Scollect API Documentation

Introduction

Welcome to **Scollect's official API**. This tool allows you to access collection data from a variety of clients and agencies, eliminating the need for direct server connections. You don't need to use Scollect to take advantage of this API, but you do need a basic understanding of **APIs and Bearer tokens**. This document assumes you have this knowledge.

If you need to refresh your understanding of **REST APIs**, please refer to the provided reference.

All API calls will use the endpoint:
`https://api.smaxcollectionsoftware.com:8000`
For simplicity, this will be referred to as **/api** throughout this document.

We'll explore each API call, demonstrating **the request and response bodies**. Please note that some text types in the response body will have specific formats. For example:

- **Dates**: `yyyy-mm-dd`
- **Social Security Numbers (SSN)**: `XXX-XX-XXXX`
- **Phone Numbers**: `XXX-XXX-XXXX`

Important Note

If you see "state": "Success" in the error sections throughout this document, note that this does not indicate a successful operation. It simply means that the API server was reachable and processed the request, but an error may have still occurred.

Using Postman

This document assumes you are familiar with **Postman**, a popular API testing tool that allows developers to easily interact with and debug REST APIs. **Postman** provides a user-friendly interface to construct HTTP requests, add headers, authenticate using Bearer tokens, and analyze API responses.

All examples in this document will be presented using the **Postman application**. If you are new to Postman, you may want to visit their official documentation to understand how to:

- Set up API requests
- Manage authentication tokens
- View response data
- Save and organize API collections

Using Postman to Test the API

To interact with the API, you will use Postman to send requests in JSON format. Follow these steps to configure your request properly:

Step 1: Open Postman

Ensure you have Postman installed and running on your system.

Step 2: Set the HTTP Method to POST or GET

Select GET or POST as the request method, depending on the API specification.

Step 3: Enter the API Endpoint URL

Input the following endpoint in the request URL field:
https://api.smaxcollectionsoftware.com:8000/<apiname>

Step 4: Configure the Request Body (if specified)

- Click on the Body tab.
- Select the raw radio button.
- Change the format to JSON (application/json).
- Enter the following JSON payload: (depending on the request)

Step 5: Add Headers

- Click on the Headers tab.
- Ensure you have the following header:

  Key: Content-Type
  Value: application/json

Step 6: Add Authorization (Bearer Token)

If the API requires authentication using a Bearer Token, follow these steps:

1. Navigate to the Authorization tab in Postman.
2. From the Auth Type dropdown menu, select Bearer Token.
3. In the Token field, enter the token value you received from the authentication API.
4. Postman will automatically include the token in the request header when sending the request.

Authentication-related API calls will be marked with a star (⭐) for easy identification throughout this document.

POST: Login

The /login endpoint enables users to log in using their API key and PIN. Upon successful authentication, an access token is generated and returned in the response, allowing the user to access protected resources.

Request Body

```
{
  "apikey": "BwsxgFT1cj4gbpmHnMCU" (text),
  "pin": "C2C"  (text)
}
```

Response

```
{
   "result": {
      "access_token":
"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJmcmVzaCI6ZmFsc2UsImlhdCI6MTc0MjIyMj
kwNywianRpIjoiYWQ0YjAzYmItYmU1Ny00N2I0LThmMzktYjhjYzUxZjdiYjYxIiwidHlwZSI6I
mFjY2VzcyIsInN1YiI6IkJ3c3hnRlQxY2o0Z2JwbUhuTUNVIiwibmJmIjoxNzQyMjIyOTA3LCJj
c3JmIjoiOWJiNzkwOTMtZWVlMi00NzlkLTg2ODItNmQ0N2M4ZDM4YzImIiwiZXhwIjoxNzQy
MjIzODA3LCJwaW4iOiJDMkMifQ.auztODpuR6ya4CCM8UbmceV5aAFoeS4AKONUvm05In
I"
   },
   "state": "SUCCESS"
}
```

Error

```
{
   "result": [
      {
         "error": "Invalid API Credentials."
      },
      401
   ],
   "state": "SUCCESS"
}
```

GET: /getportfoliolist

This endpoint retrieves a list of portfolios by making an internal request to a private API. It requires JWT authentication, extracts the token from the Authorization header, and includes it in the internal request headers. The response returns a task ID, which is used to check and return the status of the request. Error handling is included for token expiration and general exceptions.

Request Body

There is no request body in this endpoint.

Response

```
{
  "result": [
    {
      "client": "C2C",
      "dateplaced": "Thu, 01 Dec 2022 00:00:00 GMT",
      "id": 707,
      "portfolio": "C2CLA 12.1.22"
    },
    {
      "client": "C2C",
      "dateplaced": "Thu, 01 Dec 2022 00:00:00 GMT",
      "id": 708,
      "portfolio": "C2C PDL TEST"
    },
    {
      "client": "C2C",
      "dateplaced": "Thu, 08 Dec 2022 00:00:00 GMT",
      "id": 709,
      "portfolio": "C2C Demo File to Load_120522"
    },
    {
      "client": "Concepts2Code",
      "dateplaced": "Wed, 06 Mar 2024 00:00:00 GMT",
      "id": 710,
      "portfolio": "C2C 101 120"
    },
    {
      "client": "Concepts2Code",
      "dateplaced": "Thu, 14 Mar 2024 00:00:00 GMT",
      "id": 711,
      "portfolio": "Buffalo Credit"
    },
    {
      "client": "Concepts2Code",
      "dateplaced": "Tue, 26 Mar 2024 00:00:00 GMT",
      "id": 712,
      "portfolio": "Buffalo Loans"
    },
    {
      "client": "C2C",
      "dateplaced": "Tue, 19 Nov 2024 00:00:00 GMT",
      "id": 715,
      "portfolio": "New Added Portfolio"
    },
    {
      "client": "New Client",
      "dateplaced": "Thu, 13 Feb 2025 00:00:00 GMT",
      "id": 716,
      "portfolio": "NewPortfolioEdited"
    },
    {
      "client": "Concepts2Code",
      "dateplaced": "Thu, 20 Feb 2025 00:00:00 GMT",
      "id": 717,
      "portfolio": "NewPortfolio220256"
    },
    {
      "client": "C2C",
      "dateplaced": "Wed, 26 Mar 2025 00:00:00 GMT",
      "id": 720,
      "portfolio": "NewToday45"
    },
    {
      "client": "C2C",
      "dateplaced": "Wed, 02 Apr 2025 00:00:00 GMT",
      "id": 721,
      "portfolio": "NewPortfolioNameManual"
    },
    {
      "client": "New Client Manual",
      "dateplaced": "Tue, 15 Apr 2025 00:00:00 GMT",
      "id": 722,
      "portfolio": "New Portfolio Manual Name"
    }
  ],
  "state": "SUCCESS"
}
```

GET: /getaccountbysocial/<ssn>

This endpoint retrieves account information based on a provided social security number by forwarding the request to a private API. It requires a valid JWT token, extracts the token from the request header, and uses it to authorize the internal request. The response includes a task ID, which is then used to check and return the task's status. Error handling is in place for expired tokens and other exceptions.

