BLOCKCHAIN BACKEND

REQUIREMENTS AND API CALLS

1 Basics

1.1 Processes

This is a preparatory documentation, its purpose is to lay down the basic concept of API communication. The described technical details - processes, calls, addresses, parameters, etc. - may change during the consultations and implementation, until the parties decide jointly that what is described in the document is considered final.

Basic functions cover:

- ⇒ Authentication
- ⇒ Create account
- ⇒ Top up account
- ⇒ Account debit
- ⇒ Setting the account balance to a given value
- ⇒ Inquiry of account balance

The basic assumption is that the account identifiers are determined by the blockchain API client aggregator system. These are alphanumeric strings with a minimum of 8 and a maximum of 24 characters. In calls placed on endpoints, lowercase letters are always converted to uppercase, and account IDs are stored this way. After the POC, it is strongly recommended to include some kind of verification mechanism that enables the verification of individual account identifiers.

Furthermore, the basic assumption concerning the account balance is that:

- ⇒ The system maintains one general balance for each account.
- ⇒ The balance is validated by the aggregator system, only tracking is done on the blockchain side.
- ⇒ The balance can be negative.
- ⇒ The balance is always displayed with two decimal places, both in the parameters of the calls and in the output of the endpoints.

If the above assumptions are changed, the description of the endpoints will change, for example if it is necessary to handle multiple currencies, then the parameterization and output of most endpoints will also change.

When developing the scope of the PoC project, it is worth considering that the technology offers extra credibility, so it may be worth implementing part of the business logic on the blockchain side.

1.2 Basic settings

All calls must be over HTTPS, content-type for all calls is ‘application/x-www-form-urlencoded’

- ⇒ Test system: https://35.158.186.93:9000
- ⇒ Production system: TBD

1.3 Test credentials

In the test environment, the following credentials can be used for API calls:

- ⇒ login: aggregator

⇒ password: k9tV9MdXYmRp

During access to the test environment, the calling party's IP address is not validated, but the live environment will only be accessible from a pre-defined range.

1.4 RESPONSE STATUSES

The API reports the status of the performed operations using HTTP status codes, these possible values are:

⇒ 200: Success

⇒ 401: Unsuccessful validation of authentication token

⇒ 422: Validation error of incoming request data

⇒ 500: Server side error

1.5 BLOCK EXPLORER

It offers the possibility to browse account and transaction data recorded in the blockchain. The exact functionality will be determined later.

⇒ URL: TBD

2 API Authentication

Pre-authentication is obviously not required. Success of this call is a prerequisite for further requests as the JWT token – returned in the response body of this call - should be sent in the subsequential requests' header in the following format:

```
Authorization: Bearer <JWT_TOKEN>
```

The credentials used in the examples provide access to the test system.

2.1 Request

```
Method: POST
Endpoint: /api/auth
Host: 35.158.186.93:9000
Content-Type: application/x-www-form-urlencoded
```

Sample call

```
curl    -X POST https://35.158.186.93:9000/api/auth           \
        -H "Content-Type: application/x-www-form-urlencoded"  \
        -d "login=aggregator&password=k9tV9MdXYmRp"
```

Request body

With 'application/x-www-form-urlencoded' set as content type, the keys and values are encoded in key-value tuples separated by '&', with a '=' between the key and the value. Non-alphanumeric characters in both keys and values are percent (aka URL) encoded.

Parameter	Format	Max length	Description	Sample
login	string	255	User's registered login name.	aggregator
password	string	255	Password for login in clear text.	k9tV9MdXYmRp

2.2 Responses

HTTP status indicates the outcome of the call. Content type is always "application/json"

2.2.1 Successful authentication: 200

Schema

Parameter	Obligatory	Type	Max length	Description
id	no	string	10	Alphanumeric user identifier.
token	**yes**	**string**	**256**	**JWT token as a string**
expiration	yes	date-time	-	UTC time of expiration in RFC3339 format.
last_login	no	date-time	-	Date and time (UTC) of last login in RFC3339 format. In case of null value or parsing error it is omitted or 0001-01-01T00:00:00Z will be returned.
message	no	string	256	Additional human readable message, if any.

Example response


```
{
  "id": "string",
  "token": "string",
  "expiration": "2020-11-04T11:47:47.149Z",
  "last_login": "2020-11-04T11:47:47.149Z",
  "message": "string"
}
```


2.2.2 Errors: 401, 422, 500

These are http error codes that indicate a possible error:

⇒ 401: Invalid username and/or password supplied

⇒ 422: Validation error of incoming request data

⇒ 500: Server side error

Schema

Parameter	Obligatory	Type	Max length	Description
message	yes	string	256	Human readable message text.
code	no	string	256	Numeric error code for internal use only.

