

# **Informe de la metodología de gestión de la configuración**

**PSG2-2223-g7-72**

**Proceso Software y Gestión 2, Universidad de Sevilla**

# Resumen

En el siguiente documento se detalla el **historial** de **cambios** realizados en el código para el *Sprint 2* del proyecto psg2-2223-g7-72. En primer lugar el equipo va a trabajar sobre una sola rama (Trunk-based development) para forzar conflictos y resolverlos. En segundo lugar el equipo va a trabajar siguiendo el sistema de **Git flow**, de tal forma que se podrá comprobar las ventajas y desventajas de cada una de estas estrategias.

# Antecedentes

Al iniciar el proyecto los desarrolladores comenzaron a realizar las tareas A2.5 (detalladas en el [siguiente apartado](#)) sin que aparecieran conflictos, debido a la siguiente causa:

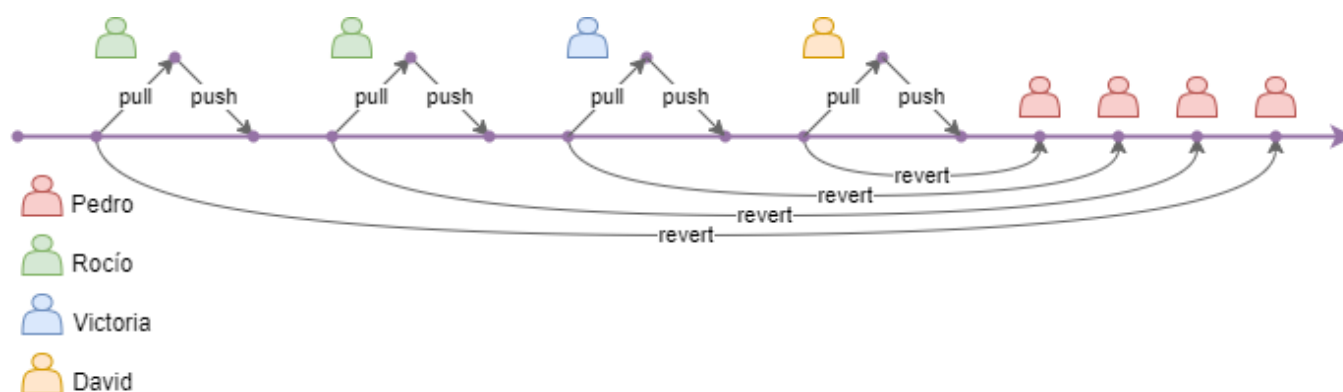
***Se estaba realizando un PULL del repositorio antes de comenzar con la tarea que correspondía.***

Durante este periodo inicial de tiempo los desarrolladores siguieron los siguientes pasos:

1. Hacer **PULL** del repositorio antes de empezar a trabajar.
2. Realizar sus cambios en el código.
3. Hacer **PUSH** al repositorio, para que otros compañeros pudieran bajar sus cambios en el siguiente **PULL**.

De esta forma, el objetivo de la tarea (generar conflictos y solucionarlos) no podía cumplirse, por lo que se decidió realizar **REVERT** de las tareas realizadas hasta el momento y volver a empezar siguiendo las pautas detalladas en el backlog de la asignatura.

A continuación se detalla mediante un esquema los commits realizados:



---

`git revert [hash-commit(s)]` : Realiza un commit deshaciendo los cambios de un commit o commits en concreto. Por tanto, en los siguientes apartados de este documento partimos desde este punto, donde a pesar de haber realizado algunos commits, el proyecto ha vuelto a su estado inicial.

