***API description***

General information about bank - GET /api/bank-info

The method returns general information about bank: bank name, branches count, customers count, amounts of funds by currency.

**Authorization**: required.

Response format:

1. If the user is authorized:

{

"name": "Alphabank",

"customers\_count": "1000",

"funds\_amount": {

EUR: "1000",

USD: "2000",

GBP: "5000",

…

}

}

**Status code:** 200

1. If the user is not authorized:

**Status code:** 403

***CLIENTS***

Add a client - POST /api/client

The method adds a new client.

**Authorization**: required.

Request body:

{

"first\_name": “John”,

"last\_name": “Smith”,

"tax\_code": "012345",

"status": 1,

"email": "mail@mail.com",

"phone": "19990006677",

"address": "London, Main str. 56",

"manager\_id": 1

}

Successful response format:

{"result": true}

L

Response format in case of an error or multiple errors: Status code: 200

{

"result": false

“violations”: [

{

“first\_name”: "Name must not be empty"

},

{

"address": "Address must not be empty"

}

}

Response format if the user is not authorized: Status code: 403

Get client by id - GET /api/client/{id}

The method returns information about client.

**Authorization**: required.

Response format:

1. If the user is authorized:

{

"first\_name": “John”,

"last\_name": “Smith”,

"tax\_code": "012345",

"status": 1,

"email": "mail@mail.com",

"phone": "19990006677",

"address": "London, Main str. 56",

“created\_at”: 2011-12-03T10:15:30,

“updated\_at”: 2015-12-03T10:15:30,

"manager\_id": 1

}

**Status code:** 200

1. If the user is not authorized:

**Status code:** 403

Get all clients by status - GET /api/clients/{status}

The method returns clients by status.

**Authorization**: required.

Response format:

If the user is authorized:

{

"clients": [

{

"first\_name": “John”,

"last\_name": “Smith”,

"tax\_code": "012345",

"status": 1,

"email": "mail@mail.com",

"phone": "19990006677",

"address": "London, Main str. 56",

“created\_at”: 2011-12-03T10:15:30,

“updated\_at”: 2015-12-03T10:15:30,

"manager\_id": 1

},

…

]

}

**Status code:** 200

If the user is not authorized:

**Status code:** 403

Get all clients - GET /api/clients/all

The method returns all bank clients.

**Authorization**: required.

Response format:

1. If the user is authorized:

{

"clients": [

{

"first\_name": “John”,

"last\_name": “Smith”,

"tax\_code": "012345",

"status": 1,

"email": "mail@mail.com",

"phone": "19990006677",

"address": "London, Main str. 56",

“created\_at”: 2011-12-03T10:15:30,

“updated\_at”: 2015-12-03T10:15:30,

"manager\_id": 1

},

…

]

}

**Status code:** 200

1. If the user is not authorized:

**Status code:** 403

Change information about client - PUT /api/client/{id}

The method puts the changes into client information.

**Authorization**: required.

Request parameters:

{

"first\_name": “John”,

"last\_name": “Smith”,

"tax\_code": "012345",

"status": 1,

"email": "mail@mail.com",

"phone": "19990006677",

"address": "London, Main str. 56",

}

Successful response format:

{"result": true}

L

Response format in case of an error or multiple errors: Status code: 200

{

"result": false

“violations”: [

{

“first\_name”: "Name must not be empty"

},

{

"address": "Address must not be empty"

}

}

Response format if the user is not authorized: Status code: 403

***ACCOUNTS***

Add an account - POST /api/account

The method adds a new account.

**Authorization**: required.

Request body:

{

"name": “My account”,

"type": “Saving”,

"status": 1

"balance": 100000.50,

"currency\_code": "EUR",

"client\_id": 1,

“created\_at”: 2011-12-03T10:15:30

}

}

Successful response format:

{"result": true}

L

Response format in case of an error or multiple errors: Status code: 200

{

"result": false

“errors”: {

“name”: "Name must not be empty",

"balance": "Balance must not be null",

…

}

}

Response format if the user is not authorized: Status code: 403

Get account by id - GET /api/account/{id}

The method returns information about account.

