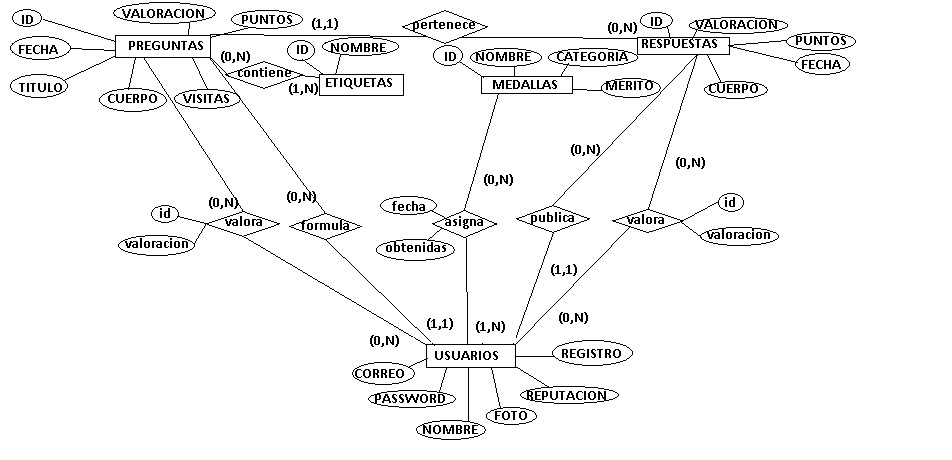
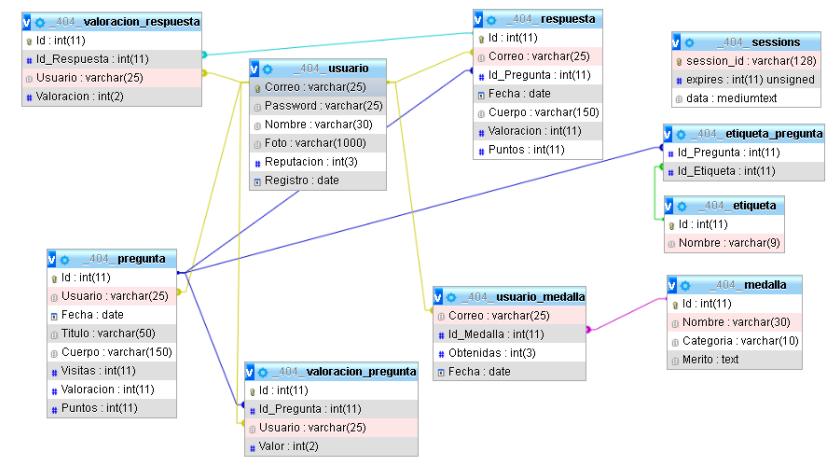
**PRESENTACIÓN**

**Diseño de la base de datos**

* **Modelo entidad-relación**



* **Modelo relacional**



**Estructura de la aplicación**

* **Controladores**
* **ControllerQuestion**

Contiene todas las funciones que se van a utilizar para las direcciones del router de preguntas (RouterQuestions).

* **ControllerSession**

Contiene todas las funciones que se van a utilizar para las direcciones del router de sesión (RouterSession).

* **ControllerUser**

Contiene todas las funciones que se van a utilizar para las direcciones del router del usuario (RouterUser).

* **Routers**
* **RouterUser**

Contiene todos los manejadores de ruta que pertenezca a la dirección /usuarios.

* **RouterSession**

Contiene todos los manejadores de ruta que pertenezca a la dirección /.

* **RouterQuestions**

Contiene todos los manejadores de ruta que pertenecen a la dirección /preguntas.

* **Modelos**
* **DAOUsers**

Contiene todas las operaciones que manipulan la bbdd para obtener información de los usuarios registrados en la plataforma.

* **DAOInteractions**

Contiene todas las operaciones que manipulan la bbdd para obtener información de las preguntas, respuestas y etiquetas. También gestiona las medallas.

* **Otros módulos**

**App.js:** aplicación

**Config.js:** guarda los datos de la BBDD

**Utils.js:** funcionalidad para la aplicación (creación de tags)

* **Views**
* **AllQuestions**
* **Error**
* **FindByTag**
* **FindByText**
* **FindUser**
* **Home**
* **Login**
* **NewQuestion**
* **Question**
* **SignUp**
* **Unanswered**
* **User**
* **Users**

Plantillas de ejs.

* **Encabezado**
* **QuestionPanel**
* **UserPanel**

Subplantillas de ejs.