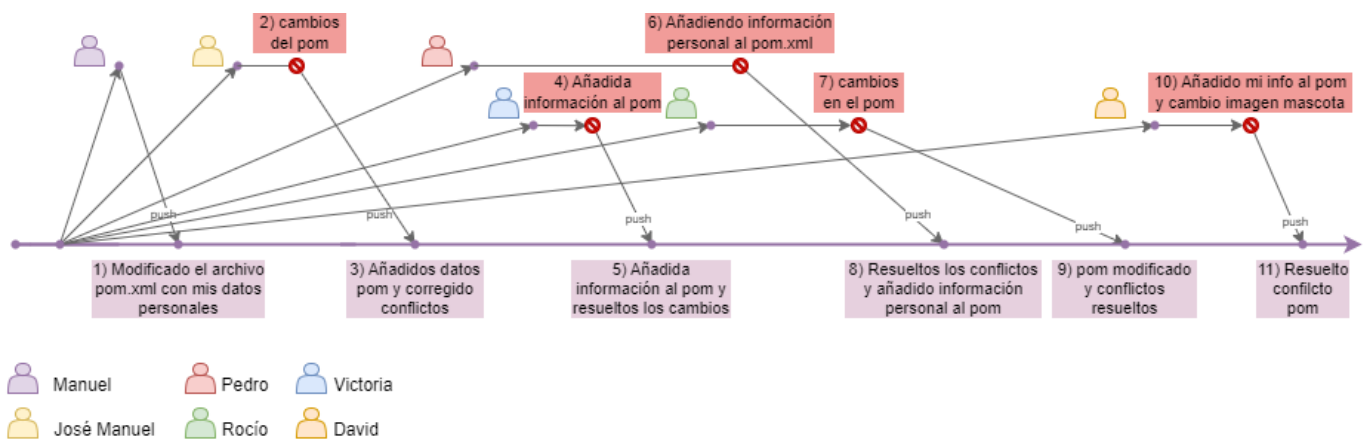
---

# Tareas realizadas por cada desarrollador

- Pedro González Marcos
  - **A2.5a:** Agregar información personal al `pom.xml`
  - **A2.5e):** Cambiar el mensaje de la página de Bienvenida
- Rocío López Moyano
  - **A2.5a:** Agregar información personal al `pom.xml`
  - **A2.5b:** Cambiar la información del `pom.xml` según atributos
  - **A2.5h:** Cambiar la información del `info.yml`
- José Manuel Moreno Guerrero
  - **A2.5a:** Agregar información personal al `pom.xml`
- Victoria de Carmen Ruíz Delgado
  - **A2.5a:** Agregar información personal al `pom.xml`
  - **A2.5d:** Cambiar el color de fondo del menú
- Manuel Vázquez Martín
  - **A2.5a:** Agregar información personal al `pom.xml`
  - **A2.5c:** Cambiar el color de los botones del encabezado
  - **A2.5g:** Cambiar el color de fondo del encabezado de la tabla
- David Zarandieta Ortiz
  - **A2.5a:** Agregar información personal al `pom.xml`
  - **A2.5f:** Cambiar la imagen de la página de Bienvenida

# Historial de cambios A2.5

A continuación se detalla mediante un esquema los cambios realizados en el proyecto, se indica con un círculo rojo cada vez que aparece un conflicto y se explica en el apartado siguiente como se han solucionado:

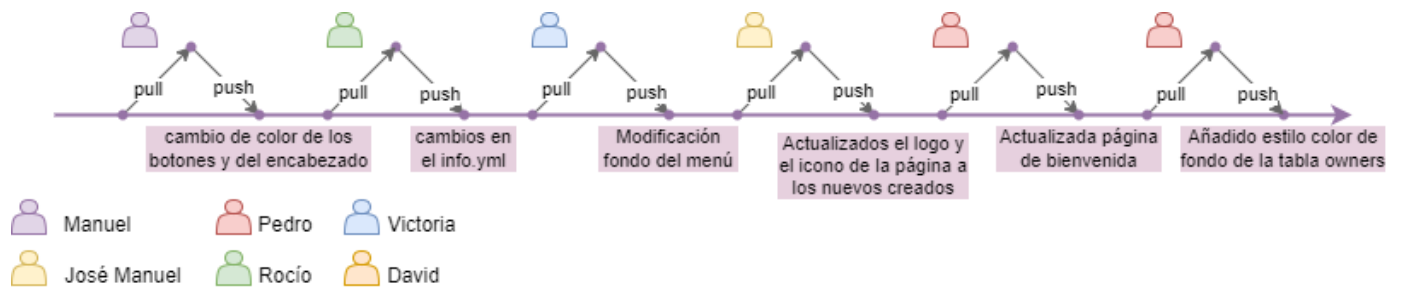


La imagen muestra los **commits** realizados en **orden cronológico** de izquierda a derecha. Los commits que presentaron conflictos se representan encuadrados en rojo mientras que los que no presentaron conflictos se representan encuadrados en morado.

