API Specification

Core API Interface

*<API>*

*Version 1.0a*

*13/05/2021*

**VERSION HISTORY**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Version | Maker | Make date | Reviewer | Review date | Document changes |
| 1.0 | BEN | 13/05/2021 | BEN | 13/05/2021 | Create document |
|  |  |  |  |  |  |

Contents

[I. API LIST 5](#_Toc87257919)

[1. AUTH 5](#_Toc87257920)

[1.1. POST /user/login 5](#_Toc87257921)

[2. API 6](#_Toc87257922)

[2.1. GET /user/userlist 6](#_Toc87257923)

[2.2. GET /user/userpolicy 9](#_Toc87257924)

[2.3. GET /user/userrole 11](#_Toc87257925)

[2.4. POST /user/changepwd 12](#_Toc87257926)

[2.5. POST /user/adduser 13](#_Toc87257927)

[2.6. POST /user/viewuser 15](#_Toc87257928)

[2.7. POST /user/deleteuser 17](#_Toc87257929)

[2.8. POST /user/modifyuser 17](#_Toc87257930)

[2.9. POST /user/addpolicy 19](#_Toc87257931)

[2.10. POST /user/viewpolicy 21](#_Toc87257932)

[2.11. POST /user/deletepolicy 23](#_Toc87257933)

[2.12. POST /user/modifypolicy 23](#_Toc87257934)

[2.13. POST /user/resetpassword 26](#_Toc87257935)

[2.14. POST /user/adduserrole 27](#_Toc87257936)

[2.15. POST /user/viewuserrole 28](#_Toc87257937)

[2.16. POST /user/deleteuserrole 30](#_Toc87257938)

[2.17. POST /user/modifyuserrole 30](#_Toc87257939)

[3. SYSTEM 32](#_Toc87257940)

[3.1. GET /system/getbanklist 32](#_Toc87257941)

[3.2. GET /system/systemcode 34](#_Toc87257942)

[3.3. POST /system/bank/add 35](#_Toc87257943)

[3.4. POST /system/bank/view 37](#_Toc87257944)

[3.5. POST /system/bank/modify 38](#_Toc87257945)

[3.6. POST /system/bank/delete 39](#_Toc87257946)

[3.7. POST /system/lookup 40](#_Toc87257947)

[3.8. POST /system/getinfor 41](#_Toc87257948)

[3.9. POST /system/getmenuinvoke 42](#_Toc87257949)

[3.10. POST /system/updatemenuinvoke 43](#_Toc87257950)

[4. IPC 46](#_Toc87257951)

[4.1. GET /ipc/getlistschedule 46](#_Toc87257952)

[4.2. POST /ipc/addschedule 48](#_Toc87257953)

[4.3. POST /ipc/viewschedule 49](#_Toc87257954)

[4.4. POST /ipc/modifyschedule 51](#_Toc87257955)

[4.5. POST /ipc/deleteschedule 52](#_Toc87257956)

[5. FAST 53](#_Toc87257957)

[5.1. GET /fast/fastbank 53](#_Toc87257958)

[5.2. GET /fast/transaction/outgoing 55](#_Toc87257959)

[5.3. GET /fast/transaction/incoming 57](#_Toc87257960)

[5.4. POST /fast/transaction/view 59](#_Toc87257961)

[5.5. POST /fast/addfastbank 61](#_Toc87257962)

[5.6. POST /fast/viewfastbank 62](#_Toc87257963)

[5.7. POST /fast/modifyfastbank 63](#_Toc87257964)

[5.8. POST /fast/deletefastbank 64](#_Toc87257965)

[5.9. POST /fast/getreceiveracno 65](#_Toc87257966)

[5.10. POST /fast/makefullfundtransfer 66](#_Toc87257967)

[5.11. POST /fast/makehalffundtransfer 66](#_Toc87257968)

[5.12. POST /fast/reversetransaction 67](#_Toc87257969)

[5.13. POST /fast/makeacknowledgement 68](#_Toc87257970)

[5.14. POST /fast/accountinquiry 68](#_Toc87257971)

[5.15. POST /fast/incomingtransaction 69](#_Toc87257972)

[5.16. POST /fast/outgoingtransaction 70](#_Toc87257973)

[5.17. POST /fast/syncoutgoingtran 70](#_Toc87257974)

[5.18. POST /fast/syncincomingtran 71](#_Toc87257975)

[II. Document for developer 73](#_Toc87257976)

[1. Directory Structure Project 73](#_Toc87257977)

[2. Webapi Configuration Details 74](#_Toc87257978)

[3. Automatic Build Instructions 76](#_Toc87257979)

[4. Reusable Functions 77](#_Toc87257980)

[5. Instructions For Creating A Report 78](#_Toc87257981)

[5.1. Step 1: Create new report 78](#_Toc87257982)

[5.2. Step 2: Config in database 83](#_Toc87257983)

[6. Api Programming Tutorial 85](#_Toc87257984)

[6.1. Microsoft .NET Core API Workflow 85](#_Toc87257985)

[6.2. Tutorial For API Not Connecting To O9core 85](#_Toc87257986)

[6.3. Tutorial For API Connecting To O9core 89](#_Toc87257987)

# API LIST

## AUTH

API Authenticate

### POST /user/login

Login to O9Core Banking System

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/user/login |
| Function Description | Login to O9Core Banking System |
| Content-type | application/json |
| Request Body | {  "username": "ac04",  "password": "4QrcOUm6Wau+VuBX8g+IPg=="  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | username | String | Username to login to system | | password | String | Password of user to login | | encrypt | Boolean | Password is encrypting or not. If you send plain password => set this to "false" | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Login successful",  "result": {  "username": "duong",  "firstname": "vu",  "lastname": "duong",  "status": "N",  "expiredate": "12/12/2021",  "branchid": 1,  "gender": 1,  "deptid": 1,  "userlevel": null,  "policyid": null,  "token": "eyJhbGciOiJIUI6IkpXVCJ9.eyJpZCI6ImR1b25nIiwidHiOjE2MjE4NDAxNDZ9.63x5TWLIggvkX99TijsSEqaYQ",  "cdlist": {},  "searchfunc": {},  "form": {},  "menu": []  }  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Login successful | | username | String | Username to login to system | | password | String | Password to login to system | | firstname | String | First of the name of user | | lastname | String | Last of the name of user | | status | String | N: Normal; C: Closed | | expiredate | Date | Date of expire for user login to System | | branchid | Number | Branch ID of the user | | deptid | String | Department of the user | | userlevel | Number | 1: Admin ; 2: IT; 3: Teller | | policyid | Number | 1 | |

## API

### GET /user/userlist

List of user profile in system

REQUEST

|  |  |
| --- | --- |
| Request type | GET |
| URL | http://{{IP\_ADDRESS}}/user/userlist |
| Function Description | Get list of user profile in system |
| Content-type | application/json |
| Request Body | {} |
| Field Description |  |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Get User List successful",  “result”:  {  [  {  "username”: “ac01”,  “firstname”: “Nguyen Van”,  “lastname”: “A”,  “gender”: 1,  “address”: “Phnom Penh”,  “email”: [nguyenvana@gmail.com](mailto:nguyenvana@gmail.com),  “birthday”: “01/01/1990”,  “phone”:”036795965151”,  “status”:” N”,  “usercreated”:” admin”,  “datecreated”: “23/04/2021”,  “usermodified”: “admin”,  “datemodified”: “23/04/2021”,  “islogin”: 1,  “expiretime”: “24/04/2021 08:00:00 AM”,  “branchid”: “9999”  “deptid”: “302”,  “userlevel”: 1,  “coreuserid”: 2290,  “corelgname”:”ac01”,  “productid”: “FAST”,  “isshow”: 1,  “policyid”: 1,  “failnumber”: 2  },  {  "username”: “ac02”,  “firstname”: “Nguyen Thi”,  “lastname”: “B”,  “gender”: 2,  “address”: “Phnom Penh”,  “email”: nguyenthib[@gmail.com](mailto:nguyenvana@gmail.com),  “birthday”: “02/02/1991”,  “phone”:”037795563150”,  “status”:” N”,  “usercreated”:” admin1”,  “datecreated”: “22/04/2021”,  “usermodified”: “admin1”,  “datemodified”: “22/04/2021”,  “islogin”: 0,  “expiretime”: “23/04/2021 08:00:00 AM”,  “branchid”: “0001”  “deptid”: “101”,  “userlevel”: 2,  “coreuserid”: 4521,  “corelgname”:”ac02”,  “productid”: “FAST”,  “isshow”: 0,  “policyid”: 1,  “failnumber”: 4  }  ]  }  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Login successful | | username | String | Username to login to system | | password | String | Password to login to system | | Firstname | String | First of the name of user | | Lastname | String | Last of the name of user | | Gender | Number | 1: Male ; 2: Female; 3: Others | | Address | String | Address of user | | Email | String | Email of user | | Birthday | Date | Date of birth of user | | Phone | String | Phone off user | | Status | String | N: Normal; C: Closed | | Usercreated | String | User created for this user | | Datecreated | Date | Date created for this user | | Usermodifed | String | User modified for this user | | Datemodified | Date | Date modified for this user | | islogin | Number | 1: True ; 2:False | | expiredate | Date | Date of expire for user login to System | | Branchid | Number | Branch ID of the user | | Deptid | String | Department of the user | | Userlevel | Number | 1: Admin ; 2: IT; 3: Teller | | coreuserid | String | User has logon to O9Core | | corelgname | String | User has logon to O9Core | | productid | String | FAST / RIPPLE / MB / IB | | isshow | Number | 1: True; 2:False | | policyid | Number | 1 | | failnumber | Number | Number of login failed | |

### GET /user/userpolicy

List of user policy in system

REQUEST

|  |  |
| --- | --- |
| Request type | GET |
| URL | http://{{IP\_ADDRESS}}/user/userpolicy |
| Function Description | Get list of user policy in system |
| Content-type | application/json |
| Request Body | {} |
| Field Description |  |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Get User Policy successful",  “result”:  {  [  {  " policyid”: 1,  “descr”: “Default policy”,  “isdefault”: 1,  “effrom”: “2020-10-07 11:18:00.000”,  “efto”: “2022-12-31 00:00:00.000”,  “pwdhis”: 3,  “pwdagemax”: 0,  “minpwdlen”: 6,  “pwdcplx”: 0,  “pwdcplxlc”: 0,  “pwdcplxuc”: 0,  “pwdcplxsc”: 0,  “pwccplxsn”: 0,  “datemodified”: “23/04/2021”,  “islogin”: 1,  “timelginrequire”: 0,  “lginfr”: “00:00:00”  “lginto”: “23:59:00”,  “llkoutthrs”: 5,  “resetlkout”: 0,  “usercreate”:” admin”,  “datecreate”: “2021-04-23 12:09:49.987”,  “usermodify”: “admin1”,  “datemodify”: “2021-04-23 12:09:49.987”,  “contractid”: Null,  “isbankedit”:1,  “iscorpedit”:0,  “baseonpolicy”:0,  “pwdft”: “JITS”  }  ]  }  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Login successful | | policyid | String | ID of Policy | | Descr | String | Default policy | | Isdefault | Number | 1: Yes; 2: No | | Effrom | Date | Effect from date | | Efto | Date | Effect to date | | Pwdhis | Number | 1,2,3,4,5 | | Pwdagemax | Number | Password age max : 0 | | Minpwdlen | Number | Min password length: 6 | | Pwdcplx | Number | Password complex: 0 | | Pwdcplxlc | Number | Password complex: 0 | | Pwdcplxuc | Number | Password complex: 0 | | Pwdcplxsc | Number | Password complex: 0 | | Pwccplxsn | Number | Password complex: 0 | | Timelginrequire | Number | Time login required: 0 | | Lginfr | String | Login from: 00:00:00 | | Lginto | String | Login to: 23:59:00 | | Llkoutthrs | String | Login out history: 5 | | Resetlkout | String | Reset login out: 0 | | usercreate | String | User create policy | | datecreate | Date | Date create policy | | usermodify | String | User modify policy | | contractid | String | Contract number | | Isbankedit | Number | 1: True; 2:False | | Iscorpedit | Number | 1: True; 2:False | | Baseonpolicy | Number | 1: True; 2:False | | Pwdft | String | Password default | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | S\_USERPOLICY | |

### GET /user/userrole

List of user role in system

REQUEST

|  |  |
| --- | --- |
| Request type | GET |
| URL | http://{{IP\_ADDRESS}}/user/userrole |
| Function Description | Get list of user role in system |
| Content-type | application/json |
| Request Body | {} |
| Field Description |  |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Get User Role Successful",  “result”:  {  [  {  " roleid”: 1,  “rolename”: “Administrator”,  “roledescription”: “System role”,  “usertype”: “ADMIN”,  “contractno”: NULL,  “usercreated”: “system”,  “datecreated”: “2021-04-23 16:15:39.810”,  “usermodified”: “system”,  “datemodified”: “2021-04-23 16:15:39.810”,  “serviceid”: “FAST”,  “status”: ‘N’,  “isshow”: 1,  } ,  {  " roleid”: 2,  “rolename”: “teller”,  “roledescription”: “A Banker has an authority to do only the first basic and necessary IB services for customers”,  “usertype”: “TELLER”,  “contractno”: NULL,  “usercreated”: “system”,  “datecreated”: “2021-04-23 16:15:39.810”,  “usermodified”: “system”,  “datemodified”: “2021-04-23 16:15:39.810”,  “serviceid”: “FAST”,  “status”: ‘N’,  “isshow”: 1,  }  ]  }  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Login successful | | roleid | Number | Role ID of user | | rolename | String | Name of role | | Roledescription | String | Description about Role | | usertype | String | Type of user role: ADMIN | | contractno | String | Contract number | | usercreated | String | User created Role | | datecreated | Date | Date created role | | usermodified | String | User modified role | | datemodified | Number | Date modified role | | serviceid | Number | Service ID: FAST | | status | Number | Status of Role: N / C | | isshow | Number | Show up information: 1 | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | S\_USERROLE | |

### POST /user/changepwd

Change password of the user

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/user/changepwd |
| Function Description | Change password of the user |
| Content-type | application/json |
| Request Body | {  "username": "duong",  "oldpass": "1234567",  "newpass": "123456",  "encrypt": false  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | username | String | Username to login to system | | oldpass | String | Current password of user to login | | newpass | String | New password of user to login | | encrypt | Boolean | Password is encrypting or not. If you send plain password => set this to "false" | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Change password successful",  "result":  {  “ischanged”: 1  }  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Login successful | | ischanged | Number | 1: True ; 2: False | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | S\_USRAC | |

### POST /user/adduser

Add new the user

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/user/adduser |
| Function Description | Add more than user |
| Content-type | application/json |
| Request Body | {  "username”: “ac01”,  “firstname”: “Nguyen Van”,  “lastname”: “A”,  “gender”: 1,  “address”: “Phnom Penh”,  “email”: [nguyenvana@gmail.com](mailto:nguyenvana@gmail.com),  “birthday”: “01/01/1990”,  “phone”:”036795965151”,  “status”:” N”,  “usercreated”:” admin”,  “datecreated”: “23/04/2021”,  “islogin”: 0,  “expiretime”: “10/05/2022 08:00:00 AM”,  “branchid”: “9999”  “deptid”: “302”,  “userlevel”: 1,  “productid”: “FAST”,  “policyid”: 1,  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | username | String | Username to login to system | | Firstname | String | First of the name of user | | Lastname | String | Last of the name of user | | Gender | Number | 1: Male ; 2: Female; 3: Others | | Address | String | Address of user | | Email | String | Email of user | | Birthday | Date | Date of birth of user | | Phone | String | Phone off user | | Status | String | N: Normal; C: Closed | | Usercreated | String | User created for this user | | Datecreated | Date | Date created for this user | | islogin | Number | 1: True ; 2:False | | expiredate | Date | Date of expire for user login to System | | Branchid | Number | Branch ID of the user | | Deptid | String | Department of the user | | Userlevel | Number | 1: Admin ; 2: IT; 3: Teller | | productid | String | FAST / RIPPLE / MB / IB | | policyid | Number | 1 | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | | Username | String | Username to login to system | |

