAccountReferenceFields:  
- IBAN  
multicurrencyAccountLevelSupported: SUBACCOUNT  
aisPisSessionsSupported: false  
checkTppRolesFromCertificateSupported: true  
aspspNotificationsSupported:  
- NONE  
authorisationConfirmationRequestMandated: false  
authorisationConfirmationCheckByXs2a: true  
checkUriComplianceToDomainSupported: true  
tppUriComplianceResponse: REJECT  
psuInInitialRequestIgnored: true

## 3 Anwendungsfälle mit Nutzerdaten

---

### 3.1 Simulation der Nutzerauthentifizierung (SCA)

Eine starke Nutzerauthentifizierung ist für Single Payments, die Löschung von Payments sowie die Anlage von Consents vorgeschrieben. Diese Nutzerauthentifizierung wird in der weiteren Dokumentation als SCA (Strong Customer Authentication) bezeichnet.

Für die folgenden Testfälle steht der Testkunde „PSU Successful“ mit der IBAN "DE29799905990000603740" und der BIC „TEST7999“ zur Verfügung. Die Benutzerkennung für den Testkunden lautet: 603740. Passwort und TAN sind in diesem Fall für unsere Sandbox frei wählbar und werden nicht geprüft.

### 3.2 Zahlungsinitiierung

Um ein Payment zu initiieren, ersetzen Sie die Debitor-IBAN in Ihrem Request entsprechend mit der IBAN des Testkunden. Um das Payment zu autorisieren, benutzen Sie den Redirect-Link aus der Response der Payment-Initiierung.

Der nachfolgende Code beschreibt einen beispielhaften cURL Command, der ein Single Payment für den PSU "PSU-Successful" anlegt:

#### Ablaufbeschreibung:

##### 1.) Aufruf Sandbox XS2A-API:

##### Beispiel-Request:

```
curl --location --request POST
'https://api-mock.sparda.de/mock/3.0.0/v1/payments/sepa-credit-transfers' \
\
--header 'X-Request-ID: 5a151c54-6ff5-47c0-a70d-dea8d7876335' \
--header 'Accept: application/json' \
--header 'Content-Type: application/json' \
--header 'PSU-IP-Address: 192.168.1.2' \
--header 'tpp-redirect-uri: https://tppapi.sopra-ft.com' \
--header 'X-BIC: TEST7999' \

--data-raw '{
  "debtorAccount": {
    "currency": "EUR",
    "iban": "DE29799905990000603740"
  },
  "instructedAmount": {
    "currency": "EUR",
    "amount": "612.00"
  }
}
```

```

    },
    "creditorAccount": {
      "currency": "EUR",
      "iban": "DE76799905990000603767"
    },
    "creditorName": "Max Mustermann",
    "remittanceInformationUnstructured": "Demo-Überweisung in der
Sandbox",
    "requestedExecutionDate": "2021-09-30"
  }'

```

In der Response wird der IDP-Link für die Kunden-Autorisierung zurückgegeben:

