

Memoria Proyecto PF08:

**Dados Zombies con Spring
MVC y MONGODB**

José Manuel Romero Rodríguez

Dados Zombies con Spring MVC y MongoDB

Índice de contenidos.

1. Introducción
2. Requisitos
3. Estructura de directorios de la aplicación.
4. Base de datos PF08 MongoDB
5. Probando la aplicación en un servidor local
6. Diagrama de casos de uso
7. Diagrama de clases
8. Código Fuente
9. Ficheros de configuración

1. Introducción

Las herramientas que he utilizado son las siguientes:

- [Spring Source Tool Suite](#) version 3.7.3 basado en Eclipse Mars
- [Eclipse Mars para desarrolladores Java EE](#)
- [Maven\(incluido en STS\)](#)
- [Log4j\(incluido en STS\)](#)
- [JBoss AS 7.1](#)
- [MongoChef 3.5.1](#)
- [MongoDB 3.2.5](#)

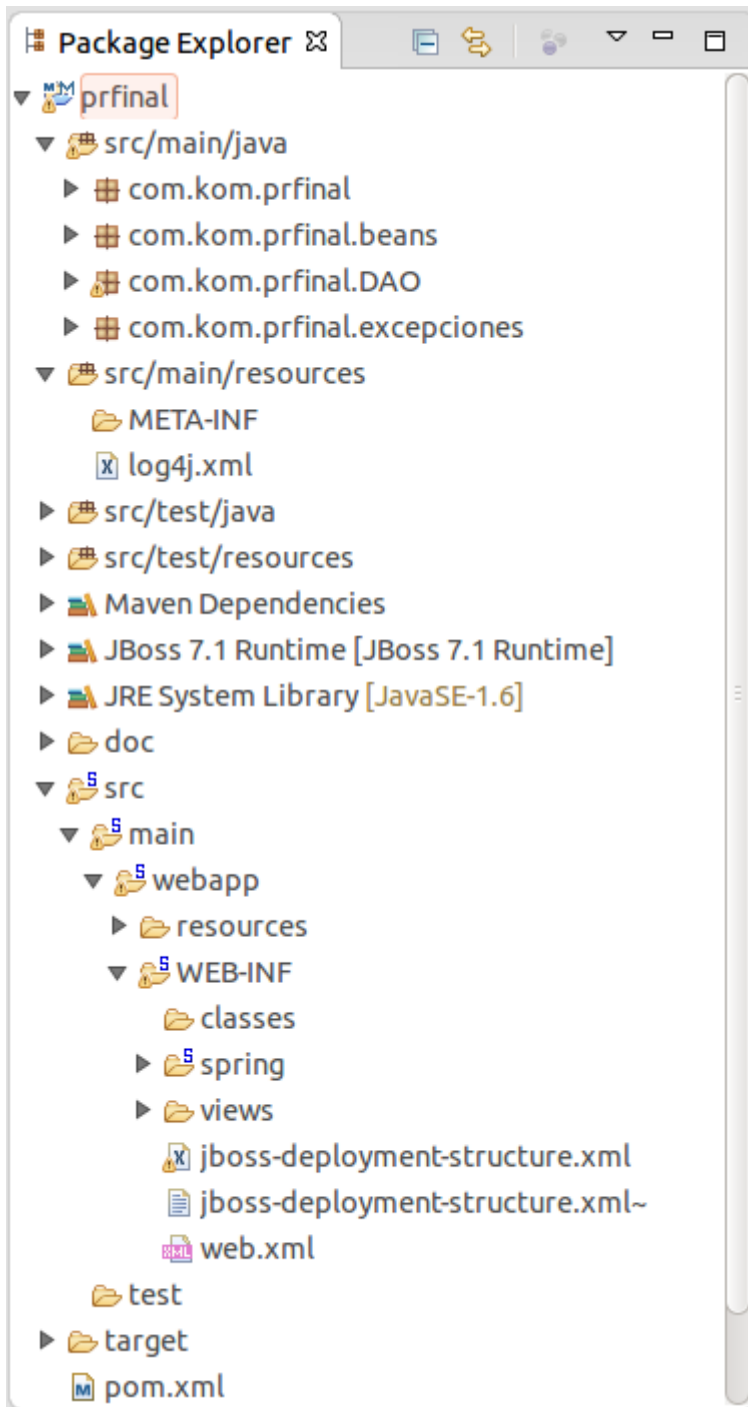
2. Requisitos

Los requisitos necesarios para la ejecución de la aplicación es tener instalado un servidor de aplicaciones, en este caso se ha usado JBOSS AS 7.1, y MongoDB como sistema gestor de bases de datos.

