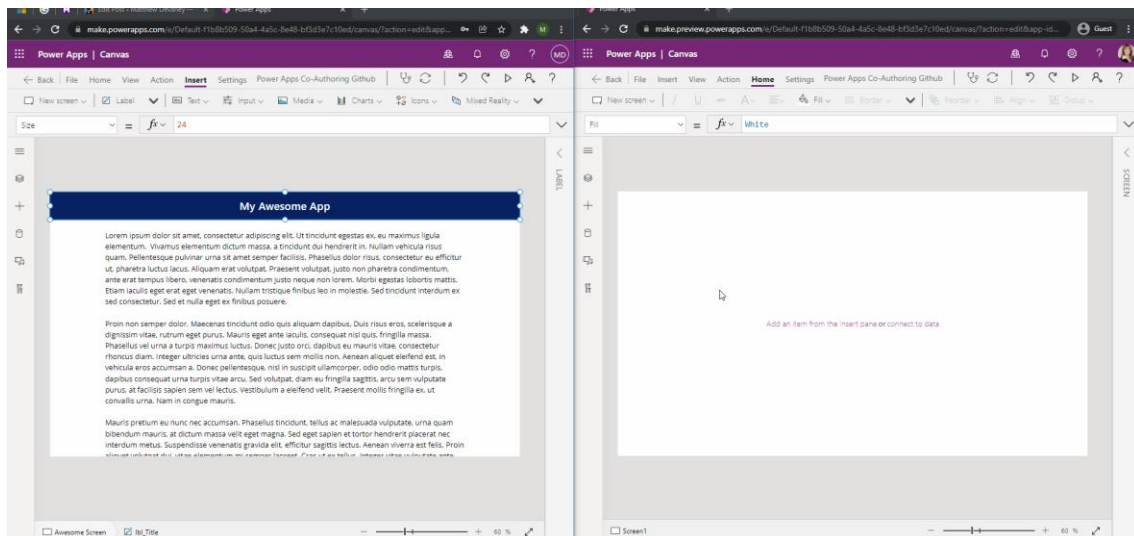


## Introducción

Con Power Apps, varios desarrolladores pueden trabajar en una aplicación al mismo tiempo para acelerar el proceso de desarrollo. Así es como se ve la colaboración entre dos desarrolladores en Power Apps.

Ambos desarrolladores en el ejemplo a continuación tienen la misma aplicación abierta. El primer desarrollador en el lado izquierdo agrega una barra de título a la pantalla y algo de texto. Luego, el primer desarrollador presiona el botón de sincronización para guardar y confirmar los cambios. El segundo desarrollador en el lado derecho quiere ver los cambios realizados por el primer desarrollador. Entonces, el segundo desarrollador también presiona el botón de sincronización y aparecen la barra de título y el texto.

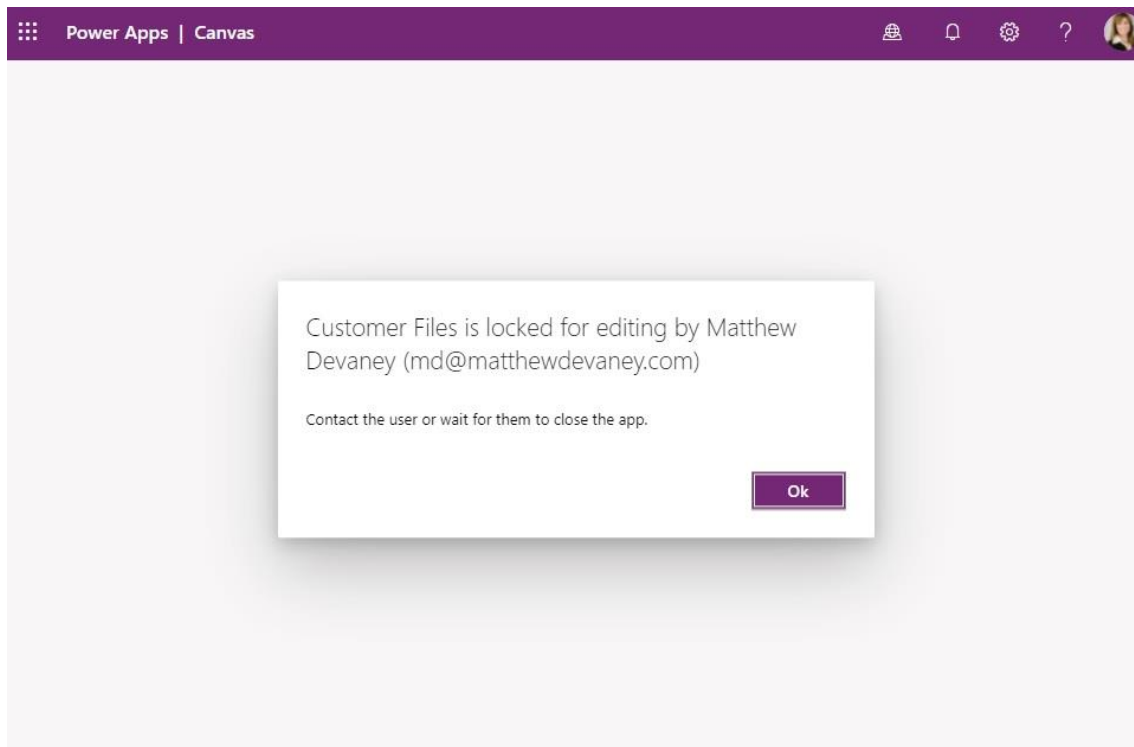


## ¿Qué es Git y cómo proporciona control de versiones?

Para que varios desarrolladores colaboren en Power Apps, la configuración experimental *de control de versiones de Git* debe estar activada. Git es una herramienta que utilizan los desarrolladores de software para el control de versiones, una forma de realizar un seguimiento de quién realizó qué cambios en una aplicación y cuándo. También permite a los desarrolladores que están trabajando en su propia instancia de la aplicación la capacidad de fusionar su código con el de otros. Si surge un problema con el código de la aplicación, los desarrolladores pueden usar Git para comparar los cambios entre cada versión de la aplicación y volver a una anterior.

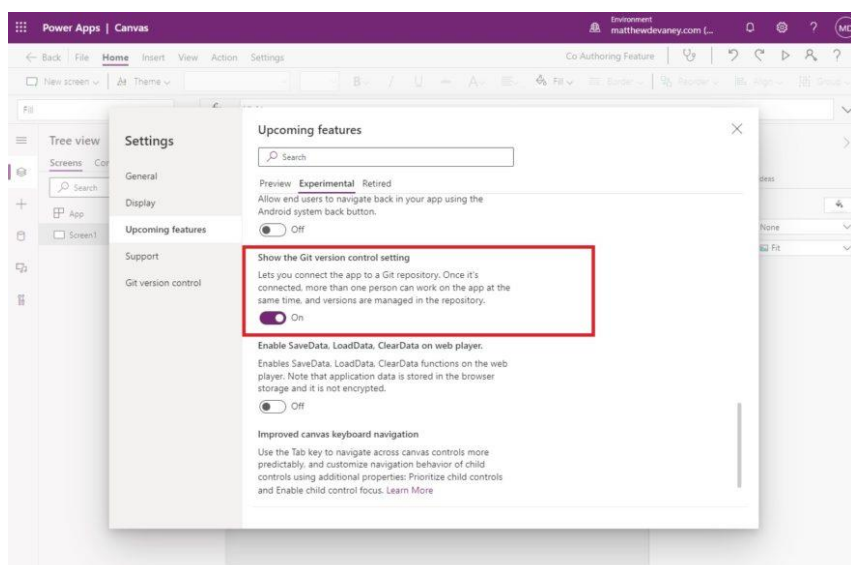
## Habilitar la colaboración con el control de versiones de Git

Si más de un desarrollador intenta abrir la misma aplicación, aparecerá el mensaje “Power Apps está bloqueada para edición. Comuníquese con el usuario o espere a que cierre la aplicación”.



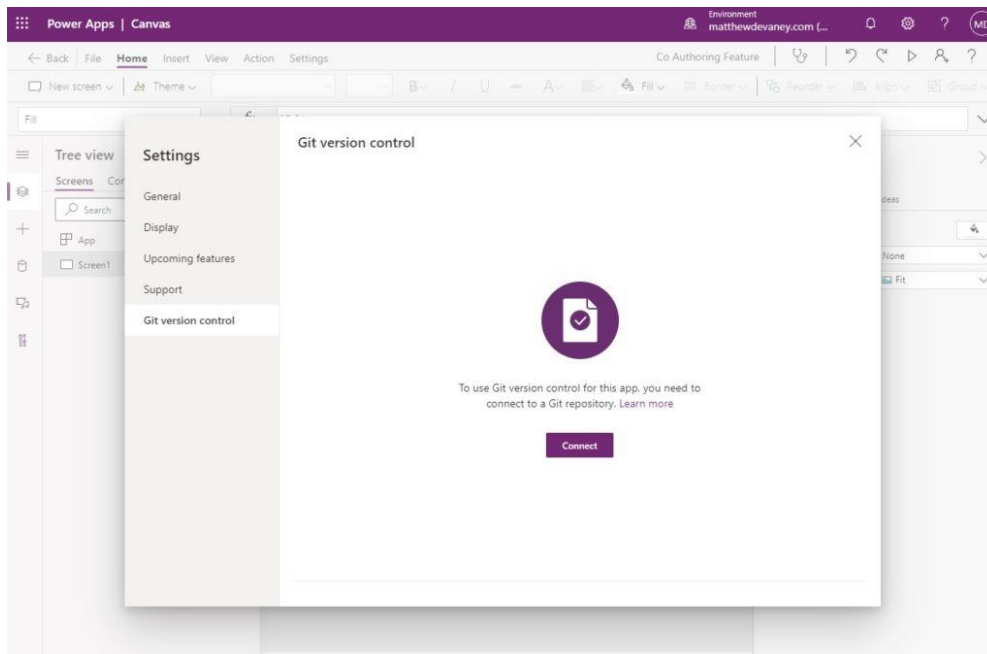
Cuando el control de versiones de Git está habilitado, varios desarrolladores pueden trabajar en la misma aplicación y el mensaje de bloqueo para edición ya no aparecerá. ¿Cómo es posible esto? Power Apps abre una instancia independiente de la aplicación en el modo Studio para cada desarrollador. Cada vez que un desarrollador guarda su trabajo, este se confirma (es decir, se guarda) en un repositorio Git compartido (una carpeta). Al guardar, se fusiona el nuevo código con el código existente en el repositorio aportado por otros desarrolladores. Luego, el código recién fusionado se descarga en Power Apps Studio para que el desarrollador continúe trabajando en él.