#### Beispiel-Response:

```

{
  "transactionStatus": "RCVD",
  "paymentId":
  "O-hbq9zocm9Eu2MX0MUk9q_U3MWZ0wW_Z9EoENjV0cIDJ1HsOUMgfx3GSrJeW4bEC6eRMBmFY
  YCEfmx6GgWLqs8E8ZcgkupoAptjViDqg52c=_psGLvQpt9Q",
  "_links": {
    "scaRedirect": {
      "href":
      "https://idp-mock.sparda-n.de.schulung.sparda.de/oauth2/authorize?bic=TES
      T7999&client_id=PSDDE-BAFIN-TEST&redirect_uri=https://tppapi.sopra-
      ft.com&response_type=code&scope=PIS:tx-
      e6d1112b002732b2c0c0db0cc80134747f1adb1c22c777405b34fabcc790d90c&code_cha
      llenge_method=S256&code_challenge=lRPHsFD6rWW1zJlodyWWMrdV0K9uY29EXe_L7ZM
      _SZc"
    },
    "self": {
      "href":
      "https://api-mock.sparda.de/mock/3.0.0/v1/payments/sepa-credit-transfers/
      O-hbq9zocm9Eu2MX0MUk9q_U3MWZ0wW_Z9EoENjV0cIDJ1HsOUMgfx3GSrJeW4bEC6eRMBmFY
      CEFmx6GgWLqs8E8ZcgkupoAptjViDqg52c=_psGLvQpt9Q"
    },
    "status": {
      "href":
      "https://api-mock.sparda.de/mock/3.0.0/v1/payments/sepa-credit-transfers/
      O-hbq9zocm9Eu2MX0MUk9q_U3MWZ0wW_Z9EoENjV0cIDJ1HsOUMgfx3GSrJeW4bEC6eRMBmFY
      CEFmx6GgWLqs8E8ZcgkupoAptjViDqg52c=_psGLvQpt9Q/status"
    },
    "scaStatus": {
      "href":
      "https://api-mock.sparda.de/mock/3.0.0/v1/payments/sepa-credit-transfers/
      O-hbq9zocm9Eu2MX0MUk9q_U3MWZ0wW_Z9EoENjV0cIDJ1HsOUMgfx3GSrJeW4bEC6eRMBmFY
      CEFmx6GgWLqs8E8ZcgkupoAptjViDqg52c=_psGLvQpt9Q/authorisations/487119c6-
      6cab-4abe-83d5-3d1f3da0a6dd"
    }
  }
}

```

Für die hier in der Sandbox vorbelegte code\_challenge wurde folgender code\_verifier hinterlegt:

„N6WgAgTXVwLUca7mIPIEDmYJuccOqXSJq9Wf95ul1ZFn253J6orTxdUAOW4RxPEO2Ktwe75nKeQ  
pUxZ0vCdLvr4Plzwn8aVcJEZoOjaq4EH4XcBO6Dx1Nt3CzCjp0gyK“



## 2.) Login des Kunden und Auswahl des Freigabeverfahrens:

Durch Aufruf des IDP-Link aus der Response können Sie sich mit den Logindaten des PSU Successful autorisieren.

### Beispiel-Request:

```
curl --location --request GET
'https://idp-mock.sparda-n.de.schulung.sparda.de/oauth2/authorize?bic=TES
T7999&client_id=PSDDE-BAFIN-TEST&redirect_uri=https://tpp.api.sopra-ft.co
m&response_type=code&scope=PIS:tx-e6d1112b002732b2c0c0db0cc80134747f1adb1
c22c777405b34fabcc790d90c&code_challenge_method=S256&code_challenge=lRPHs
FD6rWW1zJlodyYWMRdV0K9uY29EXe_L7ZM_SZc'
```



## Anmeldung



## Freigabeverfahren auswählen

Mögliche Verfahren:

- SecureApp
- chipTAN
- mobileTAN

3.) Anzeige der Payment Daten am IDP und Bestätigung mit der TAN des PSU Successful:

## mobileTAN Freigabe

### Überweisung anlegen durch Sopra Financial Technology GmbH

Der beauftragte Betrag wird an den angegebenen Empfänger überwiesen.

Verwendungszweck	Demo-Überweisung in der Sandbox
Empfängerkonto	DE76799905990000603767
Betrag	612.00 EUR
Ausführungsdatum	2021-09-30
Empfängername	Max Mustermann
Auftraggeberkonto	DE29799905990000603740

mobileTAN

**Freigeben**

4.) Bei erfolgreicher Eingabe erfolgt ein automatischer Redirect des Kunden in die APP des TPP; an der URL hängt der Autorisierungscode:

<https://tppapi.sopra-ft.com/?code=tac-4d9c367630ff8418bc991a03f44ed2597dcaa8a12f7f1d1c1adddd017bf36268>

5.) Aufruf des Token-Endpunkts am IDP mit dem Autorisierungscode:

#### Beispiel-Request:

```
curl --location --request POST 'https://idp-mock.sparda.de/oauth2/token' \
--header 'Content-Type: application/x-www-form-urlencoded' \
--data-urlencode 'grant_type=authorization_code' \
--data-urlencode 'code=tac-4d9c367630ff8418bc991a03f44ed2597dcaa8a12f7f1d1c1adddd017bf36268' \
--data-urlencode 'redirect_uri=https://tppapi.sopra-ft.com' \
--data-urlencode 'client_id=PSDDE-BAFIN-TEST' \
--data-urlencode 'code_verifier=N6WgAgTXVwLUca7mIPleDmYjUccOqXSJq9Wf95uI1ZFn253J6orTxdUAOW4RxPEO2Ktwe75nKeQpUxZ0vCdLVr4PIzwn8aVcJEZoOjaq4EH4XcBO6Dx1Nt3CzCjp0gyK'
```

