



Project Name: **Proyecto
FilmFlow - No Country**

Issue: [FILMFLOW-28](#)

Prepared By: Mateo Joel Lopez

 **CONFIDENTIAL**



TEST PLAN

API TESTING: Peliculas DB



TABLE OF CONTENTS

1.	Introduction.....	5
1.1.	Document Overview	5
1.2.	Test Plan FILMFLOW-28 Details	5
2.	SUMARIO:.....	5
2.1.1.	LEVEL TEST PLAN / FUNCIONALIDAD: THEMOMIEDB - PELICULAS	5
3.	OBJETIVO:.....	5
4.	SUPOSICIONES:	6
5.	SCOPE (FOCO):	8
6.	RIESGOS:	8
7.	TEST DATA (ENDPOINTS A TESTEAR):	10
8.	RECURSOS / ITEMS:	13
8.1.	□RECURSOS DE SISTEMA:	14
9.	TEST DELIVERY - LOS ENTREGABLES DE QA	15
9.1.1.	Entregables que deben ser realizado ANTES DE la fase de TEST EXECUTION:	15
9.1.2.	Entregables que deben ser realizado DURANTE la fase de TEST EXECUTION:	15
9.1.3.	Entregables que deben ser realizado DESPUÉS DE la fase de TEST EXECUTION:	15
9.2.	Requirements covered by the Tests in this Test Plan	15
9.3.	Overall Execution Status	15
9.4.	Defects	17
9.5.	Tests Summary	17
10.	Test Executions	21
10.1.	FILMFLOW-29: US 13 TS - 01 API TESTING Obtener una lista de géneros	21
10.1.1.	FILMFLOW-30: US 13 TS - 01 TC - 01 Lista de generos validos	21
10.1.2.	FILMFLOW-31: US 13 TS - 01 TC - 02 Estado 200 OK.....	27
10.1.3.	FILMFLOW-32: US 13 TS - 01 TC - 03 Objeto JSON	30
10.1.4.	FILMFLOW-33: US 13 TS - 01 TC - 04 Campos id y name	33
10.1.5.	FILMFLOW-34: US 13 TS - 01 TC - 05 Solicitud GET incorrecta	37
10.1.6.	FILMFLOW-35: US 13 TS - 01 TC - 06 Lista de generos NO está vacía.....	41



10.1.7. FILMFLOW-36: US 13 TS - 01 TC - 07 Generos duplicados.....	44
10.2. FILMFLOW-37: US 15 TS 01 Obtener una lista de películas populares.....	48
10.2.1. FILMFLOW-38: US 15 TS 01 TC 01 Verificar estado 200 OK	48
10.2.2. FILMFLOW-39: US 15 TS 01 TC 02 Propiedad 'popularity' en todos los objetos	52
10.2.3. FILMFLOW-40: US 15 TS 01 TC 03 Verificar que la respuesta sea un objeto JSON	55
10.2.4. FILMFLOW-41: US 15 TS 01 TC 04 La propiedad POPULARITY es un numero.....	59
10.2.5. FILMFLOW-42: US 15 TS 01 TC 05 Las peliculas no tienen orden ascendente o descendente	63
10.3. FILMFLOW-43: US 17 TS 01 Validar el formato de las URL de las imágenes.....	67
10.3.1. FILMFLOW-44: US 17 TS 01 TC 01 Verificar imagen válida cuando la URL de la imagen comienza con "https://".	67
10.3.2. FILMFLOW-45: US 17 TS 01 TC 02 Verificar estado 200 cuando se hace la solicitud GET.....	71
10.3.3. FILMFLOW-46: US 17 TS 01 TC 03 Verificar que todas las peliculas tengan la propiedad image	74
10.4. FILMFLOW-47: US 19 TS 01 Obtener una lista de películas	78
10.4.1. FILMFLOW-48: US 19 TS 01 TC 01 Verificar que se reciban 20 películas en la respuesta cuando se realiza la solicitud GET.	78
10.4.2. FILMFLOW-49: US 19 TS 01 TC 02 Verificar que el código de estado de la respuesta sea 200 (Éxito) cuando se realiza la solicitud GET.....	82
10.4.3. FILMFLOW-50: US 19 TS 01 TC 03 Verificar que la respuesta contiene la información requerida para cada película, como título, descripción, fecha de lanzamiento e imagen.	85
10.4.4. FILMFLOW-51: US 19 TS 01 TC4 Verificar que las fechas se devuelven en un formato específico, como "AAAA-MM-DD,"	89
10.5. FILMFLOW-52: US 18 TS 01 Validar valores de "vote_average" en un rango y "vote_count" mínimo	90
10.5.1. FILMFLOW-53: US 18 TS 01 TC 01 Endpoint accesible	90
10.5.2. FILMFLOW-57: US 18 TS 01 TC 05 Verificar array vacio.....	93
10.5.3. FILMFLOW-56: US 18 TS 01 TC 04 Campo vote_count valido	96
10.5.4. FILMFLOW-55: US 18 TS 01 TC 03 vote_Average valido.....	100
10.5.5. FILMFLOW-54: US 18 TS 01 TC 02 Array de peliculas	104
10.6. FILMFLOW-58: US 16 TS 01 Obtener una lista de próximos estrenos	107
10.6.1. FILMFLOW-59: US 16 TS 01 TC 01 Formato de fechas de lanzamiento	107
10.6.2. FILMFLOW-61: US 16 TS 01 TC 03 Propiedad release_date	111
10.6.3. FILMFLOW-60: US 16 TS 01 TC 02 Fechas de lanzamiento pasadas.....	115
10.7. FILMFLOW-68: US 14 TS 01 Verificacion de pertenencia.....	119
10.7.1. FILMFLOW-70: US 14 TS 01 TC 01 Genero Accion.....	119
10.7.2. FILMFLOW-71: US 14 TS 01 TC 02 Genero Aventura	122
10.7.3. FILMFLOW-72: US 14 TS 01 TC 03 Genero Animacion.....	124
10.7.4. FILMFLOW-73: US 14 TS 01 TC 04 Genero comedia.....	127
10.7.5. FILMFLOW-74: US 14 TS 01 TC 05 Genero crimen.....	129



10.7.6. FILMFLOW-75:	US 14 TS 01 TC 06 Genero documental.....	132
10.7.7. FILMFLOW-76:	US 14 TS 01 TC 07 Genero drama	135
10.7.8. FILMFLOW-77:	US 14 TS 01 TC 08 Genero familia	137
10.7.9. FILMFLOW-78:	US 14 TS 01 TC 09 Genero fantasia	140
10.7.10. FILMFLOW-79:	US 14 TS 01 TC 10 Genero historia	142
10.7.11. FILMFLOW-80:	US 14 TS 01 TC 11 Genero terror	144
10.7.12. FILMFLOW-81:	US 14 TS 01 TC 12 Genero musica	147
10.7.13. FILMFLOW-82:	US 14 TS 01 TC 13 Genero misterio	150
10.7.14. FILMFLOW-83:	US 14 TS 01 TC 14 Genero romance	152
10.7.15. FILMFLOW-84:	US 14 TS 01 TC 15 Genero ciencia ficcion	155
10.7.16. FILMFLOW-85:	US 14 TS 01 TC 16 Genero ciencia pelicula de tv	157
10.7.17. FILMFLOW-86:	US 14 TS 01 TC 17 Genero ciencia suspense	160
10.7.18. FILMFLOW-87:	US 14 TS 01 TC 18 Genero belica	162
10.7.19. FILMFLOW-88:	US 14 TS 01 TC 19 Genero western	164
10.8. FILMFLOW-96:	US 20 TS 01 Obtener una lista de generos (Series de TV)	167
10.8.1. FILMFLOW-97:	US 20 TS 01 TC 01 Generos validos	167
10.8.2. FILMFLOW-98:	US 20 TS 01 TC 02 EndPoint accesible	169
10.8.3. FILMFLOW-99:	US 20 TS 01 TC 03 Array de objetos	172
10.8.4. FILMFLOW-100:	US 20 TS 01 TC 04 Campos válidos	174
10.8.5. FILMFLOW-101:	US 20 TS 01 TC 05 Lista de series vacias	176
10.9. FILMFLOW-103:	US 21 TS 01 Verificación de Disponibilidad de Imágenes de Portada de Series	178
10.9.1. FILMFLOW-104:	US 21 TS 01 TC 01 Verificación de Disponibilidad de Imágenes.....	178
11. Appendix A: Approval.....		182



1. Introduction

1.1. Document Overview

This test report provides a summary of a test plan. This document has been generated automatically from the test management platform.

1.2. Test Plan FILMFLOW-28 Details

Begin Date	End Date
Summary	Description
API TESTING: Peliculas DB	<p>1.2.1.1. <i>PLAN DE PRUEBAS FUNCIONAL NIVEL API TESTING</i></p> <p>2. SUMARIO:</p> <p>2.1.1. LEVEL TEST PLAN / FUNCIONALIDAD: EDB - PELICULAS</p> <ul style="list-style-type: none">Este plan se llevará a cabo a nivel de pruebas de INTEGRACIÓN.Este plan se llevará a cabo en POSTMAN de manera automatizada.Este plan se llevará a regresión.Este plan se realizara para determinar QUE pruebas se harán y COMO se realizarán las pruebas <p>3. OBJETIVO:</p> <p>Validar las funciones básicas para encontrar peliculas, encontrar peliculas populares y filtrado por género usando POSTMAN de forma automatizada.</p>



4. SUPOSICIONES:

- Suponemos que la API permite la capacidad de ordenar las películas de acuerdo con diferentes parámetros, como por popularidad o fecha de lanzamiento, y que las pruebas verificarán que el ordenamiento personalizado funciona según lo previsto.
- Suponemos que al solicitar películas a través de la API, se obtendrá un número específico de resultados (por ejemplo, 20 películas), y las pruebas validarán que la cantidad de resultados coincida con la configuración esperada.
- Si la API devuelve fechas, suponemos que las fechas se devuelven en un formato específico, como "AAAA-MM-DD," y las pruebas verificarán que las fechas sigan este formato.
- Suponemos que las URL de las imágenes se devuelven en un formato específico, como una URL completa a una imagen, y las pruebas validarán que las URL de las imágenes sean válidas.
- Suponemos que cada película tiene un identificador único (por ejemplo, el campo "id") y que las pruebas verificarán que no hay duplicados en los resultados.
- Suponemos que los géneros proporcionados por la API son válidos y coinciden con los géneros en una base de datos central. Las pruebas confirmarán que los géneros son coherentes.
- Suponemos que la función de búsqueda en la API funciona correctamente, y las pruebas verificarán que



se pueden encontrar películas utilizando términos de búsqueda válidos.

- Suponemos que cada película en los resultados contiene información completa y que las pruebas validarán que los campos esenciales NOMBRE, IMAGEN Y DESCRIPCIÓN están presentes en cada respuesta.
- Suponemos que la base de datos de películas en la API puede cambiar con el tiempo debido a actualizaciones o adiciones de nuevas películas, y las pruebas considerarán estos cambios.
- Suponemos que "vote_average" es un valor numérico que representa la calificación promedio de una película en un rango válido, generalmente entre 0 y 10. Las pruebas validarán que los valores estén dentro de este rango.
- Suponemos que "vote_count" es un valor numérico que representa la cantidad de votos o calificaciones que ha recibido una película. Las pruebas verificarán que el valor sea igual o mayor que un valor mínimo (por ejemplo, 0 o 1, indicando que al menos una persona ha calificado la película).
- Suponemos que el endpoint de **Películas Populares** devuelve una lista de películas populares en función de ciertos criterios de popularidad. Las pruebas verificarán que las películas en la lista cumplan con criterios específicos de popularidad y que estén clasificadas de acuerdo con esos criterios.
- Suponemos que el endpoint de **Próximos Estrenos** proporciona una lista de películas que se lanzarán próximamente. Las pruebas verificarán que las



películas incluidas sean de hecho próximos estrenos y que las fechas de lanzamiento cumplan con las expectativas.

- Suponemos que las películas próximas a estrenarse pertenecen a géneros apropiados para un lanzamiento en el futuro. Las pruebas validarán que las películas sean coherentes con esta suposición.
- Suponemos que el endpoint de **Películas Mejor Valoradas** devuelve una lista de las películas mejor valoradas en un momento dado. Las pruebas verificarán que las películas incluidas tengan calificaciones que las califiquen como "mejor valoradas" y que estén ordenadas en consecuencia.

5. SCOPE (FOCO):

- Capacidad de alcanzar solamente las funciones CRUD descritas en los criterios de aceptación de las historias de usuario.
- NO se VALIDARÁN los campos que no fueron descritos en la FUNCIONALIDAD.

6. RIESGOS:

RIESGO - (LO QUE SE NECESITA SOLVENTAR)	MITIGACIÓN - (CON LO QUE SE PUEDE SOLVENTAR).
.	