### POST /user/viewuser

View the user

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/user/viewuser |
| Function Description | View user |
| Content-type | application/json |
| Request Body | {  "username”: “ac01”  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | username | String | Username to login to system | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "View User successful",  "result":  {  "username”: “ac01”,  “firstname”: “Nguyen Van”,  “lastname”: “A”,  “gender”: 1,  “address”: “Phnom Penh”,  “email”: nguyenvana@gmail.com,  “birthday”: “01/01/1990”,  “phone”:”036795965151”,  “status”:” N”,  “usercreated”:” admin”,  “datecreated”: “23/04/2021”,  “usermodified”: “admin”,  “datemodified”: “23/04/2021”,  “islogin”: 1,  “expiretime”: “24/04/2021 08:00:00 AM”,  “branchid”: “9999”  “deptid”: “302”,  “userlevel”: 1,  “coreuserid”: 2290,  “corelgname”:”ac01”,  “productid”: “FAST”,  “isshow”: 1,  “policyid”: 1,  “failnumber”: 2  }  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Create new user successful/  Create new user failure | | username | String | Username to login to system | | Firstname | String | First of the name of user | | Lastname | String | Last of the name of user | | Gender | Number | 1: Male ; 2: Female; 3: Others | | Address | String | Address of user | | Email | String | Email of user | | Birthday | Date | Date of birth of user | | Phone | String | Phone off user | | Status | String | N: Normal; C: Closed | | Usercreated | String | User created for this user | | Datecreated | Date | Date created for this user | | islogin | Number | 1: True ; 2:False | | expiredate | Date | Date of expire for user login to System | | Branchid | Number | Branch ID of the user | | Deptid | String | Department of the user | | Userlevel | Number | 1: Admin ; 2: IT; 3: Teller | | productid | String | FAST / RIPPLE / MB / IB | | policyid | Number | 1 | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | S\_USRAC | | Store procedure | ADM\_USER\_VIEW | |

### POST /user/deleteuser

Delete the user

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/user/deleteuser |
| Function Description | Delete user |
| Content-type | application/json |
| Request Body | {  "username”: “ac01”  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | username | String | Username to login to system | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | S\_USRAC | |

### POST /user/modifyuser

Modify the user

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/user/modifyuser |
| Function Description | Modify user |
| Content-type | application/json |
| Request Body | {  “firstname”: “Nguyen Van”,  “lastname”: “B”,  “gender”: 0,  “address”: “Phnom Penh - Cambodia”,  “birthday”: “01/01/1990”,  “phone”:”036795965151”,  “usercreate”:” admin”,  “datecreated”: “23/05/2021”,  “usermodified”: “admin”,  “datemodified”: “23/05/2021”,  “expiretime”: “24/04/2022 08:00:00 AM”,  “branchid”: “9999”  “deptid”: “302”,  “userlevel”: 1,  “coreuserid”: 2290,  “corelgname”:”ac01”,  “policyid”: 1,  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | firstname | String | First of the name of user | | lastname | String | Last of the name of user | | gender | Number | 1: Male ; 2: Female; 3: Others | | address | String | Address of user | | birthday | Date | Date of birth of user | | phone | String | Phone off user | | usercreated | String | User created for this user | | datecreated | Date | Date created for this user | | expiredate | Date | Date of expire for user login to System | | branchid | Number | Branch ID of the user | | deptid | String | Department of the user | | Userlevel | Number | 1: Admin ; 2: IT; 3: Teller | | policyid | Number | 1 | | coreuserid | String | User has logon to O9Core | | corelgname | String | User has logon to O9Core | | policyid | Number | 1 | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | | Ismodify | Number | 1: True ; 2: False | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | S\_USRAC | | Store procedure | ADM\_USER\_UPDATE | |

### POST /user/addpolicy

Add user policy

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/user/addpolicy |
| Function Description | Add more than policy |
| Content-type | application/json |
| Request Body | {  "policyid": 1,  "descr": "Default policy",  "isdefault": true,  "effrom": "07/10/2020 00:00:00",  "efto": "31/12/2022 00:00:00",  "pwdhis": 3,  "pwdagemax": 0,  "minpwdlen": 6,  "pwdcplx": false,  "pwdcplxlc": false,  "pwdcplxuc": false,  "pwdcplxsc": false,  "pwccplxsn": false,  "timelginrequire": false,  "lginfr": "00:00:00",  "lginto": "23:59:00",  "llkoutthrs": "5",  "resetlkout": "0",  "usercreate": "Ygn1\_manager",  "datecreate": "26/04/2016 12:09:49",  "usermodify": "hoangnv",  "datemodify": "07/10/2020 10:04:48",  "contractid": "",  "isbankedit": true,  "iscorpedit": false,  "baseonpolicy": 0,  "pwdft": "SBILH"  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | policyid | String | ID of Policy | | Descr | String | Default policy | | Isdefault | Number | 1: Yes; 2: No | | Effrom | Date | Effect from date | | Efto | Date | Effect to date | | Pwdhis | Number | 1,2,3,4,5 | | Pwdagemax | Number | Password age max : 0 | | Minpwdlen | Number | Min password length: 6 | | Pwdcplx | Number | Password complex: 0 | | Pwdcplxlc | Number | Password complex: 0 | | Pwdcplxuc | Number | Password complex: 0 | | Pwdcplxsc | Number | Password complex: 0 | | Pwccplxsn | Number | Password complex: 0 | | Timelginrequire | Number | Time login required: 0 | | Lginfr | String | Login from: 00:00:00 | | Lginto | String | Login to: 23:59:00 | | Llkoutthrs | String | Login out history: 5 | | Resetlkout | String | Reset login out: 0 | | usercreate | String | User create policy | | datecreate | Date | Date create policy | | usermodify | String | User modify policy | | contractid | String | Contract number | | Isbankedit | Number | 1: True; 2:False | | Iscorpedit | Number | 1: True; 2:False | | Baseonpolicy | Number | 1: True; 2:False | | Pwdft | String | Password default | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | S\_USERPOLICY | |

### POST /user/viewpolicy

Get policy by Id

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/user/viewpolicy |
| Function Description | Get policy by Id |
| Content-type | application/json |
| Request Body | {  "policyid": 1  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | policyid | String | ID of Policy | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": {  "policyid": 1,  "descr": "Default policy",  "isdefault": true,  "effrom": "07/10/2020 00:00:00",  "efto": "31/12/2022 00:00:00",  "pwdhis": 3,  "pwdagemax": 0,  "minpwdlen": 6,  "pwdcplx": false,  "pwdcplxlc": false,  "pwdcplxuc": false,  "pwdcplxsc": false,  "pwccplxsn": false,  "timelginrequire": false,  "lginfr": "00:00:00",  "lginto": "23:59:00",  "llkoutthrs": "5",  "resetlkout": "0",  "usercreate": "Ygn1\_manager",  "datecreate": "26/04/2016 12:09:49",  "usermodify": "hoangnv",  "datemodify": "07/10/2020 10:04:48",  "contractid": "",  "isbankedit": true,  "iscorpedit": false,  "baseonpolicy": 0,  "pwdft": "SBILH"  }  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Create new user successful/  Create new user failure | | policyid | String | ID of Policy | | Descr | String | Default policy | | Isdefault | Number | 1: Yes; 2: No | | Effrom | Date | Effect from date | | Efto | Date | Effect to date | | Pwdhis | Number | 1,2,3,4,5 | | Pwdagemax | Number | Password age max : 0 | | Minpwdlen | Number | Min password length: 6 | | Pwdcplx | Number | Password complex: 0 | | Pwdcplxlc | Number | Password complex: 0 | | Pwdcplxuc | Number | Password complex: 0 | | Pwdcplxsc | Number | Password complex: 0 | | Pwccplxsn | Number | Password complex: 0 | | Timelginrequire | Number | Time login required: 0 | | Lginfr | String | Login from: 00:00:00 | | Lginto | String | Login to: 23:59:00 | | Llkoutthrs | String | Login out history: 5 | | Resetlkout | String | Reset login out: 0 | | usercreate | String | User create policy | | datecreate | Date | Date create policy | | usermodify | String | User modify policy | | contractid | String | Contract number | | Isbankedit | Number | 1: True; 2:False | | Iscorpedit | Number | 1: True; 2:False | | Baseonpolicy | Number | 1: True; 2:False | | Pwdft | String | Password default | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | S\_USERPOLICY | |

### POST /user/deletepolicy

Delete user policy

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/user/deletepolicy |
| Function Description | Delete user policy |
| Content-type | application/json |
| Request Body | {  "policyid": 1  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | policyid | String | ID of Policy | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | S\_USERPOLICY | |

### 2.12. POST /user/modifypolicy

Modify user policy

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/user/modifypolicy |
| Function Description | Modify user policy |
| Content-type | application/json |
| Request Body | {  "policyid": 1,  "descr": "Default policy",  "isdefault": true,  "effrom": "07/10/2020 00:00:00",  "efto": "31/12/2022 00:00:00",  "pwdhis": 3,  "pwdagemax": 0,  "minpwdlen": 6,  "pwdcplx": false,  "pwdcplxlc": false,  "pwdcplxuc": false,  "pwdcplxsc": false,  "pwccplxsn": false,  "timelginrequire": false,  "lginfr": "00:00:00",  "lginto": "23:59:00",  "llkoutthrs": "5",  "resetlkout": "0",  "usercreate": "Ygn1\_manager",  "datecreate": "26/04/2016 12:09:49",  "usermodify": "hoangnv",  "datemodify": "07/10/2020 10:04:48",  "contractid": "",  "isbankedit": true,  "iscorpedit": false,  "baseonpolicy": 0,  "pwdft": "SBILH"  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Create new user successful/  Create new user failure | | policyid | String | ID of Policy | | Descr | String | Default policy | | Isdefault | Number | 1: Yes; 2: No | | Effrom | Date | Effect from date | | Efto | Date | Effect to date | | Pwdhis | Number | 1,2,3,4,5 | | Pwdagemax | Number | Password age max : 0 | | Minpwdlen | Number | Min password length: 6 | | Pwdcplx | Number | Password complex: 0 | | Pwdcplxlc | Number | Password complex: 0 | | Pwdcplxuc | Number | Password complex: 0 | | Pwdcplxsc | Number | Password complex: 0 | | Pwccplxsn | Number | Password complex: 0 | | Timelginrequire | Number | Time login required: 0 | | Lginfr | String | Login from: 00:00:00 | | Lginto | String | Login to: 23:59:00 | | Llkoutthrs | String | Login out history: 5 | | Resetlkout | String | Reset login out: 0 | | usercreate | String | User create policy | | datecreate | Date | Date create policy | | usermodify | String | User modify policy | | contractid | String | Contract number | | Isbankedit | Number | 1: True; 2:False | | Iscorpedit | Number | 1: True; 2:False | | Baseonpolicy | Number | 1: True; 2:False | | Pwdft | String | Password default | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | | Ismodify | Number | 1: True ; 2: False | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | S\_USERPOLICY | |

### POST /user/resetpassword

Change password of the user

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/user/resetpassword |
| Function Description | Reset password |
| Content-type | application/json |
| Request Body | {  "username": "duong",  "email": "duongvv@jits.com.vn",  "code":"376432"  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | username | String | Username to login to system | | email | String | Email of user | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Create new user successful/  Create new user failure | | Username | String | Username to login to system | | email | String | Email of user | | code | String | Optional field. Set null if first time request, set field with code receive in email | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | S\_USRA | |

### POST /user/adduserrole

Add user role

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/user/adduserrole |
| Function Description | Add more than role |
| Content-type | application/json |
| Request Body | {  "rolename": "Guest",  "roledescription": "System role",  "usertype": "SEMS",  "contractno": null,  "usercreated": "SYSTEM",  "datecreated": "05/01/2010 00:00:00",  "usermodified": "SYSTEM",  "datemodified": "22/04/2010 00:00:00",  "serviceid": "SEMS",  "status": null,  "isshow": "N"  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | roleid | Number | Role ID of user | | rolename | String | Name of role | | Roledescription | String | Description about Role | | usertype | String | Type of user role: ADMIN | | contractno | String | Contract number | | usercreated | String | User created Role | | datecreated | Date | Date created role | | usermodified | String | User modified role | | datemodified | Number | Date modified role | | serviceid | Number | Service ID: FAST | | status | Number | Status of Role: N / C | | isshow | Number | Show up information: 1 | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | S\_USERROLE | |

### POST /user/viewuserrole

View user role by roleid

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/user/viewuserrole |
| Function Description | View user role |
| Content-type | application/json |
| Request Body | {  "roleid": 1  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | roleid | Number | Role ID of user | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": {  "roleid": 1,  "rolename": "Guest",  "roledescription": "System role",  "usertype": "SEMS",  "contractno": null,  "usercreated": "SYSTEM",  "datecreated": "05/01/2010 00:00:00",  "usermodified": "SYSTEM",  "datemodified": "22/04/2010 00:00:00",  "serviceid": "SEMS",  "status": null,  "isshow": "N"  }  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Create new user successful/  Create new user failure | | roleid | Number | Role ID of user | | rolename | String | Name of role | | Roledescription | String | Description about Role | | usertype | String | Type of user role: ADMIN | | contractno | String | Contract number | | usercreated | String | User created Role | | datecreated | Date | Date created role | | usermodified | String | User modified role | | datemodified | Number | Date modified role | | serviceid | Number | Service ID: FAST | | status | Number | Status of Role: N / C | | isshow | Number | Show up information: 1 | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | S\_USERROLE | |

### POST /user/deleteuserrole

Delete user role

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/user/deleteuserrole |
| Function Description | Delete user role |
| Content-type | application/json |
| Request Body | {  "roleid": 6  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | roleid | Number | Role ID of user | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | S\_USERROLE | |

### POST /user/modifyuserrole

Modify user role

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/user/modifyuserrole |
| Function Description | Modify user role |
| Content-type | application/json |
| Request Body | {  "roleid": 6,  "rolename": "Duong",  "roledescription": "System role",  "usertype": "SEMS",  "contractno": "",  "usercreated": "SYSTEM",  "datecreated": "25/05/2021 15:00:28",  "usermodified": "SYSTEM",  "datemodified": "22/04/2010 00:00:00",  "serviceid": "SEMS",  "status": "",  "isshow": "N"  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | roleid | Number | Role ID of user | | rolename | String | Name of role | | Roledescription | String | Description about Role | | usertype | String | Type of user role: ADMIN | | contractno | String | Contract number | | usercreated | String | User created Role | | datecreated | Date | Date created role | | usermodified | String | User modified role | | datemodified | Number | Date modified role | | serviceid | Number | Service ID: FAST | | status | Number | Status of Role: N / C | | isshow | Number | Show up information: 1 | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | S\_USERROLE | |

