**Updates: 28.11.2025**

**Updates That Need to Be Modified:**

1. **UI Improvements:** The user interface requires enhancement. Please standardize the font across the platform and reduce the font size to ensure the module appears clean and organized. If there is any uncertainty regarding the UI implementation, kindly share the wireframes for review so that we can select the most suitable design.
2. **Service Page Layout:** The service page should open in full-frame mode. Pop-ups on this page should be eliminated.
3. **Complete Variable Inclusion:** Ensure that all relevant variables are included in each section, as some are currently missing.
4. **Tooltip Implementation:** Tooltips should be added for all variables to improve clarity and usability.
5. **Asset Linking:** The asset linking process needs refinement. It should be clear where assets can be added and how linked assets can be viewed in relation to the service.
6. **Usage Metrics:** Implement automatic usage metrics. For example, if a keyword is linked to a service, the system should automatically populate the count of how many times the keyword is used for that service.
7. **Master Table Integration:** Variables must be linked to their corresponding master tables. For instance, the Industry ID should be connected to the industry table.
8. **Data Consistency and Workflow:** All information should be properly saved and function seamlessly within the workflow.

**Next Steps:**  
Please prioritize these updates. We will first thoroughly test the service page before proceeding to subsequent modules.

05.12.25:

1. Make the Description box larger and make in horizontal

A screenshot of a computer

AI-generated content may be incorrect.

1. Navigation 🡪 Parent Menu section is missing.

A screenshot of a computer

AI-generated content may be incorrect.

1. Strategic 🡪 Content type (Dropdown is missing), Secondary persona id is missing, linked campaigns id,

A screenshot of a computer

AI-generated content may be incorrect.

1. Content 🡪 h4\_list | h5\_list | body content | internal links | external links | image\_alt\_texts | wordcount | reading\_time\_minutes

A screenshot of a computer

AI-generated content may be incorrect.

1. SEO 🡪 Seo Score, ranking summary is missing

A screenshot of a computer

AI-generated content may be incorrect.

1. Technical 🡪 redirect\_from\_urls | hreflang\_group\_id | include\_in\_sitemap | core\_web\_vitals\_status | tech\_seo\_status | sitemap\_priority

A screenshot of a computer

AI-generated content may be incorrect.

1. Linking 🡪 Can’t be able to Link with Asset (kindly link with asset)

A screenshot of a computer

AI-generated content may be incorrect.

1. Governance 🡪 Brand is not displaying,created by | created at | updated by | updated at | version number is missing

A screenshot of a computer

AI-generated content may be incorrect.

08.12.25:

**Asset Workflow:**

1. Upload the asset and select the appropriate repository (e.g., SMM, Content, SEO).
2. Specify the asset type (Image, Video, Audio), set its status, and enter the mapped services, sub-services, or pages (as listed in the Service Master or Sub-Service Master), then upload the asset.
3. If the asset is mapped to a service in the Service Master, it should automatically link under that service.
4. If the asset is mapped to a sub-service, it should automatically link under the corresponding sub-service.
5. After uploading, the asset will be linked to the relevant service. Users can also search within a specific repository to view a list of assets. From this list, they can select and link the desired asset(s) to the service.

In **Create New Service → Linking**:

1. The Asset Library Management section should appear first.
2. The linking metadata should be displayed based on the Asset Management section.
3. In the linking metadata, the asset count and sub-service count should be fetched in real-time based on the linking.

For example: If a service is linked with a sub-service, the sub-service count should automatically update to 1.

**In Create New Service 🡪 Content:**

1. In Body content editor 🡪 When I type a letter, its typing under the S.NO.

**In Create New Sub service 🡪 Linking:**

1. All asset management items and the assets linked to the sub-service should be listed under the Linking section.
2. All fields available in the service master should also be present in the sub-service master.

09.12.25:

Under **Assets**, add a new section titled **Asset Applications**.  
This section should provide a dropdown with the following options:

* Web
* SEO
* SMM

**When “Web” is selected**, the following fields must be displayed:

* Title
* Description
* Keywords
* URL
* H1
* H2
* H2
* Thumbnail/Blog Image (Upload option)
* Body Content

**When “SMM” is selected**, an additional dropdown should appear with individual platforms:

* Facebook/Instagram
* Twitter
* LinkedIn

Based on the selected platform, the following fields should be shown separately for each:

* Title
* Tag
* URL
* Description
* Hashtags
* Media Upload (Image/Video/Carousel/gif)

Each selected social media platform should also have a **Preview** option.

10.12.25:

**1. Asset Submission – User Entry Fields (In Order)**

Users creating or submitting an asset must complete the following fields in the exact sequence:

1. **Asset Application – WEB, SEO, SMM**
2. **Service**
3. **Sub-Service Linking**
4. **Keywords – Link with keyword master table**
5. **Title**
6. **Description**
7. **URL**
8. **H1**
9. **H2**
10. **Asset Type**
11. **Asset Category**
12. **Asset Format**
13. **Repository**
14. **Image Upload Option – preview option based on the SMM.**
15. **Body Content**
16. **SEO Score (Integrate AI for SEO score check)**
17. **Grammar Score (Integrate AI for grammar score check**)
18. **Usage Status**
19. **Status**

**Submit Button** – Triggers workflow and sends the asset for approval.

**2. Approval Workflow**

**2.1 User Submission**

* The submitter fills all fields listed above.
* Upon clicking **Submit**, the asset moves to **Pending QC Review** status.
* The asset **cannot** link to a service or sub-service until QC approval is completed.

**3. QC Review Module**

**3.1 View Access**

* When QC personnel open the item, it should appear strictly in **List View**.
* **Fields must NOT be editable** in QC view.
* QC reviewers should only see structured data, without body editing access.

**3.2 QC Input Fields**

QC team can input the following:

1. **QC Score**
2. **Checklist Completion**
3. **Remarks / Comments**
4. **Submit Review**

**3.3 post-QC Action**

* If **Approved**:  
  The asset becomes active and automatically links to the selected **Service** or **Sub-Service**.
* If **Rejected**:  
  It returns to the creator with remarks for correction, and status changes to **Rework Required**.

**4. Functional Requirements Summary**

**4.1 User Mode (Asset Creator)**

* Full edit access to all required fields.
* Ability to submit for approval.
* Cannot mark QC score.
* Cannot approve or activate linking.

**4.2 QC Mode**

* List view only (read-only asset data).
* Input allowed only on QC scoring fields.
* No modification to original content.
* Ability to approve or reject.

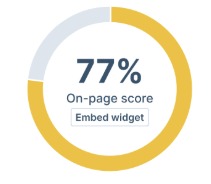
**4.3 System Behaviour**

* Linking to service/sub-service becomes **active only after QC approval**.
* SEO and Grammar scores are mandatory before submission.
* Workflow log must capture:
  + Submission time
  + Submitted by
  + QC reviewer
  + Approval/rejection timestamps

**5. Output / UI Expectations**

* Clean, structured form for submission.
* QC list view with summarized fields.
* Dedicated QC panel for scoring and remarks.
* Automated status transitions:
  + **Draft → Pending QC → Approved → Linked**
  + Or **Pending QC → Rejected → Rework Required**

For Grammer Score: You can Implement like this



SCREENS:

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

AFTER SUBMITTING APPROVAL:

A screenshot of a computer

AI-generated content may be incorrect.

**In List View 🡪 Add ID, Rework count, status**

While Clicking on Review,

A screenshot of a computer

AI-generated content may be incorrect.

* 1. Add Approve, Reject, Rework
  2. The checklist should change based on the application

FOR QC REVIEW:

**SEO – Top 5 Quality Checks**

1. Meta title & description optimized with primary keyword
2. H1, H2 structure correctly applied and keyword-aligned
3. Image alt text optimized and relevant
4. Page speed and mobile responsiveness verified
5. Internal links added and all links functioning

**WEB – Top 5 Quality Checks**

1. All pages load smoothly without errors
2. Forms, buttons, and links fully functional
3. Website responsive across all devices
4. Visual consistency (fonts, spacing, colors) maintained
5. SSL active and no security warnings

**SMM – Top 5 Quality Checks**

1. Caption aligned with brand tone and error-free
2. Correct post dimensions (image/video) used for each platform
3. Hashtags relevant and optimized
4. Visual/creative matches brand guidelines
5. All links, tags, and CTAs verified and functioning

KEY POINTS TO DEVELOPER:

1. In the QC list view, the QC team should see a “Review” action.
2. For the user, the status should display as “Approved,” and the asset should be linked to the respective service.

**Approval Workflow**

**2.1 User Submission**

* The submitter fills all fields listed above.
* Upon clicking **Submit for QC approval**, the asset moves to **Pending QC Review** status.
* The asset **cannot** link to a service or sub-service until QC approval is completed.

**3. QC Review Module**

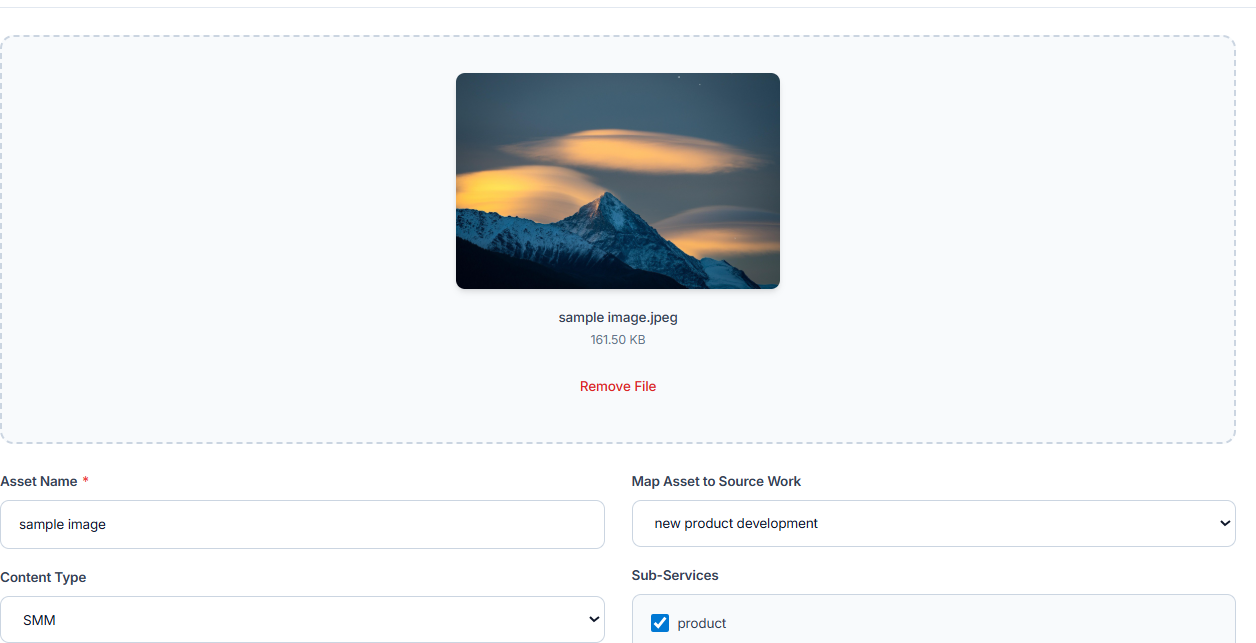
**3.1 View Access**

* When QC person open the item, it should appear strictly in **List View**.
* **Fields must NOT be editable** in QC view.
* QC reviewers should only see structured data, without body editing access.
* After approval, the Asset should link to the service page.

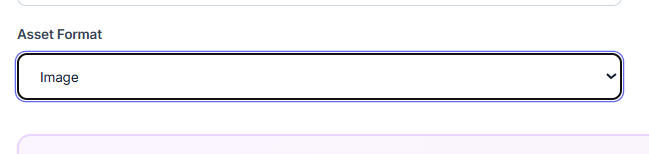
Edit Access – User Submission:  
In the QC review stage, users should only have access to the **Edit** and **Delete** options under Actions.  
Only the admin should see the **Review** action, including **Approve**, **Reject**, and **Rework** options.

16.12.25:

* 1. In SMM 🡪 It should have only one image upload



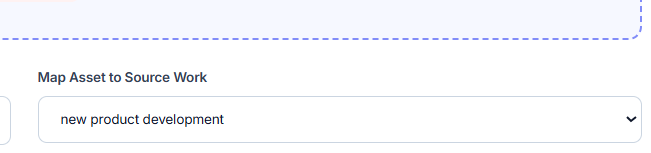
* 1. In Web, SMM, SEO 🡪 Asset format should link with Asset Master:



* 1. Remove Usage status:



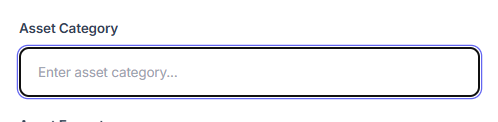
4.Rename this to “Map Asset to Services”

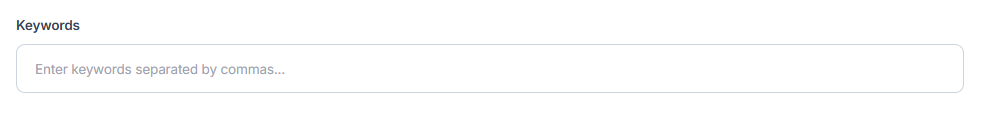


5.Make this content type static, it is now editable even after choosing WEB



6. Convert the **Asset Category** into a master table so that any entries added in the backend are automatically listed here, allowing users to select from them.

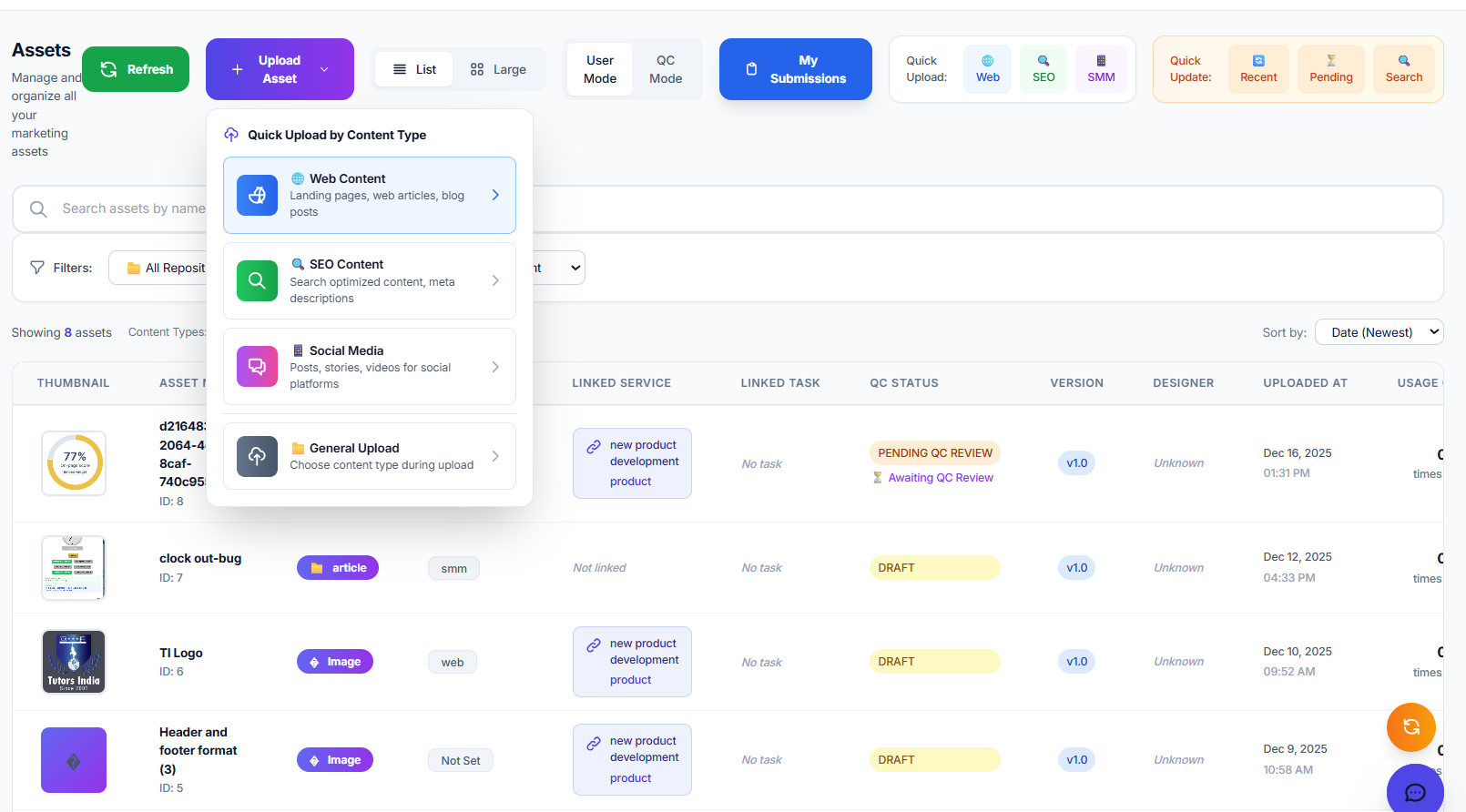


7.Keywords should integrate master database 

17.12.25:

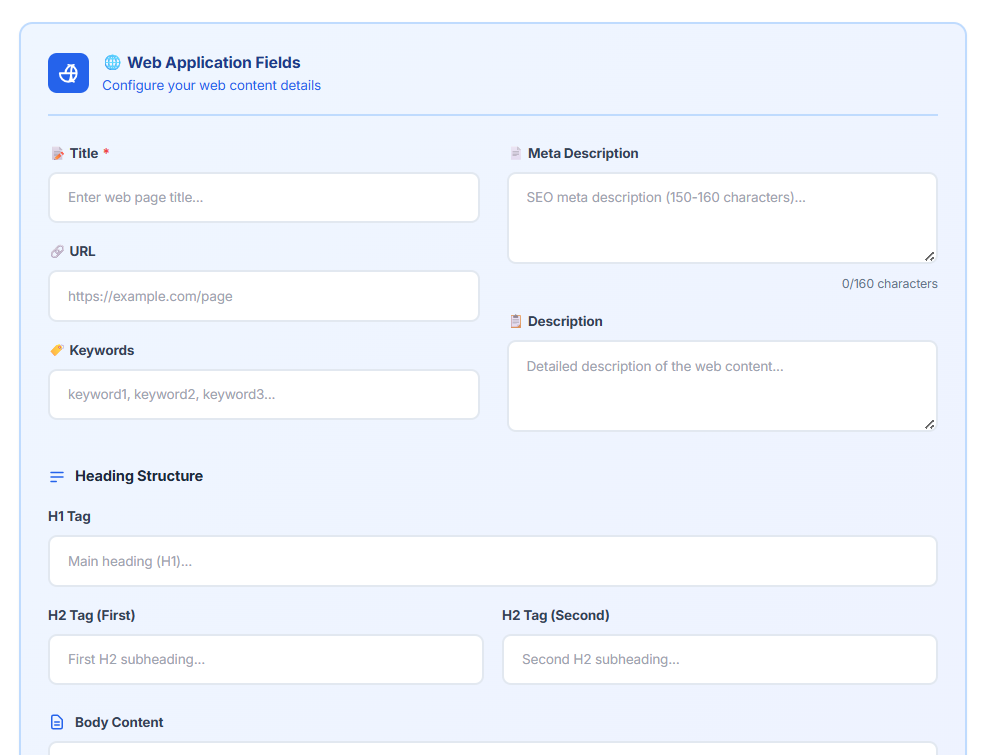
Step 1:

By Clicking one Upload New asset 🡪 The User will click on “WEB”, “SEO”, “SMM”



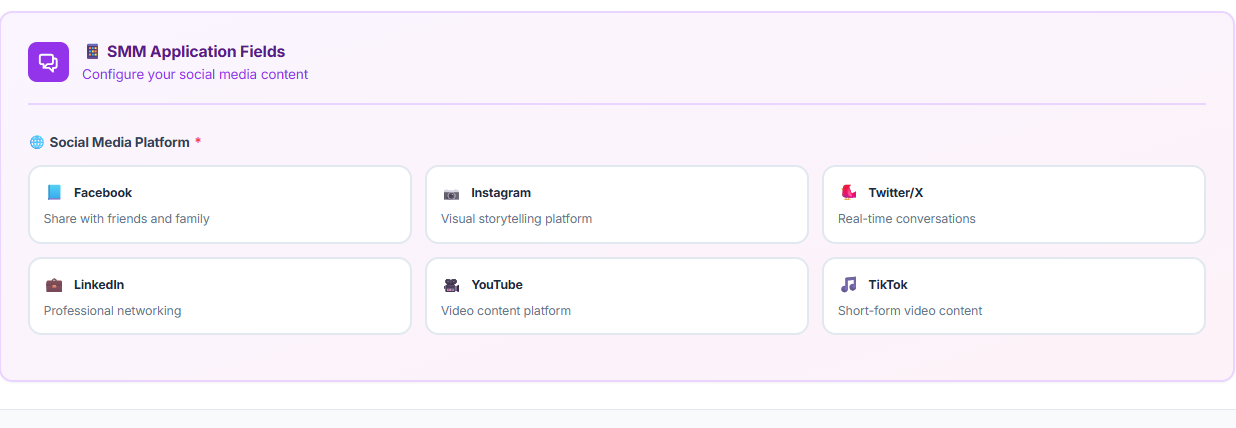
Step 2:

While clicking on Web 🡪 the Web application field should appear first Up to Quality check



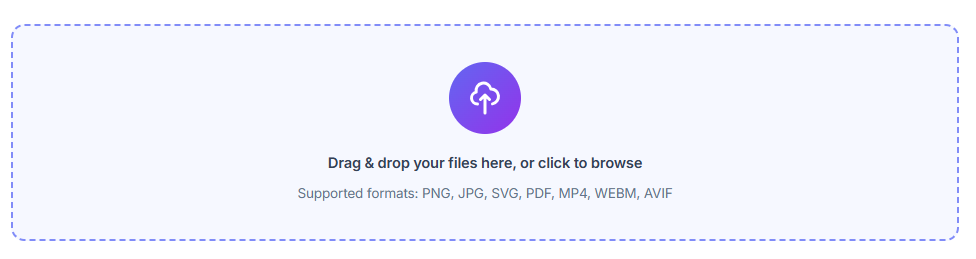
While clicking on SEO 🡪 the SEO application field should appear:



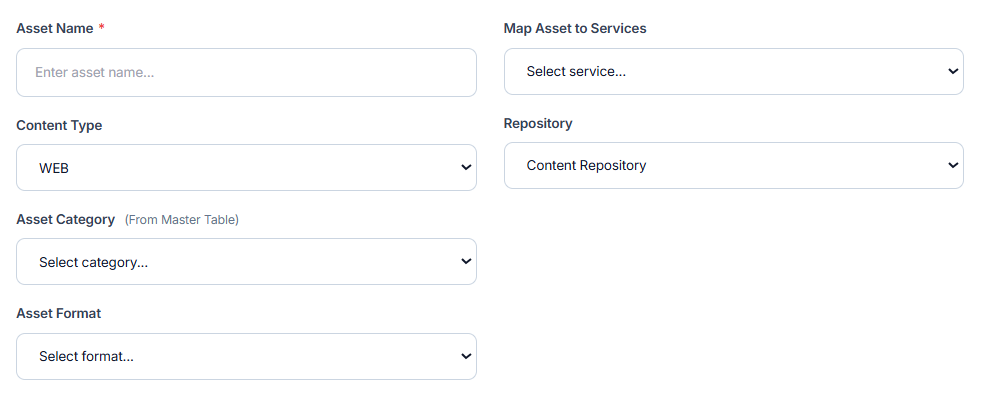
While Clicking on SMM 🡪 The SMM application should appear  
  


Step 3:

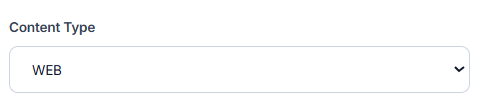
After that display, the Upload option



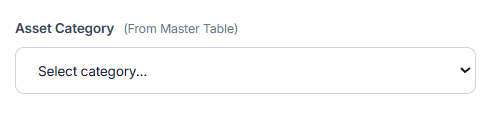
Step 4 : After that, Display the below fields



Step 5: Set the content type to ‘Web’. If the user selects SEO, lock the content type as SEO; if the user selects SMM, lock it as SMM. Whatever the user chooses when uploading the asset, it should automatically be frozen to that selection.

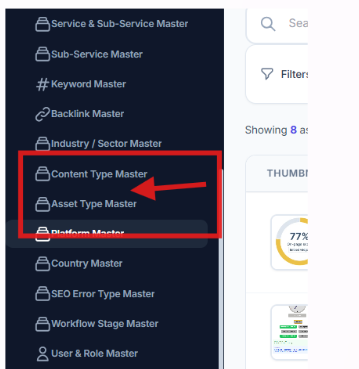


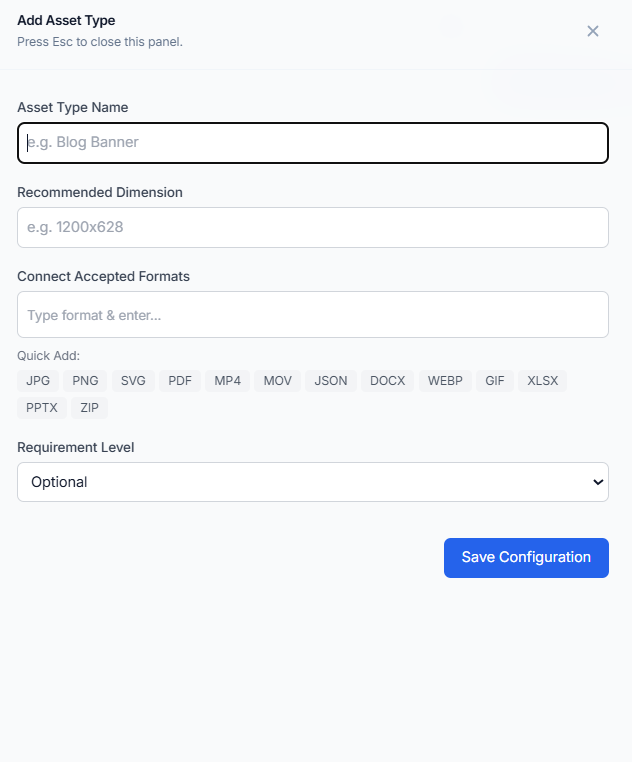
Step 6: The Asset Category should be linked to the Asset Master table.  
If an Asset Category is added in the Master table, it should appear in this list for the user to select.



Step 7: Create a MASTER TABLE for asset category

Add a new master table in above Asset Category



Step 8: Same as Asset type 🡪 create an Asset category like this:  
  


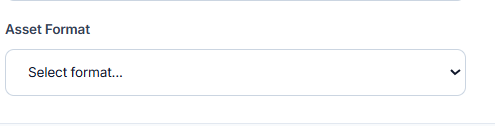
Only include the fields:  
  
1. Brand – Dropdown (Pubrica, Stats work, Food Research lab, PhD assistance, tutors India)

2. Asset Category Name   
  
3. word count

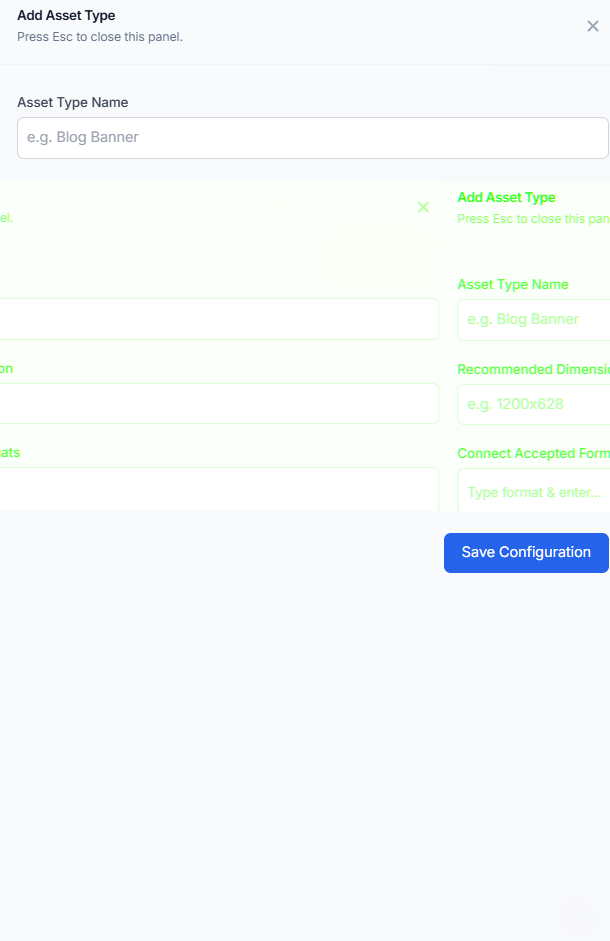
And with a submit button  
  
the user will submit the asset category in the master table, that will list in the Assets

Step 8:

The Asset format should link with master table



Link with Asset type field with the Asset type master:



23.12.25:

Corrections to be modified:   
  
For Web:

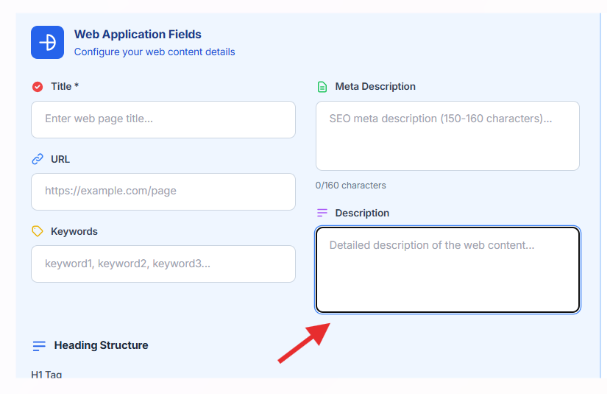
* 1. The Asset Category Master table should be connected to Asset Category. When the admin adds a category in the Asset Category Master table, it should automatically link to the Asset Category Field. The categories should show up in a dropdown list.
  2. The Asset Type Master table should be linked to Asset Type. When the Admin enters data in the Asset Type Master table, it should automatically link with the corresponding Asset Type. The Asset Type should then appear as a dropdown selection
  3. Remove Asset format fields
  4. Remove the Add icons near the Asset type, Asset Category,
  5. The Quality Score should right beside Body Content
  6. When clicking on 'Map Asset to Services,' the sub-services should be displayed below. The user will select the subservices
  7. File Upload should be below the Body content

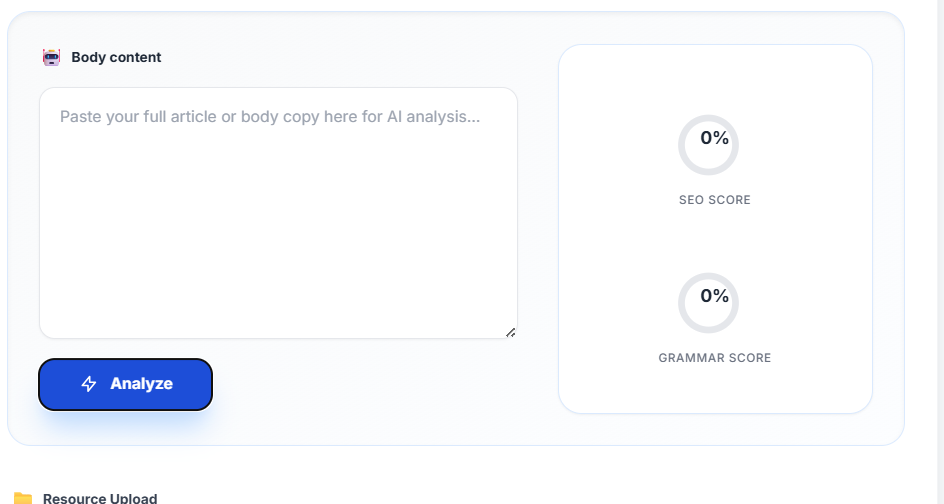
**For SMM:**

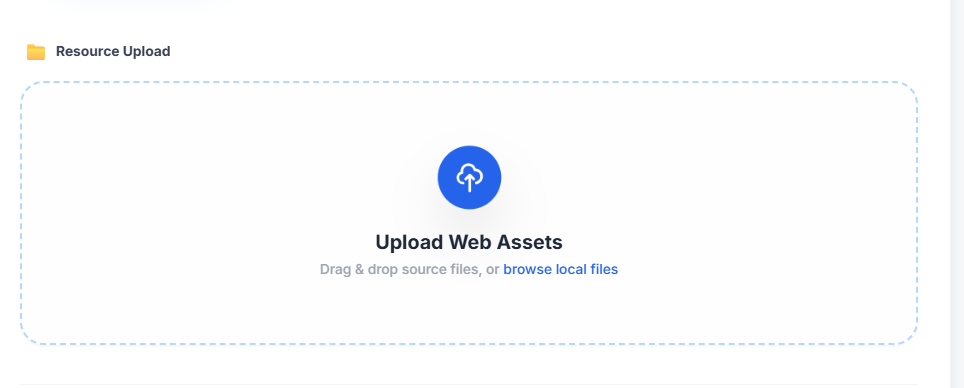
1. There should be only one upload option.
2. The upload option should include a preview, displayed on the right side.
3. Remove TikTok from the list of social media platforms.
4. Include Asset Classifications: Content Type, Repository dropdown, Asset Category (linked to the master table), Asset Format (linked to the master table), and Map Asset to Services with a dropdown selection for sub-services.
5. Remove the Add icons near the Asset type, Asset Category, Asset format

Corrections in the Screens: (FOR WEB)

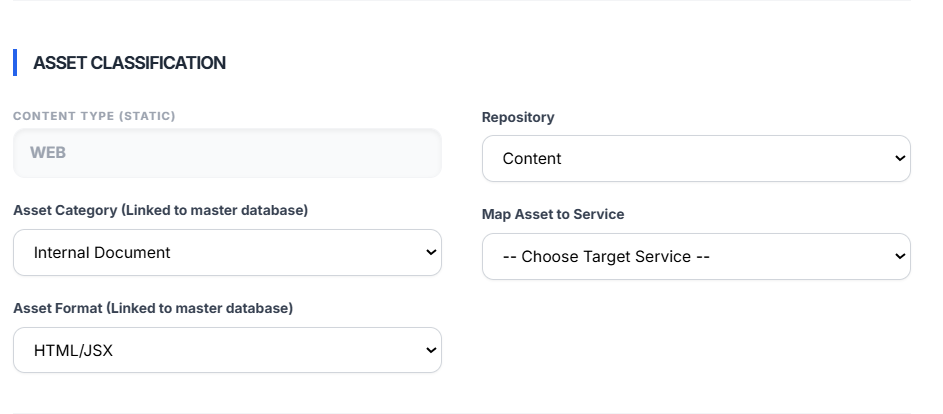
1. Remove this Description box:



1. Below the Web application fields 🡪 Add the body content and display the AI score next to it, with the 'Analyse Score' option shown below.  
   
