# DAM handover

Table of contents

[DAM handover 1](#_Toc181182825)

[**Project Background** 2](#_Toc181182826)

[**Function Introduction** 2](#_Toc181182827)

[**DAM & Hubspot** 4](#_Toc181182828)

[**Main object & parameter configuration** 4](#_Toc181182829)

[**Code Explanation** 6](#_Toc181182830)

## **Project Background**

Based on Hubspot storage service, it provides a tool for file upload and download interaction. Users can use the uploaded display link, edit file-related information and display sorting.

Env:

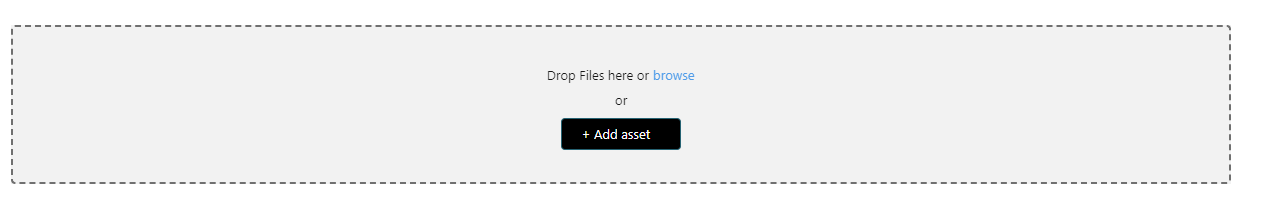
[Dev1a360v5](https://here2serve--dev1a360v5.sandbox.lightning.force.com/lightning/o/Product2/list)

[UATD](https://here2serve--uatd.sandbox.lightning.force.com/lightning/r/Product2/01tBU000000BE4TYAW/view)

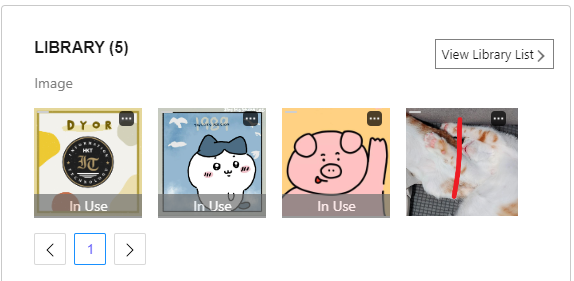
## **Function Introduction**

Search for Apps: products, select a data item to enter, click the ATTACHMENTS module and browse down.

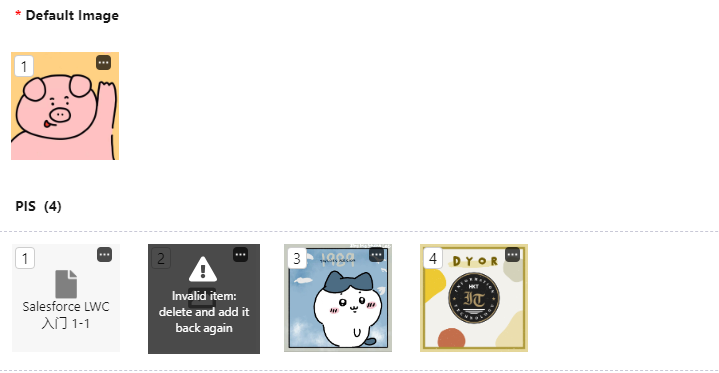
**Add Asset**

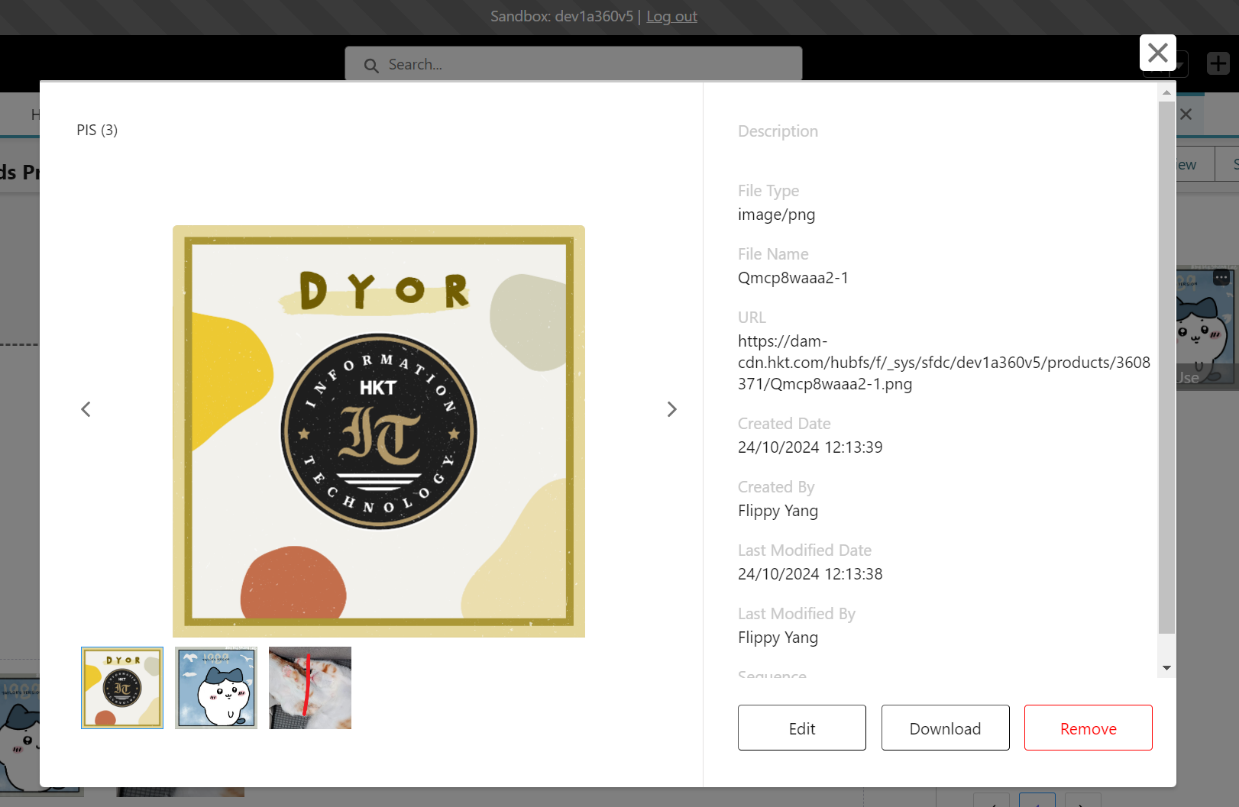
You can upload files by clicking Add or dragging and dropping files directly

