

A thick dark blue vertical bar runs down the left side of the page. A blue arrow-shaped banner points to the right from this bar, containing the text 'DESPLIEGUE DE APLICACIONES WEB'. In the lower-left area, there are several thin, curved, light blue lines that sweep upwards and to the right.

DESPLIEGUE DE APLICACIONES WEB

PRÁCTICA DE CONTROL DE VERSIONES CON GIT Y GITHUB

AUTORES: Erik Aylagas López y Alberto Martínez Pérez
CURSO: 2º CFGS Desarrollo de Aplicaciones Web (DAW)
MÓDULO: Despliegue de Aplicaciones Web
FECHA DE ENTREGA: 24 de enero de 2024

ÍNDICE

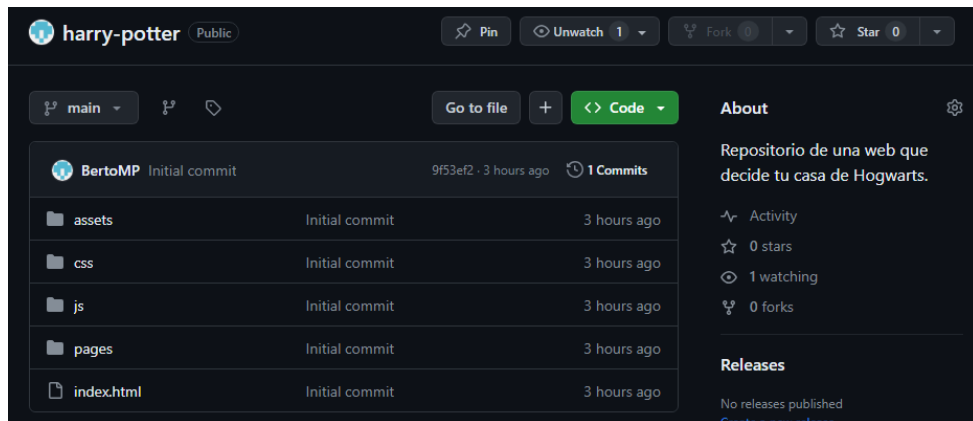
INTRODUCCIÓN.....	3
A. APLICACIÓN QUE SE VA A UTILIZAR.....	3
B. REPARTO DE ROLES	3
C. REPARTO DE TAREAS	4
CAMBIOS REALIZADOS POR EL DESARROLLADOR SENIOR.....	5
A. CAMBIOS REALIZADOS EN EL DIRECTORIO COMÚN	5
B. CAMBIOS REALIZADOS POR EL DESARROLLADOR SENIOR EN SUS FICHEROS.....	11
C. DESCARGAR CAMBIOS REALIZADOS POR NUESTRO COMPAÑERO	13
CAMBIOS REALIZADOS POR EL DESARROLLADOR JUNIOR.....	15
A. CAMBIOS EN LOS ARCHIVOS PROPIOS.....	15
B. CAMBIOS REALIZADOS SOBRE LOS ARCHIVOS COMUNES.....	18
CONFIGURACIÓN DE UN REPOSITORIO PÚBLICO	22
A. SINCRONIZACIÓN ENTRE REPOSITORIO LOCAL Y REMOTO	22
B. CREACIÓN DE UN ARCHIVO README.MD	25
C. CONFIGURACIÓN DE SEGURIDAD	29
D. AÑADIR COLABORADORES AL REPOSITORIO	30
E. ENVÍO DE UNA PULL-REQUEST	31
F. GESTIÓN DE UNA PULL-REQUEST	36

INTRODUCCIÓN

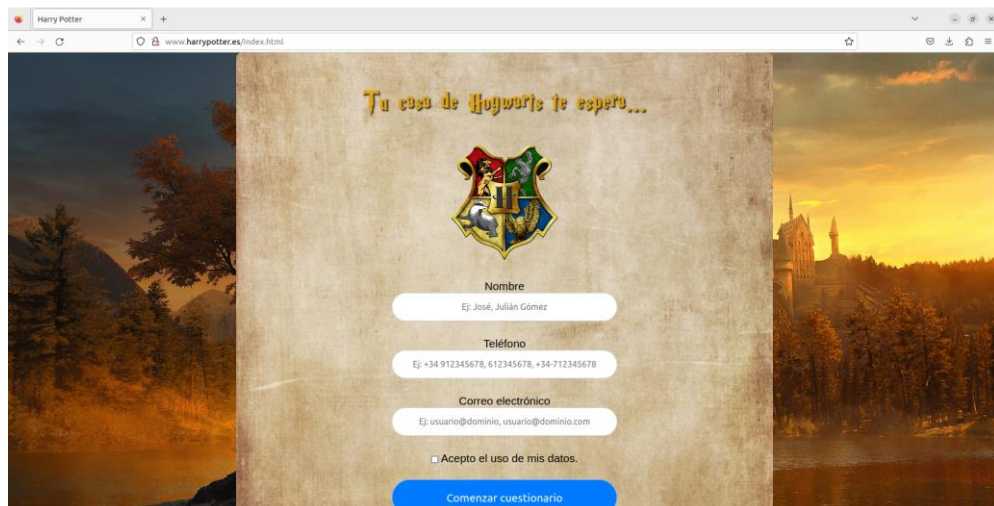
INTRODUCCIÓN

A. APLICACIÓN QUE SE VA A UTILIZAR

Para realizar la práctica se ha decidido utilizar una aplicación simple basada en HTML + CSS + JavaScript, la cual se encuentra alojada en el siguiente repositorio de GitHub: <https://github.com/BertoMP/harry-potter>



Esta aplicación una vez desplegada en un servidor web se ve de la siguiente forma:



B. REPARTO DE ROLES

En el grupo de trabajo habrá dos roles, un rol senior que será realizado por Alberto Martínez Pérez y un rol junior que será realizado por Erik Aylagas López.

C. REPARTO DE TAREAS

Para la realización de estas prácticas vamos a repartir las tareas de la siguiente forma:

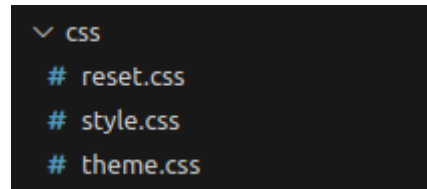
- **Tareas comunes:** Nos encargaremos en común del estilado de la página, es decir, los dos integrantes del equipo podremos realizar cambios sobre el directorio css/ del proyecto y sobre los archivos que en él se encuentran. Pudiendo realizarse por tanto cualquier modificación sobre los ficheros que se encuentran en el directorio, así como la creación de nuevos ficheros y de eliminación de los ya presentes desde el inicio.
- **Tareas que realizar por el desarrollador junior:** El desarrollador junior se va a encargar de los ficheros index.html y js/landing.js para realizar toda la interfaz y funciones que tengan que ver con la web inicial de la aplicación. De forma que como tareas podrá realizar cualquier modificación sobre el contenido de estos ficheros, así como el renombramiento del fichero landing.js. Así mismo podrá crear nuevos ficheros que sean necesarios para mejorar o modularizar el código y, además,
- **Tareas que realizar por el desarrollador senior:** El desarrollador senior se va a encargar de los ficheros pages/questionnaire.html, js/questionnaire.js, js/questions.js que son los ficheros que se encargan de toda la lógica y generación de la interfaz visual del cuestionario nuestra aplicación. Podrá realizar modificaciones sobre el contenido de los ficheros indicados, así como su renombramiento.

El grupo realizará una reunión semanal en la que se decidirán los cambios que se deben realizar durante la semana, así como poner en común los cambios que se han de realizar a continuación o un nuevo reparto de roles.

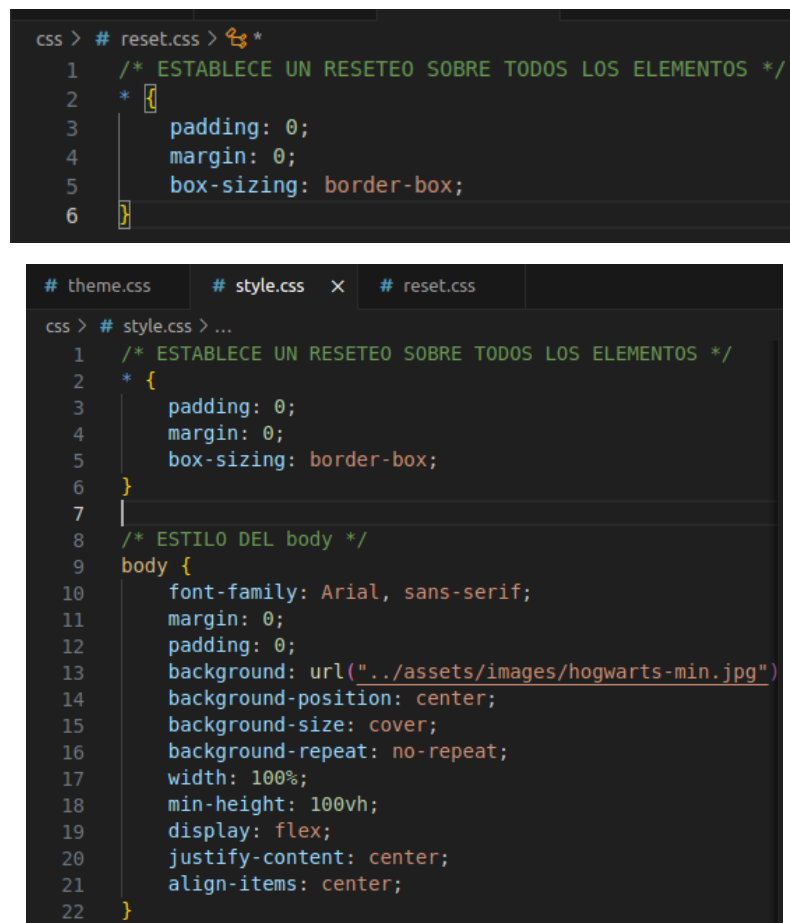
CAMBIOS REALIZADOS POR EL DESARROLLADOR SENIOR

A. CAMBIOS REALIZADOS EN EL DIRECTORIO COMÚN

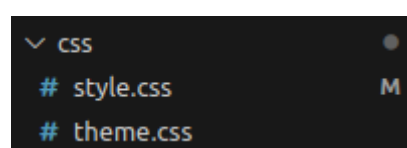
El estado actual del proyecto en la carpeta común es el siguiente:



He decidido incluir los datos del documento reset.css dentro del fichero style.css:



De esta forma este fichero reset.css ya no es necesario por lo que se puede eliminar:



CAMBIOS REALIZADOS POR EL DESARROLLADOR SENIOR

Una vez hecho esto voy a crear un commit con los cambios realizados, para ello lo primero que debemos ver es el estado del repositorio, para ello usamos el comando:

git status

```
alumno@albertoMartinez:~/harry-potter$ git status
En la rama main
Tu rama está actualizada con 'origin/main'.

Cambios no rastreados para el commit:
  (usa "git add/rm <archivo>..." para actualizar a lo que se le va a hacer commit)
  (usa "git restore <archivo>..." para descartar los cambios en el directorio de trabajo)
    borrados:      css/reset.css
    modificados:    css/style.css

sin cambios agregados al commit (usa "git add" y/o "git commit -a")
```

Esto nos informa de que un archivo ha sido borrado y que otro ha sido modificado y que estos cambios deben ser rastreados por git (pasar al área de stage) antes de ser confirmados, para ello utilizamos el comando

git add .

```
alumno@albertoMartinez:~/harry-potter$ git status
En la rama main
Tu rama está actualizada con 'origin/main'.

Cambios a ser confirmados:
  (usa "git restore --staged <archivo>..." para sacar del área de stage)
    borrados:      css/reset.css
    modificados:    css/style.css
```

Ahora debemos confirmar los cambios haciendo un commit, para ello utilizamos el comando:

git commit -m "mensaje informativo del commit"

```
alumno@albertoMartinez:~/harry-potter$ git commit -m "Añadido contenido de reset.css a style.css"
[main 73105cf] Añadido contenido de reset.css a style.css
2 files changed, 7 insertions(+), 6 deletions(-)
delete mode 100644 css/reset.css
```

Estos cambios se encuentran en el repositorio local, para subirlos a nuestro repositorio remoto debemos hacer un push de los cambios y para ello primero nos tendremos que autenticar como usuarios de GitHub a través de GitHub CLI, para ello usamos el comando:

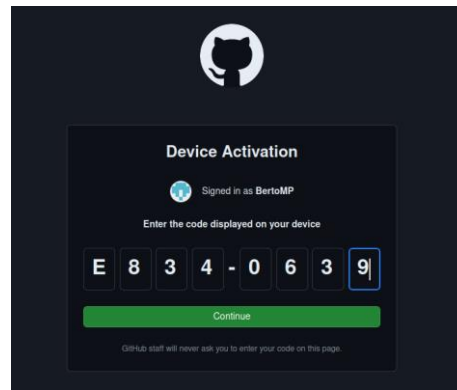
gh auth login

```
alumno@albertoMartinez:~/harry-potter$ gh auth login
? What account do you want to log into? GitHub.com
? What is your preferred protocol for Git operations? HTTPS
? Authenticate Git with your GitHub credentials? Yes
? How would you like to authenticate GitHub CLI? Login with a web browser

! First copy your one-time code: E834-0639
- Press Enter to open github.com in your browser...
```

Esto hará que se abra una ventana del navegador donde debemos incluir el código que se nos ha proporcionado:

CAMBIOS REALIZADOS POR EL DESARROLLADOR SENIOR

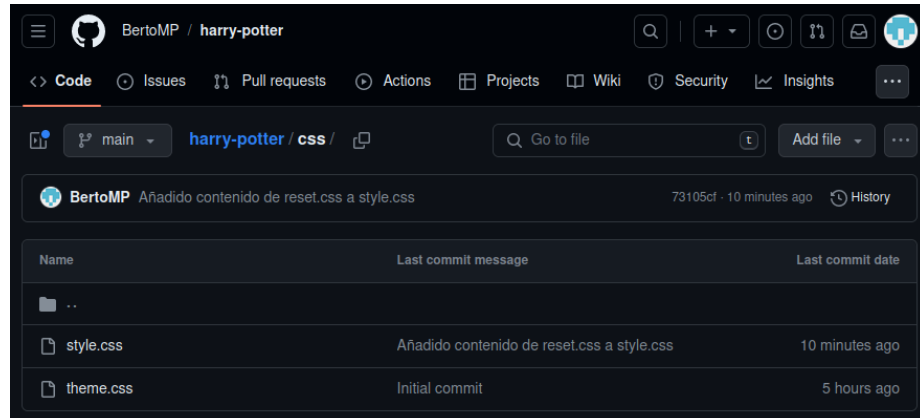


Una vez hecho esto volvemos a la terminal y realizamos el push de los cambios con el siguiente comando:

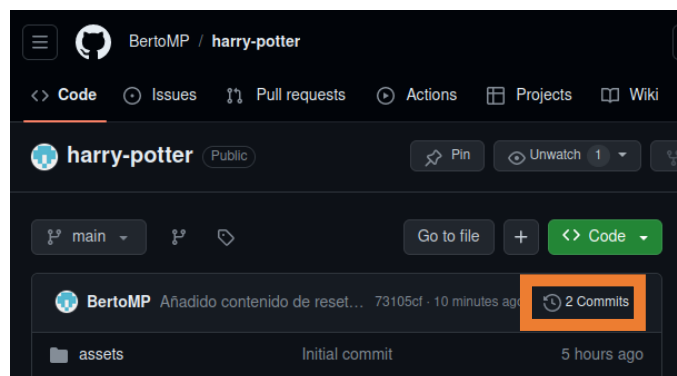
`git push`

```
alumno@albertoMartinez:~/harry-potter$ git push
Enumerando objetos: 7, listo.
Contando objetos: 100% (7/7), listo.
Compresión delta usando hasta 2 hilos
Comprimiendo objetos: 100% (4/4), listo.
Escribiendo objetos: 100% (4/4), 496 bytes | 496.00 KiB/s, listo.
Total 4 (delta 2), reusados 0 (delta 0), pack-reusados 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/BertoMP/harry-potter.git
9f53ef2..73105cf main -> main
```

De esta forma los cambios estarán presentes en nuestro repositorio remoto:

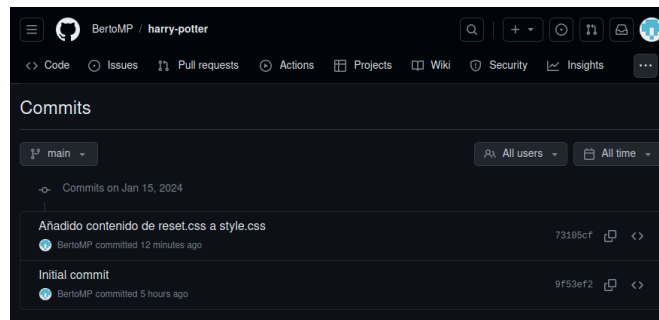


Además, en el menú principal del proyecto si hacemos clic en commits:



CAMBIOS REALIZADOS POR EL DESARROLLADOR SENIOR

Veremos el commit que acabamos de realizar:



Además, decido sacar el estilado propio de los botones del cuestionario a otro archivo .css independiente llamado buttons.css:

```
# style.css x
css > # style.css > div>button
134 /* ESTILO PARA LOS div>button */
135 div>button {
136     height: 3.7em;
137     margin-bottom: 0;
138     opacity: .8;
139     box-shadow: 5px 5px 5px rgba(0, 0, 0, 0.5);
140     transition: .5s all;
141     background: url("../assets/images/background-buttons.jpg");
142     background-position: center;
143     background-repeat: no-repeat;
144     background-size: cover;
145     color: black;
146     font-family: sans-serif;
147     font-weight: bold;
148     font-size: 1.4em;
149     display: flex;
150     justify-content: center;
151     align-items: center;
152 }
153
154 /* ESTILO PARA LOS div>button AL PASAR EL CURSOR SOBRE ELLOS */
155 div>button:hover {
156     opacity: 1;
157     filter: brightness(110%);
158 }
```

```
# style.css # buttons.css x
css > # buttons.css > div>button:hover
1 /* ESTILO PARA LOS div>button */
2 div>button {
3     height: 3.7em;
4     margin-bottom: 0;
5     opacity: .8;
6     box-shadow: 5px 5px 5px rgba(0, 0, 0, 0.5);
7     transition: .5s all;
8     background: url("../assets/images/background-buttons.jpg");
9     background-position: center;
10    background-repeat: no-repeat;
11    background-size: cover;
12    color: black;
13    font-family: sans-serif;
14    font-weight: bold;
15    font-size: 1.4em;
16    display: flex;
17    justify-content: center;
18    align-items: center;
19 }
20
21 /* ESTILO PARA LOS div>button AL PASAR EL CURSOR SOBRE ELLOS */
22 div>button:hover {
23     opacity: 1;
24     filter: brightness(110%);
25 }
```


CAMBIOS REALIZADOS POR EL DESARROLLADOR SENIOR

También cambio el tipo de letra de estos botones para que en lugar de ser una sans-serif sea la fuente temática que se creó en el archivo theme.css.

```
# style.css M # buttons.css U X
css > # buttons.css > div>button
1  /* ESTILO PARA LOS div>button */
2  div>button {
3      height: 3.7em;
4      margin-bottom: 0;
5      opacity: .8;
6      box-shadow: 5px 5px 5px rgba(0, 0, 0, 0.5);
7      transition: .5s all;
8      background: url("../assets/images/background-buttons.jpg");
9      background-position: center;
10     background-repeat: no-repeat;
11     background-size: cover;
12     color: black;
13     font-family: var(--fuente-tematica);
14     font-weight: bold;
15     font-size: 1.4em;
16     display: flex;
17     justify-content: center;
18     align-items: center;
19 }
20
21 /* ESTILO PARA LOS div>button AL PASAR EL CURSOR SOBRE ELLOS */
22 div>button:hover {
23     opacity: 1;
24     filter: brightness(110%);
25 }
```

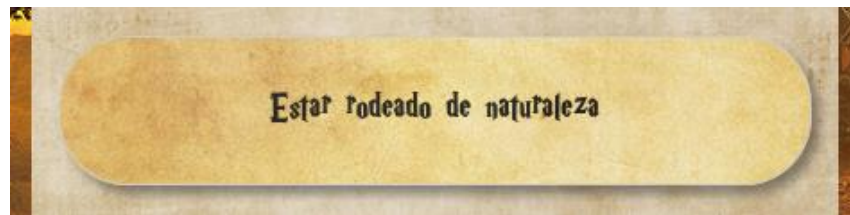
Añado esta nuevo CSS al archivo questionnaire.html:

```
# style.css M # buttons.css U questionnaire.html X
pages > questionnaire.html > html > head > link
1  <!DOCTYPE html>
2  <!-- Declaración del tipo de documento especificando la versión de HTML e idioma -->
3  <html lang="es">
4  <head>
5      <!-- Sección head que contiene metadatos sobre el documento -->
6      <meta charset="UTF-8"> <!-- Especifica la codificación de caracteres -->
7      <title>Cuestionario</title> <!-- Establece el título de la página -->
8      <!-- Enlaces a hojas de estilo externas para la presentación -->
9      <link rel="stylesheet" href="../css/style.css">
10     <link rel="stylesheet" href="../css/theme.css">
11     <link rel="stylesheet" href="../css/buttons.css">
12 </head>
13 <body>
14 <!-- Contenido principal del cuerpo de la página -->
15 <footer>
16     <!-- Sección de pie de página que incluye el script del cuestionario -->
17     <script src="../js/questionnaire.js" type="module"></script>
18 </footer>
19 </body>
20 </html>
```

De forma que los botones pasan de tener el siguiente aspecto:



A este otro:



Vamos a confirmar estos cambios y a subirlos al repositorio remoto, para ello si hacemos un git status veremos lo siguiente:

```
alumno@albertoMartinez:~/harry-potter$ git status
En la rama main
Tu rama está actualizada con 'origin/main'.

Cambios no rastreados para el commit:
  (usa "git add <archivo>..." para actualizar lo que será confirmado)
  (usa "git restore <archivo>..." para descartar los cambios en el directorio de trabajo)
    modificados:   css/style.css
    modificados:   pages/questionnaire.html

Archivos sin seguimiento:
  (usa "git add <archivo>..." para incluirlo a lo que será confirmado)
    css/buttons.css

sin cambios agregados al commit (usa "git add" y/o "git commit -a")
```

Tenemos dos archivos modificados (style.css y questionnaire.html) y un nuevo archivo creado (buttons.css), para hacer que pasen al área de stage hacemos un git add con la opción . para afectar a todos ellos:

```
alumno@albertoMartinez:~/harry-potter$ git add .
alumno@albertoMartinez:~/harry-potter$ git status
En la rama main
Tu rama está actualizada con 'origin/main'.

Cambios a ser confirmados:
  (usa "git restore --staged <archivo>..." para sacar del área de stage)
    nuevos archivos: css/buttons.css
    modificados:     css/style.css
    modificados:     pages/questionnaire.html
```

Por último, realizamos el commit con el comando git commit:

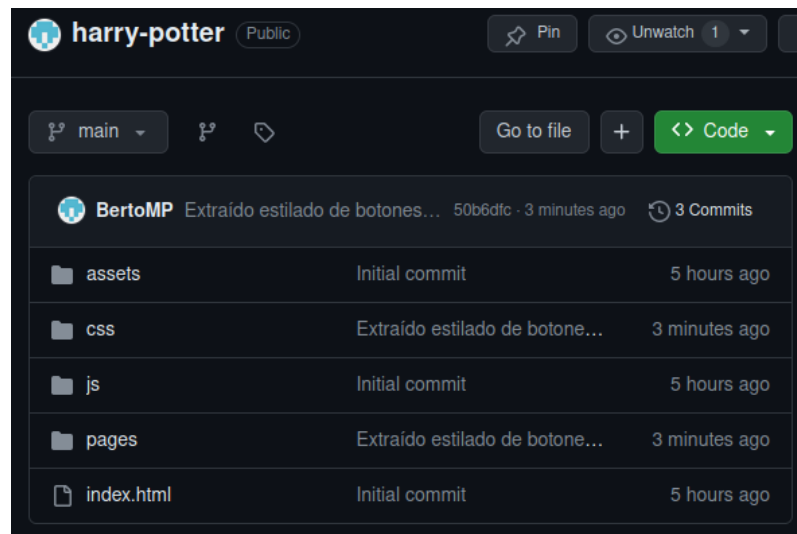
```
alumno@albertoMartinez:~/harry-potter$ git commit -m "Extraído estilado de botones a archivo css independiente."
[main 50b6dfc] Extraído estilado de botones a archivo css independiente.
 3 files changed, 26 insertions(+), 27 deletions(-)
 create mode 100644 css/buttons.css
```

Y realizamos la subida de los ficheros al repositorio remoto con un git push:

```
alumno@albertoMartinez:~/harry-potter$ git push
Enumerando objetos: 12, listo.
Contando objetos: 100% (12/12), listo.
Compresión delta usando hasta 2 hilos
Comprimiendo objetos: 100% (7/7), listo.
Escribiendo objetos: 100% (7/7), 951 bytes | 951.00 KiB/s, listo.
Total 7 (delta 4), reusados 0 (delta 0), pack-reusados 0
remote: Resolving deltas: 100% (4/4), completed with 4 local objects.
To https://github.com/BertoMP/harry-potter.git
  73105cf..50b6dfc  main -> main
```

CAMBIOS REALIZADOS POR EL DESARROLLADOR SENIOR

Si ahora accedemos a la web de nuestro repositorio remoto veremos los cambios realizados:



B. CAMBIOS REALIZADOS POR EL DESARROLLADOR SENIOR EN SUS FICHEROS

Sobre los archivos exclusivos de mi rol he decidido realizar un cambio de modificación en el archivo questions.js que se trata de un archivo tipo JSON que guarda las preguntas del cuestionario, para ello he añadido una nueva pregunta:

```
JS questions.js M x
js > JS questions.js > [e] questions > answers > points
267     },
268   ],
269 },
270 {
271   question: '¿Qué color te gusta más de los siguientes?',
272   answers: [
273     {
274       text: 'Verde',
275       house: 'Slytherin',
276       points: 10
277     },
278     {
279       text: 'Rojo',
280       house: 'Gryffindor',
281       points: 10
282     },
283     {
284       text: 'Azul',
285       house: 'Ravenclaw',
286       points: 10
287     },
288     {
289       text: 'Amarillo',
290       house: 'Hufflepuff',
291       points: 10
292     }
293   ]
294 }
295 ];
296
```

CAMBIOS REALIZADOS POR EL DESARROLLADOR SENIOR

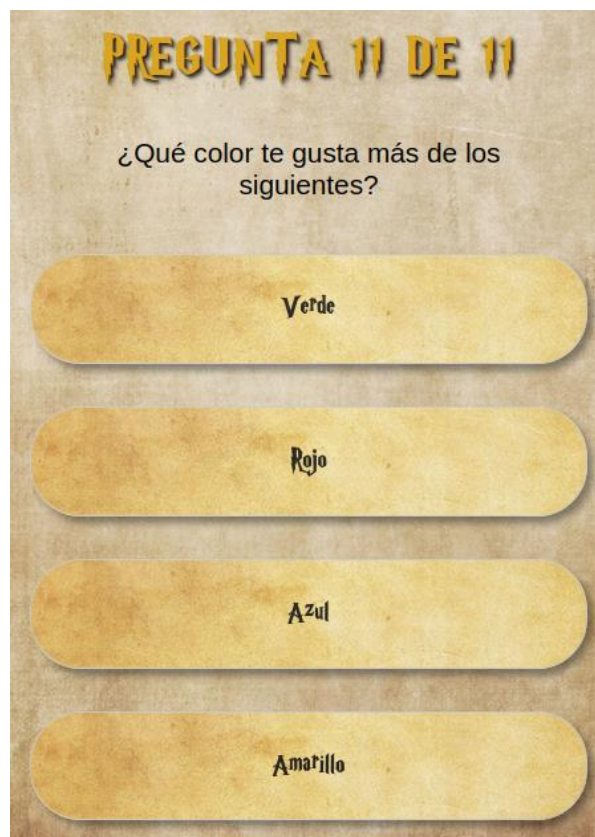
De esta forma el número máximo de preguntas pasa de 10:



A un total de 11 preguntas:



Siendo la nueva pregunta la siguiente:



Para confirmar estos cambios lo primero será ver el estado del repositorio:

```
alumno@albertoMartinez:~/harry-potter$ git status
En la rama main
Tu rama está actualizada con 'origin/main'.

Cambios no rastreados para el commit:
  (usa "git add <archivo>..." para actualizar lo que será confirmado)
  (usa "git restore <archivo>..." para descartar los cambios en el directorio de trabajo)
modificados:    js/questions.js

sin cambios agregados al commit (usa "git add" y/o "git commit -a")
```

Como vemos en la imagen hemos modificado un fichero, concretamente el fichero questions.js.

CAMBIOS REALIZADOS POR EL DESARROLLADOR SENIOR

A continuación, debemos hacer que estos cambios pasen a estar en el área de stage con un git add:

```
alumno@albertoMartinez:~/harry-potter$ git add .
alumno@albertoMartinez:~/harry-potter$ git status
En la rama main
Tu rama está actualizada con 'origin/main'.

Cambios a ser confirmados:
(usa "git restore --staged <archivo>..." para sacar del área de stage)
    modificados:    js/questions.js
```

Y pasamos a confirmar los cambios con un git commit:

```
alumno@albertoMartinez:~/harry-potter$ git commit -m "Añadida una nueva pregunta al
cuestionario"[main c5696b6] Añadida una nueva pregunta al cuestionario
1 file changed, 25 insertions(+)
```

Por último, empujamos los cambios a nuestro repositorio remoto:

```
alumno@albertoMartinez:~/harry-potter$ git push
Enumerando objetos: 7, listo.
Contando objetos: 100% (7/7), listo.
Compresión delta usando hasta 2 hilos
Comprimiendo objetos: 100% (4/4), listo.
Escribiendo objetos: 100% (4/4), 537 bytes | 537.00 KiB/s, listo.
Total 4 (delta 3), reusados 0 (delta 0), pack-reusados 0
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
To https://github.com/BertoMP/harry-potter.git
  50b6dfc..c5696b6  main -> main
```

C. DESCARGAR CAMBIOS REALIZADOS POR NUESTRO COMPAÑERO

Para descargar cambios podemos o bien utilizar el comando:

git fetch

```
alumno@albertoMartinez:~/harry-potter$ git fetch
remote: Enumerating objects: 66, done.
remote: Counting objects: 100% (29/29), done.
remote: Compressing objects: 100% (21/21), done.
remote: Total 66 (delta 10), reused 22 (delta 8), pack-reused 37
Desempaquetando objetos: 100% (66/66), 82.68 MiB | 8.85 MiB/s, listo.
Desde https://github.com/BertoMP/harry-potter
* [nueva rama]      main      -> origin/main
```

Este comando fetch nos sirve para descargar los metadatos del repositorio remoto a nuestro repositorio local, pero sin que haya un cambio físico de los archivos locales, es decir, estos no se ven modificados simplemente descargamos los metadatos para informar a nuestro repositorio local si existen cambios.

En el caso de que queramos que haya cambios físicos en los archivos, debemos utilizar el comando:

git pull

```
alumno@albertoMartinez:~/harry-potter$ git pull
```

CAMBIOS REALIZADOS POR EL DESARROLLADOR SENIOR

Este comando sí realiza una modificación sobre los archivos físicos aplicando los cambios que han sido descargados mediante git fetch (en el caso de que no hagamos un fetch previo al pull, el propio pull realizará un fetch de forma automática para actualizar los metadatos del repositorio).

CAMBIOS REALIZADOS POR EL DESARROLLADOR JUNIOR

A. CAMBIOS EN LOS ARCHIVOS PROPIOS

Hago un clone del repositorio de Harry.

```
alumno@erikaylagas:~$ git clone https://github.com/BertoMP/harry-potter.git
Clonando en 'harry-potter'...
remote: Enumerating objects: 52, done.
remote: Counting objects: 100% (15/15), done.
remote: Compressing objects: 100% (12/12), done.
remote: Total 52 (delta 3), reused 10 (delta 3), pack-reused 37
Recibiendo objetos: 100% (52/52), 82.67 MiB | 977.00 KiB/s, listo.
Resolviendo deltas: 100% (6/6), listo.
alumno@erikaylagas:~$
```

Realizo cambios del index.html.

Este es el index.html sin modificar:

Este es el código del index.html sin modificar.

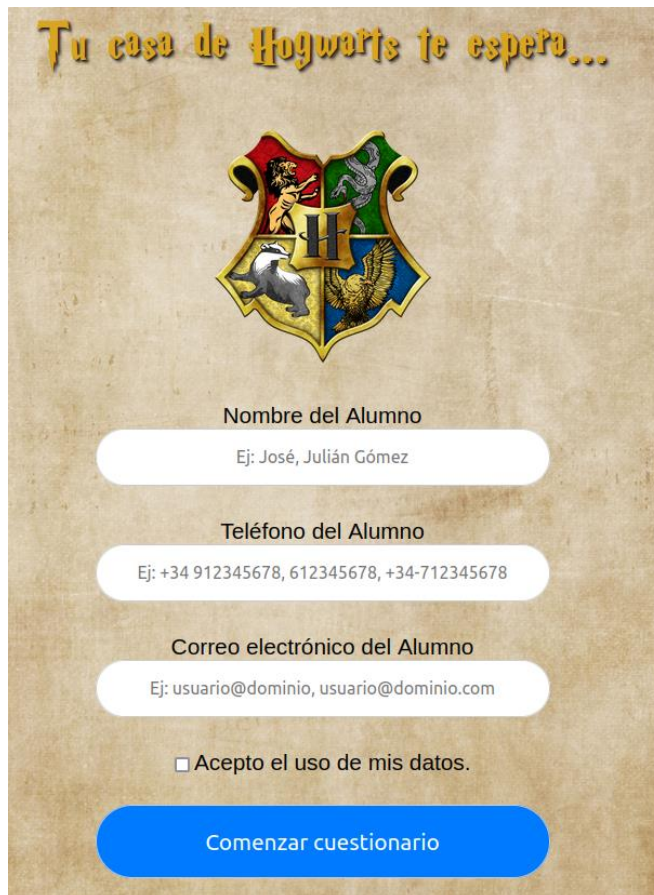
```
<section>
  <!-- Sección para el formulario de Hogwarts -->
  <section id="form-hogwarts">
    <h2>Tu casa de Hogwarts te espera...</h2> <!-- Encabezado para el formulario de Hogwarts -->
     <!-- Imagen del logo de Hogwarts -->
    <!-- Formulario para recopilar la información del usuario -->
    <form action="pages/questionnaire.html" method="get">
      <!-- Campo de entrada para el nombre del usuario -->
      <article>
        <label for="name">Nombre </label>
        <input type="text"
              name="name"
              id="name"
              placeholder="Ej: José, Julián Gómez">
      </article>
      <!-- Campo de entrada para el número de teléfono del usuario -->
      <article>
        <label for="phone">Teléfono</label>
        <input type="text"
              name="phone"
              id="phone"
              placeholder="Ej: +34 912345678, 612345678, +34-712345678">
      </article>
      <!-- Campo de entrada para la dirección de correo electrónico del usuario -->
      <article>
        <label for="email">Correo electrónico</label>
        <input type="text"
              name="email"
              id="email"
              placeholder="Ej: usuario@dominio, usuario@dominio.com">
      </article>
      <!-- Casilla de verificación para aceptar el uso de los datos -->
      <article>
        <input type="checkbox" name="accept-terms" id="accept-terms">
        <label for="accept-terms">Acepto el uso de mis datos.</label>
      </article>
      <!-- Botón para enviar el formulario y comenzar el cuestionario -->
      <article>
```

CAMBIOS REALIZADOS POR EL DESARROLLADOR JUNIOR


Modificados el index.html cambiando el texto de los label del formulario.

```
<section>
  <!-- Sección para el formulario de Hogwarts -->
  <section id="form-hogwarts">
    <h2>Tu casa de Hogwarts te espera...</h2> <!-- Encabezado para el formulario de Hogwarts -->
     <!-- Imagen del logo de Hogwarts -->
    <!-- Formulario para recopilar la información del usuario -->
    <form action="pages/questionnaire.html" method="get">
      <!-- Campo de entrada para el nombre del usuario -->
      <article>
        <label for="name">Nombre del Alumno</label>
        <input type="text"
          name="name"
          id="name"
          placeholder="Ej: José, Julián Gómez">
      </article>
      <!-- Campo de entrada para el número de teléfono del usuario -->
      <article>
        <label for="phone">Teléfono del Alumno</label>
        <input type="text"
          name="phone"
          id="phone"
          placeholder="Ej: +34 912345678, 612345678, +34-712345678">
      </article>
      <!-- Campo de entrada para la dirección de correo electrónico del usuario -->
      <article>
        <label for="email">Correo electrónico del Alumno</label>
        <input type="text"
          name="email"
          id="email"
          placeholder="Ej: usuario@dominio, usuario@dominio.com">
      </article>
      <!-- Casilla de verificación para aceptar el uso de los datos -->
      <article>
        <input type="checkbox" name="accept-terms" id="accept-terms">
        <label for="accept-terms">Acepto el uso de mis datos.</label>
      </article>
      <!-- Botón para enviar el formulario y comenzar el cuestionario -->
      <article>
        <button type="submit" id="btn-start">Comenzar cuestionario</button>
      </article>
    </form>
  </section>
</section>
```

Podemos ver los cambios como quedan.



Tu casa de Hogwarts te espera...



Nombre del Alumno

Ej: José, Julián Gómez

Teléfono del Alumno

Ej: +34 912345678, 612345678, +34-712345678

Correo electrónico del Alumno

Ej: usuario@dominio, usuario@dominio.com

☐ Acepto el uso de mis datos.

Comenzar cuestionario

CAMBIOS REALIZADOS POR EL DESARROLLADOR JUNIOR

Hago un git status para ver el estado del repositorio y nos muestra los archivos modificados.

```
alumno@erikaylagas:~/harry-potter$ git status
En la rama main
Tu rama está actualizada con 'origin/main'.

Cambios no rastreados para el commit:
  (usa "git add <archivo>..." para actualizar lo que será confirmado)
  (usa "git restore <archivo>..." para descartar los cambios en el directorio de trabajo)
    modificados:    index.html

sin cambios agregados al commit (usa "git add" y/o "git commit -a")
```

Realizo un git add . para mover los cambios y realizo un status para ver el estado.

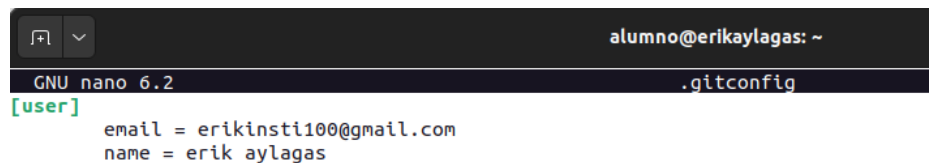
```
alumno@erikaylagas:~/harry-potter$ git add .
alumno@erikaylagas:~/harry-potter$ git status
En la rama main
Tu rama está actualizada con 'origin/main'.

Cambios a ser confirmados:
  (usa "git restore --staged <archivo>..." para sacar del área de stage)
    modificados:    index.html
```

Utilizo git config --global user.email y user.name para guardar mi correo y mi nombre

```
alumno@erikaylagas:~/harry-potter$ git config --global user.email "erikinsti100@gmail.com"
alumno@erikaylagas:~/harry-potter$ git config --global user.name "erik aylagas"
```

Podemos ver que se guarda en .gitconfig .



```
alumno@erikaylagas: ~
GNU nano 6.2 .gitconfig
[user]
    email = erikinsti100@gmail.com
    name = erik aylagas
```

Confirmamos los cambios usando git commit -m y pongo un mensaje descriptivo.

```
alumno@erikaylagas:~/harry-potter$ git commit -m "index modificado"
[main e750bd0] index modificado
 1 file changed, 3 insertions(+), 3 deletions(-)
alumno@erikaylagas:~/harry-potter$
```

Para poder subir los cambios que tenemos en el repositorio local tenemos que realizar un push pero primero tenemos que autenticarnos en GitHub.

Usamos gh auth login que nos dirá que opciones queremos usar, en mi caso uso las siguientes.

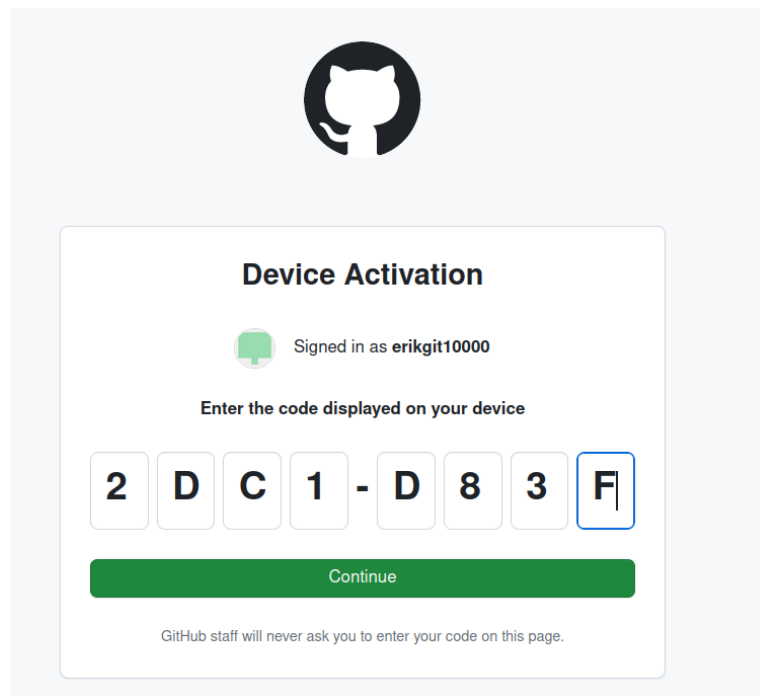
```
alumno@erikaylagas:~/harry-potter$ gh auth login
? What account do you want to log into? GitHub.com
? You're already logged into github.com. Do you want to re-authenticate? Yes
? What is your preferred protocol for Git operations? HTTPS
? Authenticate Git with your GitHub credentials? Yes
? How would you like to authenticate GitHub CLI? Login with a web browser

! First copy your one-time code: 96B5-3E6C
- Press Enter to open github.com in your browser...
Gtk-Message: 12:39:52.689: Not loading module "atk-bridge": The functionality is provided by GTK natively. Please try to not load it.
o Authentication complete. Press Enter to continue...

- gh config set -h github.com git_protocol https
✓ Configured git protocol
✓ Logged in as erikgit10000
```

CAMBIOS REALIZADOS POR EL DESARROLLADOR JUNIOR

Se nos abre una ventana del navegador y nos pide un código de confirmación

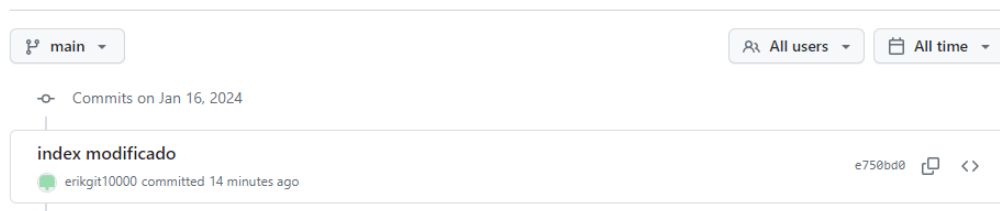


Una vez que hayas realizado los cambios, puedes ejecutar el comando git push para enviar las modificaciones desde tu repositorio local al repositorio remoto.

```
alumno@erikaylagas:~/harry-potter$ git push
Enumerando objetos: 5, listo.
Contando objetos: 100% (5/5), listo.
Compresión delta usando hasta 2 hilos
Comprimiendo objetos: 100% (3/3), listo.
Escribiendo objetos: 100% (3/3), 325 bytes | 325.00 KiB/s, listo.
Total 3 (delta 2), reusados 0 (delta 0), pack-reusados 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/BertoMP/harry-potter.git
c5696b6..e750bd0  main -> main
```

Dentro de Git podemos ver el commit que acabamos de hacer.

Commits



B. CAMBIOS REALIZADOS SOBRE LOS ARCHIVOS COMUNES

Otra modificación que realizo es que me parece mejor que estén los botones dentro del mismo css ya que separados podemos tener algún error.

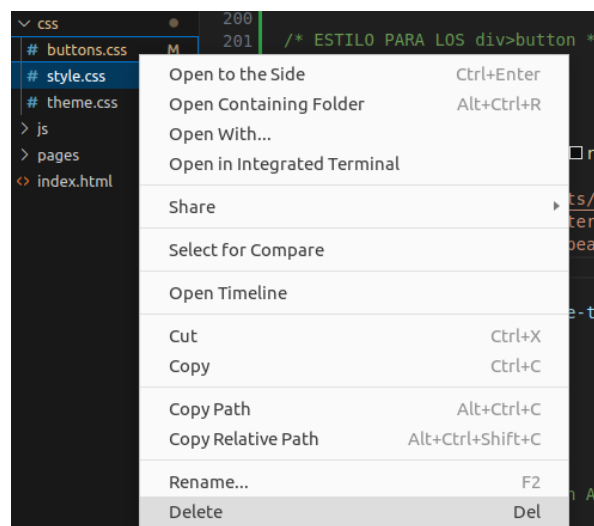
CAMBIOS REALIZADOS POR EL DESARROLLADOR JUNIOR

```
# buttons.css x
css > # buttons.css > div>buttonhover
1  /* ESTILO PARA LOS div>button */
2  div>button {
3      height: 3.7em;
4      margin-bottom: 0;
5      opacity: .8;
6      box-shadow: 5px 5px 5px rgba(0, 0, 0, 0.5);
7      transition: .5s all;
8      background: url("../assets/images/background-buttons.jpg");
9      background-position: center;
10     background-repeat: no-repeat;
11     background-size: cover;
12     color: black;
13     font-family: var(--fuente-tematica);
14     font-weight: bold;
15     font-size: 1.4em;
16     display: flex;
17     justify-content: center;
18     align-items: center;
19 }
20
21 /* ESTILO PARA LOS div>button AL PASAR EL CURSOR SOBRE ELLOS */
22 div>button:hover {
23     opacity: 1;
24     filter: brightness(110%);
25 }
```

Lo a ado al final del style.css

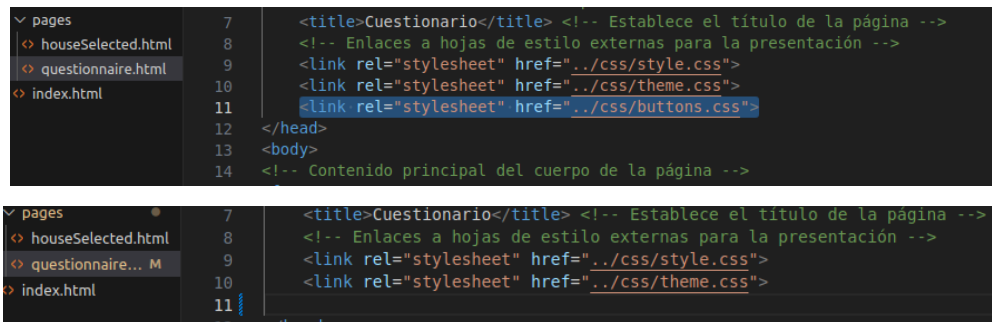
```
# buttons.css M # style.css M x
css > # style.css > div>button
199
200
201 /* ESTILO PARA LOS div>button */
202 div>button {
203     height: 3.7em;
204     margin-bottom: 0;
205     opacity: .8;
206     box-shadow: 5px 5px 5px rgba(0, 0, 0, 0.5);
207     transition: .5s all;
208     background: url("../assets/images/background-buttons.jpg");
209     background-position: center;
210     background-repeat: no-repeat;
211     background-size: cover;
212     color: black;
213     font-family: var(--fuente-tematica);
214     font-weight: bold;
215     font-size: 1.4em;
216     display: flex;
217     justify-content: center;
218     align-items: center;
219 }
220
221 /* ESTILO PARA LOS div>button AL PASAR EL CURSOR SOBRE ELLOS */
222 div>button:hover {
223     opacity: 1;
224     filter: brightness(110%);
225 }
```

Elimino el archivo buttons.css al dejarlo vac o.



CAMBIOS REALIZADOS POR EL DESARROLLADOR JUNIOR

Elimino también el enlace al no tener ya el archivo.css.



Hago un git status para ver el estado del repositorio y podemos ver los modificados y borrados

```
alumno@erikaylagas:~/harry-potter$ git status
En la rama main
Tu rama está actualizada con 'origin/main'.

Cambios no rastreados para el commit:
(usa "git add/rm <archivo>..." para actualizar a lo que se le va a hacer commit)
(usa "git restore <archivo>..." para descartar los cambios en el directorio de trabajo)
borrados:      css/buttons.css
modificados:   css/style.css
modificados:   pages/questionnaire.html

sin cambios agregados al commit (usa "git add" y/o "git commit -a")
```

Realizo un git add . y para subir los cambios y realizo un git status para comprobarlo.

```
alumno@erikaylagas:~/harry-potter$ git add .
alumno@erikaylagas:~/harry-potter$ git status
En la rama main
Tu rama está actualizada con 'origin/main'.

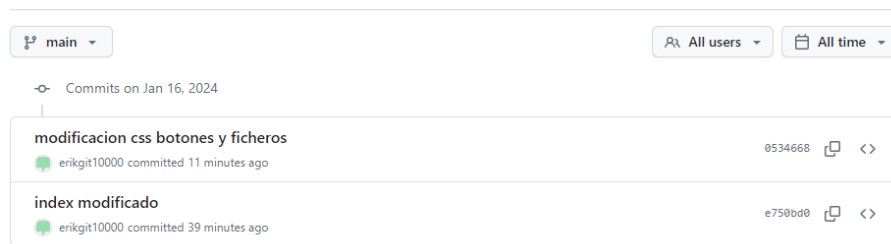
Cambios a ser confirmados:
(usa "git restore --staged <archivo>..." para sacar del área de stage)
borrados:      css/buttons.css
modificados:   css/style.css
modificados:   pages/questionnaire.html

alumno@erikaylagas:~/harry-potter$ git commit -m "modificacion css botones y ficheros"
[main 0534668] modificacion css botones y ficheros
3 files changed, 28 insertions(+), 26 deletions(-)
delete mode 100644 css/buttons.css
```

Realizo un push para subir los ficheros al repositorio remoto.

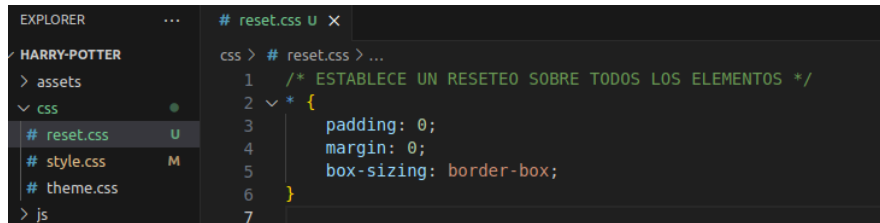
```
alumno@erikaylagas:~/harry-potter$ git push
Enumerando objetos: 11, listo.
Contando objetos: 100% (11/11), listo.
Compresión delta usando hasta 2 hilos
Comprimiendo objetos: 100% (6/6), listo.
Escribiendo objetos: 100% (6/6), 731 bytes | 365.00 KiB/s, listo.
Total 6 (delta 4), reusados 0 (delta 0), pack-reusados 0
remote: Resolving deltas: 100% (4/4), completed with 4 local objects.
To https://github.com/BertoMP/harry-potter.git
e750bd0..0534668 main -> main
```

Commits



CAMBIOS REALIZADOS POR EL DESARROLLADOR JUNIOR

Creo un fichero llamado reset y meto el código que he sacado de style porque me parece mejor tener dividido.



The screenshot shows the VS Code interface. On the left, the Explorer sidebar shows a project named 'HARRY-POTTER' with folders 'assets', 'css', and 'js'. The 'css' folder is expanded, showing 'reset.css' (selected), 'style.css', and 'theme.css'. The Editor pane on the right shows the content of 'reset.css':

```
# reset.css
1  /* ESTABLECE UN RESETEO SOBRE TODOS LOS ELEMENTOS */
2  * {
3      padding: 0;
4      margin: 0;
5      box-sizing: border-box;
6  }
```

Realizo un git status para comprobar el estado.

```
alumno@erikaylagas:~/harry-potter$ git status
En la rama main
Tu rama está actualizada con 'origin/main'.

Cambios no rastreados para el commit:
  (usa "git add <archivo>..." para actualizar lo que será confirmado)
  (usa "git restore <archivo>..." para descartar los cambios en el directorio de trabajo)
    modificados:   css/style.css

Archivos sin seguimiento:
  (usa "git add <archivo>..." para incluirlo a lo que será confirmado)
    css/reset.css

sin cambios agregados al commit (usa "git add" y/o "git commit -a")
```










Hago un add y luego un commit con un comentario para subirlo al stage.

```
alumno@erikaylagas:~/harry-potter$ git add .
alumno@erikaylagas:~/harry-potter$ git commit -m "Creo fichero reset.css para todos los elementos"
[main 083b7cb] Creo fichero reset.css para todos los elementos
2 files changed, 6 insertions(+), 7 deletions(-)
create mode 100644 css/reset.css
```

Realizo un push para terminar de subir los cambios al repositorio remoto.

```
alumno@erikaylagas:~/harry-potter$ git push
Enumerando objetos: 8, listo.
Contando objetos: 100% (8/8), listo.
Compresión delta usando hasta 2 hilos
Comprimiendo objetos: 100% (5/5), listo.
Escribiendo objetos: 100% (5/5), 527 bytes | 527.00 KiB/s, listo.
Total 5 (delta 2), reusados 0 (delta 0), pack-reusados 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/BertoMP/harry-potter.git
   0534668..083b7cb  main -> main
```

Podemos ver los commits anteriores.

Commits		
 main	 All users	 All time
Commits on Jan 16, 2024		
<hr/>		
Creo fichero reset.css para todos los elementos		
 erikgit10000 committed 1 minute ago	083b7cb	 <>
<hr/>		
modificacion css botones y ficheros		
 erikgit10000 committed 19 minutes ago	0534668	 <>
<hr/>		
index modificado		
 erikgit10000 committed 47 minutes ago	e758bd0	 <>

CONFIGURACIÓN DE UN REPOSITORIO PÚBLICO

A. SINCRONIZACIÓN ENTRE REPOSITORIO LOCAL Y REMOTO

Dentro de nuestro directorio lo primero que tenemos que hacer es inicializar nuestro repositorio local a través del comando:

`git init`

```
alumno@albertoMartinez:~/harry-potter$ git init
ayuda: Usando 'master' como el nombre de la rama inicial. Este nombre de rama pr
edeterminado
ayuda: está sujeto a cambios. Para configurar el nombre de la rama inicial para
usar en todos
ayuda: de sus nuevos repositorios, reprimiendo esta advertencia, llama a:
ayuda:
ayuda: git config --global init.defaultBranch <nombre>
ayuda:
ayuda: Los nombres comúnmente elegidos en lugar de 'master' son 'main', 'trunk'
y
ayuda: 'development'. Se puede cambiar el nombre de la rama recién creada median
te este comando:
ayuda:
ayuda: git branch -m <nombre>
Iniciado repositorio Git vacío en /home/alumno/harry-potter/.git/
```

Esto nos habrá creado una carpeta oculta llamada `.git` que almacenará el repositorio local:

```
alumno@albertoMartinez:~/harry-potter$ ls -la
total 32
drwxrwxr-x 7 alumno alumno 4096 ene 15 17:58 .
drwxr-x--- 18 alumno alumno 4096 ene 15 17:57 ..
drwxrwxr-x 5 alumno alumno 4096 ene 15 12:07 assets
drwxrwxr-x 2 alumno alumno 4096 ene 15 12:07 css
drwxrwxr-x 7 alumno alumno 4096 ene 15 17:58 .git
-rw-rw-r-- 1 alumno alumno 3047 ene 15 12:07 index.html
drwxrwxr-x 2 alumno alumno 4096 ene 15 12:07 js
drwxrwxr-x 2 alumno alumno 4096 ene 15 12:07 pages
```

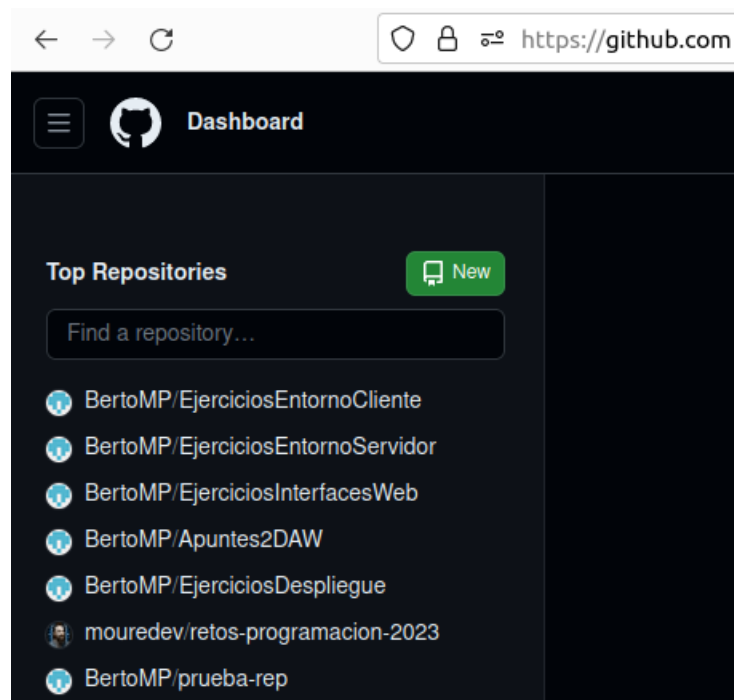
Realizamos un commit inicial, para ello primero tendremos que añadir los ficheros al área de stage, para a continuación realizar el commit:

```
alumno@albertoMartinez:~/harry-potter$ git add .
```

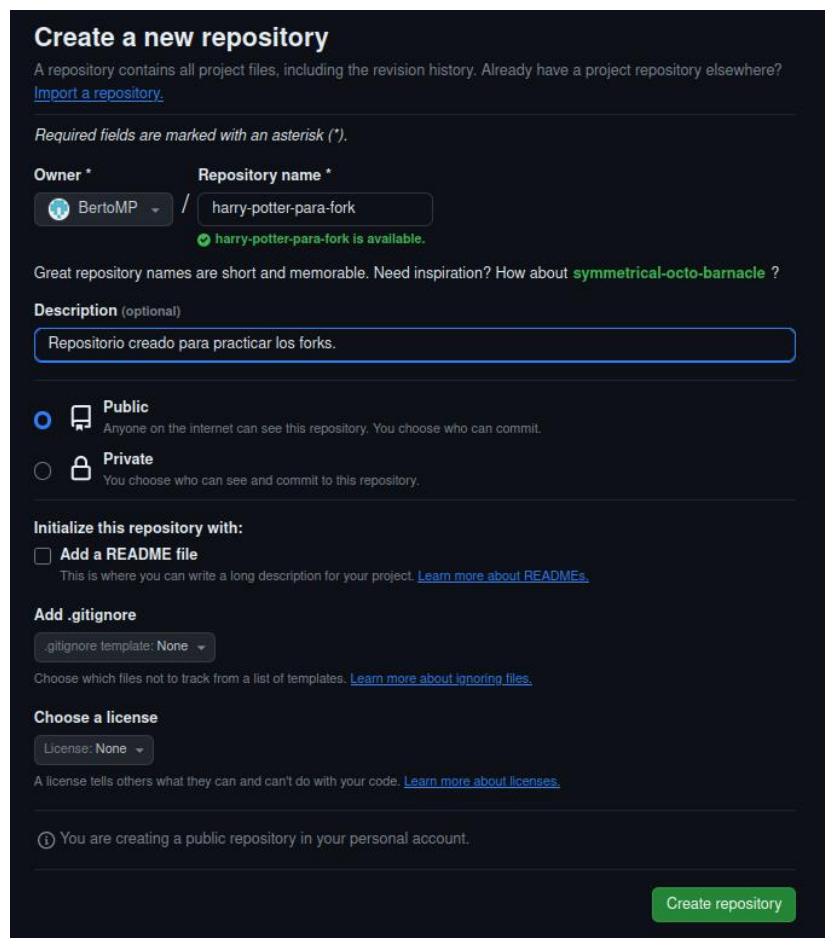
```
alumno@albertoMartinez:~/harry-potter$ git commit -m "Initial commit"
[master (commit-raiz) b796d9f] Initial commit
28 files changed, 1519 insertions(+)
create mode 100644 assets/fonts/harryPotter.ttf
create mode 100644 assets/images/background-buttons.jpg
create mode 100644 assets/images/background-min.jpg
create mode 100644 assets/images/gryffindor-min.png
create mode 100644 assets/images/gryffindor-room-min.jpg
create mode 100644 assets/images/hogwarts-min.jpg
create mode 100644 assets/images/hufflepuff-min.png
create mode 100644 assets/images/hufflepuff-room-min.png
create mode 100644 assets/images/principal-logo-min.png
create mode 100644 assets/images/ravenclaw-min.png
create mode 100644 assets/images/ravenclaw-room-min.webp
create mode 100644 assets/images/slytherin-min.png
```

CONFIGURACIÓN DE UN REPOSITORIO PÚBLICO

A continuación, nos dirigimos a GitHub y vamos a crear un nuevo repositorio remoto, para ello debemos hacer clic en el botón “New”.



En la nueva ventana que se despliega:

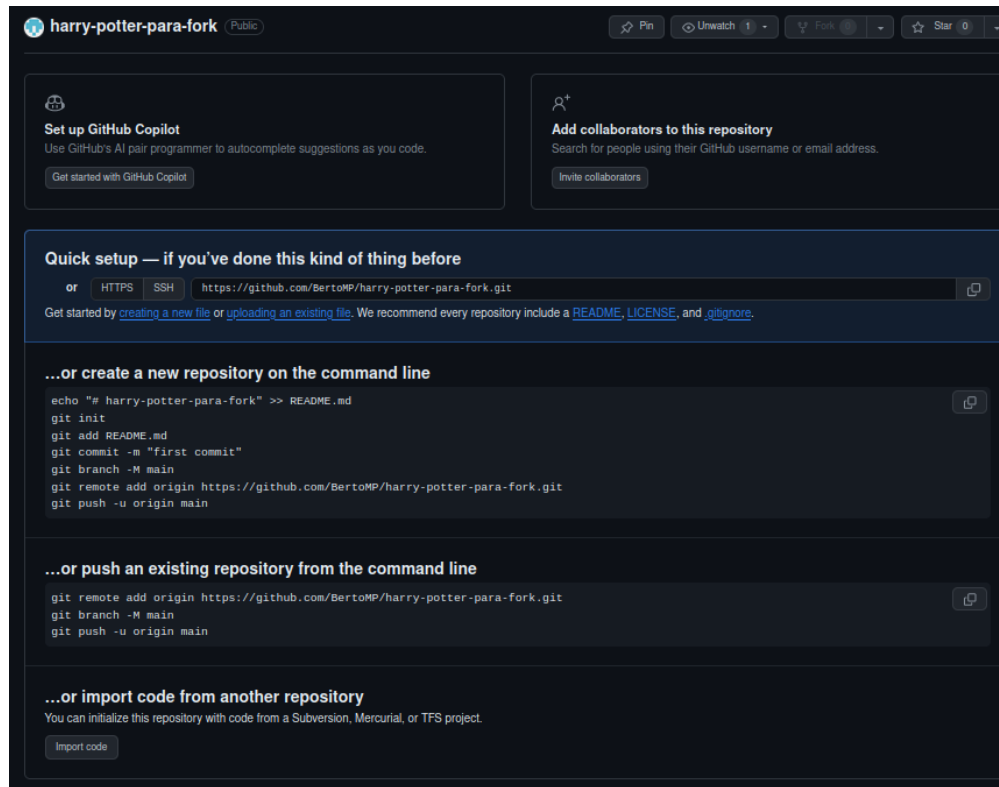
A screenshot of the 'Create a new repository' form on GitHub. The form is titled 'Create a new repository' and includes a subtitle: 'A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)'. Below this, a note states 'Required fields are marked with an asterisk (*)'. The form has two main sections: 'Owner' and 'Repository name'. The 'Owner' is set to 'BertoMP' with a dropdown arrow. The 'Repository name' is 'harry-potter-para-fork' with a dropdown arrow. A green checkmark indicates 'harry-potter-para-fork is available.'. Below these fields, a suggestion for repository names is shown: 'Great repository names are short and memorable. Need inspiration? How about [symmetrical-octo-barnacle](#) ?'. The 'Description' field is optional and contains the text 'Repositorio creado para practicar los forks.'. The 'Visibility' section has two radio buttons: 'Public' (selected) and 'Private'. The 'Initialize this repository with:' section has a checkbox for 'Add a README file'. The 'Add .gitignore' section has a dropdown menu set to 'None'. The 'Choose a license' section has a dropdown menu set to 'None'. At the bottom, there is a note: 'You are creating a public repository in your personal account.' and a green 'Create repository' button.

CONFIGURACIÓN DE UN REPOSITORIO PÚBLICO

Completamos los cambios de nombre del repositorio, de descripción del repositorio y elegimos que sea público. Además, podríamos crear un fichero README pero lo haremos posteriormente en otro apartado.

Por último, pulsamos en “Create repository”.

Esto nos habrá creado el siguiente repositorio remoto:



Para unir los repositorios debemos volver de nuevo a la línea de comandos y, lo primero será renombrar la rama de nuestro repositorio local de master a main:

```
git branch -M main
```

```
alumno@albertoMartinez:~/harry-potter$ git branch -M main
alumno@albertoMartinez:~/harry-potter$ git status
En la rama main
```

Ahora demos añadir un remoto a nuestro repositorio, para ello utilizamos el comando:

```
git remote add [nombre_remoto] [URL_remoto]
```

```
alumno@albertoMartinez:~/harry-potter$ git remote add origin https://github.com/BertoMP/harry-potter-para-fork.git
alumno@albertoMartinez:~/harry-potter$ git remote
origin
```

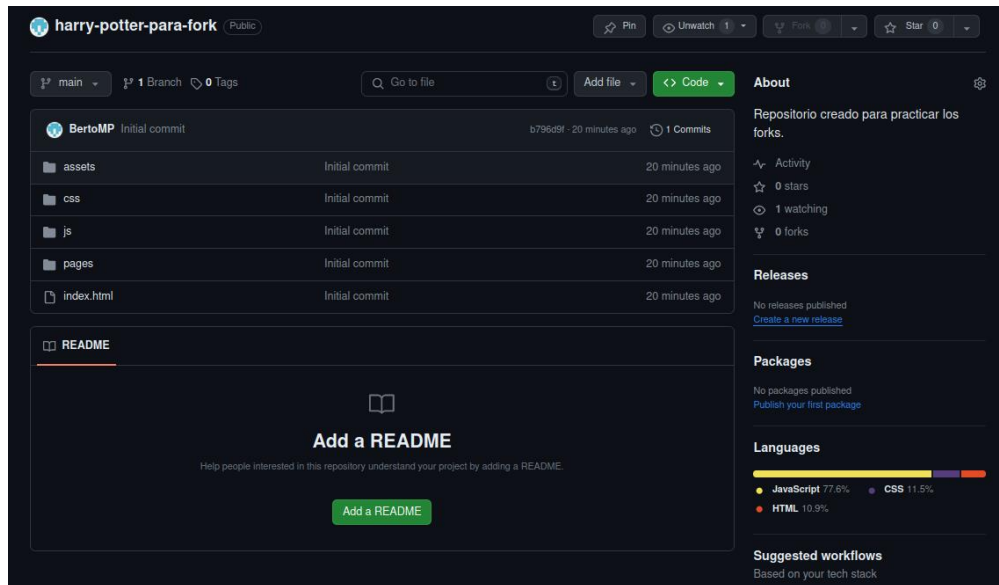
Ahora para unir nuestro remoto con nuestro local usamos el comando:

```
git push -u [nombre_remoto] [rama_a_unir]
```


CONFIGURACIÓN DE UN REPOSITORIO PÚBLICO

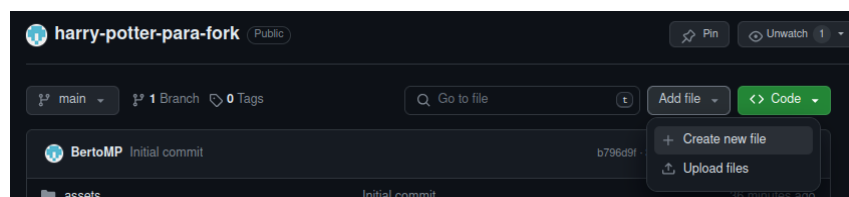
```
alumno@albertoMartinez:~/harry-potter$ git push -u origin main
Enumerando objetos: 37, listo.
Contando objetos: 100% (37/37), listo.
Compresión delta usando hasta 2 hilos
Comprimiendo objetos: 100% (36/36), listo.
Escribiendo objetos: 100% (37/37), 82.67 MiB | 13.66 MiB/s, listo.
Total 37 (delta 3), reusados 0 (delta 0), pack-reusados 0
remote: Resolving deltas: 100% (3/3), done.
To https://github.com/BertoMP/harry-potter-para-fork.git
 * [new branch]      main -> main
Rama 'main' configurada para hacer seguimiento a la rama remota 'main' de 'origin'.
```

Con esto ya hecho si volvemos a nuestro repositorio remoto veremos los archivos subidos a él:

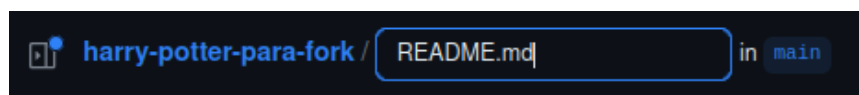


B. CREACIÓN DE UN ARCHIVO README.MD

Vamos a crear ahora un archivo README.md que sirva como explicación de lo que hace nuestra aplicación. Para crear un fichero nuevo, en nuestro repositorio hacemos clic en “Add file” y a continuación en “Create new file”:

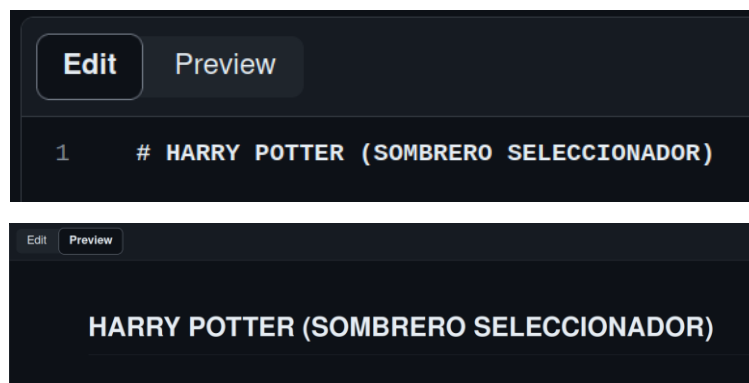


Elegimos un nombre para nuestro fichero:



Y en el cuadro de texto inferior empezamos a crear nuestro README. Vamos a empezar crear un encabezado H1 y, para ello usaremos el símbolo #:

CONFIGURACIÓN DE UN REPOSITORIO PÚBLICO



Si queremos hacer una lista no-ordenada lo haremos juntando una tabulación y un símbolo *:

```
* Slytherin      • Slytherin
* Hufflepuff     • Hufflepuff
* Ravenclaw      • Ravenclaw
* Gryffindor     • Gryffindor
```

Si utilizamos los símbolos `__` o `**` colocaremos el texto en negrita, sin embargo, si utilizamos los símbolos `*` o `_` haremos que el texto esté en cursiva:

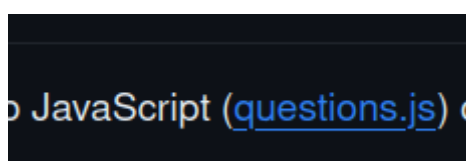
```
* _Slytherin_      • Slytherin
* __Hufflepuff__   • Hufflepuff
* *Ravenclaw*      • Ravenclaw
* **Gryffindor**   • Gryffindor
```

Para generar una tabla debemos seguir una estructura similar a la siguiente:

Casa	Puntuación
Slytherin	10
Ravenclaw	5
Gryffindor	11
Hufflepuff	3

Si queremos que nuestro README tenga un enlace, debemos hacerlo poniendo entre corchetes el texto del enlace y, a continuación, entre paréntesis la URL asociada:

```
([questions.js])(https://github.com/BertoMP/harry-potter-para-fork/blob/main/js/questions.js))
```



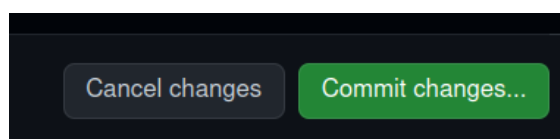
CONFIGURACIÓN DE UN REPOSITORIO PÚBLICO

Por último, si queremos enlazar un código debemos usar el símbolo ` tres veces al inicio y al final:

```
...  
{  
  question: '¿Qué color te gusta más de los siguientes?',  
  answers: [  
    {  
      text: 'Verde',  
      house: 'Slytherin',  
      points: 10  
    },  
    {  
      text: 'Rojo',  
      house: 'Gryffindor',  
      points: 10  
    },  
    {  
      text: 'Azul',  
      house: 'Ravenclaw',  
      points: 10  
    },  
    {  
      text: 'Amarillo',  
      house: 'Hufflepuff',  
      points: 10  
    }  
  ]  
}  
...
```

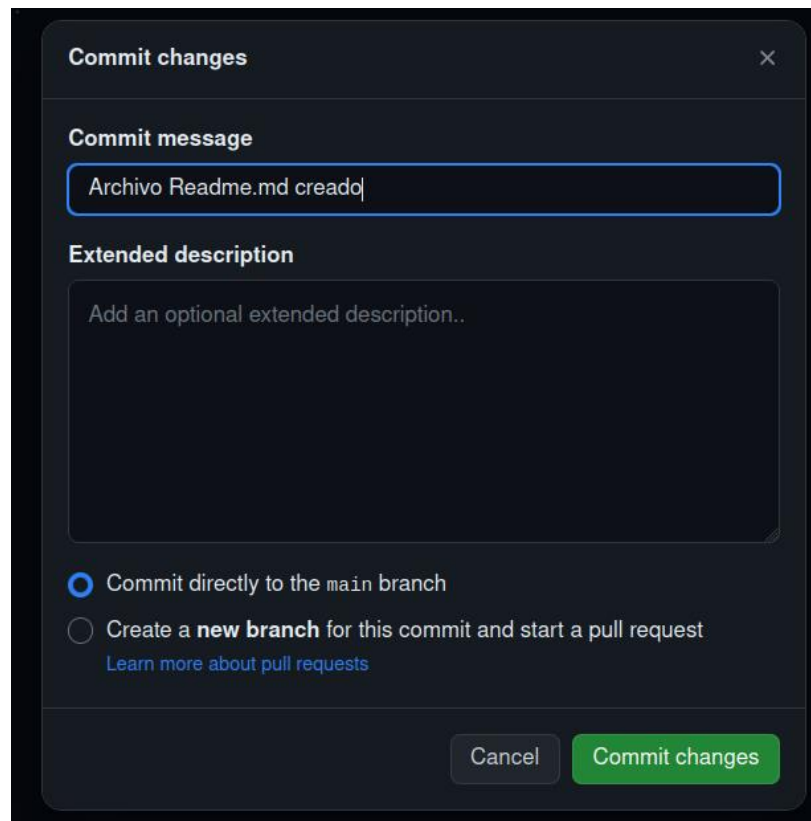
```
{  
  question: '¿Qué color te gusta más de los siguientes?',  
  answers: [  
    {  
      text: 'Verde',  
      house: 'Slytherin',  
      points: 10  
    },  
    {  
      text: 'Rojo',  
      house: 'Gryffindor',  
      points: 10  
    },  
    {  
      text: 'Azul',  
      house: 'Ravenclaw',  
      points: 10  
    },  
    {  
      text: 'Amarillo',  
      house: 'Hufflepuff',  
      points: 10  
    }  
  ]  
}
```

Para guardar los cambios de este fichero, debemos hacer clic en commit changes:



CONFIGURACIÓN DE UN REPOSITORIO PÚBLICO

Y en la ventana emergente que se abre debemos darle un texto a nuestro commit:



Si queremos bajar el README.md que hemos creado a nuestro repositorio local, debemos hacerlo a través del comando:

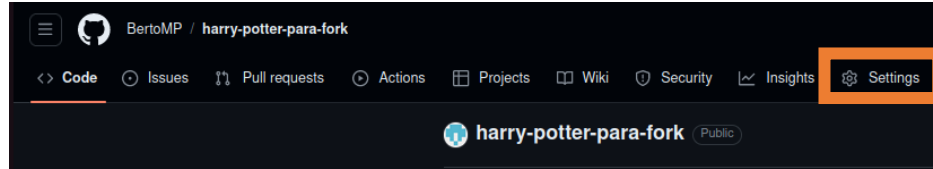
`git pull`

```
alumno@albertoMartinez:~/harry-potter$ git pull
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 3 (delta 0), pack-reused 0
Desempaquetando objetos: 100% (3/3), 1.47 KiB | 1.47 MiB/s, listo.
Desde https://github.com/BertoMP/harry-potter-para-fork
   b796d9f..89cdcc9  main      -> origin/main
Actualizando b796d9f..89cdcc9
Fast-forward
 README.md | 52 +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 1 file changed, 52 insertions(+)
 create mode 100644 README.md

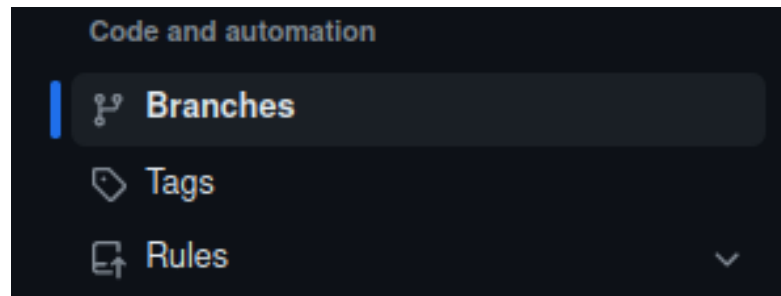
alumno@albertoMartinez:~/harry-potter$ ls -la
total 36
drwxrwxr-x  7 alumno alumno 4096 ene 15 19:40 .
drwxr-x--- 18 alumno alumno 4096 ene 15 19:28 ..
drwxrwxr-x  5 alumno alumno 4096 ene 15 12:07 assets
drwxrwxr-x  2 alumno alumno 4096 ene 15 12:07 css
drwxrwxr-x  8 alumno alumno 4096 ene 15 19:40 .git
-rw-rw-r--  1 alumno alumno 3047 ene 15 12:07 index.html
drwxrwxr-x  2 alumno alumno 4096 ene 15 12:07 js
drwxrwxr-x  2 alumno alumno 4096 ene 15 12:07 pages
-rw-rw-r--  1 alumno alumno 1437 ene 15 19:40 README.md
```

C. CONFIGURACIÓN DE SEGURIDAD

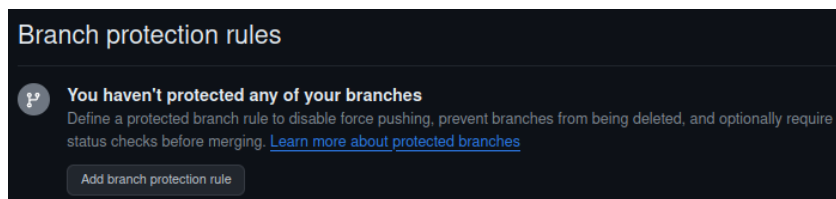
Para evitar que cualquier usuario pueda hacer cambios sobre nuestro repositorio vamos a proteger nuestra rama main, para ello nos dirigimos a “Settings”:



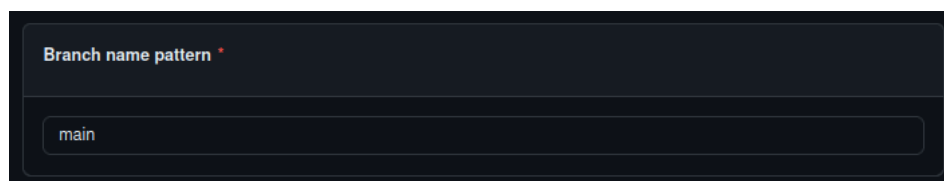
Y en la nueva ventana que se despliega nos dirigimos a “Branches”.



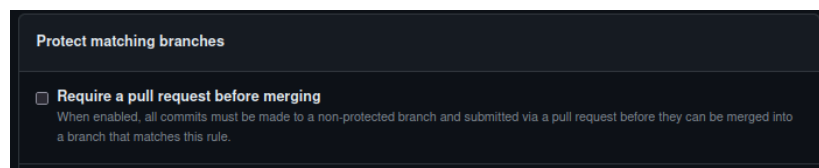
Ahora hacemos clic en “Add branch protection rule” para elegir una serie de reglas que nos sirvan para proteger la rama:



Y rellenamos el formulario con las siguientes opciones, primero determinamos el patrón de nombre de nuestra rama, en nuestro caso main:

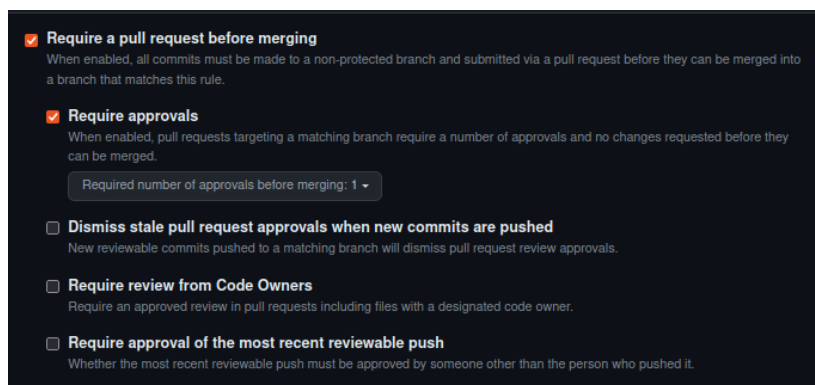


Elegimos la opción “Require a pull request before merging”:

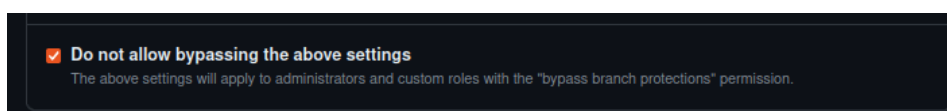


De esta forma cualquier intento de fusión de cambios necesitará de una pull-request previa:

CONFIGURACIÓN DE UN REPOSITORIO PÚBLICO



Además, activamos la opción “Do not allow bypassing the above settings” para que no se puedan sobrepasar las opciones anteriores aunque tengamos un rol de administrador:

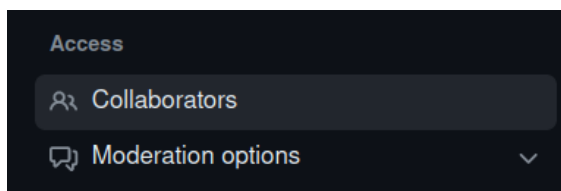


Por último, hacemos clic en create y esto generará el paquete de reglas:

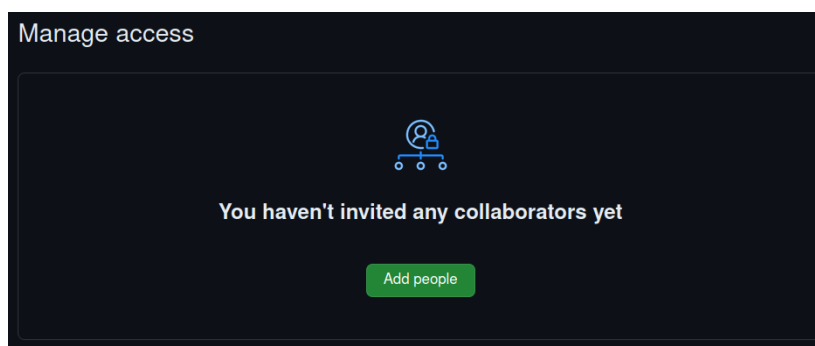


D. AÑADIR COLABORADORES AL REPOSITORIO

Para añadir a un colaborador a nuestro repositorio debemos hacerlo a través de “Collaborators” del menú “Settings” de nuestro repositorio:

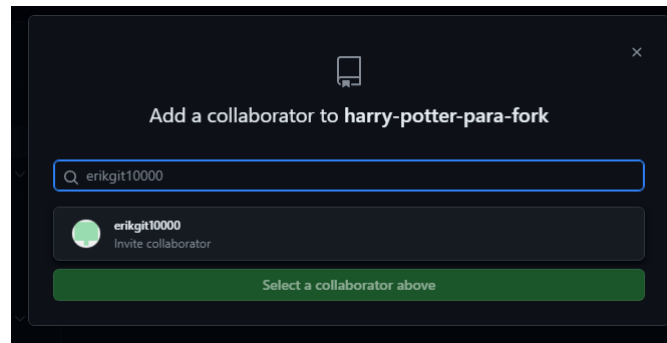


Debemos ahora hacer clic en “Add people”:

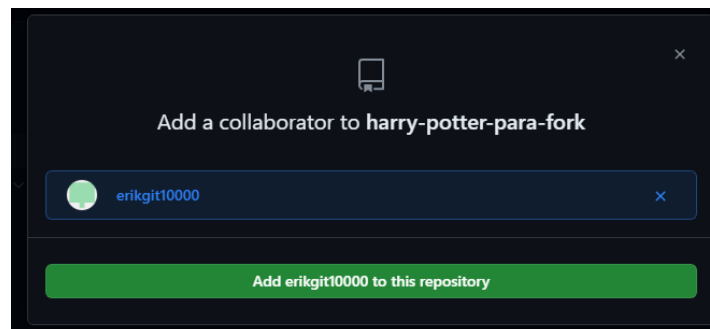


Para de esta manera buscar al colaborador que queramos añadir a nuestro repositorio y en el cuadro de búsqueda que aparece, debemos buscar a la persona que queremos añadir como colaborador del repositorio:

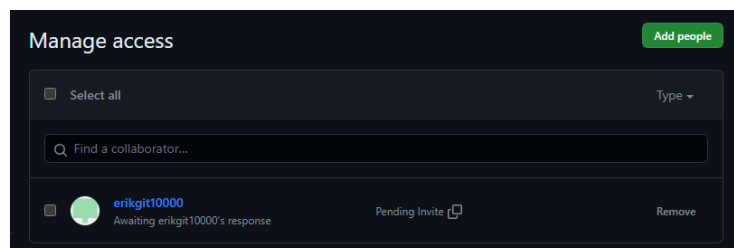
CONFIGURACIÓN DE UN REPOSITORIO PÚBLICO



A continuación, elegimos al colaborador que queremos unir a nuestro proyecto de la lista desplegable que se genera al buscar el nombre:



Y pulsamos en “Add (nombre-colaborador) to this repository”, esto enviará un mail al colaborador para que acepte el ser colaborador. Una vez acepte se convertirá en colaborador del proyecto.



