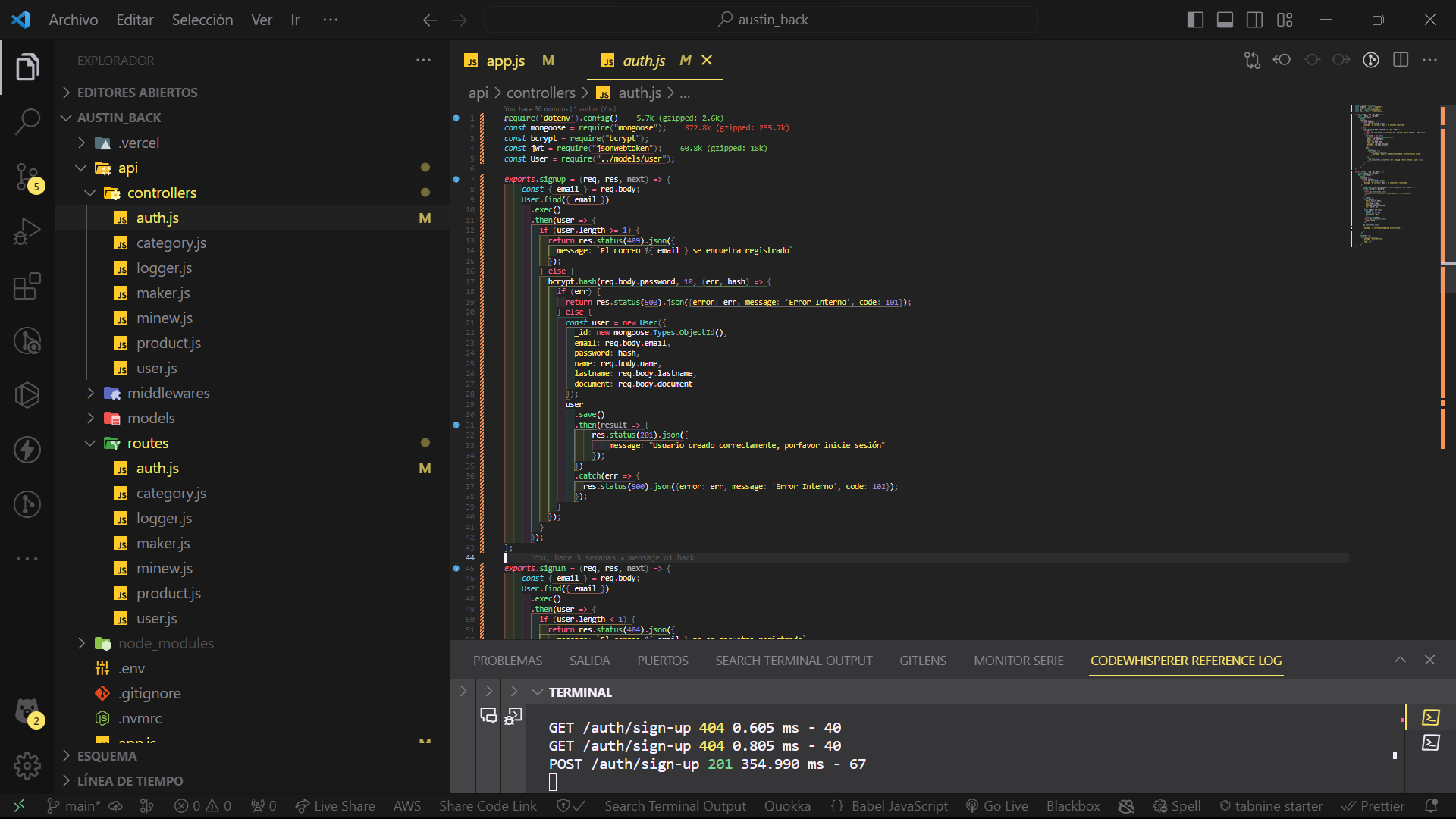
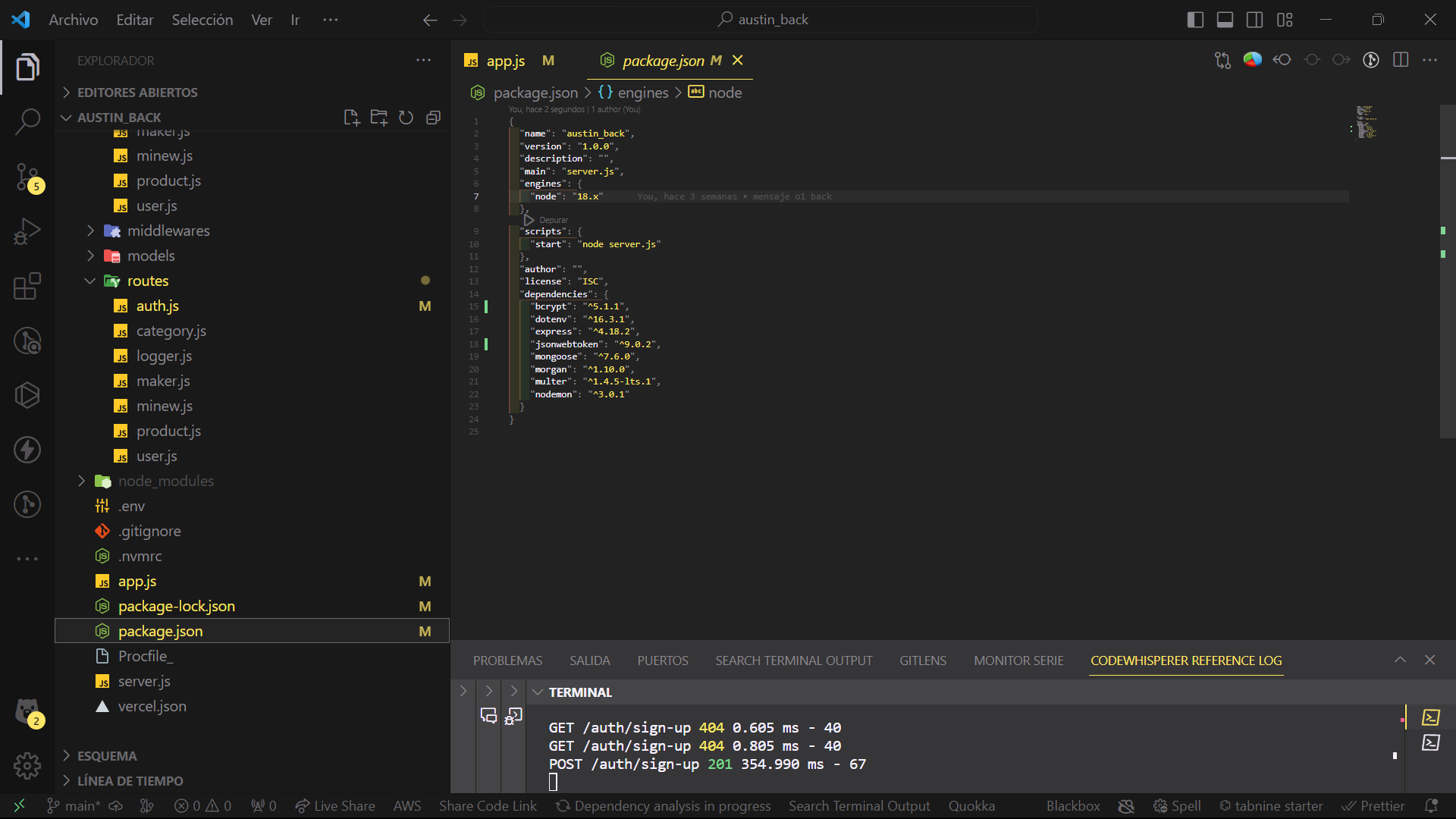
