

# DOCUMENTACIÓN PROYECTO INTERMODULAR

Michael González, Jonathan David Signes,  
Guillem Ruano, Carlos Gutiérrez

## **ÍNDEx**

- I. SPRINT 1 – Análisis del sistema
- II. SPRINT 2, Diseño de la arquitectura y del modelo (1 semana).
- III. SPRINT 3, Implementación inicial de los servicios y prototipos UI (1 semana).
- IV. SPRINT 4, Implementación completa e integraciones (1-2 semanas).
- V. SPRINT 5, Integración, pruebas y lanzamiento final (1 semana).

## **SPRINT 1 – Análisis del sistema**

**Duración:** 1 semana(8 -12 Diciembre).

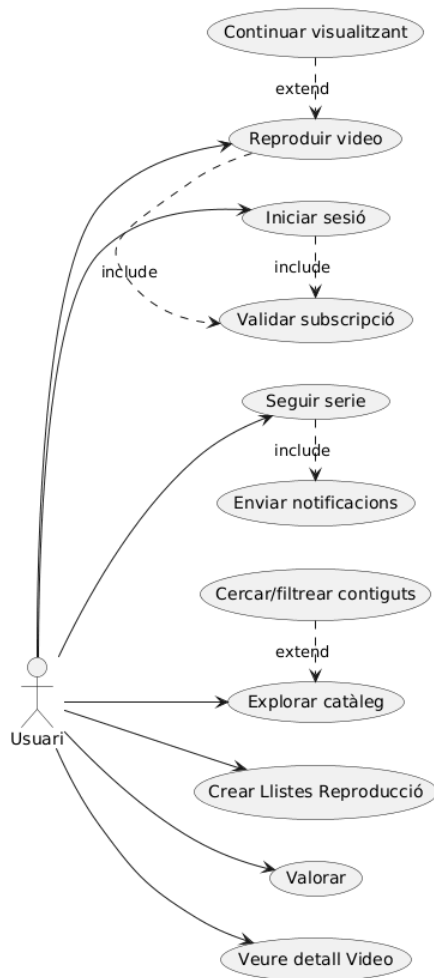
**Objetivo:** Comprender el problema, definir los requerimientos y establecer una primera visión de la arquitectura.

**Tareas:** Casos de uso y análisis de requerimientos, Modelaje Conceptual y API y arquitectura general.

## **CASOS DE USO Y REQUERIMIENTOS DEL SISTEMA**

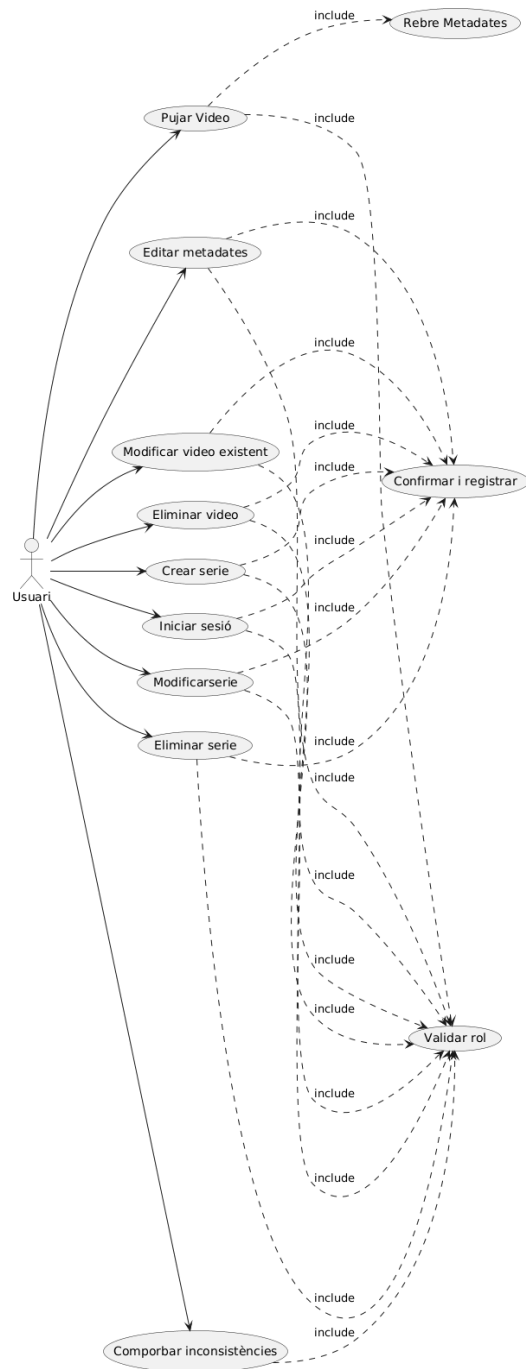
CLIENTE:

AppMóvil:



