



Documentación TareApp

TareApp es una aplicación de escritorio creada para poder administrarse las tareas, permite poder crear, editar, borrar y ver las tareas.

Documentación accesible desde: [Enlace](#)

Componentes propios

Se han creado muchos componentes propios, la mayoría son componentes para simplificar repetir los mismos estilos en componentes ya existentes como botones, inputs, selects...

Los más destacados son:

Cabecera

Este componente, se ha creado ya que la cabecera está en todas las páginas y es un componente que tiene mucho código ya que se encarga del idioma y de navegar por las demás páginas, para ello se crea este componente para evitar tener todo el código de la cabecera en el fichero Main, de esta forma queda todo más estructurado.

PopUps

He creado componentes para todos los popUps ya que son como otras vistas aparte y tienen mucho código en su interior, de esta manera el código está más separado y estructurado y así cada vez que quiero genera un nuevo PopUp solo tengo que instanciar de nuevo la clase permitiendo que todo sea más limpio y eficiente.

Componentes de tarea y lista

Estos componentes me permiten crear la estructura de las listas y tareas que se van a mostrar en los paneles de sus respectivas páginas, estos componentes se van a repetir muchas veces por lo tanto tengo que estar generando el mismo código varias veces por cada lista o tarea que se cree, para ello una forma de simplificar el código y hacerlo más eficiente es crear un componente para que solo instanciando la clase ya tenga toda la interfaz de la lista o de la tarea creada.

Partes extras del proyecto

Archivo de estilos y tipo de letra

He creado un fichero donde se encuentran todos los colores y donde se recogen las fuentes de la página, esto es útil ya que si se quieren cambiar los colores de la aplicación, solo hay que cambiar los colores de las variables de ese archivo, además he usado un tipo de letra externo, descargando la fuente Poppins.

Cifrado de contraseñas

He utilizado la clase `BCryptPasswordEncoder` para poder cifrar las contraseñas creando un Hash de 60 caracteres que no puede ser descifrado, añadiendo así una seguridad a los datos del usuario.

Modelo vista controlador

He usado el modelo vista controlador para ordenar mejor el proyecto separando la lógica de la parte del diseño y añadiendo controladores que se encargan de comunicar las vistas con los modelos.

Prueba unitaria

Todo el proyecto está comprobado, se han realizado pruebas tanto unitarias como de integración en todas las partes del proyecto para comprobar su funcionamiento, esto me ha permitido encontrar errores que ya han sido corregidos, pero para la prueba unitaria creada en el proyecto en el apartado de Test, ha sido para comprobar el registro de la aplicación ya que son varias combinaciones que se tienen que probar como:

- Que el usuario no se registre con un email que ya existe
- Que no contengan caracteres extraños el email y la contraseña
- Que no estén vacíos
- Que se cumpla el formato de la contraseña y del email.