**LIBRARY**

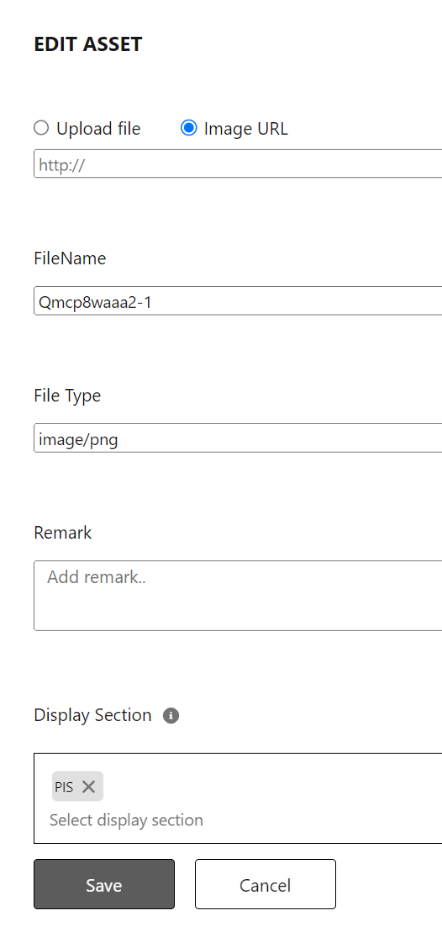
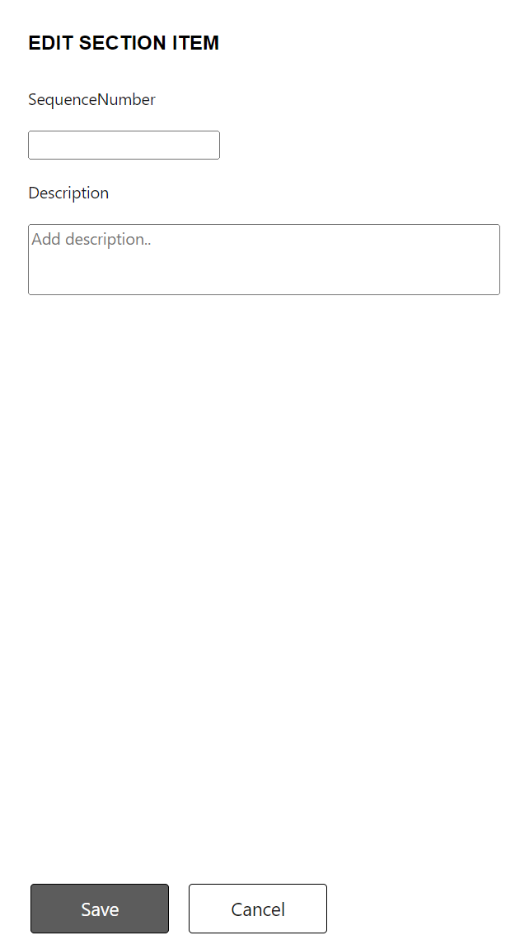
After the upload is successful, it will be displayed in the LIBRARY module on the right.

**SectionItem Show**

Drag the Default/PIS display module from the LIBRARY module The order of elements can be changed by dragging If the data associated with SectionItem, such as vlocity\_cmt\_\_VlocityAttachment\_\_c, is lost, an exclamation mark will appear as shown in the figure below, and modification is not allowed at this time

All elements can be clicked to expand the details page

**LibraryItem Edit** **SectionItem Edit**

## **DAM & Hubspot**

GitLab：<https://egit.btu.pccw.com/ruby/css/dam>

Interaction process：<https://miro.com/app/board/o9J_lyeYrWc=/>

Hubspot API：<https://developers.hubspot.com/docs/api/files/files>

Record Documentation：[DAM Components Data models and files.xlsx](https://pccw0.sharepoint.com/:x:/r/sites/CustomerSelfService/_layouts/15/Doc2.aspx?action=edit&sourcedoc=%7B92c9863a-0ae8-4b9c-8b2d-0ced28488d2f%7D&wdOrigin=TEAMS-ELECTRON.teamsSdk_ns.bim&wdExp=TEAMS-CONTROL&wdhostclicktime=1699589952899&web=1&ovuser=c5924da6-deb3-421b-aa98-57bcba0ba050%2CX0SS0027%40corphq.hk.pccw.com&clickparams=eyJBcHBOYW1lIjoiVGVhbXMtRGVza3RvcCIsIkFwcFZlcnNpb24iOiI0OS8yMzEwMTIyNzcwOCIsIkhhc0ZlZGVyYXRlZFVzZXIiOmZhbHNlfQ%3D%3D)

## **Main object & parameter configuration**

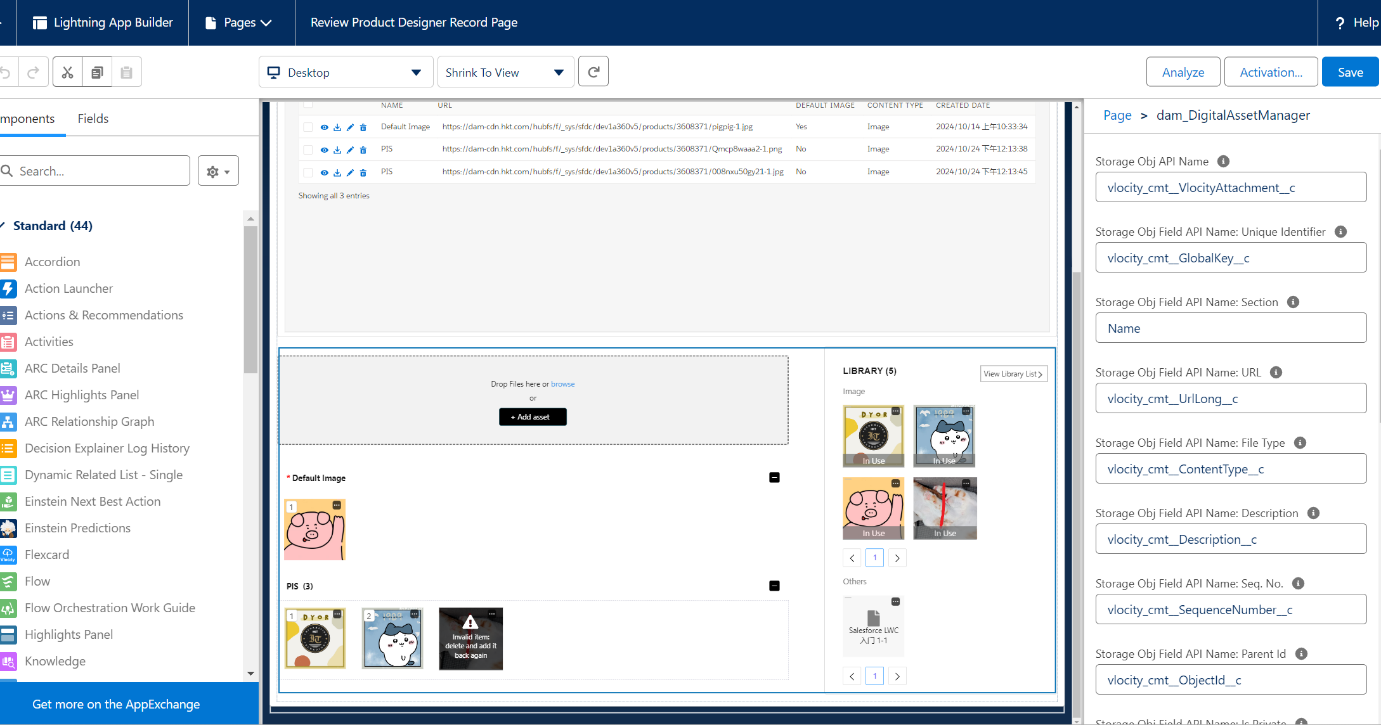
LibraryItem\_\_c：The object generated by uploading the file, that is, the Library module data