Request Body

There is no request body in this endpoint. Instead, the ssn must be formatted as ###-##-####

Response

```
{
  "result": [
    {
      "accountnumber": "12345679",
      "accruedinterest": "0.00",
      "activated": null,
      "address": "15 MAIN ST",
      "address2": "",
      "adjustments": "0.00",
      "advancedfees": "0.00",
      "asset1value": "7,000",
      "assettype1": "Cash",
      "assettype2": "Vehicles",
      "assettype2value": " 4,000",
      "assettype3": "Collectables",
      "assettype3value": " 2,000",
      "assettype4": " ",
      "assettype4value": " ",
      "assettype5": " ",
      "assettype5value": " ",
      "assignedto": "OUT",
      "attorneyaddress": " ",
      "attorneyaddress2": " ",
      "attorneybeepernumber": " ",
      "attorneycellphone": " ",
      "attorneycity": " City",
      "attorneyemailaddress": " email@email.com",
      "attorneyfax": " ",
      "attorneyfees": "0.00",
"attorneyletters": null,
"attorneyname": " ATTORNEY CONTACT",
"attorneyphone": " ",
"attorneystate": "GA",
"attorneyzip": " ",
"authorizeduser": null,
"bankruptcytype": "",
"bankruptcyverificationstatus": "",
"beepernumber": null,
"birthdate": "10/18/1965",
"calculateinterest": null,
"callbacks": null,
"cellphone": null,
"chargedoffprincipal": "0.00",
"city": "GARDEN",
"claimstatus": null,
"claimtype": null,
"clientname": "C2C",
"clientnotes": null,
"closedreason": "",
"codebtor1address": " ",
"codebtor1address2": " ",
"codebtor1city": " ",
"codebtor1contactinfo": " ",
"codebtor1employeraddress": " ",
"codebtor1employercity": " ",
"codebtor1employername": " ",
"codebtor1employerstate": " ",
"codebtor1employerzip": " ",
"codebtor1firstname": " ",
"codebtor1friends": " ",
"codebtor1fullname": " NAME LAST",
"codebtor1lastname": " ",
"codebtor1middlename": " ",
"codebtor1occupation": " ",
"codebtor1primaryphone": " ",
"codebtor1secondaryphone": " ",
"codebtor1socialsecurity": " ",
"codebtor1state": " ",
"codebtor1zip": " ",
"codebtor2address": " ",
"codebtor2address2": " ",
"codebtor2city": " ",
"codebtor2contactinfo": " ",
"codebtor2employeraddress": " ",
"codebtor2employercity": " ",
"codebtor2employername": " SONSHINE SCHOOL",
"codebtor2employerstate": " ",
"codebtor2employerzip": " ",
"codebtor2firstname": " ",
"codebtor2friends": " ",
"codebtor2fullname": " CODEBTOR 2 CONTACT",
"codebtor2lastname": " ",
"codebtor2middlename": " ",
"codebtor2occupation": " ",
"codebtor2primaryphone": " 5556667777",
"codebtor2secondaryphone": " ",
"codebtor2socialsecurity": " ",
"codebtor2state": " ",
"codebtor2zip": " ",
"collectionfees": "0.00",
"collectorname": "Mark",
"contingencyamount": "100%",
"courtaddress": "",
"courtcity": "",
"courtfees": "0.00",
"courtfiled": "10/26/2023",
"courtphone": "1112223333",
"courtstate": "",
"courtzip": "",
"creditreport1": null,
"creditreport2": null,
"creditreport3": null,
"creditreport4": null,
"creditreport5": null,
"creditscore": " ",
"currentbalance": "809.00",
"currentlyemployed": null,
"custom1": "UNITED SOUTHERN BANK",
"custom10": " ",
"custom11": " ",
"custom12": " ",
"custom13": " ",
"custom14": " ",
"custom15": "",
"custom16": " ",
"custom17": " ",
"custom18": " ",
"custom19": " ",
"custom2": "063105285",
"custom20": " ",
"custom21": " ",
"custom22": " ",
"custom23": " ",
"custom24": " ",
"custom25": " ",
"custom26": " ",
"custom27": " ",
"custom28": " ",
"custom29": " ",
"custom3": "124",
"custom30": " ",
"custom31": " ",
"custom32": " ",
"custom33": " ",
"custom34": " ",
"custom35": " ",
"custom36": " ",
"custom37": " ",
"custom38": " ",
"custom39": " ",
"custom4": "1234",
"custom40": " ",
"custom41": " ",
"custom42": " ",
"custom43": " ",
"custom44": " ",
"custom45": " ",
"custom46": " ",
"custom47": " ",
"custom48": " ",
"custom49": " ",
"custom5": "Bank Info",
"custom50": " ",
"custom51": " ",
"custom52": " ",
"custom53": " ",
"custom54": " ",
"custom55": " ",
"custom56": " ",
"custom57": " ",
"custom58": " ",
        "custom59": " ",
        "custom6": "email@email.com",
        "custom60": " ",
        "custom7": " ",
        "custom8": " ",
        "custom9": " ",
        "customertype": null,
        "dateaccountopened": "3/6/2013",
        "datechargedoff": "5/20/2013",
        "dateclosed": "1900-01-01",
        "datefiled": "10/26/2023",
        "datelastassigned": "Mon, 05 Dec 2022 12:20:57 GMT",
        "datelastdialed": "Mon, 05 Dec 2022 00:00:00 GMT",
        "datelaststatuschg": "Mon, 05 Dec 2022 12:20:57 GMT",
        "dateplaced": "12/5/2022",
        "datereceived": " ",
        "daysassignedto": null,
        "daysinstatus": null,
        "dayslastworked": 0,
        "dayssincelastpayment": null,
        "deceased": "NO",
        "deceaseddate": "null",
        "delinquencydate": " ",
        "dialerstatus": null,
        "docketnumber": "",
        "emailaddress": "TEST@EMAIL.COM",
        "employer": "SONSHINE SCHOOL",
        "employeraddress": " ",
        "employercity": " ",
        "employerphone": "1112223333",
        "employerstate": " ",
        "employerzip": " ",
        "endtime": null,
        "filenumber": "R1050",
        "firstname": "CLAUDIA",
        "fullname": "CLAUDIA LAST NAME",
        "garnishment": "NO",
        "garnishmentamount": "7,000",
        "garnishmentdate": " 11/23/23",
        "garnishmentsource": " ",
        "garnishmenttype": " ",
        "goodnumbers": null,
        "groupname": "New Alert added to this account.\r\nAdditional Information
Added.",
"hasassets": "NO",
"hasduplicates": null,
"hasimages": null,
"hotlist": null,
"hotlistcomments": null,
"id": 51,
"interestpercentage": "0.00",
"jointaddress": " ",
"jointaddress2": " ",
"jointbirthdate": " ",
"jointcity": " ",
"jointemployer": " ",
"jointemployeraddress": " ",
"jointemployercity": " ",
"jointemployerstate": " ",
"jointemployerzip": " ",
"jointfirstname": " ",
"jointfullname": " CLAUDE LASTNAME",
"jointlastname": " ",
"jointmiddlename": " ",
"jointprimaryphone": " 1231231234",
"jointsecondaryphone": " ",
"jointsocial": " ",
"jointstate": " ",
"jointzip": " ",
"judgement": "NO",
"judgementdate": " 01/01/22",
"judgementfees": "0.00",
"judgementpercentage": " ",
"lastname": "LAST NAME",
"lastpaymentamount": "0.00",
"lastpaymentdate": " ",
"lastworked": "11/14/2023",
"latefees": "0.00",
"lein": "NO",
"leinamount": " ",
"leindate": " ",
"leintype": " ",
"makepayabletoaddress": null,
"makepayabletocity": null,
"makepayabletoname": null,
"makepayabletostate": null,
"makepayabletozip": null,
"managername": null,
"middlename": " ",
"miscfees": "0.00",
"nsffees": "0.00",
"occupation": " TEACHER",
"orderbydate": "Mon, 05 Dec 2022 00:00:00 GMT",
"originalbalance": "809.00",
"originalcreditor": "C2C Loan Advance",
"originalprincipal": "809.00",
"paidtodate": " ",
"pastdue": null,
"paymentplan": null,
"penaltyfees": "0.00",
"portfolio": "C2CLA 12.1.22",
"portfoliogroup": null,
"primaryphone": null,
"principal": "809.00",
"promises": null,
"reimbursedfees": "0.00",
"reportfilenumber": "",
"salesman": null,
"salesmanpercentage": null,
"skips": null,
"socialsecuritynumber": "111-22-3333",
"spouseemployer": " SONSHINE SCHOOL",
"spouseemployeraddress": " ",
"spouseemployercity": " ",
"spouseemployerphone": " ",
"spouseemployerstate": " ",
"spouseemployerzip": " ",
"spousename": " CLAUDE LASTNAME",
"spouseoccupation": "Teacher",
"spouseworkphone": " 1112223333",
"starttime": null,
"state": "AL",
"statuscolor": null,
"statusname": "OUTSOURCE",
"statustype": null,
"statuteoflimdate": "Sun, 31 Dec 1899 00:00:00 GMT",
"title": " ",
"typeofdebt": " ",
"verificationmethod": "",
"verified": "NO",
"verifieddate": "10/26/2023",
"workphone": null,
      "zip": "12345"
    },
```

