

¿Qué es Git?

Git es un sistema de control de versiones distribuido.

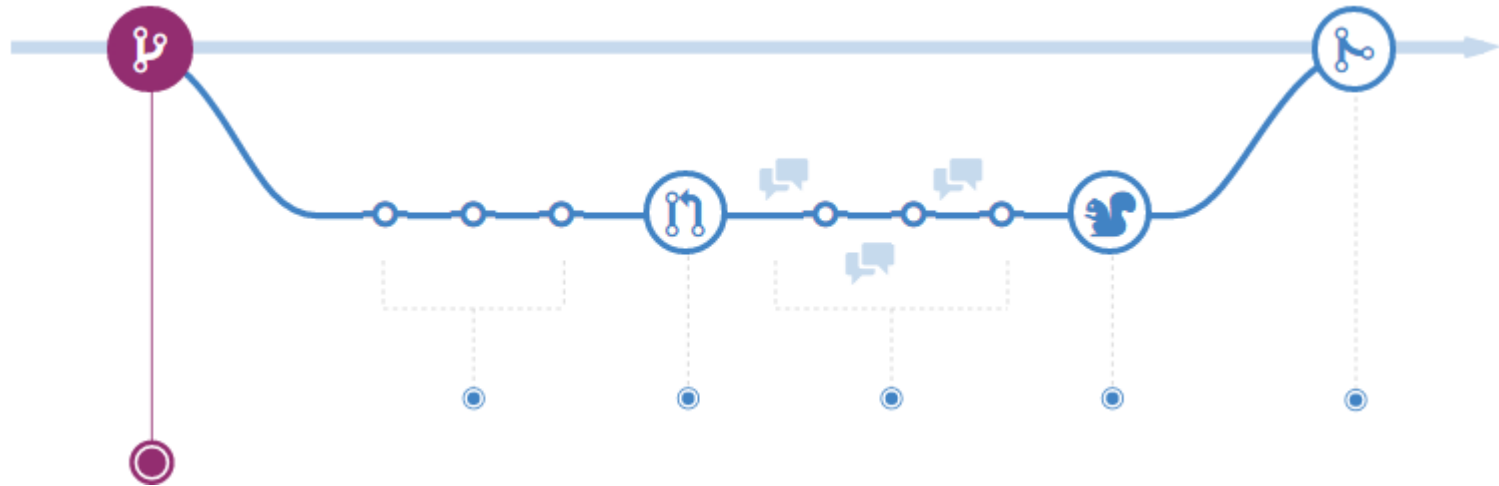
Permite a un grupo de personas trabajar en los mismos documentos (generalmente código) al mismo tiempo, sin que ninguno solape o estorbe la actividad del otro.

GitHub es una plataforma de hosting de código para el control de versiones y colaboración.

No es necesario codificar las acciones vinculadas al flujo de trabajo/colaboración.

Flujo de trabajo de *GitHub*

Flujo de trabajo de GitHub



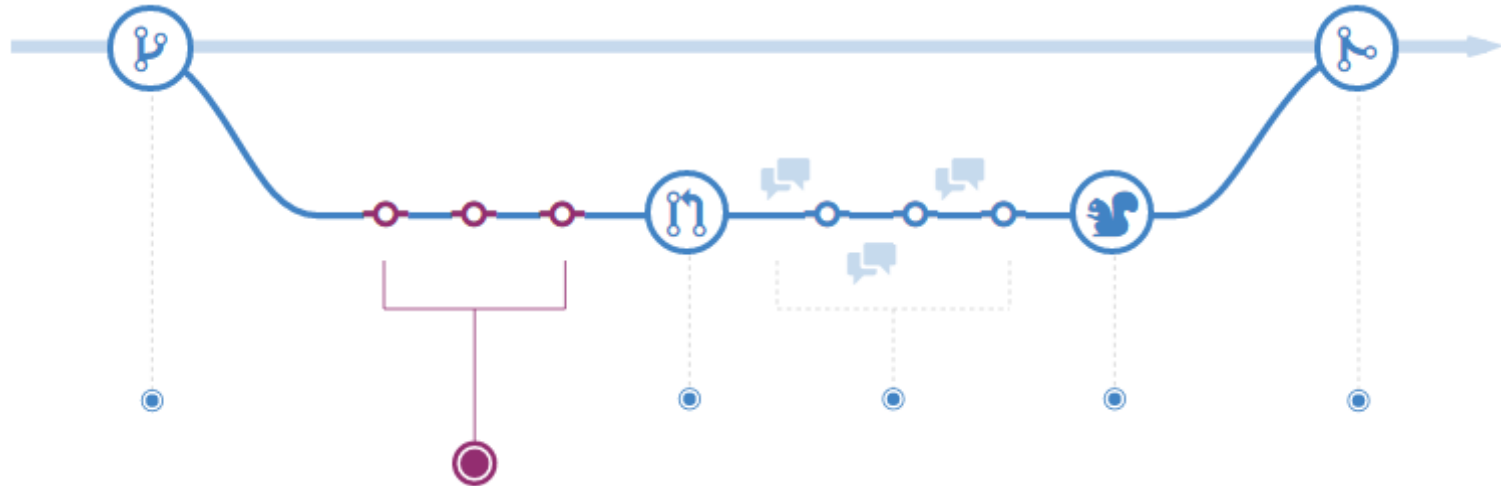
Crear una rama (*branch*)

Ayudan a manejar el flujo de trabajo en un proyecto, donde cada rama representa diferentes características e ideas en progreso.

Con cada rama se crea un entorno donde se pueden probar las nuevas ideas.

Los cambios no afectan a la rama *master*.

Flujo de trabajo de GitHub

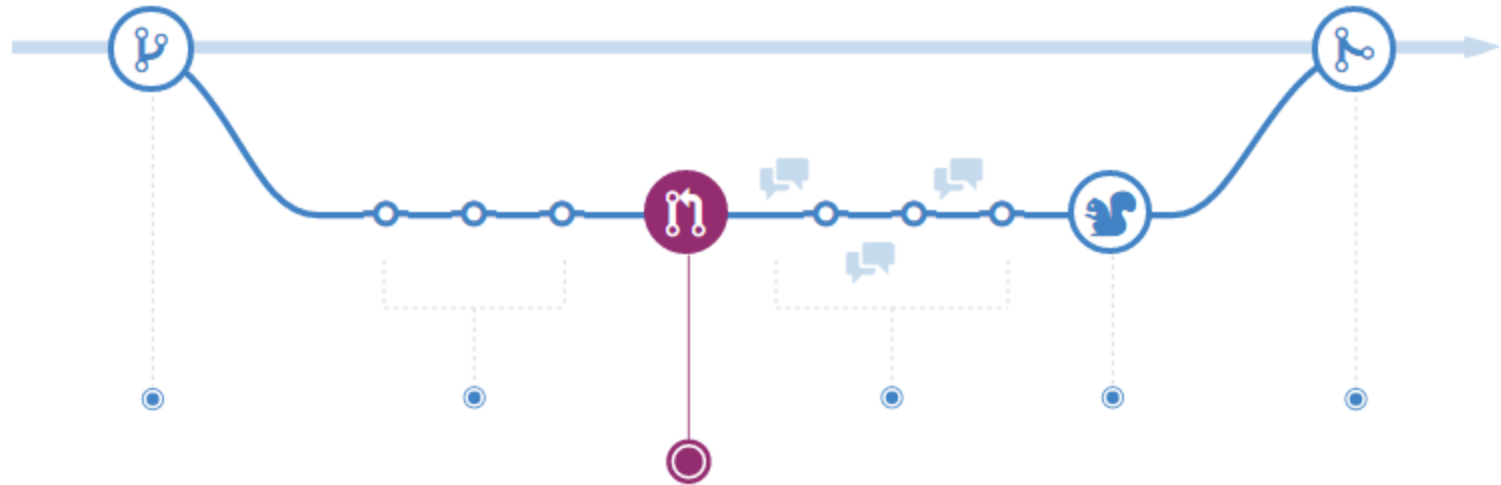


Añadir *commits*

Cada vez que se añade, edita o elimina un archivo, se está haciendo un **commit** y son añadidas a tu rama, y mantiene un **historial**.

Cada *commit* tiene un **mensaje** asociado para explicar el porqué de los cambios. Además, cada *commit* es considerado una unidad separada de cambio.

Flujo de trabajo de GitHub



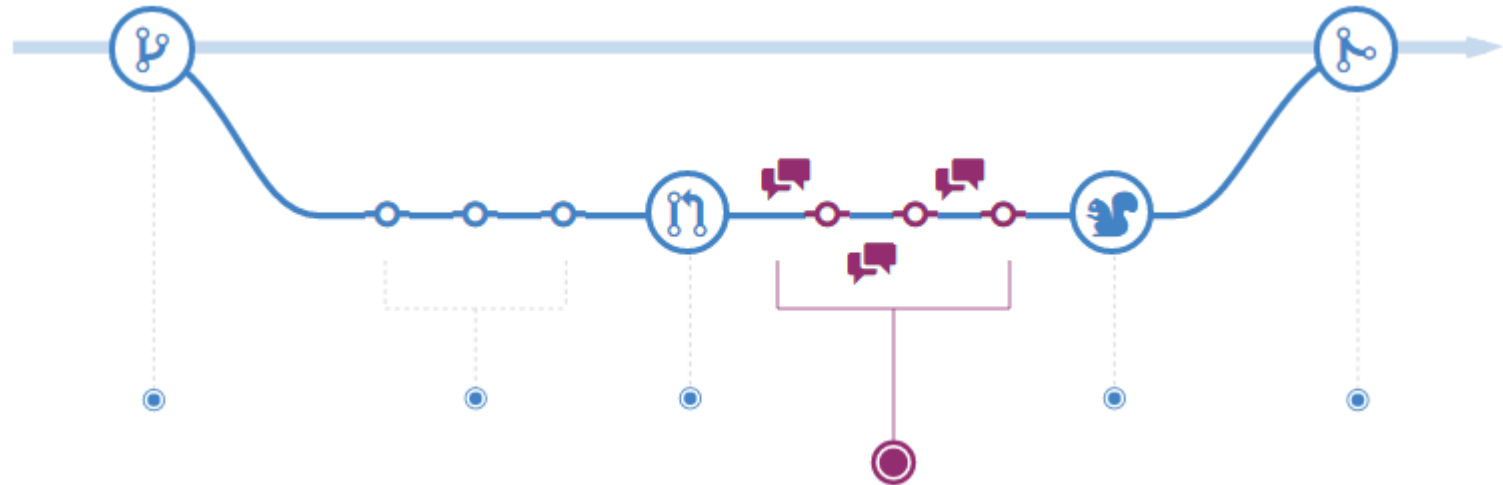
Open a *Pull Request*

Pull requests inician la discusión acerca los *commits*.

Estos están altamente integradas con el repositorio GitHub, donde cada uno puede ver exactamente qué cambios se integrarían si aceptaran un *request*. Se puede abrir un pull request en cualquier momento durante el proceso de desarrollo.

Se pueden usar *@mention* en el mensaje del pull request.

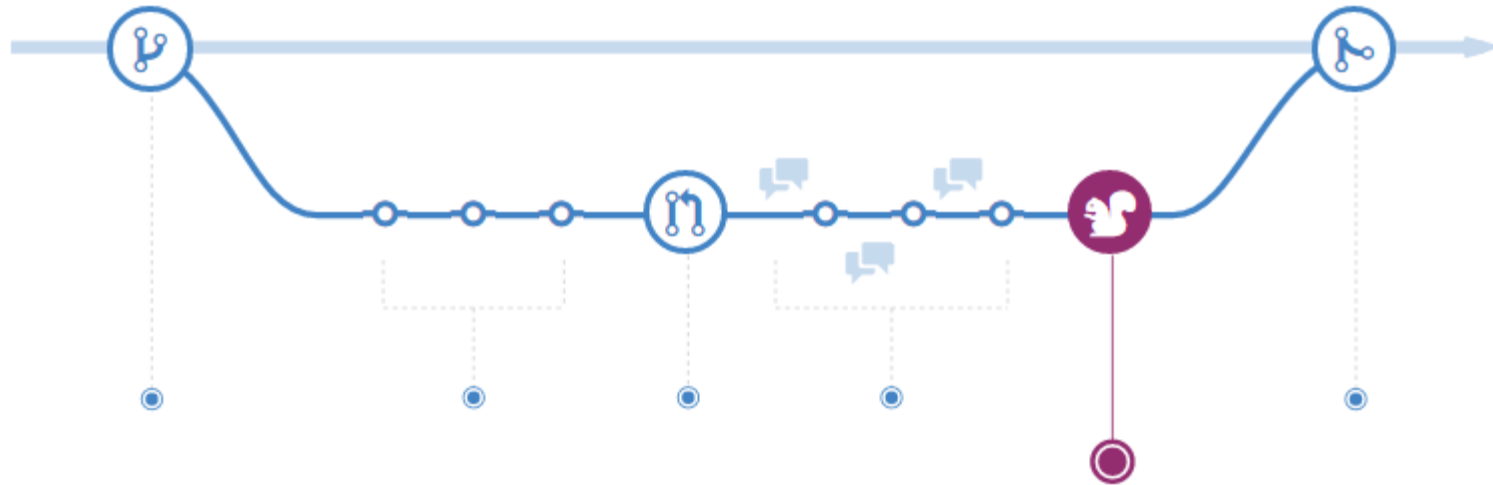
Flujo de trabajo de GitHub



Discusión y revisión

Luego que un *pull request* fue abierto, los otros miembros del equipo pueden hacer preguntas o comentarios sobre los cambios.

Flujo de trabajo de GitHub

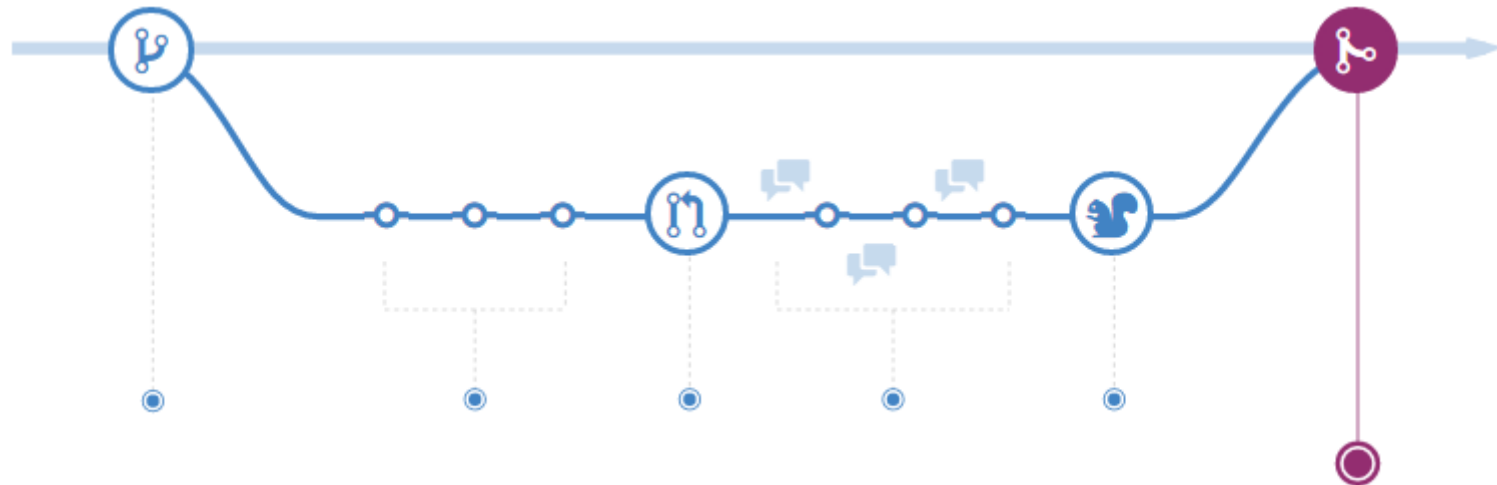


Deploy

Luego que un pull request ha sido revisado y la rama pasa ciertas pruebas propias, se puede hacer un deploy de los cambios para verificarlos en la producción.

Si la rama causa algún inconveniente se puede hacer un roll back del deploy.

Flujo de trabajo de GitHub



Merge

Luego que los cambios han sido verificados en la producción, se integran a la rama ***master***.

Los *pull requests* conservan un historial de los cambios realizados.

Repositorios *Git*Hub

Repositorios

Repositorio público (gratis):

Cualquiera puede verlo, pero solo uno mismo elige quien puede hacer *commits* (cambios) sobre él.

Repositorio privado (suscripción paga):

Por default solo uno puede verlo y elegir quien puede hacer *commits* especificando colaboradores.

