

**CFGS
DESARROLLO DE
APLICACIONES WEB**

PROYECTO FIN DE CICLO



FAT DOWN
NO HAY EXCUSAS

**GRUPO
DW2A
TUTOR
MARCOS PUENTE**

**DANIEL ESTEBAN UCLÉS
ALEJANDRO RELAÑO TORTAJADA
AMANDA SEGURA GONZÁLEZ**

FAT DOWN

ÍNDICE

- 3 EL PROYECTO**
- 4 ESTRUCTURA DEL PROYECTO**
- 4 BACK END**
- 7 FRONT END**
- 8 DIAGRAMA DE CLASES**
- 9 TECNOLOGÍAS UTILIZADAS**
- 11 GUÍA DE ESTILOS**
- 13 ANEXOS**

EL PROYECTO **FAT DOWN**

La aplicación web de fitness Fat Down surge como respuesta a un mundo en constante cambio al que debemos adaptarnos. La pandemia causada por el COVID-19 nos ha hecho reflexionar sobre la verdadera importancia de mantenerse activo no solo como el medio para mejorar nuestra salud física, sino también mental.

ELIGE CREA PERSONALIZA ¡Y A SUDAR!

Fat Down es una propuesta diferente, con consejos saludables para todo el mundo y con vídeos de rutinas para un entrenamiento completo, así como con ejercicios especialmente pensados y descritos por entrenadores que el usuario podrá añadir como desee a sus rutinas personalizadas.



Durante los momentos más duros del confinamiento, nuestros hogares se convirtieron en nuestro lugar de trabajo, nuestra clase, nuestro gimnasio... En definitiva, nuestro refugio.

Muchos se dedicaron a perfeccionar su destreza repostera, preparando cada día un pastel diferente, o también a aprender nuevas habilidades, como tocar el ukelele.

Otros, en cambio, encontraron en el ejercicio físico una vía de escape ideal para sobrellevar el día a día y, por qué no, mejorar su fuerza y resistencia. Fue entonces cuando las aplicaciones de ejercicios y vídeos de entrenamiento se convirtieron en la mejor guía para conseguir nuestros objetivos.

ESTRUCTURA DEL PROYECTO

BACK END

En el esquema mostrado a continuación se observan las capas que componen la estructura del proyecto en el lado del servidor, en lo referente al desarrollo del back end:



Entidades: Representan los objetos que se utilizarán en el proyecto y serán almacenados en la BBDD en forma de tablas

-  com.fatdown.spring.entidades
 -  Consejo.java
 -  Ejercicio.java
 -  Gif.java
 -  Imagen.java
 -  Multimedia.java
 -  Rol.java
 -  Rutina.java
 -  Usuario.java
 -  Video.java
-  com.fatdown.spring.enums
 -  Categoria.java

Repositories: Suponen la capa de abstracción que permite separar el acceso a datos a la BBDD de la implementación de dicho acceso para cada entidad.

- ✓  com.fatdown.spring.repositories
 - >  ConsejoRepository.java
 - >  EjercicioPageableRepository.java
 - >  EjercicioRepository.java
 - >  GifRepository.java
 - >  ImagenRepository.java
 - >  MultimediaRepository.java
 - >  RolRepository.java
 - >  RutinaRepository.java
 - >  UsuarioRepository.java
 - >  VideoPageableRepository.java
 - >  VideoRepository.java

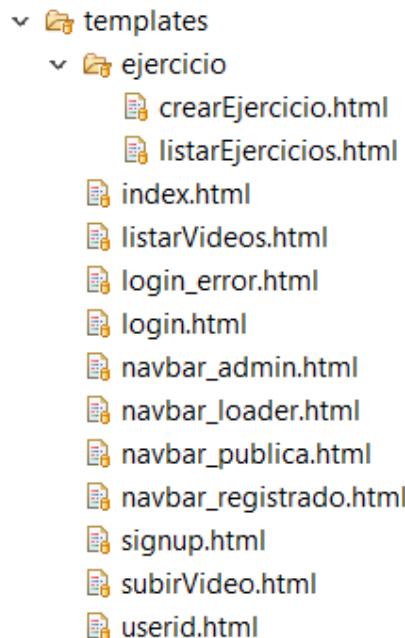
Servicios: La función de los servicios es implementar la lógica de negocios del modelo de dominio, realizando llamadas a los repositories.

