Logix MVC Accounting APIs Documentation

This documentation provides a comprehensive overview of the Accounting (Acc) API controllers in the Logix MVC solution. Each controller is responsible for a specific domain of the accounting system, such as accounts, cost centers, banks, cash handling, journals, reports, and financial operations. The endpoints are RESTful and require permissions for access, ensuring robust security.

Below you'll find detailed explanations for each controller, including its responsibility, endpoints, usage examples, and data flow diagrams where applicable.

□ BaseAccApiController

Purpose

All accounting API controllers inherit from BaseAccApiController . It defines the base route and marks all API controllers as RESTful.

- 1 [Route(\$"api/{ApiConfig.ApiVersion}/Acc/[controller]")]
- 2 [ApiController]
- 3 public abstract class BaseAccApiController : ControllerBase { }

□ AccAccountController

Overview

Manages operations for chart of accounts: searching, CRUD, loading for editing, hierarchical tree, Excel import/export, and reporting.

Name	Method	Route	Descriptio
Search	POST	/api/v1/Acc/AccAccount/Search	Searches accounts by filter
Add	POST	/api/v1/Acc/AccAccount/Add	Adds a new account
Edit	POST	/api/v1/Acc/AccAccount/Edit	Edits an existing account
Delete	DELETE	/api/v1/Acc/AccAccount/Delete	Deletes an account
GetByldForEdit	GET	/api/v1/Acc/AccAccount/GetByIdForEdit	Fetches account for edit

Name	Method	Route		Descriptio	
GetByld	GET	/api/v1/Acc/AccAccount/GetById		Gets accour	
GetByCode	GET	/api/v1/Acc/AccAccount/GetByCode		Gets accour	
DDLAccountParentByGroupId	GET	/api/v1/Acc/AccAccount/DDLAccountParentByGroupId		Loads paren accounts for dropdown	
ShowUploadAccountsExcel	POST	/api/v1/Acc/AccAccount/ShowUploadAccountsExcel		Validates Excel- uploaded accounts	
SaveAccountsExcel	POST	/api/v1/Acc/AccAccount/SaveAccountsExcel		Saves validated Excel- uploaded accounts	
DeleteAllAccAccounts	DELETE	/api/v1/Acc/AccAccount/DeleteAllAccAccounts		Deletes all accounts (dangerous)	
SearchReports	POST	/api/v1/Acc/AccAccount/SearchReports		Searches accounts for reports	
Usage Examples					
Search Accounts					
Search Accounts © Export to Postman					
Search for accounts matching f	uter criteria.				
POST https://api.example.com/api/v1/Acc/AccAccount/Search					
1.0					

Headers

Authorization string • header required Bearer <token> Request body JSON payload required for this request. { "AccAccountName": "Cash", "AccGroupId": 1 } Code examples curl -X POST "https://api.example.com/api/v1/Acc/AccAccount/Search" \ -H "Authorization: Bearer <token>" \ -H "Content-Type: application/json" \ -d '{ \"AccAccountName\": \"Cash\", \"AccGroupId\": 1 }' Responses Success { "data": [{ "AccAccountId": 1, "AccAccountName": "Cash" }] } Add New Account **Add Account** Export to Postman Create a new account record. **POST** https://api.example.com/api/v1/Acc/AccAccount/Add Headers Authorization string • header required Bearer <token> Request body JSON payload required for this request.

{ "AccAccountName": "New Account", "AccGroupId": 2 }

Code examples

```
curl -X POST "https://api.example.com/api/v1/Acc/AccAccount/Add" \
   -H "Authorization: Bearer <token>" \
   -H "Content-Type: application/json" \
   -d '{ \"AccAccountName\": \"New Account\", \"AccGroupId\": 2 }'
```

Responses

Created

```
{ "success": true, "data": { "AccAccountId": 123 } }
```

Delete Account

Delete Account

Export to Postman

Delete an account by its ID.

