TRABAJO FINAL. INTEGRACIÓN CONTINUA

20 DE FEBRERO DE 2026

Eva Huertas Morera

Contenido

1. Introducción

En el siguiente documento se detalla el proceso de integración continua de la aplicación web moviecards. El proyecto se centra en la transición de una arquitectura con un frontend situado en el proyecto moviecard hacia una arquitectura de servicios desacoplados, aplicando flujos de trabajo de integración y despliegue continuo (CI/CD) en Github.

1.1. Objetivo del trabajo

El objetivo principal es separar la lógica de negocio en un servicio independiente denominado moviecards-service. Esta nueva iteración busca ofrecer una API autónoma que pueda ser consumida por diversos clientes.

1.2. Alcance del proyecto

La evolución del sistema contempla los siguientes hitos clave:

- **Desacoplamiento:** Instalación del servicio en una URL independiente (https://moviecards-service-huertas.azurewebsites.net) y la aplicación cliente en la URL original.
- **Nuevas Funcionalidades:** Incorporación del atributo deadDate (fecha de fallecimiento) en la entidad de actores, afectando tanto al servicio como a la interfaz de usuario de la aplicación web.
- **Calidad de Software:** Implementación de un entorno de pre-producción (*staging*) y la integración de umbrales de calidad en SonarQube para restringir el despliegue si se detectan 5 o más problemas críticos.
- **Automatización:** Actualización del flujo de trabajo en GitHub Actions para evaluar las pruebas unitarias, de integración y funcionales (E2E) adaptadas a la nueva arquitectura.

Para ello se deberá seguir los apartados mencionados en el enunciado.

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.

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. NOTA: El profesor probará el funcionamiento del servicio usando Postman.

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.

Apartados para realizar una nueva versión de la aplicación para manejar la fecha de fallecimiento de los actores:

4) Modificar el código del repositorio **moviecards-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://moviecards-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.

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: 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 pre-producción, también en Azure, pero usando otro nombre para la aplicación, por ejemplo, moviecards-pre.

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.

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

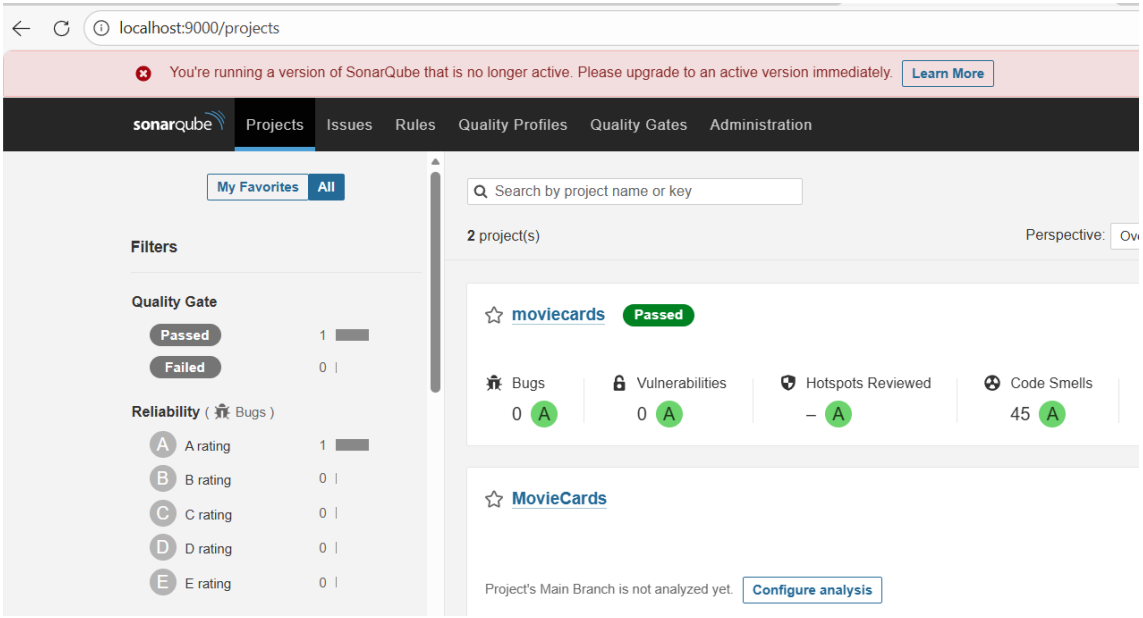
5.4) Modificar las pruebas de integración para tener en cuenta la fecha de fallecimiento de los actores.

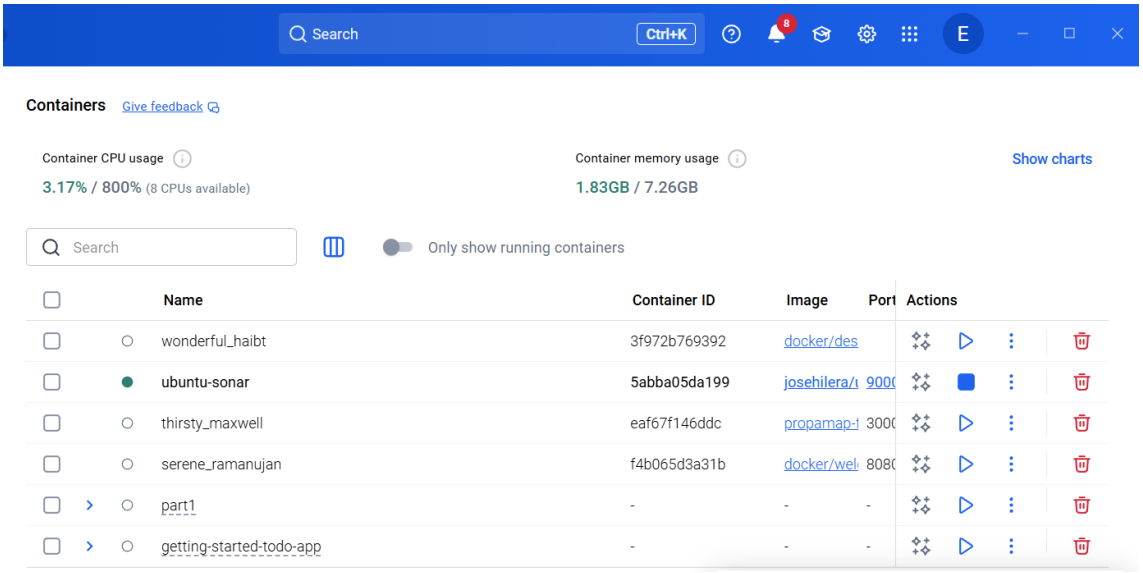
5.5) Modificar las pruebas funcionales (end to end) para tener en cuenta la fecha de fallecimiento de los actores.

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.

2. Realizar la práctica del tema 5

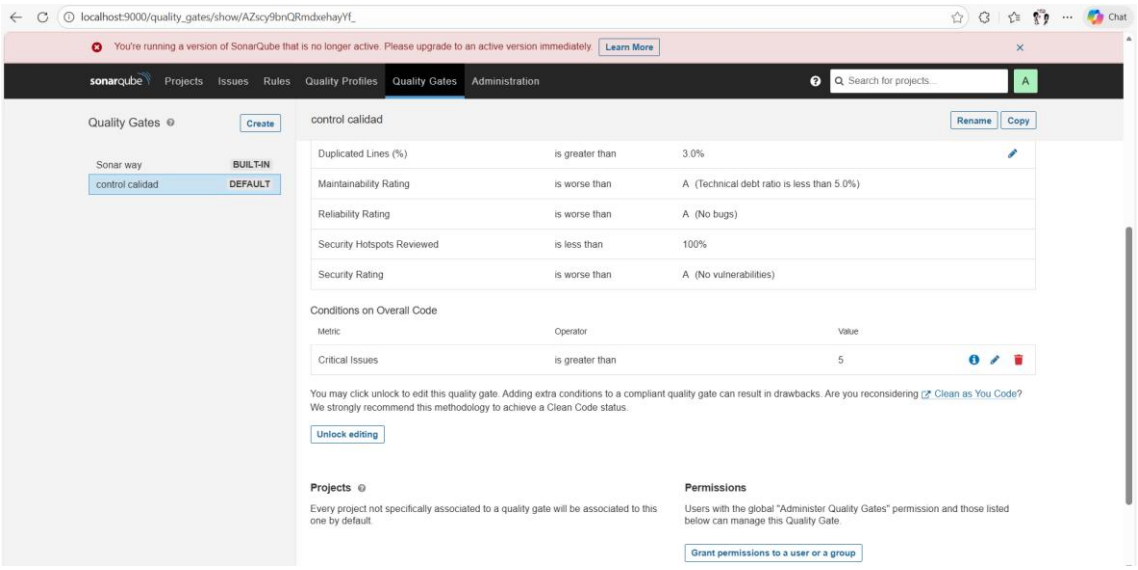
Se realizó la practica 5 guiada, que se empezó en clase, en la que se realizaron todos los pasos para la configuración del main.yaml, que cubrirán las tareas de los actions y por otra parte la ejecución tanto del Docker como del sonar, con la imagen de josehilera.