- ✓  com.fatdown.spring.servicios
 - >  ConsejoServicio.java
 - >  ConsejoServicioImpl.java
 - >  CustomUserDetailsService.java
 - >  EjercicioServicio.java
 - >  EjercicioServicioImpl.java
 - >  GifServicio.java
 - >  GifServicioImpl.java
 - >  ImagenServicio.java
 - >  ImagenServicioImpl.java
 - >  RutinaServicio.java
 - >  RutinaServicioImpl.java
 - >  UsuarioServicio.java
 - >  UsuarioServicioImpl.java
 - >  VideoServicio.java
 - >  VideoServicioImpl.java

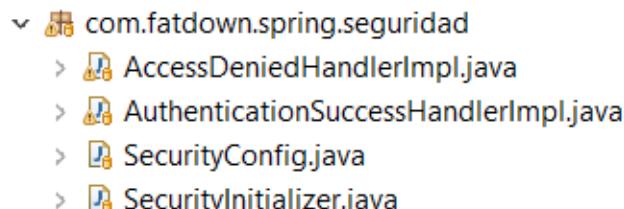
Controladores: Componen la capa que gestiona los datos de las entidades realizando llamadas a sus respectivos servicios, direccionando a los métodos y templates a los que se quiere acceder.

- ✓  com.fatdown.spring.controladores
 - >  EjercicioControlador.java
 - >  IndexControlador.java
 - >  MultimediaControlador.java
 - >  RutinaControlador.java
 - >  UsuarioControlador.java

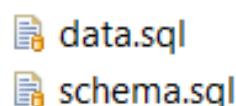
Templates: Conforman la capa en la que se volcarán los datos gestionados en los controladores, que se mostrarán por pantalla al usuario, de modo que pueda interaccionar con las distintas funcionalidades del proyecto.



Seguridad: En esta parte se contienen las clases que permiten al proyecto validar y restringir los distintos niveles de acceso a los endpoints del proyecto en base a roles, así como también regula la creación de usuarios.

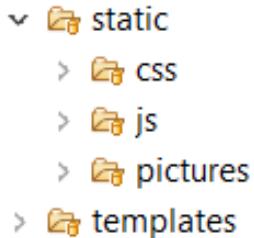


BBDD: También se han introducido ficheros .sql para generar las tablas necesarias en la base de datos de nuestro proyecto (schema.sql) así como incluir algunos registros previos de usuarios y vídeos que servirán como ejemplo (data.sql).



FRONT END

En la imagen mostrada a continuación se observan los componentes de la estructura del proyecto en el lado del cliente, en lo referente al desarrollo del front end:



Con el fin de mejorar la experiencia del usuario a la hora de navegar por las distintas vistas y páginas de la aplicación web, se han integrado entre las carpetas del proyecto hojas de estilo CSS así como JS para implementar el diseño de Bootstrap elegido al cual haremos referencia en el apartado reservado a la guía de estilos.

También se han introducido otros elementos estáticos, como imágenes que han sido utilizadas en las diferentes vistas y páginas de la aplicación web, tales como el logotipo de la marca o las divertidas imágenes que acompañan a los consejos saludables, así como las infografías motivacionales.

DIAGRAMA DE CLASES

El diseño del diagrama de clases entidad-relación que se ha implementado para este proyecto (imagen disponible en la sección de "Anexos" al final de este documento referenciado como Anexo 3) se realizó en una fase previa de diseño previa a la fase de desarrollo y codificación de la aplicación web Fat Down, con el propósito de dotar de sentido al proyecto de software así como de una estructura básica en todo caso con potencial de crecimiento.