## SYSTEM

### GET /system/getbanklist

List of bank list information

REQUEST

|  |  |
| --- | --- |
| Request type | GET |
| URL | http://{{IP\_ADDRESS}}/system/getlistbank |
| Function Description | Get list bank information |
| Content-type | application/json |
| Request Body | {} |
| Field Description |  |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Get bank list successful",  "result":  {  [  {  "bankid”: “1”,  “bankname”: “SBI Ly Hour Bank (HQ)”,  “bankcode”: “SBHRKHPP”,  “serviceid”: “FAST”  “uri”: “{ACCTNO}@10.10.100.71”,  “mt103acceptpath”: “”,  “mt103rejectpath”: “”,  “status”: “admin”,  “usercreate”:” admin”  “datecreated”:” 2021-04-26 00:00:00.000”  “usermodifed”: “admin”,  “datemodified”: “2021-04-26 00:00:00.000”,  “issender”: “Y”,  “countrycode”: “KH”,  “autogetquote”: “N”,  “autoacceptquote”: “N”  } ,  {  "bankid”: “2”,  “bankname”: “AUSTRALIA AND NEW ZEALAND BANKING GROUP LIMITED”,  “bankcode”: “ANZBAU2S”,  “serviceid”: “FAST”  “uri”: “”,  “mt103acceptpath”: “”,  “mt103rejectpath”: “”,  “status”: “admin”,  “usercreate”:” admin”  “datecreated”:” 2021-04-26 00:00:00.000”  “usermodifed”: “admin”,  “datemodified”: “2021-04-26 00:00:00.000”,  “issender”: “Y”,  “countrycode”: “KH”,  “autogetquote”: “N”,  “autoacceptquote”: “N”  }  ]  }  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Login successful | | bankid | Number | Bank ID | | bankname | String | Bank name | | Bank code | String | Bank code / SWIFT code | | serviceid | String | Service ID: FAST/RIPPLE | | uri | String | Uri connect to service | | status | String | Status: Active | Stop | Draft | | mt103acceptpath | String | Path to file accept | | mt103rejectpath | String | Path to file reject | | usercreate | String | User Created | | datecreated | Date | Date Created | | usermodifed | String | User modified | | datemodified | Date | Date modified | | issender | String | Is sender? | | countrycode | String | Country code | | autogetquote | String | Y: Yes; N: No | | autoacceptquote |  | Y: Yes; N: No | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | S\_BANK | |

### GET /system/systemcode

Get list system code information

REQUEST

|  |  |
| --- | --- |
| Request type | GET |
| URL | http://{{IP\_ADDRESS}}/system/systemcode |
| Function Description | Get list system code information |
| Content-type | application/json |
| Request Body | {} |
| Field Description |  |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Get system code successful",  "result":  {  “BANKSTS”:  {  [  {  “key”: “A”,  “value”: “Active”  },  {  “key”: “I”,  “value”: “InActive”  }  ]  },  “FASTSTS”:  {  [  {  “key”: “ANEW”,  “value”: “New”  },  {  “key”: “RCVD”,  “value”: “Received”  }  ]  }    }  } |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | S\_SYSTEMCODE | | Store procedure | SYS\_GET\_SYSTEMCODE | |

### POST /system/bank/add

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/system/bank/add |
| Function Description | Add more than user |
| Content-type | application/json |
| Request Body | {  "bankname": "duong",  "bankcode": "duonggg",  "serviceid": "BLS",  "uri": "{ACCTNO}@10.10.100.71",  "mt103acceptpath": "D:\\1.ENVIRONMENT\\2.SBILH\\RIPPLE\\LRC\\CAM\\ACCEPT",  "mt103rejectpath": "D:\\1.ENVIRONMENT\\2.SBILH\\RIPPLE\\LRC\\CAM\\REJECT",  "banksts": "A",  "usercreated": "dungvt",  "datecreated": "16/07/2020 00:00:00",  "usermodified": "hoangnv",  "datemodified": "08/01/2021 00:00:00",  "issender": "Y",  "countrycode": "KH",  "autogetquote": "N",  "autoacceptquote": "N"  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | bankid | Number | Bank ID | | bankname | String | Bank name | | Bank code | String | Bank code / SWIFT code | | serviceid | String | Service ID: FAST/RIPPLE | | uri | String | Uri connect to service | | status | String | Status: Active | Stop | Draft | | mt103acceptpath | String | Path to file accept | | mt103rejectpath | String | Path to file reject | | usercreate | String | User Created | | datecreated | Date | Date Created | | usermodifed | String | User modified | | datemodified | Date | Date modified | | issender | String | Is sender? | | countrycode | String | Country code | | autogetquote | String | Y: Yes; N: No | | autoacceptquote |  | Y: Yes; N: No | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | S\_BANK | |

### POST /system/bank/view

View system bank

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/system/bank/view |
| Function Description | View system bank |
| Content-type | application/json |
| Request Body | {  "bankid": 6  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | username | String | Username to login to system | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": {  "bankid": 6,  "bankname": "COMMONWEALTH BANK OF AUSTRALIA SYDNEY (FOR HEAD OFFICE AND ALL AUSTRALIAN BRANCHES)",  "bankcode": "CTBAAU2S",  "serviceid": "BLS",  "uri": "trans\_usd\_south@test.jpn.m003",  "mt103acceptpath": "D:\\RIPPLE\\LRC\\CAM\\ACCEPT",  "mt103rejectpath": "D:\\RIPPLE\\LRC\\CAM\\REJECT",  "banksts": "A",  "usercreated": "hoangnv",  "datecreated": "21/07/2020 16:40:52",  "usermodified": null,  "datemodified": null,  "issender": "N",  "countrycode": "AU",  "autogetquote": "N",  "autoacceptquote": "N"  }  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | | bankid | Number | Bank ID | | bankname | String | Bank name | | Bank code | String | Bank code / SWIFT code | | serviceid | String | Service ID: FAST/RIPPLE | | uri | String | Uri connect to service | | status | String | Status: Active | Stop | Draft | | mt103acceptpath | String | Path to file accept | | mt103rejectpath | String | Path to file reject | | usercreate | String | User Created | | datecreated | Date | Date Created | | usermodifed | String | User modified | | datemodified | Date | Date modified | | issender | String | Is sender? | | countrycode | String | Country code | | autogetquote | String | Y: Yes; N: No | | autoacceptquote |  | Y: Yes; N: No | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | S\_BANK | |

### POST /system/bank/modify

Modify system bank

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/system/bank/modify |
| Function Description | Modify system bank |
| Content-type | application/json |
| Request Body | {  "bankid": 110,  "bankname": "duong1111",  "bankcode": "sualan1",  "serviceid": "BLS",  "uri": "{ACCTNO}@10.10.100.71",  "mt103acceptpath": "D:\\1.ENVIRONMENT\\2.SBILH\\RIPPLE\\LRC\\CAM\\ACCEPT",  "mt103rejectpath": "D:\\1.ENVIRONMENT\\2.SBILH\\RIPPLE\\LRC\\CAM\\REJECT",  "banksts": "A",  "usercreated": "dungvt",  "datecreated": "16/07/2020 00:00:00",  "usermodified": "hoangnv",  "datemodified": "08/01/2021 00:00:00",  "issender": "Y",  "countrycode": "KH",  "autogetquote": "N",  "autoacceptquote": "N"  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | bankid | Number | Bank ID | | bankname | String | Bank name | | Bank code | String | Bank code / SWIFT code | | serviceid | String | Service ID: FAST/RIPPLE | | uri | String | Uri connect to service | | status | String | Status: Active | Stop | Draft | | mt103acceptpath | String | Path to file accept | | mt103rejectpath | String | Path to file reject | | usercreate | String | User Created | | datecreated | Date | Date Created | | usermodifed | String | User modified | | datemodified | Date | Date modified | | issender | String | Is sender? | | countrycode | String | Country code | | autogetquote | String | Y: Yes; N: No | | autoacceptquote |  | Y: Yes; N: No | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | S\_BANK | |

### POST /system/bank/delete

Delete system bank

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/system/bank/delete |
| Function Description | Delete system bank |
| Content-type | application/json |
| Request Body | {  "bankid": 6  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | bankid | Number | Bank ID | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | S\_BANK | |

### POST /system/lookup

Lookup data

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/system/lookup |
| Function Description | Lookup data |
| Content-type | application/json |
| Request Body | {  "lookupid":"BRANCH\_LKP",  "param":{  "branchcd":9999  }  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | lookupid | String | Lookup ID | | param | List | List param to filter data | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": {  "data": [  {  "code": 9999,  "name": "Branch in Ho Chi Minh City vietttttttttttttttttttttnam"  }  ]  }  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | S\_LOOPKUP | |

### POST /system/getinfor

Getinfor data

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/system/getinfor |
| Function Description | Getinfor data |
| Content-type | application/json |
| Request Body | {  "inforid":"user\_branch|user\_policy|user\_user",  "param":{  "branchid":9999,  "brname":"Branch",  "policyid":1  }  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | inforid | String | List inforid ID | | param | List | List param to filter data | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": {  "brname": "Branch in Ho Chi Minh City vietttttttttttttttttttttnam",  "address": "1",  "descr": "Default policy"  }  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | S\_GETINFOR | |

### POST /system/getmenuinvoke

Get all menu role for invoke

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/system/getmenuinvoke |
| Function Description | Get all menu role for invoke |
| Content-type | application/json |
| Request Body | {  "roleid": 4  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | roleid | Number | Role ID of user | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": {  "data": [  {  "menuid": "001",  "menuparent": "0",  "cmdtype": "M",  "type": "link",  "caption": "{\"en\" : \"Home\", \"vi\" : \"Trang chủ\", \"kh\" : \"ទាំងគ្រប់គ្រង\"}",  "menuorder": 1,  "pageid": null,  "actionid": null,  "roleid": 4,  "invoke": true  },  {  "menuid": "002",  "menuparent": "0",  "cmdtype": "M",  "type": "dropDown",  "caption": "{\"en\" : \"BO Transaction\", \"vi\" : \"Giao dịch BO\", \"kh\" : \"BO ប្រតិបត្តិការ\"}",  "menuorder": 2,  "pageid": null,  "actionid": "Search",  "roleid": 4,  "invoke": true  }  ]  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | | menuid | String | Menu ID | | menuparent | String | ID of parent | | cmdtype | String | Command type | | type | String | Type of component | | menuorder | String | Order for display | | pageid | String | ID of page | | actionid | String | Action of component | | roleid | String | Role ID | | invoke | Boolean | Invoke | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | UI\_MENU | | Table | UI\_MENUROLES | | Table | S\_USERROLEDETAIL | |

### POST /system/updatemenuinvoke

Update menu role for invoke

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/system/updatemenuinvoke |
| Function Description | Update menu role for invoke |
| Content-type | application/json |
| Request Body | {  "data": [  {  "roleid": 4,  "menuid": "001",  "invoke": true  },  {  "roleid": 4,  "menuid": "002",  "invoke": true  },  {  "roleid": 4,  "menuid": "002001",  "invoke": false  }  ]  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | | menuid | String | Menu ID | | menuparent | String | ID of parent | | cmdtype | String | Command type | | type | String | Type of component | | menuorder | String | Order for display | | pageid | String | ID of page | | actionid | String | Action of component | | roleid | String | Role ID | | invoke | Boolean | Invoke | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | UI\_MENU | | Table | UI\_MENUROLES | | Table | S\_USERROLEDETAIL | |

## IPC

### GET /ipc/getlistschedule

List of schedule information

REQUEST

|  |  |
| --- | --- |
| Request type | GET |
| URL | http://{{IP\_ADDRESS}}/ipc/getlistschedule |
| Function Description | Get list schedule information |
| Content-type | application/json |
| Request Body | {} |
| Field Description |  |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Get schedule successful",  "result":  {  [  {  " scheduleid”: “SCH00000000000000001”,  “scheduletype”: “REPEATEDLY”,  “scheduletime”: “2021-04-26 07:00:00.000”,  “schedulename”: “Get o9 session”,  “description”: “Get o9 session”,  “status”: “A”,  “usercreate”: “admin”,  “userapproved”: “admin”,  “isapproved”: “Y”,  “serviceid”: “BLS”,  “actiontype”: “”,  “actionid”: “A101”,  “nextexecute”: “2021-04-26 14:08:02.570”,  “createdate”: “2021-04-26 14:08:02.570”,  “enddate”: “2099-07-07 00:00:00.000”  } ,  {  " scheduleid”: “SCH00000000000000015”,  “scheduletype”: “REPEATEDLY”,  “scheduletime”: “2021-04-26 07:00:00.000”,  “schedulename”: “FAST Get Account Inquiry”,  “description”: “Get Account Inquiry”,  “status”: “A”,  “usercreate”: “admin”,  “userapproved”: “admin”,  “isapproved”: “Y”,  “serviceid”: “BLS”,  “actiontype”: “”,  “actionid”: “A114”,  “nextexecute”: “2021-04-26 14:08:02.570”,  “createdate”: “2021-04-26 14:08:02.570”,  “enddate”: “2099-07-07 00:00:00.000”  }  ]  }  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Login successful | | scheduleid | String | Schedule ID | | scheduletype | String | Schedule type: REPEATEDLY | | scheduletime | Date | Schedule time | | schedulename | String | Schedule name | | description | String | Description | | status | String | Status: Active | Stop | Draft | | usercreate | String | User created | | userapproved | String | User approved | | isapproved | String | Y: Yes; N:No | | serviceid | String | Service ID: BLS | | actiontype | String | Action type | | actionid | String | Action ID: Axxx | | nextexecute | Date | Next execute time | | createdate | Date | Create date | | enddate | Date | End date | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | IPC\_SCHEDULES | |

### POST /ipc/addschedule

Add new schedule

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/ipc/addschedule |
| Function Description | Add new schedule |
| Content-type | application/json |
| Request Body | {  "scheduleid": "duong1",  "scheduletype": "REPEATEDLY",  "scheduletime": "07/07/2017 08:30:00",  "schedulename": "test thoi1",  "description": "test thoi1",  "status": "S",  "usercreate": "BEN",  "userapproved": "BEN",  "isapproved": "Y",  "serviceid": "KRC",  "actiontype": "",  "actionid": "AGETTOKEN",  "nextexecute": "27/04/2021 13:50:00",  "enddate": "07/07/2099 00:00:00",  "ord": 20  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | scheduleid | String | Schedule ID | | scheduletype | String | Schedule type: REPEATEDLY | | scheduletime | Date | Schedule time | | schedulename | String | Schedule name | | description | String | Description | | status | String | Status: Active | Stop | Draft | | usercreate | String | User created | | userapproved | String | User approved | | isapproved | String | Y: Yes; N:No | | serviceid | String | Service ID: BLS | | actiontype | String | Action type | | actionid | String | Action ID: Axxx | | nextexecute | Date | Next execute time | | createdate | Date | Create date | | enddate | Date | End date | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | | scheduleid | String | Schedule ID | | scheduletype | String | Schedule type: REPEATEDLY | | scheduletime | Date | Schedule time | | schedulename | String | Schedule name | | description | String | Description | | status | String | Status: Active | Stop | Draft | | usercreate | String | User created | | userapproved | String | User approved | | isapproved | String | Y: Yes; N:No | | serviceid | String | Service ID: BLS | | actiontype | String | Action type | | actionid | String | Action ID: Axxx | | nextexecute | Date | Next execute time | | createdate | Date | Create date | | enddate | Date | End date | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | IPC\_SCHEDULES | |

