

Integración API-ASS

Este documento se presenta como una guía general de integración con la consola Aranda Self Service (ASS). Se presentan algunas operaciones de manera detallada incluyendo datos de petición, parámetros, respuestas y códigos de error. \n## Generalidades

A continuación se presentan las generalidades que deben tenerse en cuenta para el correcto funcionamiento de los servicios de ASS.

- Versión compatible de base de datos 8.0.66
- Las operaciones sobre el API deben realizarse con usuarios que tengan los permisos correspondientes asignados desde la consola Aranda PROFILE.
- La visualización de los Artículos, dependerá de la administración previa de un Administrador. Solo se visualizaran los Artículos que se encuentren en un estado "Aprobado" y tengan asociada una "Categoría".
- Las Categorías asociadas deben tener seleccionada la opción de Artículos.
- Los servicios creados vienen con el instalador de Aranda Self Service (ASS) y son solo de visualización y consulta. \n## Descripción de operaciones – title: Descripción de operaciones chapter: "descripcion" layout: apis –

A continuación esta sección se describe detalladamente cómo se consume cada uno de los servicios correspondientes a cada método. Se incluyen detalles como URI, tipo de operación, parámetros (requeridos y opcionales) y sus tipos, respuestas y códigos y mensajes de error. \n## Manejo de sesión – title: Manejo de sesión chapter: "descripcion" layout: apis –

En esta sección se describen las operaciones relacionadas con el manejo de sesiones (autenticación de usuarios y cierre de sesión). \n## Inicio de sesión – title: Inicio de sesión chapter: "descripcion" layout: apis –

Detalles de la petición:

- URI: api/v8.1/user/login
- Tipo: POST
- Encabezados requeridos:
 - Content-Type: application/json

Parámetros:

Nombre	Tipo	Obligatorio	Descripción
USERNAME	Texto	Si	Usuario que iniciará sesión.
PASSWORD	Texto	Si	Contraseña correspondiente al usuario.
LANGUAGEID	Número	No	Idioma con que se registrará la sesión. Las posibles opciones son: 1. Inglés 2. Español 3. Portugués Si no se provee, por defecto se registra en inglés.

Cuerpo de la petición:

La petición consiste de un arreglo json de tipo field-value con los posibles parámetros según se requiera. Ejemplo:

```
[
  {
    "Field": "username",
    "Value": "USUARIO_ASDK"
  },
  {
    "Field": "password",
    "Value": "CONTRASEÑA_DEL_USUARIO"
  }
]
```

Respuesta:

Como respuesta se obtiene un objeto json, con la siguiente información:

```
[
  {
    "Field": "userId",
    "Value": "####"
  }
]
```

```
},
{
  "Field": "sessionId",
  "Value": "###AAABBCC###"
},
{
  "Field": "result",
  "Value": "True"
}
]
```

Mensajes de error:

[Código|Estado HTTP|Mensaje de error| |---|---| |400|BadRequest|InvalidUserName| |400|BadRequest|InvalidPassword| |400|BadRequest|InvalidLangId| |401|Unauthorized|InvalidUserData| |500|InternalServerError|FailureOnLogin| \n## Cierre de sesión – title: Cierre de sesión chapter: “descripcion” layout: apis –

Detalles de la petición:

- URI: api/v8.1/user/logout
- Tipo: POST
- Encabezados requeridos:
 - Content-Type: application/json
 - Authorization: TOKEN

Parámetros:

Sólo se requiere realizar solicitud enviando el token de autenticación como encabezado.

Cuerpo de la petición:

Vacía.

Respuesta:

No existe un cuerpo de respuesta, la validación del resultado de la operación se puede realizar analizando el código http obtenido: 200 OK para una operación exitosa.