Esto se aprecia en la ya mencionada estructura de capas abstractas siguiendo el patrón de arquitectura Modelo-Vista-Controlador (MVC), lo cual ha permitido desarrollar la aplicación rápidamente, de forma modular, sostenible e incremental.

Por otro lado, también se puede destacar del diagrama el uso de las clases abstractas y la herencia en Java en clases como "Multimedia", "Video" o "Gif" con la intención de proporcionar a las diversas funcionalidades y entidades del proyecto de flexibilidad y adaptabilidad que permita, en un futuro, un desarrollo más avanzado en base a nuevas necesidades.

Entre las relaciones utilizadas entre las entidades se puede observar una variedad de tipos (OneToOne, OneToMany, ManyToMany, de agregación, de composición, etc) que ha afectado directamente a la forma de mapear dichas entidades y sus respectivas tablas en la BBDD, utilizando Hibernate como herramienta de mapeo objeto-relacional.

TECNOLOGÍAS UTILIZADAS

Las tecnologías utilizadas para este proyecto han sido principalmente:

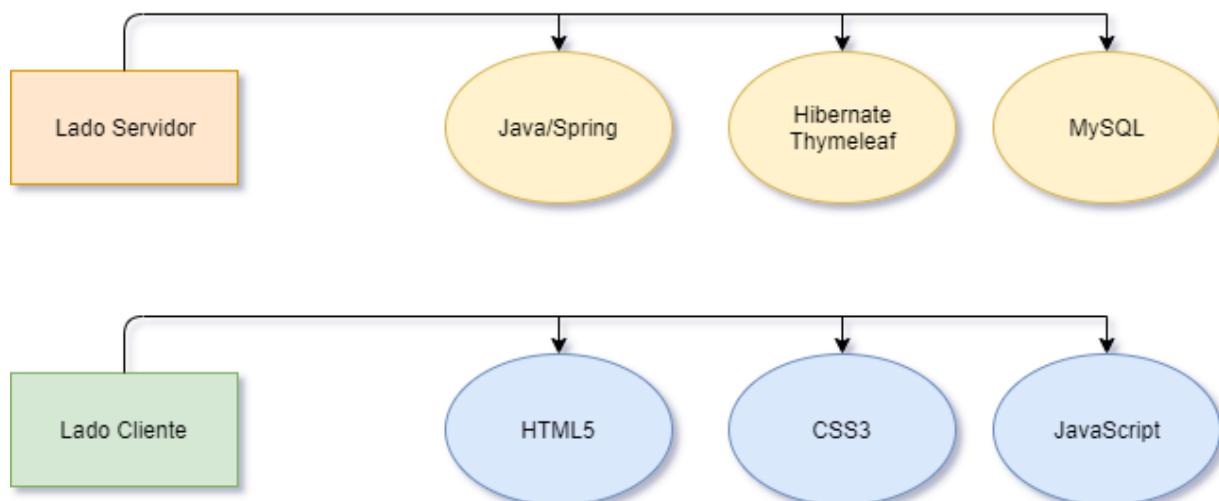
BACK END

- Java como lenguaje de programación en el lado servidor
- Spring boot como framework de desarrollo
- Hibernate como herramienta de mapeo objeto-relacional
- Thymeleaf como motor de plantillas (templates)
- Maven como herramienta de inyección de dependencias y gestión de librerías
- Git como herramienta de control de versiones
- Base de datos MySQL, uno de los SGBD Open Source más populares y utilizados en la actualidad, lo que permite una gran portabilidad y soporte del desarrollo de software realizado en este proyecto

FRONT END

- HTML5 como lenguaje de marcas
- CSS3 para hojas de estilo
- BootStrap como framework CSS que permite diseñar páginas HTML responsive
- JavaScript vanilla como lenguaje de programación en el lado cliente
- JQuery
- JavaScript Asíncrono con XML (AJAX)

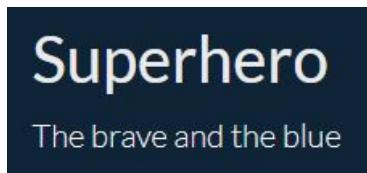
En el siguiente esquema, se puede apreciar más simplificadamente las tecnologías empleadas en el desarrollo de la aplicación web Fat Down.