SectionItem\_\_c：Drag and drop objects generated by the display module through the Library，

Key associated with LibraryItem\_\_c ->Library\_Item\_\_c，Library\_Item\_UUID

Key associated with vlocity\_cmt\_\_VlocityAttachment\_\_c ->Asset\_Item\_ID\_\_c

vlocity\_cmt\_\_VlocityAttachment\_\_c：Drag and drop objects generated by the display module through the Library，Key associated with SectionItem\_\_c -> vlocity\_cmt\_\_GlobalKey\_\_c: Asset\_Item\_ID\_\_c

**Page Configuration** The general parameters are as follows：

{  
 **"assetItemApiName"**: **"vlocity\_cmt\_\_VlocityAttachment\_\_c"**,  
 **"sectionApiName"**: **"Name"**,  
 **"urlLongApiName"**: **"vlocity\_cmt\_\_UrlLong\_\_c"**,  
 **"fileTypeApiName"**: **"vlocity\_cmt\_\_ContentType\_\_c"**,  
 **"descriptionApiName"**: **"vlocity\_cmt\_\_Description\_\_c"**,  
 **"seqNoApiName"**: **"vlocity\_cmt\_\_SequenceNumber\_\_c"**,  
 **"parentIdApiName"**: **"vlocity\_cmt\_\_ObjectId\_\_c"**,  
 **"isDefaultImage"**: **"vlocity\_cmt\_\_IsDefaultImage\_\_c"**,  
 **"assetItemIdFieldApiName"**: **"vlocity\_cmt\_\_GlobalKey\_\_c"**,  
 **"uniqueCode"**: **"ProductCode"**,  
 **"baseDir"**: **"products"**,  
 **"isDefaultImageUsed"**: **true**,  
 **"sectionNamesString"**: **"PIS"**,  
 **"inputSectionItemObjectIdFieldApiName"**: **"ProductCode"**,  
 **"recordId"**: **"01tA2000000EvsrIAC"**,  
 **"objectApiName"**: **"Product2"**  
}

These configurations are included in the configApiNameParams declaration variable in Code

