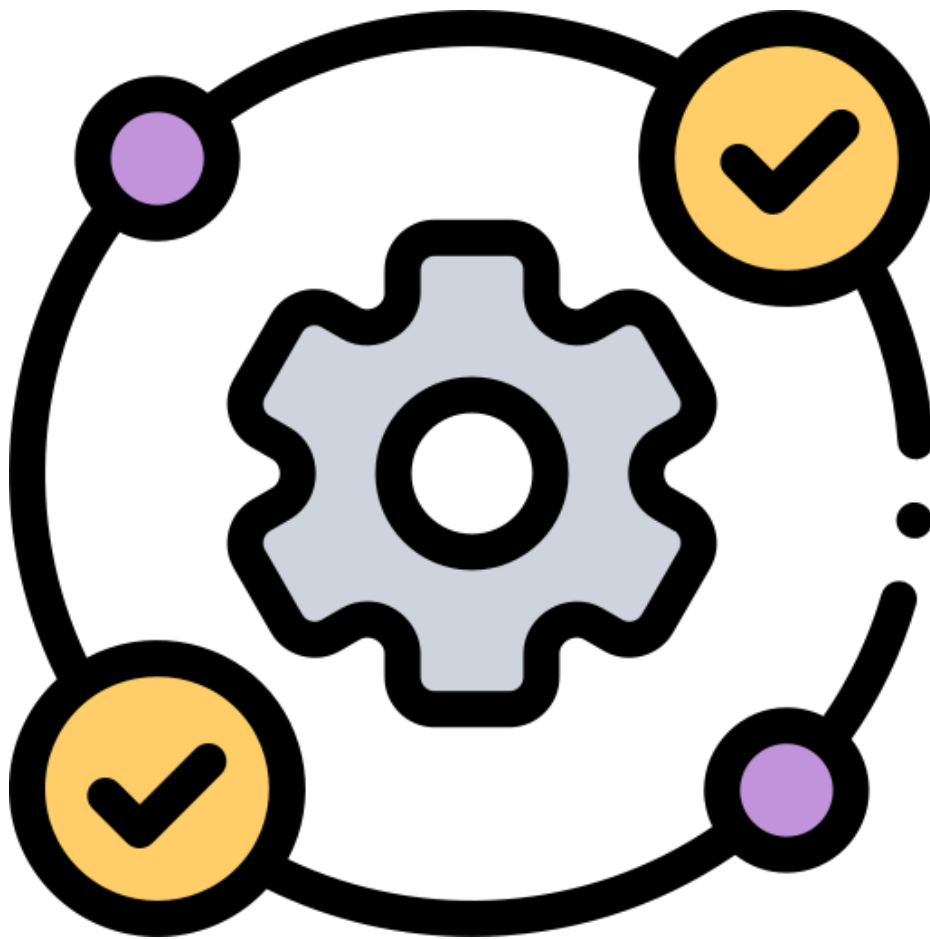


Grado en Ingeniería Informática
Desarrollo de Aplicaciones Web II
Ingeniería Inversa
Curso 2021-2022
03/05/2022



Joaquín Pastore Barrios

ÍNDICE

Introducción	2
Desarrollo	3
Estructura del directorio app/components	3
Estructura del directorio app/guards-auth	5
Estructura del directorio app/modules	6
Estructura del directorio app/services	10
Conclusión	11
Bibliografía	12

1. Introducción

El objetivo principal de esta actividad es realizar el proceso de ingeniería inversa sobre una aplicación proporcionada, elaborando un informe detallando el proceso realizado.

Este informe abarca aspectos como la estructura de la aplicación estudiada, en particular el directorio fuente, así como los elementos que componen dicha aplicación, como componentes, servicios, entre otros

Una vez realizada la observación se realizó la implementación de una aplicación que gestiona la plantilla de un equipo deportivo tomando como base la aplicación dada, de manera que el punto de partida fue esta “plantilla” en el propósito que el estudiante se desarrolla en entornos donde deben realizar un estudio previo porque existen trabajos previos en los que no han participado y que deben comprender para desempeñar sus funciones.

