

Integración API-CMDB

Este documento se presenta como una guía general de integración con Aranda CMDB. Se presentan las operaciones de manera detallada incluyendo datos de petición, parámetros, respuestas y códigos de error.

Generalidades

A continuación, se presentan las generalidades que deben tenerse en cuenta para el correcto funcionamiento del API de Aranda CMDB.

- Versión compatible con consola de Aranda CMDB Web 8.4.3 y versión de base de datos 8.0.62
- Versión Microsoft .NET Framework 4.5 o superior.
- Las operaciones sobre el API deben realizarse con usuarios que tengan los permisos correspondientes asignados desde Aranda PROFILE.
- El uso del API requiere licencias disponibles, las cuales tendrán el mismo comportamiento como si un especialista iniciara sesión en la consola.
- La creación de CI mediante el API cumple las mismas reglas de licencias para CI disponibles, tal como si se crearan desde la consola.
- Los CI's registrados mediante la integración se van a crear en el proyecto que esté configurado por defecto en la ruta "Configuración de Parámetros>Proyecto predeterminado" dentro de la consola CMDB WEB, este puede ser modificado.
- El proyecto debe tener configurado las correspondencias entre categoría y tipo de CI en la consola CMDB WEB.
- Las categorías configuradas deben tener asociados estados y su respectivo flujo.
- La consola CMDB WEB en el menú "Configuración de Parámetros>Sitio para Autorización" debe tener configurada la ruta del sitio de autenticación.
- La consulta, edición, desactivación y reactivación se puede realizar por el Id del CI o por el código del GuID.
- Los siguientes campos ahora reciben parámetro tipo texto en la creación o edición, si el valor que se envía ya existe se asocia al mismo ID que tiene en base de datos, en caso que no exista, se crea y se asocia a un nuevo ID: CurrentStatus, ProgrammedStatus, Manufacturer, Provider, ProfileCI, CostCenter, Unit, Brand, Model, Building, City, Company, Country, Department, Floor, Site y NumberDepartment.

Descripción de operaciones

En esta sección se detalla el consumo de los servicios correspondientes a cada método. Se incluyen detalles como URI, tipo de operación, parámetros (requeridos y opcionales) y sus tipos, respuestas, códigos y mensajes de error.

En esta sección se describen las operaciones relacionadas con el manejo de sesiones (autenticación de usuarios, renovación de token y cierre de sesión).

Detalles de la petición:

- URI: api/v8.6/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.
CONSOLEID	Número	No	
CONSOLEVERSION	Texto	No	Versión de la consola desde la que se inicia sesión.
DEVICEIMEI	Texto	No	IMEI del dispositivo móvil desde el que se inicia sesión.
DEVICEBRANDID	Número	No	Marca del dispositivo desde el que se inicia sesión. Las posibles opciones son: 1. Android 2. iOS
DEVICENAME	Texto	No	Nombre del dispositivo móvil desde el que se inicia sesión.
DEVICEREGISTRATIONID	Texto	No	RegistrationId del dispositivo móvil, utilizado para mensajes push.
APPLICATIONNAME	Texto	No	Nombre de la aplicación externa que consume el servicio.
FOREIGN	Texto	No	No es obligatorio siempre y cuando el nombre de la aplicación no se envíe. Las posibles opciones son: 1. True (si aplicación es foránea) 2. False (Si aplicación no es foránea)

Cuerpo de la petición:

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

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

Ejemplo para inicio de sesión con aplicación externa:

```
[
  {
    "Field": "username",
    "Value": "USUARIO_CMDB"
  },
  {
    "Field": "password",
    "Value": "CONTRASEÑA_DEL_USUARIO"
  },
  {
    "Field": "applicationname",
    "Value": "Nombre de aplicación externa"
  },
  {
    "Field": "foreign",
    "Value": "true"
  }
]
```

Respuesta:

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

```
[
  {
    "Field": "userId",
    "Value": "####"
  },
  {
    "Field": "sessionId",
    "Value": ""###AAABBCCC###"
  },
  {

```

```
"Field": "result",
"Value": "True"
}
]
```

Mensajes de error:

Código	Estado HTTP	Mensaje de error
400	BadRequest	InvalidUserName
400	BadRequest	InvalidPassword
400	BadRequest	InvalidConsoleId
400	BadRequest	InvalidConsoleVersion
400	BadRequest	InvalidDeviceId
400	BadRequest	InvalidDeviceBrandId
400	BadRequest	InvalidDeviceName
400	BadRequest	InvalidLangId
400	BadRequest	DeviceNamesRequired
400	BadRequest	InvalidForeignValue
400	BadRequest	InvalidApplicationName
500	InternalServerError	FailureOnLogin

\n### Renovación de Sesión

title: Renovación de Sesión chapter: "descripcion" layout: apis —

Detalles de la petición:

- URI: `api/v8.6/session/renew`
- Tipo: POST
- Encabezados requeridos:
 - Content-Type: `application/json`
 - Authorization: `TOKEN`

Parámetros:

Cuerpo de la petición:

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:\n### Cierre de sesión

title: Cierre de sesión chapter: "descripcion" layout: apis —

Detalles de la petición:

- URI: `api/v8.6/user/logout`
- Tipo: POST
- Encabezados requeridos: Content-Type: `application/json`
 - Content-Type: `application/json`
 - Authorization: `TOKEN`

Parámetros:

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

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
400	BadRequest	InvalidToken
500	InternalServerError	FailureCloseSession

\n## Manejo de CI

title: Manejo de CI chapter: "descripcion" layout: apis –

En esta sección se describen las operaciones relacionadas con el manejo de CI (creación, edición, consulta, borrado y reactivación). \n## Creación de CI – title: Creación de CI chapter: "descripcion" layout: apis –

Detalles de la petición:

- URI: api/v8.6/ci/add
- Tipo: POST
- Encabezados requeridos:
 - Content-Type: application/json
 - Authorization: TOKEN

Parámetros:

Nombre	Tipo	Obligatorio	Descripción
Name	Texto	Si	Nombre del CI.
ProjectId	Número	No	Id del Proyecto.
Guld	Texto	No	Id externo del CI. Este campo es obligatorio sólo si el CI proviene de una integración.
Type	Número	No	Tipo de CI. Este campo es obligatorio sólo si el CI proviene de una integración. Las opciones son: 1. Application 2. BusinessProcess 3. Catalog 4. Computer 5. Hub 6. Monitor 7. Phone 8. Printer 9. Router 10. Server 11. Service 12. SoftwareProduct 13. SystemSoftware 14. Tablet
CategoryID	Número	No	Id de la categoría. Si se envía el campo Type en la petición, este campo no debe ser ingresado.
AuthorId	Número	No	Id del autor. Si no se envía, se obtiene el del usuario asociado al token que se utiliza en la petición.
Description	Texto	No	Descripción del Ci.
Building	Texto	No	Nombre del edificio.
City	Texto	No	Nombre de ciudad.
Company	Texto	No	Nombre de la compañía.
CostCenter	Texto	No	Centro de costo.
Country	Texto	No	Nombre del país.
CurrentStatus	Texto	No	Nombre del estado actual.
Department	Texto	No	Nombre del departamento, Ej: Desarrollo.
DepreciationMethod	Número	No	Id del método de depreciación (1: Línea recta, 2: Suma de dígitos).
Floor	Texto	No	Nombre del piso, ej: 25-Sotano
GroupId	Número	No	Id de la agrupación del CI.

Nombre Impact	Tipo Número	Obligatorio No	Descripción Impacto del CI en la organización.
Manufacturer	Texto	No	Nombre de la compañía fabricante del CI.
Model	Texto	No	Modelo del CI.
DepartmentNumber	Texto	No	Número del departamento.
Price	Decimal	No	Precio del CI.
CIProfile	Texto	No	Perfil del CI.
ProgrammedStatus	Texto	No	Nombre del estado futuro del CI.
Provider	Texto	No	Nombre de la compañía proveedora del CI.
ResidualValue	Decimal	No	Valor a salvar del precio del CI.
Alias	Texto	No	Usuario responsable del CI, puede ir en los siguientes formatos: Dominio\Usuario ó Usuario@Dominio
Site	Texto	No	Sitio.
Unit	Texto	No	Unidad de medida del CI ej: pulgadas
UsefulLife	Número	No	Años de vida util del CI, para el cálculo de la depreciación
Brand	Texto	No	Marca.
AdditionalField:			
Type	Número	No	Los tipos de campos adicionales son: 1. Texto 2. Fecha 3. Lista 4. Texto largo 5. Check box 6. Número
Value	-	No	Información que se va a ingresar al campo adicional, depende del tipo de dato que espere el campo adicional
Id	Número	No	Id del campo adicional