**Custom Metadata Types**

dam\_Configuration

**Trusted URLs**

dam\_Hubspot\_Public，dam\_Hubspot\_Private

**Named Credentials**

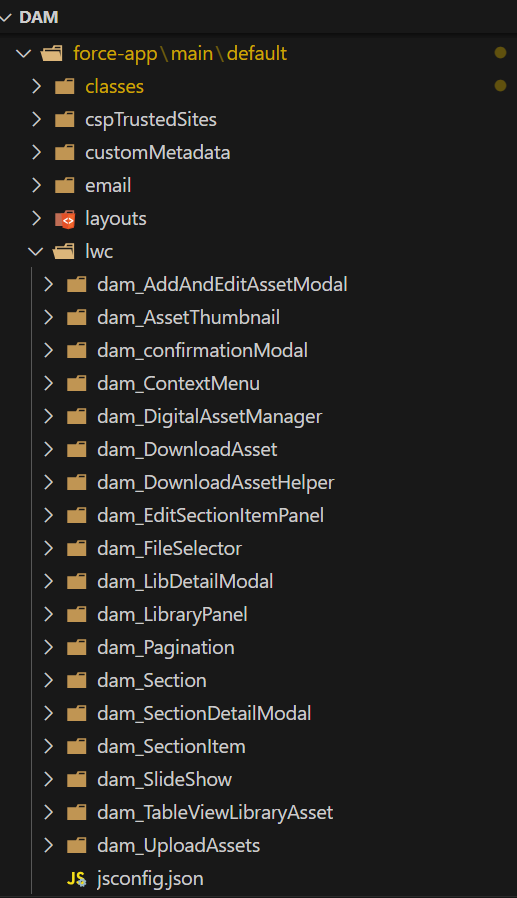
dam\_to\_mule

**Remote Site Settings**

dam\_Hubspot\_API ，dam\_hubspot\_cdn

## **Code Explanation**

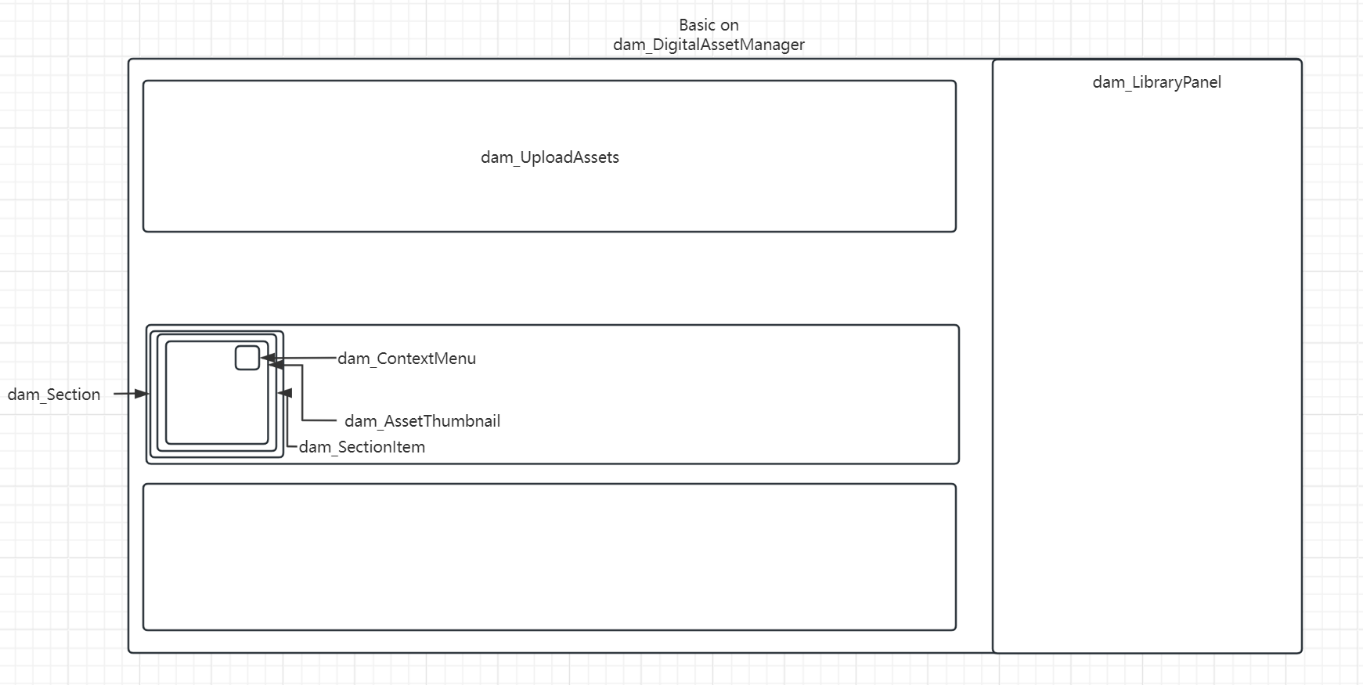
**Directory Structure**



Focus on the classes and lwc directories

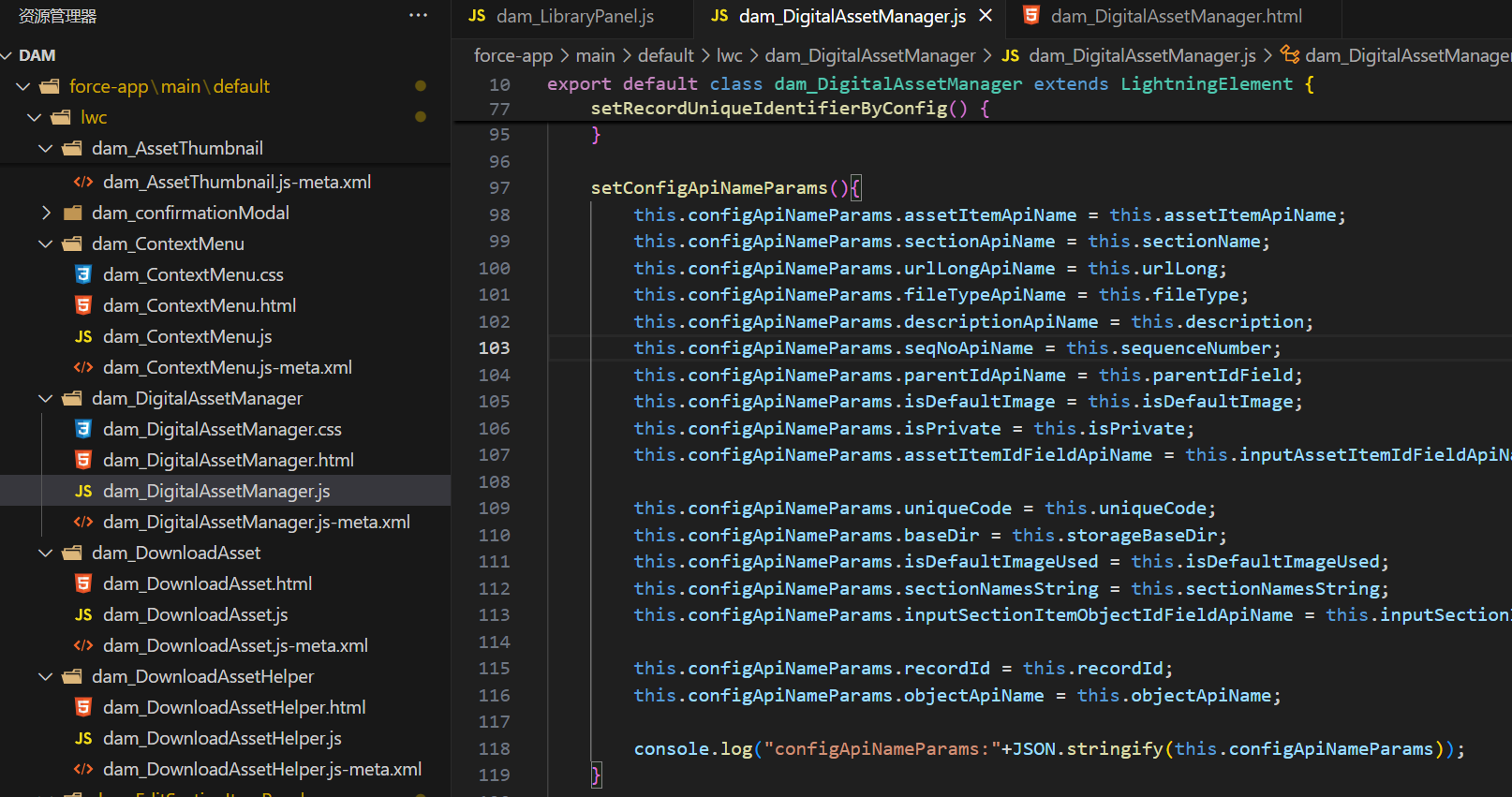
**LWC(Lightning Web Components)**

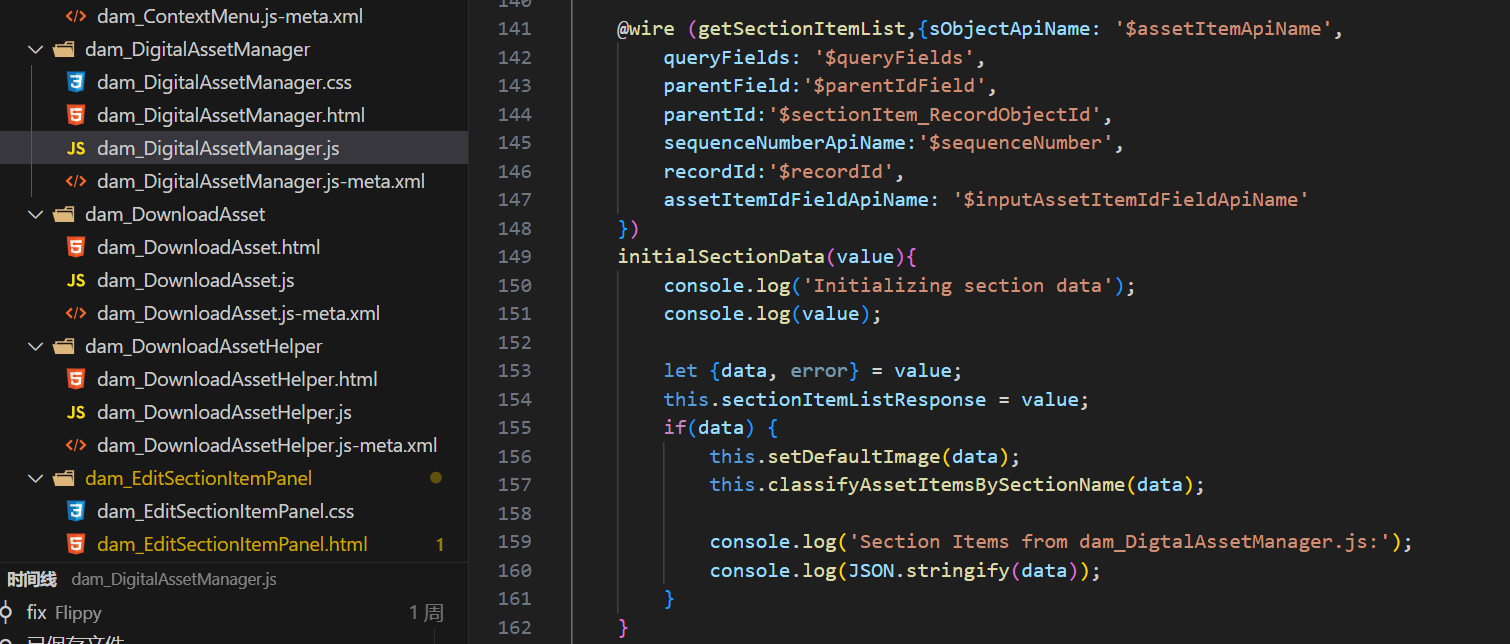
