

IES SEVERO OCHOA

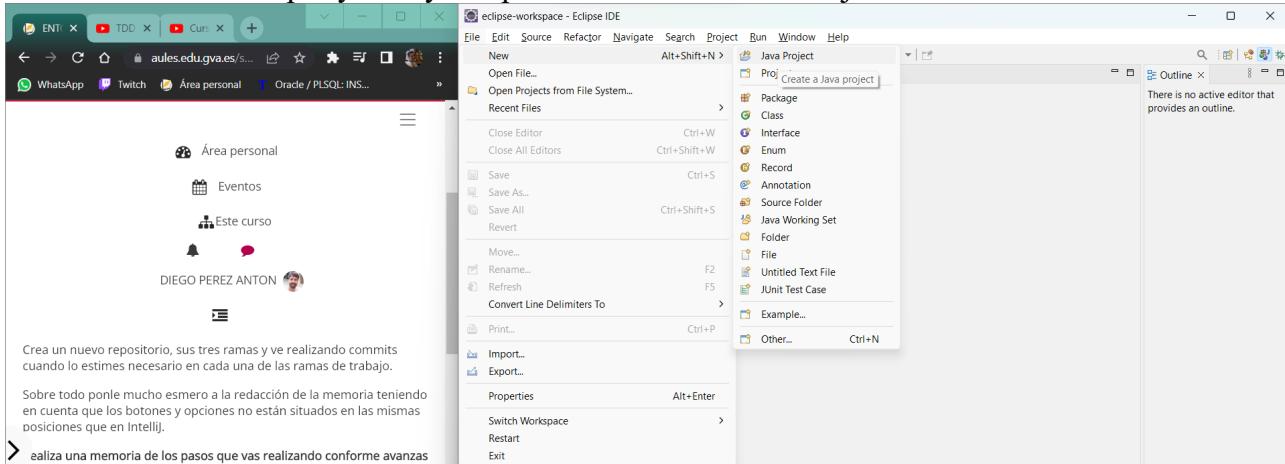
ENTORNOS DE DESARROLLO



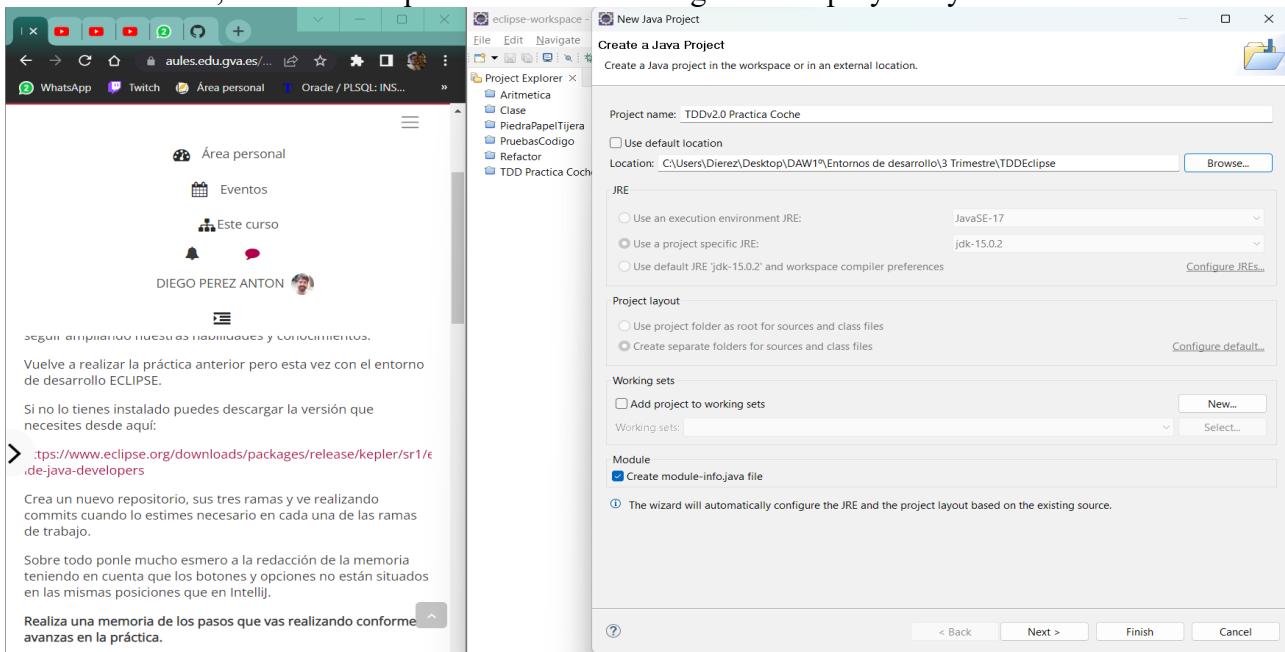
DIEGO PEREZ ANTON
2022/2023

Mi primer TDD V2.0

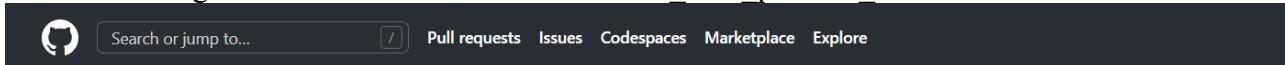
-Inicio el entorno Eclipse y le doy a la pestaña File/New/Java Project:



-En Project name pongo el nombre del proyecto, desmarco la casilla Use default location, le doy al botón Browse... , cambio la carpeta de destino donde guardar el proyecto y botón Finish:



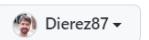
-Dentro de mi github creo un directorio llamado tdd_v2.0_practica_coche:



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Owner *



Repository name *

tdd_v2.0_practica_coche

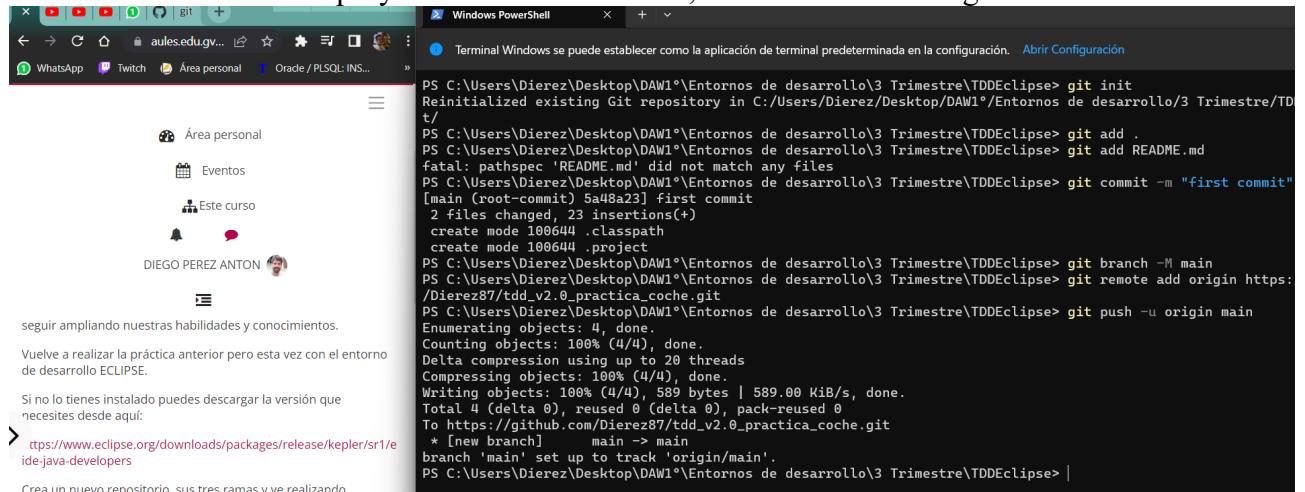
Great repository names are taken. How about [reimagined-octo-system](#)?

-Una vez creado el proyecto, me dirijo al directorio del proyecto e inicio la consola para añadir los siguientes comandos del directorio de github:

...or create a new repository on the command line

```
echo "# tdd_v2.0_practica_coche" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/Dierez87/tdd_v2.0_practica_coche.git
git push -u origin main
```

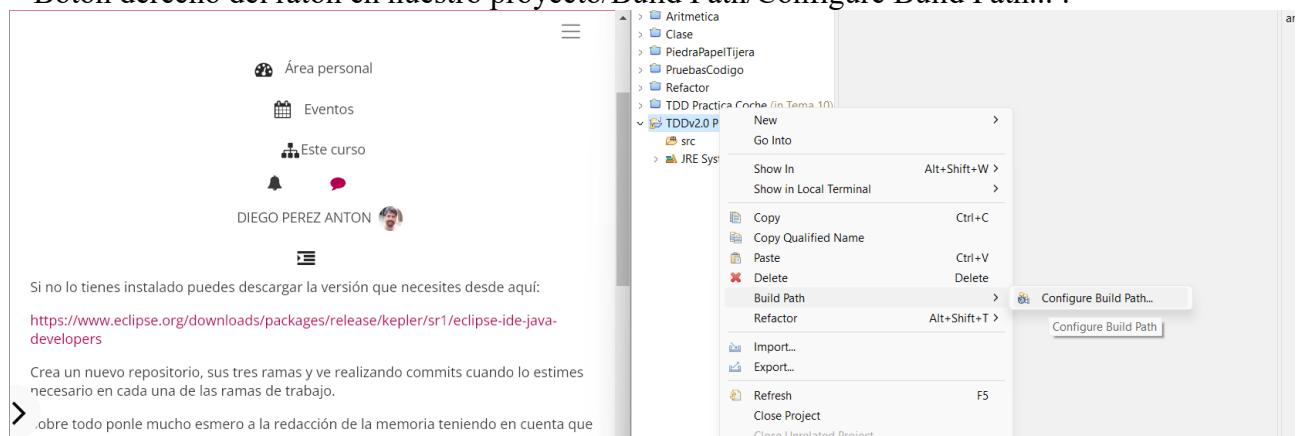
-Accedo al directorio del proyecto e inicio la consola, también he añadido git add . :



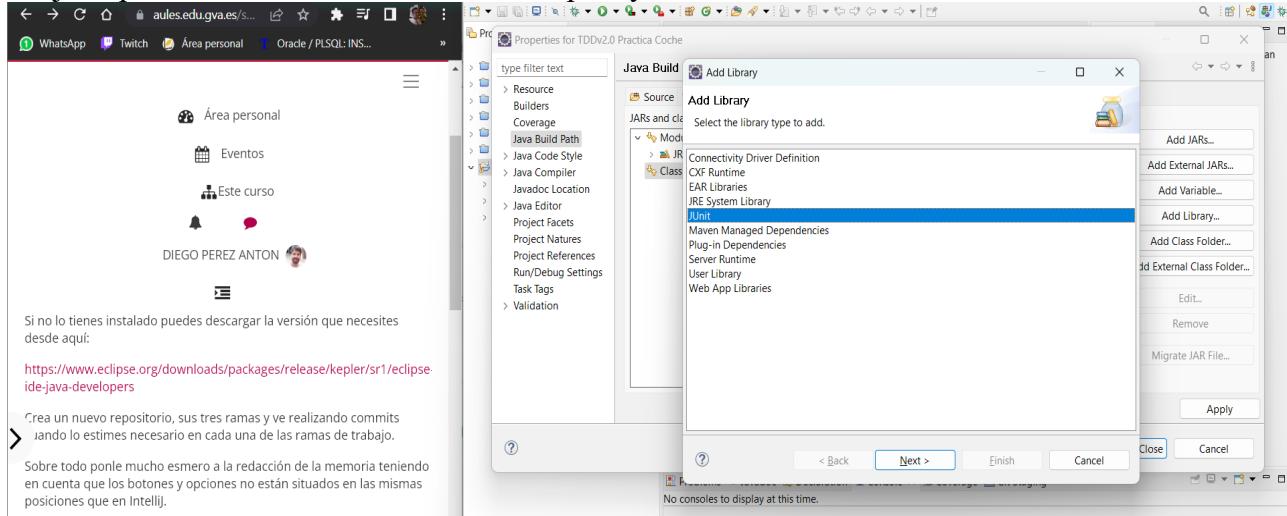
```
Windows PowerShell Terminal Windows se puede establecer como la aplicación de terminal predeterminada en la configuración. Abrir Configuración
PS C:\Users\Dieitez\Desktop\DAW1\Entornos de desarrollo\3 Trimestre\TDDEclipse> git init
Reinitialized existing Git repository in C:/Users/Diezter/Desktop/DAW1/Entornos de desarrollo/3 Trimestre/TDDEclipse/
PS C:\Users\Dieitez\Desktop\DAW1\Entornos de desarrollo\3 Trimestre\TDDEclipse> git add .
PS C:\Users\Dieitez\Desktop\DAW1\Entornos de desarrollo\3 Trimestre\TDDEclipse> git add README.md
fatal: pathspec 'README.md' did not match any files
PS C:\Users\Dieitez\Desktop\DAW1\Entornos de desarrollo\3 Trimestre\TDDEclipse> git commit -m "first commit"
[main (root-commit) 5a48a23] first commit
 2 files changed, 23 insertions(+)
   create mode 100644 .classpath
   create mode 100644 .project
PS C:\Users\Dieitez\Desktop\DAW1\Entornos de desarrollo\3 Trimestre\TDDEclipse> git branch -M main
PS C:\Users\Dieitez\Desktop\DAW1\Entornos de desarrollo\3 Trimestre\TDDEclipse> git remote add origin https://Dierez87/tdd_v2.0_practica_coche.git
PS C:\Users\Dieitez\Desktop\DAW1\Entornos de desarrollo\3 Trimestre\TDDEclipse> git push -u origin main
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 20 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 589 bytes | 589.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Dierez87/tdd_v2.0_practica_coche.git
 * [new branch] main -> main
branch 'main' set up to track 'origin/main'.
PS C:\Users\Dieitez\Desktop\DAW1\Entornos de desarrollo\3 Trimestre\TDDEclipse> |
```

-Ahora añado la librería de JUnit al proyecto:

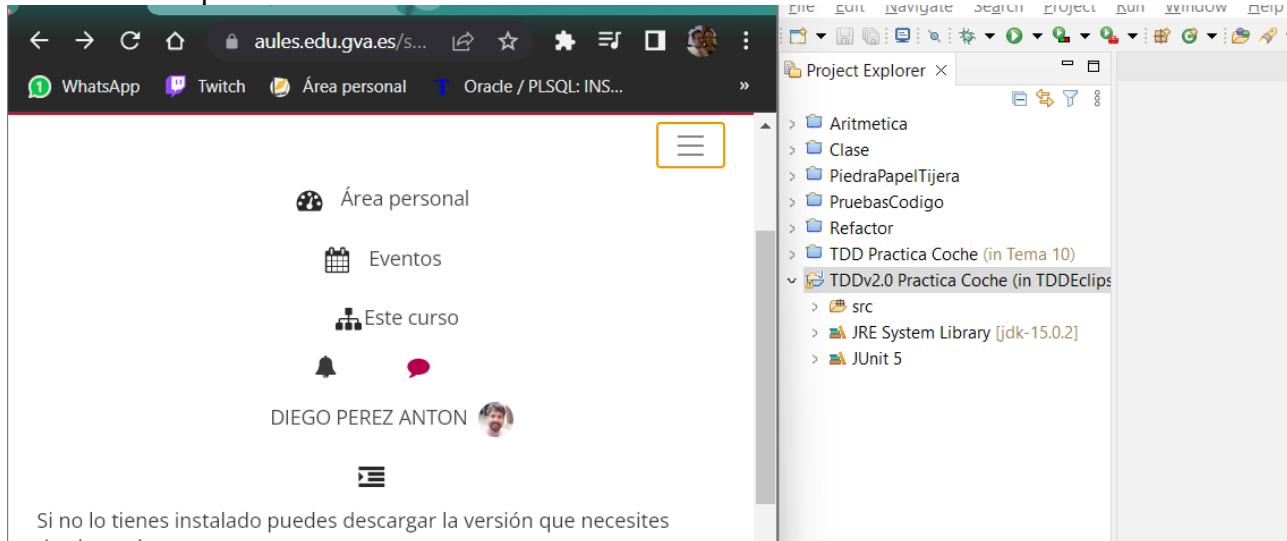
Botón derecho del ratón en nuestro proyecto/Build Path/Configure Build Path... :



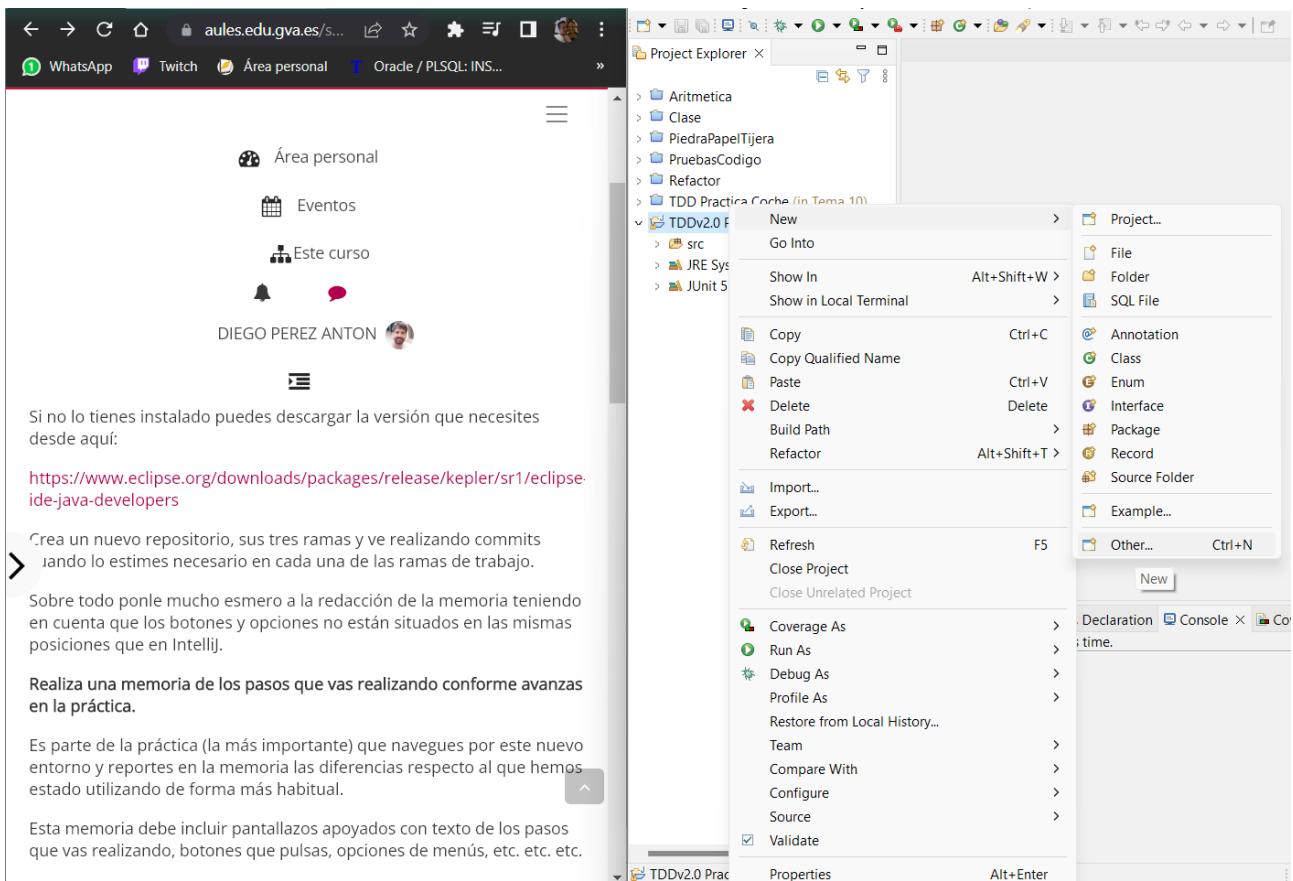
-Elijo la pestaña Libraries, selecciono Classpath y añado la librería de Junit:



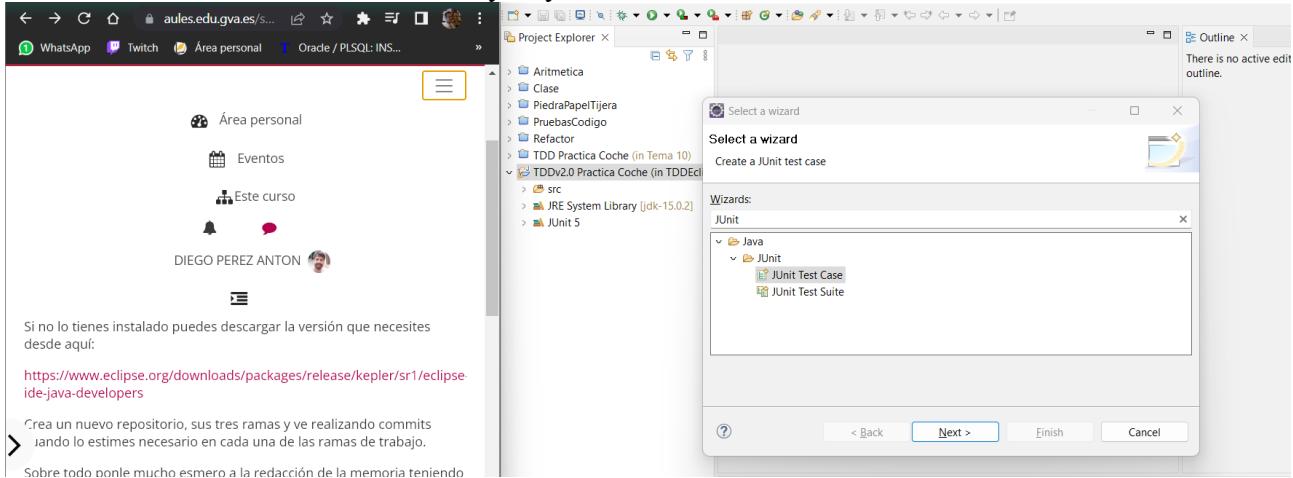
-Podemos ver que la librería ha sido añadida:



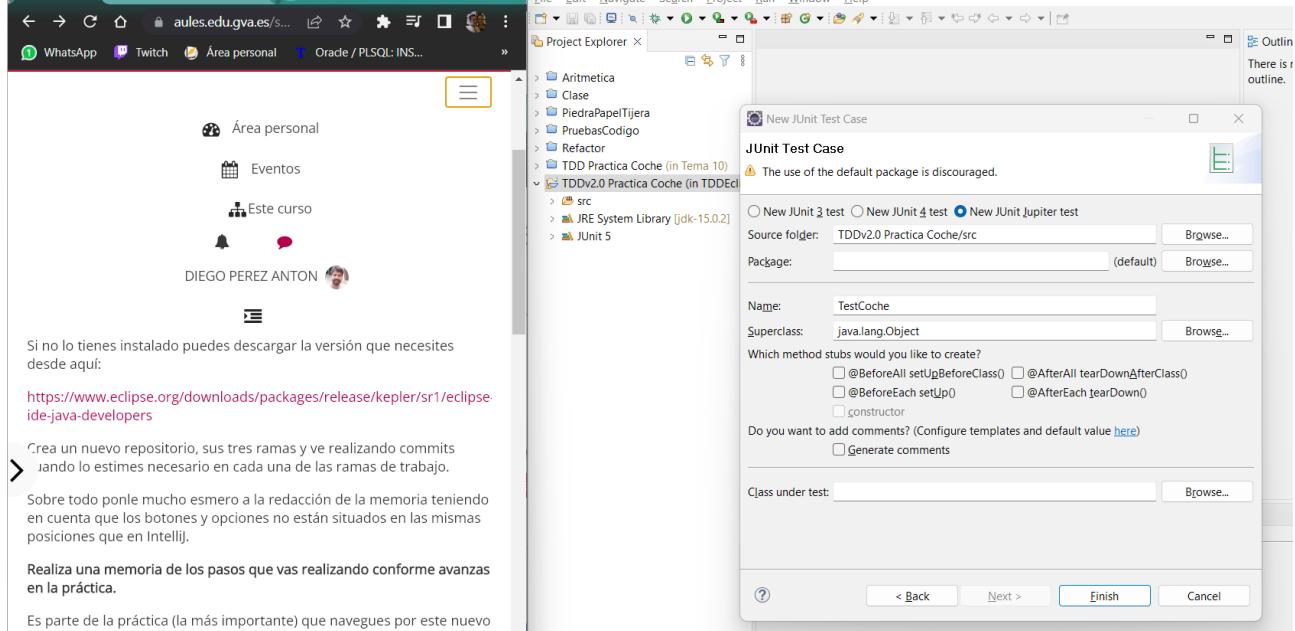
-Ahora creamos la nueva clase de prueba:
Botón derecho del ratón en el proyecto/new/Other (ya que no me da la opción de Junit)



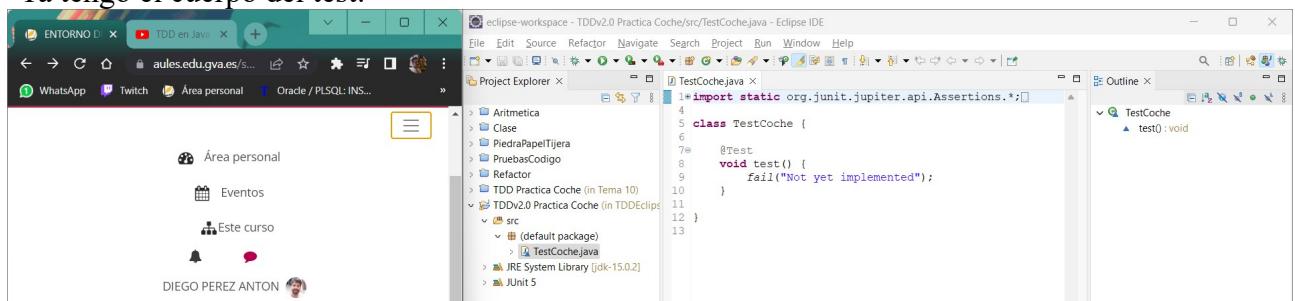
-Busco JUnit, Marco JUnit Test Case y doy al botón next:



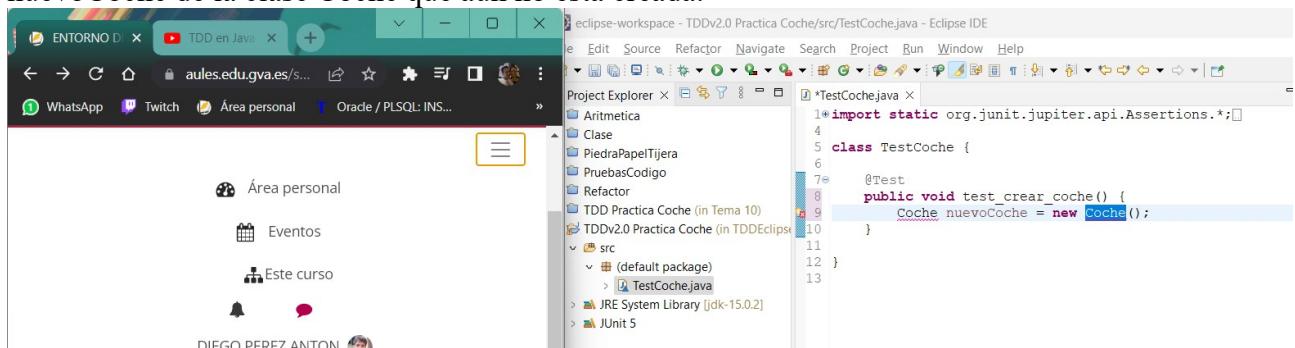
-En la siguiente ventana pongo como nombre TestCoche y botón finish:



-Ya tengo el cuerpo del test:



-Ahora creo mi primer test (public void test_crear_coche()), y a continuación creo el objeto nuevoCoche de la clase Coche que aun no está creada:



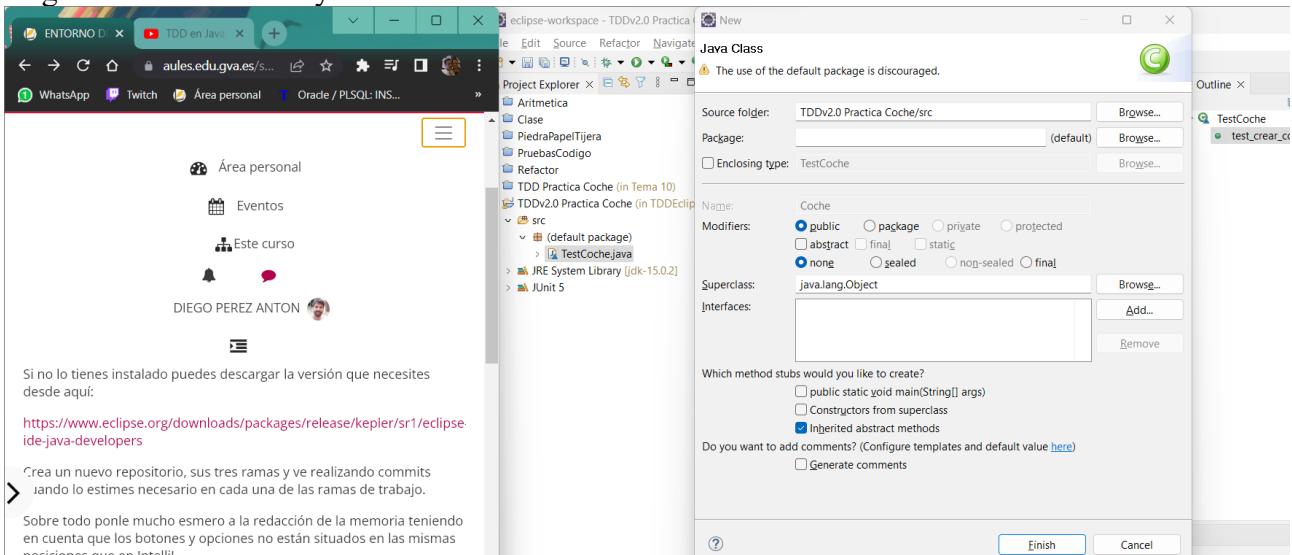
-Ahora me avisa de que la clase no está creada, click en bombilla y creo la clase Coche:

```

import static org.junit.jupiter.api.Assertions.*;
class TestCoche {
    @Test
    public void test_crear_coche() {
        Coche nuevoCoche = new Coche();
    }
}

```

-Siguiente ventana le doy al botón Finish:



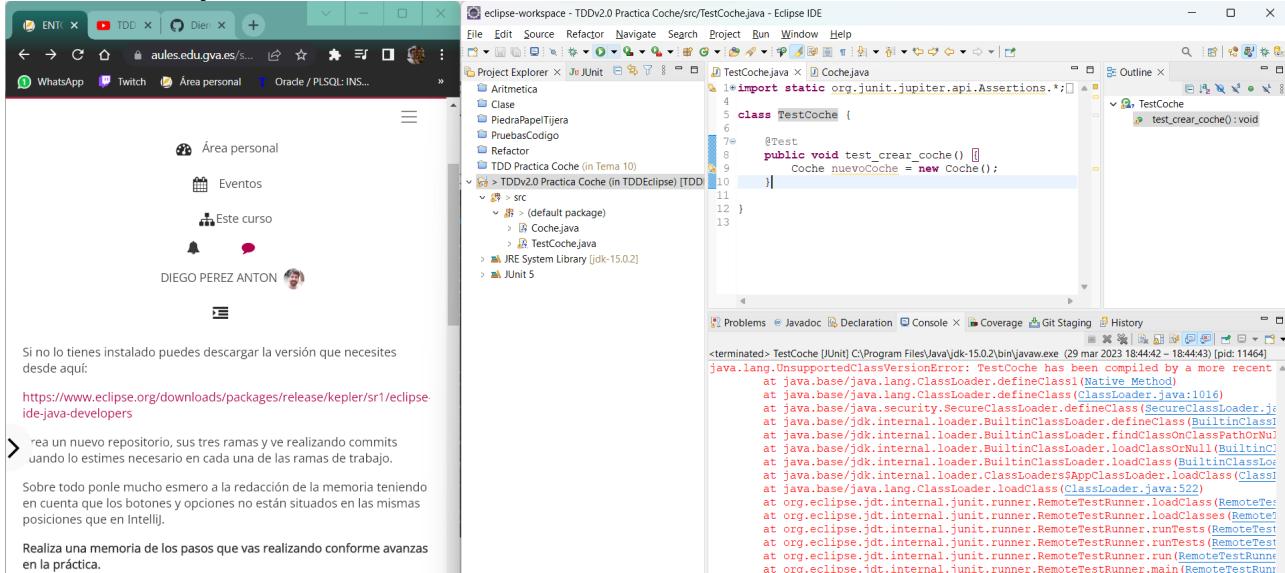
-Clase Coche creada:

```

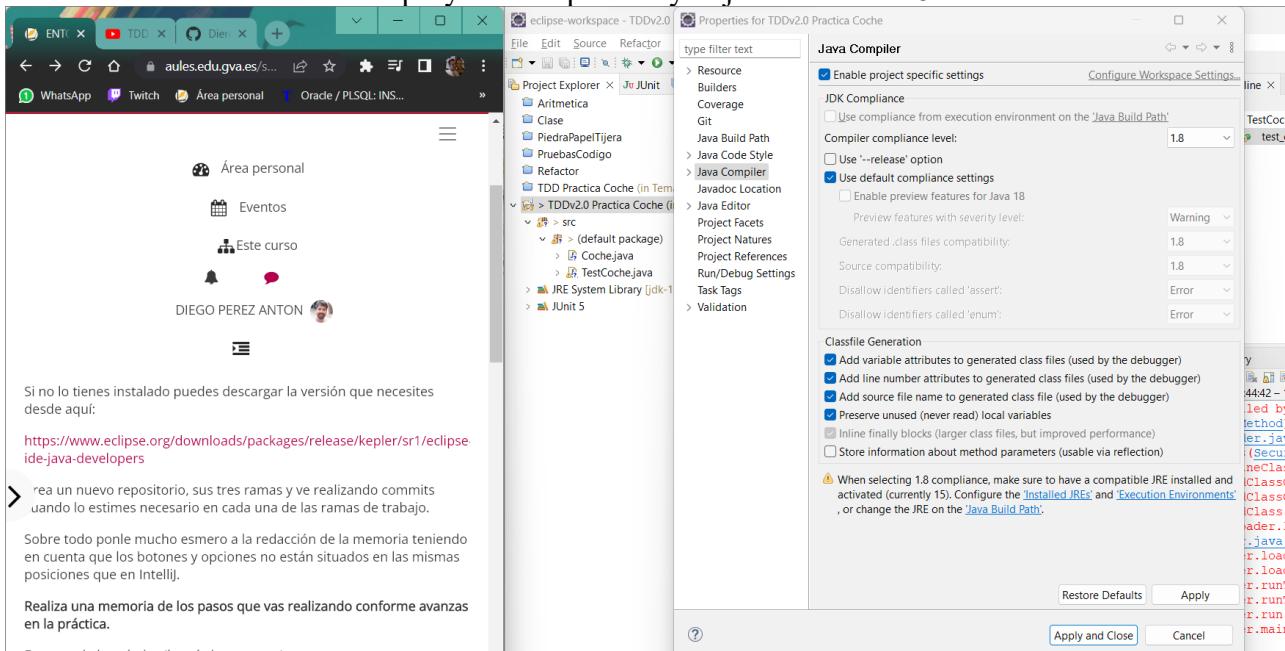
public class Coche {
}

```

-Ahora inicio mi primer test pero me da error por la incompatibilidad entre el jdk y JUnit, cambio a la versión 1.8 del jdk:



-Botón derecho del ratón en al proyecto/Properties/ y elijo la versión 1.8:



-Primera prueba realizada correctamente:

The screenshot shows the Eclipse IDE interface. On the left, there's a sidebar with links like WhatsApp, Twitch, Área personal, Este curso, and a user profile for DIEGO PEREZ ANTON. The main workspace shows a Java project named 'TestCoche'. In the 'Project Explorer' view, under 'JUnit', it says 'Runs: 1/1 Errors: 0 Failures: 0'. The code editor shows a single test method:

```
import static org.junit.jupiter.api.Assertions.*;
class TestCoche {
    @Test
    public void test_crear_coche() {
        Coche nuevoCoche = new Coche();
    }
}
```

 Below the code editor is the 'Run' view, which shows 'terminated> TestCoche [JUnit] C:\Program Files\Java\jdk-15.0.2\bin\javaw.exe (29 mar 2023 18:47:48 - 18:47:49) [pid: 2004]'. The status bar at the bottom indicates 'Si no lo tienes instalado puedes descargar la versión que necesites desde aquí: https://www.eclipse.org/downloads/packages/release/kepler/sr1/eclipse'.

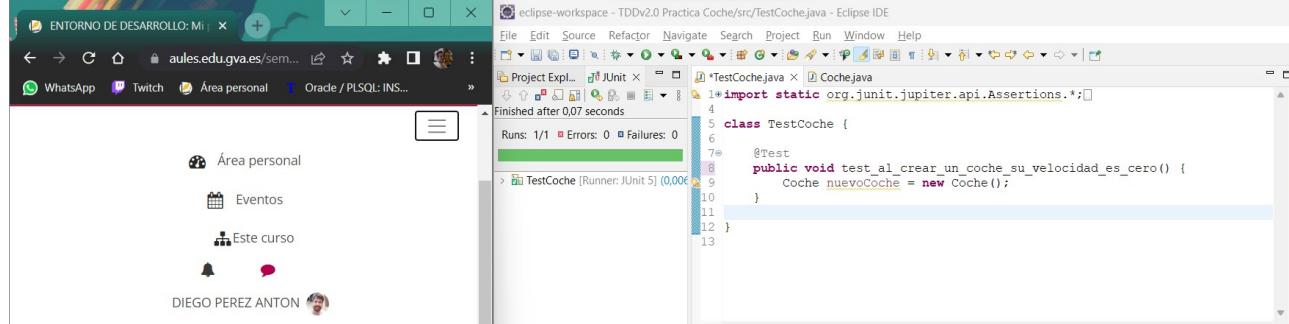
-Ahora en la pestaña Git Starting añado los archivos al Staged Changes, pongo una frase en Commit Message y doy al botón de Commit:

This screenshot shows the Eclipse IDE with the 'Git Staging' tab selected. The 'Unstaged Changes' section is empty. The 'Staged Changes' section lists five files: classpath, .gitignore, Coche.java - src, org.eclipse.jdt.core.prefs - settings, and TestCoche.java - src. To the right, the 'Commit Message' field contains the text 'Primer test superado'. The 'Author' field is set to 'Diego <dierez87@gmail.com>' and the 'Committer' field is also set to 'Diego <dierez87@gmail.com>'. At the bottom are 'Commit and Push' and 'Commit' buttons.

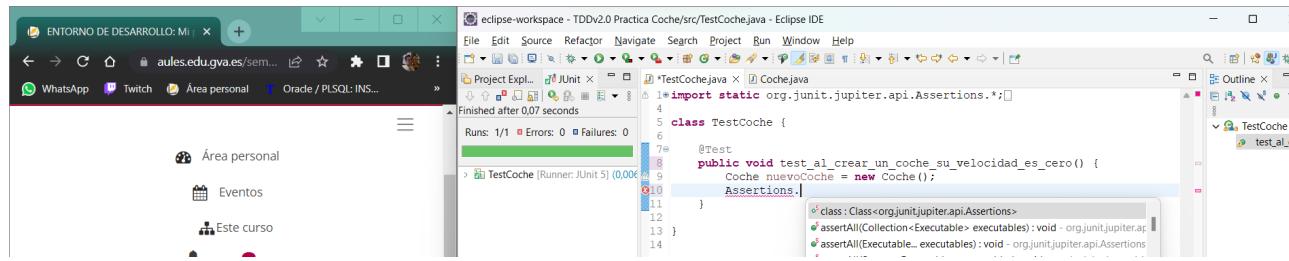
-Podemos ver que se ha realizado:

The screenshot shows the Eclipse IDE with the 'Git Repositories' tab selected. It displays a tree view of repositories. Under 'Local', there's a branch named 'main f6fad36 Primer test superado'. The code editor on the right shows the same Java test code as before. The status bar at the bottom indicates 'Si no lo tienes instalado puedes descargar la versión que necesites desde aquí: https://www.eclipse.org/downloads/packages/release/kepler/sr1/eclipse'.

-Ahora indico con test_al_crear_un_coche_su_velocidad_es_cero que la velocidad sea cero, para eso hay que indicarlo con assertions y lo importo dándole a la bombilla:

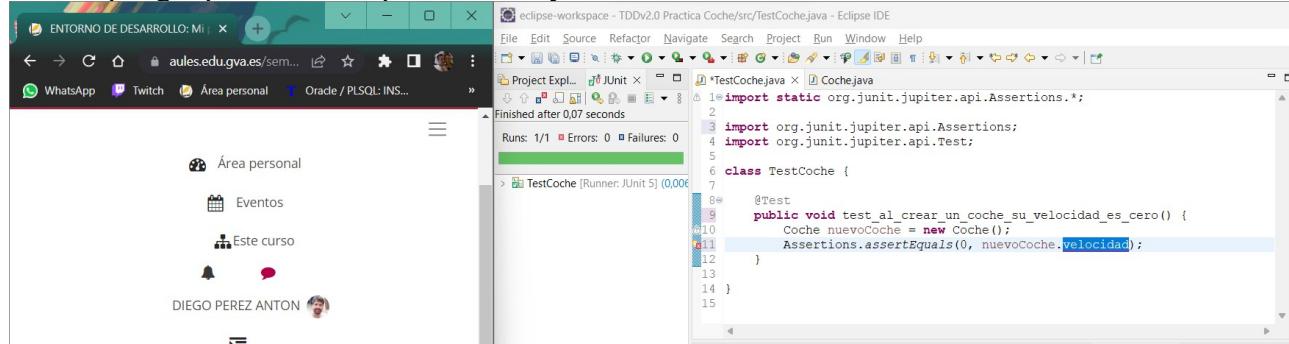


```
eclipse-workspace - TDDv2.0 Practica Coche/src/TestCoche.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Project Expl... JUnit
*TestCoche.java x Coche.java
1 import static org.junit.jupiter.api.Assertions.*;
2
3 class TestCoche {
4
5     @Test
6     public void test_al_crear_un_coche_su_velocidad_es_cero() {
7         Coche nuevoCoche = new Coche();
8         Assertions.assertAll(nuevoCoche,
9             () -> assertEquals(0, nuevoCoche.velocidad)
10        );
11    }
12 }
13
14 }
```