Error

```
{
  "result": {
    "error": "SSN of 111-22-333 not found"
  },
  "state": "SUCCESS"
}
```

POST: get_accounts_in_portfolio

This endpoint is a POST request that requires a valid JWT token in the request header, including the sub and pin fields. These fields are used to retrieve SCAPISettings and connect to the Scollect database. The request body must include a portfolio field in JSON format. The system then queries the Scollect database for accounts in the specified portfolio and returns them as a JSON response. Unless specified otherwise, assume every response field is a string.

Request Body

```
{
  "portfolio": "C2CLA 12.1.22"  (text)
}
```

Response

```
{
  "result": [
    {
      "CBR": "Mon, 01 Jan 1900 00:00:00 GMT",
      "CO AMT": "0.00",
      "CO Date": "5/20/2013",
      "Current UPB": "809.00",
      "DOB": "10/18/1965",
      "E-Mail": "TEST@EMAIL.COM",
      "Last Paid Date": " ",
      "Lot_Group_master": null,
      "MM Address": "15 MAIN ST",
      "MM City": "GARDEN",
"MM State": "AL",
"MM Zip": "12345",
"OrigDate": "3/6/2013",
"Originator": "C2C Loan Advance",
"SOLdate": "Sun, 31 Dec 1899 00:00:00 GMT",
"SSN": "111-22-3333",
"Total Paid since CO": 0.0,
"co_first": " ",
"co_last": " ",
"co_ssn": " ",
"codebtor1primaryphone": " ",
"creditor": "C2C",
"custom1": "UNITED SOUTHERN BANK",
"custom10": " ",
"custom11": " ",
"custom12": " ",
"custom13": " ",
"custom14": " ",
"custom15": "",
"custom16": " ",
"custom17": " ",
"custom18": " ",
"custom19": " ",
"custom2": "063105285",
"custom20": " ",
"custom21": " ",
"custom22": " ",
"custom23": " ",
"custom24": " ",
"custom25": " ",
"custom26": " ",
"custom27": " ",
"custom28": " ",
"custom29": " ",
"custom3": "124",
"custom30": " ",
"custom31": " ",
"custom32": " ",
"custom33": " ",
"custom34": " ",
"custom35": " ",
"custom36": " ",
"custom37": " ",
"custom38": " ",
"custom39": " ",
"custom4": "1234",
"custom40": " ",
"custom41": " ",
"custom42": " ",
"custom43": " ",
"custom44": " ",
"custom45": " ",
"custom46": " ",
"custom47": " ",
"custom48": " ",
"custom49": " ",
"custom5": "Bank Info",
"custom50": " ",
"custom51": " ",
"custom52": " ",
"custom53": " ",
"custom54": " ",
"custom55": " ",
"custom56": " ",
"custom57": " ",
"custom58": " ",
"custom59": " ",
"custom6": "email@email.com",
"custom60": " ",
"custom7": " ",
"custom8": " ",
"custom9": " ",
"description": " ",
"firstname": "CLAUDIA",
"lastactive": "11/14/2023",
"lastname": "LAST NAME",
"level": " ",
"lot_group": "C2CLA 12.1.22",
"lotno": "R1050",
"mm_notes": "",
"origacctno": "12345679",
"phone1": "111-222-3333",
"phone10": "",
"phone11": "",
"phone12": "",
"phone13": "",
"phone14": "",
"phone15": "",
      "phone2": "222-333-4444",
      "phone3": "451-589-8544",
      "phone4": "111-111-1111",
      "phone5": "222-222-2222",
      "phone6": "333-333-3333",
      "phone7": "444-444-4444",
      "phone8": "555-555-5555",
      "phone9": "",
      "phonestatus1": "Good Number",
      "phonestatus10": "",
      "phonestatus11": "",
      "phonestatus12": "",
      "phonestatus13": "",
      "phonestatus14": "",
      "phonestatus15": "",
      "phonestatus2": "Disconnected",
      "phonestatus3": "Disconnected",
      "phonestatus4": "UNKNOWN",
      "phonestatus5": "UNKNOWN",
      "phonestatus6": "UNKNOWN",
      "phonestatus7": "UNKNOWN",
      "phonestatus8": "UNKNOWN",
      "phonestatus9": "",
      "producttype": " "
    }
  ],
  "state": "SUCCESS"
}
```

Error

```
{
  "result": {
    "error": "Missing portfolio"
  },
  "state": "SUCCESS"
}
```

GET: getattempts/<filenumber>

This process defines a route /getattempts/<filenumber> that accepts GET requests. It requires a valid JWT token in the request header, which contains the sub and pin fields. The process uses these fields to get the SCAPISettings from the database and connect to the scollect database. The process also expects a filenumber as a path parameter. The process queries the scollect database for the attempts log of the account with the given filenumber and returns them as a JSON response. Unless specified otherwise, assume every response field is a string.

Request Body

api/getattempts/R1001

Response

```
{
  "result": {
    "result": [
      {
        "agent": "agent",
        "date": "Wed, 04 Oct 2023 00:00:00 GMT",
        "direction": "Outbound",
        "dispo": "LEFT MESSAGE MACHINE",
        "filenumber": "R1001",
        "maindatabaseid": 2,
        "phonenumber": "555-589-5050",
        "time": "09:45",
        "type": "MANUAL",
        "wassecond": 0 (number)
      },
      {
        "agent": "agent",
        "date": "Wed, 04 Oct 2023 00:00:00 GMT",
        "direction": "Outbound",
        "dispo": "SPOKE TO DEBTOR",
        "filenumber": "R1001",
        "maindatabaseid": 2,
        "phonenumber": "555-589-5050",
        "time": "10:23",
        "type": "MANUAL",
        "wassecond": 0 (number)
      },
      {
        "agent": "agent",
        "date": "Tue, 23 Jan 2024 00:00:00 GMT",
        "direction": "Outbound",
        "dispo": "LEFT MESSAGE MACHINE",
        "filenumber": "R1001",
        "maindatabaseid": 2,
        "phonenumber": "222-222-2222",
        "time": "07:24",
        "type": "MANUAL",
        "wassecond":  0 (number)
      }
    ]
  },
  "state": "SUCCESS"
}
```

Error

```
{
  "result": {
    "result": []
  },
  "state": "SUCCESS"
}
```

GET: /getemails/<filenumber>

The getemails API is a way to retrieve the emails sent to a customer in a database. You need to provide the file number of the customer as a parameter in the URL. You also need a valid token to access the API. Unless specified otherwise, assume every response field is a string.