The basic layout is shown in the figure



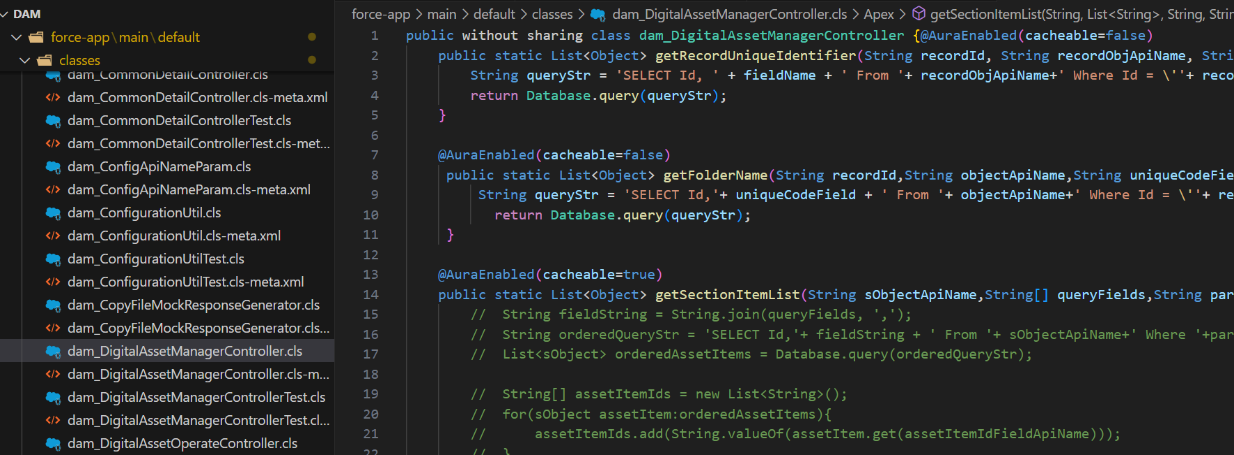
**Main frame**

dam\_DigitalAssetManager Initialize component configuration and use the configuration parameters in subsequent subcomponent passes



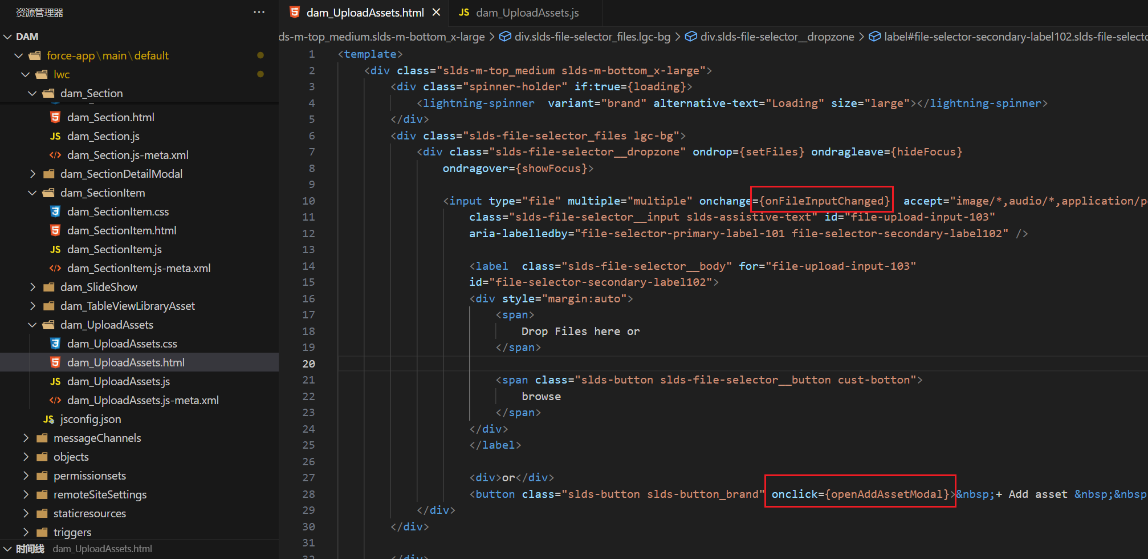
Read SectionIteml list

getSectionItemList is declared above in JS and used ->dam\_DigitalAssetManagerController

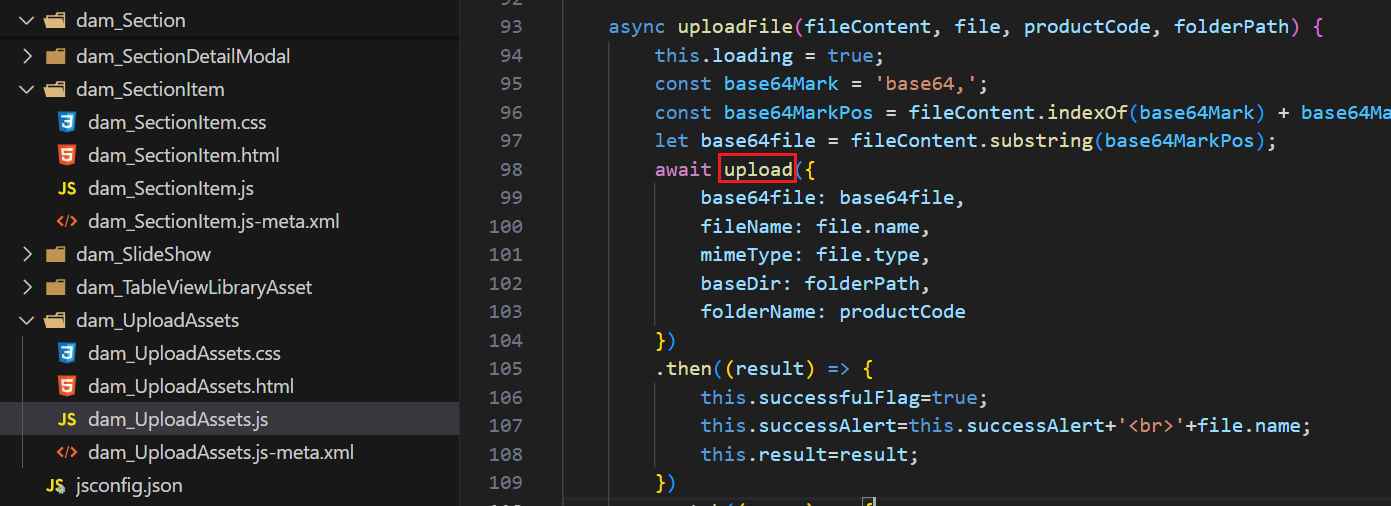
Verify data integrity while obtaining it, otherwise mark it for component to display error message