No poder guardar el TOKEN de TM DB u olvidarla	<ul style="list-style-type: none">🎬 Solicitar el TOKEN al PM o DEV del Backend.🎬 Guardarlo en un archivo con nombre .env en la carpeta Backend del repositorio.🎬 Guardarlo en una variable en POSTMAN
No conocer como usar la API	Solicitar capacitación y documentación al PM o a los desarrolladores Backend sobre la misma
No tener acceso al Workspace de POSTMAN	Crear un workspace e invitar a los Tester QA para que puedan trabajar en él.
Falta de acceso a internet	Asegurarse de que el equipo de pruebas tenga acceso a Internet estable durante todo el proceso de pruebas.
Cambios no documentados en la API	Establecer una comunicación efectiva con el equipo de desarrollo para recibir notificaciones sobre cambios planificados en la API. Además, se debe monitorear la documentación y las actualizaciones de la API de TMDb para estar al tanto de los cambios.



Desconexion de recursos de desarrollo	Mantener una comunicación constante con el equipo de desarrollo para asegurarse de que los recursos y entornos necesarios para las pruebas estén disponibles y en línea.
Cambio en la estructura de datos de la API	Establecer una estrategia de pruebas sólida que incluya pruebas de regresión para verificar que la API sigue funcionando después de cambios en la estructura de datos.

7. TEST DATA (ENDPOINTS A REAR):

DESCRIPCION	RUTA ENDPOINT
LISTA DE 20 PELICULAS.	http://localhost:4000/getmovies
LISTA DE GÉNEROS.	http://localhost:4000/filters/genres
LISTA DE PELICULAS QUE COINCIDAN CON EL FILTRO SOLICITADO	http://localhost:4000/filters/genre/28



(Acción en este caso).	
LISTA DE PELICULAS POPULARES	http://localhost:4000/list/popmovies
LISTA DE PROXIMOS ESTRENOS	http://localhost:4000/list/upcomings
LISTA DE PELICULAS MAS VALORADAS	http://localhost:4000/list/top
LISTA CON ENDPOINTS DE GENEROS DE PELICULAS	<ol style="list-style-type: none">1. Acción (28): http://localhost:4000/filters/genre/282. Aventura (12): http://localhost:4000/filters/genre/123. Animación (16): http://localhost:4000/filters/genre/164. Comedia (35): http://localhost:4000/filters/genre/355. Crimen (80): http://localhost:4000/filters/genre/806. Documental (99): http://localhost:4000/filters/genre/997. Drama (18): http://localhost:4000/filters/genre/18



- | | |
|--|---|
| | <ul style="list-style-type: none">8. Familia (10751):
http://localhost:4000/filters/genre/107519. Fantasía (14):
http://localhost:4000/filters/genre/1410. Historia (36):
http://localhost:4000/filters/genre/3611. Terror (27):
http://localhost:4000/filters/genre/2712. Música (10402):
http://localhost:4000/filters/genre/1040213. Misterio (9648):
http://localhost:4000/filters/genre/964814. Romance (10749):
http://localhost:4000/filters/genre/1074915. Ciencia ficción (878):
http://localhost:4000/filters/genre/87816. Película de TV (10770):
http://localhost:4000/filters/genre/1077017. Suspense (53):
http://localhost:4000/filters/genre/5318. Bélica (10752):
http://localhost:4000/filters/genre/10752 |
|--|---|



	19. Western (37): http://localhost:4000/filters/genre/37
GENEROS DE SERIES DE TV	http://localhost:4000/filters/genres?tv=true
PAGINAS CON SERIES DE TV	<ol style="list-style-type: none">1. http://localhost:4000/tv?page=12. http://localhost:4000/tv?page=23. http://localhost:4000/tv?page=34. http://localhost:4000/tv?page=45. http://localhost:4000/tv?page=56. http://localhost:4000/tv?page=67. http://localhost:4000/tv?page=78. http://localhost:4000/tv?page=89. http://localhost:4000/tv?page=910. http://localhost:4000/tv?page=10

8. RECURSOS / ITEMS:

- 🎬 POSTMAN: Como herramienta principal para probar las API
- 🎬 Documentación del proyecto: Como fuente de guía para saber como usar las API.
- 🎬 A DIFERENCIA de UI testing, (que serian mas casos de pruebas y mas pasos de prueba), el API TESTING podra generar menos casos de prueba y menos pasos de prueba.



- 📄 Test plan (SOLO EN JIRA CONFLUENCE) para planificación de las pruebas

8.1. ☐ RECURSOS DE SISTEMA:

HERRAMIENTAS DEL PROYECTO	DESCRIPCIÓN DEL USO
SLACK	Para comunicación fluida, corrida, actualización de temas, noticias, y más.
Jira con XRAY	Herramienta principal, gestor de incidencias para trabajar y seguir las tareas. CLICK PARA ACCEDER AL ESPACIO DE TRABAJO JIRA
CONFLUENCE	Herramienta de Documentación definitiva, Docs de Jira, Proyecto, Plans, etc. CLICK PARA ACCEDER A LA DOCUMENTACION EN JIRA CONFLUENCE
DISCORD	Para realizar las Planning, resolver problemas rápido, etc. CLICK PARA ACCESO A CANAL DE DISCORD
POSTMAN	Herramienta principal para probar las API CLICK PARA ACCESO AL ESPACIO DE TRABAJO EN POSTMAN (QA)



9. TEST DELIVERY - LOS ENTREGABLES DE QA

(Tareas por realizar en cada etapa del Testing)

9.1.1. Entregables que deben ser realizado ANTES DE la TEST EXECUTION:

- 📄 Documento de plan de pruebas
- 📄 Documento de casos de prueba

9.1.2. Entregables que deben ser realizado DURANTE la TEST EXECUTION:

- 📄 Test Scripts
- 📄 Documento: Matriz de trazabilidad
- 📄 Error logs y logs de ejecucion.

9.1.3. Entregables que deben ser realizado DESPUÉS DE TEST EXECUTION:

- 📄 Resultados de test/reportes
- 📄 Reporte de defectos
- 📄 Instalacion/ Guias de procedimiento para los test

9.2. Requirements covered by the Tests in this Test Plan

Key	Summary	Workflow Status
-----	---------	-----------------

9.3. Overall Execution Status

Of the 52 Tests contained on FILMFLOW-28:

Status	#Tests	Percentage
--------	--------	------------



TODO	0	0%
EXECUTING	0	0%
PASS	50	96.2%
FAIL	2	3.8%
ABORTED	0	0%



9.4. Defects

Key	Summary	Priority
FILMFLOW-62	API Peliculas a estrenar Hay peliculas con fechas pasadas	Medium
FILMFLOW-105	API Hay series que no tienen imagen disponible	Medium

9.5. Tests Summary

Key	Summary	Issue Assignee	Requirements	Test Executions	Latest Status
FILMFLOW-60	US 16 TS 01 TC 02 Fechas de lanzamiento pasadas	Mateo Joel Lopez		FILMFLOW-58	FAILED
FILMFLOW-104	US 21 TS 01 TC 01 Verificación de Disponibilidad de Imágenes	Mateo Joel Lopez		FILMFLOW-103	FAILED
FILMFLOW-30	US 13 TS - 01 TC - 01 Lista de generos validos	Mateo Joel Lopez		FILMFLOW-29	PASSED
FILMFLOW-31	US 13 TS - 01 TC - 02 Estado 200 OK	Mateo Joel Lopez		FILMFLOW-29	PASSED
FILMFLOW-32	US 13 TS - 01 TC - 03 Objeto JSON	Mateo Joel Lopez		FILMFLOW-29	PASSED
FILMFLOW-33	US 13 TS - 01 TC - 04 Campos id y name	Mateo Joel Lopez		FILMFLOW-29	PASSED
FILMFLOW-34	US 13 TS - 01 TC - 05 Solicitud GET incorrecta	Mateo Joel Lopez		FILMFLOW-29	PASSED



FILMFLO W-35	US 13 TS - 01 TC - 06 Lista de generos NO está vacía.	Mateo Joel Lopez		FILMFLOW-29	PASSED
FILMFLO W-36	US 13 TS - 01 TC - 07 Generos duplicados.	Mateo Joel Lopez		FILMFLOW-29	PASSED
FILMFLO W-38	US 15 TS 01 TC 01 Verificar estado 200 OK	Mateo Joel Lopez		FILMFLOW-37	PASSED
FILMFLO W-39	US 15 TS 01 TC 02 Propiedad 'popularity' en todos los objetos	Mateo Joel Lopez		FILMFLOW-37	PASSED
FILMFLO W-40	US 15 TS 01 TC 03 Verificar que la respuesta sea un objeto JSON	Mateo Joel Lopez		FILMFLOW-37	PASSED
FILMFLO W-41	US 15 TS 01 TC 04 La propiedad POPULARITY es un numero	Mateo Joel Lopez		FILMFLOW-37	PASSED
FILMFLO W-42	US 15 TS 01 TC 05 Las peliculas no tienen orden ascendente o descendente	Mateo Joel Lopez		FILMFLOW-37	PASSED
FILMFLO W-44	US 17 TS 01 TC 01 Verificar imagen válida cuando la URL de la imagen comienza con "https://".	Mateo Joel Lopez		FILMFLOW-43	PASSED
FILMFLO W-45	US 17 TS 01 TC 02 Verificar estado 200 cuando se hace la solicitud GET.	Mateo Joel Lopez		FILMFLOW-43	PASSED
FILMFLO W-46	US 17 TS 01 TC 03 Verificar que todas las peliculas tengan la propiedad image	Mateo Joel Lopez		FILMFLOW-43	PASSED
FILMFLO W-48	US 19 TS 01 TC 01 Verificar que se reciban 20 películas en la respuesta cuando se realiza la solicitud GET.	Mateo Joel Lopez		FILMFLOW-47	PASSED
FILMFLO W-49	US 19 TS 01 TC 02 Verificar que el código de estado de la respuesta sea 200 (Éxito) cuando se realiza la solicitud GET.	Mateo Joel Lopez		FILMFLOW-47	PASSED



FILMFLOW-50	US 19 TS 01 TC 03 Verificar que la respuesta contiene la información requerida para cada película, como título, descripción, fecha de lanzamiento e imagen.	Mateo Joel Lopez		FILMFLOW-47	PASSED
FILMFLOW-51	US 19 TS 01 TC4 Verificar que las fechas se devuelven en un formato específico, como "AAAA-MM-DD,"	Mateo Joel Lopez		FILMFLOW-47	PASSED
FILMFLOW-53	US 18 TS 01 TC 01 Endpoint accesible	Mateo Joel Lopez		FILMFLOW-52	PASSED
FILMFLOW-57	US 18 TS 01 TC 05 Verificar array vacío	Mateo Joel Lopez		FILMFLOW-52	PASSED
FILMFLOW-56	US 18 TS 01 TC 04 Campo vote_count válido	Mateo Joel Lopez		FILMFLOW-52	PASSED
FILMFLOW-55	US 18 TS 01 TC 03 vote_Average válido	Mateo Joel Lopez		FILMFLOW-52	PASSED
FILMFLOW-54	US 18 TS 01 TC 02 Array de películas	Mateo Joel Lopez		FILMFLOW-52	PASSED
FILMFLOW-59	US 16 TS 01 TC 01 Formato de fechas de lanzamiento	Mateo Joel Lopez		FILMFLOW-58	PASSED
FILMFLOW-61	US 16 TS 01 TC 03 Propiedad release_date	Mateo Joel Lopez		FILMFLOW-58	PASSED
FILMFLOW-70	US 14 TS 01 TC 01 Género Acción	Mateo Joel Lopez		FILMFLOW-68	PASSED
FILMFLOW-71	US 14 TS 01 TC 02 Género Aventura	Mateo Joel Lopez		FILMFLOW-68	PASSED
FILMFLOW-72	US 14 TS 01 TC 03 Género Animación	Mateo Joel Lopez		FILMFLOW-68	PASSED



FILMFLOW-W-73	US 14 TS 01 TC 04 Genero comedia	Mateo Joel Lopez		FILMFLOW-68	PASSED
FILMFLOW-W-74	US 14 TS 01 TC 05 Genero crimen	Mateo Joel Lopez		FILMFLOW-68	PASSED
FILMFLOW-W-75	US 14 TS 01 TC 06 Genero documental	Mateo Joel Lopez		FILMFLOW-68	PASSED
FILMFLOW-W-76	US 14 TS 01 TC 07 Genero drama	Mateo Joel Lopez		FILMFLOW-68	PASSED
FILMFLOW-W-77	US 14 TS 01 TC 08 Genero familia	Mateo Joel Lopez		FILMFLOW-68	PASSED
FILMFLOW-W-78	US 14 TS 01 TC 09 Genero fantasia	Mateo Joel Lopez		FILMFLOW-68	PASSED
FILMFLOW-W-79	US 14 TS 01 TC 10 Genero historia	Mateo Joel Lopez		FILMFLOW-68	PASSED
FILMFLOW-W-80	US 14 TS 01 TC 11 Genero terror	Mateo Joel Lopez		FILMFLOW-68	PASSED
FILMFLOW-W-81	US 14 TS 01 TC 12 Genero musica	Mateo Joel Lopez		FILMFLOW-68	PASSED
FILMFLOW-W-82	US 14 TS 01 TC 13 Genero misterio	Mateo Joel Lopez		FILMFLOW-68	PASSED
FILMFLOW-W-83	US 14 TS 01 TC 14 Genero romance	Mateo Joel Lopez		FILMFLOW-68	PASSED
FILMFLOW-W-84	US 14 TS 01 TC 15 Genero ciencia ficcion	Mateo Joel Lopez		FILMFLOW-68	PASSED
FILMFLOW-W-85	US 14 TS 01 TC 16 Genero ciencia pelicula de tv	Mateo Joel Lopez		FILMFLOW-68	PASSED