Example response


```
{
  "message": "string",
  "code": "0"
}
```

3 New wallet

Create a new account and set the initial balance. The account number is not generated in blockchain or on the API, only formal verification is done here. If the specified account number has already been created, it returns with a validation error.

Beyond the scope of the POC, the possibility of managing balances in multiple currencies and expanding the stored reference fields should also be considered.

3.1 Request

```
Method: POST
Endpoint: /api/account/init
Host: 35.158.186.93:9000
Content-Type: application/x-www-form-urlencoded
```

Sample call

```
curl    --location --request POST 'https://35.158.186.93:9000/api/account/init'        \
        --header 'Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.Lsdfew'   \
        --header 'Content-Type: application/x-www-form-urlencoded'                      \
        --data-urlencode 'account=8YVPETVKAU84'                                         \
        --data-urlencode 'balance=100.00'                                               \
        --data-urlencode 'reference=ext_ref_data'
```

This call will create account ‘8YVPETVKAU84’ if it doesn't already exist and set its balance to ‘100.00’.

Parameters

Parameter	Format	Max length	Obligatory	Description	Default	Sample
account	string	24	yes	Alphanumeric value, minimum length 8 characters. Lowercase letters are converted to uppercase, and the account number is stored this way.	-	8YVPETVKAU84
balance	float	12	no	Initial balance to be set. Its total length can be 12 characters, in a form that can be converted to float32. Values less than two decimal places are automatically truncated. Thus, the smallest value can be ''-99999999.99' and the largest value can be '999999999.99'.	0.00	100.00
Reference	String	256	no	The external reference used in the aggregator system. Even base64 encoded JSON or XML can be stored.	-	ext_ref_data

3.2 Responses

HTTP status indicates the outcome of the call. Content type is always “application/json”

3.2.1 Success: 200

The account and its initial balance have been recorded.

Schema

Key	Obligatory	Type	Max length	Description
account	yes	string	24	The ID of the created account.
balance	yes	float	12	The set balance.
endpoint	yes	string (enum)	-	Value is always 'init'
txid	yes	string	64	Blockchain transaction identifier.

Example response

```
{
  "account":      "8YVPETVKAU8",
  "balance":      100.00,
  "endpoint":     "init",
  "txid":         "831118d4dd47d5d71e030561001b50490979c9073b9234cd052e4b1b15b5e25a"
}
```

3.2.2 Errors: 401, 422, 500

Possible HTTP error codes:

⇒ 401: Unsuccessful validation of authentication token.

⇒ 422: Validation error of incoming request data. This may indicate a formal validation error (for example, the balance cannot be converted) or the specified account number already exists. The 'message' field of the response contains the description of the cause of the error.

⇒ 500: Server side error.

Schema

Parameter	Obligatory	Type	Max length	Description
message	yes	string	256	Human readable message text.
code	no	string	256	Numeric error code for internal use.

Example response body

```
{
  "message": "string",
  "code": "0"
}
```

4 TOP UP BALANCE

Topping up an existing account.

Beyond the scope of the POC, the possibility of managing balances in multiple currencies and expanding the stored reference fields should also be considered.

4.1 REQUEST

```
Method: POST
Endpoint: /api/account/credit
Host: 35.158.186.93:9000
Content-Type: application/x-www-form-urlencoded
```

SAMPLE CALL

```
curl    --location --request POST 'https://35.158.186.93:9000/api/account/credit'      \
        --header 'Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.Lsdfew'   \
        --header 'Content-Type: application/x-www-form-urlencoded'                      \
        --data-urlencode 'account=8YVPETVKAU84'                                         \
        --data-urlencode 'add=10.00'                                              \
        --data-urlencode 'reference=ext_ref_data'
```

This call adds "10.00" to the balance of the account with ID "8YVPETVKAU84".

PARAMETERS

PARAMETER	FORMAT	MAX LENGTH	OBLIGATORY	DESCRIPTION	DEFAULT	SAMPLE
account	string	24	yes	Alphanumeric value, minimum length 8 characters. Lowercase letters are converted to uppercase, and account number is stored this way.	-	8YVPETVKAU84
add	float	12	yes	Amount to be added to the account balance. Its total length can be 12 characters, in a form that can be converted to float32. Values less than two decimal places are automatically truncated. Negative values produce a validation error. Thus, the smallest value can be '0.00' and the largest value can be '999999999.99'.	-	10.00
reference	String	256	no	The external reference used in the aggregator system. Even base64 encoded JSON or XML can be stored.	-	ext_ref_data

4.2 RESPONSES

HTTP status indicates the outcome of the call. Content type is always “application/json”

4.2.1 SUCCESS: 200

The operation was completed successfully.