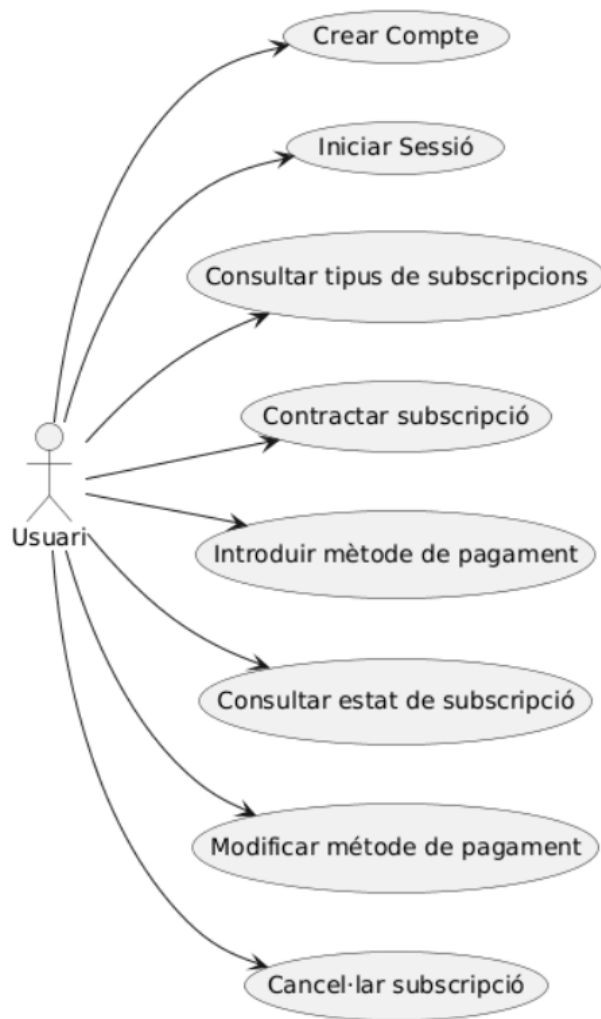
Los casos de uso de la App Móvil permiten al usuario iniciar sesión y, una vez validada su suscripción, explorar y buscar en el catálogo, ver detalles de vídeos, reproducir y continuar visualizaciones, seguir series, crear listas de reproducción, valorar contenidos y recibir notificaciones\*\*, integrando así todas las funcionalidades de consumo, interacción y personalización dentro de la experiencia de streaming.

## Escritorio Admin:



Los casos de uso del Cliente de Escritorio Admin permiten al administrador iniciar sesión y, tras validar su rol, realizar la gestión integral del catálogo multimedia, incluyendo subir vídeos (recibiendo sus metadates), crear, modificar y eliminar tanto vídeos como series, editar metadatos, confirmar y registrar cambios, y comprobar inconsistencias en el sistema, centralizando así todas las tareas de administración y control de calidad del contenido.

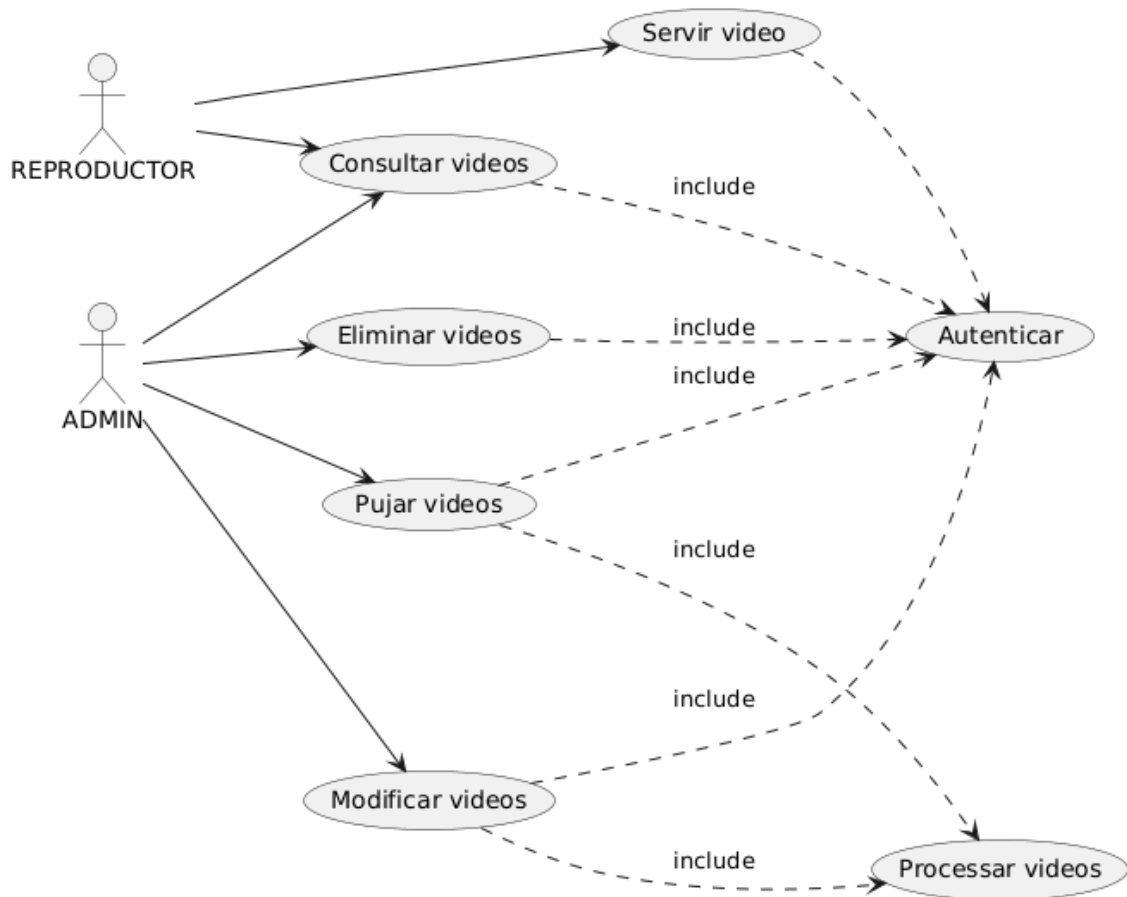
## Portal Web Odoo:



Los casos de uso de Odoo permiten al usuario gestionar su cuenta y suscripción (crear cuenta, iniciar sesión, consultar, contratar y cancelar suscripciones, así como gestionar métodos de pago), mientras que el administrador tiene capacidades extendidas para realizar operaciones CRUD completas sobre suscripciones y métodos de pago, controlando así toda la capa comercial y de usuarios del sistema.

## SERVIDOR:

Express:



Los casos de uso del servidor Express se centran en la gestión y streaming de vídeos en formato HLS: permite servir los vídeos para su reproducción y, previa autenticación, ofrece a los administradores operaciones CRUD completas (consultar, subir, modificar y eliminar vídeos), actuando como el gestor técnico del contenido multimedia en el sistema.

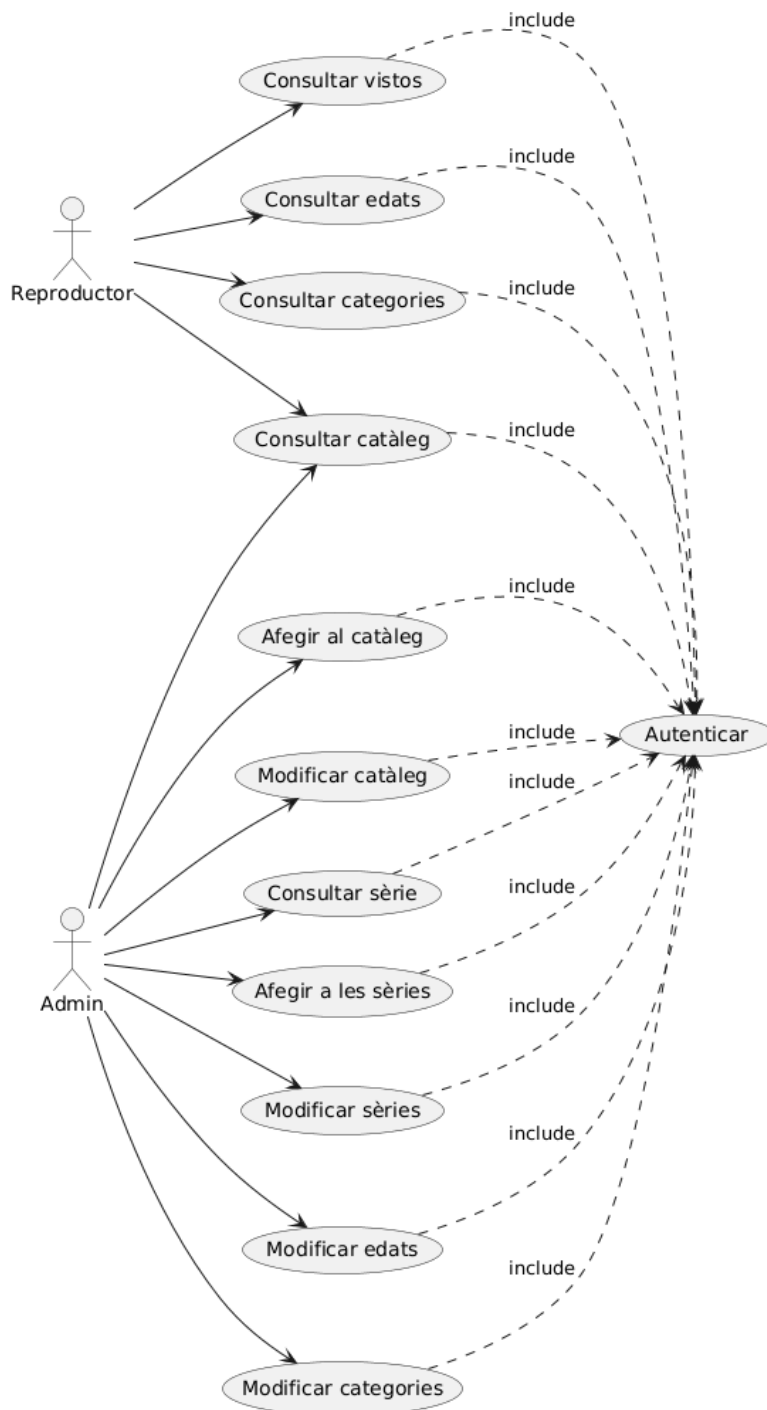
Odoo:



Los casos de uso del servidor Odoo giran en torno a la *gestión de usuarios y suscripciones*: permite *autenticar* usuarios, crear cuentas, consultar roles y tipos de suscripción, contratar, cancelar o cambiar suscripciones, modificar métodos de pago y enviar notificaciones, sirviendo así como el módulo central de identidad, comercialización y comunicación del sistema.