### POST /ipc/viewschedule

View schedule

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/ipc/viewschedule |
| Function Description | View schedule |
| Content-type | application/json |
| Request Body | {  "scheduleid": "duong1"  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | scheduleid | String | Schedule ID | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": {  "scheduleid": "duong1",  "scheduletype": "REPEATEDLY",  "scheduletime": "07/07/2017 08:30:00",  "schedulename": "test thoi1",  "description": "test thoi1",  "status": "S",  "usercreate": "BEN",  "userapproved": "BEN",  "isapproved": "Y",  "serviceid": "KRC",  "actiontype": "",  "actionid": "AGETTOKEN",  "nextexecute": "27/04/2021 13:50:00",  "createdate": "14/06/2021 21:47:08",  "enddate": "07/07/2099 00:00:00",  "ord": 20  }  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | | scheduleid | String | Schedule ID | | scheduletype | String | Schedule type: REPEATEDLY | | scheduletime | Date | Schedule time | | schedulename | String | Schedule name | | description | String | Description | | status | String | Status: Active | Stop | Draft | | usercreate | String | User created | | userapproved | String | User approved | | isapproved | String | Y: Yes; N:No | | serviceid | String | Service ID: BLS | | actiontype | String | Action type | | actionid | String | Action ID: Axxx | | nextexecute | Date | Next execute time | | createdate | Date | Create date | | enddate | Date | End date | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | IPC\_SCHEDULES | |

### POST /ipc/modifyschedule

Modify schedule

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/ipc/modifyschedule |
| Function Description | Modify schedule |
| Content-type | application/json |
| Request Body | {  "scheduleid": "duong",  "scheduletype": "REPEATEDLY",  "scheduletime": "07/07/2017 08:30:00",  "schedulename": "test thoi1",  "description": "test thoi1",  "status": "S",  "usercreate": "BEN",  "userapproved": "BEN",  "isapproved": "Y",  "serviceid": "KRC",  "actiontype": "",  "actionid": "AGETTOKEN",  "nextexecute": "27/04/2021 13:50:00",  "createdate": "07/07/2017 00:00:00",  "enddate": "07/07/2099 00:00:00",  "ord": 20  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | scheduleid | String | Schedule ID | | scheduletype | String | Schedule type: REPEATEDLY | | scheduletime | Date | Schedule time | | schedulename | String | Schedule name | | description | String | Description | | status | String | Status: Active | Stop | Draft | | usercreate | String | User created | | userapproved | String | User approved | | isapproved | String | Y: Yes; N:No | | serviceid | String | Service ID: BLS | | actiontype | String | Action type | | actionid | String | Action ID: Axxx | | nextexecute | Date | Next execute time | | createdate | Date | Create date | | enddate | Date | End date | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | | scheduleid | String | Schedule ID | | scheduletype | String | Schedule type: REPEATEDLY | | scheduletime | Date | Schedule time | | schedulename | String | Schedule name | | description | String | Description | | status | String | Status: Active | Stop | Draft | | usercreate | String | User created | | userapproved | String | User approved | | isapproved | String | Y: Yes; N:No | | serviceid | String | Service ID: BLS | | actiontype | String | Action type | | actionid | String | Action ID: Axxx | | nextexecute | Date | Next execute time | | createdate | Date | Create date | | enddate | Date | End date | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | IPC\_SCHEDULES | |

### POST /ipc/deleteschedule

Delete schedule

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/ipc/deleteschedule |
| Function Description | Delete schedule |
| Content-type | application/json |
| Request Body | {  "scheduleid": "duong1"  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | scheduleid | String | Schedule ID | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | IPC\_SCHEDULES | |

## FAST

### GET /fast/fastbank

List of bank join into FAST NBC

REQUEST

|  |  |
| --- | --- |
| Request type | GET |
| URL | http://{{IP\_ADDRESS}}/fast/fastbank |
| Function Description | Get list of bank join into FAST NBC |
| Content-type | application/json |
| Request Body | {} |
| Field Description |  |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Get FAST NBC successful",  "result":  {  [  {  " bankid”: SBHRKHPP,  “bankname”: “SBI LYHOUR BANK PLC”,  “bankstreet”: “#85-88, Street 289, Phum 14, Sangkat Boeng Kak Pir, Khan Tuol Kouk, Phnom Penh”,  “bankbuilding”: “36”,  “bankpostalcode”: “12150”,  “banktown”: “SBILH”,  “bankcountry”: “CAMBODIA”,  “username”: “admin”,  “password”: “123456”,  “participatecode”: “SBILH001”  } ,  {  " bankid”: KRTHKHPPXXX,  “bankname”: “KRUNG THAI BANK PUBLIC COMPANY LIMITED (PP)”,  “bankstreet”: “No. 149, Jawaharlal Nehru (St. 215), 12153, Phnom Penh, Cambodia”,  “bankbuilding”: “12”,  “bankpostalcode”: “12153”,  “banktown”: “KTB”,  “bankcountry”: “CAMBODIA”,  “username”: “admin”,  “password”: “123456”,  “participatecode”: “KTBPP01”  }  ]  }  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Login successful | | bankid | Number | Bank ID | | bankname | String | Name of the bank | | bankstreet | String | Bank Street | | bankbuilding | String | Bank Building | | bankpostalcode | String | Bank postal code | | banktown | String | Bank town | | bankcountry | Date | Bank country | | username | String | User name | | password | Number | Password | | participatecode | Number | Participate code | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | FAST\_BANK | | Store procedure | FAST\_BANKLIST | |

### GET /fast/transaction/outgoing

Get outgoing transaction list

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/fast/transaction/outgoing |
| Function Description | Get outgoing transaction list |
| Content-type | application/json |
| Request Body | {  "page":1,  "numberperpage": 15,  "searchfunc":"FAST\_OUTGOING",  "search":  {  "ipctransid": 1200  }  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | page | Number | Number page | | numberperpage | Number | Number row per page | | searchfunc | String | Search function for search | | ipctransid | Number | Transaction ID for search | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Get Transaction Successful",  "result":  {  [  {  "ipctransid”: 5589,  “transtype”: “FAST\_OUTGOING”,  “transdate”: “2021-04-28 19:40:51.833”  “senderacno”: “999901117”,  “sendername”: “BEN”,  “senderbank”: “SBILH BANK PLC”,  “senderamount”: 10000,  “senderccrcd”: “KHR”,  “receiveracno”: “999901112”,  “receivername”: “KABI”,  “receiverbank”: “ACLEDA Bank Plc (ACLEDA)”,  “receiveramount”: 10000,  “receiverccrcd”: “KHR”,  “txrefid”: “210201300000062998”,  “paymentid”:” SBHRKHPP/ACLBKHPP/999900007”,  “status”: “Pending”  “description”: “SBILH send payment to ACLEDA”  }  ]  }  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Get Transaction successful | | ipctransid | Number | Transaction ID | | Instructionid | String | Instruction ID | | transtype | String | Transaction type | | transdate | Date | Transaction date time | | senderacno | String | Sender account | | sendername | String | Sender name | | senderbank | String | Sender bank | | senderamount | Number | Sender amount | | senderccrcd | String | Sender currency | | senderfee | Number | Sender fee | | receiveracno | String | Receiver account | | receivername | String | Receiver name | | receiverbank | String | Receiver bank | | receiveramount | Number | Receiver amount | | receiverccrcd | String | Receiver currency | | receiverfee | Number | Receiver fee | | txrefid | String | Transaction references | | paymentid | String | Payment ID | | pmtref | String | Payment Internal References | | description | String | Description | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | IPCLOGTRANS | | Store procedure | FAST\_GETOUTGOING | | View | V\_FAST\_OUTGOING | |

### GET /fast/transaction/incoming

Get incoming transaction list

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/fast/transaction/incoming |
| Function Description | Get incoming transaction list |
| Content-type | application/json |
| Request Body | {  "page":1,  "numberperpage": 15,  "searchfunc":"FAST\_OUTGOING",  "search":  {  "ipctransid": 1200  }  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | page | Number | Number page | | numberperpage | Number | Number row per page | | searchfunc | String | Search function for search | | ipctransid | Number | Transaction ID for search | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Get Transaction Successful",  "result":  {  [  {  "ipctransid”: 5589,  “transtype”: “FAST\_INCOMING”,  “transdate”: “2021-04-28 19:40:51.833”  “senderacno”: “999901117”,  “sendername”: “BEN”,  “senderbank”: “ACLEDA Bank Plc (ACLEDA)”,  “senderamount”: 10000,  “senderccrcd”: “KHR”,  “receiveracno”: “999901112”,  “receivername”: “KABI”,  “receiverbank”: “SBILH BANK Plc”,  “receiveramount”: 10000,  “receiverccrcd”: “KHR”,  “txrefid”: “210201300000062998”,  “paymentid”:” SBHRKHPP/ACLBKHPP/999900007”,  “status”: “Pending”  “description”: “SBILH send payment to ACLEDA”  }  ]  }  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Get Transaction successful | | ipctransid | Number | Transaction ID | | Instructionid | String | Instruction ID | | transtype | String | Transaction type | | transdate | Date | Transaction date time | | senderacno | String | Sender account | | sendername | String | Sender name | | senderbank | String | Sender bank | | senderamount | Number | Sender amount | | senderccrcd | String | Sender currency | | senderfee | Number | Sender fee | | receiveracno | String | Receiver account | | receivername | String | Receiver name | | receiverbank | String | Receiver bank | | receiveramount | Number | Receiver amount | | receiverccrcd | String | Receiver currency | | receiverfee | Number | Receiver fee | | txrefid | String | Transaction references | | paymentid | String | Payment ID | | description | String | Description | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | IPCLOGTRANS | | Store procedure | FAST\_INCOMING | | View | V\_FAST\_INCOMING | |

### POST /fast/transaction/view

Get transaction detail by transaction references (TRANSID)

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/fast/addfastbank |
| Function Description | Get transaction detail by transaction references (TRANSID) |
| Content-type | application/json |
| Request Body | {  "ipctransid": 12353  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | ipctransid | Number | Transaction ID | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Get Transaction Successful",  "result":  {  "ipctransid”: 5589,  “instructionid”: “SBHRKHPP/999900007”,  “transtype”: “FAST\_OUTGOING”,  “transdate”: “2021-04-28 19:40:51.833”  “senderacno”: “999901117”,  “sendername”: “BEN”,  “senderbank”: “SBILH BANK PLC”,  “senderamount”: 10000,  “senderccrcd”: “KHR”,  “senderfee”: 10,  “receiveracno”: “999901112”,  “receivername”: “KABI”,  “receiverbank”: “ACLEDA Bank Plc (ACLEDA)”,  “receiveramount”: 10000,  “receiverccrcd”: “KHR”,  “receiverfee”: 10,  “txrefid”: “210201300000062998”,  “status”: “Pending”,  “paymentid”:” SBHRKHPP/ACLBKHPP/999900007”,  “description”: “SBILH send payment to ACLEDA”  }  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Get Transaction successful | | ipctransid | Number | Transaction ID | | Instructionid | String | Instruction ID | | transtype | String | Transaction type | | transdate | Date | Transaction date time | | senderacno | String | Sender account | | sendername | String | Sender name | | senderbank | String | Sender bank | | senderamount | Number | Sender amount | | senderccrcd | String | Sender currency | | senderfee | Number | Sender fee | | receiveracno | String | Receiver account | | receivername | String | Receiver name | | receiverbank | String | Receiver bank | | receiveramount | Number | Receiver amount | | receiverccrcd | String | Receiver currency | | receiverfee | Number | Receiver fee | | txrefid | String | Transaction references | | Status | String | Status of transaction | | paymentid | String | Payment ID | | description | String | Description | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | IPCLOGTRANS | | Store procedure | FAST\_GETTRANSDETAIL | |

### POST /fast/addfastbank

Add new fast bank

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/fast/addfastbank |
| Function Description | Add new fast bank |
| Content-type | application/json |
| Request Body | {  "bankid": "duong",  "bankname": "KRUNG THAI BANK PUBLIC COMPANY LIMITED (PP)",  "bankstreet": "No. 149, Jawaharlal Nehru (St. 215), 12153, Phnom Penh, Cambodia\r\n",  "bankbuilding": "12",  "bankpostalcode": "12153",  "banktown": "KTB",  "bankcountry": "CAMBODIA",  "username": "admin",  "password": "123456",  "participatecode": "KTBPP01"  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | bankid | Number | Bank ID | | bankname | String | Name of the bank | | bankstreet | String | Bank Street | | bankbuilding | String | Bank Building | | bankpostalcode | String | Bank postal code | | banktown | String | Bank town | | bankcountry | Date | Bank country | | username | String | User name | | password | Number | Password | | participatecode | Number | Participate code | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | FAST\_BANK | |

### POST /fast/viewfastbank

View fast bank

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/fast/viewfastbank |
| Function Description | View fast bank |
| Content-type | application/json |
| Request Body | {  "bankid": "KRTHKHPPXXX"  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | bankid | String | Id of fast bank | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": {  "bankid": "KRTHKHPPXXX",  "bankname": "KRUNG THAI BANK PUBLIC COMPANY LIMITED (PP)",  "bankstreet": "No. 149, Jawaharlal Nehru (St. 215), 12153, Phnom Penh, Cambodia\r\n",  "bankbuilding": "12",  "bankpostalcode": "12153",  "banktown": "KTB",  "bankcountry": "CAMBODIA",  "username": "admin",  "password": "123456",  "participatecode": "KTBPP01"  }  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Create new user successful/  Create new user failure | | bankid | Number | Bank ID | | bankname | String | Name of the bank | | bankstreet | String | Bank Street | | bankbuilding | String | Bank Building | | bankpostalcode | String | Bank postal code | | banktown | String | Bank town | | bankcountry | Date | Bank country | | username | String | User name | | password | Number | Password | | participatecode | Number | Participate code | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | FAST\_BANK | |

### POST /fast/modifyfastbank

Modify fast bank

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/fast/modifyfastbank |
| Function Description | Modify fast bank |
| Content-type | application/json |
| Request Body | {  "bankid": "duong",  "bankname": "KRUNG THAI BANK PUBLIC COMPANY LIMITED (PP)",  "bankstreet": "No. 149, Jawaharlal Nehru (St. 215), 12153, Phnom Penh, Cambodia\r\n",  "bankbuilding": "12",  "bankpostalcode": "12153",  "banktown": "KTB",  "bankcountry": "CAMBODIA",  "username": "duong",  "password": "123456",  "participatecode": "KTBPP01"  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | bankid | Number | Bank ID | | bankname | String | Name of the bank | | bankstreet | String | Bank Street | | bankbuilding | String | Bank Building | | bankpostalcode | String | Bank postal code | | banktown | String | Bank town | | bankcountry | Date | Bank country | | username | String | User name | | password | Number | Password | | participatecode | Number | Participate code | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Login successful | | bankid | Number | Bank ID | | bankname | String | Name of the bank | | bankstreet | String | Bank Street | | bankbuilding | String | Bank Building | | bankpostalcode | String | Bank postal code | | banktown | String | Bank town | | bankcountry | Date | Bank country | | username | String | User name | | password | Number | Password | | participatecode | Number | Participate code | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | FAST\_BANK | |

### POST /fast/deletefastbank

Delete fast bank

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/fast/deletefastbank |
| Function Description | Delete fast bank |
| Content-type | application/json |
| Request Body | {  "bankid": "KRTHKHPPXXX"  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | bankid | String | Id of fast bank | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | |
| Source data | |  |  | | --- | --- | | Type | Name | | Database | SQL Server | | Table | FAST\_BANK | |