Una vez que el otro desarrollador acepte la invitación, aparecerá en la ventana principal de nuestro repositorio en la sección contributors:



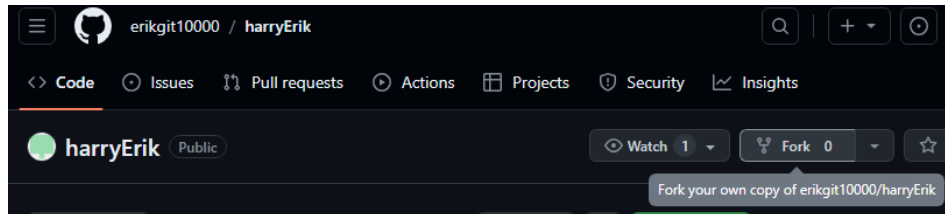
E. ENVÍO DE UNA PULL-REQUEST

Para la realización de una Pull-Request lo primero que se debe realizar es una bifurcación o fork de un repositorio público, en este caso utilizaremos

CONFIGURACIÓN DE UN REPOSITORIO PÚBLICO

el repositorio alojado en la dirección:
<https://github.com/erikgit10000/harryErik>.

Para realizar un fork debemos hacer click en “Fork” en el repositorio que queremos bifurcar:



Esto nos abrirá una nueva ventana donde debemos elegir el nombre que le daremos al fork, una vez elegido pulsamos en “Create fork”.

Create a new fork

A fork is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project.

Required fields are marked with an asterisk (*).

Owner * Repository name *

BertoMP / harryErik-fork-Alberto

harryErik-fork-Alberto is available.

By default, forks are named the same as their upstream repository. You can customize the name to distinguish it further.

Description (optional)

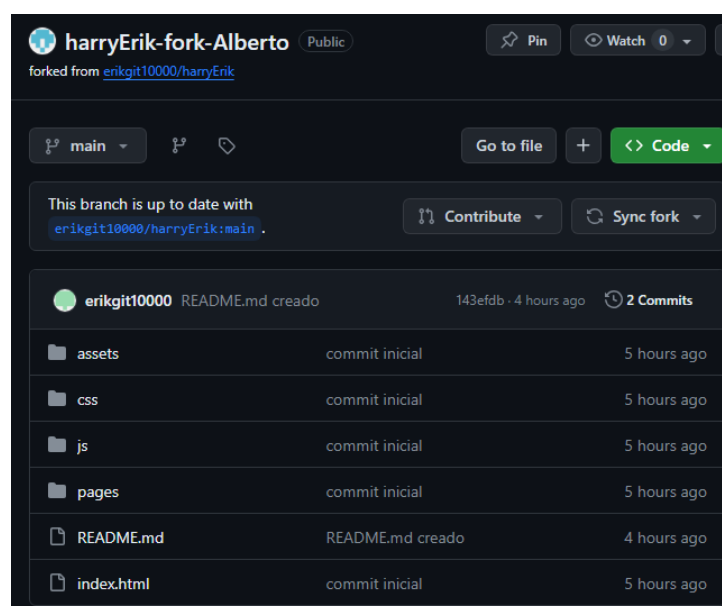
☒ Copy the main branch only

Contribute back to erikgit10000/harryErik by adding your own branch. [Learn more.](#)

You are creating a fork in your personal account.

Create fork

Esto nos generará un nuevo repositorio en nuestro listado de repositorios:



CONFIGURACIÓN DE UN REPOSITORIO PÚBLICO

Para descargar el repositorio y poder realizar cambios debemos hacer una clonación del repositorio, para ello utilizamos el comando:

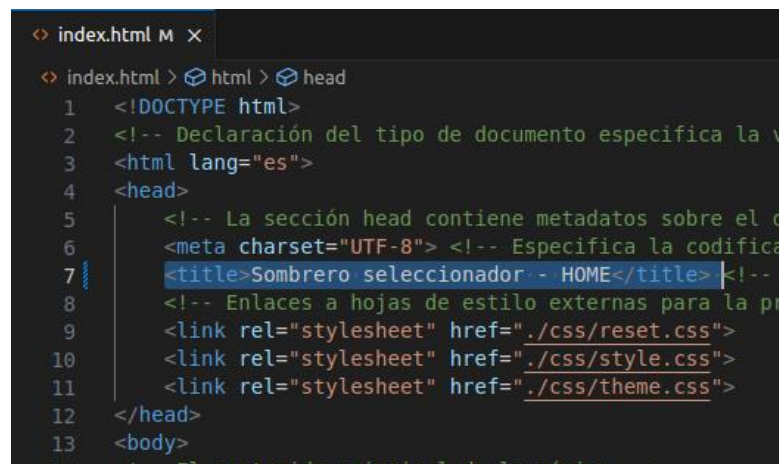
```
git clone [URL_repositorio]
```

```
alumno@albertoMartinez:~$ git clone https://github.com/BertoMP/harryErik-fork-Alberto.git
Clonando en 'harryErik-fork-Alberto'...
remote: Enumerating objects: 40, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 40 (delta 0), reused 0 (delta 0), pack-reused 37
Recibiendo objetos: 100% (40/40), 82.67 MiB | 17.98 MiB/s, listo.
Resolviendo deltas: 100% (3/3), listo.
```

Esto nos habrá creado un directorio con el proyecto:

```
alumno@albertoMartinez:~$ ls -la
total 128
drwxr-x--- 19 alumno alumno 4096 ene 16 18:57 .
drwxr-xr-x  3 root    root    4096 ene 15 15:25 ..
-rw-r----- 1 alumno alumno 2619 ene 15 19:43 .bash_history
-rw-r--r--  1 alumno alumno  220 ene 15 15:25 .bash_logout
-rw-r--r--  1 alumno alumno 3771 ene 15 15:25 .bashrc
drwx----- 12 alumno alumno 4096 ene 15 17:43 .cache
drwx----- 13 alumno alumno 4096 ene 15 17:03 .config
drwxr-xr-x  2 alumno alumno 4096 ene 16 18:43 Descargas
drwxr-xr-x  2 alumno alumno 4096 ene 15 15:47 Documentos
drwxrwxr-x  3 alumno alumno 4096 ene 15 17:09 .dotnet
drwxr-xr-x  2 alumno alumno 4096 ene 15 15:47 Escritorio
drwxrwxr-x  7 alumno alumno 4096 ene 16 18:56 harryErik-fork-Alberto
drwxrwxr-x  7 alumno alumno 4096 ene 16 18:55 harry-potter
drwxr-xr-x  2 alumno alumno 4096 ene 15 15:47 Imágenes
```

Ahora realizamos una modificación en el repositorio, por ejemplo, modificamos el título de la página principal:



```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE html>
<!-- Declaración del tipo de documento especifica la versión de HTML -->
<html lang="es">
<head>
  <!-- La sección head contiene metadatos sobre el documento -->
  <meta charset="UTF-8"> <!-- Especifica la codificación de caracteres -->
  <title>Sombrero seleccionador - HOME</title> <!-- Título de la página -->
  <!-- Enlaces a hojas de estilo externas para la página -->
  <link rel="stylesheet" href="./css/reset.css">
  <link rel="stylesheet" href="./css/style.css">
  <link rel="stylesheet" href="./css/theme.css">
</head>
<body>
  <!-- El contenido principal de la página -->
</body>
</html>
```

Y realizamos una confirmación sobre nuestros cambios haciendo primero un paso al stage de los archivos modificados (git add .) y, a continuación un commit (git commit).

```
alumno@albertoMartinez:~/harryErik-fork-Alberto$ git status
En la rama main
Tu rama está actualizada con 'origin/main'.

Cambios no rastreados para el commit:
  (usa "git add <archivo>..." para actualizar lo que será confirmado)
  (usa "git restore <archivo>..." para descartar los cambios en el directorio de trabajo)
    modificados:   index.html

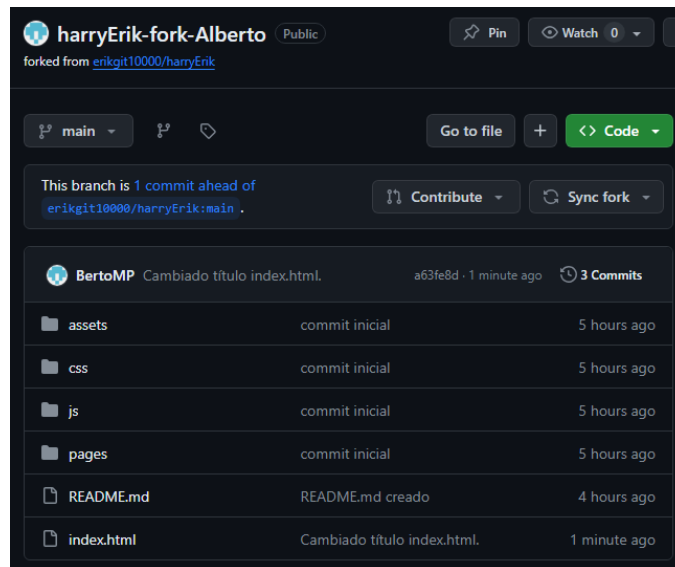
sin cambios agregados al commit (usa "git add" y/o "git commit -a")
alumno@albertoMartinez:~/harryErik-fork-Alberto$ git add .
alumno@albertoMartinez:~/harryErik-fork-Alberto$ git commit -m "Cambiado titulo index.html."
[main a63fe8d] Cambiado titulo index.html.
1 file changed, 1 insertion(+), 1 deletion(-)
```

CONFIGURACIÓN DE UN REPOSITORIO PÚBLICO

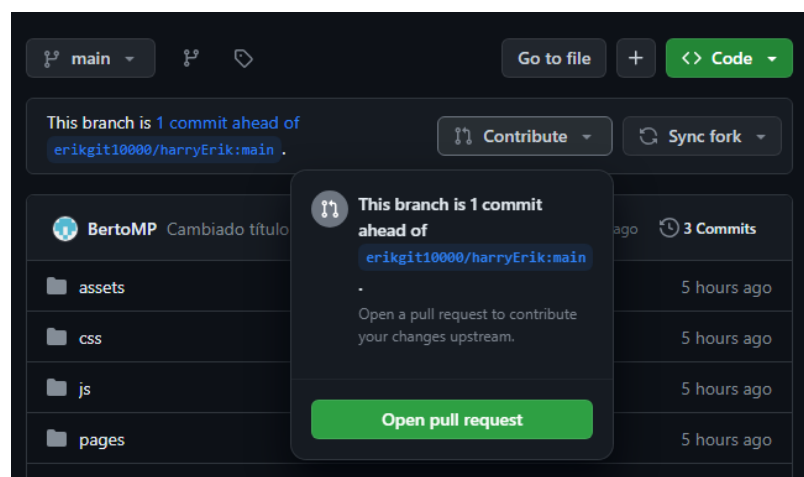
Ahora empujamos estos cambios a nuestro repositorio bifucardo:

```
alumno@albertoMartinez:~/harryErik-fork-Alberto$ git push
Enumerando objetos: 5, listo.
Contando objetos: 100% (5/5), listo.
Compresión delta usando hasta 2 hilos
Comprimiendo objetos: 100% (3/3), listo.
Escribiendo objetos: 100% (3/3), 357 bytes | 357.00 KiB/s, listo.
Total 3 (delta 2), reusados 0 (delta 0), pack-reusados 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/BertoMP/harryErik-fork-Alberto.git
143efdb..a63fe8d main -> main
```

Esto habrá hecho que los cambios se vean en nuestro repositorio remoto de GitHub:

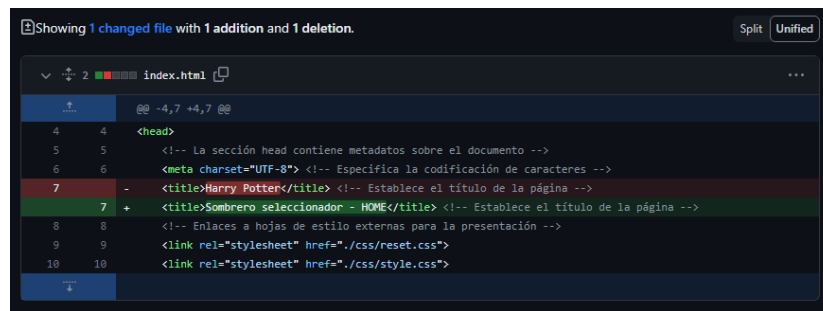


Ahora para contribuir en la mejora del repositorio original proponiendo nuestros cambios debemos hacer click en “Contribute” y, a continuación, en “Open pull request”:



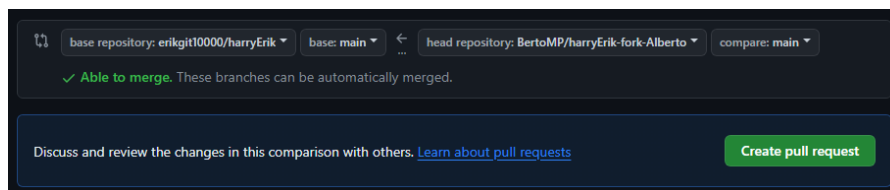
Lo que se nos muestra en la ventana siguiente es un resumen de los cambios que queremos proponer:

CONFIGURACIÓN DE UN REPOSITORIO PÚBLICO

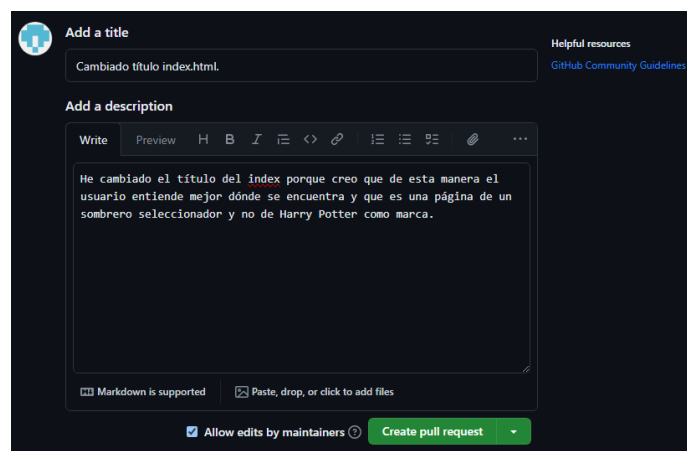


```
Showing 1 changed file with 1 addition and 1 deletion.
Split Unified
index.html
@@ -4,7 +4,7 @@
4 4 <head>
5 5 <!-- La sección head contiene metadatos sobre el documento -->
6 6 <meta charset="UTF-8"> <!-- Especifica la codificación de caracteres -->
7 7 - <title>Harry Potter</title> <!-- Establece el título de la página -->
7 7 + <title>Sombrero seleccionador - HOME</title> <!-- Establece el título de la página -->
8 8 <!-- Enlaces a hojas de estilo externas para la presentación -->
9 9 <link rel="stylesheet" href="./css/reset.css">
10 10 <link rel="stylesheet" href="./css/style.css">
```

El siguiente paso será crear la pull-request como tal, para ello hacemos clic en el botón “create pull request” de esta ventana:



En la siguiente ventana se nos pide que completemos una serie de campos, concretamente un título para nuestra pull-request (que por defecto será el mensaje que hayamos puesto en nuestro commit) y una descripción de los cambios a realizar (en este campo de texto podemos describir más en detalle lo que hemos realizado en los cambios y por qué):



Add a title

Cambiado título index.html

Add a description

Write Preview H B I

He cambiado el título del index porque creo que de esta manera el usuario entiende mejor dónde se encuentra y que es una página de un sombrero seleccionador y no de Harry Potter como marca.

