

2025

TRABAJO FINAL

INTEGRACIÓN CONTINUA EN EL DESARROLLO ÁGIL
ANTORANZ ESTEBAN FRANCISCO JAVIER

Índice de contenido

URLS 3

SOLUCIONES A APARTADOS 3

 PUNTO 1 3

 PUNTO 2 4

 PUNTO 3 11

 PUNTO 4 12

 PUNTO 5 14

WORKFLOW FINAL 22

Índice de imágenes

| | |
|--|----|
| Ilustración 1 Nombre de los commits. | 3 |
| Ilustración 2 Credenciales github | 4 |
| Ilustración 3 Fichero main.yaml..... | 5 |
| Ilustración 4 Creación un Actor Status 200 | 5 |
| Ilustración 5 Listar Actores..... | 6 |
| Ilustración 6 Obtener Actor por Id | 6 |
| Ilustración 7 Crear Pelicula Status 200..... | 7 |
| Ilustración 8 Listar películas. | 8 |
| Ilustración 9 Obtener película ID | 9 |
| Ilustración 10 Inscribir Actor en una Peli | 9 |
| Ilustración 11 Consultar Actor Con película | 10 |
| Ilustración 12 Consultar película con Actor | 10 |
| Ilustración 13 Refactorización de Código MileStone..... | 11 |
| Ilustración 14 Commit 0963498 | 12 |
| Ilustración 15 Commit 943699c..... | 12 |
| Ilustración 16 Crear Actor con Fecha de Muerte | 13 |
| Ilustración 17 Consultar Actor con Fecha de Muerte | 13 |
| Ilustración 18 Milestone Apartado 5..... | 14 |
| Ilustración 19 Nuevo job stage | 15 |
| Ilustración 20 Commit f0adf23 | 15 |
| Ilustración 21 Crear Actor en Pre | 16 |
| Ilustración 22 Listar Actor en Pre | 16 |
| Ilustración 23 Crear Actor Producción | 17 |
| Ilustración 24 Listado Actores Producción | 17 |
| Ilustración 25 Commit 202befb | 18 |
| Ilustración 26 Commit 05aecb1 | 18 |
| Ilustración 27 Commit 91159d7e..... | 19 |
| Ilustración 28 Commit 8184571 | 19 |
| Ilustración 29 Commit 65518c3..... | 20 |
| Ilustración 30 Nueva Regla limitar críticas a 5..... | 21 |
| Ilustración 31 Pantalla general SonarQ | 21 |
| Ilustración 32 Pantalla Detalle Issues | 22 |
| Ilustración 33 Workflow Último Commit..... | 22 |

URLS

- 1. Moviecards: [fjavierae/moviecards](#)
- 2. Moviecards-service: [fjavierae/moviecards-service](#)
- 3. URL Pro Moviecards: [FichasPeliculasApp | Aplicación de gestión de fichas de películas](#)
- 4. URL Pre Moviecards: [FichasPeliculasApp | Aplicación de gestión de fichas de películas](#)
- 5. URL Moviecards-service: [moviecards-service-antoranz.azurewebsites.net](#)

SOLUCIONES A APARTADOS

PUNTO 1

- 1) Realizar la práctica del tema 5 de la asignatura para la creación de la aplicación moviecards. La aplicación debe quedar funcionando en [https://moviecards.apellido.azurewebsites.net](#), donde “apellido” es el apellido del alumno.

Se puede revisar en el github de moviecards los commits realizados según las instrucciones de la practica realizada mayormente en clase.

| | | | |
|---|----------|--|--|
| Creado trabajo qa | e98a359 | | |
| Antoranz committed on Dec 20, 2024 · 4 / 4 | | | |
| Creado trabajo qa | 57bf8ad | | |
| Antoranz committed on Dec 20, 2024 · 3 / 4 | | | |
| Creado trabajo qa | 41ab7b6 | | |
| Antoranz committed on Dec 20, 2024 · 1 / 4 | | | |
| Pruebas sin fallos | c78e619 | | |
| Antoranz committed on Dec 20, 2024 · 4 / 4 | | | |
| Fallo forzado en prueba unitaria | b1cefeff | | |
| Antoranz committed on Dec 20, 2024 · 1 / 4 | | | |
| Trabajo test creado | 1ce19a8 | | |
| Antoranz committed on Dec 20, 2024 · 4 / 4 | | | |
| Trabajo test creado | f36228d | | |
| Antoranz committed on Dec 20, 2024 · 1 / 4 | | | |
| Trabajo test creado | b9f9aec | | |
| Antoranz committed on Dec 20, 2024 | | | |
| Error de compilacion corregido | 854b59b | | |
| Antoranz committed on Dec 20, 2024 · 4 / 4 | | | |
| Error de compilación forzado | d87c2cb | | |
| Antoranz committed on Dec 20, 2024 · 0 / 4 | | | |
| Creado main.yaml | ebf7a76 | | |
| Antoranz committed on Dec 20, 2024 · 4 / 4 | | | |
| Commit inicial | dfdcF88 | | |
| Antoranz committed on Dec 20, 2024 | | | |

Ilustración 1 Nombre de los commits.

PUNTO 2

- 2) Crear un nuevo repositorio llamado moviecards-service con el código del servicio y desplegarlo en Azure. Debe quedar funcionando en <https://moviecards-service-apellido.azurewebsites.net>, donde “apellido” es el apellido del alumno. Para este repositorio no es necesario crear ningún proyecto asociado al repositorio, ni tampoco milestones ni issues.

Para este ejercicio se siguieron los pasos de la práctica para crear el servicio en azure y se crearon los secretos en el repositorio adecuados y los referenciamos en el fichero “.yaml”.