FILMFLO W-86	US 14 TS 01 TC 17 Genero ciencia suspense	Mateo Joel Lopez		FILMFLOW-68	PASSED
FILMFLO W-87	US 14 TS 01 TC 18 Genero belica	Mateo Joel Lopez		FILMFLOW-68	PASSED
FILMFLO W-88	US 14 TS 01 TC 19 Genero western	Mateo Joel Lopez		FILMFLOW-68	PASSED
FILMFLO W-97	US 20 TS 01 TC 01 Generos validos	Mateo Joel Lopez		FILMFLOW-96	PASSED
FILMFLO W-98	US 20 TS 01 TC 02 EndPoint accesible	Mateo Joel Lopez		FILMFLOW-96	PASSED
FILMFLO W-99	US 20 TS 01 TC 03 Array de objetos	Mateo Joel Lopez		FILMFLOW-96	PASSED
FILMFLO W-100	US 20 TS 01 TC 04 Campos válidos	Mateo Joel Lopez		FILMFLOW-96	PASSED
FILMFLO W-101	US 20 TS 01 TC 05 Lista de series vacias	Mateo Joel Lopez		FILMFLOW-96	PASSED

10. Test Executions

10.1. [FILMFLOW-29](#): US 13 | TS - 01 | API TESTING | Obtener una lista de géneros

10.1.1. [FILMFLOW-30](#): US 13 | TS - 01 | TC - 01 | Lista de generos validos

Execution Status	Assignee	Executed By	Started On	Finished On
------------------	----------	-------------	------------	-------------



PASSED	Mateo Joel Lopez	Mateo Joel Lopez		
--------	------------------	------------------	--	--

10.1.1.1. Execution Defects

There are 0 Execution Defects on this Test Run.

10.1.1.2. Execution Evidence

There are 0 Execution Evidence on this Test Run.

10.1.1.3. Comment

None.

10.1.1.4. Test Description

Verificar que se reciba una lista de géneros válidos cuando se envía una solicitud GET a "http://localhost:4000/filters/genres".

10.1.1.5. Test Issue Attachments

FILMFLOW-30 doesn't have attachments.

10.1.1.6. Test Details

This Manual Test has 4 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------



1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	<p>COLECCION: DB PELICULAS TESTING</p> <p>SOLICITUD: GET: US 13 API TESTING Obtener una lista de géneros</p> <p>URL: [http://localhost:4000/filters/genres] http://localhost:4000/filters/genres]</p>	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 13 API TESTING Obtener una lista de géneros" en tu colección "DB PELICULAS TESTING".		<p>Visualizar la ventana correspondiente a la solicitud GET</p> <p>US 13 API TESTING Obtener una lista de géneros</p>					PASSED



3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{noformat}// TC1: Verificar que se r eciba una lista de géneros válidos pm.test("Verificar que se reciba una lista de géneros válidos", function () { // Realizar la co mprobación del c ódigo de estado HTTP pm.response.to .have.status(200); // Parsear la re spuesta JSON var jsonRespo nse = pm.respons e.json(); // Verificar que la respuesta sea un objeto JSON pm.expect(json Response).to.be. an('array'); // Verificar que cada género cont enga al menos lo s campos "id" y "n</pre>	Lograr ver correctamente el script en la pestaña TESTS					PASSED
---	---	--	--	--	--	--	--	--------




		<pre>ame" pm.expect(json Response).to.be. an('array').that.ha s.property(0).that. has.all.keys('id', 'n ame'); // Verificar que los géneros en la lista sean válidos var validGenre s = [{ "id": 28, "na me": "Acción" }, { "id": 12, "na me": "Aventura" }, { "id": 16, "na me": "Animación" }, { "id": 35, "na me": "Comedia" }, { "id": 80, "na me": "Crimen" }, { "id": 99, "na me": "Documental " }, { "id": 18, "na me": "Drama" }, { "id": 10751, "name": "Familia" }, { "id": 14, "na</pre>						
--	--	---	--	--	--	--	--	--



		<pre>me": "Fantasía" }, { "id": 36, "name": "Historia" }, { "id": 27, "name": "Terror" }, { "id": 10402, "name": "Música" }, { "id": 9648, "name": "Misterio" }, { "id": 10749, "name": "Romance" }, { "id": 878, "name": "Ciencia ficción" }, { "id": 10770, "name": "Película de TV" }, { "id": 53, "name": "Suspense" }, { "id": 10752, "name": "Bélica" }, { "id": 37, "name": "Western" }]; pm.expect(jsonResponse).toEqual(validGenres);</pre>					
--	--	--	--	--	--	--	--



		}); {noformat}						
4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: Verificar que se reciba una lista de géneros válidos, tenga el estado PASS					

10.1.1.7. Requirements linked with this test
None.

10.1.2. [FILMFLOW-31](#): US 13 | TS - 01 | TC - 02 | Estado 200 OK

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.1.2.1. Execution Defects
There are 0 Execution Defects on this Test Run.

10.1.2.2. Execution Evidence
There are 0 Execution Evidence on this Test Run.



10.1.2.3. Comment

None.

10.1.2.4. Test Description

Verificar que se reciba una lista de géneros válidos cuando se envía una solicitud GET a "http://localhost:4000/filters/genres".

10.1.2.5. Test Issue Attachments

FILMFLOW-31 doesn't have attachments.

10.1.2.6. Test Details


This Manual Test has 4 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------



1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	<p>COLECCION: DB PELICULAS TESTING</p> <p>SOLICITUD: GET: US 13 API TESTING Obtener una lista de géneros</p> <p>URL: [http://localhost:4000/filters/genres] http://localhost:4000/filters/genres]</p>	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 13 API TESTING Obtener una lista de géneros" en tu colección "DB PELICULAS TESTING".		<p>Visualizar la ventana correspondiente a la solicitud GET</p> <p>US 13 API TESTING Obtener una lista de géneros</p>					PASSED



3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{code:javascript}// TC2: VERIFICAR STATUS 200 pm.test("Verificar código de estado HTTP 200", function () { pm.response.to.h ave.status(200); });{code}</pre>	Lograr ver correctamente el script en la pestaña TESTS					PASSED
4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar código de estado HTTP 200", tenga el estado PASS					PASSED

10.1.2.7. *Requirements linked with this test*
None.

10.1.3. [FILMFLOW-32](#): US 13 | TS - 01 | TC - 03 | Objeto JSON

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		



10.1.3.1. Execution Defects

There are 0 Execution Defects on this Test Run.

10.1.3.2. Execution Evidence

There are 0 Execution Evidence on this Test Run.

10.1.3.3. Comment

None.

10.1.3.4. Test Description

Verificar que la respuesta sea un objeto JSON cuando se envía una solicitud GET a "http://localhost:4000/filters/genres".

10.1.3.5. Test Issue Attachments

FILMFLOW-32 doesn't have attachments.

10.1.3.6. Test Details

This Manual Test has 4 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------



1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING SOLICITUD: GET: US 13 API TESTING Obtener una lista de géneros URL: [http://localhost:4000/filters/genres] http://localhost:4000/filters/genres]	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 13 API TESTING Obtener una lista de géneros" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET US 13 API TESTING Obtener una lista de géneros					PASSED



3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{code:javascript}// TC3: VERIFICAR que sea array de objetos JSON pm.test("Verificar respuesta como array de objetos JSON", function () { pm.expect(pm.response.json()).to.be.an('array'); });{code}</pre>	Lograr ver correctamente el script en la pestaña TESTS					PASSED
4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar respuesta como array de objetos JSON", tenga el estado PASS				image.png 	PASSED

10.1.3.7. *Requirements linked with this test*
None.

10.1.4. [FILMFLOW-33](#): US 13 | TS - 01 | TC - 04 | Campos id y name

Execution Status	Assignee	Executed By	Started On	Finished On
------------------	----------	-------------	------------	-------------



PASSED	Mateo Joel Lopez	Mateo Joel Lopez		
---------------	------------------	------------------	--	--

10.1.4.1. Execution Defects

There are 0 Execution Defects on this Test Run.

10.1.4.2. Execution Evidence

There are 0 Execution Evidence on this Test Run.

10.1.4.3. Comment

None.

10.1.4.4. Test Description

Verificar que cada género en la lista contenga al menos los campos "id" y "name" cuando se envía una solicitud GET a "http://localhost:4000/filters/genres".

10.1.4.5. Test Issue Attachments

FILMFLOW-33 doesn't have attachments.

10.1.4.6. Test Details

This Manual Test has 4 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------

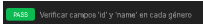


1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING SOLICITUD: GET: US 13 API TESTING Obtener una lista de géneros URL: [http://localhost:4000/filters/genres] http://localhost:4000/filters/genres]	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 13 API TESTING Obtener una lista de géneros" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET US 13 API TESTING Obtener una lista de géneros					PASSED



3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{code:javascript}// TC4:Verificar que cada género en la lista contenga al menos los campos "id" y "name" pm.test("Verificar campos 'id' y 'name' en cada género", function () { // Obtener la respuesta JSON var jsonResponse = pm.response.json (); // Verificar que la respuesta sea un arreglo pm.expect(jsonR esponse).to.be.an ('array'); // Iterar a través de cada género en el arreglo y verificar los campos 'id' y 'name'</pre>	Lograr ver correctamente el script en la pestaña TESTS					PASSED
---	---	---	--	--	--	--	--	--------



		<pre>jsonResponse.for Each(function (genre) { pm.expect(genre) .to.have.property(' id'); pm.expect(genre) .to.have.property(' name'); }); });{code}</pre>						
4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar campos 'id' y 'name' en cada género", tenga el estado PASS				image.png 	PASSED

10.1.4.7. Requirements linked with this test
None.

10.1.5. [FILMFLOW-34](#): US 13 | TS - 01 | TC - 05 | Solicitud GET incorrecta

Execution Status	Assignee	Executed By	Started On	Finished On
------------------	----------	-------------	------------	-------------



PASSED	Mateo Joel Lopez	Mateo Joel Lopez		
--------	------------------	------------------	--	--

10.1.5.1. Execution Defects

There are 0 Execution Defects on this Test Run.

10.1.5.2. Execution Evidence

There are 0 Execution Evidence on this Test Run.

10.1.5.3. Comment

None.

10.1.5.4. Test Description

Verificar que la respuesta no sea un código de estado HTTP 404 (Not Found) cuando se envía una solicitud GET a una URL incorrecta.

10.1.5.5. Test Issue Attachments

FILMFLOW-34 doesn't have attachments.

10.1.5.6. Test Details


This Manual Test has 4 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------



1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING SOLICITUD: GET: US 13 API TESTING Obtener una lista de géneros URL: [http://localhost:4000/filters/genres] http://localhost:4000/filters/genres]	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 13 API TESTING Obtener una lista de géneros" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET US 13 API TESTING Obtener una lista de géneros					PASSED



3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{code:javascript}// TC5: Verificar que la respuesta no sea un código de estado HTTP 404 (Not Found) cuando se envía una solicitud GET a una URL incorrecta pm.test("Verificar que no sea un código de estado HTTP 404", function () { pm.response.to.not.have.status(404); }},{code}</pre>	Lograr ver correctamente el script en la pestaña TESTS					PASSED
4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar que no sea un código de estado HTTP 404", tenga el estado PASS					PASSED

10.1.5.7. Requirements linked with this test
None.



10.1.6. [FILMFLOW-35](#): US 13 | TS - 01 | TC - 06 | Lista de generos NO está vacía.

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.1.6.1. Execution Defects

There are 0 Execution Defects on this Test Run.

10.1.6.2. Execution Evidence

There are 0 Execution Evidence on this Test Run.

10.1.6.3. Comment

None.

10.1.6.4. Test Description

Verificar que la lista de géneros **NO** esté vacía cuando se envía una solicitud GET a "http://localhost:4000/filters/genres".

10.1.6.5. Test Issue Attachments

FILMFLOW-35 doesn't have attachments.



