

PRUEBA DE CONOCIMIENTOS EN JAVA PARA UNA VACANTE EN LA EMPRESA KIRITEK

CREADO POR:

PROGRAMADOR WEB Y MOVIL

RAMIRO JAVIER MEDINA URIBE

PROYECTO:

API REST FUL EN SPRING BOOT CON JPA Y
MYSQL

INDICE

Contenido

REQUERIMIENTOS DEL PROYECTO.....	3
IDENTIDADES.....	3
MODELO RELACIONAL DE LA BASE DE DATOS.....	5
PATRON DE DISEÑO	6
ENDPOINTS.....	7
POST MAN	8

REQUERIMIENTOS DEL PROYECTO

Requerimientos:

- Java 8 o superior.
- PostMan
- SPRING BOOT SUITE 4 (algun otro IDE como Eclipse, Visual Code o IntelliJ IDEA)
- Mysql

IDENTIDADES

Se definieron como identidades las siguientes clases:

```
com.test.api.model
├── AuxComentario.java
├── Comentario.java
├── Libro.java
├── Reseña.java
└── Usuario.java
```

```
public class Usuario {

    public static interface BaseUser {
    }

    public static interface SingleUser extends BaseUser {
    };

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Integer id;
    @JsonView(BaseUser.class)
    @Column(unique=true)
    private String email;
    @JsonView(SingleUser.class)
    private String password;
    @JsonView(BaseUser.class)
    private String username;
    @OneToMany(mappedBy = "usuario")
    private List<Comentario> comentarios;
    @OneToMany(mappedBy = "usuario")
    private List<Reseña> reseña;
```

Clase usuario que contiene dos variables extras el mapeo de las llaves foráneas.

Clase Reseña instancia también dos variables de tipo usuario y libro para crear la relación con esas clases.

```
public class Reseña {
    public static interface BaseReseña extends BaseUser, BaseComentario {
    }
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Integer id;
    @JsonView(BaseReseña.class)
    private String reseña;
    // Relacion Reseña a Comentarios
    @JsonView(BaseReseña.class)
    @OneToMany(mappedBy = "reseña")
    private List<Comentario> comentarios;
    // Relacion Reseña - Usuario
    @JsonView(BaseReseña.class)
    @ManyToOne
    private Usuario usuario;
    // Relacion Reseñas - Libro
    @JsonView(BaseReseña.class)
    @ManyToOne
    private Libro libro;
```

```
public class Libro {

    public static interface BaseLibro extends BaseReseña { //
    }

    public static interface SingleLibro extends BaseLibro {
    };

    @JsonView(BaseLibro.class)
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Integer id;
    @JsonView(BaseLibro.class)
    private String titulo;
    @JsonView(BaseLibro.class)
    private String sinopsis;
    @JsonView(BaseLibro.class)
    private String portada;
    private Date fecha_publicacion;
    @JsonView(SingleLibro.class)
    @OneToMany(mappedBy = "libro")
    private Reseña reseñas;
}
```