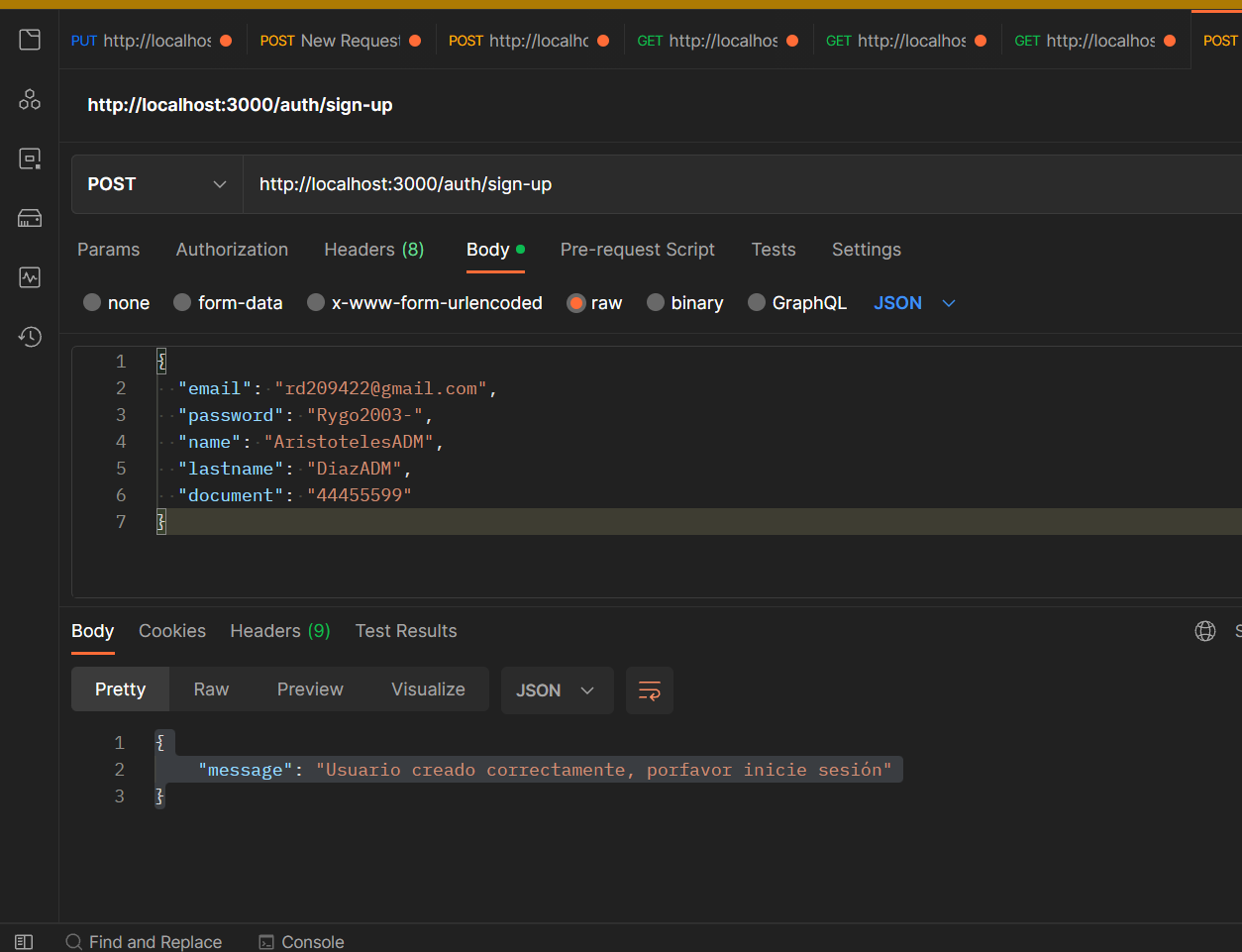
AUTH