Request Body

api/getemails/R1166

Response

```
{
  "result": [
      {
        "DateLoaded": "Thu, 07 Mar 2024 15:39:00 GMT",
        "EmailID": "1bd709a7-8cac-4c1e-a4f7-e82b21d9c14e",
        "IPAddress": "34.227.69.252",
        "ReferenceNumber": "R1166",
        "Subject": "Keep Your Account in Good Standing",
        "ToAddress": "egallagher@concepts2code.com",
        "TrackingType": "OkToSend",
        "UserAgent": "",
        "id": 2098037
      },
      {
        "DateLoaded": "Thu, 07 Mar 2024 15:39:02 GMT",
        "EmailID": "1bd709a7-8cac-4c1e-a4f7-e82b21d9c14e",
        "IPAddress": "72.21.217.15",
        "ReferenceNumber": "R1166",
        "Subject": "Keep Your Account in Good Standing",
        "ToAddress": "egallagher@concepts2code.com",
        "TrackingType": "Delivery",
        "UserAgent": "",
        "id": 2098038
      },
      {
        "DateLoaded": "Thu, 07 Mar 2024 15:39:48 GMT",
        "EmailID": "1bd709a7-8cac-4c1e-a4f7-e82b21d9c14e",
        "IPAddress": "72.65.17.16",
        "ReferenceNumber": "R1166",
        "Subject": "Keep Your Account in Good Standing",
        "ToAddress": "egallagher@concepts2code.com",
        "TrackingType": "Open",
        "UserAgent": "Mozilla/4.0 (compatible; ms-office; MSOffice 16)",
        "id": 2098039
      }
    ],
    "state": "SUCCESS"
}
```

Error

```
{
  "result": {
    "result": []
  },
  "state": "SUCCESS"
}
```

GET: /getemailstats/<filenumber>

The getemailstats API is a way to retrieve the email statistics of a customer in a database. You need to provide the file number of the customer as a parameter in the URL. You also need a valid token to access the API. Unless specified otherwise, assume every response field is a string.

Request Body

api/getemailstats/D3406

Response

```
{
  "result": [
    {
      "filenumber": "D3406",
      "last_email_date": "Mon, 01 Jan 1900 00:00:00 GMT",
      "lastclickdate": null,
      "lastopenedemail": null,
      "lastportallogin": "Mon, 01 Jan 1900 00:00:00 GMT",
      "linkclickcount": 0,
      "openedcount": 0,
      "registeredportal": 0,
      "sentcount": 0
    }
  ],
  "state": "SUCCESS"
}
```

Error

```
{
  "result": {
    "result": []
  },
  "state": "SUCCESS"
}
```

GET: /getphones/<filenumber>

The getphones API is a way to retrieve the phone numbers of a customer in a database. You need to provide the file number of the customer as a parameter in the URL. You also need a valid token to access the API. Unless specified otherwise, assume every response field is a string.

Request Body

api/getphones/R1000

Response

```
{
  "result": [
    {
      "assignedto": null,
      "balance": null,
      "filenumber": "R1000",
      "firstname": "IVANA",
      "id": 15171286,
      "isagoodnumber": null,
      "lastchange": "Mon, 27 Nov 2023 12:15:27 GMT",
      "lastname": "LAST NAME",
      "maindatabaseid": 1,
      "phonenumber": "123-456-7890",
      "phonestatus": "UNKNOWN",
      "phonetype": "UNKNOWN",
      "source": "",
      "typeofdebt": null
    },
    {
      "assignedto": "agent",
      "balance": "$0.00",
      "filenumber": "R1000",
      "firstname": "",
      "id": 15171583,
      "isagoodnumber": "UNKNOWN",
      "lastchange": "0000-00-00 00:00:00",
      "lastname": "",
      "maindatabaseid": 1,
      "phonenumber": "222-333-4444",
      "phonestatus": "UNKNOWN",
      "phonetype": "UNKNOWN",
      "source": "USER",
      "typeofdebt": "debt"
    }
  ],
  "state": "SUCCESS"
}
```

Error

```
{
  "result": {
    "error": "File number D3406 does not have associated phone numbers."
  },
  "state": "SUCCESS"
}
```

GET: /getnotes/<filenumber>

The GET /getnotes/{filenumber} endpoint retrieves all notes associated with a specific customer based on the provided file number. A valid authentication token is required to access the data. If the account exists, the API returns a list of notes, including timestamps, messages, and collector details. If no notes are found or the account does not exist, an error message is returned.

Request Body

api/getnotes/R1000

Response

```
{
   "result": [
      {
         "collectorname": "System",
         "filenumber": "R1000",
         "id": 34610632,
         "log.filenumber": "1",
         "logdate": "12/1/2022",
         "logdateandtime": "12/1/2022 AT 1:16 PM",
         "logmessage": "|ACCOUNT IMPORTED INTO DATABASE|",
         "logtime": "1:16 PM"
      },
      {
         "collectorname": "agent",
         "filenumber": "R1000",
         "id": 34610938,
         "log.filenumber": "1",
         "logdate": "11/27/2023",
         "logdateandtime": "11/27/2023 AT 12:14 PM",
         "logmessage": "|Account Change |\r\nFirst Name from IYANNA to IVANA\r\nLast Name from NAVARRO to LAST NAME\r\nEmail Name from TEST@gmail.com1 to EMAIL@EMAIL.COM\r\nAddress from PO BOX 1858 to PO BOX 0123\r\nAddress2 from TORRANCE COUNTY) to \r\nCity from MORIARTY to CITY\r\nState from NM to ST\r\nZip from 87035 to 12345",
         "logtime": "12:14 PM"
      },
      {
         "collectorname": "agent",
         "filenumber": "R1000",
         "id": 34610939,
         "log.filenumber": "1",
      "logdate": "11/27/2023",
      "logdateandtime": "11/27/2023 AT 12:15 PM",
      "logmessage": "|DELETED| PHONE NUMBER  505-855-5505",
      "logtime": "12:15 PM"
   },
   {
      "collectorname": "agent",
      "filenumber": "R1000",
      "id": 34610940,
      "log.filenumber": "1",
      "logdate": "11/27/2023",
      "logdateandtime": "11/27/2023 AT 12:15 PM",
      "logmessage": "|DELETED| PHONE NUMBER  505-832-5477",
      "logtime": "12:15 PM"
   },
   {
      "collectorname": "agent",
      "filenumber": "R1000",
      "id": 34610941,
      "log.filenumber": "1",
      "logdate": "11/27/2023",
      "logdateandtime": "11/27/2023 AT 12:15 PM",
      "logmessage": "Changed phonenumbe id: 15171286 from 505-855-5555 to
123-456-7890 Type: UNKNOWN Status: UNKNOWN",
      "logtime": "12:15 PM"
   },
   {
      "collectorname": "agent",
      "filenumber": "R1000",
      "id": 34610942,
      "log.filenumber": "1",
      "logdate": "11/27/2023",
      "logdateandtime": "11/27/2023 AT 12:16 PM",
      "logmessage": "|Added Alert| BANKRUPT ACCOUNT",
      "logtime": "12:16 PM"
   },
   {
      "collectorname": "agent",
      "filenumber": "R1000",
      "id": 34610943,
      "log.filenumber": "1",
      "logdate": "11/27/2023",
      "logdateandtime": "11/27/2023 AT 12:43 PM",
      "logmessage": "Changed Custom Field: Label4 from   to 12345",
    "logtime": "12:43 PM"
},
{
    "collectorname": "agent",
    "filenumber": "R1000",
    "id": 34611022,
    "log.filenumber": "1",
    "logdate": "1/12/2024",
    "logdateandtime": "1/12/2024 AT 9:15 AM",
    "logmessage": "|Added Alert| ",
    "logtime": "9:15 AM"
},
{
    "collectorname": "agent",
    "filenumber": "R1000",
    "id": 34611023,
    "log.filenumber": "1",
    "logdate": "1/12/2024",
    "logdateandtime": "1/12/2024 AT 9:17 AM",
    "logmessage": "|Added Alert| Added new alert.",
    "logtime": "9:17 AM"
},
{
    "collectorname": "agent",
    "filenumber": "R1000",
    "id": 34611024,
    "log.filenumber": "1",
    "logdate": "1/12/2024",
    "logdateandtime": "1/12/2024 AT 9:19 AM",
    "logmessage": "|Added Alert| ",
    "logtime": "9:19 AM"
},
{
    "collectorname": "agent",
    "filenumber": "R1000",
    "id": 34611025,
    "log.filenumber": "1",
    "logdate": "1/12/2024",
    "logdateandtime": "1/12/2024 AT 9:20 AM",
    "logmessage": "|Added Alert| Added new alert.",
    "logtime": "9:20 AM"
},
{
    "collectorname": "agent",
      "filenumber": "R1000",
      "id": 34611026,
      "log.filenumber": "1",
      "logdate": "1/12/2024",
      "logdateandtime": "1/12/2024 AT 9:24 AM",
      "logmessage": "Added Phonenumber : 111-222-3333 Type: UNKNOWN Status:
UNKNOWN",
      "logtime": "9:24 AM"
    },
    {
      "collectorname": "agent",
      "filenumber": "R1000",
      "id": 34611027,
      "log.filenumber": "1",
      "logdate": "1/12/2024",
      "logdateandtime": "1/12/2024 AT 9:26 AM",
      "logmessage": "Added Phonenumber : 222-333-4444 Type: UNKNOWN Status:
UNKNOWN",
      "logtime": "9:26 AM"
    }
  ],
  "state": "SUCCESS"
}
```

