

Implementación de Manuales

El manual de Buenas Prácticas de documentación aporta la información pertinente para diseñar, organizar y gestionar proyectos de documentación siguiendo las etapas sugeridas y estableciendo recomendaciones generales en la gestión de contenido y la estructura del proyecto de documentación.



1. Gestión Repositorios

Conozca cómo administrar y configurar los repositorios en Azure DevOps donde se organizan los diferentes manuales y contenidos asociados a los productos de Aranda.

2. Gestión Contenido

Conozca el ciclo de vida de un proyecto de documentación en Azure DevOps e identifique cómo organizar los contenidos (texto, imágenes, tablas) para optimizar las tareas de documentación.

3. Pautas de Documentación

Identifique los lineamientos generales a tener en cuenta en un proceso de documentación destacando la estructura de información a implementar y las recomendaciones gramaticales más relevantes.

Para Quién es este manual

Este manual está pensado en los encargados de generar y actualizar la documentación de los diferentes manuales asociados a los productos de Aranda.

Empezar a implementar

Teniendo en cuenta las instrucciones compartidas podrá implementar, gestionar y publicar un proyecto de documentación.

Configurar Repositorio

Configuración repositorio Azure

La configuración del repositorio en Azure nos permitirá definir de manera organizada los componentes del proyecto de documentación

Para administrar un proyecto de documentación se deben tener en cuenta estas 3 etapas de la configuración:



1. Crear repositorio

Cada proyecto de documentación debe tener un repositorio en Azure para cada uno de los manuales por producto.

2. Clonar repositorio

Para mantener el estándar en la gestión del proyecto de documentación y optimizar el trabajo es recomendable reutilizar o clonar la información de otro repositorio.

3. Estructurar repositorio

Organizar la información de un proyecto de documentación, de carácter colaborativo, es determinante para administrar de forma eficiente los contenidos generados, facilitando encontrar y actualizar la información.

El repositorio de un proyecto en Azure DevOps contiene toda la información para organizar los contenidos requeridos.

Crear Repositorio

Un repositorio en Azure permite almacenar y organizar la información de un proyecto de documentación.

Cada proyecto de documentación debe tener un repositorio en Azure para cada uno de los manuales por producto.

📌 **Nota:** - 1. El equipo de documentación será el encargado de crear y definir el nombre del repositorio. El repositorio inicial se crea sin datos y posteriormente se unifica con el repositorio clonado para el manual requerido por el encargado del producto

- 2. El nombre del repositorio debe estar compuesto por las siglas del producto y si lo requiere el tipo de manual. Ejemplo docs-asms-admin

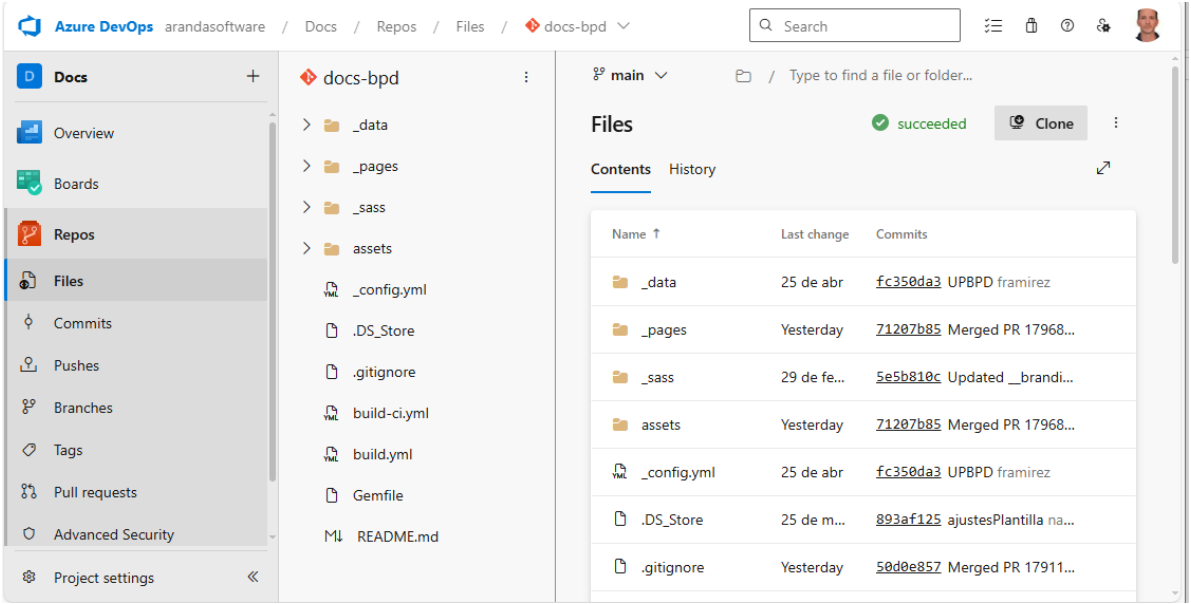
Importante

1. Tener instalado en la máquina local, el **visual studio code**.
2. Contar con Licencia de **Azure DevOps** y verificar permisos para entrar y editar los contenidos del [Repositorio Documentación Aranda](#)
3. Para comenzar solicita al área de Documentación la creación del repositorio para el producto.
4. Para visualizar el manual localmente debes seguir los pasos descritos en este link [Visualización Jekyll](#)

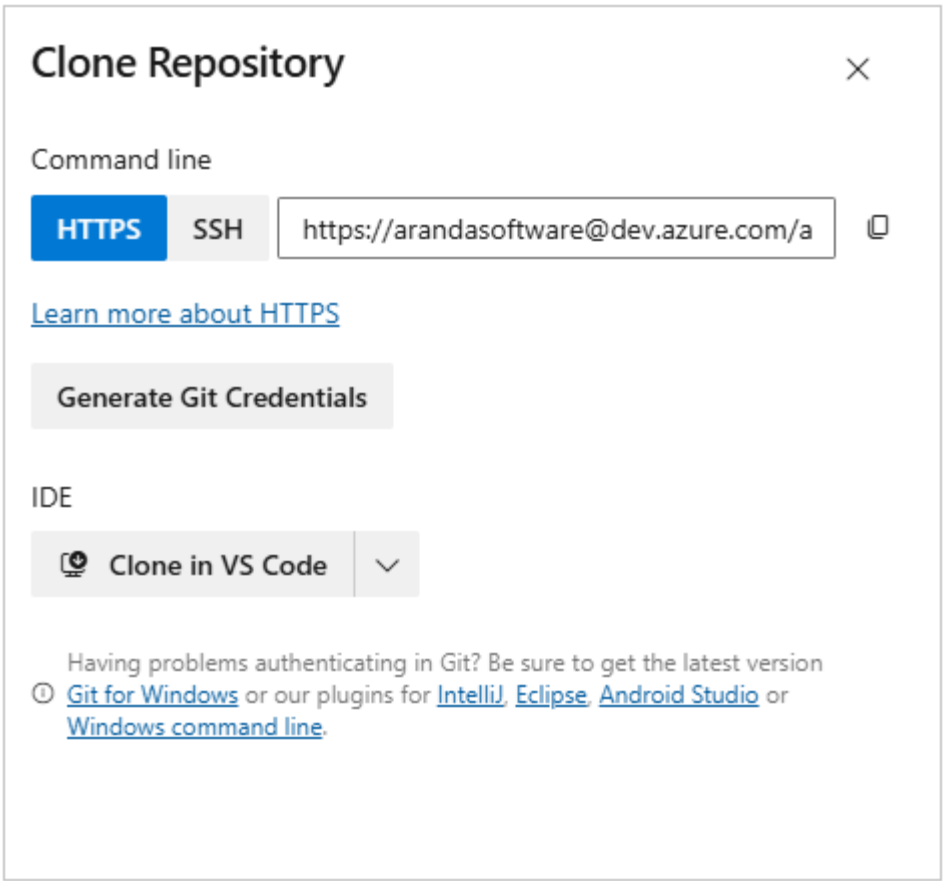
Clonar Repositorio

Cómo clonar una Repositorio

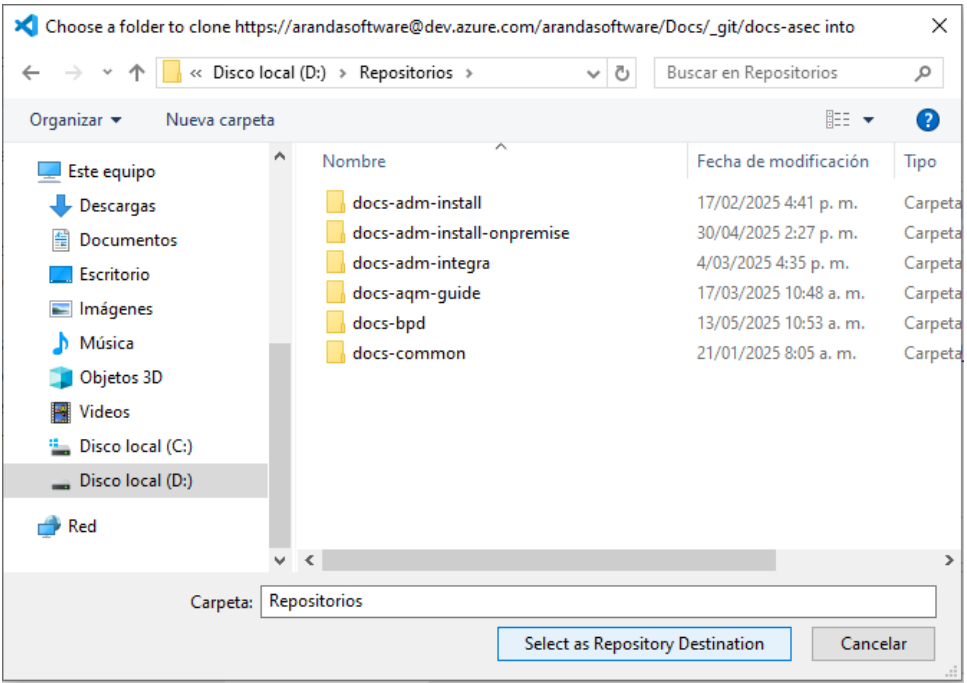
1. Ingrese a la consola de Azure DevOps, en el menú principal seleccione la opción Repos (repositorio).



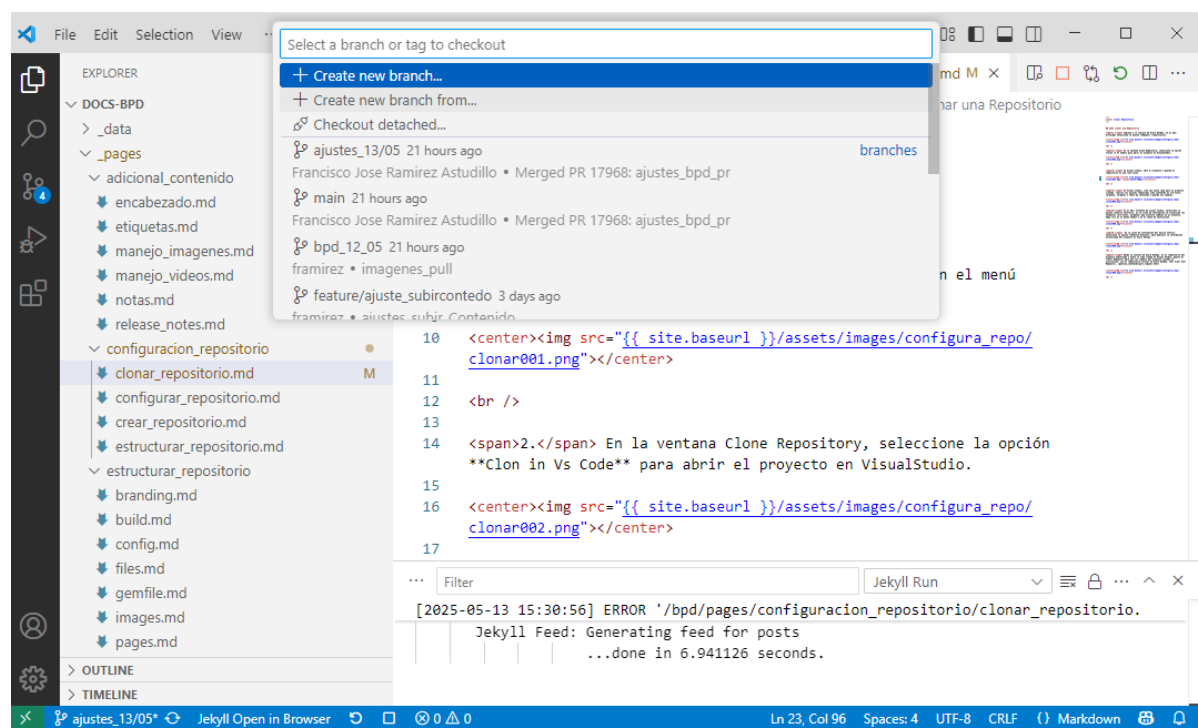
2. En la ventana Clone Repository, seleccione la opción Clon in Vs Code para abrir el proyecto en VisualStudio.



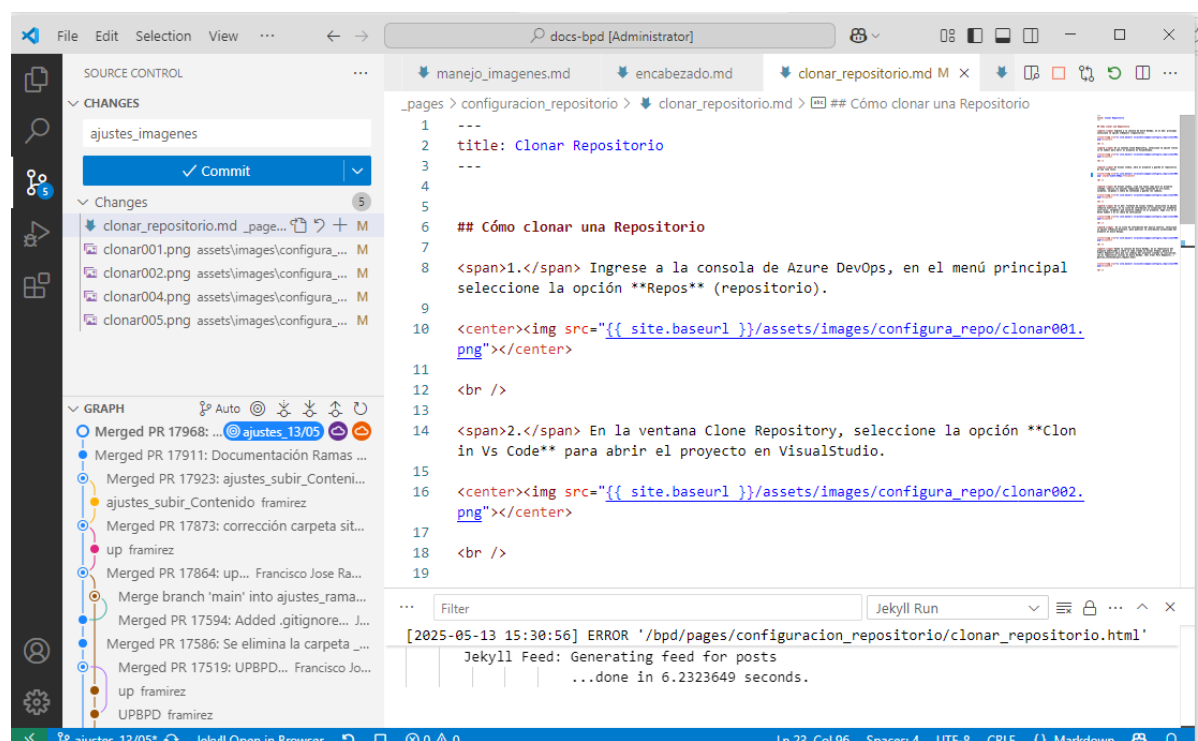
3. En Visual studio, abra el proyecto y guarde el repositorio en una ruta local.



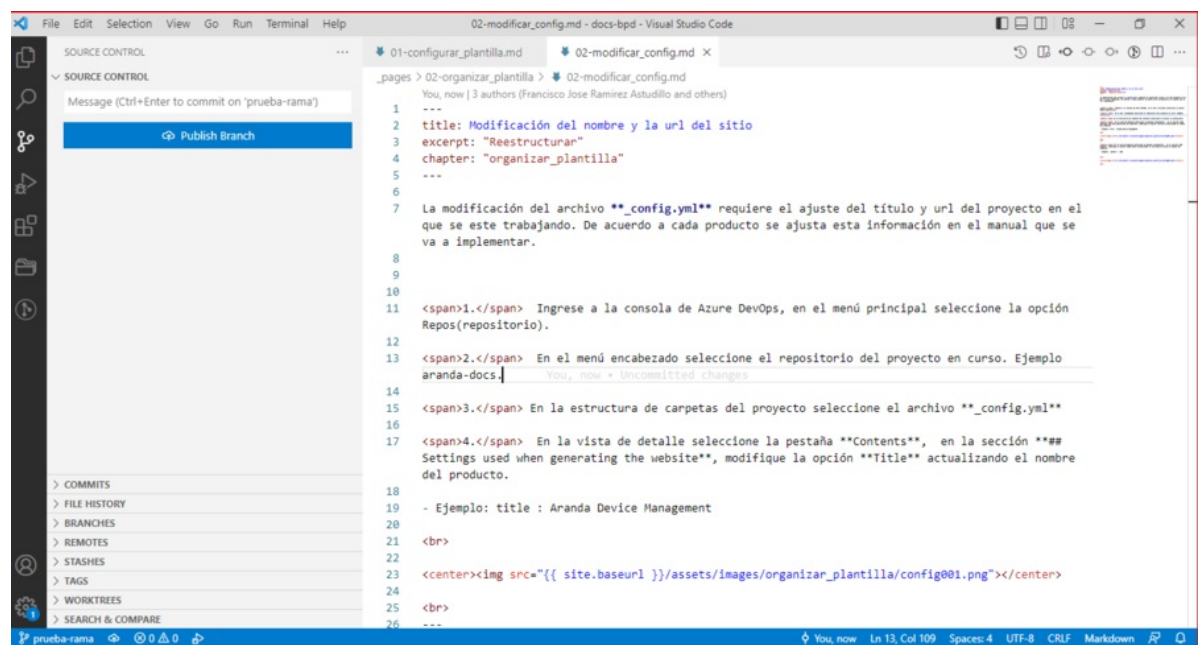
4. En Visual studio, cree una nueva rama para el proyecto clonado, realice los ajustes requeridos como contenido en artículos, carpetas, imágenes o tabla de contenido y guarde los cambios.



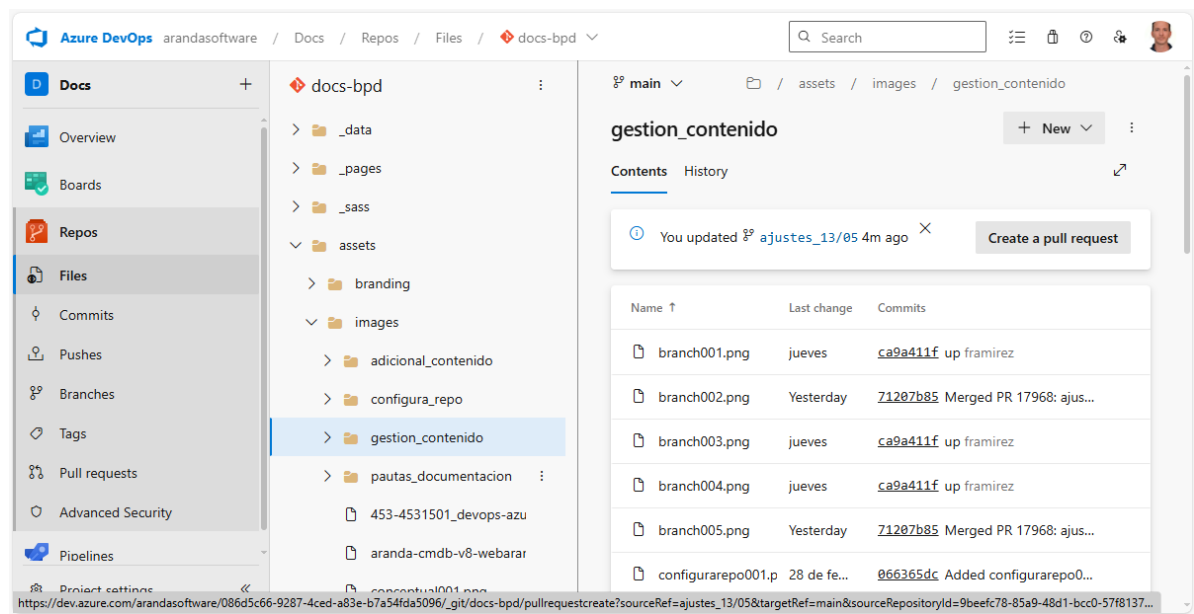
5. En el menú flotante de visual studio, seleccione la opción **Source Control**, en la vista de información se visualizan los elementos (artículos, imágenes) que tuvieron cambios en el proyecto. haga clic en el botón **Commit** o en el check de veificación



6. En la vista de información del Source Control, seleccione la opción **Publish Branch**, para publicar la información actualizada del proyecto en Azure DevOps.



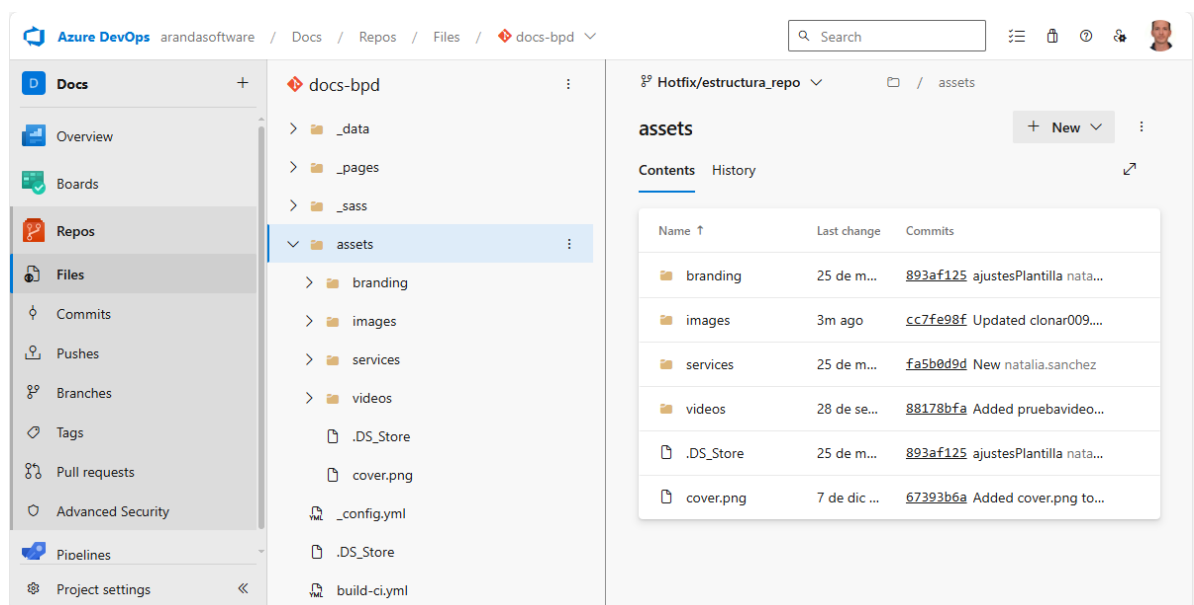
7. Desde la consola de Azure DevOps, en el repositorio del proyecto modificado y sobre la rama creada en Visual Studio, genere el **Pull Request** para que los cambios del proyecto clonado se sincronicen, con el repositorio inicial en Azure DevOps. [Ver Crear Pull Request](#)



Estructurar repositorio

Cómo acceder a la estructura de un repositorio

1. Ingrese a la consola de Azure DevOps, en el menú principal seleccione la opción **Repos** (repositorio) o **Files**.
2. En el menú encabezado seleccione el repositorio del proyecto en curso. Ejemplo docs-bpd.



3. La estructura de datos del repositorio del proyecto está compuesta por carpetas, artículos Markdown, imágenes y archivos de configuración. En las siguientes secciones podrá gestionar la estructura del repositorio:

_data

La carpeta **_data** contiene la estructura de la tabla de contenido del proyecto, en el archivo sidebar.yml.

Organizar de forma correcta el sidebar nos permite visualizar el proyecto en el orden requerido.