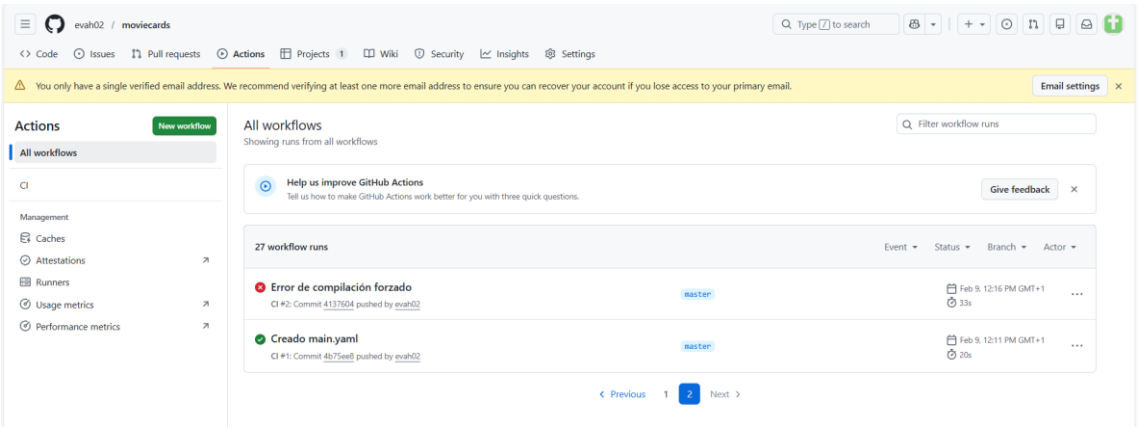


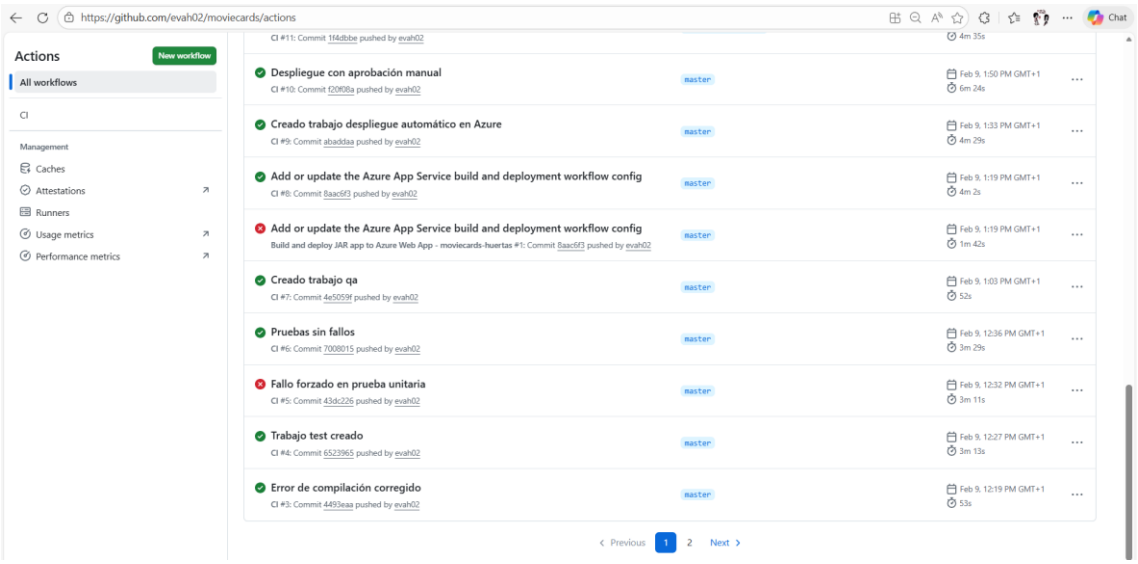
Una vez realizado todo eso se pudo visualizar la página del proyecto lanzado en la siguiente imagen. También se hicieron los cambios en cuanto al formato de la página por eso se ve de color gris y amarillo.

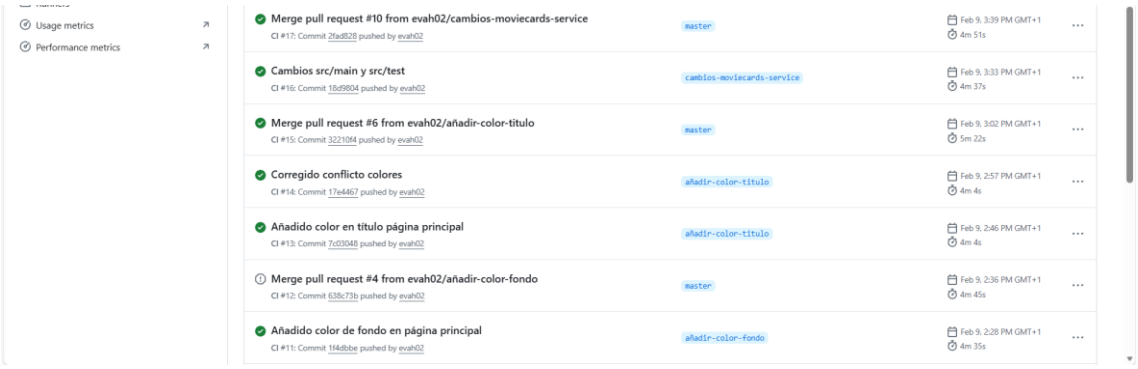
Las restricciones que se pusieron a sonarqube inicialmente, fueron las que se ven en la imagen, con 5 errores críticos al principio y que mas tarde en otra parte de la práctica guiada se cambió a 8, para que no diesen error en los actions.



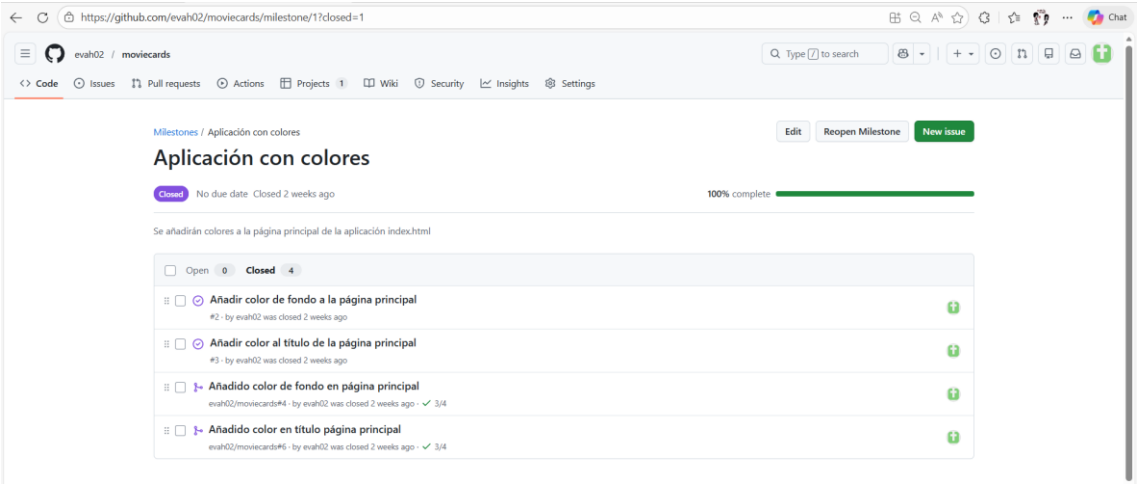
A continuación, se muestran todos los github actions de la práctica guiada.







Además de los actions se generaron los milestone y los issues pertinentes a esos cambios, que se muestran a continuación.