6.) In der Response erhalten Sie den Access-Token und den Refresh-Token:

**Beispiel-Response:**

```
{
  "access_token": "tat-
68c780b1b33b4fa91718ee5d8c7e1a039dea8069232efe4f087ebe96232a20ea",
  "refresh_token": "trt-
9b52b0e507c9e05216bac0a17d25d008cdabef8ce04831d375b62d465e122208",
  "scope": "PIS:tx-
e6d1112b002732b2c0c0db0cc80134747f1adb1c22c777405b34fabcc790d90c",
  "token_type": "Bearer",
  "expires_in": 300
}
```

PSU-Beschreibung	IBAN	SCA Status	Transaction Status
PSU-Successful	DE29799905990000603740	finalised	ACTC/ACSC*

(\*) Der Status ist abhängig von dem Payment-Typen. Ein Single Payment erhält nach seiner Ausführung den Transaction Status "executed" in der Sandbox. Ein Future-Dated Payment wird ausgeführt, wenn das definierte Datum "requestedExecutionDate" erreicht wurde. Ein ähnliches Verhalten ist für Periodic Payments implementiert, die vom Enddatum "endDate" abhängen.

### 3.3 Zahlungsinittierung ohne IBAN

Eine Zahlungsinittierung ohne Angabe einer IBAN ist ebenfalls möglich. Folgen Sie der Ablaufbeschreibung unter 3.2 zur Zahlungsinittierung. Nach erfolgreichem Login wird zusätzlich eine Kontoliste angezeigt, wo im Falle von mehr als einem Girokonto die Auswahl der korrekten IBAN des PSU Successful zu erfolgen hat. Anschließend erfolgt die SCA und weitere Schritte wie im vorhergehenden Absatz beschrieben.

**Beispiel-Request:**

```
curl --location --request POST
'https://api-mock.sparda.de/mock/3.0.0/v1/payments/sepa-credit-transfers'
\
--header 'X-Request-ID: 5a151c54-6ff5-47c0-a70d-dea8d7876335' \
--header 'Accept: application/json' \
--header 'Content-Type: application/json' \
--header 'PSU-IP-Address: 192.168.1.2' \
--header 'tpp-redirect-uri: https://tppapi.sopra-ft.com' \
--header 'X-BIC: TEST7999' \
```

```
--data-raw '{
  "debtorAccount": {},
  "instructedAmount": {
    "currency": "EUR",
    "amount": "612.00"
  },
  "creditorAccount": {
    "currency": "EUR",
    "iban": "DE76799905990000603767"
  },
  "creditorName": "Max Mustermann",
  "remittanceInformationUnstructured": "Demo-Überweisung in der
Sandbox",
  "requestedExecutionDate": "2021-09-30"
}'
```



## Kontoauswahl

Bitte Konto auswählen:

- (DE29799905990000603740)
- (DE29799905990100603740)
- (DE29799905990200603740)

**Auswählen**

### 3.4 Löschen eines Payments (Terminüberweisung oder Dauerauftrag)

Um ein Payment zu löschen, fügen Sie Ihre Payment-Id in den Delete Payment Endpoint ein. Das Löschen eines Payments benötigt den *expliziten Start des Autorisierungsvorgangs*. Das bedeutet, dass dem startAuthorization-Link in der Response gefolgt werden muss, um die Autorisierung zu starten. Benutzen Sie danach den SCA Redirect-Link wie im vorherigen Absatz beschrieben.