[Ver Modificar Sidebar \(Tabla de Contenido\)](#)

_pages

La carpeta **_pages** tiene los artículos del proyecto agrupados en carpetas, de acuerdo al orde jerárquico de aparición de la información.

[Ver Organizar Carpetas y Artículos](#)

_branding-dir.scss

La carpeta **branding** tiene el archivo con la información de los estilos y logos del proyecto, en el archivo **_branding-dir.scss**. Esta información debe ser actualizada en relación con las características gráficas de cada producto de Aranda.

[Ver Ajustes Branding](#)

assets/images

La carpeta `images` de la sección `assets`. agrupa todas las imágenes que se carguen al proyecto. Estas imágenes serán llamadas desde cada uno de los artículos configurados en la carpeta `pages`.

[Ver Modificar Imágenes](#)

_config.yml

El archivo `_config.yml` tiene datos generales de configuración de la plantilla y después de clonado el proyecto, podrá modificar información relevante como el título, la url del proyecto generado y adicionar los links (enlaces tipo variable) requeridos para acceder a documentación externa (otros repositorios de productos Aranda) y realizar su posterior llamado a través de esta variable.

[Ver Modificar Archivo _config.yml](#)

_build.yml

El archivo `_build.yml` contiene las instrucciones del sitio y su navegación.

[Ver Modificar Archivo _build.yml](#)

_Gemfile

El `_Gemfile` es un archivo de texto en un proyecto que define las dependencias (bibliotecas y frameworks) que necesita la aplicación.

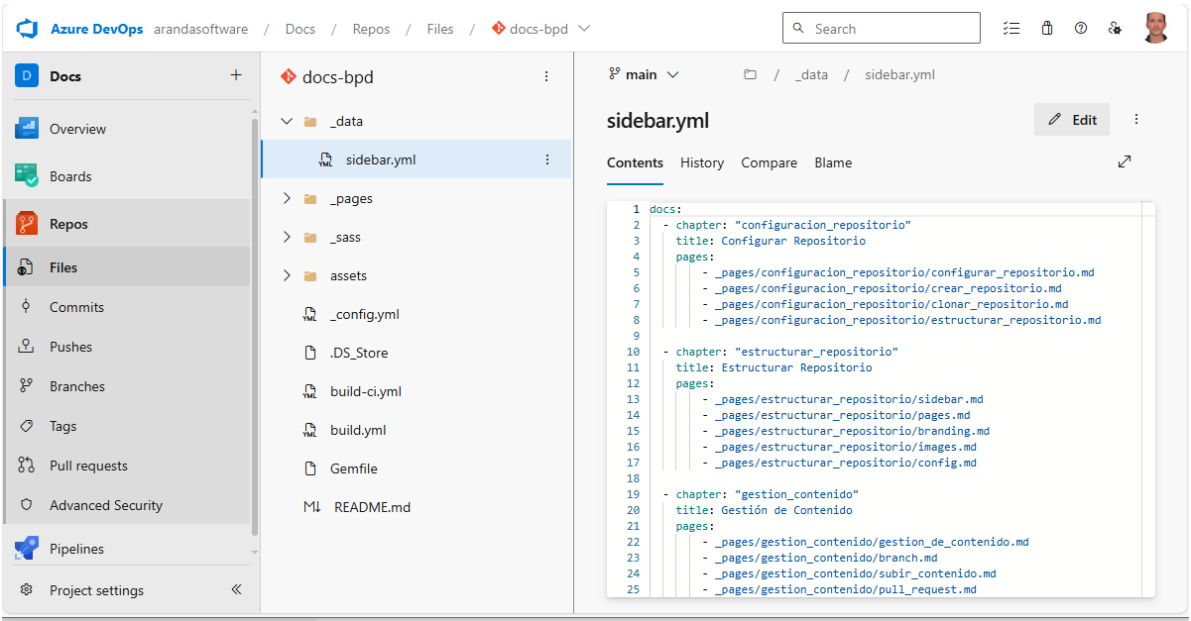
[Ver Modificar Archivo _gemfile.yml](#)

Estructurar Repositorio

Sidebar/Tabla de contenido

La modificación del archivo `sidebar.yml`, permite organizar la estructura de la tabla de contenido del proyecto/manual de acuerdo al orden lógico de uso de la funcionalidad.

Los campos que se deben organizar en el sidebar son `title` y `chapter`.



Cómo modificar el archivo _sidebar

1. Ingrese a la consola de Azure DevOps, en el menú principal seleccione la opción **Repos** (repositorio).
2. En el menú encabezado seleccione el repositorio del proyecto/manual en curso. Ejemplo aranda-docs.

📌 **Nota:** Antes de modificar la la tabla de contenido en el sidebar, cree una rama para registrar los cambios. [Ver creación de Ramas en un Proyecto](#)
3. En la vista de información seleccione la carpeta `_data` y el archivo `sidebar.yml`; en la vista detalle del sydebar haga clic en **Edit** para organizar la tabla de contenido del manual que podrá visualizar el usuario final.

Los elementos como está estructurado el sidebar o tabla de contenido son los siguientes:

Nombre	Descripción
chapter	Es la categoría principal que agrupa los artículos de un tema de la funcionalidad.
title	El es el nombre visible del capítulo que agrupa los artículos de un tema de la funcionalidad y se presentará en forma jerárquica en la tabla de contenido del manual.
Pages	Corresponde al nombre de los artículos de un tema específico de la funcionalidad y se relaciona la ruta dentro del repositorio donde se encuentra el artículo. Ejemplo: <code>_pages/nombre_del_chapter/nombre_del_articulo.md</code>

- 📌 **Advertencia:** El nombre del Chapter del artículo debe coincidir con el nombre del chapter relacionado en el sidebar; de esta forma la tabla de contenido estructurada en el sydebar podrá tener visible las categorías establecidas.

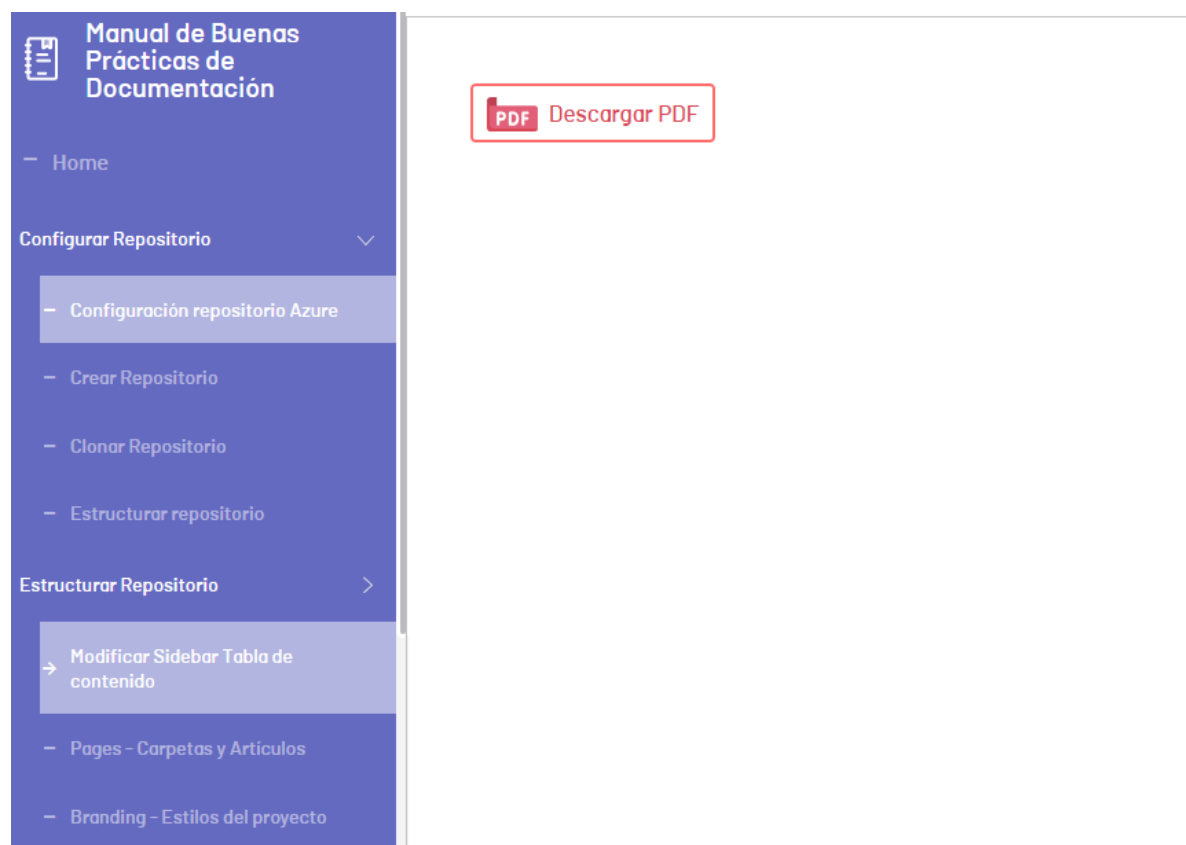
📌 **Nota:** Los artículos relacionados en la estructura del sidebar pueden pertenecer a un solo chapter(capítulo) o a varios chapter.

Estructura básica del sidebar:

```
docs:
- chapter: "configuracion_repositorio"
  title: Configurar Repositorio
  pages:
    - _pages/configuracion_repositorio/configurar_repositorio.md
    - _pages/configuracion_repositorio/crear_repositorio.md
    - _pages/configuracion_repositorio/clonar_repositorio.md
    - _pages/configuracion_repositorio/estructurar_repositorio.md

- chapter: "estructurar_repositorio"
  title: Estructurar Repositorio
  pages:
    - _pages/estructurar_repositorio/sidebar.md
    - _pages/estructurar_repositorio/pages.md
    - _pages/estructurar_repositorio/branding.md
    - _pages/estructurar_repositorio/images.md
    - _pages/estructurar_repositorio/config.md
```

Resultado de la estructura del sidebar:



La información ingresada en el sidebar debe tener en cuenta lo siguiente:

Recomendaciones:

- 1. Los nombres en *Chapter*, *Title* y *Pages* No debe tener espacios entre las palabras.
- 2. Los nombres en *Chapter* y *Pages* deben estar escritos en minúsculas.-
- 3. Los nombres del *Title* pueden estar escritos en minúscula/mayúscula y pueden llevar tildes
- 4. Para unir palabras se debe utilizar guion o raya al piso (- _).
- 5. No se deben incluir caracteres especiales, ni tildes.

4. Después de modificar el sydebar haga clic en **Commit** para actualizar los cambios.

5. Para implementar los ajustes en el proyecto, en la vista detalle del sydebar seleccione la opción **Pull Request**. [Ver Crear Pull Request](#)

Importante

- La definición y organización correcta del “Chapter”, posibilita la visualización de los contenidos en el menú de navegación. Si un concepto no tiene este chapter no será visible al momento de hacer la publicación.
- Cada artículo dentro de la opción Pages, tiene referenciado en su encabezado el nombre del Título (Title) y en nombre del Capítulo (chapter) al que pertenecen.

_Pages/ Folders (Carpetas)

La opción **_pages** en Azure DevOps permite agrupar la información del proyecto en carpetas y artículos Markdown. Organizar y estructurar los contenidos debe hacerse teniendo en cuenta el orden lógico de funcionamiento del producto, presentando al usuario final.

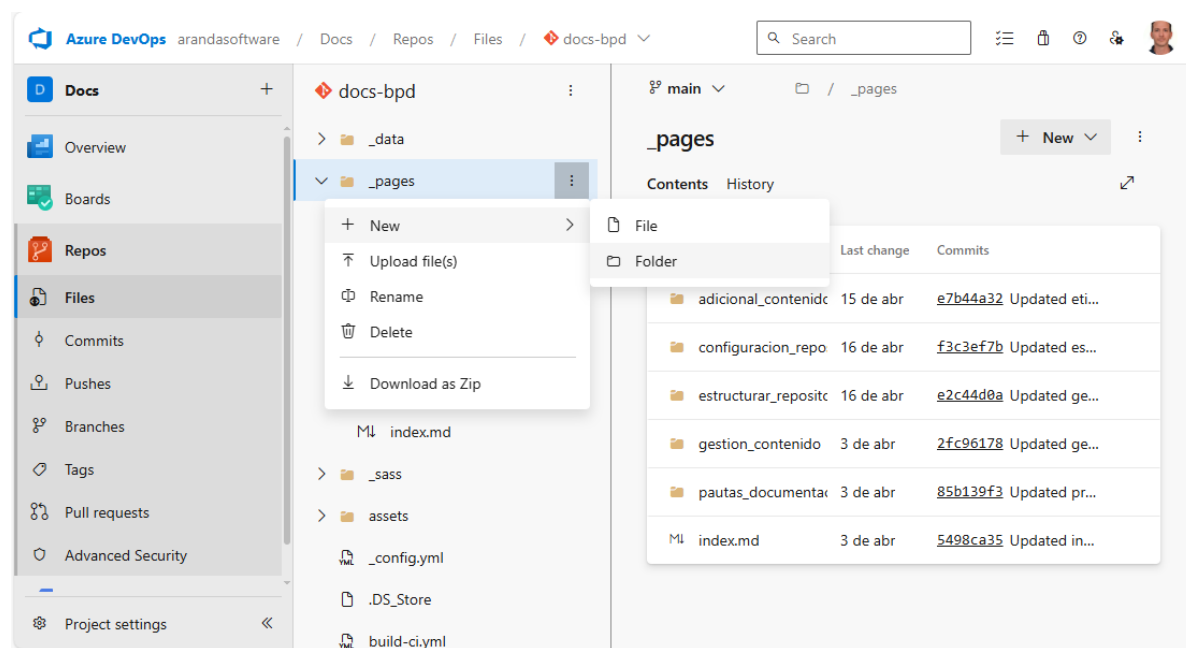
Crear Carpetas en el proyecto

1. Ingrese a la consola de Azure DevOps, en el menú principal seleccione la opción **Repos** (repositorio).

2. En el menú encabezado seleccione el repositorio del proyecto/manual en curso. Ejemplo docs-bpd.

📌 **Nota:** Antes de modficar la carpeta, cree una rama para registrar los cambios. [Ver creación de Ramas en un Proyecto](#)

3. Para crear una nueva carpeta, En la vista de información seleccione la carpeta **_pages** y en la opción more opttions, seleccione **New/Folder**.



4. Ingrese el nombre de la carpeta y el nombre del primer artículo asociado.

New folder

New folder name

/_pages/configuracion_repo...

configuracion_bpd

Use slashes to create multiple subfolders like "sub/folder".

New file name

nuevo.md

Git folders cannot be empty, so a placeholder file will be added. Its content can be edited before commit.

Cancel

Create

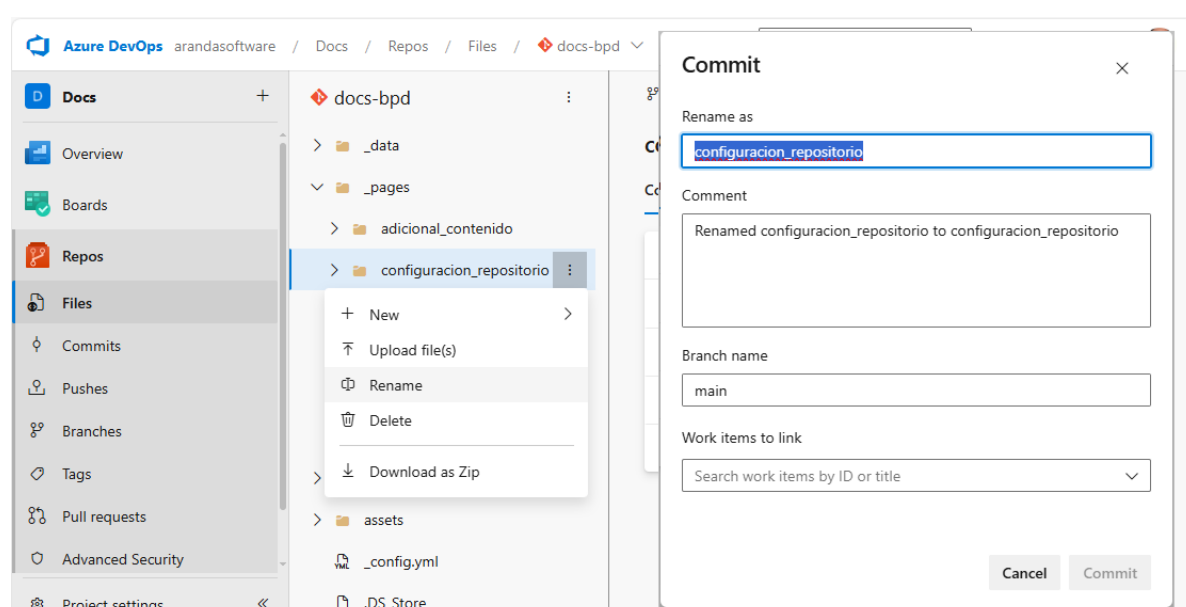
RECOMENDACIONES:

- 1. El nombre de la carpeta No debe tener espacios entre las palabras.
- 2. El nombre de la carpeta debe estar escrito en minúsculas.
- 3. El nombre de la carpeta corresponde al nombre del capítulo o chapter que agrupa los contenidos o conceptos de la funcionalidad.
- 4. Dentro de la opción **_pages** se define una carpeta por cada por cada tema o capítulo del manual, la cual está constituida por artículos que corresponde a un concepto principal de la funcionalidad.

4. En la vista detalle del artículo creado, ingrese el encabezado del documento ([Ver Encabezado Artículos](#)) y el contenido requerido. Haga clic en el botón **Commit** para guardar los cambios propuestos.

Renombrar una Carpeta

5. Para renombrar una carpeta, en la vista de información seleccione la carpeta **_pages** y una carpeta creada; en la opción more options, seleccione **Rename**. Se habilita la ventana para actualizar el nombre de la carpeta.



Eliminar una Carpeta

5. Para eliminar una carpeta, en la vista de información seleccione la carpeta `_pages` y una carpeta creada; en la opción `more options`, seleccione `Delete`. Se habilita la ventana para eliminar la carpeta del repositorio.
6. Después de crear, renombrar o eliminar la carpeta, haga clic en el botón `Commit` para guardar los cambios propuestos y [generar el Pull Request](#).

Importante a Tener en Cuenta

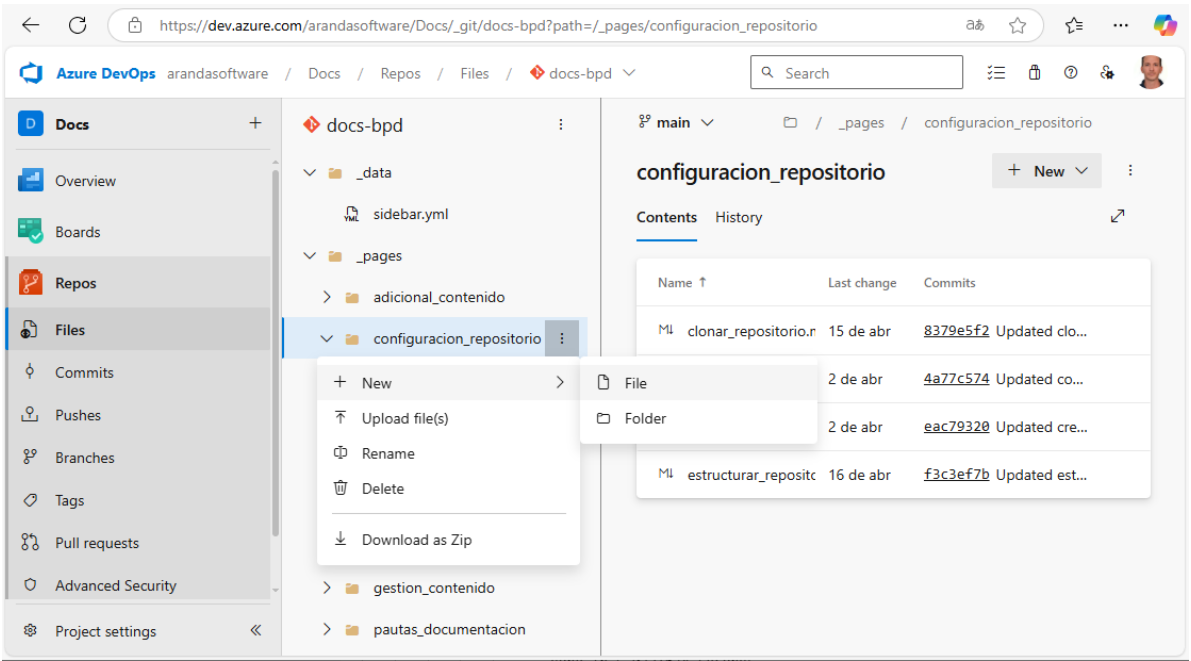
- La correcta gestión de carpetas, artículos e imágenes, epermitea optimizar los procesos actualización, revisión y cambios en el proyecto de documentación.
- [Agregar contenido\(imágenes, texto, tablas\)](#) en un proyecto de Documentación.

_Pages/ Files (Artículos)

La opción `_pages` en Azure DevOps permite agrupar la información del proyecto en carpetas y artículos Markdown. Organizar y estructurar los contenidos debe hacerse teniendo en cuenta el orden lógico de funcionamiento del producto, presentando al usuario final.

Crear artículos y contenidos en el proyecto

1. Ingrese a la consola de Azure DevOps, en el menú principal seleccione la opción `Repos`(repositorio).
2. En el menú encabezado seleccione el repositorio del proyecto/manual en curso. Ejemplo `docs-bpd`.
- 📌 **Nota:** Antes de modificar el artículo, cree una rama para registrar los cambios.[Ver creación de Ramas en un Proyecto](#)
3. Para crear una nuevo artículo, En la vista de información seleccione una carpeta del proyecto en `_pages` y en la opción `more options`, seleccione `New/File`.



3. Ingrese el nombre del artículo y haga clic en el botón `Create`.

New file

New file name

/_pages/configuracion_repo...

nuevo_articulo.md

Use slashes to create multiple subfolders like "sub/folder".

