

## 1. Línea de control y bitácora del documento

Identificador	FODD-001
Proyecto	Prueba técnica programador
Fecha	2022-08-16
Lenguaje	Java, Python.
Revisores	Juan Camilo Castro Pinto David Tafur

Esta prueba de programación se deberá implementar utilizando algún framework en Java o Python que permita crear servicios web tipo REST. Al finalizar la prueba se debe comprimir todos los archivos fuentes (evitar archivos compilados) y enviar como mensaje adjunto al correo [juan.castro@software-colombia.com](mailto:juan.castro@software-colombia.com) con el siguiente asunto *[Nombre completo del aspirante] - [No. Documento de identidad] - Prueba Back End*. **En el cuerpo del mensaje se debe mencionar el framework utilizado y las instrucciones necesarias para ejecutar el proyecto.**

La prueba consiste en utilizar el siguiente archivo CSV: [Descargar aquí](#), para crear una serie de servicios web tipo REST. El archivo es un compilado de películas y se ve de la siguiente manera:



ID	Film	Genre	Studio	Score	Year
1	Zack and Miri Make a Porno	Romance	The Weinstein Company	70	2008
2	Youth in Revolt	Comedy	The Weinstein Company	52	2010
3	You Will Meet a Tall Dark Stranger	Comedy	Independent	35	2010
4	When in Rome	Comedy	Disney	44	2010
5	What Happens in Vegas	Comedy	Fox	72	2008
6	Water For Elephants	Drama	20th Century Fox	72	2011
7	WALL-E	Animation	Disney	89	2008
8	Waitress	Romance	Independent	67	2007
9	Waiting For Forever	Romance	Independent	53	2011
10	Valentine's Day	Comedy	Warner Bros.	54	2010
11	Tyler Perry's Why Did I get Married	Romance	Independent	47	2007
12	Twilight: Breaking Dawn	Romance	Independent	68	2011
13	Twilight	Romance	Summit	82	2008
14	The Ugly Truth	Comedy	Independent	68	2009
15	The Twilight Saga: New Moon	Drama	Summit	78	2009
16	The Time Traveler's Wife	Drama	Paramount	65	2009
17	The Proposal	Comedy	Disney	74	2009
18	The Invention of Lying	Comedy	Warner Bros.	47	2009
19	The Heartbreak Kid	Comedy	Paramount	41	2007
20	The Duchess	Drama	Paramount	68	2008

Las columnas se explican a continuación:

Nombre	Descripción	Tipo
ID	Identificador único de la película	Int
Film	Nombre de la película	String
Genre	Género de la película	String
Studio	Estudio que produjo la película	String
Score	Puntuación de la audiencia (0 - 100)	Int
Year	Año de estreno de la película	Int



Con base en la anterior información, se pide crear los siguientes servicios web tipo REST:

1. **Obtener una película por ID:** Este servicio debe responder al método GET y recibirá un ID como parámetro de la URL y debe retornar la película correspondiente a ese ID. Por ejemplo:

Petición:

GET <https://localhost:8080/movie?id=7>

Respuesta:

```
{
  "id": 7,
  "film": "WALL-E",
  "genre": "Animation",
  "studio": "Disney",
  "score": 89,
  "year": 2008
}
```

2. **Listar las películas ordenadas alfabéticamente:** Este servicio debe responder al método GET y recibirá una variable llamada *total* como parámetro de la URL que indicará cuantos elementos máximo debe devolver el servicio y una variable llamada *order* la cual indicará si el orden alfabético es asc (ascendente) o desc (descendente). El servicio deberá devolver una lista con los elementos que cumplan estos criterios. Por ejemplo:

Petición:

GET <https://localhost:8080/movies?total=3&order=asc>



Respuesta:

```
[
  {
    "id": 77,
    "film": "(500) Days of Summer",
    "genre": "Comedy",
    "studio": "Fox",
    "score": 81,
    "year": 2009
  },
  {
    "id": 76,
    "film": "27 Dresses",
    "genre": "Comedy",
    "studio": "Fox",
    "score": 71,
    "year": 2008
  },
  {
    "id": 75,
    "film": "A Dangerous Method",
    "genre": "Drama",
    "studio": "Independent",
    "score": 89,
    "year": 2011
  }
]
```



3. **Agregar una película:** Este servicio debe responder a un método POST que reciba la información de la película que se desea agregar y arroje como respuesta un mensaje de éxito. Por ejemplo:

Petición:

POST <https://localhost:8080/movie>

Cuerpo de la petición:

```
{
  "id": 78,
  "film": "Parasite",
  "genre": "Drama",
  "studio": "Barunson E&A",
  "score": 97,
  "year": 2019
}
```

Respuesta:

```
{
  "message": "La película fue creada con éxito"
}
```

Finalmente la nueva película debe poder ser consultada utilizando el servicio 1