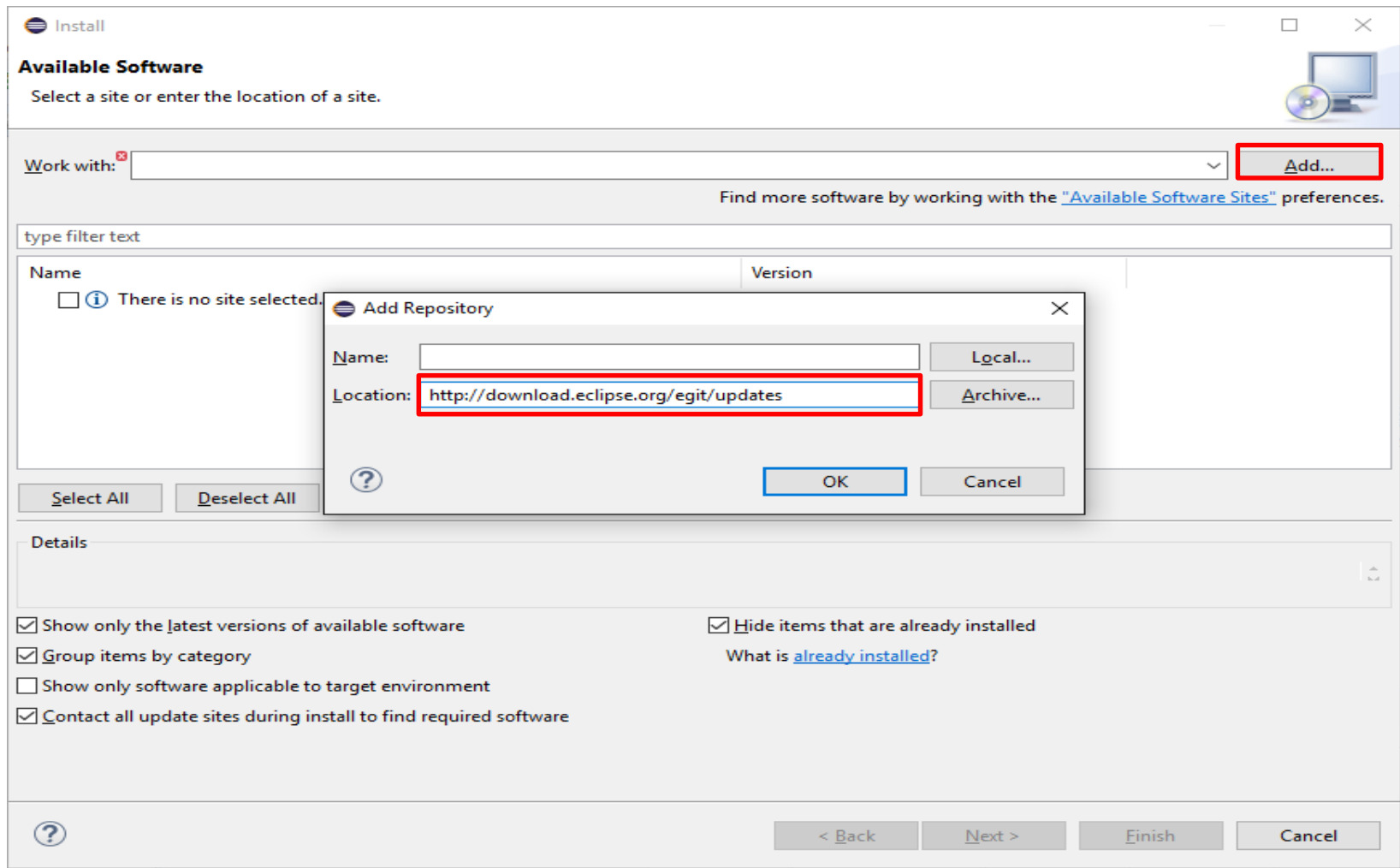
Repositorio local: es el alojado en tu computadora. Uno puede trabajar en este repositorio sin estar conectado a Internet.

Repositorio remote: (<http://GitHub.com>) Se pueden enviar links de los repositories a las personas para compartir código o colaborar en el desarrollo de un proyecto.

Vinculación Eclipse – Git

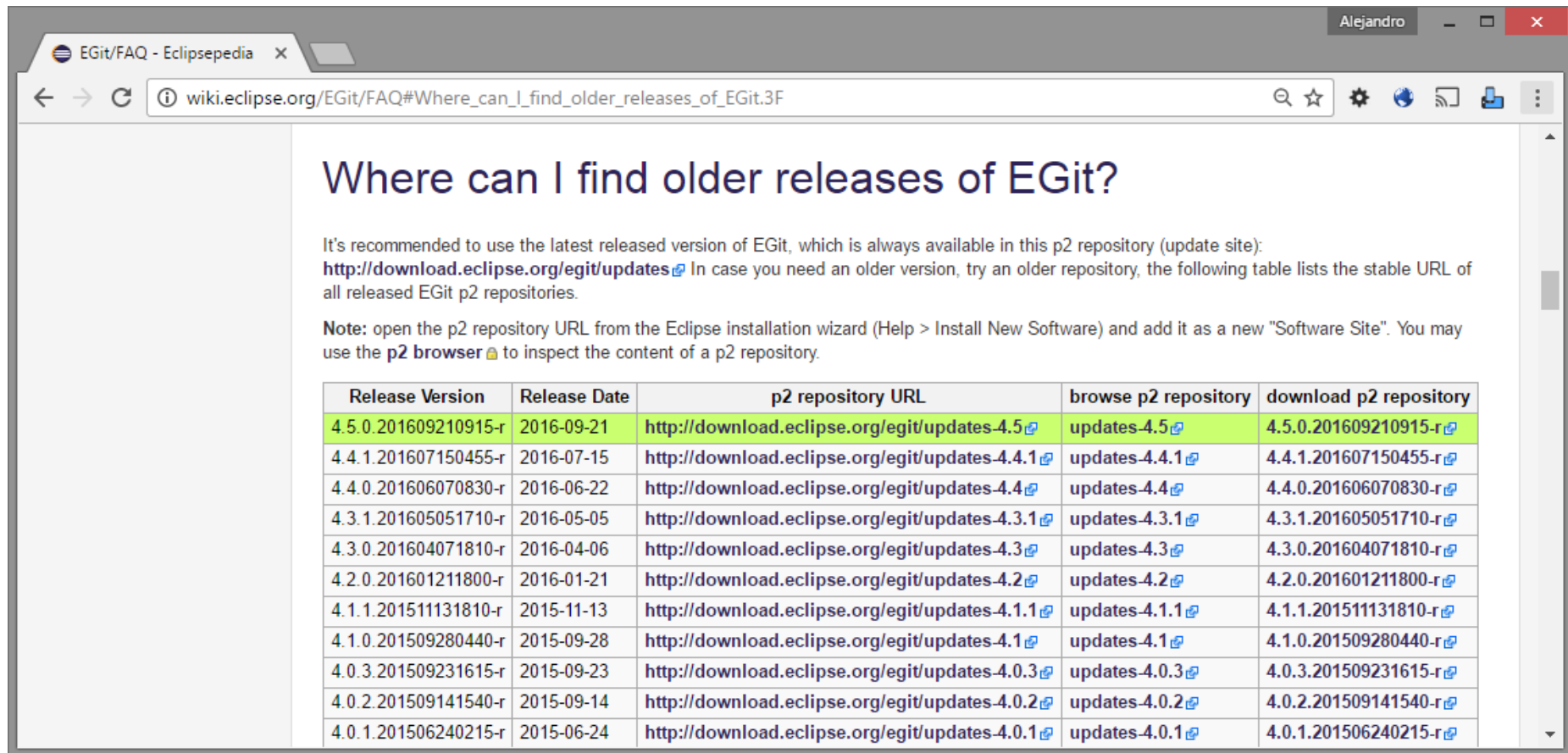
Eclipse – Git

En Eclipse, ir a **Help > Install new software**.



Eclipse – Git: Incompatibilidades con versiones antiguas del IDE Eclipse

En Eclipse, ir a **Help > Install new software**.



EGit/FAQ - Eclipsepedia

wiki.eclipse.org/EGit/FAQ#Where_can_I_find_older_releases_of_EGit.3F

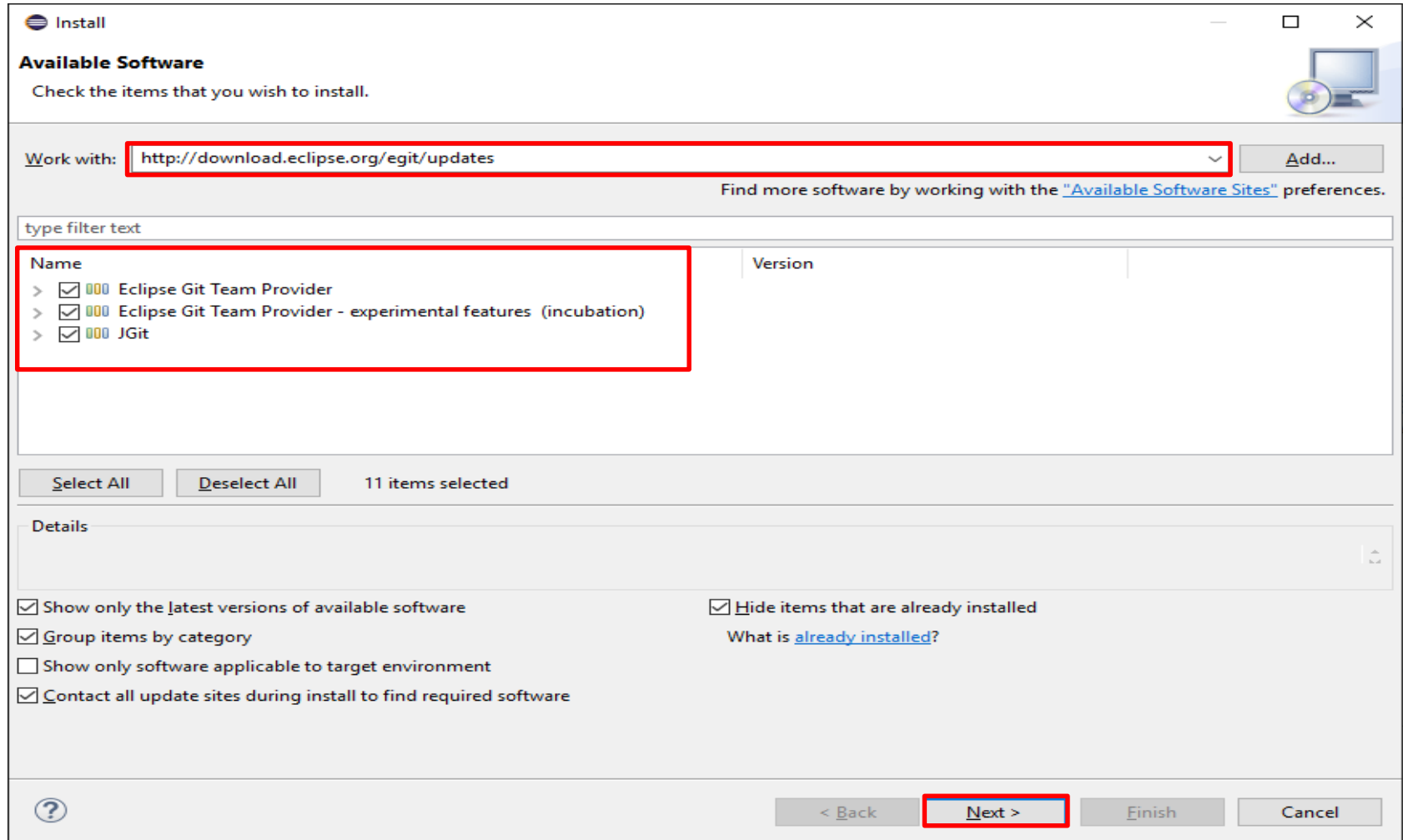
Where can I find older releases of EGIt?

It's recommended to use the latest released version of EGIt, which is always available in this p2 repository (update site):
<http://download.eclipse.org/egit/updates> In case you need an older version, try an older repository, the following table lists the stable URL of all released EGIt p2 repositories.

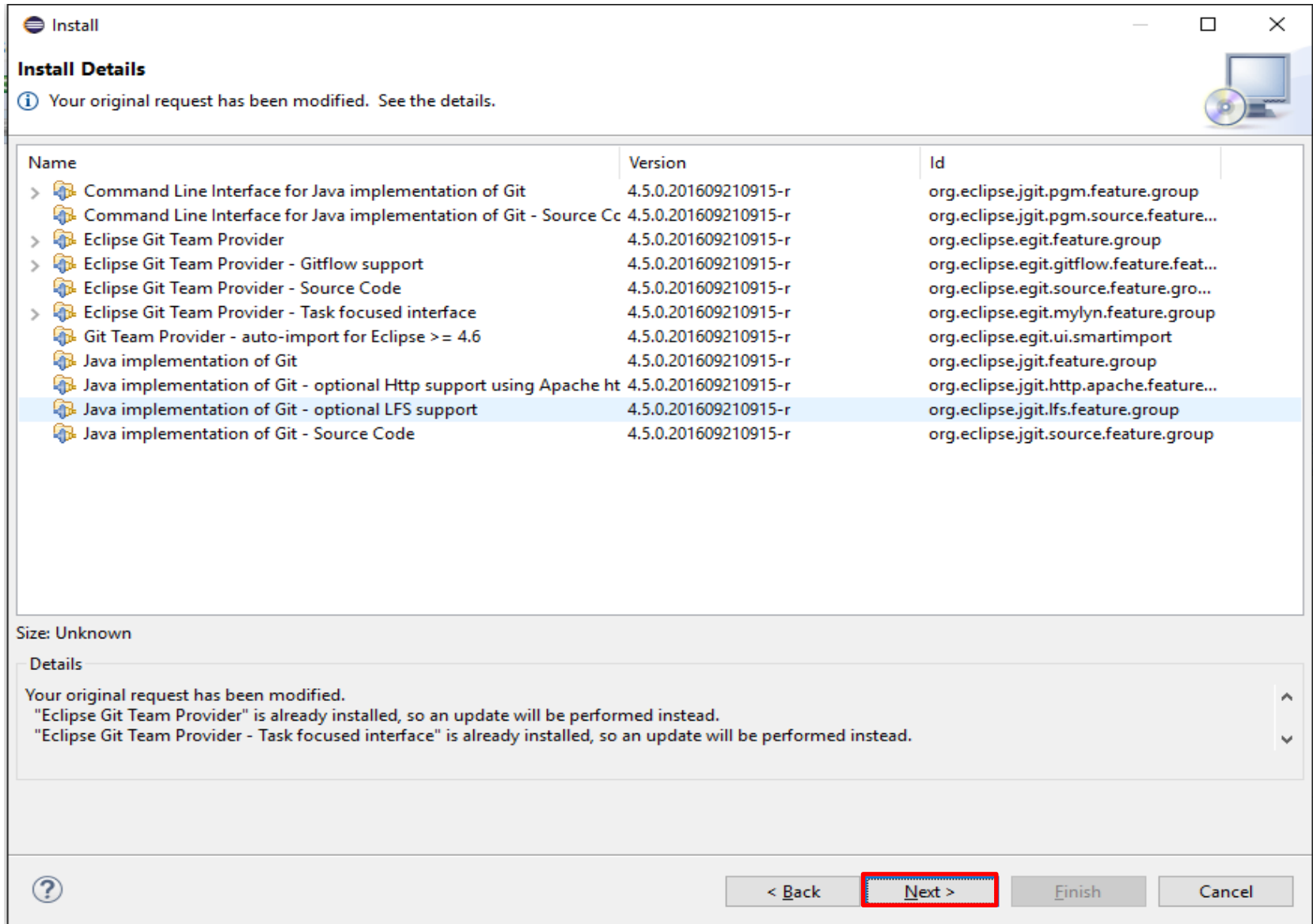
Note: open the p2 repository URL from the Eclipse installation wizard (Help > Install New Software) and add it as a new "Software Site". You may use the **p2 browser** to inspect the content of a p2 repository.