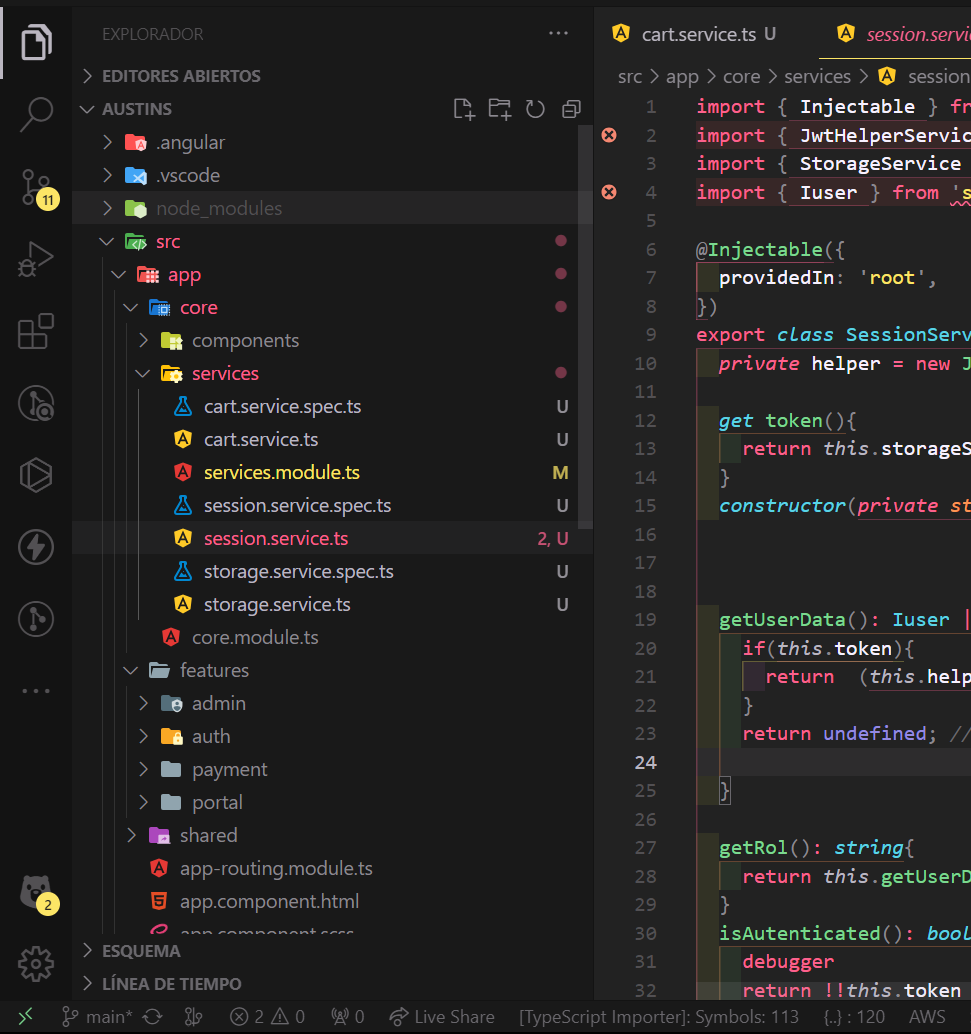
Ya implementado en la api el registro de usuario y el encriptado de la contraseña y almacenarlo en mongo en la Nube procedemos a implementar el sign-in de administrador