GUÍA DE ESTILOS

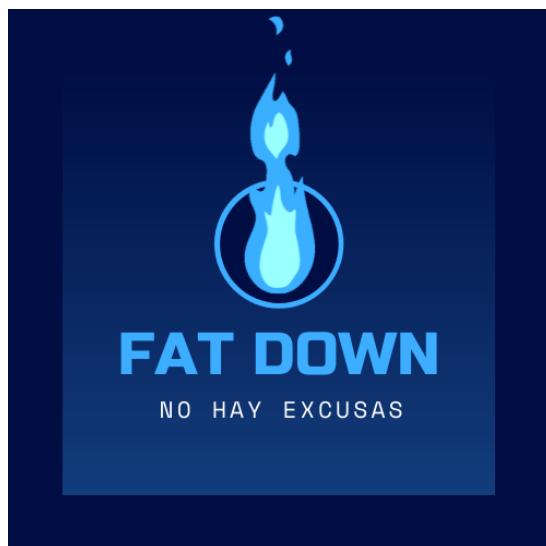
BOOTSTRAP

Los estilos de este proyecto se han basado en el tema de Bootstrap/Bootswatch "Superhero", que se puede acceder desde el siguiente enlace: [aquí](https://bootswatch.com/superhero/) (<https://bootswatch.com/superhero/>).



La decisión se decantó por este tema debido a la naturaleza del proyecto, una web sobre fitness, tratando de valorar la experiencia del usuario y buscando llamar la atención del mismo, empleando tonos que suelen encontrarse en otras webs o aplicaciones de fitness similares.

LOGO



El logo creado, además de perseguir la homogeneidad con el estilo seleccionado anteriormente, contiene una llama azul que simboliza la energía que infusionamos en la actividad física, en el que se incluye el nombre de la marca y aplicación web homónima "Fat Down", junto al lema "No hay excusas" que acompaña al conjunto, en un intento de animar al usuario a sacar fuerzas de flaqueza y conseguir sus objetivos fitness.

COLORES

#000e44

≈ Stratos

#000080

Navy Blue

#3dadff

≈ Dodger Blue

#Ff9900

≈ Orange Peel

#Ffcc99

Peach Orange

Los colores empleados han sido cuidadosamente seleccionados siguiendo el patrón empleado en el tema "Superhero" de Bootstrap, tratando de mantener coherencia y homogeneidad en todas las vistas y páginas de la aplicación web.

Se han utilizado tanto colores análogos (azules) para crear transiciones suaves como complementarios para crear contraste y resaltar información (naranjas).