**Upload files**

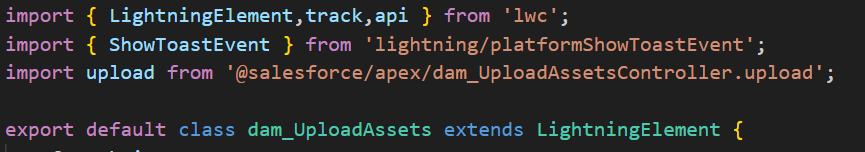
dam\_UploadAssets Key trigger method



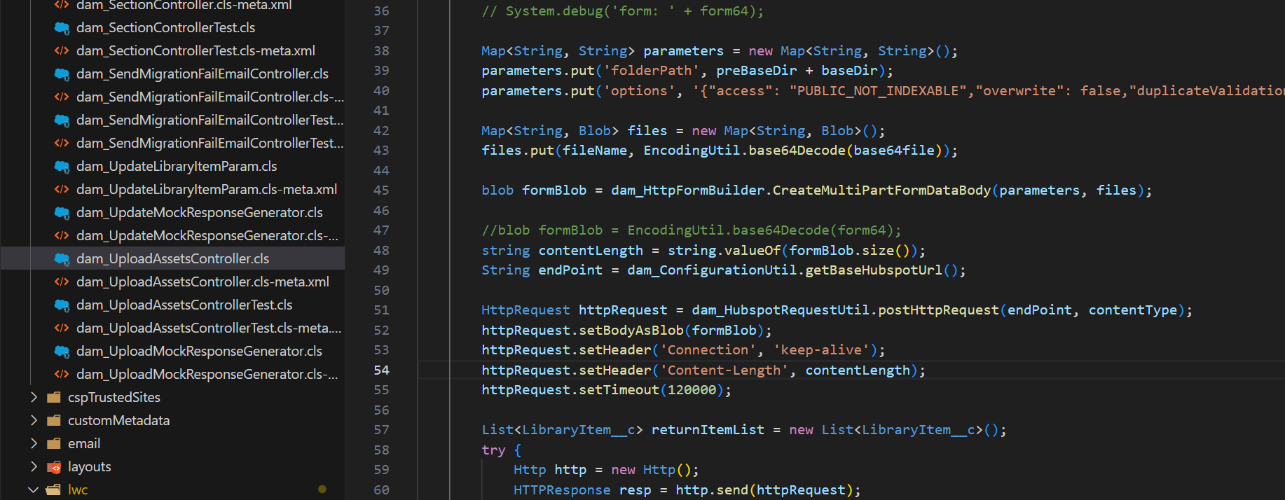
Key Implementation Methods



This method is declared above in JS

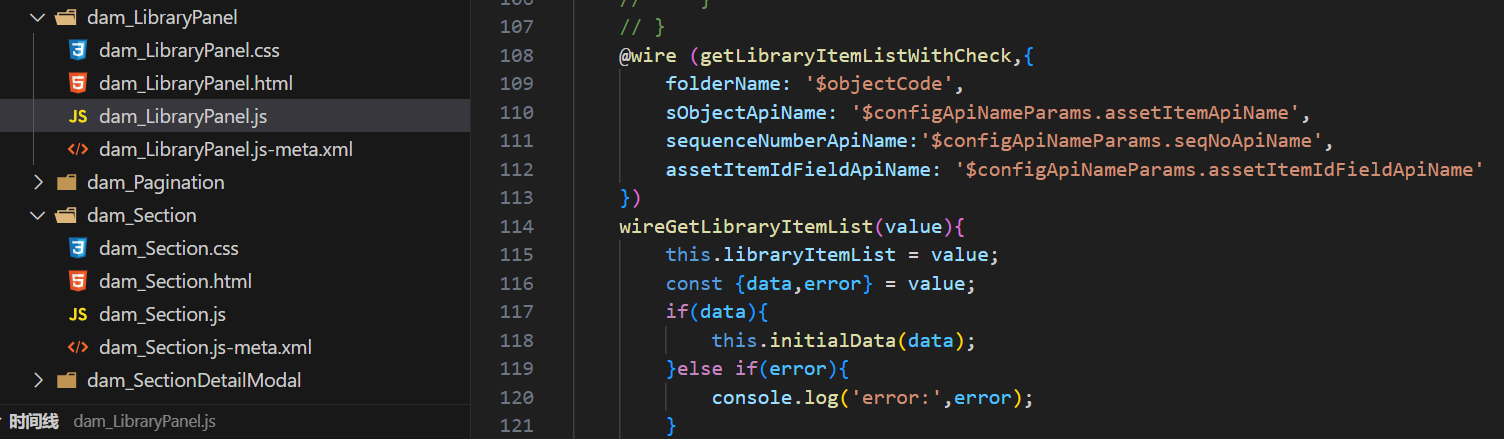


Core upload method section: dam\_ConfigurationUtil is used to obtain the API address configuration



**LIBRARY Module**

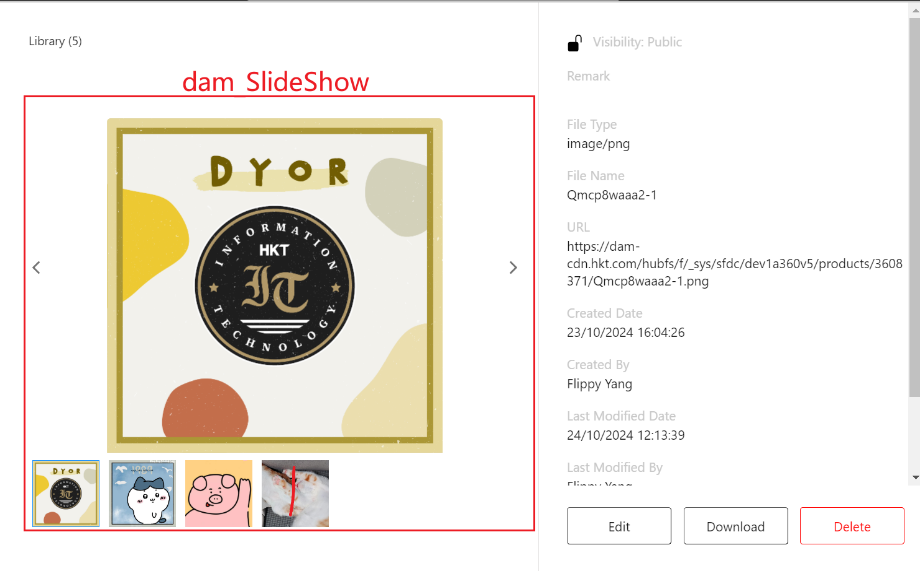
dam\_LibraryPanel Read LibraryItem List dam\_GetLibraryItemPanel.getLibraryItemListWithCheck