Clase Libro, contiene una variable de tipo reseña para el mapeo de la llave foránea de la clase reseña.

```
public class Comentario {
    public static interface BaseComentario {
    }

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Integer id;
    @JsonView(BaseComentario.class)
    private String comentario;
    // Parametro Relacion Comentarios - Reseña

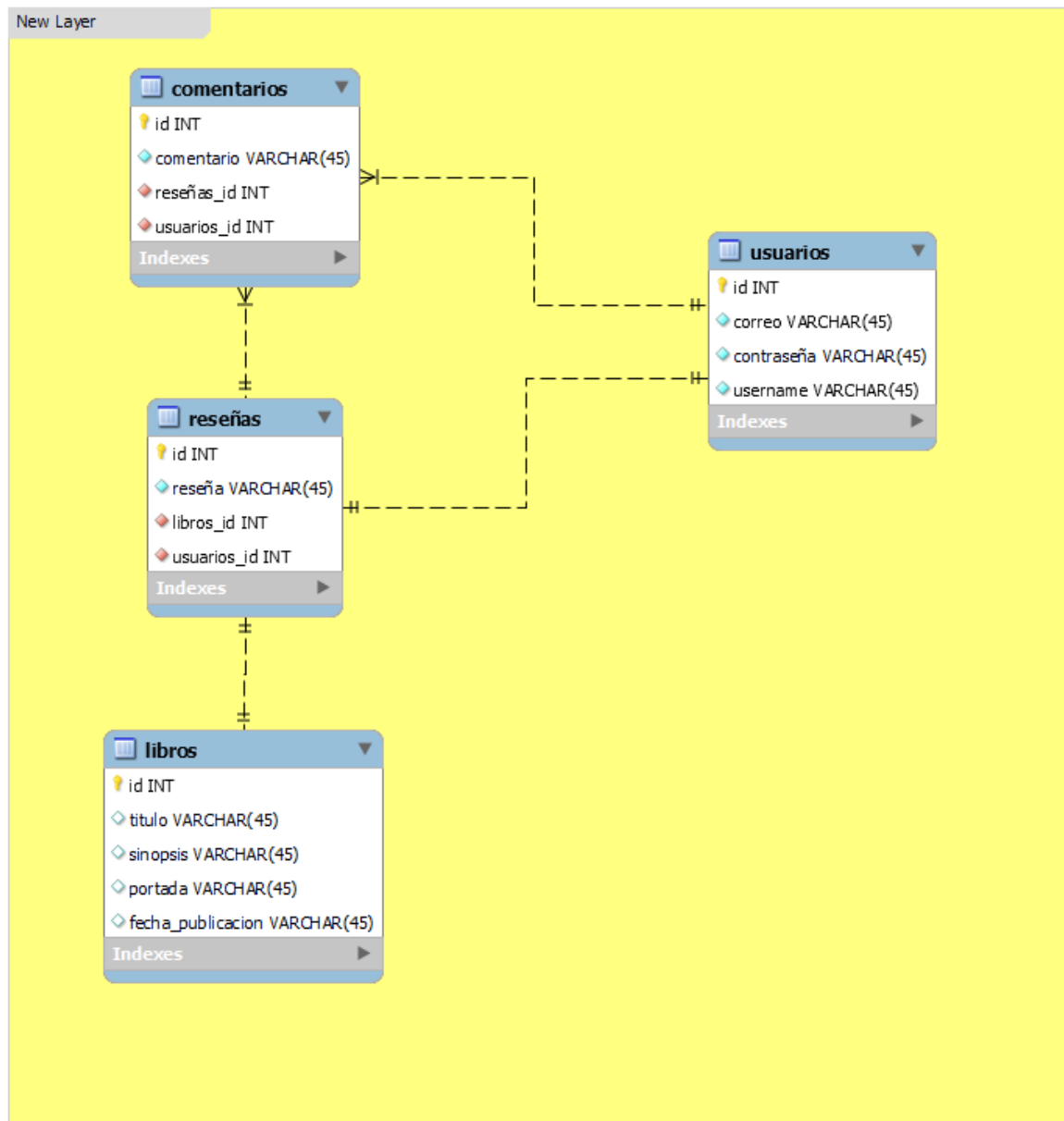
    @ManyToOne
    private Reseña reseña;
    @JsonView(BaseComentario.class)
    @ManyToOne
    private Usuario usuario;
}
```

Clase Comentario, instancia 2 variables extras de la clase reseña y usuario para la creación de las llaves foráneas.

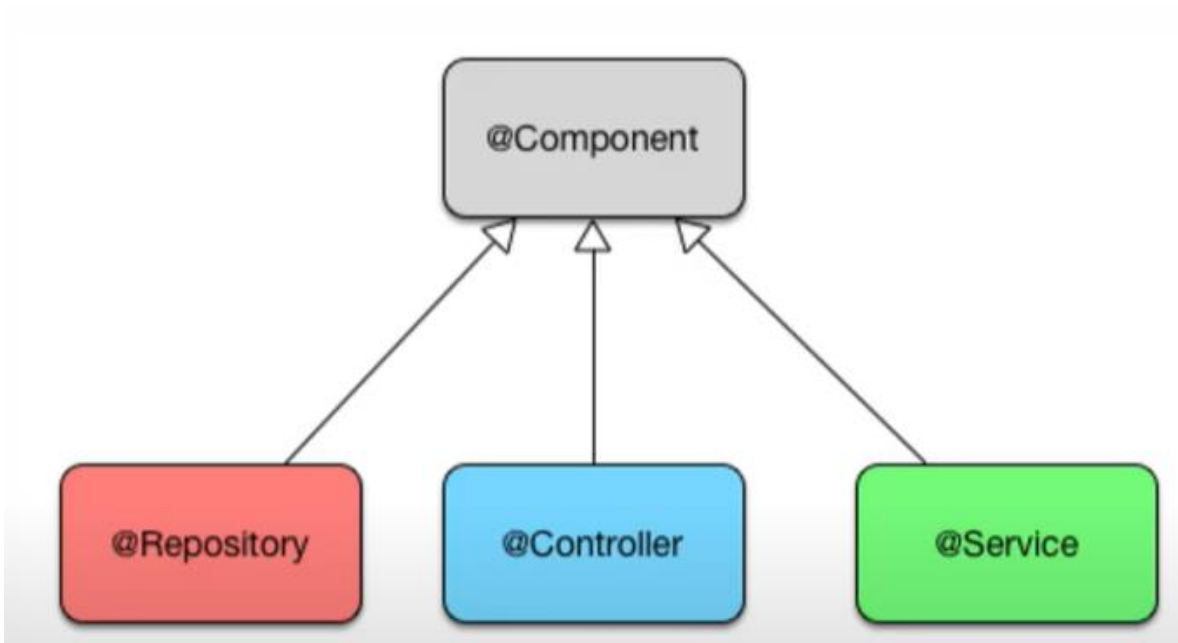
Se creó una clase AuxComentario como alternativa para guardar el id de la llave foránea dentro de la clase Comentario, ya que JPA requiere de instanciar un objeto de la clase de la llave foránea para insertar un nuevo campo a la tabla, se adaptó como alternativa la creación de esta clase auxiliar para poder enviar desde una petición post con formato json solamente el id del usuario y también de la reseña.

```
public class AuxComentario {
    private String comentario;
    private Integer usuario_id;
    private Integer reseña_id;
    public AuxComentario(String comentario, Integer usuario_id, Integer reseña_id) {
        super();
        this.comentario = comentario;
        this.usuario_id = usuario_id;
        this.reseña_id = reseña_id;
    }
    public String getComentario() {
        return comentario;
    }
    public void setComentario(String comentario) {
        this.comentario = comentario;
    }
    public Integer getUsuario_id() {
        return usuario_id;
    }
    public void setUsuario_id(Integer usuario_id) {
        this.usuario_id = usuario_id;
    }
    public Integer getReseña_id() {
        return reseña_id;
    }
    public void setReseña_id(Integer reseña_id) {
        this.reseña_id = reseña_id;
    }
}
```