DELETE

https://api.example.com/api/v1/Acc/AccAccount/Delete

Headers

Authorization string • header required

Bearer <token>

Query parameters

Id string required

Account Id

Code examples

```
curl -X DELETE "https://api.example.com/api/v1/Acc/AccAccount/Delete?Id=Account+Id" \
   -H "Authorization: Bearer <token>"
```

Responses

200 403

Deleted { "success": true } Forbidden { "error": "AccessDenied" }

Data Flow Example

```
Diagram Error
Parse error on line 3: ...Controller→|Search(filter)|AccAccountS ------^Expecting 'SQE', 'DOUBLECIRCLEEND', 'PE', '-)', 'STADIUMEND', 'SUBROUTINEEND', 'PIPE', 'CYLINDEREND', 'DIAMOND_STOP', 'TAGEND', 'INVTRAPEND', 'UNICODE_TEXT', 'TEXT', 'TAGSTART', got 'PS'

▶ Show details

$\delta$ Fix diagram
```

☐ AccAccountsCostcenter

Overview

Handles operations mapping accounts to cost centers, including search, CRUD, and retrieval by account or Id.

Endpoints

- POST /api/v1/Acc/AccAccountsCostcenter/Search Search account/cost center links
- POST /api/v1/Acc/AccAccountsCostcenter/Add Add link
- POST /api/v1/Acc/AccAccountsCostcenter/Edit Edit link
- DELETE /api/v1/Acc/AccAccountsCostcenter/Delete Delete link by Id
- GET /api/v1/Acc/AccAccountsCostcenter/GetByIdForEdit Retrieve for editing
- GET /api/v1/Acc/AccAccountsCostcenter/GetById Retrieve by Id
- GET /api/v1/Acc/AccountsCostcenter/GetByAccountsId Retrieve all cost centers for an account

Usage Example

Search Account-Cost Center Links

Search Account-Costcenter



Find links between accounts and cost centers.

POST

https://api.example.com/api/v1/Acc/AccAccountsCostcenter/Search

Headers

Authorization string • header required

Bearer <token>

Request body

JSON payload required for this request.

```
{ "AccAccountId": 101 }
```

Code examples

```
curl -X POST "https://api.example.com/api/v1/Acc/AccAccountsCostcenter/Search" \
   -H "Authorization: Bearer <token>" \
   -H "Content-Type: application/json" \
   -d '{ \"AccAccountId\": 101 }'
```

Responses

Success

```
{ "data": [ { "Id": 1, "AccAccountId": 101, "CcId": 5 } ] }
```

☐ ACCBankUsersApiController

Overview

Manages user permissions for bank records.

- GET /api/v1/Acc/ACCBankUsersApi/GetUsersPermissionACCBank Get users with permissions on a bank
- POST /api/v1/Acc/ACCBankUsersApi/Edit Set users with permission

• DELETE /api/v1/Acc/ACCBankUsersApi/Delete — Remove a user's permission from a bank

Usage Example

Get Users with Bank Permission

Get Users Permission for Bank

Export to Postman

Get users who have permission for a bank account.

GET

https://api.example.com/api/v1/Acc/ACCBankUsersApi/GetUsersPermissionACCBank

Headers

Authorization string • header required

Bearer <token>

Query parameters

id string required

Bank Id

Code examples

```
curl -X GET "https://api.example.com/api/v1/Acc/ACCBankUsersApi/GetUsersPermissionACCBank?id=Bank+Id" \
   -H "Authorization: Bearer <token>"
```

Responses

Success

```
{ "data": [ { "Id": 1, "Name": "John" } ] }
```

☐ AccAccountsTreeController

Overview

Provides tree-structured data of the chart of accounts.

Endpoints

 $\bullet \quad \text{GET /api/v1/Acc/AccAccountsTree/AccountsTree} \ \ - \ \ \text{Retrieves the full accounts tree}$

Usage Example

Get Accounts Tree

Export to Postman

Get tree structure of all accounts.

GET

https://api.example.com/api/v1/Acc/AccAccountsTree/AccountsTree



Authorization string • header required

Bearer <token>

Code examples

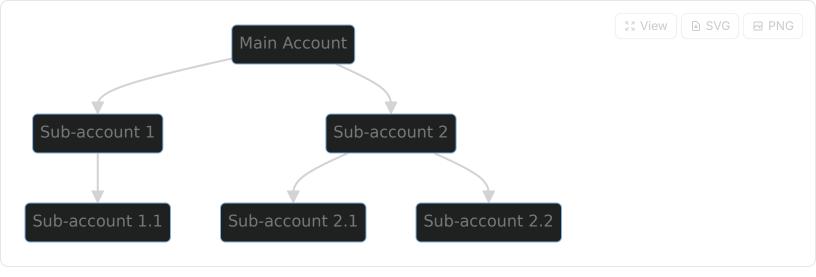
```
curl -X GET "https://api.example.com/api/v1/Acc/AccAccountsTree/AccountsTree" \
   -H "Authorization: Bearer <token>"
```