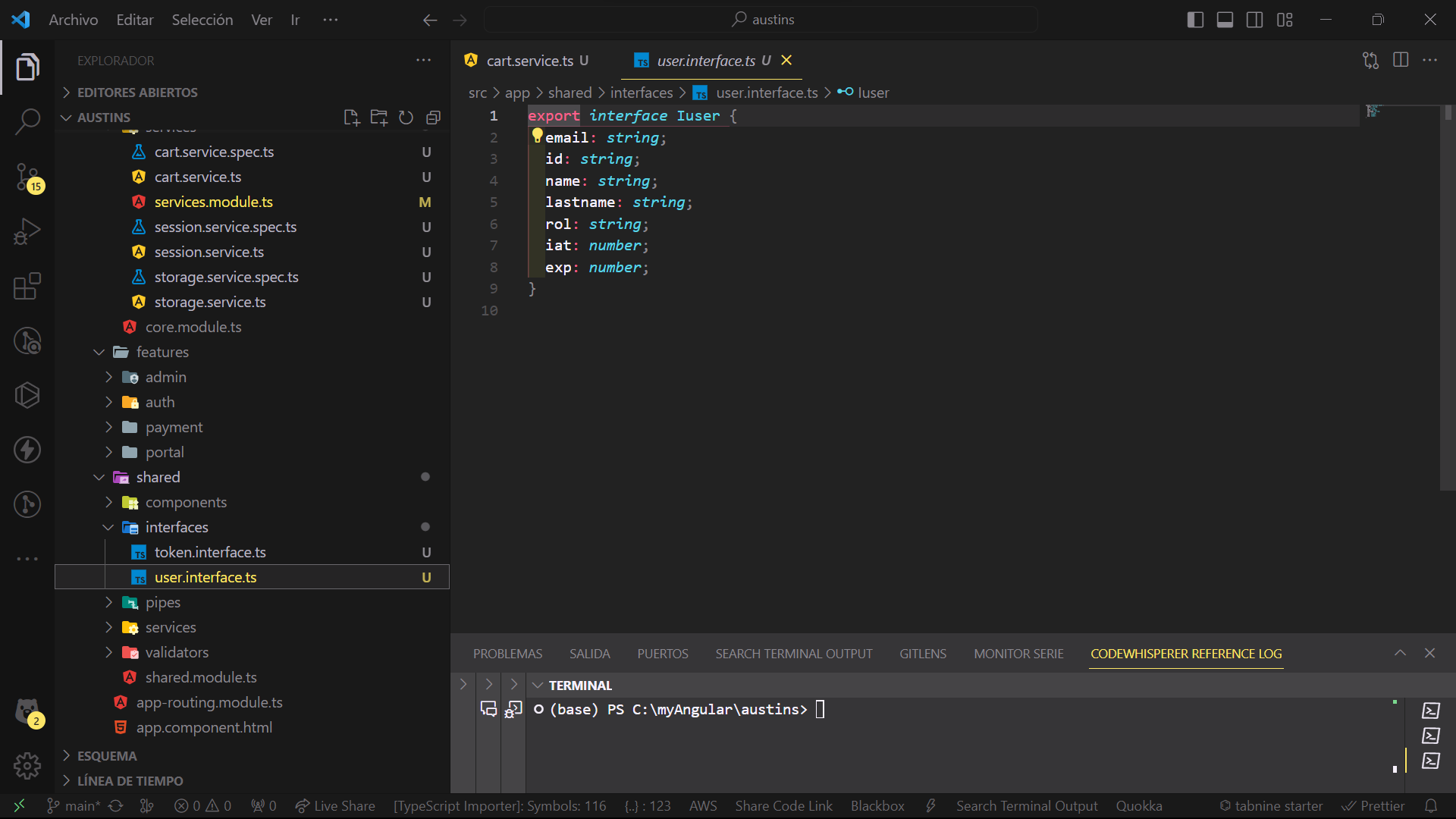


