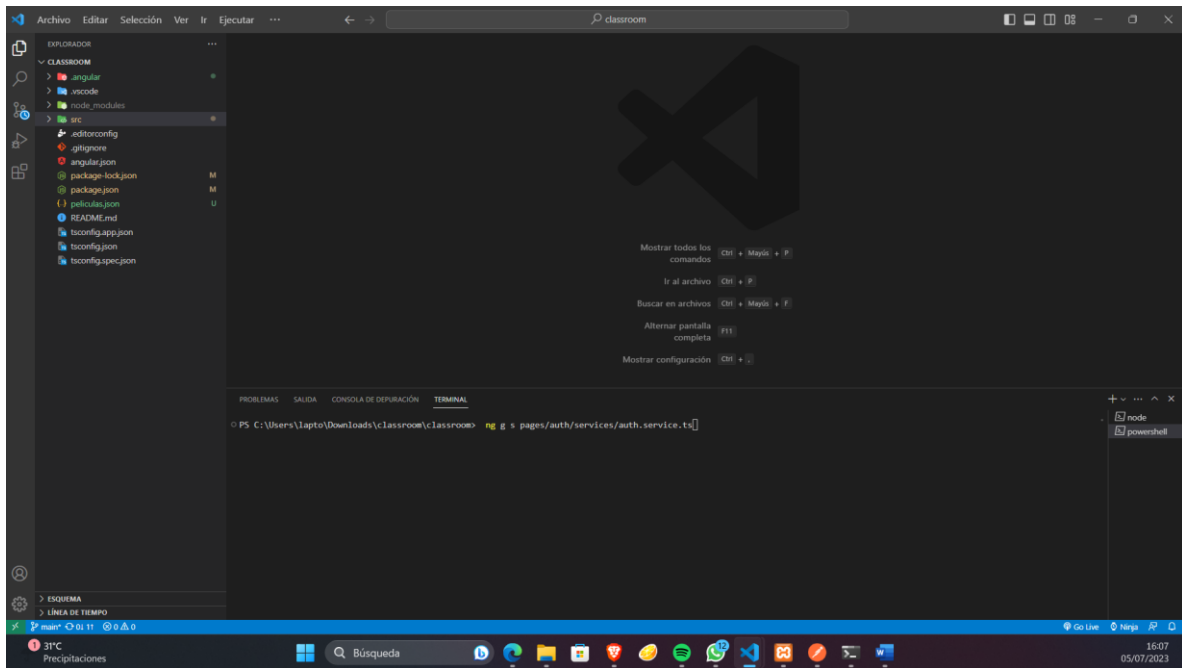
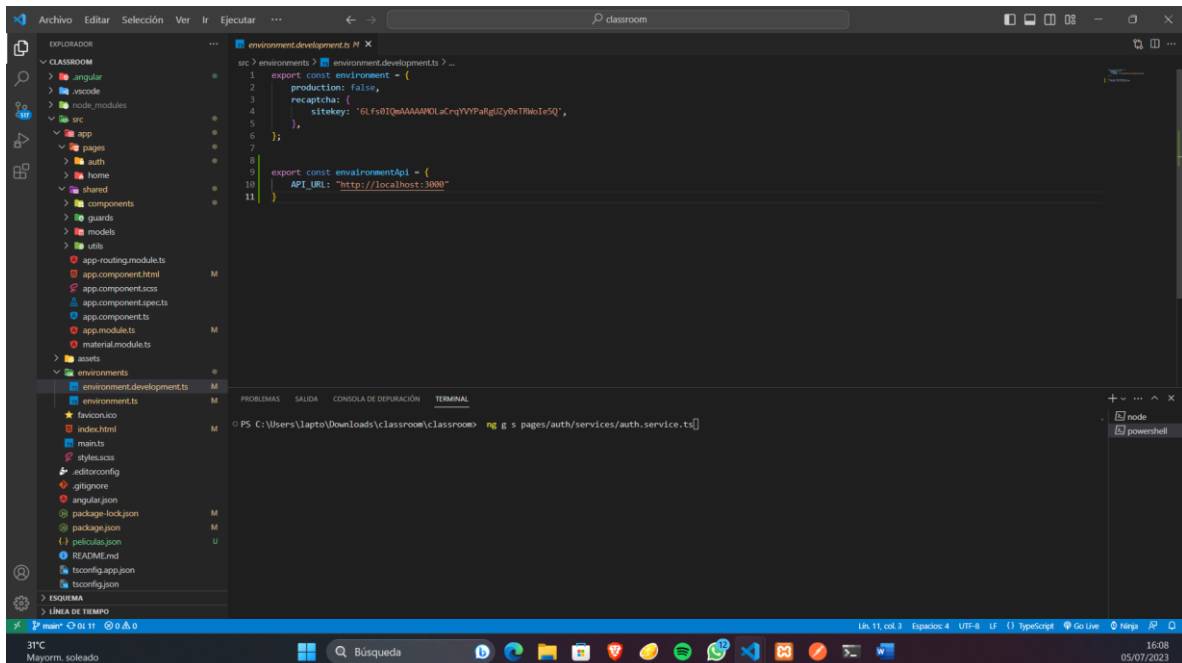


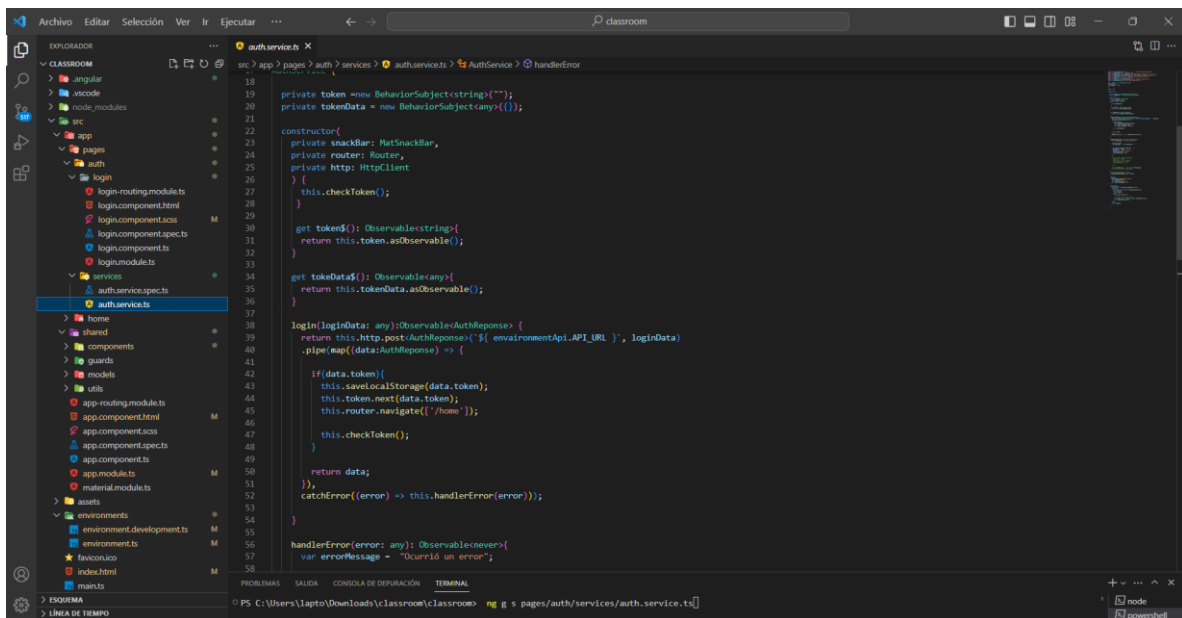
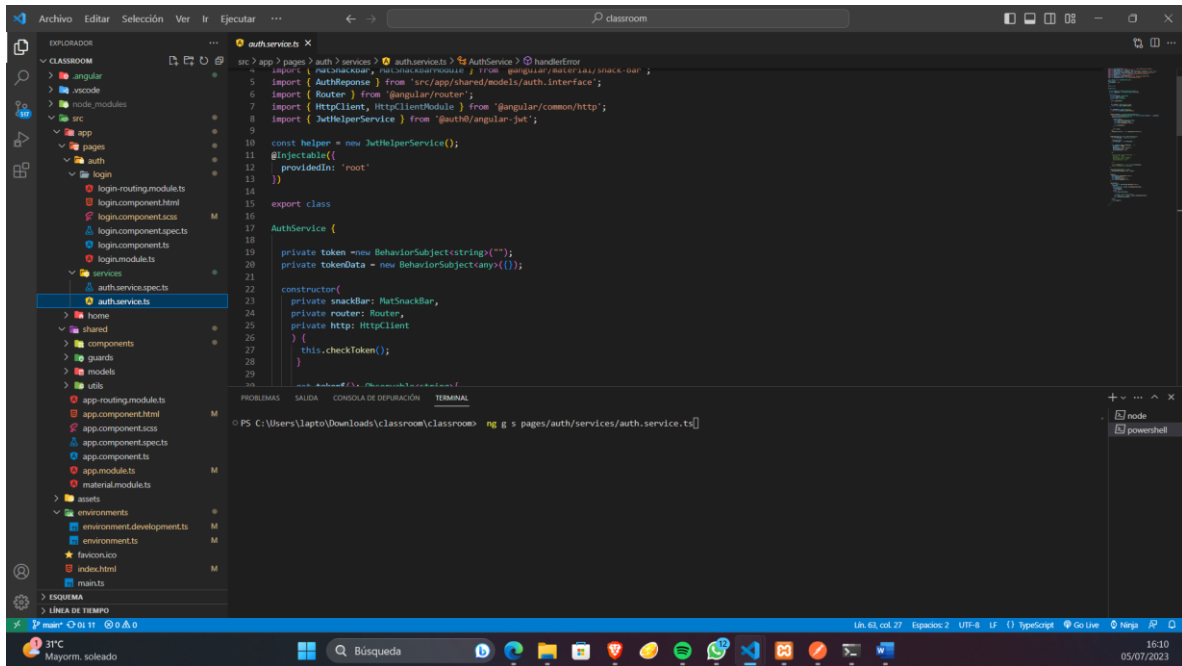
## Iniciando nuestra aplicación classroom angular

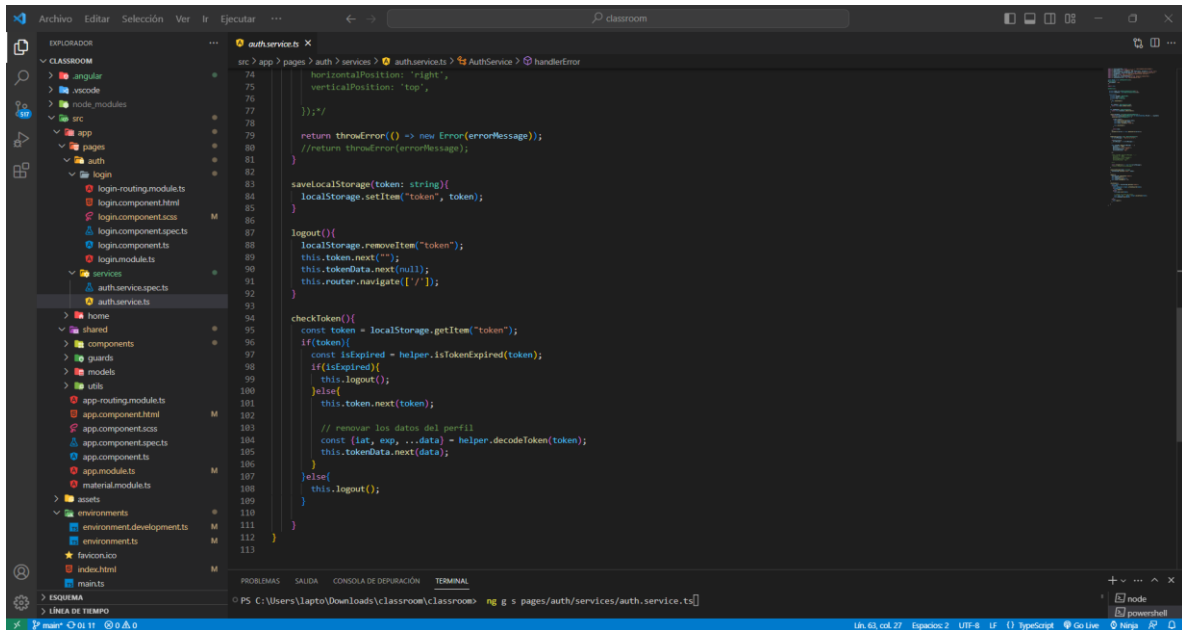


## Agregamos nuestro environment con el link de nuestra api

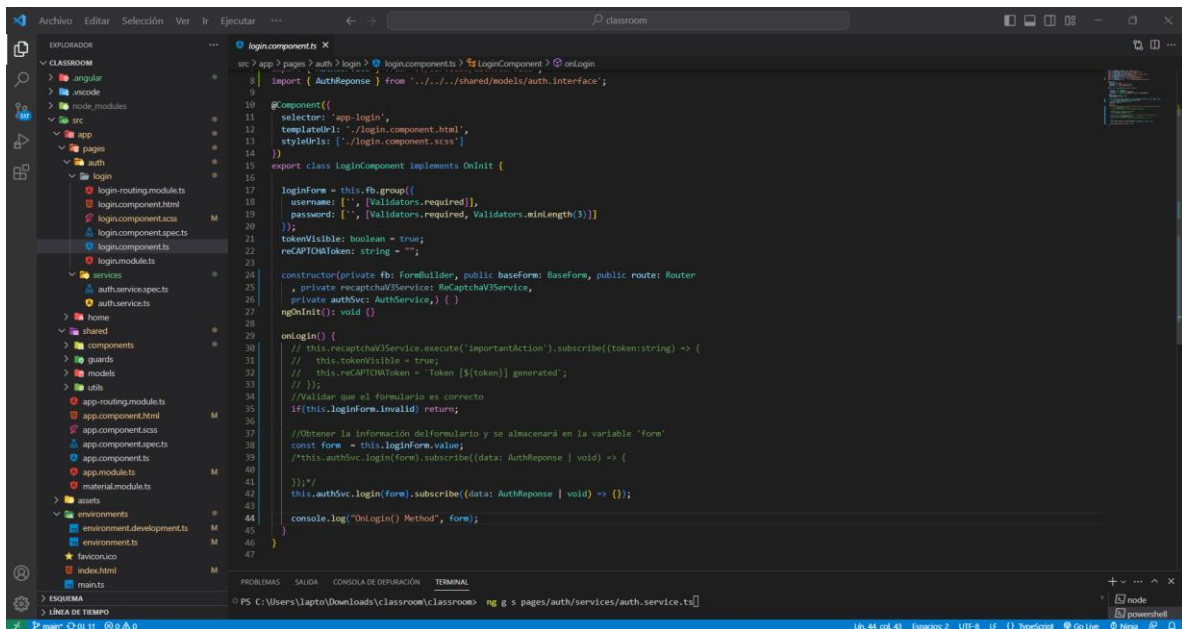


## En nuestro componente login creamos nuestro servicio auth.service.ts

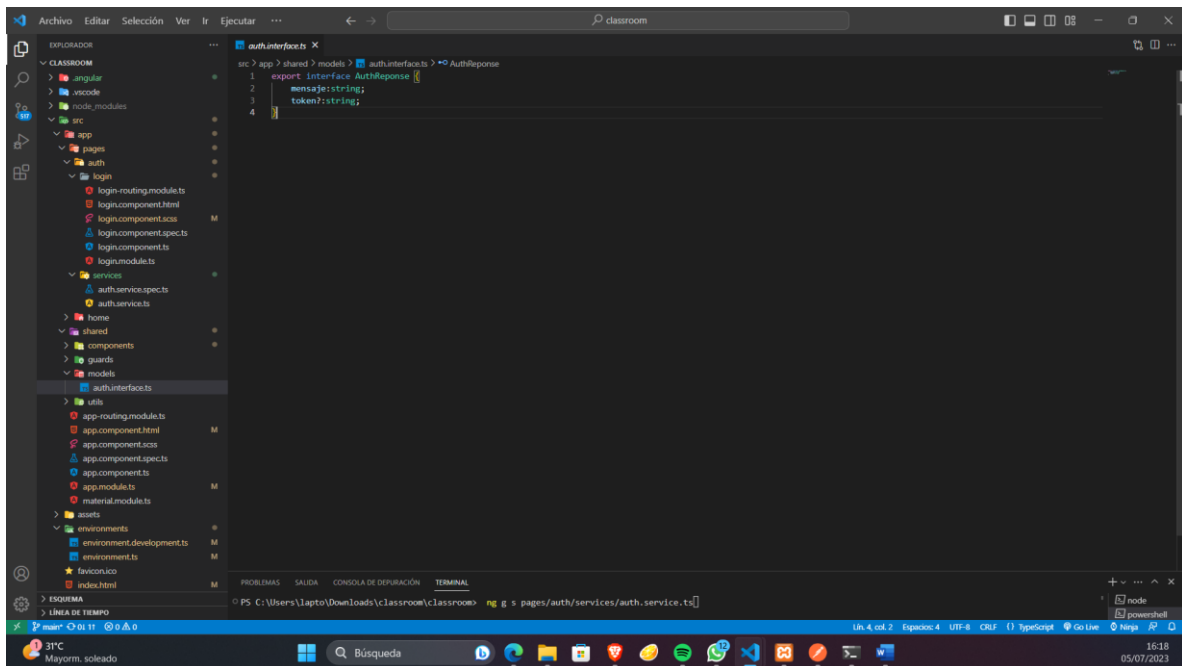




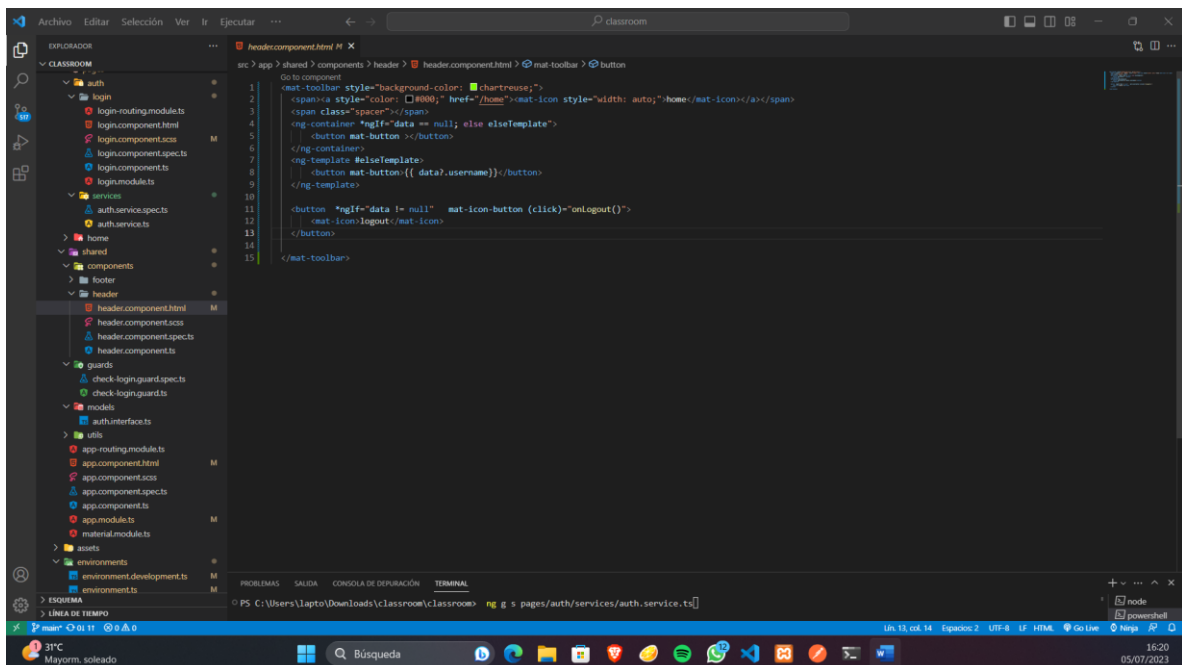
En nuestro login.component.ts actualizamos la función login y mandamos llamar a nuestro servicio

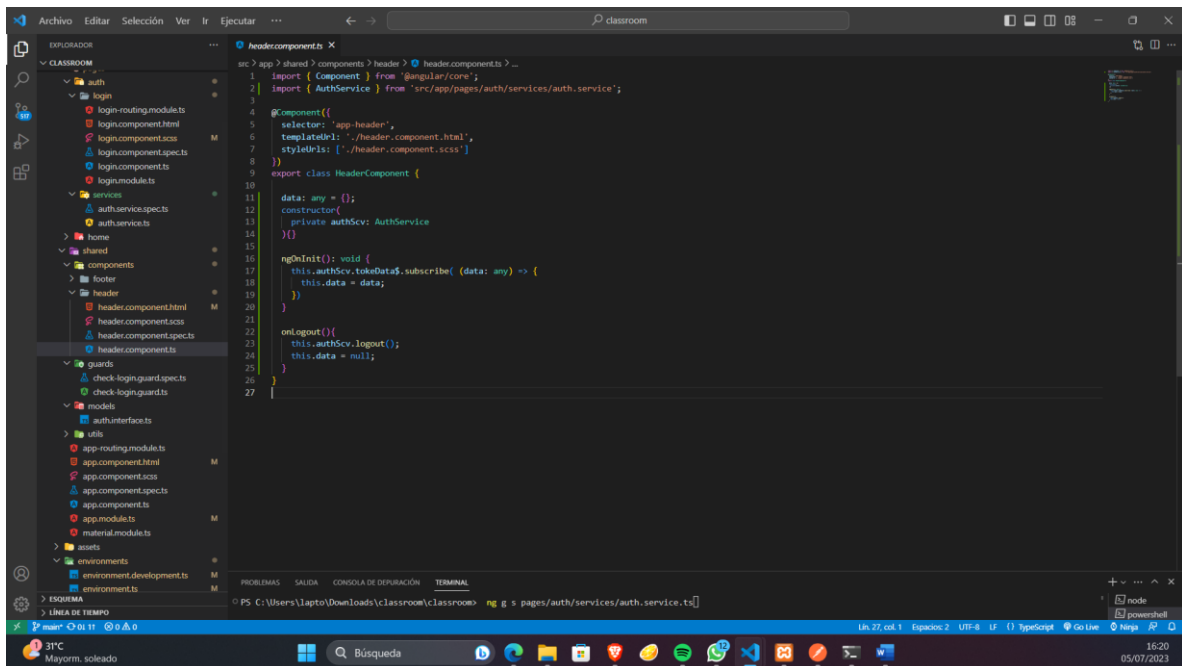


Creamos nuestro modelo auth.interface.ts

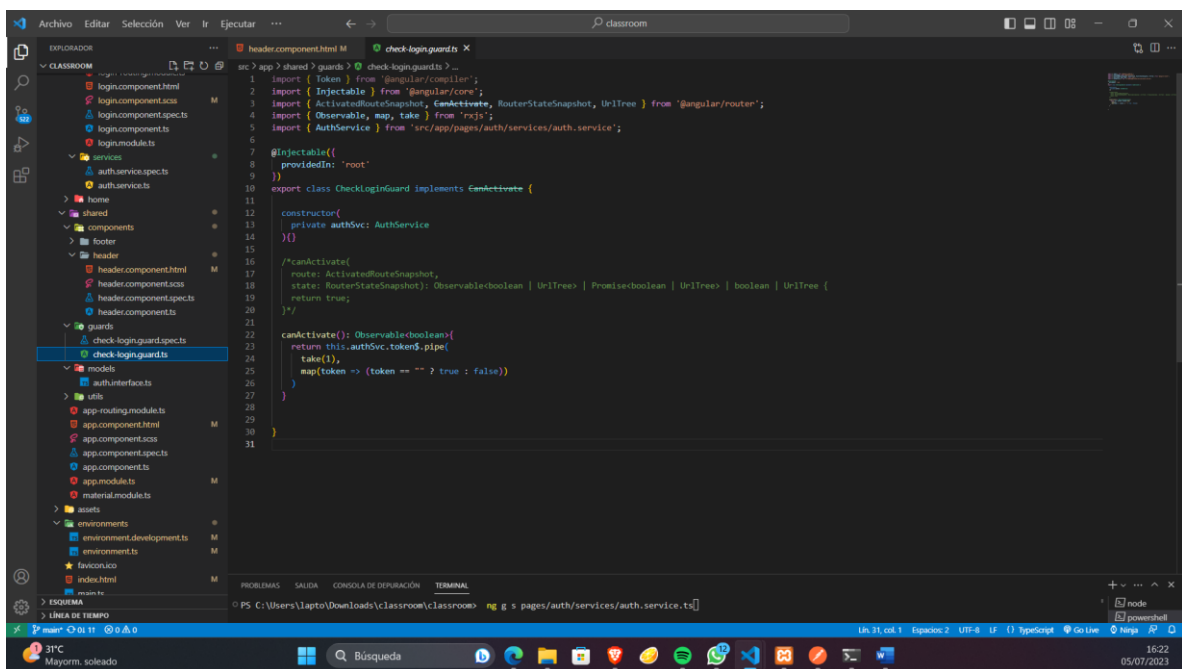


**Nuestro componente header lo actualizamos para que se vea el nombre y apellido y también podamos ejecutar la función de cerrar sesión**

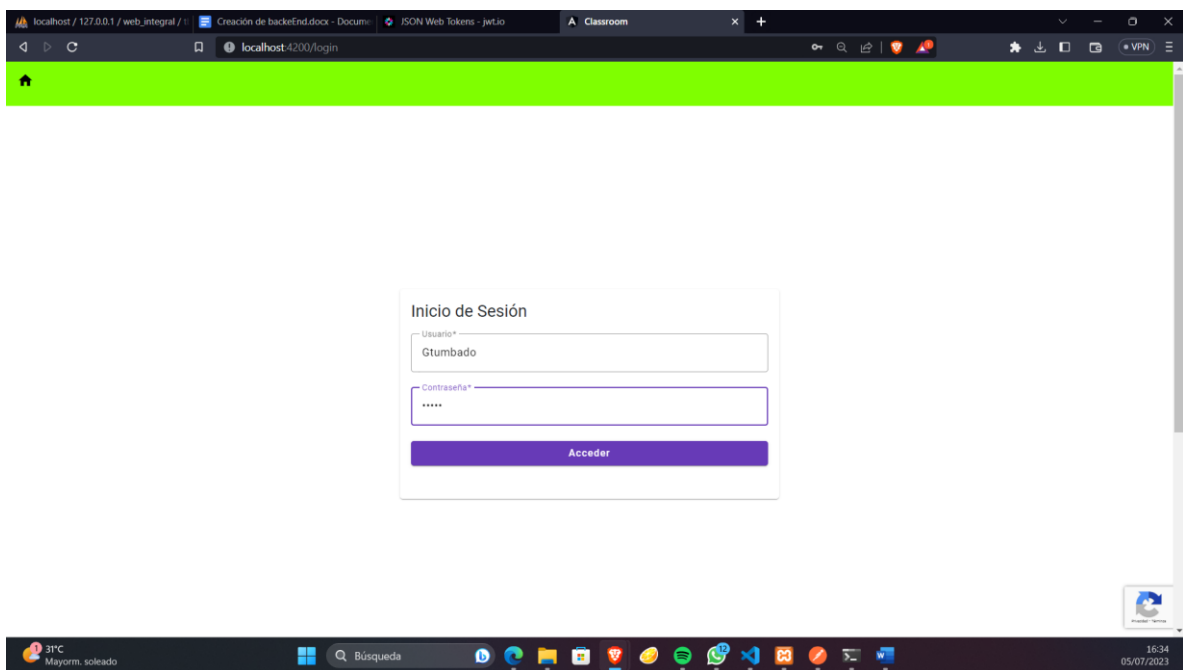
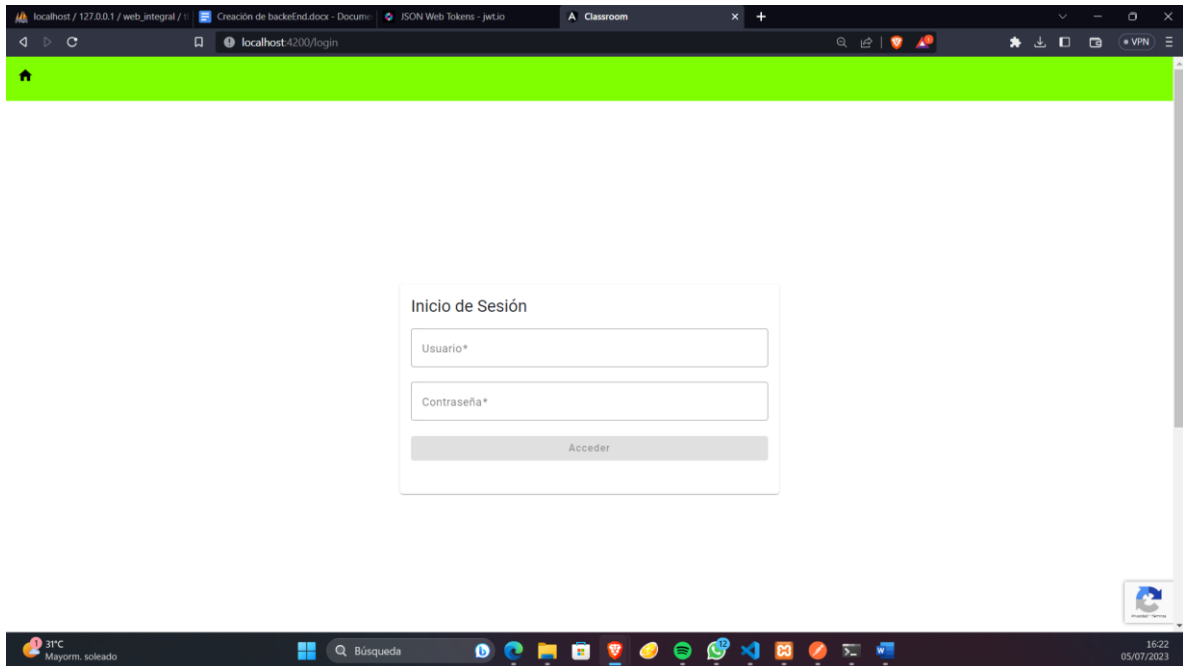


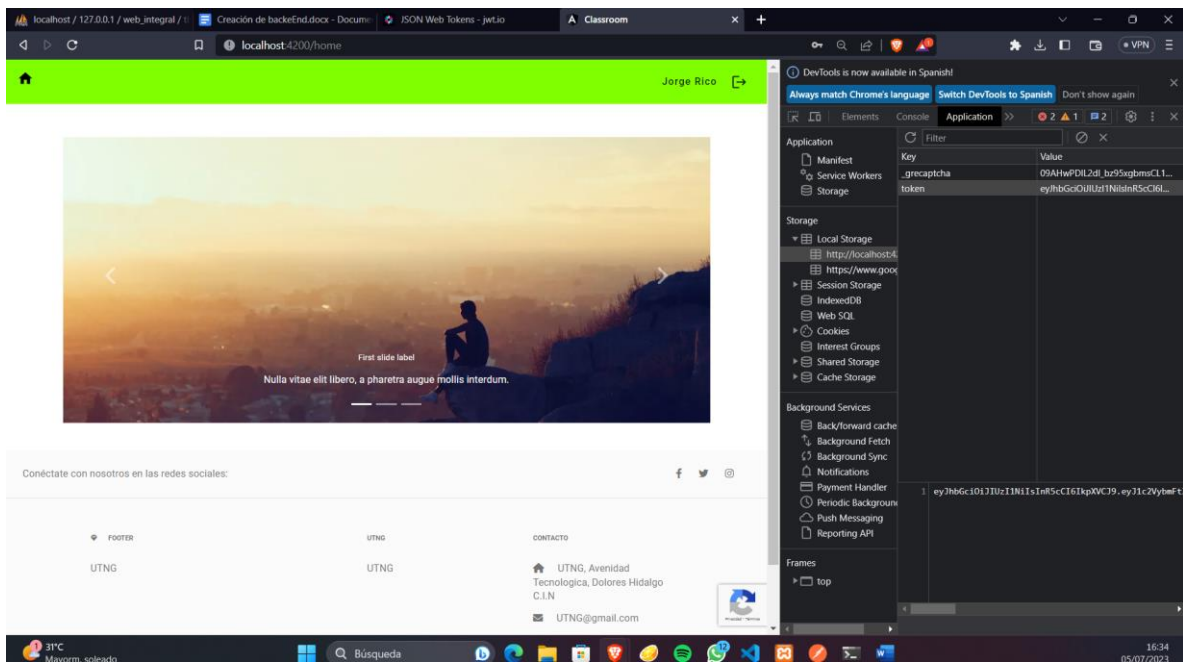
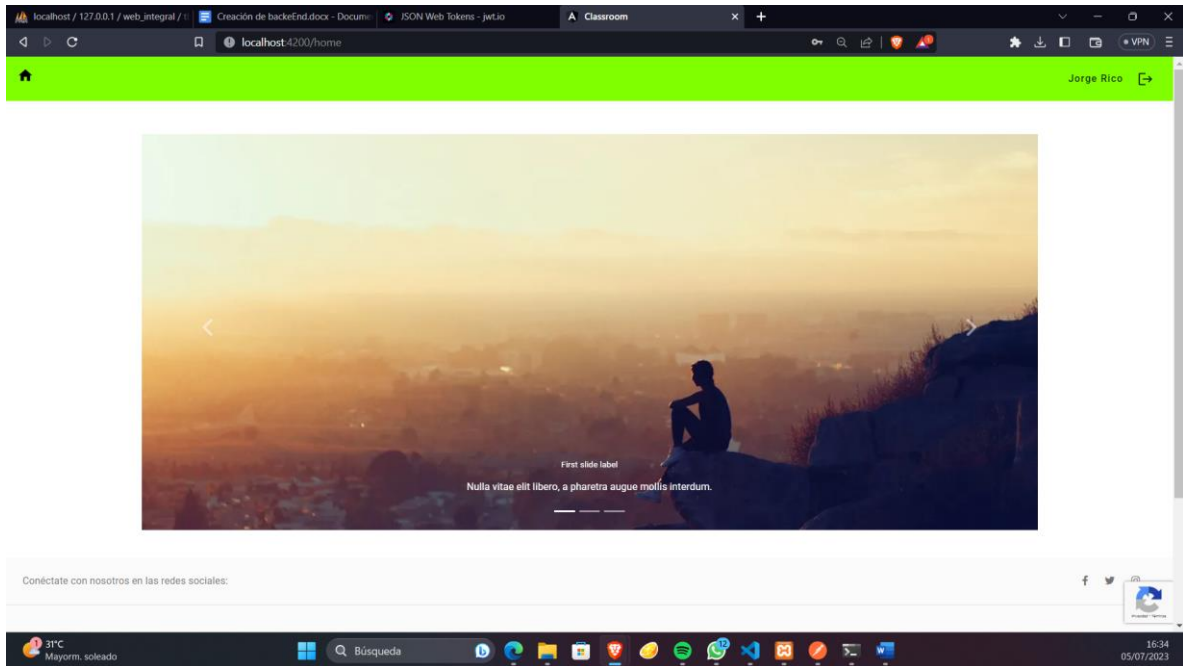


Para la protección de rutas creamos un guard y llamamos un método de nuestro service