10.1.6.6. Test Details


This Manual Test has 4 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING SOLICITUD: GET: US 13 API TESTING Obtener una lista de géneros URL: [http://localhost:4000/filters/genres] http://localhost:4000/filters/genres]	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 13 API TESTING Obtener una lista de géneros" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET US 13 API TESTING Obtener una lista de géneros					PASSED



3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{code:javascript}// TC6:TC8: Verificar que la lista de géneros no esté vacía pm.test("Verificar que la lista de géneros no esté vacía", function () { // Verificar que la respuesta sea un arreglo pm.expect(pm.res ponse.json()).to.b e.an('array'); // Verificar que el arreglo no esté vacío pm.expect(pm.res ponse.json()).to.h ave.length.above(0); }); {code}</pre>	Lograr ver correctamente el script en la pestaña TESTS						PASSED
---	---	--	--	--	--	--	--	--	--------



4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar que la lista de géneros no esté vacía", tenga el estado PASS					PASSED
---	--	--	--	--	--	--	---	--------

10.1.6.7. Requirements linked with this test
None.

10.1.7. **FILMFLOW-36**: US 13 | TS - 01 | TC - 07 | Generos duplicados.

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.1.7.1. Execution Defects
There are 0 Execution Defects on this Test Run.

10.1.7.2. Execution Evidence
There are 0 Execution Evidence on this Test Run.

10.1.7.3. Comment
None.



10.1.7.4. Test Description

Verificar que la respuesta no contenga géneros duplicados cuando se envía una solicitud GET a "http://localhost:4000/filters/genres".

10.1.7.5. Test Issue Attachments

FILMFLOW-36 doesn't have attachments.

10.1.7.6. Test Details

This Manual Test has 4 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------




1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING SOLICITUD: GET: US 13 API TESTING Obtener una lista de géneros URL: [http://localhost:4000/filters/genres] http://localhost:4000/filters/genres]	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 13 API TESTING Obtener una lista de géneros" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET US 13 API TESTING Obtener una lista de géneros					PASSED



3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{code:javascript}// TC7: C7: Verificar que la respuesta no contenga géneros duplicados pm.test("Verificar que no haya géneros duplicados", function () { // Obtener la respuesta JSON var jsonResponse = pm.response.json (); // Verificar que la respuesta sea un arreglo pm.expect(jsonR esponse).to.be.an ('array'); // Verificar que no haya géneros duplicados var genres = jsonResponse.ma p(JSON.stringify); var</pre>	Lograr ver correctamente el script en la pestaña TESTS						PASSED
---	---	---	--	--	--	--	--	--	--------



		<pre>uniqueGenres = [...new Set(genres)]; pm.expect(unique Genres.length).to. eql(genres.length); }); {code}</pre>						
4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar que no haya géneros duplicados", tenga el estado PASS					PASSED

10.1.7.7. Requirements linked with this test
None.

10.2. [FILMFLOW-37](#): US 15 | TS 01 | Obtener una lista de películas populares

10.2.1. [FILMFLOW-38](#): US 15 | TS 01 | TC 01 | Verificar estado 200 OK

Execution Status	Assignee	Executed By	Started On	Finished On
------------------	----------	-------------	------------	-------------



PASSED	Mateo Joel Lopez	Mateo Joel Lopez		
---------------	------------------	------------------	--	--

10.2.1.1. Execution Defects

There are 0 Execution Defects on this Test Run.

10.2.1.2. Execution Evidence

There are 0 Execution Evidence on this Test Run.

10.2.1.3. Comment

None.

10.2.1.4. Test Description

Verificar que la respuesta de la solicitud GET a "http://localhost:4000/list/popmovies" sea un código de estado HTTP 200 (OK).

10.2.1.5. Test Issue Attachments

FILMFLOW-38 doesn't have attachments.

10.2.1.6. Test Details

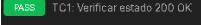
This Manual Test has 4 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------



1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	<p>COLECCION: DB PELICULAS TESTING</p> <p>SOLICITUD: GET: US 15 API TESTING Obtener una lista de películas populares</p> <p>URL: [http://localhost:4000/list/popmovies http://localhost:4000/list/popmovies]</p>	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 15 API TESTING Obtener una lista de películas populares" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET US 15 API TESTING Obtener una lista de películas populares					PASSED



3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{noformat}TC1: Verificar que la respuesta de la solicitud GET a "http://localhost:4 000/list/popmovie s" sea un código de estado HTTP 200 (OK). pm.test("TC1: Verificar estado 200 OK", function () { pm.response.to.h ave.status(200); }); {noformat}</pre>	Lograr ver correctamente el script en la pestaña TESTS					PASSED
4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: TC1: Verificar estado 200 OK, tenga el estado PASS				image.png 	PASSED

10.2.1.7. Requirements linked with this test
None.



10.2.2. [FILMFLOW-39](#): US 15 | TS 01 | TC 02 | Propiedad 'popularity' en todos los objetos

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.2.2.1. Execution Defects

There are 0 Execution Defects on this Test Run.

10.2.2.2. Execution Evidence

There are 0 Execution Evidence on this Test Run.

10.2.2.3. Comment

None.

10.2.2.4. Test Description

Verificar que las películas en la respuesta tengan la propiedad "popularity" cuando se envía una solicitud GET a "http://localhost:4000/list/popmovies".

10.2.2.5. Test Issue Attachments

FILMFLOW-39 doesn't have attachments.

10.2.2.6. Test Details

This Manual Test has 4 Steps.




Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING SOLICITUD: GET: US 15 API TESTING Obtener una lista de películas populares URL: [http://localhost:4000/list/popmovies http://localhost:4000/list/popmovies]	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 15 API TESTING Obtener una lista de películas populares" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET US 15 API TESTING Obtener una lista de películas populares					PASSED



3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{noformat}//TC2: Verificar que las películas en la respuesta tengan la propiedad "popularity" cuando se envía una solicitud GET a "http://localhost:4 000/list/popmovie s". pm.test("TC2: Propiedad 'popularity' en todos los objetos", function () { const peliculas = pm.response.json ().data; pm.expect(pelicul as).to.be.an('arra y').that.is.not.emp ty; peliculas.forEach((movie) => { pm.expect(movie)</pre>	Lograr ver correctamente el script en la pestaña TESTS					PASSED
---	---	---	--	--	--	--	--	--------



		<pre>.to.have.property('popularity'); }); }); {noformat}</pre>						
4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: TC2: Propiedad 'popularity' en todos los objetos, tenga el estado PASS					PASSED

10.2.2.7. *Requirements linked with this test*
None.

10.2.3. **FILMFLOW-40**: US 15 | TS 01 | TC 03 | Verificar que la respuesta sea un objeto JSON

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.2.3.1. *Execution Defects*
There are 0 Execution Defects on this Test Run.

10.2.3.2. *Execution Evidence*
There are 0 Execution Evidence on this Test Run.



10.2.3.3. Comment

None.

10.2.3.4. Test Description

Verificar que la respuesta de la solicitud GET a "http://localhost:4000/list/popmovies" sea un objeto JSON.

10.2.3.5. Test Issue Attachments

FILMFLOW-40 doesn't have attachments.

10.2.3.6. Test Details

This Manual Test has 4 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------



1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	<p>COLECCION: DB PELICULAS TESTING</p> <p>SOLICITUD: GET: US 15 API TESTING Obtener una lista de películas populares</p> <p>URL: [http://localhost:4000/list/popmovies http://localhost:4000/list/popmovies]</p>	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 15 API TESTING Obtener una lista de películas populares" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET US 15 API TESTING Obtener una lista de películas populares					PASSED



3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{noformat}//TC3: Verificar que la respuesta de la solicitud GET a "http://localhost:4 000/list/popmovie s" sea un objeto JSON que contiene un arreglo de películas populares. pm.test("TC3: Verificar que la respuesta sea un objeto JSON", function () { pm.response.to.b e.json; });{noformat}</pre>	Lograr ver correctamente el script en la pestaña TESTS					PASSED
4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: TC3: Verificar que la respuesta sea un objeto JSON, tenga el estado PASS					PASSED



10.2.3.7. *Requirements linked with this test*
None.

10.2.4. [FILMFLOW-41](#): US 15 | TS 01 | TC 04 | La propiedad POPULARITY es un numero

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.2.4.1. *Execution Defects*
There are 0 Execution Defects on this Test Run.

10.2.4.2. *Execution Evidence*
There are 0 Execution Evidence on this Test Run.

10.2.4.3. *Comment*
None.

10.2.4.4. *Test Description*
Verificar que la propiedad POPULARITY sea un número.

10.2.4.5. *Test Issue Attachments*
FILMFLOW-41 doesn't have attachments.



10.2.4.6. Test Details

This Manual Test has 4 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING SOLICITUD: GET: US 15 API TESTING Obtener una lista de películas populares URL: [http://localhost:4000/list/popmovies http://localhost:4000/list/popmovies]	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED




2	Haz clic en la solicitud "US 15 API TESTING Obtener una lista de películas populares" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET US 15 API TESTING Obtener una lista de películas populares						PASSED
---	---	--	--	--	--	--	--	--	--------



3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{noformat}//TC4: Verificar que la propiedad POPULARITY sea un número. pm.test("TC4: La propiedad POPULARITY es un numero", function () { const peliculas = pm.response.json ().data; const popularidad = peliculas[1].popul arity peliculas.forEach((movie) => { pm.expect(pelicul as[1].popularity).t o.be.a('number'); }); });{noformat}</pre>	Lograr ver correctamente el script en la pestaña TESTS						PASSED
---	---	---	--	--	--	--	--	--	--------



4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: TC4: La propiedad POPULARITY es un numero, tenga el estado PASS					PASSED
---	--	--	--	--	--	--	---	---------------

10.2.4.7. Requirements linked with this test

None.

10.2.5. **FILMFLOW-42**: US 15 | TS 01 | TC 05 | Las peliculas no tienen orden ascendente o descendente

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.2.5.1. Execution Defects

There are 0 Execution Defects on this Test Run.

10.2.5.2. Execution Evidence

There are 0 Execution Evidence on this Test Run.

10.2.5.3. Comment

None.



10.2.5.4. Test Description

Verificar que la lista de películas populares no tengan un orden ascendente ni descendente, es decir, que tenga un orden aleatorio cuando se envía la solicitud GET.

10.2.5.5. Test Issue Attachments

FILMFLOW-42 doesn't have attachments.

10.2.5.6. Test Details

This Manual Test has 4 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------



1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	<p>COLECCION: DB PELICULAS TESTING</p> <p>SOLICITUD: GET: US 15 API TESTING Obtener una lista de películas populares</p> <p>URL: [http://localhost:4000/list/popmovies http://localhost:4000/list/popmovies]</p>	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 15 API TESTING Obtener una lista de películas populares" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET US 15 API TESTING Obtener una lista de películas populares					PASSED



3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{noformat}//TC5: Verificar que la lista de peliculas populares no tengan un orden ascendente ni descendente, es decir, que tenga un orden aleatorio cuando se envia la solicitud GET. pm.test("TC5: Las peliculas no tienen orden ascendente o descendente", function () { const movies = pm.response.json ().data; const sortedMovies = [...movies].sort((a, b) => a.popularity - b.popularity); pm.expect(movie s).to.not.eql(sorte dMovies); });{noformat}</pre>	Lograr ver correctamente el script en la pestaña TESTS						PASSED
---	---	--	--	--	--	--	--	--	--------



4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: TC5: Las peliculas no tienen orden ascendente o descendente, tenga el estado PASS				image.png 	PASSED
---	--	--	--	--	--	--	-------------------------------	---------------

10.2.5.7. Requirements linked with this test
None.

10.3. **FILMFLOW-43**: US 17 | TS 01 | Validar el formato de las URL de las imágenes

10.3.1. **FILMFLOW-44**: US 17 | TS 01 | TC 01 | Verificar imagen válida cuando la URL de la imagen comienza con "https://".

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.3.1.1. Execution Defects
There are 0 Execution Defects on this Test Run.

10.3.1.2. Execution Evidence
There are 0 Execution Evidence on this Test Run.



10.3.1.3. Comment

None.

10.3.1.4. Test Description

None.

10.3.1.5. Test Issue Attachments

FILMFLOW-44 doesn't have attachments.