☐ Markdown is supported ☒ Paste, drop, or click to add files

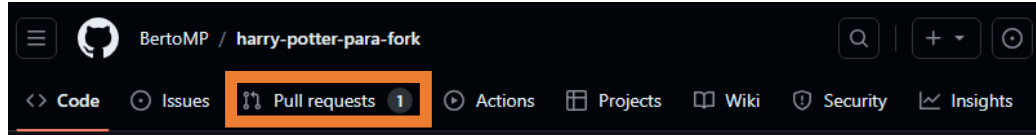
☒ Allow edits by maintainers [Create pull request](#)

Por último, hacemos clic en “Create pull request” y esto habrá creado nuestra pull-request:

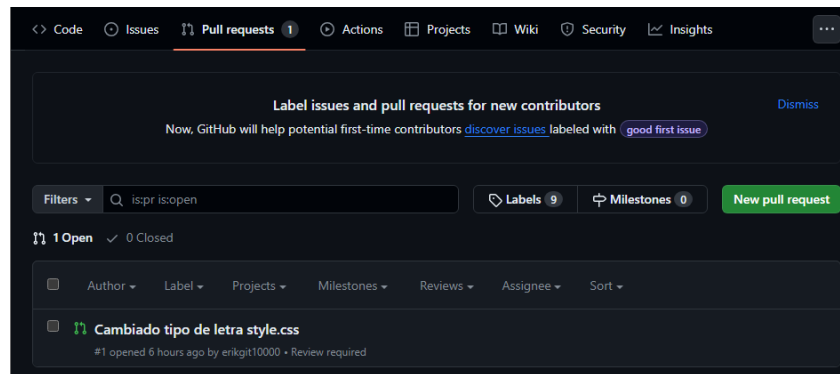


F. GESTIÓN DE UNA PULL-REQUEST

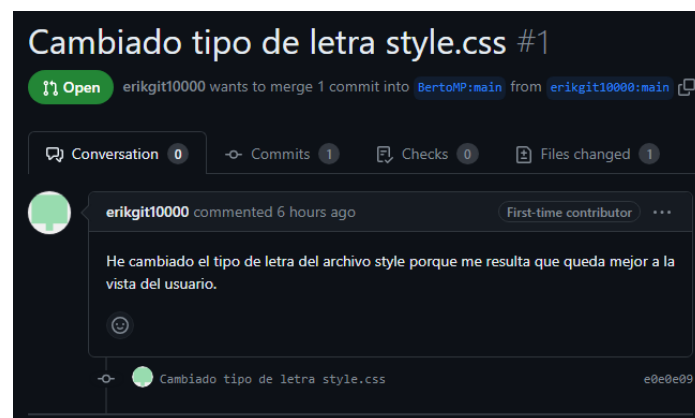
En el caso de que seamos nosotros los que recibamos una pull-request de otra persona, nos aparecerá reflejada en la barra principal del repositorio, concretamente en la sección pull request.



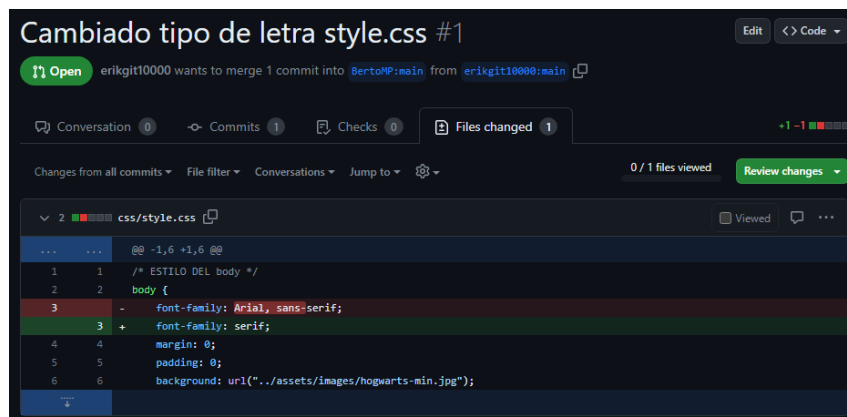
Si entramos en esta sección veremos la pull-request.



Haciendo clic sobre ella accederemos a la información.

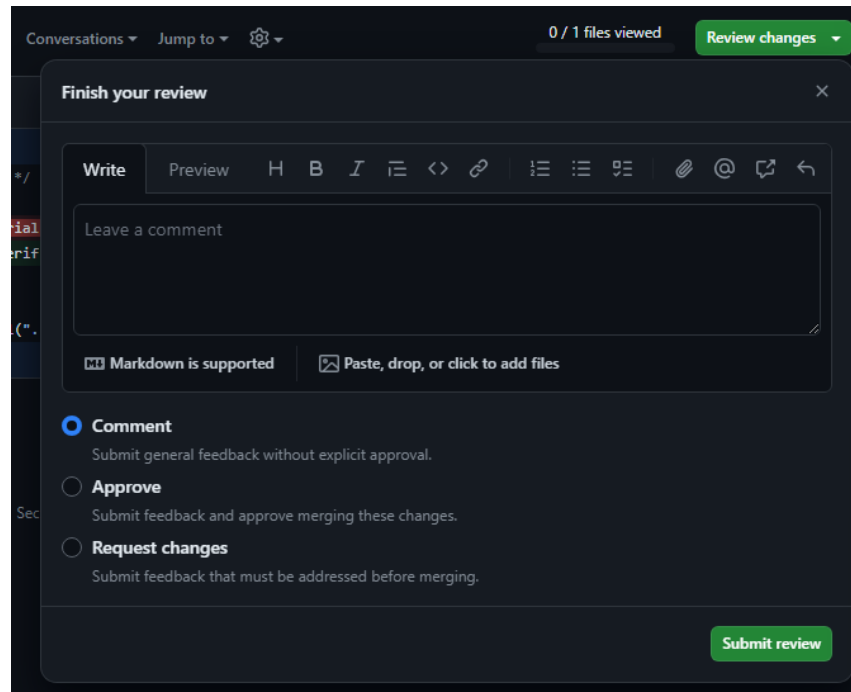


Si accedemos desde aquí a la pestaña "Files changed" podremos ver los cambios que se han realizado en los ficheros del repo.

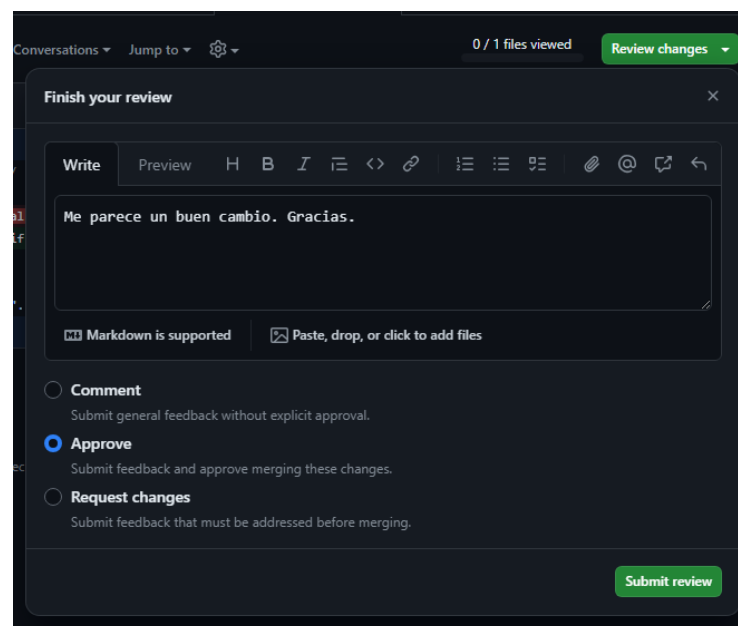


CONFIGURACIÓN DE UN REPOSITORIO PÚBLICO

Desde aquí si hacemos clic en el botón “review changes” podremos decidir si simplemente hacemos un comentario acerca de los cambios (comment), si aceptamos los cambios y los fusionamos a nuestro repositorio (approve) o si pedimos que sean revisados antes de esa fusión (request changes):

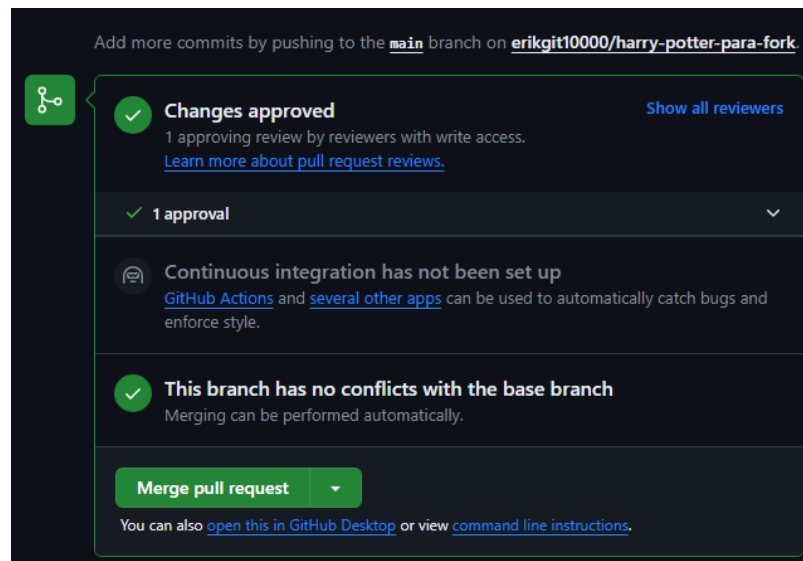


En este caso los vamos a aprobar, para ello elegimos la opción approve, escribimos un comentario en el cuadro de texto (es opcional pero da feedback al programador que ha realizado la pull-request) y pulsamos sobre submit review.

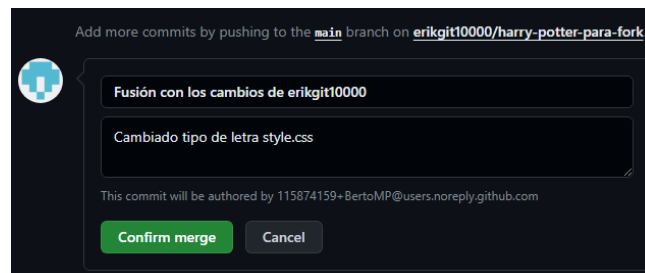


CONFIGURACIÓN DE UN REPOSITORIO PÚBLICO

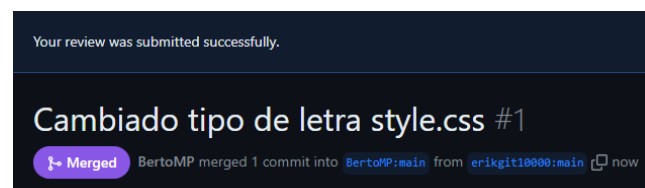
Una vez hecho esto se nos devolverá a la pestaña “conversation” donde debemos finalizar la gestión haciendo clic sobre el botón “merge pull request”:



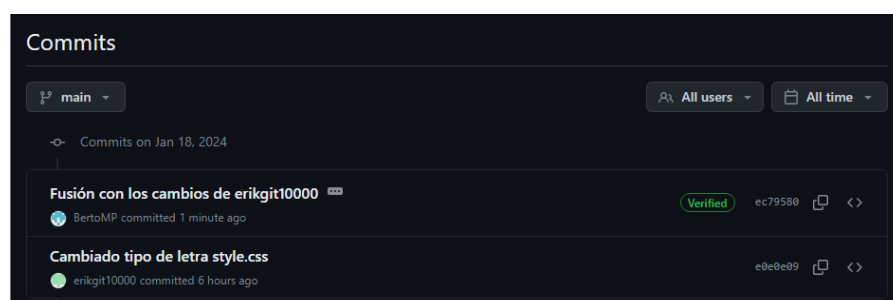
Esto nos solicitará que demos un nombre a esta acción ya que de forma automática se producirá un commit cuando se produzca la fusión de cambios.



Pulsamos en “confirm merge” y se ejecutará la fusión.



Si ahora vamos al apartado “code” de nuestro repositorio y accedemos a los commits, veremos los nuevos commits que se han generado:

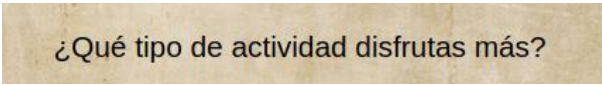


CONFIGURACIÓN DE UN REPOSITORIO PÚBLICO

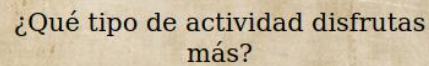
Para bajarnos estos cambios a nuestro repositorio local, debemos realizar un pull:

```
alumno@albertoMartinez:~/harry-potter$ git pull
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 5 (delta 2), reused 4 (delta 2), pack-reused 0
Desempaquetando objetos: 100% (5/5), 1.27 KiB | 130.00 KiB/s, listo.
Desde https://github.com/BertoMP/harry-potter-para-fork
      89cdcc9..ec79580  main      -> origin/main
Actualizando 89cdcc9..ec79580
Fast-forward
 css/style.css | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)
```

De forma que ahora el estilado de la letra pasa de estar en una fuente no serifada:

A screenshot of a text box with a light beige, textured background. The text inside is "¿Qué tipo de actividad disfrutas más?" written in a black serif font.

A tener una letra serifada:

A screenshot of a text box with a light beige, textured background. The text inside is "¿Qué tipo de actividad disfrutas más?" written in a black serif font, identical to the one above.