Spring:



El módulo SpringBoot expone una API que permite a los usuarios finales *consultar el catálogo*, los vídeos vistos y las series disponibles, mientras que los administradores, previa autenticación, pueden gestionar completamente el catálogo mediante operaciones *CRUD* sobre vídeos, series, categorías y clasificaciones por edad, centralizando así toda la lógica de gestión y consulta de contenidos multimedia en el sistema.

## **REQUERIMIENTOS DEL SISTEMA**

Express

### **Funcionales:**

1. Se sirven vídeos y thumbnails.
2. Se puede modificar vídeos.
3. Se pueden consultar vídeos.
4. Se pueden servir vídeos.
5. Se pueden eliminar vídeos .
6. Se pueden procesar vídeos.

### **No Funcionales:**

1. Hecho en Node JS y Express.
  2. TypeScript lenguaje
  3. Validación mediante Odoo
- 

Odoo

### **Funcionales:**

1. Se pueden crear usuarios.
2. Se pueden modificar usuarios.
3. Se puede Iniciar Sesión.
4. Se puede comprobar el token.
5. Eliminar usuarios.
6. CRUD de suscripciones.
7. CRUD métodos de pago.

### **No Funcionales:**

1. Lenguaje Python para los Módulos.
2. XML para las views.
3. Web integrada de Odoo.

---

Spring

**Funcionales:**

1. CRUD catálogo,
2. CRUD series,
3. CRUD categorías,
4. CRUD edades,
5. CRUD vídeos vistos,
6. CRUD series seguidas

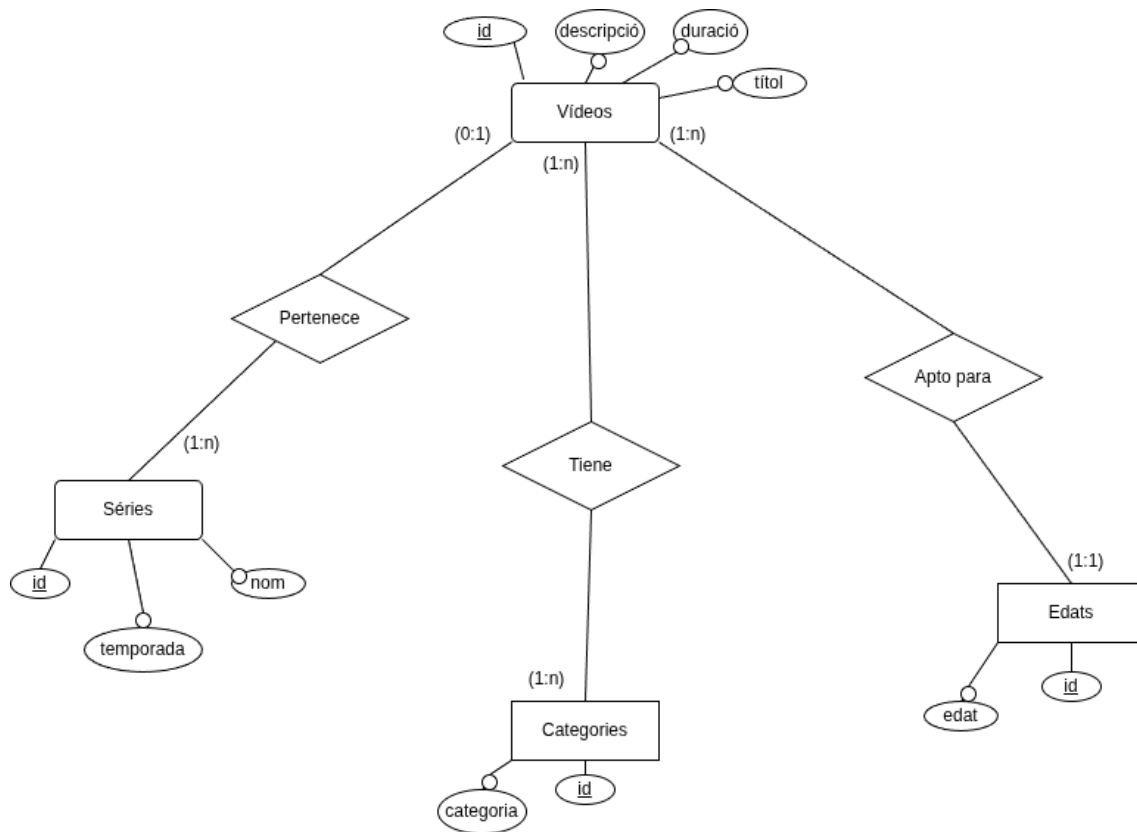
**No Funcionales:**

1. Programado en JAVA.
2. Uso de Hibernate.
3. Uso de Spring
4. Base de datos relacional MySQL.
5. Base de deatos no relacional MongoDB.
6. Validar mediante Odoo.