#### Ablaufbeschreibung:

1.) Aufruf XS2A-Sandbox-API:

#### Beispiel-Request:

```
curl --location --request DELETE
'https://api-mock.sparda.de/mock/3.0.0/v1/payments/sepa-credit-transfers/
O-hbq9zocm9Eu2MX0MUk9q_U3MWZ0wW_Z9EoENjV0cIDJ1HsOUMgfox3GSrJeW4bEC6eRMBmFY
CEfmx6GgWLqs8E8ZcggkupoAptjViDqg52c=_=_psGLvQpt9Q' \
```

```
--header 'X-Request-ID: 3bceb0ab-2670-4460-9703-c836293cf46f' \  
--header 'TPP-Redirect-URI: https://tppapi.sopra-ft.com' \  
--header 'X-BIC: TEST7999'
```

In der Response wird der IDP-Link für die Kunden-Autorisierung zurückgegeben:

#### Beispiel-Response:

```
{  
  "transactionStatus": "ACSP",  
  "_links": {  
    "scaRedirect": {  
      "href":  
"https://idp-mock.sparda-n.de.schulung.sparda.de/oauth2/authorize?bic=TES  
T7999&client_id=PSDDE-BAFIN-TEST&redirect_uri=https://tppapi.sopra-ft.com  
&response_type=code&scope=PIS:tx-d634a632ca211edab4669093035f28c293dff626  
b83a144868699d044a13c35a&code_challenge_method=S256&code_challenge=lRPHsF  
D6rWW1zJlodkYWMMrdV0K9uY29EXe_L7ZM_SZc"  
    },  
    "self": {  
      "href":  
"https://api-mock.sparda.de/mock/3.0.0/v1/payments/sepa-credit-transfers/  
O-hbq9zocm9Eu2MX0MUK9q_U3MWZ0wW_Z9EoENjV0cIDJ1HsOUMgfx3GSrJew4bEC6eRMBmFY  
CEFmx6GgWLqs8E8ZcgkupoAptjViDqg52c=_psGLvQpt9Q"  
    },  
    "status": {  
      "href":  
"https://api-mock.sparda.de/mock/3.0.0/v1/payments/sepa-credit-transfers/  
O-hbq9zocm9Eu2MX0MUK9q_U3MWZ0wW_Z9EoENjV0cIDJ1HsOUMgfx3GSrJew4bEC6eRMBmFY  
CEFmx6GgWLqs8E8ZcgkupoAptjViDqg52c=_psGLvQpt9Q/status"  
    },  
    "scaStatus": {  
      "href":  
"https://api-mock.sparda.de/mock/3.0.0/v1/payments/sepa-credit-transfers/  
O-hbq9zocm9Eu2MX0MUK9q_U3MWZ0wW_Z9EoENjV0cIDJ1HsOUMgfx3GSrJew4bEC6eRMBmFY  
CEFmx6GgWLqs8E8ZcgkupoAptjViDqg52c=_psGLvQpt9Q/cancellation-authorisati  
ons/23dbcf5c-c9b2-4a47-8d49-3bd8845d5a17"  
    }  
  }  
}
```

#### 2.) Login des Kunden:

In der Response des Aufrufs erhalten Sie einen Link, über den Sie den Kunden an den IDP weiterleiten. Hier können Sie sich mit den Logindaten des PSU Successful autorisieren.

#### Beispiel-Request:

```
curl --location --request GET  
'https://idp-mock.sparda-n.de.schulung.sparda.de/oauth2/authorize?bic=TES  
T7999&client_id=PSDDE-BAFIN-TEST&redirect_uri=https://tppapi.sopra-ft.com  
&response_type=code&scope=PIS:tx-79da42d0d3ecd20bcae93bfc60cd16436f203d38  
f61b8dc3ad7501df7e8fe73b&code_challenge_method=S256&code_challenge=lRPHsF  
D6rWW1zJlodkYWMMrdV0K9uY29EXe_L7ZM_SZc'
```

**Sparda-Bank**

## Anmeldung

603740

.....

**Anmelden**

3.) Anzeige der Payment Daten am IDP und Bestätigung mit der TAN des PSU Successful:

**Sparda-Bank**

## mobileTAN Freigabe

**Überweisung löschen durch Sopra Financial Technology GmbH**

Die Überweisung wird gelöscht.