### POST /fast/getreceiveracno

Verify fast account number

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/fast/getreceiveracno |
| Function Description | Verify fast account number |
| Content-type | application/json |
| Request Body | {  "biccd": "KRTHKHPPXXX",  "account": "123456789"  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | biccd | String | Code of fast bank | | account | String | Account number | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | |
| Source data | |  |  | | --- | --- | | Type | Name | |

### POST /fast/makefullfundtransfer

FAST make full fund transfer

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/fast/makefullfundtransfer |
| Function Description | FAST make full fund transfer |
| Content-type | application/json |
| Request Body | {  "ipctranid":"12943"  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | ipctranid | String | Transaction id | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | |
| Source data | |  |  | | --- | --- | | Type | Name | |

### POST /fast/makehalffundtransfer

FAST make half fund transfer

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/fast/makehalffundtransfer |
| Function Description | FAST make half fund transfer |
| Content-type | application/json |
| Request Body | {  "ipctranid":"12943"  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | ipctranid | String | Transaction id | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | |
| Source data | |  |  | | --- | --- | | Type | Name | |

### POST /fast/reversetransaction

Reverse FAST transaction

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/fast/reversetransaction |
| Function Description | Reverse FAST transaction |
| Content-type | application/json |
| Request Body | {  "ipctranid":"12943"  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | ipctranid | String | Transaction id | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | |
| Source data | |  |  | | --- | --- | | Type | Name | |

### POST /fast/makeacknowledgement

FAST make acknowledgement

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/fast/makeacknowledgement |
| Function Description | FAST make acknowledgement |
| Content-type | application/json |
| Request Body | {  "ipctranid":"12943"  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | ipctranid | String | Transaction id | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | |
| Source data | |  |  | | --- | --- | | Type | Name | |

### POST /fast/accountinquiry

Get information FAST account

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/fast/accountinquiry |
| Function Description | Get information FAST account |
| Content-type | application/json |
| Request Body | {  "participant\_code": 32435436  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | participant\_code | String | Account number | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | |
| Source data | |  |  | | --- | --- | | Type | Name | |

### POST /fast/incomingtransaction

Get FAST incoming transaction

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/fast/incomingtransaction |
| Function Description | Get FAST incoming transaction |
| Content-type | application/json |
| Request Body | {  "payee\_participant\_code": "32435436",  "ccy":"USD"  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | participant\_code | String | Account number | | ccy | String | Currency | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | |
| Source data | |  |  | | --- | --- | | Type | Name | |

### POST /fast/outgoingtransaction

Get FAST outgoing transaction

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/fast/outgoingtransaction |
| Function Description | Get FAST outgoing transaction |
| Content-type | application/json |
| Request Body | {  "payer\_participant\_code": "32435436",  "instruction\_ref": "LETMEIN"  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | participant\_code | String | Account number | | instruction\_ref | String |  | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | |
| Source data | |  |  | | --- | --- | | Type | Name | |

### POST /fast/syncoutgoingtran

FAST sync outgoing transaction

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/fast/syncoutgoingtran |
| Function Description | FAST sync outgoing transaction |
| Content-type | application/json |
| Request Body | {  "SESSIONID": "00000178-d477-51e4-0000-0178d47751ed"  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | SESSIONID | String | SESSION ID of O9 | |

RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | |
| Source data | |  |  | | --- | --- | | Type | Name | |

### POST /fast/syncincomingtran

FAST sync incoming transaction

REQUEST

|  |  |
| --- | --- |
| Request type | POST |
| URL | http://{{IP\_ADDRESS}}/fast/syncincomingtran |
| Function Description | FAST sync incoming transaction |
| Content-type | application/json |
| Request Body | {  "SESSIONID": "00000178-d477-51e4-0000-0178d47751ed"  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | SESSIONID | String | SESSION ID of O9 | |

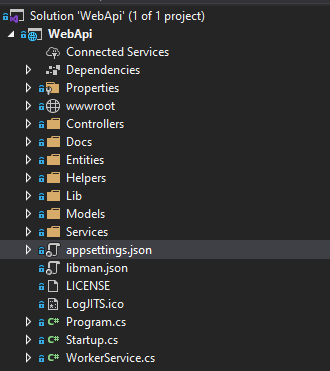
RESPONSE

|  |  |
| --- | --- |
| Status code | Success: 201  Wrong Format/Data: 400 or 415  Failure : 500 |
| Response Body Result | {  "errorcode": 0,  "messagedetail": "Successful",  "result": null  } |
| Field Description | |  |  |  | | --- | --- | --- | | Name | Type | Description | | errorcode | Number | 0: Success; 9998: Failure | | messagedetail | String | Successful/  Failure | |
| Source data | |  |  | | --- | --- | | Type | Name | |

# Document for developer

## Directory Structure Project

* Open project WebApi with Visual Studio 2019:





|  |  |
| --- | --- |
| Folders | Description |
| Properties | the default folder of .net core framework |
| wwwroot | configure webui for api |
| Controllers | configure routers to redirect urls to corresponding handler functions |
| Entities | configuring objects mapped from the database |
| Helpers | configuring objects mapped from the database |
| Lib | storing third party libraries |
| Models | configure objects as output or input to apis |
| Services | this is the most important folder, is where the main processing code for all apis |

* There are also some files to pay attention to:

|  |  |
| --- | --- |
| Files | Description |
| appsettings.json | the main configuration for the API system |
| Startup.cs | where the code that runs the API initializers and configurators |
| WorkerService.cs | contains code that runs in the background on a schedule |
| Helpers\Common\Codetypes.cs | file contains error messages |
| Helpers\Common\GlobalVariable.cs | contains environment variables |

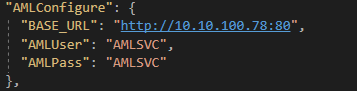
## Webapi Configuration Details

* Open file appsettings.json with Visual Studio 2019, we see the following configurations:
  + AppSettings: configure connections and authentication types for core API



|  |  |
| --- | --- |
| Keys | Description |
| Secret | Secret key to generate JWT token string |
| DatabaseProvider | name of the type of database used |
| DBConnectionString | the connection string to the database, encrypted |
| Memcached | Connection string to Memcached |
| PoolConnection | the number of connections of an ActiveMQ pool |
| ExpiryDateJWT | number of days of existence of JWT token |
| UsingJWT | whether to use JWT authentication or not. true: yes, false: no |
| UsingLicense | whether to use license authentication or not. true: yes, false: no |

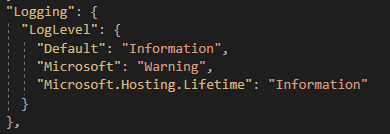
* + AMLConfigure: Configure connection to AML server



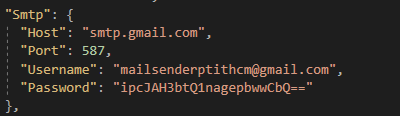
* + DefaultUser: Configure default users of o9core user



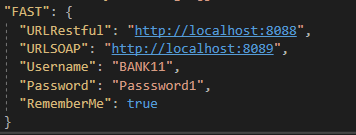
* + Logging: Configure Log mode for api



* + Smtp: Configure mail server for api



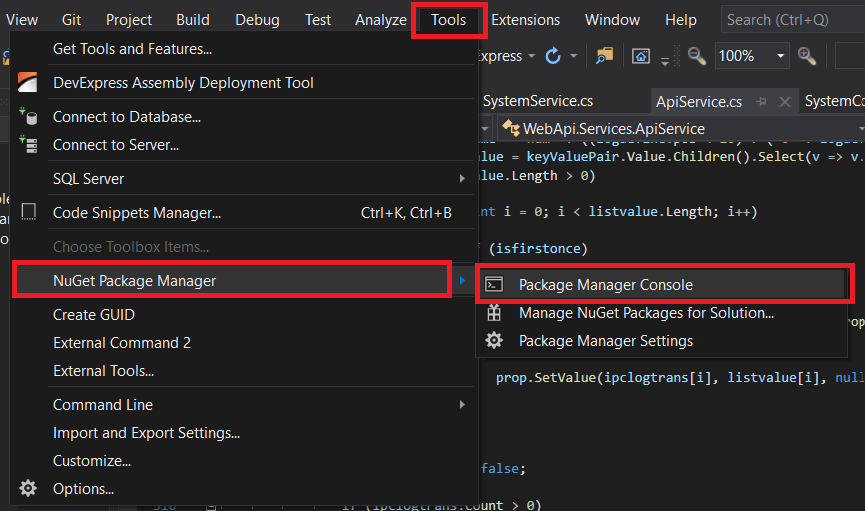
* + FAST: configure connections to the FAST server



|  |  |
| --- | --- |
| Keys | Description |
| URLRestful | the address of the Restful webservice of NBC FAST |
| URLSOAP | the address of the SOAP webservice of NBC FAST |
| Username | CM username |
| Password | CM password |

## Automatic Build ENTITIES

* Click Tool -> NuGet Package Manger -> Package Manger Console



* Type the following command into the Console window and then press Enter:

Scaffold-DbContext "Server=localhost;Database=FAST;Trusted\_Connection=True;" Microsoft.EntityFrameworkCore.SqlServer -OutputDir Entities -t S\_BANK

Details :

|  |  |
| --- | --- |
| Keys | Description |
| Server=localhost;Database=FAST;  Trusted\_Connection=True; | the connection string to the database |
| Microsoft.EntityFrameworkCore.SqlServer | library to migrate data |
| Entities | the directory containing entities |
| S\_BANK | is the table name. In the case of multiple tables, we separate the tables by commas |

* Note: After the build, it will create [database name]Context.cs we copy the data in the file and paste each corresponding part into the Helpers\DatabaseUtils\DataContext.cs file

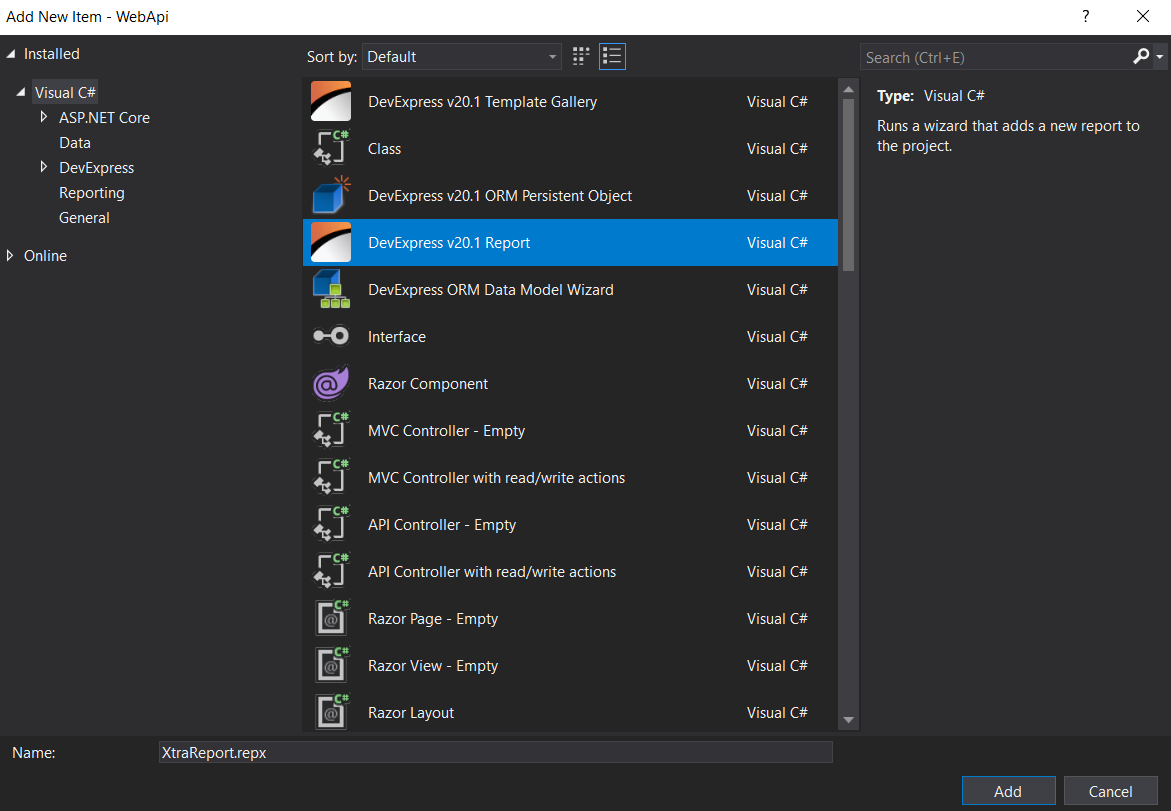
## Reusable Functions

|  |  |  |
| --- | --- | --- |
| Files | Functions | Description |
| Services\ApiService.cs | **CallFrontOffice**(TransactionRequest model, string txcode, bool isMapping = true, Users user = null) | call O9core's Front Office transaction |
| **CallFunction**(TransactionRequest model, string txproc, Users user = null) | call a function of O9core |
| Helpers\DatabaseUtils\DbUtils.cs | **GetVariableFromStoredProcedure**(string spname, IDictionary<string, object> spparams) | execute a query that returns a data variable by calling SP. **Note**: SP must be return 1 table with 1 column name is result |
| **GetDataTableFromStoredProcedure**(string spname, IDictionary<string, object> spparams=null) | execute a query that returns a table of data by calling SP |
| **FAST\_CreateMessageXML**(string trancode, object sourcedata) | generate an XML document from the configurator on the database |
| **GetEmailTemplate**(int bankid, string channelid, string trancode) | get email template from database |
| **GetDataTableFromSQL**(string sql) | execute a query that returns a table of data by calling the SQL statement |
| Services\RestClientService.cs | **CallRestAPI**(string url, string method, object requestBody, RestHeaderRequest header = null) | call a Restful webservice |
| Helpers\Utils\Utility.cs | **ConvertStringToDateTime**(string datetimeString) | convert string to datetime |
| **ConvertDateTimeToString**(DateTime datetime) | convert datetime to string format dd/MM/yyyy |
| **ConvertDateTimeToStringDatetime**(DateTime datetime) | convert datetime to string format dd/MM/yyyy HH:mm:ss |
| **IsDate**(string tempDate) | Check if the string is in the correct date time format |
| **SetValueObject**(JObject jObject, object obj) | get data from JObject and assign it to any object (requires the JObject's key to match the object's property name) |
| **FromatJObject**(object obj, string[] removefields = null) | refactor the JObject to return the result to the client |
| **FormatKeyJArray**(JArray jArray) | convert JArray keys to lowercase before returning the result to the client |
| **GetValueJObjectByKey**(JObject jObject, string KeyName) | get value of JObject by its key |
| **GetValueObjectFromName**(object obj, string name) | get the value of the object by its name. Example: ISOMessage.messageid |