## MODELAJE CONCEPTUAL

### Diagramas ER:

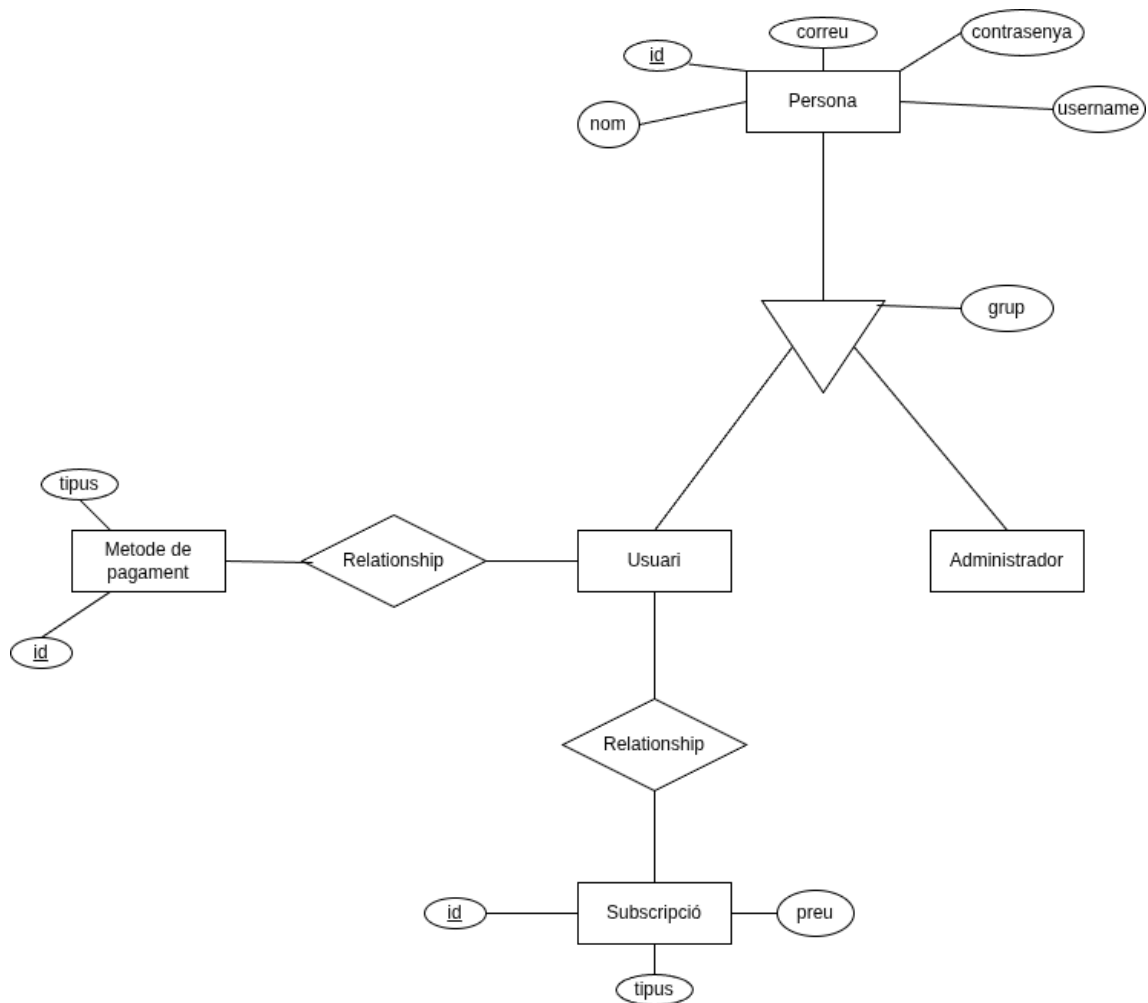
MySQL:



Este diagrama representa la base de datos de SpringBoot que usaremos para el catálogo de vídeos.

La entidad principal es *Vídeos*, la cual se relaciona con *Series*, *Categorías* y *Edades*.

PostgreSQL:



Este es la base de datos de Odoo, que usaremos para gestionar los usuarios y suscripciones.