Responses

Success

```
{ "data": [ { "AccountId": 1, "AccountName": "Assets", "Children": [ ... ] } ] }
```

Accounts Tree Diagram



□ ACCCostCenterController

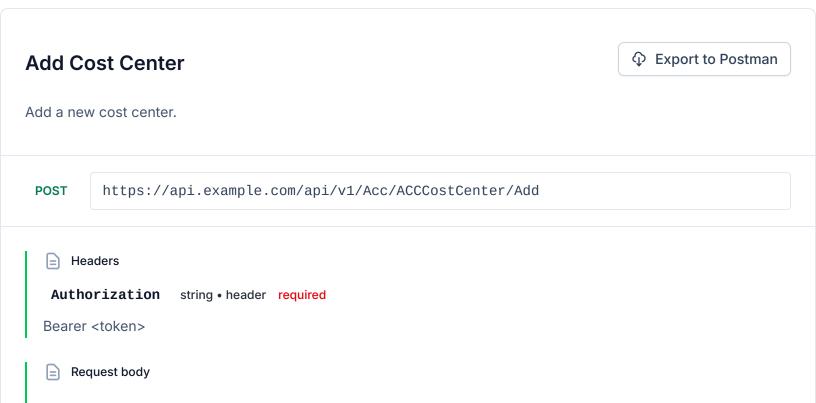
Overview

CRUD and search for cost centers.

Endpoints

- POST /api/v1/Acc/ACCCostCenter/Search Search cost centers
- POST /api/v1/Acc/ACCCostCenter/Add Add new cost center
- POST /api/v1/Acc/ACCCostCenter/Edit Edit cost center
- DELETE /api/v1/Acc/ACCCostCenter/Delete Delete cost center
- GET /api/v1/Acc/ACCCostCenter/GetByIdForEdit Get for edit
- GET /api/v1/Acc/ACCCostCenter/GetById Get by Id

Usage Example



JSON payload required for this request.

```
{ "CostCenterName": "IT Center" }
```

Code examples

```
curl -X POST "https://api.example.com/api/v1/Acc/ACCCostCenter/Add" \
   -H "Authorization: Bearer <token>" \
   -H "Content-Type: application/json" \
   -d '{ \"CostCenterName\": \"IT Center\" }'
```

Responses

Success

```
{ "success": true, "data": { "CcId": 5 } }
```

□ AccBankController

Overview

CRUD, search, and retrieval for bank accounts.

Endpoints

- GET /api/v1/Acc/AccBank/GetAll All banks
- POST /api/v1/Acc/AccBank/Search Search
- POST /api/v1/Acc/AccBank/Add Add
- POST /api/v1/Acc/AccBank/Edit Edit
- DELETE /api/v1/Acc/AccBank/Delete Delete
- GET /api/v1/Acc/AccBank/GetByIdForEdit Get for edit
- GET /api/v1/Acc/AccBank/GetById Get by Id

Usage Example

Get All Banks

Export to Postman

Retrieve all banks.

https://api.example.com/api/v1/Acc/AccBank/GetAll

```
Headers
```

Authorization string • header required

Bearer <token>

Code examples

```
curl -X GET "https://api.example.com/api/v1/Acc/AccBank/GetAll" \
  -H "Authorization: Bearer <token>"
```

Responses

Success

```
{ "data": [ { "BankId": 1, "BankName": "My Bank" } ] }
```

□ AccCashOnHandController

Overview

Manages cash on hand (petty cash) accounts.

Endpoints

- GET /api/v1/Acc/AccCashOnHand/GetAll
- POST /api/v1/Acc/AccCashOnHand/Search
- POST /api/v1/Acc/AccCashOnHand/Add
- POST /api/v1/Acc/AccCashOnHand/Edit
- DELETE /api/v1/Acc/AccCashOnHand/Delete
- GET /api/v1/Acc/AccCashOnHand/GetByIdForEdit
- GET /api/v1/Acc/AccCashOnHand/GetById

Usage Example

Get Cash On Hand By Id

Export to Postman

Fetch a cash on hand record by Id.

```
Headers

Authorization string • header required

Bearer < token >

Query parameters

id string required

Cash Account Id
```

Code examples

```
curl -X GET "https://api.example.com/api/v1/Acc/AccCashOnHand/GetById?id=Cash+Account+Id" \
   -H "Authorization: Bearer <token>"
```

Responses

Success

```
{ "data": { "Id": 8, "Name": "Main Cash" } }
```

☐ AccChequeBookApiController

Overview

Manages cheque books associated with bank accounts.

