

15.08.2025

**JENNY KATHERINE MONTENEGRO RAMOS**

P R E S E N T A

**G I T H U B**

15.08.2025

GitHub es una plataforma basada en la nube donde puedes almacenar, compartir y trabajar junto con otros usuarios para escribir código.



Git es un sistema de control de versiones que realiza un seguimiento de los cambios en los archivos. Git es especialmente útil cuando un grupo de personas y tú estáis haciendo cambios en los mismos archivos al mismo tiempo.

# ¿QUÉ ES GITHUB Y GIT?

# VENTAJAS DE USAR GITHUB

Control de versiones	Documentación centralizada
Colaboración y trabajo en equipo	Gestión de proyectos y tareas
Revisión y mejora continua del código	Ofrece acceso a una amplia comunidad de desarrolladores
Automatización de tareas EQUIPO 5	Permite controlar quién puede acceder a los repositorios y realizar cambios

GitHub ofrece numerosas ventajas para desarrolladores y empresas, incluyendo control de versiones, colaboración eficiente, automatización de tareas, documentación centralizada y una gran comunidad de desarrolladores. Permite a los equipos trabajar juntos en proyectos, realizar seguimiento de cambios, automatizar tareas repetitivas y encontrar recursos y soporte.

15.08.2025

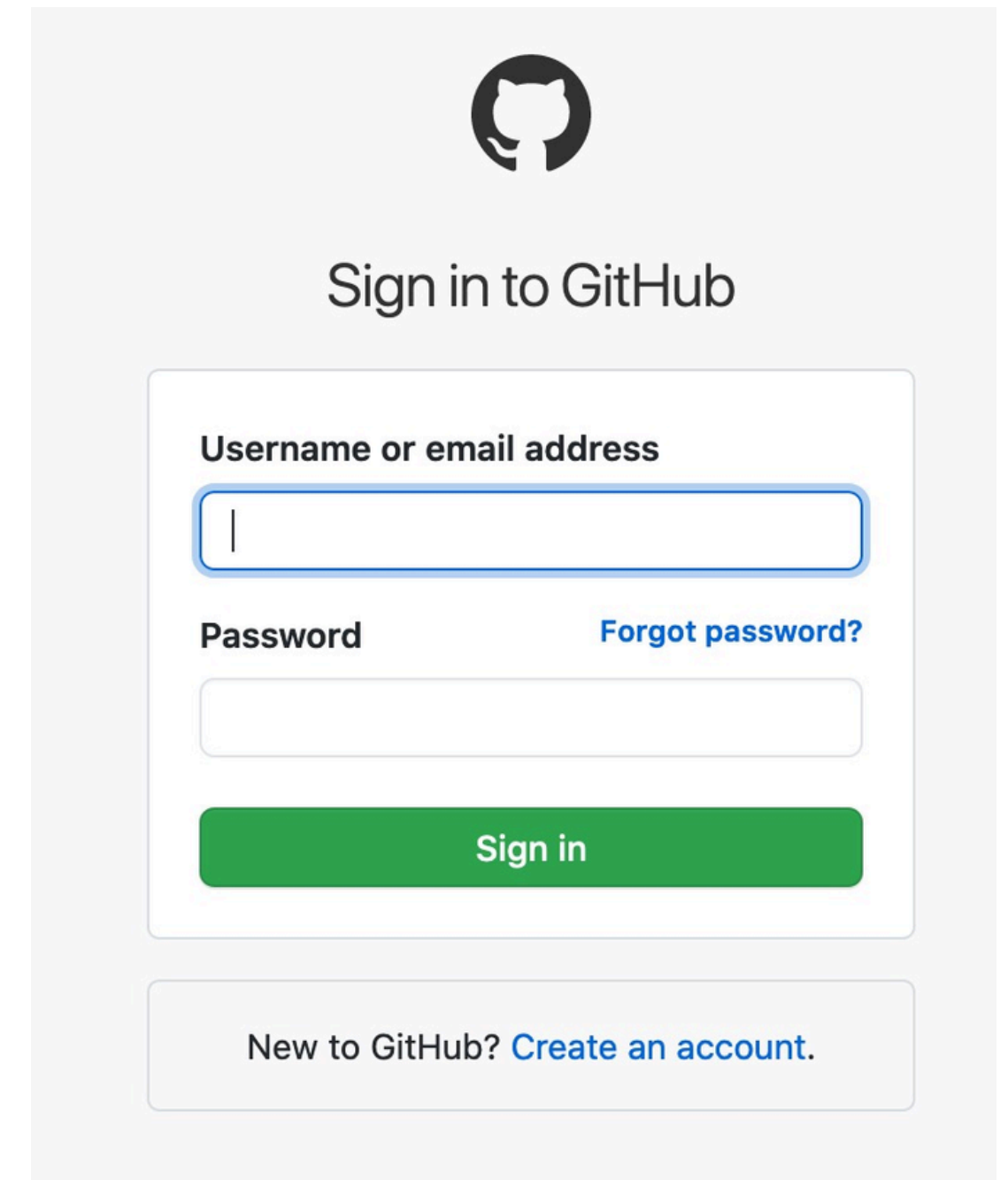
1. Vaya a <https://github.com>
1. Haga clic en Registrarse.
1. Como alternativa, haz clic en Continue with Google para registrarte con el inicio de sesión de redes sociales.
1. Sigue las indicaciones para crear tu cuenta personal.