2. Add an Upload option below the Body content:



1. Below the Upload Web Assets 🡪 Add the Asset classification



In the Above fields, The Asset Category Field should link with Asset Category Master Table, When the admin adds a category in the Asset Category Master table, it should automatically link to the Asset Category Field. The categories should show up in a dropdown list.

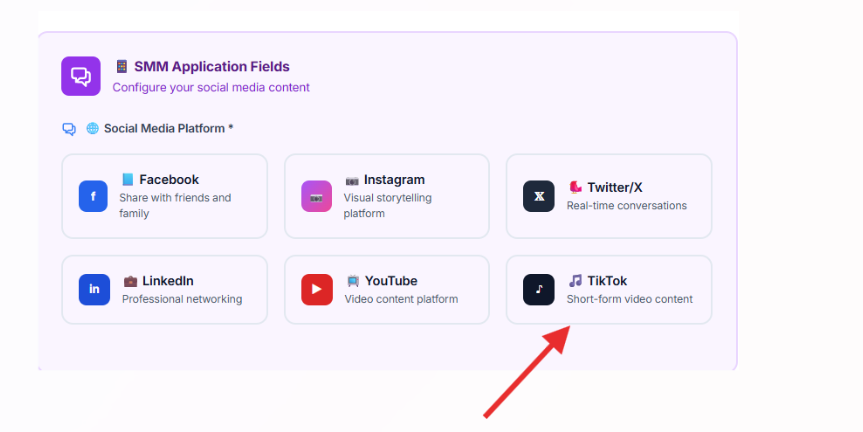
The Asset Type field should link with the Asset Type Master table, When the Admin enters data in the Asset Type Master table, it should automatically link with the corresponding Asset Type. The Asset Type should then appear as a dropdown selection

You can see the Master table of Asset type and Asset Category:

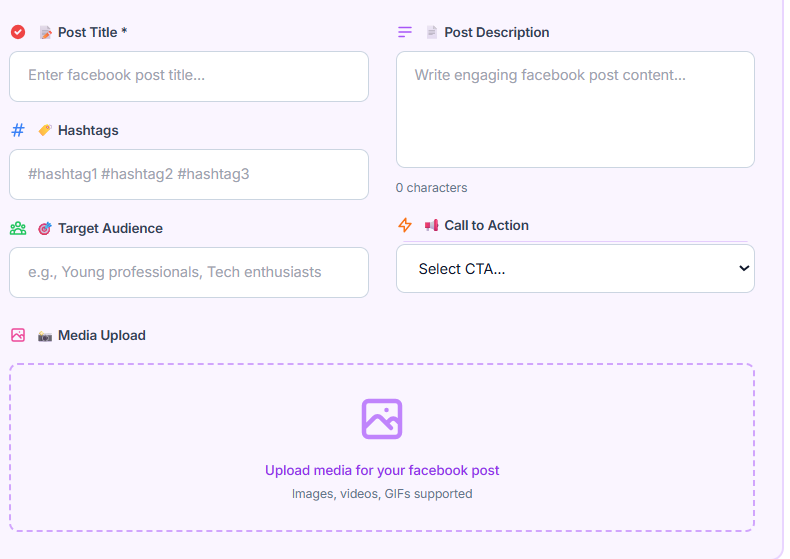


Corrections in the Screens: (FOR SMM)

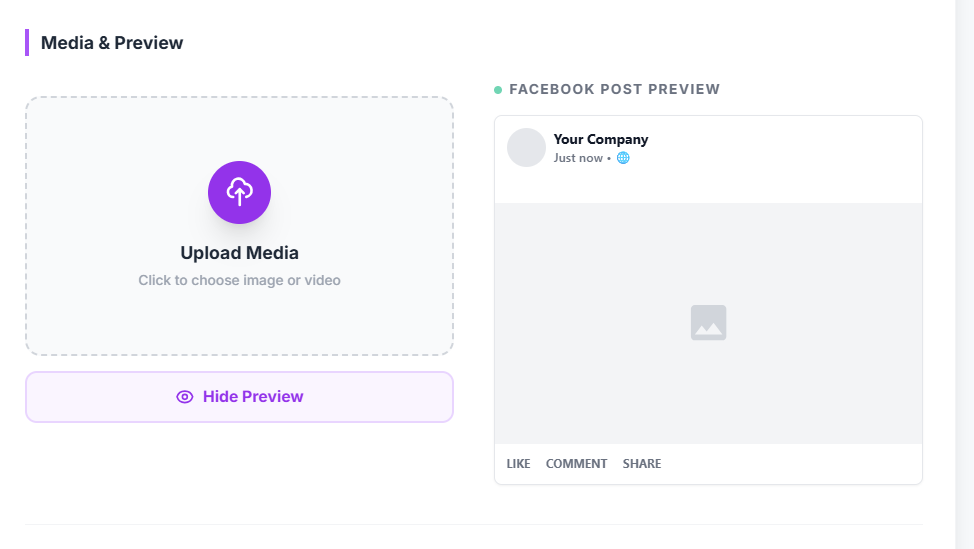
* 1. Remove Tik Tok



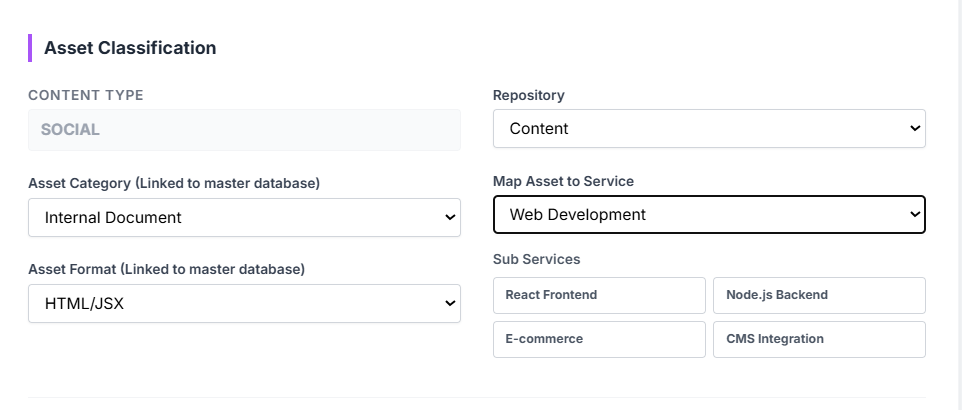
2.Based on Social media platform, the fields will display below



3.Below the Upload Option, Include the preview post option based on the social media platform



4. Below the Upload option, Include the Asset classification

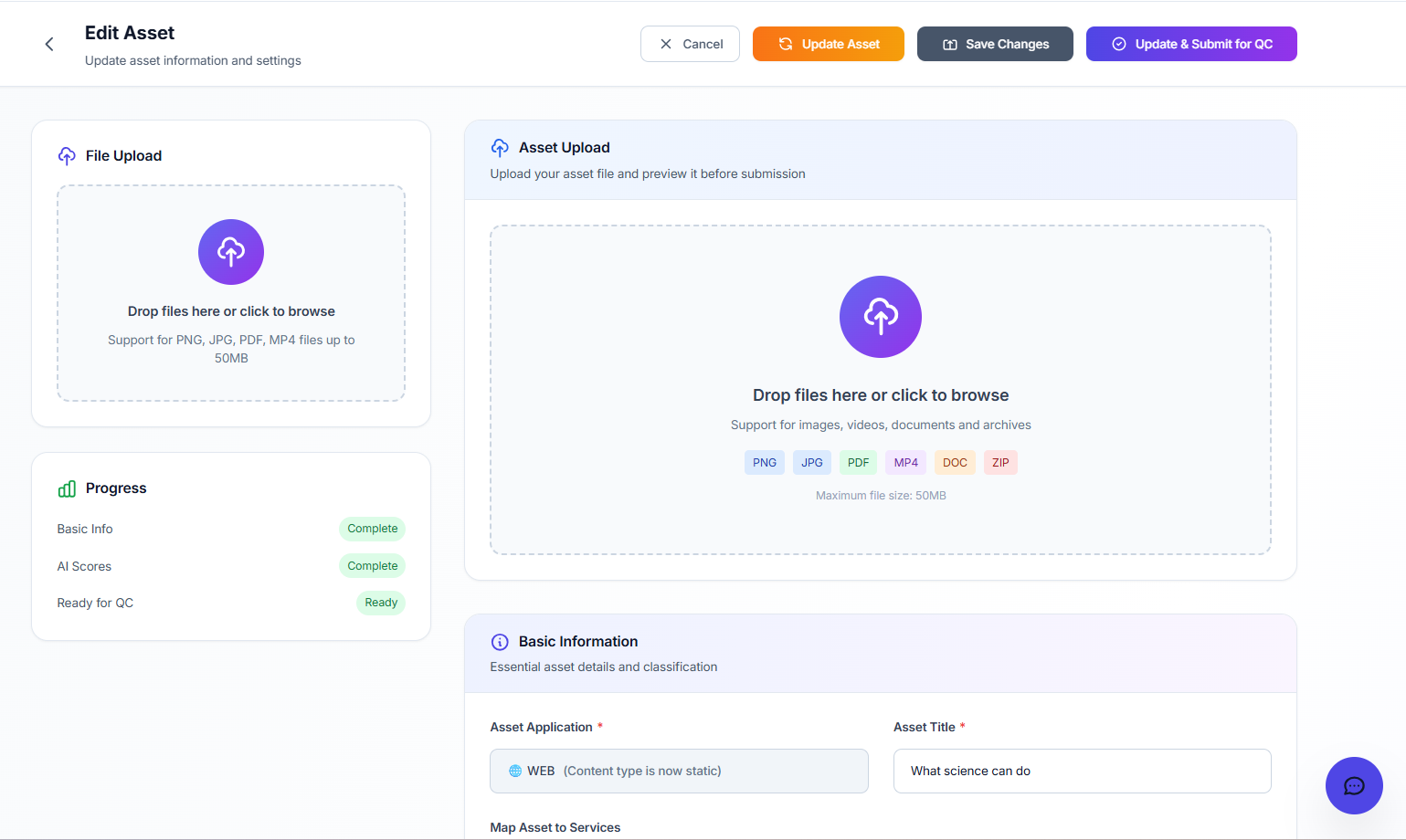


In the Above fields, The Asset Category Field should link with Asset Category Master Table, When the admin adds a category in the Asset Category Master table, it should automatically link to the Asset Category Field. The categories should show up in a dropdown list.

The Asset Type field should link with the Asset Type Master table, When the Admin enters data in the Asset Type Master table, it should automatically link with the corresponding Asset Type. The Asset Type should then appear as a dropdown selection

**ASSET CORRECTIONS:**

1.We can’t be able to Scroll the Edit Asset page:



**2.Introduce a new section under Upload Assets titled “Map Assets to Source Work” and integrate the required variables within this section.**

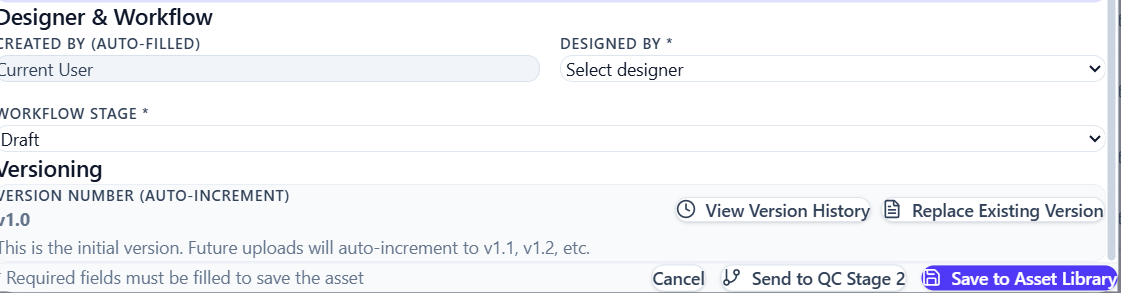
| **Field / Variable Name** | **Type** | **Description / Requirement** |
| --- | --- | --- |
| Linked Task | Dropdown / Lookup | Allows the user to select and link an existing task from the Task module. |
| Linked Campaign | Dropdown / Lookup | Displays all available campaigns for selection and linking. |
| Linked Project | Dropdown / Lookup | Lists all projects available in the Project module for linking. |
| Linked Service | Dropdown / Lookup | Shows all services from the Service Master for selection. |
| Linked Sub-Service | Dropdown / Lookup | Displays sub-services related to the selected service. |
| Linked Repository Item | Dropdown / Lookup | Allows selection of items from the Repository module to link with the record. |

SCREEN OUTCOME:   
  

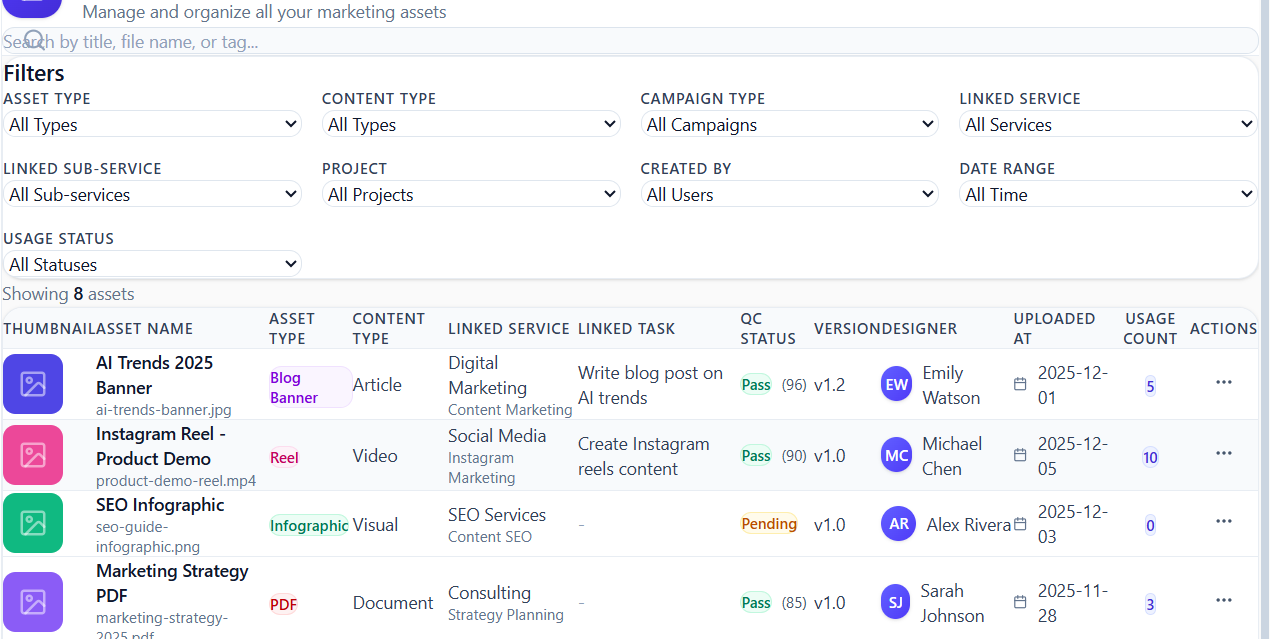

* Introduce a Another Section into upload Assets “**Designer & Workflow” and “Versioning” & integrate the below variables**
* Variables Integrate in the Upload Assets:

| **Field / Variable Name** | **Type** | **Description / Requirement** |
| --- | --- | --- |
| Created By | Auto populated / Dropdown | Automatically captures or allows selection of the user who created the record. |
| Designed By | Dropdown | Allows selection of the user responsible for designing the content or item. |
| Workflow Stage | Dropdown | Displays the current workflow status with the following options: **Draft**, **QC Pending**, **QC Completed**, **Published**. |