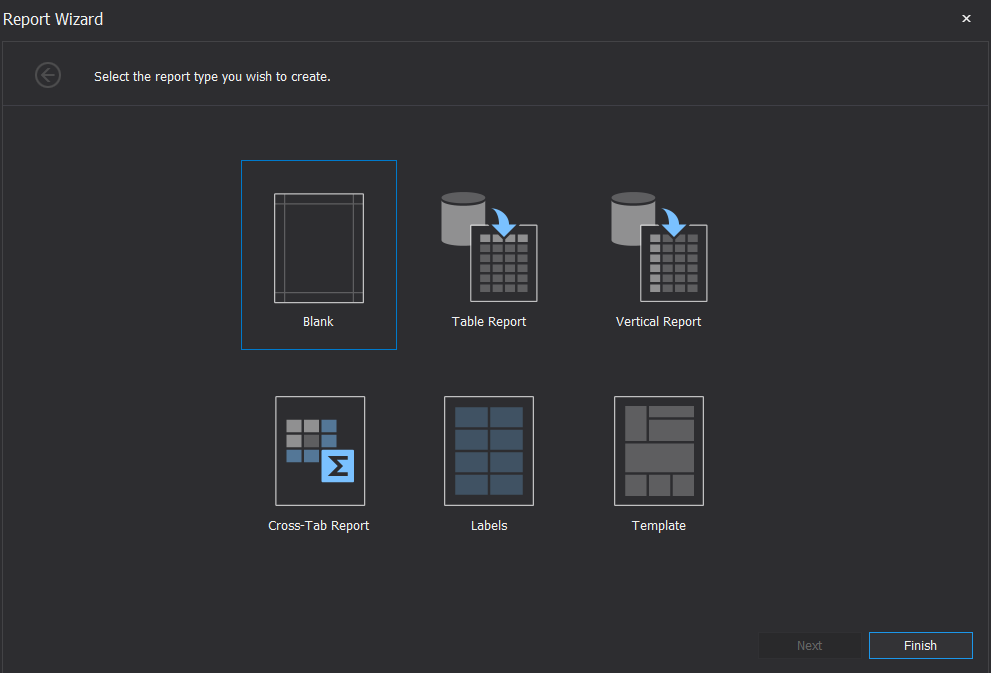
## Instructions For Creating A Report

### Step 1: Create new report

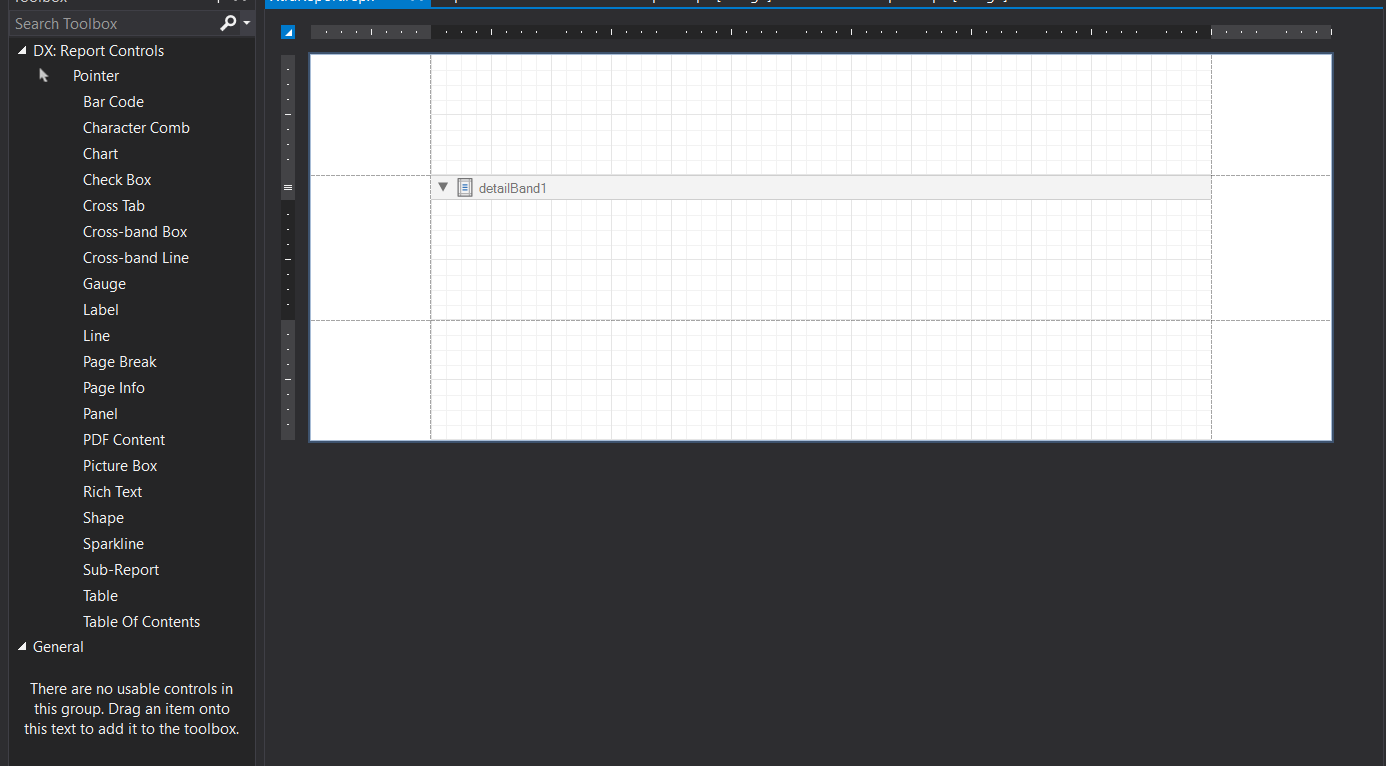
* Click to Reports folder -> Add -> New Item -> Choose DevExpress report, rename and click Add