2. Desarrollo

Para comenzar el desarrollo del informe, el directorio `src` proporcionado se integró en una aplicación recién creada. Una vez integrados, había que instalar paquetes como Bootstrap y FontAwesome, ya que sin ellos no sería posible ejecutar la aplicación.

Dentro del directorio `src`, nos centraremos en las estructuras y elementos que componen el directorio de la aplicación.

El directorio de la aplicación contiene las rutas a los diferentes componentes del sitio web y el archivo de enrutamiento que contiene los archivos principales del componente de la aplicación.

Estructura del directorio *app/components*

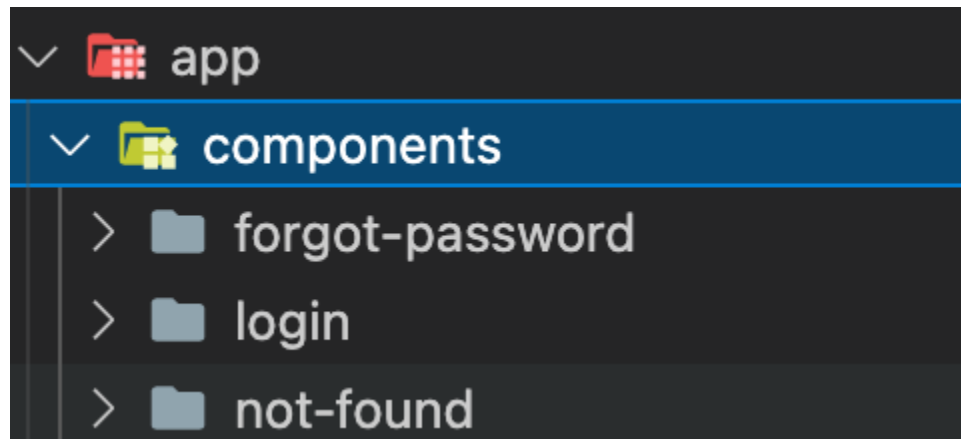
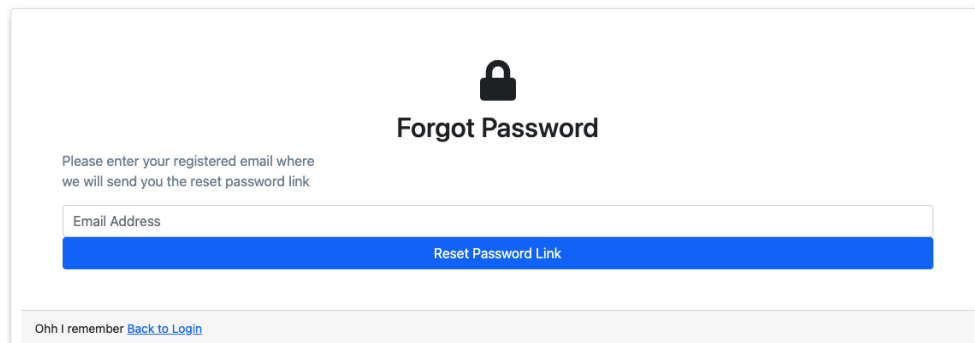


Figura 1: estructura del directorio *app/components*

Como podemos observar, existen tres componentes dentro del directorio *app/components* que se corresponden con el **forgot-password**, **login** y **not-found**.