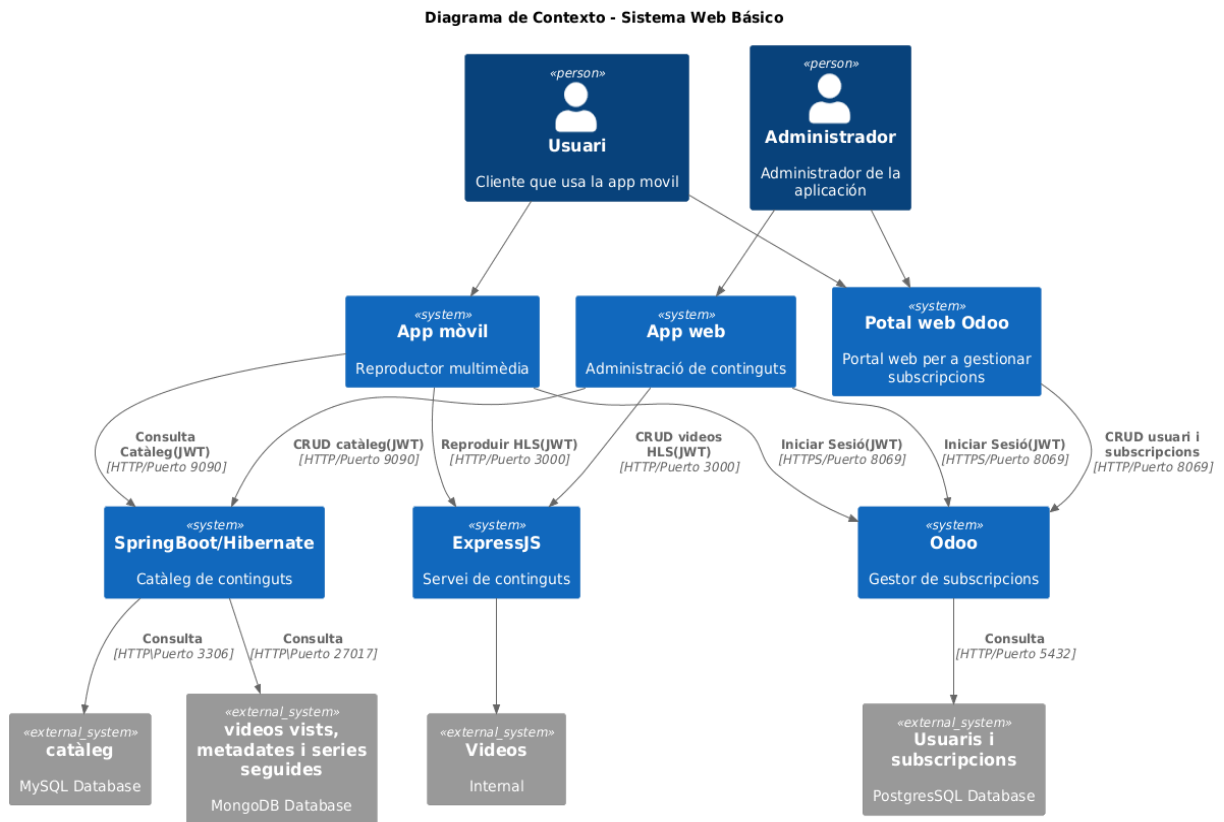
Las entidades principales de este diagrama son *Usuario* y *Administrador*, que heredan de *Persona*, a demás, existen las entidades *Método de pago* y *Suscripción* que se relacionan con *Usuario*.

MongoDB:

No es un diagrama de Entidad Relación, JSON a utilizar:

```
VideosVist{  
    • Id  
    • videoid  
    • usuarioid  
    • dataVisualitzacio  
    • minutActual  
    • completat  
}  
  
SeriesSeguides {  
    • id  
    • serieId  
    • usuarioid  
    • dataInici  
    • dataUltimAcces  
    • episodisVists  
    • notificaciones  
}  
  
MetadatosVideo{  
    • id  
    • videoid  
    • {metadata}  
    • dataProcessament  
    • format  
    • qualitat  
    • mida  
}
```

**Diagrama C4:**



**1-Usuari**, el usuario accede a la App Móvil para reproducir vídeos, donde inicia sesión mediante Odoo.

**2-Administrador**, el admin hace uso de la App Web para administrar los contenidos y hace uso del Portal Web Odoo donde gestionará las subscripciones.

**3-SpringBoot/Hibernate**, Consulta: gestiona las peticiones de la *App Móvil* para mostrar el catálogo de vídeos/ CRUD: el *Administrador* desde la *App Web* gestiona los métodos CRUD del catálogo.

**4-Express**, Reproducir: se encarga de reproducir los vídeos que pide el Cliente desde la App Móvil/ CRUD: el Admin se encarga de todos los métodos CRUD de los vídeos.

**5-Odoo**, Inicio Sesión: se encarga del *Login* tanto de la *App Web*, como de la *App Móvil*/ CRUD: el *Admin* el *Portal Web Odoo*, se encarga de gestionar las subscripciones y usuarios, más todo lo que conlleva.

**6•MySQL**, Base de datos relacional que se encarga de almacenar todo el contenido del catálogo, usado por Spring/Hibernate.