Durante el registro, se te pedirá que verifiques tu dirección de correo electrónico. Sin una dirección de correo electrónico verificada, no podrás completar algunas tareas básicas de GitHub, como crear un repositorio. Algunas empresas crean cuentas de usuario administradas para sus usuarios. No puedes registrarte para obtener una cuenta personal con una dirección de correo electrónico que ya se haya verificado para un cuenta de usuario administrada.

Si tienes problemas para comprobar la dirección de correo electrónico, hay algunos pasos de solución de problemas que puedes seguir. Para más información, consulta [Verificar tu dirección de correo electrónico](#).

#### Pasos siguientes

Ahora que has creado tu cuenta personal, empezaremos a explorar los conceptos básicos de GitHub. En el tutorial siguiente, Hola mundo, obtendrá información sobre los repositorios y cómo crear uno, y se le presentarán conceptos como ramificación, confirmaciones y solicitudes de extracción. Se recomienda encarecidamente configurar 2FA para tu cuenta. 2FA es una capa adicional de seguridad que puede ayudar a mantener tu cuenta segura. Para más información, consulta [Configurar la autenticación de dos factores](#).

A screenshot of the GitHub sign-in page. At the top is the GitHub logo (Octocat). Below it is the text "Sign in to GitHub". There are two input fields: "Username or email address" and "Password". To the right of the password field is a link "Forgot password?". Below the password field is a green button labeled "Sign in". At the bottom, there is a link "New to GitHub? Create an account.".

Sign in to GitHub

Username or email address

Password [Forgot password?](#)

Sign in

New to GitHub? [Create an account.](#)