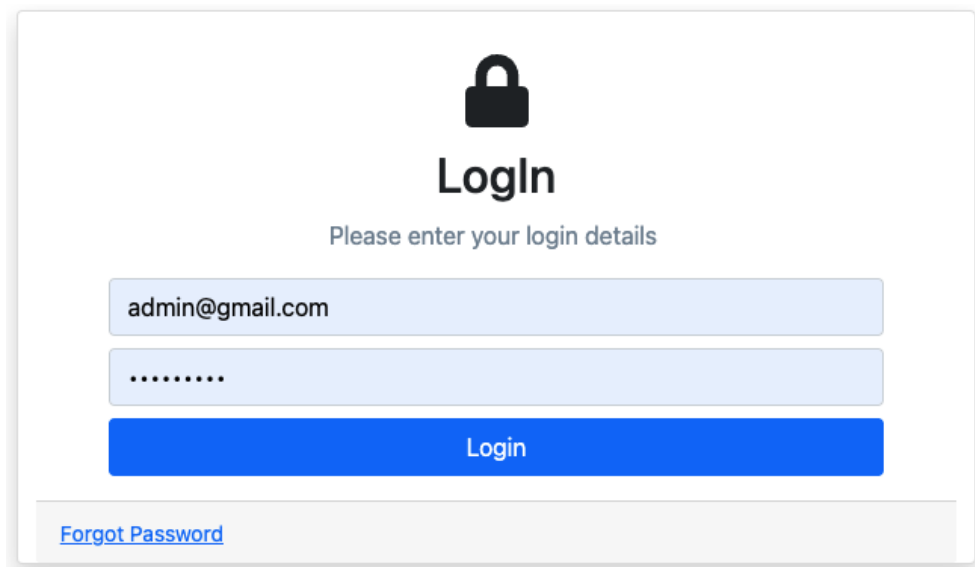
El componente **forgot-password** tiene el objetivo de permitir al usuario recuperar la contraseña en caso de pérdida. Para ello, muestra un formulario que permite introducir el correo electrónico del usuario para poder restablecerla (Figura 2).



The screenshot shows a web form titled "Forgot Password" with a lock icon above the title. Below the title, there is a message: "Please enter your registered email where we will send you the reset password link". There is a text input field labeled "Email Address" containing the text "admin@gmail.com". Below the input field is a blue button labeled "Reset Password Link". At the bottom of the form, there is a link that says "Ohh I remember [Back to Login](#)".

Figura 2: componente forgot-password

El componente **login** (figura 3) es aquel que permite que los usuarios puedan acceder a la aplicación haciendo uso de sus credenciales. Además, contiene el enlace correspondiente al componente **forgot-password** descrito anteriormente.



The screenshot shows a web form titled "Login" with a lock icon above the title. Below the title, there is a message: "Please enter your login details". There are two text input fields: the first one contains "admin@gmail.com" and the second one contains ".....". Below the input fields is a blue button labeled "Login". At the bottom of the form, there is a link that says "[Forgot Password](#)".

Figura 3: componente login

Finalmente, el último componente se llama **not-found** y su objetivo es mostrar un mensaje de error al usuario cuando trata de acceder a una ruta que no esté registrada, ofreciendo la posibilidad de volver a la página principal (figura 4).

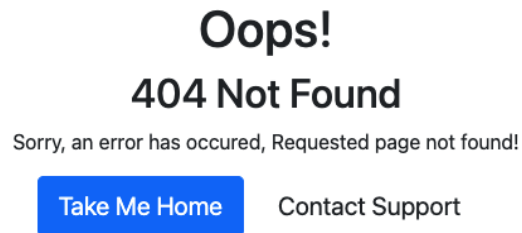


Figura 4: componente **not-found**

Estructura del directorio *app/guards-auth*

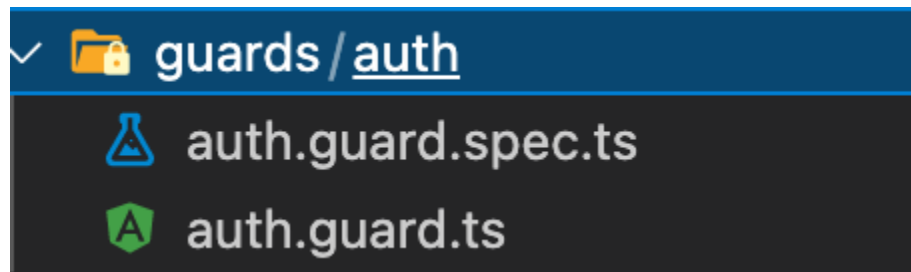


Figura 5: estructura del directorio *app/guards-auth*

Este directorio corresponde a un guard, que es básicamente una interfaz que te permite proteger las rutas para que solo los usuarios con permiso puedan acceder a ellas, por lo que si un usuario no tiene los privilegios necesarios, la acción a realizar en consecuencia, redirigirlos a otra página o realizar las acciones especificadas en el archivo.