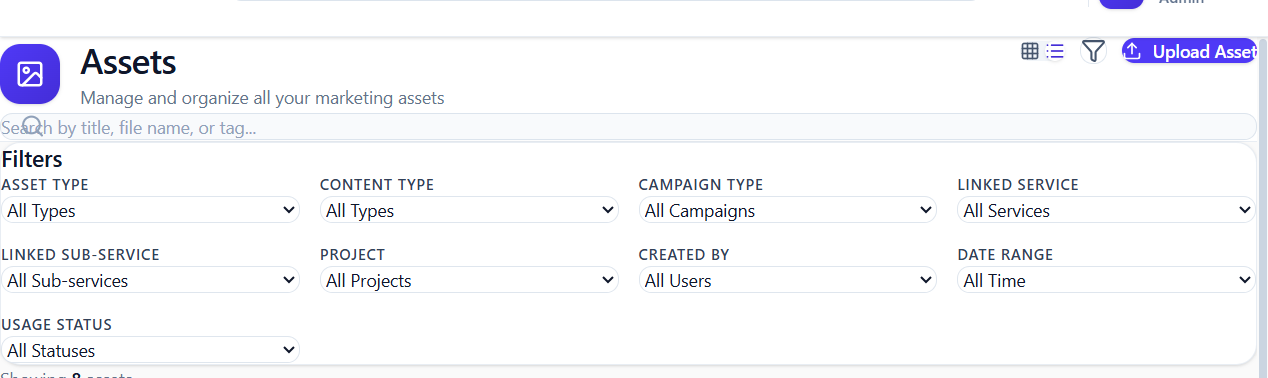
**SCREEN OUTCOME:**



3.The List view of Assets page should look like this:



**4.Filters to be implemented:**

  
 **Variables to Add:**

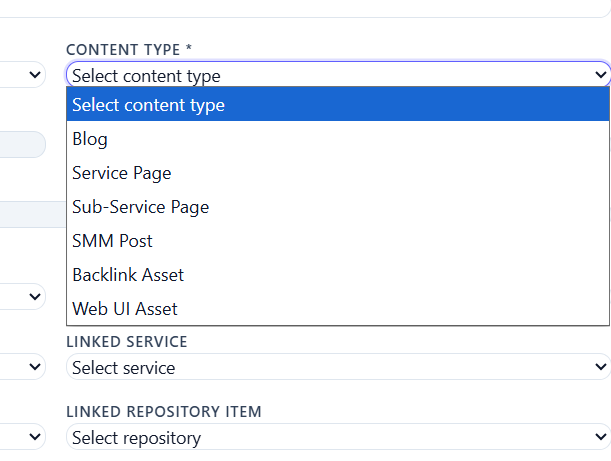
| **Field / Variable Name** | **Type** | **Description / Requirement** |
| --- | --- | --- |
| Asset Type | Dropdown | Allows selection of asset type. Default option: **All Types added in asset type master** |
| Asset Category | Dropdown | Lists all available Asset Category. Default option: **All Categories added in Asset Category master** |
| Campaign Type | Dropdown | Displays all campaign categories. Default option: **All Campaigns added in Campaigns** |
| Linked Service | Dropdown | Lists all services configured in the Service Master. Default option: **All Services** |
| Linked Sub-Service | Dropdown | Displays sub-services mapped to the selected service. Default option: **All Sub-Services**. |
| Project | Dropdown | Lists all projects available in the system. Default option: **All Projects**. |
| Created By | Dropdown | Allows filtering based on asset creator. Lists all system users. Default option: **All Users**. |
| Date Range | Date Picker | Enables filtering assets based on a selected date range. Default option: **All Time**. |
| Usage Status | Dropdown | Displays asset usage states (e.g., Used, Unused, Archived). Default option: **All Statuses**. |

25.12.25:

**In Upload Asset Page:**

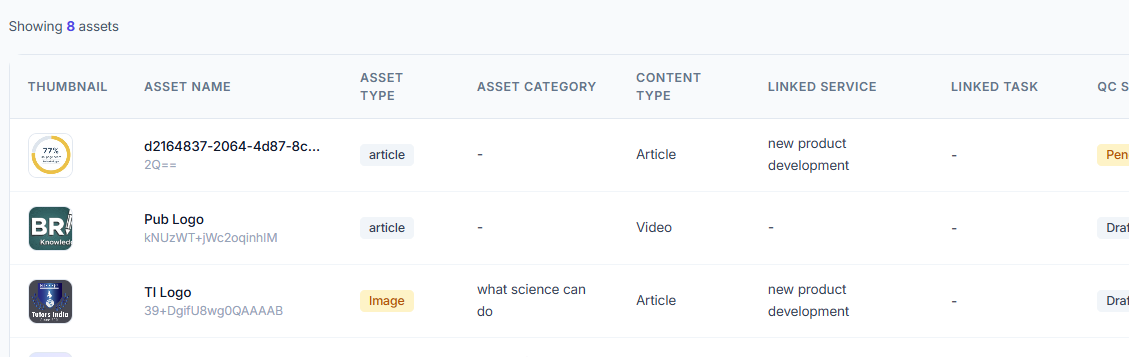
Add **Content Type** to the Asset Classification with the following dropdown options.

| **Field / Variable Name** | **Type** | **Description / Requirement** |
| --- | --- | --- |
| Content Type | Dropdown | A single dropdown field used to classify the content asset. Available options: **Blog, Service Page, Sub-Service Page, SMM Post, Backlink Asset, Web UI Asset**. |



**In The Asset List View:**

Based on the Content Type selected during asset upload; the same Content Type should be fetched and displayed in the List View. The List View must include a dedicated column for Content Type to reflect this information accurately.



**Asset Detail – Side View Panel Documentation – Connect with backend as well as Frontend**

When a user clicks on a particular Asset from the Assets list, the asset details should open in a **side view panel** (not a full page).  
The side view must be divided into the following **four sections**:

1. **Metadata**
2. **Mapping & Links**
3. **QC Panel**
4. **Usage Panel**

**1. Metadata Section**

**Purpose:**  
Displays all asset-level information entered **during asset upload**.  
These fields should also be available as **columns and filters in the Asset List View**.

| **Field / Variable Name** | **Type** | **Description / Requirement** |
| --- | --- | --- |
| Asset ID | Auto generated | Unique identifier for the asset. |
| Asset Type | Dropdown (read-only here) | Asset type selected during upload (e.g., Blog Banner). |
| Content Type | Dropdown (read-only here) | Content classification selected during upload (e.g., Article, Blog, SMM Post). |
| Dimensions | Text | Asset dimensions (e.g., 1920x1080).( Fetch from Asset type Master of the Asset) |
| Size | Text | File size (e.g., 2.4 MB). ) (Fetch from Asset type Master of the Asset) |
| Format | Text | File format (e.g., JPEG, PNG, MP4).) (Fetch from Asset type Master of the Asset) |
| Version | Text / Auto | Current asset version (e.g., v1.2). |
| Created By | User Reference | User who uploaded/created the asset. |
| Updated By | User Reference | User who last updated the asset. |
| Created Date | Date | Asset creation date. |
| Last Updated Date | Date | Last modified date of the asset. |

**2. Mapping & Links Section**

**Purpose:**  
Displays all **linkages selected while uploading the asset**.  
These mappings should also be available in the **List View and Filters**.

Data Source: Asset Upload → “Map Assets to Source Work” section

| **Field / Variable Name** | **Type** | **Description / Requirement** |
| --- | --- | --- |
| Linked Task | Reference / Text | Task linked during asset upload. |
| Linked Campaign | Reference / Dropdown | Campaign associated with the asset. |
| Linked Project | Reference / Dropdown | Project linked to the asset. |
| Linked Service | Reference / Dropdown | Service mapped during upload. |
| Linked Sub-Service | Reference / Dropdown | Sub-service mapped during upload. |
| Linked Repository Item | Reference | Repository item associated with the asset (if applicable). |

**Behaviour:**

* These values must **render in the asset side view**
* Must be **searchable and filterable in the Asset List View**

**3. QC Panel Section**

**Purpose:**  
Displays **QC results fetched from the Asset Review / QC workflow**.  
This section is **read-only for users** and populated after QC review.

Data Source: QC Review Module

| **Field / Variable Name** | **Type** | **Description / Requirement** |
| --- | --- | --- |
| QC Score | Numeric | Final QC score (e.g., 96 / 100). |
| QC Status | Status | QC result (Pass / Fail / Rework). |
| Reviewer | User Reference | Admin/QC reviewer who evaluated the asset. |
| QC Date | Date | Date when QC was completed. |

**QC Checklist & Scoring (Repeatable Items)**

| **Field / Variable Name** | **Type** | **Description / Requirement** |
| --- | --- | --- |
| Checklist Item Name | Text | QC parameter name (e.g., Image Resolution & Quality). |
| Reviewer Comment | Text | Reviewer feedback for the checklist item. |
| Score | Numeric | Score assigned for the item (e.g., 18/20). |
| Result | Status | Pass / Fail per checklist item. |

**4. Usage Panel Section**

**Purpose:**  
Displays **where and how the asset is used**, fetched dynamically from usage records.

Data Source: Usage Tracking / Manual Linking

**Website Usage**

| **Field / Variable Name** | **Type** | **Description / Requirement** |
| --- | --- | --- |
| Website URL | URL | Web pages where the asset is used. |

**Social Media Usage**

| **Field / Variable Name** | **Type** | **Description / Requirement** |
| --- | --- | --- |
| Platform Name | Text | Social platform (e.g., LinkedIn, Twitter). |
| Post Link | URL | Direct link to the post using the asset. |

**Backlink Submissions**

| **Field / Variable Name** | **Type** | **Description / Requirement** |
| --- | --- | --- |
| Domain Name | Text | Backlink submission website. |
| Approval Status | Status | Approved / Pending / Rejected. |

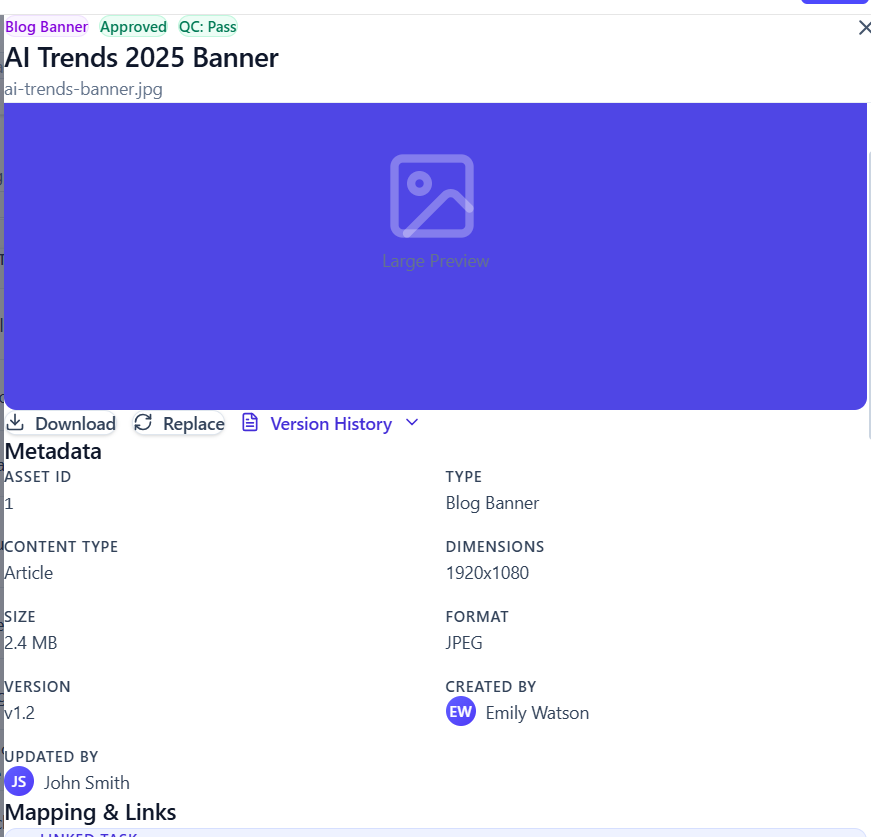
**Engagement Metrics**

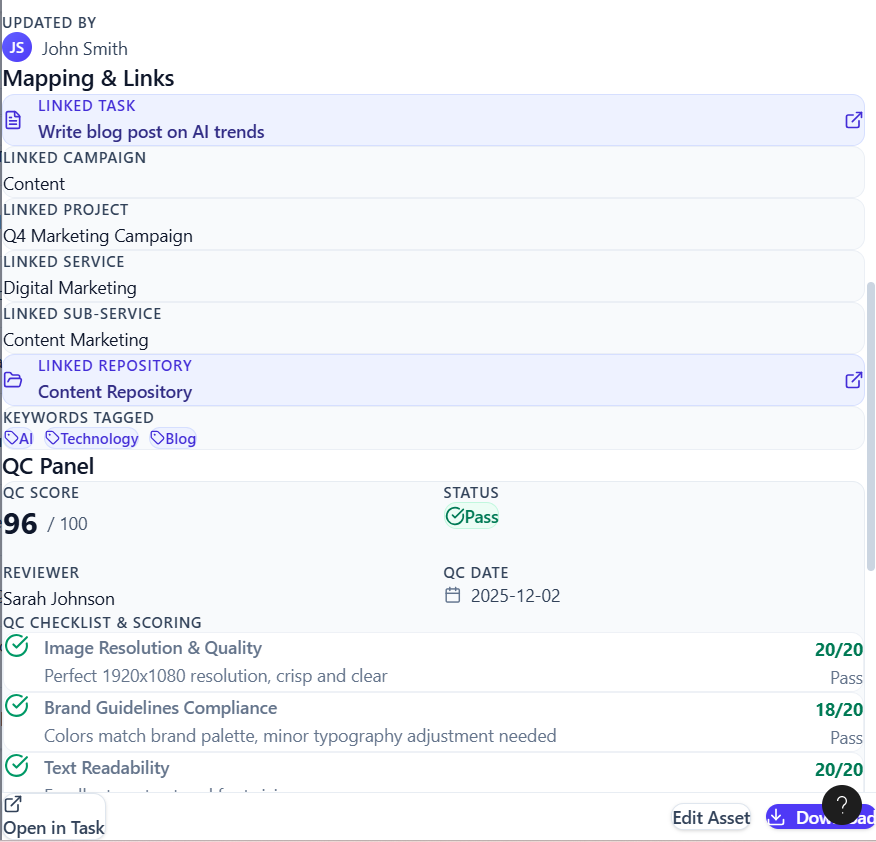
| **Field / Variable Name** | **Type** | **Description / Requirement** |
| --- | --- | --- |
| Impressions | Numeric | Total impressions count. |
| Clicks | Numeric | Total clicks. |
| CTR | Percentage | Click-through rate. |
| Shares | Numeric | Total social shares. |

**Summary for Developers**

* **Metadata & Mapping fields**  
  → Entered during asset upload  
  → Must appear in **side view + list view + filters**
* **QC Panel**  
  → Fetched from QC workflow  
  → Read-only for users
* **Usage Panel**  
  → Dynamically fetched from usage tracking  
  → Supports URLs, social posts, backlinks, and metrics
* **UI Requirement**  
  → Asset details must open in a **side view panel**, not a full page  
  → List View & Filters must match **Figma designs exactly**

SCREEN OUTCOME:



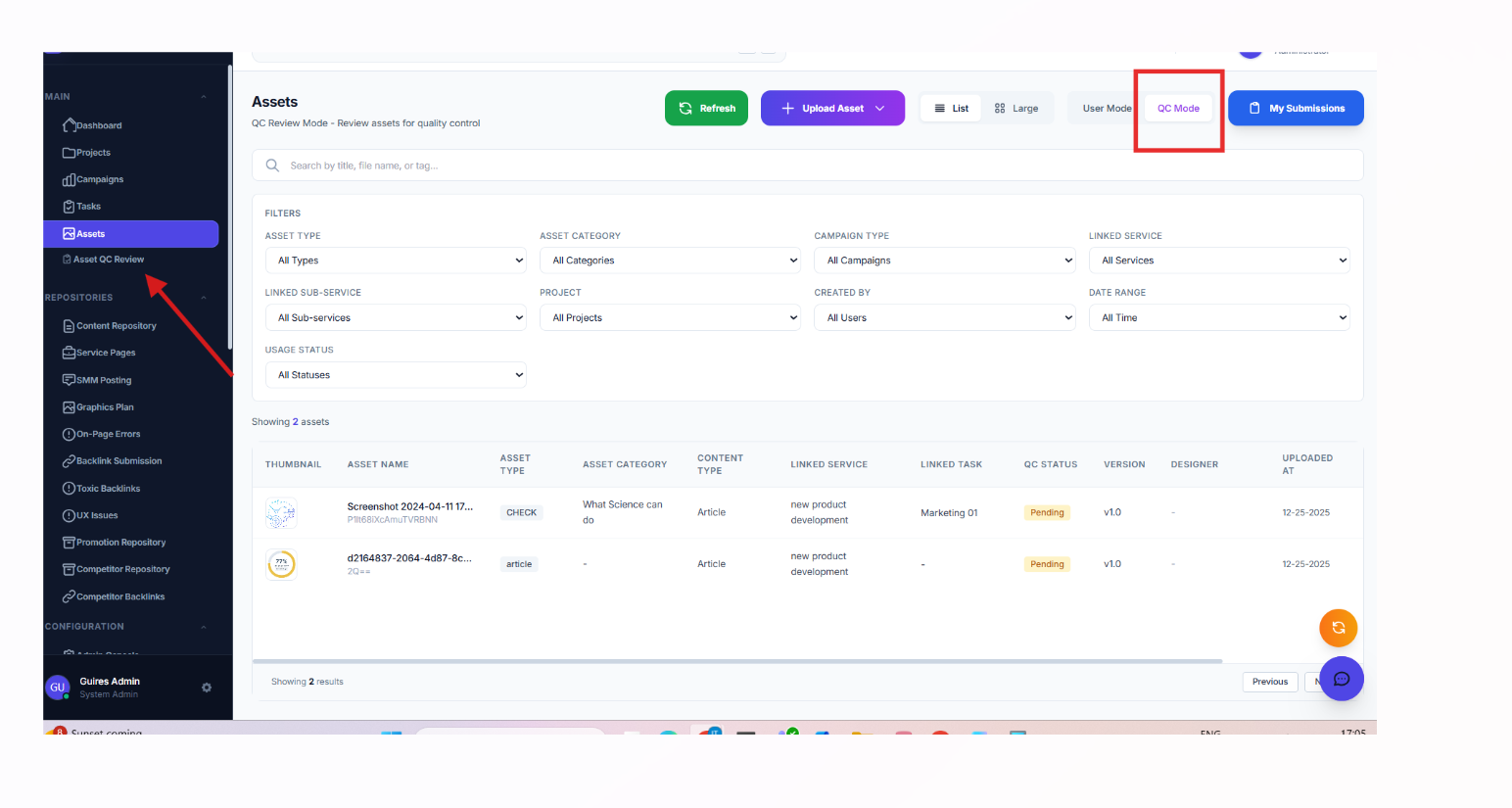




Provide role-based access for **User-level** and **Admin-level** logins for QC checks:

* **Admin Level:**  
  Admin users are responsible for performing QC checks. They will review the asset, assign a QC score through the QC panel, and submit the QC results. Once the asset passes QC, it should be automatically linked to the corresponding Service page.
* **User Level:**  
  If an asset is marked as **Rework** during QC, it should be clearly indicated as *Rework* in the user view. The user will then make the necessary updates and resubmit the asset for QC review.

We do not require a separate QC mode. Once an asset is submitted and send to Asset to QC stage 2 , it should automatically move to the **Asset QC Review** section in the sidebar

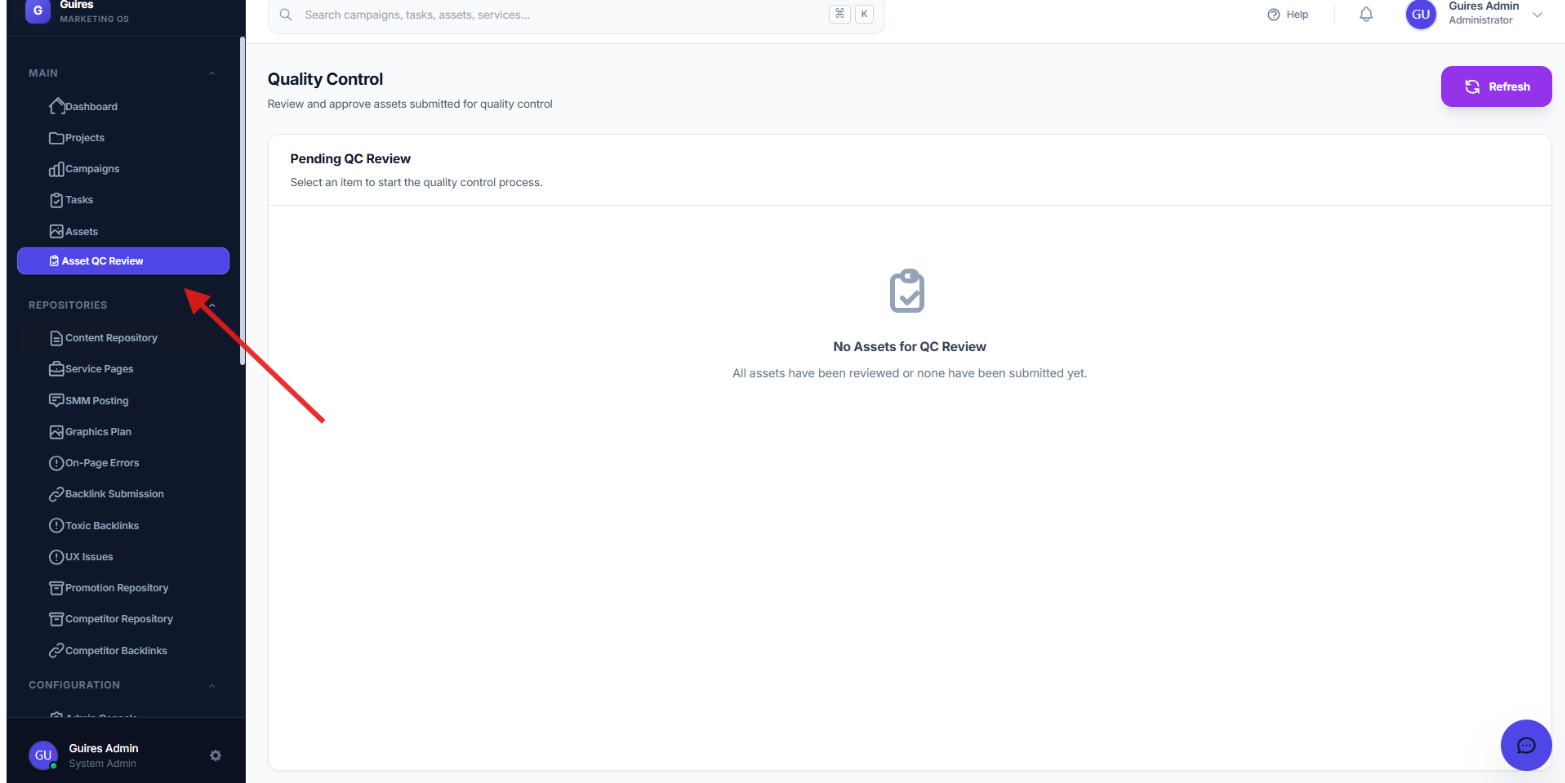


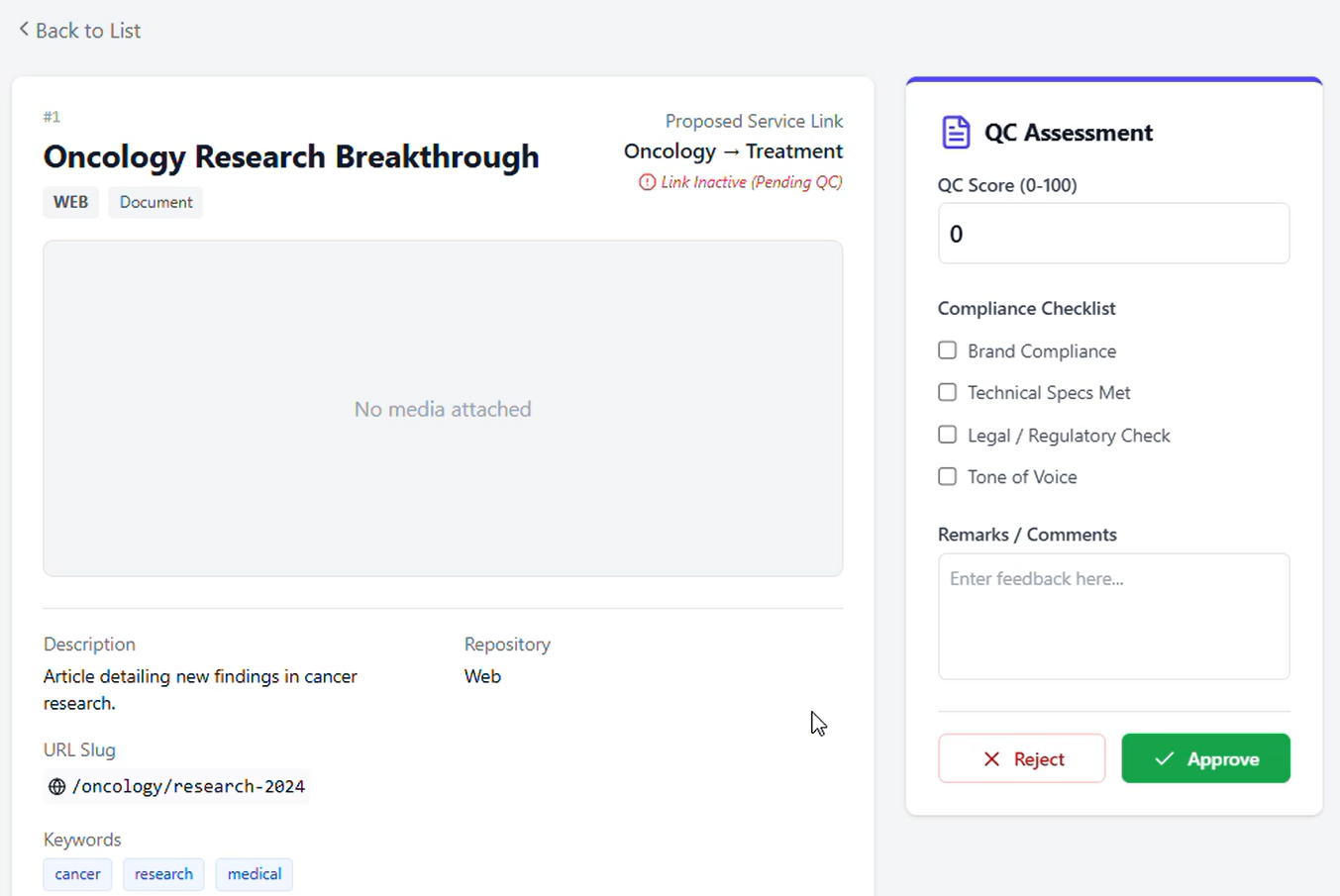
After it moved to the QC review:

**Asset List View – Field Definitions**

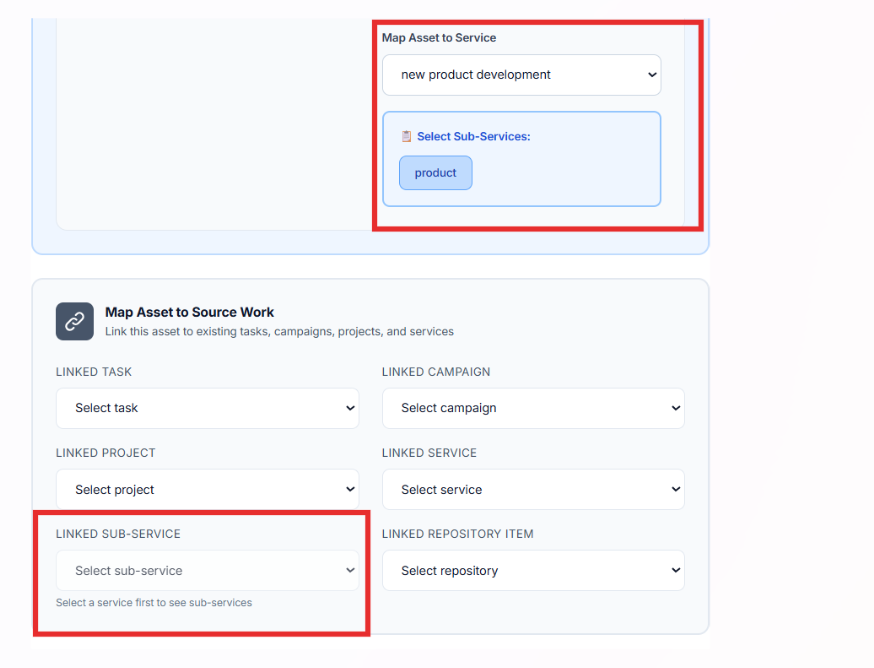
| **Field / Variable Name** | **Type** | **Description / Requirement** |
| --- | --- | --- |
| **Thumbnail** | **Image** | **Displays a small preview image of the asset. Should load from the stored asset thumbnail URL.** |
| **Asset Name** | **Text** | **Name of the asset entered during asset upload. Should be clickable to open the asset side-view panel.** |
| **Asset Type** | **Text / Label** | **Displays the asset type (e.g., Banner, Video, Document) selected during upload.** |
| **Asset Category** | **Text / Label** | **Shows the asset category mapped in the Asset Category Master.** |
| **Content Type** | **Text / Label** | **Displays the selected content type (e.g., Blog, Service Page, SMM Post) chosen during asset upload.** |
| **Linked Service** | **Text** | **Shows the service mapped to the asset under “Map Assets to Source Work.”** |
| **Linked Task** | **Text** | **Displays the task linked to the asset (if mapped during upload).** |
| **QC Status** | **Status Label** | **Indicates the current QC state (e.g., Pending, Pass, Fail, Rework). Should update dynamically based on QC actions.** |
| **Version** | **Text** | **Displays the current version of the asset (e.g., v1.0, v1.2).** |
| **Designer** | **Text / User Name** | **Shows the designer or creator assigned during asset upload or workflow configuration.** |
| **Uploaded At** | **Date & Time** | **Displays the asset upload date and time. Should follow system date-time format.** |
| **Usage Count** | **Numeric** | **Indicates the number of times the asset is used or referenced across services, pages, or campaigns.** |
| **Review Action** | **Action Button** | **Provides actions such as View, Send to QC, Rework, or Approve, based on user role and asset status.** |

**SCREEN VIEW:**





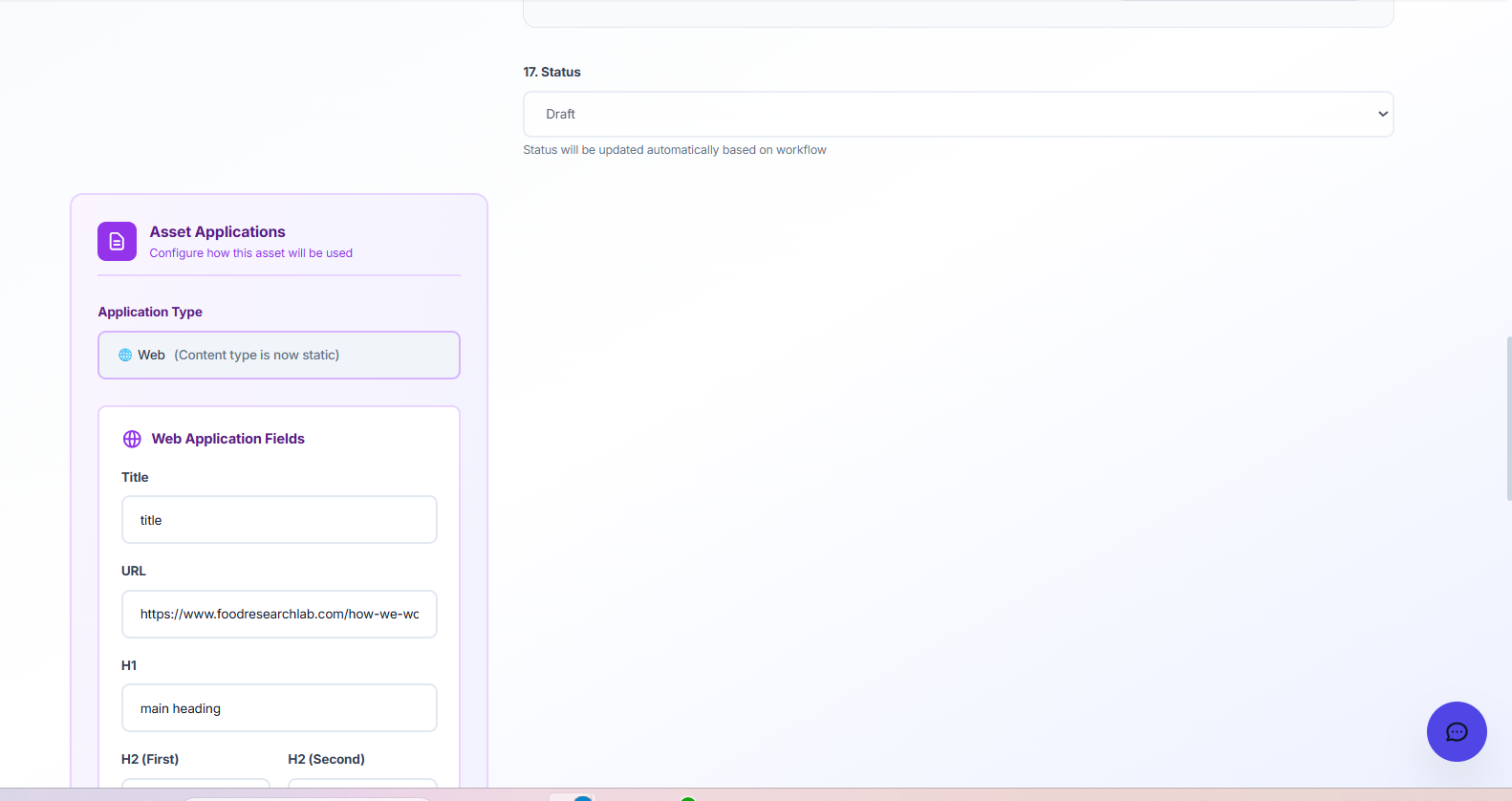
**Corrections:**

* 1. Since the Linked Sub-Service is already available under Map Asset to Source Work, please remove the **Map Asset to Service option**.
  2. In the **Edit** option, the fields are currently unorganized. Only the fields that were originally entered for the selected asset type should be editable.

