

Open Banking Payments project

- User stories for API endpoints related to 'Accounts' EPIC, covering features like 'Account List', and 'Account Details' & 'Account Nickname'.
- User stories written in GHERKIN format, depicting dummy request/response codes and error messages with dummy JSON responses for proposed API endpoints and a Sequence Diagram

EPIC: Accounts

The following user stories details on managing savings/checking accounts in an Open Banking platform.

Req ID : UserStory_1_Account list

UC_1.1_AI - Happy flow - Scenario 1: Successful Account List Retrieval

Given the user is authenticated with a valid token

When the user requests the list of accounts

Then the system should validate the token and user authorization

And the system should respond with a list of accounts

And the response should include account details such as id, name, type and app

And the response should be paginated to handle a large number of accounts

API endpoint : *GET baseurl/v1/accounts/account_list*

Sample Request:

Headers

{

Authorization: <OAuth_2>

}

Sample Response:

Status Code: 200 OK

```
{  
  "accounts": [  
    {  
      "id": "9012",  
      "name": "Hansa",  
      "type": "CHECKING",  
      "App": "DNA"  
    },  
  
    {  
      "id": "9013",  
      "name": "Krish",  
      "type": "SAVINGS",  
      "App": "Core"  
    }  
  ]  
}
```

UC_1.1.2_AL - Negative_usecase

Failure Scenario 1.2: Account List Retrieval - Invalid Token - Unauthorized access

API endpoint : *GET baseurl/v1/accounts/account_list*

Given the user provides an invalid or expired token

When the user attempts to access account information

Then the system should reject the request with a 401 Unauthorized status

And the system should provide an error message indicating invalid or expired credentials

Sample Response:

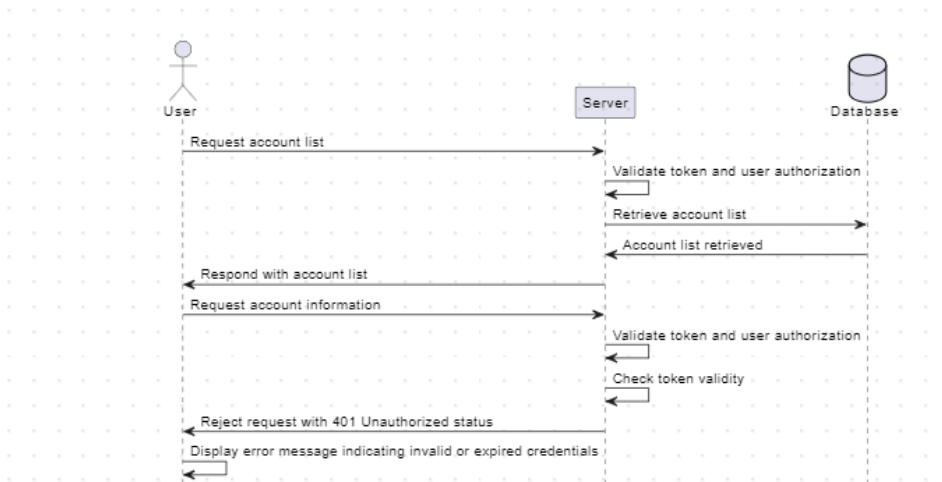
Headers

```
{  
  Authorization: <OAuth_2>  
}
```

Sample Response:

Status: 401 Unauthorized

```
{  
  "error": "Invalid or expired token"  
}
```



UserStory_2 Account Details

As a user, I want to retrieve detailed information about a specific account so that I can view its balance, transactions, and other relevant data.

UC_2.1: Successful Account Details Retrieval

Given the user is authenticated with a valid token

When the user requests details for a specific account, e.g., "0101"

Then the system should validate the token and user authorization

And the system should verify the existence of the requested account

And the system should respond with the details of the specified account

And the response should include account information such as account number, type, balance, and transaction history

API endpoint : ***GET baseurl/v1.2.3/accounts/account_id***

Request

Headers

```
{  
  Authorization: <access_token>  
}  
Header Key - account_id value - 0101
```

Response

Status Code: 200 OK

```
{  
  "id": "1010",  
  "name": "Hansa",  
  "type": "CHECKING",  
  "Transcationtype": "A2A",  
  "balance": 1234.56,  
  "InstID" : "121"  
  "currency": "USD",  
  "iban": "DE89370400440532013000",  
  "transactions": [  
    {  
      "id": "0001",  
      "date": "2023-12-12",  
      "description": "Deposit",  
      "amount": 1000.00,  
      "balance": 2234.56  
    },  
    {  
      "id": "0002",  
      "date": "2023-12-11",  
      "description": "Payment",  
      "amount": -500.00,  
      "balance": 1234.56  
    }  
  ]  
}
```

}

User Story 3: Account Nickname

As a customer, I want to assign a nickname to my account so that I can easily identify it.

UC_3.1 - Successful Nickname Update

Account Nickname - Update account nickname

Given the user is authenticated with a valid token

When the user updates the nickname for a specific account, e.g "10101" to "Emergency Fund"

Then the system should validate the token and user authorization

And the system should verify the existence of the specified account

And the system should update the account nickname

And the system should respond with a success message

And the response should include the updated account details, including the new nickname

API endpoint : *PUT baseurl/v1.2.3/accounts/account_id/nickname*

Sample Request

Header

```
{  
  Authorization: Bearer <Outh_2>  
  Content-Type: application/json  
}
```

Body

```
{  
  "nickname": "Emergency Fund"  
}
```

Sample Response

```
{  
  "id": "1010",  
  "name": "Hansat",
```

```
"type": "CHECKING",  
"balance": 1234.56,  
"currency": "USD",  
"nickname": "Emergency_Fund",  
"iban": "DE89370400440532013000"  
}
```