Технічні характеристики спеціалізованих інтерфейсів

v.1.0

Версія	Опис змін
v.1.0	Початкова версія.

У цьому документі наведено загальний опис технічних характеристик. Детальний опис відповідно до версії, яку використовує НПП знаходиться за адресою https://docs.api.upc.ua/api-services/compliance-apis

1. ЗАГАЛЬНІ ТЕХНІЧНІ ХАРАКТЕРИСТИКИ

1.1. Архітектура та стандарт

Спеціалізовані інтерфейси побудовані за принципами REST API та базуються на специфікації XS2A (Access to Account), яка реалізована через платформу відкритого банкінгу (Open Banking Platform, OBP). XS2A є стандартним інтерфейсом доступу третіх сторін до платіжних рахунків відповідно до вимог PSD2.

1.2. Транспортний протокол

Протокол: HTTPS (HTTP over TLS/SSL). Усі запити передаються виключно по захищеному каналу HTTPS. Використання незахищеного протоколу HTTP не допускається.

1.3. Формат даних

Формат тіла запитів та відповідей: JSON (JavaScript Object Notation). Заголовок `Content-Type: application/json` обов'язковий для всіх запитів, що містять тіло (POST).

1.4. Версіонування

Поточна версія обох інтерфейсів: v2. Версія зазначається у базовому шляху URL: `/pis/v2/` та `/ais/v2/`.

1.5. Обов'язкові заголовки HTTP (загальні для обох інтерфейсів)

- `X-Request-Id` — унікальний ідентифікатор запиту (UUID формат), наприклад: `dc7b16a5-4ac8-4fdc-9c4e-9f9d0387dc07`. Використовується для ідентифікації та трасування запиту.
- `Content-Type` — тип вмісту тіла запиту: `application/json` (для POST-запитів).
- `PSU-ID` — ідентифікатор користувача платіжних послуг (PSU) у системі банку.
- `PSU-IP-Address` — IP-адреса пристрою PSU.
- `Consent-ID` — ідентифікатор підтвердженої AIS-згоди (для запитів до AIS після авторизації).

- `Client-Redirect-URI` — URL для перенаправлення PSU після успішної авторизації (SCA Redirect).
- `Client-Redirect-Nok-URI` — URL для перенаправлення PSU у разі невдалої авторизації (SCA Redirect).

2. ТЕХНІЧНІ ХАРАКТЕРИСТИКИ ІНТЕРФЕЙСУ PIS (PAYMENT INITIATION SERVICE)

2.1. Базовий URL-шлях

`/pis/v2/`

2.2. Ендпоінти (кінцеві точки)

2.2.1. Ініціювання платежу

- **Метод та шлях:** `POST /pis/v2/payments/{payment-product}`
- **Параметр шляху:** `payment-product` — тип платіжного продукту (наразі підтримується: `instant-credit-transfers`).

Тіло запиту (JSON):

- `paymentIdentification.endToEndId` — наскрізний ідентифікатор платежу (рядок).
- `debtorAccount.iban` — IBAN рахунку платника.
- `debtorAccount.currency` — валюта рахунку платника (наприклад, `"UAH"`).
- `instructedAmount.currency` — валюта суми переказу.

- `instructedAmount.amount` — сума переказу (рядок у десятковому форматі, наприклад `"12.21"`).
- `creditor.name` — ім'я отримувача.
- `creditor.creditorId` — ідентифікатор отримувача.
- `creditor.creditorIdType` — тип ідентифікатора отримувача (наприклад, `"PSPT"`).
- `creditorAccount.iban` — IBAN рахунку отримувача.
- `creditorAccount.currency` — валюта рахунку отримувача.
- `remittanceInformationUnstructured` — масив рядків із призначенням платежу.

Відповідь (JSON): `paymentId` (UUID), `transactionStatus` (`"RCVD"`), `_links.startAuthorisation.href`.

2.2.2. Запуск авторизації платежу (SCA)

- **Метод та шлях:** `POST /pis/v2/payments/{payment-product}/{payment-id}/authorisations`
- **Параметри шляху:** `payment-product`, `payment-id` (отримано з попереднього кроку).
- **Заголовки:** `Client-Redirect-URI`, `Client-Redirect-Nok-URI` (обов'язкові для Redirect SCA).

Відповідь для Decoupled SCA (JSON): `scaStatus` (`"received"`), `authorisationId` (UUID), `psuMessage` (текст повідомлення для PSU).