Verwendungszweck	Demo-Überweisung in der Sandbox
Empfängerkonto	DE76799905990000603767
Betrag	612.00 EUR
Ausführungsdatum	2021-09-30
Empfängername	Max Mustermann
Auftraggeberkonto	DE29799905990000603740

mobileTAN

**Freigeben**

4.) Bei erfolgreicher Eingabe erfolgt ein automatischer Redirect des Kunden in die APP des TPP; an der URL hängt der Autorisierungscode:

```
https://tppapi.sopra-ft.com/?code=tac-4ec9bb45d4dfae7cb685f8e5b584078b616417872130597f8b1abc9bbc290470
```

5.) Aufruf des Token-Endpunkts am IDP mit dem Autorisierungscode:

**Beispiel-Request:**

```
curl --location --request POST 'https://idp-mock.sparda.de/oauth2/token' \
--header 'Content-Type: application/x-www-form-urlencoded' \
--data-urlencode 'grant_type=authorization_code' \
--data-urlencode 'code=tac-4ec9bb45d4dfae7cb685f8e5b584078b616417872130597f8b1abc9bbc290470' \
--data-urlencode 'redirect_uri=https://tppapi.sopra-ft.com' \
--data-urlencode 'client_id=PSDDE-BAFIN-TEST' \
--data-urlencode 'code_verifier=N6WgAgTXVwLUca7mIP1EDmYjUccOqXSJq9Wf95uI1ZFfn253J6orTxdUAOW4RxPEO2Ktwe75nKeQpUxZ0vCdLVr4PIzwn8aVcJEZoOjaq4EH4XcBO6Dx1Nt3CzCjp0gyK'
```

6.) In der Response erhalten Sie den Access-Token und den Refresh-Token:

**Beispiel-Response:**

```
{
  "access_token": "tat-
c9822797b090cf3258943bb6f20e1c47eda6ed2169b18d535e7607f0fe0f61b5",
  "refresh_token": "trt-
1ba4717db30ad5ff93d5f783f7faf87cefc04ca40ca1f8572c266ab1013b774b",
  "scope": "PIS:tx-
d634a632ca211edab4669093035f28c293dff626b83a144868699d044a13c35a",
  "token_type": "Bearer",
  "expires_in": 300
}
```

PSU-Beschreibung	IBAN	SCA Status	Transaction Status
PSU-Successful	DE29799905990000603740	finalised	CANCEL*

(\*) Es ist nur möglich Payments zu löschen, die noch nicht ausgeführt wurden. Da Single Payments direkt ausgeführt werden, ist eine Löschung nur bei Future-Dated Payments oder Periodic Payments möglich.

### 3.5 Payment abfragen

Um Zahlungsdaten abzufragen, fügen Sie Ihre Payment-Id in den GET Payment Data Endpoint ein.

**Beispiel-Request:**

```
curl --location --request GET
'https://api-mock.sparda.de/mock/3.0.0/v1/payments/sepa-credit-transfers/
O-hbq9zocm9Eu2MX0MUk9q_U3MWZ0wW_Z9EoENjV0cIDJ1HsOUMgffx3GSrJeW4bEC6eRMBmFY
CEFmx6GgWLqs8E8ZcgkupoAptjViDqg52c=_psGLvQpt9Q' \

--header 'X-Request-ID: 02ee5aec-88bd-46d7-a040-b26867d767ab' \
--header 'Accept: application/json' \
--header 'Content-Type: application/json' \
--header 'PSU-IP-Address: 192.168.1.2' \
--header 'Authorization: Bearer tat-
68c780b1b33b4fa91718ee5d8c7e1a039dea8069232efe4f087ebe96232a20ea'
```

Um den Transaction Status eines Payments abzufragen, fügen Sie Ihre Payment-Id in den GET Payment Status Endpoint ein.

**Beispiel-Request:**

```
curl --location --request GET
'https://api-mock.sparda.de/mock/3.0.0/v1/payments/sepa-credit-transfers/
O-hbq9zocm9Eu2MX0MUk9q_U3MWZ0wW_Z9EoENjV0cIDJ1HsOUMgffx3GSrJeW4bEC6eRMBmFY
CEFmx6GgWLqs8E8ZcgkupoAptjViDqg52c=_psGLvQpt9Q/status' \
--header 'X-Request-ID: 37c3fd04-c38c-4030-8dcd-ee72c4ae69c2' \
--header 'Authorization: Bearer
tat-c9822797b090cf3258943bb6f20e1c47eda6ed2169b18d535e7607f0fe0f61b5'
```