ANEXOS

- **ANEXO 1**

REPOSITORIO GITHUB

HTTPS://GITHUB.COM/ALEXRET/FATDOWN.GIT

- **ANEXO 2**

ENLACES A VÍDEOS DE DEMOSTRACIÓN

- I. DEMO USUARIO ADMINISTRADOR**

AQUÍ

HTTPS://YOUTU.BE/CVSJGX1GJZW

- II. DEMO USUARIO REGISTRADO**

AQUÍ

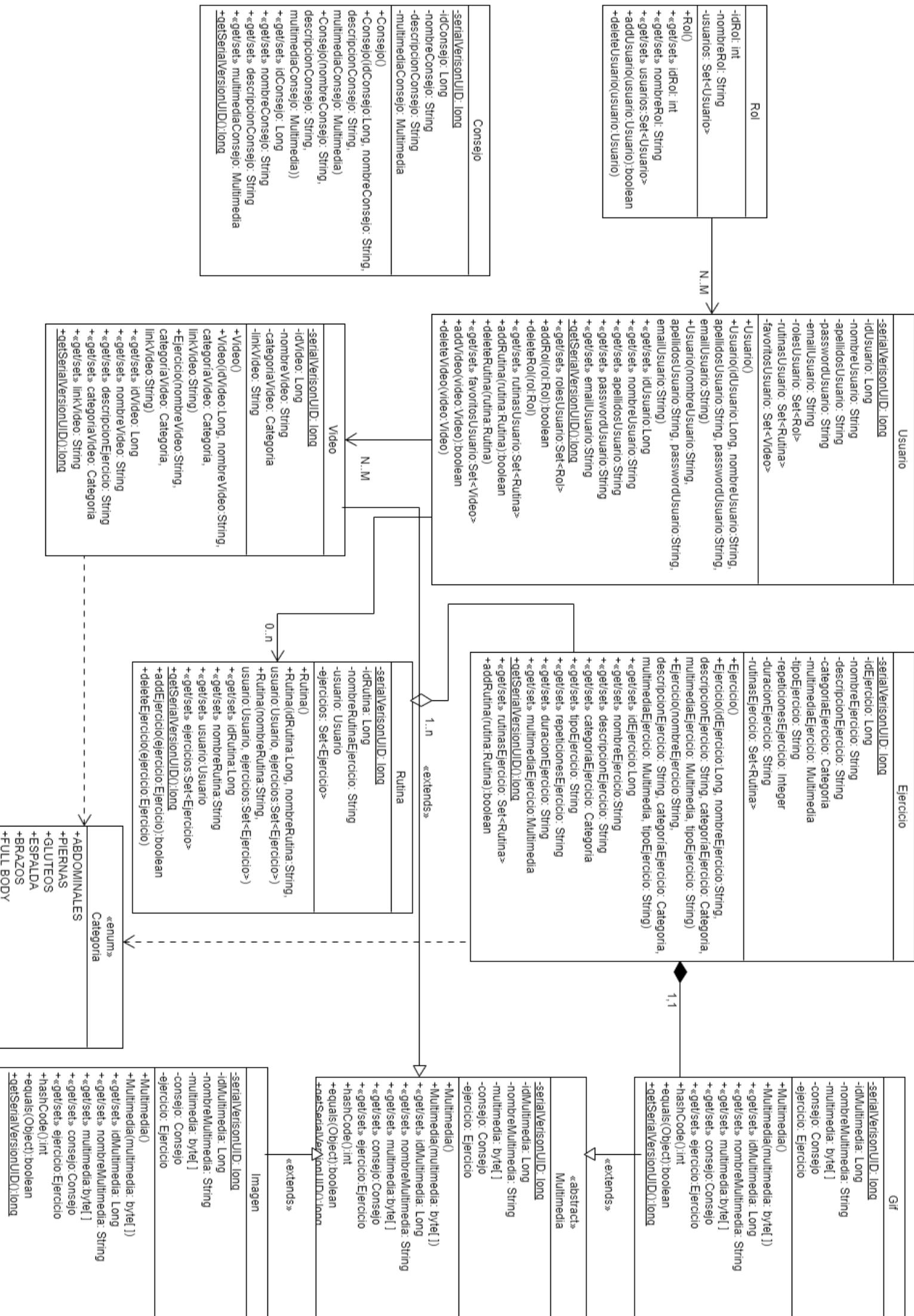
HTTPS://YOUTU.BE/YG0Z8NRIABU

- **ANEXO 3**

DIAGRAMA DE CLASES E/R

- **ANEXO 4**

INFOGRAFÍAS MOTIVACIONALES



MISS 5 ASPIRACIONES FITNESS

¡MOTÍVATE Y CONSIGUE
TUS METAS CON FAT
DOWN!



1. Contenido más motivador



2. Mi meme de Homer Simpson favorito



3. Mejor video para entrenar



4. Rutina de entrenamiento ideal



5. Inspiración para entrenar



MI DESAFÍO FAT DOWN

FAT DOWN

NO HAY EXCUSAS

HOY HE HECHO...

- 20 SENTADILLAS
- 30' PLANCHA ISOMÉTRICA
- 15 BURPEES
- 25' SUPERMAN
- 20 PATADA TRASERA
- 30' CURL DE BÍCEPS
ISOMÉTRICO



FAT DOWN

NO HAY EXCUSAS