10.3.1.6. Test Details

This Manual Test has 4 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------




1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	<p>COLECCION: DB PELICULAS TESTING</p> <p>SOLICITUD: GET: US 17 API TESTING Validar el formato de las URL de las imágenes</p> <p>URL: [http://localhost:4000/getmovies]http://localhost:4000/getmovies]</p>	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 17 API TESTING Validar el formato de las URL de las imágenes" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET US 17 API TESTING Validar el formato de las URL de las imágenes					PASSED



3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{noformat}//TC#1: Verificar imagen válida cuando la URL de la imagen comienza con "https://". pm.test("Las URLs de las imagenes comienzan con 'https://'", function () { const movies = pm.response.json (); _.each(movies, function (movie) { pm.expect(movie. image).to.match(/ ^https:\/\//); }); }); {noformat}</pre>	Lograr ver correctamente el script en la pestaña TESTS						PASSED
---	---	---	--	--	--	--	--	--	--------



4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Las URLs de las imagenes comienzan con 'https://'", tenga el estado PASS					PASSED
---	--	--	--	--	--	--	---	--------

10.3.1.7. Requirements linked with this test
None.

10.3.2. FILMFLOW-45: US 17 | TS 01 | TC 02 | Verificar estado 200 cuando se hace la solicitud GET.

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.3.2.1. Execution Defects
There are 0 Execution Defects on this Test Run.

10.3.2.2. Execution Evidence
There are 0 Execution Evidence on this Test Run.

10.3.2.3. Comment
None.



10.3.2.4. Test Description
None.

10.3.2.5. Test Issue Attachments
FILMFLOW-45 doesn't have attachments.

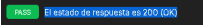
10.3.2.6. Test Details
This Manual Test has 4 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------



1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	<p>COLECCION: DB PELICULAS TESTING</p> <p>SOLICITUD: GET: US 17 API TESTING Validar el formato de las URL de las imágenes</p> <p>URL: [http://localhost:4000/getmovies]http://localhost:4000/getmovies]</p>	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 17 API TESTING Validar el formato de las URL de las imágenes" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET US 17 API TESTING Validar el formato de las URL de las imágenes					PASSED



3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{noformat} //TC#2: Verificar estado 200 cuando se hace la solicitud GET. pm.test("El estado de respuesta es 200 (OK)", function () { pm.response.to.h ave.status(200); });{noformat}</pre>	Lograr ver correctamente el script en la pestaña TESTS					PASSED
4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "TEI estado de respuesta es 200 (OK)", tenga el estado PASS				image.png 	PASSED

10.3.2.7. *Requirements linked with this test*
None.

10.3.3. FILMFLOW-46: US 17 | TS 01 | TC 03 | Verificar que todas las peliculas tengan la propiedad image

Execution Status	Assignee	Executed By	Started On	Finished On
------------------	----------	-------------	------------	-------------



PASSED	Mateo Joel Lopez	Mateo Joel Lopez		
--------	------------------	------------------	--	--

10.3.3.1. Execution Defects
There are 0 Execution Defects on this Test Run.

10.3.3.2. Execution Evidence
There are 0 Execution Evidence on this Test Run.

10.3.3.3. Comment
None.

10.3.3.4. Test Description
None.

10.3.3.5. Test Issue Attachments
FILMFLOW-46 doesn't have attachments.

10.3.3.6. Test Details
This Manual Test has 4 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------




1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	<p>COLECCION: DB PELICULAS TESTING</p> <p>SOLICITUD: GET: US 17 API TESTING Validar el formato de las URL de las imágenes</p> <p>URL: [http://localhost:4 000/getmovies]htt p://localhost:4000 /getmovies]</p>	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 17 API TESTING Validar el formato de las URL de las imágenes" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET US 17 API TESTING Validar el formato de las URL de las imágenes					PASSED



3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{noformat} //TC#3: Verificar que todas las películas tengan la propiedad image. pm.test("Todas las películas tienen la propiedad 'image", function () { const movies = pm.response.json (); _.each(movies, function (movie) { pm.expect(movie) .to.have.property(' image'); }); }); }{noformat}</pre>	Lograr ver correctamente el script en la pestaña TESTS						PASSED
---	---	---	--	--	--	--	--	--	--------



4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Todas las películas tienen la propiedad 'image'", tenga el estado PASS					PASSED
---	--	--	--	--	--	--	---	---------------

10.3.3.7. Requirements linked with this test
None.

10.4. [FILMFLOW-47](#): US 19 | TS 01 | Obtener una lista de películas

10.4.1. [FILMFLOW-48](#): US 19 | TS 01 | TC 01 | Verificar que se reciban 20 películas en la respuesta cuando se realiza la solicitud GET.

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.4.1.1. Execution Defects
There are 0 Execution Defects on this Test Run.

10.4.1.2. Execution Evidence
There are 0 Execution Evidence on this Test Run.



10.4.1.3. Comment

None.

10.4.1.4. Test Description

None.

10.4.1.5. Test Issue Attachments

FILMFLOW-48 doesn't have attachments.

10.4.1.6. Test Details

This Manual Test has 4 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------



1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	<p>COLECCION: DB PELICULAS TESTING</p> <p>SOLICITUD: GET: US 19 API TESTING Obtener una lista de películas</p> <p>URL: [http://localhost:4000/getmovies]http://localhost:4000/getmovies]</p>	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 19 API TESTING Obtener una lista de películas" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET US 19 API TESTING Obtener una lista de películas					PASSED



3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{noformat}//TC1: Verificar que se reciban 20 películas en la respuesta cuando se realiza la solicitud GET. pm.test("La lista contiene 20 películas", function () { pm.response.to.h ave.status(200); pm.expect(pm.res ponse.json()).to.b e.an('array').and.t o.have.lengthOf(2 0); });{noformat}</pre>	Lograr ver correctamente el script en la pestaña TESTS					PASSED
4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "La lista contiene 20 películas", tenga el estado PASS				image.png 	PASSED

10.4.1.7. Requirements linked with this test
None.



10.4.2. [FILMFLOW-49](#): US 19 | TS 01 | TC 02 | Verificar que el código de estado de la respuesta sea 200 (Éxito) cuando se realiza la solicitud GET.

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.4.2.1. Execution Defects

There are 0 Execution Defects on this Test Run.

10.4.2.2. Execution Evidence

There are 0 Execution Evidence on this Test Run.

10.4.2.3. Comment

None.

10.4.2.4. Test Description

None.

10.4.2.5. Test Issue Attachments

FILMFLOW-49 doesn't have attachments.

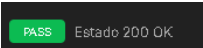
10.4.2.6. Test Details

This Manual Test has 4 Steps.



Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING SOLICITUD: GET: US 19 API TESTING Obtener una lista de películas URL: [http://localhost:4000/getmovies]http://localhost:4000/getmovies]	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 19 API TESTING Obtener una lista de películas" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET US 19 API TESTING Obtener una lista de películas					PASSED



3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{noformat}//TC2: Verificar que el código de estado de la respuesta sea 200 (Éxito) cuando se realiza la solicitud GET. pm.test("Respons e status code should be 200", function () { pm.response.to.h ave.status(200); }); {noformat}</pre>	Lograr ver correctamente el script en la pestaña TESTS					PASSED
4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: {{Estado 200 OK}}, tenga el estado PASS				image.png 	PASSED

10.4.2.7. Requirements linked with this test
None.



10.4.3. [FILMFLOW-50](#): US 19 | TS 01 | TC 03 | Verificar que la respuesta contiene la información requerida para cada película, como título, descripción, fecha de lanzamiento e imagen.

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.4.3.1. Execution Defects

There are 0 Execution Defects on this Test Run.

10.4.3.2. Execution Evidence

There are 0 Execution Evidence on this Test Run.

10.4.3.3. Comment

None.

10.4.3.4. Test Description

None.

10.4.3.5. Test Issue Attachments

FILMFLOW-50 doesn't have attachments.



10.4.3.6. Test Details

This Manual Test has 4 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING SOLICITUD: GET: US 19 API TESTING Obtener una lista de películas URL: [http://localhost:4000/getmovies]http://localhost:4000/getmovies]	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 19 API TESTING Obtener una lista de películas" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET US 19 API TESTING Obtener una lista de películas					PASSED




3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{noformat}//TC3: Verificar que la respuesta contiene la información requerida para cada película, como título, descripción, fecha de lanzamiento e imagen. pm.test("Todas las películas tienen propiedad título, descripción, fecha e imagen", function () { pm.response.to.h ave.status(200); pm.response.to.b e.json; var jsonData = pm.response.json (); pm.expect(jsonD ata).to.be.an('arra y');</pre>	Lograr ver correctamente el script en la pestaña TESTS						PASSED
---	---	---	--	--	--	--	--	--	--------



		<pre>jsonData.forEach(function (movie) { pm.expect(movie) .to.have.property("title"); pm.expect(movie) .to.have.property("overview"); pm.expect(movie) .to.have.property("release_date"); pm.expect(movie) .to.have.property("image"); }); }); {noformat}</pre>						
--	--	--	--	--	--	--	--	--



4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Todas las peliculas tienen propiedad titulo, descripción, fecha e imagen", tenga el estado PASS					PASSED
---	--	--	---	--	--	--	---	--------

10.4.3.7. Requirements linked with this test
None.

10.4.4. [FILMFLOW-51](#): US 19 | TS 01 | TC4 | Verificar que las fechas se devuelven en un formato específico, como "AAAA-MM-DD,"

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.4.4.1. Execution Defects
There are 0 Execution Defects on this Test Run.

10.4.4.2. Execution Evidence
There are 0 Execution Evidence on this Test Run.



10.4.4.3. Comment

None.

10.4.4.4. Test Description

None.

10.4.4.5. Test Issue Attachments

FILMFLOW-51 doesn't have attachments.

10.4.4.6. Test Details

This Manual Test has 0 Steps.

10.4.4.7. Requirements linked with this test

None.

10.5. [FILMFLOW-52](#): US 18 | TS 01 | Validar valores de "vote_average" en un rango y "vote_count" mínimo

10.5.1. [FILMFLOW-53](#): US 18 | TS 01 | TC 01 | Endpoint accesible

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		



10.5.1.1. Execution Defects

There are 0 Execution Defects on this Test Run.

10.5.1.2. Execution Evidence

There are 0 Execution Evidence on this Test Run.

10.5.1.3. Comment

None.

10.5.1.4. Test Description

Verificar que la respuesta devuelva un código de estado 200.

10.5.1.5. Test Issue Attachments

FILMFLOW-53 doesn't have attachments.

10.5.1.6. Test Details


This Manual Test has 4 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------



1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	<p>COLECCION: DB PELICULAS TESTING</p> <p>SOLICITUD: GET: US 18 API TESTING Validar valores de "vote_average" en un rango y "vote_count" mínimo</p> <p>URL: [http://localhost:4000/getmovies http://localhost:4000/getmovies]</p>	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 18 API TESTING Validar valores de "vote_average" en un rango y "vote_count" mínimo" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET US 18 API TESTING Validar valores de "vote_average" en un rango y "vote_count" mínimo					PASSED



3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{noformat}pm.test("Endpoint es accesible", function () {pm.response.to.have.status(200);});{noformat}</pre>	Lograr ver correctamente el script en la pestaña TESTS					PASSED
4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Endpoint es accesible", tenga el estado PASS					PASSED

10.5.1.7. Requirements linked with this test
None.

10.5.2. [FILMFLOW-57](#): US 18 | TS 01 | TC 05 | Verificar array vacio

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.5.2.1. Execution Defects
There are 0 Execution Defects on this Test Run.



10.5.2.2. Execution Evidence

There are 0 Execution Evidence on this Test Run.

10.5.2.3. Comment

None.

10.5.2.4. Test Description

Verificar que la respuesta de no esté vacía.

10.5.2.5. Test Issue Attachments

FILMFLOW-57 doesn't have attachments.

10.5.2.6. Test Details


This Manual Test has 4 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------



1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	<p>COLECCION: DB PELICULAS TESTING</p> <p>SOLICITUD: GET: US 18 API TESTING Validar valores de "vote_average" en un rango y "vote_count" mínimo</p> <p>URL: [http://localhost:4000/getmovies http://localhost:4000/getmovies]</p>	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 18 API TESTING Validar valores de "vote_average" en un rango y "vote_count" mínimo" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET US 18 API TESTING Validar valores de "vote_average" en un rango y "vote_count" mínimo					PASSED



3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{noformat}pm.test("El array de peliculas no esta vacio", function () { pm.response.to.not.be.empty; }); {noformat}</pre>	Lograr ver correctamente el script en la pestaña TESTS					PASSED
4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "El array de peliculas no esta vacio", tenga el estado PASS					PASSED

10.5.2.7. *Requirements linked with this test*
None.

10.5.3. [FILMFLOW-56](#): US 18 | TS 01 | TC 04 | Campo vote_count valido

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.5.3.1. *Execution Defects*
There are 0 Execution Defects on this Test Run.



10.5.3.2. Execution Evidence

There are 0 Execution Evidence on this Test Run.

10.5.3.3. Comment

None.

10.5.3.4. Test Description

Verificar que, para cada película en la respuesta, el campo "vote_count" sea un valor numérico igual o mayor que 0.

10.5.3.5. Test Issue Attachments

FILMFLOW-56 doesn't have attachments.