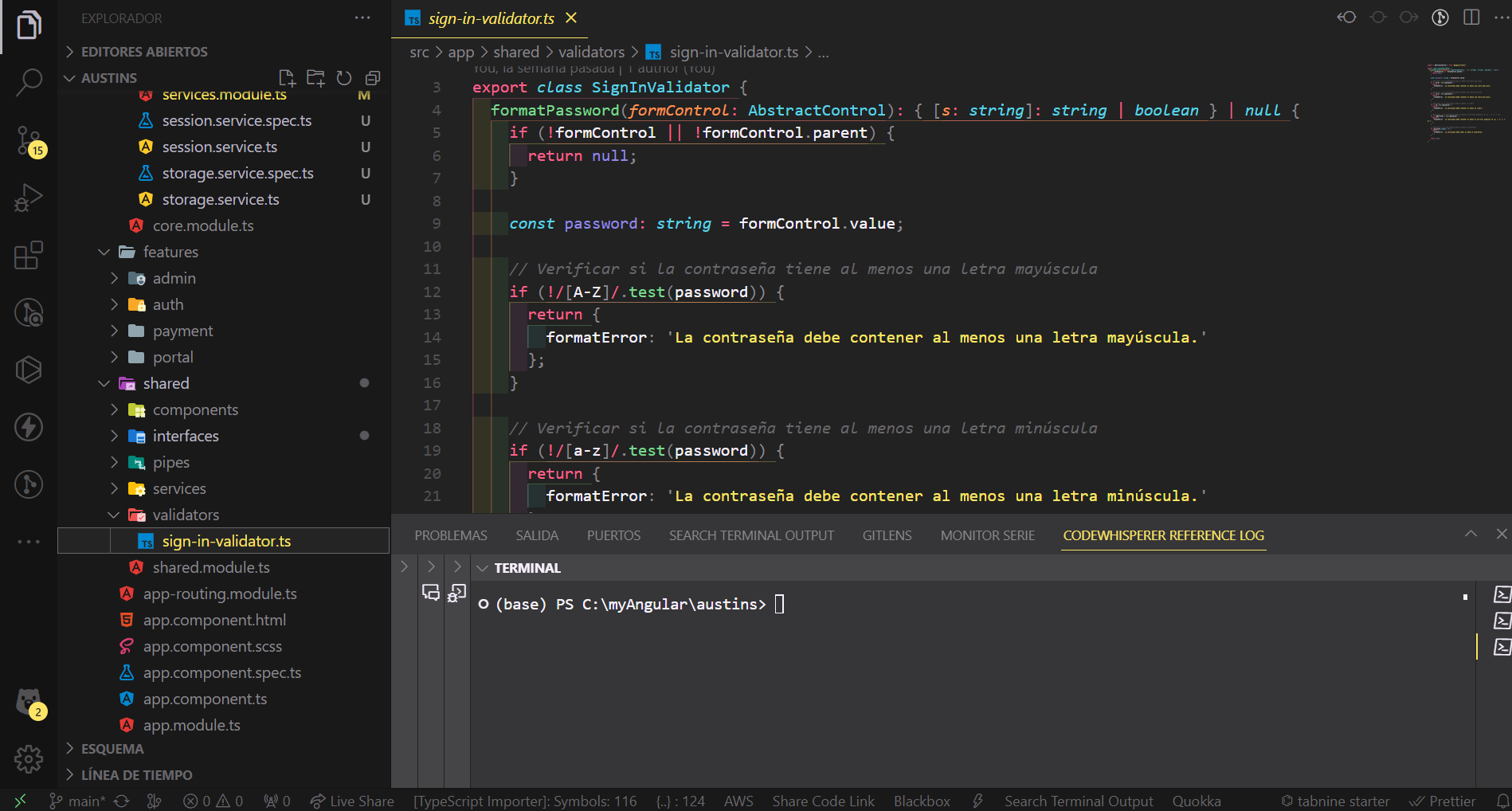
Dependencias hasta el momento en el back