Відповідь для Redirect SCA (JSON): `scaStatus` (`"received"`), `authorisationId` (UUID), `_links.scaRedirect.href` (URL для перенаправлення PSU на SCA-сторінку банку).

2.2.3. Перевірка статусу платежу

- **Метод та шлях:** `GET /pis/v2/payments/{payment-product}/{payment-id}/status`

- **Параметри шляху:** `payment-product`, `payment-id`.

Відповідь (JSON): `transactionStatus` — статус транзакції (наприклад, `"ACCC"` — успішно завершено).

2.3. Процедура взаємодії (PIS Flow)

1. TPP ініціює платіж (`POST /payments/{payment-product}`) → отримує `paymentId`.
2. TPP запускає авторизацію (`POST /payments/{payment-product}/{payment-id}/authorisations`) → отримує `authorisationId` та або `psuMessage` (Decoupled) або `scaRedirect` URL (Redirect).
3. PSU авторизує платіж (через застосунок банку або банківську веб-сторінку залежно від режиму SCA).
4. TPP перевіряє статус платежу (`GET /payments/{payment-product}/{payment-id}/status`) → отримує `transactionStatus`.
5. TPP інформує PSU про результат виконання платежу.

2.4. Коди помилок, специфічні для PIS

- `404` — `PRODUCT_UNKNOWN`: вказаний платіжний продукт не підтримується банком.
- `400` — `PAYMENT_FAILED`: запит на ініціювання платежу не вдалося опрацювати.
- `403` — `SERVICE_BLOCKED`: сервіс недоступний для PSU через незалежне блокування з боку банку.

3. ТЕХНІЧНІ ХАРАКТЕРИСТИКИ ІНТЕРФЕЙСУ AIS (ACCOUNT INFORMATION SERVICE)

3.1. Базовий URL-шлях

`/ais/v2/`

3.2. Ендпоінти (кінцеві точки)

3.2.1. Створення AIS-згоди

- **Метод та шлях:** `POST /ais/v2/consents/account-access`

Тіло запиту (JSON):

- `access.payments[].account.iban` — IBAN рахунку, до якого надається доступ.
- `access.payments[].rights` — масив прав доступу: `"accountDetails"`, `"balances"`, `"transactions"`. Якщо вказано `"balances"` або `"transactions"`, право `"accountDetails"` надається неявно.
- `consentType` — тип згоди: `"detailed"`.
- `recurringIndicator` — ознака повторюваного доступу: `true` (постійний) або `false` (одноразовий).
- `validTo` — дата закінчення дії згоди (формат: `YYYY-MM-DD`).
- `frequencyPerDay` — максимальна кількість запитів на добу без присутності PSU (максимум: 4).

Відповідь (JSON): `consentId` (UUID), `consentStatus` (`"received"`), `_links.startAuthorisation.href`.

3.2.2. Запуск авторизації згоди (SCA)

- **Метод та шлях:** `POST /ais/v2/consents/account-access/{consent-id}/authorisations`
- **Параметр шляху:** `consent-id` (отримано з попереднього кроку).
- **Заголовки:** `Client-Redirect-URI`, `Client-Redirect-Nok-URI`.

Відповідь для Decoupled SCA (JSON): `scaStatus`, `authorisationId`, `psuMessage`.

Відповідь для Redirect SCA (JSON): `scaStatus`, `authorisationId`, `_links.scaRedirect.href`.

3.2.3. Перевірка статусу згоди

- **Метод та шлях:** `GET /ais/v2/consents/account-access/{consent-id}/status`

Відповідь (JSON): `consentStatus` — статус згоди (`"received"`, `"valid"`, `"rejected"` тощо).

3.2.4. Отримання переліку рахунків

- **Метод та шлях:** `GET /ais/v2/accounts`
- **Параметр запиту:** `withBalance=true` (для отримання балансів у відповіді).
- **Заголовок:** `Consent-ID` — ідентифікатор підтвердженої AIS-згоди.

Відповідь (JSON): масив об'єктів `accounts` з полями: `iban`, `currency`, `resourceId`, `name`, `balances[].balanceAmount`, `balances[].balanceType`, `balances[].referenceDate`.

3.2.5. Отримання історії транзакцій

- **Метод та шлях:** `GET /ais/v2/accounts/{account-id}/transactions`

Параметри запиту:

- `dateFrom` — дата початку вибірки (без додаткового SCA доступна до 90 днів назад; для давніших транзакцій потрібна одноразова AIS-згода з `recurringIndicator: false` та правом `"transactions"`).
- `limit` — максимальна кількість записів транзакцій у відповіді (для пагінації).
- `offset` — кількість пропущених записів від початку вибірки (для пагінації).