Error

```
{
  "result": {
    "error": "File number R100 does not have any notes or may not exist."
  },
  "state": "SUCCESS"
}
```

GET: /getpayments/<filenumber>

The GET /getpayments/{filenumber} endpoint retrieves the payment history of a specific customer based on the provided file number. A valid authentication token is required to access the data. If the account exists, the API returns a list of recorded payments, including payment dates, amounts, and types. If no matching record is found, an error message is returned.

Request Body

api/getpayments/R10001

Response

```
{
  "result": [
    {
      "datetaken": "2/20/2024",
      "filenumber": "R1001",
      "merchant": null,
      "paymentamount": "15.00",
      "paymentdate": "Tue, 20 Feb 2024 00:00:00 GMT",
      "paymentmethod": null,
      "portfolio": "C2C PDL TEST",
      "typeofpayment": "IN HOUSE"
    },
    {
      "datetaken": "2/20/2024",
      "filenumber": "R1001",
      "merchant": null,
      "paymentamount": "150.00",
      "paymentdate": "Tue, 27 Feb 2024 00:00:00 GMT",
      "paymentmethod": null,
      "portfolio": "C2C PDL TEST",
      "typeofpayment": "IN HOUSE"
    },
    {
      "datetaken": "2/20/2024",
      "filenumber": "R1001",
      "merchant": null,
      "paymentamount": "150.00",
      "paymentdate": "Tue, 05 Mar 2024 00:00:00 GMT",
      "paymentmethod": null,
      "portfolio": "C2C PDL TEST",
      "typeofpayment": "IN HOUSE"
    }
  ],
  "state": "SUCCESS"
}
```

Error

```
{
   "result": [
      {
         "error": "File number R100 does not exist."
      },
      404
   ],
   "state": "SUCCESS"
}
```

GET: /getaccount/<filenumber>

The GET /getaccount/{filenumber} endpoint retrieves detailed account information for a specific customer based on the provided file number. A valid authentication token is required to access the data. If the account exists, the API returns comprehensive account details, including personal, financial, and employment information. If no matching record is found, an empty result is returned.

Request Body

api/getaccount/R1001

Response

```
{
   "result": [
      {
         "accountnumber": "12345679",
         "accruedinterest": "0.00",
         "activated": null,
         "address": "15 STREET NAME",
         "address2": "",
         "adjustments": "0.00",
         "advancedfees": "0.00",
         "asset1value": "5500",
"assettype1": "Vehicles",
"assettype2": "",
"assettype2value": "",
"assettype3": "",
"assettype3value": "",
"assettype4": " ",
"assettype4value": " ",
"assettype5": " ",
"assettype5value": " ",
"assignedto": "mall",
"attorneyaddress": " ",
"attorneyaddress2": "",
"attorneybeepernumber": " ",
"attorneycellphone": " ",
"attorneycity": " ",
"attorneyemailaddress": " ",
"attorneyfax": " ",
"attorneyfees": "0.00",
"attorneyletters": null,
"attorneyname": "",
"attorneyphone": " ",
"attorneystate": "",
"attorneyzip": " ",
"authorizeduser": null,
"bankruptcytype": "",
"bankruptcyverificationstatus": "",
"beepernumber": null,
"birthdate": "1/1/1900",
"calculateinterest": null,
"callbacks": null,
"cellphone": null,
"chargedoffprincipal": "0.00",
"city": "CITY",
"claimstatus": null,
"claimtype": null,
"clientname": "C2C",
"clientnotes": null,
"closedreason": "",
"codebtor1address": " ",
"codebtor1address2": " ",
"codebtor1city": " ",
"codebtor1contactinfo": " ",
"codebtor1employeraddress": " ",
"codebtor1employercity": " ",
"codebtor1employername": " ",
"codebtor1employerstate": " ",
"codebtor1employerzip": " ",
"codebtor1firstname": " ",
"codebtor1friends": " ",
"codebtor1fullname": " ",
"codebtor1lastname": " ",
"codebtor1middlename": " ",
"codebtor1occupation": " ",
"codebtor1primaryphone": " ",
"codebtor1secondaryphone": " ",
"codebtor1socialsecurity": " ",
"codebtor1state": " ",
"codebtor1zip": " ",
"codebtor2address": " ",
"codebtor2address2": " ",
"codebtor2city": " ",
"codebtor2contactinfo": " ",
"codebtor2employeraddress": " ",
"codebtor2employercity": " ",
"codebtor2employername": " ",
"codebtor2employerstate": " ",
"codebtor2employerzip": " ",
"codebtor2firstname": " ",
"codebtor2friends": " ",
"codebtor2fullname": " ",
"codebtor2lastname": " ",
"codebtor2middlename": " ",
"codebtor2occupation": " ",
"codebtor2primaryphone": " ",
"codebtor2secondaryphone": " ",
"codebtor2socialsecurity": " ",
"codebtor2state": " ",
"codebtor2zip": " ",
"collectionfees": "0.00",
"collectorname": "Mark",
"contingencyamount": "7050",
"courtaddress": "",
"courtcity": "",
"courtfees": "0.00",
"courtfiled": "10/6/2023",
"courtphone": "",
"courtstate": "",
"courtzip": "",
"creditreport1": null,
"creditreport2": null,
"creditreport3": null,
"creditreport4": null,
"creditreport5": null,
"creditscore": " ",
"currentbalance": "809.00",
"currentlyemployed": null,
"custom1": "UNITED SOUTHERN BANK",
"custom10": " ",
"custom11": " ",
"custom12": " ",
"custom13": " ",
"custom14": " ",
"custom15": " ",
"custom16": " ",
"custom17": " ",
"custom18": " ",
"custom19": " ",
"custom2": "063105285",
"custom20": " ",
"custom21": " ",
"custom22": " ",
"custom23": " ",
"custom24": " ",
"custom25": " ",
"custom26": " ",
"custom27": " ",
"custom28": " ",
"custom29": " ",
"custom3": "124",
"custom30": " ",
"custom31": " ",
"custom32": " ",
"custom33": " ",
"custom34": " ",
"custom35": " ",
"custom36": " ",
"custom37": " ",
"custom38": " ",
"custom39": " ",
"custom4": "45",
"custom40": " ",
"custom41": " ",
"custom42": " ",
"custom43": " ",
"custom44": " ",
"custom45": " ",
"custom46": " ",
"custom47": " ",
"custom48": " ",
"custom49": " ",
"custom5": " ",
"custom50": " ",
"custom51": " ",
"custom52": " ",
"custom53": " ",
"custom54": " ",
"custom55": " ",
"custom56": " ",
"custom57": " ",
"custom58": " ",
"custom59": " ",
"custom6": " ",
"custom60": " ",
"custom7": " ",
"custom8": " ",
"custom9": " ",
"customertype": null,
"dateaccountopened": "3/6/2013",
"datechargedoff": "5/20/2013",
"dateclosed": "1900-01-01",
"datefiled": "10/6/2023",
"datelastassigned": "Thu, 01 Dec 2022 13:14:01 GMT",
"datelastdialed": "Thu, 01 Dec 2022 00:00:00 GMT",
"datelaststatuschg": "Thu, 01 Dec 2022 13:14:01 GMT",
"dateplaced": "12/1/2022",
"datereceived": " ",
"daysassignedto": null,
"daysinstatus": null,
"dayslastworked": 0,
"dayssincelastpayment": null,
"deceased": "NO",
"deceaseddate": null,
"delinquencydate": " ",
"dialerstatus": null,
"docketnumber": "dgmhdm",
"emailaddress": "EMAIL@EMAIL.COM",
"employer": "SONSHINE SCHOOL",
"employeraddress": " 123 MAIN STt",
"employercity": " CITY",
"employerphone": "1112223333",
"employerstate": "ST",
"employerzip": " 12345",
"endtime": null,
"filenumber": "R1001",
"firstname": "MALIA",
"fullname": "MALIA GUEST",
"garnishment": "NO",
"garnishmentamount": " ",
"garnishmentdate": " ",
"garnishmentsource": " ",
"garnishmenttype": " ",
"goodnumbers": null,
"groupname": "",
"hasassets": "NO",
"hasduplicates": null,
"hasimages": null,
"hotlist": null,
"hotlistcomments": null,
"id": 2,
"interestpercentage": "0.00",
"jointaddress": " ",
"jointaddress2": " ",
"jointbirthdate": " ",
"jointcity": " ",
"jointemployer": " ",
"jointemployeraddress": " ",
"jointemployercity": " ",
"jointemployerstate": " ",
"jointemployerzip": " ",
"jointfirstname": " ",
"jointfullname": " ",
"jointlastname": " ",
"jointmiddlename": " ",
"jointprimaryphone": " ",
"jointsecondaryphone": " ",
"jointsocial": " ",
"jointstate": " ",
"jointzip": " ",
"judgement": "NO",
"judgementdate": " 01/25/23",
"judgementfees": "0.00",
"judgementpercentage": " ",
"lastname": "GUEST",
"lastpaymentamount": "0.00",
"lastpaymentdate": " ",
"lastworked": "1/23/2024",
"latefees": "0.00",
"lein": "NO",
"leinamount": "",
"leindate": " ",
"leintype": " ",
"makepayabletoaddress": null,
"makepayabletocity": null,
"makepayabletoname": null,
"makepayabletostate": null,
"makepayabletozip": null,
"managername": null,
"middlename": " ",
"miscfees": "0.00",
"nsffees": "0.00",
"occupation": " TEACHER",
"orderbydate": "Thu, 01 Dec 2022 00:00:00 GMT",
"originalbalance": "809.00",
"originalcreditor": "C2C Loan Advance",
"originalprincipal": "809.00",
"paidtodate": " ",
"pastdue": null,
"paymentplan": null,
"penaltyfees": "0.00",
"portfolio": "C2C PDL TEST",
"portfoliogroup": null,
"primaryphone": null,
"principal": "809.00",
"promises": null,
"reimbursedfees": "0.00",
"reportfilenumber": "000-00-0000",
"salesman": null,
"salesmanpercentage": null,
"skips": null,
"socialsecuritynumber": "000-00-0000",
"spouseemployer": "",
"spouseemployeraddress": " ",
"spouseemployercity": " ",
"spouseemployerphone": " ",
         "spouseemployerstate": " ",
         "spouseemployerzip": " ",
         "spousename": "",
         "spouseoccupation": "",
         "spouseworkphone": "",
         "starttime": null,
         "state": "AL",
         "statuscolor": null,
         "statusname": "Bankruptcy",
         "statustype": null,
         "statuteoflimdate": "Sun, 20 May 2018 00:00:00 GMT",
         "title": " ",
         "typeofdebt": " ",
         "verificationmethod": "",
         "verified": "NO",
         "verifieddate": "10/6/2023",
         "workphone": null,
         "zip": "ZIP"
      }
   ],
   "state": "SUCCESS"
}
```