Release Version	Release Date	p2 repository URL	browse p2 repository	download p2 repository
4.5.0.201609210915-r	2016-09-21	http://download.eclipse.org/egit/updates-4.5	updates-4.5	4.5.0.201609210915-r
4.4.1.201607150455-r	2016-07-15	http://download.eclipse.org/egit/updates-4.4.1	updates-4.4.1	4.4.1.201607150455-r
4.4.0.201606070830-r	2016-06-22	http://download.eclipse.org/egit/updates-4.4	updates-4.4	4.4.0.201606070830-r
4.3.1.201605051710-r	2016-05-05	http://download.eclipse.org/egit/updates-4.3.1	updates-4.3.1	4.3.1.201605051710-r
4.3.0.201604071810-r	2016-04-06	http://download.eclipse.org/egit/updates-4.3	updates-4.3	4.3.0.201604071810-r
4.2.0.201601211800-r	2016-01-21	http://download.eclipse.org/egit/updates-4.2	updates-4.2	4.2.0.201601211800-r
4.1.1.201511131810-r	2015-11-13	http://download.eclipse.org/egit/updates-4.1.1	updates-4.1.1	4.1.1.201511131810-r
4.1.0.201509280440-r	2015-09-28	http://download.eclipse.org/egit/updates-4.1	updates-4.1	4.1.0.201509280440-r
4.0.3.201509231615-r	2015-09-23	http://download.eclipse.org/egit/updates-4.0.3	updates-4.0.3	4.0.3.201509231615-r
4.0.2.201509141540-r	2015-09-14	http://download.eclipse.org/egit/updates-4.0.2	updates-4.0.2	4.0.2.201509141540-r
4.0.1.201506240215-r	2015-06-24	http://download.eclipse.org/egit/updates-4.0.1	updates-4.0.1	4.0.1.201506240215-r

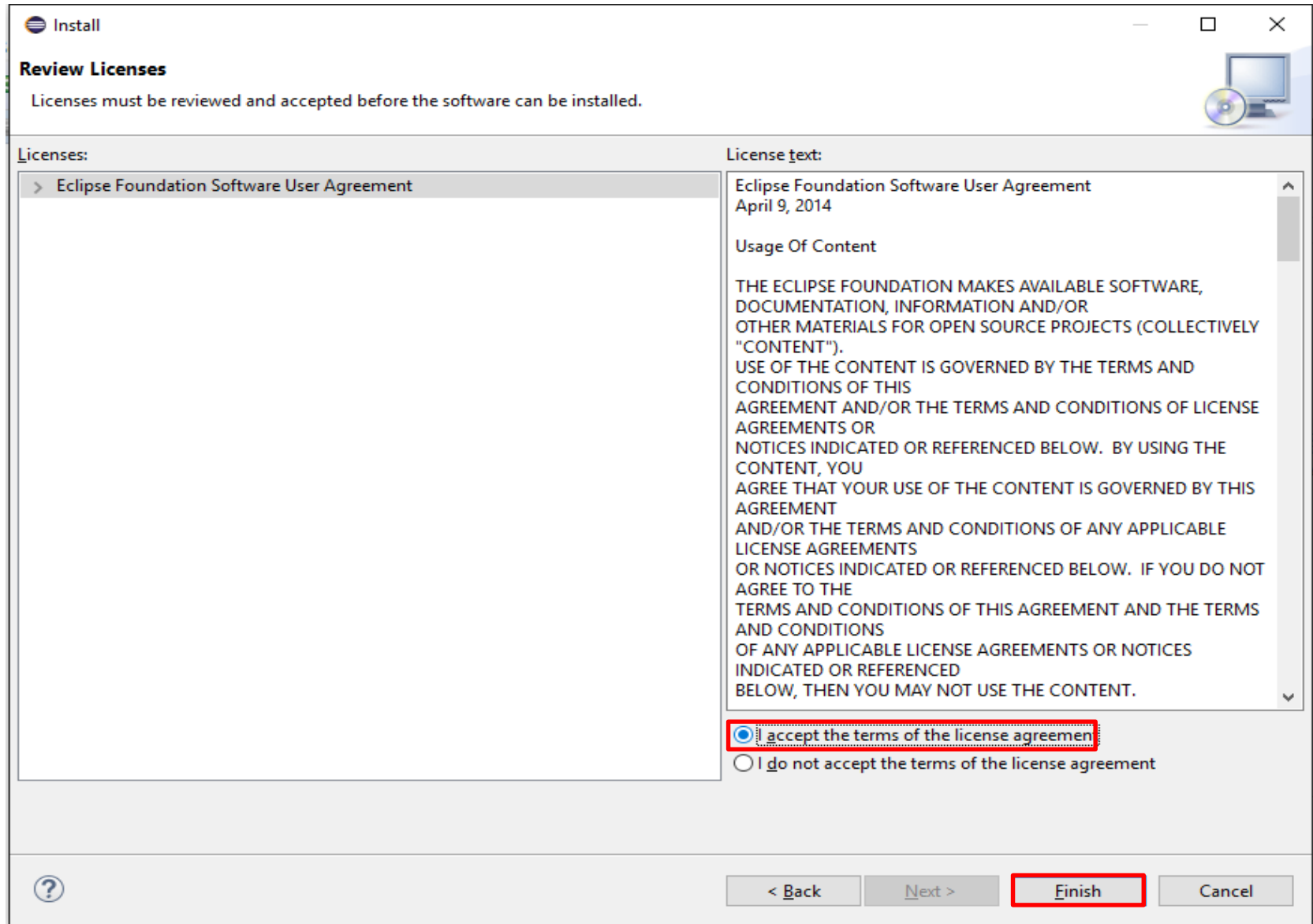
Eclipse – Git



Eclipse – Git

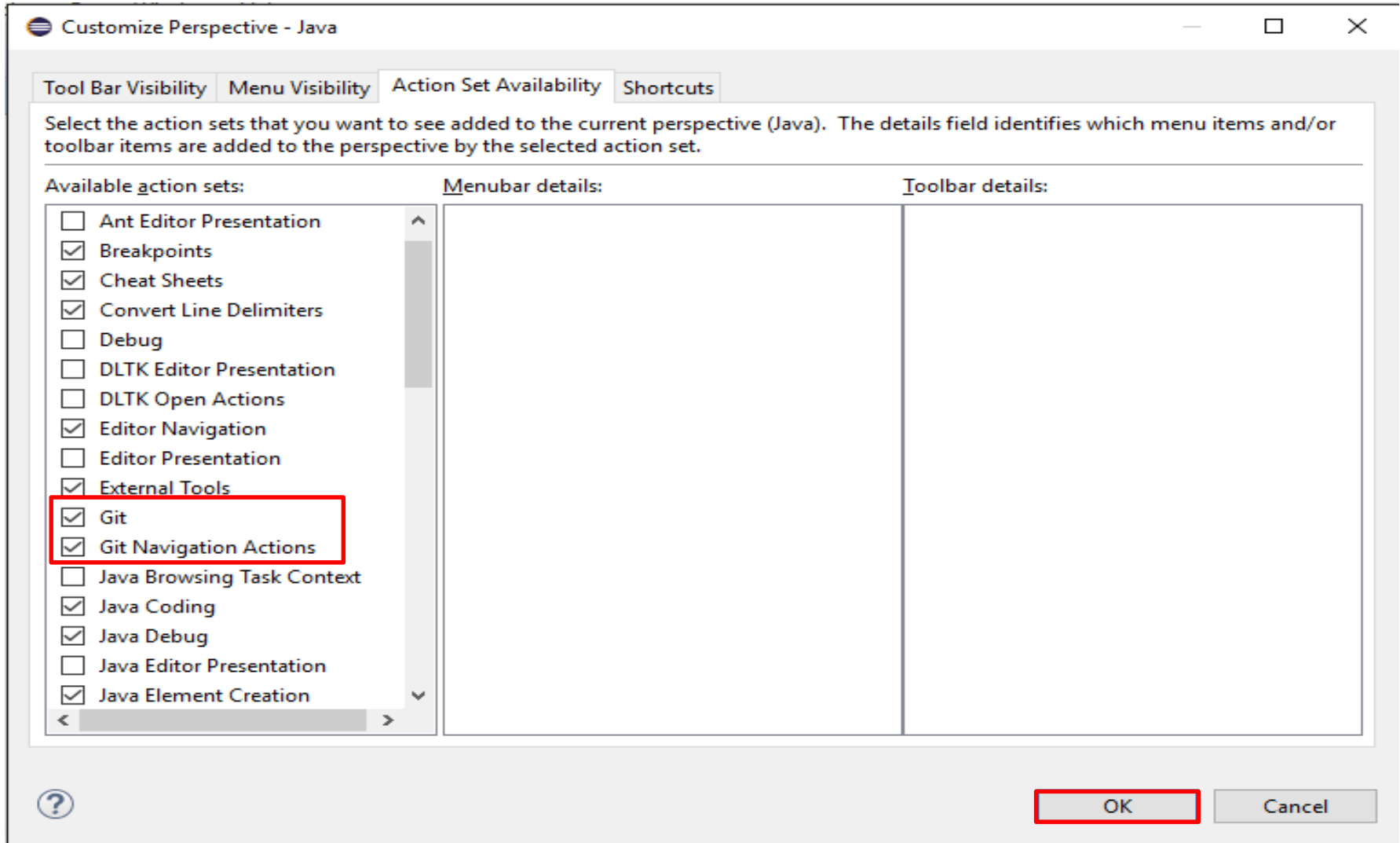


Eclipse – Git

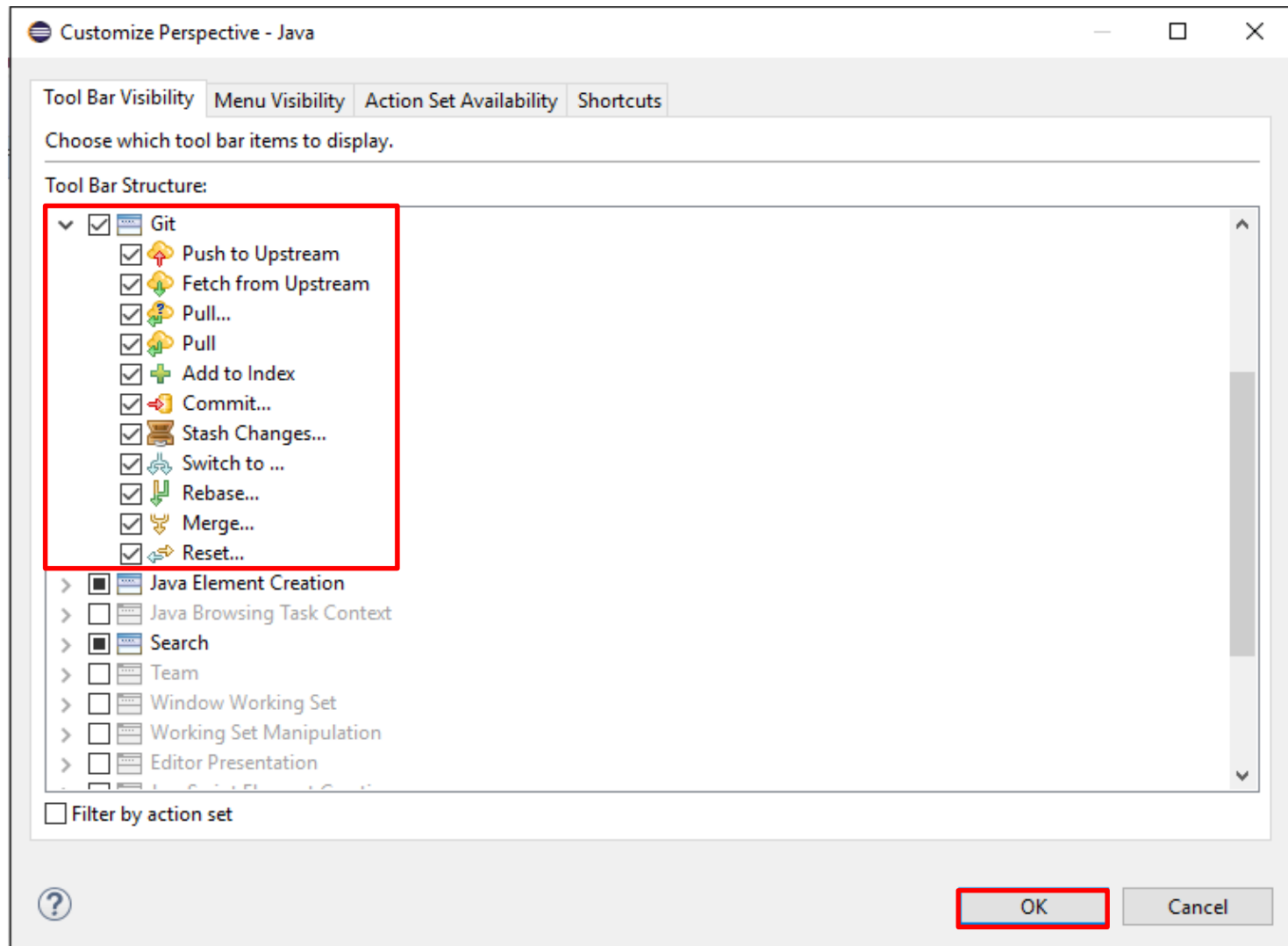


Configuración de la barra de herramientas en Eclipse

Ir a **Window > Perspective > Customize perspective...**

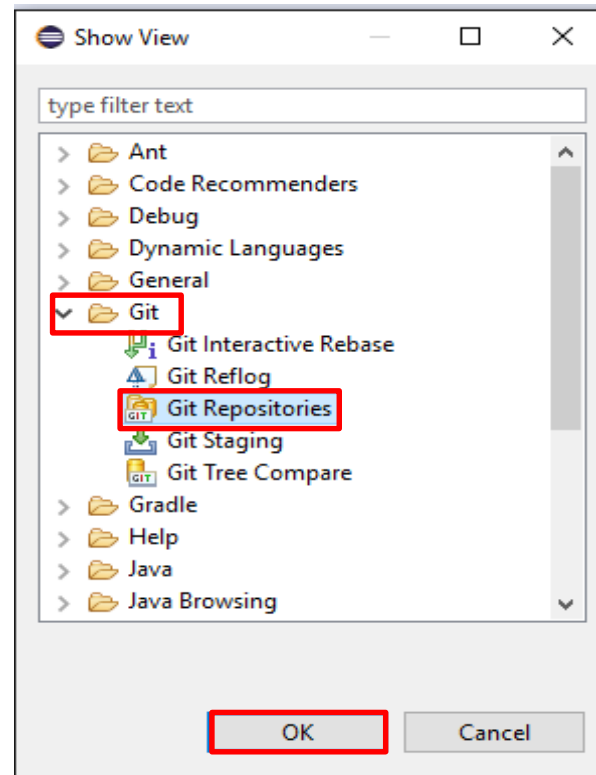


Configuración de la barra de herramientas en Eclipse



Configuración de la barra de herramientas en Eclipse

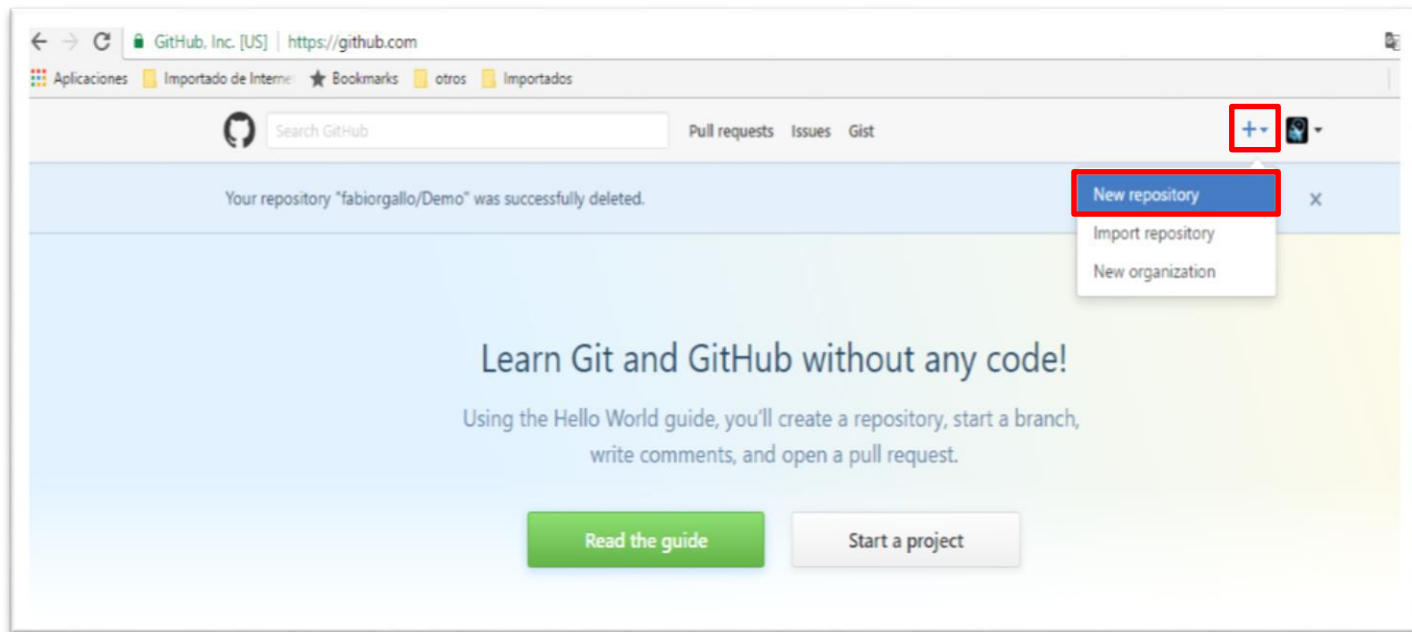
En Eclipse, ir a **Window > Show view > Other...**



Creación de repositorio en la web

Creación de repositorio en *GitHub*



Ir a <https://github.com/> y crear una cuenta, activarla y luego iniciar sesión.



Creación de repositorio en *GitHub*

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner:  fabiorgallo / Repository name: Demo 

Great repository names are short and memorable. Need inspiration? How about [expert-waddle](#).

Description (optional): Repositorio de ejemplo para la presentación.

☒ Public
Anyone can see this repository. You choose who can commit.

☐ Private
You choose who can see and commit to this repository.

