


Open · Opened 2 days ago by  **Rubén Montero**

## GET /movies/{movieID} (404)

### Resumen

- Si la película especificada no existe, devolveremos un código de error HTTP **404 Not Found**
- Para ello, añadiremos un **if** y lanzaremos una excepción **ResourceException** en **GetMovieByID.java**

### Descripción

Ahora mismo, si visitamos:

- **http://localhost:8104/movies/4984** (el ID no existe)

...la página de respuesta indicará **Internal Server Error**, y en IntelliJ IDEA podemos ver:

Exception or error caught in server resource

```
org.restlet.resource.ResourceException: Internal Server Error (500) - The server encountered an unexpected condition which pr
    at org.restlet.resource.ServerResource.doHandle(ServerResource.java:527)
    at org.restlet.resource.ServerResource.get(ServerResource.java:723)
    ...
```

### Un código de error adecuado

Las respuestas HTTP van siempre acompañadas de un código:

- 1xx Información
- 2xx Éxito
- 3xx Redirección
- 4xx Error cliente
- 5xx Error servidor

Hasta ahora no nos hemos preocupado por ello, pues hemos programado el *happy path* por el cual las cosas **siempre** salen bien y una respuesta 2xx es generada automáticamente. Predeciblemente, **200 OK**.

Pero ahora, ante un cliente que visita un **/movies/{movieID}** con un **{movieID}** inválido... ¡está saltando un **NullPointerException** que Restlet traduce en *Internal Server Error*!

Nuestra **Movie movie** es **null**.

### Controlémoslo

En **GetMovieByID.java**, podemos controlar dicho escenario:

```
@Get
public StringRepresentation getEndpointResponse() {
    String movieID = getAttribute("movieID");
    Integer movieInt = Integer.parseInt(movieID);
    MoviesConnector connector = new MoviesConnector();
    Movie movie = connector.getByID(movieInt);
    if (movie == null) {
        // ...
    }
    // ...
}
```

#### ResourceException

Para encargar a Restlet que *devuelva un código HTTP* en concreto, lo hacemos *lanzando* una excepción.

Igual que el típico **throw new RuntimeException();** que solíamos escribir. Pero en esta ocasión, será:

```
if (movie == null) {
```

```
    throw new ResourceException(404, "Error phrase");  
}
```

¡Listo!

## La tarea

---

Se pide que:

- Controles el caso en que `{movieID}` apunta a una película que no existe.
- Si eso sucede, el servidor REST devolverá un error `404` con *frase* de error: `"The movie was not found"`

## Por último

Verifica que el `test` funciona correctamente.

Haz `commit` y `push` para subir los cambios al repositorio.

---



[Rubén Montero @ruben.montero](#) changed milestone to [%Sprint 2](#) 2 days ago