PSU-Beschreibung	IBAN	SCA Status	Transaction Status
PSU-Successful	DE29799905990000603740	finalised	ACTC/ACSC*

### 3.6 Erstellung eines Global/Dedicated Consent

Um Accounts abfragen zu können, ist es vorab notwendig, einen sogenannten Dedicated oder Global Consent anzulegen. Dabei handelt es sich um eine Art Einverständniserklärung mit einer definierten Gültigkeit, die es erlaubt, je nach Art des Consents Accounts für einen festgelegten Gültigkeitszeitraum abzufragen.



## Ablaufbeschreibung:

### 1.) Aufruf Sandbox XS2A-API:

#### Beispiel-Request:

```
curl --location --request POST
'https://api-mock.sparda.de/mock/3.0.0/v1/consents' \

--header 'X-Request-ID: 0b5236ef-d0bd-42c9-85ad-2ddd7e66f55b' \
--header 'Accept: application/json' \
--header 'Content-Type: application/json' \
--header 'PSU-IP-Address: 192.168.1.2' \
--header 'TPP-Redirect-URI: https://tppapi.sopra-ft.com' \
--header 'TPP-Nok-Redirect-URI: https://tppapi.sopra-ft.com' \
--header 'X-BIC: TEST7999' \

--data-raw '{
  "access": {
    "allPsd2": "allAccounts"
  },
  "recurringIndicator": true,
  "validUntil": "2021-12-31",
  "frequencyPerDay": 4
}'
```

In der Response wird der IDP-Link für die Kunden-Autorisierung zurückgegeben:

#### Beispiel-Response:

```
{
  "consentStatus": "received",
  "consentId":
  "SpU4KWjuP-d0LQYhfkOvWbZh0pR9KgbPbVruVLG3avPM6NEvGGs9k8ZrrGTt6SuoEggjY9Wf
  5NJUUUpjYiwd6Z3eGDd8_WXXE6Y4C13Cr9H8=_psGLvQpt9Q",
  "_links": {
    "scaRedirect": {
      "href":
      "https://idp-mock.sparda-n.de.schulung.sparda.de/oauth2/authorize?bic=TES
      T7999&client_id=PSDDE-BAFIN-TEST&redirect_uri=https://tppapi.sopra-ft.com
      &response_type=code&scope=AIS:tx-7bc6419c1fc14dea02945fdeea4c621a0d98371f
      fe393f8039ea89ealfaeae9a&code_challenge_method=S256&code_challenge=lRPHsF
      D6rWWlzlJlodkYWMRdV0K9uY29EXe_L7ZM_Szc"
    },
    "self": {
      "href":
      "https://api-mock.sparda.de/mock/3.0.0/v1/consents/SpU4KWjuP-d0LQYhfkOvWb
      Zh0pR9KgbPbVruVLG3avPM6NEvGGs9k8ZrrGTt6SuoEggjY9Wf5NJUUUpjYiwd6Z3eGDd8_WXX
      E6Y4C13Cr9H8=_psGLvQpt9Q"
    },
    "status": {
      "href":
      "https://api-mock.sparda.de/mock/3.0.0/v1/consents/SpU4KWjuP-d0LQYhfkOvWb
```

```
Zh0pR9KgbPbVruVLG3avPM6NEvGGs9k8ZrrGTt6SuoEggjY9Wf5NJUUUpjYiwd6Z3eGDd8_WXX
E6Y4C13Cr9H8=_psGLvQpt9Q/status"
    },
    "scaStatus": {
      "href":
"https://api-mock.sparda.de/mock/3.0.0/v1/consents/SpU4KWjuP-d0LQYhfkOvWb
Zh0pR9KgbPbVruVLG3avPM6NEvGGs9k8ZrrGTt6SuoEggjY9Wf5NJUUUpjYiwd6Z3eGDd8_WXX
E6Y4C13Cr9H8=_psGLvQpt9Q/authorisations/7460edd1-d638-4072-82c9-302148d
e9799"
    }
  }
}
```

## 2.) Login des Kunden:

In der Response des Aufrufs erhalten Sie einen Link, über den Sie den Kunden an den IDP weiterleiten. Hier können Sie sich mit den Logindaten des PSU Successful autorisieren.