Vaya al menú Configuración en Power Apps y luego navegue hasta la sección *Próximamente funciones* . Busque la *configuración Mostrar el control de versiones de Git* en funciones experimentales y actívela.

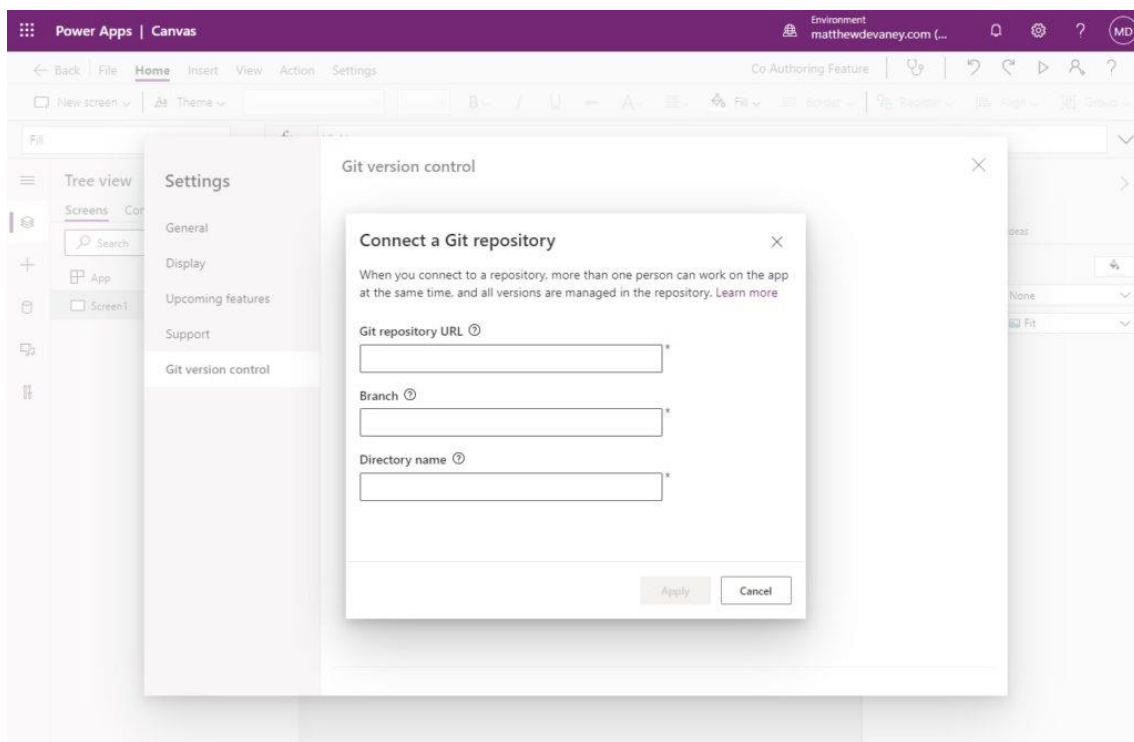


## Conectar Power Apps a un repositorio Git

Una vez que se cambia la configuración del control de versiones de Git, debemos conectar Power Apps a un repositorio de Git. Vaya a la sección *Control de versiones de Git* del menú de configuración y seleccione *Conectar*.



Aparece un formulario que solicita la URL del repositorio Git, la rama y el nombre del directorio. Para completar estos campos, primero tendremos que crear un repositorio Git. Esto es algo que se hace fuera de Power Apps.



## Cómo elegir un servidor para el repositorio Git

Git es una herramienta genérica y existen muchos servicios en línea que pueden alojar un repositorio Git. ¿Qué host debería elegir para Power Apps? Recomiendo estos dos servicios que ofrece Microsoft y cada uno tiene un nivel gratuito:

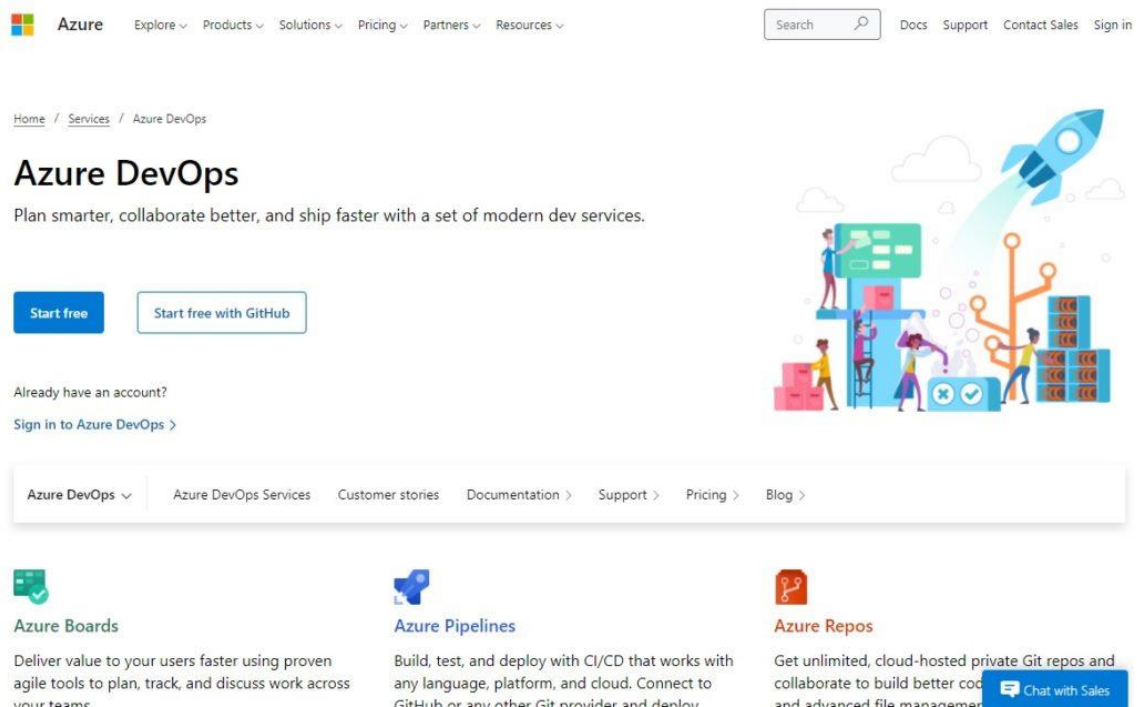
- [Github](#) : la comunidad de código abierto más popular del mundo
- [Azure DevOps](#) : una herramienta de colaboración para desarrolladores que también incluye repositorios git

Para este tutorial, utilizaremos **Azure DevOps** porque es la opción preferida de Hitachi Solutions.

*Importante: si elige un servicio alternativo, tenga en cuenta que la coautoría de Power Apps no admite repositorios locales en este momento .*

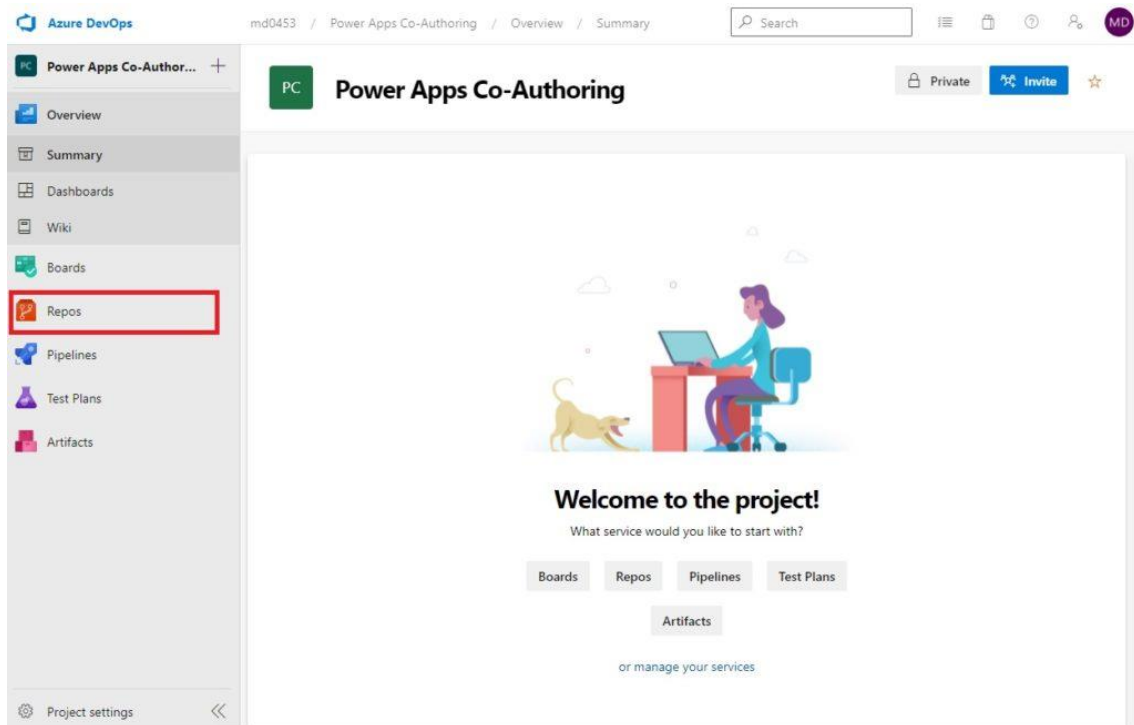
## Regístrese para obtener una cuenta gratuita de Azure DevOps

Vaya a [dev.azure.com](https://dev.azure.com) e inicie sesión en su cuenta. Si no tiene una cuenta de Azure Dev Ops, puede registrarse de forma gratuita. Solo necesitamos el nivel gratuito para configurar un repositorio de Git para que varios desarrolladores colaboren en Power Apps.