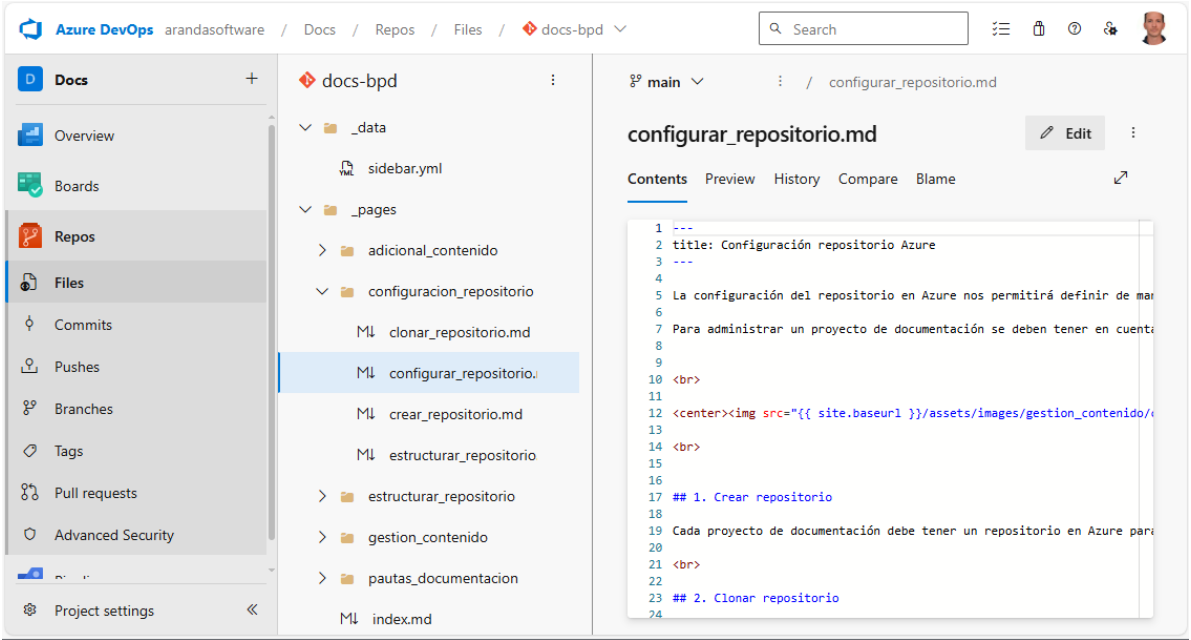
Cancel

Create

RECOMENDACIONES:

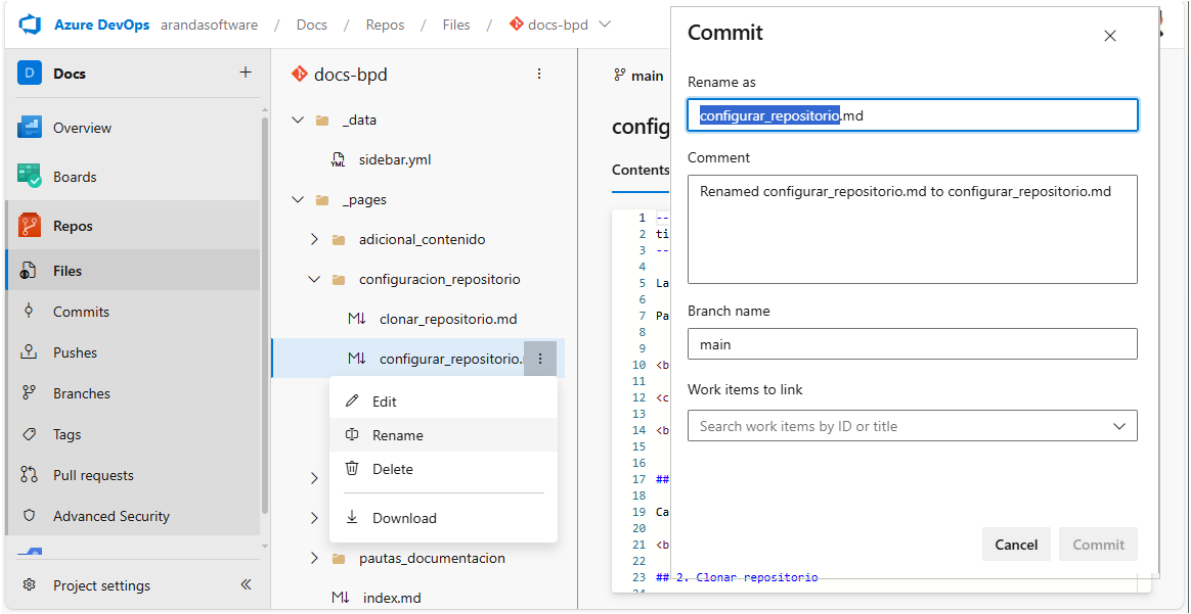
- El nombre del artículo debe ser corto, claro y corresponder al tema tratado.
- El nombre del artículo debe tener la extensión “.md” (markdown) para ser reconocido al publicar el manual.
- El nombre del artículo No debe tener espacios entre las palabras.
- El nombre del artículo debe estar escrito en minúsculas.
- Para unir palabras del nombre del artículo podrá utilizar guion o raya al piso (– _).
- El nombre del artículo No debe incluir caracteres especiales, ni tildes.

4. En la vista detalle del artículo creado, ingrese el encabezado del documento ([Ver Encabezado Articulos](#) y el contenido requerido (imágenes, texto, tablas) y los comentarios respectivos. Haga clic en el botón Commit para guardar los cambios propuestos.



Renombrar Artículo

5. Para renombrar un artículo, en la vista de información seleccione la carpeta `_pages` y un artículo creado; en la opción more options, seleccione `rename`. Se habilita la ventana para actualizar el nombre del artículo.



Eliminar una Artículo

5. Para eliminar una carpeta, en la vista de información seleccione la carpeta `_pages` y un artículo creado; en la opción more options, seleccione `Delete`. Se habilita la ventana para eliminar el artículo del repositorio.

6. Después de crear, renombrar o eliminar el artículo, [genere el Pull Request](#) para implementar los ajustes en el proyecto.

Importante a Tener en Cuenta

- La correcta gestión de carpetas, artículos e imágenes, epermitea optimizar los procesos actualización, revisión y cambios en el proyecto de documentación.
- [Agregar contenido \(imágenes, texto, tablas\)](#) en un proyecto de Documentación.

Branding - Estilos del proyecto

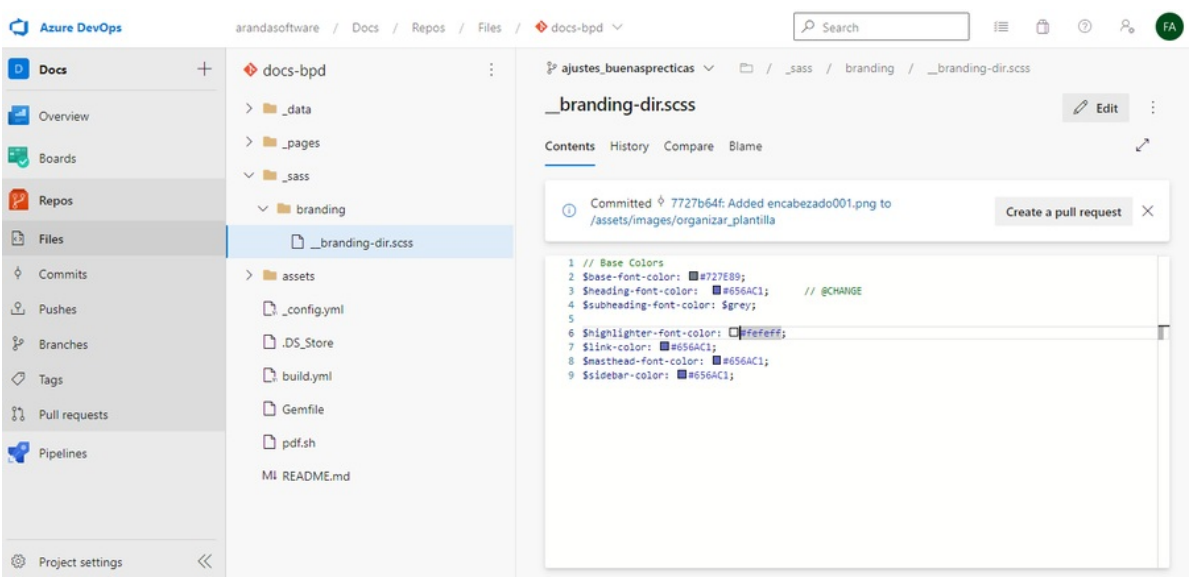
La modificación de la información del **branding**, permite configurar y establecer los estilos, colores, logos y plantillas que pertenecen a un proyecto (manual) específico. Definir los estilos refuerza la identidad gráfica de cada producto.

Para personalizar los estilos del branding de acuerdo al producto, debe tener en cuenta lo siguiente:

1. Ingrese a la consola de Azure DevOps, en el menú principal seleccione la opción **Repos** (repositorio).
2. En el menú encabezado seleccione el repositorio del proyecto en curso. Ejemplo docs-bpd.

📌 **Nota:** Antes de modificar la información de los estilos del proyecto, cree una rama para registrar los cambios. [Ver creación de Ramas en un Proyecto](#)

3. En la estructura de carpetas del proyecto seleccione la carpeta **_sass**, despliegue la información y seleccione la carpeta **branding** y el archivo **__branding-dir.scss** .



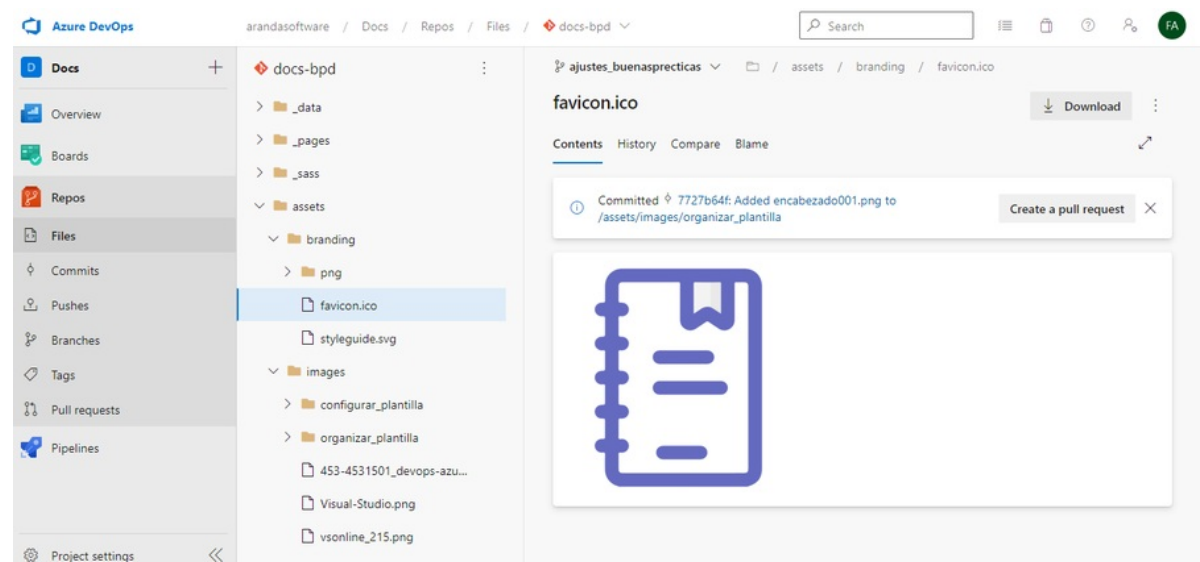
4. En la vista de detalle de los estilos **__branding-dir.scss** modifique los colores de acuerdo al estilo que corresponde por producto y al terminar haga clic en el botón **Commit** para guardar los cambios propuestos.
5. Después de ajustar el branding, [genere el Pull Request](#) para implementar los ajustes en el proyecto.

Catálogo colores por producto

Colores por Producto	Logos
Aranda Service Management	HEX #00A1C6.
Aranda Enterprise Mobility Management	HEX #00A1C6
Aranda Field Service	HEX #00a99d
Aranda Query Manager	HEX #73a1d1
Aranda Service Management	HEX #f4846e
Aranda Database Tools	HEX #33058d
Aranda Virtual Agent	HEX #656AC1
Integraciones Aranda	HEX #656AC1
Aranda Datasafe	HEX #04A99D
Aranda Pass Recovery	HEX #425B68
Aranda	HEX #656AC1

Logos

1. En la estructura de carpetas del proyecto seleccione la carpeta **assets** , despliegue la información y seleccione la carpeta **branding** y el archivo **favicon.ico** para reemplazar o incluir los logos asociados al producto.



[Link descarga](#)

assets/images

La opción **images** en Azure DevOps permite organizar y estructurar las imágenes, pantallas, gráficos, videos que serán referenciados en los diferentes artículos del proyecto. Antes de adjuntar los archivos es necesario definir la estructura para organizar las imágenes.

Cómo organizar Carpetas de imágenes en el proyecto

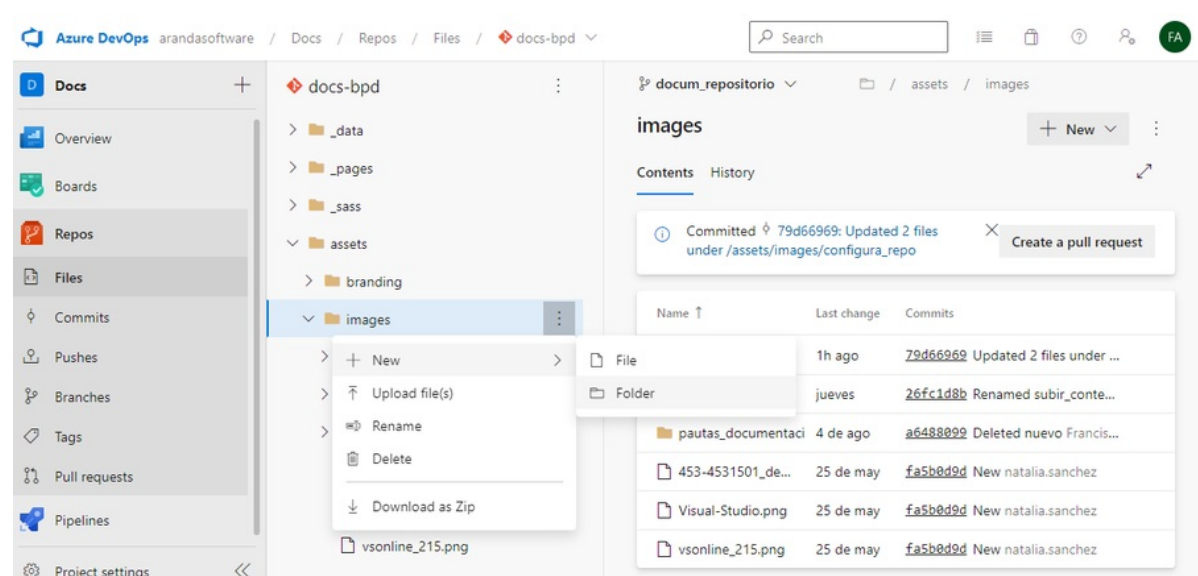
- 1. Ingrese a la consola de Azure DevOps, en el menú principal seleccione la opción **Repos** (repositorio).
- 2. En el menú encabezado seleccione el repositorio del proyecto/manual en curso. Ejemplo **docs-bpd**.

📌 **Nota:** Antes de modificar la carpeta, cree una rama para registrar los cambios. [Ver creación de Ramas](#)

en un Proyecto

Nota: Las imágenes se podrán organizar por carpetas agrupadas de acuerdo a los conceptos requeridos utilizando la estructura definida en la sección de Folders (carpetas).

3. Para crear una nueva carpeta de imágenes, en la vista de información seleccione la carpeta **assets**; al desplegarla elija la carpeta **images** y en la opción **more options**, seleccione **New/Folder**.



4. Ingrese el nombre de la carpeta de imágenes y el nombre del primer archivo de imagen o video asociado, y haga clic en el botón **Create**.

New folder

New folder name

/assets/images/

carpeta_imagenes_configuracion

Use slashes to create multiple subfolders like "sub/folder".

New file name

imagen.png

Git folders cannot be empty, so a placeholder file will be added. Its content can be edited before commit.

Cancel

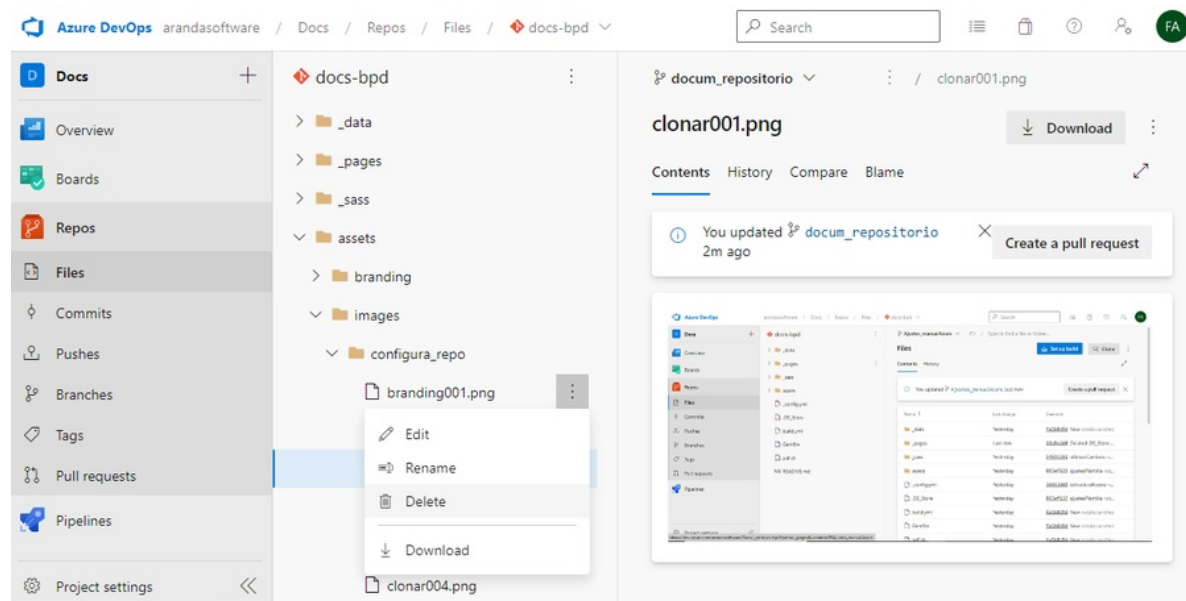
Create

EJEMPLO:

- 1. El nombre de las carpetas de imágenes deben coincidir con el nombre del capítulo o chapter que agrupa los contenidos o conceptos de la funcionalidad.
- 2. El nombre de la carpeta debe tener el consecutivo numérico y el nombre que identifique el capítulo, así:
 - 01-nombre_carpeta
 - 02-nombre_carpeta
 - 03-nombre_carpeta

5. En la vista detalle del archivo de la imagen creada, haga clic en el botón **Commit** para guardar los cambios propuestos.

6. Para visualizar una imagen, en la vista de información seleccione la carpeta **assets** y en la carpeta **images** seleccione una carpeta de imagenes del proyecto y haga clic en la imagen que desea visualizar o eliminar .



7. Para renombrar una carpeta de imágenes, en la vista de información seleccione la carpeta **assets** y una carpeta creada; en la opción **more options**, seleccione **rename**. Se habilita la ventana para actualizar el nombre de la carpeta. Actualice el nombre y en la ventana que se habilita haga clic en el botón **Commit**.

8. Después de crear o renombrar la carpeta de imágenes, [genere el Pull Request](#).

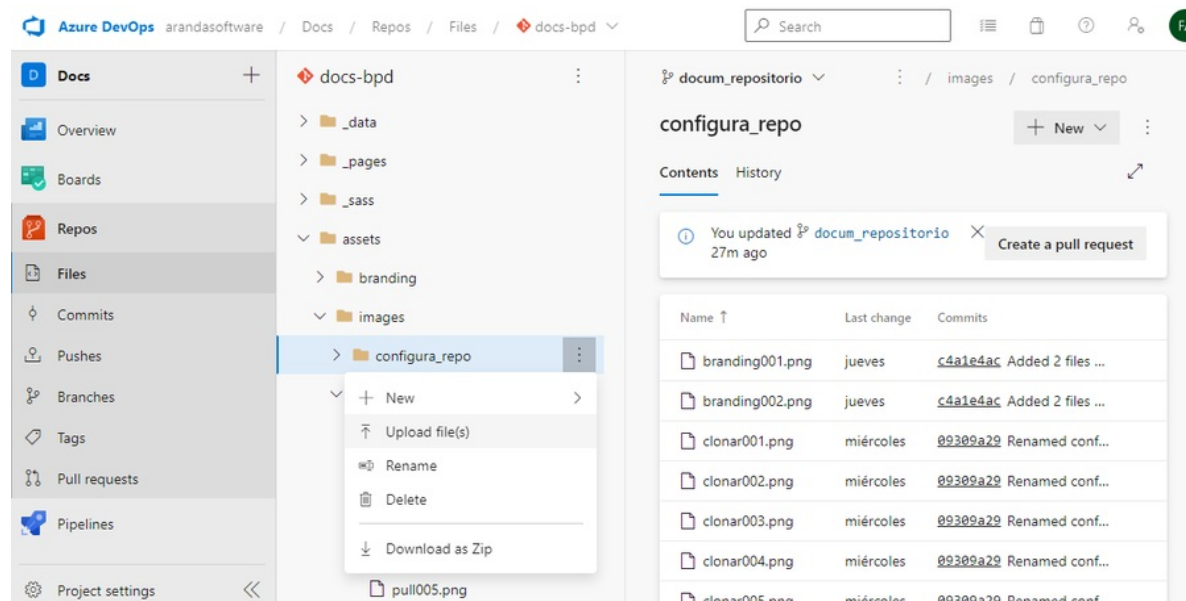
Cómo subir imágenes al proyecto

1. Ingrese a la consola de Azure DevOps, en el menú principal seleccione la opción **Repos** (repositorio).

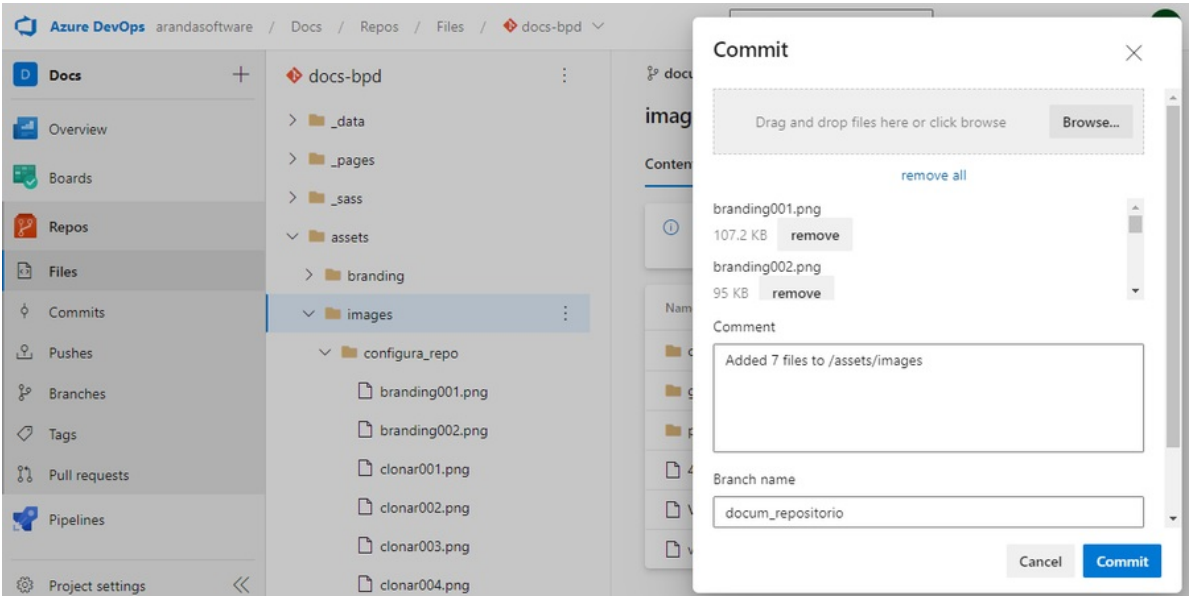
2. En el menú encabezado seleccione el repositorio del proyecto/manual en curso. Ejemplo **docs-bpd**.

✎ **Nota:** Antes de modificar el artículo, cree una rama para registrar los cambios. [Ver creación de Ramas en un Proyecto](#)

3. Para agregar imágenes al proyecto, en la vista de información seleccione la carpeta **assets**; al desplegarla elija la carpeta **images** o una carpeta de imágenes adicionada y en la opción **more options**, seleccione la opción **upload Files**.



4. En la ventana que se habilita seleccione la imagen o imágenes que requiere agregar al proyecto y haga clic en el botón **Commit** para guardar los cambios propuestos.



5. Para renombrar o borrar una imagen, en la vista de información seleccione la carpeta **assets**, al desplegarla elija la carpeta **images** y una imagen creada; en la opción more options, seleccione **rename** o **delete**. En la ventana que se habilita haga clic en el botón **Commit** para guardar los cambios propuestos.

6. Después de agregar o renombrar una imagen, [genere el **Pull Request**] para implementar los ajustes en el proyecto.

Importante a Tener en Cuenta

- El correcto uso de la numeración en las carpetas y archivos de imágenes, es importante para optimizar los procesos actualización, revisión y cambios en el proyecto de documentación.
- [Manejo de imágenes en un proyecto de Documentación](#)

_Config.yml/Actualización

La modificación del archivo **_config.yml** requiere el ajuste del título y url del proyecto en el que se este trabajando. De acuerdo a cada producto se ajusta esta información en el manual que se va a implementar.