Endpoints

- GET /api/v1/Acc/AccChequeBookApi/GetAll
- GET /api/v1/Acc/AccChequeBookApi/GetByBankAll
- POST /api/v1/Acc/AccChequeBookApi/Add
- POST /api/v1/Acc/AccChequeBookApi/Edit
- POST /api/v1/Acc/AccChequeBookApi/Delete
- GET /api/v1/Acc/AccChequeBookApi/GetByIdForEdit
- GET /api/v1/Acc/AccChequeBookApi/GetById

Usage Example

Add Cheque Book

Add a cheque book for a bank account.

POST

https://api.example.com/api/v1/Acc/AccChequeBookApi/Add

Headers

Authorization string • header required

Bearer <token>

Request body

JSON payload required for this request.

```
{ "BankId": 1, "StartNo": "1001", "EndNo": "1100" }
```

Code examples

```
curl -X POST "https://api.example.com/api/v1/Acc/AccChequeBookApi/Add" \
   -H "Authorization: Bearer <token>" \
   -H "Content-Type: application/json" \
   -d '{ \"BankId\": 1, \"StartNo\": \"1001\", \"EndNo\": \"1100\" }'
```

Responses

Created

```
{ "success": true }
```

☐ AccAccountsLevelController

Overview

Handles the management and update of account levels (the digit/structure of account numbers).

Endpoints

- GET /api/v1/Acc/AccAccountsLevel/GetAll List levels
- POST /api/v1/Acc/AccAccountsLevel/Edit Update digit count and color for a level

Usage Example

Update Account Level Digits

Export to Postman

Update the digit count and color for an account level.

POST

https://api.example.com/api/v1/Acc/AccAccountsLevel/Edit



Authorization string • header required

Bearer <token>

Request body

JSON payload required for this request.

```
{ "LevelId": 3, "NoOfDigit": 5, "Color": "#0000ff" }
```

Code examples

```
curl -X POST "https://api.example.com/api/v1/Acc/AccAccountsLevel/Edit" \
   -H "Authorization: Bearer <token>" \
   -H "Content-Type: application/json" \
   -d '{ \"LevelId\": 3, \"NoOfDigit\": 5, \"Color\": \"#0000ff\" }'
```

Responses

Updated

```
{ "success": true }
```

ACCCashonhandUsersApiController

Overview

Manages user assignments to cash on hand records.

- GET /api/v1/Acc/ACCCashonhandUsersApi/GetUsersPermissionAccCashonhand
- POST /api/v1/Acc/ACCCashonhandUsersApi/Edit
- DELETE /api/v1/Acc/ACCCashonhandUsersApi/Delete

Get Users for Cash On Hand

Export to Postman

Get users with permissions for a cash on hand record.

GET

https://api.example.com/api/v1/Acc/ACCCashonhandUsersApi/GetUsersPermissionAccCashonha

Headers

Authorization string • header required

Bearer <token>

Query parameters

id string required

Cash Account Id

Code examples

```
curl -X GET "https://api.example.com/api/v1/Acc/ACCCashonhandUsersApi/GetUsersPermissionAccCashonhand?id=Cash+Account+Id" \
-H "Authorization: Bearer <token>"
```

Responses

Success

```
{ "data": [ { "Id": 1, "Name": "Alice" } ] }
```

☐ AccCostCenterTreeApiController

Overview

Returns a tree structure of all cost centers.

- GET /api/v1/Acc/AccCostCenterTreeApi/CostCentertTree
- GET /api/v1/Acc/AccCostCenterTreeApi/GetById

- GET /api/v1/Acc/AccCostCenterTreeApi/GetByload
- POST /api/v1/Acc/AccCostCenterTreeApi/Edit
- DELETE /api/v1/Acc/AccCostCenterTreeApi/Delete

Usage Example

Get Cost Center Tree

Export to Postman

Return a tree of all cost centers for the facility.