Error

```
{
   "result": {
      "result": []
   },
   "state": "SUCCESS"
}
```

POST /insertphone

The POST /insertphone endpoint allows authorized users to add a new phone number to a customer's account. By providing the required details, including the file number, phone number, type, status, and source, the system records the phone information in the database. If the operation is successful, a confirmation message is returned; otherwise, an error is provided if the account is not found.

Request Body

```
{
   "filenumber": "R1000",  (text)
   "phonenumber": "123-456-7890",  (text)
  "phonetype": "UNKNOWN",  (text)
  "phonestatus": "UNKNOWN",  (text)
  "source": "LOAD"  (text)
}
```

Response

```
{
   "result": {
      "success": "Phone number inserted successfully"
   },
   "state": "SUCCESS"
}
```

Error

```
{
   "result": {
      "error": "Data not found for this filenumber"
   },
   "state": "SUCCESS"
}
```

PUT /updatephone

The updatephone API lets you update the type and status of a phone number tied to a specific file. To use it, you must include a valid token, a filenumber, and a properly formatted phone number (###-###-####), along with at least one of phonetype or phonestatus—neither may be omitted.

Request Body

```
{
   "filenumber": "R1001",
   "phonenumber": "407-253-7828",
   "phonetype":"UNKNOWN",
   "phonestatus": "UNKNOWN"
}
```

Response

```
{
   "result": {
      "success": "Phone number updated successfully"
   },
   "state": "SUCCESS"
}
```

Error

```
{
   "result": [
      {
         "error": "You need to provide at least one field to update"
      },
      400
   ],
   "state": "SUCCESS"
}
```

POST /insertattempt

The POST /insertattempt endpoint records an attempt to contact a customer in the system. Users must provide details such as the phone number, outcome of the attempt, agent handling the contact, type of attempt, and file number. A valid authentication token is required. If the attempt is successfully logged, a confirmation message is returned; otherwise, an error is provided if the file number is not found.

Request Body

```
{
   "phonenumber": "123-456-7890", (text)
   "dispo": "SPOKE TO DEBTOR", (text)
   "agent": "agent", (text)
   "type": "MANUAL", (text)
   "filenumber": "R1000" (text)
}
```

Response

```
{
   "result": {
      "success": "Attempt inserted successfully"
   },
   "state": "SUCCESS"
}
```

Error

```
{
   "result": [
      {
         "error": "Data not found for this filenumber"
      },
      400
   ],
   "state": "SUCCESS"
}
```

POST /InsertNoteline

The POST /InsertNoteline endpoint allows authorized users to add a log entry to an account. By providing the necessary details, including the account number, collector's name, and a message, a new record is created in the system with the current date and time. If successful, a confirmation message is returned; otherwise, an error is provided if the account is not found.

Request Body

```
{
   "filenumber": "R1000",
   "collectorname": "mall",
   "logmessage":"Test"
}
```

Response

```
{
   "result": {
      "success": "Noteline inserted successfully"
   },
   "state": "SUCCESS"
}
```

Error

```
{
   "result": [
      {
         "error": "Data not found for this filenumber"
      },
      400
   ],
   "state": "SUCCESS"
}
```

PUT /updatepermissions

The PUT /updatepermissions endpoint allows authorized users to update specific account permissions. By providing a valid authentication token and the required details, users can modify permission settings for a given account. If the update is successful, a confirmation message is returned; otherwise, an error is provided if the account is not found.

