

Práctica Obligatoria

Un cliente encargado de realizar formaciones a empresas ha decidido crear un curso nuevo para poder enseñar *javascript* aplicado a interacciones con elementos HTML. Para ello, nos ha pedido que implementemos una pequeña página web en la que mostremos algunos ejemplos en los que se utilice este lenguaje de programación. Su idea es que el producto final tenga un aspecto similar al siguiente esquema:



Nuestro equipo de programadores está compuesto por 3 personas. Por este motivo, hemos decidido dividir el trabajo de la siguiente forma:

- El usuario 1 que es el más experimentado, se encargará de crear el repositorio del proyecto e implementará la estructura inicial de la página en la que incluirá la cabecera, la barra de navegación con el acceso a Home y el *footer* de la página. En la parte central del home incluirá una breve descripción del curso.
- El usuario 2 creará la sección de “Modificar contenido HTML” en el que incluirá algún ejemplo similar al que existe en el siguiente [enlace](#) así como una explicación de dicho código. También desarrollará la sección “Modificar atributos HTML”

- El usuario 3 que acaba de llegar a la empresa implementará la sección “Modificar estilos CSS” de la misma forma que ha hecho el usuario 2. Añadirá ejemplos similares a [este](#) (comentando el código).

Además, debido a la complejidad del proyecto, se ha decidido seguir la metodología Git Flow para poder realizar el desarrollo.

Así pues, nuestra tarea consistirá en crear un repositorio en GitHub en el que simulemos toda la implementación acordada además de documentar todo el proceso correctamente para que nuestro cliente vea que somos una empresa seria y trabajamos siguiendo una metodología (esto incluye mensajes de *commit* siguiendo las buenas prácticas vistas en clase). Esta documentación, deberá estar incluida en el GitHub Pages del propio repositorio del proyecto y deberá contener:

- Una introducción teórica de qué es Git, la metodología Git Flow y por qué la utilizamos en nuestro desarrollo. (0,75 puntos)



DESPLIEGUE DE APLICACIONES WEB

[Home](#)[Modificar contenido HTML](#)[Modificar estilos CSS](#)[Modificar atributos HTML](#)

Que es GIT?

Git es un sistema de control de versiones distribuido que permite a los desarrolladores colaborar en proyectos de software de manera eficiente y organizada. Git almacena el historial de cambios de un proyecto en forma de ramas, que son versiones independientes del código.

Que es la metodologia GitFlow?

Gitflow es un modelo alternativo de creación de ramas en Git en el que se utilizan ramas de función y varias ramas principales. Fue Vincent Driessen en nvie quien lo publicó por primera vez y quien lo popularizó. En comparación con el desarrollo basado en troncos, Gitflow tiene diversas ramas de más duración y mayores confirmaciones.

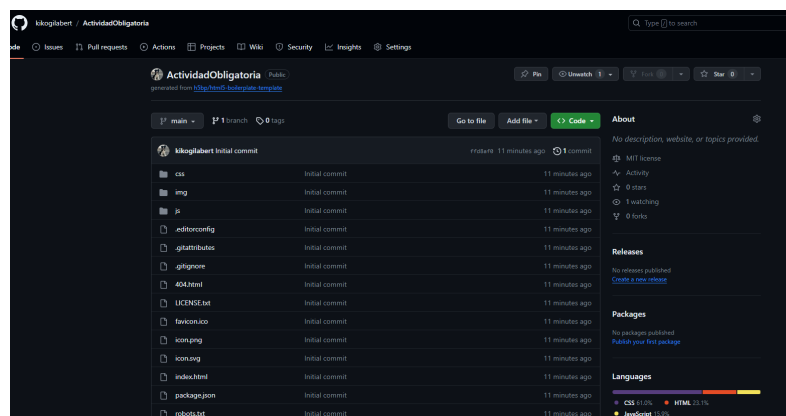
Porque deberiamos usar GitFlow?