# CREACIÓN DE CUENTA EN GITHUB

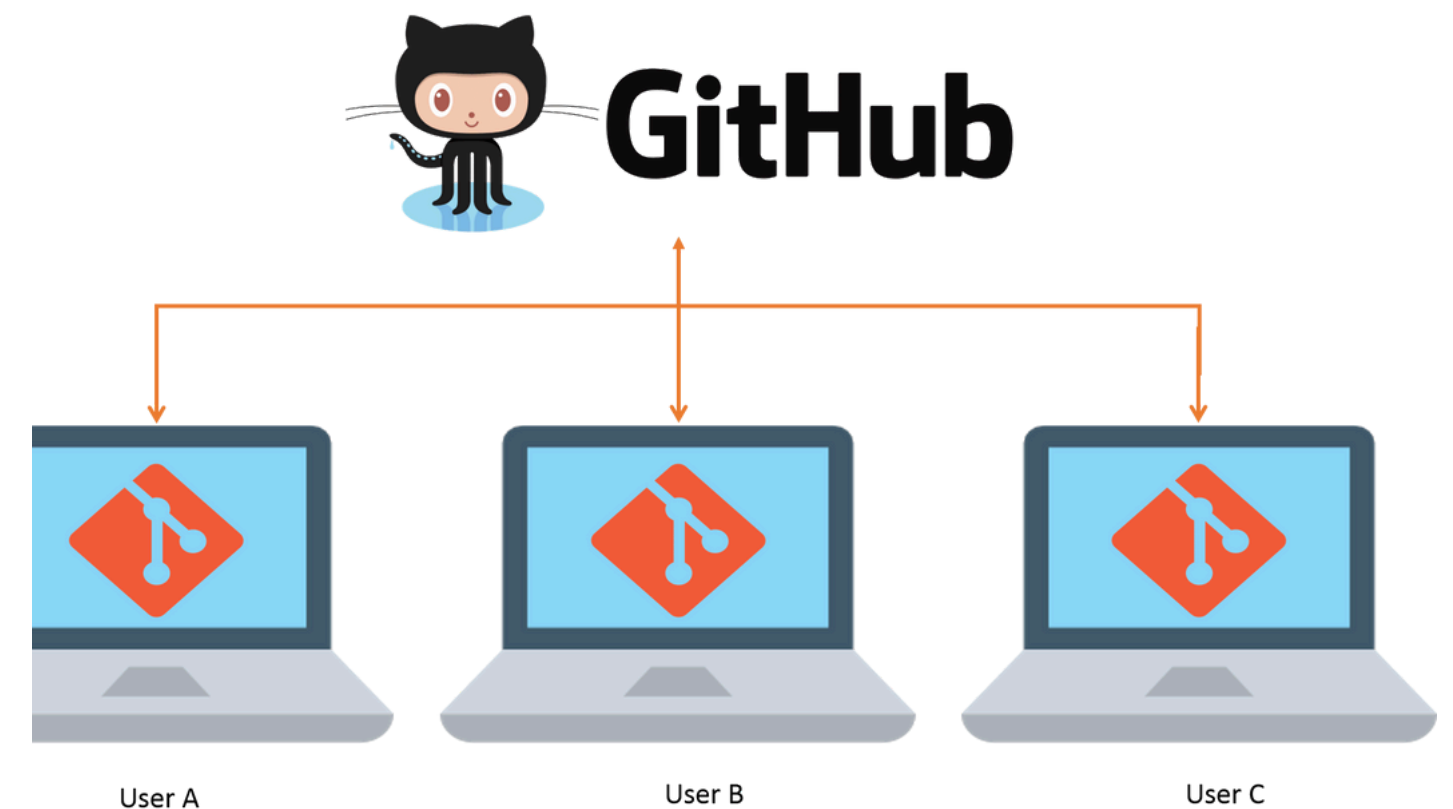
### Paso 1: Creación de un repositorio

Lo primero será crear un repositorio. Puedes considerar un repositorio como una carpeta que contiene elementos relacionados, como archivos, imágenes, vídeos o incluso otras carpetas. Normalmente, un repositorio agrupa los elementos que pertenecen al mismo "proyecto" o aquello en lo que trabajas.

A menudo, los repositorios incluyen un archivo LÉAME con información sobre el proyecto. Los archivos LÉAME están escritos en Markdown, un lenguaje fácil de leer y escribir para aplicar formato al texto sin formato. En el siguiente tutorial, [Configuración del perfil](#) obtendrás más información sobre Markdown.

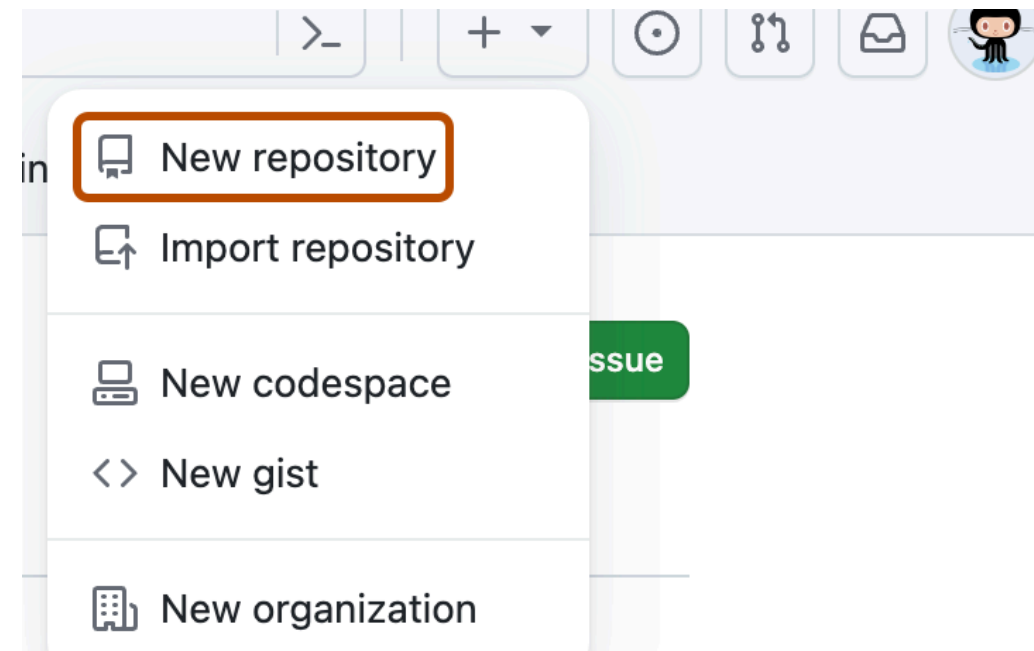
GitHub permite agregar un archivo LÉAME al mismo tiempo que crea el nuevo repositorio. GitHub también ofrece otras opciones comunes, tales como un archivo de licencia, pero no tienes que seleccionar ninguna de ellas ahora mismo.

El repositorio hello-world puede ser un lugar donde almacene ideas, recursos o incluso comparta y debata cosas con otros usuarios.



# CREACIÓN DE UN REPOSITORIO

En la esquina superior derecha de cualquier página, selecciona y luego haz clic en Nuevo repositorio.



En el cuadro "Nombre del repositorio", escriba hello-world.

En el cuadro "Descripción", escriba una breve descripción. Por ejemplo, escribe "Este repositorio es para practicar el flujo de GitHub".

Seleccione si el repositorio será Público o Privado.

Seleccione Agregar un archivo LÉAME.

Haga clic en Create repository (Crear repositorio).

# CREACION DE UNA RAMA

15.08.2025

La ramificación te permite tener versiones diferentes de un repositorio en una ocasión.

