

NOMBRE DE LA APLICACIÓN

CFGS DAW

CURSO 2020/2021

NOMBRE/S DEL/DE LOS ALUMNO/S

Santiago Rodríguez y Rodrigo Pascual

COLEGIO CALASANZ SALAMANCA

1. Introducción	3
2. Descripción de la aplicación	4
3. Tecnologías escogidas y justificación	5
4. Diseño de la aplicación	6
5. Arquitectura de la aplicación	7
5.1. Estructura del proyecto	7
6. Manual de despliegue	8

1. Introducción

El proyecto del fin de grado superior lo vamos a realizar de forma conjunta los alumnos: Santiago Rodríguez Ortiz de Zárate y Rodrigo Pascual Montero. La temática del proyecto consiste en el desarrollo de una aplicación web de un gimnasio en la cual plasmaremos los conocimientos adquiridos en estos dos años de curso. El nombre de nuestra aplicación será "InfoGym", la cual servirá para la gestión de un gimnasio.

Nuestra aplicación es muy intuitiva y con ella el usuario del gimnasio podrá hacer varias tareas para hacer más fácil su día a día en nuestras instalaciones.

Para poder para poder empezar a usar la aplicación el Usuario deberá de hacer un registro completando un pequeño formulario, una vez finalizado se usará el Usuario y la contraseña escogidas para entrar en la aplicación .

2. Descripción de la aplicación

Es una aplicación que en primer lugar nos aparece una ventana con el login y el registro, en el caso de que el usuario este en la base de datos poniendo el usuario y la contraseña valdría para acceder, en caso contrario el usuario tendría que hacer un registro. En este registro le aparecerá un formulario que tendrá que rellenar donde tendrá que poner el nombre, los apellidos, el email, el Nick o nombre de usuario y la contraseña, dando a continuar volvemos al login en donde poniendo el Nick y la contraseña accedemos a la primera vista donde la aplicación nos da la bienvenida con 4 opciones:

-ir a la página principal

Aquí en la página principal podemos ver una serie de videos de diferente actividades de los grupos musculares que se pueden ejercitar, así como una serie de botones que pueden redirigirte a las otras pestañas donde poder funcionar como la página del inicio o las siguientes a comentar.

-ver las actividades

Aquí veremos una tabla con las diferentes actividades

-ver tu perfil

Aquí nos muestra nuestro perfil en la aplicación, si le damos a modificar usuario nos sale el formulario del principio para poder modificar alguna cosa que queramos guardándose en la base de datos con la modificación realizada

-salir

La aplicación te da las gracias por su visita y te redirige a la página principal de la aplicación.



3. Tecnologías escogidas y justificación

Las tecnologías escogidas tras mucha meditación ha sido la utilización del lenguaje de Java (Java es un lenguaje de programación y una plataforma informática comercializada por primera vez en 1995 por Sun Microsystems. Hay muchas aplicaciones y sitios web que no funcionarán a menos que tenga Java instalado y cada día se crean más. Java es rápido, seguro y fiable. Desde portátiles hasta centros de datos, desde consolas para juegos hasta súper computadoras, desde teléfonos móviles hasta Internet, Java está en todas partes.) ya que ha sido el lenguaje más utilizado estos dos últimos años y nos da la facilidad para poder hacer una aplicación de calidad, para ello hemos utilizado un Framework, esto nos ha facilitado aun más el trabajo , dicho Framework es Spring.

Spring facilita la creación de aplicaciones independientes de grado de producción que pueden simplemente ejecutarse, tomando una visión concreta de la plataforma Spring y las bibliotecas de terceros para que pueda comenzar con el mínimo esfuerzo

Para las vistas se ha utilizado el lenguaje de html el cual es

“HTML significa **lenguaje de marcado de hipertexto**, y le permite al usuario crear y estructurar secciones, párrafos, encabezados, enlaces y elementos de cita en bloque (blockquotes) para páginas web y aplicaciones.

HTML no es un lenguaje de programación, lo que significa que no tiene la capacidad de crear una funcionalidad dinámica. En cambio, hace posible organizar y formatear documentos, de manera similar a Microsoft Word”

4. Diseño de la aplicación

La aplicación está diseñada para que el cliente interactúe y esté unido a una base de datos la cual es la encargada de almacenar todos los datos de los mismo .