Estructura del directorio *app/modules*

Este directorio se divide en dos grandes directorios: *admin* y *user*. El directorio del administrador contiene los componentes y archivos de enrutamiento y módulo correspondientes a los usuarios con privilegios administrativos, y el directorio de usuario contiene los archivos de módulo y enrutamiento correspondientes a otros usuarios.

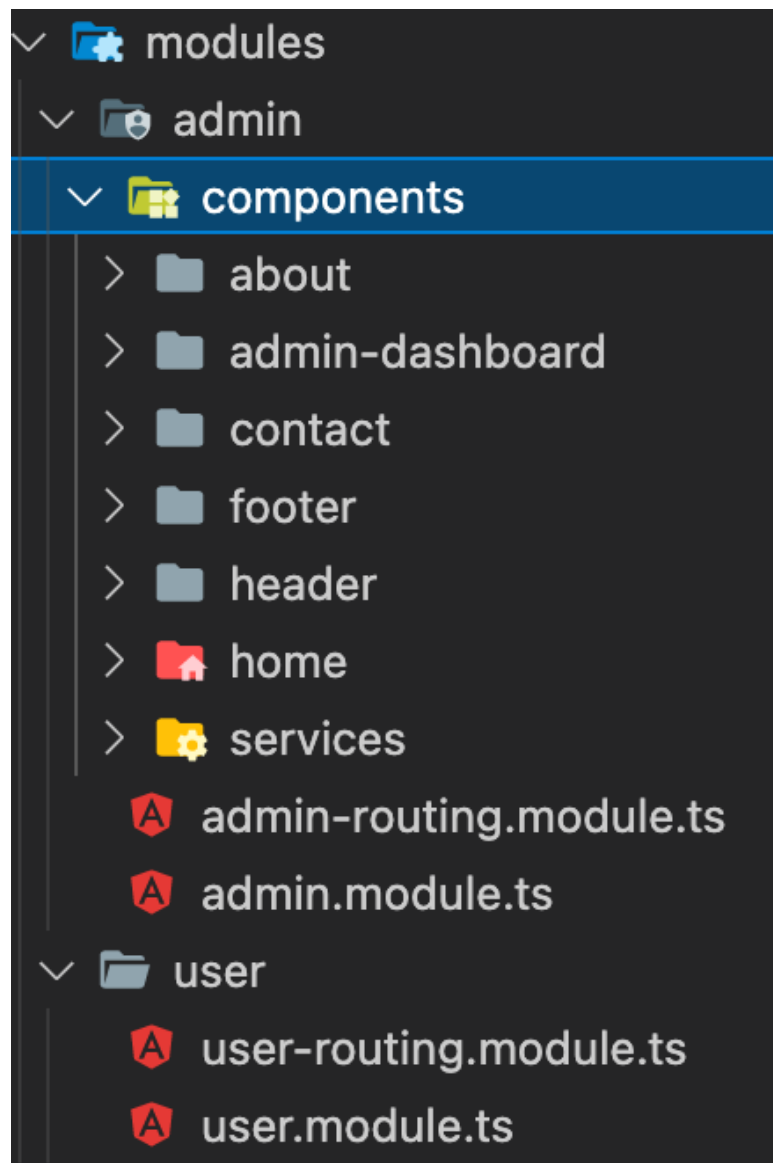


Figura 6: estructura del directorio *app/modules*

El directorio *admin/components* contiene los componentes accesibles a los usuarios administradores, que son:

- **about:** Contiene elementos de ejemplo pero el objetivo principal sería la de ofrecer información sobre la empresa, como la típica página 'Sobre nosotros'. Se ubica en la ruta */admin/about*.