GET

https://api.example.com/api/v1/Acc/AccCostCenterTreeApi/CostCentertTree



Headers

Authorization string • header required

Bearer <token>

Code examples

```
curl -X GET "https://api.example.com/api/v1/Acc/AccCostCenterTreeApi/CostCentertTree" \
   -H "Authorization: Bearer <token>"
```

Responses

Tree

```
{ "data": [ { "CcId": 1, "CostCenterName": "Projects", "Children": [ ... ] } ] }
```

☐ AccFacilitiesApiController

Overview

Facility management endpoints for searching, retrieving, and updating facility data and profile.

- POST /api/v1/Acc/AccFacilitiesApi/Search
- POST /api/v1/Acc/AccFacilitiesApi/GetAll
- GET /api/v1/Acc/AccFacilitiesApi/GetByIdForEdit
- POST /api/v1/Acc/AccFacilitiesApi/Edit

- GET /api/v1/Acc/AccFacilitiesApi/GetFacilityStamp
- GET /api/v1/Acc/AccFacilitiesApi/EditStamp
- POST /api/v1/Acc/AccFacilitiesApi/GetMyFacilityProfile
- POST /api/v1/Acc/AccFacilitiesApi/EditProfile
- POST /api/v1/Acc/AccFacilitiesApi/GetAccFacility

Usage Example

Export to Postman **Get Facility Profile** Get the profile of the current facility. https://api.example.com/api/v1/Acc/AccFacilitiesApi/GetMyFacilityProfile **POST** Headers Authorization string • header required Bearer <token> Code examples curl -X POST "https://api.example.com/api/v1/Acc/AccFacilitiesApi/GetMyFacilityProfile" \ -H "Authorization: Bearer <token>" Responses Facility profile

☐ AccFirstTimeBalancesApiController

Overview

Manages first time balances (opening balances) for a financial year.

{ "data": { "FacilityId": 1, "FacilityName": "ABC Corp" } }

- GET /api/v1/Acc/AccFirstTimeBalancesApi/GetAll
- POST /api/v1/Acc/AccFirstTimeBalancesApi/Search

- POST /api/v1/Acc/AccFirstTimeBalancesApi/Add
- POST /api/v1/Acc/AccFirstTimeBalancesApi/Edit
- DELETE /api/v1/Acc/AccFirstTimeBalancesApi/Delete
- GET /api/v1/Acc/AccFirstTimeBalancesApi/GetByIdForEdit
- GET /api/v1/Acc/AccFirstTimeBalancesApi/GetById

Usage Example

Add First Time Balance

Export to Postman

Create a new first time balance entry.

POST

https://api.example.com/api/v1/Acc/AccFirstTimeBalancesApi/Add

Headers

Authorization string • header required

Bearer <token>

Request body

JSON payload required for this request.

```
{ "JCode": "FTB-2024", "Amount": 10000 }
```

Code examples

```
curl -X POST "https://api.example.com/api/v1/Acc/AccFirstTimeBalancesApi/Add" \
   -H "Authorization: Bearer <token>" \
   -H "Content-Type: application/json" \
   -d '{ \"JCode\": \"FTB-2024\", \"Amount\": 10000 }'
```

Responses

Created

```
{ "success": true }
```

This is a partial documentation. The solution includes many more API controllers for all accounting operations: journals, periods, groups, profiles, settlements, shared resources, reports, transfers, closing years, petty cash, types of expenses, and VAT groups. Each controller exposes several endpoints for CRUD and business actions. All endpoints enforce user permissions and return standardized responses.

For a complete API reference, continue with the same format for each controller and endpoint!

System Architecture

	KM View	SVG	₽ PNG
(C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-C-			

□ Security & Permissions

- · Every endpoint checks for user permissions before executing the business logic.
- Unauthorized calls receive an AccessDenied response.
- All modifications require valid input and model state.

☐ Response Format

All API responses are wrapped in a standardized structure, typically like:

```
1 {
2     "success": true,
3     "data": { ... },
4     "message": ""
5 }
```

Errors return:

```
1 {
2    "success": false,
3    "message": "Error description"
4 }
```

☐ Further Reading

- For detailed request/response models, refer to the DTO definitions.
- System configuration (e.g., account level titles, cost center visibility) is fetched dynamically based on facility.

Tip: Always use the correct HTTP method (GET, POST, DELETE) and provide the required headers, especially Authorization.

For help, contact your API administrator or refer to the in-code XML comments for each controller.