* **Public**
* **Profile\_imgs**

Fotos necesarias para realizar la práctica.

* **CSS**
* **AllQuestions**
* **FindByTag**
* **FindByText**
* **FindUser**
* **Home**
* **Login**
* **NewQuestion**
* **Question**
* **SignUp**
* **Unanswered**
* **User**
* **Users**

Damos formato a los ejs correspondientes.

**Rutas gestionadas por el servidor**

* **RouterSession (/)**

**/ :** redirige al login.

**/login:** muestra y gestiona lo ocurrido en la ventana de inicio de sesión. Lleva a un formulario para poder iniciar sesión o registrarse.

**/signUp:** muestra y gestiona los eventos ocurridos en la ventana de registro. Esta ruta lleva a un formulario que rellenamos para crear la cuenta. También te da la opción de volver a la ventana de inicio de sesión.

**/logOut:** maneja la petición de cerrar sesión y redirige al login.

* **RouterUser (/usuarios)**

**/user/:correo:** se accede al pincha sobre un usuario y se encarga de cargar la página de dicho usuario o del usuario con el correo que pasamos por parámetro.

**/users:** carga la ventana con todos los usuarios registrados en la plataforma.

**/findUser:** se encarga de buscar a los usuarios que en su nombre contengan una cadena y los muestra en su propia ventana.

**/userImg:** se encarga de cargar la foto de perfil del usuario registrado.

**/rutaImg/:foto:** se encarga de generar la ruta de la foto con el nombre que se le pasa por parámetro, donde está guardada la foto del usuario.

* **RouterQuestions (/preguntas)**

**/home**: carga la página principal.

**/allQuestions:** carga todas las preguntas registradas en la BBDD, las ordena por fecha y las muestra en su propia ventana.

**/unanswered:** carga en su ventana todas las preguntas que todavía no tienen respuesta.

**/question/:questId:** carga la información detallada de una pregunta, incrementa su número de visita y asigna una medalla si es necesario.

**/findByTag/:tagId**: carga en su ventana todas las preguntas que contengan el tag proporcionado como parámetro.

**/newQuestion:** carga un formulario para poder publicar una pregunta.

**/findByText:** carga en su ventana todas las preguntas que contengan en su título o en su cuerpo una cadena proporcionada por el usuario.

**/answer/:Id:** asigna a una pregunta una respuesta perteneciente al usuario registrado.

**/likeP/:Id:** comprueba si el usuario ha votado esa pregunta y, en caso de no haberla votado, registra ese voto y asigna una medalla en caso de que la pregunta haya alcanzado el número de puntos.

**/dislikeP:Id:** comprueba si el usuario ha votado una pregunta y en caso de no haberla votado registra su voto.

**/likeR/:Id:** comprueba si el usuario ha votado esa respuesta y, en caso de no haberla votado, registra ese voto y asigna una medalla en caso de que la respuesta haya alcanzado el número de puntos.

**/dislikeR:Id:** comprueba si el usuario ha votado una respuesta y en caso de no haberla votado registra su voto.

**Método utilizado para restringir el acceso a las rutas para los usuarios no logueados**

**Middleware de control**: se encuentra en los routers RouterUser y RouterQuestions. Si detecta que el usuario está logueado, redirige a la función solicitada; en caso contrario, te manda al login.

**Middleware iniciado:** se encuentra en el RouterSession. Si detecta que el usuario ya tiene una sesión iniciada, no deja acceder ni al login ni al signUp. En caso contrario, te deja acceder a las rutas.

**Gestión de los errores 404 y 500**

Son dos middleware que se llaman en el controller y cuando dan error van a esos middleware. En aquellos lugares donde puedan saltar esos errores (generalmente en los de BBDD, sobre todo cuando no se conecta o falla en servidor), se manda ese error al middleware correspondiente y se carga la información del error en la ventana Error.ejs.

**Listado de los paquetes utilizados**

**EJS**

Nos proporciona las plantillas y las funciones para poder cargar la información que nos manda la aplicación en el correspondiente html.

**BODY-PARSER**

Desde una petición tipo post actualizamos o creamos una entidad.

**EXPRESS**

Nos permite utilizar el router y la aplicación.

**MYSQL**

Para gestión de bases de datos.

**PATH**

Genera los directorios dentro del proyecto.

**EXPRESS SESSION y EXPRESS MYSQL SESSION**

Encargados de generar la información de la sesión actual y almacenarla en la base de datos.

Alejandro Ruiz Martín

Verónica Calzada Álvarez

4ºE Ingeniería del Software AW