Esta base de datos está formada por cuatro tablas, dichas tablas son :



Usuarios/ Actividades/Beneficios /Instalaciones a continuación veremos como están formadas:

Actividades -----

```
`id` int NOT NULL,  
`nombre` varchar(45) NOT NULL,
```

Usuarios-----

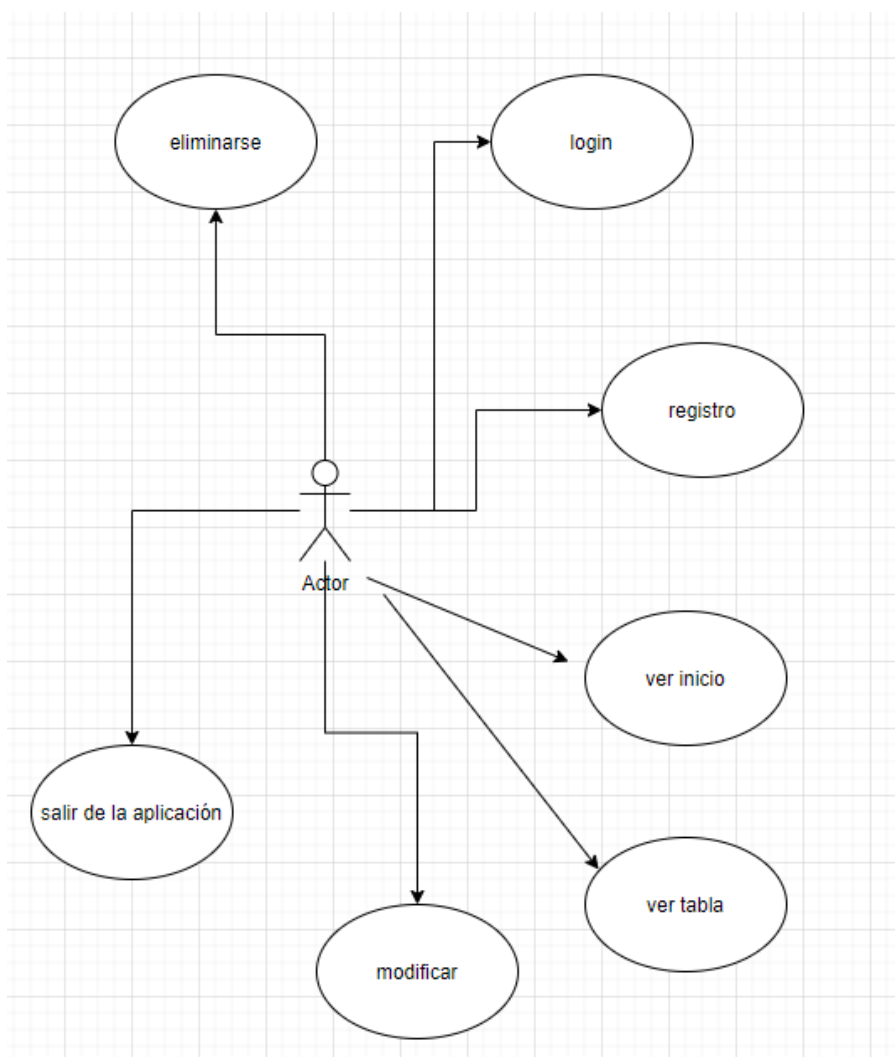
```
PRIMARY KEY (`id`) `id` bigint NOT NULL AUTO_INCREMENT,  
`apellido` varchar(255) DEFAULT NULL,  
`email` varchar(255) DEFAULT NULL,  
`nombre` varchar(255) DEFAULT NULL,  
`password` varchar(255) DEFAULT NULL  
`username` varchar(255) DEFAULT NULL,
```

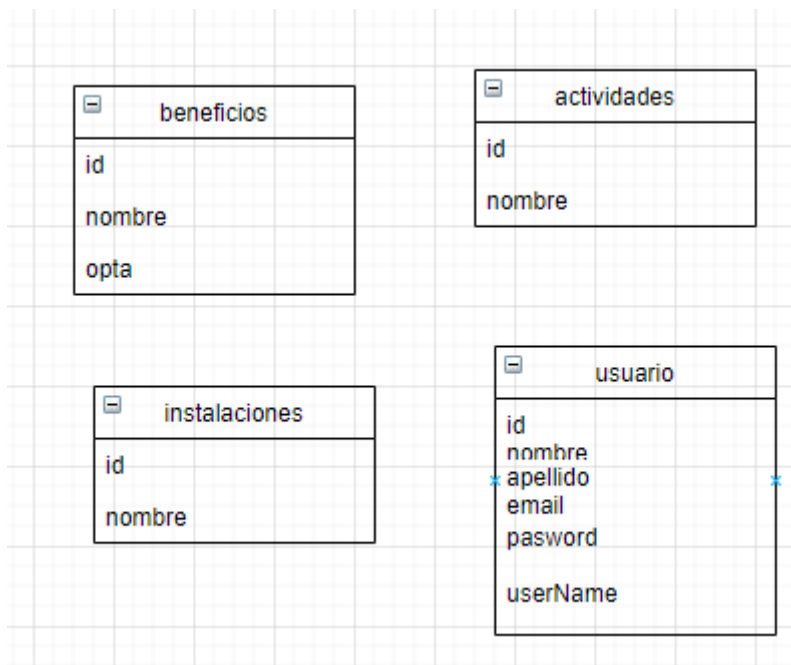
Beneficios-----

```
`id` int NOT NULL,  
`nombre` varchar(45) NOT NULL,  
`opta` tinyint(1) NOT NULL,  
PRIMARY KEY (`id`)
```