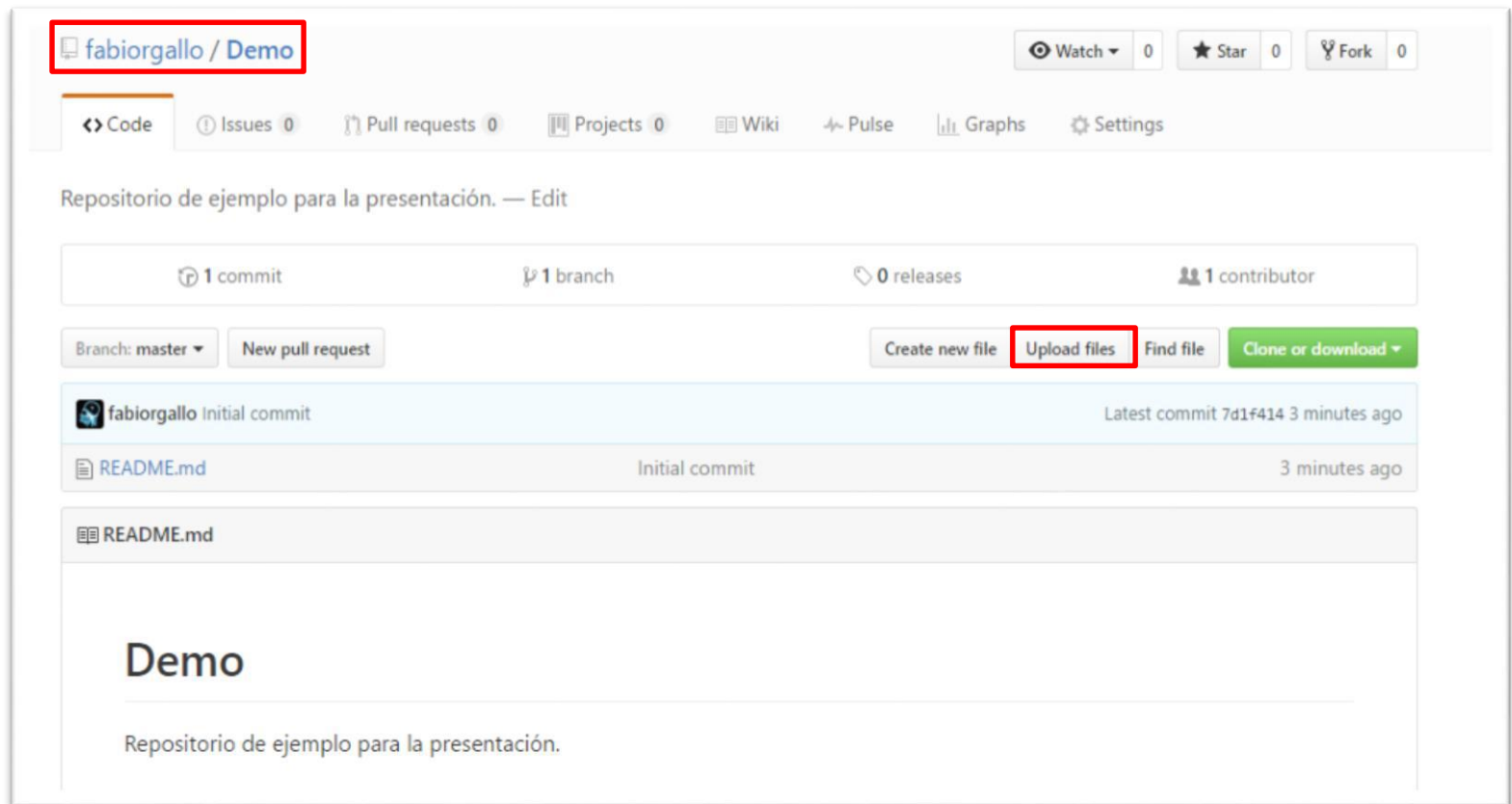
☒ Initialize this repository with a README
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: None | Add a license: None ⓘ

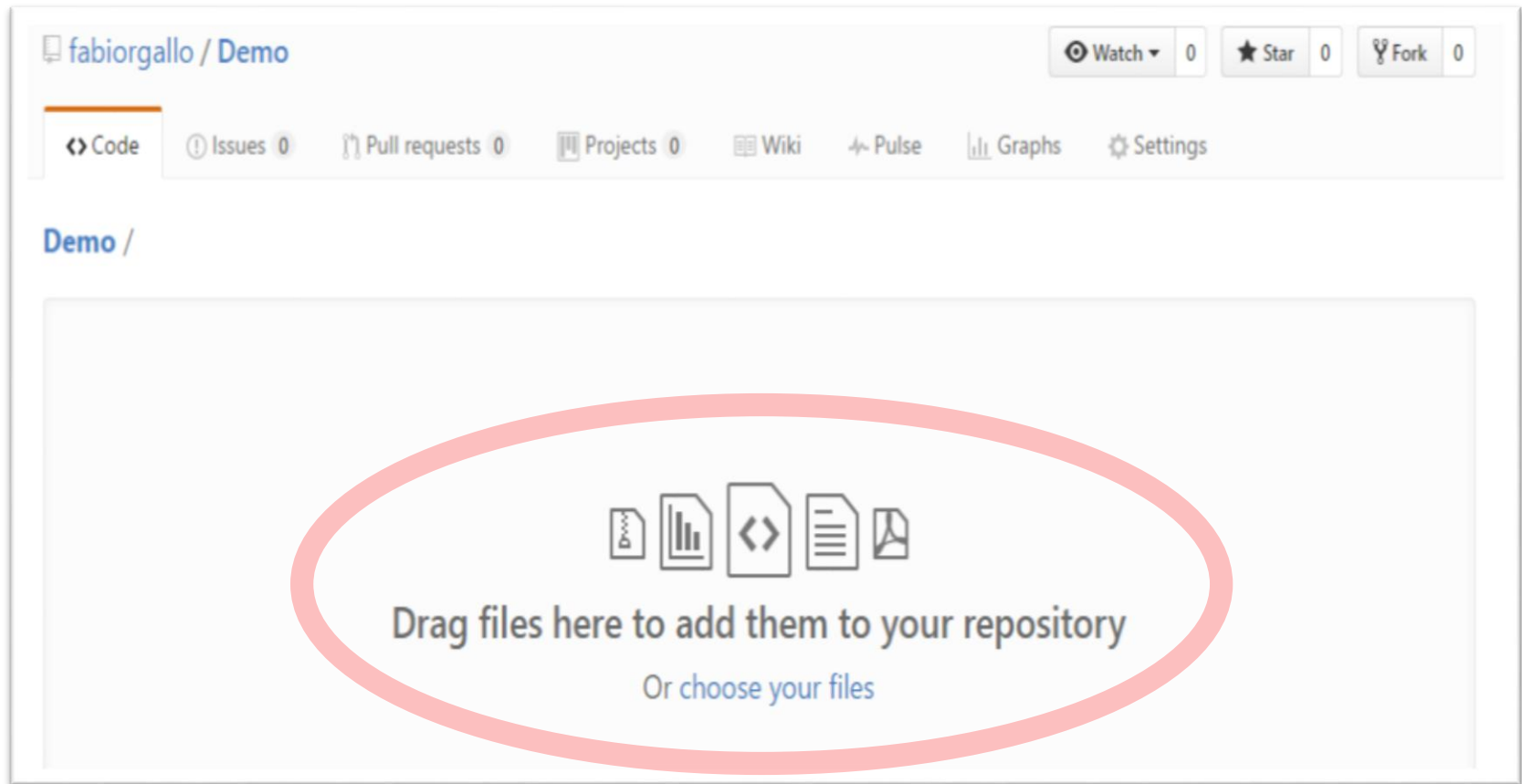
Create repository

Al tildar esta opción crea una rama *master* por defecto. Así permite luego agregar archivos.

Creación de repositorio en *GitHub*



Creación de repositorio en *GitHub*




Para el ejemplo cargamos el proyecto Java del juego **Snake**.

Creación de repositorio en *GitHub*

Lista de archivos subidos.

/Snake/nbproject/project.properties	×
/Snake/nbproject/project.xml	×
/Snake/Snake.png	×
/Snake/src/Main.java	×
/Snake/src/GUI.java	×
/Snake/Type.png	×
/Snake/src/Snake.java	×

 **Commit changes**

Subida del proyecto

Subida del proyecto inicial "Snake".

☒ Commit directly to the `master` branch.

☐ Create a new branch for this commit and start a pull request. [Learn more about pull requests.](#)

Commit changes Cancel

Enviar solicitud de colaboración

fabriorgallo / Demo 1

Watch 0 Star 0 Fork 0

<> Code Issues 0 Pull requests 0 Projects 0 Wiki Pulse Graphs Settings 2

Options

Collaborators 3

Branches

Webhooks

Integrations & services

Deploy keys

Collaborators Push access to the repository

This repository doesn't have any collaborators yet. Use the form below to add a collaborator.

Search by username, full name or email address

You'll only be able to find a GitHub user by their email address if they've chosen to list it publicly. Otherwise, use their username instead.

PSS-UNS 4

Add collaborator 5

El sistema envía un mail automáticamente con la invitación al correo del usuario. Este tiene que aceptarla para colaborar.

Options

Collaborators 3

Branches

Webhooks

Integrations & services

Deploy keys

Collaborators Push access to the repository

Awaiting PSS-UNS's response

Copy invite link Cancel invite

Search by username, full name or email address

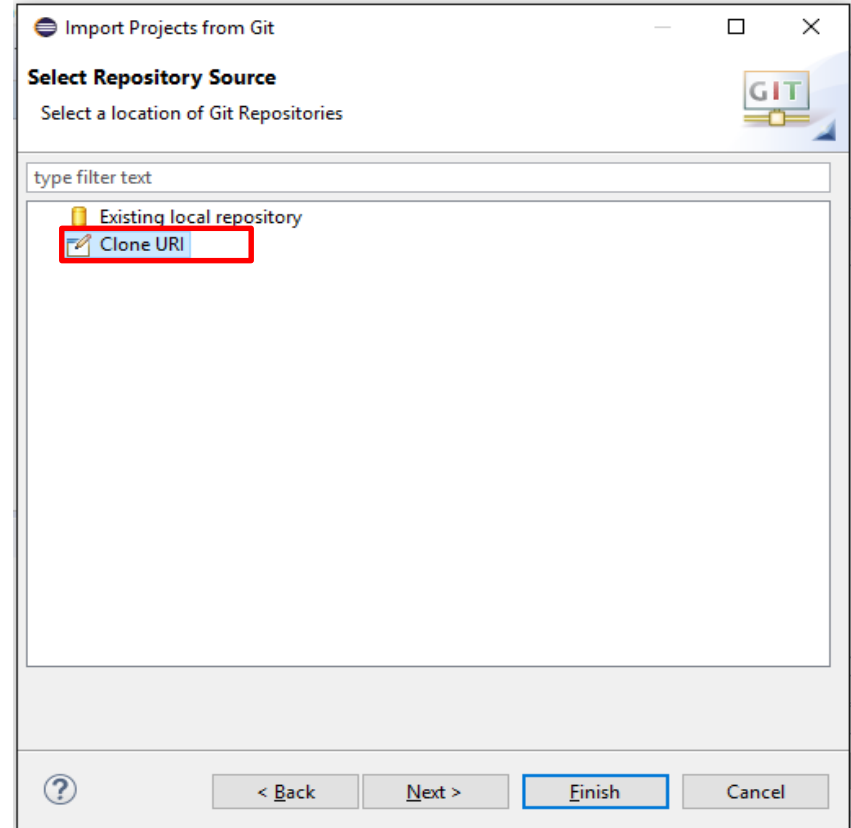
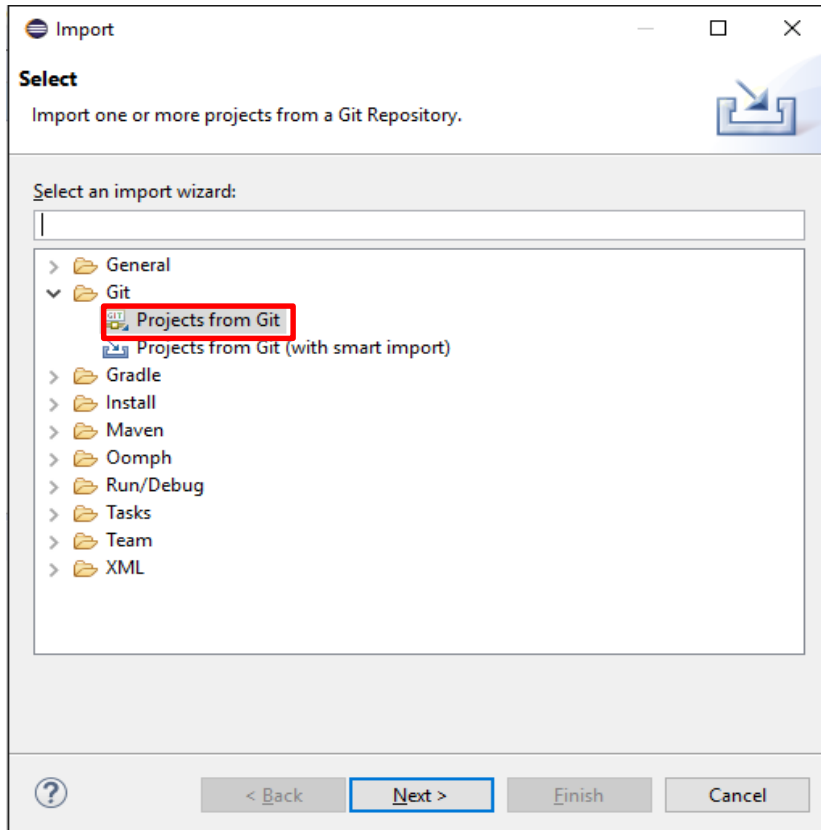
You'll only be able to find a GitHub user by their email address if they've chosen to list it publicly. Otherwise, use their username instead.

Add collaborator 5

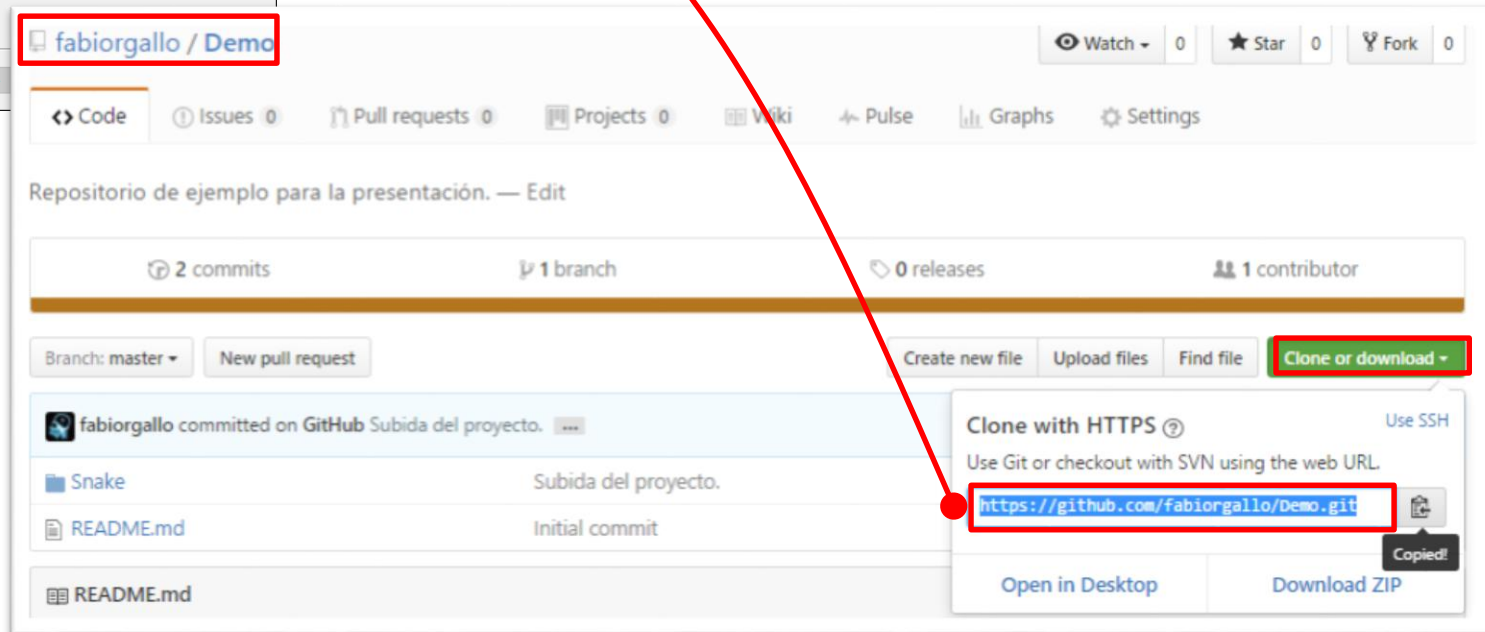
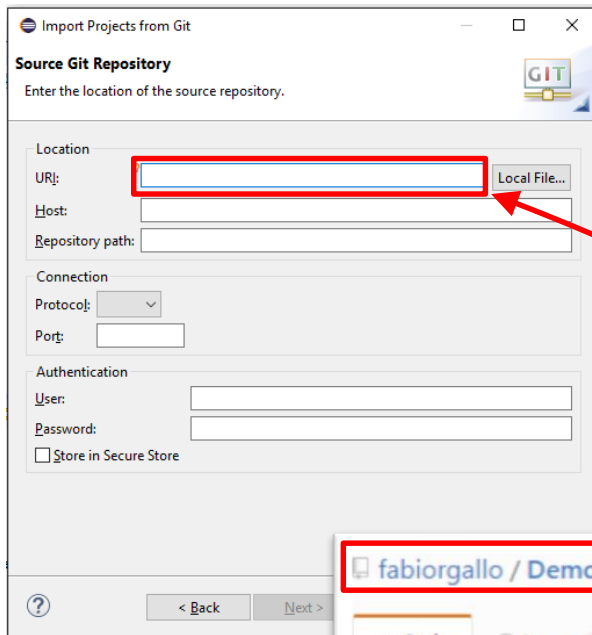
Abrir repositorio en Eclipse

Abrir repositorio en Eclipse

Ir a **File > Import**



Abrir repositorio en Eclipse



Abrir repositorio en Eclipse

Import Projects from Git

Source Git Repository

Enter the location of the source repository.

Location

URI: Local File...

Host:

Repository path:

Connection

Protocol:

Port:

Authentication

User:

Password:

☐ Store in Secure Store

< Back Next > Finish Cancel

Import Projects from Git

Branch Selection

Select branches to clone from remote repository. Remote tracking branches will be created to track updates for these branches in the remote repository.