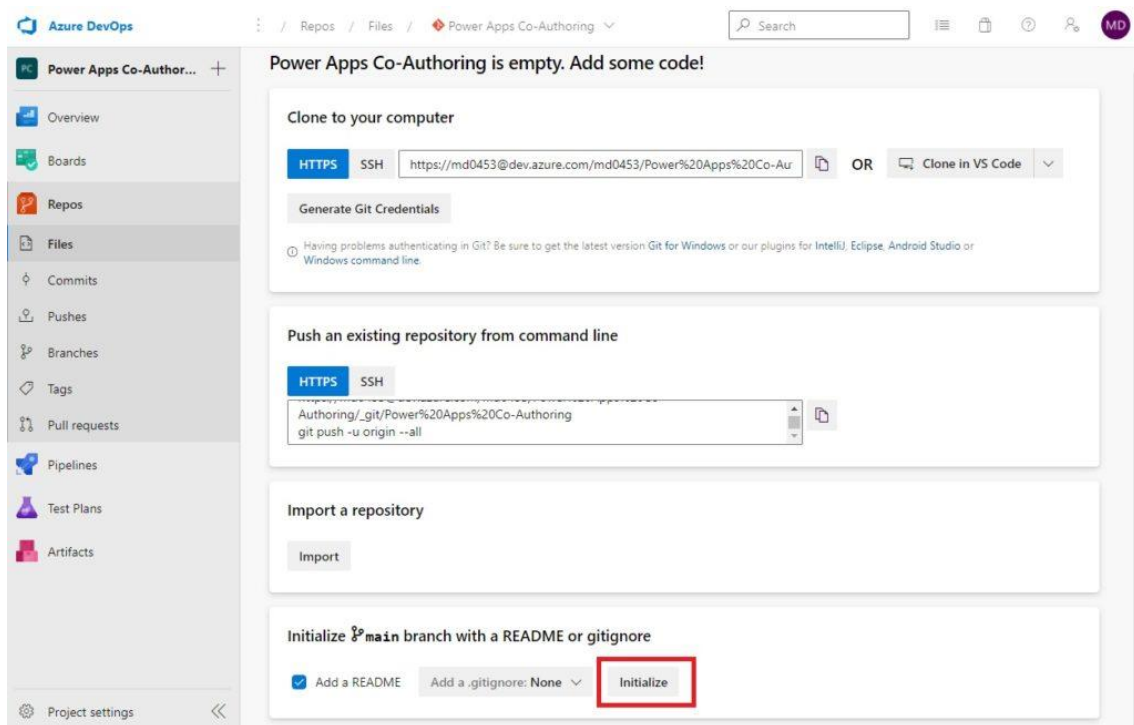


## Crear un nuevo repositorio en Azure Dev Ops

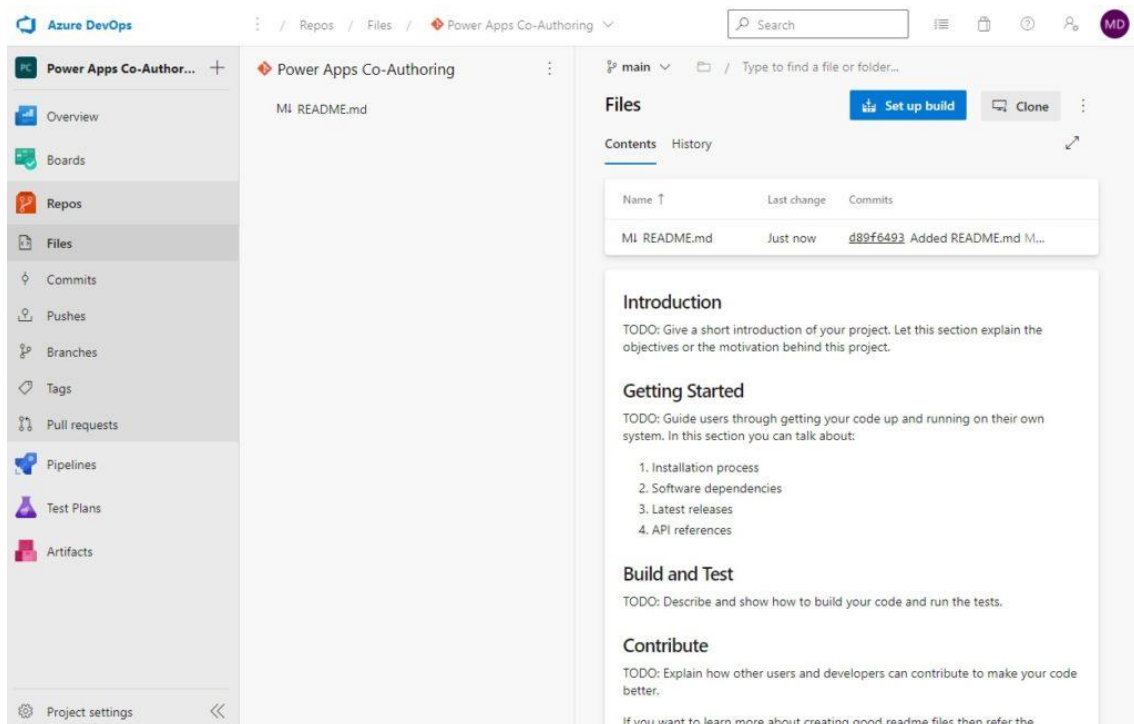
A continuación, creamos un nuevo repositorio para alojar nuestra aplicación Power Apps. Crea un nuevo proyecto en Azure Dev Ops. Luego, selecciona *Repositorios* en el menú de navegación izquierdo.



Inicialice la rama principal del repositorio con un archivo README.md.



Ya se ha creado el repositorio de Azure Dev Ops. Podemos obtener la ubicación del repositorio desde la URL en la barra de direcciones de nuestro navegador web.



## Ingresar información del repositorio de Auzre Dev Ops en Power Apps

Vuelva a Power Apps y complete el formulario Conectar un repositorio Git con estos valores. Haga clic en Aplicar cuando haya terminado.

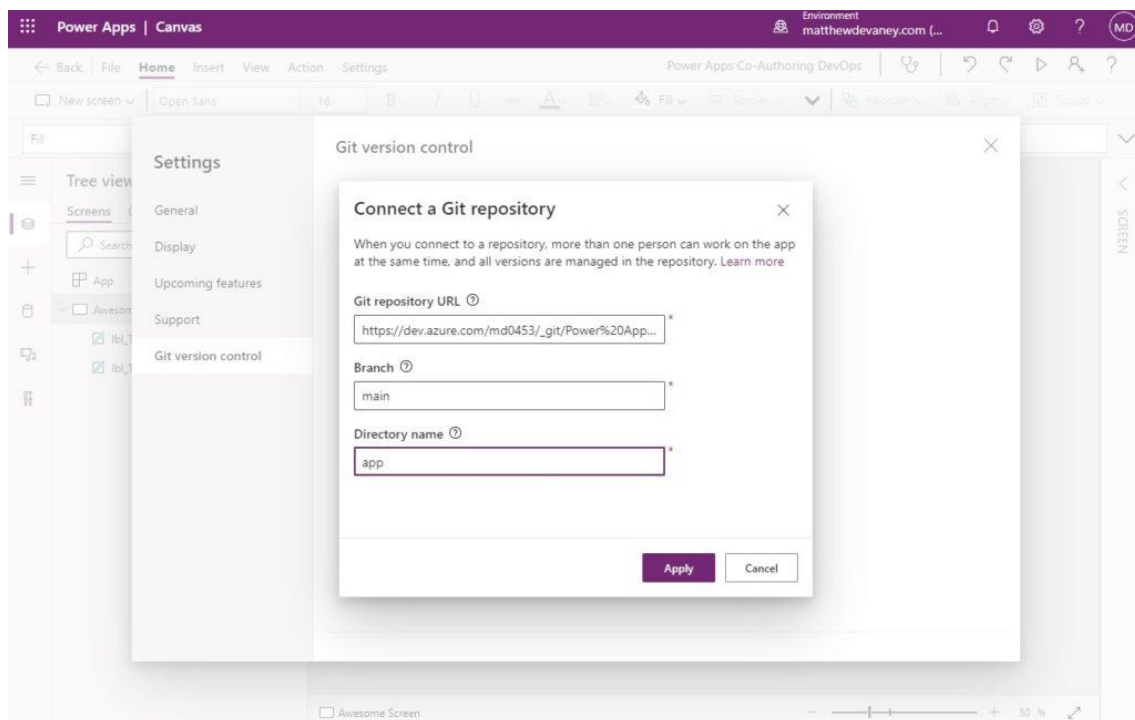
| Campo                   | Valor   |
|-------------------------|---|
| URL del repositorio Git | <p><code>https://dev.azure.com/nombre_organización/nombre_proyecto/_git/nombre_repositorio</code></p> <p><i>Ejemplo:</i> <code>https://dev.azure.com/md0453/Power%20Apps%20Co-Authored/_git/Power%20Apps%20Co-Authored</code></p> |
| Rama                    | principal   |
| Nombre del directorio   | aplicación  |

La *URL del repositorio de Git* es una dirección web que podemos escribir en nuestro navegador web para ver el repositorio. *App* es el nombre de la carpeta donde se almacenará

nuestro código fuente de Power Apps. Power Apps no permite que el código fuente se almacene en la carpeta raíz del repositorio.

