**Roland Jasko**

**Grupo 2**

**#Instrucciones**

Los controladores realizan un CRUD basico para las siguientes entidades:

->Users (path controller: /users)

->Events (path controller: /events)

->Languages (path controller: /languages)

->Locations (path controller: /locations)

->Reviews (path controller: /reviews)

->Roles (path controller: /roles)

**#URL acceso**

http:// 44.208.195.232:8080/Grupo2-0.0.1-SNAPSHOT/'path controller'

**#JSON y comentarios**

->Estas son las estructuras JSON para cada entidad. La url de cada entidad muestra una ruta GET para obtener los datos de las entidades.

->Las solicitudes GET y POST se manejan con la siguiente url:

->http:// 44.208.195.232:8080/Grupo2-0.0.1-SNAPSHOT/path\_controller

->Adicionalmente se añade un metodo GET para poder encontrar un usuario a traves de su id con la siguiente url:

->http:// 44.208.195.232:8080/Grupo2-0.0.1-SNAPSHOT/path\_controller/id

->Los metodos DELETE y PUT necesitan el id del usuario para borrar y/o modificar los datos. La ruta url para estos metodos seran:

->http:// 44.208.195.232:8080/Grupo2-0.0.1-SNAPSHOT/path\_controller/id

->Adicionalmente se añade al controlador de la entidad Users un metodo login. La respuesta de este controlador sera el valor del id del usuario, si se encuentra uno, o un valor "-1" en el caso de no encontrar un usuario con el id. La solicitud necesita para la comprobacion los campos de email y pass. Se utiliza en este caso la siguiente url:

->http:// 44.208.195.232:8080/Grupo2-0.0.1-SNAPSHOT/users/login

->JSON para users:

{

"id":

"userName":

"email":

"pass":

"imgUrl":

"firstName":

"lastName":

"age":

"des":

"phone":

"address":

"points":

}

->JSON para users/login:

{

"email":

"pass":

}

->Events:

{

"id":

"imgUrl":

"title":

"eventDate":

"des":

"createdBy":

"language\_id":

}

->Roles:

{

"id":

"name":

}

->Reviews:

{

"id":

"des":

"ratingReview":

}

->Locations:

{

"id":

"location":

}

->Languages:

{

"id":

"language":

}

**#Instrucciones para el despliegue**

UF1846 - A1 - Instalar Tomcat en AWS - EC2 y desplegar App Spring

Instrucciones

Pasos a seguir para realizar la actividad

Utilizar vuestra VPC 10.0.0.0/16

Elegir una Subnet PUBLICA (Ejemplo 10.0.2.0/24)

Revisar que tiene activo el check "Asignar Automáticamente IP pública" y que está conectada correctamente al IGW (Internet Gateway)

Crear un Security Group que tenga abiertos por lo menos los puertos 22 y 8080

Crear un servidor virtual (EC2) de tipo Ubuntu 22 y tamaño "Medium" en la subnet pública elegida anteriormente

Asignar una IP fija (Elastic IP) al servidor.

Lanzar y conectarse al el servidor por SSH desde la terminal local.

Instalar Tomcat 10 siguiendo las instrucciones en el fichero adjunto.

Despliegue de la Aplicación de backend (Spring) en el nuevo application server Tomcat.

Crear el paquete WAR de la aplicación usando los comandos oportunos en VSCode

Desplegar el fichero .war utilizando la página de administración de Tomcat

Entregar un fichero de Texto Plano (TXT) con el link a la API de vuestra aplicación de backend, las indicaciones y los Json necesarios para testear el correcto funcionamiento a través de Postman.