Gitflow es un modelo alternativo de creación de ramas en Git en el que se utilizan ramas de función y varias ramas principales. Fue Vincent Driessen en nvie quien lo publicó por primera vez y quien lo popularizó. En comparación con el desarrollo basado en troncos, Gitflow tiene diversas ramas de más duración y mayores confirmaciones.

Igual más adelante añado fotos.

- Explicación detallado del proceso realizado (incluyendo capturas) en el que:

- El **usuario 1 *** que es el más experimentado, se encargará de crear el repositorio del proyecto,



```

Lenovo@DESKTOP-URU9V56 MINGW64 ~/Desktop/KIKO/2DAW/Despliegue de aplicaciones/ActividadObligatoria/ActividadObligatoria (main)
$ npm install

up to date, audited 363 packages in 1s

58 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities

Lenovo@DESKTOP-URU9V56 MINGW64 ~/Desktop/KIKO/2DAW/Despliegue de aplicaciones/ActividadObligatoria/ActividadObligatoria (main)
$ npm run start

> @0.0.1 start
> webpack serve --open --config webpack.config.dev.js

<i> [webpack-dev-server] Project is running at:
<i> [webpack-dev-server] Loopback: http://localhost:8080/
<i> [webpack-dev-server] On Your Network (IPv4): http://172.28.70.30:8080/

```

adecuará este para poder utilizar la metodología git flow (creación de ramas iniciales)

```

Lenovo@DESKTOP-URU9V56 MINGW64 ~/Desktop/KIKO/2DAW/Despliegue de aplicaciones/ActividadObligatoria/ActividadObligatoria (master)
$ git checkout -b develop
Switched to a new branch 'develop'

```

```

ktop\KIKO\2DAW\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> git checkout develop
ktop\KIKO\2DAW\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> git branch feature_home
ktop\KIKO\2DAW\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> git checkout feature_home
ktop\KIKO\2DAW\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria>

```

Se implementará la estructura inicial del proyecto utilizando el siguiente boilerplate ([link](#)). Esta página inicial incluirá la cabecera, la barra de navegación con el acceso a Home y el *footer* de la página. (1'5 puntos)



He commiteado los cambios en la rama *feature_home* y los he subido al origen. Después he cambiado a la rama *develop* y he hecho un merge para añadir los cambios a mi rama develop.

- Además, el **usuario 1** que quiere que todo funcione correctamente, creará dos hooks (utilizando la librería husky) para el proyecto y que harán lo siguiente:

```
oria\ActividadObligatoria> npm install husky
```

```
scripts : {  
  "test": "echo \"Error: no test specifi",  
  "start": "webpack serve --open --confi",  
  "build": "webpack --config webpack.con",  
  "prepare" : "husky install"  
},
```

- El primero comprobará que siempre que se realice un commit en la rama hotfix exista una persona asignada para verificar que no se rompe nada. Para ello, deberá comprobar que el mensaje de commit contiene la frase “A corregir por: ...” (0,75 puntos)

```
ActividadObligatoria > .husky > pre-commit
1  #!/bin/bash
2  commit_message=$(git log -1 --pretty=%B)
3
4  if [ "$(git rev-parse --abbrev-ref HEAD)" == "hotfix_errorhome" ] && [[ $commit_message != *"A corregir por: *" ]] then
5      echo "Error: Debes asignar a alguien para verificar el hotfix. El mensaje de commit debe contener 'A corregir por: ...'."
6      exit 1
7  fi
```

Comprobación de que no funciona con un formato diferente al creado:

```
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> git commit -m "Esto es una prueba"
Error: Debes asignar a alguien para verificar el hotfix. El mensaje de commit debe contener 'A corregir por: ...'.
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria>
```

Comprobación de que funciona con el formato declarado:

```
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> git commit -m "A corregir por: Lolito"
[hotfix_errorhome c6f47a5] A corregir por: Lolito
1 file changed, 1 insertion(+), 1 deletion(-)
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria>
```

- El segundo simplemente mostrará en consola un mensaje con un dibujo hecho con caracteres ascii una vez hagamos el checkout de una rama del proyecto. Podéis coger ejemplos de esta [página](#). (0,5 puntos)

[illegible]