Mensajes de error:

[Código | Estado HTTP | Mensaje de error | |-----|-----|-----| |401| Unauthorized| InvalidToken| |500| InternalServerError| FailureCloseSession| \n## Manejo de artículos – title: Manejo de artículos chapter: “descripcion” layout: apis –

En esta sección se describen las operaciones relacionadas con el manejo de los artículos (Listado, Detalles, Adjuntos y Relacionados). \n## Listado de artículos – title: Listado de artículos chapter: “descripcion” layout: apis –

Detalles de la petición:

- URI: api/v8.1/article/list
- Tipo: POST
- Encabezados requeridos:
 - Content-Type: application/json
 - Authorization: TOKEN

Parámetros:

Nombre	Tipo	Obligatorio	Descripción
CategoryId	Numérico	No	Id de la categoría.
FieldSearch	String	No	Tipo de búsqueda.
PrivacyId	Numérico	Si	Privacidad del artículo.
ProjectId	Numérico	Si	Id del Proyecto.
TextSearch	String	No	Texto a buscar.
TypeId	Numérico	No	Id Tipo de artículo.

Cuerpo de la petición:

En el cuerpo de la petición tienen que venir los parámetros PrivacyId y ProjectId, como mínimo para realizar la consulta y a su vez estar diligenciados.

El campo ProjectId, solo recibe datos enteros y debe ser diligenciado con el ID del proyecto que será consultado.

El campo PrivacyId, maneja la privacidad y debe ser diligenciado con los siguientes valores enteros:

- 0, Consulta los artículos privados y públicos.
- 1, Consulta los artículos públicos.

Ejemplo

```
{
  "PrivacyId":0,
  "ProjectId":1
}
```

Para consultas más específicas, se pueden adicionar los siguientes parámetros:

- **CategoryId**, se debe ingresar el ID de la categoría a consultar.
- **TextSearch**, consulta un texto en específico del Artículo, el cual debe ir entre comillas "".
- **TypeId**, se debe ingresar el id del "Tipo" de artículo.
- El parámetro **FieldSearch**, puede enviar vacío o con las siguientes asignaciones para realizar las consultas de acuerdo a su tipo:
 - *LastVisited*: Últimos visitados.
 - *FavoriteByUser*: Favoritos del usuario.
 - *QualifiedByUser*: Calificados por el usuario.
 - *MostVisitedByUser*: Más visitados por el usuario.
 - *AddedByProject*: Añadidos por proyecto.
 - *MostVisitedByProject*: Más visitados por proyecto.
 - *QualifiedByProject*: Calificados por proyecto.
 - *AddedByCategory*: Añadidos por categoría.
 - *MostVisitedByCategory*: Más visitados por categoría.
 - *QualifiedByCategory*: Calificados por categoría.

Ejemplo:

```
{
  "CategoryId":0,
  "FieldSearch":"QualifiedByUser",
  "PrivacyId":1,
  "ProjectId":2,
  "TextSearch":"Ingresar texto",
  "TypeId":3
}
```

Respuesta:

El servicio retornará un Json con la siguiente estructura:

```
[
  {
    "Description": "Descripción del artículo",
    "Id": valor numérico correspondiente al id,
    "Title": "Titulo del Artículo"
  }
]
```

Mensajes de error:

| Código | Estado HTTP | Mensaje de error | |-----|-----|-----| | 400 | BadRequest | DataSearchIsNull | | 400 | BadRequest | PrivacyIdsNull | | 400 | BadRequest | InvalidProjectId | | 400 | BadRequest | InvalidPrivacyId | | 401 | Unauthorized | InvalidToken | | 500 | InternalServerError | FailureSearchArticle | \n## Detalle del artículo – title: Detalle del artículo chapter: "descripcion" layout: apis –

La consulta del detalle del Artículo, se realizara dependiendo del ID que se envié en la consulta.

Detalles de la petición:

- URI: `api/v8.1/article/{id}/detail`
- Tipo: GET
- Encabezados requeridos:
 - Content-Type: application/json
 - Authorization: TOKEN

Parámetros:

Nombre	Tipo de dato	Obligatorio	Descripción
Id	Número	Si	Id del artículo

Cuerpo de la petición:

Vacía.