10.5.3.6. Test Details

This Manual Test has 4 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------



1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	<p>COLECCION: DB PELICULAS TESTING</p> <p>SOLICITUD: GET: US 18 API TESTING Validar valores de "vote_average" en un rango y "vote_count" mínimo</p> <p>URL: [http://localhost:4000/getmovies http://localhost:4000/getmovies]</p>	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 18 API TESTING Validar valores de "vote_average" en un rango y "vote_count" mínimo" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET US 18 API TESTING Validar valores de "vote_average" en un rango y "vote_count" mínimo					PASSED



3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{noformat}pm.test ("Todas las peliculas tienen vote_count con valor numérico igual o mayor que 0.", function () { var movies = pm.response.json (); _.each(movies, function (movie) { pm.expect(movie. vote_count).to.be. a('number'); pm.expect(movie. vote_count).to.be. at.least(0); }); });{noformat}</pre>	Lograr ver correctamente el script en la pestaña TESTS						PASSED
---	---	---	--	--	--	--	--	--	--------



4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Todas las peliculas tienen vote_count con valor numérico igual o mayor que 0.", tenga el estado PASS					PASSED
---	--	--	--	--	--	--	---	--------

10.5.3.7. Requirements linked with this test
None.

10.5.4. **FILMFLOW-55:** US 18 | TS 01 | TC 03 | vote_Average valido

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.5.4.1. Execution Defects
There are 0 Execution Defects on this Test Run.

10.5.4.2. Execution Evidence
There are 0 Execution Evidence on this Test Run.



10.5.4.3. Comment

None.

10.5.4.4. Test Description

Verificar que, para cada película en la respuesta, el campo "vote_average" sea un valor numérico dentro del rango válido (0-10).

10.5.4.5. Test Issue Attachments

FILMFLOW-55 doesn't have attachments.

10.5.4.6. Test Details

This Manual Test has 4 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------




1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	<p>COLECCION: DB PELICULAS TESTING</p> <p>SOLICITUD: GET: US 18 API TESTING Validar valores de "vote_average" en un rango y "vote_count" mínimo</p> <p>URL: [http://localhost:4000/getmovies http://localhost:4000/getmovies]</p>	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 18 API TESTING Validar valores de "vote_average" en un rango y "vote_count" mínimo" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET US 18 API TESTING Validar valores de "vote_average" en un rango y "vote_count" mínimo					PASSED



3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{noformat}pm.test ("Todas las películas tienen vote_average con un valor numérico dentro del rango válido de 0 a 10", function () { var movies = pm.response.json (); _.each(movies, function (movie) { pm.expect(movie. vote_average).to. be.a('number'); pm.expect(movie. vote_average).to. be.within(0, 10); }); });{noformat}</pre>	Lograr ver correctamente el script en la pestaña TESTS						PASSED
---	---	---	--	--	--	--	--	--	--------



4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: " Todas las peliculas tienen vote_average con un valor numérico dentro del rango válido de 0 a 10", tenga el estado PASS				image.png 	PASSED
---	--	--	---	--	--	--	--	---------------

10.5.4.7. Requirements linked with this test
None.

10.5.5. [FILMFLOW-54](#): US 18 | TS 01 | TC 02 | Array de peliculas

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.5.5.1. Execution Defects
There are 0 Execution Defects on this Test Run.

10.5.5.2. Execution Evidence
There are 0 Execution Evidence on this Test Run.



10.5.5.3. Comment

None.

10.5.5.4. Test Description

Verificar que la respuesta contenga un objeto JSON con un arreglo de películas.

10.5.5.5. Test Issue Attachments

FILMFLOW-54 doesn't have attachments.

10.5.5.6. Test Details


This Manual Test has 4 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------



1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	<p>COLECCION: DB PELICULAS TESTING</p> <p>SOLICITUD: GET: US 18 API TESTING Validar valores de "vote_average" en un rango y "vote_count" mínimo</p> <p>URL: [http://localhost:4000/getmovies http://localhost:4000/getmovies]</p>	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 18 API TESTING Validar valores de "vote_average" en un rango y "vote_count" mínimo" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET US 18 API TESTING Validar valores de "vote_average" en un rango y "vote_count" mínimo					PASSED



3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{noformat}pm.test("La respuesta obtinee un array de peliculas", function () { pm.expect(pm.response.json()).to.be.an('array'); });{noformat}</pre>	Lograr ver correctamente el script en la pestaña TESTS					PASSED
4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "La respuesta obtinee un array de peliculas", tenga el estado PASS				image.png 	PASSED

10.5.5.7. Requirements linked with this test
None.

10.6. [FILMFLOW-58](#): US 16 | TS 01 | Obtener una lista de próximos estrenos

10.6.1. [FILMFLOW-59](#): US 16 | TS 01 | TC 01 | Formato de fechas de lanzamiento

Execution Status	Assignee	Executed By	Started On	Finished On
------------------	----------	-------------	------------	-------------



PASSED	Mateo Joel Lopez	Mateo Joel Lopez		
---------------	------------------	------------------	--	--

10.6.1.1. Execution Defects

There are 0 Execution Defects on this Test Run.

10.6.1.2. Execution Evidence

There are 0 Execution Evidence on this Test Run.

10.6.1.3. Comment

None.

10.6.1.4. Test Description

Verificar que las fechas de lanzamiento futuras se muestren en el formato “AAAA-MM-DD” cuando no se ha producido ningún error.

10.6.1.5. Test Issue Attachments

FILMFLOW-59 doesn't have attachments.

10.6.1.6. Test Details

This Manual Test has 4 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------

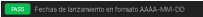


1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING SOLICITUD: GET: US 16 API TESTING Obtener una lista de próximos estrenos URL: [http://localhost:4000/getmovies]http://localhost:4000/getmovies]	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 16 API TESTING Obtener una lista de próximos estrenos" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET US 16 API TESTING Obtener una lista de próximos estrenos					PASSED



3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{noformat}pm.test ("Fechas de lanzamiento en formato AAAA- MM-DD", function () { var responseData = pm.response.json (); var isFormatValid = true; for (var i = 0; i < responseData.len gth; i++) { var releaseDate = responseData[i].r elease_date; if (!/^\\d{4}- \\d{2}- \\d{2}\$/.test(releas eDate)) { isFormatValid = false; break; } } }</pre>	Lograr ver correctamente el script en la pestaña TESTS					PASSED
---	---	---	--	--	--	--	--	--------



		<pre>pm.expect(isForm atValid).to.be.true ; }); {noformat}</pre>						
4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Fechas de lanzamiento en formato AAAA-MM-DD", tenga el estado PASS				image.png 	PASSED

10.6.1.7. Requirements linked with this test
None.

10.6.2. [FILMFLOW-61](#): US 16 | TS 01 | TC 03 | Propiedad release_date

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.6.2.1. Execution Defects

There are 0 Execution Defects on this Test Run.



10.6.2.2. Execution Evidence

There are 0 Execution Evidence on this Test Run.

10.6.2.3. Comment

None.

10.6.2.4. Test Description

Verificar que todas las peliculas obtenidas en el llamado, tengan la propiedad release_date (fecha de lanzamiento)

10.6.2.5. Test Issue Attachments

FILMFLOW-61 doesn't have attachments.

10.6.2.6. Test Details

This Manual Test has 4 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------




1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	<p>COLECCION: DB PELICULAS TESTING</p> <p>SOLICITUD: GET: US 16 API TESTING Obtener una lista de próximos estrenos</p> <p>URL: [http://localhost:4000/getmovies]http://localhost:4000/getmovies]</p>	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 16 API TESTING Obtener una lista de próximos estrenos" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET US 16 API TESTING Obtener una lista de próximos estrenos					PASSED



3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{noformat}pm.test ("Todas las películas tienen la propiedad release_date", function () { var responseData = pm.response.json (); var allMoviesHaveRel easeDate = true; for (var i = 0; i < responseData.len gth; i++) { var movie = responseData[i]; if (!movie.hasOwnP roperty("release_ date")) { allMoviesHaveRel easeDate = false; break; } }</pre> <pre>pm.expect(allMov</pre>	Lograr ver correctamente el script en la pestaña TESTS					PASSED
---	---	--	--	--	--	--	--	--------



		iesHaveReleaseDate).to.be.true;}); {noformat}						
4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Todas las películas tienen la propiedad release_date", tenga el estado PASS			FILMFLOW-62	image.png  image.png	PASSED

10.6.2.7. Requirements linked with this test
None.

10.6.3. [FILMFLOW-60](#): US 16 | TS 01 | TC 02 | Fechas de lanzamiento pasadas

Execution Status	Assignee	Executed By	Started On	Finished On
FAILED	Mateo Joel Lopez	Mateo Joel Lopez		

10.6.3.1. Execution Defects
There are 0 Execution Defects on this Test Run.

10.6.3.2. Execution Evidence
There are 0 Execution Evidence on this Test Run.



10.6.3.3. Comment
None.

10.6.3.4. Test Description
Verificar que no se muestren fechas de lanzamiento pasadas en cada pelicula.

10.6.3.5. Test Issue Attachments
FILMFLOW-60 doesn't have attachments.

10.6.3.6. Test Details
This Manual Test has 4 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Sta
------	--------	------	-----------------	-------------	---------	---------	----------	-----



1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	<p>COLECCION: DB PELICULAS TESTING</p> <p>SOLICITUD: GET:</p> <p>US 16 API TESTING Obtener una lista de próximos estrenos</p> <p>URL:</p> <p>[http://localhost:4000/getmovies http://localhost:4000/getmovies]</p>	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow						PAS
2	Haz clic en la solicitud "US 16 API TESTING Obtener una lista de próximos estrenos" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET						PAS



3	En la pestaña "Tests" de la solicitud, puedes escribir el siguiente script de prueba en JavaScript:	<pre>{noformat}//----- var currentDate = new Date().toISOString().split('T')[0]; // Imprime la fecha actual en la consola console.log("Fecha actual: " + currentDate); // Analiza la respuesta JSON var responseData = pm.response.json(); // Itera a través de las películas en la respuesta y muestra el nombre junto con la fecha de lanzamiento for (var i = 0; i < responseData.length; i++) { var movie = responseData[i]; var releaseDate = movie.release_date; var movieTitle = movie.title; console.log("Fecha de lanzamiento de '" + movieTitle + "': " + releaseDate); } pm.test("Las fechas de lanzamiento son posteriores a la actual", function () { var responseData = pm.response.json(); var currentDate = new Date(); var noPastDates = true; var moviesWithPastDates = []; for (var i = 0; i < responseData.length; i++) { var releaseDate = new Date(responseData[i].release_date); if (releaseDate < currentDate) { noPastDates = false; moviesWithPastDates.push(responseData[i].title); } } });</pre>	Lograr ver correctamente el script en la pestaña TESTS						PAS
---	---	--	--	--	--	--	--	--	-----



		<pre> } if (!noPastDates) { pm.expect(noPastDates, "Las siguientes películas tienen fechas de lanzamiento pasadas: " + moviesWithPastDates.join(", ")).to.be.true; } }); {noformat}</pre>						
4	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Las fechas de lanzamiento son posteriores a la actual", tenga el estado PASS					FAIL

10.6.3.7. Requirements linked with this test
None.

10.7. [FILMFLOW-68](#): US 14 | TS 01 | Verificacion de pertenencia

10.7.1. [FILMFLOW-70](#): US 14 | TS 01 | TC 01 | Genero Accion

Execution Status	Assignee	Executed By	Started On	Finished On
------------------	----------	-------------	------------	-------------



PASSED	Mateo Joel Lopez	Mateo Joel Lopez		
--------	------------------	------------------	--	--

10.7.1.1. Execution Defects
There are 0 Execution Defects on this Test Run.

10.7.1.2. Execution Evidence
There are 0 Execution Evidence on this Test Run.

10.7.1.3. Comment
None.

10.7.1.4. Test Description
Verificar que todas las películas en la respuesta pertenezcan al género de Acción (ID: 28).

10.7.1.5. Test Issue Attachments
FILMFLOW-70 doesn't have attachments.

10.7.1.6. Test Details
This Manual Test has 3 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------



1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING FOLDER: GET: US 14 API TESTING Obtener una lista de películas por género	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "ID: 28 - Acción" del folder: "US 14 API TESTING Obtener una lista de películas por género" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET					PASSED



3	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar que todas las películas sean del género Acción (ID: 28)", tenga el estado PASS					PASSED
---	--	--	--	--	--	--	--	--------

10.7.1.7. Requirements linked with this test
None.

10.7.2. [FILMFLOW-71](#): US 14 | TS 01 | TC 02 | Genero Aventura

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.7.2.1. Execution Defects
There are 0 Execution Defects on this Test Run.

10.7.2.2. Execution Evidence
There are 0 Execution Evidence on this Test Run.

10.7.2.3. Comment
None.



10.7.2.4. Test Description

Verificar que todas las películas en la respuesta pertenezcan al género de Aventura (ID: 12).

10.7.2.5. Test Issue Attachments

FILMFLOW-71 doesn't have attachments.