Observaciones :

- El campo CITotal no hace parte de los parámetros, ya que se calcula automáticamente por la herramienta y es de uso interno.
- La petición es sensible a mayúsculas y minúsculas, por tanto, se deben dejar los nombres de los parámetros tal como están en la tabla de parámetros.

Cuerpo de la petición:

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

```
{
  "Name": "Nombre del Ci",
  "CategoryId": 19,
  "GroupId": -1,
  "CurrentStatus": "En uso",
  "AuthorId": 3,
  "ProgrammedStatus": "Estado Programado Prueba",
  "Model": "GXP1200",
  "Alias": "Usuario@Dominio ",
  "AdditionalField": [
    {
      "Id": ###,
      "Type": #,
      "Value": "Valor"
    }
  ]
}
```

Respuesta:

Como respuesta se obtiene un identificador único del CI creado.

Mensajes de error:

Código	Estado HTTP	Mensaje de error
400	BadRequest	InvalidEventName
400	BadRequest	InvalidUserId
400	BadRequest	InvalidGuld
400	BadRequest	InvalidType
400	BadRequest	InvalidApplicationName
400	BadRequest	InvalidApplicationId
400	BadRequest	InvalidCiId
400	BadRequest	InvalidProjectIdByDefault
400	BadRequest	InvalidSubcategoryId
400	BadRequest	InvalidStateName
400	BadRequest	InvalidCompany
400	BadRequest	InvalidIdSettingGeneral
400	BadRequest	CatalogDescriptionIsRequired
400	BadRequest	InvalidCategoryId
404	NotFound	NotFoundStatesOfCMDB
500	InternalServerError	FailureAddCi
500	InternalServerError	FailureAddCiOrGetIdCi

\n## Consulta de CI

title: Consulta de CI chapter: "descripcion" layout: apis —

Detalles de la petición:

- URI: `api/v8.6/ci/{id}?isGuld={isGuld}`
- Tipo: GET
- Encabezados requeridos:
 - Content-Type: application/json
 - Authorization: TOKEN

La estructura URI de la petición se implementa de la siguiente forma:

- Si es por CI: `api/v8.6/ci/IdCI?isGuld=False`
- Si es por Guld: `api/v8.6/ci/Guld?isGuld=True`

Parámetros:

Nombre	Tipo de dato	Obligatorio	Descripción
Id	Texto	Si	Id del CI o Guld que se va a consultar
IsGuld	Booleano	Si	En caso que el Id que se envíe sea Guld el valor será "true", si corresponde al Id de un CI entonces será "false"

Respuesta:

Como respuesta retornara un objeto json, con la siguiente información:

```
{
  "AcceptDate": null,
  "AdditionalField": [
    {
      "Alias": "ADD_STR_1",
```

```
"Caption": "Prueba caracteristica en 80",
  "Id": 92,
  "IdLookField": 0,
  "Mandatory": false,
  "Order": 0,
  "Type": 1,
  "Value": "Caracteristica en 80",
  "ValueLookField": null,
  "Visible": false
},
{
  "Alias": "ADD_STR_2",
  "Caption": "Home",
  "Id": 213,
  "IdLookField": 0,
  "Mandatory": false,
  "Order": 0,
  "Type": 1,
  "Value": "prueba additional home",
  "ValueLookField": null,
  "Visible": false
},
{
  "Alias": "ADD_STR_3",
  "Caption": "Aranda",
  "Id": 217,
  "IdLookField": 0,
  "Mandatory": false,
  "Order": 0,
  "Type": 6,
  "Value": null,
  "ValueLookField": null,
  "Visible": false
},
},
],
"Address": null,
"ApplicationName": null,
"AssetTag": null,
"AuthorId": 2017,
"BackgroundCurrentStatus": "255.255.255",
"BarCode": null,
"Brand": "Nueva marca prueba 2",
"BrandId": "71",
"Building": null,
"BuildingId": null,
"CategoryId": 9,
"CheckOutDate": null,
"CheckInDate": null,
"CiProfile": null,
"CiProfileId": null,
"CiTotal": 0,
"City": "Panama Prueba crear 6",
"CityId": null,
"Company": "Disc Guadalajara",
"CompanyId": null,
"CostCenter": null,
"CostCenterId": "5495",
"Country": "Mexico",
"CountryId": null,
"CurrentStatus": "Estado Prueba 2",
"CurrentStatusId": 718,
"Department": "Development",
"DepartmentId": null,
"DepartmentNumber": null,
"DepartmentNumberId": null,
"DepreciationMethod": null,
>Description": "Prueba catalogo crear centro costo id",
"Floor": null,
"FloorId": null,
"ForegroundCurrentStatus": "0.0.0",
"Foreign": false,
"GroupId": -1,
"Guid": "Guid_154_iyuhd_03052016_1652",
"Id": 5238,
"ImageFileName": null,
"Impact": null,
"LicenseNumber": null,
"Manufacturer": "Prueba Manufacturer",
"ManufacturerId": 929,
"MapFilename": null,
"Model": "GXP1200",
"ModelId": "12",
>Name": "AddCi_03052016_1652",
"Price": null,
"ProfileId": 0,
"ProgrammedDate": null,
"ProgrammedStatus": "Estado Programado Prueba",
"ProgrammedStatusId": 717,
"ProjectId": null,
"Provider": "Company Prueba 2",
"ProviderId": 929,
"RFID": null,
"ReponsibleDate": null,
"ResidualValue": null,
"ResponsibleId": null,
"Serial": null,
"Site": null,
"SiteId": null,
"Size": null,
"StatId": 718,
"Type": null,
"Unit": null,
"UnitId": 1,
"UsefulLife": null,
"UserIdModifier": null
}
```

Mensajes de error:

Código	Estado HTTP	Mensaje de error
400	BadRequest	InvalidCild
404	NotFound	NotFoundCild
500	InternalServerError	FailureGetDetailCi

Edición CI

title: Edición CI chapter: "descripcion" layout: apis —

Detalles de la petición:

- URI: `api/v8.6/ci/{id}?isGuld={isGuld}`
- Tipo: PUT
- Encabezados requeridos:
 - Content-Type: `application/json`
 - Authorization: `TOKEN`

La estructura URI de la petición se implementa de la siguiente forma:

- Si es por CI: `api/v8.6/ci/IdCI?isGuld=False`
- Si es por Guld: `api/v8.6/ci/Guld?isGuld=True`

Parámetros

Código	Tipo de dato	Obligatorio	Descripción
Id	Texto	Si	Id del CI o Guld a editar
IsGuld	Booleano	Si	En caso que el Id que se envíe sea Guld el valor será "true", si corresponde al Id de un CI entonces será "false"

Adicionalmente es posible enviar como parámetros en el cuerpo de la petición los valores que se quieren editar. Estos valores corresponden a los mismos disponibles en el proceso de creación.

Cuerpo de la petición:

Ejemplo:

```
{
  "Description": "AAABBBCCDDDEEE",
  "AdditionalField": [
    {
      "Id": 486,
      "Type": 1,
      "Value": "Windows 15"
    }
  ]
}
```

Respuesta:

Valor booleano indicando el resultado de la operación.

Mensajes de error:

Código	Estado HTTP	Mensaje de error
400	BadRequest	InvalidToken
400	BadRequest	InvalidUserId
400	BadRequest	InvalidCiData
400	BadRequest	InvalidCild
400	BadRequest	InvalidCategoryId
400	BadRequest	InvalidProjectIdByDefault
400	BadRequest	InvalidSubcategoryId
500	InternalServerError	FailureParseCi
500	InternalServerError	FailureUpdateCi

\n## Borrar CI

title: Borrar CI chapter: "descripcion" layout: apis –

Detalles de la petición:

- URI: `api/v8.6/ci/{id}?isGuld={isGuld}`
- Tipo: DELETE
- Encabezados requeridos:
 - Content-Type: application/json
 - Authorization: TOKEN

La estructura URI de la petición se implementa de la siguiente forma:

- Si es por CI: `api/v8.6/ci/IdCI?isGuld=False`
- Si es por Guld: `api/v8.6/ci/Guld?isGuld=True`

Parámetros:

Nombre	Tipo de dato	Obligatorio	Descripción
Id	Texto	Si	Id del CI o Guld a borrar
IsGuld	Booleano	Si	En caso que el Id que se envíe sea Guld el valor será "true", si corresponde al Id de un CI entonces será "false"

Cuerpo de la petición

Vacía

Respuesta:

Valor booleano indicando el resultado de la operación.