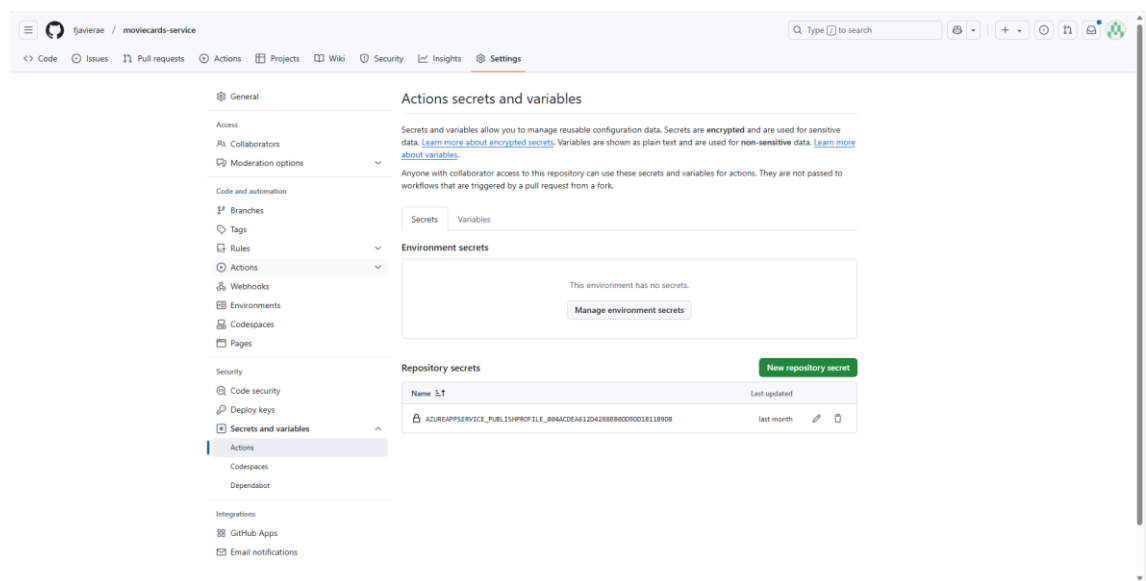


Ilustración 2 Credenciales github

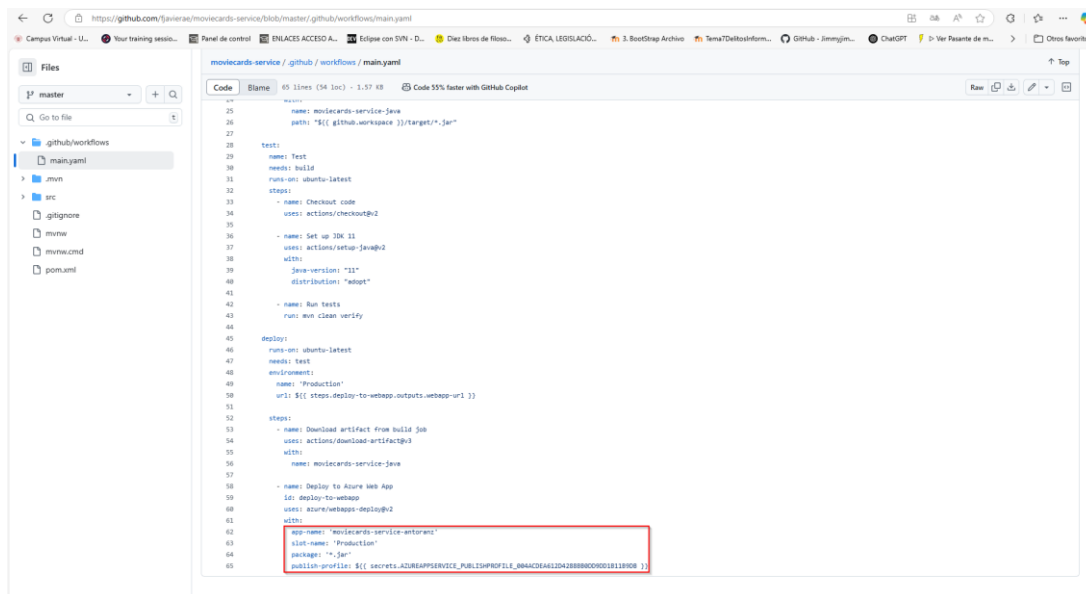


Ilustración 3 Fichero main.yml

Para comprobar el resultado del apartado realizaremos una serie de pruebas en postman:

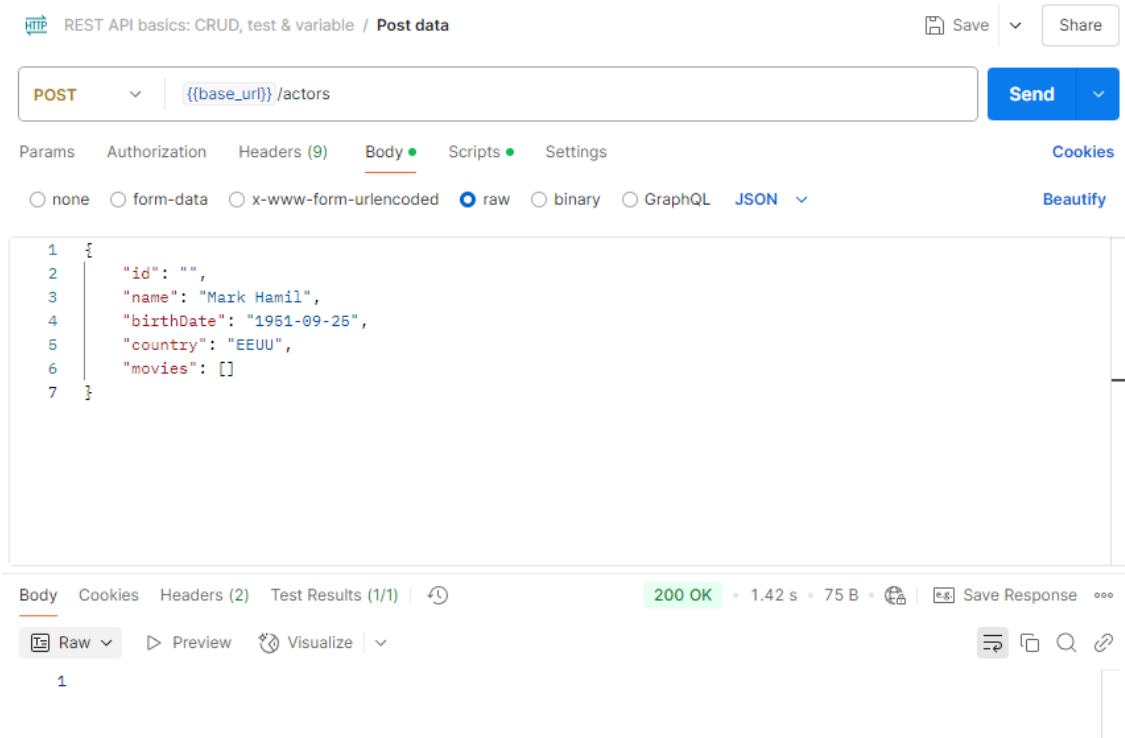


Ilustración 4 Creación un Actor Status 200

REST API basics: CRUD, test & variable / **Get data** Save Share

GET ⌵ `{{base_url}}/actors` Send ⌵

Params Authorized https://moviecards-service-antoranz.azurewebsites.net/ Cookies

☒ none ☐ form-data GraphQL

C Collection Variables in request →

This request does not have a body

Body Cookies Headers (3) Test Results (1/1) 🔄 200 OK • 1.17 s • 235 B • 🌐 📄 Save Response ⋮

{} JSON ⌵ ▶ Preview 🔍 Visualize ⌵ 📄 🔍 🔗

```
1 [
2   {
3     "id": 1,
4     "name": "Mark Hamil",
5     "birthDate": "1951-09-25T00:00:00.000+00:00",
6     "deadDate": null,
7     "country": "EEUU",
8     "movies": []
9   }
10 ]
```

Ilustración 5 Listar Actores

GET ⌵ `{{base_url}}/actors/1` Send ⌵

Params Authorized https://moviecards-service-antoranz.azurewebsites.net/ Cookies

Query Params C Collection Variables in request →

| | Key | | Description | ... | Bulk Edit |
|--|-----|-------|-------------|-----|-----------|
| | Key | Value | Description | | |

Body Cookies Headers (3) Test Results (1/1) 🔄 200 OK • 585 ms • 233 B • 🌐 📄 Save Response ⋮

{} JSON ⌵ ▶ Preview 🔍 Visualize ⌵ 📄 🔍 🔗

```
1 {
2   "id": 1,
3   "name": "Mark Hamil",
4   "birthDate": "1951-09-25T00:00:00.000+00:00",
5   "deadDate": null,
6   "country": "EEUU",
7   "movies": []
8 }
```

Ilustración 6 Obtener Actor por Id

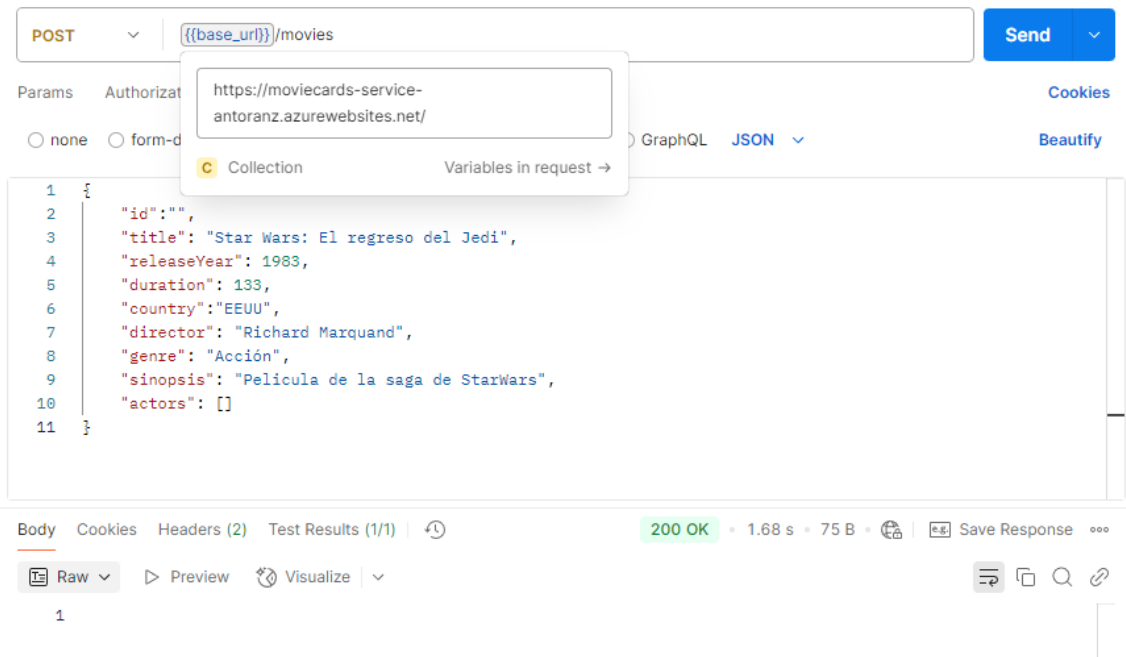


Ilustración 7 Crear Pelicula Status 200.

