## **Production Engineering**



## Cinema Managing Application

## The Storm Dragonflies

Alina Georgescu Ovidiu Iordache Daria Runceanu Grupa 344

#### Cinema

- id
- name
- email
- street
- number
- city

## **Auditorium**

- id
- name
- seatNumber

#### Client

- id
- name

#### Movie

- id
- title
- year
- description
- type
- name

- /cinema/add\_cinema
  (POST)
- /cinema/cinemas (GET)
- /cinema/update\_cinema
- ➤ (PUT)
- /cinema/get\_cinema/{id}
- ➤ (GET)
- /cinema/name/{name}
- ➤ (GET)
- cinema/city/{city}
- ➤ (GET)

- /auditorium/add-auditorium (POST)
- auditorium/get-auditorium (GET)
- /cinema/get\_cinema/{id}
- → (GET)
- auditorium/get-auditoriumby-name/{name} (GET)
- auditorium/auditoriums-formovie/{movield} (GET)
- /get-cinemaauditoriums/{cinema\_id}(GET)

- /client
  (GET, POST)
- /clienti/{id}
  (GET, PUT, DELETE)

- /movies (GET, POST)
- /movies/{id}
  GET, PUT, DELETE)

# **TESTING**



**Unit Tests** 

Class	Tests	Failures	Ignored	Duration	Success rate
<u>AuditoriumTest</u>	6	0	0	0.003s	100%
<u>CinemaTest</u>	13	0	0	0.039s	100%
ClientTest	6	0	0	0.002s	100%



Integration Tests



**Performance** Tests



**Behavior Driven Tests** 

Deprecated Gradle features were used in this build, making it incompatible with Gradle 8.0.

You can use '--warning-mode all' to show the individual deprecation warnings and determine if they come from your own scripts or plugins.

See https://docs.gradle.org/7.2/userguide/command line interface.html#sec:command line warnings

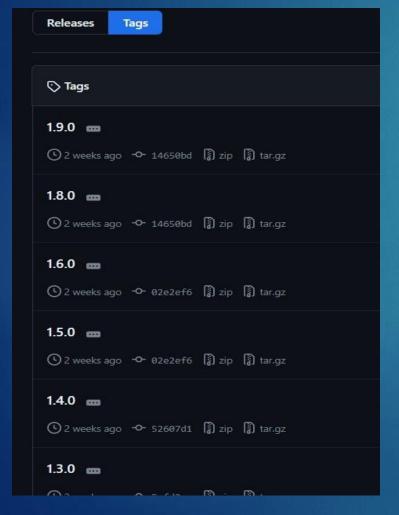
**BUILD SUCCESSFUL in 16s** 

5 actionable tasks: 1 executed, 4 up-to-date

Sample # †	Start Time	Thread Name	Label		Status	Bytes	Sent Bytes	Latency	Connect Ti
473	14:09:39.657	Thread Gro	HTTP Request		<b>©</b>	290	131	34	
474	14:09:39.628	Thread Gro	HTTP Request		⊗	290	131	34	
475	14:09:39.644	Thread Gro	HTTP Request		<b>©</b>	290	131	55	
476	14:09:39.630	Thread Gro	HTTP Request	71	⊗	290	131	28	
477	14:09:39.665	Thread Gro	HTTP Request	36	<b>©</b>	290	131	16	
478	14:09:39.653	Thread Gro	HTTP Request	50	<b>⊘</b>	290	131	29	
479	14:09:39.662	Thread Gro	HTTP Request	46	<b>©</b>	290	131	24	
480	14:09:39.654	Thread Gro	HTTP Request	54	<b>©</b>	290	131	27	
481	14:09:39.662	Thread Gro	HTTP Request	47	<b>©</b>	290	131	19	
482	14:09:39.664	Thread Gro	HTTP Request	45	⊗	290	131		
483	14:09:39.673	Thread Gro	HTTP Request	38	<b>©</b>	290	131	23	
484	14:09:39.663	Thread Gro	HTTP Request	56	€	290	131	40	
485	14:09:39.653	Thread Gro	HTTP Request	66	<b>©</b>	290	131	27	
486	14:09:39.641	Thread Gro	HTTP Request	78	<b>⊘</b>	290	131	42	
487	14:09:39.659	Thread Gro	HTTP Request	61	<b>©</b>	290	131	39	
488	14:09:39.651	Thread Gro	HTTP Request		<b>©</b>	290	131		
489	14:09:39.675	Thread Gro	HTTP Request	48	<b>©</b>	290	131	26	
490	14:09:39.676	Thread Gro	HTTP Request		€	290	131	33	
491	14:09:39.681	Thread Gro	HTTP Request	56	<b>©</b>	290	131	18	
492	14:09:39.685	Thread Gro	HTTP Request	52	<b>⊗</b>	290	131	18	
493	14:09:39.685	Thread Gro	HTTP Request	61	<b>©</b>	290	131	25	
494	14:09:39.699	Thread Gro	HTTP Request	50	⊗	290	131	24	
495	14:09:39.697	Thread Gro	HTTP Request	53	<b>©</b>	290	131	17	
496	14:09:39.651	Thread Gro	HTTP Request	99	⊗	290	131	37	
497	14:09:39.648	Thread Gro	HTTP Request	103	<b>©</b>	290	131	49	
498	14:09:39.698	Thread Gro	HTTP Request	53	<b>⊘</b>	290	131		
499	14:09:39.701	Thread Gro	HTTP Request	51	<b>⊗</b>	290	131	36	

# CI/CD

Using Jenkins we automated the process of developing the app by continuously checking the new features added pushing the new builds to Docker Hub and publishing new versions on GitHub.



## MONITORING