Mensajes de error:

Código	Estado HTTP	Mensaje de error
400	BadRequest	InvalidToken
404	BadRequest	InvalidUserId
404	BadRequest	InvalidCiId
500	InternalServerError	FailureDeleteCi

Reactivar CI

title: Reactivar CI chapter: "descripcion" layout: apis –

Detalles de la petición:

- URI: `api/v8.6/ci/{ci}/reactivate?isGuld={isGuld}`
- Tipo: PUT
- Encabezados requeridos:
 - Content-Type: application/json
 - Authorization: TOKEN

La estructura URI de la petición se implementa de la siguiente forma:

- Si es por CI: `api/v8.6/ci/IdCI/reactivate?isGuld=False`
- Si es por Guld: `api/v8.6/ci/Guld/reactivate?isGuld=True`

Parámetros:

Nombre	Tipo de dato	Obligatorio	Descripción
Id	Texto	Si	Id del CI o Guld a reactivar.
IsGuld	Booleano	Si	En caso que el Id que se envíe sea Guld el valor será "true", si corresponde al Id de un CI entonces será "false"

Cuerpo de la petición

Vacía

Respuesta:

Valor booleano indicando el resultado de la operación.

Mensajes de error:

Código	Estado HTTP	Mensaje de error
400	BadRequest	InvalidToken
404	BadRequest	InvalidUserId
404	BadRequest	InvalidCiId
500	InternalServerError	FailureReactivateCi

Campos Adicionales

title: Campos Adicionales chapter: "descripcion" layout: apis –

Consulta Campos Adicionales

Detalles de la petición:

- URI: `api/v8.6/additionalfield`

- Tipo: POST
- Encabezados requeridos:
 - Content-Type: application/json
 - Authorization: TOKEN

Parámetros:

Nombre	Tipo de dato	Obligatorio	Descripción
ProjectId	Número	No	Id del proyecto del Ci.
Cid	Texto	No	Id del Ci, si aplicación es foránea recibe su id foráneo.
CategoryId	Número	No	Id de la categoría del Ci.
TypeId	Número	No	Tipo de Ci.

Cuerpo de la petición:

Ejemplo:

```
{
  "TypeId": "14"
}
```

Respuesta:

```
[
  {
    "Alias": "ADD_STR_1",
    "Caption": "Prueba característica en 80",
    "Id": 92,
    "IdLookField": 0,
    "Mandatory": false,
    "Order": 0,
    "Type": 1,
    "Value": null,
    "ValueLookField": null,
    "Visible": false
  },
  {
    "Alias": "ADD_STR_2",
    "Caption": "Home",
    "Id": 213,
    "IdLookField": 0,
    "Mandatory": false,
    "Order": 0,
    "Type": 1,
    "Value": null,
    "ValueLookField": null,
    "Visible": false
  },
  ...
]
```

Mensajes de error:

Código	Estado HTTP	Mensaje de error
400	BadRequest	InvalidToken
404	BadRequest	InvalidUserId
400	BadRequest	InvalidCid
400	BadRequest	InvalidProjectIdByDefault
404	NotFound	NotFoundCid
404	NotFound	NotFoundTypeId
500	InternalServerError	FailureGetListAdditionalFields

\\n## Edición Campos Adicionales

title: Edición Campos Adicionales chapter: "descripcion" layout: apis –

Detalles de la petición:

- URI: `api/v8.6/additionalfield/{cid}?isGuld={isGuld}`
- Tipo: PUT
- Encabezados requeridos:
 - Content-Type: `application/json`
 - Authorization: `TOKEN`

La estructura URI de la petición se implementa de la siguiente forma:

- Si es por CI: `api/v8.6/additionalfield/IdCI?isGuld=False`
- Si es por Guld: `api/v8.6/additionalfield/Guld?isGuld=True`

Parámetros:

Nombre	Tipo de dato	Obligatorio	Descripción
Id	Texto	Si	Id del CI o Guld que se va a editar
IsGuld	Booleano	si	En caso que el Id que se envíe sea Guld el valor será "true", si corresponde al Id de un CI entonces será "false"

Nota: En el cuerpo de la petición, se envía el Id que corresponde al campo adicional que se va a editar, éste se puede obtener mediante la consulta de campos adicionales descrita en el apartado anterior.

