PROYECTO FINAL

Programación Web I – diciembre 2022

Descripción

Profesor: Fazzini, Santiago Oscar Nombre Aplicación: RottenApples

Tiempo estimado de desarrollo: 3 semanas

Institución Académica: Universidad de Ciencias Económicas y Empresariales

CONTENIDO

- Presentación del proyecto 2
- Tecnologías Utilizadas 3
- Distribución de la aplicación 4
- Diagrama Lógico de la Base de Datos 5
- Diagrama de los componentes 6
- Documentación 7

Presentación del proyecto

El proyecto consiste en una aplicación web para poder publicar reviews (opiniones) de distintas películas. Estas reviews se van a poder compartir con distintos usuarios que se encuentren registrados en la plataforma. Las principales características del proyecto son las siguientes:

- 1. El **usuario** va a poder registrarse e iniciar sesión.
- 2. El **usuario** va a poder publicar su opinión de las películas vio.
- 3. El **usuario** va a poder eliminar y modificar las reviews que publico.
- 4. El **usuario** va a poder ver las reviews de todos los otros usuarios.
- 5. El **usuario** va a poder seguir otros usuarios.
- 6. El **usuario** va a poder ver las reviews de los otros usuarios que sigue.
- 7. El **usuario** va a poder bloquear a los usuarios que lo siguen.
- 8. Los **administradores** van a poder hacer lo mismo que usuario normal.
- 9. Los administradores van a poder eliminar películas, usuarios y reviews.
- 10. Los **administradores** van a poder agregar nuevas películas al sistema.
- 11. Los administradores no van a poder eliminar a otros administradores.
- 12. Los **administradores** no van a poder editar las películas.
- 13. Siempre va a tener que quedar al menos una película registrada en el sistema.
- 14. No se van a poder registrar 2 email idénticos.
- 15. No se van a poder registrar 2 nombres de usuarios idénticos.
- 16. No se puede crear una película que tenga el mismo nombre, que una película ya registrada.
- 17. El listado de las películas se va a obtener a través de cierta información que se encuentra registrada en la base de datos.
- 18. Las contraseñas de todos los usuarios se van a encontrar hasheadas, para resguardar la seguridad de todas las cuentas.

Tecnologías Utilizadas

- HTML5
- CSS
- Bootstrap V5
- Angular Js
- PHP
- MySQL
- JWT
- NPM Paquetes (SweetAlert2 / Axios)

Distribución de la aplicación

La aplicación web va a estar compuesta por los siguientes componentes:

- **RegistroComponent**: En este componente el usuario va a poder registrarse en la aplicación web, e iniciar sesión.
- **GlobalComponent**: En este componente se van a ver las reviews de todos los usuarios, sin importar si el usuario registrado lo sigue o no.
- **ReviewsComponent**: En este componente solo se van a ver las reviews de las cuentas a las que el usuario registrado está siguiendo.
- **MisReviewsComponent**: En este componente el usuario registrado va a poder publicar, editar y eliminar reviews sus propias reviews.
- **MiPerfilComponent**: En este componente el usuario registrado va a poder seguir a otras cuentas. Además, va a poder ver que usuarios lo están siguiendo, de forma de que si lo desea va a poder bloquear a los mismos.
- AdministradorComponent: En este componente, cuando el usuario registrado tiene los privilegios de administrador, va a poder eliminar a los usuarios, reviews y películas que desee.
 - AgregarPeliculaComponent: En este componente, que forma parte de AdministradorComponent, el administrador va a poder registrar nuevas películas en la base de datos.
- HomeCompontent: Pantalla principal de la aplicación web.
- **FooterComponent**: Pie de pagina de la aplicación web, donde se comparte información de esta.
- HeaderComponent: Barra de navegación, que le va a permitir al usuario cambiar de rutas.

Para una mejor comprensión, se adjunta en la página 6 un diagrama con los componentes y sus respectivas métodos y atributos.

Diagrama Lógico de la Base de Datos

DIAGRAMA LÓGICO

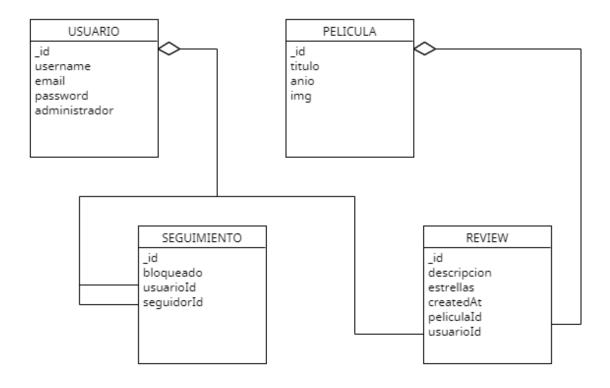


Diagrama de Componentes

DIAGRAMA DE COMPONENTES

- getReviewSeguidoresConParametros() - getUsuario() getReviewConParametros()+ resetFormulario() mis-reviews reviews + postReview() + deleteReview(id) + patchReview(review) + miFormulario getUsuario() getPelicula() + reviews + peliculas + usuario header + jwtHelper + iseExpanded + usuario + logOut() - getPeliculaTituloConParametros(titulo) + resetFormulario() + postPelicula agregar-pelicula administrador getReview()
getPelicula()
getCantidadPeliculas()
deleteUsuario(id)
deletePelicula(id)
deletePelicula(id)
deletePelicula(id) getUsuarioNormal() + reviews + peliculas - catnPeliculas + miFormulario + anioActual getUsuario() + usuarios resetMiFormulario()
 getUsernameConParametros(username)
 getEmailConParametros(email)
 login()
 postUsuario() - getSiguiendo.ConParametros()
- getSeguidoresConParametros()
+ getBuscarlusuario.ConParametros()
+ pastSeguimiento()
+ patchSeguimiento()
+ patchSeguimientoDesbloquear() mi-perfil registro + usuariosEncontrados · resetMiFormulario() + miFormulario2 + miFormulario + miFormulario + credenciales + siguiendo + seguidores getUsuario

6

Bibliografía

- Documentación compartida en las clases de Programación Web I
- Angular https://angular.io/docs
- Axios https://axios-http.com/docs/intro
- SweetAlert2 https://sweetalert2.github.io/#download