

# Proyecto 2º Trimestre

---

AGGREGATIONS

**Contenido**

Contenido ..... 1

1. Estructura de la base de datos ..... 2

    1.1. Películas..... 2

    1.2. Series ..... 2

    1.3. Países..... 3

    1.4. Categorías..... 3

    1.5. Usuarios..... 3

2. Operadores usados: ..... 4

## 1. Estructura de la base de datos

He realizado una base de datos de un videoclub en línea. Gracias a esta base de datos la empresa podrá gestionar que usuarios pueden acceder a X películas o series, además de obtener variedad de información sobre los archivos multimedia.

La base de datos está compuesta por 5 colecciones:

### 1.1. Películas

```
{
  _id: 1,
  titulo: "Battleship",
  clasificacionEdad: 16,
  id_categoria: [01, 02, 05, 19],
  fechaSalida: new Date("2012-04-11"),
  duracionMinutos: 131,
  precioEuros: 3
}
```

**id:** Es el identificador único de cada película.

**título:** Campo alfanumérico que indica el título de la película.

**clasificacionEdad:** Campo numérico que indica el mínimo de edad requerido que necesita tener un usuario para acceder a la compra de dicha película.

**Id categoría:** Campo de tipo array numérico que hace referencia a las categorías de las películas, cuyos nombre se guardan en otra colección.

**fechaSalida:** Campo de tipo fecha que indica en que año salió la película.

**duracionMinutos:** Campo numérico que indica la duración en minutos de la película.

**precioEuros:** Campo numérico que indica el precio de la película sin IVA.

### 1.2. Series

```
{
  _id: 1,
  nombre: "La Casa de Papel",
  clasificacionEdad: 12,
  id_categoria: [01, 17],
  salidas: {fechasTemporadas: [new Date("2017-05-02"), new
Date("2018-05-01"), new Date("2019-07-04")], terminada: true},
  temporadas: 3,
  precioEuros: 10
}
```

**id:** Es el identificador único de cada serie.

**nombre:** Campo alfanumérico que indica el nombre de la serie.

**clasificacionEdad:** Campo numérico que indica el mínimo de edad requerido que necesita tener un usuario para acceder a la compra de dicha serie.

**id categoría:** Campo de tipo array numérico que hace referencia a las categorías de las películas, cuyos nombre se guardan en otra colección.

**salidas:** Campo de tipo documento, que contiene un array con campos de tipo fechas que indican cuando ha salido cada temporada, y también un campo de tipo booleano que indica si la serie está terminada.

**temporadas:** Campo numérico que indica el número de temporadas que tiene una serie.

**precioEuros:** Campo numérico que indica el precio de la serie sin IVA.

### 1.3. Países

```
{
  _id: 1,
  país: "espana",
  iva: 0.21,
  seriesPermitidas: ["La Casa de Papel", "Los 100", "Breaking Bad",
"Juego de Tronos", "Las Winx", "Death Note", "Prison Break", "Narcos",
"La que se avecina", "El Juego del Calamar"],
  peliculasPermitidas: ["Battleship", "Ice Age", "Toy Story",
"Cars", "Venganza", "Ready Player One", "Deadpool", "El Dictador", "Iron
Man", "Shrek", "Shin Shan: Perdidos en la Jungla", "Mortal Kombat",
"Chicken Little", "Harry Potter y la piedra filosofal", "Interstellar"]}
```

**id:** Es el identificador único de cada país.

**país:** Campo alfanumérico que indica el nombre de cada país.

**iva:** Campo numérico que indica el porcentaje de IVA que tiene cada país.

**seriesPermitidas:** Campo de tipo array alfanumérico que indica las series permitidas en cada país.

**peliculasPermitidas:** Campo de tipo array alfanumérico que indica las películas permitidas en cada país.

### 1.4. Categorías

```
{
  id_categoria: 01,
  nombre_categoria: "accion"
}
```

**id:** Es el identificador único de cada categoría.

**nombre\_categoria:** Campo alfanumérico que indica el nombre de la categoría.

### 1.5. Usuarios

```
{
  _id: 1,
  nombreUsuario: "franBuiPer",
  edad: 18,
  país: "espana",
}
```

**Id:** Es el identificador único de cada usuario

**nombreUsuario:** Campo alfanumérico que indica el nombre del usuario.

**edad:** Campo numérico que indica la edad del usuario.

**país:** Campo de tipo alfanumérico que indica de que país pertenece el usuario.

## **2. Operadores usados:**

**\$group:** Agrupa los documentos por la condición que le pongamos.

**\$match:** Filtra los documentos para dejar pasar solo los documentos que coincidan con las condiciones especificadas.

**\$project:** Pasa los documentos con los campos que le digamos a la siguiente etapa. Los campos especificados pueden ser campos existentes del documento o campos personalizados.

**\$lookup:** Permite unir 2 colecciones. Se elige una clave primaria y una clave foránea, y crea un campo nuevo en la colección base cuyo valor es un Array formado por los datos de la otra colección.

**\$sort:** Ordena los documentos para mostrarlos de manera ascendente o descendente.

**\$unwind:** Rompe un Array de documentos para generar un documento para cada elemento.

**\$arrayElemAt:** Devuelve el elemento de un array de la posición que le indiquemos.

**\$addFields:** Agrega nuevos campos a los documentos.

Todos estos, además de operadores de comparación, operación, etc, ya vistos en el primer trimestre.