1. Ingrese a la consola de Azure DevOps, en el menú principal seleccione la opción **Repos** (repositorio).

2. En el menú encabezado seleccione el repositorio del proyecto en curso. Ejemplo **docs-bpd**.

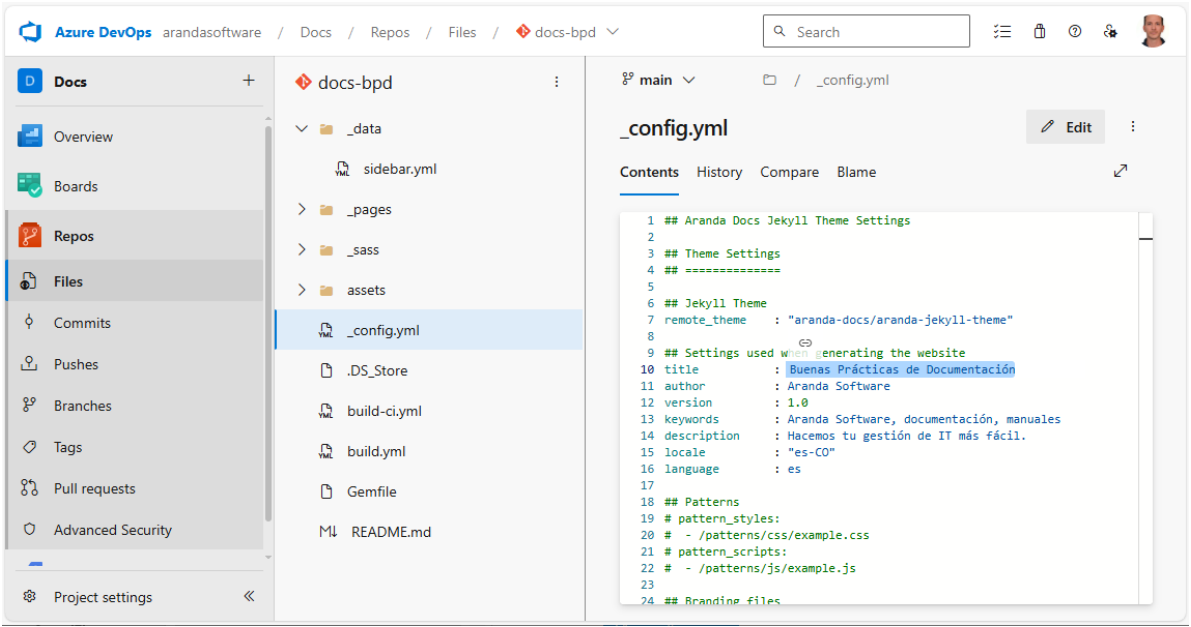
📌 **Nota:** Antes de modificar el nombre y la url del sitio, cree una rama para registrar los cambios.[Ver creación de Ramas en un Proyecto](#)

3. En la estructura de carpetas del proyecto seleccione el archivo **_config.yml**

Actualizar Título del proyecto

4. En la vista de detalle del **_config.yml** seleccione la pestaña **Contents**, en la sección **## Settings used when generating the website**, modifique la opción **Title** actualizando el nombre del producto.

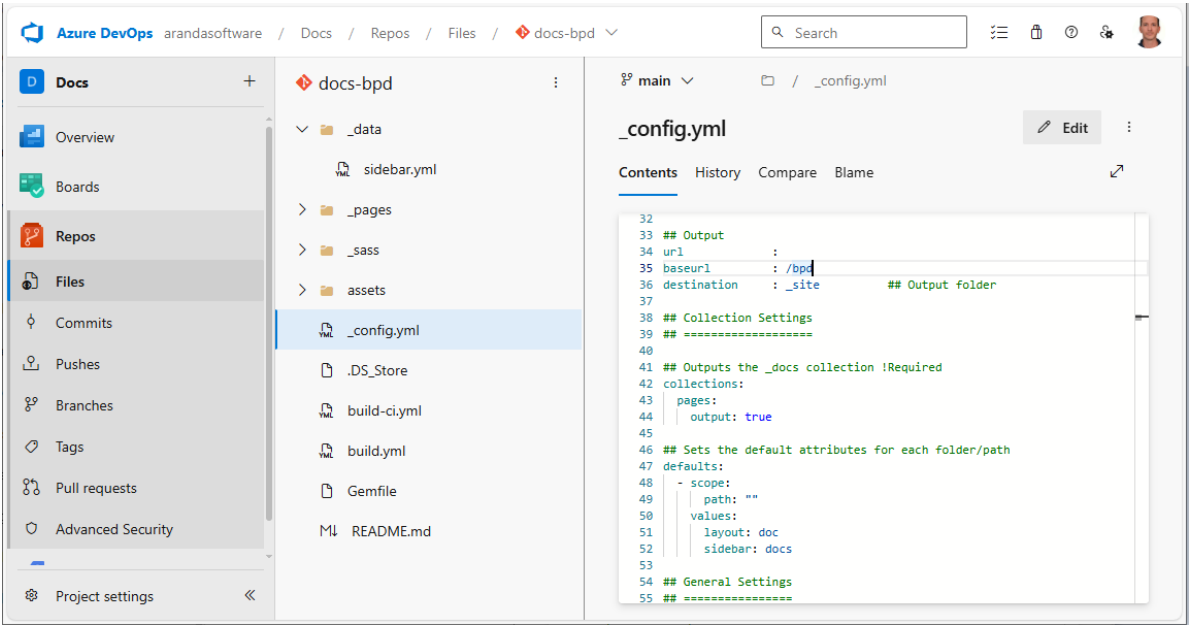
📌 **Ejemplo:** *title : Buenas Prácticas de Documentación*



Actualizar URL del proyecto

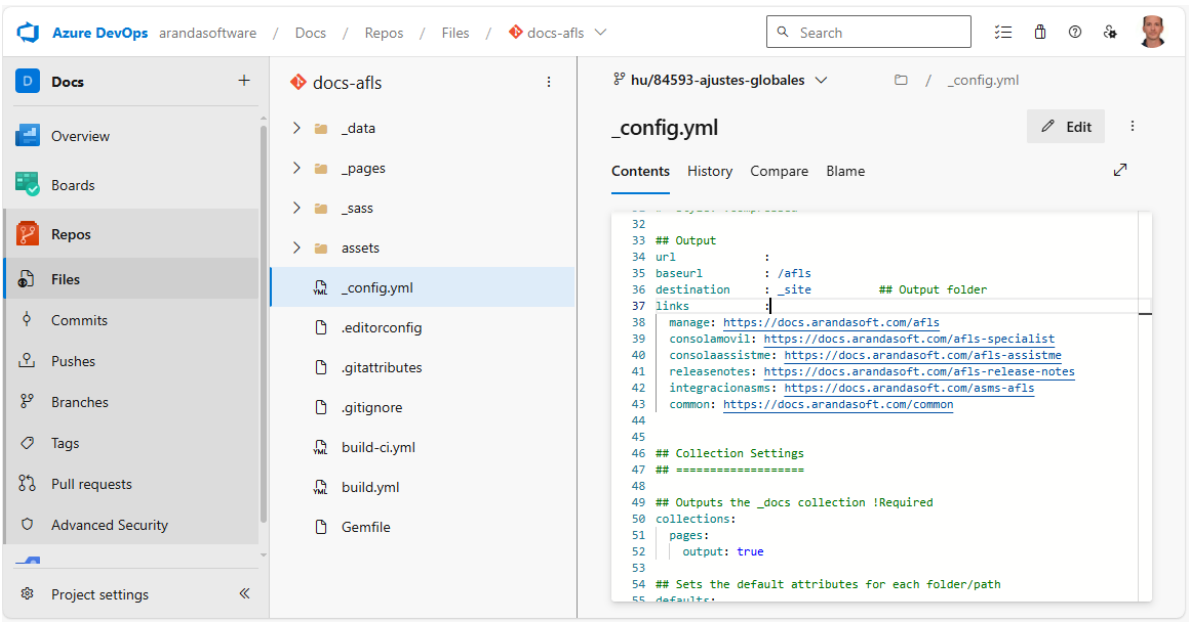
5. En la vista de detalle del `_config.yml` seleccione la pestaña **Contents**, en la sección **## Output**, modifique la opción `baseurl` actualizando la ruta base de acceso al manual del producto.

Ejemplo: `baseurl : /bpd`



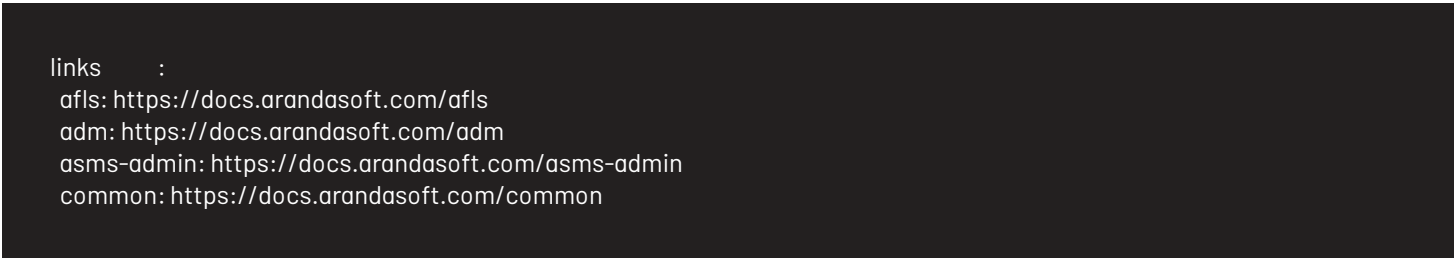
Incluir variables para enlaces externos

1. Si requiere realizar links (enlaces) externos entre manuales (repositorios) de productos de Aranda, en la vista de detalle del `_config.yml` seleccione la pestaña **Contents**, dentro de la sección **## Otput**, agregue el nombre `Links` y relacione las variables para los enlaces externos.



2. La estructura de los links sera así: `baseurl:url`

Estructura de los links con variables en `_config.yml`



2. La estructura de los links sera así: `baseurl:url`

Estructura de los links con variables desde un artículo:

[Módulos Aranda Commons]({{site.links.common}}/pages/configuracion_general/modulos_common.html)

Resultado con links con variables desde un artículo:

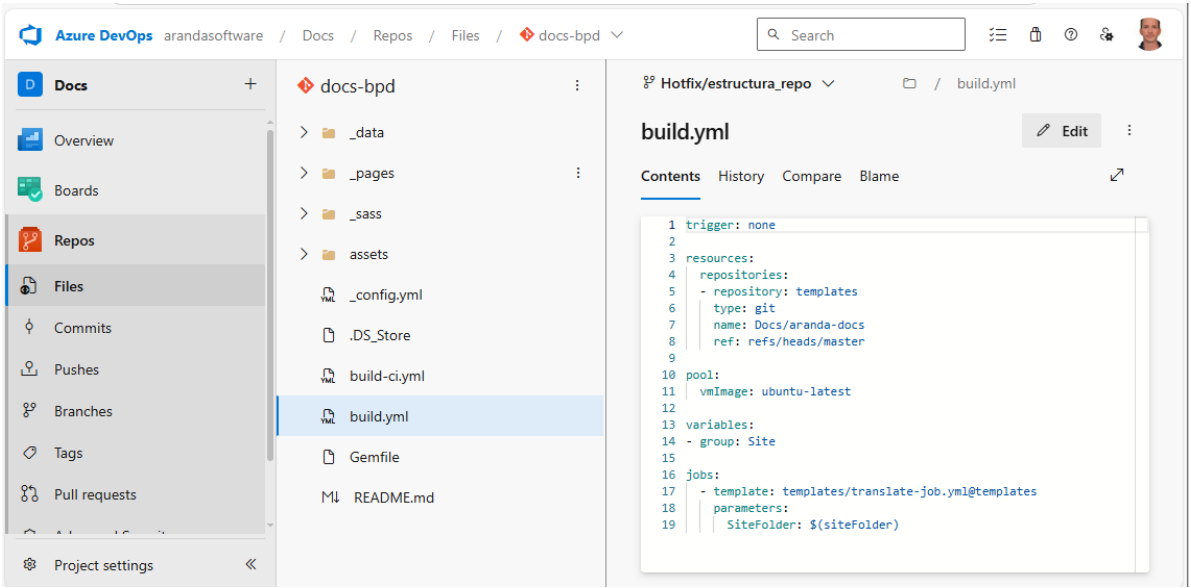
build.yml/Actualización

La modificación del archivo build.yml es requerida, si no tiene configurados los valores sugeridos.

1. Ingrese a la consola de Azure DevOps, en el menú principal seleccione la opción Repos (repositorio).
2. En el menú encabezado seleccione el repositorio del proyecto en curso. Ejemplo docs-bpd.

📌 **Nota:** Antes de actualizar el el archivo, cree una rama para registrar los cambios. [Ver creación de Ramas en un Proyecto](#)

3. En la estructura de carpetas del proyecto seleccione el archivo build.yml



4. En la vista de detalle seleccione la pestaña Contents, actualizando el archivo buil.yml . Haga clic en el botón Commit para guardar los cambios propuestos.

La información que debe estar actualizada en el archivo build.yml es la siguiente:

```
trigger: none

resources:
  repositories:
    - repository: templates
      type: git
      name: Docs/aranda-docs
      ref: refs/heads/master

pool:
  vmImage: ubuntu-latest

variables:
  - group: Site

jobs:
  - template: templates/translate-job.yml@templates
    parameters:
      SiteFolder: $(siteFolder)
```

5. Después de actualizar e archivo, debe [generar el Pull Request](#).

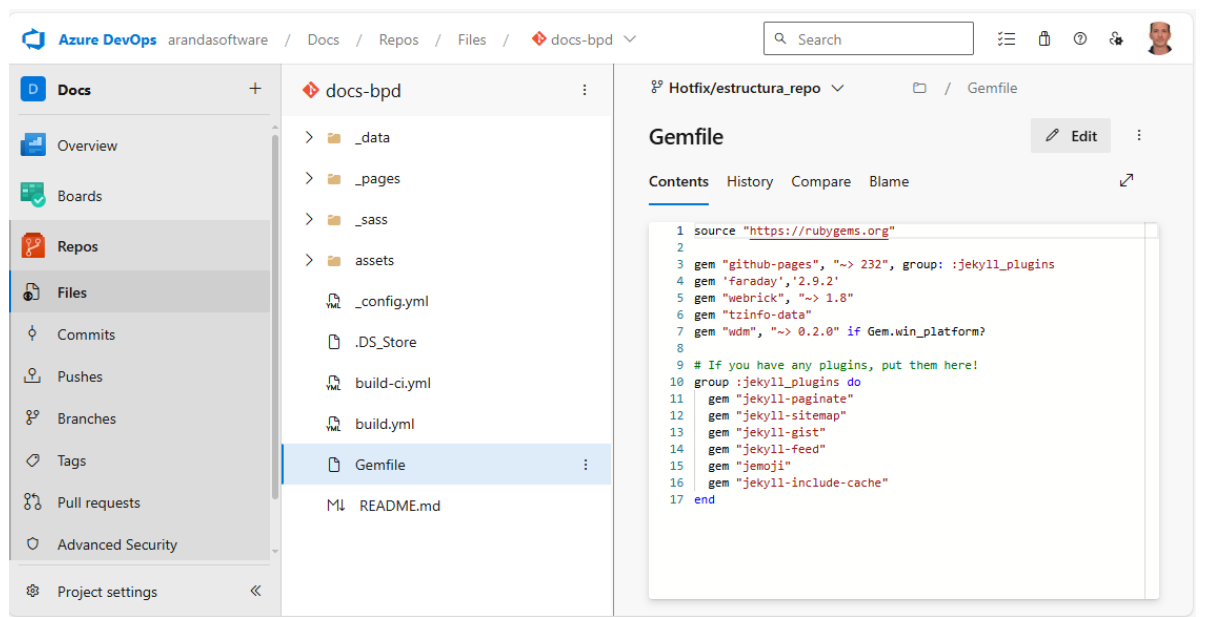
Gemfile/Actualización

La modificación del archivo Gemfile es requerida, si no tiene configurados los valores sugeridos.

1. Ingrese a la consola de Azure DevOps, en el menú principal seleccione la opción Repos (repositorio).
2. En el menú encabezado seleccione el repositorio del proyecto en curso. Ejemplo docs-bpd.

📌 **Nota:** Antes de actualizar el el archivo, cree una rama para registrar los cambios. [Ver creación de Ramas en un Proyecto](#)

3. En la estructura de carpetas del proyecto seleccione el archivo Gemfile



4. En la vista de detalle seleccione la pestaña **Contents**, actualizando el archivo Gemfile. Haga clic en el botón **Commit** para guardar los cambios propuestos.

La información que debe estar actualizada en el archivo **_Gemfile** es la siguiente:

```
source "https://rubygems.org"

gem "github-pages", "~> 232", group: :jekyll_plugins
gem 'faraday','2.9.2'
gem "webrick", "~> 1.9"
gem "rexml", ">= 3.3.9"
gem "tzinfo-data"
gem "wdm", "~> 0.2.0" if Gem.win_platform?

# If you have any plugins, put them here!
group :jekyll_plugins do
  gem "jekyll-paginate"
  gem "jekyll-sitemap"
  gem "jekyll-gist"
  gem "jekyll-feed"
  gem "jemoji"
  gem "jekyll-include-cache"
end
```

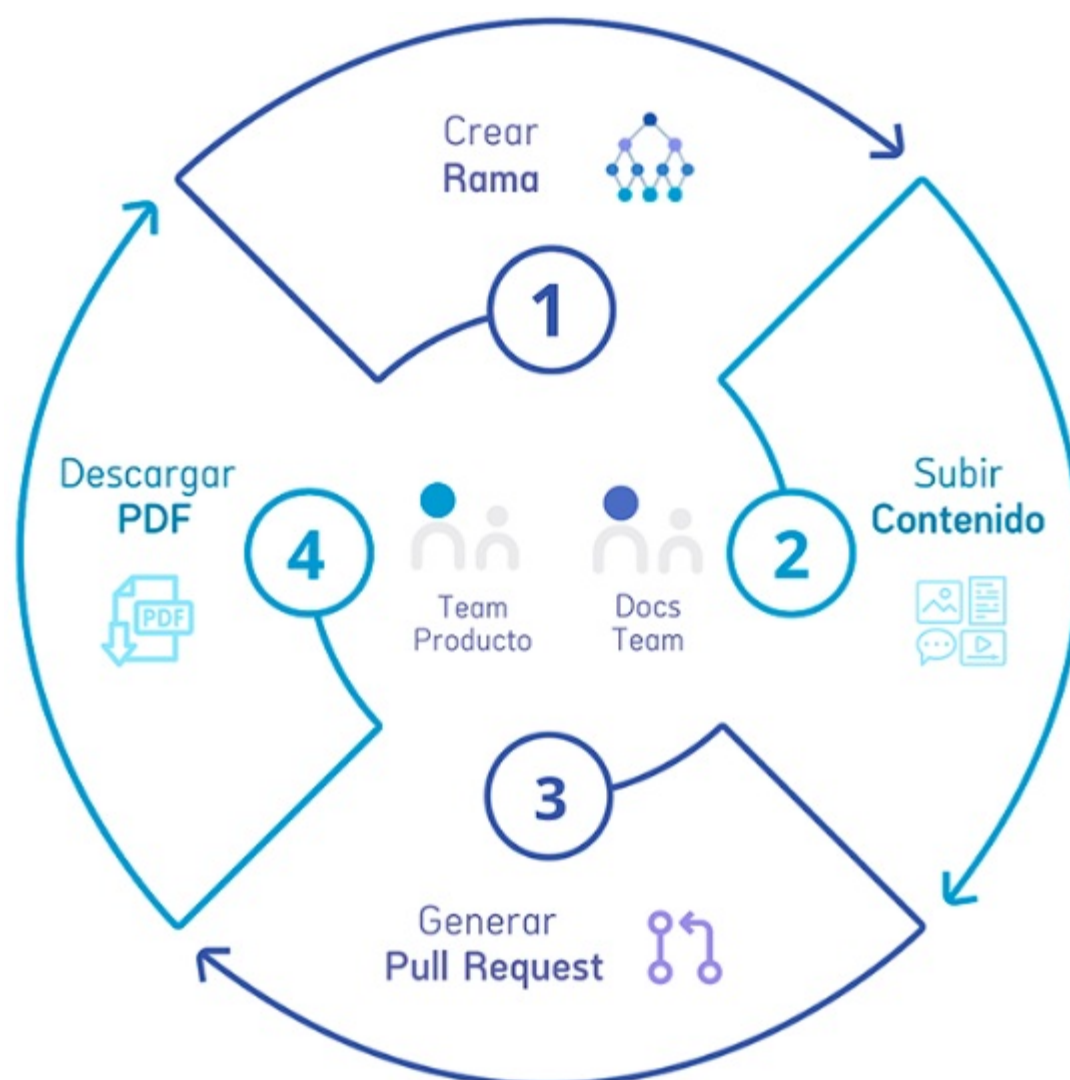
5. Después de actualizar e archivo, debe [generar el Pull Request](#).

Gestión de Contenido

Gestión de contenido en un Proyecto de documentación

La administración de los contenidos en un proyecto de documentación en un repositorio en Azure DevOps permite realizar las modificaciones (creación o edición de artículos; creación o edición de carpetas; subir o borrar imágenes) actualización y mantenimiento de los manuales de usuario e información relacionada a cada uno de los productos de Aranda.

Definir los contenidos en un proyecto de documentación debe tener en cuenta las siguientes etapas:



1. Crear Rama

Las ramas en Azure DevOps son un recurso que permite la gestión organizada y sincronizada de un proyecto en Azure DevOps.

La rama principal (Main) aloja el contenido actualizado de todo el proyecto. Las actualizaciones al contenido son bifurcaciones de la rama principal y se comportan como versiones paralelas hasta la unificación del contenido.

Para cada cambio que se realice en el proyecto (creación o edición de artículos, carpetas, archivos e imágenes, modificación del sidebar) se debe crear una rama.

La gestión de las ramas en un entorno colaborativo (varios usuarios aportan) permite actualizar el contenido con una retroalimentación permanente (revisión, corrección) y calificada.

2. Subir contenidos

La información más relevante del funcionamiento del producto debe estar representada en el tipo de contenido que se entrega al usuario final. Descripciones del proceso, definiciones, imágenes de contexto, tablas de apoyo, videos, son algunos de los recursos que integran un proyecto de documentación.

3. Generar PullRequest

La solicitud de cambios (Pull Request) permite mantener un proyecto estable y actualizado, con los aportes permanentes de los usuarios involucrados.

Los usuarios registrados pueden crear solicitudes de incorporación de cambios (Pull Request) a partir de ramas generadas. Los equipos de trabajo que realizan sus aportes al proyecto, podrán revisar los Pull Request y agregar comentarios sobre los cambios, revisar el historial de cambios y aprobar o rechazar los cambios propuestos.

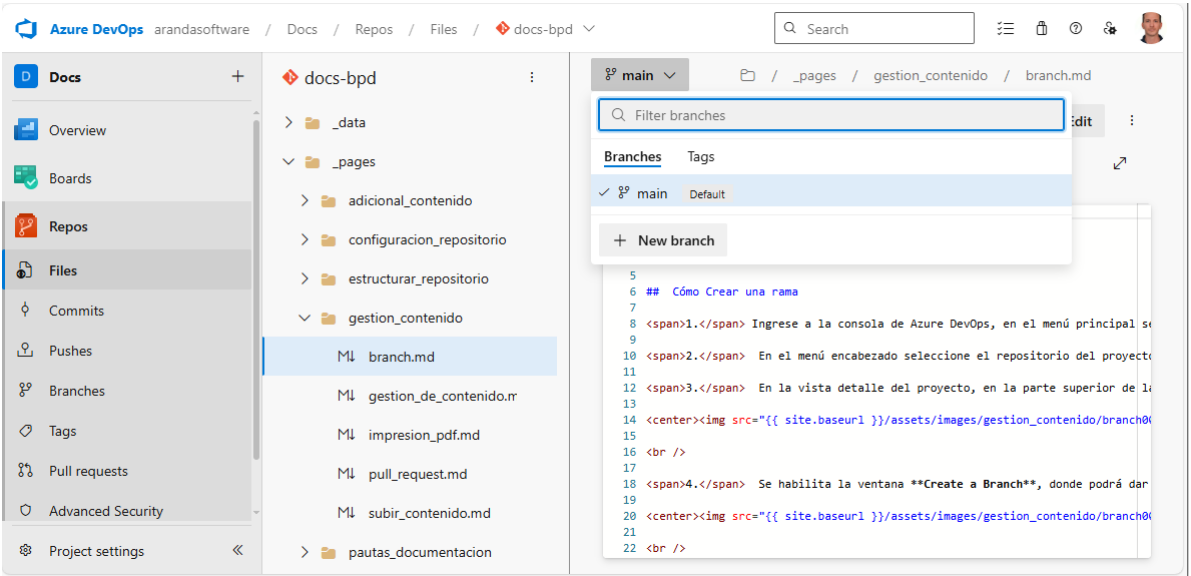
4. Impresión PDF

El Usuario siempre tendrá disponible la acción para imprimir en formato PDF el manual respectivo del proyecto de documentación (Manual de Usuario, manual de administración, guía de instalación, release notes, entre otros).