Пагінація: TPP збільшує значення `offset` на кількість вже отриманих записів. Ознакою завершення є відповідь із кількістю записів, меншою за `limit`, або порожній список. Якщо параметри пагінації не передані, кількість повернутих транзакцій визначається конкретним банком.

3.3. Процедура взаємодії (AIS Flow)

1. TPP створює AIS-згоду (`POST /consents/account-access`) → отримує `consentId`.
2. TPP запускає авторизацію згоди (`POST /consents/account-access/{consent-id}/authorisations`) → отримує `authorisationId` та або `psuMessage` (Decoupled) або `scaRedirect` URL (Redirect).
3. PSU авторизує згоду (через застосунок банку або банківську веб-сторінку).
4. TPP перевіряє статус згоди (`GET /consents/account-access/{consent-id}/status`) до отримання статусу `"valid"`.
5. TPP отримує інформацію про рахунки та транзакції, передаючи `Consent-ID` у заголовку запиту.

3.4. Обмеження та ліміти

- Максимальна частота запитів без присутності PSU: 4 рази на добу (`frequencyPerDay ≤ 4`).
- Доступна глибина історії транзакцій без додаткового SCA: 90 календарних днів.

3.5. Коди помилок, специфічні для AIS

- `401` — `CONSENT_INVALID`: згода недійсна або не надає необхідних прав доступу.
- `403` — `CONSENT_UNKNOWN`: згода не існує.
- `429` — `ACCESS_EXCEEDED`: перевищено денний ліміт у 4 GET-запити без присутності PSU.
- `400` — `FORMAT_ERROR` (пагінація транзакцій): параметр `dateFrom` містить дату, що перевищує 90 днів у минулому, для повторюваної AIS-згоди.

4. ПРОЦЕДУРИ СИЛЬНОЇ АВТЕНТИФІКАЦІЇ КЛІЄНТА (SCA)

Обидва інтерфейси (PIS та AIS) підтримують два режими SCA:

4.1. Decoupled SCA (роз'єднана автентифікація)

Процедура: PSU отримує повідомлення або push-сповіщення від банку та авторизує операцію безпосередньо у застосунку банку, не залишаючи інтерфейс TPP. TPP отримує у відповіді поле `psuMessage` з текстом для PSU. Банк самостійно сповіщає PSU про необхідність авторизації.

4.2. Redirect SCA (автентифікація з перенаправленням)

Процедура: TPP перенаправляє PSU на URL-адресу банківської SCA-сторінки (`scaRedirect.href`), де PSU проходить автентифікацію та підтверджує операцію. Після завершення авторизації банк перенаправляє PSU назад до TPP за адресою `Client-Redirect-URI` (успіх) або `Client-Redirect-Nok-URI` (невдача).

5. ІНСТРУМЕНТИ ТА ВИМОГИ ДО СТОРОННЬОГО НПП

5.1. Реєстрація та сертифікати

Для підключення до спеціалізованих інтерфейсів сторонній НПП повинен:

- Пройти реєстрацію на Developer Portal (`portal.api.upc.ua`).
- Отримати та налаштувати цифрові сертифікати для взаємної автентифікації TLS (mTLS) — відповідно до розділу `"Certificates"` порталу.
- Налаштувати електронні підписи запитів — відповідно до розділу `"Electronic signatures"` порталу.
- Зареєструвати застосунок у розділі `"Applications"` та оформити підписку на відповідний API-сервіс.

5.2. Технічні вимоги

- HTTP-клієнт з підтримкою HTTPS/TLS та мутуальної автентифікації (mTLS).

- Підтримка формату JSON для серіалізації та десеріалізації даних.
- Здатність формувати унікальні UUID для заголовка `X-Request-Id`.
- Реалізація логіки опитування статусу (polling) для перевірки статусу платежу та згоди.
- Реалізація механізму пагінації для обходу великих наборів транзакцій (параметри `limit` та `offset`).
- Наявність публічно доступних URL-адрес для `Client-Redirect-URI` та `Client-Redirect-Nok-URI` (для Redirect SCA).

5.3. Середовища

- **Sandbox** (тестове середовище): доступне для розробки та тестування інтеграції.
- **Production** (продуктивне середовище): для реальної взаємодії з платіжними рахунками PSU.

Previous
Опис функціоналу спеціалізованих інтерфейсів

Next
Опис програми тестування