Una vez desplegada la aplicación en el servidor, hace uso de MongoDB y hará una conexión en localhost con el puerto por defecto a una base de datos llamada PF08 (que la creará en el caso de que no exista).

3. Estructura de directorios de la aplicación.

Esta es la estructura de directorios:



Los ficheros de configuracion que se han configurado son los siguientes:

- pom.xml, este es el archivo central de Maven, este archivo contiene Meta-Información del proyecto, como el nombre, la organización y lo mas importante, las dependencias, además también podemos añadir plugins para Maven.

Se han incluido las dependencias de MongoJavaDriver(3.2.2) y de GSON(2.6.2)

```
<!-- MongoDB -->
<dependency>
  <groupId>org.mongodb</groupId>
  <artifactId>mongo-java-driver</artifactId>
  <version>3.2.2</version>
</dependency>

<!-- GSON -->
<dependency>
  <groupId>com.google.code.gson</groupId>
  <artifactId>gson</artifactId>
  <version>2.6.2</version>
</dependency>
```

- log4j.xml, es el fichero de configuración del Log4j.

Se ha configurado un appender para llevar el control de log de errores que se guardara en el fichero dentro de la carpeta de log de JBOSS(standalone/log/errorPF08.log)

```
<appender name="file" class="org.apache.log4j.RollingFileAppender">
  <param name="append" value="true" />
  <param name="maxFileSize" value="1024KB" />
  <param name="maxBackupIndex" value="5" />
  <param name="File" value="../standalone/log/errorPF08.log" />
  <layout class="org.apache.log4j.PatternLayout">
    <param name="ConversionPattern"
      value="%d{yyyy-MM-dd HH:mm:ss} %-5p %c{1}:%L - %m%n" />
  </layout>
</appender>
```

- jboss-deployment-structure.xml, fichero necesario para la configuración del JBOSS para que use el log4j, es necesario ponerlo para que no tome los loggers por defecto de JBOSS y use log4j.

```
<jboss-deployment-structure>
  <deployment>
    <exclusions>
      <module name="org.apache.log4j" />
    </exclusions>
  </deployment>
</jboss-deployment-structure>
```


•

Estos ficheros xml de configuración completos y los demás que se han utilizado están en la documentación al final del código fuente.

4. Base de datos MongoDB (PF08)

En las siguientes imágenes podemos ver las colecciones de las que hace uso la aplicación:

▼  PF08


▶  Partidas

▶  Usuarios

Key	Value	Type
▼ (1) { _id : 572ef432cd060c0b26d4ae03 }	{ 5 fields }	Document
_id	572ef432cd060c0b26d4ae03	ObjectId
Nombre	admin	String
estado	0	Int32
njugadores	4	Int32
▼ jugadores	[4 elements]	Array
▶ 0	{ 3 fields }	Object
▼ 1	{ 3 fields }	Object
login	kom	String
cerebros	0	Int32
estado	1	Int32
▼ 2	{ 3 fields }	Object
login	almu	String
cerebros	0	Int32
estado	0	Int32
▶ 3	{ 3 fields }	Object

▼  PF08

▶  Partidas

▶  Usuarios

Key	Value	Type
▶ (1) { _id : 571c90da9dbb6e1381597564 }	{ 6 fields }	Document
▼ (2) { _id : 57247adbcd060c0a57aa47b7 }	{ 6 fields }	Document
_id	57247adbcd060c0a57aa47b7	ObjectId
login	kom	String
pass	kom	String
email	kom@kom.com	String
puntos	11	Int32
nombre	king of metal	String

5. Probando la aplicación en un servidor local

Para probar la aplicación en un servidor local (en este caso JBOSS AS 7.1), hacemos click con el botón derecho sobre el nombre de nuestro proyecto, Run As -> Run on server, o lo desplegamos dentro de la configuración de JBOSS.

Nos saldrá la pantalla de bienvenida y ya podemos usar el juego.



La aplicación nos dará los siguientes opciones:

- **Reglas:**

Aquí podemos ver como se juega al juego con un manual y un vídeo explicativo:

!MMM! !CEREEEBROS!

En tu turno, agita el cubilete con los dados dentro, coge tres dados sin mirarlos, y lanzalos. Cada dado es una victima humana.