**Click LibraryItem**

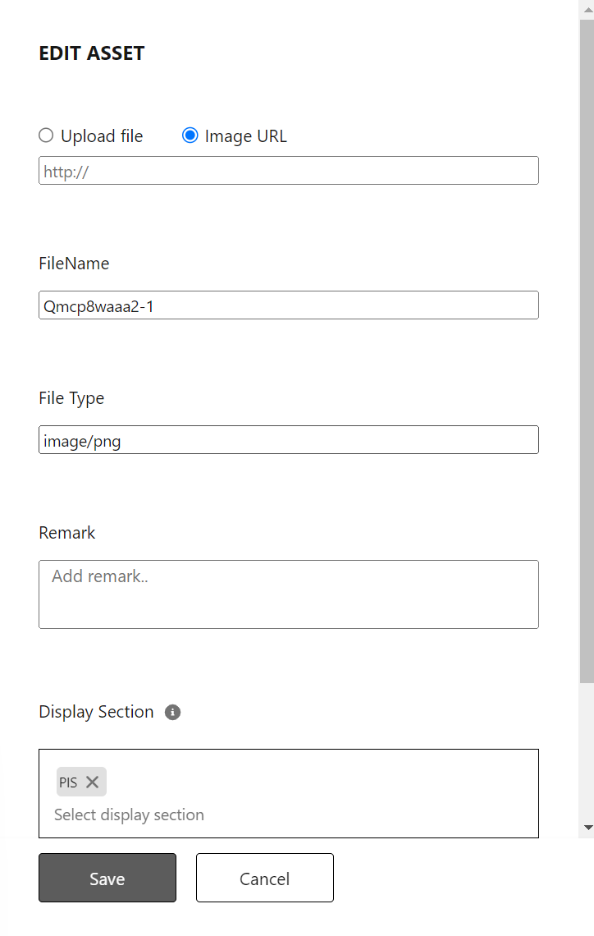
Key trigger method, onitemclicked={openModalDetail}

Page basic on dam\_LibDetailModal



**Click Edit**

Basic on dam\_AddAndEditAssetModal，The Display Section drop-down box can be configured by configApiNameParams. sectionNamesString



Key trigger method

dam\_AddAndEditAssetModal.upsertAssetData

dam\_DigitalAssetOperateController.updateLibraryItemWithFile/updateLibraryItemWithoutFile

The method mainly involves logic: whether to re-upload the file for replacement, whether the Display Section has been modified, if it has been added, change it to PULIC permission and add a SectionItem and Attachment data, if it has been removed, change it to PRIVATE permission and delete the SectionItem and Attachment data (removeSecionItems)

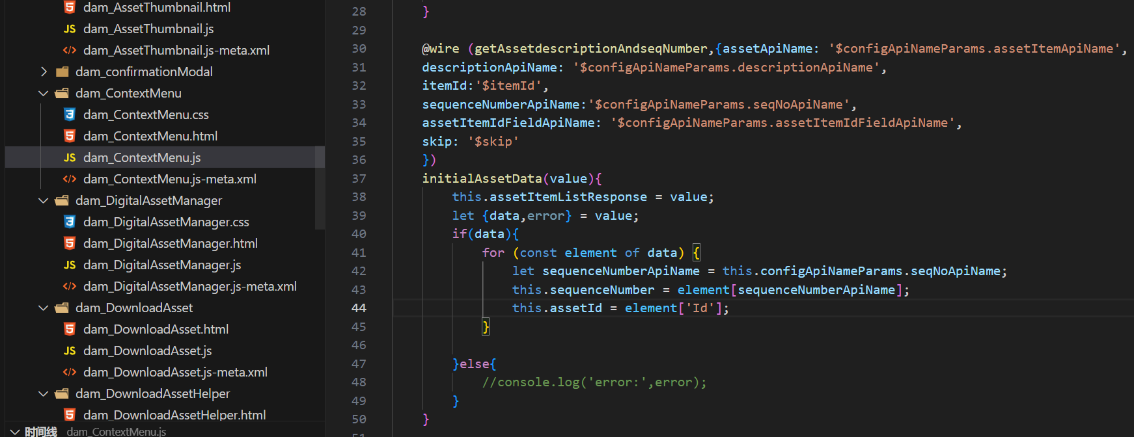
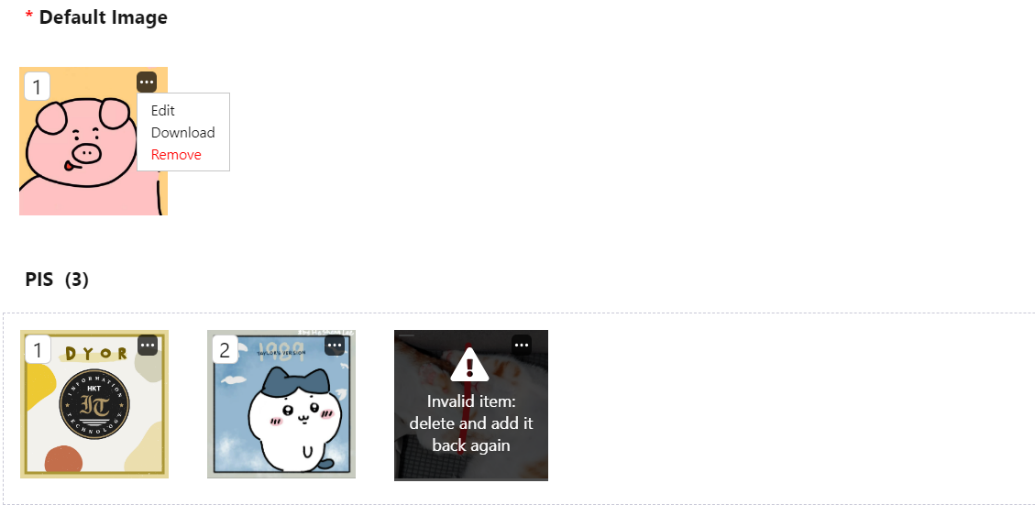
**Section Module**

Sections will be sorted with serial numbers, which are read from dam\_AssetThumbnail

The Edit and Remove functions are triggered by HandleEdit and HandleRemove of dam\_ContextMenu and passed to the upper component for further processing.

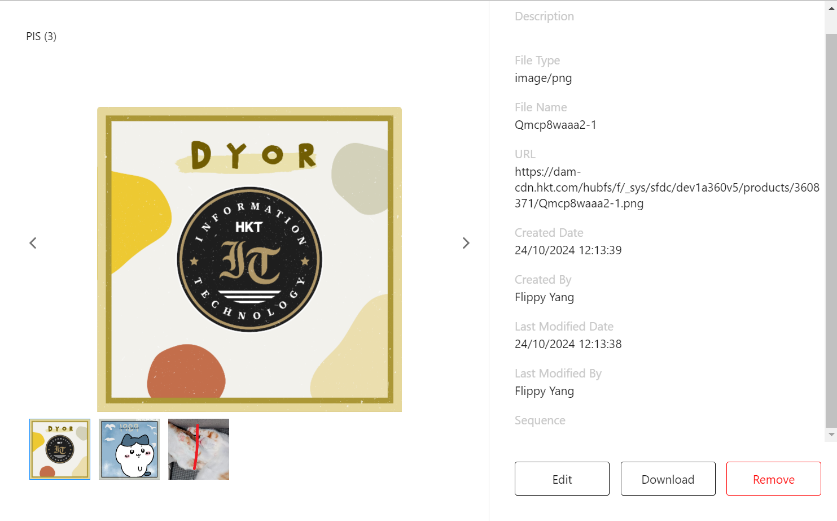
For example: Edite->HandleEdit-> dam\_AssetThumbnail.handlerEdit-> dam\_SectionItem.handlerEdit -> dam\_Section.handlerEdit