Request Body

```
{
  "filenumber":"R1000",
  "fieldname":"callspouse",
  "fieldvalue":"1"
}
```

Response

```
{
   "result": {
      "success": "Account permissions table updated successfully"
   },
   "state": "SUCCESS"
}
```

Error

```
{
   "result": {
      "error": "Data not found for this filenumber"
   },
   "state": "SUCCESS"
}
```

PUT /updatedbase

This process defines a route /updatedbase that accepts PUT requests. It requires a valid JWT token in the request header, which contains the sub and pin fields. The process uses these fields to get the SCAPISettings from the database and connect to the scollect database. The process also expects a JSON payload in the request body, which must contain the filenumber, fieldname, and fieldvalue fields. The process updates the dbase table with the given field value for the account with the given filenumber and returns a success message.

Request Body

```
{
   "filenumber": "R1000",
   "fieldname":"firstname",
   "fieldvalue":"IVANA"
}
```

Response

```
{
   "result": {
      "success": "Dbase table updated successfully"
   },
   "state": "SUCCESS"
}
```

Error

```
{
   "result": [
      {
         "error": "The field s does not exist in the dbase table."
      },
      404
   ],
   "state": "SUCCESS"
}
```

POST insert_payplan_external

This endpoint works the same exact way as insert payments external but instead of inserting one singular payment date and payment amount, you pass it in as a list.

Request Body

```
{
   "filenumber": "r456456" ,
   "paymentdate": "2023-04-15",
   "payorname": "John Doe",
   "paymentmethod": "Credit Card",
   "paymentstatus": "PENDING",
   "typeofpayment": "Online",
   "checkaccountnumber": "9876543210",
   "checkroutingnumber": "012345678",
   "cardtype": "Visa",
   "cardnumber": "4111111111111111",
   "threedigitnumber": "111",
   "cardexpirationmonth": "12",
   "cardexpirationyear": "2025",
   "cardexpirationdate": "12/2025",
   "paymentamount": "100.00",
   "checkaccounttype": "Checking",
   "acceptedfees": "0",
   "printed": "false",
   "invoice": "INV123456",
   "paymentdata": [
       {
           "paymentamount": "1.00",
           "paymentdate": "2025-09-01"
       },
       {
           "paymentamount": "1.00",
           "paymentdate": "2025-10-01"
       }
   ]
}
```

POST insert_payments_external

The POST /insert_payments_external endpoint allows authorized users to securely record payment transactions in the system. Users must provide payment details, including file number, payment date, payor name, payment method, and other relevant information. The request must be authenticated with a valid JWT token. The endpoint validates the submitted data, ensuring all required fields are present. If successful, the payment is stored in the database, and a confirmation message is returned. If any required information is missing, an error response is provided.

Request Body

```
{
   "filenumber": "12345" ,                  (datatype: String, required: True)
   "paymentdate": "2023-04-15",             (datatype: String, required: True)
   "payorname": "John Doe",                 (datatype: String, required: True)
   "paymentmethod": "CREDIT CARD",                  (datatype: String, required: True)
   "paymentstatus": "PENDING",              (datatype: String, required: True)
   "typeofpayment": "Online", (can be anything)                     (datatype: String, required:
True)
   "checkaccountnumber": "9876543210",      (datatype: String, required: True)
   "checkroutingnumber": "012345678",       (datatype: String, required: True)
   "cardtype": "Visa",                      (datatype: String, required: True)
   "cardnumber": "4111111111111111",        (datatype: String, required: True)
   "threedigitnumber": "111",               (datatype: String, required: True)
   "cardexpirationmonth": "12",             (datatype: String, required: True)
   "cardexpirationyear": "2025",            (datatype: String, required: True)
   "cardexpirationdate": "12/2025",         (datatype: String, required: True)
   "paymentamount": "100.00",               (datatype: String, required: True)
   "checkaccounttype": "Checking",          (datatype: String, required: True)
   "acceptedfees": "0 or 1",                (datatype: String, required: True)
   "printed": "false",                      (datatype: String, required: True)
   "invoice": "INV123456"                   (datatype: String, required: True)
}
```

Response

```
{
   "state": "SUCCESS",
   "message": "Payment successfully inserted"
}
```

Error

```
{
   "result": {
      "error": “”
   },
   "state": "SUCCESS"
}
```

POST /createCallback

This API endpoint allows the user to create a callback record. A valid Bearer Authentication token is required.Use this API endpoint to create a callback record. You'll need a valid Bearer Authentication token.

Request Body

```
{
  "FileNumber": "R1000",
  "Name": "IVANA LAST NAME",
  "Assignedto":"mall",
  "Callbackdate": "2025-03-17",
  "Callbacktime": "1:43PM",
  "Callbackdatemysql": "2025-03-17"
}
```

Response

```
{
  "result": {
    "Success": "Callback has been uploaded"
  },
  "state": "SUCCESS"
}
```

Error

```
{
  "result": {
    "result": []
  },
  "state": "SUCCESS"
}
```

POST /run_payment_external

This API endpoint allows external systems to run or process a payment for a specific filenumber

Request Body

```
{
   "filenumber": "12345",              (datatype: String, required: True)
   "paymentdate": "2023-04-15",    (datatype: String, required: True)
   "paymentamount": "9.00"                    (datatype: String, required: True)
}
```

Response

```
{
   "result": "Payment Result>",
   "approvalcode": "<Payment Approval Code>",
   "error": "<Payment Approval Code>",

}
```

Error

```
{
   "result": {
      "result": []
   },
   "state": "SUCCESS"
}
```

POST /getAIaccountdata

The POST /getAIaccountdata endpoint retrieves key account details for a specified customer using their file number. A valid authentication token is required. If the file number exists, the API returns information such as the customer's balance, contact details, birthdate, and payment-related data. If no matching record is found, an error message is returned.

Request Body

```
{
   "FileNumber": "R1000"
}
```

Response

```
{
   "result": {
      "success": [
         {
            "birthdate": "1/1/1900",
            "currentbalance": "589.00",
            "filenumber": "R1000",
            "fullname": "IVANA LAST NAME",
            "id": 1,
            "minimumonetimepayment": "0.00",
            "minimumpayment": "0.00",
            "phone1": "123-456-7890",
            "phone2": "222-333-4444",
            "sifamount": "0.00",
            "sifnumberofpay": 0,
            "socialsecuritynumber": "0000"
         }
      ]
   },
   "state": "SUCCESS"
}
```

Error

```
{
   "result": [
      {
         "error": "This file number does not exist in the system"
      },
      404
   ],
   "state": "SUCCESS"
}
```

POST /getAIpaymentdata

The POST /getAIpaymentdata endpoint retrieves payment-related information for a specific customer based on the provided file number. Users may optionally filter results by payment method and the last four digits of a card number. A valid authentication token is required. If the file number exists, the API returns details such as the customer's balance, minimum payment, contact information, and settlement options. If the file number is not found, an error message is returned.

Request Body

```
{
   "FileNumber": "r2121118",
   "paymentMethod": "CREDIT CARD", (optional)
   "last4Card":"1111" (optional)
}
```

Response

```
{
   "result": {
      "success": [
         {
            "cardzipcode": "",
            "last4card": "1111",
            "maindatabaseid": 176,
            "paymentmethod": "CREDIT CARD"
         }
      ]
   },
   "state": "SUCCESS"
}
```

Error

```
{
   "result": [
      {
         "error": "The payment data based on your search criteria does not exist"
      },
      404
```

```
  ],
  "state": "SUCCESS"
}
```

POST /get_lastpayinfo

This post update gets the last posted payment for a given account.