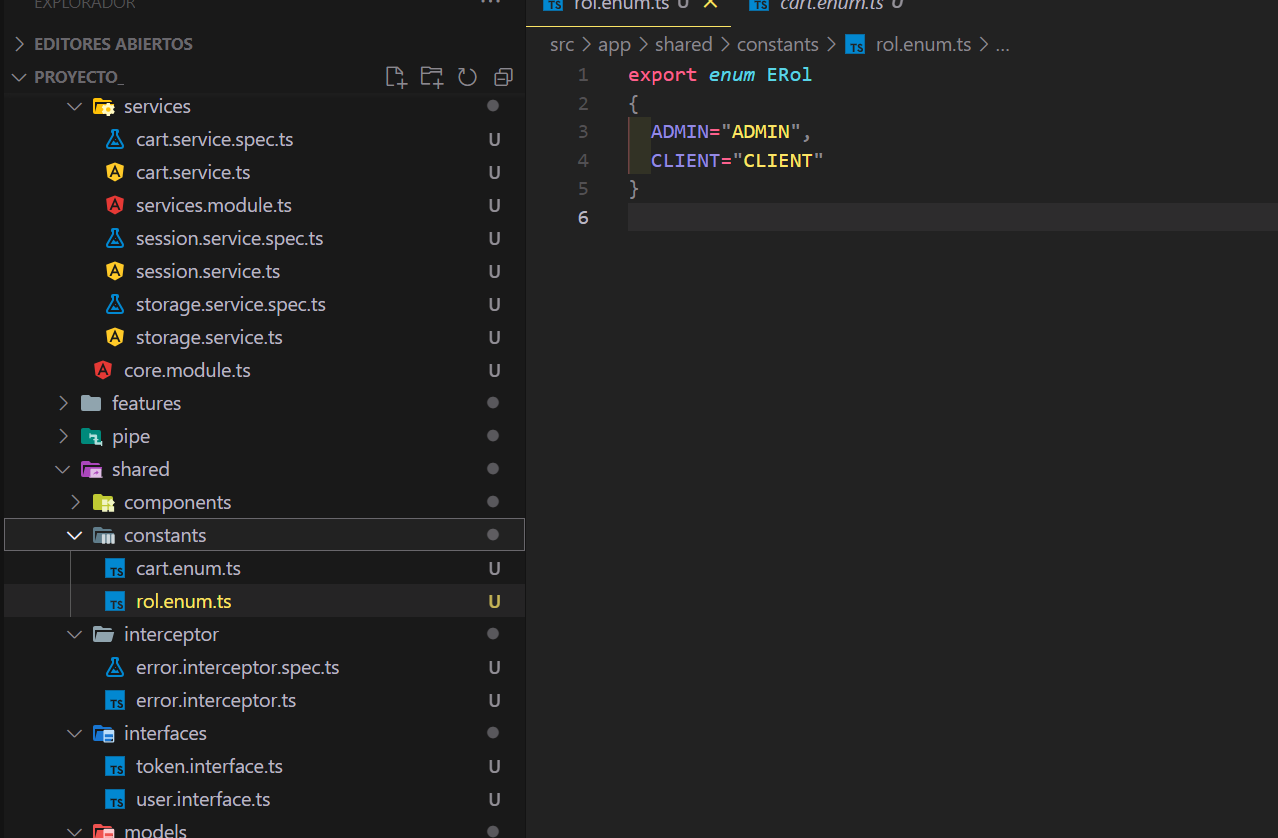


Resultado de la inserción del usuario tener en cuenta que cuando se registra de esta manera sele asigna un rol pol defaul que es del cliente