Respuesta:

El servicio retornará un Json con la siguiente estructura:

```
{
  "Description": "Descripción del artículo",
  "Id": 1, "Title": "Titulo del Artículo",
  "AuthorId": null, "AuthorName": null,
  "AuthorUserName": null, "CategoryId": null,
  "CategoryName": null, "ClassId": 0,
  "Content": Contenido Html,
  "CreateReasonId": 0,
  "Created": "/Date(1336768613863-0500)/",
}
```

```
"GroupId": null,
"Interfaceld": null,
"IsRatedByUser": true,
"IsUserFavorite": false,
"Ispublic": false,
"KeyWords": "Palabras Claves del articulo",
"LastModified": "/Date(1336768922067-0500)/",
"ModifiedDate": null,
"ModifierUserId": null,
"ModifierUserName": null,
"OpenedDate": null,
"ProjectId": 7,
"ProjectName": "PPQA",
"Rated": 1,
"RatingTotal": 5.5,
"ReasonDescription": null,
"ReasonId": 0,
"ResponsibleGroupId": 0,
"ResponsibleGroupName": null,
"ResponsibleId": 0,
"ResponsibleName": null,
"SolutionId": null,
"StatusDescription": null,
"StatusId": 0,
"TotalHit": 12,
"TypeDescription": null,
"TypeId": 5,
"TypeName": null,
"Url": ""
}
```

Mensajes de error:

| Código | Estado HTTP | Mensaje de error | |-----|-----|-----| |400| BadRequest | InvalidArticleId | |401| Unauthorized | InvalidToken | |404| NotFound | InvalidArticleId | |500| InternalServerError | FailureGetDetailArticle | \n## Adjuntos del articulo – title: Adjuntos del articulo chapter: “descripcion” layout: apis –

La consulta de los Adjuntos asociados a un Artículo, se realiza dependiendo del ID que se envié en la consulta.

Detalles de la petición:

- URI: api/v8.1/article/{id}/file/list
- Tipo: GET
- Encabezados requeridos:
 - Content-Type: application/json
 - Authorization: TOKEN

Parámetros:

Nombre	Tipo de dato	Obligatorio	Descripción
Id	Número	Si	Id del articulo

Cuerpo de la petición:

Vacía.

Respuesta:

Dependiendo de la cantidad de adjuntos asociados al Artículo, el servicio retornará un bloque por cada adjunto en un Json con la siguiente estructura:

```
[
{
"Created": "/Date(1473801979043-0500)/",
"Id": 3721,
"Name": "Nombre del adjunto.docx",
"Size": 634311,
"Url": "Ubicación del adjunto"
}
]
```

Mensajes de error:

| Código | Estado HTTP | Mensaje de error | |-----|-----|-----| |400| BadRequest | InvalidArticleId | |401| Unauthorized | InvalidToken | |404| NotFound | InvalidArticleId | |500| InternalServerError | FailureListFiles | \n## Artículos Relacionados – title: Artículos Relacionados chapter: “descripcion” layout: apis –

La consulta de los Artículos relacionados, se realizara sobre el ID del Artículo que se envía. El campo idPrivacy, maneja la privacidad de los artículos relacionados y debe ser diligenciado con los siguientes valores enteros:

- 0, Consulta los artículos relacionados con estado privado.
- 1, Consulta los artículos relacionados con estado público.
- 2, Consulta todos los artículos sin importar su privacidad.

Detalles de la petición:

- URI: `api/v8.1/article/{id}/{idPrivacy}/related/list`
- Tipo: GET
- Encabezados requeridos:
 - Content-Type: application/json
 - Authorization: TOKEN

Parámetros:

Nombre	Tipo de dato	Obligatorio	Descripción
Id	Número	Si	Id del articulo
idPrivacy	Número	Si	Id privacidad del articulo

Cuerpo de la petición:

Vacía.

Respuesta:

El servicio retornará un Json con la siguiente estructura:

```
[  
{  
  "Description": "Ejemplo de descripción",  
  "Id": 1,  
  "Title": "Titulo del Artículo"  
}]
```

Mensajes de error:

Código	Estado HTTP	Mensaje de error
400	BadRequest	InvalidArticleId
400	BadRequest	InvalidIdPrivacy
401	Unauthorized	InvalidToken
404	NotFound	InvalidArticleId
500	InternalServerError	FailureListRelatedArticles