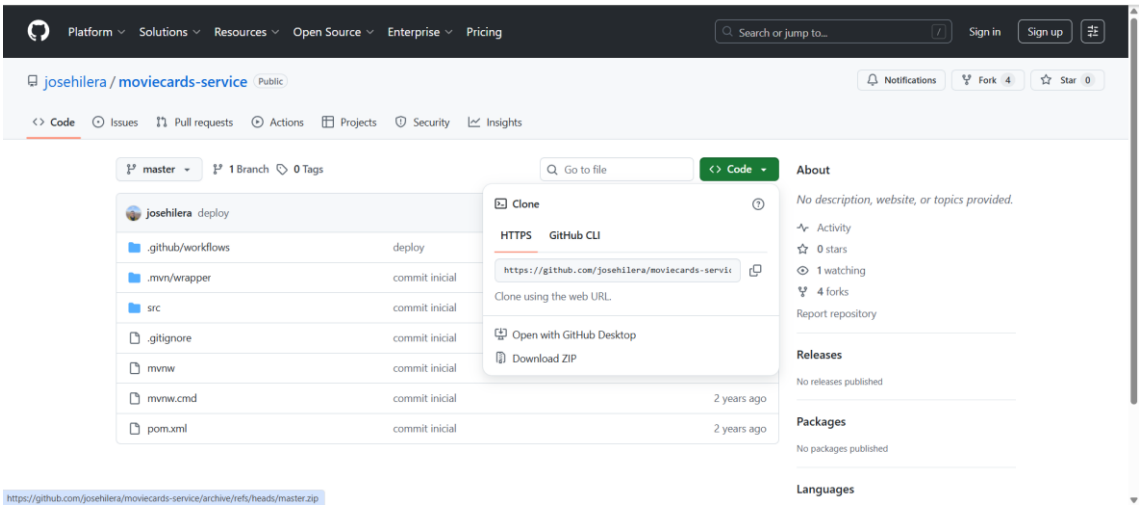


3. Configuración de moviecards-service

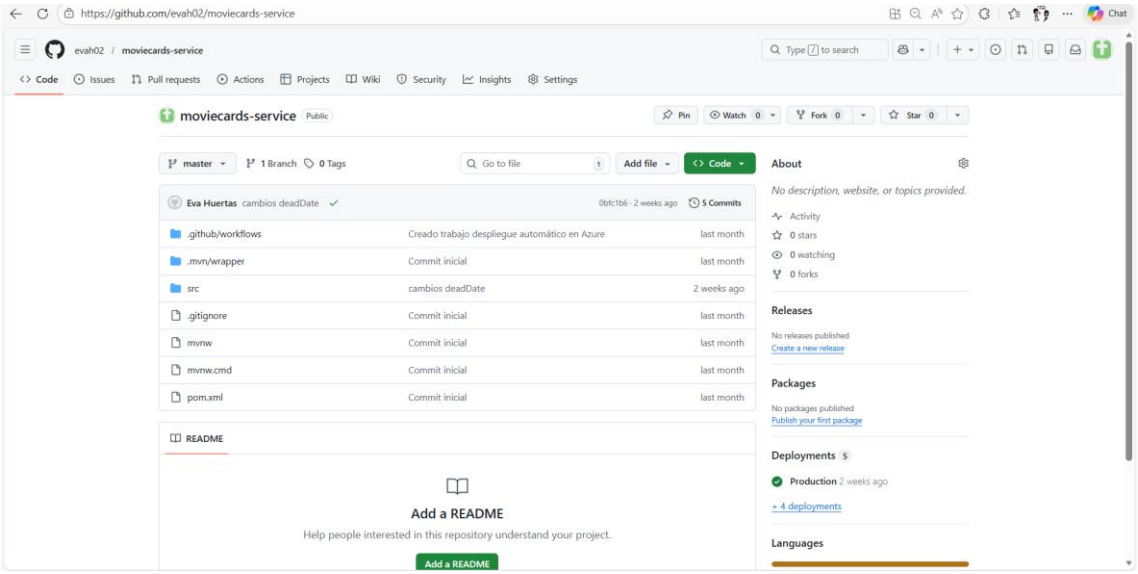
3.1. Creación del repositorio

Siguiendo las recomendaciones, se ha procedido de la siguiente manera:

- Se ha utilizado el código base proporcionado por el profesor desde el repositorio oficial.



- Se creó un nuevo repositorio local mediante el comando git init y se vinculó a un nuevo repositorio remoto en GitHub denominado moviecards-service.

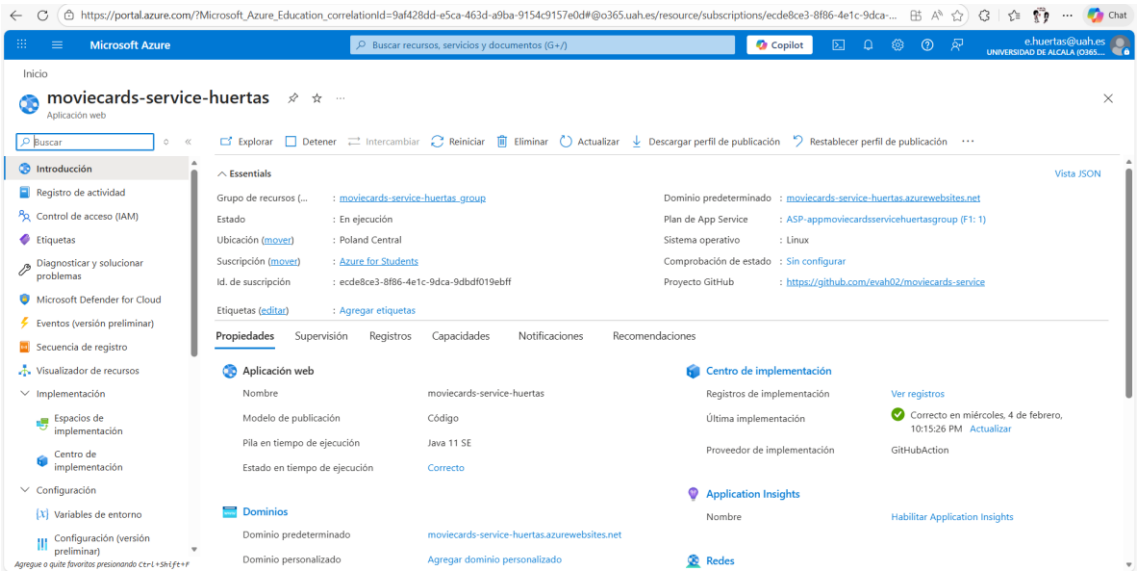


- No se asociaron proyectos, milestones ni issues a este repositorio específico, ya que el enfoque principal fue su despliegue y disponibilidad funcional.

3.1. Despliegue en Azure App Service

El servicio se ha desplegado en la plataforma Azure, garantizando que sea accesible de forma independiente.

- **URL de acceso:** El servicio está operativo en https://moviecards-service-apellido.azurewebsites.net (sustituyendo "apellido" por el correspondiente).



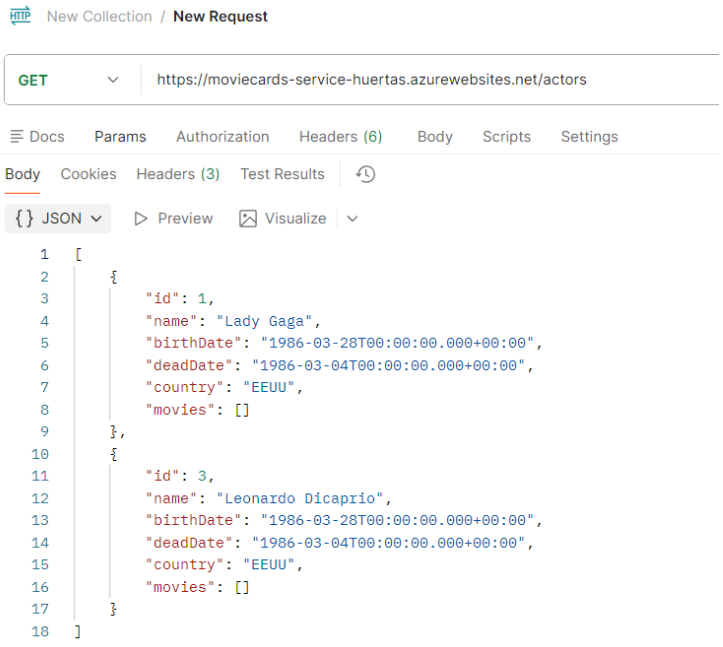
- **Configuración de CI/CD:** Se configuró un flujo de trabajo mediante GitHub Actions (.github/workflows/main.yaml), actualizando el secreto AZUREAPPSERVICE_PUBLISHPROFILE generado por Azure para permitir despliegues automáticos cada vez que se detectan cambios en el código.

3.3. Verificación de funcionamiento (Postman)

Para asegurar la integridad del servicio antes de conectarlo con la aplicación cliente, se realizaron pruebas manuales utilizando Postman. Se validaron los siguientes puntos finales (endpoints):

- **Actores:** Pruebas de creación (POST /actors) y listado (GET /actors).
- **Películas:** Verificación de creación (POST /movies) y consulta de detalles (GET /movies/idM).
- **Relaciones:** Se comprobó la funcionalidad de inscribir actores en películas mediante el endpoint /movies/insc/idA/idM.