### Beispiel-Request:

```
curl --location --request GET
'https://idp-mock.sparda-n.de.schulung.sparda.de/oauth2/authorize?bic=TES
T7999&client_id=PSDDE-BAFIN-TEST&redirect_uri=https://tppapi.sopra-ft.com
&response_type=code&scope=AIS:tx-7bc6419c1fc14dea02945fdeea4c621a0d98371f
fe393f8039ea89ea1faeae9a&code_challenge_method=S256&code_challenge=lRPHsF
D6rWW1zJlodkaYWMRdV0K9uY29EXe_L7ZM_SZc'
```



## Anmeldung

3.) Anzeige der Consent Daten am IDP und Bestätigung mit der TAN des PSU Successful:



## mobileTAN Freigabe

**Consent anlegen durch Sopra Financial Technology GmbH**

Die Zustimmung zur automatischen Kontoabfrage wird für alle Ihre Zahlungsverkehrskonten und deren Kontoinformationen (allgemeine Kontoinformationen, Saldo, Umsätze) erteilt.

gültig bis	2021-11-17
wiederholender Zugriff	ja

mobileTAN

**Freigeben**

4.) Bei erfolgreicher Eingabe erfolgt ein automatischer Redirect des Kunden in die APP des TPP; an der URL hängt der Autorisierungscode:

```
https://tppapi.sopra-ft.com/?code=tac-56a36d3ac308795077864240049a150b6c905f584a8f8c2b1b25c3aefe841b24
```

5.) Aufruf des Token-Endpunkts am IDP mit dem Autorisierungscode:

**Beispiel-Request:**

```
curl --location --request POST 'https://idp-mock.sparda.de/oauth2/token' \
--header 'Content-Type: application/x-www-form-urlencoded' \
--data-urlencode 'grant_type=authorization_code' \
--data-urlencode 'code=tac-56a36d3ac308795077864240049a150b6c905f584a8f8c2b1b25c3aefe841b24' \
--data-urlencode 'redirect_uri=https://XS2ATEST.produktion.api.sopra-ft.com' \
--data-urlencode 'client_id=PSDDE-BAFIN-TEST' \
--data-urlencode 'code_verifier=N6WgAgTXVwLUca7mIPleDmYjUccOqXSJq9Wf95uI1ZFn253J6orTxdUAOW4RxPEO2Ktwe75nKeQpUxZ0vCdLvr4PIzwn8aVcJEZoOjaq4EH4XcBO6Dx1Nt3CzCjp0gyK'
```

6.) In der Response erhalten Sie den Access-Token und den Refresh-Token:

**Beispiel-Response:**

```
{
  "access_token": "tat-
09032fbfa220b40e6ac666afdceelac563c1294de7b74bfbfd52e9d45d66f1208",
  "refresh_token": "trt-
a79f30547fa1762c821cbeb7eb0699ab4e9447f8b5108948f87300b8519c059f",
  "scope": "AIS:tx-
7bc6419c1fc14dea02945fdeea4c621a0d98371ffe393f8039ea89ea1faeae9a",
  "token_type": "Bearer",
  "expires_in": 300
}
```

Um einen Dedicated Consent anzulegen, ersetzen Sie die IBAN in Ihrem Request mit der IBAN des genannten Testkunden. Den Consent können Sie, wie im vorherigen Absatz beschrieben, mit dem SCA Redirect-Link autorisieren.

Der nachfolgende Code beschreibt einen beispielhaften cURL Command, der einen Dedicated Consent für PSU Successful anlegt:

**Beispiel-Request:**

```
curl --location --request POST
'https://api-mock.sparda.de/mock/3.0.0/v1/consents' \

--header 'X-Request-ID: 729d3568-75a4-4a08-b73c-2911a9718552' \
--header 'Accept: application/json' \
--header 'Content-Type: application/json' \
--header 'PSU-IP-Address: 192.168.1.2' \
--header 'TPP-Redirect-URI: https://tppapi.sopra-ft.com' \
--header 'TPP-Nok-Redirect-URI: https://tppapi.sopra-ft.com' \
--header 'X-BIC: TEST7999' \

--data-raw '{
  "access": {
    "accounts": [
      {
        "iban": "DE29799905990000603740",
        "currency": "EUR"
      }
    ],
    "balances": [
      {
        "iban": "DE29799905990000603740",
        "currency": "EUR"
      }
    ],
    "transactions": [
      {
        "iban": "DE29799905990000603740",
        "currency": "EUR"
      }
    ]
  },
  "recurringIndicator": false,
  "validUntil": "2021-12-31",
  "frequencyPerDay": 4
}'
```