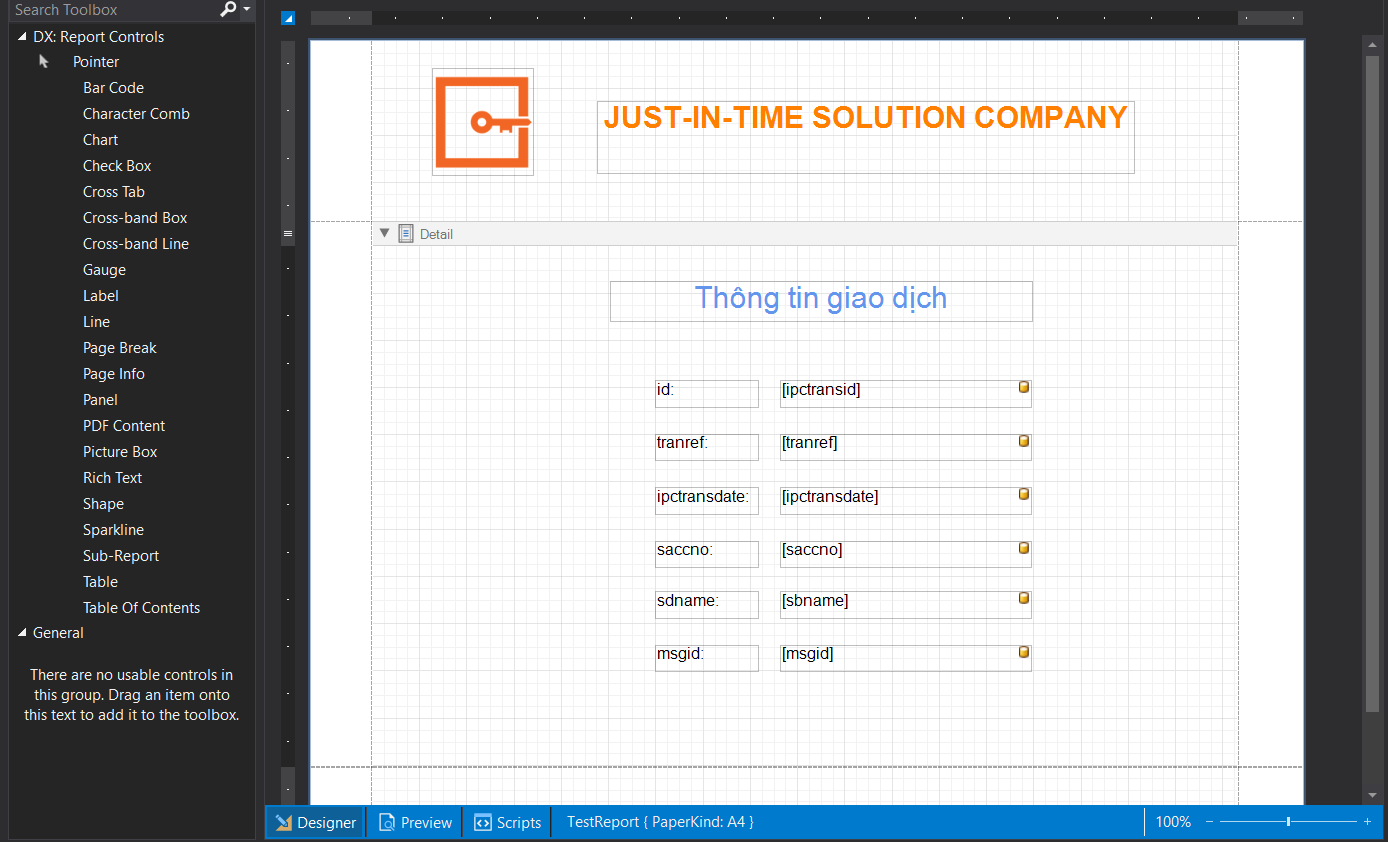


* Choose Blank -> Finish

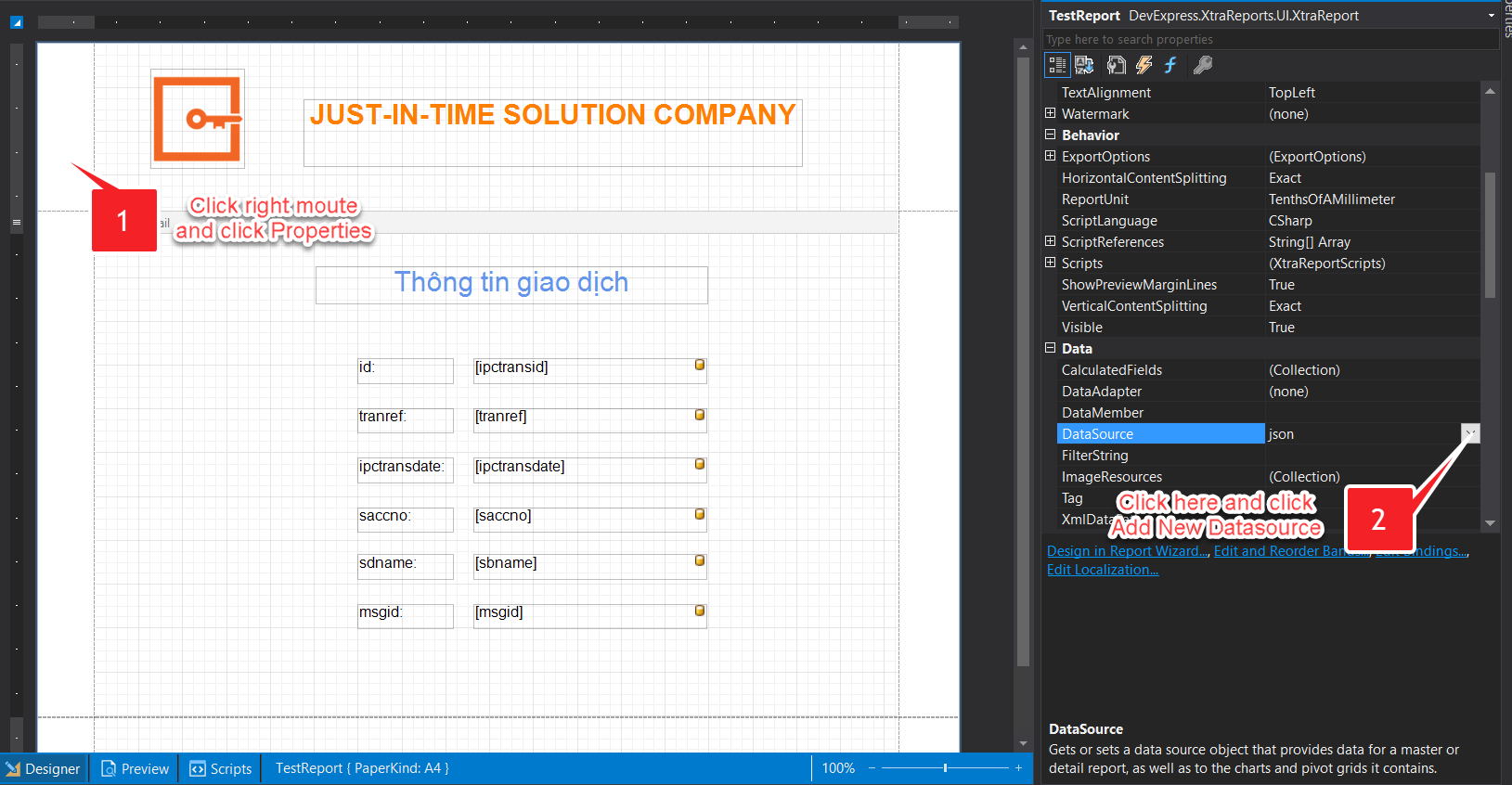


* Drag and drop controls on Toolbox for report

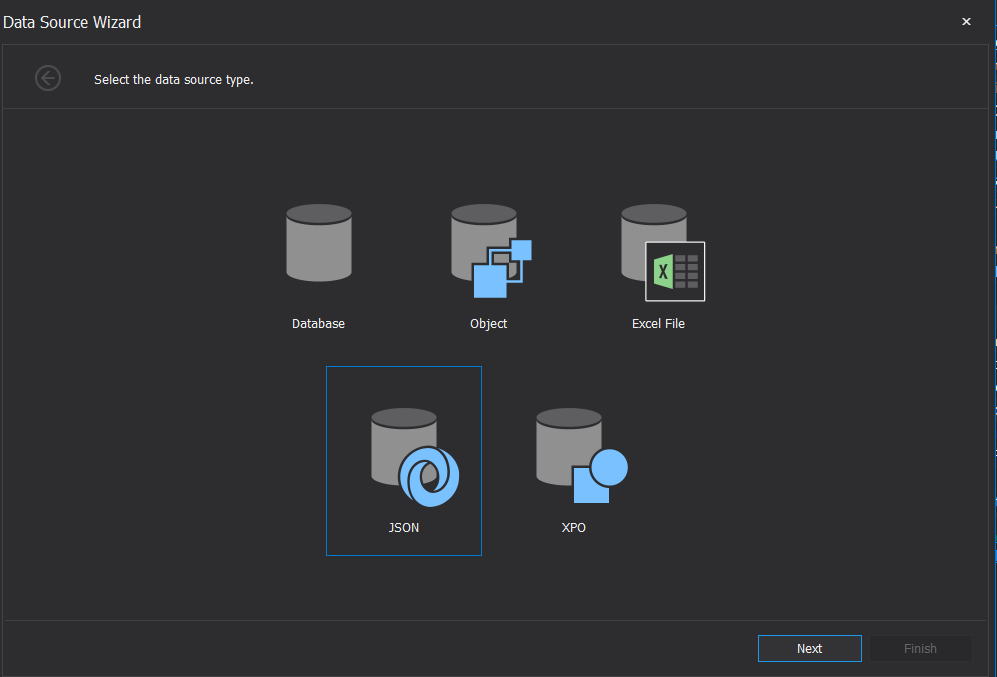




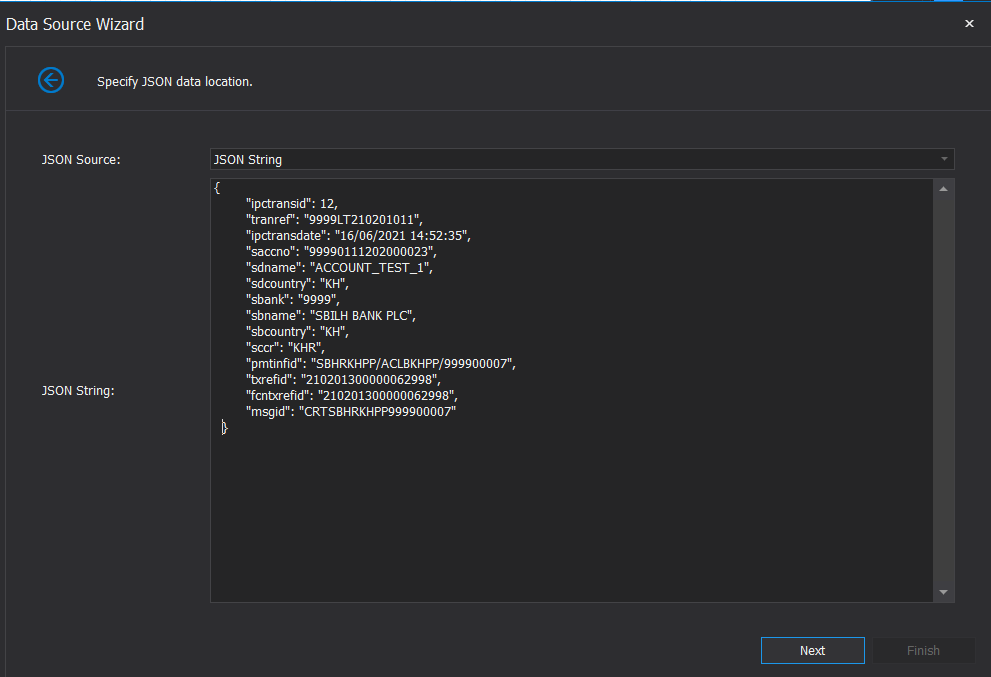
* Add new data source

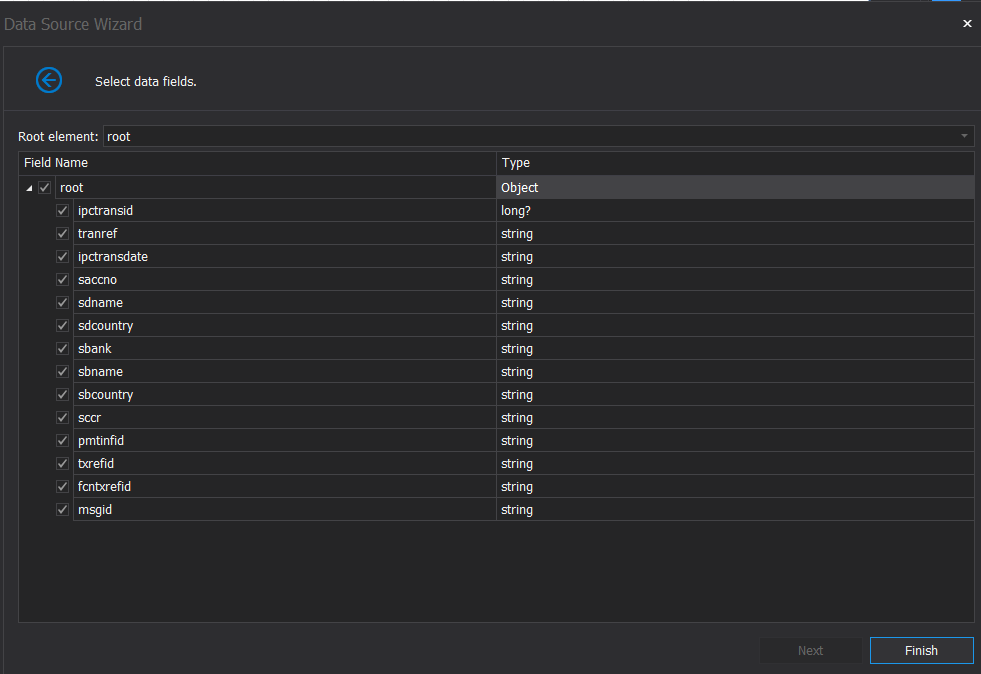


* Choose JSON -> Next

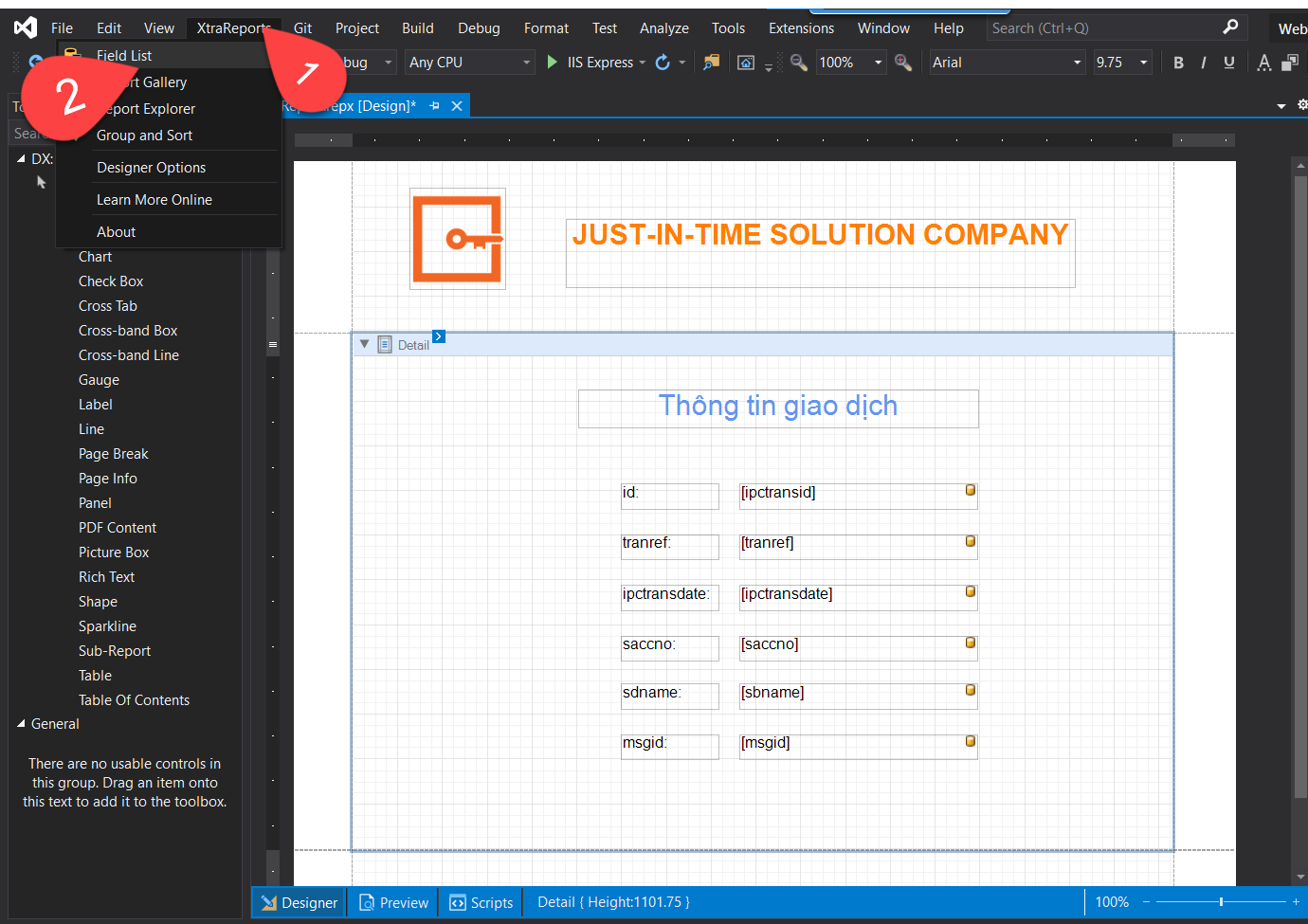


* Choose JSON Source: JSON String and paste JSON source template to JSON String. Then click Next -> Finish

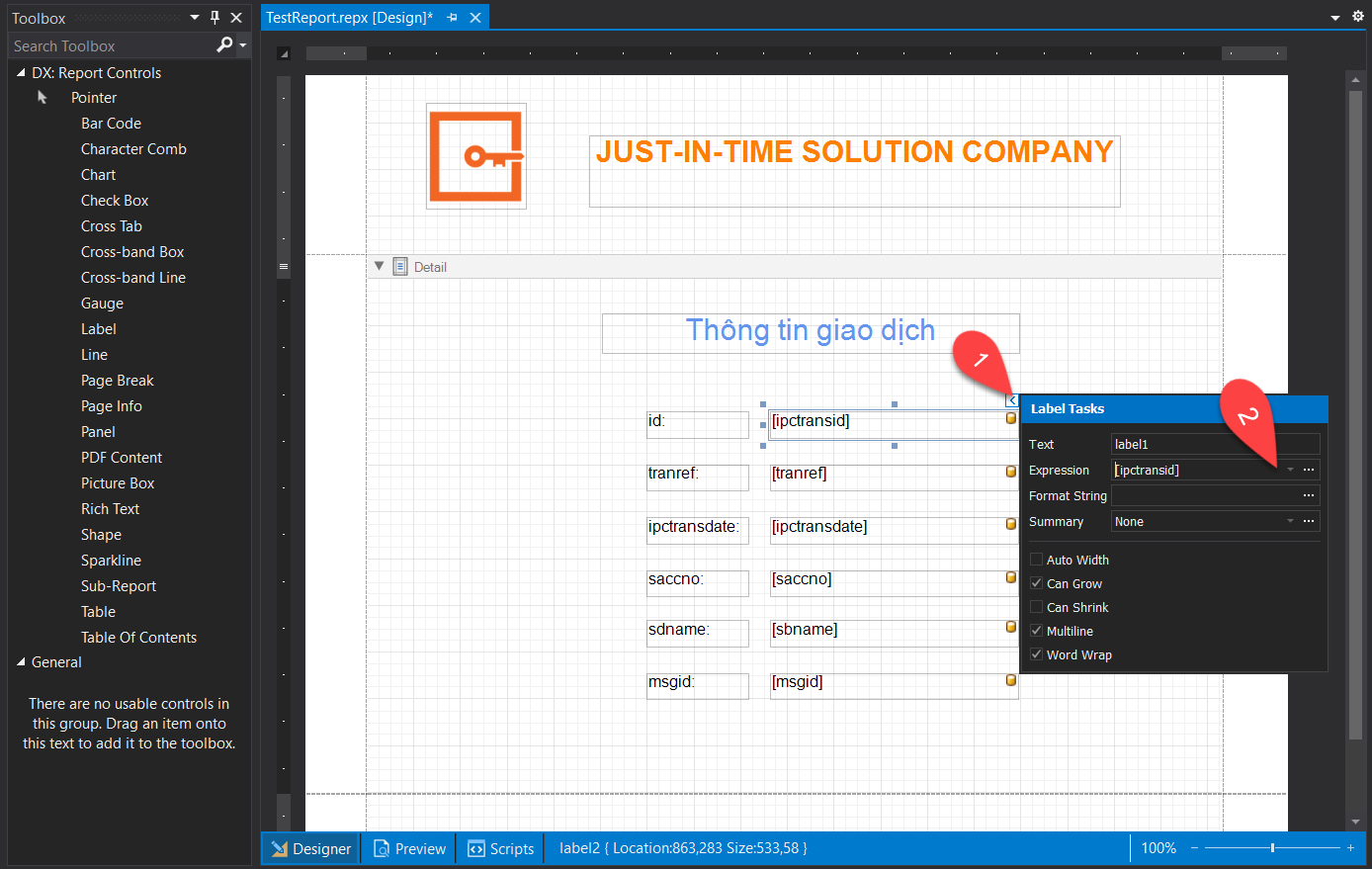




* To review the Datasource we choose XtraReport -> Field List



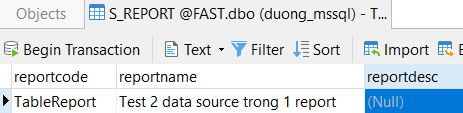
* To select datasource for any field



Find out more at: [Reporting | Reporting | DevExpress Documentation](https://docs.devexpress.com/XtraReports/2162/reporting)

### Step 2: Config in database

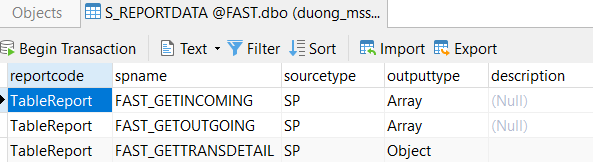
* Step 1: config on table S\_REPORT



Details:

|  |  |
| --- | --- |
| Columns | Description |
| reportcode | the file name(.repx) of the report saved on the server side |
| reportname | the name of the report |
| reportdesc | the description of the report |

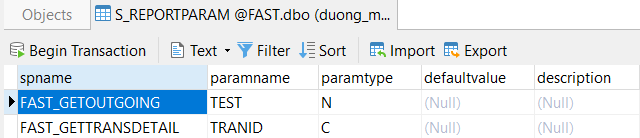
* Step 2: Config data source for report at table S\_REPORTDATA:



Details:

|  |  |
| --- | --- |
| Columns | Description |
| reportcode | the file name(.repx) of the report saved on the server side |
| spname | Name of Stored procedure in database, where supply data |
| sourcetype | SP: Stored procedure |
| outputtype | Array: Stored procedure return a table a lot of rows  Object: Stored procedure return a table have only 1 row |
| description | the description of the report data source |