Schema

Key	Obligatory	Type	Max length	Description
account	yes	string	24	The ID of the account on which the operation took place.
balance	yes	float	12	New balance. Its total length can be a maximum of 12 characters, in a form that can be converted to float32. The smallest value can be '-99999999.99' and the largest value can be '999999999.99'.
endpoint	yes	string (enum)	-	Value is always 'credit'
txid	yes	string	64	Blockchain transaction identifier.

Example response


```
{
  "account":      "8YVPETVKAU8",
  "balance":      110.00,
  "endpoint":     "credit",
  "txid":         "831118d4dd47d5d71e030561001b50490979c9073b9234cd052e4b1b15b5e25a"
}
```


4.2.2 Errors: 401, 422, 500

Possible HTTP error codes:

⇒ 401: Unsuccessful validation of authentication token.

⇒ 422: Validation error of incoming request data. This may indicate a formal validation error (for example, the amount cannot be converted) or the specified account number does not exist. The 'message' field of the response contains the description of the cause of the error. This error code also occurs if the balance would be too high as a result of the operation.

⇒ 500: Server side error.

Schema

Parameter	Obligatory	Type	Max length	Description
message	yes	string	256	Human readable message text.
code	no	string	256	Numeric error code for internal use.

Example response body


```
{
  "message": "string",
  "code": "0"
}
```

5 Account debit

The account is charged on this endpoint. It should be clarified whether the account balance can drop below 0, and if so, under what conditions. Beyond the scope of the POC, the possibility of managing balances in multiple currencies and expanding the stored reference fields should also be considered.

5.1 Request

```
Method: POST
Endpoint: /api/account/debit
Host: 35.158.186.93:9000
Content-Type: application/x-www-form-urlencoded
```

Sample call

```
curl    --location --request POST 'https://35.158.186.93:9000/api/account/debit'       \
        --header 'Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.Lsdfew'   \
        --header 'Content-Type: application/x-www-form-urlencoded'                      \
        --data-urlencode 'account=8YVPETVKAU84'                                         \
        --data-urlencode 'charge=10.00'                                                 \
        --data-urlencode 'reference=ext_ref_data'
```

This call will debit "10.00" from the balance of the account ID "8YVPETVKAU84".

Parameters

Parameter	Format	Max length	Obligatory	Description	Default	Sample
account	string	24	yes	Alphanumeric value, minimum length 8 characters. Lowercase letters are converted to uppercase, and account number is stored this way.	-	8YVPETVKAU84
charge	float	12	yes	Amount to be charged from the account balance. Its total length can be 12 characters, in a form that can be converted to float32. Values less than two decimal places are automatically truncated. Negative values produce a validation error. Thus, the smallest value can be '0.00' and the largest value can be '999999999.99'.	-	10.00
reference	String	256	no	The external reference used in the aggregator system. Even base64 encoded JSON or XML can be stored.	-	ext_ref_data

5.2 Responses

HTTP status indicates the outcome of the call. Content type is always “application/json”

5.2.1 Success: 200

The operation was completed successfully.

Schema

Key	Obligatory	Type	Max length	Description
account	yes	string	24	The ID of the account on which the operation took place.
balance	yes	float	12	New balance. Its total length can be a maximum of 12 characters, in a form that can be converted to float32. The smallest value can be '-99999999.99' and the largest value can be '999999999.99'.
endpoint	yes	string (enum)	-	Value is always 'debit'
txid	yes	string	64	Blockchain transaction identifier.

Example response


```
{
  "account":      "8YVPETVKAU8",
  "balance":      100.00,
  "endpoint":     "debit",
  "txid":         "831118d4dd47d5d71e030561001b50490979c9073b9234cd052e4b1b15b5e25a"
}
```


5.2.2 Errors: 401, 422, 500

Possible HTTP error codes:

⇒ 401: Unsuccessful validation of authentication token.

⇒ 422: Validation error of incoming request data. This may indicate a formal validation error (for example, the amount cannot be parsed) or the specified account number does not exist. The 'message' field of the response contains the description of the cause of the error. This error code also occurs if the balance would be too low as a result of the operation.

⇒ 500: Server side error.

Schema

Parameter	Obligatory	Type	Max length	Description
message	yes	string	256	Human readable message text.
code	no	string	256	Numeric error code for internal use.

Example response body


```
{
  "message": "string",
  "code": "0"
}
```

6 Account sync

Setting the account balance to a specified value.

Beyond the scope of the POC, the possibility of managing balances in multiple currencies and expanding the stored reference fields should also be considered.