Los rojos son las mas resistentes. Los verdes son las mas debiles, y los amarillos son las de resistencia media.

Los dados tienen tres simbolos:

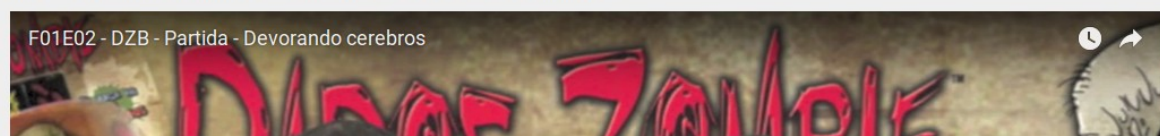
Cerebro - Te zampas el cerebro de tu victima. Pon los dados con el Cerebro a tu izquierda.

Disparo - ¡Tu victima se ha defendido! Pon los dados con el Disparo a tu derecha.

Huellas - Tu victima ha escapado. Mantén los dados con las Huellas delante de ti. Si escoges lanzar los dados de nuevo, volveras a lanzar estos dados, junto con otros nuevos, suficientes para llegar al total de tres.

!Cereeeebros!

Continúa jugando hasta que alguien llegue a los trece Cerebros. Entonces se finaliza la ronda. El jugador que tenga mas puntos al final de la ronda es el ganador.



- **Mi cuenta:**

Para poder entrar en la cuenta de usuario como en *Jugar* previamente tendremos que estar logueados en el sistema, podremos hacerlo mediante el siguiente formulario:



CONVIERTETE EN UN ZOMBIE Y COME CEREEEBROS

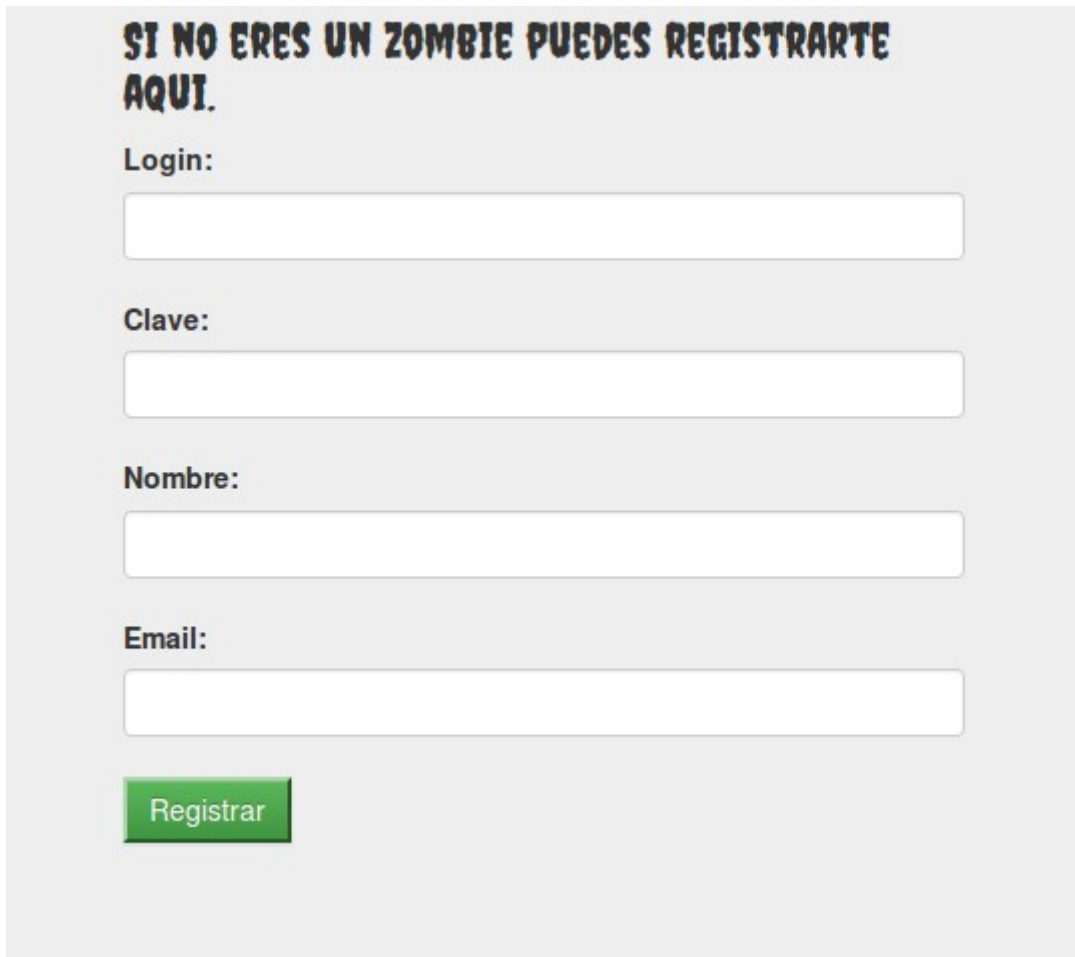
INTRODUZCA LOGIN Y CLAVE

Login:

Clave:

Entrar

Si no estamos registrados podemos hacerlo en el formulario de registro:



SI NO ERES UN ZOMBIE PUEDES REGISTRARTE AQUI.

Login:

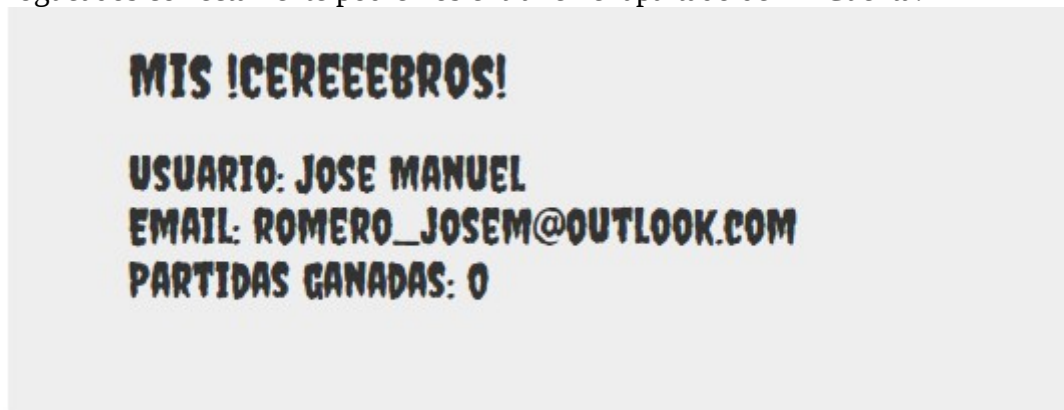
Clave:

Nombre:

Email:

Registrar

Una vez logueados correctamente podremos entrar en el apartado de Mi Cuenta:



donde se nos informa de nuestros datos y de las partidas ganadas, y tambien podemos acceder al juego:

- **Jugar**

Lo primero que tendremos que hacer es crear una partida nueva o unirnos a una existente:

LISTA DE PARTIDAS DISPONIBLES

Id	Creador	Jugadores	
572ef432cd060c0b26d4ae03	admin	4	<button>Unirse</button>
572ef54acd060c0b261e0984	radanisa	1	<button>Unirse</button>

Crear Partida

Una vez dentro de una partida tendremos que comenzar nuestra partida creada o esperar a que la comience otro usuario, solo el creador de la partida puede comenzar la partida.

ESPERANDO ZOMBIES PARA COMENZAR LA PARTIDA...




Nombre	Turno
admin	1

Comenzar Partida

Una vez la partida de comienzo nos ponemos a realizar nuestras tiradas y a comer los máximos cerebros posibles, el primer zombie que consiga 13 o mas sera el ganador:

Durante la partida podemos tanto realizar tiradas como plantarnos para acumular los cerebros conseguidos, pero ojo si acumulas 3 disparos durante una jugada moriras y perderas los cerebros conseguidos.

NOMBRE	CEREBROS
ADMIN	0
KOM	0



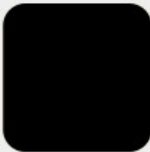


DISPAROS: 0
CEREBROS: 2

Tirar

Plantarse

Esta es la pantalla de espera mientras juegan los demas jugadores:

NOMBRE	CEREBROS
ADMIN	4
KOM	0



ESPERANDO A QUE COMAN
LOS DEMAS ZOMBIES

Cuando haya un ganador se nos informara y nos llevara a la pantalla de fin de partida, donde nos muestra la puntuación de todos los jugadores y el ganador:

RESULTADO DE LA PARTIDA

[

GANADOR: ADMIN

CEREBROS CONSEGUIDOS: 13

. USUARIO: KOM

CEREBROS CONSEGUIDOS: 2

]

Código Fuente y ficheros de configuración