*Main* es el nombre de la rama de desarrollo predeterminada. En el desarrollo de software tradicional, se crean ramas (es decir, copias) de la aplicación para que funcionen en funciones específicas y luego se fusionan nuevamente con la rama principal una vez que se completa la función. Power Apps solo funciona en la rama *principal*.

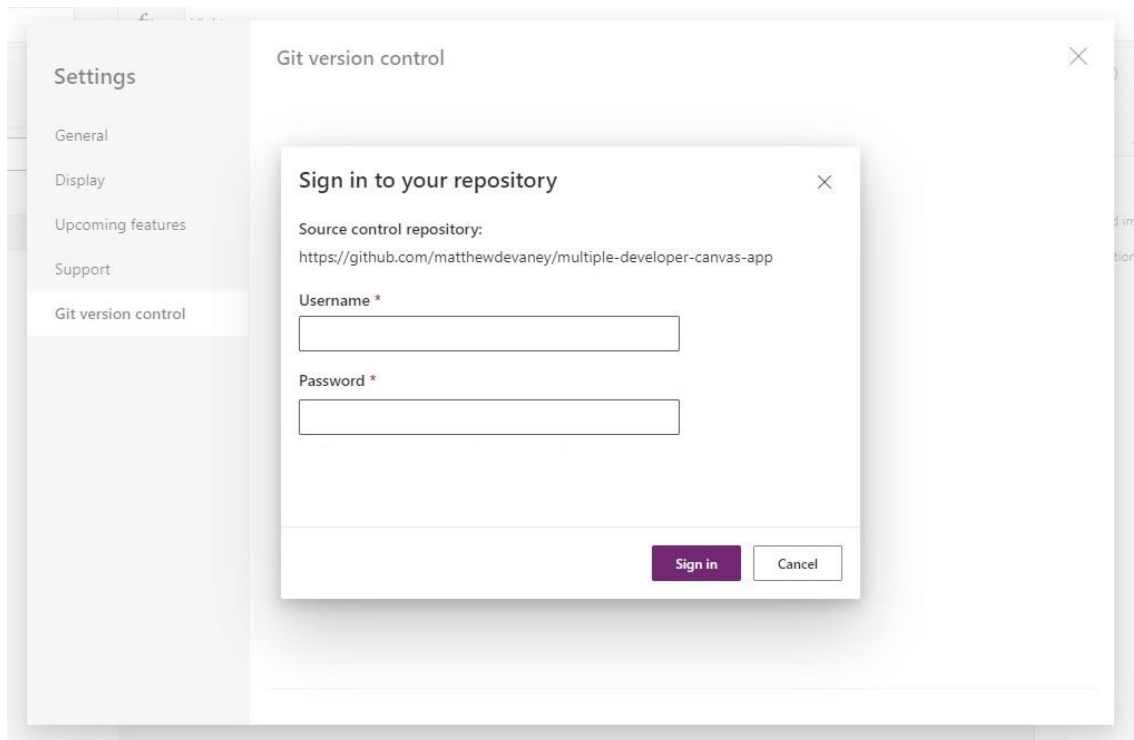
*Importante: debemos elegir un nombre de directorio que no exista actualmente en el repositorio y permitir que Power Apps lo cree. Si el nombre del directorio ya existe, Power Apps no podrá conectarse.*



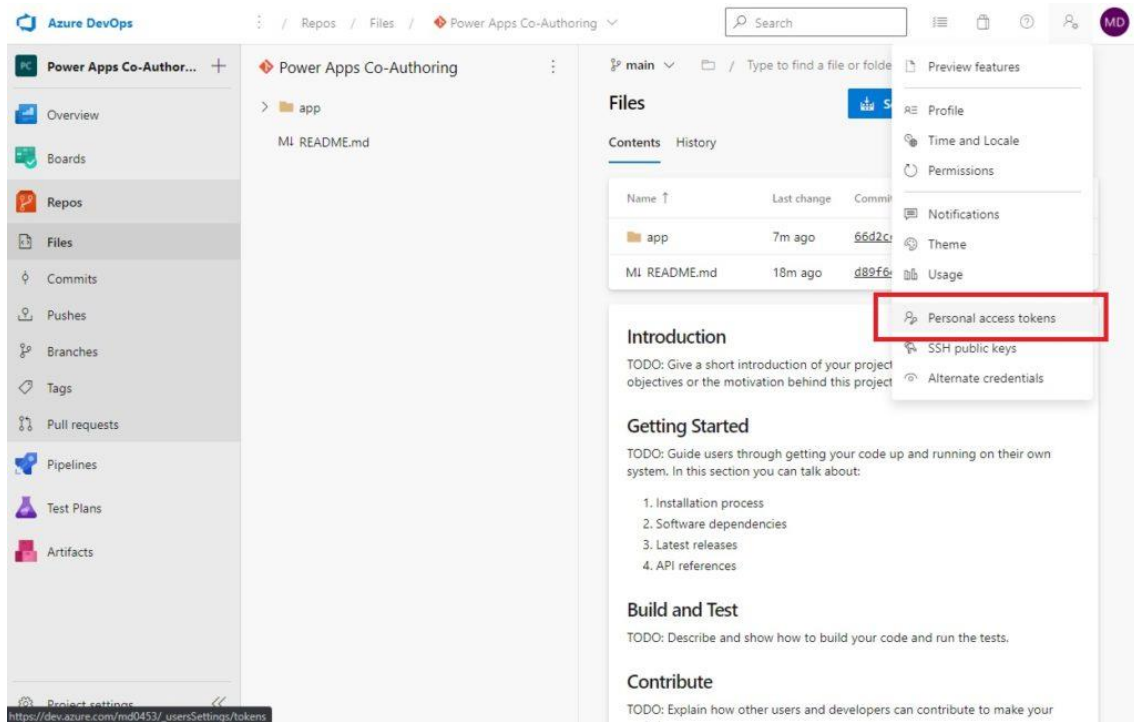
## Crear un token de acceso personal

Se requiere un tipo especial de contraseña denominada token de acceso personal para que Power Apps inicie sesión en el repositorio de Azure Dev Ops. Un desarrollador creará el token de acceso personal y lo compartirá con el resto del equipo de desarrollo. Debemos volver a Azure Dev Ops para crear uno.

*Importante: un token de cuenta personal no es la contraseña de su cuenta de Azure Dev Ops.*

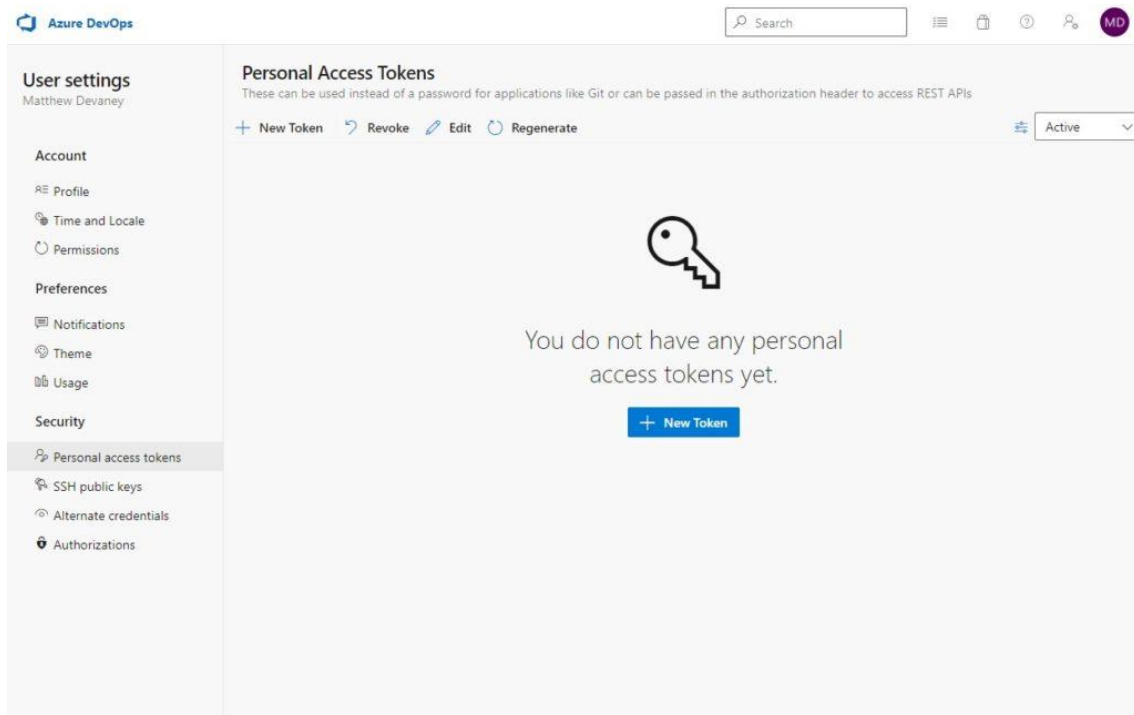


Abra Azure Dev Ops y haga clic en el avatar en la esquina superior derecha. Seleccione *Tokens* de *acceso* *personal*.