GET

[[base_url]]/movies

Send

Params

Authorization

Query Params

https://moviecards-service-antoranz.azurewebsites.net/

Collection

Variables in request →

| Key | Description | Bulk Edit |
|-----|-------------|-------------|
| Key | Value | Description |

Body

Cookies

Headers (3)

Test Results (1/1)

200 OK

488 ms

323 B

Save Response

{ } JSON

Preview

Visualize

```
1  [
2    {
3      "id": 2,
4      "title": "Star Wars: El regreso del Jedi",
5      "releaseYear": 1983,
6      "duration": 133,
7      "country": "EEUU",
8      "director": "Richard Marquand",
9      "genre": "Acción",
10     "sinopsis": "Película de la saga de StarWars",
11     "actors": []
12   }
13 ]
```

Ilustración 8 Listar películas.

REST API basics: CRUD, test & variable / consultar Pelicula

GET `{{base_url}}/movies/2` Send

Params Authorizat

Query Params

Collection Variables in request →

| Key | Value | Description | Bulk Edit |
|-----|-------|-------------|-----------|
| Key | Value | Description | |

Body Cookies Headers (3) Test Results (1/1) 200 OK 518 ms 321 B Save Response

JSON Preview Visualize

```
1 {
2   "id": 2,
3   "title": "Star Wars: El regreso del Jedi",
4   "releaseYear": 1983,
5   "duration": 133,
6   "country": "EEUU",
7   "director": "Richard Marquand",
8   "genre": "Acción",
9   "sinopsis": "Película de la saga de StarWars",
10  "actors": []
11 }
```

Ilustración 9 Obtener película ID

GET `{{base_url}}/movies/insc/1/2` Send

Params Authorizat

Query Params

Collection Variables in request →

| Key | Value | Description | Bulk Edit |
|-----|-------|-------------|-----------|
| Key | Value | Description | |

Body Cookies Headers (2) Test Results (1/1) 200 OK 188 ms 75 B Save Response

Raw Preview Visualize

1

Ilustración 10 Inscribir Actor en una Peli

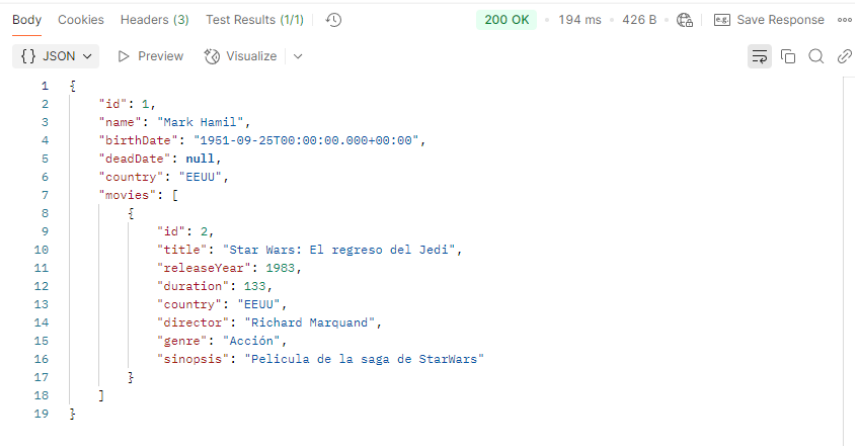
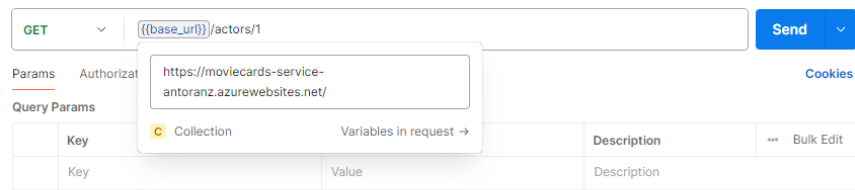


Ilustración 11 Consultar Actor Con película

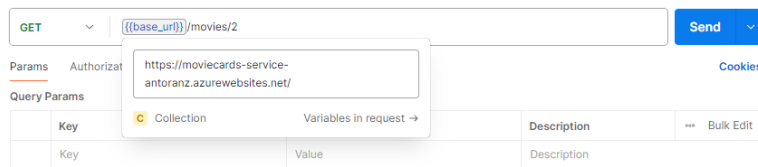


Ilustración 12 Consultar película con Actor

PUNTO 3

- 3) Realizar una nueva versión de la aplicación moviecards, modificando el código del repositorio moviecards de la práctica del tema 5, para que utilice el servicio creado en el apartado anterior. Debe quedar funcionando en la misma URL de la versión anterior <https://moviecards-apellido.azurewebsites.net>. Para ello, hay que crear en el proyecto un nuevo sprint (milestone) con los siguientes issues: • 3.1) Modificar el código de la aplicación en src/main. • 3.2) Modificar el código de las pruebas en src/test

Para la resolución de este apartado se siguieron las modificaciones que indico el profesor en la parte de ayuda y se creó un nuevo Milestone con el nombre de “Refactorización código”

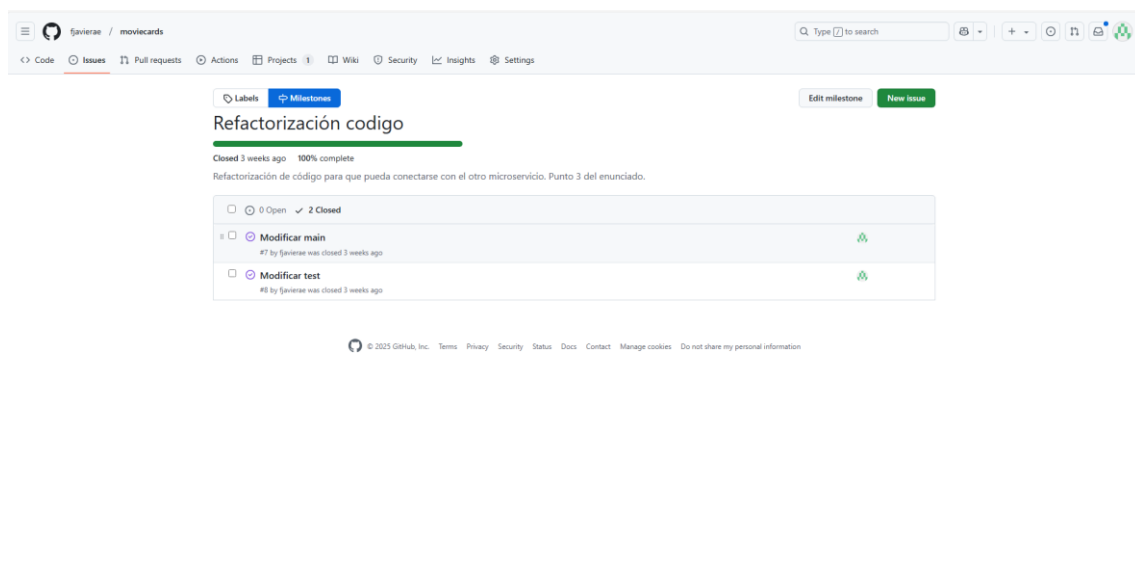


Ilustración 13 Refactorización de Código MileStone

Los cambios de este milestone se pueden ver en la rama: “tarea-refactorizacion”, concretamente la mayoría de los cambios están en el commit: “Refactorización de main y test”. En un principio en esta rama se eliminaron las clases de los repositorios porque no le vi mucho sentido que estuviesen en un servicio que es front pero en commits posteriores se restauraron para que se usasen en las modificaciones que había que hacer a los test para incluir la fecha de fallecimiento.