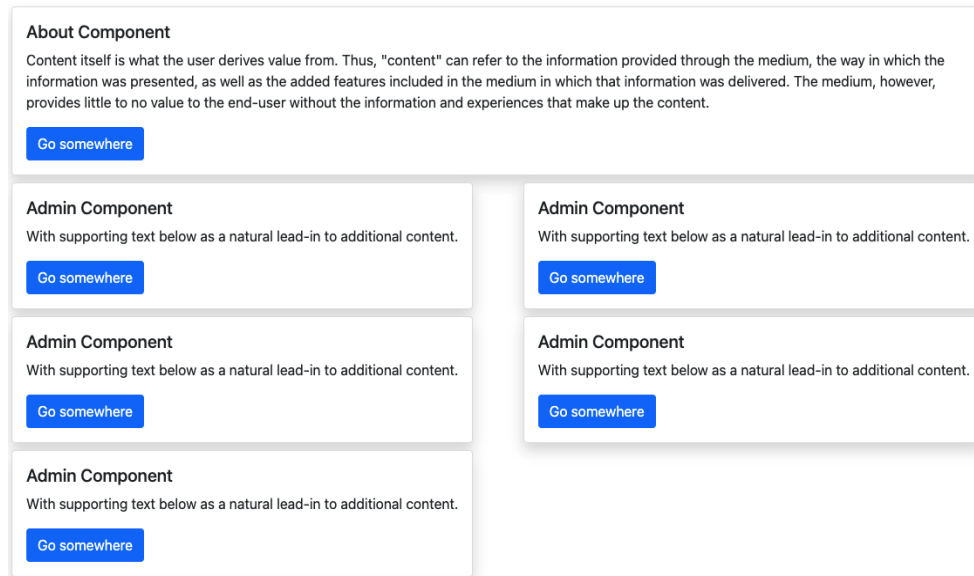


Figura 7: componente about

- **admin-dashboard:** Constituye el componente principal de esta aplicación bajo la ruta */admin/home* y contiene los componentes header y footer.

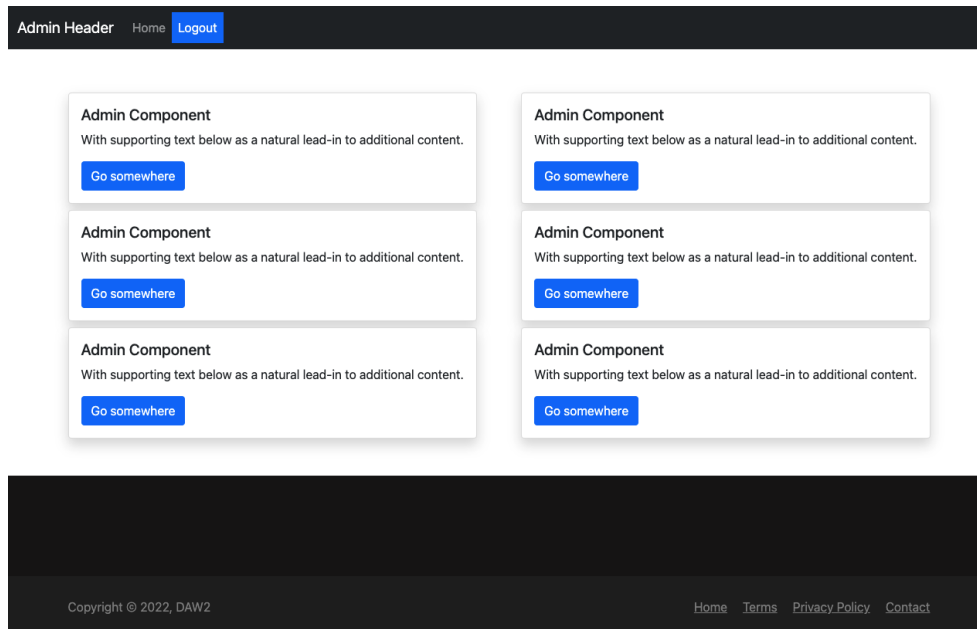


Figura 8: componente admin-dashboard

- **contact:** Es un componente cuyo objetivo es ofrecer los métodos de contacto establecidos. Se encuentra en la ruta */admin/contact*.