**Authorization**: required.

Response format:

If the user is authorized:

{

"name": “My account”,

"type": “Saving”,

"status": 1

"balance": 100000.50,

"currency\_code": "EUR"

“created\_at”: 2011-12-03T10:15:30,

“updated\_at”: 2015-12-03T10:15:30,

"client": {

“first\_name”: “John”,

“last\_name”: “Smith”,

“email”: “[mail@mail.com](mailto:mail@mail.com)”,

“phone”: “19990004455”

}

}

**Status code:** 200

If the user is not authorized:

**Status code:** 403

Get all accounts by status - GET /api/accounts/{status}

The method returns accounts by status.

**Authorization**: required.

Response format:

If the user is authorized:

{

"accounts": [

"name": “My account”,

"type": “Saving”,

"status": 1

"balance": 100000.50,

"currency\_code": "EUR"

“created\_at”: 2011-12-03T10:15:30,

“updated\_at”: 2015-12-03T10:15:30,

"client": {

“first\_name”: “John”,

“last\_name”: “Smith”,

“email”: “[mail@mail.com](mailto:mail@mail.com)”,

“phone”: “19990004455”

}

…

]

}

**Status code:** 200

If the user is not authorized:

**Status code:** 403

Get all accounts - GET /api/accounts/all

The method returns all bank accounts.

**Authorization**: required.

Response format:

If the user is authorized:

{

"accounts": [

"name": “My account”,

"type": “Saving”,

"status": 1

"balance": 100000.50,

"currency\_code": "EUR"

“created\_at”: 2011-12-03T10:15:30,

“updated\_at”: 2015-12-03T10:15:30,

"client": {

“first\_name”: “John”,

“last\_name”: “Smith”,

“email”: “[mail@mail.com](mailto:mail@mail.com)”,

“phone”: “19990004455”

}

…

]

}

**Status code:** 200

If the user is not authorized:

**Status code:** 403

Change account - PUT /api/account /{id}

The method put the changes into account.

**Authorization**: required.

Request body:

{

"name": “My account”,

"type": “Saving”,

"status": 1

"balance": 100000.50,

"currency\_code": "EUR"

}

Successful response format:

{"result": true}

L

Response format in case of an error or multiple errors: Status code: 200

{

"result": false

“violations”: [

{

“first\_name”: "Name must not be empty"

},

{

"address": "Address must not be empty"

}

}

Response format if the user is not authorized: Status code: 403

***MANAGERS***

Add a manager - POST /api/manager

The method adds a new manager.

**Authorization**: required.

Request body:

{

"first\_name": “John”,

"last\_name": “Smith”,

"status": 1,

}

Successful response format:

{"result": true}

L

Response format in case of an error or multiple errors: Status code: 200

{

"result": false

“violations”: [

{

“first\_name”: "Name must not be empty"

},

{

"address": "Address must not be empty"

}

}

Response format if the user is not authorized: Status code: 403

Get manager by id - GET /api/ manager/{id}

The method returns information about manager.

**Authorization**: required.

Response format:

If the user is authorized:

{

"first\_name": “John”,

"last\_name": “Smith”,

"status": 1,

“created\_at”: 2011-12-03T10:15:30,

“updated\_at”: 2015-12-03T10:15:30

}

**Status code:** 200

If the user is not authorized:

**Status code:** 403

Get all managers by status - GET /api/ clients/{status}

The method returns managers by status.

**Authorization**: required.

Response format:

If the user is authorized:

{

"managers": [

{

"first\_name": “John”,

"last\_name": “Smith”,

"status": 1,

“created\_at”: 2011-12-03T10:15:30,

“updated\_at”: 2015-12-03T10:15:30

},

…

]

}

**Status code:** 200

If the user is not authorized:

**Status code:** 403

Get all managers - GET /api/managers/all

The method returns all bank managers.

**Authorization**: required.

Response format:

If the user is authorized:

{

"managers": [

{

"first\_name": “John”,

"last\_name": “Smith”,

"status": 1,

“created\_at”: 2011-12-03T10:15:30,

“updated\_at”: 2015-12-03T10:15:30

},

…

]

}

**Status code:** 200

If the user is not authorized:

**Status code:** 403

Change information about manager - PUT /api/manager/{id}

The method put the changes into manager’s information.