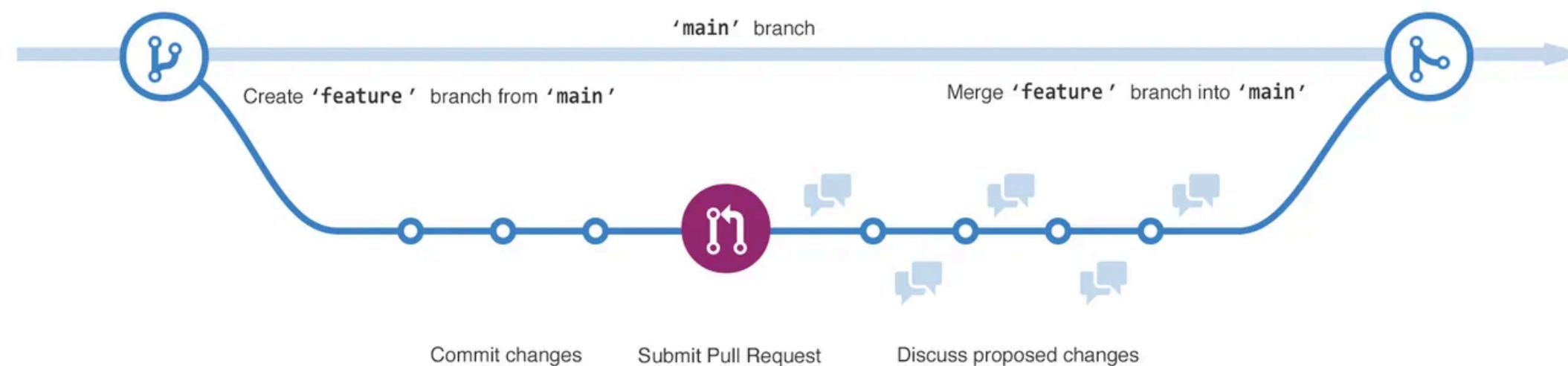
De manera predeterminada, el repositorio tiene una rama llamada main que se considera la rama definitiva. Puede crear ramas adicionales fuera de main en el repositorio.

La ramificación es útil cuando quieres agregar características nuevas a un proyecto sin cambiar la fuente de código principal. El trabajo que se hace en las diferentes ramas no se mostrará en la rama principal sino hasta que la fusiones, lo cual veremos más adelante en esta guía. Puede utilizar ramas para experimentar y realizar ediciones antes de confirmarlas en main.

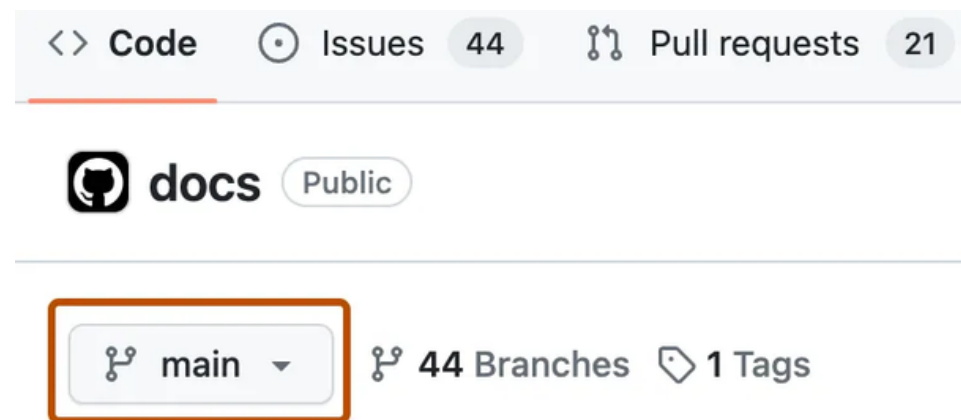
Al crear una rama a partir de la rama main, se realiza una copia, o instantánea, de main como estaba en ese momento en el tiempo. Si alguien más ha realizado cambios en la rama main mientras trabajaba en el rama, podría extraer esas actualizaciones.

En este diagrama se muestra:

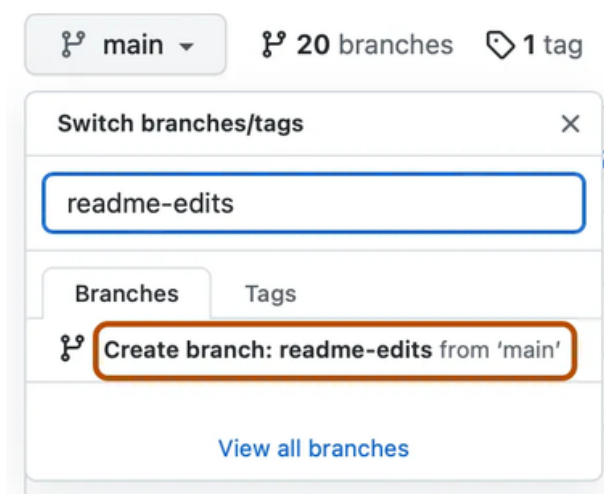
- La rama main
- Una nueva rama denominada feature
- El recorrido que feature lleva a cabo fases para "Confirmar cambios", "Enviar solicitud de incorporación de cambios" y "Analizar los cambios propuestos" antes de combinarlos en main



1. Haga clic en la pestaña Código del repositorio hello-world.
2. Encima de la lista de archivos, haga clic en el menú desplegable que indica main.



3. Escriba un nombre de rama, readme-edits, en el cuadro de texto.
4. Haga clic en Crear rama: readme-edits desde main.



Ahora tiene dos ramas, main y readme-edits. Ahora mismo, se ven idénticas. A continuación, agregarás cambios a la rama readme-edits nueva.