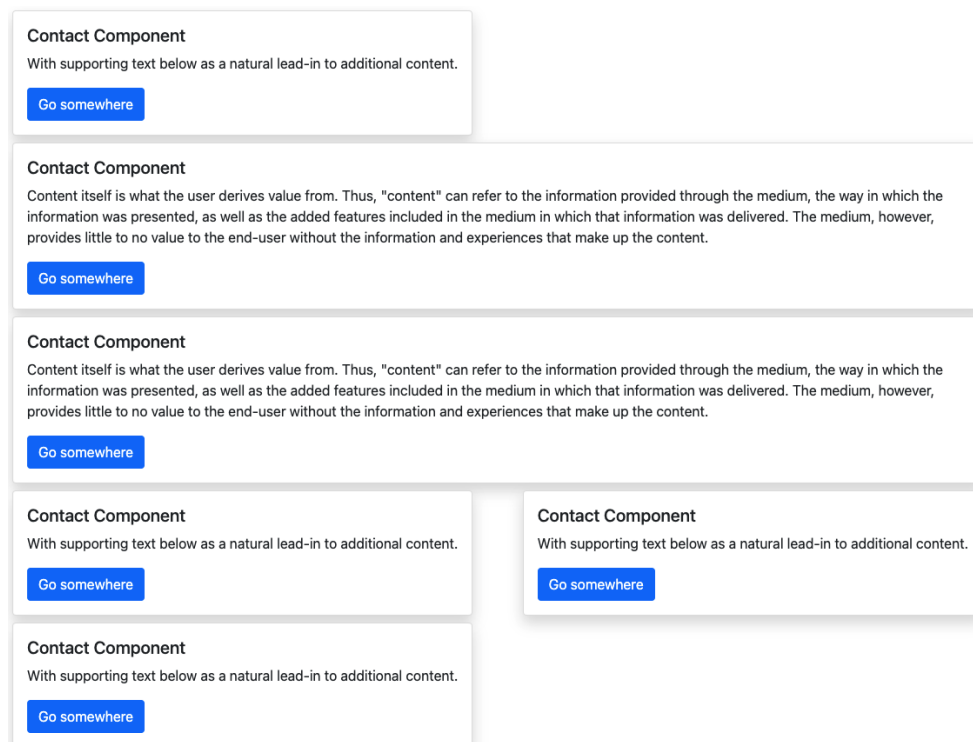


Figura 9: componente contact

- **footer:** Es el componente correspondiente al footer del sitio web.

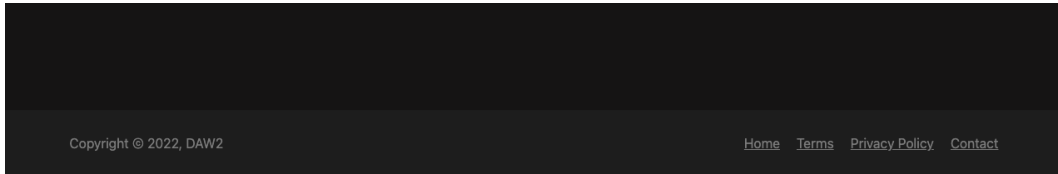


Figura 10: componente footer

- **header:** Es el componente correspondiente al header del sitio web y que contiene la barra de navegación.



Figura 11: componente header

- **home:** Componente que contiene el contenido de **admin-dashboard** descrito anteriormente.
- **services:** Constituye el componente dedicado a mostrar los servicios a los que está dedicado el sitio web. Se ubica en la ruta */admin/services*.