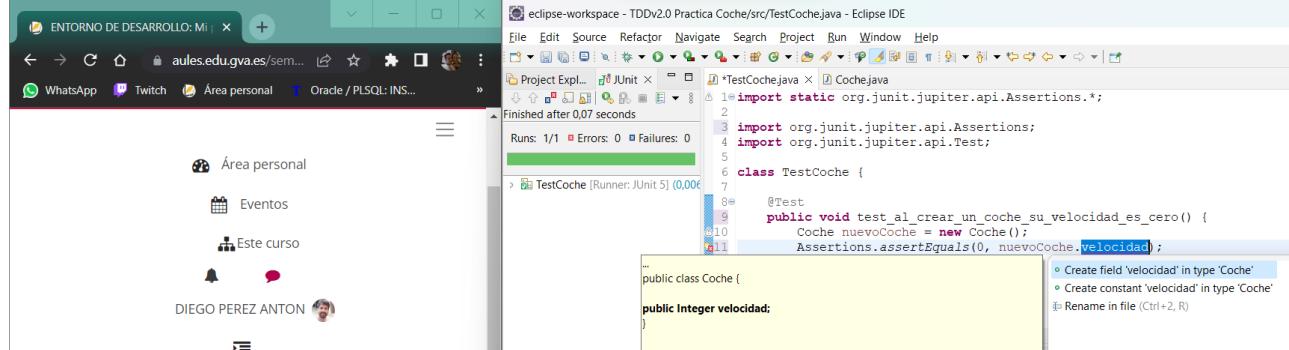
```
eclipse-workspace - TDDv2.0 Practica Coche/src/TestCoche.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Project Expl... JUnit
*TestCoche.java x Coche.java
1 import static org.junit.jupiter.api.Assertions.*;
2
3 class TestCoche {
4
5     @Test
6     public void test_al_crear_un_coche_su_velocidad_es_cero() {
7         Coche nuevoCoche = new Coche();
8         Assertions.assertAll(nuevoCoche,
9             () -> assertEquals(0, nuevoCoche.velocidad)
10        );
11    }
12 }
13
14 }
```

-Ahora pongo que el valor esperado sea 0 y el valor real será nuevoCoche.velocidad:



```
eclipse-workspace - TDDv2.0 Practica Coche/src/TestCoche.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Project Expl... JUnit
*TestCoche.java x Coche.java
1 import static org.junit.jupiter.api.Assertions.*;
2
3 import org.junit.jupiter.api.Assertions;
4 import org.junit.jupiter.api.Test;
5
6 class TestCoche {
7
8     @Test
9     public void test_al_crear_un_coche_su_velocidad_es_cero() {
10        Coche nuevoCoche = new Coche();
11        Assertions.assertEquals(0, nuevoCoche.velocidad);
12    }
13
14 }
```

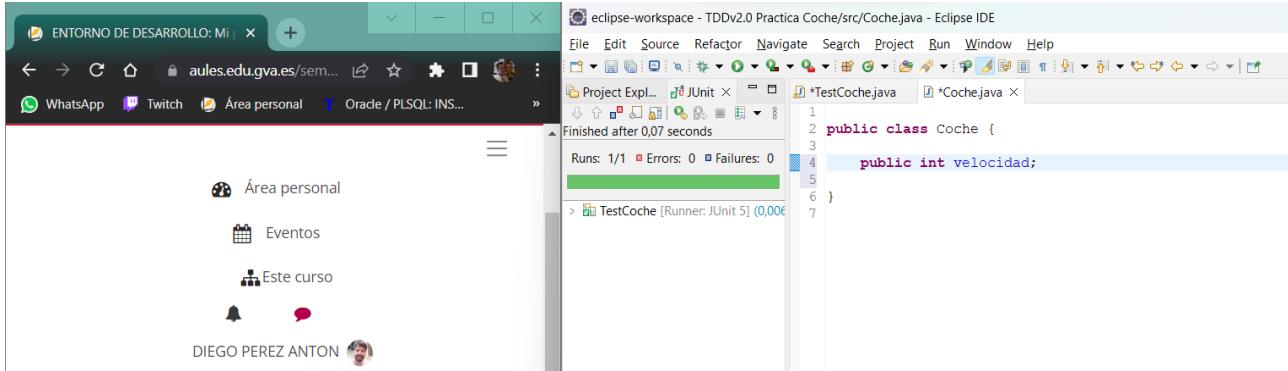
-Aparece en rojo velocidad porque no tenemos aun creado el símbolo, dispongo a crearlo:



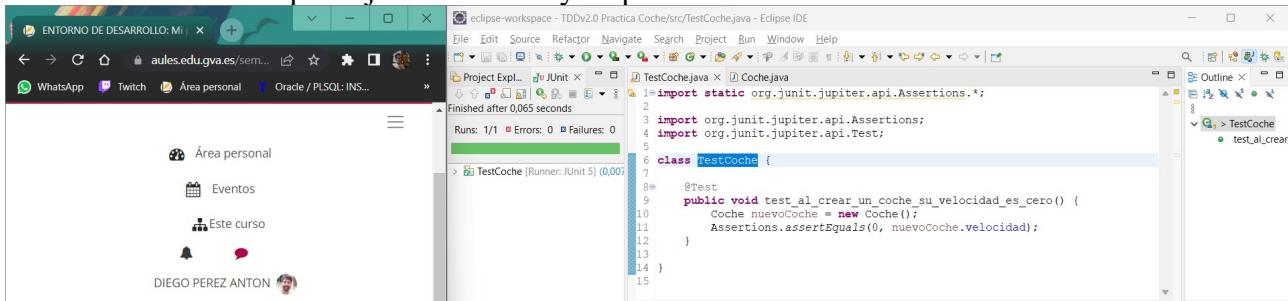
```
eclipse-workspace - TDDv2.0 Practica Coche/src/TestCoche.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Project Expl... JUnit
*TestCoche.java x Coche.java
1 import static org.junit.jupiter.api.Assertions.*;
2
3 import org.junit.jupiter.api.Assertions;
4 import org.junit.jupiter.api.Test;
5
6 class TestCoche {
7
8     @Test
9     public void test_al_crear_un_coche_su_velocidad_es_cero() {
10        Coche nuevoCoche = new Coche();
11        Assertions.assertEquals(0, nuevoCoche.velocidad);
12    }
13
14 }
```

```
public class Coche {
    public Integer velocidad;
}
```

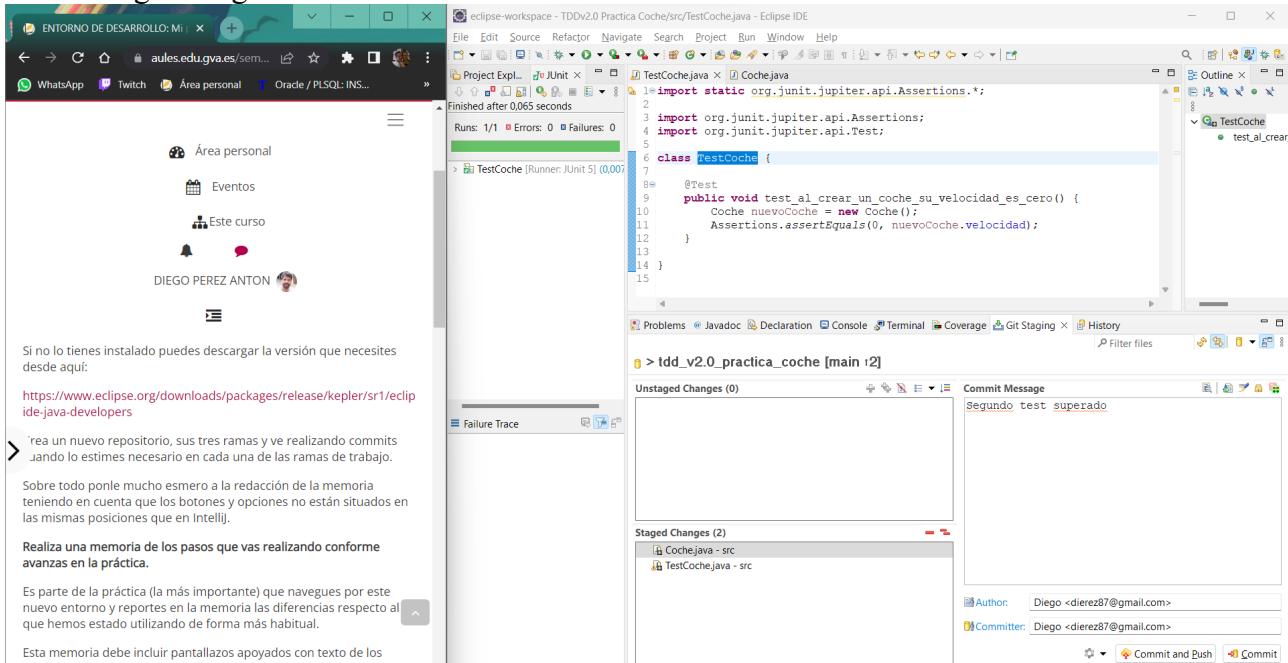
-Creado:



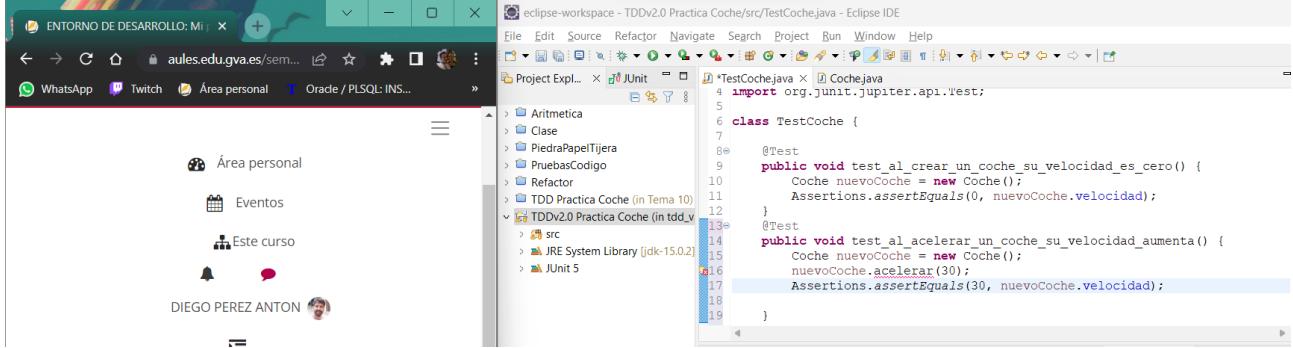
-Inicio dándole al run para ejecutar los test y ha pasado correctamente:



-Ahora hago el segundo commit de los cambios realizados:



-Creo otro test con public void test_al_acelerar_un_coche_su_velocidad_aumenta() le digo que nuevo.coche tenga la velocidad de 30 y la Aserción que tenga como velocidad 30:

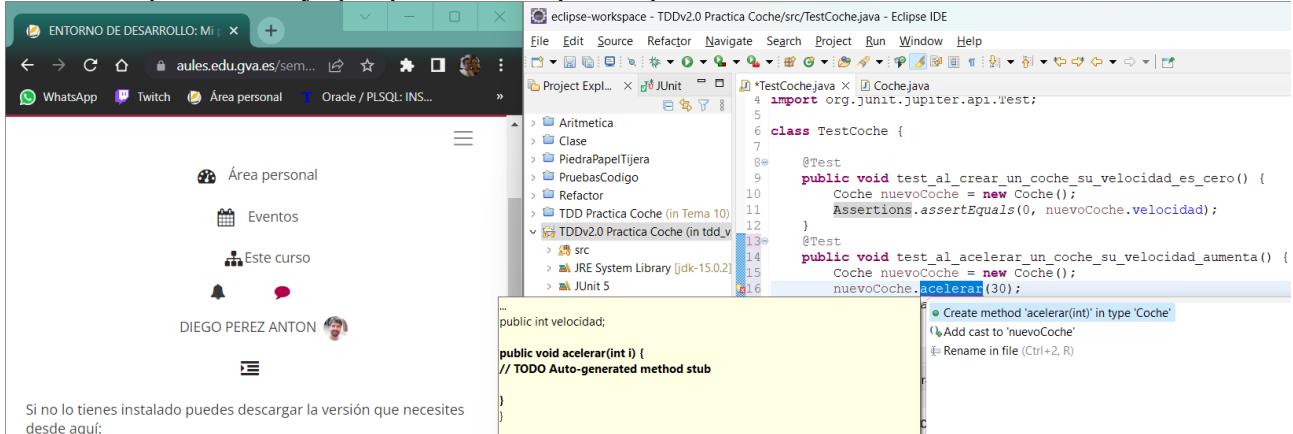


The screenshot shows the Eclipse IDE interface. On the left, there's a sidebar with various icons for WhatsApp, Twitch, Área personal, Oracle / PLSQL: INS..., Área personal, Eventos, Este curso, and a user profile for DIEGO PEREZ ANTON. The main workspace shows the Project Explorer and the Java Editor. In the Java Editor, the code for `TestCoche.java` is being edited:

```
import org.junit.jupiter.api.Test;
import static org.junit.jupiter.api.Assertions.assertEquals;

public class TestCoche {
    @Test
    public void test_al_crear_un_coche_su_velocidad_es_cero() {
        Coche nuevoCoche = new Coche();
        Assertions.assertEquals(0, nuevoCoche.velocidad);
    }
    @Test
    public void test_al_acelerar_un_coche_su_velocidad_aumenta() {
        Coche nuevoCoche = new Coche();
        nuevoCoche.acelerar(30);
        Assertions.assertEquals(30, nuevoCoche.velocidad);
    }
}
```

-Acelerar aparece en rojo porque no existe por lo que creo el método:



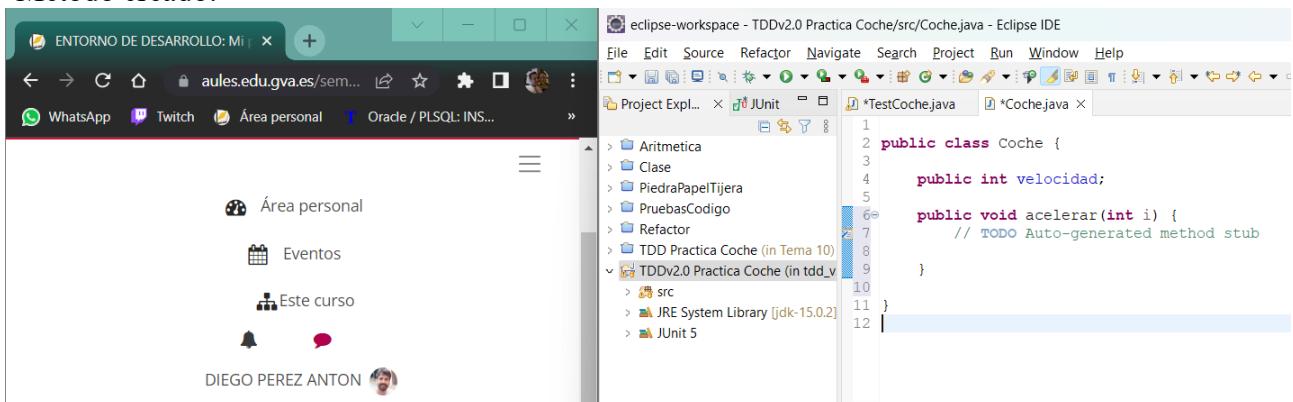
The screenshot shows the Eclipse IDE interface. The sidebar and project structure are identical to the previous screenshot. In the Java Editor, the code for `Coche.java` is being edited:

```
public int velocidad;
public void acelerar(int i) {
    // TODO Auto-generated method stub
}
```

A tooltip appears over the `acelerar` method, suggesting options: "Create method 'acelerar(int)' in type 'Coche'", "(+) Add cast to 'nuevoCoche'", and "Rename in file (Ctrl+2, R)".