Todos los commits representados en la imagen anterior se realizaron el 15 de febrero de 2023, podemos verlos también en GitHub, ordenados en este caso de forma ascendente:

Commits on Feb 15, 2023		
Resuelto conflicto pom	davidzarandieta committed 2 days ago	e4f05e6
Merge branch 'main' of https://github.com/gii-is-psg2-2223-g7-72	davidzarandieta committed 2 days ago	187c95e
Añadido mi info al pom y cambio imagen mascota	davidzarandieta committed 2 days ago	3e619cd
pom modificado y conflictos resueltos	RukisRo committed 2 days ago	73bf1ed
Resueltos los conflictos y añadido información personal al pom	pgmarc committed 2 days ago	4c8ac61
cambios en el pom	RukisRo committed 2 days ago	0a1a191
Añadida información al pom y resueltos los cambios	VictoriaCRD committed 2 days ago	f6267f3
Añadida información al pom	VictoriaCRD committed 2 days ago	d7a1522
Añadiendo información personal al pom.xml	pgmarc committed 2 days ago	9b710d8
Añadidos datos pom y corregido conflictos	josmorgue3 committed 2 days ago	9027a1e
cambios del pom	josmorgue3 committed 2 days ago	abe3ea4
Modificado el archivo pom.xml con mis datos personales	manvazmar3 committed 2 days ago	0529be8

Además para completar las tareas del apartado A2.5 se realizaron algunas modificaciones más, esta vez sin forzar conflictos y realizando un **PULL** antes de comenzar a realizar cambios en el código local:

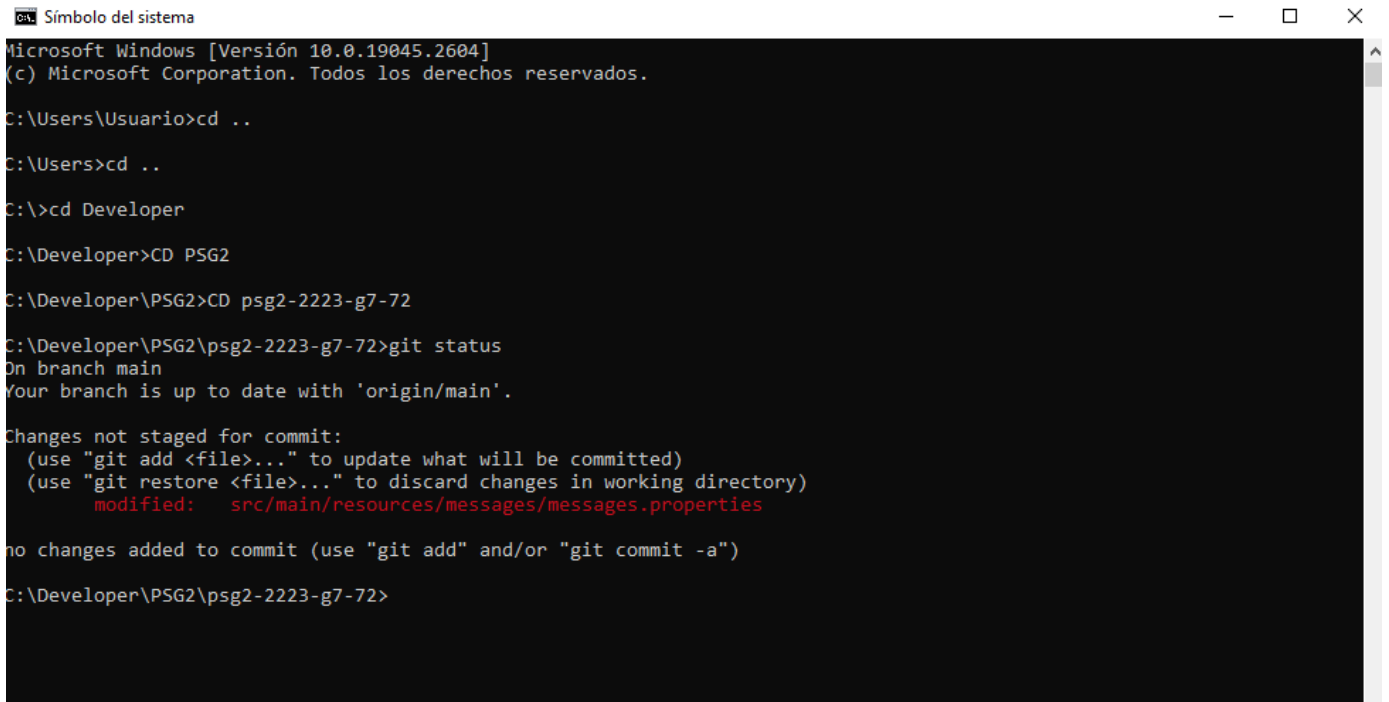


# Resolución de conflictos

A continuación se detalla los pasos seguidos por el equipo para generar conflictos de código:

1. Tras haber realizado un cambio en el código local sin haber hecho **PULL**, realizamos la siguiente secuencia de comandos:

- `git status`



```
Símbolo del sistema
Microsoft Windows [Versión 10.0.19045.2604]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Usuario>cd ..
C:\Users>cd ..
C:\>cd Developer
C:\Developer>CD PSG2
C:\Developer\PSG2>CD psg2-2223-g7-72
C:\Developer\PSG2\psg2-2223-g7-72>git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   src/main/resources/messages/messages.properties

no changes added to commit (use "git add" and/or "git commit -a")

C:\Developer\PSG2\psg2-2223-g7-72>
```

- `git add .`
- `git commit -m "mensaje"`
- `git push origin main`

Tras hacer el **PUSH** surge un **conflicto** en el código ya que una de las líneas que se ha modificado en local ya había sido modificada por anterioridad por otro desarrollador, pero estos cambios no habían sido incluidos en el código local.

```
ca Símbolo del sistema
modified: src/main/resources/messages/messages.properties

no changes added to commit (use "git add" and/or "git commit -a")

C:\Developer\PSG2\psg2-2223-g7-72>git add .

C:\Developer\PSG2\psg2-2223-g7-72>git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   src/main/resources/messages/messages.properties

C:\Developer\PSG2\psg2-2223-g7-72>git commit -m "modificado el messages.properties para forzar un conflicto"
[main 69f3c66] modificado el messages.properties para forzar un conflicto
1 file changed, 1 insertion(+), 1 deletion(-)

C:\Developer\PSG2\psg2-2223-g7-72>git push
To https://github.com/gii-is-psg2/psg2-2223-g7-72.git
! [rejected]        main -> main (fetch first)
error: failed to push some refs to 'https://github.com/gii-is-psg2/psg2-2223-g7-72.git'
hint: Updates were rejected because the remote contains work that you do
hint: not have locally. This is usually caused by another repository pushing
hint: to the same ref. You may want to first integrate the remote changes
hint: (e.g., 'git pull ...') before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.

C:\Developer\PSG2\psg2-2223-g7-72>
```

- `git pull origin main`
- `code src/main/resources/messages/messages.properties`

Con este último comando se abre Visual Studio Code, donde se compara el código que genera el conflicto. En la parte superior de la ventana se puede aceptar el cambio que corresponda:

- Aceptar cambio actual
- Aceptar cambio entrante
- Aceptar ambos cambios
- Comparar cambios

En este caso se selecciona *Aceptar cambio entrante* para dejar el código como estaba antes de introducir la modificación:

```
messages.properties X
C: > Developer > PSG2 > psg2-2223-g7-72 > src > main > resources > messages > messages.properties
Aceptar cambio actual | Aceptar cambio entrante | Aceptar ambos cambios | Comparar cambios
1 <<<<<< HEAD (Cambio actual)
2 welcome=Mensaje para generar conflicto
3 =====
4 welcome=Welcome to the PSG2-2223-G7-72 Petclinic
5 >>>>>> 1f6db974be7645496bb359c7980cf5ea982d48af (Cambio entrante)
6 required=is required
7 notFound=has not been found
8 duplicate=is already in use
9 nonNumeric=must be all numeric
10 duplicateFormSubmission=Duplicate form submission is not allowed
11 typeMismatch.date=invalid date
12 typeMismatch.birthDate=invalid date
```

El código vuelve a ser correcto y ya no tiene conflictos:



C: > Developer > PSG2 > psg2-2223-g7-72 > src > main > resources > messages > messages.properties

```
1 welcome=Welcome to the PSG2-2223-G7-72 Petclinic
2 required=is required
3 notFound=has not been found
4 duplicate=is already in use
5 nonNumeric=must be all numeric
6 duplicateFormSubmission=Duplicate form submission is not allowed
7 typeMismatch.date=invalid date
8 typeMismatch.birthDate=invalid date
9
```