# REALIZACION Y CONFIRMACION DE CAMBIOS

Cuando has creado una rama en el paso anterior, GitHub te ha llevado a la página de código de la nueva rama readme-edits, que es una copia de main.

Puedes hacer y guardar cambios a los archivos de tu repositorio. En GitHub, los cambios guardados se llaman confirmaciones. Cada confirmación tiene un mensaje de confirmación asociado, el cual es una descripción que explica por qué se realizó algún cambio en particular. Los mensajes de confirmación capturan la historia de tus cambios para que otros contribuyentes puedan entender lo que hiciste y por qué.



1. En la rama readme-edits que has creado, haz clic en el archivo README.md.
2. Para editar el archivo, haz clic en .
3. En el editor, escribe un poco sobre ti mismo.
4. Haga clic en Commit changes (Confirmar cambios).
5. En el cuadro "Confirmar cambios", escribe un mensaje de confirmación que describa los cambios.
6. Haga clic en Confirmar cambios.

Estos cambios solo se realizarán en el archivo LÉAME de la rama readme-edits, por lo que ahora contiene contenido diferente al de main.

15.08.2025

Ahora que tiene cambios en una rama derivada de main, puede abrir una solicitud de incorporación de cambios.

Las solicitudes de cambios son el núcleo de la colaboración en GitHub. Cuando abres una solicitud de cambios, estás proponiendo tus cambios y solicitando que alguien revise e integre tu contribución y la fusione en su rama. Las solicitudes de cambio muestran diffs o diferencias del contenido de ambas ramas. Los cambios, adiciones y sustracciones se muestran en varios colores. Tan pronto hagas una confirmación, puedes abrir una solicitud de cambios y comenzar un debate, incluso antes de que se termine el código.

En este paso, abrirás una solicitud de cambios en tu propio repositorio y, después, la combinarás personalmente. Es una forma genial de practicar el flujo de GitHub antes de trabajar en proyectos más grandes.

# **APERTURA DE UNA SOLICITUD DE CAMBIOS**

## EQUIPO 5

15.10.2030

1. Haga clic en la pestaña Solicitudes de incorporación de cambios del repositorio hello-world.
2. Haz clic en Nueva solicitud de incorporación de cambios.
3. En el cuadro Comparaciones de ejemplo, seleccione la rama que ha creado, readme-edits, para compararla con main (la original).
4. Mira tus cambios en los diffs en la página de Comparar, asegúrate que son lo que quieres enviar.



5. Haga clic en Create pull request (Crear solicitud de incorporación de cambios).
6. Dale un título a tu solicitud de cambios y escribe una descripción breve de estos. Puedes incluir emojis y arrastrar y soltar imágenes y gifs.
7. Haga clic en Crear solicitud de incorporación de cambios.

### **Revisión de una solicitud de incorporación de cambios**

Cuando empieces a colaborar con otros usuarios, es la hora a la que les pedirías su revisión. Esto permite a los colaboradores comentar o proponer cambios en la solicitud de cambios antes de combinar los cambios en la rama main.

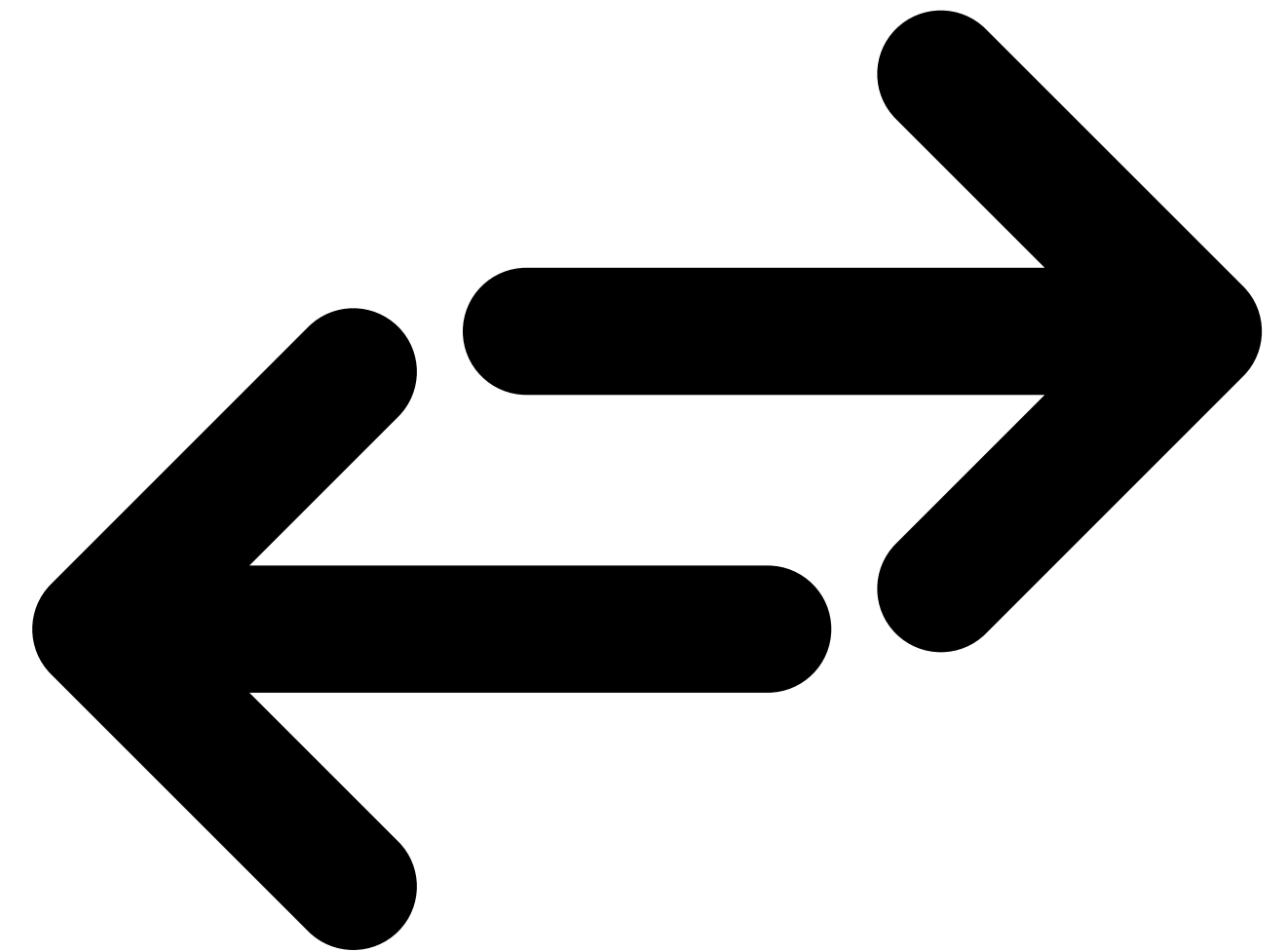
En este tutorial no se tratará la revisión de las solicitudes de cambios, pero si te interesa obtener más información, consulta [Acerca de las revisiones de solicitudes de extracción](#). Como alternativa, prueba el curso "Revisión de solicitudes de cambios" de [GitHub Skills](#).