Si no lo tienes instalado puedes descargar la versión que necesites desde aquí:

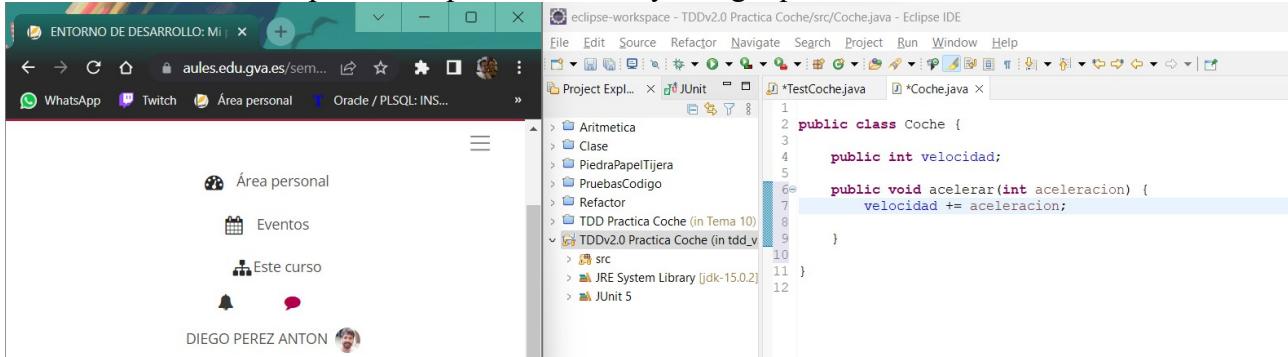
-Método creado:



The screenshot shows the Eclipse IDE interface. The sidebar and project structure remain the same. In the Java Editor, the code for `Coche.java` is now fully defined:

```
public class Coche {
    public int velocidad;
    public void acelerar(int i) {
        // TODO Auto-generated method stub
    }
}
```

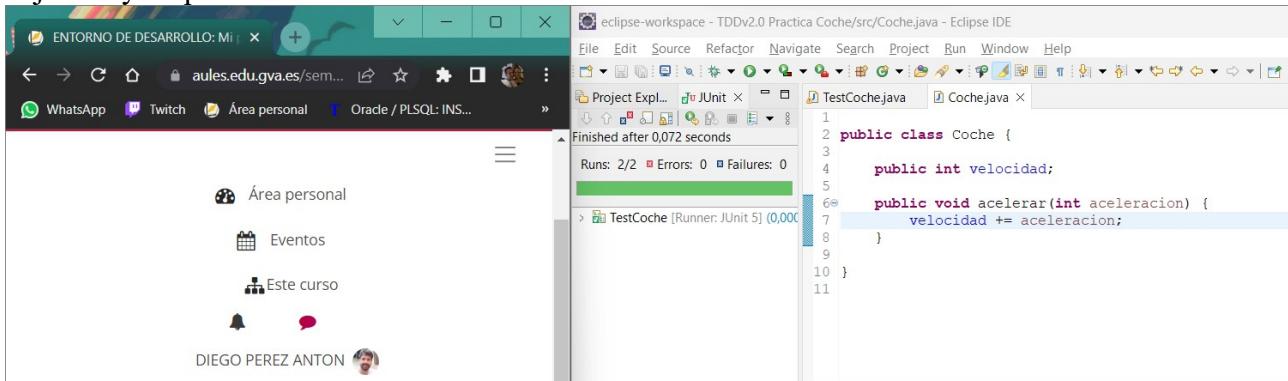
-Cambio el nombre del parámetro por aceleracion y le digo que velocidad aumente en aceleracion:



The screenshot shows the Eclipse IDE interface. On the left, there's a sidebar with links like 'Área personal', 'Eventos', 'Este curso', and notifications. The main workspace shows the 'Project Explorer' and 'JUnit' perspectives. In the 'Project Explorer', there are several projects and files listed. In the 'JUnit' perspective, two files are open: 'TestCoche.java' and 'Coche.java'. The code in 'Coche.java' is as follows:

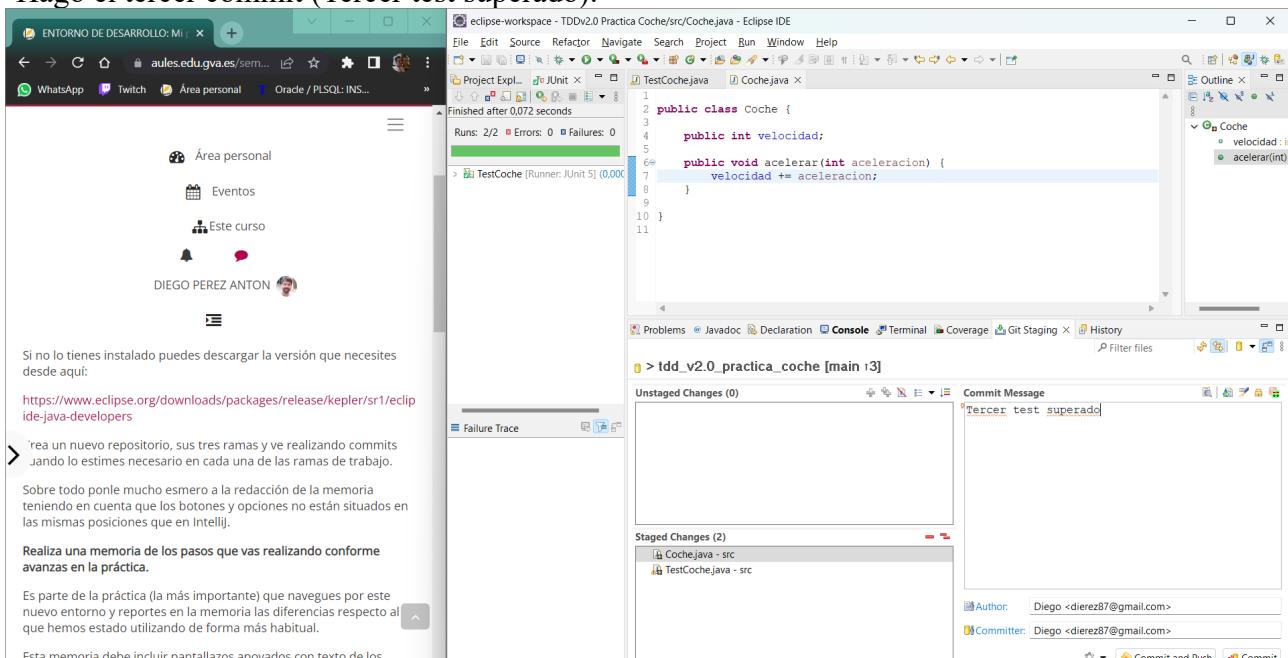
```
1 public class Coche {  
2     public int velocidad;  
3  
4     public void acelerar(int aceleracion) {  
5         velocidad += aceleracion;  
6     }  
7 }  
8  
9  
10 }  
11  
12 }
```

-Ejecuto y he pasado el test correctamente:



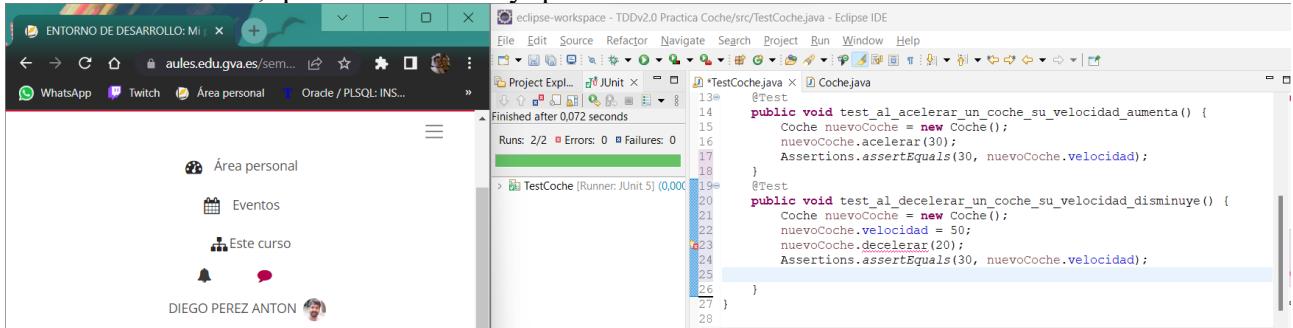
The screenshot shows the Eclipse IDE interface. The sidebar and project structure are similar to the previous screenshot. In the 'JUnit' perspective, the results of the test run are displayed. It shows 'Runs: 2/2 Errors: 0 Failures: 0'. The code in 'Coche.java' remains the same as in the previous screenshot.

-Hago el tercer commit (Tercer test superado):



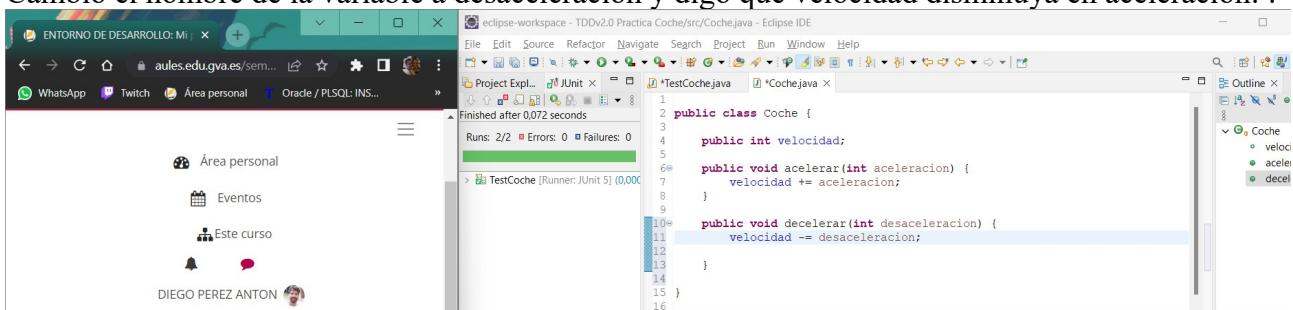
The screenshot shows the Eclipse IDE interface with the 'Git Staging' view open. The sidebar and project structure are consistent. The 'Git Staging' view shows the commit message 'Tercer test superado' and lists staged changes: 'Coche.java - src' and 'TestCoche.java - src'. The commit details show the author and committer as 'Diego <dierez87@gmail.com>'.

-Ahora creo otro método que sea decelerar, primero indico que la velocidad inicial sea 50, porque si parte de 0 y tiene que desacelerar no queremos que sea un numero negativo la velocidad. Indico la velocidad inicial 50, que desacelere 20 y que la velocidad final sea 30:



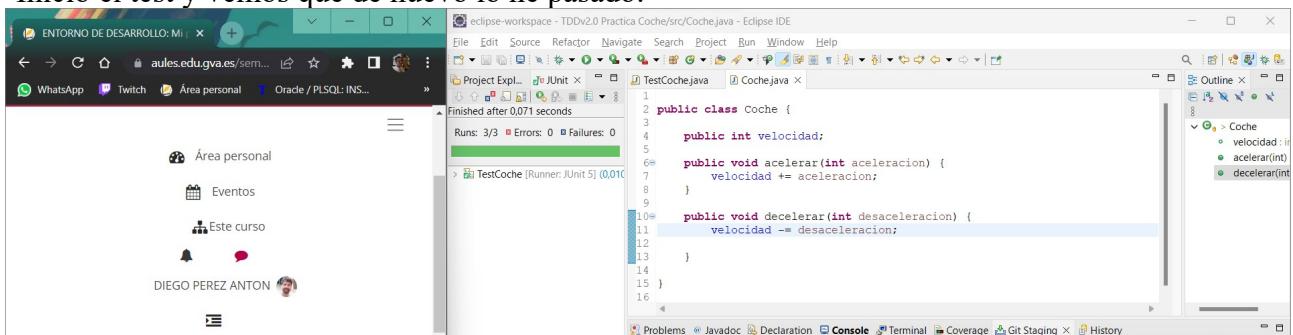
The screenshot shows the Eclipse IDE interface. On the left, there's a sidebar with links to WhatsApp, Twitch, Área personal, Oracle / PLSQL: INS..., and DIEGO PEREZ ANTON. The main workspace displays two Java files: `testCoche.java` and `Coche.java`. `testCoche.java` contains JUnit test cases for the `Coche` class. The first test, `test_al_acelerar_un_coche_su_velocidad_aumenta()`, creates a new `Coche` object, sets its initial velocity to 30, and then calls `acelerar(30)`. It uses `Assertions.assertEquals(30, nuevoCoche.velocidad)` to verify the final velocity. The second test, `test_al_decelerar_un_coche_su_velocidad_disminuye()`, creates a new `Coche` object with an initial velocity of 50, calls `decelerar(20)`, and uses `Assertions.assertEquals(30, nuevoCoche.velocidad)` to verify the final velocity. The `Coche.java` file defines the `Coche` class with methods `velocidad`, `acelerar(int aceleracion)`, and `decelerar(int desaceleracion)`.