Branches of https://github.com/fabiorgallo/Demo.git:

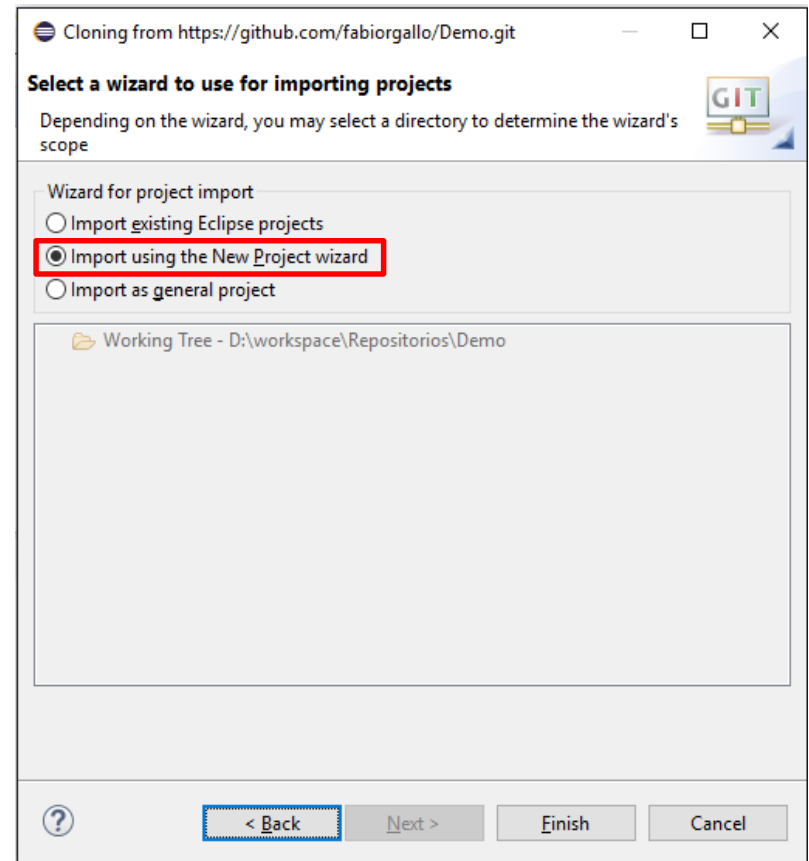
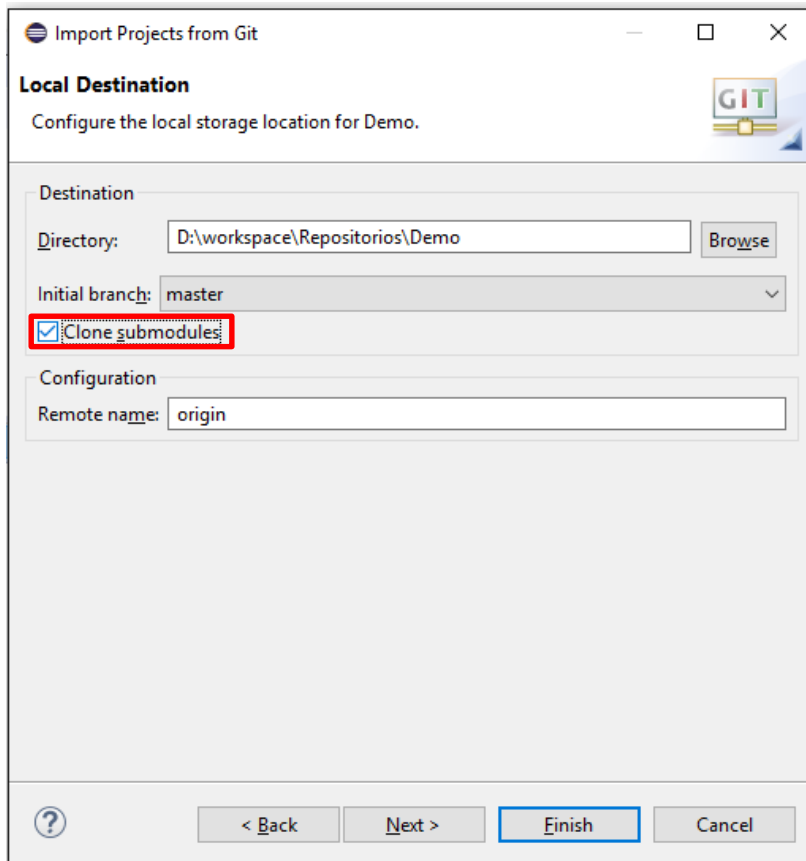
type filter text

☒ master

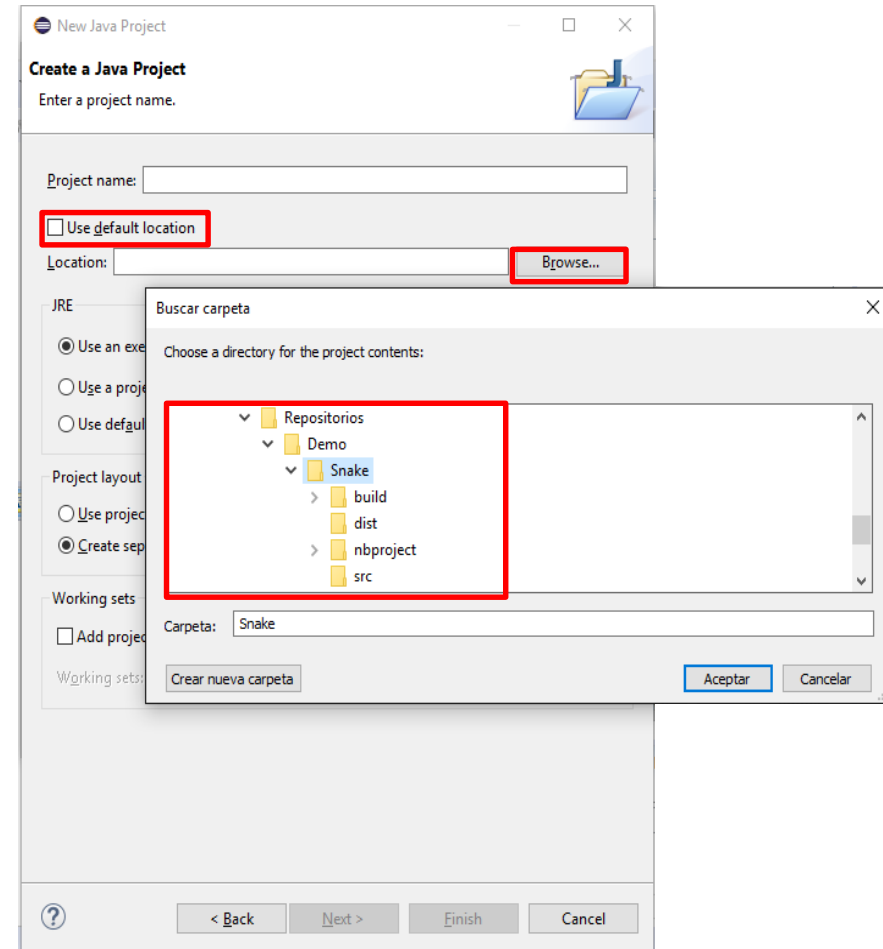
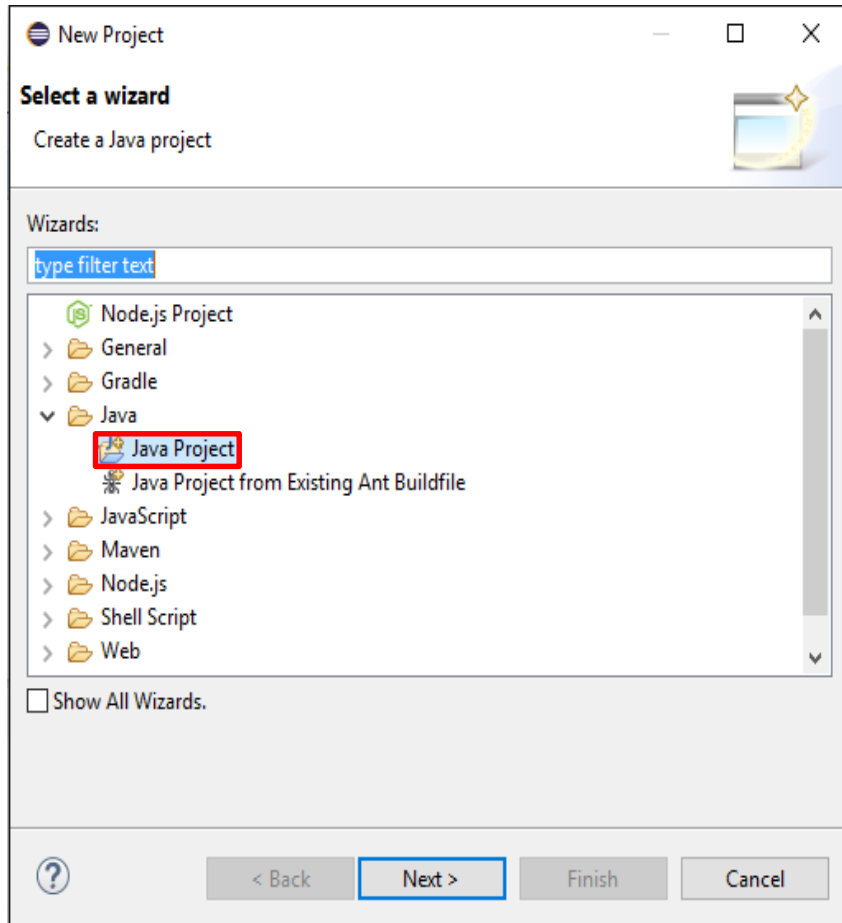
Select All Deselect All

< Back Next > Finish Cancel

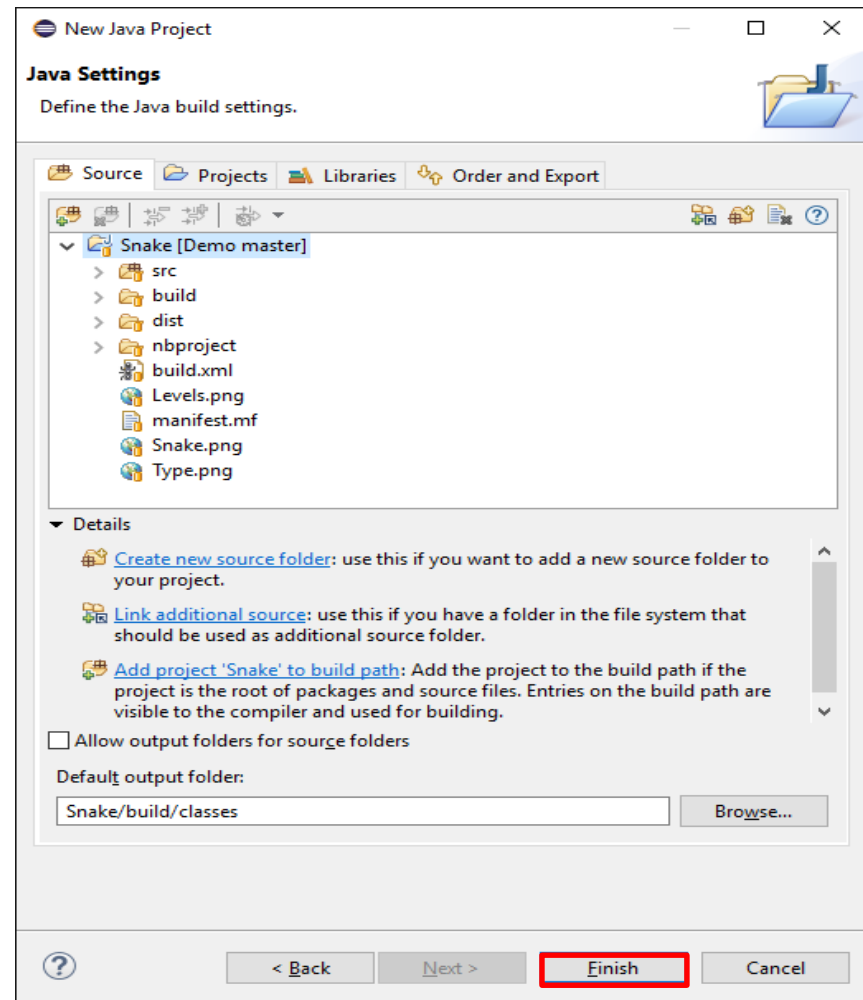
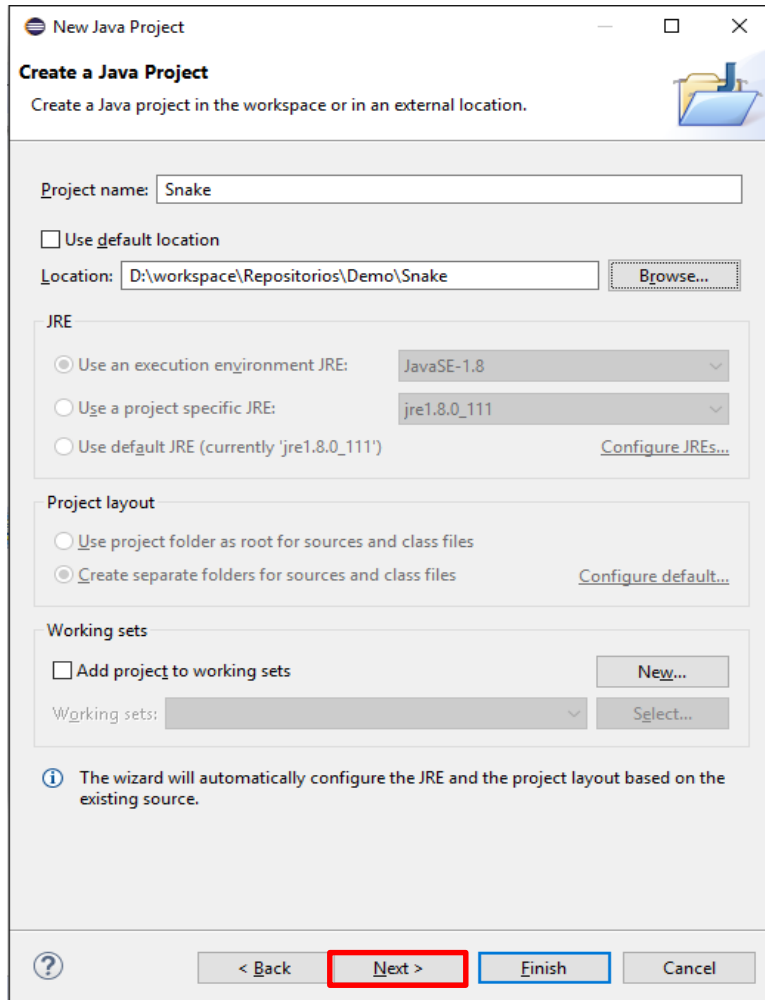
Abrir repositorio en Eclipse