# FUSIÓN DE LA SOLICITUD DE CAMBIOS

---

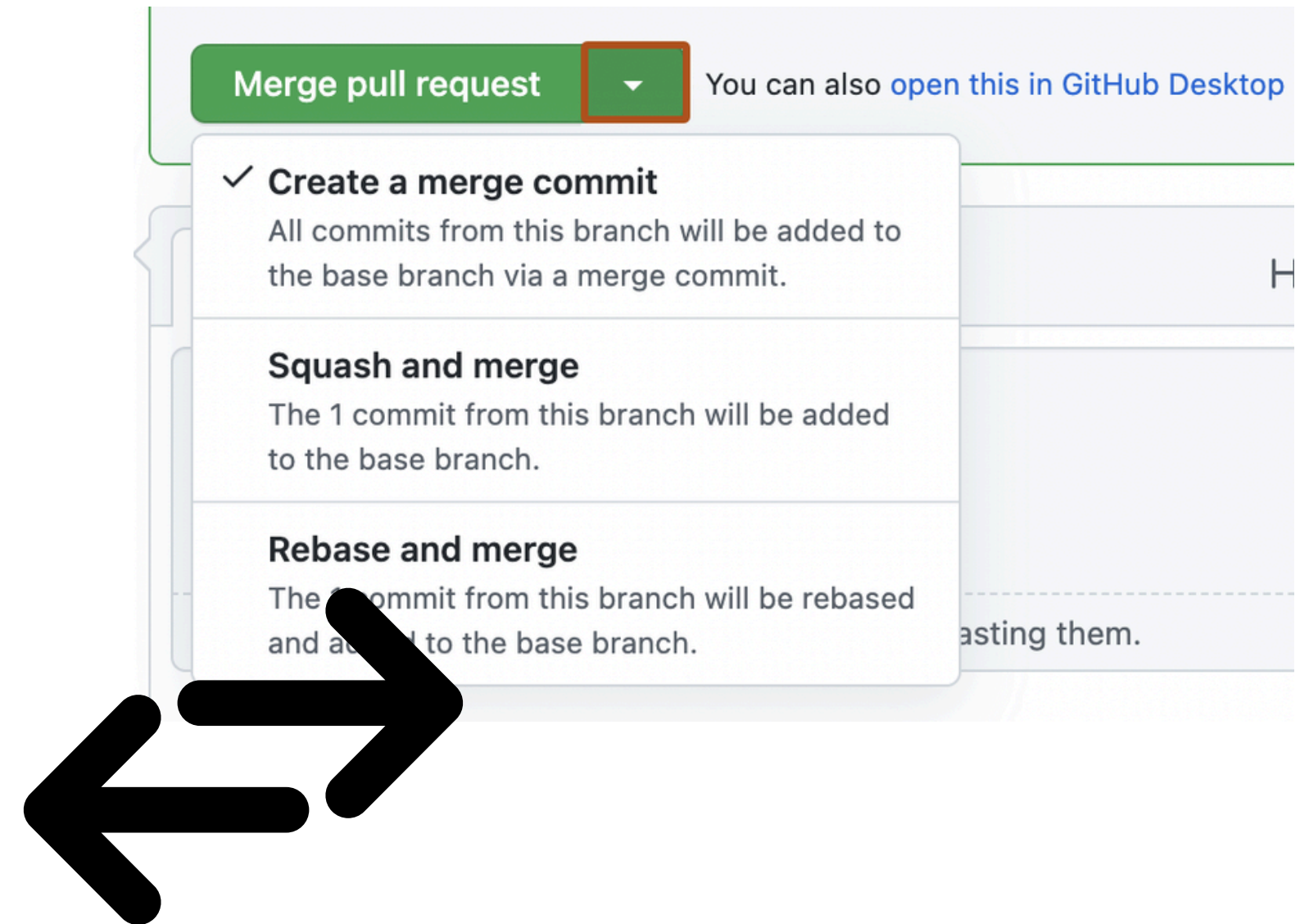
En este paso final, combinará la rama readme-edits en la rama main. Después de combinar la solicitud de incorporación de cambios, los cambios en la rama readme-edits se incorporarán a main.