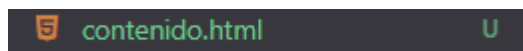
[illegible]

- El **usuario 2 ***, crea dos features (feature/contenidoHTML y feature/atributosHTML, una por cada sección que debe implementar). Además, añadirá el siguiente hook (1 punto):

feature_contenidoHTML:

primero he creado la rama con `git checkout -b feature_contenidoHTML` desde la rama `develop`:

después creamos un html nuevo para dicha feature:



y añadido código:

```
<body>
  <div id="menu">
    <header>
      
      <h1 class="titulo">DESPLIEGUE DE APLICACIONES WEB</h1>
    </header>
    <main>
      <nav>
        <ul class="ulmenu">
          <li class="menu"> <a href="index.html">Home</a></li>
          <li class="menu"> <a href="contenido.html">Modificar contenido HTML</a></li>
          <li class="menu"> <a href="atributos.html">Modificar atributos HTML</a></li>
          <li class="menu"> <a href="listas.html">Modificar atributos HTML</a></li>
        </ul>
      </nav>
    </div>
    <section>
      <h2>What Can JavaScript Do?</h2>
      <p id="demo">JavaScript can change HTML content.</p>
      <button type="button" onclick='document.getElementById("demo").innerHTML = "Hello JavaScript!">Click Me!</button>
    </section>
  </main>
</body>
```



DESPLIEGUE DE APLICACIONES WEB

Home

Modificar contenido HTML

Modificar estilos CSS

Modificar atributos HTML

What Can JavaScript Do?

JavaScript can change HTML content.

[Click Me!](#)

Bienvenido a esta pagina creada por Kiko

Para consultas, no dudes en ponerte en contacto con nosotros en info@tusilio.com o a través de nuestro número de teléfono (123) 456-7890.

```
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> git add .
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> git commit "usuario2: New feature for home"
error: pathspec 'usuario2: New feature for home' did not match any file(s) known to git
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> git commit -m "usuario2: New feature for home"
[feature_contenidoHTML f2b7fec] usuario2: New feature for home
2 files changed, 67 insertions(+), 1 deletion(-)
create mode 100644 contenido.html
```


```
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> git checkout develop
Switched to branch 'develop'
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> git merge feature_contenidoHTML
Updating ffd8af0..f2b7fec
Fast-forward
 contenido.html | 66 +
 css/style.css | 291 +---
 img/git.png | Bin 0 -> 4340 bytes
 index.html | 50 +-
 package-lock.json | 4126 +++++
 5 files changed, 4310 insertions(+), 223 deletions(-)
 create mode 100644 contenido.html
 create mode 100644 img/git.png
 create mode 100644 package-lock.json
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria>
```

feature atributos HTML:

creamos la rama de la nueva feature:

```
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> git checkout -b feature_atributosHTML
Switched to a new branch 'feature_atributosHTML'
```

y creamos el archivo atributos.html para crear la nueva feature.

 atributos.html

Yo he añadido el siguiente código

```
<body>
<div id="menu">
<header>

<h1 class="titulo">DESPLIEGUE DE APLICACIONES WEB</h1>
</header>
<main>
<nav>
<ul class="ulmenu">
<li class="menu"><a href="index.html">Home</a></li>
<li class="menu"><a href="contenido.html">Modificar contenido HTML</a></li>
<li class="menu"><a href="atributos.html">Modificar atributos HTML</a></li>
<li class="menu"><a href="listas.html">Modificar atributos HTML</a></li>
</ul>
</nav>
</div>
<section>
<h2>What Can JavaScript Do?</h2>
<p>JavaScript can change HTML attribute values.</p>
<p>In this case JavaScript changes the value of the src (source) attribute of an image.</p>
<button onclick="document.getElementById('myImage').src='https://www.w3schools.com/js/pic_bulbon.gif'">Turn on the light</button>

<button onclick="document.getElementById('myImage').src='https://www.w3schools.com/js/pic_bulboff.gif'">Turn off the light</button>
</section>
</main>
<br>
```



DESPLIEGUE DE APLICACIONES WEB

[Home](#)

[Modificar contenido HTML](#)

[Modificar atributos HTML](#)

[Modificar atributos HTML](#)

What Can JavaScript Do?