Branch- Ramas

Cómo crear una rama

1. Ingrese a la consola de Azure DevOps, en el menú principal seleccione la opción Repos (repositorio).
2. En el encabezado del menú, seleccione el repositorio del proyecto/manual en curso. Ejemplo docs-bpd.
3. En la vista detalle del proyecto, en la parte superior de la consola, podrá encontrar las ramas disponibles. Por defecto encontrará la rama principal (Main). En el menú que se despliega seleccione la opción New Branch.



4. Se habilita la ventana Create a branch, donde podrá dar un nombre a la rama, definir su rama base y asociar uno o varios Work items.
5. En el campo Name ingrese el nombre de la rama en relación con la funcionalidad que se documenta o modifica.

Create a branch

Name *

hotfix/ajustes_branch

Based on

release

Work items to link 1

Clear all

Search work items by ID or title

User Story 121019: Manual Buenas Prácticas

Updated viernes, Active

Cancel

Create

6. En el campo Based on, seleccione el tipo de rama (main,feature o release) acorde con la rama que se está creando.

Create a branch

Name *

feature/nombre_ajuste

Based on

 main

▼

Work items to link

1

[Clear all](#)

Search work items by ID or title

▼

 User Story 121019: Manual Buenas Prácticas

Updated Just now,

●

 Active

×

Cancel

Create

Los tipos de ramas definidos son:

Tipo de Rama	Propósito	Based on	Destino de Merge
main (dev)	Rama principal y base de todo cambio en el documento. Código estable y en progreso.	Inicial del repositorio	N/A
release/	Rama de publicación lista para entregar relacionada a la versión de producto a liberar o liberada.	main	N/A
feature/	Rama que reúne la documentación de nueva funcionalidad en el producto (Cambios).	main	main
hu/	Rama que se asocia a la documentación en relación a la historia de usuario que hace parte de la nueva funcionalidad.	feature/{abc}	feature/{abc}
hotfix/	Ajustes críticos en la versión publicada (release) que deben aplicarse inmediatamente.	release/x.y.z	release/x.y.z y con la main

📌 Notas:

- 1. Reemplaza {abc} por el nombre de la rama feature/.
- 2. Reemplaza x.y.z por el número de versión de release correspondiente.

Convenciones de Nomenclatura nombres de ramas

Para mantener consistencia y claridad en el repositorio:

Rama	Estructura	Ejemplo
release	release/{MAJOR}.{MINOR}.{PATCH}	release/1.2.0.
feature	feature/{ID-WI}-{descripción-corta}	feature/123-login-social.
hu	hu/{ID-WI}-{título-hu}	hu/456-mejoras-documentación.
hotfix	hotfix/{ID-WI}-{descripción}	hotfix/789-corrección-tipo.

Tip: Al nombrar las ramas, se recomienda utilizar guiones (-) y minúsculas, manteniendo los nombres

descriptivos pero concisos.

7. Al terminar haga clic en el botón **Create**.

📌 **Notas:**

- 1. La rama creada debe estar basada en ramas principales como main, feature o release.
- 2. Pueden existir múltiples ramas que evidencien cambios de diferentes usuarios sobre el proyecto. Es importante no mantener ramas abiertas en las que no se estén trabajando.
- 3. Antes de realizar una actualización sobre un archivo valide que no hayan usuarios trabajando en ese documento al mismo tiempo. Aunque estén haciendo cambios en ramas independientes, puede generarse un conflicto al momento de unificar el contenido.

📌 **ADVERTENCIAS:**

- 1. Sobre la rama main (dev o master) No se deben realizar cambios, esta rama unifica al final los cambios generados en ramas como release o feature.
- 2. La rama creada debe estar basada en ramas principales como **main**, **feature** o **release**.

Cómo Visualizar las ramas

- 1. Ingrese a la consola de Azure DevOps, en el menú principal seleccione la opción **Branches** (ramas).
- 2. En la vista detalle de ramas, en la parte superior de la consola, seleccione la opción **All** donde podrá crear nuevas ramas o ver la estructura creada para tipos de ramas como feature, hu, release, main.

Branches

New branch

Mine All Stale

Search branch name

Branch	C...	Author	Aut...	Behind Ahead	S...	P...
feature						
ajusteimagenes	5265	Nancy...	30 ...	0 0	✓	
Migrate_GlobalSettings	f22e	Julieth...	30 ...	14 0		
migrate-providers	7126	Nancy...	26 ...	19 9		
hu						
96392-conf-dias-festivos	7668	Julieth...	28 ...	19 2		1
release						
9.22.0	571f	Nancy...	14 ...	24 0	✓	
main	5265	Nancy...	30 ...		✓	★
master	5b64	Jerem...	9 d...	650 0	✓	

📌 **Buenas practicas:**

- 1. Si una rama remota ya no es requerida, ya sea porque se haya realizado el merge o cerrado un Pull Request, se debe eliminar. Lo recomendable es eliminarla durante el Pull Request.
- 2. Al fusionar la rama al entorno principal main, feature/ o release/, se podrá eliminar de forma local con seguridad para evitar confusión.

Documentación relacionada en Azure Devops

- [Acerca de las Ramas](#)
- [Estrategia de Bifurcación](#)
- [Crear una Rama](#)

Subir contenido al Proyecto

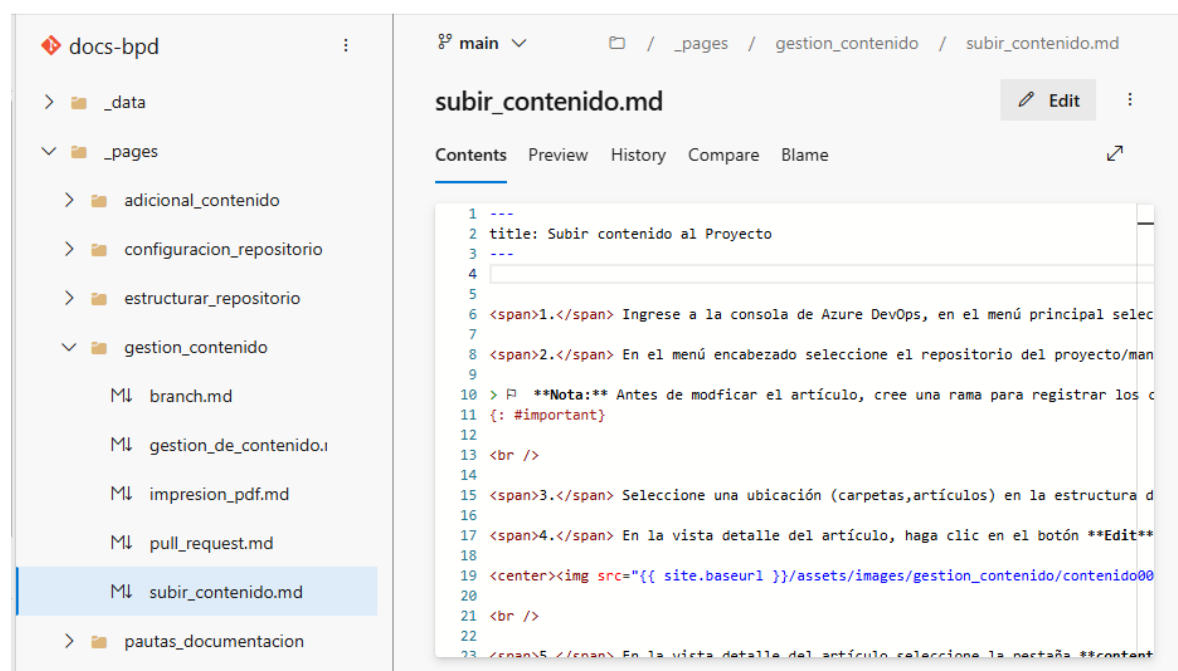
- 1. Ingrese a la consola de Azure DevOps, en el menú principal seleccione la opción **Repos** (repositorio).

2. En el menú encabezado seleccione el repositorio del proyecto/manual en curso. Ejemplo docs-bpd.

📌 **Nota:** Antes de modificar el artículo, cree una rama para registrar los cambios. [Ver creación de Ramas en un Proyecto](#)

3. Seleccione una ubicación (carpetas, artículos) en la estructura del proyecto, donde subir el contenido [Ver Creación Artículos y Carpetas](#)

4. En la vista detalle del artículo, haga clic en el botón Edit para agregar el contenido requerido (texto, imágenes, videos, tablas).



5. En la vista detalle del artículo seleccione la pestaña contents para activar la vista de Edición e incluir la información requerida. El contenido del artículo puede variar pero su estructura debe tener en cuenta los siguientes elementos:



Tipo de Contenido

La información que puede relacionar en una página o artículo de un proyecto, se puede remitir a texto o a diferentes recursos que complementan el contenido como:

Tipo	Descripción
Encabezado	El encabezado del artículo está compuesto por el Título del artículo (title) y pueden estar escritos en minúscula/mayúscula y llevar tildes. El nombre del título debe resumir las características de la funcionalidad.
Numeración	La secuencia de pasos permite describir una funcionalidad de una manera ordenada y orienta al usuario en el manejo de la funcionalidad. Conozca cómo implementar las etiquetas markdown de Numeración con span o Numeración con un dígito
Imágenes	Las imágenes, pantallas y gráficos son complemento al contenido para ampliar o contextualizar la descripción de la funcionalidad. Conozca las Recomendaciones en el manejo de imágenes en un proyecto de documentación
Tablas	El uso de las tablas permite organizar y clasificar la información de la funcionalidad, así como resaltar un contenido específico. Ejemplo: En la descripción de los campos de un formulario. Conozca cómo implementar las etiquetas markdown para Tablas
Enlaces	Los links en un artículo permiten relacionar documentación interna o externa que complementa el contenido expuesto o es un elemento de continuidad en el documento. Con las etiquetas de markdown podrá realizar links entre artículos , links externos , links externos con variables y anclas dentro de un artículo.
Notas	Las notas son mensajes que permiten generar una advertencia o recomendación en la documentación de la funcionalidad. Con las etiqueta tipo Nota de markdown podrá realizar Ejemplos , Notas , Advertencias , Enlaces Relacionados y Definiciones

6. En la vista detalle del artículo seleccione la pestaña Preview para activar la vista previa del documento.

bpd_subir_conten... / _pages / 03-gestion_contenido / 03-subir_contenido.mc

03-subir_contenido.md

CommitRevert

ContentsPreviewHighlight changes

title	excerpt	chapter
Subir contenido al Proyecto	Subir	gestion_contenido

Subir contenido

1. Ingrese a la consola de Azure DevOps, en el menú principal seleccione la opción **Repos (repositorio)**.

2. En el menú encabezado seleccione el repositorio del proyecto/manual en curso. Ejemplo docs-bpd.

Nota

Antes de modificar el artículo, cree una rama para registrar los cambios. [Ver creación de Ramas en un Proyecto](#)

3. Seleccione una ubicación (carpetas, artículos) en la estructura del proyecto, donde subir el contenido [Ver Creación Artículos y Carpetas](#)

4. En la vista detalle del artículo, haga clic en el botón **Edit** para agregar el contenido requerido (texto, imágenes, videos, tablas)

5. En la vista detalle del artículo seleccione la pestaña **contents** para activar la vista de Edición e incluir la información requerida. El contenido del artículo puede variar pero su estructura debe tener en cuenta los siguientes elementos:

Sección	Descripción
Encabezado	El encabezado del artículo está compuesto por el Título del artículo (title) y el capítulo (chapter) al que pertenece.(Ver Encabezado Artículos)
Numeración	describir una funcionalidad de una manera ordenada y secuencial,

NumeraciónEtiquetasPantallas: Tablas

y el contenido requerido (imagenes, texto, tablas) y los comentarios respectivos. Haga clic en el botón **Commit** para guardar los cambios propuestos.

7. Después de agregar el contenido, haga clic en el botón Commit para guardar los cambios propuestos.
8. Después de crear o renombrar el artículo, [genere el Pull Request](#) para implementar los ajustes en el proyecto.

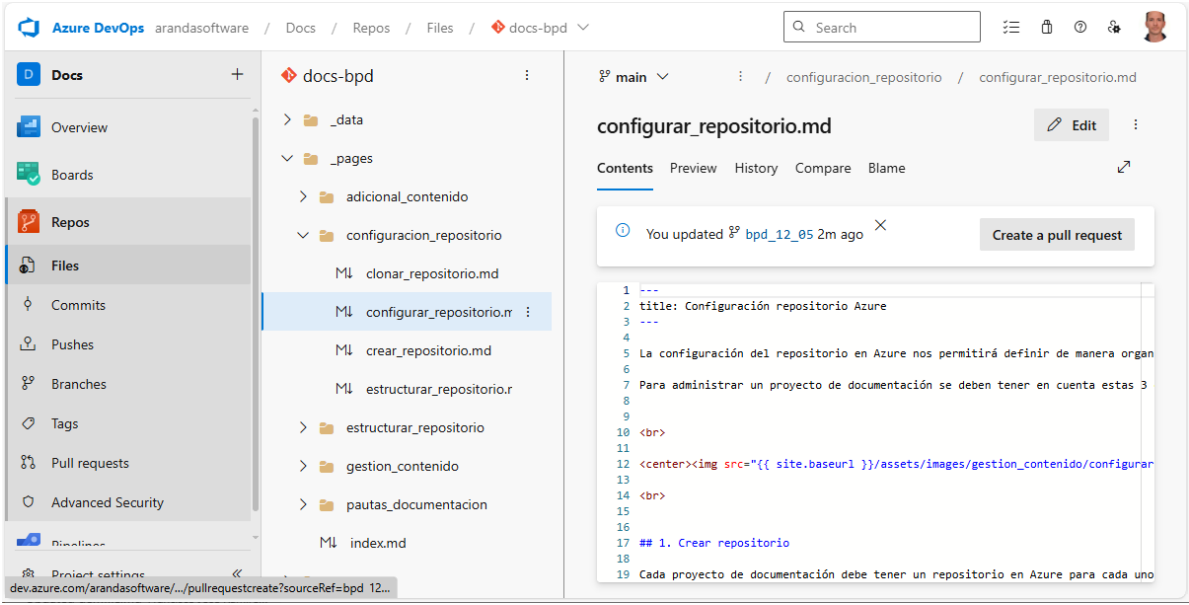
Enlaces importantes a referenciar en este artículo

- Definir encabezado. [Ver Encabezado Artículos](#)
- Recomendaciones para subir imágenes al proyecto de documentación. [Ver Manejo de las imágenes](#)
- Recomendaciones para subir videos al proyecto de documentación. [Ver Manejo de las videos](#)
- Incluir imágenes, numeración, tablas o links en un artículo en formato markdown. [Ver Uso de etiquetas en un Artículo](#)

Pull Request

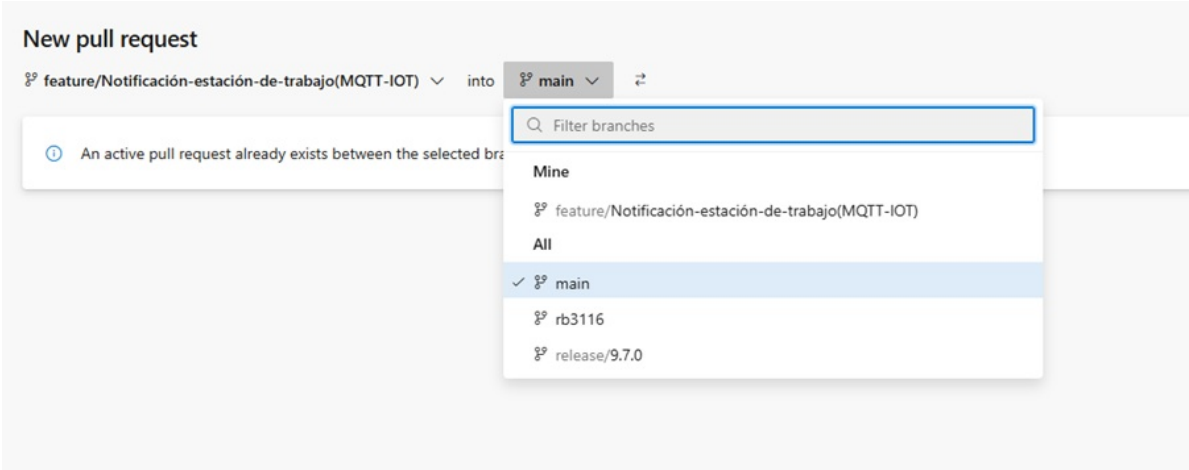
Generar un PR en un Proyecto

1. Después de realizar cambios o actualizaciones en la documentación del proyecto (carpetas, imágenes, artículos, archivos), en la vista de información del repositorio se habilita la opción **Create a pull request**. Seleccione esta opción para iniciar el proceso de enviar los cambios asociados a la rama creada y unificar la información en la rama base (main, feature/ o release/).



2. En el encabezado de la ventana **New pull request** configura la rama de origen y destino destino de acuerdo a la siguiente tabla:

Origen	Destino
feature/{abc}	main (master o dev)
hu/{abc}	feature/{abc}
hotfix/{abc}	release/{x.y.z} (no eliminar la rama; posteriormente se debe enviar hacia main)



New Pull Request complete los datos de la actualización: título, descripción (opcional), commits (obligatorio), archivos seleccionados para revisar los cambios y relacionar el o los Work Items. Al finalizar seleccione **Crear pull request**.

New pull request

bpd_12_05 into main ↺

Overview Files 4 Commits 2

Title

ajuste creacion pull request

Description Add commit messages

Describe the code that is being reviewed

0/4000

Markdown supported. Drag & drop, paste, or select files to insert. Link work items.

@ # 🔗 📎 ↵ **B** *I* ⚡ ☰ ☷ ☹

Reviewers Add required reviewers

Juan Pablo Jaramillo Bedoya × +

Work items to link 1 Clear all

Search work items by ID or title ▼

User Story 121019: Manual Buenas Prácticas
Updated 20m ago, ● Active ×

Tags

Create ▼

1. El responsable de generar el contenido y los usuarios relacionados para revisar las actualizaciones, recibirán una notificación por correo electrónico del Pull Request.
2. El equipo de documentación tendrá un rol de revisión del contenido (corrección de estilo, redacción y ajuste al formato de documentación) y posterior a su aprobación, podrá unificarse la información.

📌 **A tener en cuenta:** Para evitar conflictos durante el envío del Pull Request se recomienda incorporar los cambios de la rama base.

Azure DevOps | arandasoftware | Docs | Repos | Pull requests | docs-bpd

Docs +

- Overview
- Boards
- Repos
- Files
- Commits
- Pushes
- Branches
- Tags
- Pull requests**
- Advanced Security
- Pipelines
- Project settings

ajustes_bpd_pr Active #17968  Francisco Jose Ramirez Astudillo proposes to merge [bpd_12_05](#) into [main](#)

Overview | Files | Updates | Commits

 All required checks succeeded

[View 3 checks](#) ⓘ

 Required reviewers have approved

 No merge conflicts
Last checked 3m ago

Description 

Approve  **Complete** 

-  Complete
-  Mark as draft
-  Abandon

Reviewers **Add** 

Required

-   **Docs Team**
Approved via Francisco Jose Ra... 

Optional

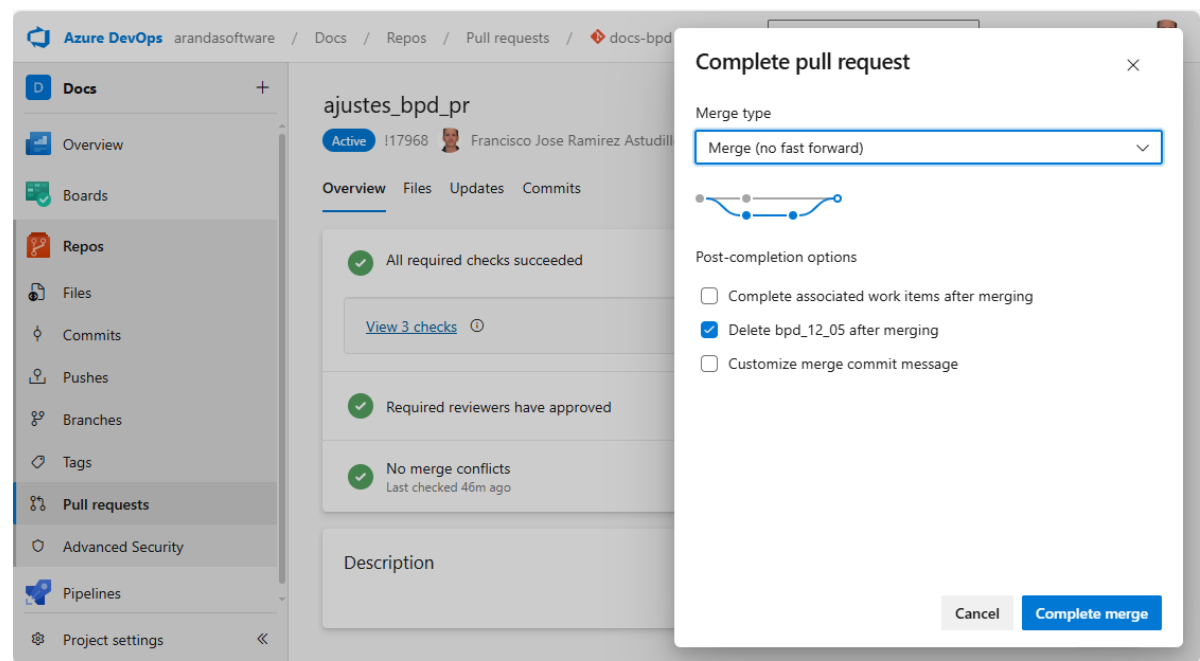
-   **Francisco Jose Ramirez Astudillo**
Approved

Tags **+**

No tags

Work items **+**

6. Se habilita la ventana **Complete Pull Request** donde debe unificar la información en la rama base. Mantener marcadas las opciones de **Complete** y **Delete**.



⚠ **Importante:** Al realizar el *merge* de hotfix/{abc} → release/{x.y.z}, desmarque la opción **Delete** para poder enviar un segundo Pull Request hacia la rama main.