6.1 Request

```
Method: POST
Endpoint: /api/account/sync
Host: 35.158.186.93:9000
Content-Type: application/x-www-form-urlencoded
```

Sample call

```
curl    --location --request POST 'https://35.158.186.93:9000/api/account/sync'        \
        --header 'Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.Lsdfew'   \
        --header 'Content-Type: application/x-www-form-urlencoded'                      \
        --data-urlencode 'account=8YVPETVKAU84'                                         \
        --data-urlencode 'balance=200.00'                                               \
        --data-urlencode 'reference=ext_ref_data'
```

This call will set the balance of the account ID "8YVPETVKAU84" to "200.00".

Parameters

Parameter	Format	Max length	Obligatory	Description	Default	Sample
account	string	24	yes	Alphanumeric value, minimum length 8 characters. Lowercase letters are converted to uppercase, and account number is stored this way.	-	8YVPETVKAU84
balance	float	12	yes	Balance to be set. Its total length can be 12 characters, in a form that can be converted to float32. Values less than two decimal places are automatically truncated. Thus, the smallest value can be ''-99999999.99' and the largest value can be '999999999.99'.	-	200.00
reference	String	256	no	The external reference used in the aggregator system. Even base64 encoded JSON or XML can be stored.	-	ext_ref_data

6.2 Responses

HTTP status indicates the outcome of the call. Content type is always “application/json”

6.2.1 Success: 200

The operation was completed successfully.

Schema

Key	Obligatory	Type	Max length	Description
account	yes	string	24	The ID of the account on which the operation took place.
balance	yes	float	12	New balance. Its total length can be a maximum of 12 characters, in a form that can be converted to float32. The smallest value can be '-99999999.99' and the largest value can be '999999999.99'.
endpoint	yes	string (enum)	-	Value is always 'sync'
txid	yes	string	64	Blockchain transaction identifier.

Example response

```
{
  "account":      "8YVPETVKAU8",
  "balance":      100.00,
  "endpoint":     "sync",
  "txid":         "831118d4dd47d5d71e030561001b50490979c9073b9234cd052e4b1b15b5e25a"
}
```

6.2.2 Errors: 401, 422, 500

Possible HTTP error codes:

⇒ 401: Unsuccessful validation of authentication token.

⇒ 422: Validation error of incoming request data. This may indicate a formal validation error (for example, the amount cannot be parsed) or the specified account number does not exist. The 'message' field of the response contains the description of the cause of the error. This error code also occurs if the balance would be too low or too high as a result of the operation.

⇒ 500: Server side error.

Schema

Parameter	Obligatory	Type	Max length	Description
message	yes	string	256	Human readable message text.
code	no	string	256	Numeric error code for internal use.

Example response body

```
{
  "message": "string",
  "code": "0"
}
```

7 Balance

Query the current balance of the account.

7.1 Request

```
Method: GET
Endpoint: /api/account/balance
Host: 35.158.186.93:9000
Content-Type: application/x-www-form-urlencoded
```

Sample call

```
curl    -X GET                          \
        -H "accept: application/json"   \
        'https://35.158.186.93:9000/api/account/balance?account=8YVPETVKAU84'
```

This call will query the balance of the account ID "8YVPETVKAU84".

Parameters

Parameter	Format	Max length	Obligatory	Description	Default	Sample
account	string	24	yes	Alphanumeric value, minimum length 8 characters. Lowercase letters are converted to uppercase, and account number is stored this way.	-	8YVPETVKAU84

7.2 Responses

HTTP status indicates the outcome of the call. Content type is always “application/json”

7.2.1 Success: 200

The operation was completed successfully.

Schema

Key	Obligatory	Type	Max length	Description
account	yes	string	24	The ID of the account on which the operation took place.
balance	yes	float	12	Current balance. Its total length can be a maximum of 12 characters, in a form that can be converted to float32. The smallest value can be '-99999999.99' and the largest value can be '999999999.99'.
endpoint	yes	string (enum)	-	Value is always ‘balance’
txid	yes	string	64	Blockchain transaction identifier.

Example response

```
{
  "account":      "8YVPETVKAU8",
  "balance":      200.00,
  "endpoint":     "balance",
  "txid":         "831118d4dd47d5d71e030561001b50490979c9073b9234cd052e4b1b15b5e25a"
}
```

7.2.2 Errors: 401, 422, 500

Possible HTTP error codes:

⇒ 401: Unsuccessful validation of authentication token.

⇒ 422: Validation error of incoming request data. This may indicate a formal validation error (for example, too long account number) or the specified account number does not exist. The 'message' field of the response contains the description of the cause of the error.

⇒ 500: Server side error.

Schema

Parameter	Obligatory	Type	Max length	Description
message	yes	string	256	Human readable message text.
code	no	string	256	Numeric error code for internal use.

Example response body

```
{
  "message": "string",
  "code": "0"
}
```