

FxaAdditionsExclusionController.cs

This controller manages **additions and exclusions** on fixed assets. It provides endpoints to search, create, edit, and delete addition/exclusion records, and to populate dropdowns.

Dependencies:

- `IFxaServiceManager.FxaAdditionsExclusionService`
- `IAccServiceManager.AccFacilityService`
- `IPermissionHelper`
- `ICurrentData`
- `ILocalizationService`
- `IApiDDLHelper`
- `IWebHostEnvironment`

☐ Endpoints

Route	Method	Description
/Search	POST	Search additions/exclusions
/Add	POST	Create a new addition/exclusion
/DDL FxaAddExcType	GET	Load addition/exclusion type dropdown list
/GetByIdForEdit	GET	Retrieve a record for editing
/Edit	POST	Update an existing record
/Delete	DELETE	Delete a record

☐ Search

Performs filtered search on non-deleted records belonging to the current facility.

- **Route:** POST /Search
- **Input DTO:** `FxaAdditionsExclusionFilterDto`
- **Key Logic:**
 - Permission check (1986 , Show)
 - Default null numeric filters to 0
 - Date range filter using yyyy/MM/dd
 - Map VW entity to filter DTO

```
1 [HttpPost("Search")]
2 public async Task<ActionResult> Search(FxaAdditionsExclusionFilterDto filter)
3 {
4     // permission
5     filter.Id ??= 0;
6     filter.FixedAssetNo ??= 0;
7     var items = await fxaServiceManager.FxaAdditionsExclusionService
8         .GetAllVW(a => !a.IsDeleted && a.FacilityId == session.FacilityId
```



```
9         && (filter.Id == 0 || a.Id == filter.Id)
10         && (filter.FixedAssetNo == 0 || a.FixedAssetNo == filter.FixedAssetNo)
11         && (string.IsNullOrEmpty(filter.FixedAssetName) || a.FixedAssetName == filter.FixedAssetName));
12
13     // apply date filter, map to DTO, return
14 }
```

Add

Creates a new addition/exclusion record and generates a ZATCA QR code if successful.

- **Route:** POST /Add
- **Input DTO:** FxaAdditionsExclusionDto
- **Key Logic:**
 - Permission check (1986 , Add)
 - Model-state validation
 - Call service Add(obj)
 - On success, fetch facility data and generate QR code

DDL FxaAddExcType

Loads a dropdown of addition/exclusion types by typeId .

- **Route:** GET /DDL FxaAddExcType?TypeId={TypeId}
- **Returns:** SelectList of FxaAdditionsExclusionTypeDto

GetByIdForEdit

Fetches a record's details for editing, including related journal info.

- **Route:** GET /GetByIdForEdit?id={id}
- **Key Logic:**
 - Permission check (1986 , Edit)
 - Validate id > 0
 - Fetch VW record
 - Map to FxaAdditionsExclusionEditDto
 - Fetch related AccJournalMaster and set JId , JCode , PeriodId

Edit

Updates an existing addition/exclusion and regenerates the ZATCA QR.

- **Route:** POST /Edit
- **Input DTO:** FxaAdditionsExclusionEditDto
- **Key Logic:**
 - Permission check (1986 , Edit)
 - Model-state validation
 - Call service Update(obj)
 - On success, regenerate QR code

Delete

Soft-deletes a record by id .

- **Route:** DELETE /Delete?id={id}

- **Key Logic:**
 - Permission check (1986 , Delete)
 - Validate id > 0
 - Call service Remove(id)

FxaFixedAssetTypeController.cs

Manages **fixed asset types**, enabling CRUD operations and hierarchical relationships.

Dependencies:

- IFxaServiceManager.FxaFixedAssetTypeService
- IPermissionHelper
- ICurrentData
- ILocalizationService

Endpoints

Route	Method	Description
/GetAll	GET	Retrieve all asset types
/Search	POST	Filter asset types
/Add	POST	Create a new asset type
/GetByIdForEdit	GET	Fetch type for editing
/Edit	POST	Update an existing type
/Delete	DELETE	Remove a type

Class Overview

- **FxaFixedAssetTypeVm**: ViewModel
- **FxaFixedAssetTypeDto**: DTO for creation
- **FxaFixedAssetTypeEditDto**: DTO for editing

GetAll

Returns all non-deleted types for the current facility, populating MainAssetName for child types.

```
1 [HttpGet("GetAll")]
2 public async Task<ActionResult> GetAll()
3 {
4     // permission(901, Show)
5     var items = await fxaServiceManager.FxaFixedAssetTypeService.GetAllVw(
6         t => !t.IsDeleted && t.FacilityId == session.FacilityId);
7 }
```

```
8      // map to FxaFixedAssetTypeVm, fetch parent name if ParentId != 0
9  }
```

☐ Search

Filters by code, name, parent, rates, age, and account codes.