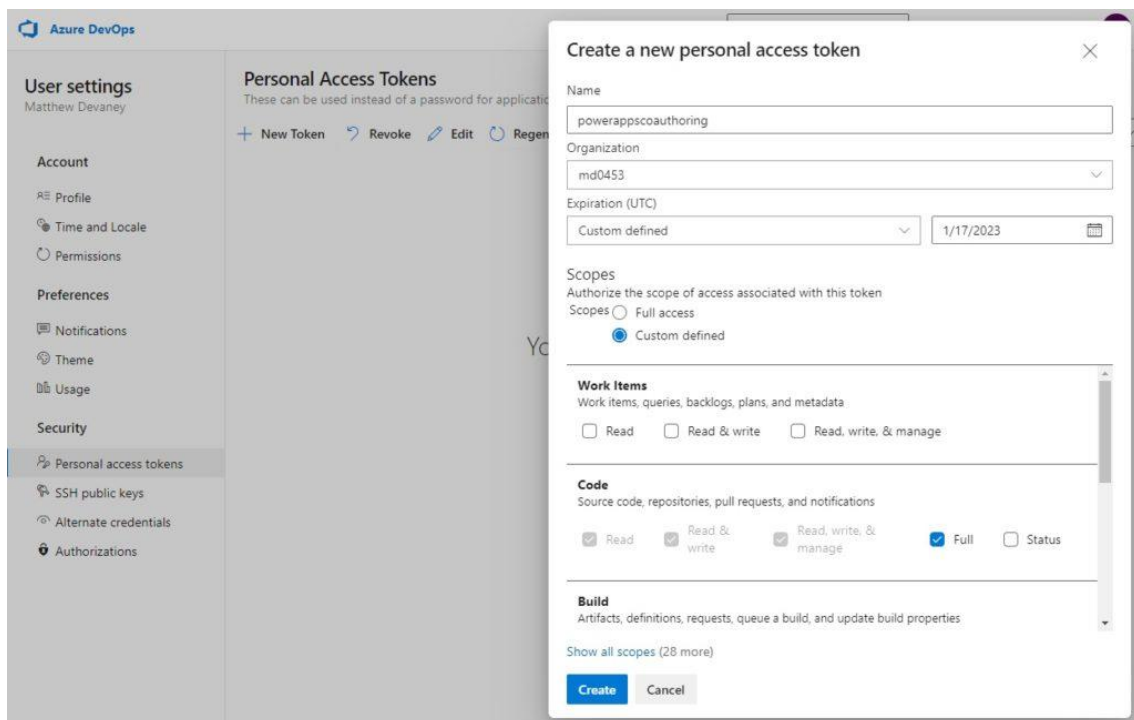




Crear un nuevo token .



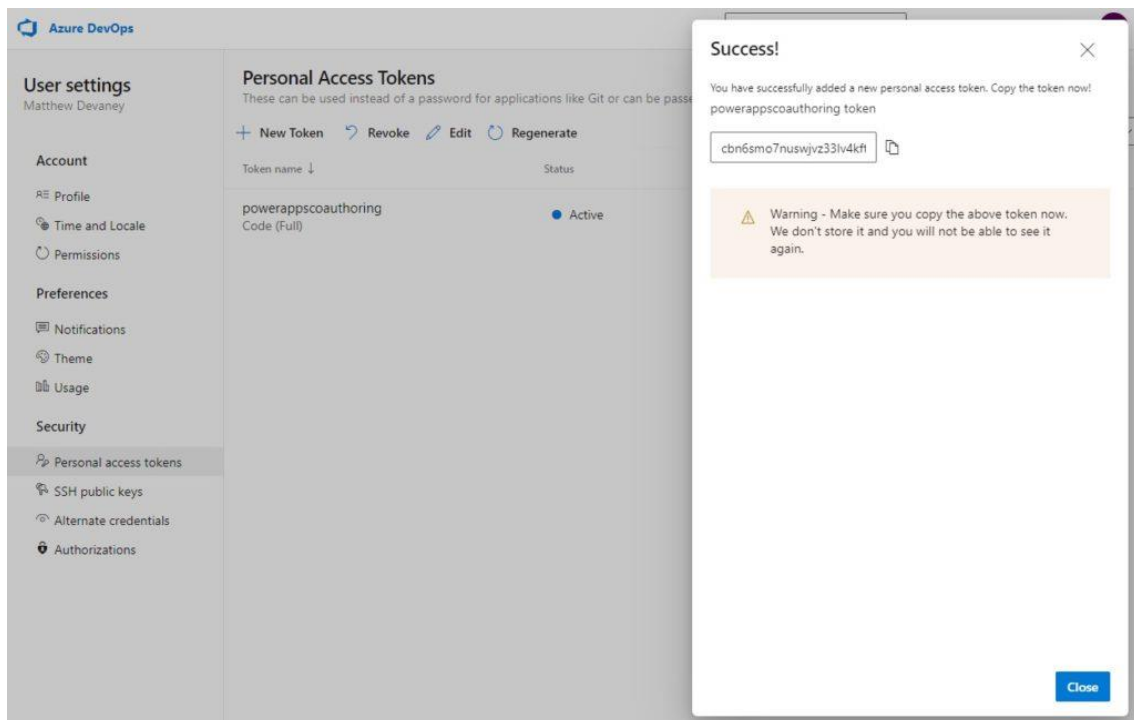
Nombre el token de acceso personal, elija una fecha de vencimiento y marque la casilla de verificación *completa* para darle a Power Apps control total sobre el código fuente y luego haga clic en *Crear* .



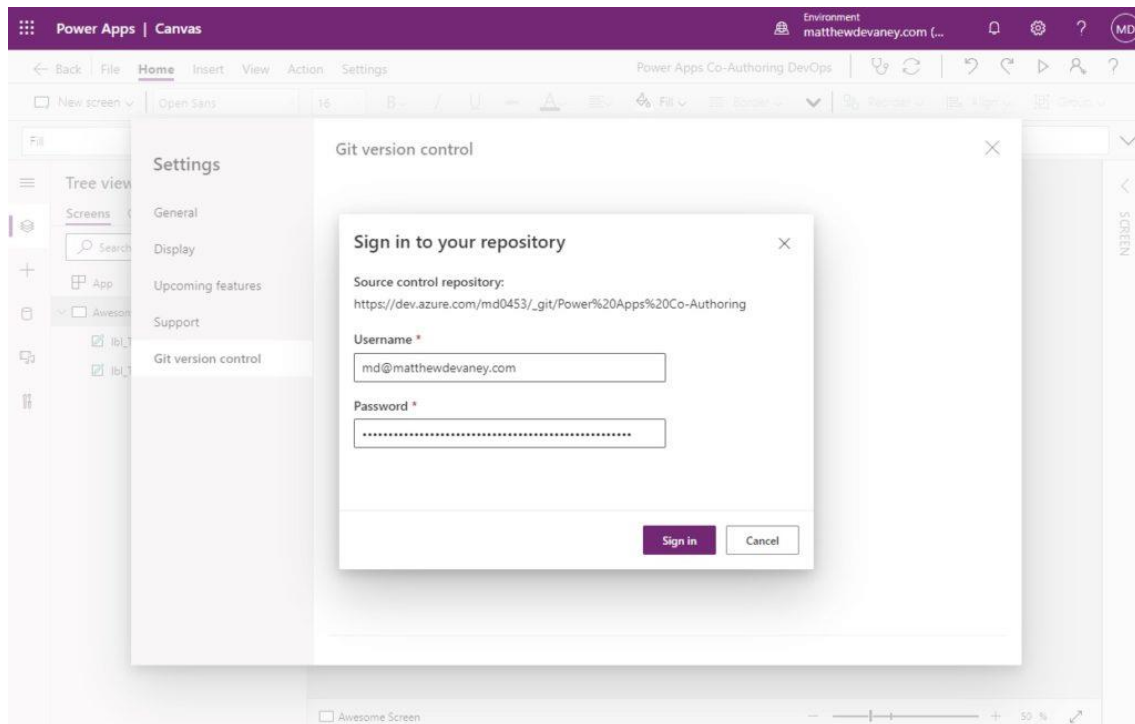
## Iniciar sesión en el repositorio de Azure Dev Ops

Azure Dev Ops se abrirá en la pantalla de éxito después de hacer clic en Crear token. Copie el token de acceso personal al portapapeles.