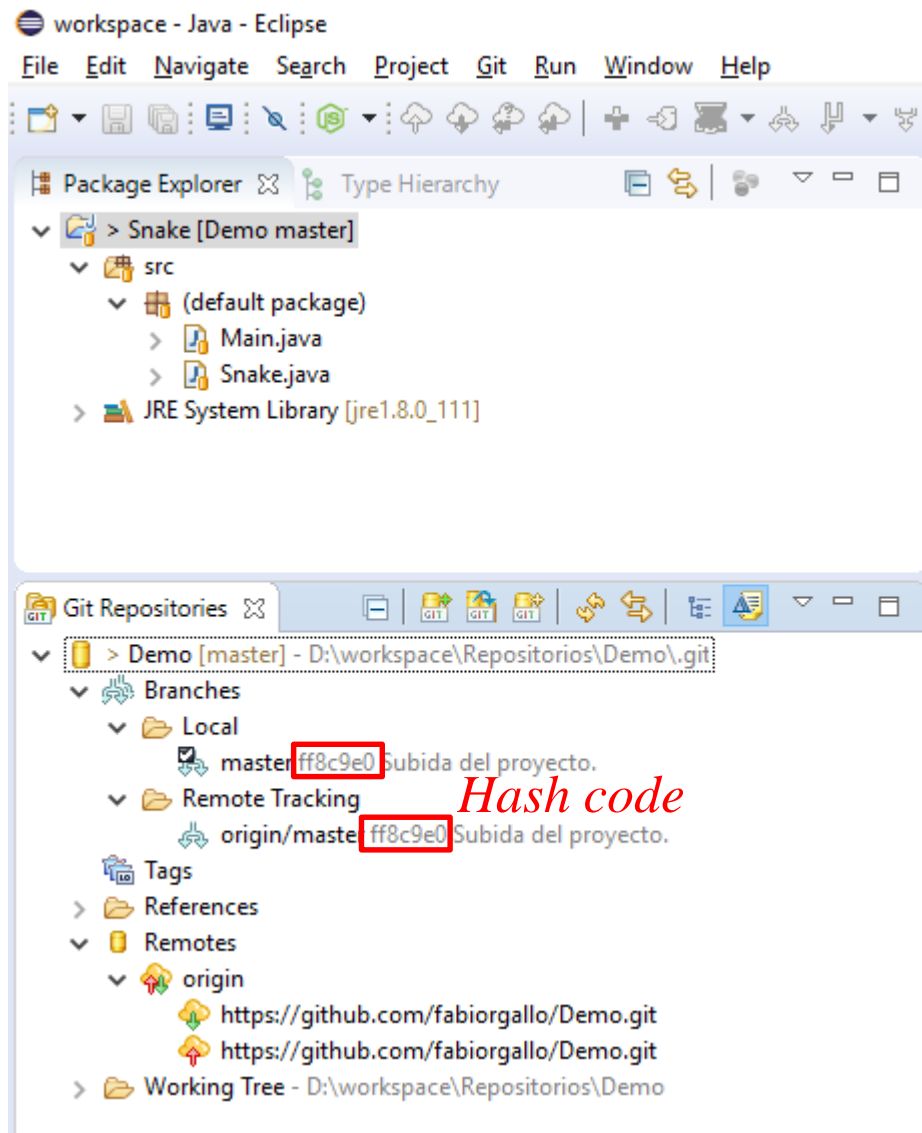
Abrir repositorio en Eclipse



Abrir repositorio en Eclipse



Abrir repositorio en Eclipse



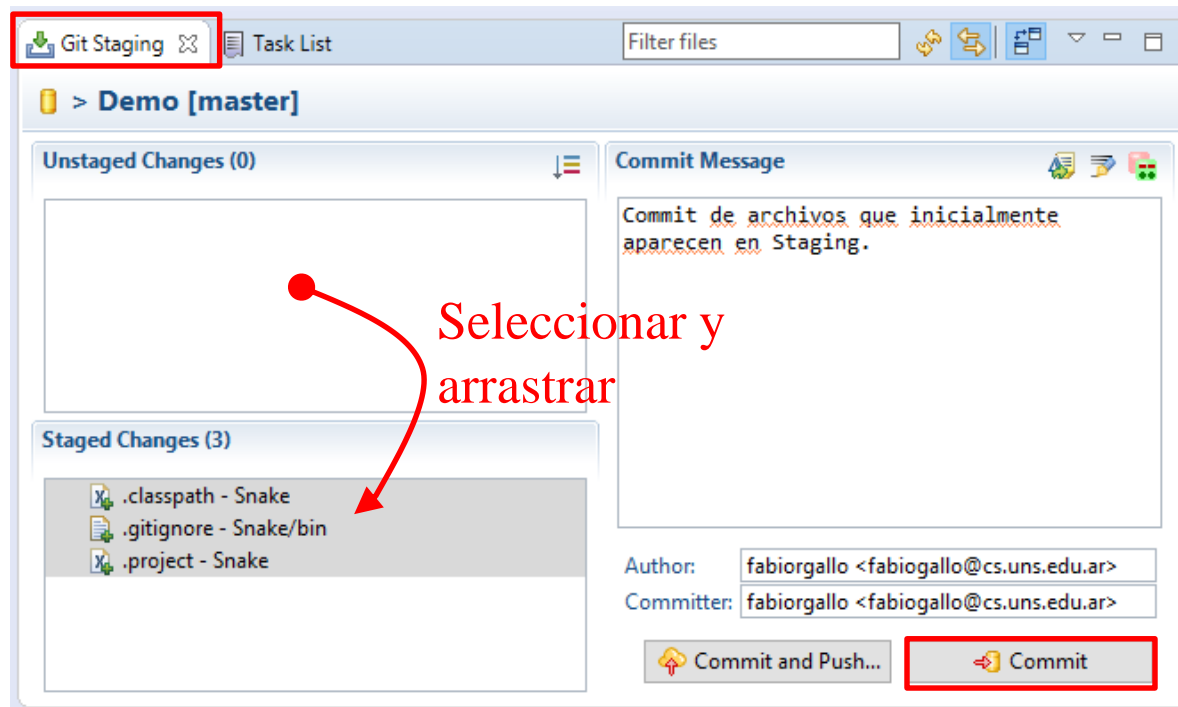
Antes de comenzar a trabajar verificar que coincidan los hash codes.

Sino, presionar



Abrir repositorio en Eclipse

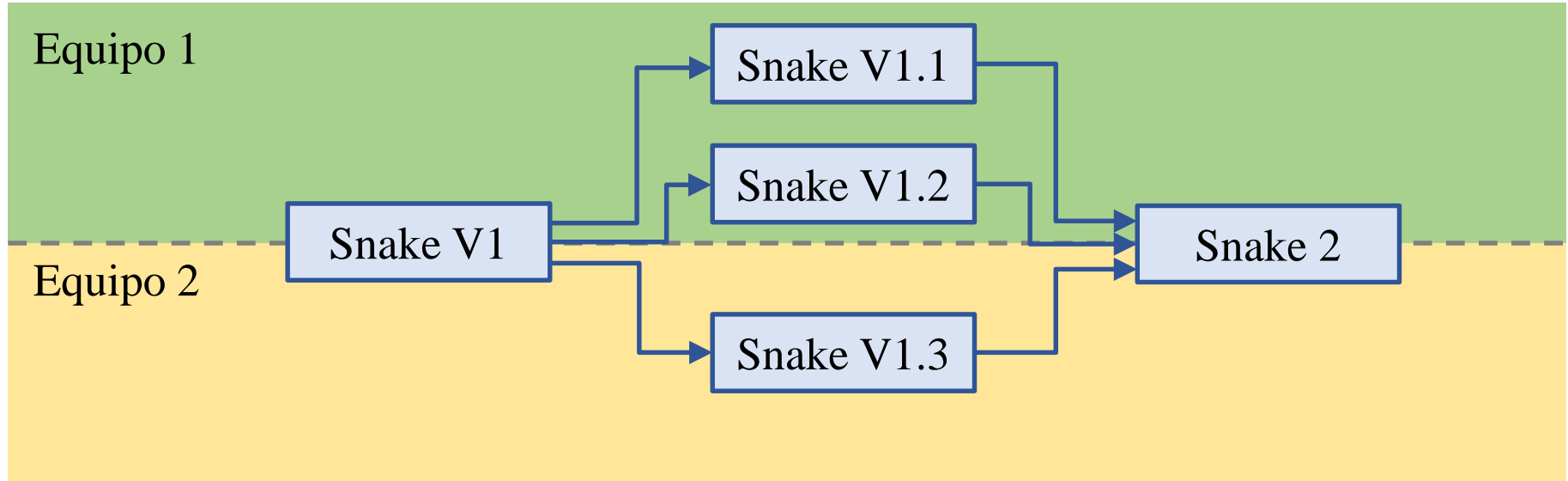
Importante: En este ejemplo, al cargar el repositorio aparecen archivos pendientes que tienen que ser integrados a la rama *master*.



Luego de hacer *commit*, los hash codes de la rama master local y la remota son distintos.

Creación de ramas en Eclipse

Plan de trabajo

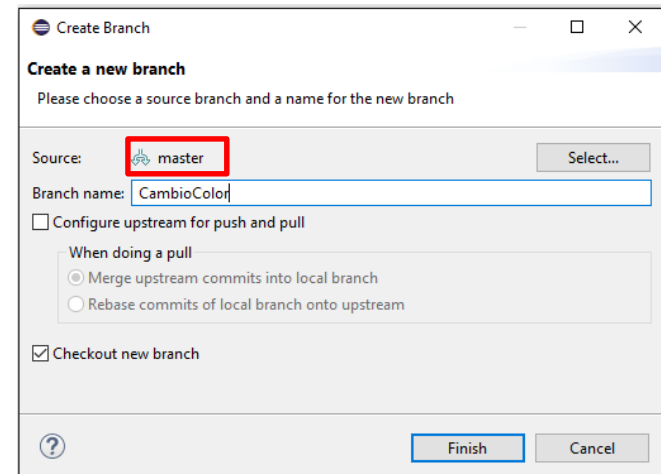
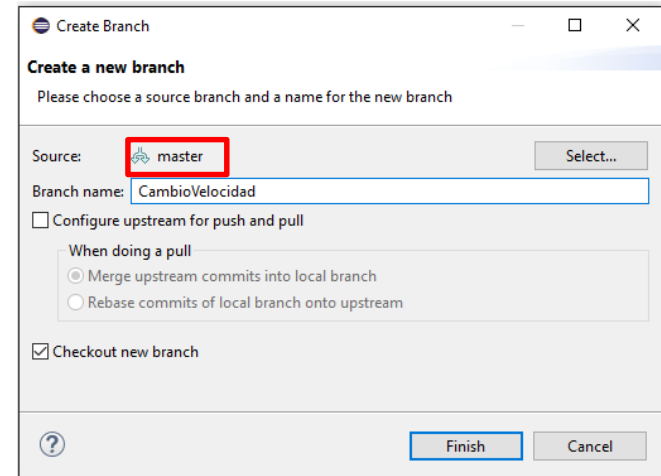
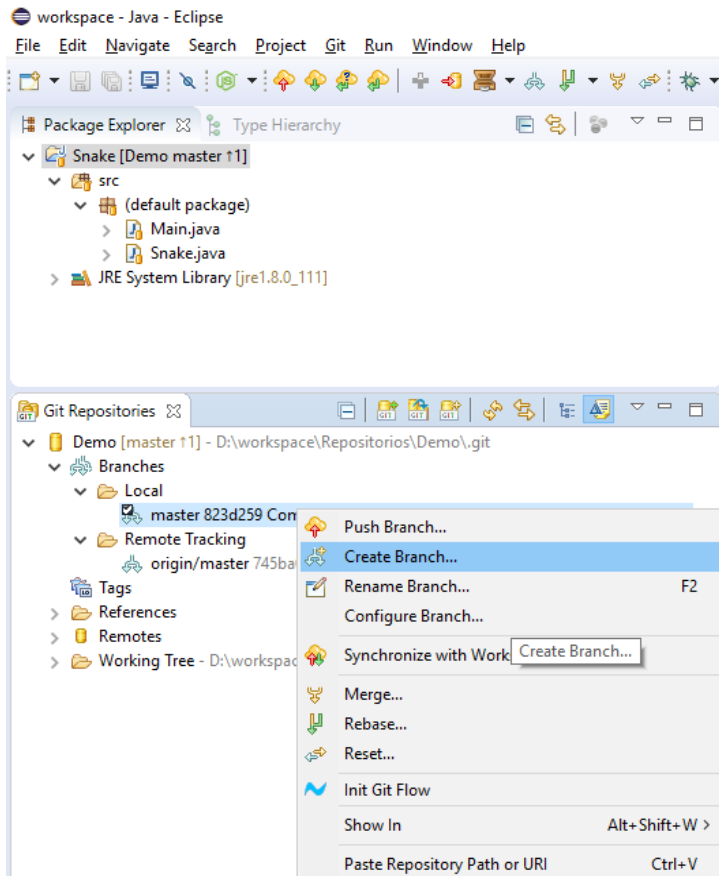


V1.1: Cambio de la velocidad por defecto.

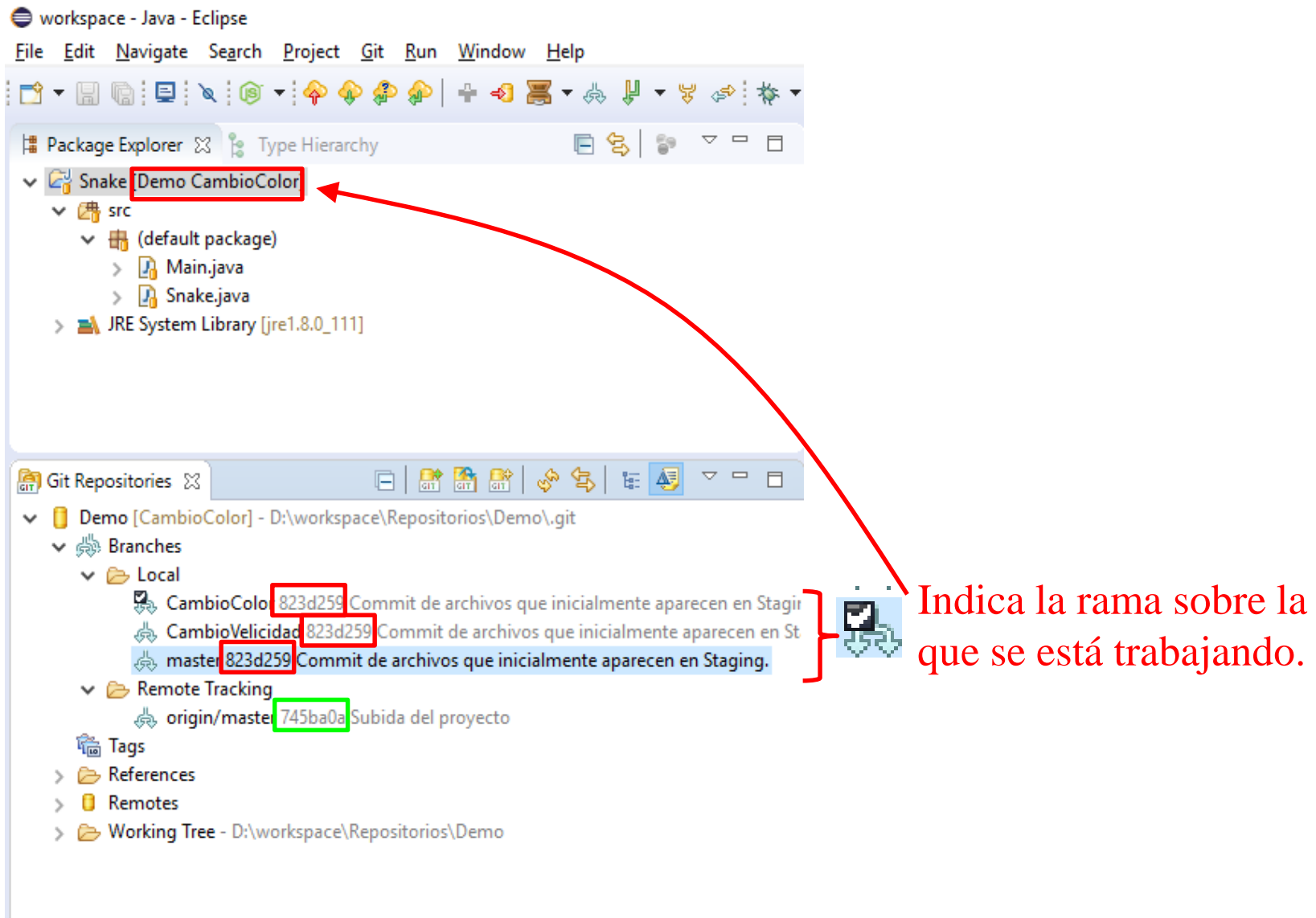
V1.2: Cambio de color del área de juego.

V1.3: Traducción de las opciones del menú.

Creación de ramas



Título



workspace - Java - Eclipse

File Edit Navigate Search Project Git Run Window Help

Package Explorer Type Hierarchy

Snake Demo CambioColor

src

(default package)

Main.java

Snake.java

JRE System Library [jre1.8.0_111]

Git Repositories

Demo [CambioColor] - D:\workspace\Repositorios\Demo\.git

Branches

Local

CambioColor 823d259 Commit de archivos que inicialmente aparecen en Staging

CambioVelocidad 823d259 Commit de archivos que inicialmente aparecen en Staging

master 823d259 Commit de archivos que inicialmente aparecen en Staging.

Remote Tracking

origin/master 745ba0a Subida del proyecto

Tags

References

Remotes

Working Tree - D:\workspace\Repositorios\Demo

Indica la rama sobre la que se está trabajando.

Cambio en rama “CambioVelocidad”

The screenshot shows the Eclipse IDE interface with the following components:

- Package Explorer:** Shows the project structure. The file `Snake.java` is highlighted with a red box.
- Git Repositories:** Shows the local branches. The branch `CambioVelocidad` is selected and highlighted with a red box. The commit message is "Commit de archivos que inicialmente aparecen en".
- Editor:** Displays the source code of `Snake.java`. The line `private int selectedSpeed = SNAKE_RUNNING_SPEED_FAST; // antes era ... = ..._FASTER;` is highlighted with a red box.

```
11 private JPanel board;
12 private JButton[] snakeBodyPart;
13 private JButton bonusfood;
14 private JTextArea scoreViewer;
15
16 // Constants
17 private final int SNAKE_RUNNING_SPEED_FASTEST = 25;
18 private final int SNAKE_RUNNING_SPEED_FASTER = 50;
19 private final int SNAKE_RUNNING_SPEED_FAST = 100;
20 private final int BOARD_WIDTH = 500;
21 private final int BOARD_HEIGHT = 250;
22 private final int SCORE_BOARD_HEIGHT = 20;
23 private final int SNAKE_LENGTH_DEFAULT = 4;
24 private final int SNAKE_BODY_PART_SQURE = 10;
25 private final int BONUS_FOOD_SQURE = 15;
26 private final Point INIT_POINT = new Point(100, 150);
27
28 // Others values
29 private enum GAME_TYPE {NO_MAZE, BORDER, TUNNEL};
30 private int selectedSpeed = SNAKE_RUNNING_SPEED_FAST; // antes era ... = ..._FASTER;
31 private GAME_TYPE selectedGameType = GAME_TYPE.NO_MAZE;
32 private int totalBodyPart;
33 private int directionX;
34 private int directionY;
35 private int score;
36 private Point pointOfBonusFood = new Point();
37 private boolean isRunningLeft;
38 private boolean isRunningRight;
39 private boolean isRunningUp;
40 private boolean isRunningDown;
41 private boolean isBonusFoodAvailable;
42 private boolean isRunning;
43 private Random random = new Random();
44
```




Commit de rama “CambioVelocidad”

Snake.java Git Staging Task List Filter files

> Demo [CambioVelocidad]

Unstaged Changes (0)

Staged Changes (2)

- Snake.class - Snake/bin
- Snake.java - Snake/src

Commit Message

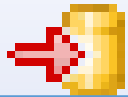
Cambio de la velocidad por defecto. Ahora es la más lenta de 3 posibles.

Author: fabiorgallo <fabiorgallo@cs.uns.edu.ar>

Committer: fabiorgallo <fabiorgallo@cs.uns.edu.ar>

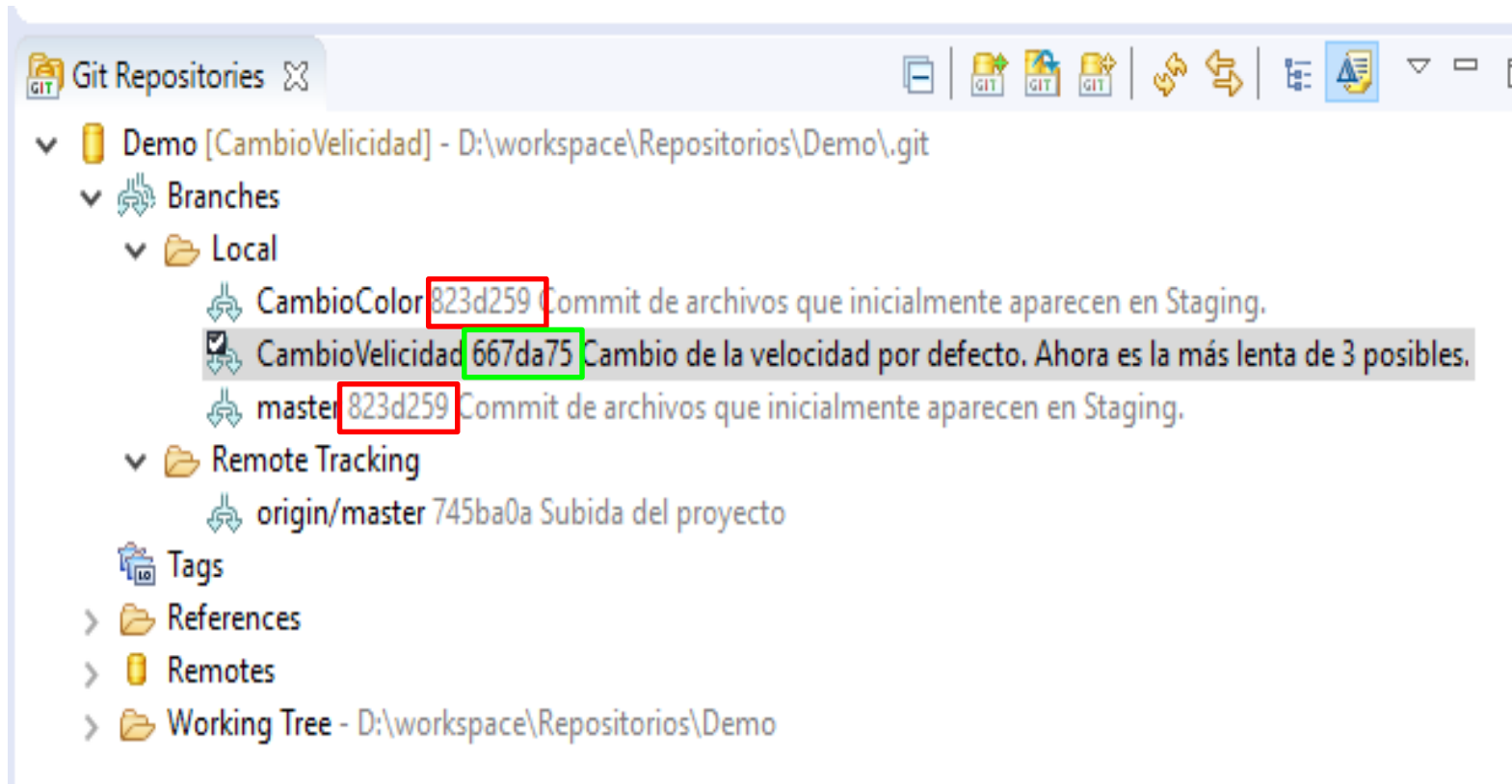
Commit and Push... Commit

Significa que los cambios en estos archivos si o si van a ser incluidos en el próximo *commit*.



Commit de rama “CambioVelocidad”

Luego de hacer el *commit*, la rama tiene que cambiar su hash code.



Cambio en rama “CambioColor”

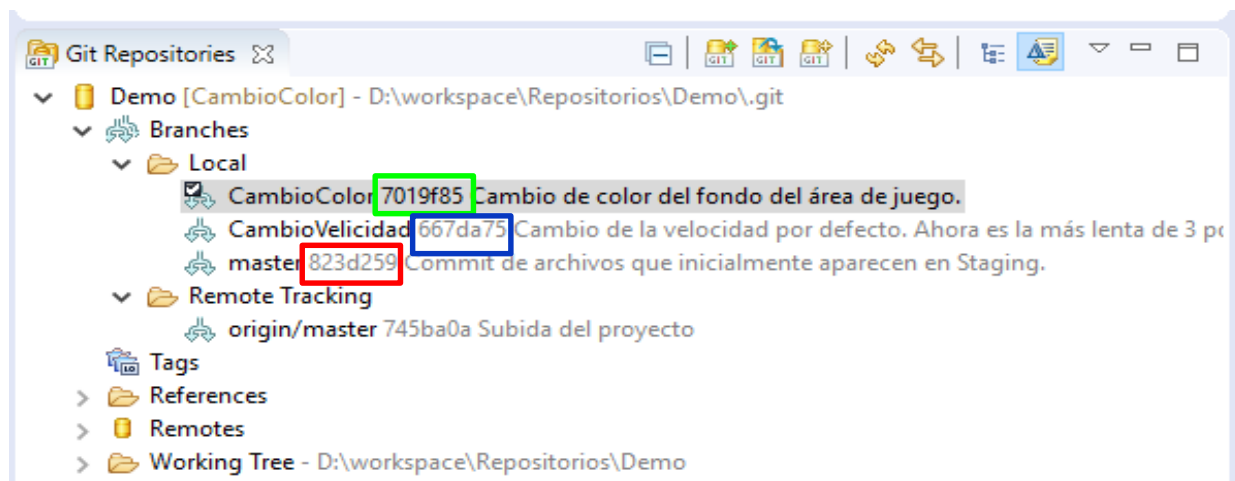
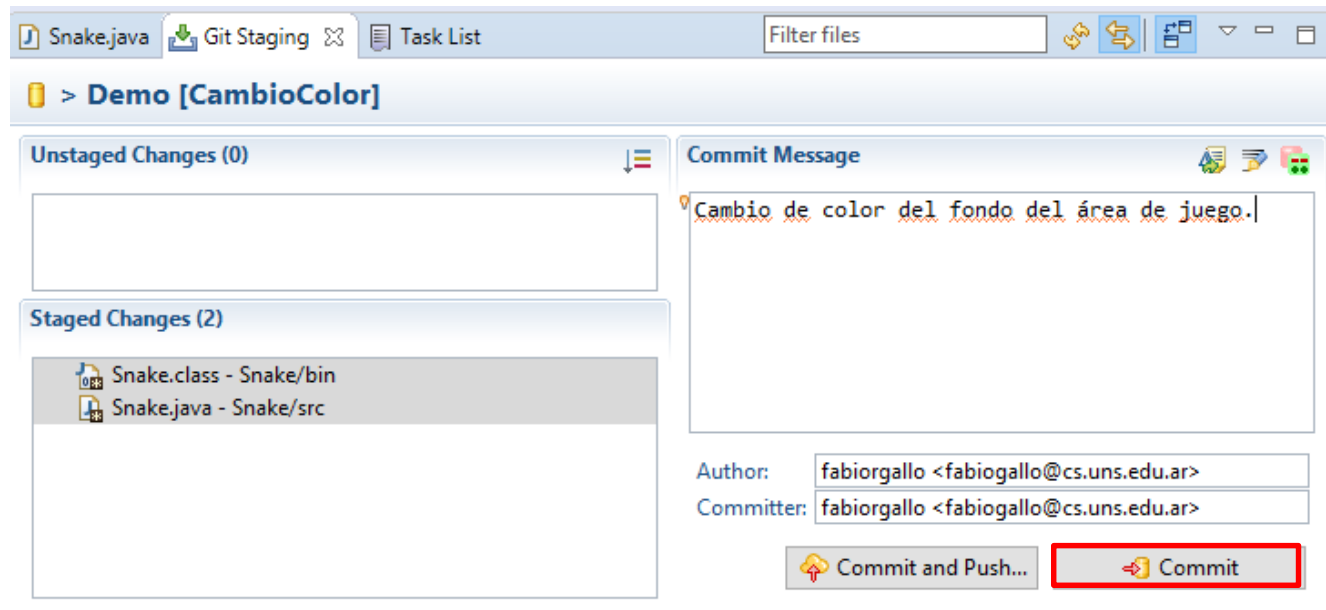
The screenshot shows the Eclipse IDE interface. The Package Explorer on the left shows the project structure: Snake > src > (default package) > Snake.java. The Git Repositories view at the bottom left shows the local branches: Demo [CambioColor], Local, and Remote Tracking. The 'CambioColor 823d259 Commit de archivos que inicialmente aparecen en Staging.' branch is selected and highlighted with a red box. The main editor shows the Snake.java file with Java code. Line 72, `board.setBackground(Color.GRAY); // antes era Color.WHITE`, is highlighted with a red box. The code includes UI setup for a snake game, including a menu bar, score panel, and game board.

```
60
61 //Create Menu bar with functions
62 setJMenuBar(frame);
63 // Start of UI design
64 JPanel scorePanel = new JPanel();
65 scoreViewer = new JTextArea("Score ==>" + score);
66 scoreViewer.setEnabled(false);
67 scoreViewer.setBackground(Color.BLACK);
68
69 board = new JPanel();
70 board.setLayout(null);
71 board.setBounds(0, 0, BOARD_WIDTH, BOARD_HEIGHT);
72 board.setBackground(Color.GRAY); // antes era Color.WHITE
73 scorePanel.setLayout(new GridLayout(0, 1));
74 scorePanel.setBounds(0, BOARD_HEIGHT, BOARD_WIDTH, SCORE_BOARD_HEIGHT);
75 scorePanel.setBackground(Color.RED);
76 scorePanel.add(scoreViewer); // will contain score board
77
78 frame.getContentPane().setLayout(null);
79 frame.getContentPane().add(board);
80 frame.getContentPane().add(scorePanel);
81 frame.setVisible(true);
82 frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
83 frame.addKeyListener(new KeyAdapter() {
84     @Override
85     public void keyPressed(KeyEvent e) {
86         snakeKeyPressed(e);
87     }
88 });
```

Notar que el cambio de velocidad en esta rama no está reflejado.



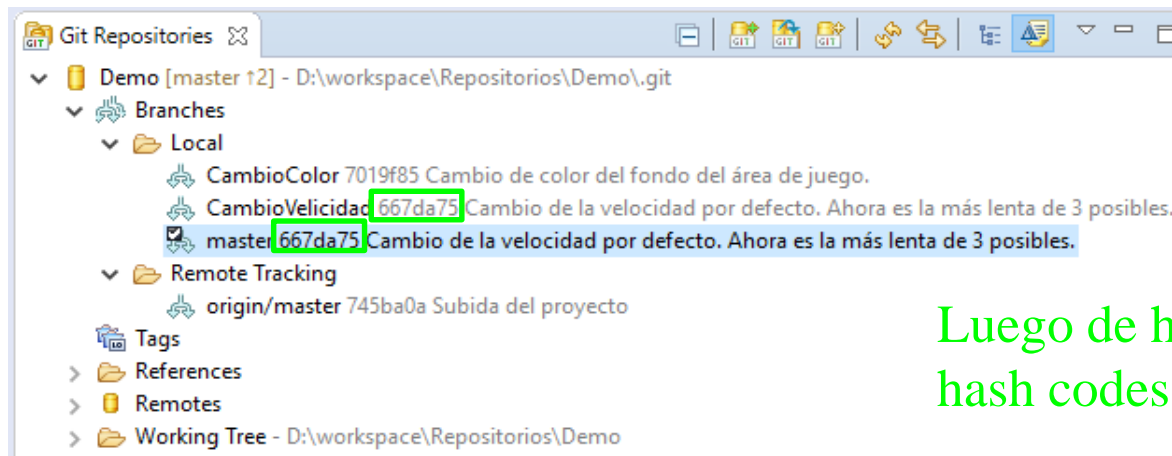
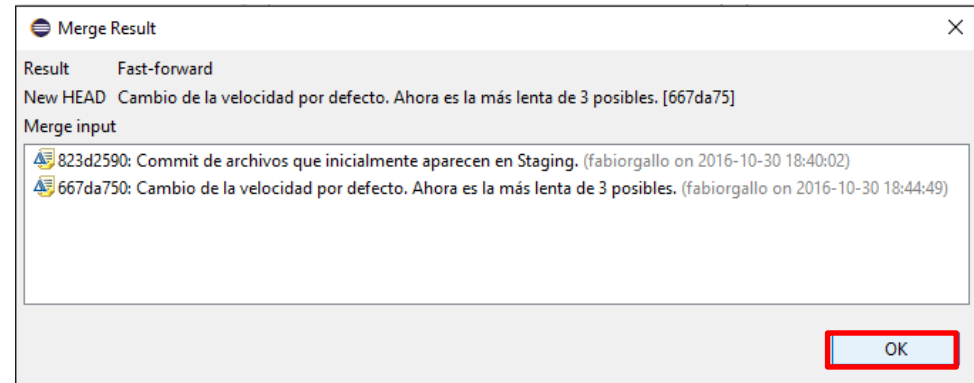
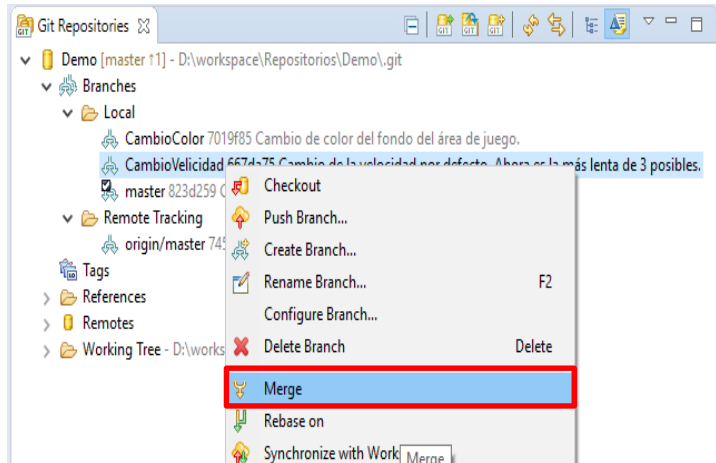
Commit de rama “CambioColor”





Merging de ramas creadas a la rama *master*

“CambioVelocidad”

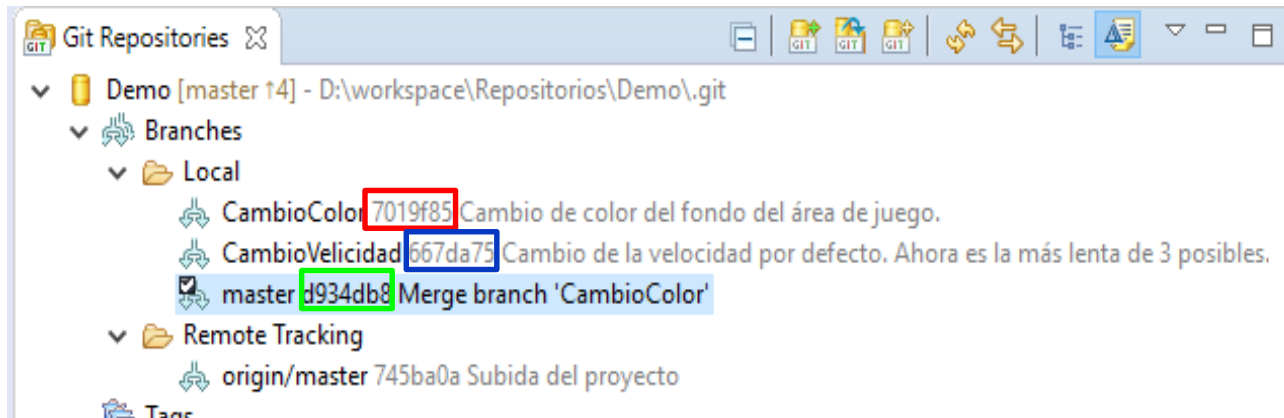
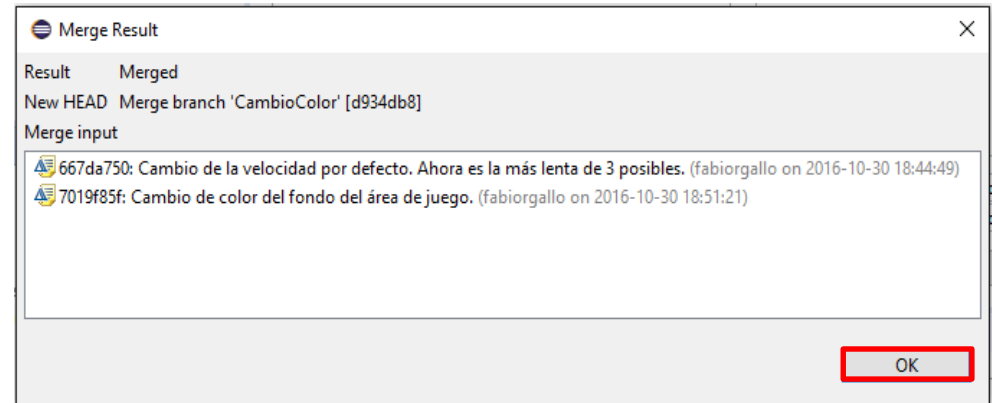
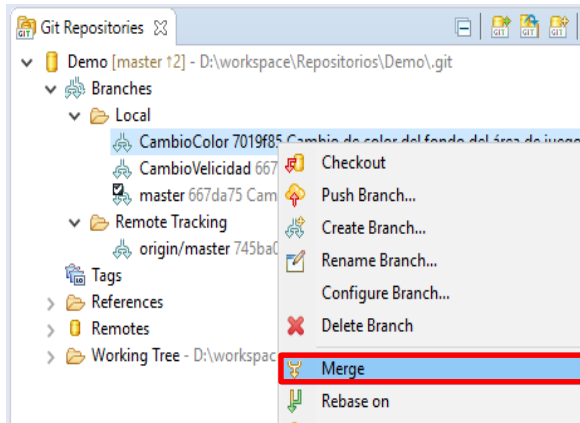


Luego de hacer *merging*, los hash codes deben ser iguales.



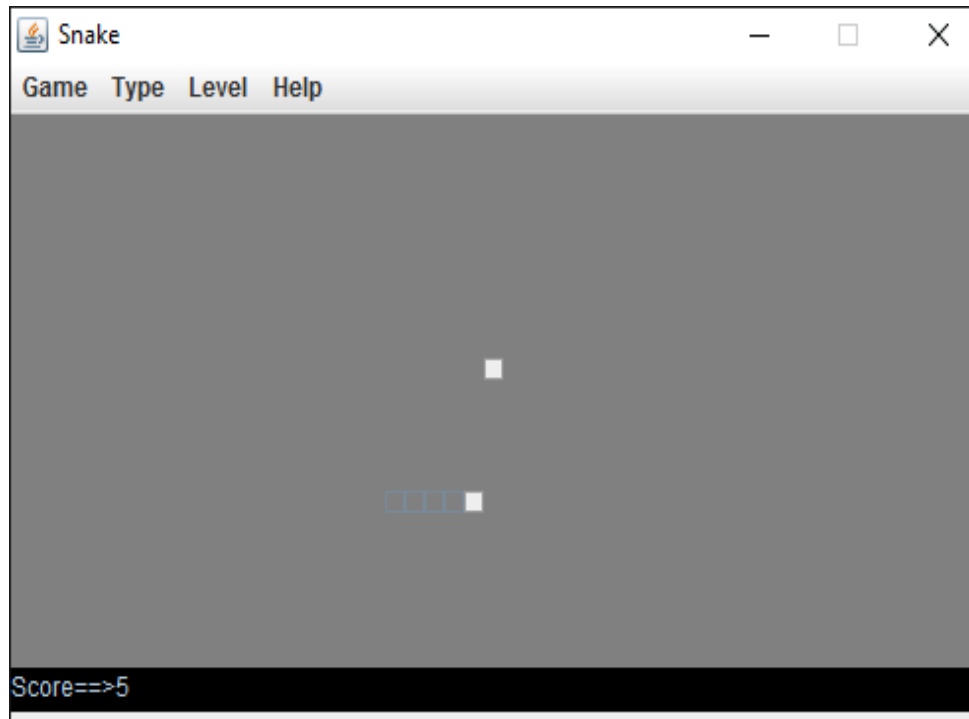
Merging de ramas creadas a la rama *master*

“CambioColor”



Verificar el merge localmente

Para comprobar que ambas ramas fueron integradas exitosamente, podemos ver el código de *master*. Realizamos la ejecución del mismo y observamos que la velocidad y el cambio de color están integrados.



Historial

Desde el menú contextual seleccionar **Team > Show in History**.

Repository: Demo

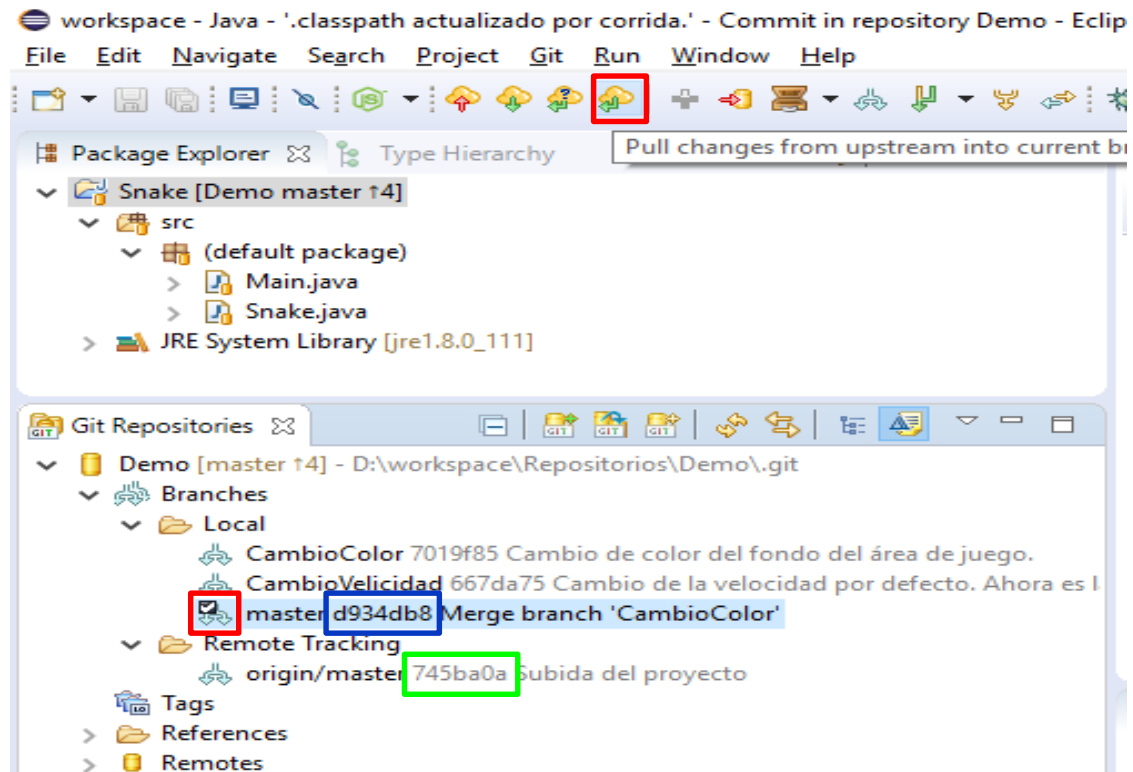
Id	Message	Author	Authored D...	Committer	Committed...
d934db8	master HEAD Merge branch 'CambioColor'	fabiorgallo	51 minutes ago	fabiorgallo	51 minutes ago
7019f85	CambioColor Cambio de color del fondo del área de	fabiorgallo	57 minutes ago	fabiorgallo	57 minutes ago
667da75	CambioVelocidad Cambio de la velocidad por defecto	fabiorgallo	63 minutes ago	fabiorgallo	63 minutes ago
823d259	Commit de archivos que inicialmente aparecen en Sta	fabiorgallo	68 minutes ago	fabiorgallo	68 minutes ago
745ba0a	origin/master Subida del proyecto	fabiorgallo	73 minutes ago	GitHub	73 minutes ago
652e7ef	Initial commit	fabiorgallo	75 minutes ago	fabiorgallo	75 minutes ago

Push en Eclipse



Push al repositorio remoto desde Eclipse

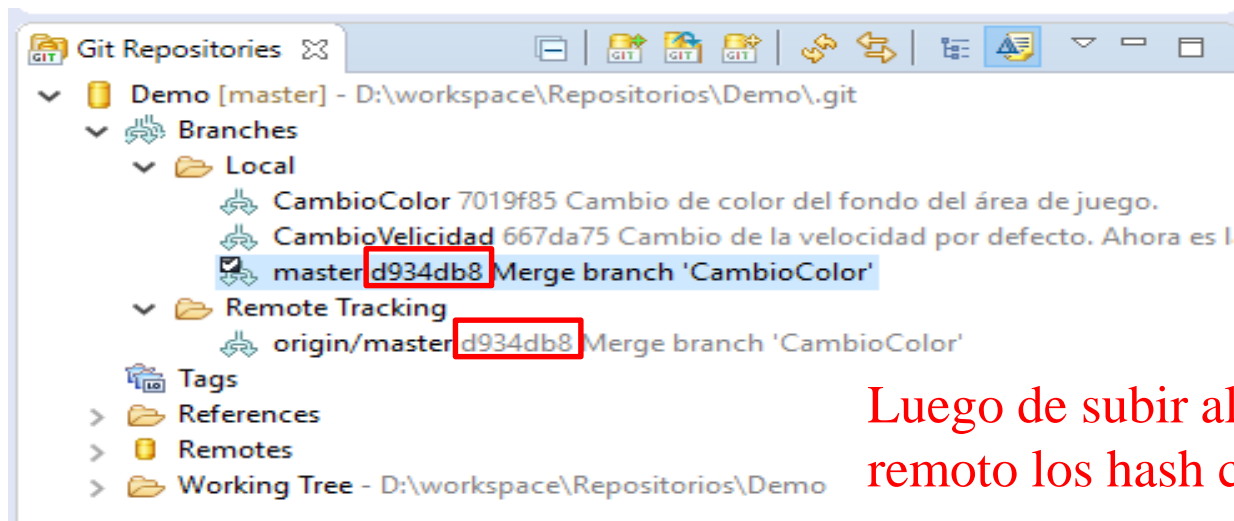
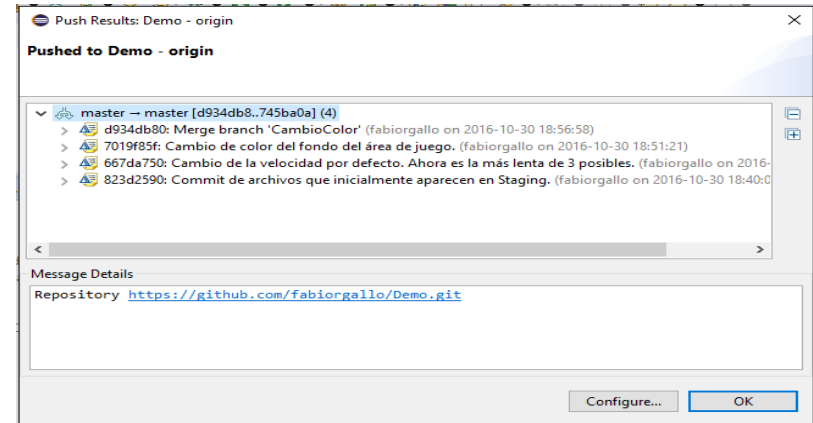
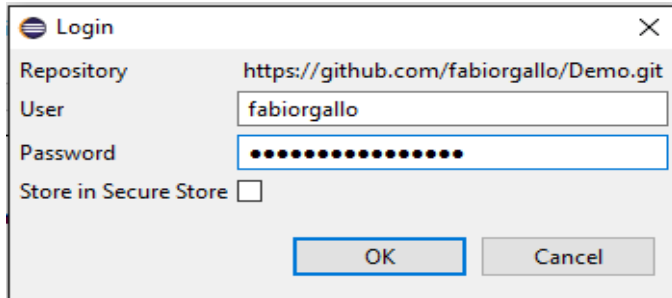
Una vez hecho los cambios en el repositorio **local**, hay que integrarlos al repositorio **remoto**.



Notar que antes de hacer *push* los hash codes de las ramas *master* local y remoto son distintos.



Push al repositorio remoto desde Eclipse



Luego de subir al repositorio remoto los hash codes coinciden



Push al repositorio remoto desde Eclipse

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import java.util.*;

...

private enum GAME_TYPE {NO_MAZE, BORDER, TUNNEL};
private int selectedSpeed = SNAKE_RUNNING_SPEED_FAST; // antes era ... = ..._FASTER
private GAME_TYPE selectedGameType = GAME_TYPE.NO_MAZE;
private int totalBodyPart;
private int directionX;
private int directionY;

...

board = new JPanel();
board.setLayout(null);
board.setBounds(0, 0, BOARD_WIDTH, BOARD_HEIGHT);
board.setBackground(Color.GRAY); // antes era Color.WHITE
scorePanel.setLayout(new GridLayout(0, 1));
scorePanel.setBounds(0, BOARD_HEIGHT, BOARD_WIDTH, SCORE_BOARD_HEIGHT);
scorePanel.setBackground(Color.RED);
scorePanel.add(scoreViewer); // will contain score board
```

Se puede verificar en el repositorio web que los cambios fueron integrados.

Colaboración - Conflictos

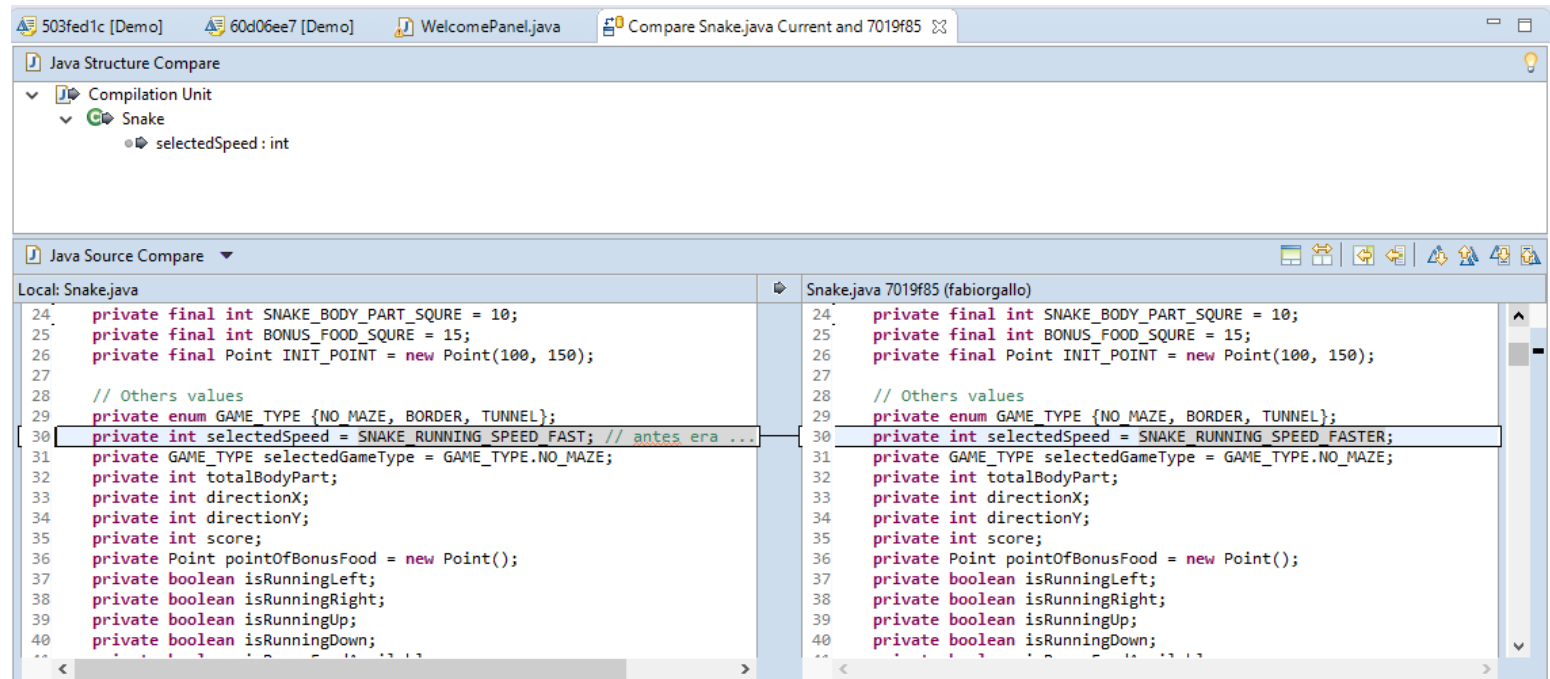
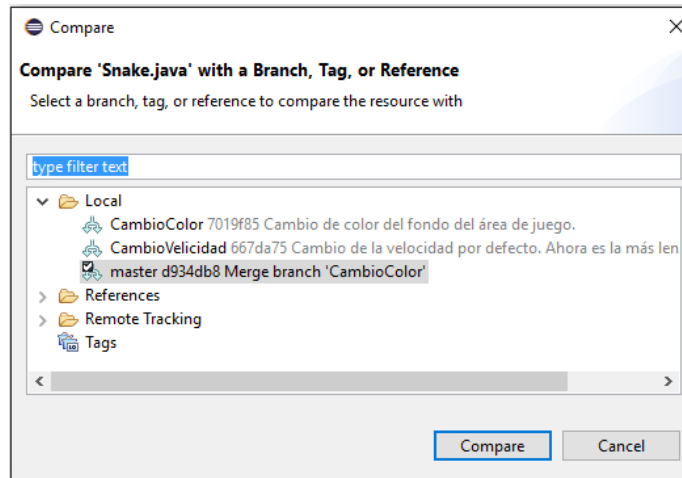
Una vez que alguien acepta la solicitud de colaboración puede cargar el proyecto en un repositorio local desde Eclipse. Cuando cada colaborador sube al repositorio remoto su trabajo pueden haber conflictos con lo realizado por otros. Eclipse proporciona perspectivas que ayudan a su resolución.

Desde el menú contextual de una clase de una determinada rama seleccionar

Compare with > Branch, tag or reference

El siguiente ejemplo muestra la comparación de la clase `Snake.java` de la rama *master* del repositorio local con la misma clase en la rama *CambioColor*.

Colaboración - Conflictos



Recursos utilizados para la presentación

- Git version control with Eclipse (EGit) – Tutorial
<http://www.vogella.com/tutorials/EclipseGit/article.html#completely-new-to-git>
- Remote Git Server and Merging:Git & Eclipse
<https://www.youtube.com/watch?v=KfeqnerMmE>
- Egit/user guide
http://wiki.eclipse.org/EGit/User_Guide
- Usar Git en Eclipse (Egit) Pull, commit y Fetch
<https://www.youtube.com/watch?v=5UBsESBfzzo>