-desacelerar aparece en rojo porque aun no he creado el método, por lo que dispongo a crearlo. Cambio el nombre de la variable a desaceleracion y digo que velocidad disminuya en aceleracion: :



This screenshot shows the Eclipse IDE after renaming the variable. The `decelerar` method in `Coche.java` is now named `desacelerar`. The code runs successfully, and the test results show 2/2 runs completed with 0 errors and 0 failures.

-Inicio el test y vemos que de nuevo lo he pasado:



The screenshot shows the Eclipse IDE after running the tests again. The test results now show 3/3 runs completed with 0 errors and 0 failures, indicating that the changes have been successful.

-Hago el cuarto commit (Cuarto test superado):

The screenshot shows the Eclipse IDE interface. On the left, there's a sidebar with links like 'Área personal', 'Eventos', 'Este curso', and a user profile for 'DIEGO PEREZ ANTON'. The main workspace shows Java code for a 'Coche' class and its test class 'TestCoche'. The code defines methods for accelerating and decelerating. In the bottom right, the 'Git Staging' view is open, showing a staged change for 'Coche.java - src' and 'TestCoche.java - src'. A commit message is being typed: 'Cuarto test superado'. Below it, author and committer details are listed: 'Author: Diego <dielez87@gmail.com>' and 'Committer: Diego <dielez87@gmail.com>'. The status bar at the bottom indicates 'Unstaged Changes (0)'.

-Ahora voy a crear otro test que sea public void

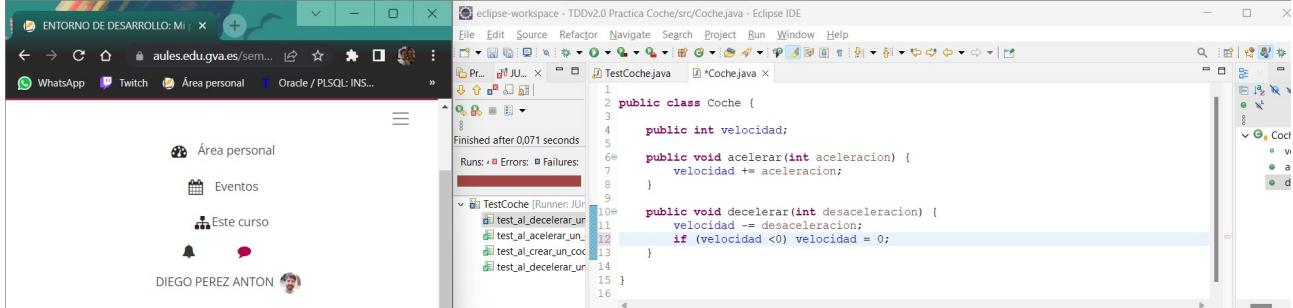
test_al_decelerar_un_coche_su_velocidad_no_puede_ser_menor_que_cero() e indico que la velocidad inicial sea 50, que desacelera 80 y que la velocidad final sea 0 (que se quede parado):

The screenshot shows the Eclipse IDE interface. The code editor displays a Java test class 'TestCoche.java' with two test methods: 'test_al_decelerar_un_coche_su_velocidad_disminuye()' and 'test_al_decelerar_un_coche_su_velocidad_no_puede_ser_menor_que_cero()'. The second test method is currently selected. The code sets up a new car with an initial velocity of 50, decelerates by 80, and asserts that the final velocity is 0. The status bar at the bottom indicates 'Runs: 2 Errors: 0 Failures: 0'.

-Podemos ver que el test falla porque se esperaba 0, pero obtiene -30:

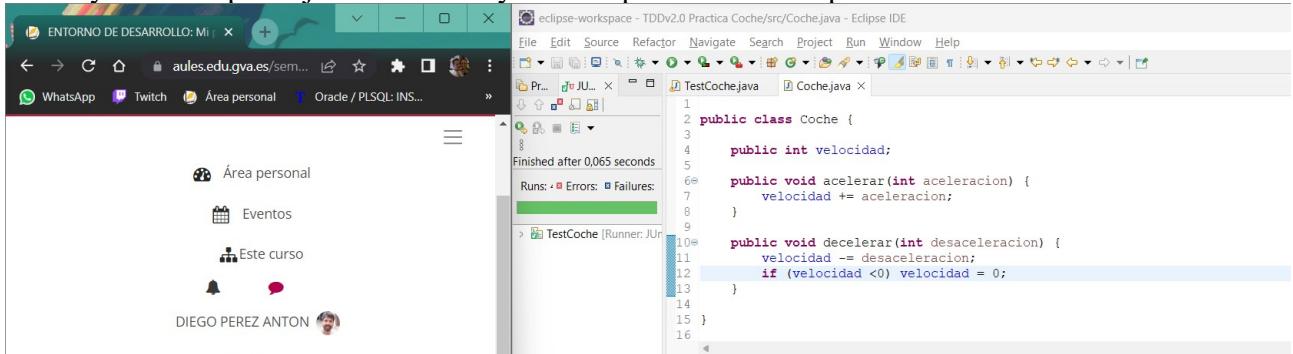
The screenshot shows the Eclipse IDE interface. The code editor displays the same Java test class 'TestCoche.java'. The 'test_al_decelerar_un_coche_su_velocidad_no_puede_ser_menor_que_cero()' test has failed. The status bar at the bottom indicates 'Runs: 2 Errors: 1 Failures: 0'. The 'Problems' view shows an error for the failing test: 'java.lang.AssertionError: expected [0] but found [-30]'.

-Voy a la clase Coche y ponemos que en desacelerar si la velocidad es menor que cero, velocidad sea igual que cero:



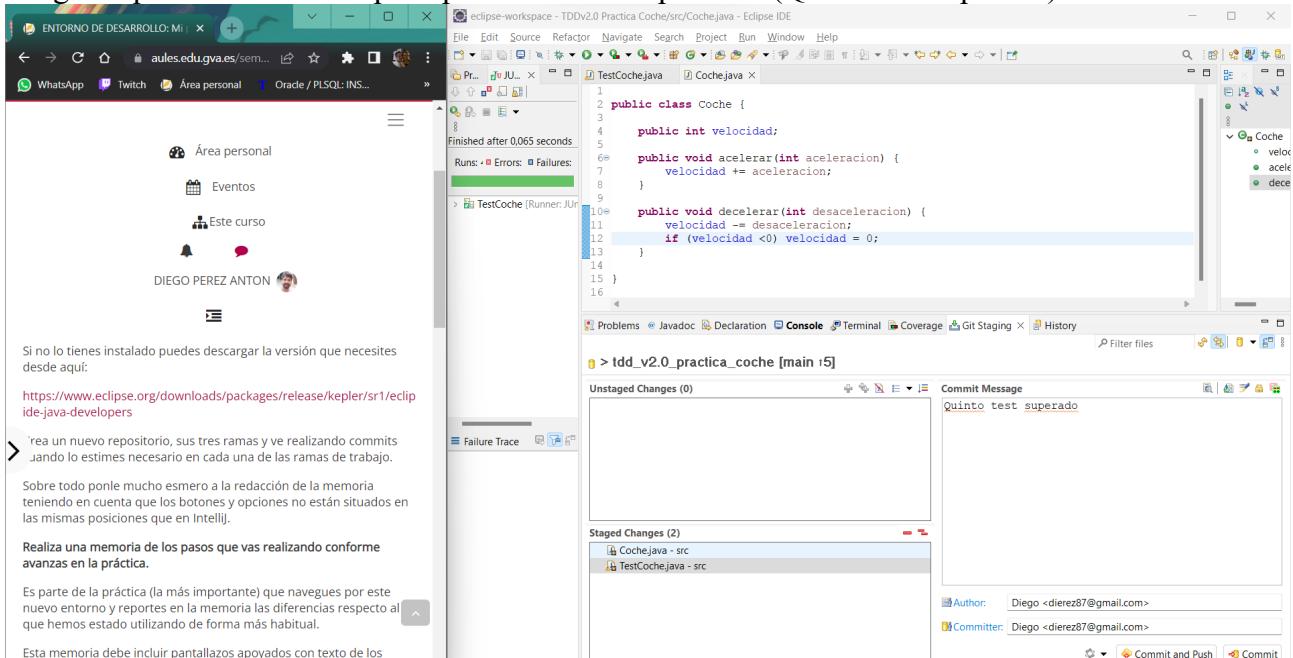
```
public class Coche {
    public int velocidad;
    public void acelerar(int aceleracion) {
        velocidad += aceleracion;
    }
    public void decelerar(int desaceleracion) {
        velocidad -= desaceleracion;
        if (velocidad < 0) velocidad = 0;
    }
}
```

-Le doy a Iniciar para ejecutar los test y vemos que todos los he pasado:



```
public class Coche {
    public int velocidad;
    public void acelerar(int aceleracion) {
        velocidad += aceleracion;
    }
    public void decelerar(int desaceleracion) {
        velocidad -= desaceleracion;
        if (velocidad < 0) velocidad = 0;
    }
}
```

-Hago el quinto commit and push para subirlo al repositorio(Quinto test superado):



Si no lo tienes instalado puedes descargar la versión que necesites desde aquí:
<https://www.eclipse.org/downloads/packages/release/kepler/sr1/eclipse-ide-java-developers>

crea un nuevo repositorio, sus tres ramas y ve realizando commits dando lo estimes necesario en cada una de las ramas de trabajo.

Sobre todo ponle mucho esmero a la redacción de la memoria teniendo en cuenta que los botones y opciones no están situados en las mismas posiciones que en IntelliJ.

Realiza una memoria de los pasos que vas realizando conforme avanzas en la práctica.

Es parte de la práctica (la más importante) que navegues por este nuevo entorno y reportes en la memoria las diferencias respecto a que hemos estado utilizando de forma más habitual.

Esta memoria debe incluir pantallazos apoyados con texto de los

Unstaged Changes (0)

Staged Changes (2)

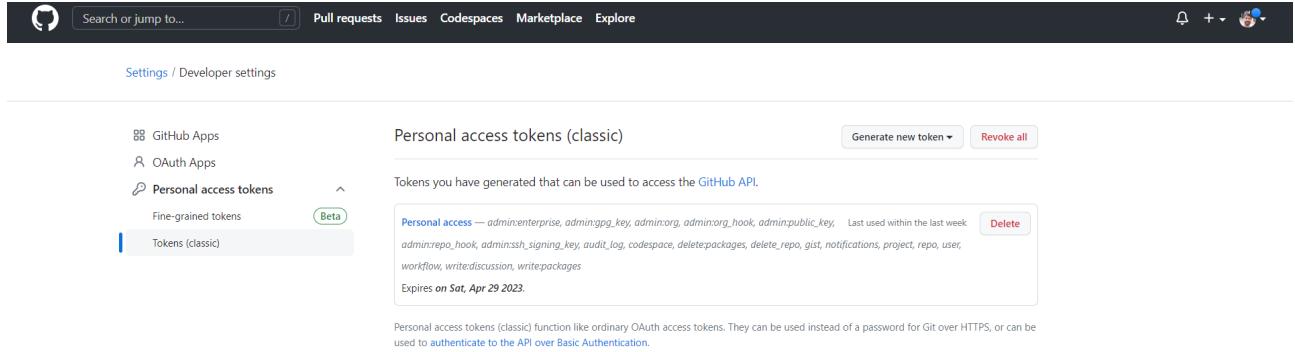
Coche.java - src
TestCoche.java - src

Commit Message
Quinto test superado

Author: Diego <dieriez87@gmail.com>
Committer: Diego <dieriez87@gmail.com>

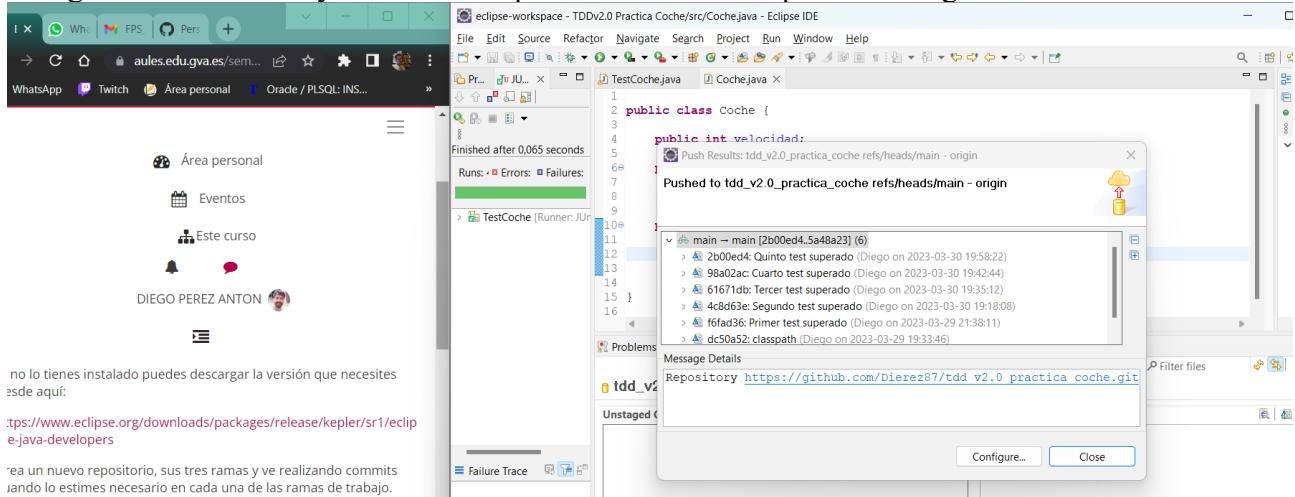
Commit and Push Commit

-Como me pide credenciales, creo un token en gibhub, dentro de settings/Developer settings creo el token:



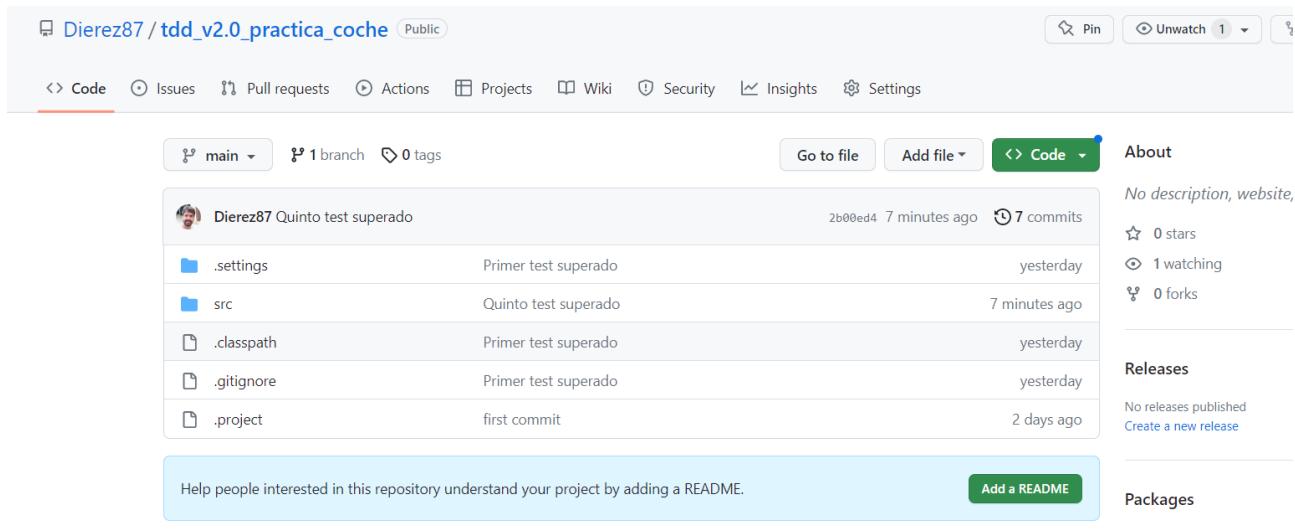
The screenshot shows the GitHub developer settings page. In the sidebar, under 'Personal access tokens', the 'Tokens (classic)' tab is selected. A single token is listed with the name 'Personal access' and a detailed scope: 'admin:enterprise, admin:pgp_key, admin:org, admin:org_hook, admin:public_key, admin:repo_hook, admin:ssh_signing_key, audit_log, codespace, delete_packages, delete_repo, gist, notifications, project, repo, user, workflow, write:discussion, write:packages'. It was last used within the last week and expires on Saturday, April 29, 2023. There are 'Generate new token' and 'Revoke all' buttons at the top right.

