



# GymBuddy

David Castro Portero

Desarrollo de  
Aplicaciones Web

2023-2024

# GYMBUDDY



# ÍNDICE

1	Introducción	3
2	Identificación de las necesidades	4
3	Análisis del mercado	5
4	Justificación del proyecto	6
5	Stack tecnológico	7
6	Modelo de Datos	9
7	Prototipo de la Aplicación Web	12
8	Definición API REST	18
9	Manual de Despliegue	19
9.1	Pasos Previos	19
9.2	Variables de entorno	19
9.3	Base de Datos	20
9.4	Configuración Springboot	21
9.5	Configuración Angular	22
9.6	Despliegue AWS	23
9.7	Conexión con la Base de Datos	27
9.8	Despliegue Aplicación Angular	29
9.9	Error Con Cors	31
10	Conclusión y Postmortem	33



## Introducción

GymBuddy conecta la palabra 'Gym' de gimnasio y 'Buddy' de la famosa frase del culturista Ronnie Coleman "Yeah Buddy" haciendo la fusión de estas dos palabras un título con gancho y doble sentido.

GymBuddy es una aplicación en la cual ves los mejores ejercicios y entrenamientos recomendados por especialistas en el mundo del deporte. Se busca brindar a los usuarios una experiencia única y efectiva que les permita alcanzar sus metas de acondicionamiento físico de manera eficiente.

La combinación de entrenamientos personalizados, la colaboración con profesionales certificados y las herramientas de seguimiento hacen que la plataforma sea única en el mercado, brindando a los usuarios una experiencia completa y efectiva para su etapa de acondicionamiento físico.

Mi aplicación está diseñada para aquellos que buscan un enfoque profesional y personalizado para su entrenamiento físico. Se dirige a personas de todas las edades y niveles de condición física que valoran la orientación experta y desean alcanzar sus metas de manera efectiva y segura.



# Identificación de las necesidades

Listado de requisitos funcionales:

1. Registrarse
2. Iniciar Sesión
3. Obtener los Ejercicios con sus atributos.
4. Obtener los Entrenamientos con sus atributos.
5. Obtener los Usuarios con sus atributos por parte del administrador.
6. Añadir, Editar y Eliminar Usuarios por parte del administrador.
7. Añadir y Eliminar Ejercicios por parte del administrador y del entrenador.
8. Añadir y Eliminar Entrenamientos por parte del administrador y del entrenador.
9. Apuntarse y desapuntarse a un entrenamiento por parte del usuario registrado.
10. Ver los entrenamientos a los que un usuario está apuntado.



## Análisis del mercado

Comparativa de herramientas similares, puntos fuertes y débiles de la aplicación web que se quiere desarrollar respecto a las existentes.

- [Entrenamientos.com](https://entrenamientos.com): Aplicación menos centrada en ejercicios y entrenamientos del día a día en el gimnasio, si no que más centrada en personas mayores, y en problemas de salud.



## Justificación del proyecto

Desde hace tiempo, voy al gimnasio y entreno, y siempre he hecho la rutina de ejercicios yo por mi propia cuenta, sin apenas conocimiento al principio.

Y a parte, no he contratado ningún entrenador personal debido al gasto que eso supone, por ello he decidido crear una aplicación donde tengas múltiples opciones de entrenamientos a tu disposición y medida.

Mi decisión surge tras una exhaustiva investigación de mercado, donde he observado la existencia de diversas aplicaciones dedicadas a rutinas de entrenamiento. Sin embargo, mi propuesta se distingue por la incorporación de expertos altamente cualificados en el ámbito del fitness. Este enfoque diferenciador garantiza una oferta de calidad superior, respaldada por el conocimiento y la experiencia de profesionales reconocidos en la industria del acondicionamiento físico.

El compromiso con la excelencia y la calidad del servicio es el punto importante de esta aplicación. Proporcionaré a los usuarios una experiencia única y enriquecedora, donde cada rutina de entrenamiento esté cuidadosamente diseñada para maximizar los resultados y la seguridad.

En resumen, quiero crear una aplicación que cambie las aplicaciones convencionales de entrenamiento, elevando los estándares de calidad y ofreciendo a los usuarios una experiencia enriquecedora en su día a día hacia un estilo de vida más saludable y activo.



# Stack Tecnológico

## Desarrollo Back-End:

- **Tecnología:** Spring Boot
- **Justificación:** Spring Boot es un framework robusto y ampliamente utilizado en el desarrollo de aplicaciones Java. Facilita la creación de aplicaciones empresariales y APIs RESTful gracias a su configuración simplificada y su amplia gama de características integradas. Su ecosistema es maduro y bien documentado, lo que asegura soporte y recursos para resolver cualquier inconveniente. Además, la comunidad de Spring Boot es muy activa, lo que proporciona una fuente constante de nuevas ideas y soluciones.

## Desarrollo Front-End:

- **Tecnología:** Angular
- **Justificación:** Angular es un framework de front-end desarrollado y mantenido por Google. Es ideal para construir aplicaciones web dinámicas y de alta calidad. Su arquitectura basada en componentes facilita la reutilización de código y el mantenimiento a largo plazo de la aplicación. Angular también ofrece un sólido sistema de gestión de dependencias y una excelente integración con otras herramientas de desarrollo, lo que acelera el proceso de desarrollo y garantiza una mayor eficiencia.

## Base De Datos:

- **Tecnología:** MySQL
- **Justificación:** MySQL es una de las bases de datos relacionales más populares y confiables del mercado. Su capacidad para manejar grandes volúmenes de datos y sus potentes características de seguridad lo hacen ideal para aplicaciones web como GymBuddy. Además, MySQL es altamente escalable y tiene una amplia comunidad de soporte, lo que garantiza que cualquier problema pueda ser rápidamente abordado.



## Diseño:

- **Tecnología:** Tailwind CSS y Figma
- **Tailwind CSS:** Este framework de CSS es conocido por su enfoque utilitario, permitiendo un diseño rápido y eficiente sin la necesidad de escribir CSS personalizado para cada componente. Esto facilita la creación de interfaces modernas y responsivas con un mínimo esfuerzo.
- **Figma:** Figma es una herramienta de diseño colaborativa que permite crear prototipos interactivos y compartir diseños con facilidad. Su capacidad para trabajar en tiempo real con otros diseñadores y desarrolladores hace que el proceso de diseño sea más eficiente y coherente.









## 2. **usuario\_rol:**

**usuario\_id:** Identificador del usuario, clave foránea que referencia a la tabla **usuario**.

**roles\_usuario:** Roles asignados al usuario, posiblemente definido por un conjunto de valores enumerados.

## 3. **usuario\_entrenamiento:**

**entrenamiento\_id:** Identificador del entrenamiento, clave foránea que referencia a la tabla **entrenamiento**.

**usuario\_id:** Identificador del usuario, clave foránea que referencia a la tabla **usuario**.

## 4. **ejercicio:**

**id:** Identificador único del ejercicio.

**descripción:** Descripción del ejercicio.

**imagen\_url:** URL de la imagen asociada al ejercicio.

**nombre:** Nombre del ejercicio.

**creador\_id:** Identificador del usuario que creó el ejercicio, clave foránea que referencia a la tabla **usuario**.

## 5. **entrenamiento:**

**id:** Identificador único del entrenamiento.

**dificultad:** Nivel de dificultad del entrenamiento, definido por un conjunto de valores enumerados.

**imagen\_url:** URL de la imagen asociada al entrenamiento.

**nombre:** Nombre del entrenamiento.

**creador\_id:** Identificador del usuario que creó el entrenamiento, clave foránea que referencia a la tabla **usuario**.



## 6. ejercicio\_entrenamiento:

**entrenamiento\_id:** Identificador del entrenamiento, clave foránea que referencia a la tabla **entrenamiento**.

**ejercicio\_id:** Identificador del ejercicio, clave foránea que referencia a la tabla **ejercicio**.

Las relaciones entre las tablas son las siguientes:

- **usuario** tiene una relación de uno a muchos con **usuario\_rol** a través del campo **usuario\_id**.
- **usuario** tiene una relación de uno a muchos con **ejercicio** a través del campo **creador\_id**.
- **usuario** tiene una relación de uno a muchos con **entrenamiento** a través del campo **creador\_id**.
- **entrenamiento** y **ejercicio** tienen una relación de muchos a muchos a través de la tabla intermedia **ejercicio\_entrenamiento**, donde **entrenamiento\_id** y **ejercicio\_id** son claves foráneas que referencian a **entrenamiento** y **ejercicio** respectivamente.

Este modelo de datos está diseñado para manejar información de usuarios, sus roles, ejercicios creados por ellos, entrenamientos, la relación entre ejercicios y entrenamientos y entre usuarios y ejercicios.

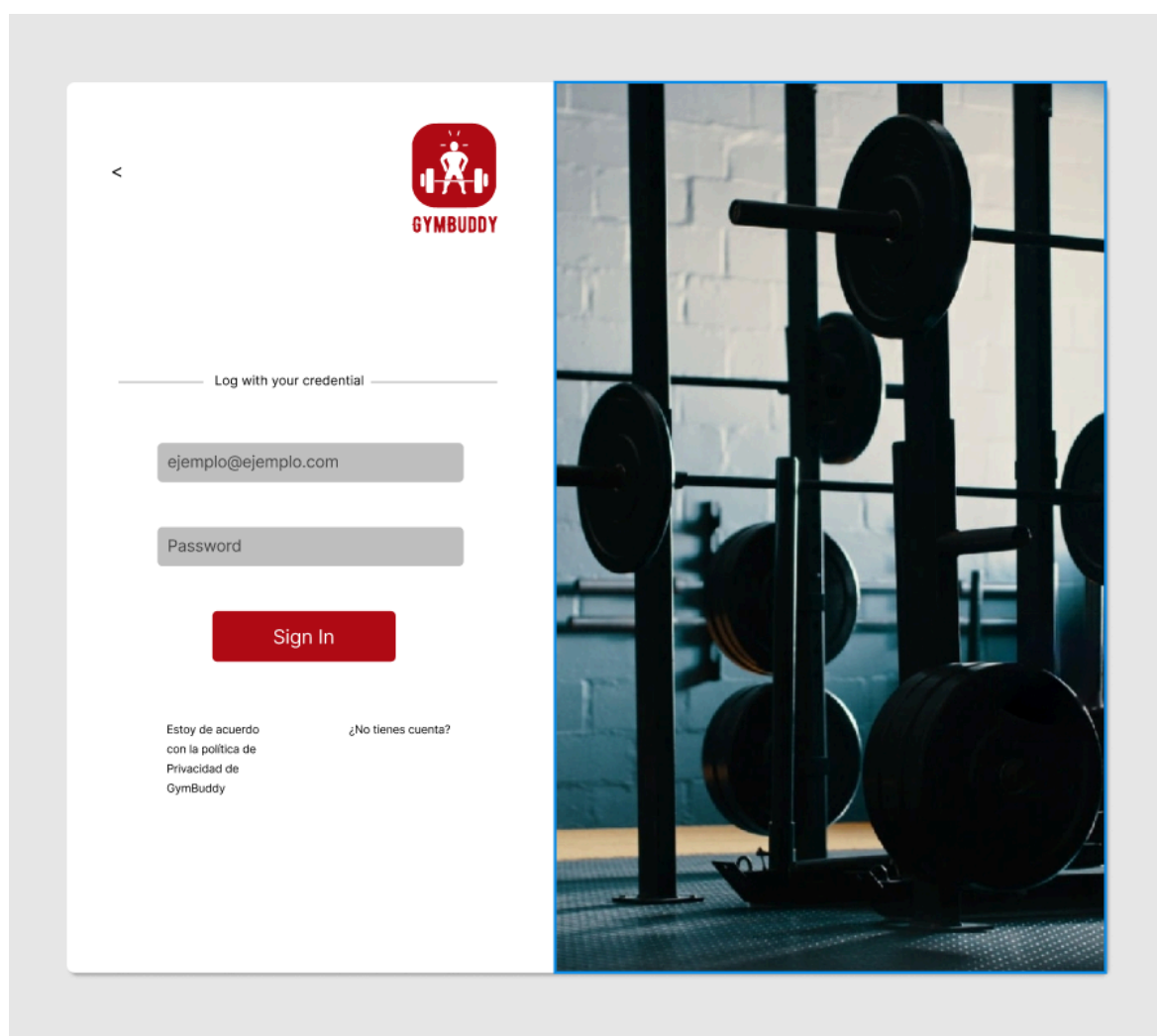


## Prototipo de la Aplicación Web

A continuación se mostrarán algunas de las vistas más importantes diseñadas en Figma. Y en este enlace se encuentra el enlace al proyecto de Figma [Enlace](#).

### Vista de Login:

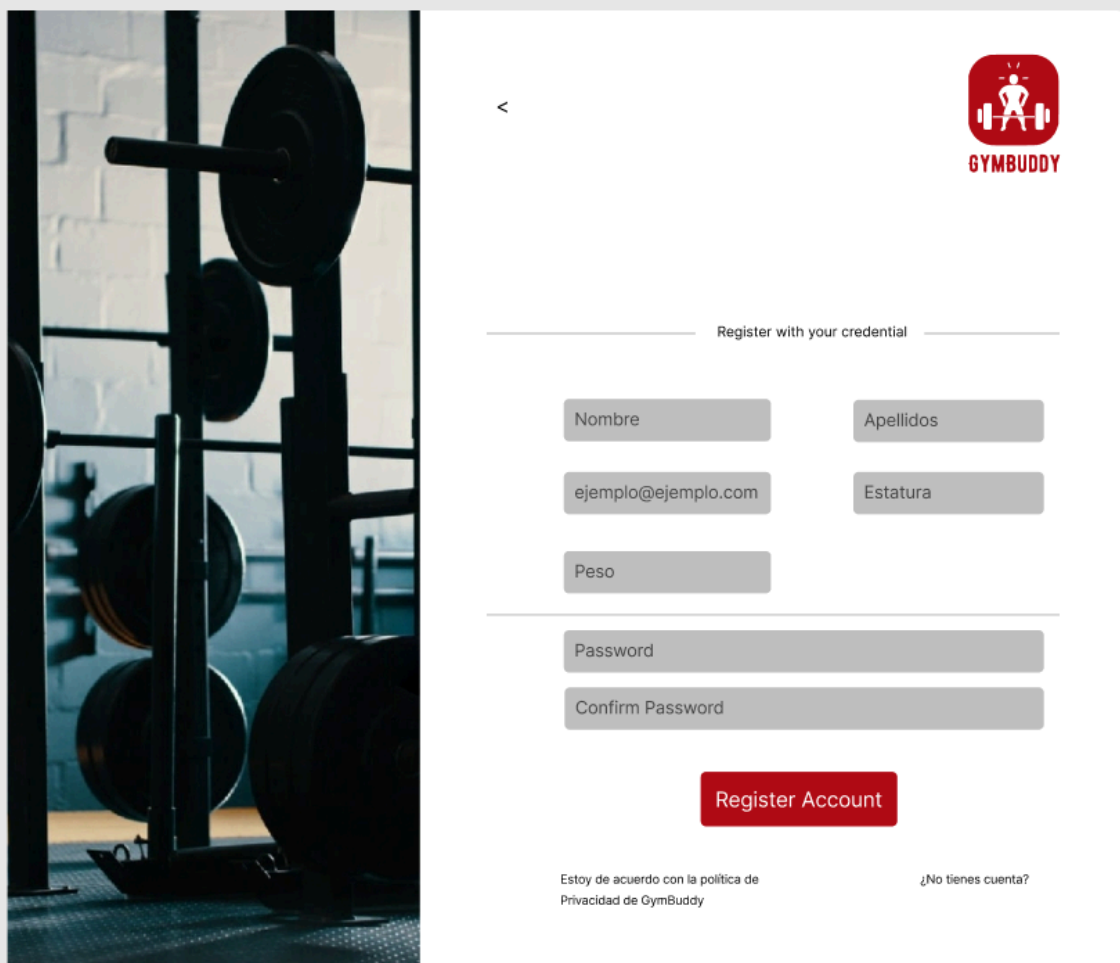
La vista de login de GymBuddy ofrece a los usuarios la posibilidad de acceder de manera segura y rápida a su cuenta personal. Incluye un formulario con campos para ingresar correo electrónico y contraseña, facilitando así el inicio de sesión. Además, presenta un diseño intuitivo y responsivo que asegura una experiencia de usuario eficiente en diferentes dispositivos.





### Vista de Inicio de Sesión:

La vista de registro en GymBuddy permite a nuevos usuarios crear una cuenta personalizada. Incluye campos para nombre, apellido, correo electrónico, peso, estatura, contraseña y confirmación de contraseña. Este formulario asegura que los datos ingresados sean validados y protegidos adecuadamente, proporcionando una experiencia de registro fluida y segura.



The screenshot shows a registration form for 'GYMBUDDY'. The form is white and positioned on the right side of the image, which has a background of a gym with weight racks. The form includes a back arrow icon at the top left, the GYMBUDDY logo at the top right, and a section titled 'Register with your credential'. The fields are: 'Nombre' (Name), 'Apellidos' (Last Name), 'ejemplo@ejemplo.com' (Email), 'Estatura' (Height), 'Peso' (Weight), 'Password', and 'Confirm Password'. A red 'Register Account' button is at the bottom. At the very bottom, there are links for 'Estoy de acuerdo con la política de Privacidad de GymBuddy' and '¿No tienes cuenta?'.

<

GYMBUDDY

Register with your credential

Nombre

Apellidos

ejemplo@ejemplo.com

Estatura

Peso

Password

Confirm Password

Register Account

Estoy de acuerdo con la política de Privacidad de GymBuddy

¿No tienes cuenta?



### Crear Nuevo Ejercicio:

En GymBuddy, la funcionalidad "Añadir Nuevo Ejercicio" permite a los usuarios autorizados introducir información detallada sobre nuevos ejercicios. Esta característica incluye campos para ingresar el nombre del ejercicio, una descripción detallada, el nombre del creador del ejercicio, y la capacidad de seleccionar y adjuntar una imagen representativa. Esto asegura que cada ejercicio esté debidamente documentado y presentado de manera visualmente atractiva para los usuarios.



Usuarios

Login

### Crear Nuevo Ejercicio

Nombre:

Descripción:

Creador:

URL de la Imagen:

Enviar



© 2024 GymBuddy, Inc

© 2024 GymBuddy, Inc. All rights reserved



## Crear Nuevo Entrenamiento:

En GymBuddy, los administradores y entrenadores pueden crear nuevos entrenamientos personalizados. Esta función permite especificar el nombre del entrenamiento, su nivel de dificultad mediante un campo select, seleccionar uno o varios ejercicios disponibles, indicar el creador del entrenamiento, y adjuntar una imagen representativa para una mejor visualización y comprensión del mismo.



Usuarios

Login

### Crear Nuevo Entrenamiento

Nombre:

Escribe el nombre del entrenamiento.

Dificultad:



Ejercicios

Press Banca  
Sentadilla  
Flexiones

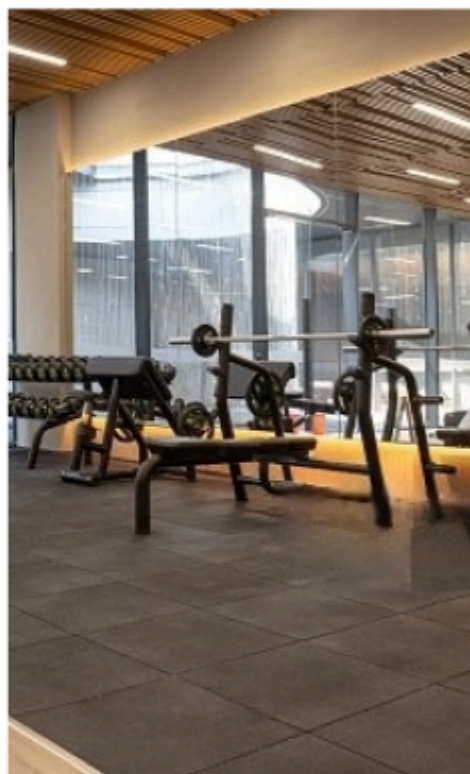
Creador:



URL de la imagen:

Seleccionar archivo

Enviar



© 2024 GymBuddy, Inc


© 2024 GymBuddy, Inc. All rights reserved

## Página Principal:

La página principal de GymBuddy proporciona a los usuarios una vista completa de todos los ejercicios y entrenamientos disponibles en la plataforma. Aquí se pueden explorar y buscar diferentes rutinas de ejercicio y programas de entrenamiento. La página está diseñada para facilitar la navegación intuitiva y el acceso rápido a la información detallada de cada ejercicio y entrenamiento, fomentando así una experiencia eficiente y satisfactoria para los usuarios de GymBuddy.

[Login](#)

## Entrenamientos

<div><b>Entrenamiento De Fuerza</b></div> <div></div> <div>En este entrenamiento, realizas ejercicios como levantamiento de pesas, flexiones o sentadillas, con el objetivo de aumentar la fuerza y la masa muscular.</div> <div>Más Información</div>	<div><b>Entrenamiento De Resistencia</b></div> <div></div> <div>Este entrenamiento implica actividades como correr, plancha o andar en bicicleta durante periodos prolongados a un ritmo moderado o intenso.</div> <div>Más Información</div>	<div><b>Entrenamiento De Potencia</b></div> <div></div> <div>Este entrenamiento se centra en ejercicios explosivos y rápidos, como saltos, lanzamientos de balón medicinal y levantamientos olímpicos.</div> <div>Más Información</div>
--	---	--

## Ejercicios

<div><b>Press Banca Con Barra</b></div> <div></div> <div>El press de banca, en este caso con barra, es un ejercicio de peso libre que trabaja principalmente la zona superior del cuerpo.</div> <div>Más Información</div>	<div><b>Sentadilla Libre</b></div> <div></div> <div>La sentadilla trabaja directamente los músculos del muslo, cadera y glúteos, y fortalece los tendones y ligamentos de piernas y caderas.</div> <div>Más Información</div>	<div><b>Peso Muerto Con Barra</b></div> <div></div> <div>El peso muerto es un ejercicio, en este caso con barra, que consiste en levantarlo desde el suelo hasta la cintura.</div> <div>Más Información</div>
---	--	---




© 2024 GymBuddy, Inc

© 2024 GymBuddy, Inc. All rights reserved



Listado De Usuarios:

En GymBuddy, la funcionalidad de "Listado de Usuarios" permite a los administradores y usuarios autorizados ver, editar y eliminar usuarios registrados en la plataforma. Esta característica proporciona un acceso centralizado a la información de los usuarios, permitiendo realizar acciones como actualizar detalles de perfil, modificar permisos y eliminar cuentas según sea necesario. Esto asegura una gestión eficiente y segura de la base de usuarios de GymBuddy, manteniendo la integridad y seguridad de la plataforma.




Usuarios

Login

Lista de Usuarios

New

ID	EMAIL	NOMBRE	APELLIDOS	ROL	EDIT	DELETE
2	david.castro@gmail.com	David	Castro Portero	ROL_ADMIN	<div>Edit</div>	Can't delete yourself
3	abel.garcia@gmail.com	Abel	Garcia Sanchez	ROL_USER	<div>Edit</div>	<div>Delete</div>
12	test@wwe.es	Test		ROL_USER	<div>Edit</div>	<div>Delete</div>



© 2024 GymBuddy, Inc

© 2024 GymBuddy, Inc. All rights reserved



## Definición API REST

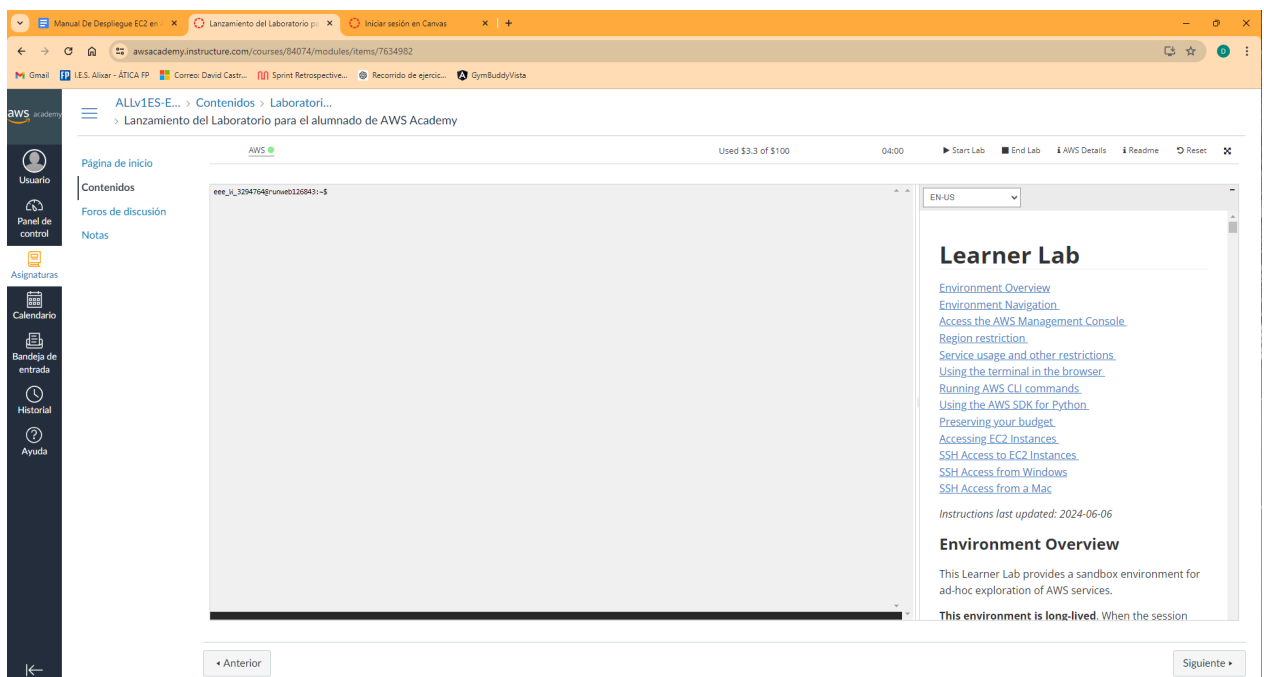
A través de este [enlace](#) accederemos a todos los endpoints de la API con su documentación.



# Manual de Despliegue

## Pasos Previos

Buscamos [AWS Academy](#) e iniciamos sesión. Una vez iniciada sesión, nos aparecerá una pantalla, en ella seleccionamos "AWS Academy Learner Lab [84074]", el cual es el que tenemos disponible en esta ocasión. Seleccionamos Módulos, accedemos al laboratorio y lo iniciamos.



## Variables de entorno

Nos creamos un .env a la altura del proyecto y le añadimos las siguientes variables

```
.env
1 MYSQL_ROOT_PASSWORD=
2 MYSQL_DATABASE=gymbuddy
3 MYSQL_USER=davidGymbuddy
4 MYSQL_PASSWORD=
5 MYSQL_HOST=localhost
6 MYSQL_PORT=3306
7 CORS_ALLOWED_ORIGIN=http://localhost:4200
8 STATIC_LOCATIONS=classpath:/static/
```

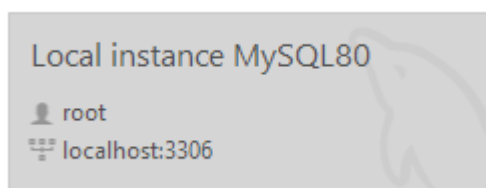


## Base de Datos

Accedemos a la aplicación MySQL Workbench y nos creamos una conexión. Accedemos a ella,

nos creamos un usuario, accedemos a roles administrativos y seleccionamos todas las opciones.

Luego a esto, nos creamos una base de datos con caracteres utf-8



Script-BBDD-GymBuddy\* usuario entrenamiento ejercicio ejercicio\_entrenamiento usuario\_rol entrenamiento usuario\_entrenamiento Administration - Users and Privileges

Local instance MySQL80  
Users and Privileges

User Accounts

User	From Host
davidGymbuddy	%
mysql.infoschema	localhost
mysql.session	localhost
mysql.sys	localhost
root	localhost

Details for account davidGymbuddy@%

Login Account Limits Administrative Roles Schema Privileges

Role	Description
<input checked="" type="checkbox"/> DBA	grants the rights to perform all tasks
<input checked="" type="checkbox"/> MaintenanceAdmin	grants rights needed to maintain server
<input checked="" type="checkbox"/> ProcessAdmin	rights needed to assess, monitor, and kill any user process
<input checked="" type="checkbox"/> UserAdmin	grants rights to create users logins and reset passwords
<input checked="" type="checkbox"/> SecurityAdmin	rights to manage logins and grant and revoke server an...
<input checked="" type="checkbox"/> MonitorAdmin	minimum set of rights needed to monitor server
<input checked="" type="checkbox"/> DBManager	grants full rights on all databases
<input checked="" type="checkbox"/> DBDesigner	rights to create and reverse engineer any database sche...
<input checked="" type="checkbox"/> ReplicationAdmin	rights needed to setup and manage replication
<input checked="" type="checkbox"/> BackupAdmin	minimal rights needed to backup any database
<input checked="" type="checkbox"/> Custom	custom role

Global Privileges

- ☒ ALTER
- ☒ ALTER ROUTINE
- ☒ CREATE
- ☒ CREATE ROUTINE
- ☒ CREATE TABLESPACE
- ☒ CREATE TEMPORARY TABLES
- ☒ CREATE USER
- ☒ CREATE VIEW
- ☒ DELETE
- ☒ DROP
- ☒ EVENT
- ☒ EXECUTE
- ☒ FILE
- ☒ GRANT OPTION
- ☒ INDEX
- ☒ INSERT
- ☒ LOCK TABLES
- ☒ PROCESS
- ☒ REFERENCES
- ☒ RELOAD
- ☒ REPLICATION CLIENT
- ☒ REPLICATION SLAVE
- ☒ SELECT
- ☒ SHOW DATABASES
- ☒ SHOW VIEW
- ☒ SHUTDOWN
- ☒ SUPER
- ☒ TRIGGER
- ☒ UPDATE

Name: GymBuddy Specify the name of the schema here. You can use any combination of ANSI letters, numbers and the underscore character for names

Rename References Refactor model, changing all references found in view, triggers, stored procedures and functions from the old schema name to the new one.

Charset/Collation: utf8 utf8\_bin The character set and its collation selected here will be used when no other charset/collation is set for a database object (it uses the



## Configuración Springboot

En nuestra clase main del back añadimos lo siguiente, a su vez en el pom añadimos la dependencia de dotenv.

Una vez hecho esto empaquetamos la aplicación en un fichero JAR a través del siguiente comando, **mvn package -DskipTests**, o en el caso de tener mvnw, **./mvnw package -DskipTests**.

```
src-api > GymBuddy > src > main > java > gymbuddy > app > GymBuddyApplication.java > ...
17
18 /**
19  * La clase principal para iniciar la aplicación GymBuddy.
20  */
21 @SpringBootApplication
22 @EnableJpaRepositories
23 public class GymBuddyApplication {
24
25     /**
26      * El método principal para iniciar la aplicación GymBuddy.
27      *
28      * @param args Argumentos de línea de comandos.
29      */
30     public static void main(String[] args) {
31         if (System.getenv(name: "AWS_EXECUTION_ENV") == null) {
32             int maxDepth = 5;
33             Optional<Path> envPathOptional = findEnvFile(Paths.get(first:"").toAbsolutePath(), maxDepth);
34             if (envPathOptional.isPresent()) {
35                 Path envPath = envPathOptional.get();
36                 Dotenv dotenv = Dotenv.configure().directory(envPath.getParent().toString()).load();
37                 dotenv.entries().forEach(entry -> {
38                     System.setProperty(entry.getKey(), entry.getValue());
39                 });
40             } else {
41                 System.out.println(x:".env file not found!");
42             }
43         }
44
45         SpringApplication.run(primarySource:GymBuddyApplication.class, args);
46     }
47
48     private static Optional<Path> findEnvFile(Path startPath, int maxDepth) {
49         Path currentPath = startPath;
50         int depth = 0;
51         while (currentPath != null && depth <= maxDepth) {
52             Path envFile = currentPath.resolve(other:".env");
53             if (Files.exists(envFile)) {
54                 return Optional.of(envFile);
55             }
56             currentPath = currentPath.getParent();
57             depth++;
58         }
59         return Optional.empty();
60     }
61 }
```

```
<dependency>
  <groupId>io.github.cdimascio</groupId>
  <artifactId>dotenv-java</artifactId>
  <version>3.0.0</version>
</dependency>
```



## Configuración Angular

Creamos una carpeta environments donde meteremos un environments.ts para el local y un environments.prod.ts para AWS. Luego, en todos los lados donde haya <http://localhost:8080/api> por apiUrl.

```
src-frontend > GymBuddyVista > src > environments > ts environment.prod.ts > ...  
1 export const environment = {  
2   production: true,  
3   apiUrl: 'http://gymbuddy.us-east-1.elasticbeanstalk.com/api'  
4 };
```

```
src-frontend > GymBuddyVista > src > environments > ts environment.ts > ...  
1 export const environment = {  
2   production: false,  
3   apiUrl: 'http://localhost:8080/api'  
4 };
```

```
import { environment } from 'src/environments/environment'  
  
@Injectable({  
  providedIn: 'root'  
})  
export class EjerciciosService {  
  private baseUrl = environment.apiUrl+'/ejercicios';  
}
```

Seguido a esto en angular.json añadimos lo siguiente. Una vez realizado ejecutamos con el comando **ng s -o** y empaquetamos con **ng build --configuration production**

```
"configurations": {  
  "production": {  
    "fileReplacements": [  
      {  
        "replace": "src/environments/environment.ts",  
        "with": "src/environments/environment.prod.ts"  
      }  
    ],  
  },  
}
```



## Despliegue de la Aplicación en AWS

Accedemos a Elastic Beanstalk en AWS y creamos un nuevo entorno. A continuación, vamos a configurar el entorno. Le ponemos un nombre y un dominio, seleccionamos la plataforma Java con la versión que tengamos y cargamos el .jar generado anteriormente y en el nombre de la versión ponemos el nombre de la Snapshot.

Elastic Beanstalk > Environments

Entornos (1) Información 🔄 Acciones Creación de entorno

### Configuración del entorno Información

**Nivel de entorno Información**  
Amazon Elastic Beanstalk tiene dos tipos de niveles de entorno para admitir diferentes tipos de aplicaciones web.

☒ Entorno de servidor web  
Ejecute un sitio web, una aplicación web o una API web que atienda solicitudes HTTP. [Más información](#)

☐ Entorno de trabajo  
Ejecute una aplicación de proceso de trabajo que procese cargas de trabajo de ejecución prolongada bajo demanda o realice tareas de forma programada. [Más información](#)

**Información de la aplicación Información**

Nombre de aplicación

Gymbuddy

La longitud máxima es de 100 caracteres.

► Etiquetas de aplicación (opcional)

**Información del entorno Información**  
Elija el nombre, el subdominio y la descripción del entorno. No se pueden cambiar más adelante.

Nombre del entorno

Gymbuddy-env-1

Debe tener entre 4 y 40 caracteres. El nombre solo puede contener letras, números y guiones. No puede comenzar ni terminar por un guion. Este nombre debe ser único dentro de una región de su cuenta.

Dominio

gymbuddy .us-east-1.elasticbeanstalk.com

Verificar disponibilidad

### Plataforma Información

**Tipo de plataforma**

☒ Plataforma administrada  
Plataformas publicadas y mantenidas por Amazon Elastic Beanstalk. [Más información](#)

☐ Plataforma personalizada  
Plataformas creadas y de su propiedad. Esta opción no está disponible si no tiene plataformas.

**Plataforma**

Java

**Ramificación de la plataforma**

Corretto 21 running on 64bit Amazon Linux 2023

**Versión de la plataforma**

4.2.5 (Recommended)

### Código de aplicación Información

☐ Aplicación de ejemplo

☐ Versión existente  
Versiones de la aplicación que ha cargado.

☒ Cargar el código  
Cargue un paquete de código fuente desde su equipo o copie uno desde Amazon S3.

**Etiqueta de versión**  
Nombre único para esta versión del código de la aplicación.

Etiqueta de versión

La etiqueta de la versión de la aplicación es obligatoria  
Origen del código fuente. Tamaño máximo de 500 MB

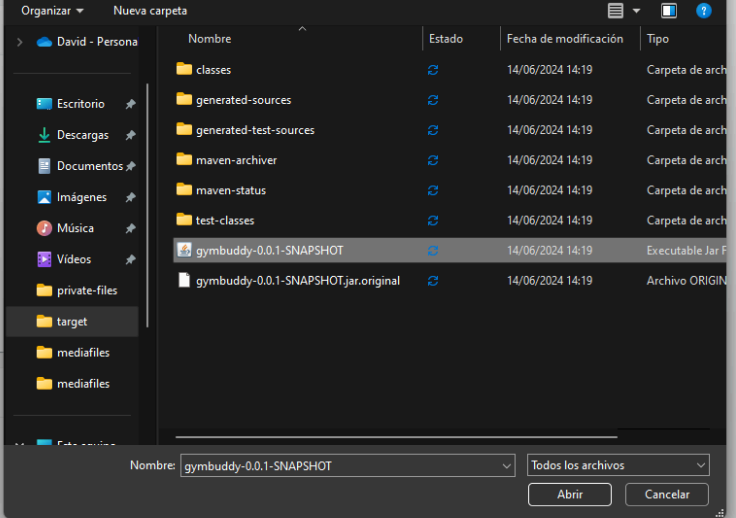
☒ Archivo local

Cargar aplicación

Elegir archivo

El archivo debe tener un tamaño máximo de archivo inferior a 500 MB

☐ URL pública de S3



Una vez realizado esto pasamos al servicio, aquí añadimos el rol de LabRole, la clave vockey y el perfil LabInstanceProfile.

## Configuración del acceso al servicio Información

### Acceso al servicio

Los roles de IAM, asumidos por Elastic Beanstalk como rol de servicio, y los perfiles de instancia de EC2 permiten a Elastic Beanstalk crear y administrar su entorno. Tanto el rol de IAM como el perfil de instancia deben estar asociados a políticas administradas de IAM que contengan los permisos necesarios. [Más información](#)

**Rol de servicio**

☐ Crear y utilizar un nuevo rol de servicio

☒ Usar un rol de servicio existente

**Roles de servicio existentes**  
Elija un rol de IAM existente para que Elastic Beanstalk asuma como rol de servicio. El rol de IAM existente debe tener las políticas administradas de IAM necesarias.

LabRole

**Par de claves de EC2**  
Seleccione un par de claves de EC2 para iniciar sesión de forma segura en sus instancias de EC2. [Más información](#)

vockey

**Perfil de instancia de EC2**  
Elija un perfil de instancia de IAM con políticas administradas que permitan a las instancias de EC2 realizar las operaciones necesarias.

LabInstanceProfile

[Ver los detalles de los permisos](#)





Pasamos al paso 3, la VPC la dejamos por defecto y configuramos la base de datos poniendo los datos de mysql y nuestro usuario y contraseña. En el paso 4 lo dejamos por defecto.

☒ Habilitar base de datos

### Restaurar una instantánea: *opcional*

Restaurar una instantánea existente a partir de una base de datos utilizada anteriormente.

Instantánea

Ninguno ▼

### Configuración de base de datos

Elija un motor y un tipo de instancia para la base de datos del entorno.

Motor

mysql ▼

Versión del motor

8.0.35 ▼

Clase de instancia

db.t3.small ▼

Almacenamiento

Elija un número comprendido entre 5 GB y 1024 GB.

5

 GB

Nombre de usuario

davidGymbuddy

Contraseña

.....

Disponibilidad

Baja (una AZ) ▼

**Política de eliminación de bases de datos**

Esta política se aplica al desacoplar una base de datos o al terminar el entorno asociado a ella.

☐ Crear instantánea

Elastic Beanstalk guarda una instantánea de la base de datos y, a continuación, la elimina. Puede restaurar una base de datos a partir de una instantánea al agregar una base de datos a un entorno de Elastic Beanstalk o al crear una base de datos independiente. Es posible que se apliquen cargos por almacenar instantáneas de base de datos.

☐ Retener

La base de datos desacoplada seguirá estando disponible y operativa fuera de Elastic Beanstalk.

☒ Eliminar



En el paso 5 agregamos las siguientes variables, una vez realizado esto, si accedemos al dominio que se genera en el entorno y vemos que nos da el siguiente fallo

CORS_ALLOWED_ORIGIN	http://localhost:8081	Eliminar
GRADLE_HOME	/usr/local/gradle	Eliminar
M2	/usr/local/apache-maven/bin	Eliminar
M2_HOME	/usr/local/apache-maven	Eliminar
MYSQL_DATABASE	testdb	Eliminar
MYSQL_HOST	127.0.0.1	Eliminar
MYSQL_PASSWORD	changeit	Eliminar
MYSQL_PORT	3306	Eliminar
MYSQL_USER	userapp	Eliminar
SERVER_PORT	5000	Eliminar

## 502 Bad Gateway

---

nginx

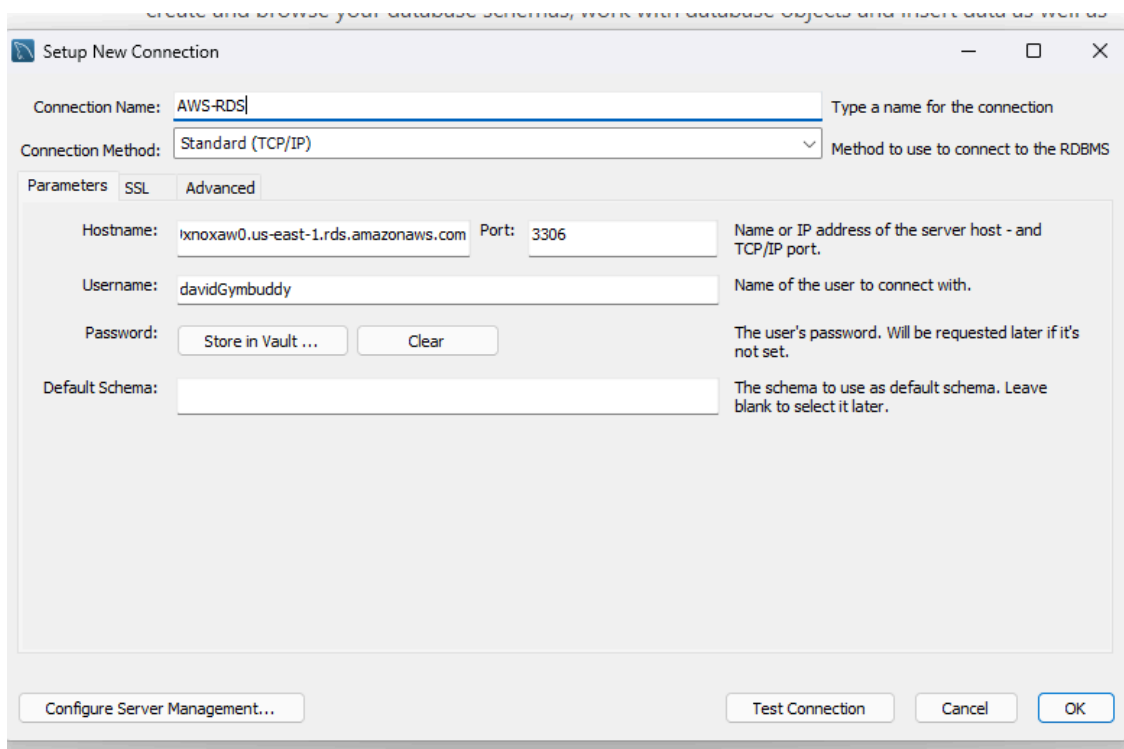
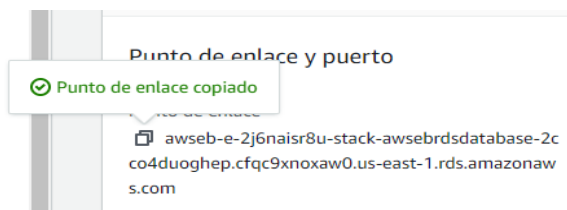


## Conexión con la base de datos

Accedemos a RDS y nos metemos a la instancia que se nos ha creado, nos aseguramos que sea pública y nos metemos al grupo de seguridad. En esta pestaña añadimos las siguientes reglas de entradas. Copiamos el punto de entrada y nos creamos una conexión con el hostname del punto de entrada

Reglas de entrada <small>Información</small>						
ID de la regla del grupo de seguridad	Tipo <small>Información</small>	Protocolo <small>Información</small>	Intervalo de puertos <small>Información</small>	Origen <small>Información</small>	Descripción: opcional <small>Información</small>	
sgr-01de2c410f770bab9	Todos los TCP	TCP	0 - 65535	Anywhere-I...	0.0.0.0/0	Eliminar
sgr-Def24b31d77523284	SSH	TCP	22	Anywhere-I...	0.0.0.0/0	Eliminar
sgr-0a494b535eb6a5cfd	MYSQL/Aurora	TCP	3306	Mi IP	46.27.105.116/32	Eliminar
Agregar regla						

Copiamos el punto de entrada y nos creamos una conexión con el hostname del punto de entrada y le añadimos el nombre de usuario puesto anteriormente.






A continuación actualizamos la variable MYSQL\_HOST por el punto de enlace de RDS . Una vez actualizado, creamos una base de datos y nos aparecerán las tablas generadas automáticamente.

MYSQL\_HOST

awseb-e-3xaummn5wv-stack-awsebrds

Eliminar



Name: GymBuddy

Rename References

Charset/Collation: utf8 utf8\_unicode\_c

**gymbuddy**

Tables

ejercicio

ejercicio\_entrenamiento

entrenamiento

usuario

usuario\_entrenamiento

usuario\_rol

Views

Stored Procedures

Functions



## Despliegue Aplicación Angular

Accedemos a S3 y nos metemos al bucket creado por Elastic Beanstalk, una vez dentro, le damos a propiedades y vamos a alojamiento de sitios web estáticos, habilitamos esta opción con las siguientes configuraciones

### Editar alojamiento de sitios web estáticos

[Información](#)

#### Alojamiento de sitios web estáticos

Utilice este bucket para alojar un sitio web o redirigir solicitudes. [Más información](#)

Alojamiento de sitios web estáticos

☐ Desactivar

☒ Habilitar

Tipo de alojamiento

☒ Alojar un sitio web estático

Utilice el punto de enlace del bucket como dirección web. [Más información](#)

☐ Redirigir las solicitudes de un objeto

Redirija las solicitudes a otro bucket o dominio. [Más información](#)

Para que sus clientes puedan obtener acceso al contenido en el punto de enlace del sitio web, debe hacer que todo el contenido sea legible públicamente. Para ello, puede editar la configuración Bloquear acceso público de S3 del bucket. Para obtener más información, consulte [Utilizar Bloquear acceso público de Amazon S3](#)

Documento de índice

Especifique la página predeterminada o de inicio del sitio web.

index.html

Documento de error - *opcional*

Esto se devuelve cuando se produce un error.

index.html

Reglas de redireccionamiento: *opcionales*

Redirija las reglas, escritas en JSON, para redirigir automáticamente las solicitudes de páginas web de contenido específico. [Más información](#)

Le damos a Permisos y añadimos la siguiente política de privacidad, luego en objetos, cargamos la carpeta dist generada anteriormente y la hacemos pública.