**Authorization**: required.

Request parameters:

{

"first\_name": “John”,

"last\_name": “Smith”,

"status": 1,

}

Successful response format:

{"result": true}

L

Response format in case of an error or multiple errors: Status code: 200

{

"result": false

“violations”: [

{

“first\_name”: "Name must not be empty"

},

{

"address": "Address must not be empty"

}

}

Response format if the user is not authorized: Status code: 403

***AGREEMENTS***

Add an agreement - POST /api/agreement

The method adds a new agreement.

**Authorization**: required.

Request body:

{

"account\_id": 20,

"product\_id": 20,

"status": 1,

"sum": 5000.50,

“interest\_rate”: 0.02

}

Successful response format:

{"result": true}

L

Response format in case of an error or multiple errors: Status code: 200

{

"result": false

“violations”: [

{

“first\_name”: "Name must not be empty"

},

{

"address": "Address must not be empty"

}

]

}

Response format if the user is not authorized: Status code: 403

Get agreement by id - GET /api/agreement/{id}

The method returns information about agreement.

**Authorization**: required.

Response format:

If the user is authorized:

{

"account\_id": 20,

"product\_id": 20,

"status": 1,

"sum": 5000.50,

“interest\_rate”: 0.02,

“created\_at”: 2011-12-03T10:15:30,

“updated\_at”: 2015-12-03T10:15:30

}

**Status code:** 200

If the user is not authorized:

**Status code:** 403

Get all agreements by status - GET /api/agreements/{status}

The method returns agreements by status.

**Authorization**: required.

Response format:

If the user is authorized:

{

"agreements": [

{

"account\_id": 20,

"product\_id": 20,

"status": 1,

"sum": 5000.50,

“interest\_rate”: 0.02,

“created\_at”: 2011-12-03T10:15:30,

“updated\_at”: 2015-12-03T10:15:30

},

…

]

}

**Status code:** 200

If the user is not authorized:

**Status code:** 403

Get all agreements - GET /api/agreements/all

The method returns all bank agreements.

**Authorization**: required.

Response format:

If the user is authorized:

{

"agreements": [

{

"account\_id": 20,

"product\_id": 20,

"status": 1,

"sum": 5000.50,

“interest\_rate”: 0.02,

“created\_at”: 2011-12-03T10:15:30,

“updated\_at”: 2015-12-03T10:15:30

},

…

]

}

**Status code:** 200

If the user is not authorized:

**Status code:** 403

Change information about agreement - PUT /api/agreement/{id}

The method put the changes into agreement’s information.

**Authorization**: required.

Request parameters:

{

"account\_id": 20,

"product\_id": 20,

"status": 1,

"sum": 5000.50,

“interest\_rate”: 0.02

}

Successful response format:

{"result": true}

L

Response format in case of an error or multiple errors: Status code: 200

{

"result": false

“violations”: [

{

“sum”: "Sum must be greater then 0"

}

]

}

Response format if the user is not authorized: Status code: 403

***PRODUCTS***

Add a product - POST /api/product

The method adds a new product.

**Authorization**: required.

Request body:

{

"name": “Bank product “,

"limit": 20,

"status": 1,

“currency\_code”: “EUR”,

"product\_limit ": 500050,

“interest\_rate”: 0.02,

“created\_at”: 2011-12-03T10:15:30,

“updated\_at”: 2015-12-03T10:15:30

}

Successful response format:

{"result": true}

L

Response format in case of an error or multiple errors: Status code: 200

{

"result": false

“violations”: [

{

“sum”: "Sum must be greater then 0"

}

]

}

Response format if the user is not authorized: Status code: 403

Get product by id - GET /api/product/{id}

The method returns information about product.

**Authorization**: required.

Response format:

If the user is authorized:

{

"name": “Bank product “,

"status": 1,

“currency\_code”: “EUR”,

"product\_limit": 500050,

“interest\_rate”: 0.02

“created\_at”: 2011-12-03T10:15:30,

“updated\_at”: 2015-12-03T10:15:30

}

**Status code:** 200

If the user is not authorized:

**Status code:** 403

Get all products by status - GET /api/products/{status}

The method returns products by status.