MODELO RELACIONAL DE LA BASE DE DATOS



PATRON DE DISEÑO



Controladores : Contiene la lógica de la aplicación y pasa los datos de entrada del usuario al servicio

Servicios : El middleware entre el controlador y el repositorio. Recopila datos del controlador, realiza la validación y la lógica comercial, y llama a los repositorios para la manipulación de datos.

Repositorios : Capa para interactuar con modelos y realizar operaciones de base de datos.

ENDPOINTS

(En amarillo los requeridos)

END POINT LIBROS

{localhost:8080}/api/libros

METODOS

GET {localhost:8080}/api/libros{id}

PUT {localhost:8080}/api/libros{id}

DELETE {localhost:8080}/api/libros{id}

POST {localhost:8080}/api/libros

```
{
  "titulo"           : "",
  "sinopsis"         : "",
  "portada"          : "",
  "fecha_publicacion" : ""
}
```

} JSON FORMAT

END POINT USUARIOS

{localhost:8080}/api/usuarios

METODOS

GET {localhost:8080}/api/usuarios {id}

PUT {localhost:8080}/api/usuarios {id}

DELETE {localhost:8080}/api/usuarios {id}

END POINT SIGNIN Y SIGNUP

POST {localhost:8080}/api/usuarios/signin

```
{
  "email"       : "",
  "password"    : ""
}
```

} JSON FORMAT

POST {localhost:8080}/api/usuarios/signup

```
{
  "email"       : "",
  "password"    : "",
  "username"    : ""
}
```

} JSON FORMAT

END POINT COMENTARIOS

METODOS

POST {localhost:8080}/api/comentarios

```
{
  "comentario"      : "",
  "reseña_id"       : "",
  "usuario_id"      : ""
}
```

} JSON FORMAT

END POINT RESEÑAS

{localhost:8080}/api/reseñas

METODOS

GET

POST MAN

LOGIN DEL USUARIO

localhost:8080/api/usuarios/signin

POSTlocalhost:8080/api/usuarios/signin

ParamsAuthorizationHeaders (8)BodyPre-request ScriptTestsSettings

noneform-datax-www-form-urlencodedrawbinaryGraphQLJSON

```
1 {
2   ... "email": "javi@gmail.com",
3   ... "password": "123",
4   ... "username": "Javier"
5 }
```

BodyCookiesHeaders (5)Test Results

200 OK121 ms

PrettyRawPreviewVisualizeText

1 Usuario Logeado Correctamente

LECTURA DE TODOS LOS LIBROS ORDENADOS ALFABÉTICAMENTE Y POR MAS RECIENTE.