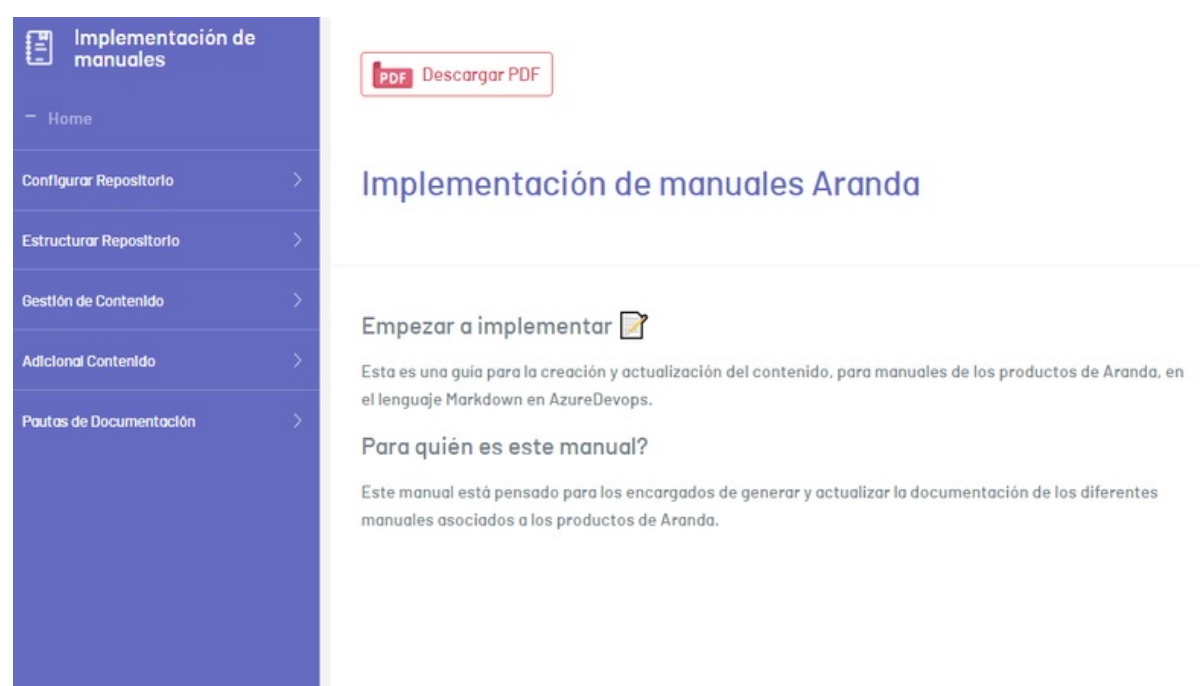
Documentación relacionada en Azure Devops

- [Creación y Requisitos de Pull Request en Doumentación Azure DevOps](#)
- [Revisión de Pull Request en Doumentación Azure DevOps](#)
- [Completar Pull Request en Doumentación Azure DevOps](#)

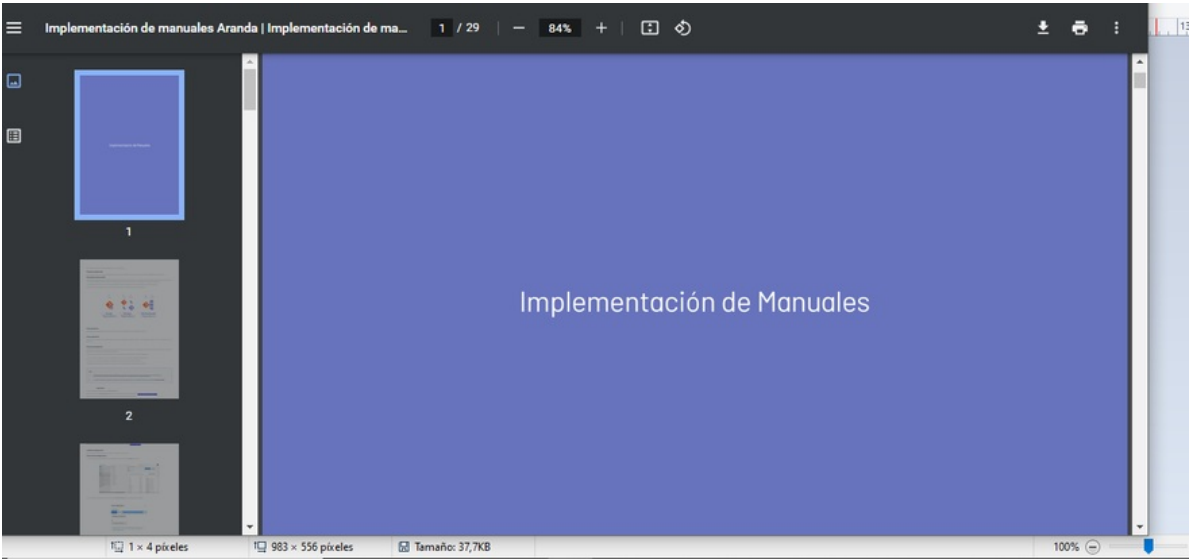
Impresión Manual en PDF

Terminado el proyecto de documentación el documentador de producto o un usuario final podrá consultar el manual en línea desde el portal de documentación o imprimir en PDF el contenido del mismo.

1. Ingrese en el [portal de documentación](#) y seleccione un producto del manual que requiere imprimir.
2. Seleccione el manual requerido.
3. En la plantilla de cada manual de producto, en la parte superior de cada uno de los artículos, seleccione la opción **Descargar PDF**



4. Se descarga de forma autotomática el manual de producto, para su impresión o consulta.

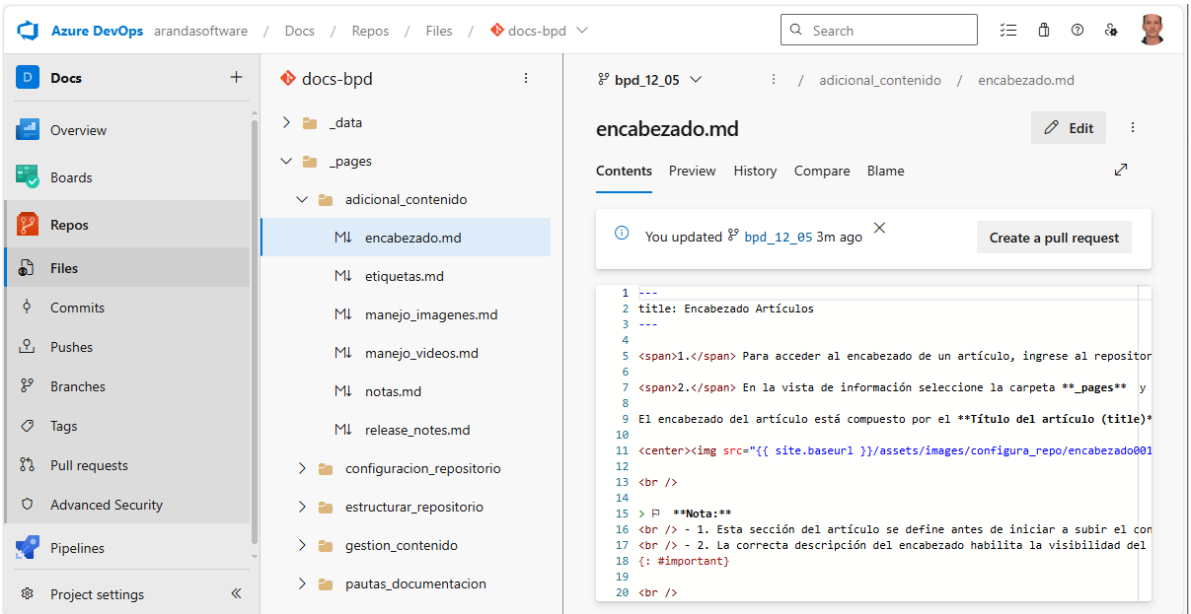


Adicional Contenido

Encabezado Artículos

1. Para acceder al encabezado de un artículo, ingrese al repositorio de su proyecto de documentación (ejemplo docs-bpd).
2. En la vista de información seleccione la carpeta `_pages` y defina una ubicación (carpetas) y un artículo, en la estructura del proyecto.

El encabezado del artículo está compuesto por el **Título del artículo (title)** al que pertenece.



Nota:

- 1. Esta sección del artículo se define antes de iniciar a subir el contenido, en el encabezado del documento.
- 2. La correcta descripción del encabezado habilita la visibilidad del artículo cuando el proyecto esté público.

Ejemplo



Importante a Tener en cuenta y Enlaces relacionados

- El nombre del artículo debe escribirse en minúsculas, sin tildes y separado con caracteres como guión o raya al piso (- _)
- [Agregar Tipo de Contenido a un proyecto de documentación](#)

Manejo de Imágenes

Las pantallas se utilizan como recurso para describir o complementar las características de una funcionalidad en la medida que sirvan de refuerzo y apoyo a los contenidos generados. Debe tener presente las siguientes recomendaciones:

Sentido común

Si las imágenes utilizadas, no aportan la descripción de la funcionalidad dándole contexto o si la información presentada en las pantallas no es visible para el usuario, es mejor no utilizarlas.

Tamaño

El uso adecuado de los tamaños de una pantalla o un gráfico, le aporta o le juega en contra a los contenidos que queremos presentar. La sugerencia del tamaño de las pantallas o imágenes a utilizar es el siguiente:

Tipo de Pantalla	Descripción	Tamaño sugerido	Uso
Consola completa	La pantalla de consola de producto agrupa los elementos principales de la funcionalidad (administración, configuración, monitoreo).	Máximo 900 Píxeles de ancho	Se puede utilizar para dar contexto general en dónde se encuentra el usuario. Usualmente al principio de un artículo y preferiblemente evitar replicar la imagen en el mismo documento.
Sección de la consola	Es una captura parcial de la funcionalidad (ventana, listados)	Máximo 500 píxeles de ancho	Se puede utilizar para destacar un concepto o un ítem especial y para darle continuidad al desarrollo del contenido
Instaladores	Pantalla describen el proceso de instalación del producto	Máximo 800 Píxeles de Ancho por 600 de Alto	Sólo para el manual de instalación.
Dispositivo Móvil	Pantalla que representa una consola móvil	Máximo 370 Píxeles x de Ancho por 660 Píxeles de Alto.	Imágenes utilizadas en los manuales donde se describe la consola móvil

Imagen Consola Completa

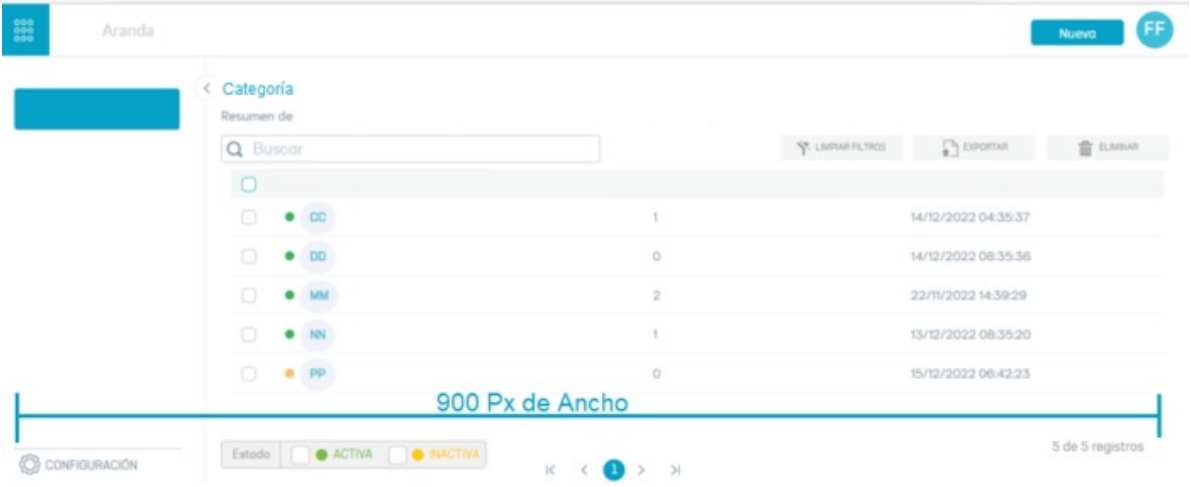


Imagen Sección Consola

Configuración

Sincronización

Unificación de campos

Ultima sincronización:
--/--/--- 00:00:00

Periodicidad

☐ Una vez

☐ Diariamente

☐ Semanalmente

☒ Mensualmente

Inicio:

2022-06-15

12:00:00

☐ Sincronizar zona horaria

Meses:

Enero

X

Abril

X

Julio

X

Octubre

X

☒ Días:

1

X

☐ El:

Imagen Instaladores

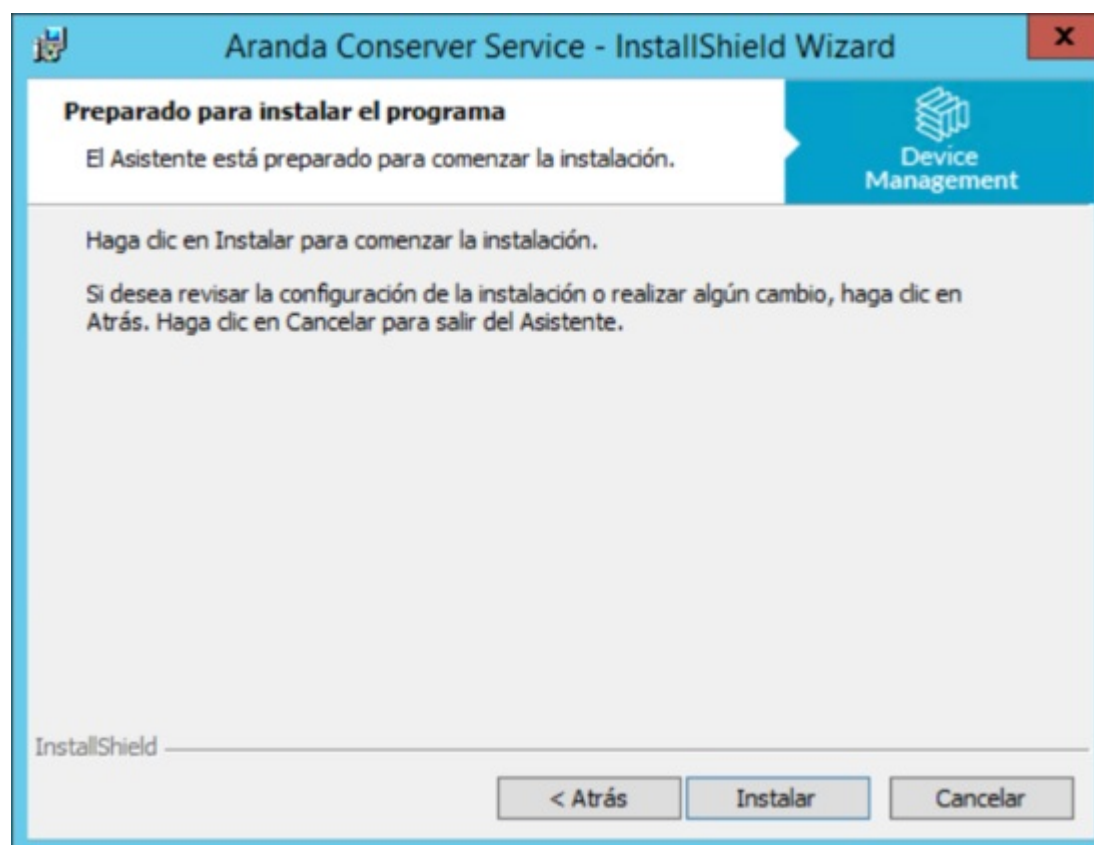


Imagen Dispositivo Móvil

Operator 5:23 p.m.

< Crear caso 1 →

Infraestructura
Infraestructura

Tipo*
Cambios

Asunto
Ingrese brevemente el asunto

Descripción
Ingrese brevemente la descripción

Ci
Digite por lo menos una letra

Psa
Selecione...

Scope
Escribe aquí...

BackOutPlan
Selecione...

5 → Agregar adjuntos

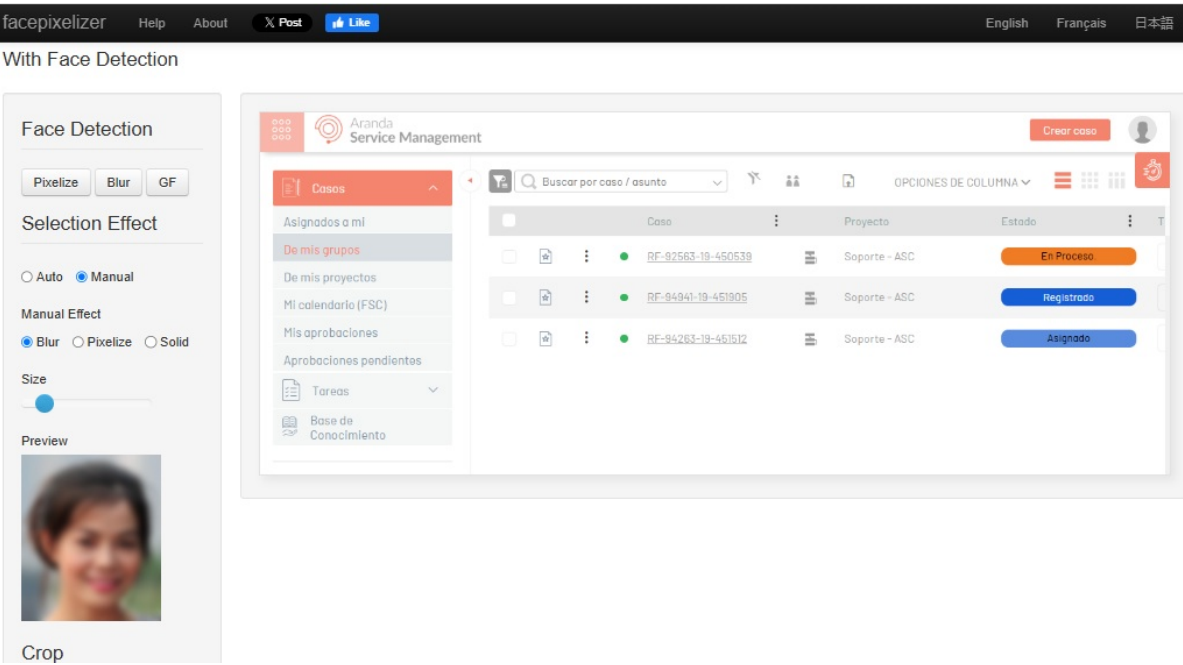
Ocultar información Sensible

Si la información registrada en pantallas utilizadas para ilustrar y describir una funcionalidad del producto, tienen información sensible para el cliente o para Aranda, se podrá difuminar la información relevante.

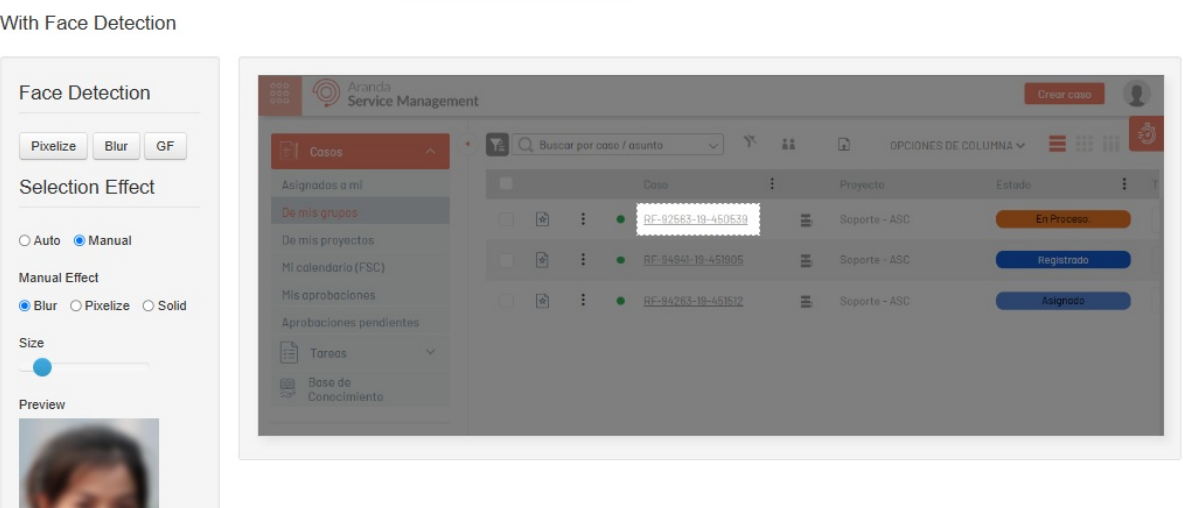
1. Ingrese al editor de imágenes [facepixelizer](#).



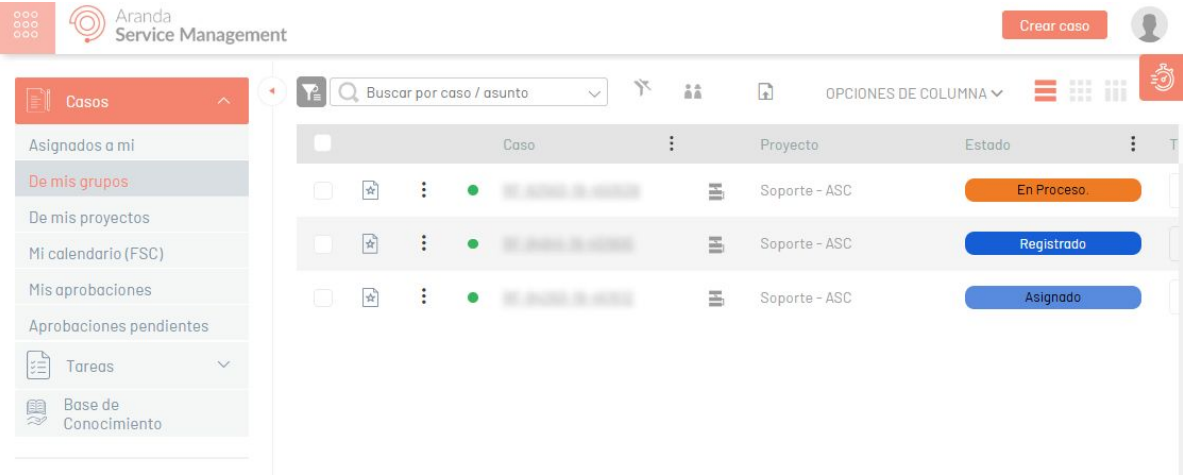
2. Arrastre la pantalla con la información que desea ocultar, al área de trabajo del editor.



- 3. En la sección Selection Effect escoja la opción **Blur**.
- 4. En la sección Size defina el grado de difuminación de la información a ocultar.
- 5. En el área de trabajo del editor, con el mouse defina la sección de información que desea ocultar.



- 6. En la sección Save haga clic en el botón **Save Image** para confirmar los cambios realizados.



Resolución

La captura de las pantallas de la funcionalidad podrá relizarse teniendo en cuenta la resolución de cada equipo. La resolución sugerida es 1600 x 800.

Proporcionalidad

En una secuencia de pantallas del mismo tipo (consola web) el tamaño de cada imagen debe ser igual a la anterior.

Continuidad

Las imágenes como el texto deben mantener una secuencia lógica del uso del producto, para evitar que el usuario pierda el hilo de la explicación.

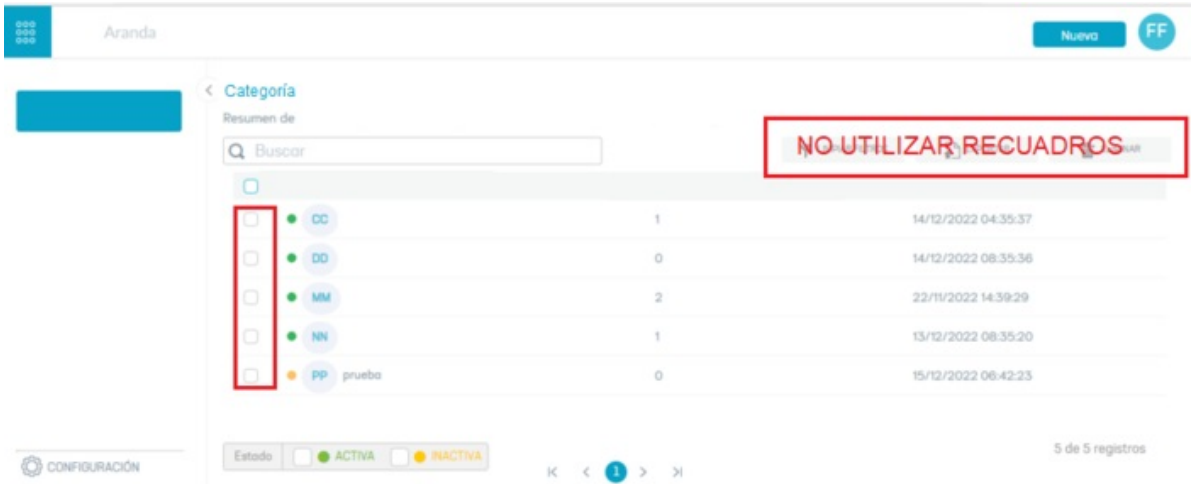
Exceso de pantallas

La descripción de una funcionalidad debe ser un equilibrio de información entre contenido e imágenes. Utilizar pantallas o sólo texto como único recurso para describir una funcionalidad, no ayuda en el aprendizaje por parte del usuario final.

Las pantallas deben aportar algo nuevo, de lo contrario, se convierte en información repetida.

Sin Recuadros

La pantalla es un complemento al contenido y la información que presenta debe defenderse sólo. Por estandar en los estilos y diseño de los manuales, no se debe usar este tipo de marcas.



Links de apoyo para manejo de imágenes

- [Extensión para Extraer imágenes de un documento en PDF](#)
- [Extensión para Redimensionar el tamaño de una imagen al estandar requerido](#)
- [Editor de Imágenes Facepixelizer](#)

Manejo de Videos

Los videos o Gifs animados pueden ser un recurso útil para describir o complementar las características de una funcionalidad en la medida que sirvan de refuerzo y apoyo a los contenidos generados.

Debe tener presente las siguientes recomendaciones:

Grabe Escritorio y Pantalla

1. Desde el dispositivo móvil o desktop seleccione una aplicación o habilite la opción (en dispositivos) para grabar el escritorio en el computador personal o la pantalla en el dispositivo móvil, regsitrando los elementos representativos de la funcionalidad.

📌 **Nota:** Un video generado no debe superar el minuto de duración, es un recurso complementario al contenido del manual.

En los procesos de actualización de la documentación, los videos son recursos que implican mayor tiempo en la generación y actualización y no es recomendable utilizar sólo videos.

2. Exporte el video en formato Mp4 o Gif. Si el video generado está en un formato diferente a Mp4, debe convertirlo a MP4.