En la siguiente imagen se indica uno de ellos, en este caso el listado de los actores.



4. Modificaciones en la aplicación principal (moviecards)

En esta fase se actualizó la aplicación original de la práctica del Tema 5 con lo especificado en el enunciado de la práctica.

4.1. Reconfiguración de la arquitectura

Se realizaron cambios estructurales en el código fuente (src/main) para integrar la comunicación entre servicios:

- **Inyección de dependencias:** Se añadió un Bean de RestTemplate en la clase principal MovieCardsApplication.java para habilitar las peticiones HTTP.
- **Controladores:** Se modificó CardController.java para utilizar @Autowired en la vinculación del servicio de actores.
- **Servicios de negocio:** Se actualizaron ActorServiceImpl.java y MovieServiceImpl.java para sustituir las interfaces JPA por llamadas a la API externa mediante RestTemplate.

4.2. Implementación de la comunicación REST

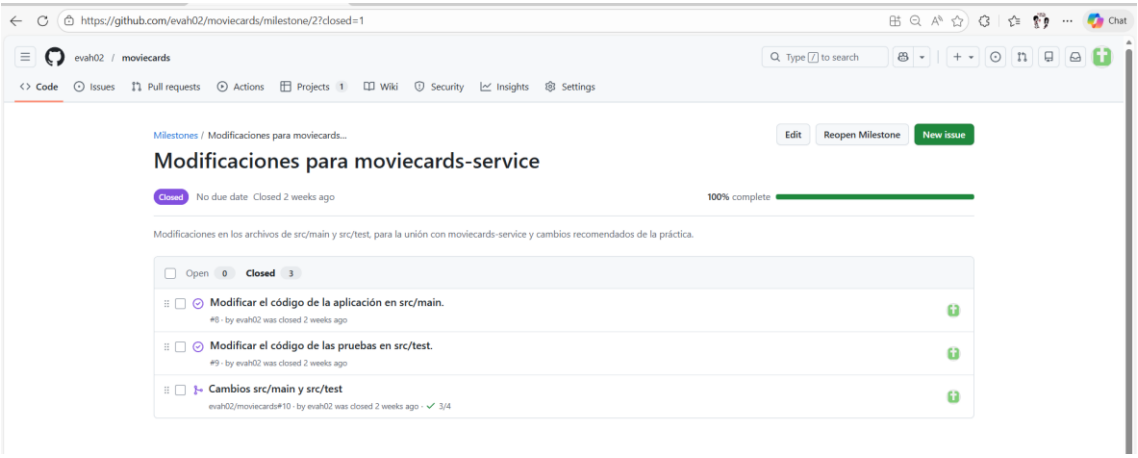
Para conectar con el servicio desplegado en Azure, se aplicaron las siguientes modificaciones técnicas:

- **Endpoints:** Se configuraron las URLs base apuntando a https://moviecards-service-huertas.azurewebsites.net/ tanto para actores como para películas.
- **Operaciones CRUD:**
    - El método findAll() ahora utiliza template.getForObject para recuperar listas de objetos desde el servicio.
    - El método save() se reestructuró para diferenciar entre inserciones (POST) y actualizaciones (PUT) dependiendo de la existencia de un ID previo.
    - El método getById() consulta directamente al endpoint específico del recurso (ej. /actors/{id}).

4.3. Gestión del proyecto y Metodología Ágil

Siguiendo los principios de desarrollo ágil, estos cambios se organizaron de la siguiente manera:

- Se creó un nuevo Sprint (Milestone) específico en el proyecto de GitHub para agrupar las tareas de migración.
- Se vincularon Issues para el seguimiento de la modificación del código de la aplicación y la actualización de las pruebas.



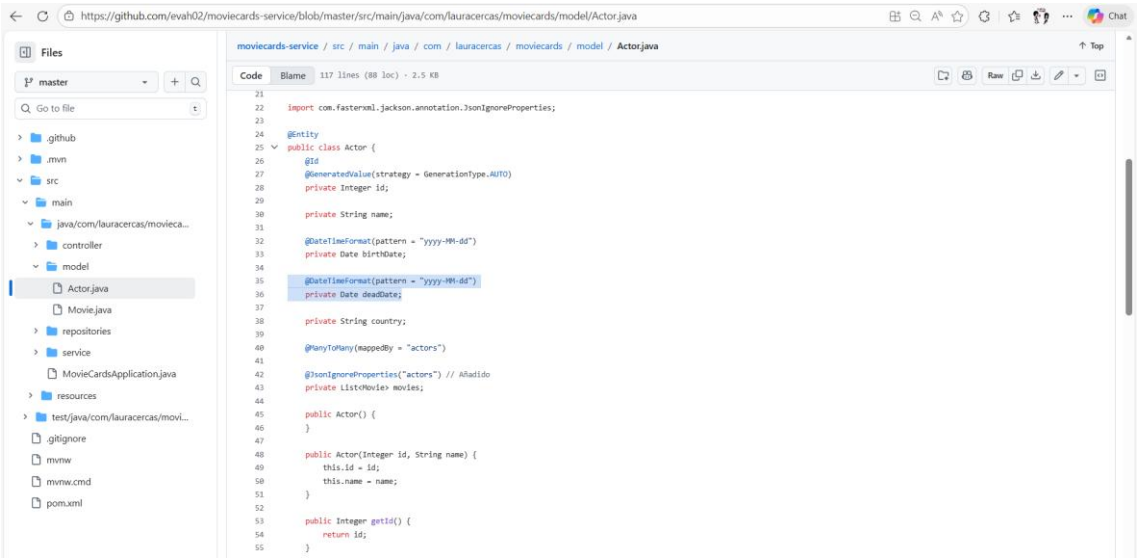
5. Evolución del Servicio: Gestión de la Fecha de Fallecimiento

Este apartado describe la actualización del modelo de datos en el servicio para permitir el manejo de información más completa de los actores.

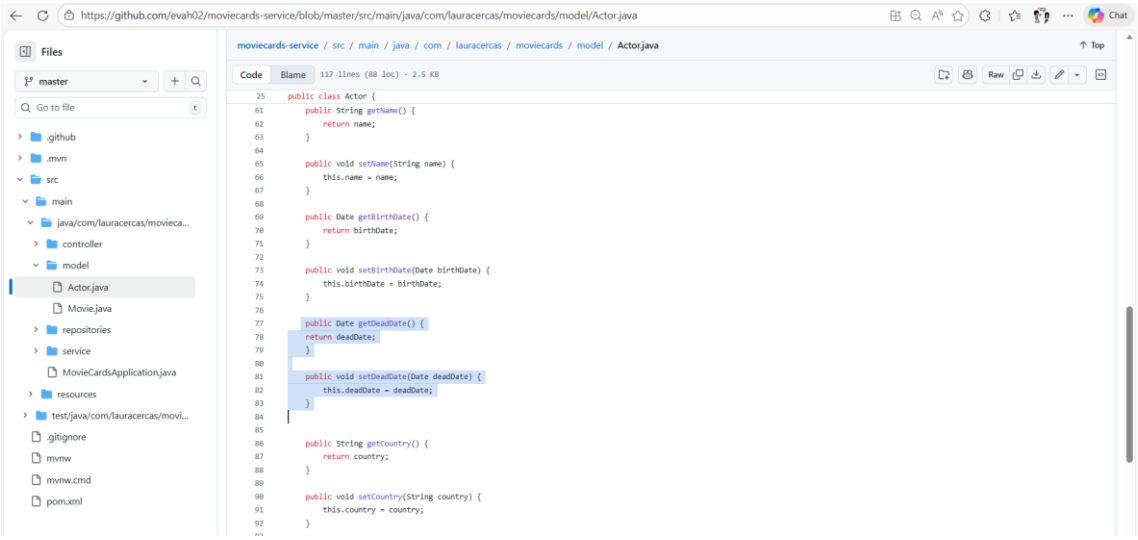
5.1. Modificación del Modelo de Datos

Se ha actualizado el código fuente del repositorio moviecards-service para integrar el nuevo requisito:

- **Atributo deadDate:** Se ha añadido un nuevo atributo llamado deadDate en la clase Actor para almacenar la fecha de fallecimiento.



- **Actualización del POJO:** Se han incluido los métodos *getter* y *setter* correspondientes para asegurar que el nuevo campo sea serializable y pueda ser procesado por la API REST.