PSU-Beschreibung	IBAN	SCA Status	Transaction Status
PSU-Successful	DE29799905990000603740	finalised	ACTC/ACSC*

### 3.7 Consent Löschung

Um einen Consent zu löschen, fügen Sie Ihre Consent-ID in den Delete Consent Endpoint ein. Um die Autorisierung der Löschung durchzuführen, folgen Sie der Ablaufbeschreibung aus 3.6 Erstellung eines global/dedicated Consents.

#### Beispiel-Request:

```
curl --location --request DELETE
'https://api-mock.sparda.de/mock/3.0.0/v1/consents/SpU4KWjuP-d0LQYhfkOvWb
Zh0pR9KgbPbVruVLG3avPM6NEvGGs9k8ZrrGTt6SuoEggjY9Wf5NJUUpjYiwd6Z3eGDd8_WXX
E6Y4C13Cr9H8=_=_psGLvQpt9Q' \

--header 'X-Request-ID: 4f6208b3-92ff-48d8-8e6a-a58e5ce66089' \
--header 'PSU-IP-Address: 192.168.1.2' \
--header 'Authorization: Bearer tat-
09032fbfa220b40e6ac666afdcee1ac563c1294de7b74bfbfd52e9d45d66f1208'
```

PSU-Beschreibung	IBAN	Consent Status
PSU-Successful	DE29799905990000603740	terminatedByTpp

### 3.8 Abfragen von Kontodaten

Um Kontodaten abzufragen, ersetzen Sie die IBAN in Ihrem Request mit der IBAN des genannten Testkunden.

Der nachfolgende Code beschreibt einen beispielhaften cURL Command, der alle Kontodaten des PSU "PSU-Successful" abfragt:

#### Beispiel-Request:

```
curl --location --request GET
'https://api-mock.sparda.de/mock/3.0.0/v1/accounts' \
--header 'x-request-id: cd6ce532-6829-4db4-a289-fa3b274e1ed0' \
```

```

--header 'consent-id:
0uit_zyWfXTZRaKcrNl7xgOKXWRcYwPQFboiahZ3C1NfkTvWUgzN_PglWdAe4vpvkvTCsNVwv
gRgBxBSSZ-H53eGDd8_WXXE6Y4C13Cr9H8=_=_psGLvQpt9Q' \

--header 'Content-Type: application/json' \
--header 'PSU-IP-Address: 192.168.1.2' \
--header 'Authorization: Bearer tat-
8a57ecbbd5fe21d552e5ceb5d372d9aae6898ad30f6d4e1350c708a279945f6c'

```

Der nachfolgende Code beschreibt eine beispielhafte Response für eine erfolgreiche Abfrage der Kontodaten:

### Beispiel-Response:

```

{
  "accounts": [
    {
      "resourceId": "9070c27c2d953c8237c2c803f77ac54026236d0a",
      "iban": "DE29799905990000603740",
      "currency": "EUR",
      "name": "",
      "product": "GIRO",
      "cashAccountType": "CACC",
      "status": "enabled",
      "bic": "TEST7999",
      "usage": "PRIV",
      "_links": {
        "balances": {
          "href":
"https://api-mock.sparda.de/mock/3.0.0/v1/accounts/9070c27c2d953c8237c2c8
03f77ac54026236d0a/balances"
        },
        "transactions": {
          "href":
"https://api-mock.sparda.de/mock/3.0.0/v1/accounts/9070c27c2d953c8237c2c8
03f77ac54026236d0a/transactions"
        }
      },
      "ownerName": "Max Mustermann"
    }
  ]
}

```

PSU-Beschreibung	IBAN	Consent Status
PSU-Successful	DE29799905990000603740	valid