Cuerpo de la petición:

Ejemplo:

```
[  
  {  
    "Id": 494,  
    "Value": "501"  
  },  
  ...  
]
```

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
400	BadRequest	InvalidToken
404	BadRequest	InvalidUserId
400	BadRequest	NotFoundCild
400	BadRequest	InvalidCild
400	NotFound	InvalidCategoryId
400	NotFound	InvalidDateValue
400	NotFound	InvalidType
400	NotFound	InvalidDataElement
400	NotFound	InvalidBooleanValue
400	NotFound	InvalidNumericValue
500	InternalServerError	FailureUpdateAdditionalFields

\n### Escenarios Básicos – title: Escenarios Básicos chapter: “escenarios” layout: apis –

A continuación, se describen los pasos necesarios para realizar algunas de las operaciones más comunes. \n### Iniciar Sesión – title: Iniciar Sesión chapter: “escenarios” layout: apis –

1. Llamar servicio de inicio de sesión enviando los parámetros requeridos.
2. Obtener la respuesta del servicio.
3. Almacenar datos como Id del usuario y token de autorización pues serán requeridos para servicios posteriores.

\n### Renovar Sesión – title: Renovar Sesión chapter: “escenarios” layout: apis –

1. Llamar el servicio de renovación de sesión enviando el token de autorización que se tenga almacenado.
2. Si la respuesta del servicio es exitosa, el token sigue siendo válido.
3. Si la respuesta del servicio no es exitosa, es necesario obtener un nuevo token de acuerdo a lo indicado en el escenario de inicio de sesión.

\n### Crear un CI

title: Crear un CI chapter: “escenarios” layout: apis –

1. Obtener un token de autenticación de acuerdo a lo indicado en los escenarios de inicio de sesión o renovar sesión, según corresponda.
2. Consumir el servicio de creación enviando encabezado de autenticación y cuerpo de petición con los datos correspondientes al caso según corresponda.
3. Gestionar según corresponda los datos obtenidos como resultado de consumo del servicio.

\n### Obtener Detalle de CI – title: Obtener Detalle de CI chapter: “escenarios” layout: apis –

1. Obtener un token de autenticación de acuerdo a lo indicado en los escenarios de inicio de sesión o renovar sesión, según corresponda.
2. Consumir servicio de obtener detalle de CI enviando los datos correspondientes al caso a consultar.
3. Gestionar según corresponda la respuesta obtenida de consumir el servicio.

\n### Actualizar un CI – title: Actualizar un CI chapter: “escenarios” layout: apis –

1. Obtener un token de autenticación de acuerdo a lo indicado en los escenarios de inicio de sesión o renovar sesión, según corresponda.
2. Consumir el servicio de edición de CI enviando encabezado de autenticación y cuerpo de petición con los datos a editar.
3. Gestionar según corresponda los datos obtenidos como resultado de consumo del servicio.

1. Obtener un token de autenticación de acuerdo a lo indicado en los escenarios de inicio de sesión o renovar sesión, según corresponda.
2. Consumir el servicio de edición de CI enviando encabezado de autenticación y cuerpo de petición con los datos a editar.
3. Los campos adicionales tanto en la creación como en la edición requieren obligatoriamente el Id del campo adicional, tipo de campo y la información que se va a consignar en el mismo, éstos valores se pueden conocer mediante la consulta de campos adicionales.
4. Para enviar información a un campo adicional desde la creación o edición de un CI, este campo debe existir para el tipo (categoría) de CI que se va a gestionar, de lo contrario no se podrá ejecutar la acción satisfactoriamente.
5. La creación de campos adicionales se realiza únicamente desde la consola de administración de Aranda CMDB versión web y Windows.