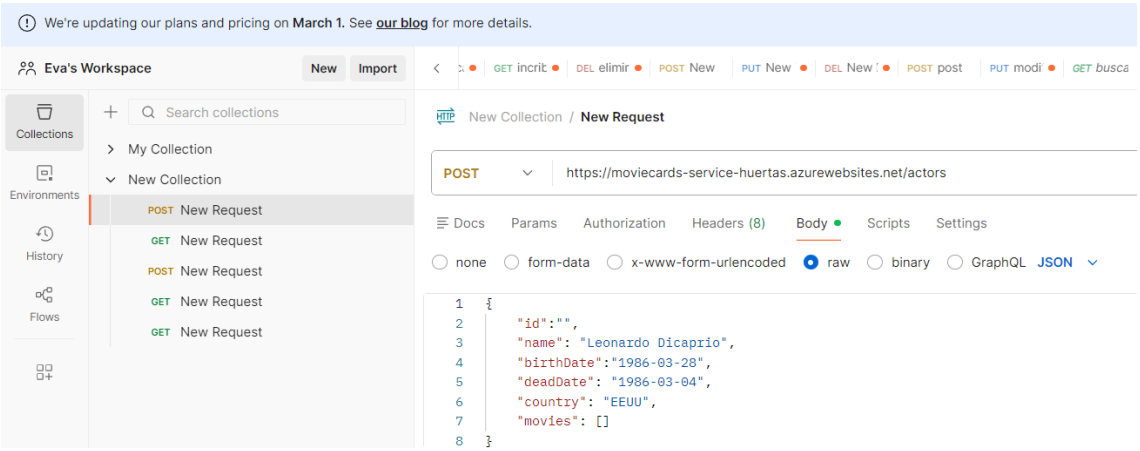
5.2. Despliegue y persistencia

- **Actualización en Azure:** Los cambios se han desplegado en la URL existente del servicio: https://moviecards-service-huertas.azurewebsites.net.
- **Integración:** No ha sido necesario crear nuevos proyectos, milestones o ramas específicas para este cambio en el repositorio del servicio, siguiendo las instrucciones de simplificación para este componente.

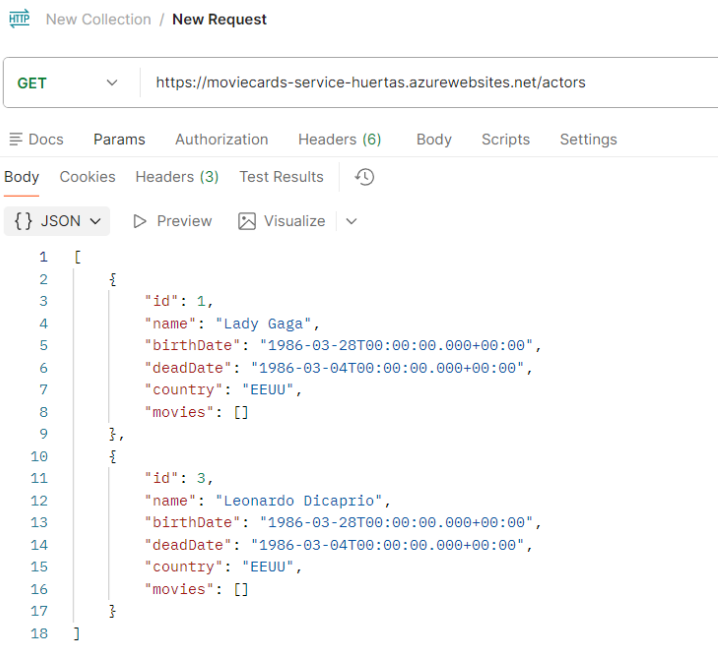
5.3. Validación de la API con Postman

Para garantizar que el servicio procesa correctamente el nuevo campo, se han realizado las siguientes pruebas:

- **Creación de actores con fallecimiento:** Se ejecutaron peticiones POST enviando un cuerpo JSON que incluye el campo "deadDate": "YYYY-MM-DD".



- **Consulta de datos:** Se verificó mediante peticiones GET que el campo deadDate se devuelve correctamente en la respuesta JSON junto con el nombre y la fecha de nacimiento.



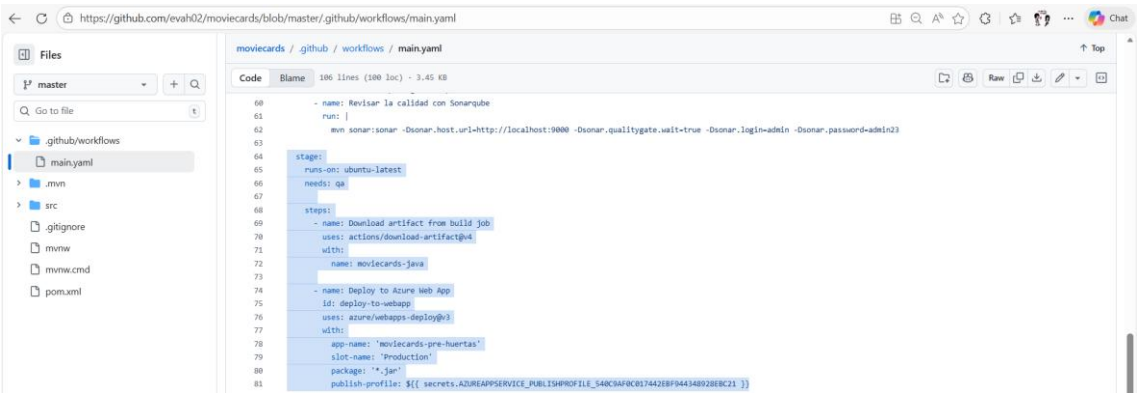
6. Implementación de funcionalidades en moviecards fecha de fallecimiento

En esta fase final, se transformó la aplicación **moviecards** para consumir la nueva información del servicio y se blindó el proceso de despliegue mediante un pipeline de integración continua más robusto.

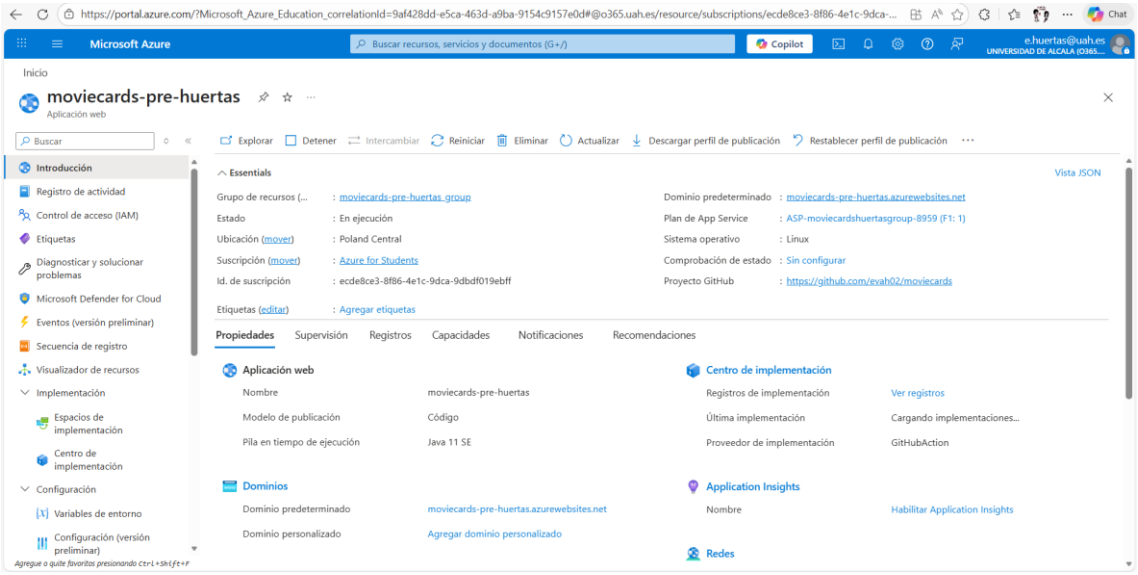
6.1. Configuración del entorno de Pre-producción (Stage)

Se ha modificado el flujo de trabajo de GitHub Actions para añadir una capa de seguridad antes del despliegue final:

- **Nuevo Job "stage":** Se insertó un trabajo llamado stage en el workflow, ubicado estratégicamente entre las fases de "qa" (análisis de calidad) y "deploy" (producción).