Instalaciones-----

```
`id` int NOT NULL,  
`nombre` varchar(45) NOT NULL,  
PRIMARY KEY (`id`)
```







Arquitectura de la aplicación

4.1. Estructura del proyecto

El proyecto está dividido en diferentes carpetas y capas , siempre utilizando el modelo vista controlador donde cada una tenga independencia una de la otra.

El proyecto tiene diferentes paquetes, estos paquetes se dividen en :

Demo : obligatorio para Spring

Controller: aquí serán los controladores de la pagina

Dao

Entity: donde están métodos getter y setter

Implementes: métodos de funcionalidad de la aplicación

Repository: conexión base de datos

Security y service.



- ▼ > LoginRegistro [boot] [devtools] [GimnasioTFG master]
 - ▼ src/main/java
 - ▼ com.example.demo
 - > LoginRegistroApplication.java
 - ▼ com.example.demo.models.controllers
 - > ActividadesController.java
 - > InicioController.java
 - > LoginController.java
 - > PerfilController.java
 - > PrincipalController.java
 - > PrivateController.java
 - > PublicController.java
 - ▼ com.example.demo.models.dao
 - > IUserDAO.java
 - ▼ com.example.demo.models.entity
 - > Usuario.java
 - ▼ com.example.demo.models.impl
 - > UsuarioServiceImpl.java
 - ▼ com.example.demo.models.repository
 - > IUserRepository.java
 - ▼ com.example.demo.models.security
 - > SecurityConfig.java
 - > UserDetailsServiceImpl.java
 - ▼ com.example.demo.models.service
 - > IUserService.java
 - ▼ src/main/resources
 - > templates
 - application.properties
 - > src/test/java
 - > JRE System Library [JavaSE-11]
 - > Maven Dependencies
 - ▼ > bin
 - ▼ > src
 - ▼ > main
 - > > java
 - ▼ > resources
 - ▼ > templates
 - > > layouts
 - actividades.html
 - index.html
 - inicio.html
 - login.html
 - logout.html
 - perfil.html
 - principal.html
 - register.html



5. Manual de despliegue

Para el despliegue hemos elegido la opción de Docker. Para hacer el despliegue dentro de la carpeta GimnasioTFG tenemos que hacer un git bash. En el git bash pondremos el comando

```
docker network create gimnasio_tfg
```

para crear la red con la que trabajaremos. A continuación pondremos el comando

```
docker build -t rodrigo313/imagenbbdd .
```

para crear la imagen de la bbdd. Una vez hecho esto utilizamos el comando

```
docker run -d --network gimnasio_tfg --name mi_mysql -e  
MYSQL_ROOT_PASSWORD=Rodrigo313 -e MYSQL_DATABASE=gimnasio -e  
MYSQL_PASSWORD=Rodrigo313 -p 3307:3306 rodrigo313/imagenbbdd
```

para crear el contenedor de la bbdd. Una vez creado la imagen y contenedor de la bbdd creamos el del backend, que sería haciendo un git bash dentro de la carpeta LoginRegistro y pondremos primero el comando

```
docker build -t rodrigo313/imagenback .
```

para crear la imagen del backend. A continuación ponemos el comando

```
docker run --network gimnasio_tfg --name gimnasio_back -p 8080:8080  
rodrigo313/imagenback
```

para crear el contenedor del backend y arrancar la aplicación. Una vez hecho esto solo tendremos que irnos a google y poner

<http://localhost:8080/gimnasio/> y ya estaríamos en la aplicación.



colegio
CALASANZ
SALAMANCA

Paseo de Canalejas, 139-159
37001 Salamanca
Tel.: 923 26 79 61 | Fax: 923 27 04 54
www.calasanzsalamanca.es
CENTRO PRIVADO CONCERTADO