For example:

* If the asset is a **Web Application** asset, only the **Web Application–related fields** should be editable.
* If the asset is an **SMM** asset, only the **SMM-related fields** should be editable.

**Fix this in the edit option of Asset View:**

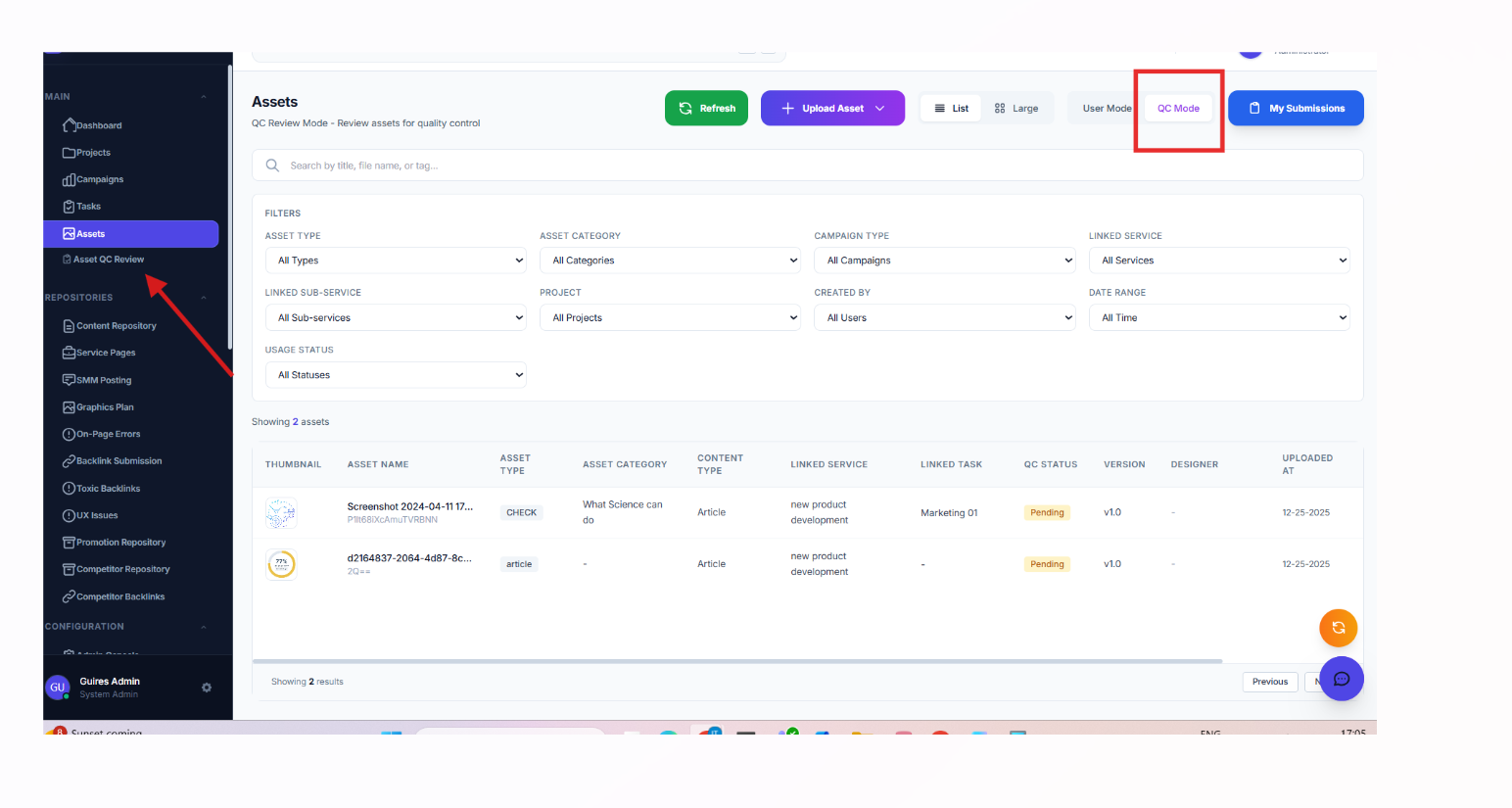


26.12.25:

Provide role-based access for **User-level** and **Admin-level** logins for QC checks:

* **Admin Level:**  
  Admin users are responsible for performing QC checks. They will review the asset, assign a QC score through the QC panel, and submit the QC results. Once the asset passes QC, it should be automatically linked to the corresponding Service page.
* **User Level:**  
  If an asset is marked as **Rework** during QC, it should be clearly indicated as *Rework* in the user view. The user will then make the necessary updates and resubmit the asset for QC review.

We do not require a separate QC mode. Once an asset is submitted and send to Asset to QC stage 2, it should automatically move to the **Asset QC Review** section in the sidebar

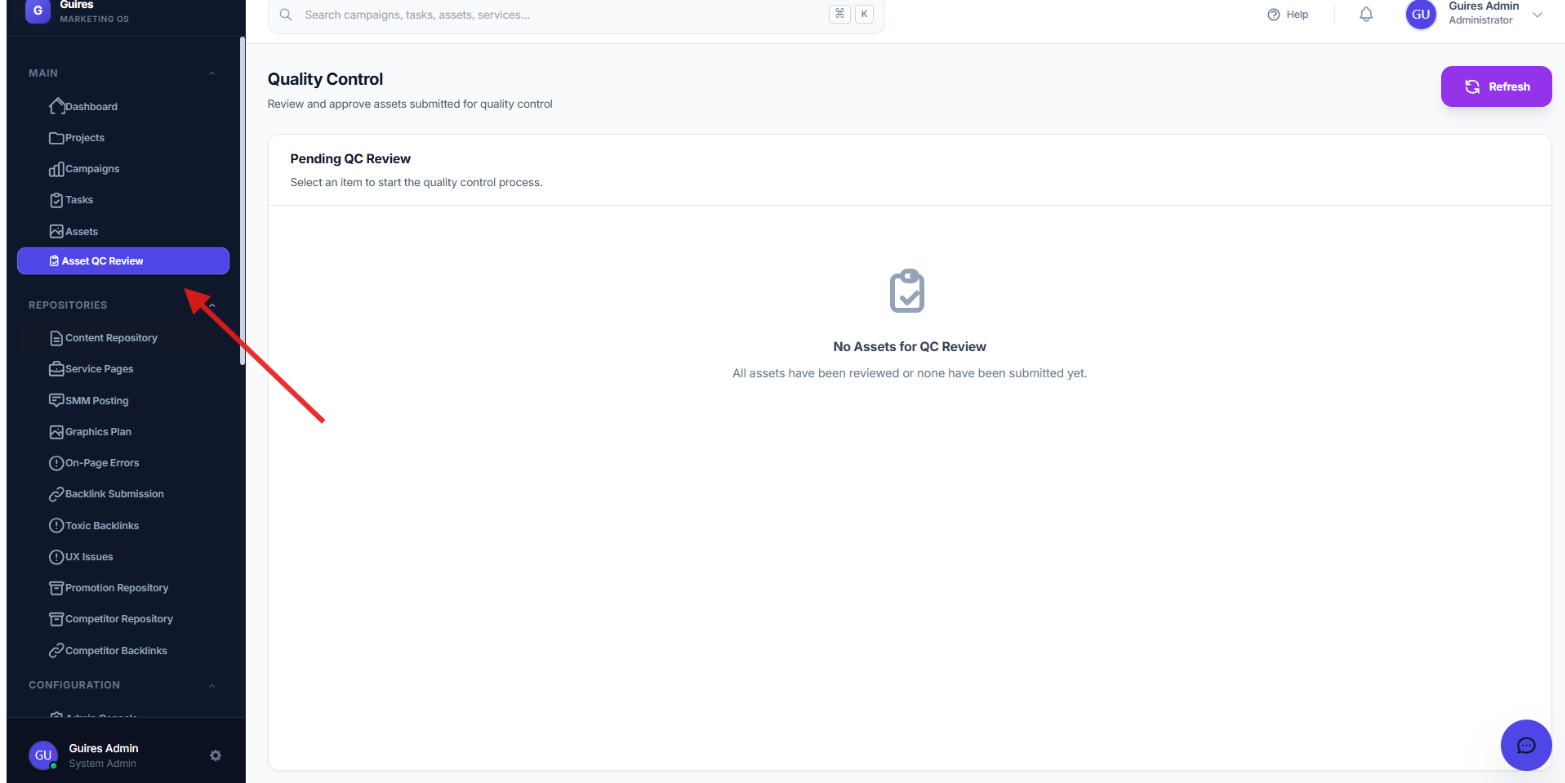


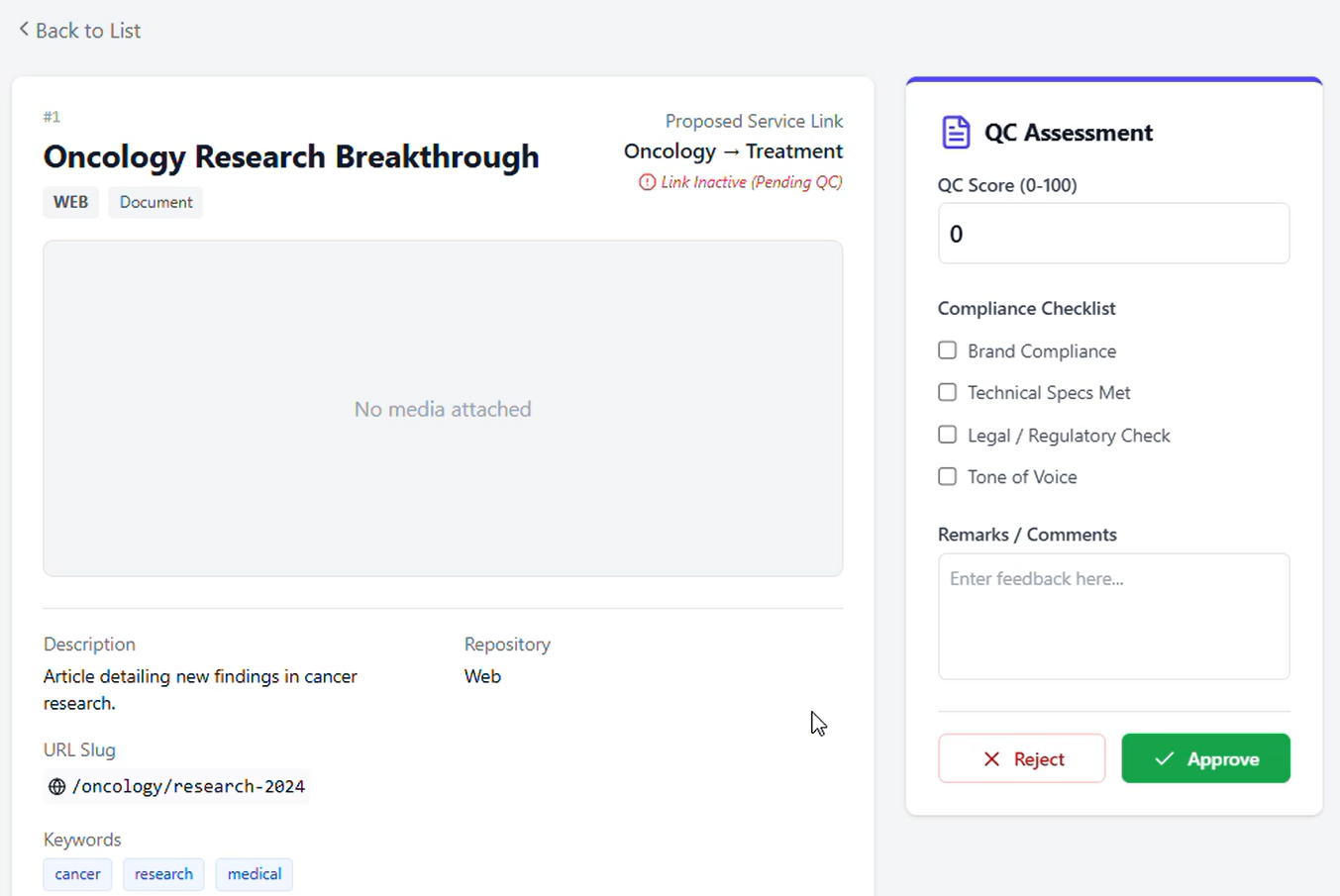
After it moved to the QC review:

**Asset List View In QC Review Field Definitions:**

| **Field / Variable Name** | **Type** | **Description / Requirement** |
| --- | --- | --- |
| **Thumbnail** | **Image** | **Displays a small preview image of the asset. Should load from the stored asset thumbnail URL.** |
| **Asset Name** | **Text** | **Name of the asset entered during asset upload. Should be clickable to open the asset side-view panel.** |
| **Asset Type** | **Text / Label** | **Displays the asset type (e.g., Banner, Video, Document) selected during upload.** |
| **Asset Category** | **Text / Label** | **Shows the asset category mapped in the Asset Category Master.** |
| **Content Type** | **Text / Label** | **Displays the selected content type (e.g., Blog, Service Page, SMM Post) chosen during asset upload.** |
| **Linked Service** | **Text** | **Shows the service mapped to the asset under “Map Assets to Source Work.”** |
| **Linked Task** | **Text** | **Displays the task linked to the asset (if mapped during upload).** |
| **QC Status** | **Status Label** | **Indicates the current QC state (e.g., Pending, Pass, Fail, Rework). Should update dynamically based on QC actions.** |
| **Version** | **Text** | **Displays the current version of the asset (e.g., v1.0, v1.2).** |
| **Designer** | **Text / Username** | **Shows the designer or creator assigned during asset upload or workflow configuration.** |
| **Uploaded At** | **Date & Time** | **Displays the asset upload date and time. Should follow system date-time format.** |
| **Usage Count** | **Numeric** | **Indicates the number of times the asset is used or referenced across services, pages, or campaigns.** |
| **Review Action** | **Action Button** | **Provides actions such as View, Send to QC, Rework, or Approve, based on user role and asset status.** |

**SCREEN VIEW:**





29.12.25:

**Admin Console – Employee Management**

**1. Overview**

This document defines the functional and UI requirements for the **Admin Console** within the Admin account. The objective is to enable administrators to manage employee accounts securely and efficiently.