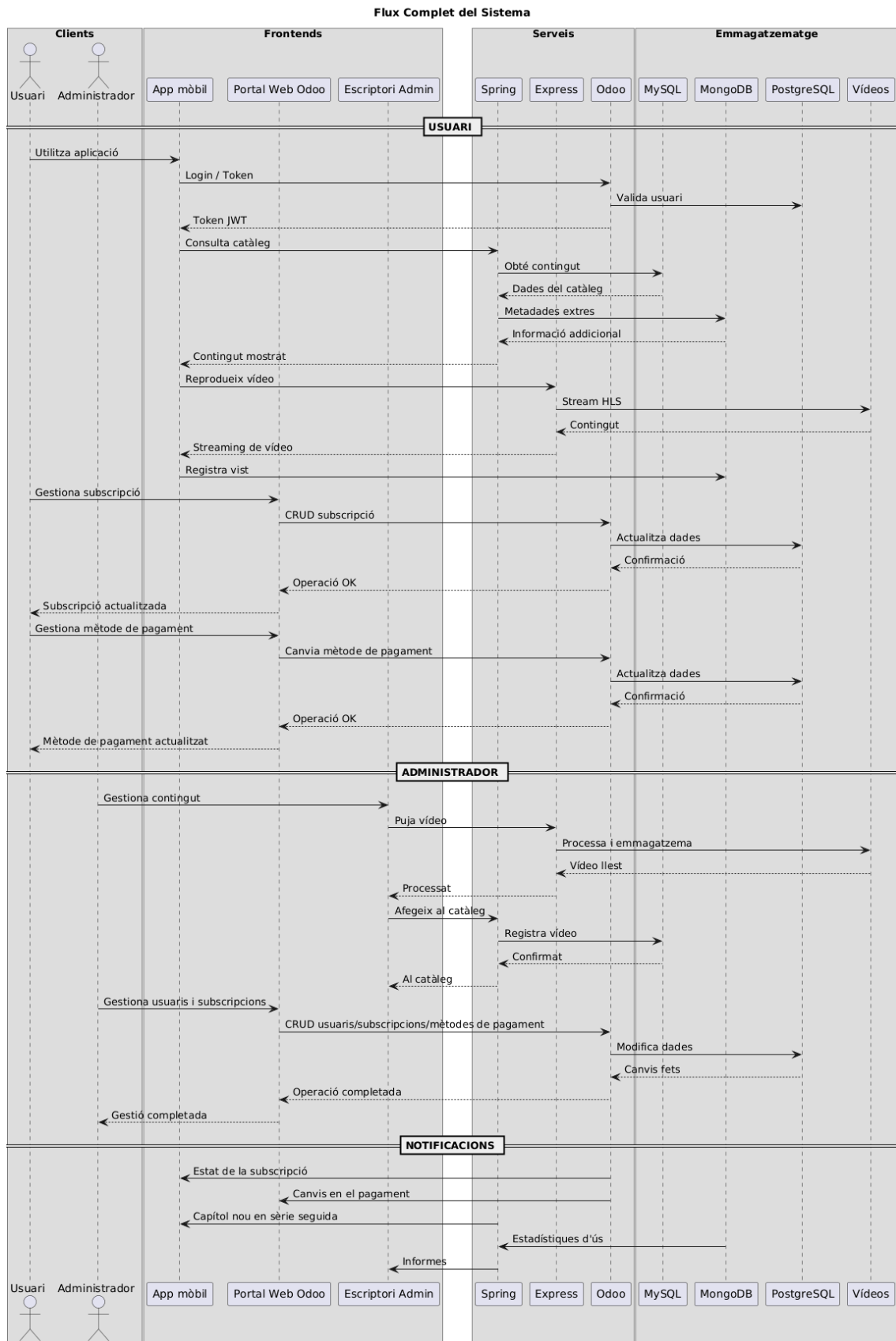
**7•MongoDB**, Base de datos no relacional que se encarga de almacenar en JSON, los vídeos vistos, series seguidas y metadatos de los vídeos,.

**8•Interna**, Base de datos interna donde guardamos los vídeos, usado en Express.

**9•PostgresSQL**, Base de datos relacional, que se encarga de almacenar los usuarios y las suscripciones.



## API/arquitectura general



Este diagrama de flujo describe el *ciclo completo de operaciones* en una plataforma de streaming, donde el *usuario* interactúa principalmente con la *App móvil* para autenticarse mediante Odoo, consultar el catálogo gestionado por Spring (que integra datos de MySQL y MongoDB), reproducir vídeos a través de Express y gestionar su suscripción y métodos de pago vía el Portal Web Odoo; mientras tanto, el *administrador* utiliza el *Esriptori Admin* para subir y catalogar contenido (mediante Express y Spring) y el *Portal Web Odoo* para gestionar usuarios y suscripciones, con Odoo coordinando toda la lógica comercial y notificando cambios relevantes a los usuarios y administradores.

## **CONTRATO INICIAL API**

### **·EXPRESS (Vídeos - Puerto 3000)**

#### **Streaming de Vídeos:**

##### **GET /api/videos/:id**

Respuesta: 200 OK

Error: 404 Not Found (si el video no existe)

Error: 500 Internal Server Error (error del servidor)

##### **GET /api/thumbnails/:id**

Respuesta: 200 OK

Error: 404 Not Found (si el thumbnail no existe)

Error: 500 Internal Server Error (error del servidor)