JavaScript can change HTML attribute values.

In this case JavaScript changes the value of the src (source) attribute of an image.



Turn on the light

Turn off the light



DESPLIEGUE DE APLICACIONES WEB

[Home](#)

[Modificar contenido HTML](#)

[Modificar atributos HTML](#)

[Modificar atributos HTML](#)

What Can JavaScript Do?

JavaScript can change HTML attribute values.

In this case JavaScript changes the value of the src (source) attribute of an image.



y volveremos a hacer el mismo procedimiento que antes una vez se a terminado la feature:

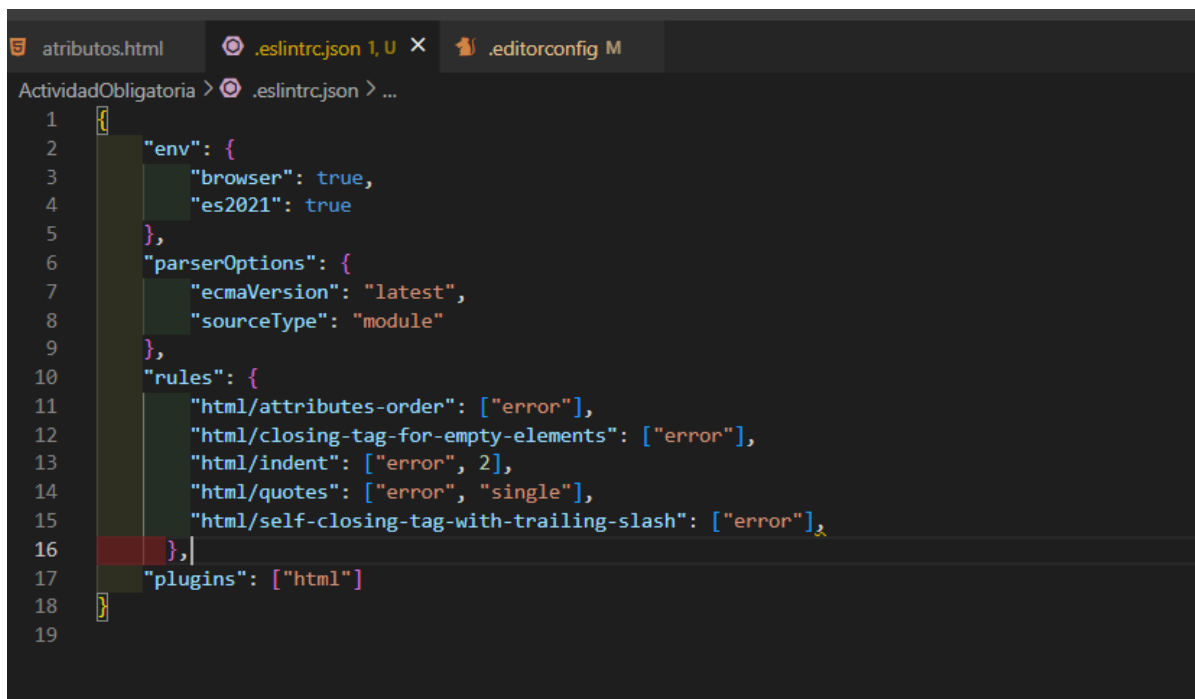
```
Switched to a new branch 'feature_atributosHTML'
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> git add .
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> git commit -m "usuario2: finished feature_atributosHTML"
[feature_atributosHTML 23b2c7f] usuario2: finished feature_atributosHTML
1 file changed, 72 insertions(+)
create mode 100644 atributos.html
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> git checkout develop
Switched to branch 'develop'
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> git merge feature_atributosHTML
Updating f2b7fec..23b2c7f
Fast-forward
 atributos.html | 72 +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
1 file changed, 72 insertions(+)
create mode 100644 atributos.html
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria>
```