Syntax example: this.dispatchEvent(new CustomEvent('editclick', { 'detail': { 'itemId': id} })); Listened by oneditclick in html and passed to the specified method

Receiving example: handleEdit(event){let id = event.detail.itemId}

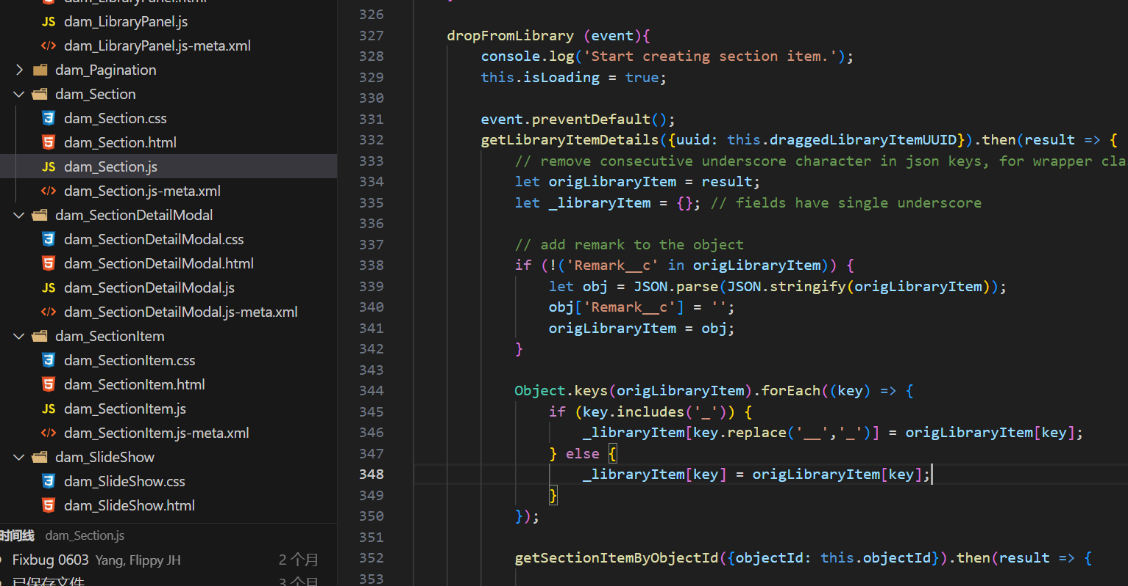
**Click SectionItem**

Page basic on dam\_SectionDetailModal



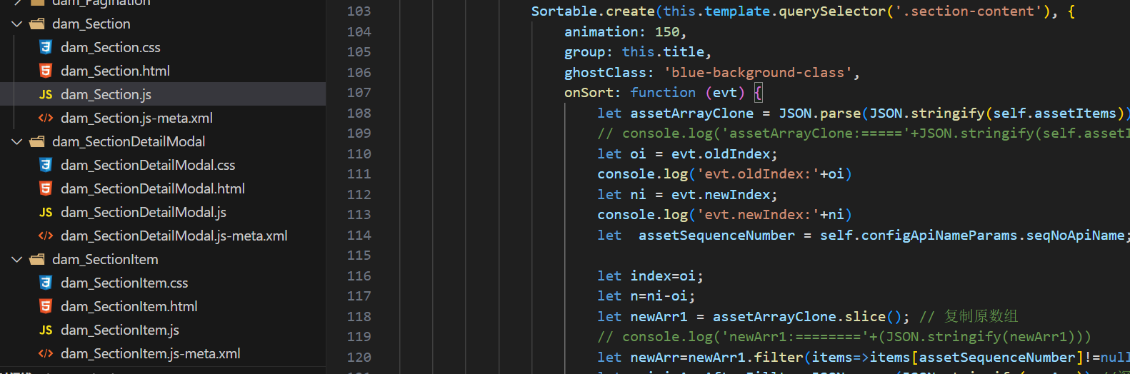
**Drag SectionItem from Library**

Trigger Method：dropFromLibrary Key Methods：dam\_SectionController .newSectionAndUpload



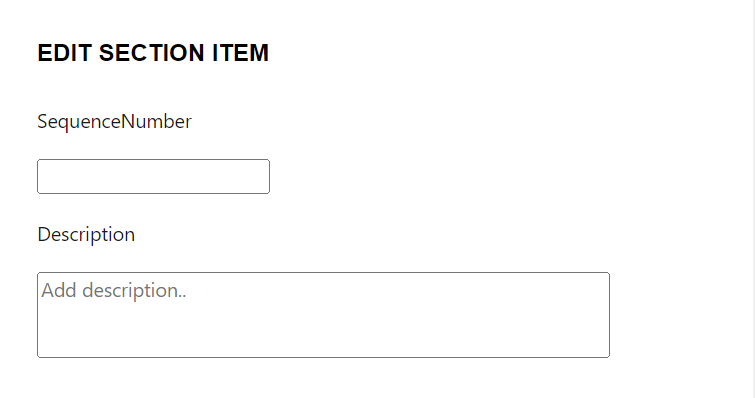
**Drag Section to change the order**

dam\_Section.Sortable.create



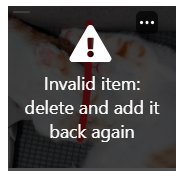
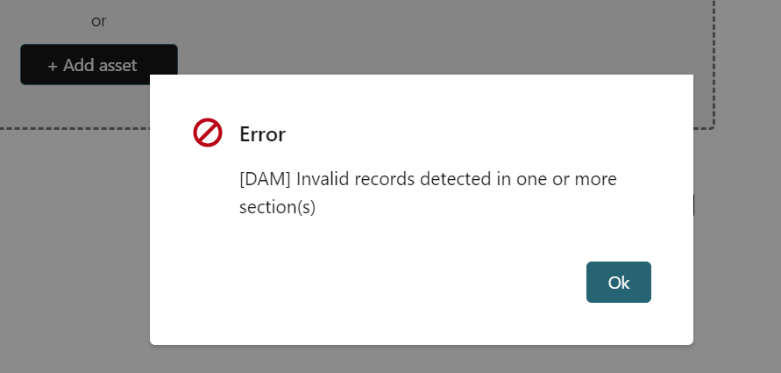
**Edit Section**

Key Methods： dam\_EditSectionItemPanel.updateData



**Abnormal Section Data Display**

If the data associated with SectionItem, such as vlocity\_cmt\_\_VlocityAttachment\_\_c or LibraryItem\_\_c, is lost, an exclamation mark will appear as shown in the figure below, and modification is not possible at this time.



Code: dam\_DigitalAssetManagerController.getSectionItemList