- **Input DTO:** `FxaFixedAssetTypeFilterDto`
- **Logic:**
 - Default null fields to `0`
 - Build predicate for all filter fields
 - Map to `FxaFixedAssetTypeVm`

☐ Add / ☒ Edit / ☐ Delete

Standard CRUD flows with permission checks (`Add=901` , `Edit=901` , `Delete=901`), model validation, and service calls.

FxaFixedAssetTransferController.cs

Handles **asset transfers** between branches, facilities, and employees.

Dependencies:

- `IFxaServiceManager.FxaFixedAssetTransferService`
- `IMainServiceManager.InvestEmployeeService`
- `IAccServiceManager`
- `IPermissionHelper`
- `ICurrentData`
- `ILocalizationService`

☐ Endpoints

Route	Method	Description
<code>/Search</code>	POST	Search transfer records
<code>/Delete</code>	DELETE	Remove a transfer
<code>/GetAssetByNo</code>	GET	Lookup asset data by number
<code>/Add</code>	POST	Create a new transfer
<code>/GetByIdForEdit</code>	GET	Fetch transfer details for editing
<code>/Edit</code>	POST	Update an existing transfer
<code>/GetAssetsByEmpCode</code>	GET	List assets owned by an employee

Route	Method	Description
/Add2	POST	Transfer multiple assets from one employee to another

❑ Search & Delete

- **Permission:** Show=799 , Delete=799
- **Filter:** Code, name, from/to locations, employees, date range
- **Mapping:** Map VW to FxaFixedAssetTransferSearchVM

❑ GetAssetByNo

Lookup an active asset by its `No` and optionally override `facilityId` .

- **Returns:** FixedAssetDataForTransferVm with branch, facility, cost center, employee, location.

❑ Add / Edit

- **DTOs:** FxaFixedAssetTransferDto , FxaFixedAssetTransferEditDto
- **Flow:** Permission → Validate → Service call

Add2

Special endpoint for transferring **multiple assets** between employees.

- **DTO:** FxaFixedAssetTransferDto2
- **Use case:** Bulk transfer in a single transaction

FxaDepreciationController.cs

Calculates and records **asset depreciation**.

Dependencies:

- IFxaServiceManager.FxaTransactionService
- IPermissionHelper
- ICurrentData
- ILocalizationService
- IMapper

❑ Endpoints

Route	Method	Description
/Search	POST	List depreciation transactions (TransTypeId = 5)
/GetLastDeprecDate	GET	Get date of last depreciation run

Route	Method	Description
/GetAssetForDeprec	POST	Compute depreciation for assets up to a given end date
/Add	POST	Record depreciation (transaction + asset entries)
/GetByIdForEdit	GET	Fetch depreciation record for editing
/Delete	DELETE	Remove a depreciation transaction

Search

- Filters by code and date range
- Maps to `FxaDepreciationFilterVM`

Asset Calculation

`GetAssetForDeprec` acts like a stored-procedure, gathering:

- Balance** from credit/debit sums
- LastDeprecDate** from previous records or asset start
- InstallmentValue** by method
- Month/day counts** to compute `DeprecValue` & `DeprecValueDialy`

Add

Delegates to `FxaTransactionService.Add_Depreciation(obj)` .

FxaPurchaseController.cs

Manages **asset purchase** records (TransTypeId = 2).

Dependencies:

- `IFxaServiceManager.FxaTransactionService`
- `IAccServiceManager`
- `IPermissionHelper`
- `ICurrentData`
- `ILocalizationService`
- `ISysConfigurationHelper`

Endpoints

Route	Method	Description
/Search	POST	Filter purchase records

Route	Method	Description
/Delete	DELETE	Delete a purchase record

Search

- Filters by code range, payment terms, branch, supplier, reference
- Date range filter
- Optional filter by `FxCode` to show only records containing a specific asset

FxaFixedAssetController.cs

Core controller for **fixed asset** CRUD, search, and detailed edit view.

Dependencies:

- `IFxaServiceManager.FxaFixedAssetService`
- `IAccServiceManager`
- `IPermissionHelper`
- `ICurrentData`
- `ILocalizationService`
- `ISysConfigurationHelper`

Endpoints

Route	Method	Description
/Search	POST	Search assets
/GetProperties	GET	Load asset-form properties (CC visibility)
/GetAssetTypeData	GET	Fetch type details for new asset entry
/Add	POST	Create a new asset
/GetByIdForEdit	GET	Fetch asset for view/edit
/FillGridViewsInEdit	GET	Populate grids: transactions, transfers, additions
/Edit	POST	Update asset
/Delete	DELETE	Remove an asset

Search

- Highly configurable filters: code, name, location, branch, status, classification, employee, addition type, date
- Hierarchical type filtering via `FxaFixedAssetTypeId_DF`

GetProperties

Fetches system config flags for cost-center fields visibility and titles.

☐ FillGridViewsInEdit

Aggregates multiple data sources for the edit screen:

1. **Transactions** (`FxaTransactionsAssetVw`)
2. **Related fixed assets**
3. **Transfers**
4. **Additions/Exclusions**

FxaBarController.cs

Generates **barcodes** and **QR codes** for fixed assets.