* Step 3: Config parameters of datasource at table S\_REPORTPARAM

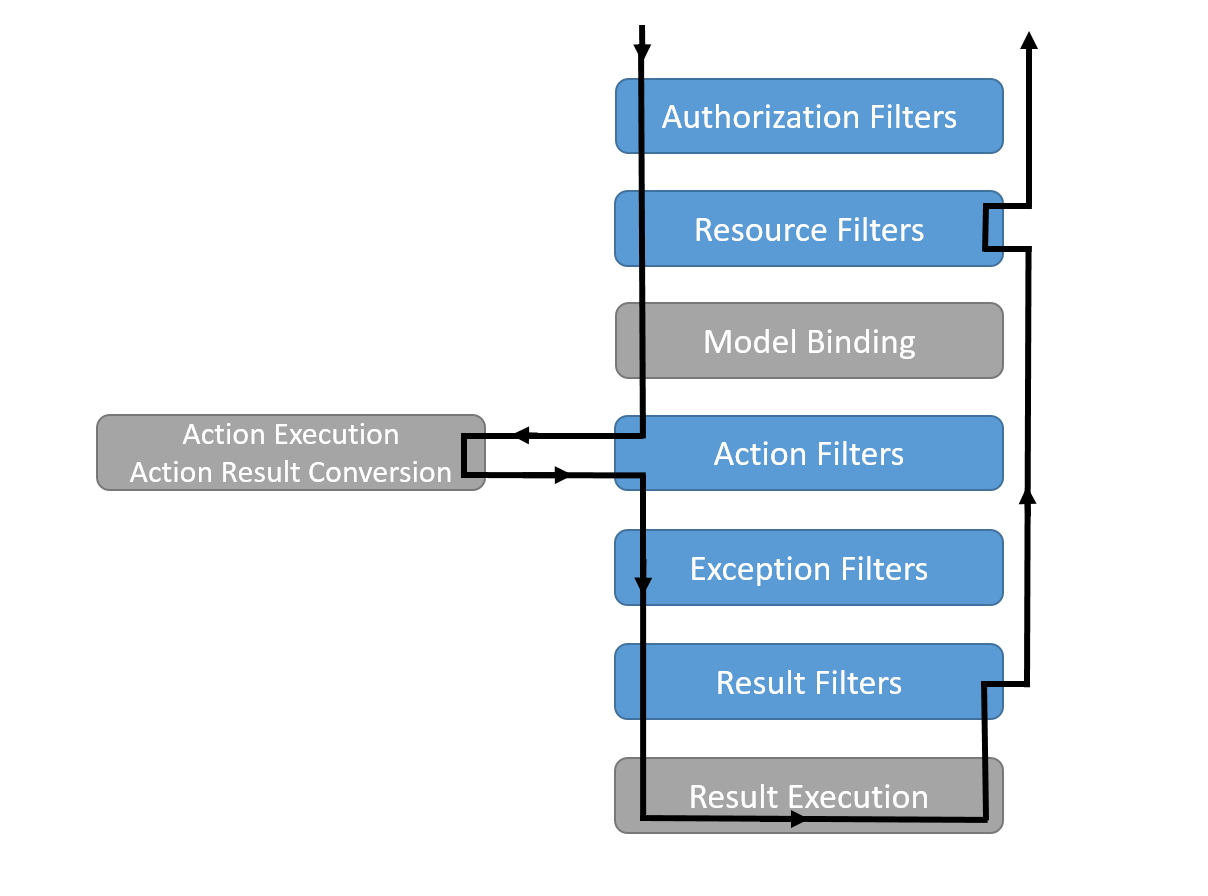


Details:

|  |  |
| --- | --- |
| Columns | Description |
| spname | Name of Stored procedure in database, where supply data |
| paramname | Name of parameter of Stored procedure |
| paramtype | N: Number  C: Character |
| defaultvalue | Default value of param |
| description | the description of the report param |

## Api Programming Tutorial

### Microsoft .NET Core API Workflow



Each filter type is executed at a different stage in the filter pipeline:

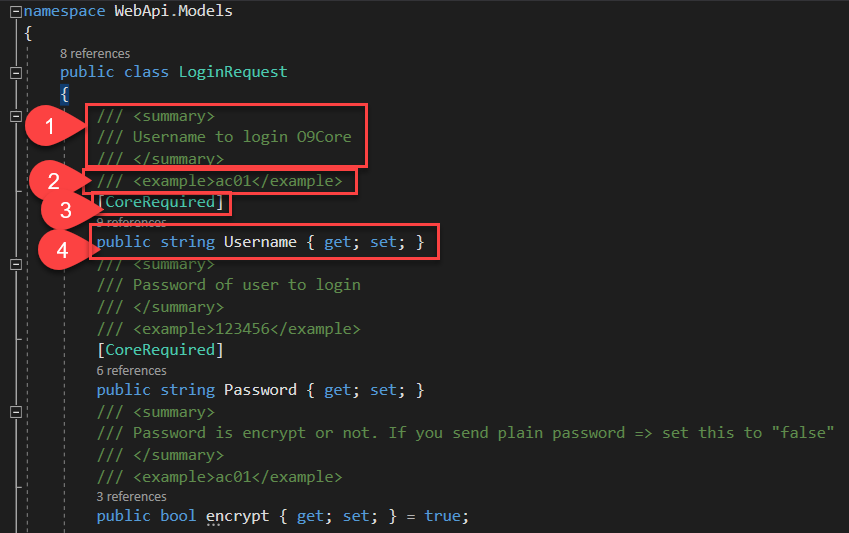
* Authorization filters run first and are used to determine whether the user is authorized for the request. Authorization filters short-circuit the pipeline if the request is not authorized.
* Resource filters:

Run after authorization.

* + OnResourceExecuting runs code before the rest of the filter pipeline. For example, OnResourceExecuting runs code before model binding.
  + OnResourceExecuted runs code after the rest of the pipeline has completed.
* Model Binding maps body of request to object defined of user.
* Action filters:
  + Run code immediately before and after an action method is called.
  + Can change the arguments passed into an action.
  + Can change the result returned from the action.
  + Are not supported in Razor Pages.
* Action Execution include Controller and Service, which primary handle data for api.
* Exception filters apply global policies to unhandled exceptions that occur before the response body has been written to.
* Result filters run code immediately before and after the execution of action results. They run only when the action method has executed successfully. They are useful for logic that must surround view or formatter execution.

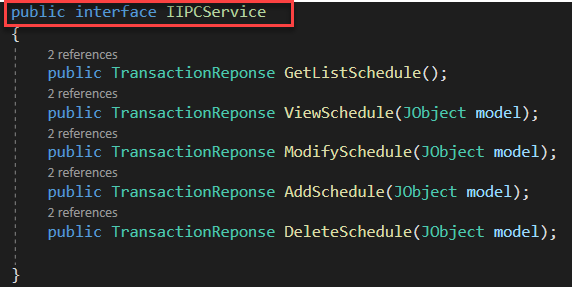
### Tutorial For API Not Connecting To O9core

* Step 1 : Create object request to map request from client
  + In folder Models, create new class

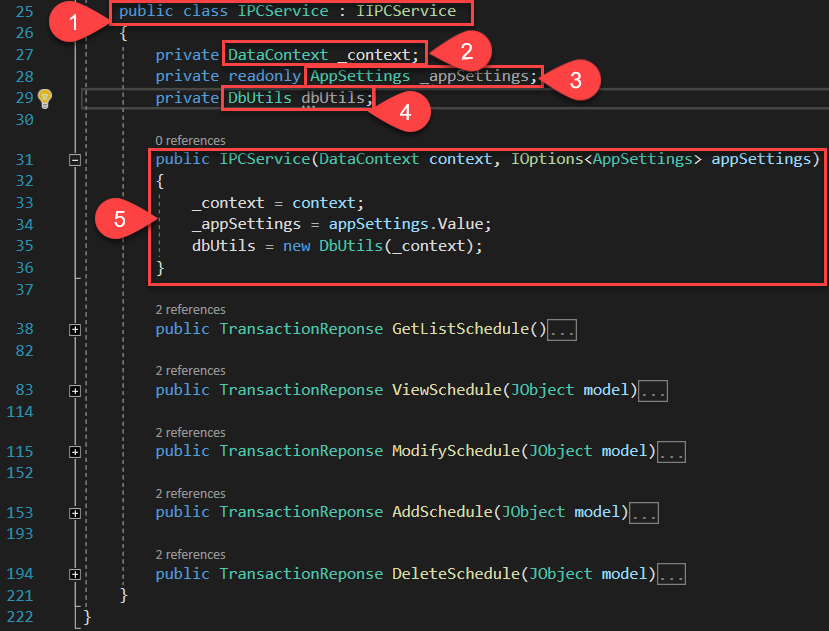


|  |  |
| --- | --- |
| Keys | Description |
| 1 | Tag summary is description for this file. Note: must set the correct structure: ///<summary>text</summary> |
| 2 | Tag example is show example value for this field. Note: must set the correct structure: ///<example>text</example> |
| 3 | This is attribute for required filed. View all attribite at file Helpers\Attribute\CoreAttribute.cs |
| 4 | Declare getter setter for this field |

* Step 2 : Create Service interface and Service class
  + In folder Service, create new class and code Service interface on this class, declare all function of service

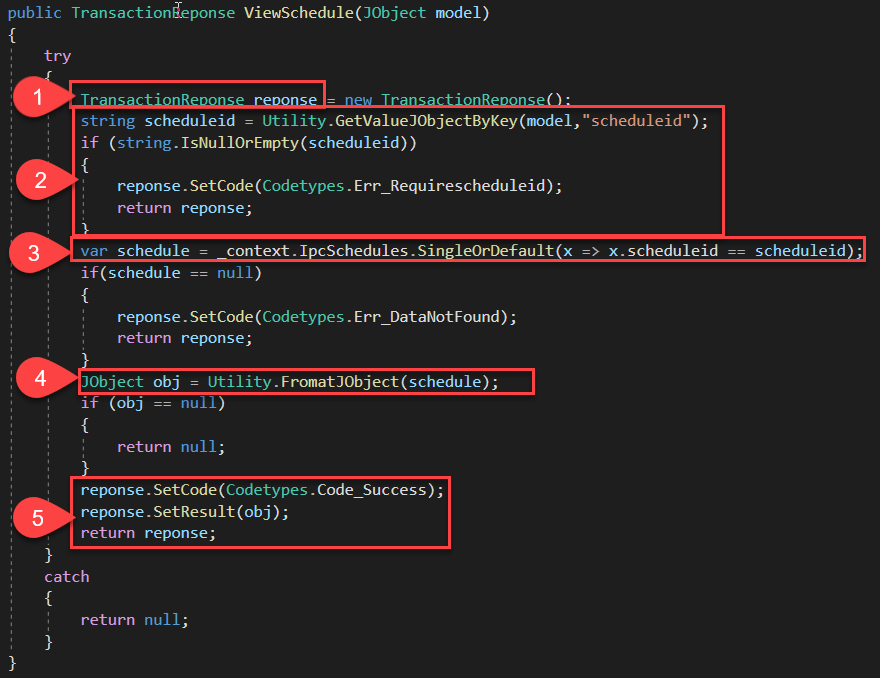


* + In this file, code Service class and all function declared in interface:



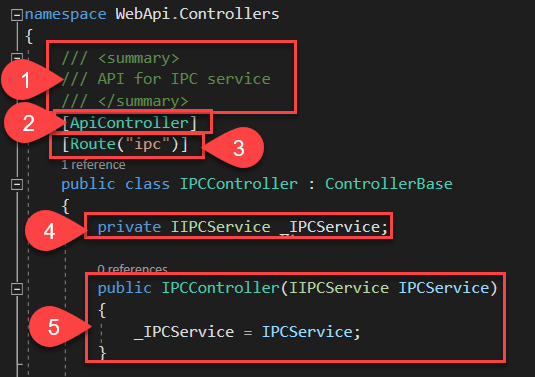
|  |  |
| --- | --- |
| Keys | Description |
| 1 | Declare class extend the interface |
| 2 | This class mapped from database to work with database, base on Entity Framework |
| 3 | This class includes api configuration |
| 4 | This class is to get information from database |
| 5 | This is base constructor of all service |

* + In this file, code function handles data for api



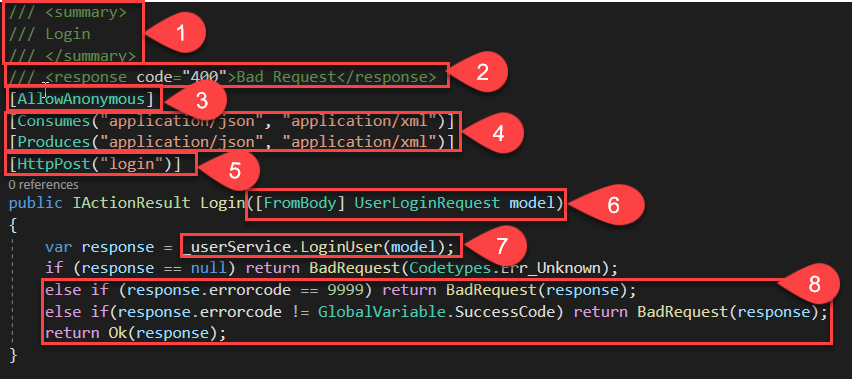
|  |  |
| --- | --- |
| Keys | Description |
| 1 | Object reponse data for all service |
| 2 | Check some field of request object |
| 3 | Get data from database, base on Entity Framework |
| 4 | Convert object to Jobject follow format of CoreAPI |
| 5 | This is base constructor of all service |

* Step 3 : Create a controller to receive the request from client and navigate to the handler function and then return the result to the client
  + Declare controller



|  |  |
| --- | --- |
| Keys | Description |
| 1 | Tag summary is description for this api. Note: must set the correct structure: ///<summary>text</summary> |
| 2 | Attribute for mark this class is api controller |
| 3 | Attribute for config route api from url |
| 4 | Declare service for handling data |
| 5 | This is base constructor of all controller |

* + Code controller for api

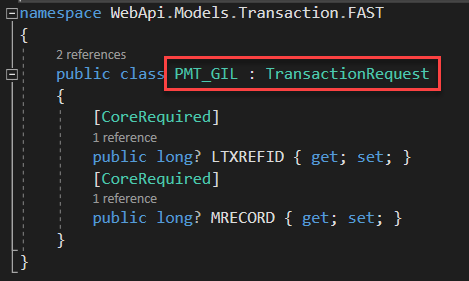


|  |  |
| --- | --- |
| Keys | Description |
| 1 | Tag summary is description for this api. Note: must set the correct structure: ///<summary>text</summary> |
| 2 | Tag response is desciption all repnose code for this api. Note: must set the correct structure: /// <response code="400">Bad Request</response> |
| 3 | Attribute to allow all client connect to api. If you want to required check JWT token, you can use Authorize attribute |
| 4 | Attribute to declare af content type of request |
| 5 | Method of api and route for it |
| 6 | Object mapping with body of request |
| 7 | Function in service, which handles data for api |
| 8 | Set reponse body and reponse code for client |

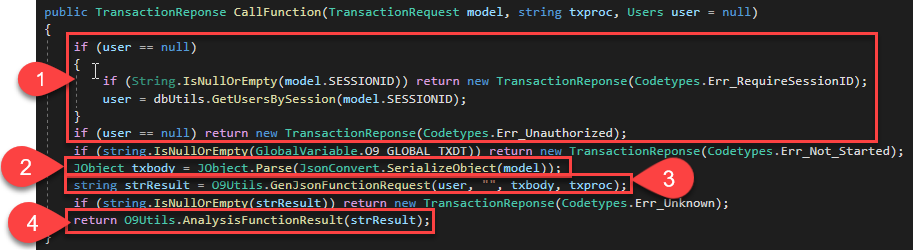
### Tutorial For API Connecting To O9core

Similar to API not connecting to O9core, but there are some differences as follows:

* Step 1 : Create object request to map request from client, it must be extend TransactionRequest class

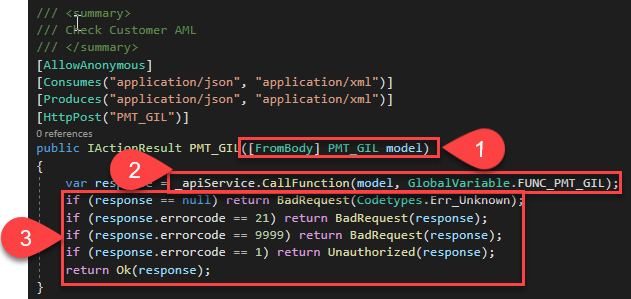


* Step 2 : Create Service interface and Service class. Similar to API not connecting to O9core, but there are some differences



|  |  |
| --- | --- |
| Keys | Description |
| 1 | Check sessionid |
| 2 | Prepare object to send to hosting |
| 3 | Send request to hosting |
| 4 | Format object to return reponse to client |

* Step 3 : Create a controller to receive the request from client and navigate to the handler function and then return the result to the client



|  |  |
| --- | --- |
| Keys | Description |
| 1 | Object mapping with body of request |
| 2 | Function in service, which handles data for api |
| 3 | Set reponse body and reponse code for client |