En ocasiones, una solicitud de cambios podría introducir cambios en el código que entren en conflicto con el código existente en main. Si existe cualquier conflicto, GitHub te alertará sobre el código conflictivo y no dejará que se combine hasta que se resuelvan los conflictos. Puedes hacer una confirmación que resuelva los conflictos o que utilice comentarios en la solicitud de cambios para debatir estos conflictos con tus miembros de equipo.



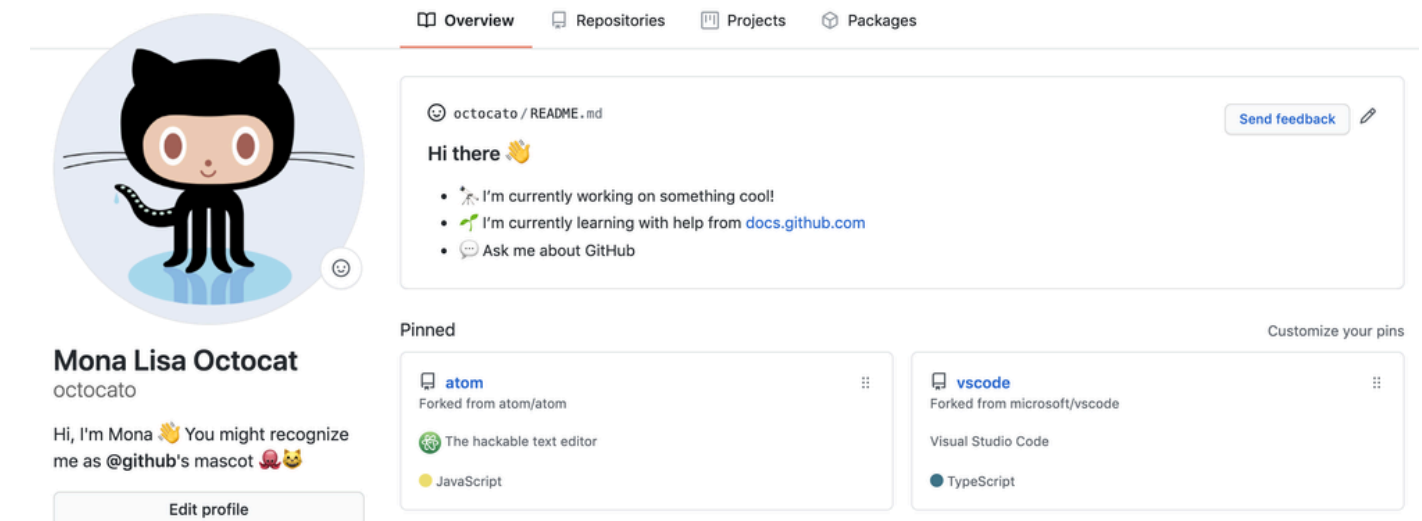
# FUSIÓN DE LA SOLICITUD DE CAMBIOS

1. En la parte inferior de la solicitud de incorporación de cambios, haz clic en Combinar solicitud de incorporación de cambios para combinar los cambios en main.
2. Haga clic en Confirmar combinación. Recibirás un mensaje indicando que la solicitud se fusionó con éxito y luego se cerró.
3. Haga clic en Eliminar rama. Ahora que la solicitud de incorporación de cambios se ha combinado y los cambios están en main, puede eliminar la rama readme-edits de forma segura. Si quieres hacer más cambios a tu proyecto, siempre puedes crear una rama nueva y repetir este proceso.
4. Haz clic de nuevo en la pestaña Código del repositorio hello-world para ver los cambios publicados en main.