10.7.2.6. Test Details

This Manual Test has 3 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING FOLDER: GET: US 14 API TESTING Obtener una lista de películas por género	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					<div>PASSED</div>



2	Haz clic en la solicitud "ID: 12 - Aventura" del folder: "US 14 API TESTING Obtener una lista de películas por género" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET					PASSED
3	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar que todas las películas sean del género Aventura (ID: 12)", tenga el estado PASS					PASSED

10.7.2.7. Requirements linked with this test
None.

10.7.3. [FILMFLOW-72](#): US 14 | TS 01 | TC 03 | Genero Animacion

Execution Status	Assignee	Executed By	Started On	Finished On
------------------	----------	-------------	------------	-------------



PASSED	Mateo Joel Lopez	Mateo Joel Lopez		
--------	------------------	------------------	--	--

10.7.3.1. Execution Defects
There are 0 Execution Defects on this Test Run.

10.7.3.2. Execution Evidence
There are 0 Execution Evidence on this Test Run.

10.7.3.3. Comment
None.

10.7.3.4. Test Description
Verificar que todas las películas en la respuesta pertenezcan al género de Animación (ID: 16).

10.7.3.5. Test Issue Attachments
FILMFLOW-72 doesn't have attachments.

10.7.3.6. Test Details
This Manual Test has 3 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------



1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING FOLDER: GET: US 14 API TESTING Obtener una lista de películas por género	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "ID: 16 - Animación" del folder: "US 14 API TESTING Obtener una lista de películas por género" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET					PASSED



3	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar que todas las películas sean del género Animacion (ID: 16)", tenga el estado PASS					PASSED
---	--	--	---	--	--	--	--	--------

10.7.3.7. *Requirements linked with this test*
None.

10.7.4. [FILMFLOW-73](#): US 14 | TS 01 | TC 04 | Genero comedia

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.7.4.1. *Execution Defects*
There are 0 Execution Defects on this Test Run.

10.7.4.2. *Execution Evidence*
There are 0 Execution Evidence on this Test Run.



10.7.4.3. *Comment*
None.

10.7.4.4. *Test Description*
Verificar que todas las películas en la respuesta pertenezcan al género de Comedia (ID: 35).

10.7.4.5. *Test Issue Attachments*
FILMFLOW-73 doesn't have attachments.

10.7.4.6. *Test Details*
This Manual Test has 3 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING FOLDER: GET: US 14 API TESTING Obtener una lista de películas por género	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED



2	Haz clic en la solicitud "ID: 35 - Comedia" del folder: "US 14 API TESTING Obtener una lista de películas por género" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET					PASSED
3	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar que todas las películas sean del género Comedio (ID: 35)", tenga el estado PASS					PASSED

10.7.4.7. Requirements linked with this test
None.

10.7.5. [FILMFLOW-74](#): US 14 | TS 01 | TC 05 | Genero crimen

Execution Status	Assignee	Executed By	Started On	Finished On
------------------	----------	-------------	------------	-------------



PASSED	Mateo Joel Lopez	Mateo Joel Lopez		
--------	------------------	------------------	--	--

10.7.5.1. Execution Defects
There are 0 Execution Defects on this Test Run.

10.7.5.2. Execution Evidence
There are 0 Execution Evidence on this Test Run.

10.7.5.3. Comment
None.

10.7.5.4. Test Description
Verificar que todas las películas en la respuesta pertenezcan al género de Crimen (ID: 80).

10.7.5.5. Test Issue Attachments
FILMFLOW-74 doesn't have attachments.

10.7.5.6. Test Details
This Manual Test has 3 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------



1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING FOLDER: GET: US 14 API TESTING Obtener una lista de películas por género	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "ID: 80 - Crimen" del folder: "US 14 API TESTING Obtener una lista de películas por género" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET					PASSED



3	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar que todas las películas sean del género Crimen (ID: 80)", tenga el estado PASS					PASSED
---	--	--	--	--	--	--	--	--------

10.7.5.7. *Requirements linked with this test*
None.

10.7.6. [FILMFLOW-75](#): US 14 | TS 01 | TC 06 | Genero documental

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.7.6.1. *Execution Defects*
There are 0 Execution Defects on this Test Run.

10.7.6.2. *Execution Evidence*
There are 0 Execution Evidence on this Test Run.

10.7.6.3. *Comment*
None.



10.7.6.4. Test Description

Verificar que todas las películas en la respuesta pertenezcan al género de Documental (ID: 99).

10.7.6.5. Test Issue Attachments

FILMFLOW-75 doesn't have attachments.

10.7.6.6. Test Details

This Manual Test has 3 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING FOLDER: GET: US 14 API TESTING Obtener una lista de películas por género	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED



2	Haz clic en la solicitud "ID: 99 - Documental" del folder: "US 14 API TESTING Obtener una lista de películas por género" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET					PASSED
3	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar que todas las películas sean del género Documental (ID: 99)", tenga el estado PASS					PASSED

10.7.6.7. Requirements linked with this test

None.



10.7.7. [FILMFLOW-76](#): US 14 | TS 01 | TC 07 | Genero drama

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.7.7.1. Execution Defects

There are 0 Execution Defects on this Test Run.

10.7.7.2. Execution Evidence

There are 0 Execution Evidence on this Test Run.

10.7.7.3. Comment

None.

10.7.7.4. Test Description

Verificar que todas las películas en la respuesta pertenezcan al género de Drama (ID: 18).

10.7.7.5. Test Issue Attachments

FILMFLOW-76 doesn't have attachments.

10.7.7.6. Test Details

This Manual Test has 3 Steps.



Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING FOLDER: GET: US 14 API TESTING Obtener una lista de películas por género	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "ID: 18 - Drama" del folder: "US 14 API TESTING Obtener una lista de películas por género" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET					PASSED



3	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar que todas las películas sean del género Drama (ID: 18)", tenga el estado PASS					PASSED
---	--	--	---	--	--	--	--	--------

10.7.7.7. *Requirements linked with this test*
None.

10.7.8. [FILMFLOW-77](#): US 14 | TS 01 | TC 08 | Genero familia

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.7.8.1. *Execution Defects*
There are 0 Execution Defects on this Test Run.

10.7.8.2. *Execution Evidence*
There are 0 Execution Evidence on this Test Run.

10.7.8.3. *Comment*
None.



10.7.8.4. Test Description

Verificar que todas las películas en la respuesta pertenezcan al género de Familia (ID: 10751).

10.7.8.5. Test Issue Attachments

FILMFLOW-77 doesn't have attachments.

10.7.8.6. Test Details

This Manual Test has 3 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING FOLDER: GET: US 14 API TESTING Obtener una lista de películas por género	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED



2	Haz clic en la solicitud "ID: 10751 - Familia" del folder: "US 14 API TESTING Obtener una lista de películas por género" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET					PASSED
3	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar que todas las películas sean del género Familia (ID: 10751)", tenga el estado PASS					PASSED

10.7.8.7. Requirements linked with this test

None.



10.7.9. [FILMFLOW-78](#): US 14 | TS 01 | TC 09 | Genero fantasia

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.7.9.1. Execution Defects

There are 0 Execution Defects on this Test Run.

10.7.9.2. Execution Evidence

There are 0 Execution Evidence on this Test Run.

10.7.9.3. Comment

None.

10.7.9.4. Test Description

Verificar que todas las películas en la respuesta pertenezcan al género de Fantasía (ID: 14).

10.7.9.5. Test Issue Attachments

FILMFLOW-78 doesn't have attachments.

10.7.9.6. Test Details

This Manual Test has 3 Steps.



Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING FOLDER: GET: US 14 API TESTING Obtener una lista de películas por género	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "ID: 14 - Fantasía" del folder: "US 14 API TESTING Obtener una lista de películas por género" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET					PASSED



3	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar que todas las películas sean del género Fantasia (ID: 14)", tenga el estado PASS					PASSED
---	--	--	--	--	--	--	--	--------

10.7.9.7. *Requirements linked with this test*
None.

10.7.10. [FILMFLOW-79](#): US 14 | TS 01 | TC 10 | Genero historia

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.7.10.1. *Execution Defects*
There are 0 Execution Defects on this Test Run.

10.7.10.2. *Execution Evidence*
There are 0 Execution Evidence on this Test Run.

10.7.10.3. *Comment*
None.



10.7.10.4. Test Description

Verificar que todas las películas en la respuesta pertenezcan al género de Historia (ID: 36).

10.7.10.5. Test Issue Attachments

FILMFLOW-79 doesn't have attachments.

10.7.10.6. Test Details

This Manual Test has 3 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING FOLDER: GET: US 14 API TESTING Obtener una lista de películas por género	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED



2	Haz clic en la solicitud "ID: 36 - Historia" del folder: "US 14 API TESTING Obtener una lista de películas por género" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET					PASSED
3	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar que todas las películas sean del género Historia (ID: 36)", tenga el estado PASS					PASSED

10.7.10.7. *Requirements linked with this test*
None.

10.7.11. [FILMFLOW-80](#): US 14 | TS 01 | TC 11 | Genero terror

Execution Status	Assignee	Executed By	Started On	Finished On
------------------	----------	-------------	------------	-------------



PASSED	Mateo Joel Lopez	Mateo Joel Lopez		
--------	------------------	------------------	--	--

10.7.11.1. Execution Defects
There are 0 Execution Defects on this Test Run.

10.7.11.2. Execution Evidence
There are 0 Execution Evidence on this Test Run.

10.7.11.3. Comment
None.

10.7.11.4. Test Description
Verificar que todas las películas en la respuesta pertenezcan al género de Terror (ID: 27).

10.7.11.5. Test Issue Attachments
FILMFLOW-80 doesn't have attachments.

10.7.11.6. Test Details
This Manual Test has 3 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------



1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING FOLDER: GET: US 14 API TESTING Obtener una lista de películas por género	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "ID: 27 - Terror" del folder: "US 14 API TESTING Obtener una lista de películas por género" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET					PASSED



3	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar que todas las películas sean del género Terror (ID: 27)", tenga el estado PASS					PASSED
---	--	--	--	--	--	--	--	--------

10.7.11.7. *Requirements linked with this test*
None.

10.7.12. [FILMFLOW-81](#): US 14 | TS 01 | TC 12 | Genero musica

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.7.12.1. *Execution Defects*
There are 0 Execution Defects on this Test Run.

10.7.12.2. *Execution Evidence*
There are 0 Execution Evidence on this Test Run.

10.7.12.3. *Comment*
None.



10.7.12.4. Test Description

Verificar que todas las películas en la respuesta pertenezcan al género de Música (ID: 10402).

10.7.12.5. Test Issue Attachments

FILMFLOW-81 doesn't have attachments.

10.7.12.6. Test Details

This Manual Test has 3 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING FOLDER: GET: US 14 API TESTING Obtener una lista de películas por género	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED



2	Haz clic en la solicitud "ID: 10402 - Música" del folder: "US 14 API TESTING Obtener una lista de películas por género" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET					PASSED
3	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar que todas las películas sean del género Musica (ID: 10402)", tenga el estado PASS					PASSED

10.7.12.7. Requirements linked with this test
None.



10.7.13. [FILMFLOW-82](#): US 14 | TS 01 | TC 13 | Genero misterio

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.7.13.1. Execution Defects

There are 0 Execution Defects on this Test Run.

10.7.13.2. Execution Evidence

There are 0 Execution Evidence on this Test Run.

10.7.13.3. Comment

None.

10.7.13.4. Test Description

Verificar que todas las películas en la respuesta pertenezcan al género de Misterio (ID: 9648).

10.7.13.5. Test Issue Attachments

FILMFLOW-82 doesn't have attachments.

10.7.13.6. Test Details

This Manual Test has 3 Steps.



Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING FOLDER: GET: US 14 API TESTING Obtener una lista de películas por género	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "ID: 9648 - Misterio" del folder: "US 14 API TESTING Obtener una lista de películas por género" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET					PASSED



3	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar que todas las películas sean del género Misterio (ID: 9648)", tenga el estado PASS					PASSED
---	--	--	--	--	--	--	--	---------------

10.7.13.7. *Requirements linked with this test*
None.

10.7.14. **FILMFLOW-83**: US 14 | TS 01 | TC 14 | Genero romance

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.7.14.1. *Execution Defects*
There are 0 Execution Defects on this Test Run.

10.7.14.2. *Execution Evidence*
There are 0 Execution Evidence on this Test Run.

10.7.14.3. *Comment*
None.



10.7.14.4. Test Description

Verificar que todas las películas en la respuesta pertenezcan al género de Romance (ID: 10749).

10.7.14.5. Test Issue Attachments

FILMFLOW-83 doesn't have attachments.