*Importante: asegúrese de guardar el token de acceso personal para usarlo en el futuro. Solo podemos ver el token una vez. Será necesario cada vez que un desarrollador abra la aplicación para realizar una edición .*

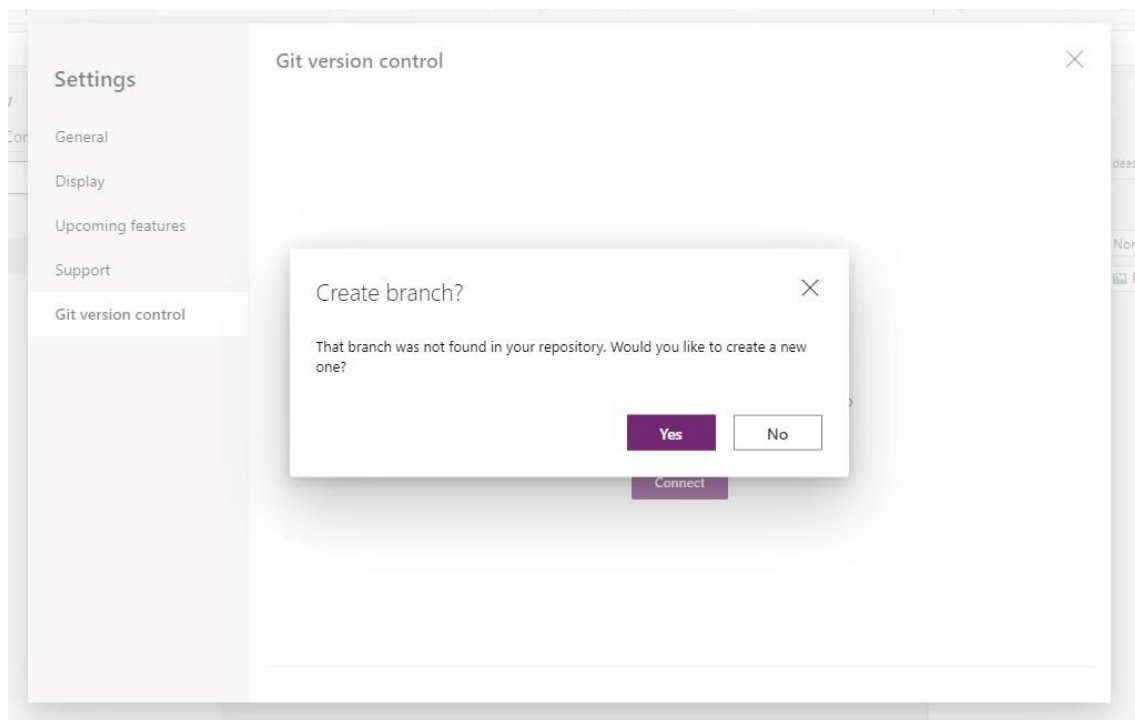


Luego regrese a Power Apps e inicie sesión con su *nombre de usuario* de Azure Dev Ops y su token de acceso personal como *contraseña* .

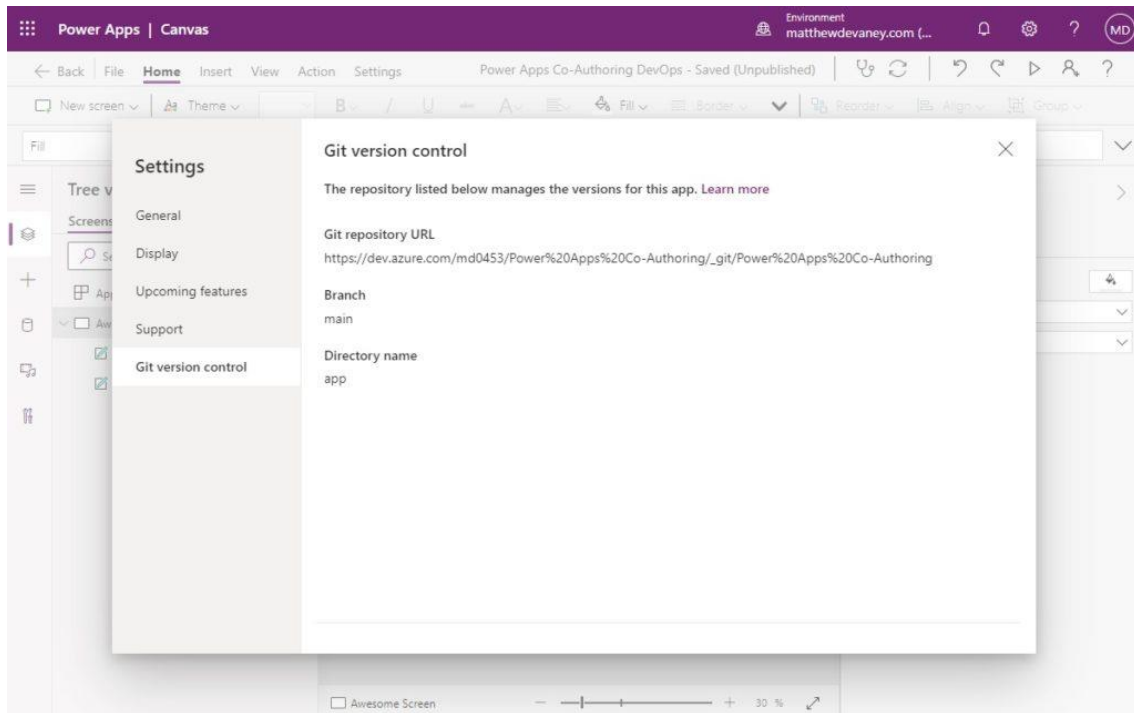


## Finalizar la configuración del control de versiones de Git

Ya casi está listo. Aparece otro mensaje que nos pregunta si queremos crear una nueva rama en el repositorio. Seleccione **Sí**.

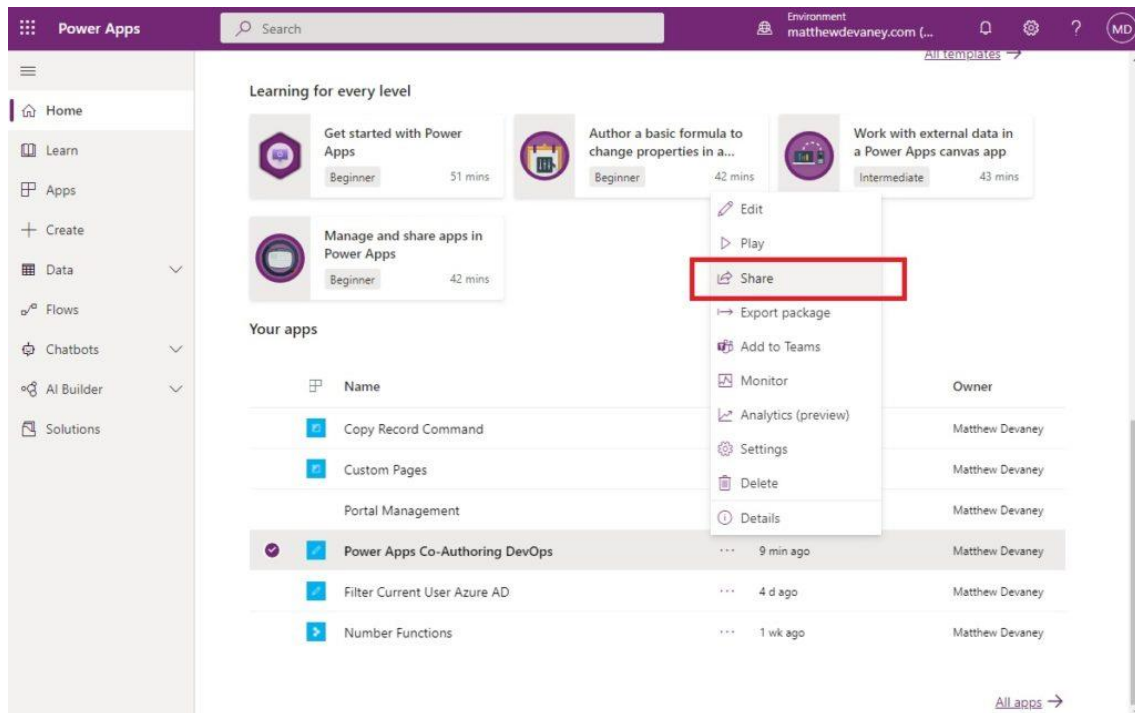


Ahora hemos configurado el control de versiones de Git en Power Apps.