The Admin Console will allow:

* Creation of new employee accounts
* Editing existing employee accounts
* Resetting employee passwords
* Activating or deactivating employee accounts
* Viewing high-level system and user summaries

**2. User Roles**

**2.1 Admin**

* Full access to the Admin Console
* Can add, edit, reset passwords, and deactivate employee accounts
* Can view user summaries and system status

**2.2 User**

* Standard employee account
* Cannot access the Admin Console
* Can log in only if the account status is **Active**

**3. Admin Console Access**

* The **Admin Console** should be visible only to users with **Admin** access role
* Non-admin users must not be able to view or access this module

**4. Dashboard Summary (Top Section)**

The Admin Console landing page should display a summary section at the top with the following metrics:

* **Total Users**: 3  
  Total number of registered employee accounts
* **Registered Accounts**: 3  
  Total accounts created in the system
* **Active Accounts**: 3  
  Number of accounts currently active and allowed to log in
* **Inactive Accounts**: 0  
  Number of accounts that are deactivated
* **System Health**: Optimal  
  Static or dynamic indicator showing overall system status

**5. Employee Management Section**

**5.1 Add New Employee**

**UI Requirement**

* A clearly visible **“Add New Employee”** button
* On click, open a form (modal or page)

**Fields Required**

* **Full Name** (Text, Mandatory)
* **Email** (Email format validation, Mandatory, Unique)
* **Password** (Secure input, Mandatory)
* **Access Role** (Dropdown: Admin / User)
* **Submit Button**

**behaviour**

* On successful submission:
  + Employee account is created
  + Account status defaults to **Active**
  + Employee can log in immediately

**5.2 Edit Employee Account**

**Functionality**

* Admin should be able to edit:
  + Full Name
  + Email
  + Access Role (Admin/User)

**Restrictions**

* Email uniqueness must be enforced
* Changes should reflect immediately after saving

**5.3 Reset Password**

**Requirements**

* Admin should have a **Reset Password** option for every employee
* Reset can be triggered directly from the employee list or profile

**Behaviour**

* On reset:
  + A new password is generated OR
  + Admin can manually set a new password
* The updated password should be effective immediately

**5.4 Deactivate Employee Account**

**UI Requirement**

* A **Deactivate Account** option for each employee

**Behaviour**

* Once deactivated:
  + User should NOT be able to log in
  + Login attempt should show the message:  
    **“User deactivated”**
* Account remains in the system but marked as **Inactive**

**6. Employee Status Handling**

**Status Values**

* **Active**: User can log in
* **Inactive (Deactivated)**: User cannot log in

**Login Validation**

* If account status = Inactive → block login and display error message

**7. Data & Validation Rules**

* Email must be unique across all accounts
* Mandatory field validation for all required inputs
* Secure handling of passwords (hashed & encrypted)

**8. Non-Functional Requirements**

* Role-based access control (RBAC)
* Secure authentication and authorization
* Responsive UI for desktop and tablet
* Audit-friendly structure (future enhancement)

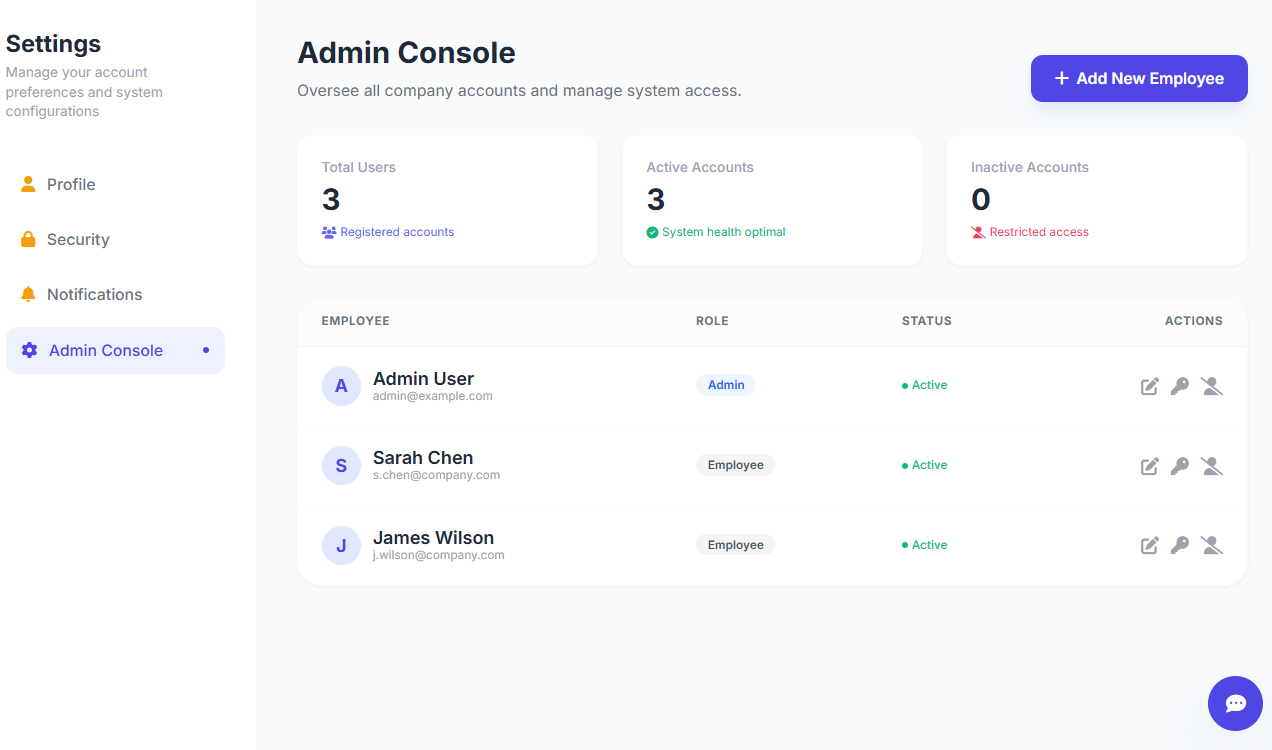
**9. Future Enhancements (Optional)**

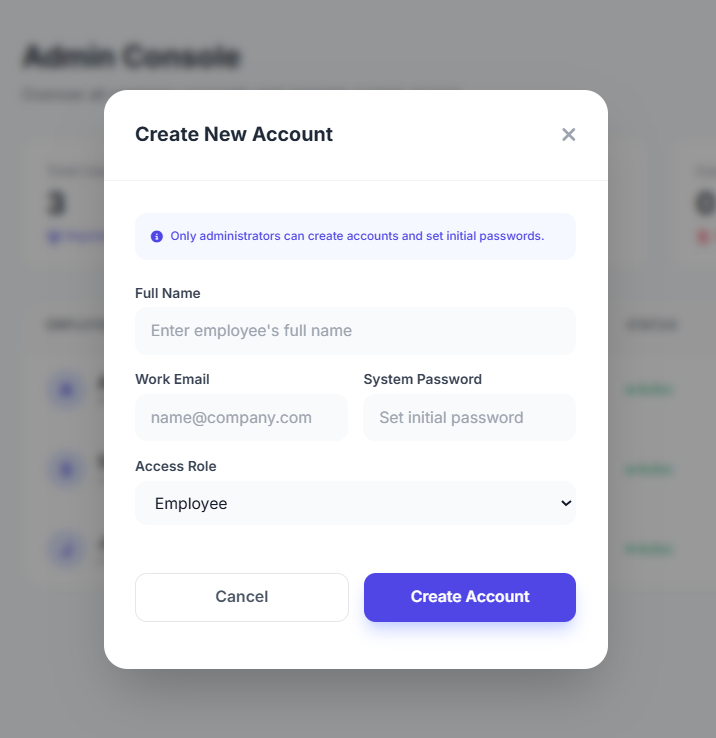
* Search and filter employees
* Bulk user activation/deactivation
* Activity logs for admin actions
* Email notification on password reset

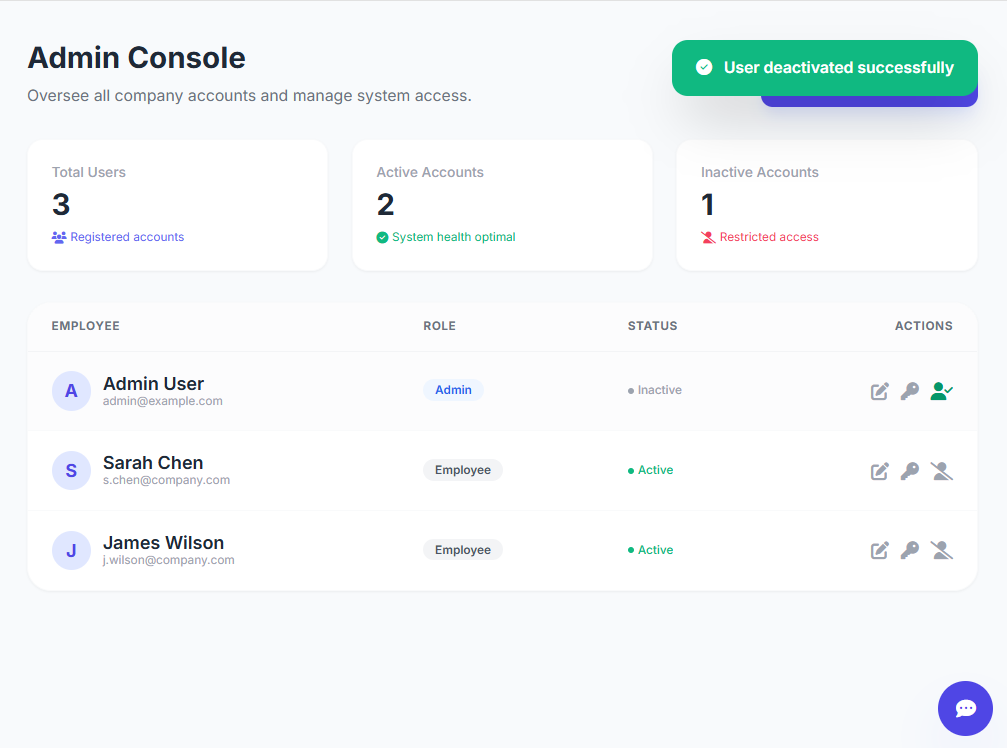
**10. Acceptance Criteria**

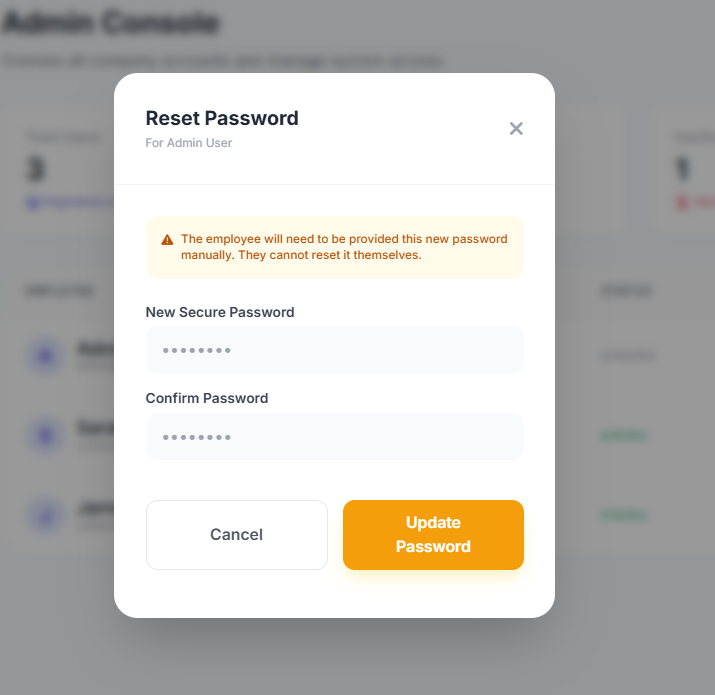
* Admin can successfully create, edit, reset, and deactivate employee accounts
* Deactivated users cannot log in and see correct error message
* Summary metrics update correctly based on user status
* Only Admin users can access the Admin Console

SCREEN OUTCOME:









30.12.25:

**Role & Permission Matrix – Detailed Requirements**

**3.1 Admin QC Asset Review Visibility**

* **Admin QC Asset Review screen must be visible ONLY to Admin accounts**
* This screen should not be accessible or visible to any other role (Employee, Manager, QC, etc.)
* Permission should be strictly enforced at both UI and API levels

**3.2 Asset QC Review Workflow for Employees**

**Asset Interaction Behaviour:**

* In the Assets module, employees should be able to click on an asset
* On click, a **side-opening panel (drawer view)** should appear

**Side Panel Should Display:**

* Asset details and metadata
* QC panel with checklist (read-only or actionable based on role)
* QC status indicators

**QC Actions:**

* Employees can submit assets for QC review
* QC Pass option should be enabled only if permitted by role
* Final QC approval remains under Admin or authorized QC role

**4. Security & Compliance Notes**

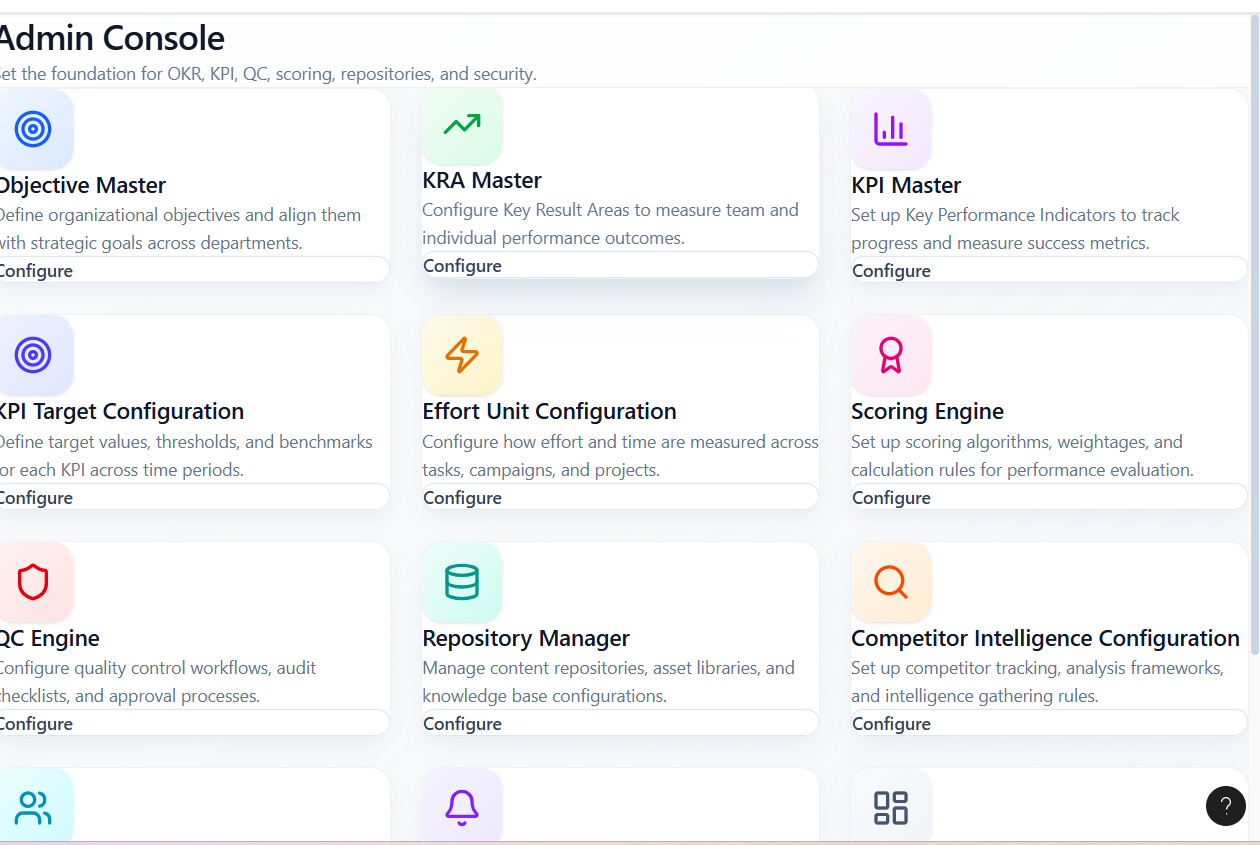
* All permission rules must be role-based and non-overridable by end users
* QC actions must be logged with timestamp and user identity
* Admin-only screens should not be accessible via direct URL

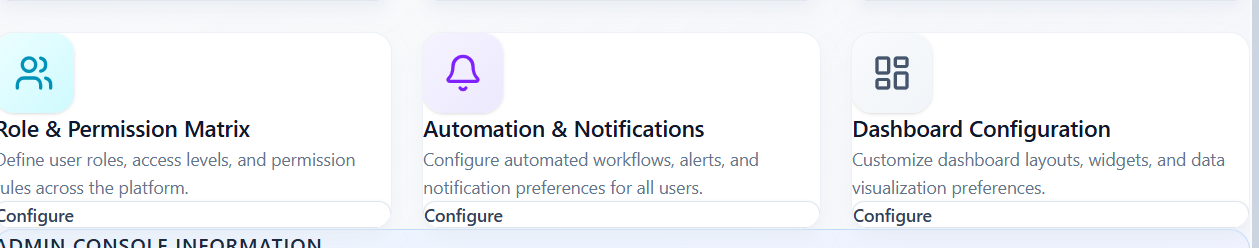
**5. Outcome**

This configuration ensures:

* Centralized administrative control
* Clear separation of duties
* Secure and auditable QC processes
* Scalable configuration management across brands and departments

**SCREENS OUTCOME:**





31.12.25:

**Roles, Permissions & Admin Console Access Control Specification**

**1. Purpose of This Document**

This document clearly defines **how user registration, roles, permissions, and admin access must work** in the system. It is written to remove ambiguity and ensure developers implement **strict access control, single-admin governance, and role-based login views**.

This specification must be followed exactly.

**2. Core Access Rule (Mandatory)**

**2.1 Admin-Only Registration Rule**

* No user account can access the software unless it is registered by an Admin account.
* Self-registration, open sign-up, or direct database entry is **strictly prohibited**.
* Every user must be created via **Admin Console → Add New Employee**.

**Exception – Main Panel Registration (Controlled)**

* Users are allowed to **initiate account creation from the Main Panel** only for registration requests.
* Main Panel registration must **not activate or grant access** to the system automatically.
* All Main Panel registrations must remain in a **Pending state** until reviewed and approved by an **Admin account**.
* Only after Admin approval and role assignment (Admin / Employee) can the user log in.

Final access control and activation authority always remains with the **Admin**.

**3. Employee Registration & Login Behaviour**

**3.1 Employee Creation Flow**

1. Admin logs into **Admin Console**
2. Admin selects **Add New Employee**
3. Admin assigns:
   * Employee details
   * Role (Admin / Employee)
   * Access status (Active / Inactive)
4. Account is saved and registered in the system

**3.2 Role-Based Login View (Critical Requirement)**

* **After registration only**, users are allowed to log in

Login behavior must follow this logic:

* If a user is registered with **Admin access**:
  + They must log in **only in Admin view**
  + Admin Console and configuration access are visible
* If a user is registered with **User / Employee access**:
  + They must log in **only in User view**
  + Admin Console and configurations must be completely hidden

Users must never be able to switch views manually.

**Role & Permission Matrix – Scope Restriction**

**3.3 Removal of Admin Asset QC Review from Configuration**

* **Admin Asset QC Review must NOT appear inside Configurations**
* Asset QC actions must be handled only through operational modules, not configuration settings

**4. Role & Permission Matrix – Allowed Content Only**

Inside **Role & Permission Matrix**, only the following **Admin Console view** must exist:

**4.1 Admin Console Dashboard**

**Purpose:** Oversee all company accounts and manage system access

**Metrics Displayed:**

* Total Users
* Registered Accounts
* Active Accounts
* Inactive Accounts

**4.2 Employee Management Table**

Columns must include:

* Employee Name
* Email ID
* Role
* Status
* Actions

**Sample Representation:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Employee** | **Email** | **Role** | **Status** | **Actions** |
| Admin User | admin@example.com | Admin | Active | — |
| Test User | user@example.com | Employee | Active | — |

**4.3 Security & Enforcement Rules (Non-Negotiable)**

* Admin Console access must be validated at **API and UI level**
* Role permissions must be enforced server-side
* Direct URL access to Admin Console by non-admin users must be blocked
* All user creation, edits, and status changes must be audit-logged

**4.4 Final Outcome**

By implementing this specification:

* No unauthorized access is possible
* Admin maintains full control over system entry
* Role-based views are strictly enforced
* Configuration remains centralized and secure

**Modules to be removed:**The following modules must be **removed from the Configurations section**:

1. Admin Console Configuration
2. Role & Permission Matrix
3. Admin QC Asset Review

The **Configurations** section should contain **the Admin Console** module.

**Role & Permission Matrix**

Within the **Admin Console → Roles & Permissions**, the admin console dashboard for add new employees with access levels

**7.1 Admin Console Dashboard**

**Purpose:**  
To monitor all company accounts and manage system access centrally.

**Metrics to be displayed:**

* Total Users
* Registered Accounts
* Active Accounts
* Inactive Accounts

**7.2 Employee Management Table**

**Mandatory Columns:**

* Employee Name
* Email ID
* Role
* Status
* Actions

**Sample Layout:**

| **Employee Name** | **Email ID** | **Role** | **Status** | **Actions** |
| --- | --- | --- | --- | --- |
| Admin User | admin@example.com | Admin | Active | — |
| Test User | user@example.com | Employee | Active | — |

02.02.26:

Assets List View:

The **Upload Asset page should be fully viewable in a proper List View**.  
Currently, only a **partial list is visible**, and users must **scroll to see additional records**, which makes it difficult to view all assets at once.

**Requirement:**

* Display the **entire asset list in a structured list/table view**.
* Ensure the list **loads and renders all records correctly** within the page layout.
* Avoid a design where only one section or limited rows are visible and the rest require excessive scrolling.
* The list view should be **consistent, readable, and aligned**, similar to standard data table behaviour (with pagination or proper scrolling if required).

Asset Side panel – QC Review:

When an **asset is clicked**, a **sidebar panel** displaying the **asset details**.  
Within this sidebar, please ensure the following are shown in the QC panel:

* The **QC checklist**
* The **score assigned by the Admin** in the **Asset QC Review module**

These details should be **fetched from the QC Review module** and **reflected in the asset details sidebar** for clear visibility and reference.

03.01.26:

This document defines the **functional, UI, and workflow requirements** for the **Upload Assets → Web** module. The objective is to ensure correct ordering of sections, clear system behaviour, and unambiguous implementation for development.

**2. Scope**

This document covers:

* Asset upload flow for **Web assets**
* Repository and task mapping
* Asset classification
* Content & SEO structure
* Editor capabilities
* AI integrations
* Resource uploads
* Workflow, QC, and versioning

The sections should be arranged in the following order:

1. **Map Assets to Source Work**
2. **Asset Classification**
3. **Content Details** – Title, Meta Description, Keywords, Headings, and Body Content
4. **Resource Upload**
5. **Designer & Workflow**
6. **Versioning**

**3. Module Flow Order (Mandatory)**

The following sections **must appear in the exact order** listed below when the user clicks:

**Upload Assets → Web**

**4. Functional Requirements**

**4.1 Map Asset to Source Work**

**Linked Repository**

* Display **all repositories** available in the system.
* Repositories must be shown in a **dropdown list**.
* The repository list must also be visible in the **sidebar** as a dropdown.

**Linked Task**

* Once a repository is selected, related tasks must load.
* User must be able to **add and select tasks**.
* Current issue: Linked Task dropdown is throwing an error and does not allow selection.
* Expected behaviour: Dropdown should load tasks without error and allow selection.

**4.2 Asset Classification**

**Changes Required**

* Remove **Repository** field from Asset Classification section.
* Update **Asset Type** dropdown with the following values only:
  + Blog Banner
  + Infographic
  + Social Post
  + Reel / Video
  + Thumbnail
  + Diagram
  + Web Graphic
  + PDF

**4.3 Content & SEO Metadata Structure**

The following fields must be **logically divided**:

**Content Keywords**

* User-entered keywords related to content.

**SEO Keywords**

* Keywords must be **fetched from Keyword Master Database**.
* Only keywords already added in Keyword Master should be listed.
* Display keywords as a **selectable list (dropdown / multi-select)**.
* User can only **select**, not manually type SEO keywords.

Additional fields:

* Title
* Meta Description
* URL

**4.4 Heading Structure**

Provide structured heading inputs:

* H1 Tag
* H2 Tag
* H2 Tag

Each tag should have a separate input field.

**4.5 Body Content Editor**

**Editor Requirements**

* Body Content input must be a **large rich-text editor**.
* Editor should function similar to **MS Word**.

**Mandatory Formatting Options**

* Bold
* Italic
* Underline
* Font size
* Font style
* Paragraph styles
* Lists (bullet & numbered)

**4.6 AI Integrations (Side Panel)**

A side panel must be displayed next to the Body Content editor with the following scores:

* **SEO Score**
* **Grammar Score**
* **AI / Plagiarism Score**

**Notes**

* These scores must be calculated using **AI integration**.
* Scores should update based on content changes.

**4.7 Resource Upload**

**Upload Capabilities**

* Provide a **multi-upload option**.
* User can upload **any number** of resources.

**Supported File Types**

* Images
* Documents
* PDFs
* Other supporting files

**4.8 Designer & Workflow Details**

**Auto / Manual Assignment**

* **Created By**
  + Auto-populated based on logged-in user.
* **Designed By**
  + Dropdown list.
  + Fetch users from **Admin Console → Users**.
* **Published By**
  + Dropdown list.
  + Fetch users from **Admin Console → Users**.
* **Verified By (SEO)**
  + Dropdown list.
  + Fetch users from **Admin Console → Users**.

**4.9 Workflow Stages**

Workflow Stage dropdown must include the following statuses:

* In Progress
* Sent to QC
* Published
* In Rework
* Moved to CW
* Moved to GD

**4.10 QC Status (Separate Field)**

QC Status must be a **separate field**, independent of Workflow Stage.

Available statuses:

* QC Pending
* Rework
* Approved
* Reject

**Workflow Logic & Status Transitions**

**Step 1: Asset Creation & Initial Work**

* When a user starts creating or editing an asset:
  + **Workflow Stage:** In Progress
  + **QC Status:** Not applicable at this stage

**Step 2: Asset Submission for QC**

* Once the user completes the asset and submits it for review:
  + **Workflow Stage:** Sent to QC
  + Asset is moved to the **QC Review Module**
  + **QC Status:** QC Pending

**Step 3: QC Review Outcomes**

**A. QC Approval**

* In the **QC Review Module**:
  + **QC Status:** Approved
* **Workflow Stage:** Remains *Sent to QC*
* Asset becomes **eligible for publishing**

**B. QC Rework Requested**

* In the **QC Review Module**:
  + **QC Status:** Rework
* In **Upload Assets / Workflow**:
  + **Workflow Stage:** In Rework
* Asset is sent back to the creator for corrections

**C. QC Rejection**

* In the **QC Review Module**:
  + **QC Status:** Rejected
* Workflow behaviour after rejection can be finalized based on business rules; however, the **Rejected status must be clearly always reflected in QC Status**.

**Step 4: Publishing the Asset**

* Once an approved asset is published:
  + **Workflow Stage:** Published
  + **QC Status:** Remains Approved

**Cross-Team Movement & Work Ownership Visibility**

**Workflow Stage Additions**

The following workflow stages must be supported:

* Moved to CW
* Moved to GD
* Moved to WD

**Behaviour & Visibility Rules**

* All teams (CW, GD, WD) will work on the **same Web Asset page**.
* When an asset is moved:
  + If moved to **CW**, the top of the Web asset page must display:
    - **“CW is working on this asset”**
  + If moved to **GD**, the top of the Web asset page must display:
    - **“GD is working on this asset”**
  + If moved to **WD**, the top of the Web asset page must display:
    - **“WD is working on this asset”**

**Purpose**

* This visibility ensures that, even though multiple teams access the same asset page, users can clearly identify:
  + **Which team is currently working on the asset**
  + **Where the workflow is currently paused or active**

**3. Key Rules & Clarifications**

* **Workflow Stage** and **QC Status** must be treated as **separate but linked states**.
* QC actions should **only update QC Status**, except:
  + When QC gives **Rework**, the **Workflow Stage must change to In Rework**.
* Publishing is the **only action** that moves the Workflow Stage to Published.

**4. Expected Outcome**

* Clear visibility of:
  + Asset lifecycle progress (Workflow Stage)
  + Quality approval state (QC Status)
* Reduced confusion between production and QC processes
* Accurate tracking of assets across Upload Assets and QC Review module

**4.11 Versioning**

**Version Control Requirements**

* System must maintain **version history** for each asset.
* Each update should create a new version.
* Previous versions must be retrievable for reference or rollback.

**5. Acceptance Criteria (Summary)**

* Correct section order is followed
* No dropdown errors in Linked Task
* Asset types match specified list
* SEO keywords are fetched only from Keyword Master
* Rich-text editor supports all formatting options
* AI scores are visible and functional
* Resource upload supports multi-file upload
* Workflow, QC, and user assignments function correctly
* Version history is maintained

**6. Notes for Development Team**

* Do not assume default behaviours.
* Follow this document strictly for implementation.
* Any deviation must be discussed before development.

SEO: (05.01.25)

**1. Purpose of This Document**

This document defines the **exact functional flow** for the SEO Asset module. The sequence below is **strict** and must be followed in UI, backend logic, and validations. No steps can be skipped or reordered.

**2. Step-by-Step Functional Flow**

**Step 1: Asset ID Selection**

**Field:** Asset ID  
**Type:** Dropdown (Single Select)

**Rules:**

* Dropdown must display **only Asset IDs already created in the system**
* No manual typing allowed
* Selection is mandatory

**Behaviour:**

* Once selected, Asset ID becomes read-only
* All subsequent sections remain disabled until Asset ID is selected

**Step 2: Map Assets to Source Work**

**Purpose:** Link the selected Asset ID to its originating work/source

**Includes:**

* All asset linkage and mapping logic

 Linked **Task** (Dropdown – Select Task) from master

 Linked **Campaign** (Dropdown – Select Campaign) from master

 Linked **Project** (Dropdown – Select Project) from master

 Linked **Service** (Dropdown – Select Service) from master

 Linked **Sub-Service** (Dropdown – Select Sub-Service) from master

 Linked **Repository Item** (Dropdown / Selector) from master

**Step 3: Asset Classification**

**Fields:**

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Source** |
| Asset Type | Dropdown | Master Database |
| Sector | Dropdown | Master Database |
| Industry | Dropdown | Master Database |

**Rules:**

* Sector and Industry must be fetched from master database
* No hardcoding allowed

**Step 4: SEO Metadata Fields & Anchor Text**

**Fields:**

|  |  |  |
| --- | --- | --- |
| Field | Type | Mandatory |
| Title | Text | Yes |
| Meta Title | Text | Yes |
| Description | Text Area | Yes |
| Service URL | URL | Optional |
| Blog URL | URL | Optiona |
| Anchor Text | Text | Yes |

**Step 5: Keywords**

**Fields:**

|  |  |  |
| --- | --- | --- |
| Field | Type | Rule |
| Primary Keyword | Integrated Dropdown/Search | Mandatory |
| LSI Keywords | Multi-select Integrated Field | Optional |

**Rules:**

* Keywords must integrate with Keyword Master
* Manual free-text entry not allowed unless enabled in keyword system

**Step 6: Domain Type & Domain Addition**

**6.1 Domain Type**

**Field:** Domain Type  
**Type:** Dropdown  
**Source:** Backlink Master

**6.2 Domain Addition (+)**

**Button:** + Add Domain

**Rules:**

* Domains must be fetched from Backlink Master
* Multiple domains allowed per SEO Asset

**Step 7: Domain Details Popup (Mandatory Interaction)**

**7.1 Popup Trigger**

* Triggered when user clicks on any added domain

**7.2 Popup Fields**

|  |  |  |
| --- | --- | --- |
| Field Name | Type | Source |
| Domain Name | Text | Backlink Master |
| URL Posted | URL | User Input |
| SEO Self QC Status | Dropdown | Manual |
| QA Status | Dropdown | Manual |

**7.3 QC Status Mapping Logic**

|  |  |
| --- | --- |
| QC Status Value | System Display Status |
| Pass | Approved |
| Fail | Rejected |
| Waiting | Pending |

* Display status must update automatically based on QC selection

**Step 8: Blog Posting – Content Editor Rule**

**Condition:**

* If Asset Type = Blog Posting

**Behaviour:**

* Display a rich text (Word-like) content editor
* Supports headings, formatting, links, and paragraphs

**If not Blog Posting:**

* Content editor must remain hidden

**Step 9: Resource File Upload**

**Field:** Resource File Upload  
**Type:** Multi-file Upload

**Rules:**

* Multiple files allowed
* Files linked to Asset ID and Asset Type

**Step 10: Designer & Workflow**

**Fields:**

|  |  |  |
| --- | --- | --- |
| Field | Type | Rule |
| Assign Team Members | Multi-select Dropdown | All Users |
| Created By | Auto | Logged-in User |
| Verified By (SEO) | Dropdown | All Users |

**Step 11: Versioning**

**Purpose:** Track changes across submissions

**Rules:**

* New version created on every re-submit after rejection
* Previous versions must remain read-only
* Version number auto-incremented by system

**Step 12: Actions – Save, Submit, Discard**

**Buttons:**

* Save (Draft)
* Submit (For QC / Approval)
* Discard

**Behavior:**

* Save stores data without validation completion
* Submit triggers validations and workflow
* Discard clears unsaved data

**3. Non-Negotiable Development Rules**

* Flow order is mandatory and system-enforced
* Master data must be dynamically fetched
* No hardcoded values where masters exist
* Conditional visibility must strictly follow Asset Type
* QC and approval logic must be system-driven