- Previo a realizar el commit, se verificará el correcto formato en los ficheros html. Para ello, se ejecutará el linter eslint utilizando el siguiente plugin ([link](#)) (0,5 puntos).

```
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria> cd .\ActividadObligatoria\
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> npm install --save-dev eslint-plugin-html

added 6 packages, removed 1 package, and audited 369 packages in 4s

63 packages are looking for funding
  run `npm fund` for details
```



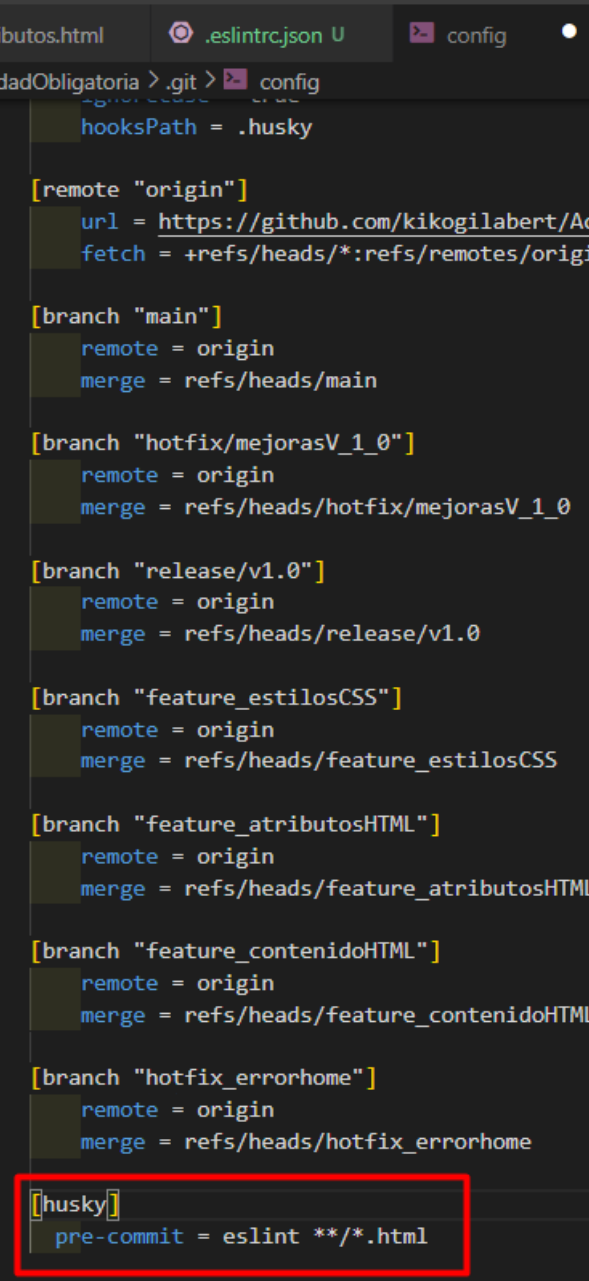
The screenshot shows the VS Code editor with the `.eslintrc.json` file open. The configuration is as follows:

```
{
  "env": {
    "browser": true,
    "es2021": true
  },
  "parserOptions": {
    "ecmaVersion": "latest",
    "sourceType": "module"
  },
  "rules": {
    "html/attributes-order": ["error"],
    "html/closing-tag-for-empty-elements": ["error"],
    "html/indent": ["error", 2],
    "html/quotes": ["error", "single"],
    "html/self-closing-tag-with-trailing-slash": ["error"],
  },
  "plugins": ["html"]
}
```