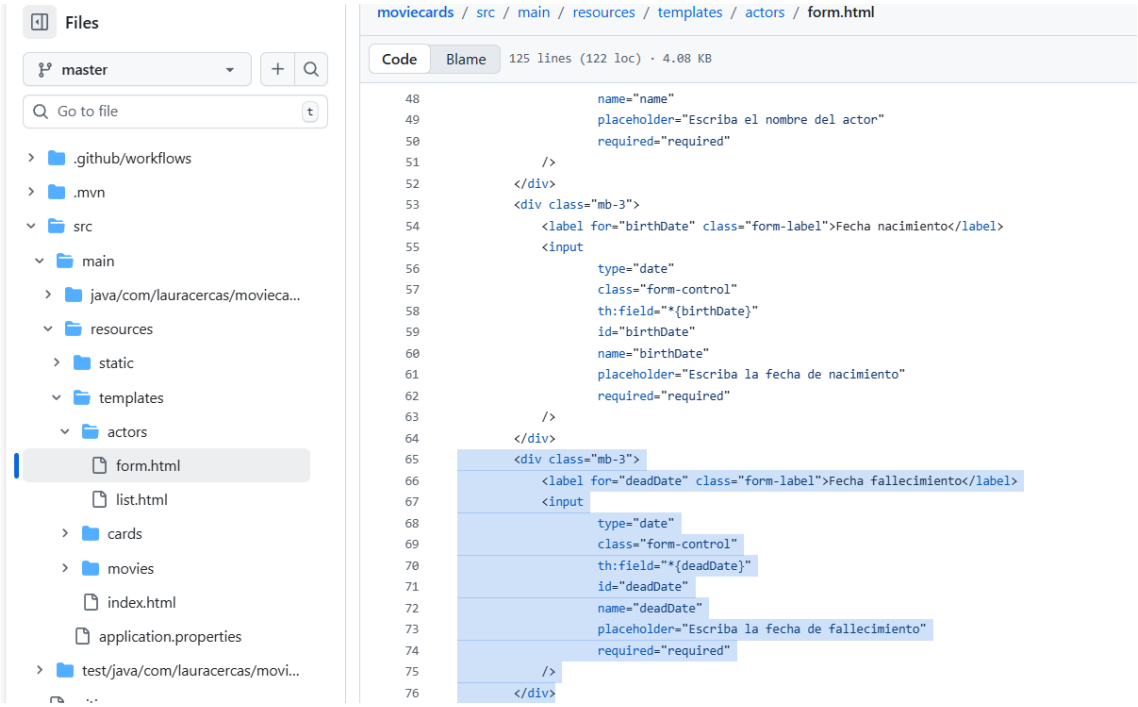
- **Entorno Aislado:** La aplicación se despliega automáticamente en un entorno de pre-producción en Azure (moviecards-pre-huertas.azurewebsites.net) para realizar validaciones finales en un entorno idéntico al real.

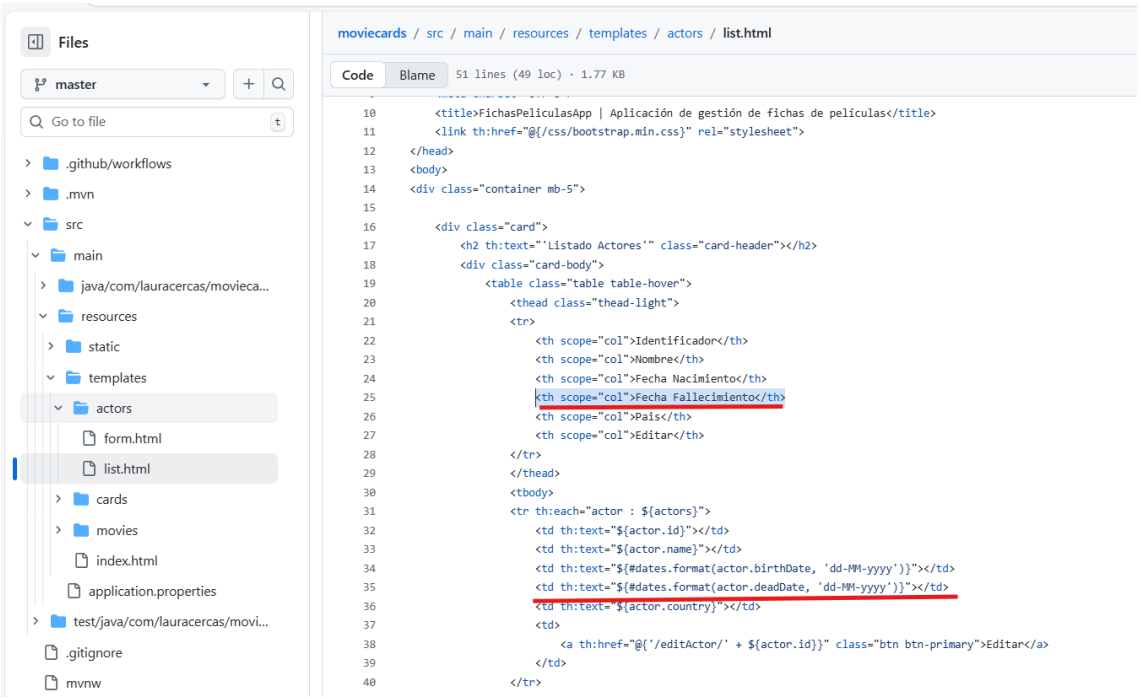


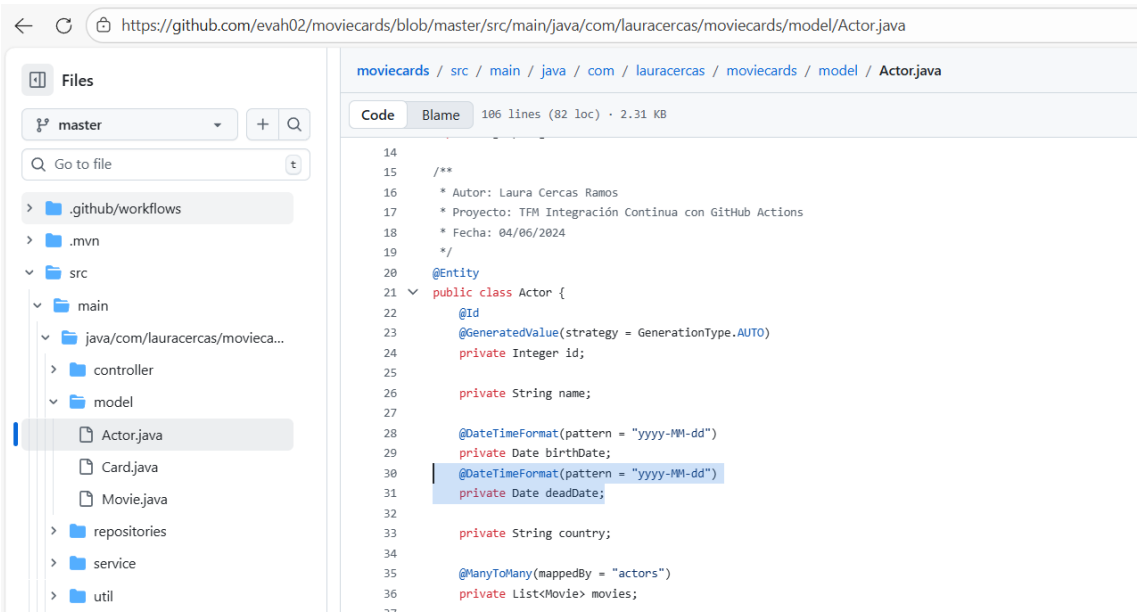
6.2. Actualización de la interfaz y lógica de negocio

Se adaptó la aplicación cliente para manejar el campo deadDate:

- **Interfaz de usuario:** Se modificó el formulario de creación de actores para solicitar la fecha de fallecimiento y se añadió una nueva columna en la tabla de listado de actores para visualizar este dato.