2. Se vuelve a realizar la siguiente secuencia de comandos para finalmente añadir los cambios al repositorio sin que se generen conflictos:

- `git add .`
- `git commit -m "mensaje"`
- `git push`

```
Unmerged paths:
(use "git add <file>..." to mark resolution)
    both modified:   src/main/resources/messages/messages.properties

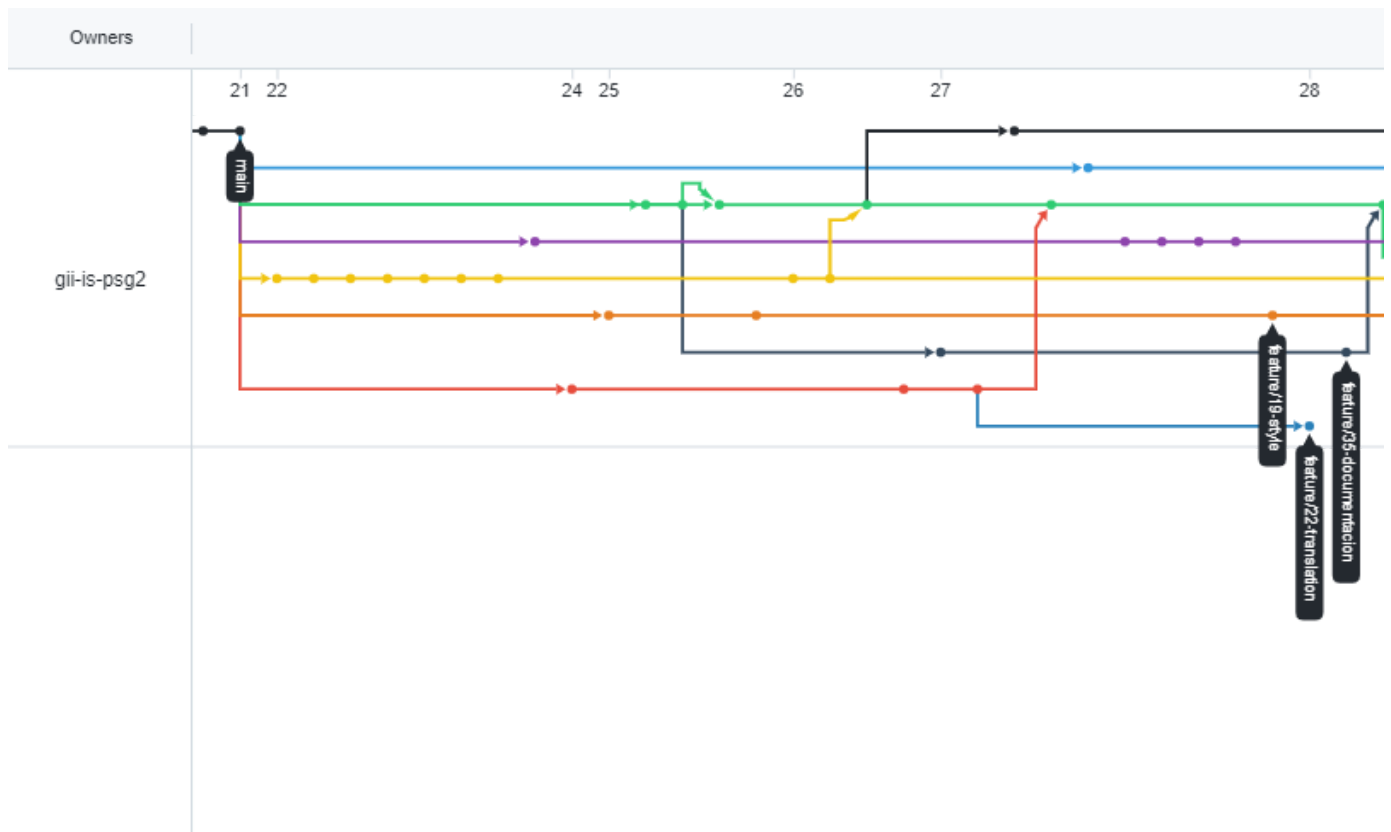
C:\Developer\PSG2\psg2-2223-g7-72>git add .
C:\Developer\PSG2\psg2-2223-g7-72>git commit -m "resuelto el conflicto de messages.properties"
[main 6a554d9] resuelto el conflicto de messages.properties
C:\Developer\PSG2\psg2-2223-g7-72>git push
Enumerating objects: 20, done.
Counting objects: 100% (20/20), done.
Delta compression using up to 4 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (8/8), 809 bytes | 404.00 KiB/s, done.
Total 8 (delta 5), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (5/5), completed with 5 local objects.
To https://github.com/gii-is-psg2/psg2-2223-g7-72.git
 1f6db97..6a554d9  main -> main
C:\Developer\PSG2\psg2-2223-g7-72>
```

Esta prueba se ha realizado para documentar la forma en que el equipo ha resuelto los conflictos del [apartado anterior](#), habiendo forzado el conflicto para tomar capturas del proceso. Los commits realizados durante la prueba no se van a añadir al historial de cambios.

# Historial de cambios A2.8 y A2.9

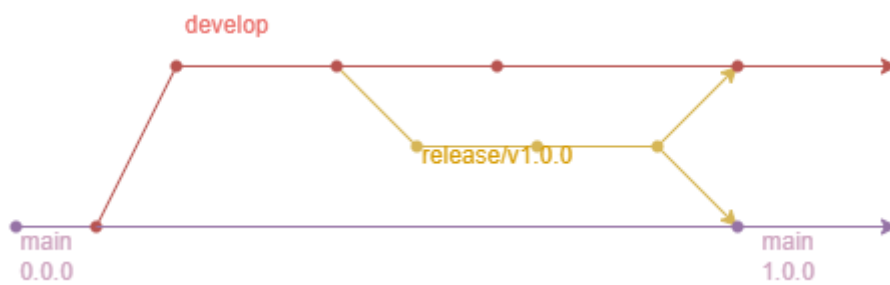
Para las tareas A2.8 y A2.9 se va a usar la metodología de git-flow. Para ello, se crea una rama `feature` para cada nueva funcionalidad de la aplicación que se fusiona con la rama `develop` (previamente se crea una PULL-REQUEST). Finalmente, una vez todas las funcionalidades están integradas en la rama `develop`, se crea a partir de ella una rama `release`. En esta rama se corrigen y añaden pequeños cambios. La rama `release` se fusiona tanto con la rama `develop` como con la rama `main`, obteniendo así la primera versión productiva de nuestro proyecto.

En las siguientes imágenes se muestra el flujo de trabajo de nuestro proyecto. De la rama `main` sacamos la rama `develop` (verde) y a partir de ella todas las demás ramas `feature`, finalmente todas las features terminan fusionadas con la rama `develop`:





Por último de la rama develop sacamos una rama release que tras realizar varios cambios en ella, fusionamos con main y con develop:

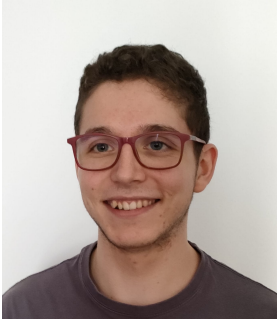






# Ventajas de Git-flow vs Trunk-based

- No se dan situaciones de conflicto o es más difícil que ocurran
- Es más fácil llevar un control de los cambios que se han realizado en el código y más fácil detectar quién los ha realizado.
- Más fácil deshacer cambios

En cuanto a las desventajas podemos mencionar que la curva de aprendizaje es mayor y podemos al principio tener fallos sobre de donde salen las ramas y hacia donde se realiza el merge.

# Anexo 1: Nuestro equipo

<p>Pedro González Marcos</p>	 A portrait of a young man with short brown hair and red-rimmed glasses, smiling. He is wearing a dark purple t-shirt.
<p>Rocío Lopez Moyano</p>	 A portrait of a young woman with long brown hair, smiling. She is wearing a blue graduation gown over a white shirt and a black tie.
<p>José Manuel Moreno Guerrero</p>	 A portrait of a young man with short dark hair, looking directly at the camera. He is wearing a black t-shirt.
<p>Victoria del Carmen Ruiz Delgado</p>	 A portrait of a young woman with long red hair, looking slightly to the side. She is wearing a green top and a necklace. The background is a colorful, abstract pattern.
<p>Manuel Vázquez Martín</p>	 A portrait of a young man with short dark hair, looking directly at the camera. He is wearing a blue graduation gown over a white shirt and a black tie. A circular logo is visible on the left side of his chest.

David Zarandieta Ortiz