Ejecuto el plugin para verificar si existe algún fallo en mis archivos .html

```
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> eslint **/*.html
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria>
```

y añado el siguiente código a mi archivo de gitconfig para asegurar que verifique antes de hacer un commit:



```
.gitconfig
[core]
  hooksPath = .husky

[remote "origin"]
  url = https://github.com/kikogilabert/ActividadObligatoria
  fetch = +refs/heads/*:refs/remotes/origin/*

[branch "main"]
  remote = origin
  merge = refs/heads/main

[branch "hotfix/mejorasV_1_0"]
  remote = origin
  merge = refs/heads/hotfix/mejorasV_1_0

[branch "release/v1.0"]
  remote = origin
  merge = refs/heads/release/v1.0

[branch "feature_estilosCSS"]
  remote = origin
  merge = refs/heads/feature_estilosCSS

[branch "feature_atributosHTML"]
  remote = origin
  merge = refs/heads/feature_atributosHTML

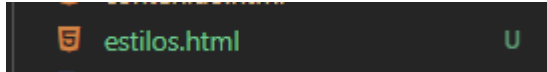
[branch "feature_contenidoHTML"]
  remote = origin
  merge = refs/heads/feature_contenidoHTML

[branch "hotfix_errorhome"]
  remote = origin
  merge = refs/heads/hotfix_errorhome

[husky]
  pre-commit = eslint **/*.html
```

- El **usuario 3** *, crea una feature (feature/estilosCSS) asociada al cambio para añadir su sección. El usuario 3, además, crea una nueva release con todas las features que etiqueta como v1.0. (2 puntos)

```
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> git checkout develop
Switched to branch 'develop'
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> git checkout -b feature_estilosCSS
Switched to a new branch 'feature_estilosCSS'
```



DESPLIEGUE DE APLICACIONES WEB

[Home](#)
[Modificar contenido HTML](#)
[Modificar atributos HTML](#)
[Modificar estilos CSS](#)

What Can JavaScript Do?

JavaScript can change the style of an HTML element.

```
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> git add .
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> git commit -m "Usuario 3: feature_estilosCSS finished"
[feature_estilosCSS dc028fd] Usuario 3: feature_estilosCSS finished
4 files changed, 67 insertions(+), 9 deletions(-)
create mode 100644 estilos.html
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria>
```

```
your branch is up to date with 'origin/main'.
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> git checkout -b release/v1.0
Switched to a new branch 'release/v1.0'
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> git merge develop
Updating ffd8af0..dc028fd
Fast-forward
 atributos.html | 63 +
 contenido.html | 66 +
 css/style.css | 291 +---
 estilos.html | 58 +
 img/git.png | Bin 0 -> 4340 bytes
 index.html | 50 +-
 package-lock.json | 4126 +++++
 7 files changed, 4431 insertions(+), 223 deletions(-)
 create mode 100644 atributos.html
 create mode 100644 contenido.html
 create mode 100644 estilos.html
 create mode 100644 img/git.png
 create mode 100644 package-lock.json
```

- El **usuario 1** * que se encarga de las pruebas, decide mejorar un poco la sección del contenido creada por el usuario 2. Para ello, crea un hotfix (hotfix/mejorasV_1_0) que efectúa dicha mejora. (1,5 puntos)

```
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> git checkout -b hotfix/mejorasV_1_0 release/v1.0
Switched to a new branch 'hotfix/mejorasV_1_0'
```

```
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> git add .
PS C:\Users\KIKO TEST2\Desktop\Despliegue de aplicaciones\ActividadObligatoria\ActividadObligatoria> git commit -m "usuariol: Mejoras implementadas en release/v1.0"
[hotfix/mejorasV_1_0 5365d19] usuariol: Mejoras implementadas en release/v1.0
```

Al finalizar el trabajo, nuestro repositorio deberá contener, además de las ramas necesarias para poder realizar el desarrollo comentado, una rama (gh-pages) en la que se incluirá toda la documentación del proceso y un proyecto creado en github en el que se dará acceso como colaborador al usuario de github pauvilarpastor. En el canvas de dicho proyecto, deberán aparecer las features y hotfix que se han creado a lo largo de esta actividad con toda la información posible. (1,5 puntos)

REPOSITORIO: <https://github.com/kikogilabert/ActividadObligatoria/tree/gh-pages>

Nota: Para simular que somos el usuario 1, 2 o 3, en el mensaje del commit incluiremos el nombre de cada uno de ellos. Los hooks de husky deben subirse al repositorio para que cualquier usuario que lo clone disponga de ellos.