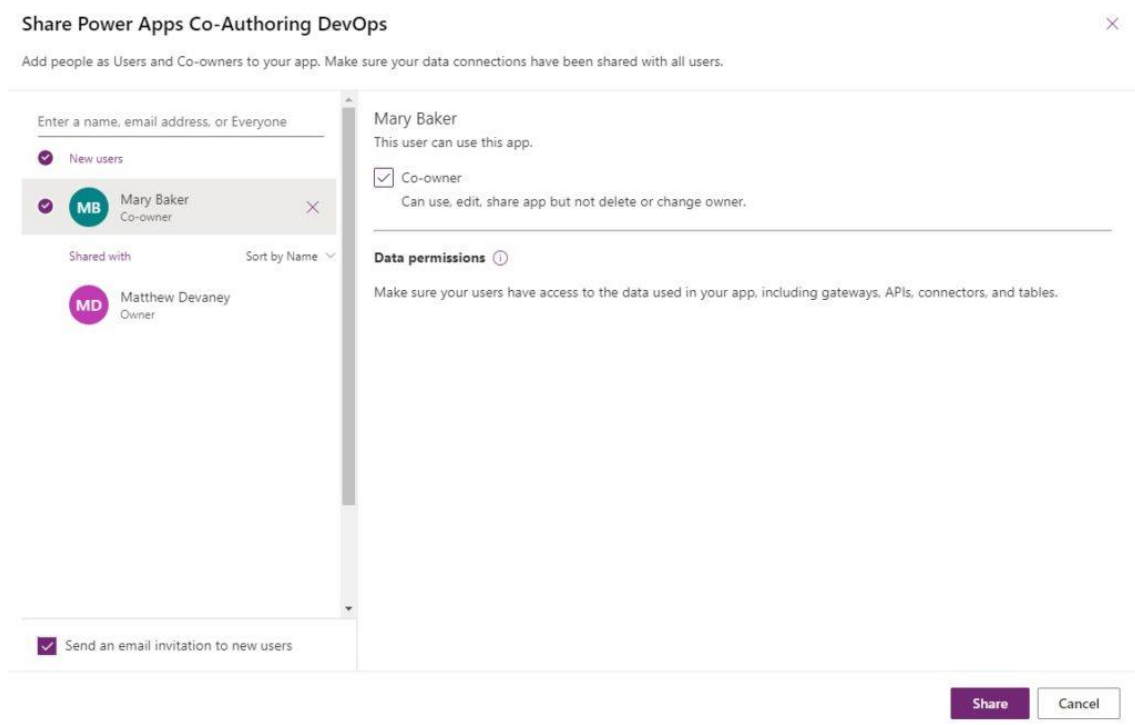


### Compartir Power Apps con otro desarrollador como copropietario

Ahora que el control de versiones de Git está configurado, varios desarrolladores pueden trabajar en la misma aplicación de Power Apps. Pero aún tenemos que compartir la aplicación y el repositorio de Azure Dev Ops con esos desarrolladores. Vaya a [make.powerapps.com](https://make.powerapps.com), busque la aplicación que conectamos a Azure Dev Ops, haga clic en los tres puntos y seleccione **Compartir**.

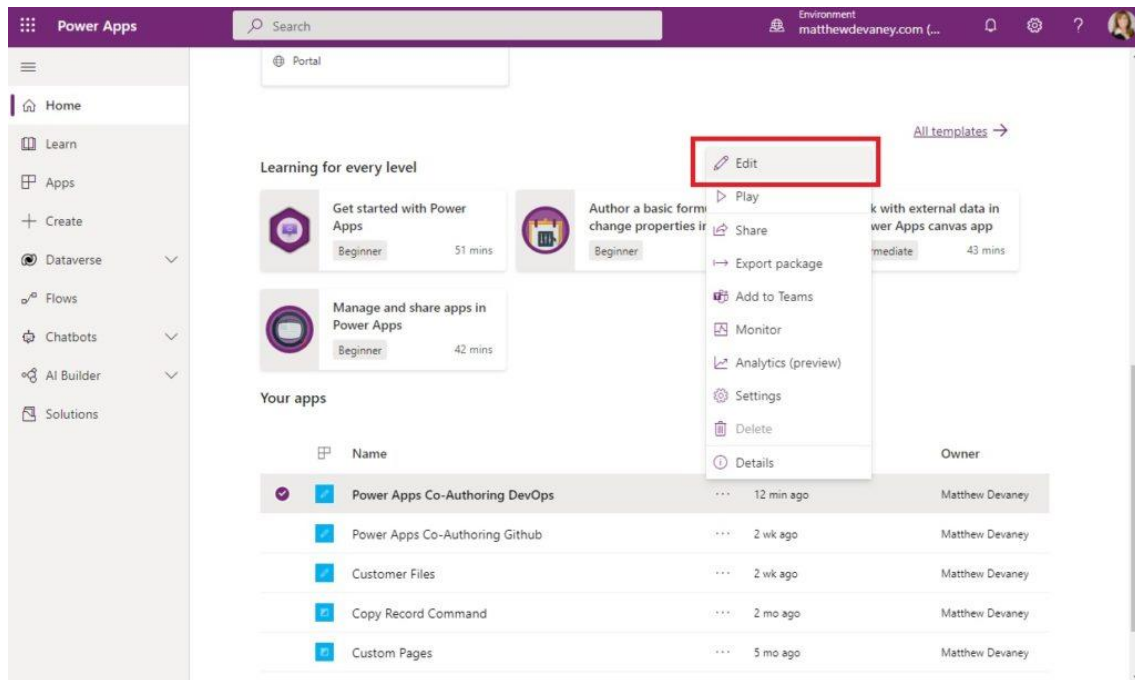


Busca a los demás desarrolladores a los que deseas dar acceso a la aplicación y marca la casilla para convertirlos en copropietarios. Selecciona Compartir cuando hayas terminado.



## Iniciar sesión en una aplicación de Power Apps al mismo tiempo que otro desarrollador

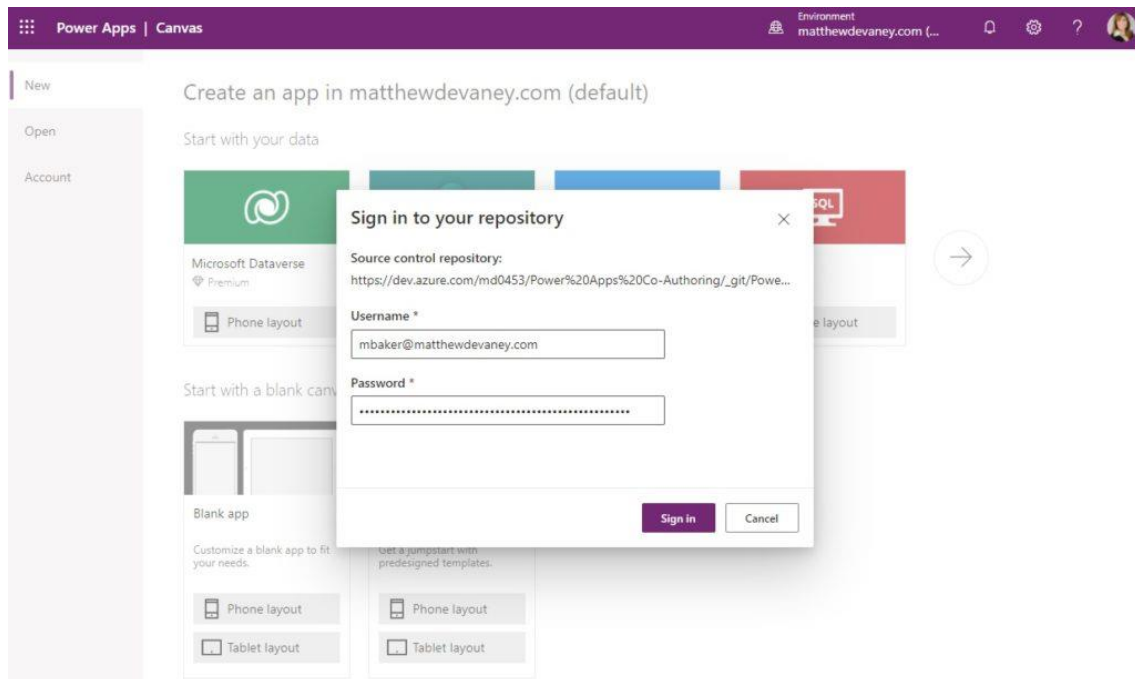
Ahora varios desarrolladores pueden trabajar en una sola aplicación Power Apps a la vez. Probemos la función para asegurarnos de que funciona. Pídale al desarrollador con el que compartió la aplicación que la abra en Power Apps Studio mientras estamos conectados.



Se le solicitará al segundo desarrollador que inicie sesión en el repositorio de Azure Dev Ops con un nombre de usuario y una contraseña.

- Nombre de usuario: es el nombre de usuario de la cuenta de Azure Dev Ops del desarrollador, no su nombre de usuario de Power Apps. El desarrollador deberá registrarse para obtener una cuenta si aún no tiene una.
- Contraseña: este es el token de acceso personal que creamos anteriormente, no la contraseña de la cuenta de Azure Dev Ops del desarrollador.

Una vez completado, Power Apps se abrirá normalmente.