-Pongo las credenciales y nos avisa de que se ha subido al repositorio de github:



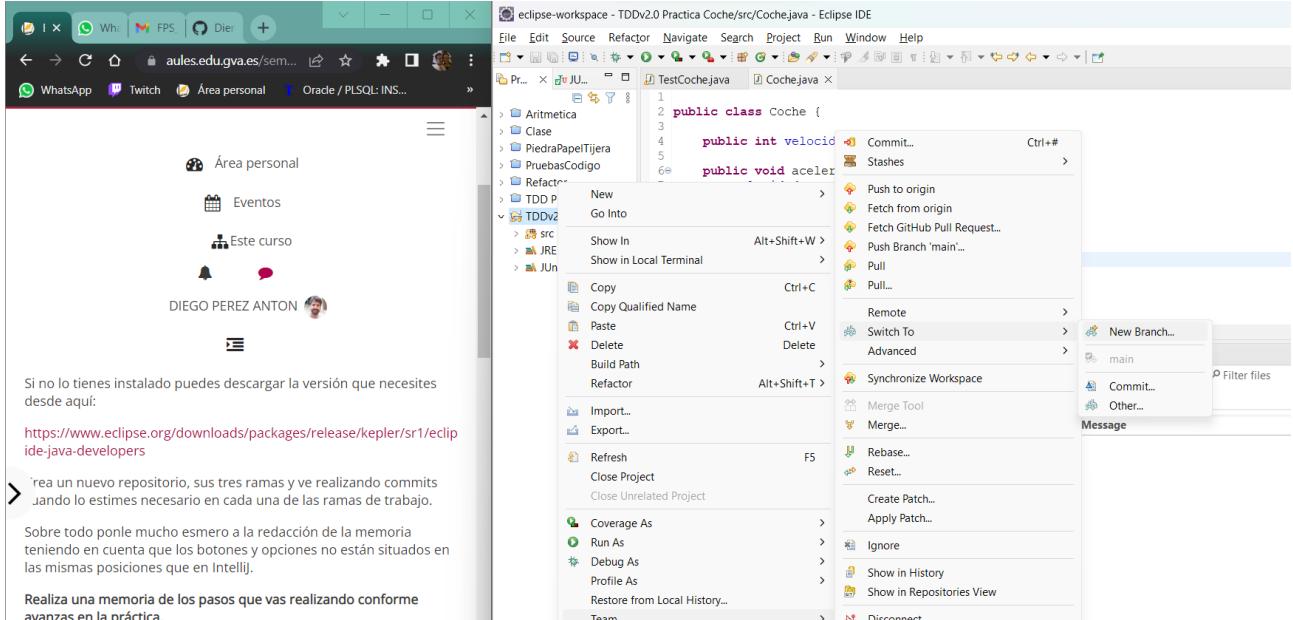
The screenshot shows the Eclipse IDE interface with a Java project named 'eclipse-workspace - TDDV2.0 Practica Coche/src/Coche.java'. A terminal window in the background shows the command 'git push' has been run successfully. The Eclipse interface also displays a GitHub pull request titled 'Pushed to tdd_v2.0_practica_coche refs/heads/main - origin' with a commit message 'Quinto test superado' and a timestamp of 'Finished after 0.065 seconds'.

-Se ha subido correctamente:

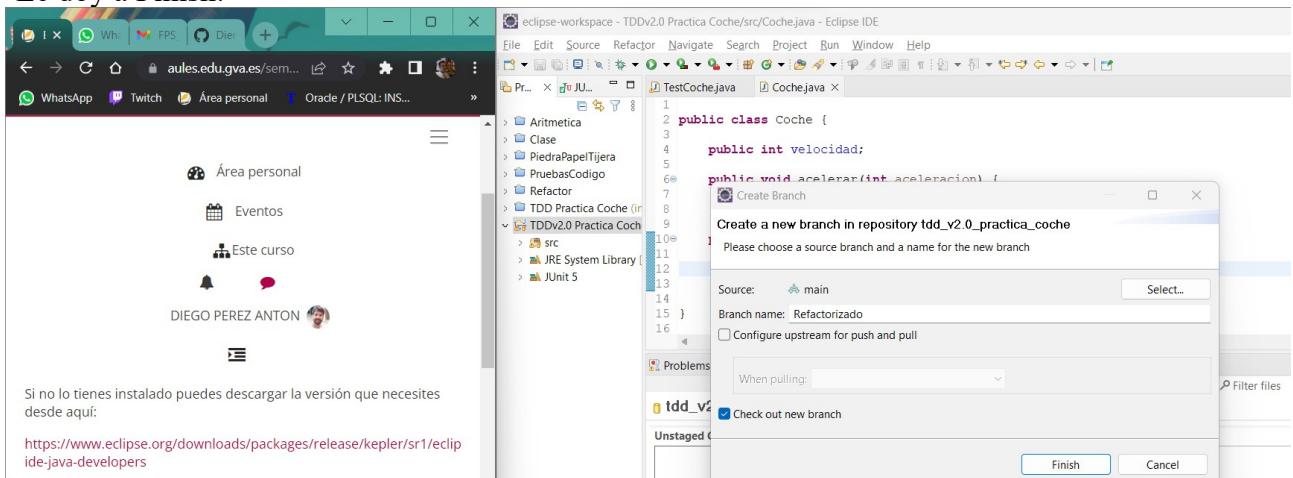


The screenshot shows the GitHub repository page for 'tdd_v2.0_practica_coche'. The repository has 1 branch ('main') and 0 tags. The commit history shows several commits from user 'Dierez87' with messages like 'Primer test superado', 'Quinto test superado', and 'Tercer test superado'. The latest commit was made 7 minutes ago. The repository has 0 stars, 1 watching, and 0 forks. There are sections for 'About', 'Releases', and 'Packages', all of which are currently empty.

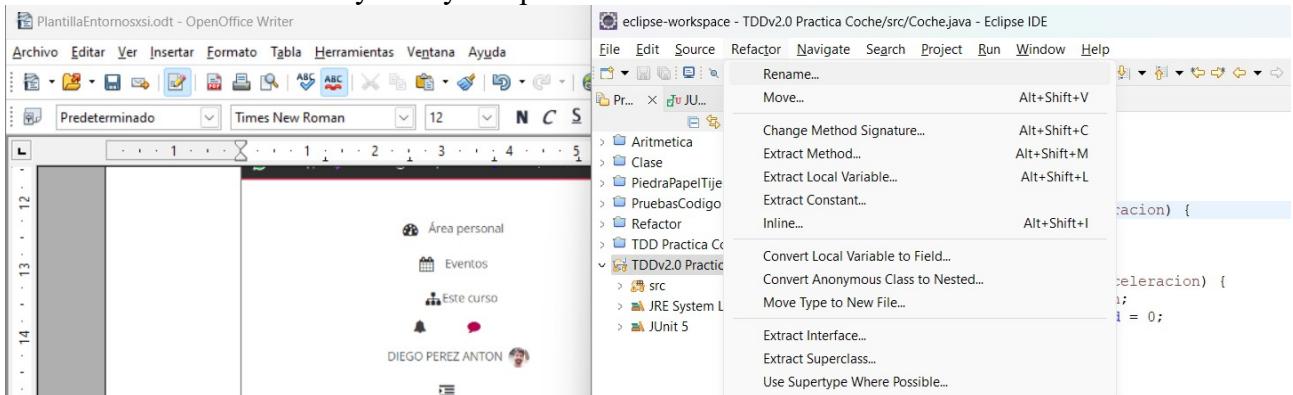
-Ahora creo la rama Refactorizado, botón derecho del ratón en el proyecto/Team/Swith To/New Branch



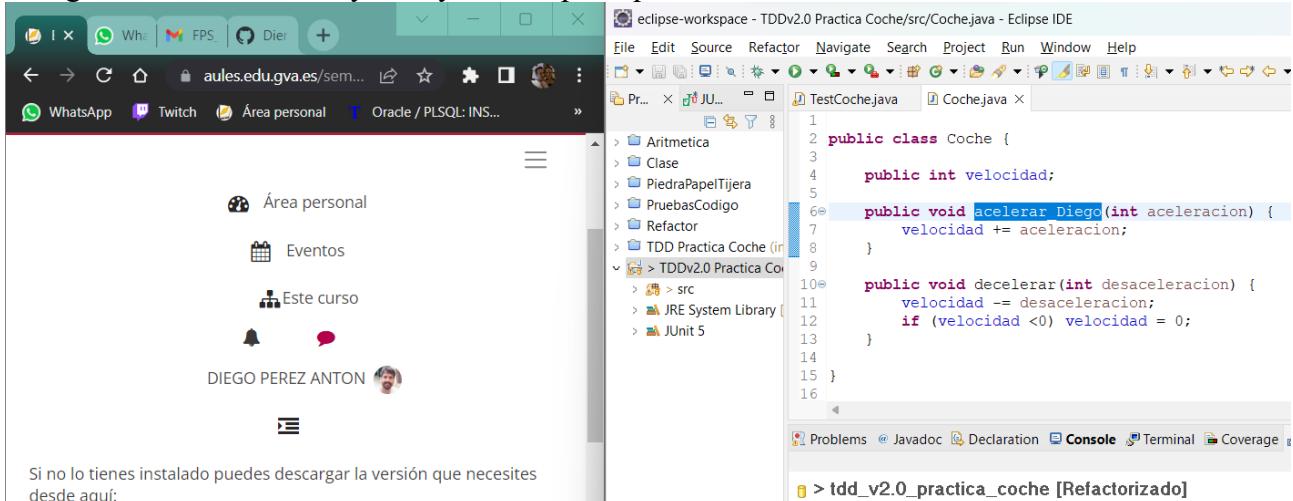
-Le doy a Finish:



-Señalo el método acelerar y le doy a la pestaña Refactor/Rename:



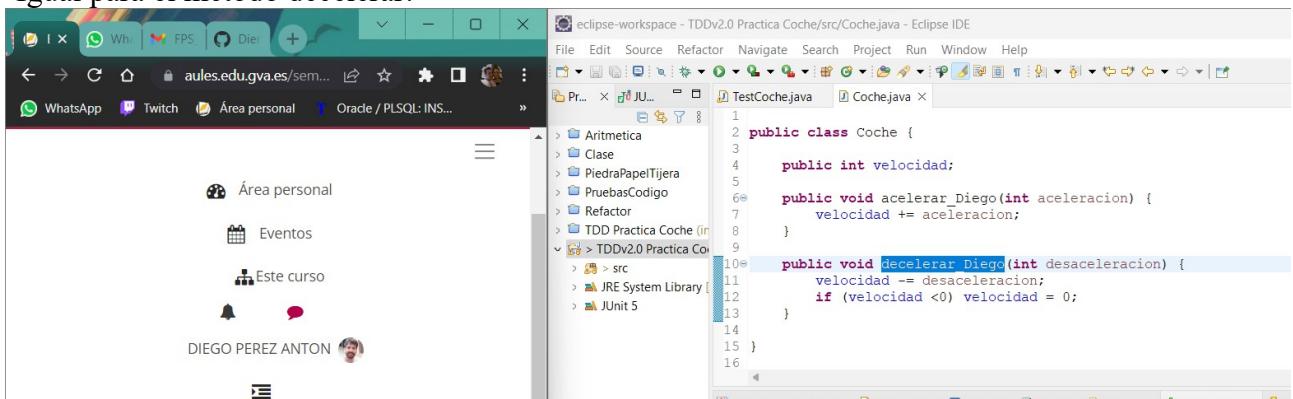
-Pongo mi nombre al final y le doy a Enter para que refactorice:



```
1 public class Coche {
2     public int velocidad;
3
4     public void acelerar_Diego(int aceleracion) {
5         velocidad += aceleracion;
6     }
7
8     public void decelerar(int desaceleracion) {
9         velocidad -= desaceleracion;
10        if (velocidad < 0) velocidad = 0;
11    }
12
13 }
14
15 }
```