Dependencies:

- `IFxaServiceManager.FxaFixedAssetService`
- `IAccServiceManager`
- `IPermissionHelper`
- `ICurrentData`
- `IWebHostEnvironment`

☐ Endpoints

Route	Method	Description
<code>/Search</code>	POST	Filter assets for barcode generation
<code>/GetBarCodes</code>	GET	Generate and return bar/QR codes URLs

☐ Search

Filters assets similarly to the main asset search, returning minimal viewmodels.

GetBarCodes

- Accepts comma-separated `assetsIds`
- Fetches facility logo + generates facility QR
- Loops assets and generates EAN-style barcode URL

BaseFxaApiController.cs

Abstract base class applying the route prefix:

```
1  [Route("api/{ApiConfig.ApiVersion}/FXA/{controller}")]
2  [ApiController]
3  public abstract class BaseFxaApiController : ControllerBase { }
```

All FXA controllers inherit this for consistent routing and CORS.

FxaReportsController.cs

Provides **reporting** endpoints covering purchase, sale, depreciation, and category reports.

Dependencies:

- IFxaServiceManager.FxaFixedAssetService
- IFxaServiceManager.FxaTransactionService
- IAccServiceManager
- IPermissionHelper
- ICurrentData
- ILocalizationService
- IApiDDLHelper

☐ Endpoints

Route	Method	Description
/GetPurchaseAssetReport	POST	Report of purchased assets
/GetSaleAssetReport	POST	Report of sold assets
/GetDepreciationReport	POST	Asset depreciation report
/GetDepreciationReport3	POST	Depreciation by hierarchy level
/GetDepreciationReport2	POST	Alternate depreciation report
/GetDepreciationReportByCategory	POST	Depreciation aggregated by category
/DDLAssetType2	GET	Dropdown level-2 asset types
/DDLAssetType3	GET	Dropdown level-3 asset types (with descendants)

FxaAdditionsExclusionTypeController.cs

CRUD for **addition/exclusion types** (metadata).

Dependencies:

- `IFxaServiceManager.FxaAdditionsExclusionTypeService`
- `IPermissionHelper`
- `ILocalizationService`

☐ Endpoints

Route	Method	Description
/Search	POST	List types
/Add	POST	Create a type
/GetByIdForEdit	GET	Fetch type for editing
/Edit	POST	Update a type
/Delete	DELETE	Remove a type

FxaRevaluationController.cs

Handles **asset revaluation** transactions (TransTypeId = 8).

Dependencies:

- `IFxaServiceManager.FxaTransactionService`
- `IFxaServiceManager.FxaTransactionsRevaluationService`
- `IAccServiceManager`
- `IPermissionHelper`
- `ICurrentData`
- `ILocalizationService`

☐ Endpoints

Route	Method	Description
/Search	POST	Filter revaluation records
/Add	POST	Record new revaluation (transaction & journal)

Route	Method	Description
/Delete	DELETE	Delete a revaluation transaction

FxaTransactionsPaymentController.cs

Manages **payments** against FXA transactions.

Dependencies:

- `IFxaServiceManager.FxaTransactionsPaymentService`
- `IAccServiceManager`
- `IPermissionHelper`
- `ICurrentData`
- `ILocalizationService`
- `IApiDDLHelper`
- `IWebHostEnvironment`

☐ Endpoints

Route	Method	Description
/Search	POST	List payment records
/Add	POST	Create a payment
/Update	POST	Modify an existing payment
/DDLFxaAddExcType	GET	Dropdown for addition/exclusion types
/GetByIdForEdit	GET	Fetch payment for editing
/Edit	POST	Update payment details
/Delete	DELETE	Remove a payment record

SharedController.cs

Provides **shared utilities** for FXA modules.

Dependencies:

- `IFxaServiceManager.FxaFixedAssetService`
- `ISysConfigurationHelper`

- `ICurrentData`

Endpoints

Route	Method	Description
<code>/GetCostCenterProperties</code>	GET	Load cost-center visibility & titles
<code>/GetAssetDataByNo</code>	GET	Lookup asset financial data by its number

FxaSaleController.cs

Manages **asset sale** lifecycle (TransTypeld = 4).

Dependencies:

- `IFxaServiceManager.FxaTransactionService`
- `IAccServiceManager`
- `IPermissionHelper`
- `ICurrentData`
- `ILocalizationService`

Endpoints

Route	Method	Description
<code>/Search</code>	POST	List sale transactions
<code>/Add</code>	POST	Record a new sale
<code>/View</code>	GET	View sale details (read-only)
<code>/GetByIdForEdit</code>	GET	Fetch sale for editing
<code>/Edit</code>	POST	Update sale record
<code>/Delete</code>	DELETE	Remove a sale transaction

Note: All controllers extend `BaseFxaApiController` and share the prefix `/api/{version}/FXA/[controller]` . They uniformly enforce permission checks, session-based facility restrictions, and return `Result<T>` wrappers.