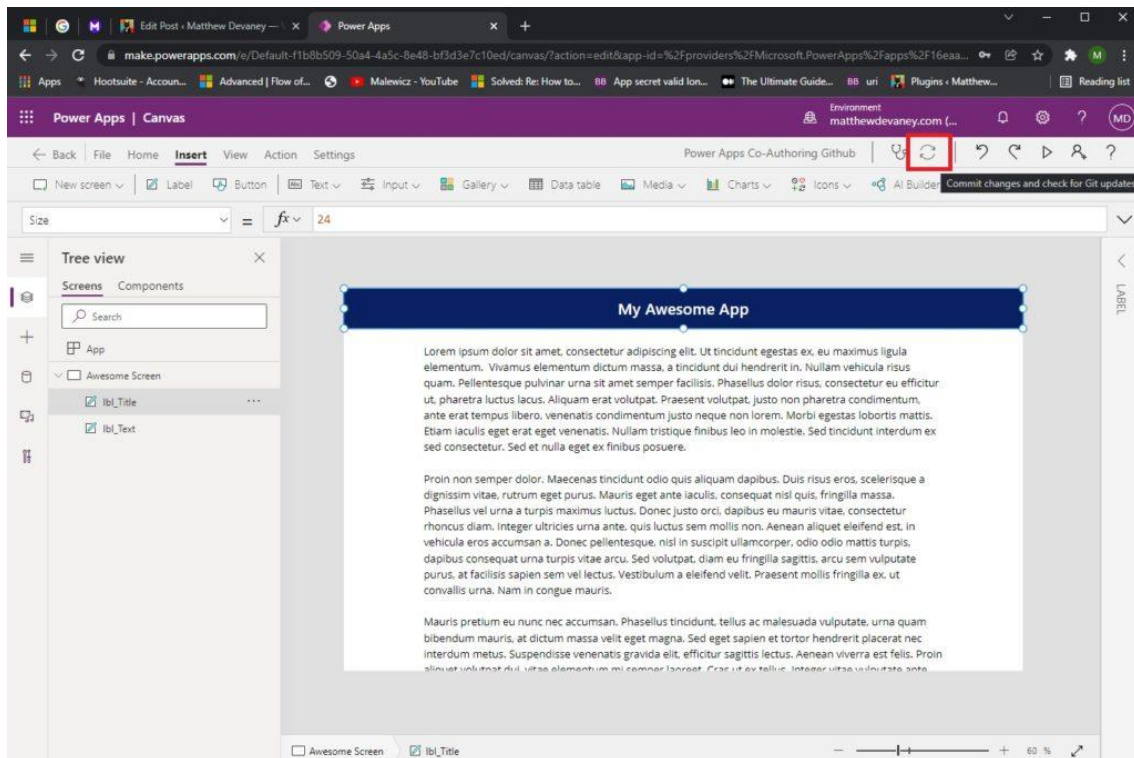


## Confirmar cambios y comprobar si hay actualizaciones

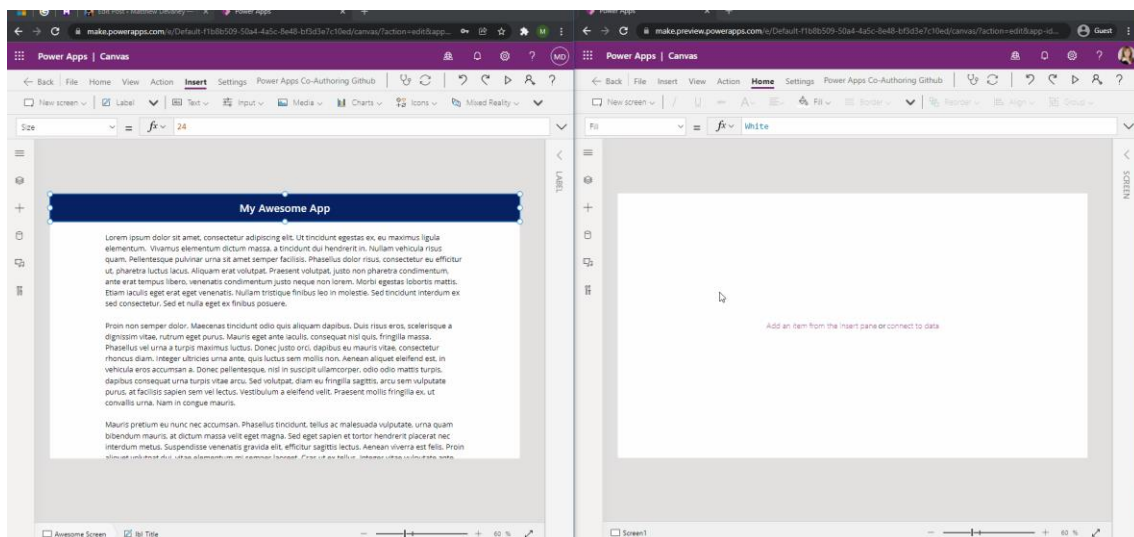
Ahora ambos desarrolladores están trabajando dentro de la misma aplicación. ¿Cómo pueden ver el progreso del otro? Power Apps no tiene una función de edición en vivo como Word, Excel o PowerPoint Online. Los desarrolladores deben guardar y sincronizar sus cambios manualmente. Esto tiene un propósito: no queremos enviar código sin terminar a la pantalla de otros desarrolladores y causar errores.

Cuando hemos completado una función, queremos enviarla al repositorio de Git para que el otro desarrollador pueda actualizar su instancia de la aplicación. Para ello, presione el botón de *sincronización* en el menú superior de Power Apps Studio. Al hacer esto, se confirman nuestros cambios en el repositorio, se fusionan con el código existente y se actualiza nuestra aplicación para mostrar los cambios realizados por otros desarrolladores.

*Importante: si dos desarrolladores trabajan en la misma propiedad de un control, prevalece la última edición realizada. El cambio más reciente reemplaza el cambio anterior. Debemos planificar con anticipación quién está trabajando en qué funciones o secciones de la aplicación para asegurarnos de no sobrescribir el trabajo de los demás.*



Así es como se ve el guardado y la sincronización para dos desarrolladores que trabajan en la misma aplicación.

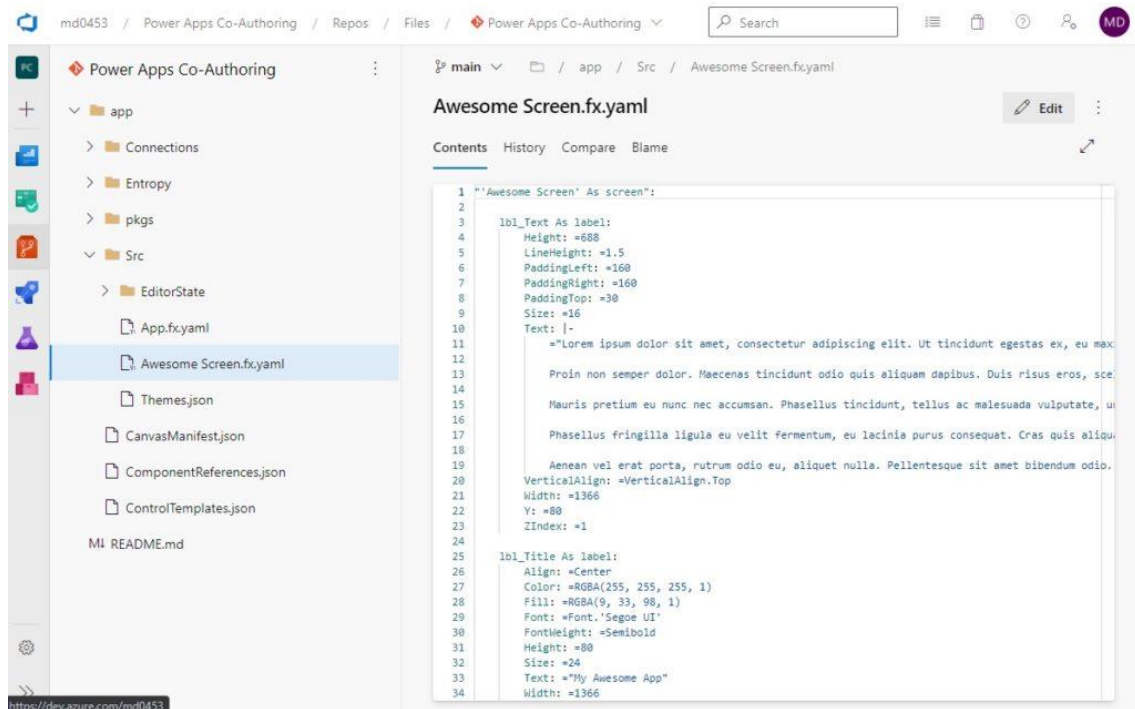


## Ver el código fuente de Power Apps en un repositorio de Azure Dev Ops

Después de que Power Apps Studio sincronice la aplicación con el repositorio, podemos ver el código fuente en Azure Dev Ops. Vaya a la carpeta *multiple-developer-canvas*



`app/app/Src` y seleccione el archivo `yaml` para la pantalla que editamos en Power Apps. Todo el código para la pantalla, sus controles y sus propiedades se encuentran aquí.



Podemos hacer algunas cosas muy útiles con el código fuente de Power Apps en Azure Dev Ops:

- Realice una revisión de código rápidamente: es más fácil revisar todo el código de una pantalla en Azure Dev Ops que explorar cada control individual y sus propiedades en Power Apps
- Editar el código sin formato: si presionamos la tecla de punto (.) mientras miramos la pantalla en Azure Dev Ops, se abrirá el archivo `yaml` en un editor [de Visual Studio Code](#) en línea . Una vez que se realicen las actualizaciones, podemos presionar el botón de sincronización en Power Apps Studio para ver los cambios.
- Cambiar el nombre de variables y colecciones: Visual Studio Code tiene la capacidad de buscar y reemplazar texto en todos los archivos y carpetas. Esto facilita el cambio de nombre de variables y colecciones para toda la aplicación. Presione la combinación de teclas `CTRL + MAYÚS + F` para usar la función *Buscar y reemplazar en archivos* .

## **Consideraciones adicionales para la coautoría de Power Apps**

Hay algunas cosas más que debe saber sobre la coautoría de Power Apps antes de usarlo:

- Esta es una función experimental y debe tratarse como tal. No la use en una aplicación de producción hasta que esté seguro de que funciona. No es el producto final y se realizarán más funciones o cambios antes de que la coautoría esté disponible para el público en general.
- Las nuevas Power Apps deben configurarse para la colaboración de forma individual. No hay forma de habilitar una configuración global para todas las aplicaciones que permita que varios desarrolladores trabajen al mismo tiempo.
- Actualmente, no es posible desconectar una aplicación conectada a Git. La única opción es eliminar el repositorio Git.
- No hay ninguna configuración disponible para restringir el acceso de un token personal a un único repositorio. Una vez que se otorga un token de acceso personal, este otorga acceso a todos los repositorios bajo la misma cuenta. Cada desarrollador debe usar el mismo token de acceso personal.