10.7.14.6. Test Details

This Manual Test has 3 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING FOLDER: GET: US 14 API TESTING Obtener una lista de películas por género	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED



2	Haz clic en la solicitud "ID: 10749 - Romance" del folder: "US 14 API TESTING Obtener una lista de películas por género" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET					PASSED
3	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar que todas las películas sean del género Romance (ID: 10749)", tenga el estado PASS					PASSED

10.7.14.7. Requirements linked with this test

None.



10.7.15. [FILMFLOW-84](#): US 14 | TS 01 | TC 15 | Genero ciencia ficcion

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.7.15.1. Execution Defects

There are 0 Execution Defects on this Test Run.

10.7.15.2. Execution Evidence

There are 0 Execution Evidence on this Test Run.

10.7.15.3. Comment

None.

10.7.15.4. Test Description

Verificar que todas las películas en la respuesta pertenezcan al género de Ciencia ficción (ID: 878).

10.7.15.5. Test Issue Attachments

FILMFLOW-84 doesn't have attachments.

10.7.15.6. Test Details

This Manual Test has 3 Steps.



Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING FOLDER: GET: US 14 API TESTING Obtener una lista de películas por género	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "ID: 878 - Ciencia ficción" del folder: "US 14 API TESTING Obtener una lista de películas por género" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET					PASSED



3	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar que todas las películas sean del género Ciencia Ficción (ID: 878)", tenga el estado PASS						PASSED
---	--	--	--	--	--	--	--	--	--------

10.7.15.7. *Requirements linked with this test*
None.

10.7.16. [FILMFLOW-85](#): US 14 | TS 01 | TC 16 | Genero ciencia pelicula de tv

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.7.16.1. *Execution Defects*
There are 0 Execution Defects on this Test Run.

10.7.16.2. *Execution Evidence*
There are 0 Execution Evidence on this Test Run.



10.7.16.3. *Comment*
None.

10.7.16.4. *Test Description*
Verificar que todas las películas en la respuesta pertenezcan al género de Película de TV (ID: 10770).

10.7.16.5. *Test Issue Attachments*
FILMFLOW-85 doesn't have attachments.

10.7.16.6. *Test Details*
This Manual Test has 3 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING FOLDER: GET: US 14 API TESTING Obtener una lista de películas por género	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED



2	Haz clic en la solicitud "ID: 10770 - Película de TV" del folder: "US 14 API TESTING Obtener una lista de películas por género" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET					PASSED
3	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar que todas las películas sean del género Pelicula de TV (ID: 10770)", tenga el estado PASS					PASSED

10.7.16.7. Requirements linked with this test

None.



10.7.17. [FILMFLOW-86](#): US 14 | TS 01 | TC 17 | Genero ciencia suspense

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.7.17.1. Execution Defects

There are 0 Execution Defects on this Test Run.

10.7.17.2. Execution Evidence

There are 0 Execution Evidence on this Test Run.

10.7.17.3. Comment

None.

10.7.17.4. Test Description

Verificar que todas las películas en la respuesta pertenezcan al género de Suspense (ID: 53)

10.7.17.5. Test Issue Attachments

FILMFLOW-86 doesn't have attachments.

10.7.17.6. Test Details

This Manual Test has 3 Steps.



Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING FOLDER: GET: US 14 API TESTING Obtener una lista de películas por género	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "ID: 53 - Suspense" del folder: "US 14 API TESTING Obtener una lista de películas por género" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET					PASSED



3	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar que todas las películas sean del género Suspense (ID: 53)", tenga el estado PASS					PASSED
---	--	--	--	--	--	--	--	--------

10.7.17.7. *Requirements linked with this test*
None.

10.7.18. [FILMFLOW-87](#): US 14 | TS 01 | TC 18 | Genero belica

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.7.18.1. *Execution Defects*
There are 0 Execution Defects on this Test Run.

10.7.18.2. *Execution Evidence*
There are 0 Execution Evidence on this Test Run.

10.7.18.3. *Comment*
None.



10.7.18.4. Test Description

Verificar que todas las películas en la respuesta pertenezcan al género de Belica (ID: 10752)

10.7.18.5. Test Issue Attachments

FILMFLOW-87 doesn't have attachments.

10.7.18.6. Test Details

This Manual Test has 3 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING FOLDER: GET: US 14 API TESTING Obtener una lista de películas por género	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED



2	Haz clic en la solicitud "ID: 10752 - Bélgica" del folder: "US 14 API TESTING Obtener una lista de películas por género" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET					PASSED
3	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar que todas las películas sean del género Belica (ID: 10752)", tenga el estado PASS					PASSED

10.7.18.7. Requirements linked with this test
None.

10.7.19. [FILMFLOW-88](#): US 14 | TS 01 | TC 19 | Genero western

Execution Status	Assignee	Executed By	Started On	Finished On
------------------	----------	-------------	------------	-------------



PASSED	Mateo Joel Lopez	Mateo Joel Lopez		
--------	------------------	------------------	--	--

10.7.19.1. Execution Defects
There are 0 Execution Defects on this Test Run.

10.7.19.2. Execution Evidence
There are 0 Execution Evidence on this Test Run.

10.7.19.3. Comment
None.

10.7.19.4. Test Description
Verificar que todas las películas en la respuesta pertenezcan al género de Western (ID: 37)

10.7.19.5. Test Issue Attachments
FILMFLOW-88 doesn't have attachments.

10.7.19.6. Test Details
This Manual Test has 3 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------



1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING FOLDER: GET: US 14 API TESTING Obtener una lista de películas por género	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "ID: 37 - Western" del folder: "US 14 API TESTING Obtener una lista de películas por género" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET					PASSED



3	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar que todas las películas sean del género Western (ID: 37)", tenga el estado PASS					PASSED
---	--	--	---	--	--	--	--	--------

10.7.19.7. *Requirements linked with this test*
None.

10.8. [FILMFLOW-96](#): US 20 | TS 01 | Obtener una lista de generos (Series de TV)

10.8.1. [FILMFLOW-97](#): US 20 | TS 01 | TC 01 | Generos validos

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.8.1.1. *Execution Defects*
There are 0 Execution Defects on this Test Run.

10.8.1.2. *Execution Evidence*
There are 0 Execution Evidence on this Test Run.



10.8.1.3. Comment

None.

10.8.1.4. Test Description

Verificar que se reciba una lista de géneros válidos

10.8.1.5. Test Issue Attachments

FILMFLOW-97 doesn't have attachments.

10.8.1.6. Test Details

This Manual Test has 3 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING GET: US 20 API TESTING Obtener una lista de generos (SERIES)	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED



2	Haz clic en la solicitud "US 20 API TESTING Obtener una lista de generos (SERIES)" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET					PASSED
3	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar que se reciba una lista de géneros válidos", tenga el estado PASS					PASSED

10.8.1.7. *Requirements linked with this test*
None.

10.8.2. **FILMFLOW-98:** US 20 | TS 01 | TC 02 | EndPoint accesible

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.8.2.1. *Execution Defects*
There are 0 Execution Defects on this Test Run.



10.8.2.2. Execution Evidence

There are 0 Execution Evidence on this Test Run.

10.8.2.3. Comment

None.

10.8.2.4. Test Description

Verificar código de estado HTTP 200

10.8.2.5. Test Issue Attachments

FILMFLOW-98 doesn't have attachments.

10.8.2.6. Test Details

This Manual Test has 3 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------



1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING GET: US 20 API TESTING Obtener una lista de generos (SERIES)	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 20 API TESTING Obtener una lista de generos (SERIES)" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET					PASSED
3	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar código de estado HTTP 200", tenga el estado PASS					PASSED

10.8.2.7. Requirements linked with this test
None.



10.8.3. [FILMFLOW-99](#): US 20 | TS 01 | TC 03 | Array de objetos

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.8.3.1. Execution Defects

There are 0 Execution Defects on this Test Run.

10.8.3.2. Execution Evidence

There are 0 Execution Evidence on this Test Run.

10.8.3.3. Comment

None.

10.8.3.4. Test Description

Verificar respuesta como array de objetos JSON

10.8.3.5. Test Issue Attachments

FILMFLOW-99 doesn't have attachments.

10.8.3.6. Test Details

This Manual Test has 3 Steps.



Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING GET: US 20 API TESTING Obtener una lista de generos (SERIES)	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic en la solicitud "US 20 API TESTING Obtener una lista de generos (SERIES)" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET					PASSED
3	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar respuesta como array de objetos JSON", tenga el estado PASS					PASSED



10.8.3.7. *Requirements linked with this test*
None.

10.8.4. [FILMFLOW-100](#): US 20 | TS 01 | TC 04 | Campos válidos

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.8.4.1. *Execution Defects*
There are 0 Execution Defects on this Test Run.

10.8.4.2. *Execution Evidence*
There are 0 Execution Evidence on this Test Run.

10.8.4.3. *Comment*
None.

10.8.4.4. *Test Description*

10.8.4.5. *Test Issue Attachments*
FILMFLOW-100 doesn't have attachments.



10.8.4.6. Test Details
This Manual Test has 3 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING GET: US 20 API TESTING Obtener una lista de generos (SERIES)	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					<div>PASSED</div>
2	Haz clic en la solicitud "US 20 API TESTING Obtener una lista de generos (SERIES)" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET					<div>PASSED</div>



3	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar campos 'id' y 'name' en cada género", tenga el estado PASS						PASSED
---	--	--	--	--	--	--	--	--	--------

10.8.4.7. Requirements linked with this test
None.

10.8.5. **FILMFLOW-101**: US 20 | TS 01 | TC 05 | Lista de series vacias

Execution Status	Assignee	Executed By	Started On	Finished On
PASSED	Mateo Joel Lopez	Mateo Joel Lopez		

10.8.5.1. Execution Defects
There are 0 Execution Defects on this Test Run.

10.8.5.2. Execution Evidence
There are 0 Execution Evidence on this Test Run.

10.8.5.3. Comment
None.



10.8.5.4. Test Description

Verificar que la lista de géneros no esté vacía

10.8.5.5. Test Issue Attachments

FILMFLOW-101 doesn't have attachments.

10.8.5.6. Test Details

This Manual Test has 3 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING GET: US 20 API TESTING Obtener una lista de generos (SERIES)	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED



2	Haz clic en la solicitud "US 20 API TESTING Obtener una lista de generos (SERIES)" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET					PASSED
3	Hacer click en el boton SEND al lado de la url de la solicitud		En la pestaña "Tests results" verificar que el test llamado: "Verificar que la lista de géneros no esté vacía", tenga el estado PASS					PASSED

10.8.5.7. Requirements linked with this test
None.

10.9. [FILMFLOW-103](#): US 21 | TS 01 | Verificación de Disponibilidad de Imágenes de Portada de Series

10.9.1. [FILMFLOW-104](#): US 21 | TS 01 | TC 01 | Verificación de Disponibilidad de Imágenes

Execution Status	Assignee	Executed By	Started On	Finished On
FAILED	Mateo Joel Lopez	Mateo Joel Lopez		



10.9.1.1. Execution Defects

There are 0 Execution Defects on this Test Run.

10.9.1.2. Execution Evidence

There are 0 Execution Evidence on this Test Run.

10.9.1.3. Comment

None.

10.9.1.4. Test Description

Se verificará que las imagenes de cada una de las páginas este disponible y tenga un estado de acceso 200 OK.

10.9.1.5. Test Issue Attachments

FILMFLOW-104 doesn't have attachments.

10.9.1.6. Test Details

This Manual Test has 4 Steps.

Step	Action	Data	Expected Result	Attachments	Comment	Defects	Evidence	Status
------	--------	------	-----------------	-------------	---------	---------	----------	--------



1	Abre la aplicación Postman y asegúrate de que tu colección y solicitud estén configuradas correctamente.	COLECCION: DB PELICULAS TESTING FOLDER: US 21 Verificación de Disponibilidad de Imágenes de Portada de Series	Visualizar la ventana correspondiente aL WORKSPACE de postman FilmFlow					PASSED
2	Haz clic derecho en el folder "US 21 Verificación de Disponibilidad de Imágenes de Portada de Series" en tu colección "DB PELICULAS TESTING".		Visualizar la ventana correspondiente a la solicitud GET					PASSED
3	Hacer click en el boton RUN FOLDER		Se debería abrir una nueva pestaña en postman llamada RUNNER donde se prepara la coleccion del folder para ejecutar los tests preparados					PASSED



4	Hacer click en RUN DB PELICULAS TESTING		Se deberia empezar a ejecutar todos los tests de la coleccion. Verificar en la pestaña ALL TESTS que todos los casos de prueba obtengan el estado PASS			FILMFLOW-105		FAILED
---	---	--	--	--	--	------------------------------	--	--------

10.9.1.7. Requirements linked with this test
None.



11. Appendix A: Approval

The undersigned acknowledge they have reviewed the **Test Plan** and agree with the approach it presents. Changes to this **Test Plan** will be coordinated with and approved by the undersigned or their designated representatives.

Signature:

Date:

Print Name:

Title:

Role: