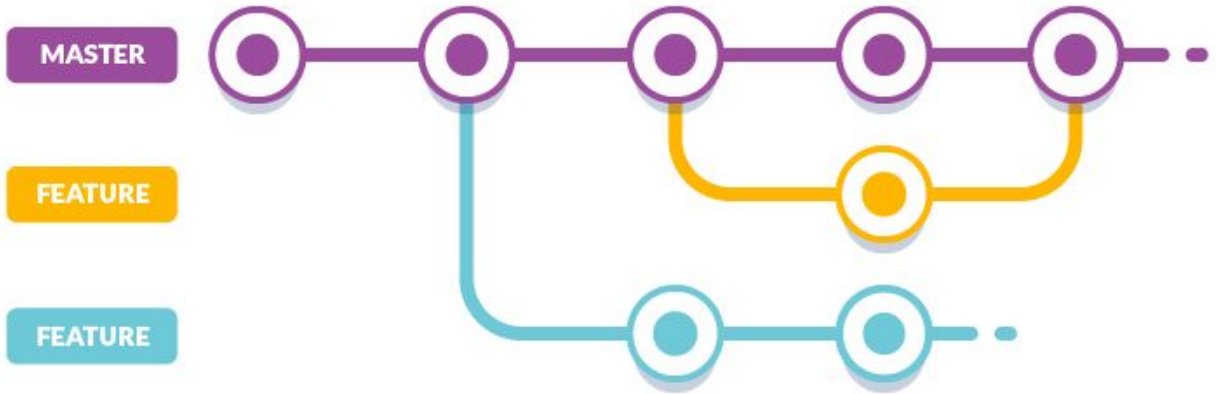
Branches, (Ramas)

IT BOARDING

BOOTCAMP

// ¿Qué son las Branches? (Ramas)

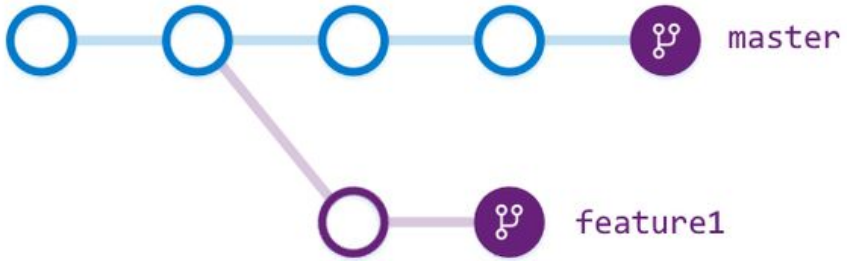
- Una rama en Git es un "espacio" dentro del repositorio donde se almacenan nuestros archivos. Por defecto en Git, la rama principal se llama master o main.



// ¿Para qué se usan las Branches?

- En Git las Ramas son espacios o entornos independientes para que un **desarrollador** (o un equipo) pueda usar y trabajar sobre un mismo **proyecto,** sin cometer errores o borrar el conjunto de archivos originales del proyecto.

- Esto permite dar flexibilidad para desarrollar el proyecto de manera más organizada, donde solo se incorpora una rama a la rama principal cuando se esté seguro que lo desarrollado funciona correctamente.



¡Veamos un EJEMPLO!

IT BOARDING

BOOTCAMP

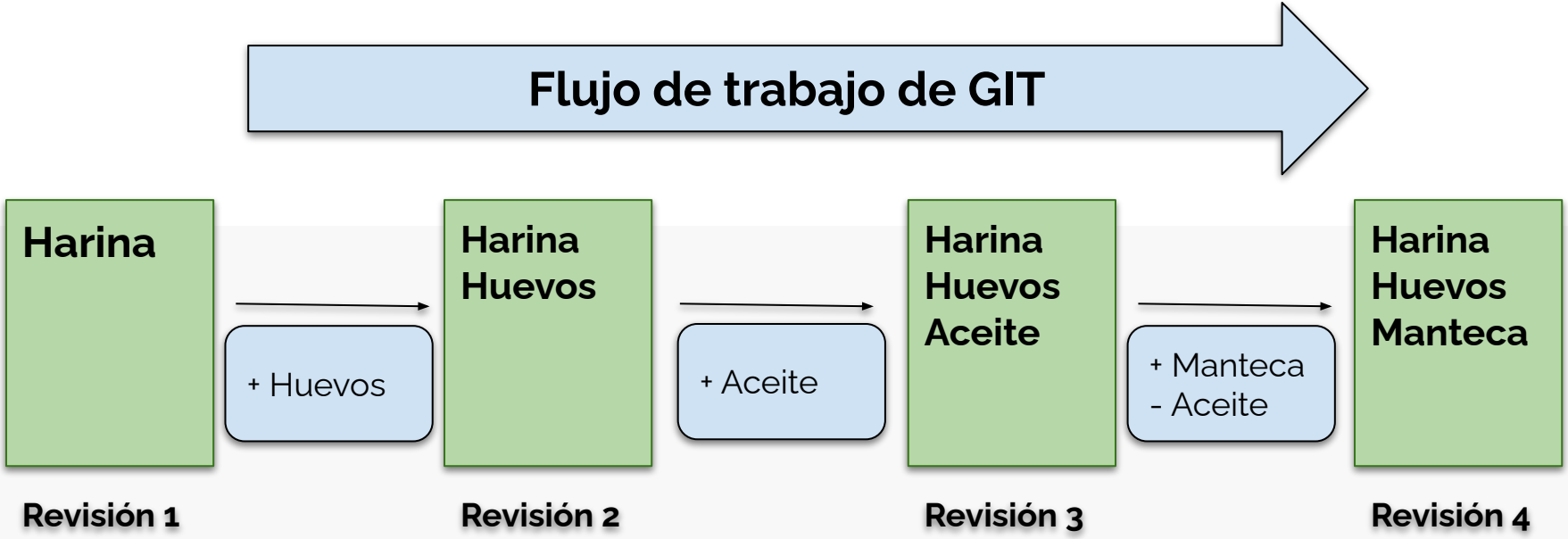
// Supongamos que tenemos una receta...

- Como no nos acordamos bien los pasos de la misma, comenzamos por ir probando distintos ingredientes. Por cada ingrediente que agreguemos vamos a crear una rama.

- En caso de que nos salga bien seguimos avanzando, y en caso de que nos equivoquemos, podemos volver a la rama anterior y seguir trabajando sobre esa.

- Una vez que probamos y encontramos la receta final, unificamos esa rama a nuestra “principal” (main o master).

// Nuestras ramas quedarían así:



¿Cómo creamos branches?

IT BOARDING

BOOTCAMP

// Comandos para Branches

// Conocer en qué rama estamos parados:

```
>_ git branch
```

// Crear una nueva rama con el nombre que queramos:

```
>_ git branch nombreBranch
```

// Crear una nueva rama a partir de una rama existente:

```
>_ git checkout -b branchNueva branchExistente
```

// Comandos para Branches

// Cambiar de rama:

```
>_  git checkout nombreBranch
```

// Eliminar una rama:

```
>_  git branch -d nombreBranch
```

// Copiar una rama:

```
>_  git branch -c nombreBranch nombreCopia
```

// Comandos para Branches

// Ver diferencias entre dos ramas:

```
>_  git diff branch1 branch2
```

// Unificar (“mergear”) ramas:

```
>_  git merge branchOrigen branchDestino
```

// Comandos para resolver conflictos

En muchas ocasiones al trabajar con Branches existen conflictos, para resolverlos es posible seguir los siguientes pasos:

// Ubicarse en la branch donde se desea que queden ambas ramas sin conflicto

```
>_  git checkout branch2
```

// Traer los datos de la banch que se desea mergear

```
>_  git pull origin branch1
```

Gracias.

IT BOARDING

BOOTCAMP