Política

```
2  "Version": "2008-10-17",
3  "Statement": [
4    {
5      "Sid": "eb-af163bf3-d27b-4712-b795-d1e33e331ca4",
6      "Effect": "Allow",
7      "Principal": {
8        "AWS": "arn:aws:iam::002949067520:role/LabRole"
9      },
10     "Action": [
11       "s3:ListBucket",
12       "s3:ListBucketVersions",
13       "s3:GetObject",
14       "s3:GetObjectVersion"
15     ],
16     "Resource": [
17       "arn:aws:s3:::elasticbeanstalk-us-east-1-002949067520",
18       "arn:aws:s3:::elasticbeanstalk-us-east-1-002949067520/resources/environments/*"
19     ]
20   },
21   {
22     "Sid": "eb-58950a8c-feb6-11e2-89e0-0800277d041b",
23     "Effect": "Deny",
24     "Principal": {
25       "AWS": "*"
26     },
27     "Action": "s3:DeleteBucket",
28     "Resource": "arn:aws:s3:::elasticbeanstalk-us-east-1-002949067520"
29   }
30 ]
```

Objetos (12) Información

Los objetos son las entidades fundamentales que se almacenan en Amazon S3. Puede utilizar el [Inventario de Amazon S3](#) para obtener una lista de todos los objetos de su bucket. Para que otras personas obtengan acceso a sus objetos, tendrá que concederles permisos de forma explícita.

Buscar objetos por prefijo

<input checked="" type="checkbox"/>	Nombre	Tipo	Última modificación	Tamaño
<input checked="" type="checkbox"/>	elasticbeanstalk	elasticbeanstalk	11 Jun 2024 8:40:55 PM CEST	0 B
<input checked="" type="checkbox"/>	1718375409591-gymbuddy-0.0.1-SNAPSHOT.jar	jar	14 Jun 2024 4:30:12 PM CEST	62.8 MB
<input checked="" type="checkbox"/>	3rdpartylicenses.txt	txt	14 Jun 2024 2:24:03 PM CEST	13.1 KB
<input checked="" type="checkbox"/>	assets/	Carpeta	-	-
<input checked="" type="checkbox"/>	favicon.ico	ico	14 Jun 2024 2:24:04 PM CEST	948.0 B
<input checked="" type="checkbox"/>	index.html	html	14 Jun 2024 2:24:05 PM CEST	2.3 KB
<input checked="" type="checkbox"/>	main.c2b4d82ade7c09e2.js	js	14 Jun 2024 2:24:06 PM CEST	332.7 KB
<input checked="" type="checkbox"/>	mediafiles/	Carpeta	-	-
<input checked="" type="checkbox"/>	polyfills.1bc4f8072e50aac6.js	js	14 Jun 2024 2:24:06 PM CEST	33.0 KB
<input checked="" type="checkbox"/>	resources/	Carpeta	-	-
<input checked="" type="checkbox"/>	runtime.266eec84a7890bb1.js	js	14 Jun 2024 2:24:07 PM CEST	904.0 B
<input checked="" type="checkbox"/>	styles.8c562f895a887af9.css	css	14 Jun 2024 2:24:07 PM CEST	17.4 KB

Acciones

Descargar como

Compartir con una URL prefirmada

Calcular el tamaño total

Copiar

Trasladar

Iniciar restauración

Consultar con S3 Select

Editar acciones

Cambiar el nombre del objeto

Editar clase de almacenamiento

Editar cifrado del lado del servidor

Editar metadatos

Editar etiquetas

Hacer público mediante ACL



## Error Con CORS

Si nos vamos a Propiedades y accedemos a la url nos da un error de CORS, por lo que debemos configurar en el back el archivo WebConfig y donde tengamos CrossOrigins, de la siguiente forma.

```
@Configuration
@EnableWebMvc
public class WebConfig implements WebMvcConfigurer {

    @Override
    public void addCorsMappings(CorsRegistry registry) {
        registry.addMapping(pathPattern: "/api/**")
            .allowedOrigins(...origins: "http://localhost:4200", "http://elasticbeanstalk-us-east-1-002949067520.s3-website-us-east-1.amazonaws.com")
            .allowedMethods(...methods: "GET", "POST", "PUT", "DELETE", "OPTIONS")
            .allowedHeaders(...headers: "**")
            .allowCredentials(allowCredentials: true);
    }
}
```

Una vez realizado esto, empaquetamos la aplicación de nuevo y nos vamos a Elastic Beanstalk y cargamos de nuevo el Snapshot

**Cargar e implementar** ×

ℹ Para implementar una versión anterior, vaya a [Página Versiones de la aplicación](#)

Cargar aplicación

**Elegir archivo**

✓ Nombre del archivo: **gymbuddy-0.0.1-SNAPSHOT.jar**

El archivo debe tener un tamaño máximo de archivo inferior a 500 MB

Etiqueta de versión

Nombre único para esta versión del código de la aplicación.

0.0.1-SNAPSHOT-13

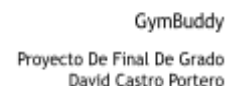
Número actual de instancias EC2: **1**

Cancelar

Implementar

[David Castro Portero](#)

31



David Castro Portero

32





# Conclusión y Postmortem

## Conclusión

GymBuddy representa un paso significativo hacia la creación de una plataforma integral para el acondicionamiento físico personalizado y profesional.

1. **Lecciones Aprendidas:** Durante el proceso, he aprendido la importancia de la planificación detallada y la flexibilidad para adaptarse a cambios.
2. **Próximos Pasos:** Para futuras iteraciones de GymBuddy, se mejorará la aplicación y se implementarán las funcionalidades no completadas

## Postmortem

Durante la ejecución del proyecto GymBuddy, se superaron varios desafíos técnicos y organizativos. A continuación, se detallan los aspectos clave evaluados:

1. **Desarrollo y Tecnología:** La elección del stack tecnológico, incluyendo Spring Boot para el back-end, Angular para el front-end, MySQL como base de datos, y herramientas como Tailwind CSS y Figma para el diseño. Estas tecnologías permitieron un desarrollo eficiente y la creación de una aplicación robusta y escalable.
2. **Retos Superados:** Durante el desarrollo, se enfrentaron desafíos técnicos como la integración compleja de componentes en Angular y la optimización de consultas en la base de datos MySQL.

A pesar de esto, debido al gran trabajo en la FCT, no se han podido completar algunas funcionalidades que se estableció al principio, pero está pensado para que a futuro esas funcionalidades se integren en la aplicación y poder mejorarla.

En este enlace se puede acceder al repositorio de GitHub con el proyecto realizado, [enlace](#).