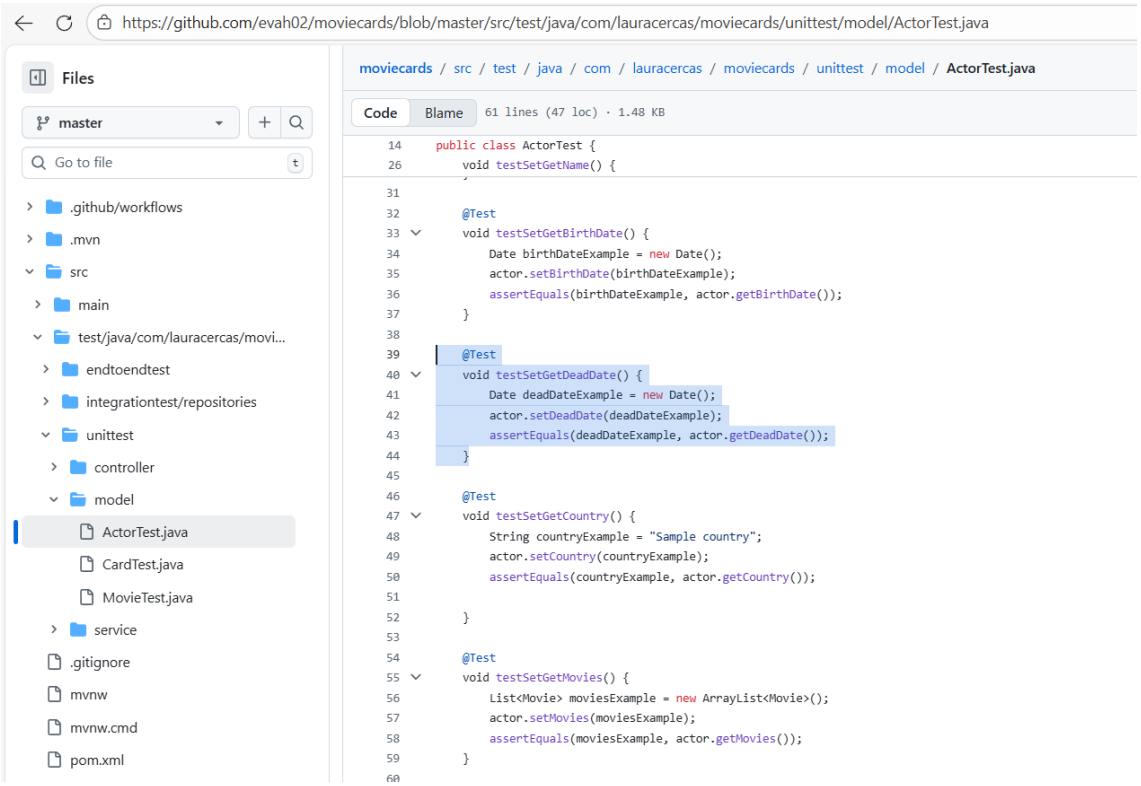


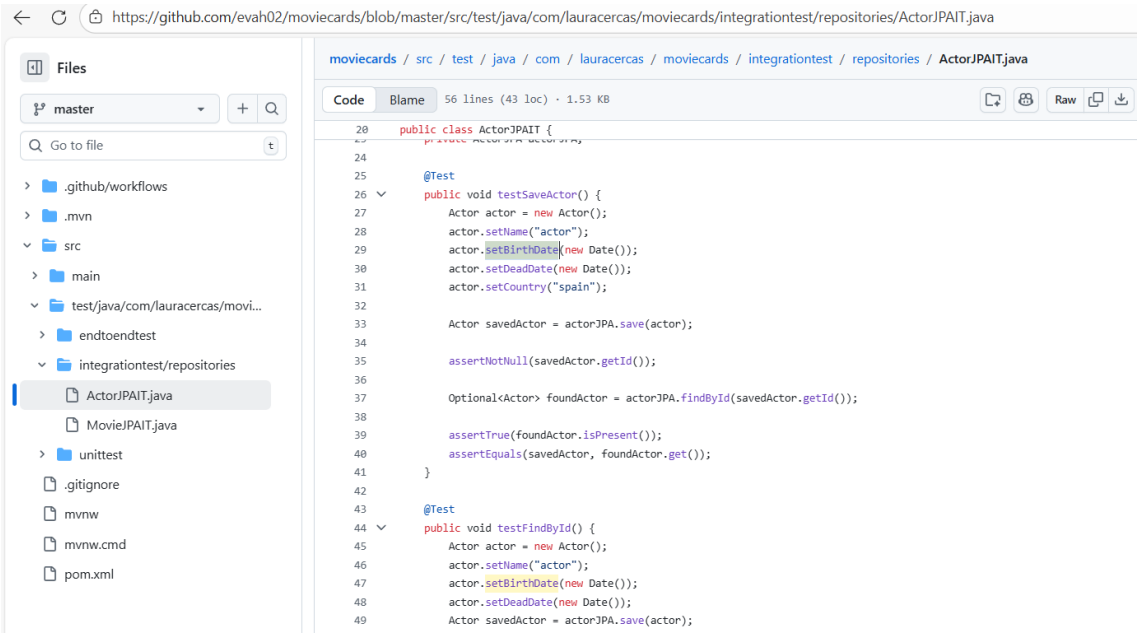
- **Gestión de ramas:** Siguiendo metodologías ágiles, los cambios se desarrollaron en una rama secundaria y, tras verificar su funcionamiento, se integraron en la rama master.

6.3. Pruebas y calidad de código

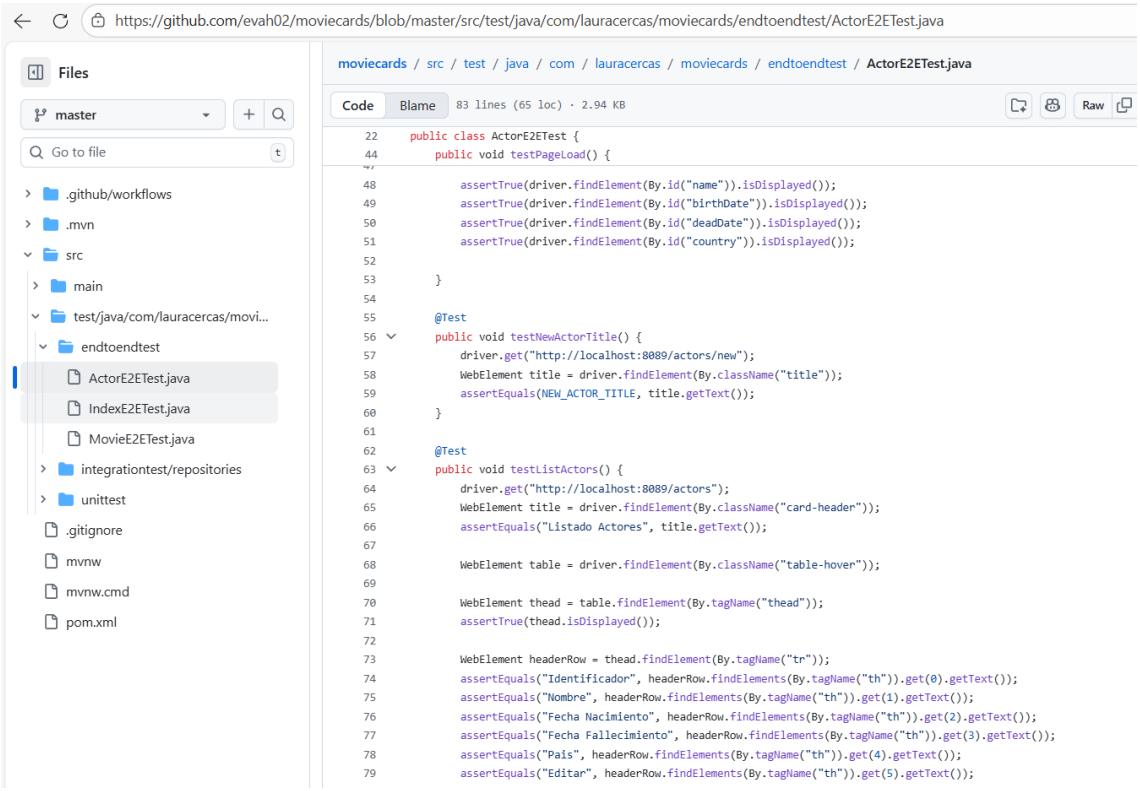
Para asegurar que los cambios no introdujeran errores, se actualizaron todos los niveles de pruebas:

- **Pruebas unitarias e integración:** Se ajustaron los tests para validar que el objeto Actor procesa correctamente el nuevo atributo y que la comunicación con el servicio REST es estable.

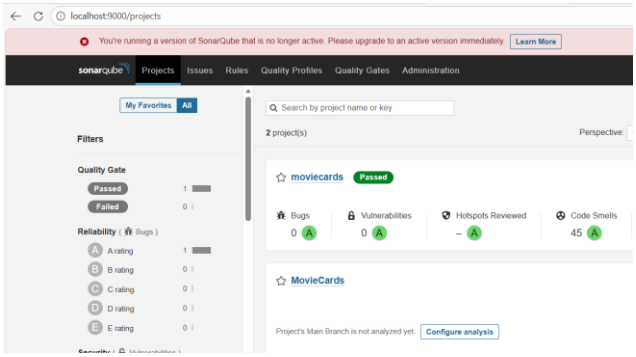


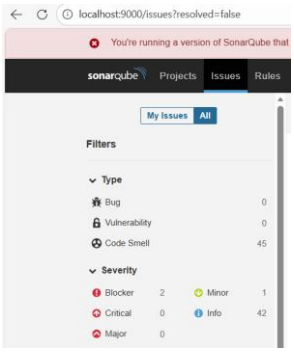


- **Pruebas funcionales (E2E):** Se modificaron las pruebas de extremo a extremo para simular el flujo completo del usuario, desde la inserción de la fecha de fallecimiento hasta su visualización en el listado.