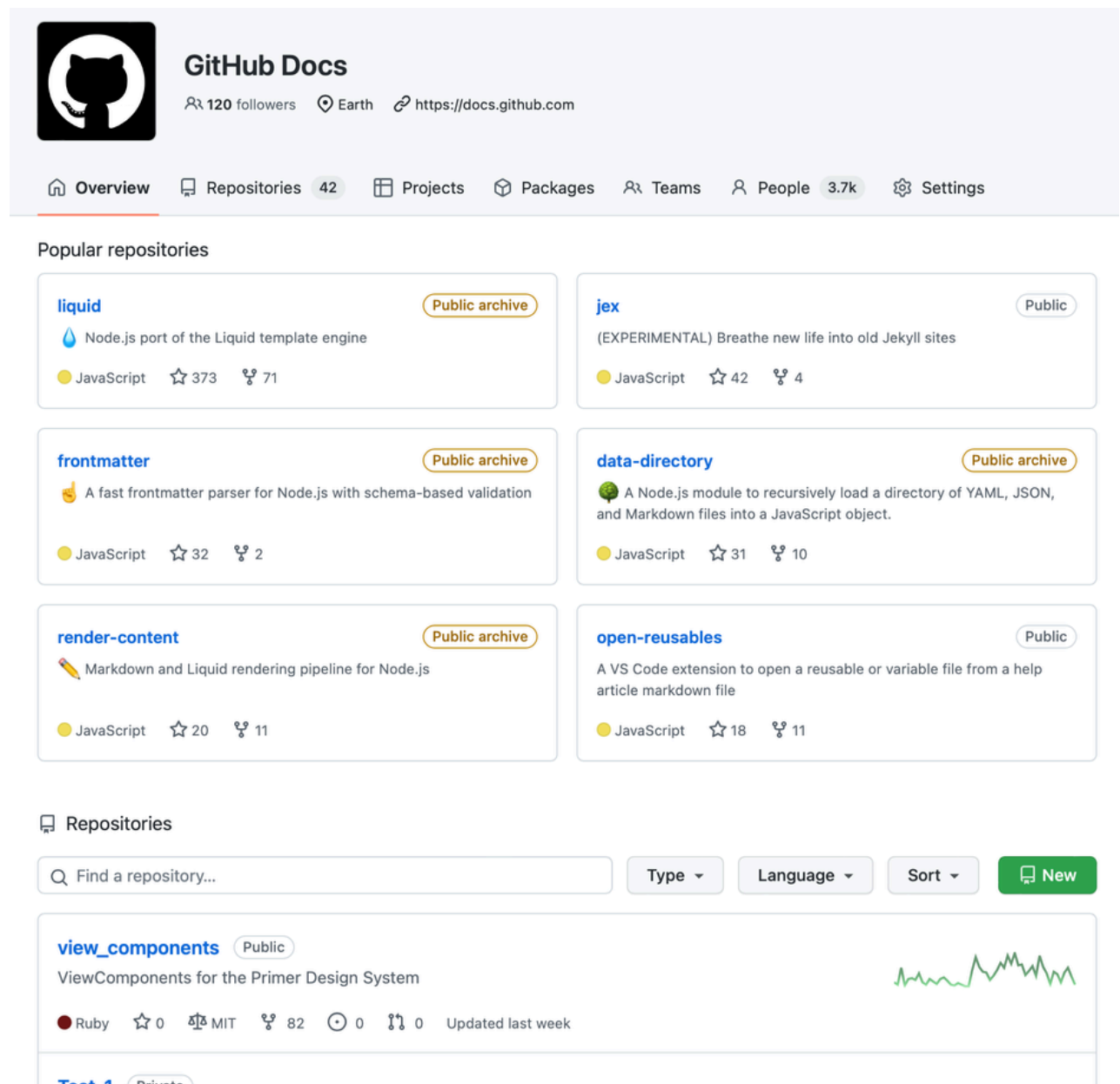


# CONFIGURAR UNA CUENTA DE GITHUB LICITUD DE CAMBIOS

1. Configurar perfil
2. Foto de perfil (opcional pero recomendable).
3. Descripción breve en tu bio (ej: Desarrollador junior, estudiante de ingeniería de sistemas).
4. Ubicación y sitio web (si tienes).
5. Elegir plan
6. GitHub ofrece plan gratuito con repositorios públicos y privados.
7. Si no necesitas más funciones, selecciona el plan Free.
8. Configurar Git localmente (opcional pero importante)







9.Configurar Git localmente (opcional pero importante)

10.Instala Git en tu computadora (En la parte inferior de la solicitud de incorporación de cambios, haz clic en Combinar solicitud de incorporación de cambios para combinar los cambios en main.

11.Haga clic en Confirmar combinación. Recibirás un mensaje indicando que la solicitud se fusionó con éxito y luego se cerró.

12.Haga clic en Eliminar rama. Ahora que la solicitud de incorporación de cambios se ha combinado y los cambios están en main, puede eliminar la rama readme-edits de forma segura. Si quieres hacer más cambios a tu proyecto, siempre puedes crear una rama nueva y repetir este proceso.

12.Haz clic de nuevo en la pestaña Código del repositorio hello-world para ver los cambios publicados en main.)

13.Configura tu usuario y correo en Git:

```
git config --global user.name "Tu Nombre"
```

```
git config --global user.email "tu_correo@example.com"
```

14.Esto asegura que tus commits muestren tu identidad.

# CONFIGURAR UNA CUENTA DE GITHUB

## LICITUD DE CAMBIOS

**GRACIAS**