Subir video al proyecto

3. Si el video está en formato gif, podrá [subir el archivo al proyecto de documentación como una imagen](#).
4. Si el video está en formato Mp4, el redactor técnico debe subir el video en arandadocs.
5. Desde un artículo del proyecto de documentación utilice la [etiqueta para hacer el llamado al video subido en arandadocs](#).

Etiqueta para subir un .Gif como imagen en proyecto de documentación:



```
<center></center>
```

Vista final del archivo .Gif en Manual de Aranda Assistme:



Etiqueta para realizar el llamado de un video almacenado en arandadocs, para proyecto de documentación:

```
<video width="620" height="540" controls>
  <source src="https://docs.arandasoft.com/videos/pruebavideo002.mp4">
</video>
```

Vista final del video en Mp4 en el manual de Buenas Prácticas de Documentación:

—

Links de apoyo para gestión de Videos

- Aplicaciones sugeridas para grabar escritorio: [screencastify](#), [Filmora](#), [obs Studio](#)
- [Cómo grabar la pantalla del móvil en iOS y Android](#)
- [Convertir video en formato Mp4](#)
- [Convertir video en gif](#)

Etiquetas y Links de Apoyo

En el proceso de documentación en los artículos de Markdown puede ser de utilidad el manejo de diferentes recursos para la elaboración de contenidos (texto, imágenes, enlaces) y a su vez para manetener un estándar en la presentación de la documentación.

En el lenguaje de Markdown se identifican etiquetas que facilitan el proceso de documentación de una forma más eficiente y consecuente con el manejo de la funcionalidad.

📌 **Nota:** Se recomienda reutilizar las etiquetas más comunes en la construcción del contenido, para mantener el estandar y estilos propuestos para los manuales generados.

Las etiquetas más utilizadas al subir contenido al proyecto de documentación son:

Numeración

Para mantener el orden numérico de un procedimiento en el lenguaje Markdown y que se respete la continuidad numérica, aunque existan párrafos intermedios, debe tener en cuenta estas opciones:

Numeración con span

Esta etiqueta permite incluir tablas, texto e imágenes entre la numeración de un procedimiento, sin romper la continuidad numérica.

```
<span>1.</span> texto,texto,texto.  
TABLA  
<span>2.</span> texto,texto,texto.  
IMAGEN  
<span>3.</span> texto,texto,texto.
```

Numeración con un Dígito

Esta etiqueta permite incluir la numeración de un procedimiento si no hay saltos de línea entre los pasos descritos.

📌 **Nota:** Para mantener la continuidad numérica entre saltos de línea (notas, tablas, imágenes), por cada línea intermedia genere un (1) espacio con la tecla tabulador “Tab” para mantener a continuidad posterior al salto de línea.

Estructura de la etiqueta/Numeración SIN salto de línea:

```
1. Ingrese a la consola de Azure DevOps.  
1. seleccione el repositorio del proyecto.  
1. Seleccione una ubicación (carpetas,articulos) en la estructura del proyecto.  
1. Haga clic en el botón Commit para guardar los cambios.
```

Resultado de la etiqueta/Numeración SIN salto de línea:

📌 **Ejemplo numeracion de 1 dígito:**

1. Ingrese a la consola de Azure DevOps.
2. seleccione el repositorio del proyecto.
3. Seleccione una ubicación (carpetas,artículos) en la estructura del proyecto.
4. Haga clic en el botón Commit para guardar los cambios.

Estructura de la etiqueta/Numeración CON salto de línea:

```
1. Ingrese a la consola de Azure DevOps.  
1. seleccione el repositorio del proyecto.  
  
> 📌 Ejemplo: prueba de ejemplo comno salto de línea  
  
1. Seleccione una ubicación (carpetas,articulos) en la estructura del proyecto.  
  
| Descripción | campo |  
| ----| ----|  
| Windows 8.1| Pro |  
| Windows 10 |Pro |  
  
1. Haga clic en el botón Commit para guardar los cambios.
```

Resultado de la etiqueta/Numeración CON salto de línea:

1. Ingrese a la consola de Azure DevOps.
- 📌 Ejemplo: prueba de ejemplo comno salto de línea
1. seleccione el repositorio del proyecto.

Descripción	campo
Windows 8.1	Pro
Windows 10	Pro

1. Seleccione una ubicación (carpetas,artículos) en la estructura del proyecto.
2. Haga clic en el botón Commit para guardar los cambios.

Agregar y centrar Imágenes

Esta etiqueta permite llamar una imagen subida previamente en la estructura del repositorio en la carpeta images, utilizando la ruta asociada y centrar la imagen en el artículo.

```
<center></center>
```

📌 Nota: Conozca [cómo subir una imagen al repositorio del proyecto](#).

Tamaño de Imágenes

Esta etiqueta permite ajustar el tamaño de la imagen al [tamaño sugerido para el tipo de pantalla a utilizar](#).

A la etiqueta de Agregar imagen se le incluye la variable Style con el tamaño requerido.

Estructura de la etiqueta/Agregar Imagen con el tamaño original:

```
<center></center>
```

Resultado de la etiqueta/Agregar Imagen con el tamaño original:



Estructura de la etiqueta/Tamaño Imagen Ajustado:

```
<center></center>
```

Resultado de la etiqueta/Tamaño Imagen Ajustado:



Links

La integración de los diferentes contenidos permite relacionar de forma acertada la dcumentación requerida.

Links entre artículos

Esta etiqueta genera un enlace entre artículos del repositorio de documentación. Debe tener en cuenta la ubicación del artículo destino para hacer el enlace de manera correcta.

- ▮ **Ejemplo:** Si el enlace es entre artículos (artículo_origen y artículo_destino) que hacen parte de la misma carpeta (carpeta_origen), el enlace sería:

```
[Nombre del llamado][articulo_destino.html]
```

- ▮ **Ejemplo:** Si el enlace es entre articulos (carpeta_origen/artículo_origen), (carpeta_destino/artículo_destino) de diferentes carpetas del proyecto de documentación, el enlace sería:

```
[Nombre del llamado](../carpeta_destino/articulo_destino.html)
```

Links entre Externos

Esta etiqueta genera un enlace absoluto desde el proyecto de documentación con sitios web externos. La estructura del enlace es:

```
[Nombre del llamado](url o ruta externa: https://docs.arandasoft.com/)
```

Links Externos con Variables

Esta etiqueta genera un enlace entre artículos de otros repositorios de documentación de Aranda. Para hacerlo debe tener en cuenta os siguientes pasos:

1. Defina [las variables en el archivo _config.yml](#) de los repositorios Aranda que tendrán enlaces externos desde los artículos del repositorio origen.
2. La etiqueta del llamado a la variable tendrá la siguiente estructura:

```
[Módulos Aranda Commons ]( {{site.links.common}}/pages/configuracion_general/modulos_common.html )
```

Anclas

Esta etiqueta permite generar un enlace entre artículos del proyecto de documentación, a una sección en particular del artículo destino. Para formar este enlace debe existir un ancla en la sección del artículo destino y un enlace desde el artículo origen.

- ▮ **Ejemplo:** Si el enlace es entre artículos (carpeta_origen/artículo_origen), (carpeta_destino/artículo_destino) de diferentes carpetas del proyecto, se utiliza la etiqueta del link externo con la url correspondiente y se agrega **#nombre_del_ancla** el enlace sería:

Estructura de la etiqueta Ancla/Link desde artículo Origen:

```
[Nombre del llamado](../nombre_carpeta_externa/nombre_del_articulo#nombre_del_ancla)
```

```
[Estructura del Contenido](../pautas_documentacion/estructura_del_contenido#vision)
```

Resultado de la etiqueta Ancla/Link desde artículo Origen

[Estructura del Contenido](#)

Estructura de la etiqueta Ancla/Nombre del ancla en Artículo Destino:

```
<span id="Nombre_del_ancla">Título artículo destino</span>
```

📌 **Nota:** El nombre del ancla del artículo destino debe coincidir con el nombre del ancla utilizado en el link del artículo origen.

Anclas / Encabezado

Esta etiqueta permite generar un enlace entre artículos del proyecto de documentación, a un encabezado (título) en particular del artículo destino. Para formar este enlace debe existir un ancla en la sección del artículo destino y un enlace desde el artículo origen.

📌 **Ejemplo:** Si el enlace es en el mismo artículo (articulo_origen y destino), se utiliza la etiqueta del link con la url correspondiente y se agrega `{#nombre_del_ancla}` el enlace sería:

Estructura de la etiqueta Ancla Encabezado/Link desde artículo Origen y destino:

```
[Nombre del llamado](nombre_del_articulo#nombre_del_ancla)
[Links de apoyo](etiquetas#links_apoyo)
```

Estructura de la etiqueta Ancla Encabezado/ ancla artículo destino

```
### Links de Apoyo {#nombre_ancla}
```

Resultado de la etiqueta Ancla Encabezado/Link desde artículo destino

[Links de apoyo](#)

Notas

Esta etiqueta permite destacar contenidos de la funcionalidad como mensajes, advertencias o recomendaciones en general.

```
> 📌 **Nota:** Información descriptiva de la nota que permite destacar contenidos de la funcionalidad.
{: #important}
```

Las etiquetas tipo Notas se podrán utilizar con variaciones tipo Advertencia, Ejemplo o Nota. Para mayor información consulta La [Etiquetas tipo Notas](#)

Tablas

Esta etiqueta permite organizar la información en una tabla en el formato markdown.

Nombre	Tipo	Obligatorio	Descripción
Texto	Texto	texto	
Texto	Texto	texto	
Texto	Texto	texto	
Texto	Texto	texto	

- Notas:
- 1. La etiqueta de tablas sólo permite generar tablas simples de filas y columnas.
 - 2. El uso correcto de los pipes o líneas verticales, que estructuran las filas y columnas de la tabla, permite que la tabla se construya de forma correcta.

Video

Esta etiqueta permite incluir videos que se han incluido en arandadocs.

```
<video width="620" height="540" controls>
  <source src="https://docs.arandasoft.com/videos/pruebavideo002.mp4">
</video>
```

Código

Esta etiqueta permite incluir la información de código de un programa, un api o una referencia de programación, que el usuario requiere conocer.

Para incluir código de un programa en markdown podrá utilizar la etiqueta de Comillas Invertidas y la etiqueta virgulilla o tilde(~).

Etiqueta virgulilla o tilde(~)

La etiqueta para incluir código en markdown debe mantener el siguiente formato:

Apertura '~~~'

Código

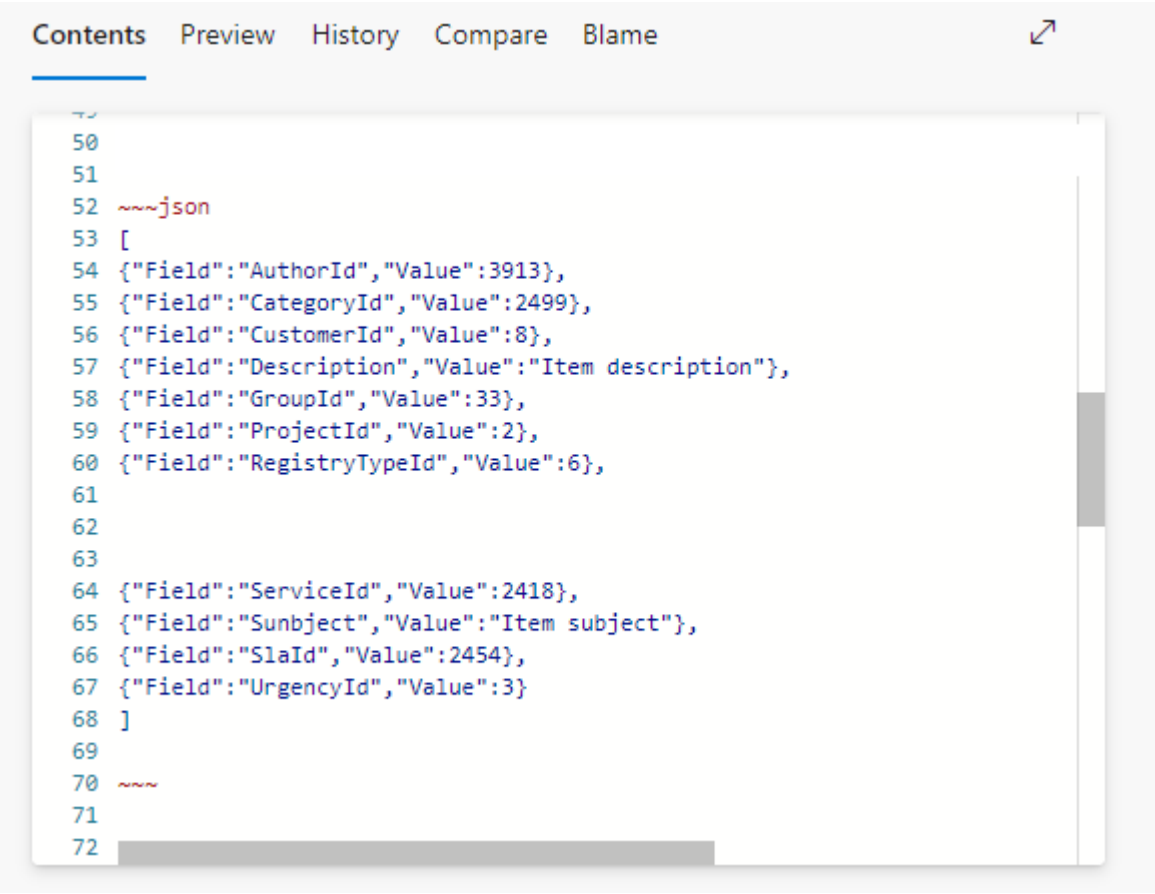
Cierre '~~~'

```
29 ## Cuerpo de la petición
30
31 ~~~
32 {
33
34     "consoleType": 1,
35
36     "password": "123",      "providerId": 0,
37
38     "userName": "e1"
39
40 }
41 ~~~
42
43 ## Respuesta
44
45 ~~~
46 {
47
48     "defaultUrl": "####",
49
50     "renewToken": "#####",      "token": "#####"
51
52 }
53 ~~~
```

🔖 **Nota:** Se recomienda No utilizar una captura de pantalla de código sino la etiqueta Markdown para incluir código, para facilitar al usuario copiar la información compartida.

🔖 **Nota:** Para determinar el estilo en una etiqueta tipo código, agregue el nombre del lenguaje de programación después de la etiqueta de apertura.

🔖 **Ejemplo:** Imagen de etiqueta markdown con tipo de programación json - Vista Editor Azure



🔖 **Ejemplo:** Imagen de etiqueta markdown con tipo de programación json - Vista pública



Etiqueta comilla invertida

La etiqueta para incluir código en markdown debe mantener el siguiente formato:

Apertura `````

Código

Cierre `````

🔖 **Ejemplo:** Imagen de etiqueta de comilla invertida tipo de programación json



```
{
  "firstName": "John",
  "lastName": "Smith",
  "age": 25
}
```

Links de apoyo para la creación de documentos en Markdown

- [Sintaxis Markdown](#)
- [Convertidor de documentos universal](#)
- [Extensión para convertir documento en formato pdf a formato markdown](#)
- [Extensión para convertir tablas a formato markdown](#)
- [Extensión para extraer imágenes de un documento Pdf](#)
- [Extensión para convertir un archivo de word a formato markdown](#) ^c

Etiquetas/Tipos de Notas

Se han diseñado diferentes notas para apoyar el contenido documentado de acuerdo a cada uno de los productos y de acuerdo a las nesecidades del contexto.

Para mantener la línea gráfica de los manuales y una coherencia visual de los mismos, copie y reutilice las etiquetas diseñadas para los diferentes nesecidades:

Ejemplo

La etiqueta de ejemplo puede ser utilizada para ampliar información de un tema especítico a modo de ejemplo.

Estructura Etiqueta:

```
> Ejemplo: Antes de ejecutar los scripts de creación de base de datos verifique que el collation de la base de datos se encuentre en (SQL_Latin1_General_CP1_CI_AI).
```

Resultado Etiqueta:

Ejemplo: Antes de ejecutar los scripts de creación de base de datos verifique que el collation de la base de datos se encuentre en **(SQL_Latin1_General_CP1_CI_AI)**.

Nota

La etiqueta tipo nota puede ser utilizada para ampliar información de un tema específico.

Estructura Etiqueta:

```
> ⓘ Nota: Antes de ejecutar los scripts de creación de base de datos verifique que el collation de la base de datos se encuentre en (SQL_Latin1_General_CP1_CI_AI).
{: #important}
```

Resultado Etiqueta:

ⓘ **Nota:** Antes de ejecutar los scripts de creación de base de datos verifique que el collation de la base de datos se encuentre en **(SQL_Latin1_General_CP1_CI_AI)**.

Advertencia

La etiqueta tipo Advertencia puede ser utilizada para alertar al usuario sobre una situación particular a resaltar o para hacer un llamado de atención en un procedimiento.

Estructura Etiqueta:

```
> ⚠ **Advertencia:** Antes de ejecutar los scripts de creación de base de datos verifique que el collation de la base de datos se encuentre en **(SQL_Latin1_General_CP1_CI_AI)**.  
{: #alert}
```

Resultado Etiqueta:

⚠ **Advertencia:** Antes de ejecutar los scripts de creación de base de datos verifique que el collation de la base de datos se encuentre en **(SQL_Latin1_General_CP1_CI_AI)**.

Enlaces Relacionads

La etiqueta tipo Enlaces Relacionados puede ser utilizada al final de cada artículo para relacionar temas afines al artículo principal.

Estructura Etiqueta:

```
> ⓘ **Enlaces Relacionados:** Antes de ejecutar los scripts de creación de base de datos verifique que el collation de la base de datos se encuentre en **(SQL_Latin1_General_CP1_CI_AI)**.  
{: #related}
```

Resultado Etiqueta:

ⓘ **Enlaces Relacionados:** Antes de ejecutar los scripts de creación de base de datos verifique que el collation de la base de datos se encuentre en **(SQL_Latin1_General_CP1_CI_AI)**.

Definiciones

La etiqueta tipo Definiciones puede ser utilizada para resaltar Términos y definiciones de producto. La etiqueta tiene la característica de desplegar la información al seleccionar la definición.

Estructura Etiqueta:

```
<details>  
  
<summary>DEFINICION: ➤</summary>  
  
Estos problemas se conocen como conflictos y se presentan cuando alguno de los valores que el agente registra periódicamente al servidor (marca de hardware, identificación asignada por el servidor o token dinámico) no coincide con el que el servidor espera. Entonces se bloquean permanentemente las solicitudes de este equipo y se muestra un conflicto en la hoja de vida respectiva en la consola.  
  
</details>{: #product}
```

Resultado Etiqueta:

► DEFINICION: ➤

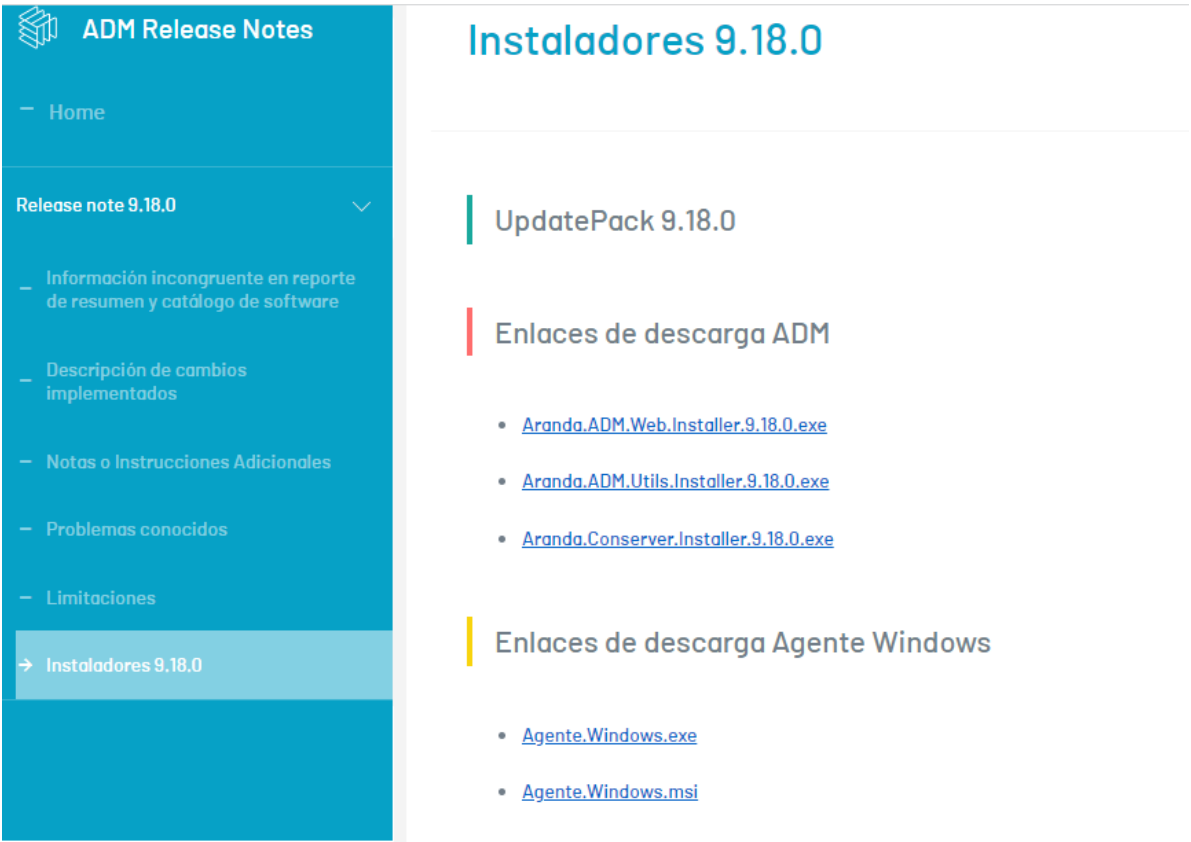
Release Notes Consideraciones

La documentación de las notas release de producto nos referencia las últimas actualizaciones, ajustes, solución a problemas o nuevas funcionalidades de la aplicación.

La descripción correcta de las notas es importante para que el usuario conozca las mejoras, novedades y soluciones de forma estructurada.

Categorías Release

Las notas release por producto organizan la información de las actualizaciones en las siguientes categorías:



Cartegorías	Descripción
Listado de Notas	Esta categoría agrupa el listado de notas de release de cada producto por versión. Las notas descritas dan respuesta a casos reportados en la mesa de servicio (problema, cambios)y deben incluirse como un artículo independiente en la estructura de release de cada producto.
Cambios o mejoras implementadas (opcional)	Esta categoría describe los cambios o mejoras del producto.
Notas Adicionales (opcional)	Esta categoría agrupa información de interés para el cliente. (ejemplo: Versión base de datos, actualización conserver, compatibilidad con otros productos)
Errores Conocidos (opcional)	Esta categoría informativa presenta los errores comunes de la aplicación.
Instaladores	Esta categoría incluye el enlace del instalador o instaladores de la versión del producto liberado. Aplica para los productos OnPremise.

📌 **Nota:** La descripción de cada nota debe tener en cuenta el formato sugerido (Ver descripción Notas)

📌 **Nota:** Las categorías definidas para organizar los release notes deben mantener el formato definido

Descripción Notas de Release

Cada nota release descrita, de acuerdo al tipo de actualización (Problema, cambio, nueva funcionalidad), debe considerar la siguiente formato para presentar la información de la actualización:

AFLS Release Notes

Home

Release Notes AFLS 9.18.1

Release Notes AFLS 9.18.0

Release Notes AFLS 9.17.2

Release Notes AFLS 9.17.1

Release Notes AFLS 9.17.0

Release Notes AFLS 9.16.2

Release Notes AFLS 9.16.1

Release Notes AFLS 9.16.0

Integración nativa entre AFLS y ASMS.

Descargar PDF

Integración nativa entre AFLS y ASMS.

CASO CHG-23434-19-300432:

Se agrega una nueva opción de integración en el módulo de Mesa de servicio para la aplicación de Aranda Service Management, ahora es posible configurar la información de conexión, seleccionar los proyectos y programar la sincronización de usuarios entre las aplicaciones de AFLS y ASMS.