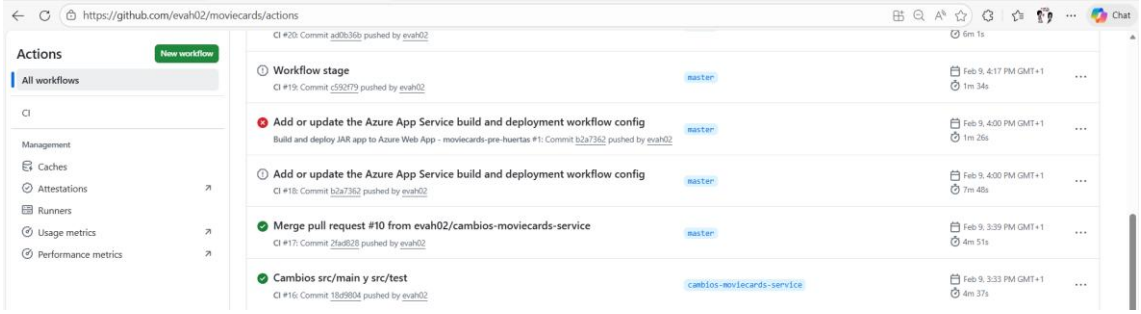


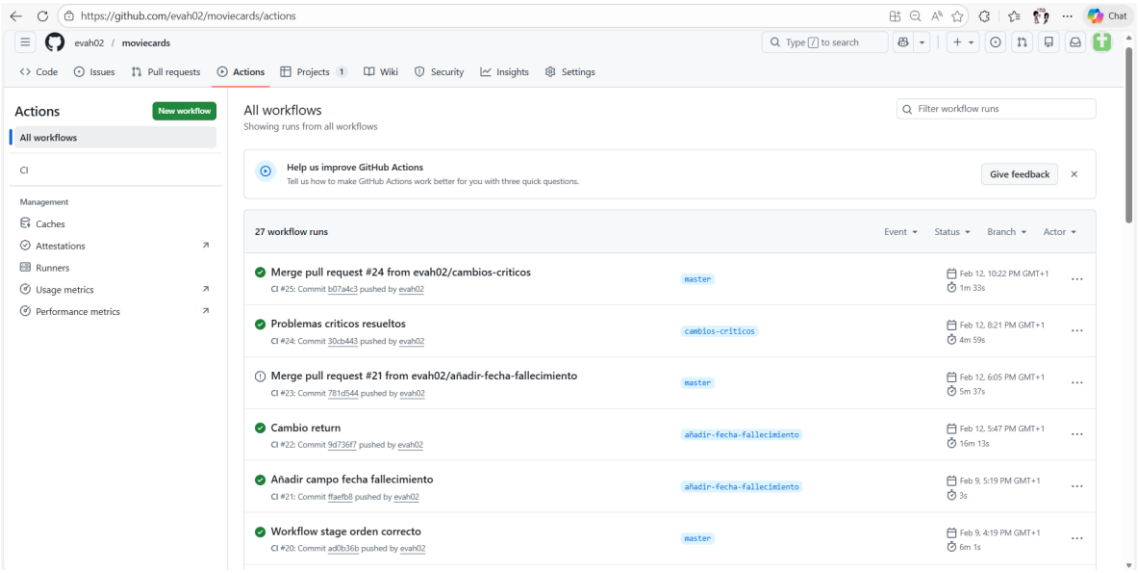
- **Control de Calidad con SonarQube:** Se estableció un umbral crítico de calidad; si el análisis detecta **5 o más problemas críticos**, el pipeline detiene automáticamente el proceso, impidiendo el despliegue a producción.



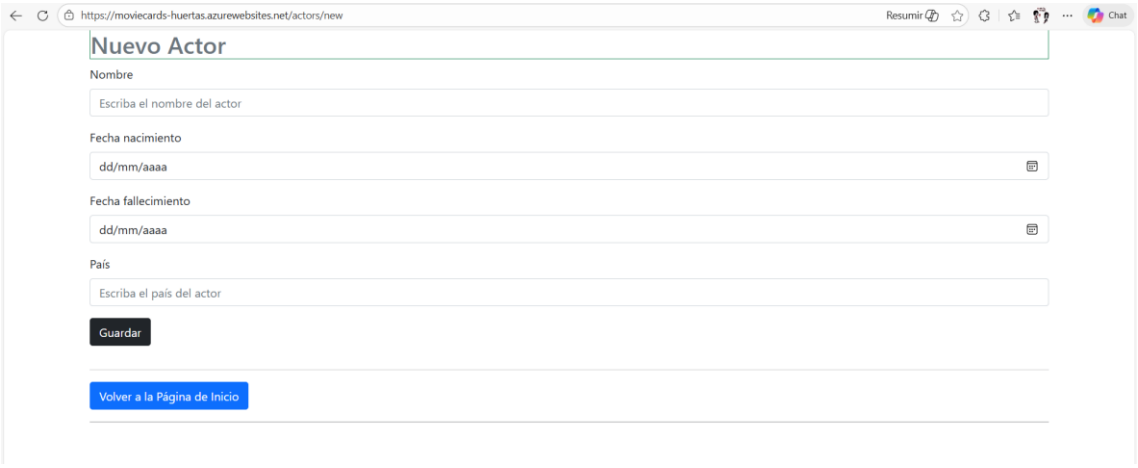


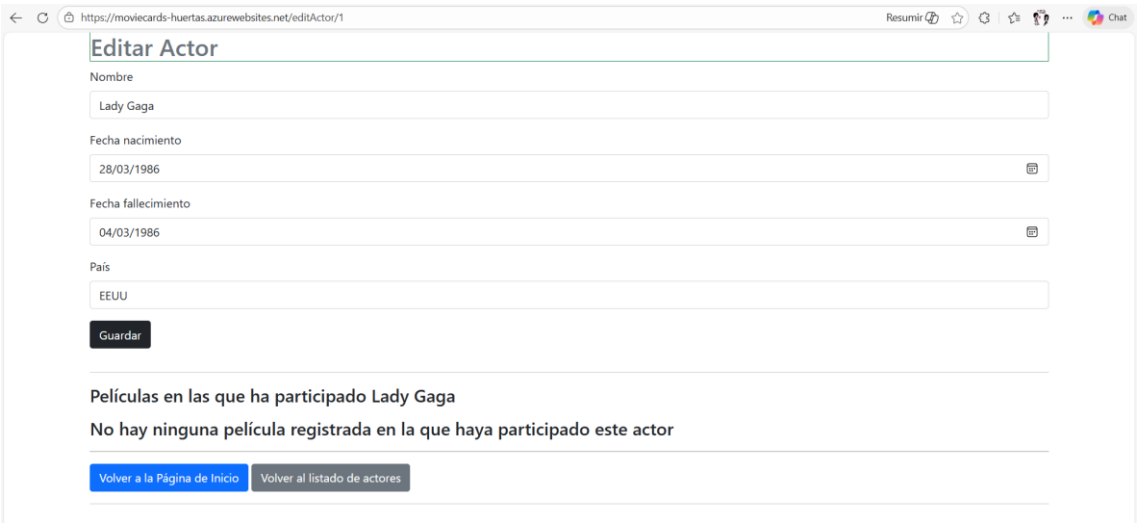
Los actions resultantes de todas las fases mencionadas después de completar la parte de la práctica 5 guiada son los que se muestran a continuación, en los cuales se han ido realizando los cambios requeridos por la práctica.





Finalmente, la aplicación quedaría de la siguiente forma aplicando todos los cambios anteriores.


https://moviecards-huertas.azurewebsites.net/actors/new
Resumir
Chat
Nuevo Actor
Nombre
Escriba el nombre del actor
Fecha nacimiento
dd/mm/aaaa
Fecha fallecimiento
dd/mm/aaaa
País
Escriba el país del actor
Guardar
Volver a la Página de Inicio


https://moviecards-huertas.azurewebsites.net/editActor/1
Resumir
Chat
Editar Actor
Nombre
Lady Gaga
Fecha nacimiento
28/03/1986
Fecha fallecimiento
04/03/1986
País
EEUU
Guardar
Películas en las que ha participado Lady Gaga
No hay ninguna película registrada en la que haya participado este actor
Volver a la Página de Inicio
Volver al listado de actores