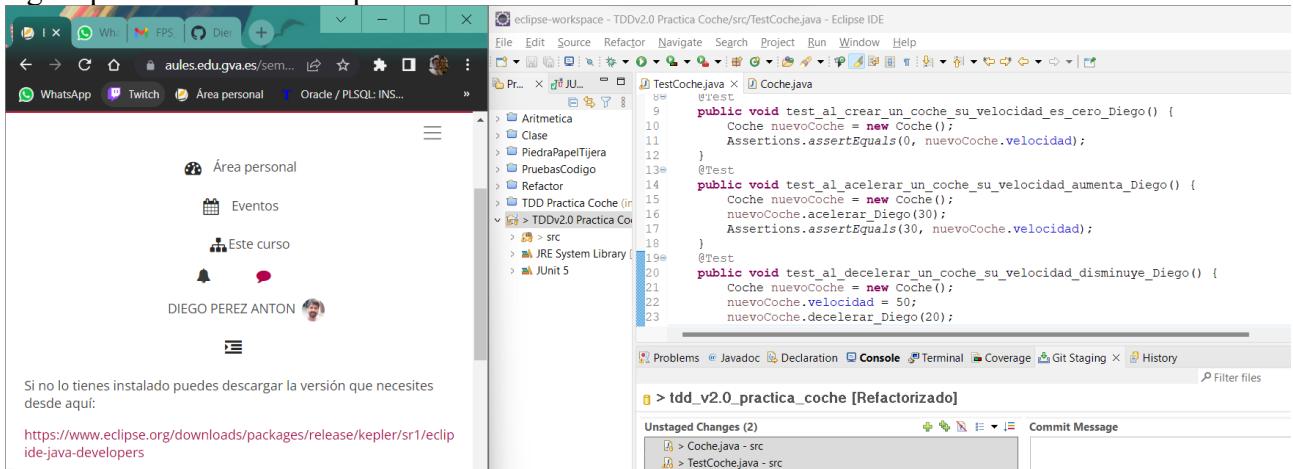
Si no lo tienes instalado puedes descargar la versión que necesites desde aquí:

-Igual para el método decelerar:



```
1 public class Coche {
2     public int velocidad;
3
4     public void acelerar_Diego(int aceleracion) {
5         velocidad += aceleracion;
6     }
7
8     public void Decelerar_Diego(int desaceleracion) {
9         velocidad -= desaceleracion;
10        if (velocidad < 0) velocidad = 0;
11    }
12
13 }
14
15 }
```

-Igual para los métodos de prueba:

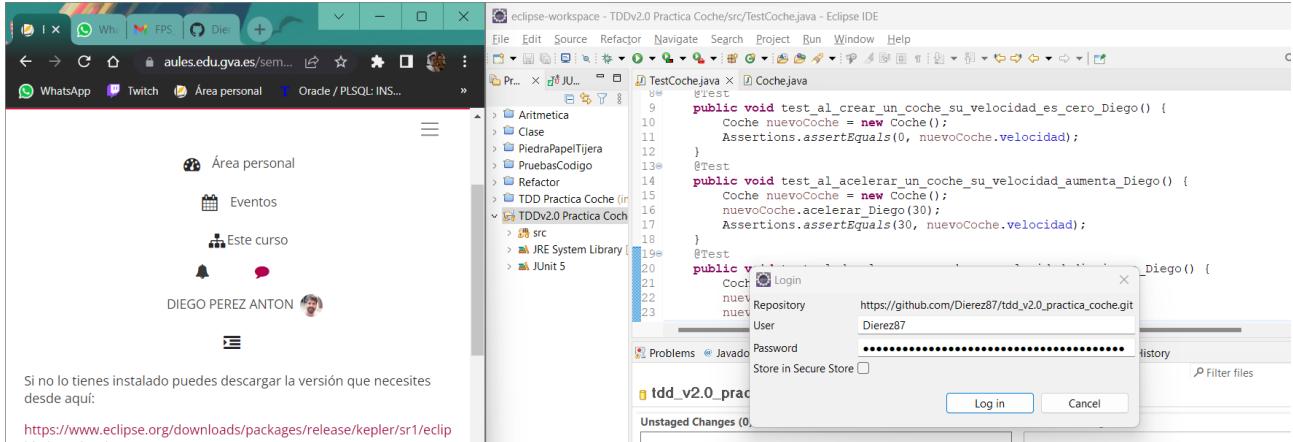


```
9
10    @Test
11    public void test_al_crear_un_coche_su_velocidad_es_cero_Diego() {
12        Coche nuevoCoche = new Coche();
13        Assertions.assertEquals(0, nuevoCoche.velocidad);
14    }
15
16    @Test
17    public void test_al_acelerar_un_coche_su_velocidad_aumenta_Diego() {
18        Coche nuevoCoche = new Coche();
19        nuevoCoche.acelerar_Diego(30);
20        Assertions.assertEquals(30, nuevoCoche.velocidad);
21    }
22
23    @Test
24    public void test_al_decelerar_un_coche_su_velocidad_disminuye_Diego() {
25        Coche nuevoCoche = new Coche();
26        nuevoCoche.velocidad = 50;
27        nuevoCoche.decelerar_Diego(20);
28    }
29
30 }
```

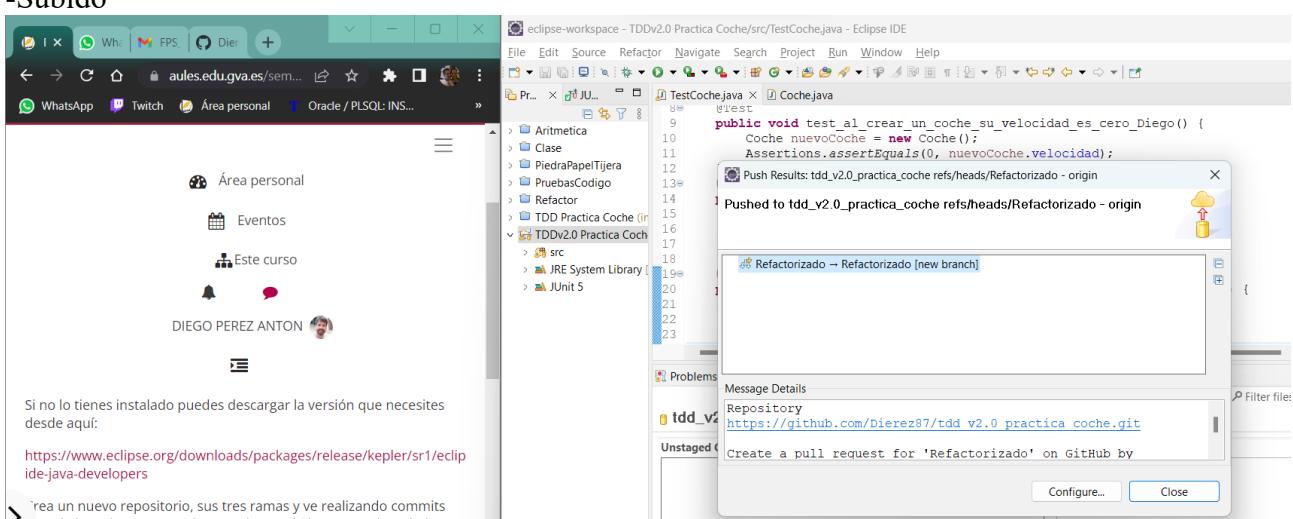
Si no lo tienes instalado puedes descargar la versión que necesites desde aquí:

<https://www.eclipse.org/downloads/packages/release/kepler/sr1/eclipse-ide-java-developers>

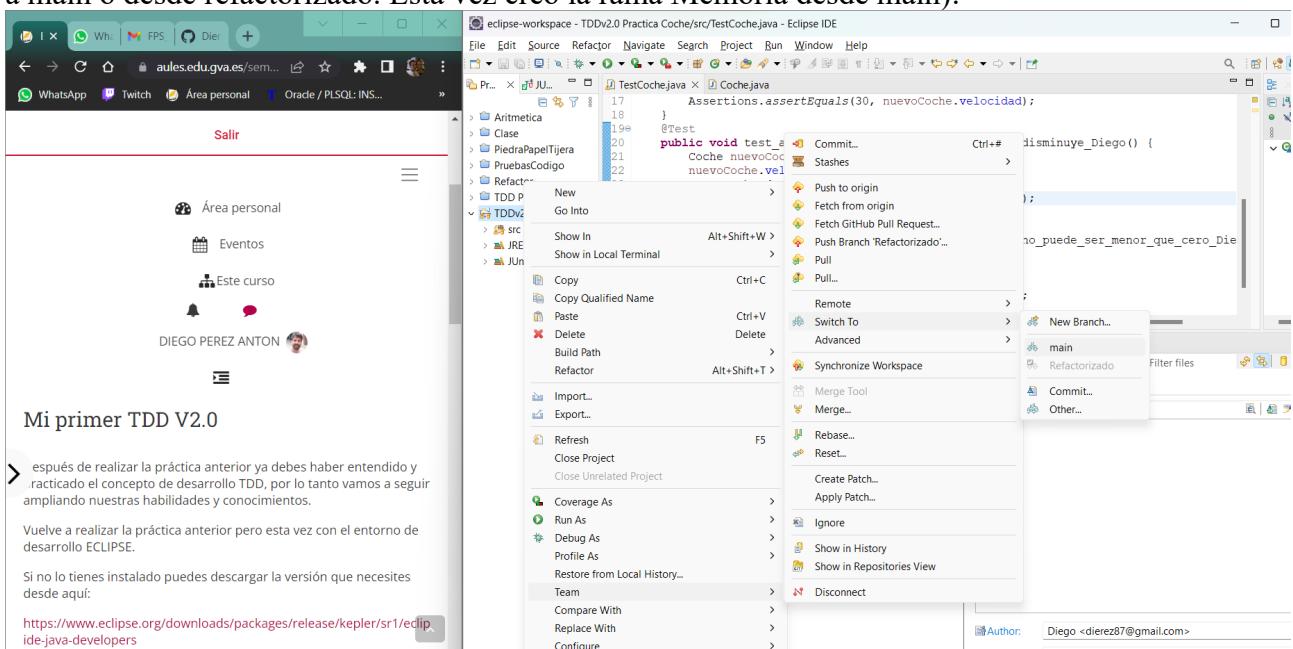
-Hago commit push para subir la rama al repositorio de github metiendo primero mis credenciales con el token creado:

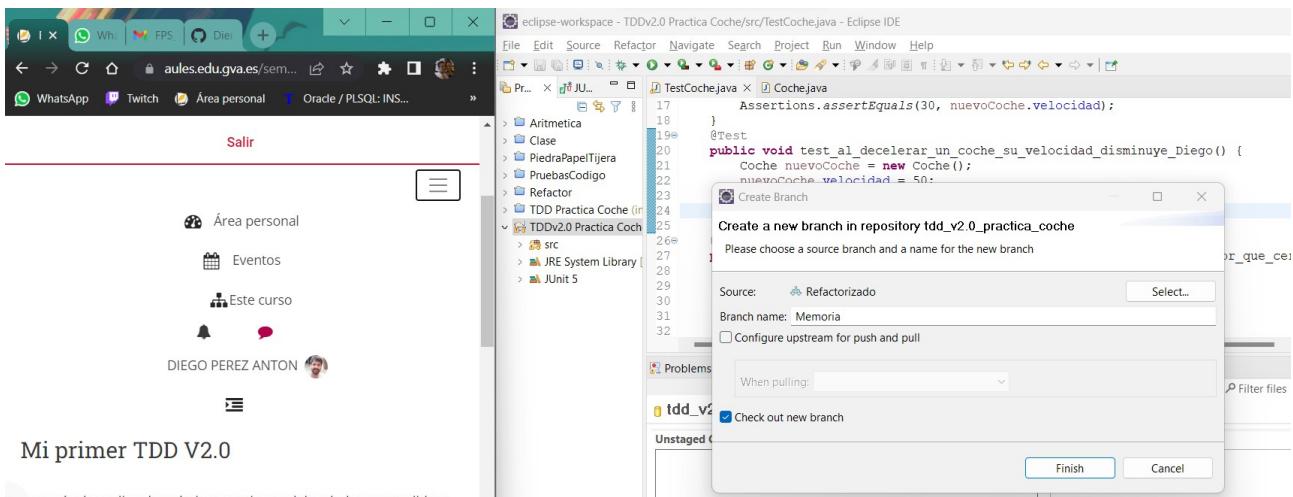


-Subido

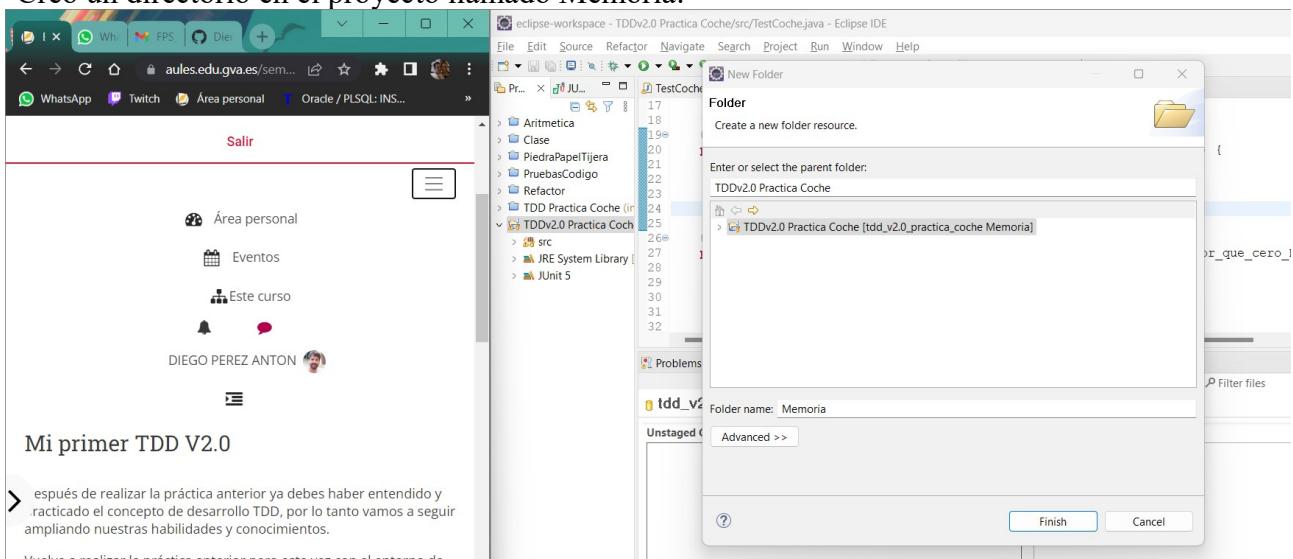


-Creo y me muevo a la rama Memoria del mismo modo que la rama refactorizado(no dice si volver a main o desde refactorizado. Esta vez creo la rama Memoria desde main):





-Cree un directorio en el proyecto llamado Memoria:



-Arrastro el trabajo realizado por Diego Pérez Antón al directorio Memoria

-Por último hago un commit and push con el comentario de Rama memoria.