**Authorization**: required.

Response format:

If the user is authorized:

{

"products": [

{

"name": “Bank product “,

"status": 1,

“currency\_code”: “EUR”,

"product\_limit": 500050,

“interest\_rate”: 0.02,

“created\_at”: 2011-12-03T10:15:30,

“updated\_at”: 2015-12-03T10:15:30

},

…

]

}

**Status code:** 200

If the user is not authorized:

**Status code:** 403

Get all products - GET /api/products/all

The method returns all bank products.

**Authorization**: required.

Response format:

If the user is authorized:

{

"products": [

{

"name": “Bank product “,

"status": 1,

“currency\_code”: “EUR”,

"product\_limit": 500050,

“interest\_rate”: 0.02,

“created\_at”: 2011-12-03T10:15:30,

“updated\_at”: 2015-12-03T10:15:30

},

…

]

}

**Status code:** 200

If the user is not authorized:

**Status code:** 403

Change information about product - PUT /api/product/{id}

The method put the changes into product’s information.

**Authorization**: required.

Request parameters:

{

"name": “Bank product “,

"status": 1,

“currency\_code”: “EUR”,

"product\_limit": 500050,

“interest\_rate”: 0.02

}

Successful response format:

{"result": true}

L

Response format in case of an error or multiple errors: Status code: 200

{

"result": false

“violations”: [

{

“sum”: "Sum must be greater then 0"

}

]

}

Response format if the user is not authorized: Status code: 403

***TRANSACTIONS***

Create a new transaction - POST /api/transaction

The method creates a new transaction.

**Authorization**: required.

Request body:

{

"debit\_account\_id": 5,

"credit\_account\_id ": 6,

"type": 1,

“amount”: 1000,

"description": “Payment for services”

}

Successful response format:

{"result": true}

L

Response format in case of an error or multiple errors: Status code: 200

{

"result": false

“violations”: [

{

“sum”: "Sum must be greater then 0"

}

]

}

Response format if the user is not authorized: Status code: 403

Get transaction by id - GET /api/transaction/{id}

The method returns information about transaction.

**Authorization**: required.

Response format:

If the user is authorized:

{

"debit\_account\_id": 5,

"credit\_account\_id ": 6,

"type": 1,

“amount”: 1000,

"description": “Payment for services”

“created\_at”: 2011-12-03T10:15:30

}

**Status code:** 200

If the user is not authorized:

**Status code:** 403

Get all transaction by type - GET /api/transactions/{type}

The method returns products by type.

**Authorization**: required.

Response format:

If the user is authorized:

{

"transactions": [

{

"debit\_account\_id": 5,

"credit\_account\_id ": 6,

"type": 1,

“amount”: 1000,

"description": “Payment for services”

“created\_at”: 2011-12-03T10:15:30

},

…

]

}

**Status code:** 200

If the user is not authorized:

**Status code:** 403

Get all client transactions - GET /api/transactions/{client\_id}

The method returns all client transactions.

**Authorization**: required.

Response format:

If the user is authorized:

{

"transactions": [

{

"debit\_account\_id": 5,

"credit\_account\_id ": 6,

"type": 1,

“amount”: 1000,

"description": “Payment for services”

“created\_at”: 2011-12-03T10:15:30

},

…

]

}

**Status code:** 200

If the user is not authorized:

**Status code:** 403

Get all client transactions for period- GET /api/transactions

The method returns all client transactions for period.

**Authorization**: required.

Parameters:

* Date and time from - 2011-12-03T10:15:30
* Date and time to - 2012-12-03T10:15:30

Response format:

If the user is authorized:

{

"transactions": [

{

"debit\_account\_id": 5,

"credit\_account\_id ": 6,

"type": 1,

“amount”: 1000,

"description": “Payment for services”

“created\_at”: 2011-12-03T10:15:30

},

…

]

}

**Status code:** 200

If the user is not authorized:

**Status code:** 403

Get all transactions - GET /api/transactions/all

The method returns all bank transactions.

**Authorization**: required.

Response format:

If the user is authorized:

{

"transactions": [

{

"debit\_account\_id": 5,

"credit\_account\_id ": 6,

"type": 1,

“amount”: 1000,

"description": “Payment for services”

“created\_at”: 2011-12-03T10:15:30

},

…

]

}

**Status code:** 200

If the user is not authorized:

**Status code:** 403