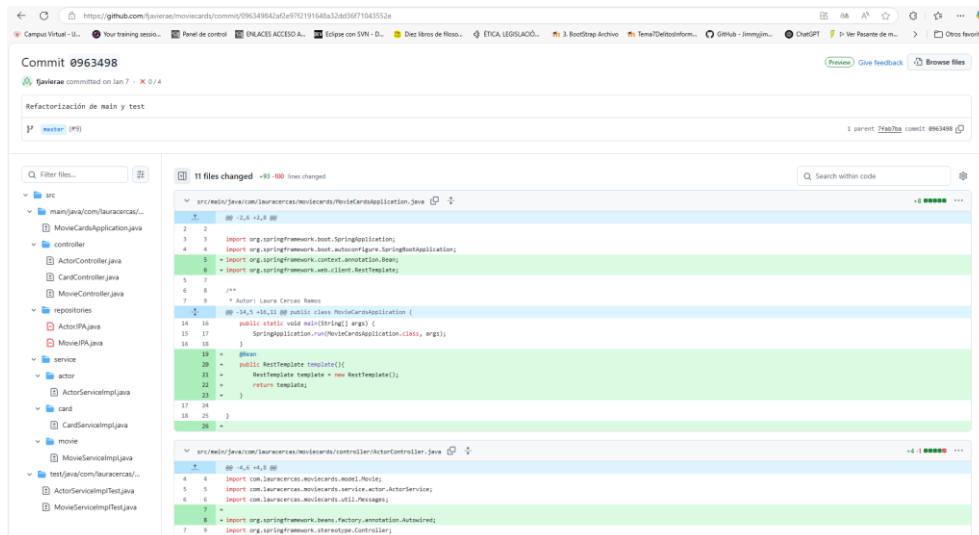


Ilustración 14 Commit 0963498

PUNTO 4

- 4) Modificar el código del repositorio moviedcards-service para añadir un nuevo atributo en la clase Actor, llamado `deadDate`, que contenga la fecha de fallecimiento del actor, y desplegar en Azure, en la misma URL del servicio: <https://moviedcards-service-apellido.azurewebsites.net>. Para este repositorio no es necesario crear ningún proyecto asociado al repositorio, ni tampoco milestones, issues, ramas ni nuevas pruebas. **NOTA:** El profesor probará el funcionamiento del servicio usando Postman para crear actores con fecha de fallecimiento.

El código modificado para esta solución se encuentra en el commit: “Punto 4) Modificar Actor para que tenga fecha de muerte”

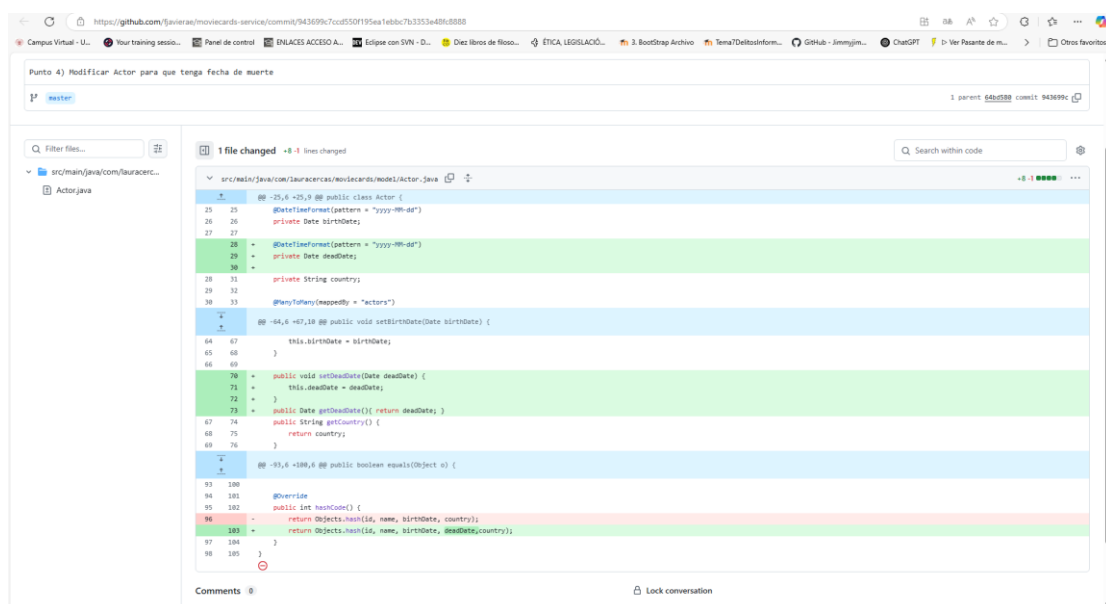


Ilustración 15 Commit 943699c

Para comprobar el resultado de este ejercicio vamos a realizar unas comprobaciones en postman:

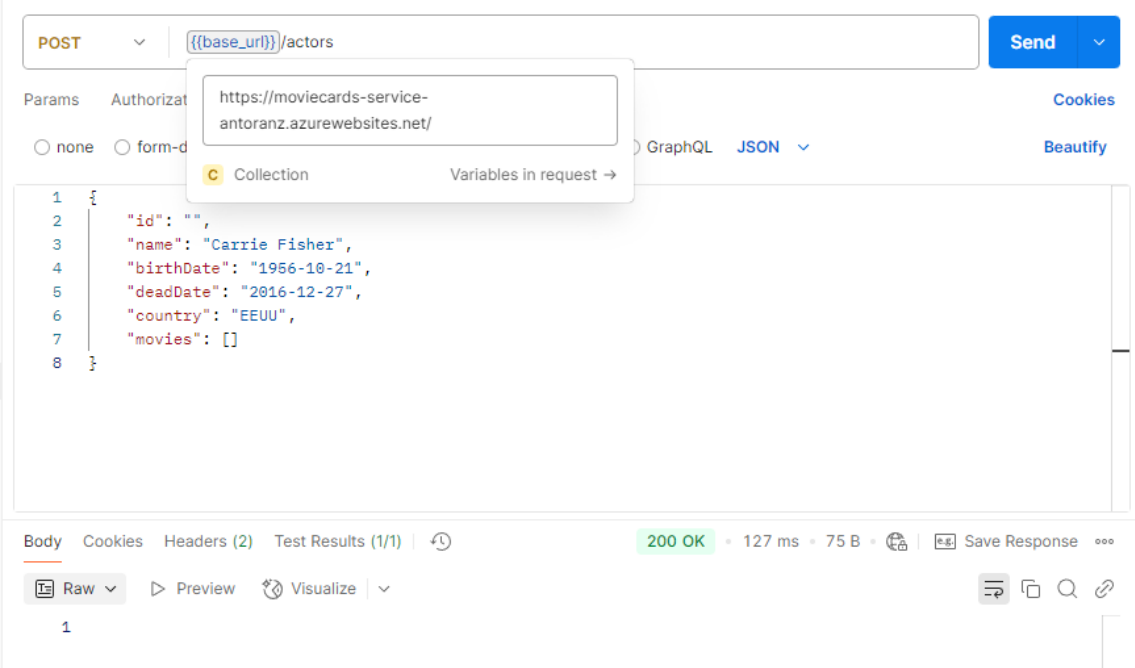


Ilustración 16 Crear Actor con Fecha de Muerte

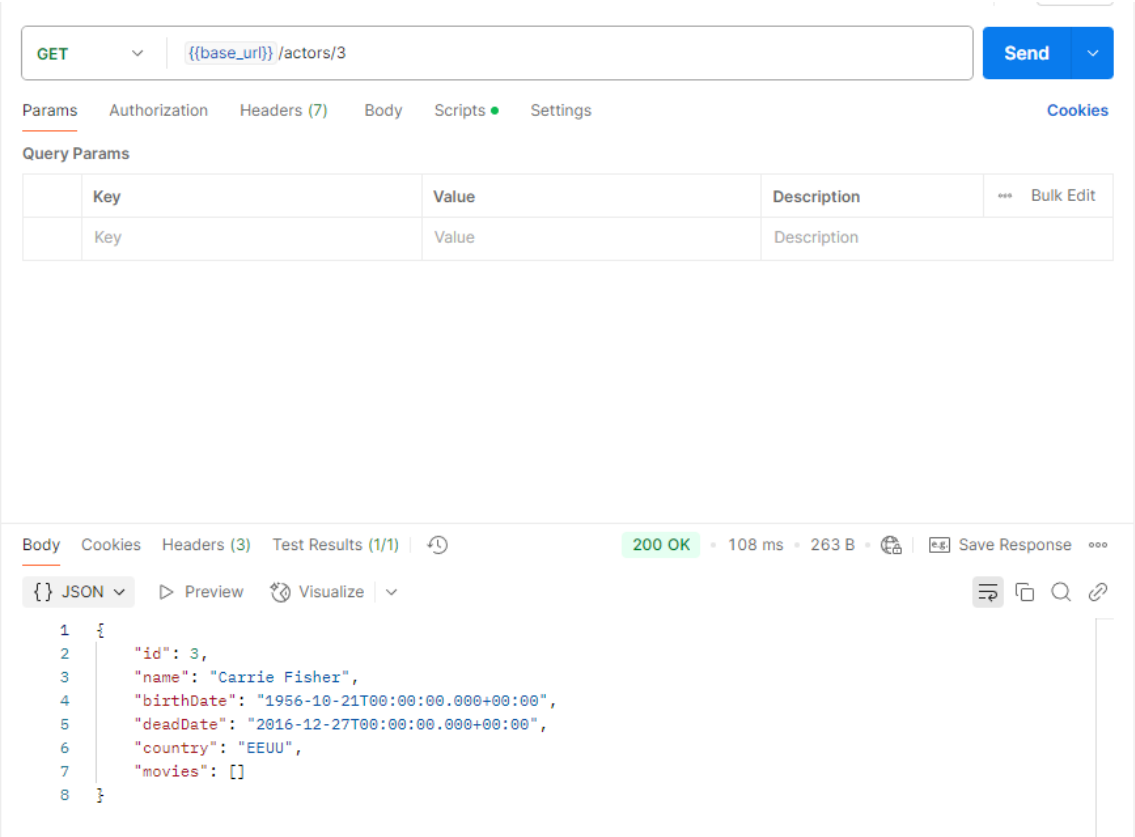


Ilustración 17 Consultar Actor con Fecha de Muerte

PUNTO 5

5) Realizar una nueva versión de la aplicación moviecards, para que en la página web, para crear un nuevo actor se pida al usuario también la fecha de fallecimiento, y en la página de listado de actores aparezca una nueva columna con la fecha de fallecimiento. Debe quedar funcionando en la misma URL de la versión anterior <https://moviecards-apellido.azurewebsites.net>. Para ello, hay que crear en el proyecto un nuevo sprint (milestone) con los siguientes issues:

Para este ejercicio se creo un milestone con una tarea para cada uno de los siguientes puntos.

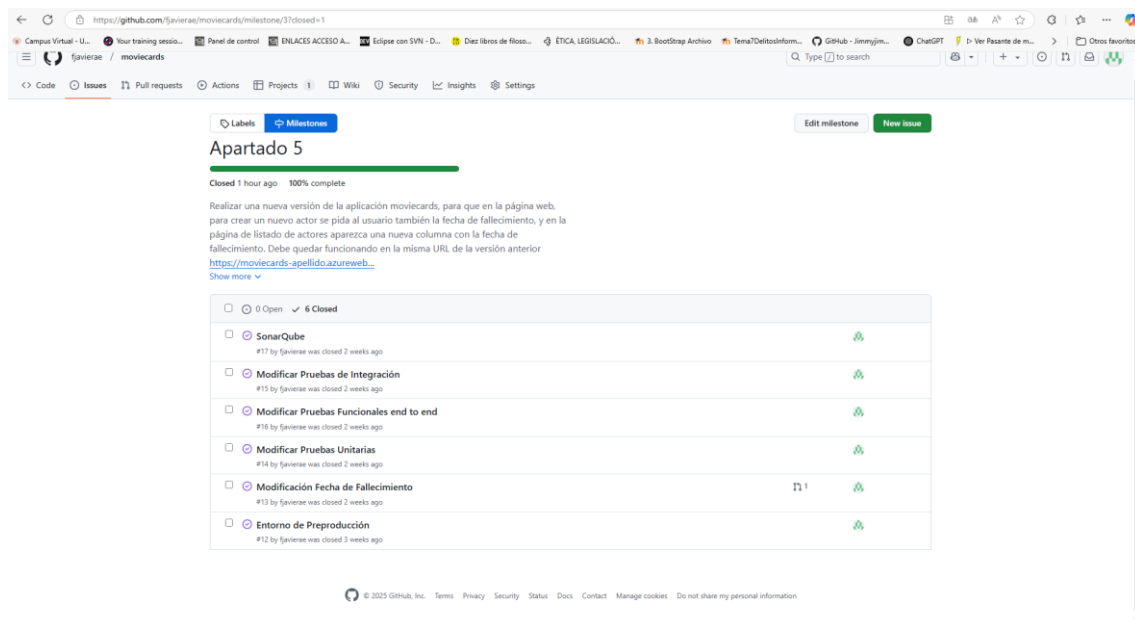


Ilustración 18 Milestone Apartado 5

5.1) Añadir al workflow del proyecto un nuevo trabajo, entre los trabajos “qa” y “deploy”, llamado “stage” para desplegar la aplicación en un entorno de preproducción, también en Azure, pero usando otro nombre para la aplicación, por ejemplo, *moviecards-pre*.

Para la solución de este apartado se ha creado un nuevo job llamado stage y se le ha asignado otro secreto de github proporcionado al crear la aplicación en azure. Para que se despliegue al hacer un commit sobre la rama master, pero sin aprobación. La aprobación nos la reservamos para producción para asegurarnos que solo lo subiremos si nos pasa el qa.

```

stage:
  needs: qa
  runs-on: ubuntu-latest
  if: github.ref=='refs/heads/master'
  environment:
    name: 'Preproduction'
    url: ${ steps.deploy-to-webapp.outputs.webapp-url }
  steps:
    - name: Download artifact from build job
      uses: actions/download-artifact@v4
      with:
        name: moviecards-java

    - name: Deploy to Azure Web App
      id: deploy-to-webapp
      uses: azure/webapps-deploy@v3
      with:
        app-name: 'moviecards-pre-antoranz'
        slot-name: 'Production'
        package: '*.jar'
        publish-profile: ${ secrets.AZUREAPPSERVICE_PUBLISHPROFILE_2F00E89734DF4CB784620207DDE5895E }

```

Ilustración 19 Nuevo job stage

5.2) Modificar el código de la aplicación para manejar la fecha de fallecimiento de los actores. **NOTA:** Debe crearse una nueva rama en el repositorio para los cambios y después integrarlos en la rama máster.

Para este caso se creó una nueva rama en el repositorio, llamada 13-modificación-fecha-de-fallecimiento. Los cambios principales se encuentran en el commit: “Añadiendo fecha de fallecimiento” f0adf23.

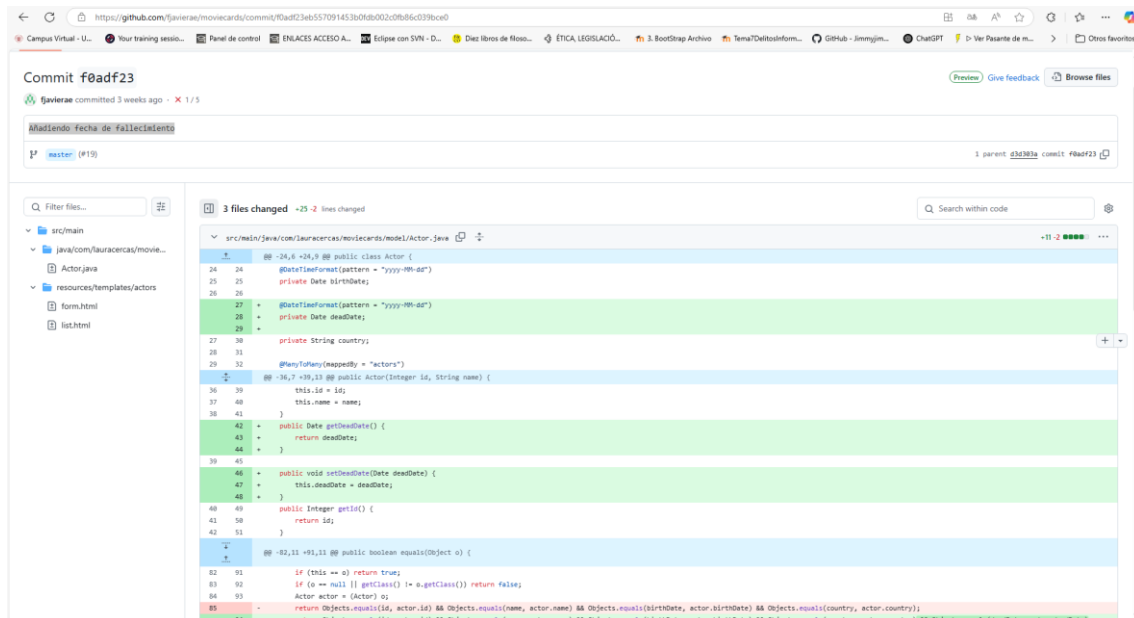


Ilustración 20 Commit f0adf23

Pruebas de la Aplicación:

← ↻ 🔍 https://moviecards-pre-antoranz.azurewebsites.net/actors/new

Campus Virtual - U... Your training sessio... Panel de control ENLACES ACCESO A... Eclipse con SVN - D... Diez libros de filos... ÉTICA LEGISLACI... 3. Bootstrap Archiv... Tema7DeitosInform... GitHub - jmmgim... ChatGPT D- Ver Pasante de m... Otros favoritos

Nuevo Actor

Nombre

Peter Mayhew

Fecha nacimiento

19/06/1944

Fecha fallecimiento

30/04/2019

País

Británico y Estado Unidense

Guardar

Volver a la Página de Inicio

Ilustración 21 Crear Actor en Pre

← ↻ 🔍 https://moviecards-pre-antoranz.azurewebsites.net/actors

Campus Virtual - U... Your training sessio... Panel de control ENLACES ACCESO A... Eclipse con SVN - D... Diez libros de filos... ÉTICA LEGISLACI... 3. Bootstrap Archiv... Tema7DeitosInform... GitHub - jmmgim... ChatGPT D- Ver Pasante de m... Otros favoritos

Listado Actores

| Identificador | Nombre | Fecha Nacimiento | Fecha Fallecimiento | País | Editar |
|---------------|---------------|------------------|---------------------|-----------------------------|--------|
| 1 | Mark Hamil | 25-09-1951 | | EEUU | Editar |
| 3 | Carrie Fisher | 21-10-1956 | 27-12-2016 | EEUU | Editar |
| 4 | Peter Mayhew | 19-06-1944 | 30-04-2019 | Británico y Estado Unidense | Editar |

Volver a la Página de Inicio

Ilustración 22 Listar Actor en Pre

Nuevo Actor

Nombre
Kenny Baker

Fecha nacimiento
24/08/1934

Fecha fallecimiento
13/08/2016

País
Inglaterra

Guardar

[Volver a la Página de Inicio](#)

Ilustración 23 Crear Actor Producción

Listado Actores

| Identificador | Nombre | Fecha Nacimiento | Fecha Fallecimiento | País | Editar |
|---------------|---------------|------------------|---------------------|-----------------------------|------------------------|
| 1 | Mark Hamill | 25-09-1951 | | EEUU | Editar |
| 3 | Carrie Fisher | 21-10-1956 | 27-12-2016 | EEUU | Editar |
| 4 | Peter Mayhew | 19-06-1944 | 30-04-2019 | Británico y Estado Unidense | Editar |
| 5 | Kenny Baker | 24-08-1934 | 13-08-2016 | Inglaterra | Editar |

[Volver a la Página de Inicio](#)

Ilustración 24 Listado Actores Producción

5.3) Modificar las pruebas unitarias para tener en cuenta la fecha de fallecimiento de los actores.

5.4) Modificar las pruebas de integración para tener en cuenta la fecha de fallecimiento de los actores.

5.5) Modificar las pruebas funcionales (end to end) para tener en cuenta la fecha de fallecimiento de los actores.

Para estos puntos se modificaron los test: ActorE2ETest.java, ActorJPAIT.java y ActorTest.java

Los cambios en esos test se pueden ver en la rama del apartado 5.1. En el commit: “Modificación test” 202befb.

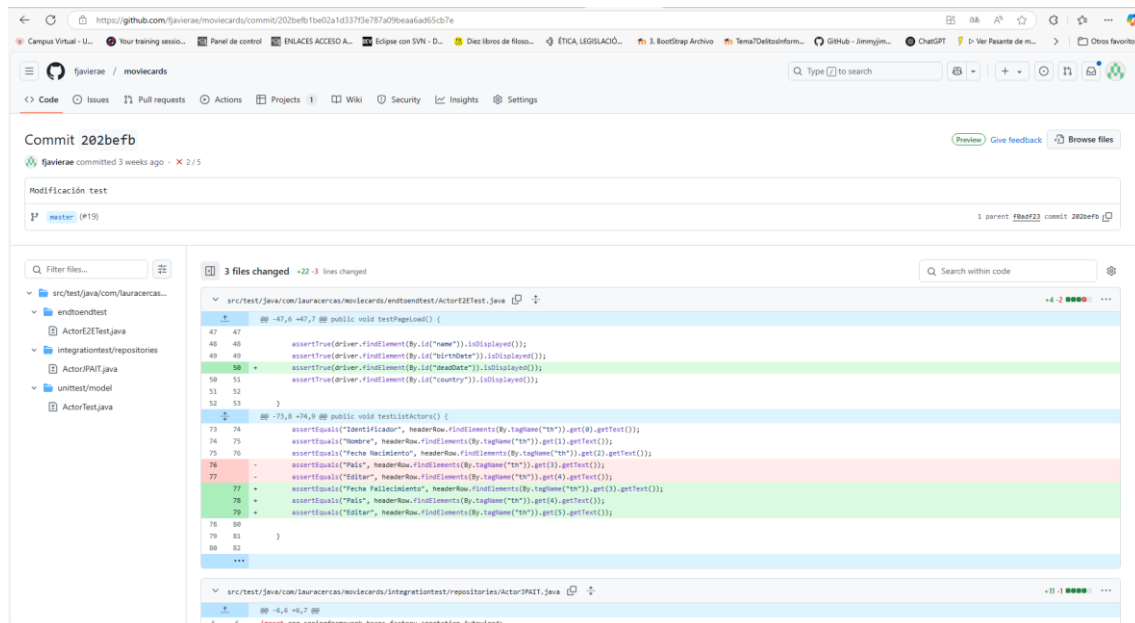


Ilustración 25 Commit 202befb

5.6) Como garantía de calidad, hay que modificar el código fuente de la aplicación hasta conseguir tener menos de 5 problemas críticos, de tal forma que, si SonarQube detecta que tiene 5 o más, no debe poderse desplegar la aplicación a producción.

Para la solución de este apartado se tuvieron que crear DTOS para solucionar problemas críticos (Commit: “Solucionando errores sonarq creamos DTOS” 05aeb1).

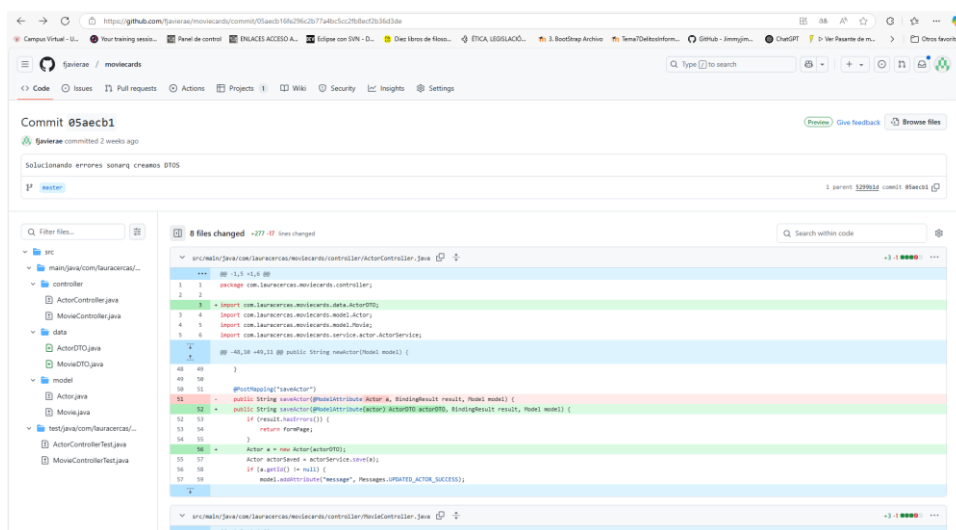


Ilustración 26 Commit 05aeb1

Otros cambios que se hicieron son la eliminación de declarar atributos de manera innecesaria cuando solo se usaban una vez.

Ej: RestTemplate template = new RestTemplate(); return template

Por: return new RestTemplate();

Además de estos cambios se tuvieron que hacer cambios en las pruebas porque detectaba errores críticos o alguna advertencia en el uso de assert en vez de assertEquals y por añadir en los métodos la notación “public” de manera innecesaria, porque ya estaban por defecto.