Request Body

```
{
  "filenumber": "R2150984"   // (datatype: String, required: True)
}
```

Successful Response

```
{
  "result": {
    "data": [
      {
        "cardaddress": "",
        "cardexpirationdate": "05/2030",
        "cardnumber": "4jde-dt32-yclc-3d28",
        "cardzipcode": "",
        "last4card": "1165",
        "paymentdate": "Mon, 26 May 2025 00:00:00 GMT",
        "payorname": "Blake McCammon"
      }
    ]
  },
  "state": "SUCCESS"
}
```

POST /update_payment_external

The POST /update_payment_external endpoint allows authorized users to update existing payment records in the system. The payment record is identified by its filenumber, and only records with paymentstatus = 'PENDING' are eligible for updates.

Users must provide the filenumber and at least one other field to update. The request must be authenticated with a valid JWT token. The system validates the input, performs the update if valid, and returns a confirmation message.

Request Body

```
{
  "filenumber": "R1002",               // (datatype: String, required: True)
  "paymentdate": "2025-04-15",          // (datatype: String, required: False)
  "payorname": "Jane Smith",            // (datatype: String, required: False)
  "paymentmethod": "ACH",               // (datatype: String, required: False)
  "paymentstatus": "PENDING",           // (datatype: String, required: False)
  "typeofpayment": "Online",            // (datatype: String, required: False)
  "checkaddress": "456 Oak Street",     // (datatype: String, required: False)
  "checkcity": "Springfield",           // (datatype: String, required: False)
  "checkstate": "IL",                   // (datatype: String, required: False)
  "checkzip": "62704",                  // (datatype: String, required: False)
  "checkaccountnumber": "123456789",    // (datatype: String, required: False)
  "checkroutingnumber": "012345678",    // (datatype: String, required: False)
  "cardtype": "MasterCard",             // (datatype: String, required: False)
  "cardnumber": "5555555555554444",     // (datatype: String, required: False)
  "threedigitnumber": "321",            // (datatype: String, required: False)
  "cardexpirationmonth": "11",          // (datatype: String, required: False)
  "cardexpirationyear": "2026",         // (datatype: String, required: False)
  "cardexpirationdate": "11/26",        // (datatype: String, required: False)
  "checkaccounttype": "Checking",       // (datatype: String, required: False)
  "acceptedfees": true,                 // (datatype: Boolean, required: False)
  "printed": "no",                      // (datatype: String, required: False)
  "invoice": "INV654321"                // (datatype: String, required: False)
}
```

Validation Rules

filenumber is required

At least one other field must be provided for update

Only updates records where paymentstatus = 'PENDING'

Fields with null, "", or [] are ignored during update

Successful Response

```
{
   "result": {
      "success": "Payments have been updated."
   },
   "state": "SUCCESS"
}
```

Error Response – Missing Fields

```
{
   "state": "TIMEOUT",
   "status": "At least one update field must be provided."
}
```

Error Response – Missing File Number

```
{
   "result": [
      {
         "error": "No payment info found with filenumber R46"
      },
      400
   ],
   "state": "SUCCESS"
}
```

POST /send_email_c2c

This endpoint receives a request to send a C2C (Concepts To Code) email. It validates the request payload and securely forwards it to a private backend API. The endpoint then monitors the resulting background task and returns its status once available. **You must have a third-party with C2C already configured in your system before using this endpoint.**

Request Body

```
{
         “sendType”: “text is required: example values is ‘email’”,
         “emailTemplate”: “text”, (is required),
         “referenceNumbers”: “array of strings”, (this is dbase.filenumber. And you just need to
provide 1),
         “updateap”: “boolean and is required.”, (if you want to update any settings in the account
         permissions table, set this to true. Otherwise, false),

}
```

Response

```
{
   "result": {
      "response": {
         "ErrorMessage": null,
         "Errors": null,
         "ResultObject": null,
         "Success": true,
         "SuccessRedirectUrl": null
      },
      "success": true
   },
   "state": "SUCCESS"
}
```

Error Response

Anything not in success

POST /send_text

This endpoint sends a templated SMS message through Solutions By Text (SBT) using a pre-approved template. It validates the request data, looks up the active subscribers for the specified file number, and forwards the message to SBT via an asynchronous background task. **You must have a Solutions By Text third-party integration configured in your system before using this endpoint.**

Request Body

```
{
   "filenumber": "316620",
   "textTemplate": "Missed 1",
   "messageType": "Multicast",
   "sbt_env": "LIVE"
}
```

Response

```
{
   "result": {
       "success": true
   },
   "state": "SUCCESS"
}
```

Error Response

Anything not in success

POST /searchbyphone

The POST /searchbyphone endpoint allows users to retrieve customer details associated with a specific phone number. A valid authentication token is required. If the phone number exists in the system, the API returns relevant customer information, including name, file number, address, and social security number (last four digits). If no matching record is found, an error message is returned.

Request Body

```
{
    "PhoneNumber":"716-628-9899"
}
```

Response

```
{
    "result": {
        "success": [
            {
                "address": "HCR 67 BOX 108",
                "birthdate": "",
                "city": "MARFA",
                "filenumber": "R1005",
                "firstname": "ANTHONY",
                "lastname": "POPE",
                "ssn": "0000",
                "state": "MT",
                "zip": "79843"
            }
        ]
    },
    "state": "SUCCESS"
}
```

Error

```
{
   "result": [
      {
         "error": "This phone number does not exist in the system"
      },
      404
   ],
   "state": "SUCCESS"
}
```

Valid Account Permission Fields

Field Name	Data Type	Description	Example
callspouse	1 or 0	Indicates if the spouse can be contacted	1
callwork	Text	Specifies if work can be called	"Allowed"
cantext	1 or 0	Indicates if text messages can be sent	0
canemail	1 or 0	Indicates if emails can be sent	1
addresscorrect	Text	Verified or corrected address	"123 Main St, NY"
dateofnotice	Date	Date of the latest notice sent	2024-03-15
cont_debtor	1 or 0	Indicates if direct contact is allowed	1
last_cont_date	Date	Last date of contact with the debtor	2024-02-10

lastcontactagent	Text	Name of the last agent who contacted the debtor	"John Smith"
call_campaign	Text	Campaign associated with the call	"Debt Recovery 2024"
firstcontact	Date	Date of the first contact	2023-12-01
firstcontactagent	Text	Name of the first agent who contacted the debtor	"Lisa Adams"
verifiedaddress	1 or 0	Indicates if the address has been verified	1
nextletter	Date	Date when the next letter is scheduled	2024-04-05
nexttext	Date	Date when the next text is scheduled	2024-04-06
nextemail	Date	Date when the next email is scheduled	2024-04-07
hotnotes	Text	Important notes about the account	"Customer requested callback"
textphone	Phone Number	Phone number used for text messages	"555-123-4567"
cancallcell	1 or 0	Indicates if the debtor's cell can be called	0
donot1099	1 or 0	Indicates if a 1099 form should not be issued	1
donottext	1 or 0	Indicates if texts should not be sent	0
donotemail	1 or 0	Indicates if emails should not be sent	0

sifamount	Number (Deci…	Settlement in full (SIF) amount	500.00
sifstartdate	Date	Start date for SIF payments	2024-01-15
sifcompletedate	Date	Completion date for SIF payments	2024-06-15
sifnumberofpay	Number	Number of payments for settlement	3
registeredportal	1 or 0	Indicates if the debtor is registered online	1
lastportallogin	Date	Date of last portal login	2024-02-28
lastopenedemail	Date	Date when the last email was opened	2024-03-02
lastclickdate	Date	Date of the last link click in an email	2024-03-03
openedcount	Number	Number of times emails were opened	5
linkclickcount	Number	Number of times links in emails were clicked	2
sentcount	Number	Number of emails sent to the debtor	10
minimumpayment	Number (Deci…	Minimum required payment	50.00
minimumonetimepayment	Number (Deci…	Minimum one-time payment required	100.00

Advanced Query Builder

The Advanced Query Builder API offers powerful, flexible SQL query generation for experienced users. Intermediate MySQL knowledge is required—misuse may lead to inefficiencies or unintended data changes. Validate queries carefully before execution, especially in production.