Para ver más detalles consultar la siguiente documentación:

- [Integración Tareas de Campo ASMS- AFLS](#)

Características de la Nota	Descripción
Título	El nombre de la nota debe relacionar la característica más importante del ajuste (Problema, cambio, actualización).
Id del caso	Si la nota hace referencia a un caso reportado en la mesa de servicio, se utiliza la sigla y el número que identifica el caso (Ejemplo: PM (Problema-33496-19-200984). Descripción específica de la nota release de acuerdo al tipo de caso.
Descripción Tipo Caso	- Cambio: Si la nota hace referencia a un tipo de caso Cambio, se describe las características de la funcionalidad. - Problema: Si la nota hace referencia a un tipo de caso problema, se describe así: - <i>Antecedentes</i>(Qué pasaba?): Se debe relacionar y describir el antecedente del problema. - <i>Solución</i>(cómo se corrigió): Se debe relacionar y describir las acciones que se tomaron para la mejora.
Pantalla (Opcional)	Si el ajuste realizado en la nota, requiere proporcionar mayor claridad al usuario, debe incluir una imagen de ejemplo que evidencie las características del ajuste.
Enlace al manual (Opcional)	Si los ajustes mencionados en la nota, son de una nueva funcionalidad o requieren ampliar la información, deben incluir un enlace al manual del producto donde se encuentra documntada esta funcionalidad.

Pautas de Documentación

Pautas Documentación

En le gestión y actualización de la documentación de manuales de Aranda se deben tener en cuenta elementos básicos para escribir, organizar y presentar los contenidos de manera adecuada y de fácil comprensión del usuario o lector final.

El objetivo final de los manuales es describir la funcionalidad, sus procesos y características con la mayor claridad y fluidez posible, convirtiendo este documento en una herramienta de soporte y consulta.

En el proceso de costrucción de los contenidos se deben tener en cuenta las siguientes fases:



1. Proceso de Documentación

En el ejercicio permanente de documentar los manuales de producto, debe tener en cuenta las consideraciones para escribir, corregir y actualizar los contenidos generados.

2. Estructura del Contenido

La estructura de la documentación de producto se consolida con una descripción lógica del uso de la funcionalidad, la formulación correcta de preguntas para desarrollar conceptos, el conocimiento de la interfaz y una visión general de la conceptualización de la funcionalidad.

3. Normas de Estilo Redacción y Ortografía

Todo proceso de escritura a nivel informal o técnico requiere mantener los niveles óptimos de redacción y ortografía que permitan compartir contenidos claros y bien expuestos.

4. Revisión de Terceros/ Autoevaluación

El proceso de generación de contenidos implica una constante tarea interdisciplinaria de construcción, validación y revisión de la información. En este proceso participan en primera instancia los autores del contenido(ingenieros de desarrollo y pruebas), el redactor técnico; El área de soporte o proyectos a nivel interno pueden hacer sus aportes o sugerencias y finalmente el cliente también ayuda a generar una retroalimentación permanente.

5. Actualización

El manual es el reflejo final de lo que hace la funcionalidad y de acuerdo a esto, debe mantener en constante construcción y actualización frente a los cambios o mejoras que se implementan en cada verión de producto,

Procesos de Documentación Manuales

Proceso de documentación para actualización de Manuales en AZURE

El proceso de documentación para los manuales construidos en AZURE, hace referencia a las actualizaciones o nuevos contenidos (texto, imágenes, tablas y funcionalidad) del manual de producto, generados por el autor (responsable) de la aplicación en la plataforma AZURE, para ser atendidos por el redactor técnico de forma puntual para el contenido generado. El proceso es el siguiente:

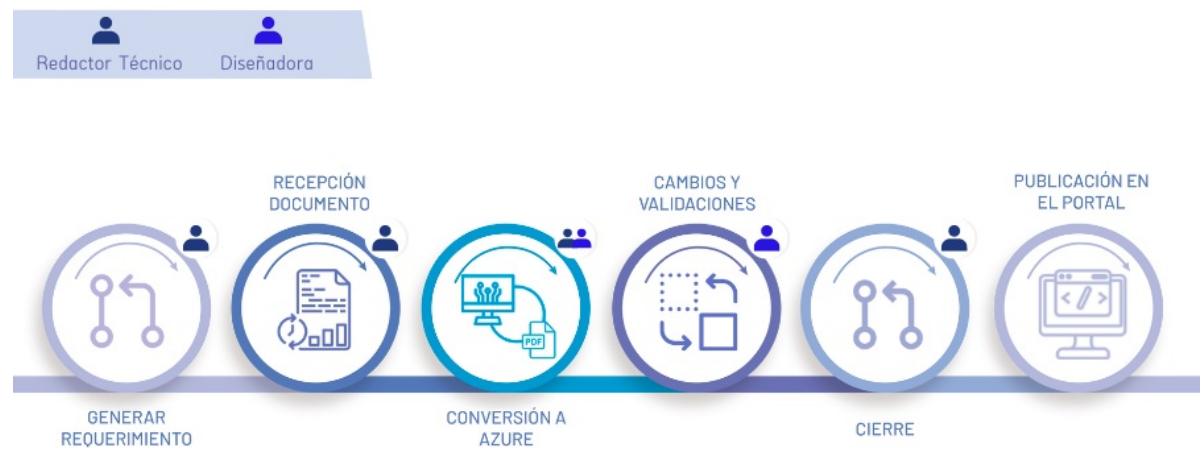


1. **Generar Requerimiento:** El Autor de producto genera requerimiento (PR) por actualización en funcionalidad con impacto en documentación.

2. **Recibo información:** El redactor técnico recibe el requerimiento de producto, se validan los cambios de contenido, se determina el impacto a nivel gráfico y de contenido y se estiman los tiempos de trabajo.
3. **Cambios y Validaciones:** El Redactor técnico realiza los ajustes respectivos de acuerdo a valoración previa.
4. **Cierre** El Autor del contenido aprueba y cierra el requerimiento (PR), integrando la rama al repositorio.

Proceso de documentación de Manuales de Versiones en PDF o WORD

El proceso de documentación para de manuales de versiones anteriores, hace referencia a las tareas de conversión, estructuración y subida del manual al entorno web del portal de documentación de Aranda SOFTWARE. Este proceso no incluye actualización de contenido e imágenes. El proceso es el siguiente:

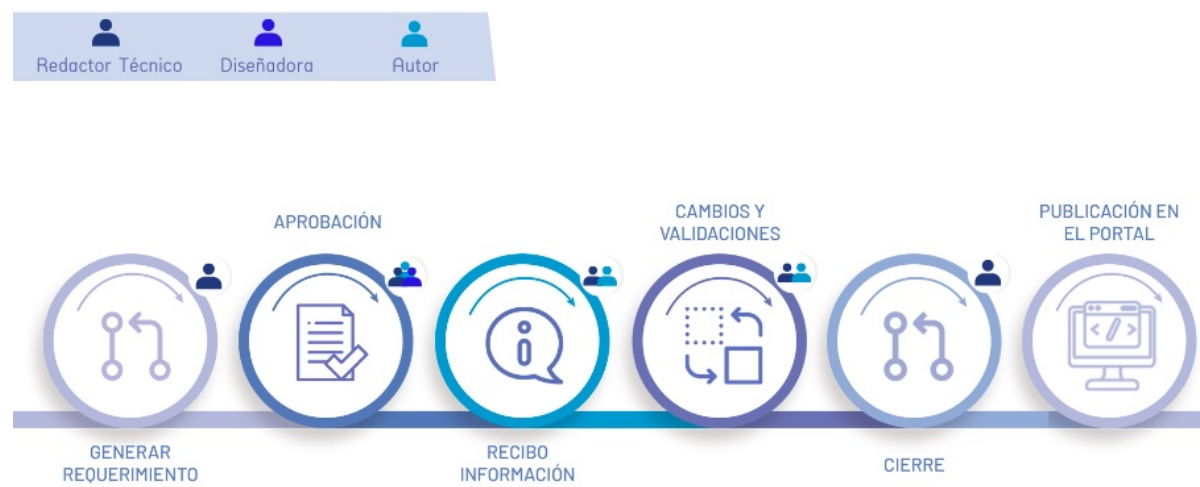


1. **Generar Requerimiento:** El redactor técnico genera requerimiento (PR) para subir los manuales de PDF a AZURE.
2. **Recepción Documento:** El redactor técnico valida la información del manual (tabla de contenido, número de páginas, imágenes), y estima los tiempos de trabajo.
3. **Conversión a Azure:** El redactor técnico en colaboración con Diseñadora UX, realiza diferentes tareas para subir el manual al entorno web de Aranda, como conversión de PDF a Markdown, clonar repositorio en Azure, organización y estructuración de tabla de contenido en nueva plantilla desde Atom o VisualStudio, configuración de estilos, llamado correcto de las imágenes o pantallas, ajuste de tablas.
4. **Cambios y Validaciones:** El equipod de documentación valida que el cambio de versión fue implementado adecuadamente. Si existen problemas en las actividades de conversión de Azure, se realizan los comentarios y sugerencias respectivas, para que el redactor técnico realice los cambios pertinentes, rechazando el requerimiento. Si la conversión fue exitosa se realiza la aprobación y cierre del requerimiento e integración a la rama principal del repositorio.
5. Se hace el llamado del manual desde el portal de documentación.

Proceso documentación de manuales para cambios en estructura

El proceso de documentación con cambios de estructura, hace referencia a la actualización de manuales en un nuevo formato de documentación donde se debe tener en cuenta los siguientes lineamientos de la [Estructura de Contenido para Manuales](#): División de Contenidos, Visión Conceptual, Orden lógico de Uso y Descripción de Procesos de Gestión.

📌 **Nota:** Este proceso no incluye actualización de contenido e imágenes para versiones V8. El contenido aprobado con la nueva estructura será generado a partir de la información existente.

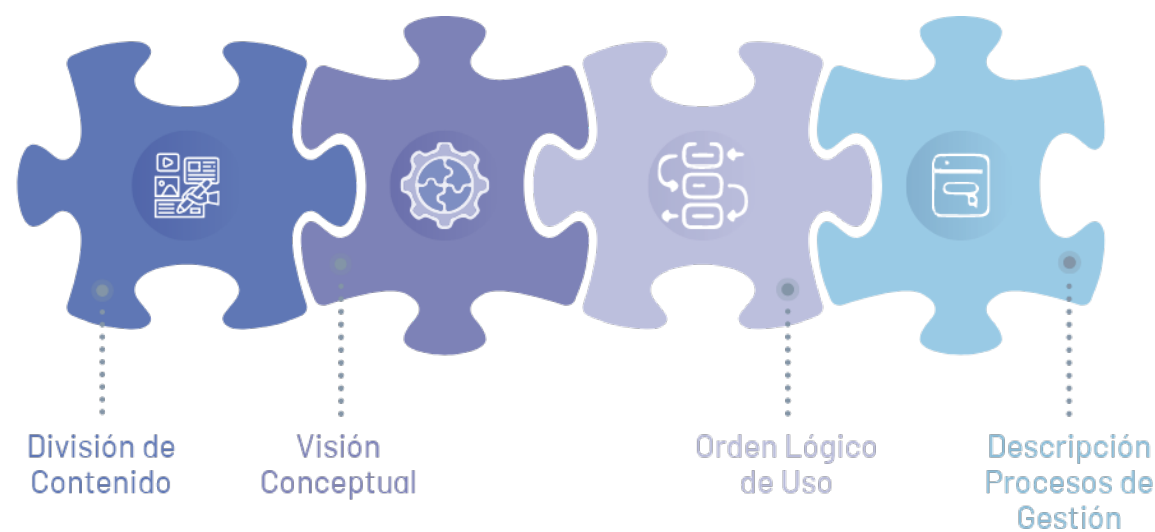


El proceso para el cambio de estructura es el siguiente:

1. **Generar Requerimiento:** El redactor técnico genera requerimiento (PR) para actualizar el manual a nivel de estructura. En espera de revisión y validación de autor de producto y diseño UX.
2. **Aprobación:** En reunión entre redactor técnico, autor de producto y diseño UX se aprueba el cambio de la documentación a nivel de estructura y se establece la planeación de las actividades.
3. **Recibo Información:** En reunión con autor de producto se amplían conceptos para incluir en estructura. El redactor técnico valida los cambios de contenido, se determina el impacto de los cambios; se estiman los tiempos de trabajo.
4. **Cambios:** El redactor técnico realiza los cambios de estructura en el manual teniendo en cuenta los lineamientos definidos en documentación.
5. **Validaciones:** El autor del producto y el redactor técnico confirman los cambios de acuerdo a las validaciones para cambios en documentación definidas.
6. **Cierre:** Se cierra el requerimiento (PR) y se integra la rama al repositorio.

Estructura del Contenido para Manuales

La definición, descripción y uso de la funcionalidades de un producto, debe tener en cuenta unos elementos básicos de la estructura de la información que se comparte con el usuario. Los elementos a considerar son los siguientes:



División de contenidos

La documentación por producto debe evaluar la mejor forma de presentar la información al usuario y organizar los contenidos afines de la siguiente manera:

- **Guía de Inicio:** Es el documento de presentación del producto y donde se describen los elementos básicos y transversales para tener un primer acercamiento a la funcionalidad con items como acceso, funcionamiento o entorno de la aplicación. Ejemplo: En esta guía podrá definir con claridad los elementos básicos de la interfaz (consola web, consola móvil) y las secciones de la consola (menú principal, menú superior, área de trabajo, barra de acceso rápido entre otros.)
- **Manual de Gestión:** Es el documento que describe los diferentes procesos de gestión del producto y está enfocado en el usuario final. En este manual se desarrollan los procesos de gestión y configuración de la funcionalidad.
- **Manual Técnico o Instalación:** Es el documento más especializado que puede incluir contenidos técnicos o referencias de instalación o gestión de componentes.

Visión Conceptual

Es un primer nivel en la documentación de producto se describen los conceptos principales a desarrollar en una sección del manual, direccionando al usuario al uso secuencial de la funcionalidad

La visión conceptual del producto es una vista general de los conceptos y su interacción con otros elementos de la funcionalidad. El usuario debe tener la idea de cómo funciona el producto y su alcance de manera integral antes de conocer sus procesos internos.

▮ **Ejemplo:** En la Visión Conceptual de la Configuración de ASEC se describe de forma general los conceptos que se desarrollan en esa sección como Usuarios, Licencias, servidor de correo, servicios de directorio y autenticación externa.

Orden lógico de uso

Tanto en la clasificación como en la documentación de los diferentes manuales, la información que se presenta debe reflejar el orden lógico del manejo de la funcionalidad.

El usuario debe recorrer el manual, desde la tabla de contenido al interior, de una forma secuencial que le permita realizar una gestión eficiente de lo que hace el producto, en el orden correcto.

- **Ejemplo:** Es necesario describir en primera instancia la configuración de un a política o un usuario, antes de presentar la gestión de la política.

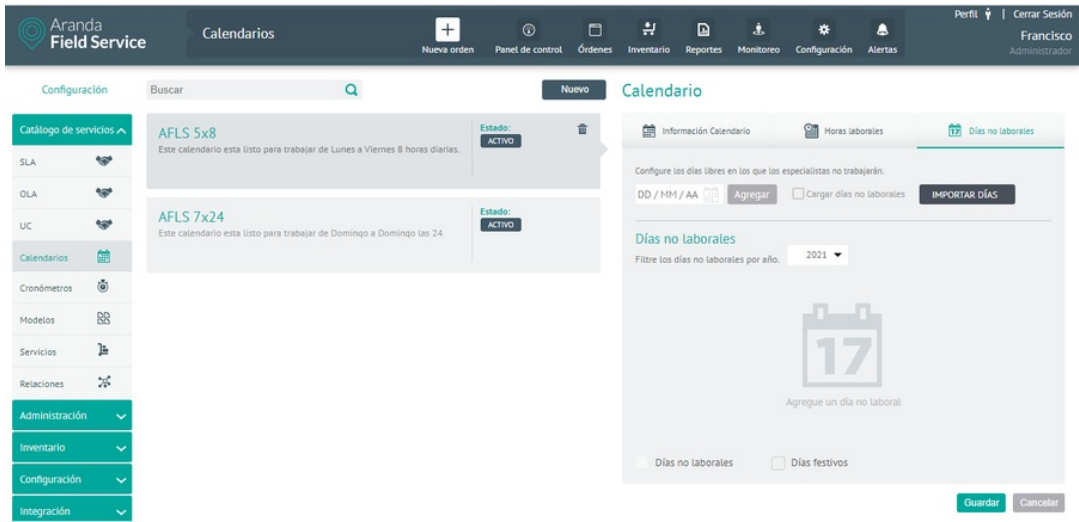
Descripción Procesos de Gestión

Durante la documentación de los procesos de gestión o configuración de una funcionalidad (Consultar, crear, editar, eliminar, exportar) puede utilizar diferentes recursos para optimizar el contenido (pantallas, notas tablas) y redactar la información con las siguientes pautas:

- **Orden Lógico:** La redacción de los contenidos debe respetar y replicar la secuencia de uso de la aplicación para describir cada uno de sus procesos
- **Ubicación y Despliegue de la Interfaz:** La descripción del proceso debe ser fiel a cómo funciona la aplicación y a través del lenguaje debe describir con claridad la ubicación del usuario frente a cada una de los componentes básicos de la consola.

- **Ejemplo:** Crear Calendarios en AFLS.

1. Para crear un Calendario Personalizado, ingrese a la consola de configuración de AFLS, en la sección de **Catálogo de Servicios** del menú principal, seleccione la opción **Calendarios**. En la vista de información seleccione el botón **Nuevo** e ingrese la siguiente información en la Vista Detalle:



- **Formular Preguntas:** Generar preguntas para describir una funcionalidad y desarrollar sus conceptos en la respuesta, puede ser un mecanismo útil para direccionar los contenidos con un objetivo claro. Dependiendo el tipo de información que se quiere compartir, podrá formular diferentes interrogantes:

Tipo de Pregunta	Alcance
¿Qué es?	Esta pregunta se utiliza para definir conceptos que deben quedar claros en la descripción de la funcionalidad. Ejemplo: Qué es un proyecto, qué es un monitor.
¿Para qué?	Esta pregunta se utiliza para destacar la utilidad de la funcionalidad y resaltar la importancia o el beneficio de un tema en particular. Es un complemento a una definición normal de un concepto que requiere un refuerzo en su descripción. Ejemplo: ¿Para qué me sirve un monitor?
¿Cómo funciona?	Esta pregunta se utiliza para describir los diferentes procedimientos de la funcionalidad. La respuesta a esta pregunta son los pasos lógicos para describir una parte de la funcionalidad. Ejemplo: ¿Cómo crear un nuevo grupo de trabajo?

📌 **Nota:** La formulación de preguntas es un punto de referencia para describir un proceso y no necesariamente acompañan el ejercicio descriptivo.

Normas de Estilo

Escribir un manual requiere construir contenido bien escrito para darle claridad al lector.

En el proceso de escritura debemos tener en cuenta la ortografía, la redacción y el estilo o forma de escribir. Podemos tener en cuenta los siguientes recomendaciones

Para escribir con ortografía:

Tip	Descripción	Consultas, trucos, ejemplos
Correctores	Utilizar un corrector de estilo antes de publicar el contenido nos permite filtrar errores básicos de ortografía o de mal tecleado	Word- Correctores de ortografía y gramática
Uso de las palabras	El uso correcto de las palabras nos ayuda a dar el contexto adecuado de lo que queremos decir o simplemente nos ayuda a confirmar alguna duda gramatical (sinónimos, conjugación correcta, ortografía)	Diccionario de la Lengua Española
Reglas de Acentuación	Utiliza las reglas de acentuación para validar el correcto acento en las palabras. No olvida palabras Agudas (común), graves(árbol) y esdrújulas(espórádico)	TIP: Todas las palabras en español, de entrada, llevan acento: Siempre hacemos más fuerza en una sílaba. Pronuncia la palabra en voz alta de forma exagerada, separando sus sílabas. Si dudas, repítelo poniendo la fuerza en cada una de ellas.
Concordancia en género y número	Todas las palabras, sustantivos, adjetivos, verbos deben concordar en género (masculino, femenino) y número	EJEMPLO: “Las políticas configuradas en Aranda Security evaluan los niveles de cumplimiento de aplicaciones de seguridad”.
Uso de la Puntuación	El correcto uso de los signos de puntuación como, puntos, comas, puntos y comas, entre otros, permiten que la lectura sea clara y fácil de entender	EJEMPLO: “Al registrar los campos haga clic en el botón Probar conexión . Si la conexión fue exitosa podrá visualizar el mensaje: La información quedó completa”.
No usar pronombres en primera persona	Evita el uso de pronombres en primera persona: “yo”, “mi”, “mío”, “mío”, “nosotros”, “nuestro”, “nuestro”.	ANTES: “En esta vista podemos observar fecha del último reporte enviado al dispositivo”. DESPUÉS: “En esta sección podrá visualizar la fecha del último reporte enviado al dispositivo”.

Para redactar correctamente:

La redacción es un proceso de organización jerárquica de la ideas que ayuda a estructurar y darle sentido a un texto.

No hay una sólo forma de redactar correctamente, pero un documento de caracter instructivo como un manual, debe considerar los siguientes lineamientos:

Tip	Descripción	Consultas
Pensar en el Otro	Antes de escribir una idea o describir un proceso debe pensar para quién escribe. Si el destinatario del contenido es un usuario final, el lenguaje debe ser preciso y no debe asumir que el lector conoce por defecto como usted, el funcionamiento de la aplicación.	EJEMPLO: “Un usuario de Aranda Security debe considerar tres etapas para la gestión y seguimiento de las políticas de cumplimiento”.
Texto descriptivo	El tipo de texto a utilizar en un manual es de carácter descriptivo, donde predomina un lenguaje concreto y claro para describir el desarrollo de una funcionalidad o la explicación de un proceso	EJEMPLO: “La definición y configuración de políticas de Seguridad permiten establecer mecanismos de diagnóstico, control y protección de la información en diferentes niveles”.
Formulación de preguntas	La definición y redacción de ciertos conceptos de la funcionalidad pueden implementarse a con la resolución a ciertos interrogantes para determinar de manera puntual qué hace, qué significa o cómo utilizo determinada funcionalidad del producto.	EJEMPLO: “¿Cómo Asociar Usuarios en AVS? Para agregar uno o más usuarios a un grupo de soporte, seleccione el campo de búsqueda <i>Asociar usuarios</i> , al hacerlo, se despliegan todos los usuarios registrados en el sistema”.
Modo verbal imperativo	Por ser un documento tipo instructivo, los verbos utilizados expresan una orden o una prohibición	EJEMPLO: “ Ingrese a la consola de Aranda Security Compliance con un usuario con rol de administrador , en la sección de <i>Configuración</i> del menú principal, seleccione la opción Licenciamiento”.
Narrador en tercera persona	La descripción del proceso debe efectuarse en tercera persona sin involucrarse en la narración. Escribir en tercera persona permite realizar una descripción más acertada.	EJEMPLO: “Administre las licencias adquiridas por el cliente y genere la solicitud de licencias para realizar una adecuada gestión de los procesos de soporte remoto de la organización”.
Tiempo presente atemporal	El tiempo verbal preferido es el presente, atemporal; se refiere a hechos sin alcance de tiempo. Es utilizado para presentar o describir características, designar verdades, indicar acciones	EJEMPLO: “En la imagen encuentra el listado de usuarios registrados”.
Estructura Narrativa	Cada oración debe estar bien estructurada (sujeto-predicado y complementos) y la idea desarrollada debe ser enunciada y concluida	EJEMPLO: La solicitud de cambios (Pull Request) permite mantener un proyecto estable y actualizado
Uso de sinónimos	Evite repetir palabras en el contexto de un mismo párrafo, utilizando sinónimos o expresiones similares.	ANTES: Se realizó un ajuste en la consola, se implementa ajuste que permite a los usuarios descargar los campos. DESPUÉS: Se realizó un ajuste en la consola, que permite a los usuarios descargar los campos.
Balance en el contenido	La información desarrollada en un documento debe mantener un equilibrio, ser concreto y no repetitivo. Demasiada información no garantiza la permanencia del lector . Información reducida da la impresión de un contenido incompleto. .	EJEMPLO: “Una política es una entidad que define las reglas y condiciones asociadas a componentes de seguridad, que se aplican a un programa bajo criterios que cumplen los marcos regulatorios de protección de la información”.

📌 **Nota:**
- 1. El responsable de generar los contenidos tiene un status comunicativo mayor al del usuario y como experto en el tema conoce el funcionamiento del producto. y esta en la capacidad de redactar desde una

visión global más no personal.

- 2. El redactor No debe escribir como habla. Las frases no deben seguir el patrón de una conversación sino el de la redacción|