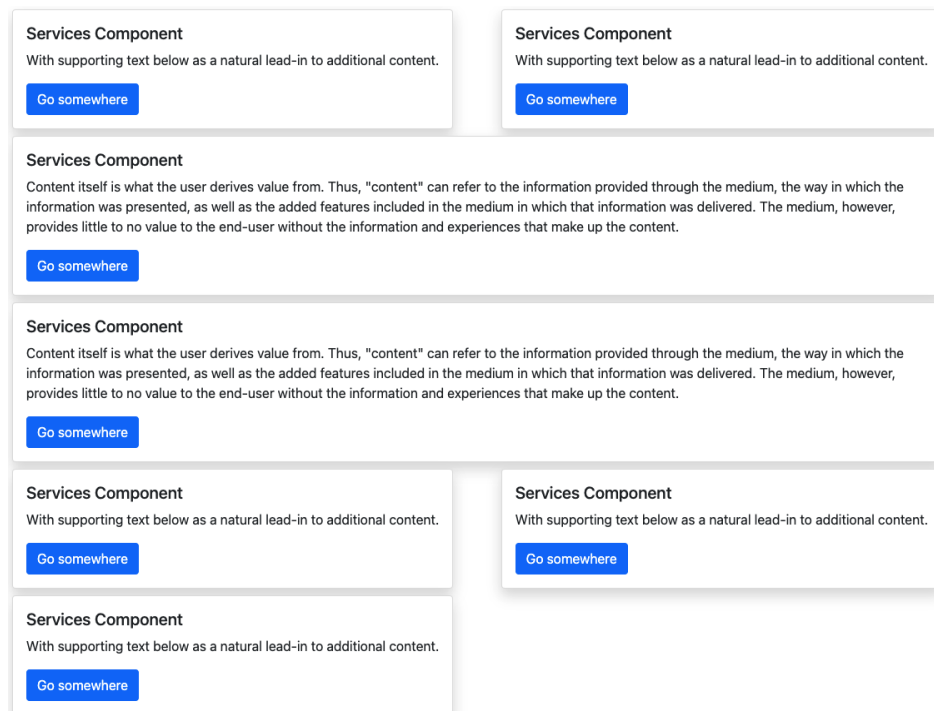


Figura 12: componente services

Estructura del directorio *app/services*

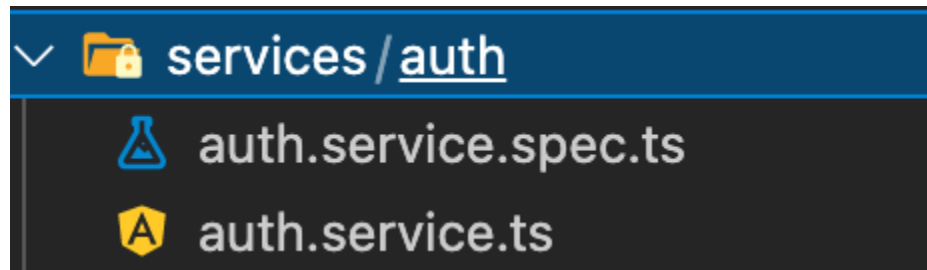


Figura 13: estructura del directorio *app/services*

Bajo este directorio se encuentra el servicio dedicado a la autenticación de usuarios a través de un token, a para lo que se han creado diferentes funciones.

3. Conclusión

En conclusión, se trata de una práctica muy interesante y útil, dado que normalmente te encuentras en el mundo laboral con proyectos que ya están parcialmente realizados, por lo que es necesario realizar un proceso de estudio de proyectos para tener una visión del conjunto.

Debido a este tipo de situaciones tan frecuentes, existe la ingeniería inversa por lo que facilita mucho este proceso y le permite brindar una perspectiva del proyecto para que pueda comenzar con las tareas específicas que se han establecido.

En este caso, la elaboración de este informe me ha facilitado mucho el desarrollo de una aplicación que tenga como base esa aplicación, por lo que creo que esta actividad me ha dado una nueva forma de analizar el problema.

4. Bibliografía

[1] Recursos Ingeniería Inversa.

Accedido en: 2022, mayo 3.

Disponible en:

https://aep22.ulpgc.es/pluginfile.php/1775843/mod_folder/content/0/src_routenavAngular_Inverse.zip?forcedownload=1

[2] Saul Lopez (2019, junio 17).

Proteger rutas con Guards en Angular.

Accedido en: 2022, mayo 3.

Disponible en:

https://www.google.com/search?q=guard+en+angular&rlz=1C5CHFA_enES993ES993&oq=guard+en+&aqs=chrome.2.69i57j0i512l5j0i22i30j69i60.3040j1j7&sourceid=chrome&ie=UTF-8