Acontinuacion se crean estos servicios mas que nada nos interesa por el momento el de sesión para crear un servicio se ejecuta ; ng g s /////   


Tambien ocuparemos interfaces para intereactuar el dato de regreso por lo tanto creamos estos archivos para el token y usuario  


Tambien tener en cuenta creado el validador de passoword y Gmail básicamente del formulario sign-in   


Adicionalmente variables de comparación   


Adicionalmente ya haber hecho de la instalación de jwt en el back de igual manera se agrega en el proyecto de angularEs importante revisar los servicios que se ocupan tanto como los validators

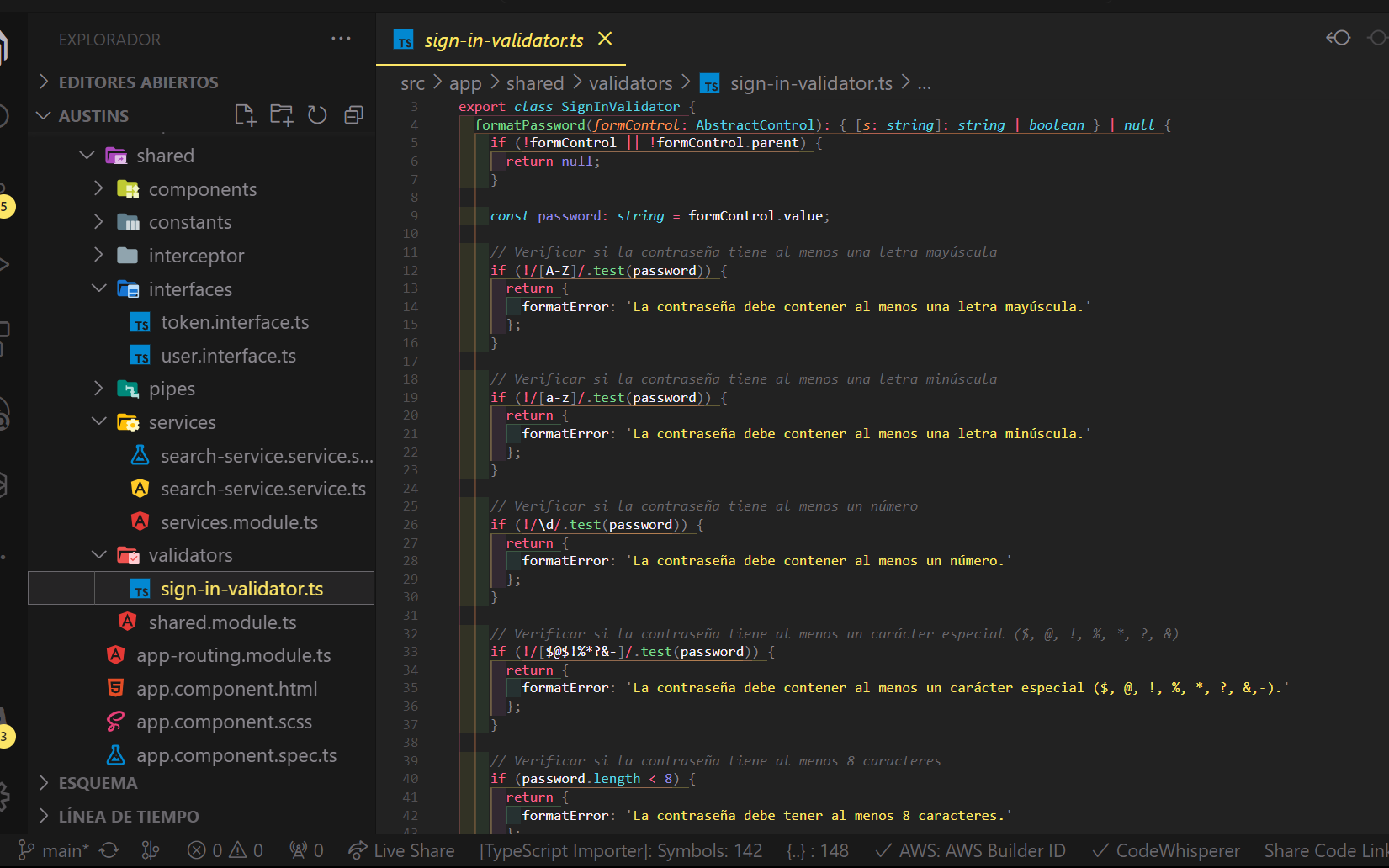
*# installation with npm*

npm install @auth0/angular-jwt

*# installation with yarn*

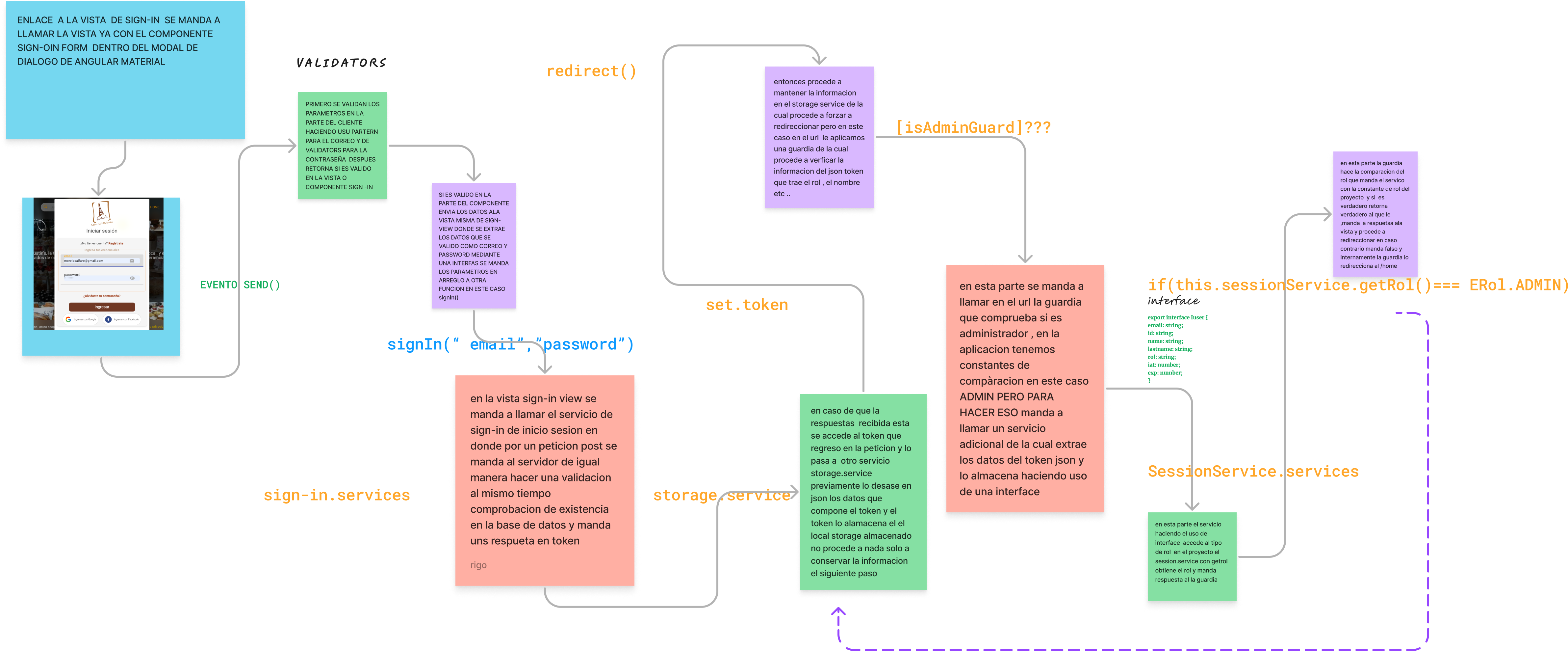
yarn add @auth0/angular-jwt

validators



Como su nombre lo indica mas que nada es como un validador de tal parámetro en este caso solo para la contraseña en esta parte podemos personalisar y hacer usu de su función en todo el proyecto esto nos servirá tanto como en el logeo y el registro pedir la contraseña de manera personalizada

Lógica de autentificación



n