##### **POST /api/videos/**

Respuesta: 201 Created

Error: 500 Internal Server Error

##### **PUT /api/videos/:id**

Respuesta: 200 OK

Error: 404 Not Found

Error: 500 Internal Server Error

##### **DELETE /api/videos/:id**

Respuesta: 204 No Content

Error: 404 Not Found

Error: 500 Internal Server Error

## ·SPRING (Vídeos - Puerto 3000)

### Gestión de Contenidos(catálogo):

#### **GET /api/catalogo/:**

Respuesta: 200 OK

Error: 500 Internal Server Error

#### **GET /api/catalogo/:id**

Respuesta: 200 OK

Error: 404 Not Found

Error: 500 Internal Server Error

#### **GET /api/catalogo/vistos**

Respuesta: 200 OK

Error: 500 Internal Server Error

#### **GET /api/catalogo/:titulo**

Respuesta: 200 OK

Error: 404 Not Found

Error: 500 Internal Server Error

#### **POST /api/catalogo/**

Respuesta: 201 Created

Error: 400 Bad Request

Error: 500 Internal Server Error

#### **POST /api/catalogo/vistos**

Respuesta: 201 Created

Error: 400 Bad Request

Error: 500 Internal Server Error

#### **PUT /api/catalogo/:id**

Respuesta: 200 OK

Error: 404 Not Found

Error: 500 Internal Server Error

**DELETE /api/catalogo/:id:**

Respuesta: 204 No Content

Error: 404 Not Found

Error: 500 Internal Server Error

**GET /api/catalogo/edades/:edat**

Respuesta: 200 OK

Error: 404 Not Found

Error: 500 Internal Server Error

**GET /api/catalogo/categorías/:categoria**

Respuesta: 200 OK

Error: 404 Not Found

Error: 500 Internal Server Error

**GET /api/catalogo/serie/:id**

Respuesta: 200 OK

Error: 404 Not Found

Error: 500 Internal Server Error

## **Series:**

### **GET /api/series/**

Respuesta: 200 OK

Error: 500 Internal Server Error

### **GET /api/series/:id**

Respuesta: 200 OK

Error: 404 Not Found

Error: 500 Internal Server Error

### **POST /api/series/**

Respuesta: 201 Created

Error: 500 Internal Server Error

### **PUT /api/series/:id**

Respuesta: 200 OK

Error: 404 Not Found

Error: 500 Internal Server Error

### **PATCH /api/series/:id**

Respuesta: 200 OK

Error: 404 Not Found

Error: 500 Internal Server Error

### **DELETE /api/series/:id**

Respuesta: 204 No Content

Error: 404 Not Found

Error: 500 Internal Server Error

## **Edades y Categorías:**

### **GET /api/edades/**

Respuesta: 200 OK

Error: 500 Internal Server Error

**POST /api/edades/**

Respuesta: 201 Created

Error: 500 Internal Server Error

**PUT /api/edades/:id**

Respuesta: 200 OK

Error: 404 Not Found

Error: 500 Internal Server Error

**PATCH /api/edades/:id**

Respuesta: 200 OK

Error: 404 Not Found

Error: 500 Internal Server Error

**DELETE /api/edades/:id**

Respuesta: 204 No Content

Error: 404 Not Found

Error: 500 Internal Server Error

**GET /api/categorias/**

Respuesta: 200 OK

Error: 500 Internal Server Error

**POST /api/ categoria /**

Respuesta: 201 Created

Error: 404 Not Found

Error: 500 Internal Server Error

**PUT /api/ categoria /:id**

Respuesta: 200 OK

Error: 404 Not Found

Error: 500 Internal Server Error

**PATCH /api/ categorias/:id**

Respuesta: 200 OK

Error: 404 Not Found

Error: 500 Internal Server Error

**DELETE /api/ categoria /:id**

Respuesta: 204 No Content

Error: 404 Not Found

Error: 500 Internal Server Error



## ·Odoo (Vídeos - Puerto 3000)

### Sistema Usuarios:

#### **GET /api/usuarios/**

Respuesta: 200 OK

Error: 500 Internal Server Error

#### **GET /api/usuarios/:id**

Respuesta: 200 OK

Error: 404 Not Found

Error: 500 Internal Server Error

#### **POST /api/usuario/**

Respuesta: 201 Created

Error: 500 Internal Server Error

#### **PUT /api/usuario/:id**

Respuesta: 200 OK

Error: 404 Not Found

Error: 500 Internal Server Error

#### **DELETE /api/usuario/:id**

Respuesta: 204 No Content

Error: 404 Not Found

Error: 500 Internal Server Error

### Sistema Suscripciones:

#### **GET /api/suscripciones/**

Respuesta: 200 OK

Error: 500 Internal Server Error

**GET /api/ suscripciones/:id**

Respuesta: 200 OK

Error: 404 Not Found

Error: 500 Internal Server Error

**POST /api/suscripcion/**

Respuesta: 201 Created

Error: 500 Internal Server Error

**PUT /api/suscripcion/:id**

Respuesta: 200 OK

Error: 404 Not Found

Error: 500 Internal Server Error

**DELETE /api/suscripcion/:id**

Respuesta: 204 No Content

Error: 404 Not Found

Error: 500 Internal Server Error

**Sistema Métodos de Pago:**

**GET /api/metodoPagamiento/**

Respuesta: 200 OK

Error: 500 Internal Server Error

**GET /api/ metodoPagamiento/:id**

Respuesta: 200 OK

Error: 404 Not Found

Error: 500 Internal Server Error

**POST /api/metodoPagamiento/**

Respuesta: 201 Created

Error: 500 Internal Server Error

**PUT /api/metodoPagamiento/:id**

Respuesta: 200 OK

Error: 404 Not Found

Error: 500 Internal Server Error

**DELETE /api/metodoPagamiento/:id**

Respuesta: 204 No Content

Error: 404 Not Found

Error: 500 Internal Server Error