Todos estos cambios también se ven reflejados en el commit “Soluciones sonarq”(91159d7e) y “solucionando errores de sonar q”(65518c3 y 8184571), este ultimo se hizo antes del paso 5 para intentar pasar el control de calidad en una rama.

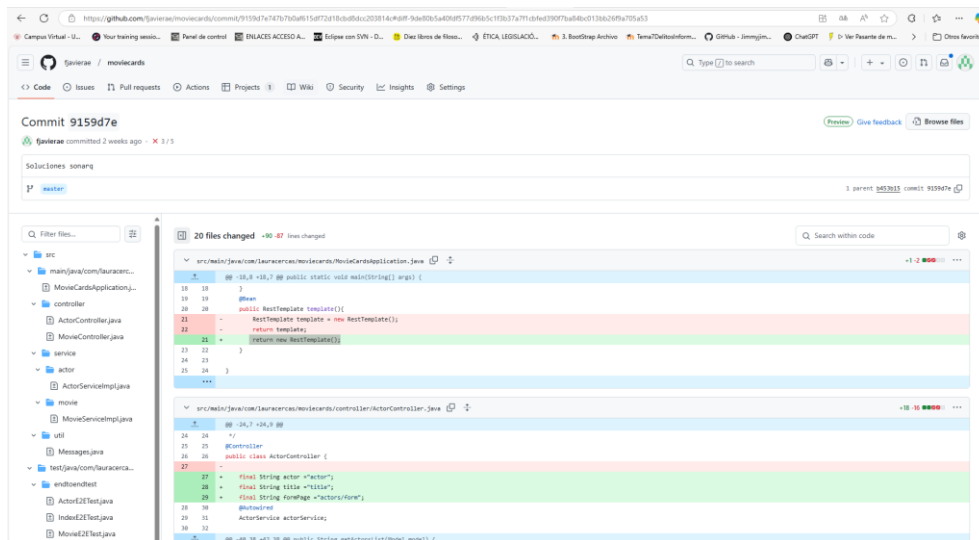


Ilustración 27 Commit 91159d7e

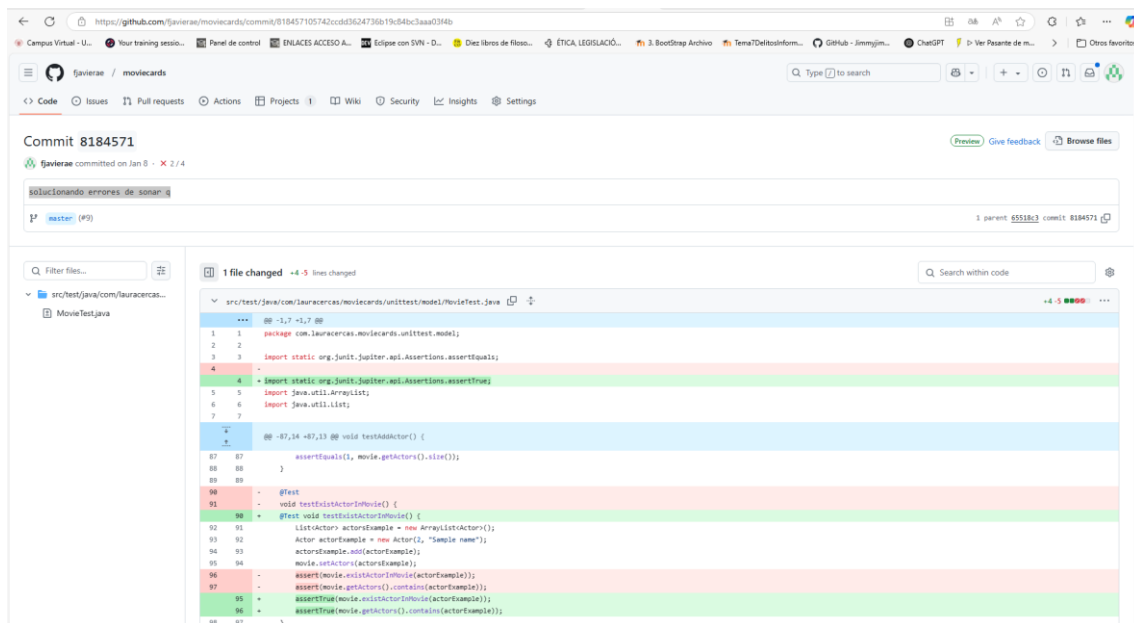


Ilustración 28 Commit 8184571

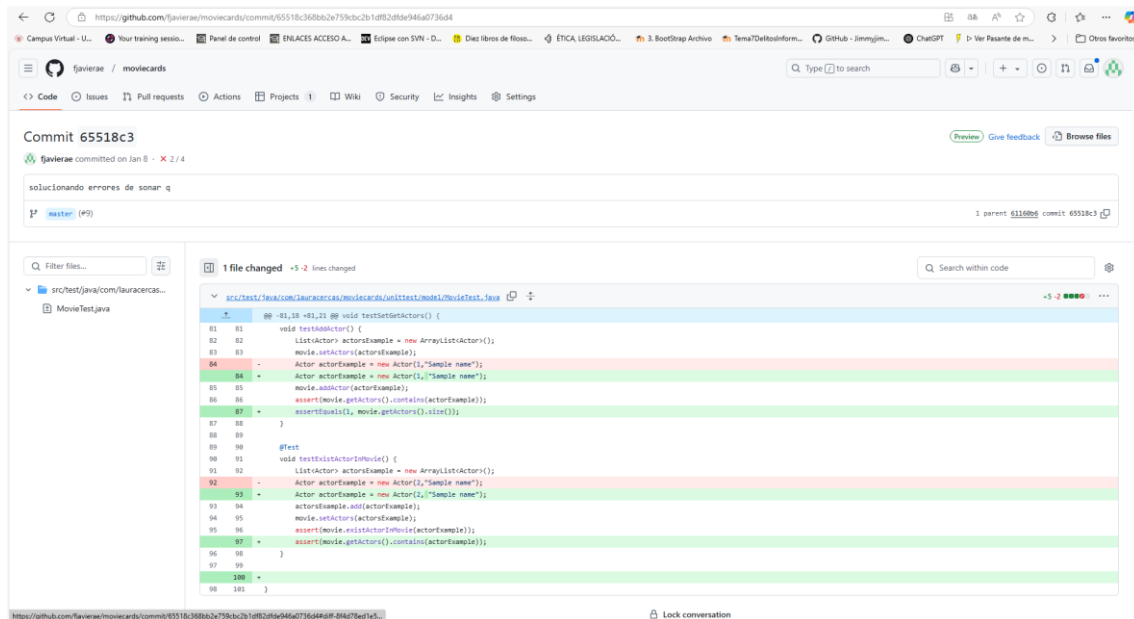


Ilustración 29 Commit 65518c3

Para acabar de mostrar el resultado de este apartado se mostrarán las capturas de la regla y de la ultima ejecución del control de calidad pasada:

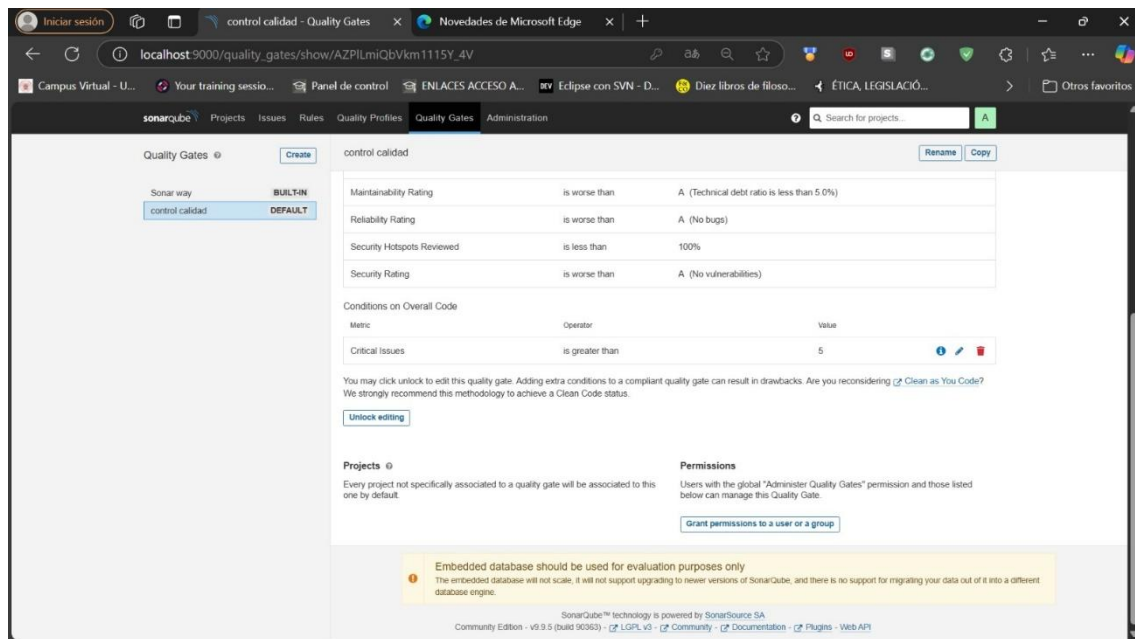


Ilustración 30 Nueva Regla limitar críticas a 5

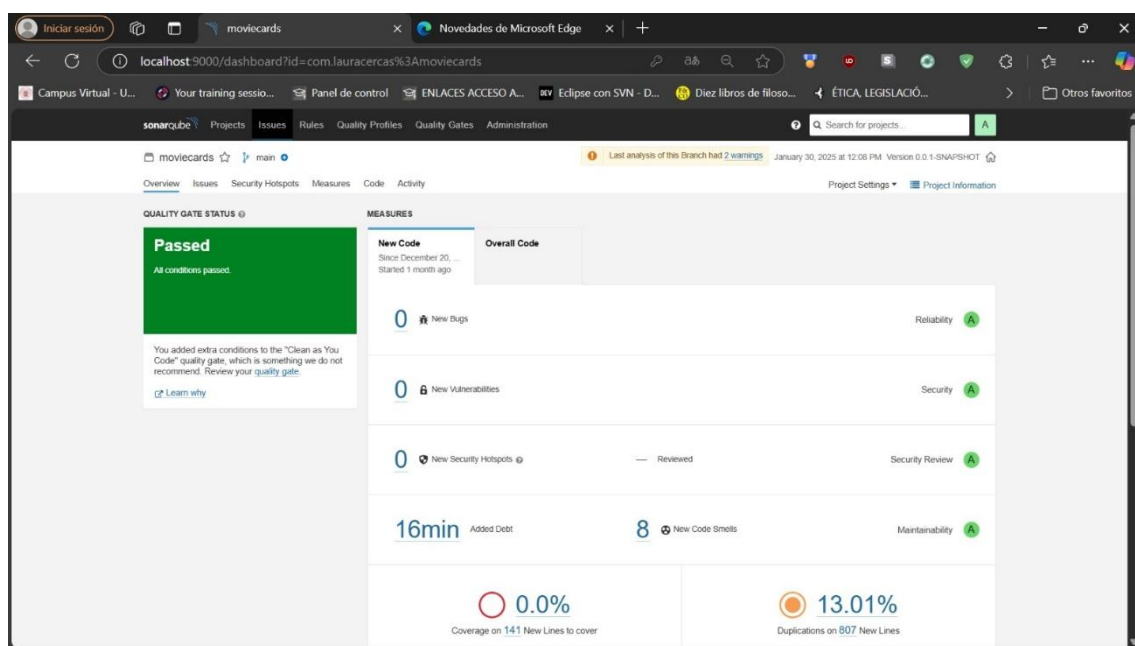


Ilustración 31 Pantalla general SonarQ

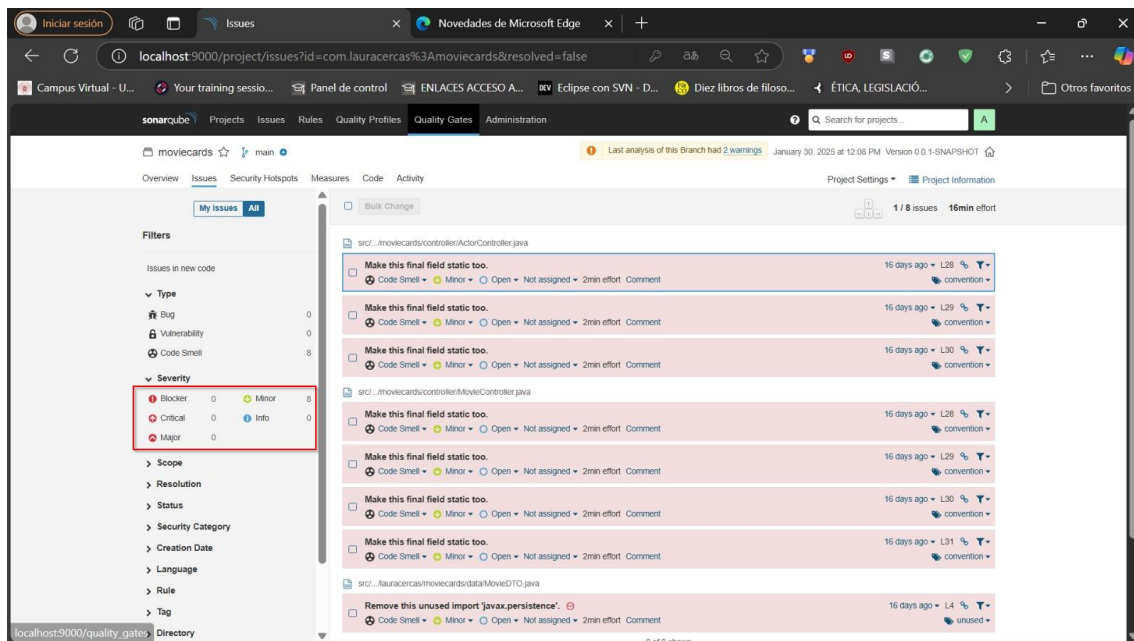


Ilustración 32 Pantalla Detalle Issues

WORKFLOW FINAL

Por último, mostramos como queda el “pipeline o workflow” en el último commit:

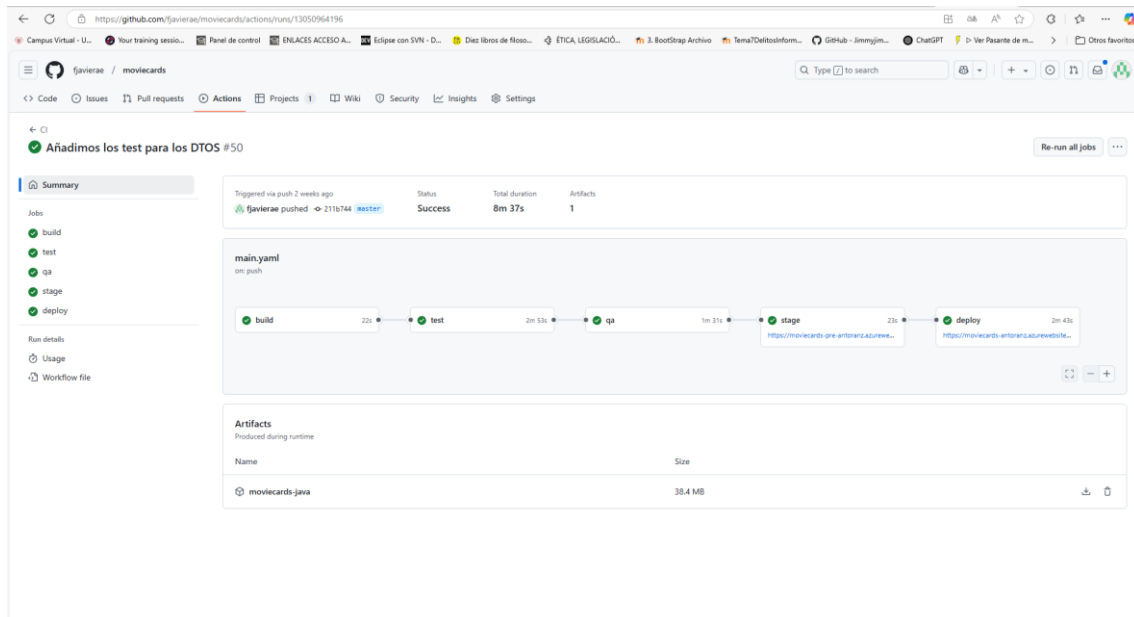


Ilustración 33 Workflow Último Commit