GETlocalhost:8080/api/libros

ParamsAuthorizationHeaders (8)BodyPre-request ScriptTestsSettings

noneform-datax-www-form-urlencodedrawbinaryGraphQLJSON

1

BodyCookiesHeaders (5)Test Results

200 OK132 ms4.75 KB

PrettyRawPreviewVisualizeJSON

```
1 [
2   {
3     "id": 5,
4     "titulo": "La mitad siniestra",
5     "sinopsis": "Thad Beaumont es el protagonista de La mitad siniestra, un escritor que durante años publicó bajo el seudónimo de George Stark varias novelas de terror. Incluso escribió para la revista People una biografía más o menos aterrador de Stark que entusiasmó a los admiradores de su alter ego. Pero ahora rico, feliz, con una atractiva esposa y padre de dos gemelos, Beaumont no necesita ya a su doble en su ascendente carrera literaria, y decide enterrarlo. Sólo que de súbito, y de una manera brutal, descubrirá que no es nada fácil deshacerse del sombrío fantasma que ha creado. En el pequeño pueblo de Castle Rock, en Maine, donde Beaumont y su esposa tienen una casa de verano, el sheriff Alan Pangborn investiga un crimen de sadismo ejemplar, sólo comparable a las delirantes imaginaciones con que Stark se complacía en sus novelas. En el lugar, la policía encuentra también un verdadero mural de sangre con las huellas dactilares del homicida y, además, idénticas a las de Thad Beaumont. La pesadilla ha comenzado y oscuras fuerzas se ciernen sobre Castle Rock, ¿de dónde vienen? Aquí está la mitad siniestra, una historia tan real y fascinante que las legiones cada vez más numerosas de lectores de Stephen King verán renacer sus más recónditas emociones bajo el dominio -y los demonios- del maestro. Este retrato del lado oscuro del sueño americano les helará la sangre y paralizará su corazón, pero lo disfrutarán como ningún otro.",
6     "portada": "mitadsiniestra.jpg"
7   },
8   {
9     "id": 4,
10    "titulo": "La casa verde",
11    "sinopsis": "Novela ejemplar en la historia del boom latinoamericano, La Casa Verde es una experiencia inoludible para todo aquel que quiera conocer en profundidad la obra
```


DETALLE DEL LIBRO Y RESEÑAS CON COMENTARIOS

GETlocalhost:8080/api/libros/2

ParamsAuthorizationHeaders (6)BodyPre-request ScriptTestsSettings

bodyCookiesHeaders (5)Test Results

Status: 200 OKTime: 13 msSize: 1.03 KBSave F

PrettyRawPreviewVisualizeJSON

```
1 {
2   "id": 2,
3   "titulo": "La metamorfosis",
4   "sinopsis": "La metamorfosis es un relato dividido en tres partes, donde se narra la transformación de Gregorio Samsa, un viajante de comercio de telas, en un monstruoso insecto
5     impacto que tendrá este acontecimiento no solo en su vida, sino en la de su familia.",
6   "portada": "metamorfosis.jpg",
7   "reseñas": {
8     "comentarios": [
9       {
10        "comentario": "Comparto la misma opinión, excelente",
11        "usuario": {
12          "id": 2,
13          "email": "sergio@kiritek.com",
14          "username": "Sergio"
15        }
16      },
17    ],
18    "usuario": {
19      "id": 1,
20      "email": "javi@gmail.com",
21      "username": "Javier"
22    }
23  }
```

LISTA DE RESEÑAS CON COMENTARIOS

GETlocalhost:8080/api/reseñas

ParamsAuthorizationHeaders (8)BodyPre-request ScriptTestsSettings

bodyCookiesHeaders (5)Test Results

Status: 200 OKTime: 25 msSize: 712 BSave Respons

PrettyRawPreviewVisualizeJSON

```
1 {
2   "reseña": "Excelente libro, me encanto demasiado. Un pequeño punto dentro de la inmensidad infinita de la galaxia. La angustia existencial y el dolor que tales afirmaciones
3     pueden provocar parecen no tener fin. No en vano, la ansiedad y los trastornos psicológicos son la verdadera pandemia del nuevo siglo que acabamos de traspasar.\n",
4   "comentarios": [
5     {
6       "comentario": "Comparto la misma opinión, excelente",
7       "usuario": {
8         "id": 2,
9         "email": "sergio@kiritek.com",
10        "username": "Sergio"
11      }
12    },
13  ],
14  "usuario": {
15    "id": 1,
16    "email": "javi@gmail.com",
17    "username": "Javier"
18  }
19 }
```

CREACION DE NUEVO COMENTARIOS

POSTlocalhost:8080/api/comentarios

ParamsAuthorizationHeaders (8)BodyPre-request ScriptTestsSettings

bodyCookiesHeaders (5)Test Results

Status: 201 CreatedTime: 69 msSiz

PrettyRawPreviewVisualizeJSON

```
1 {
2   "comentario": "Muy acertada la reseña, felicidades",
3   "reseña_id": "1",
4   "usuario_id": "1"
5 }
```