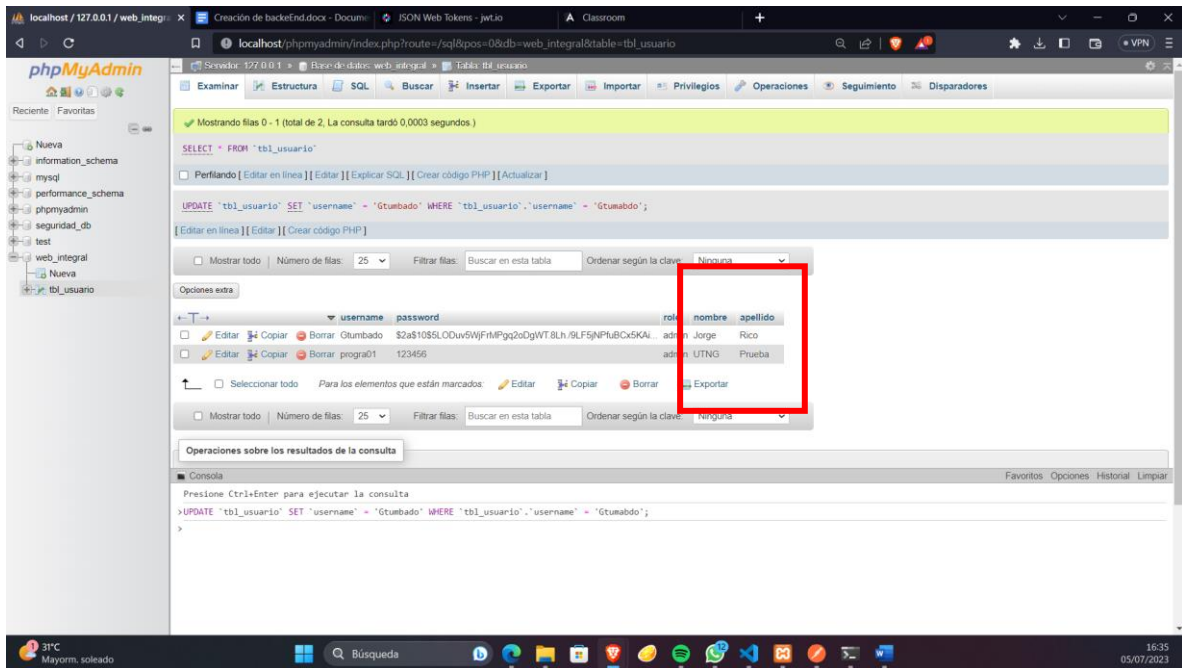


Esta seria ya la aplicación con las modificaciones agregadas





## AQUÍ ESTA EL DESAFIO QUE PIDE.



The screenshot shows the phpMyAdmin interface. The SQL query editor contains the following code:

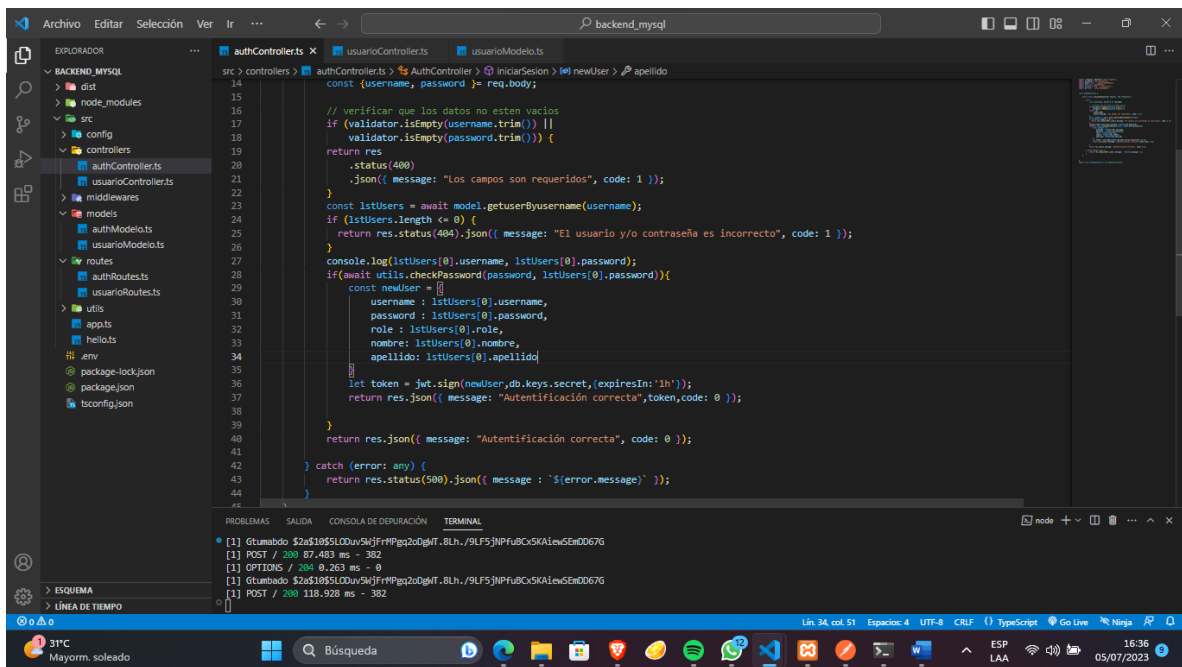
```
SELECT * FROM `tbl_usuario`  
  
UPDATE `tbl_usuario` SET `username` = 'Gtumabdo' WHERE `tbl_usuario`.`username` = 'Gtumabdo';
```

The results table shows two rows of data. The first row is highlighted with a red box:

username	password	rol	nombre	apellido
Gtumabdo	\$2a\$10\$5L0DuV5WfFmPg2oDgWT8Lh./9LF5/NPfuCCKAeVEnDD67G	admin	Jorge	Rico
progra01	123456	admin	UTNG	Prueba

The console shows the execution of the UPDATE query:

```
> UPDATE `tbl_usuario` SET `username` = 'Gtumabdo' WHERE `tbl_usuario`.`username` = 'Gtumabdo';
```



The screenshot shows the VS Code editor with the following code in the `authController.ts` file:

```
src > controllers > authController.ts > AuthController > new User > apellido  
const { username, password } = req.body;  
  
// verificar que los datos no esten vacios  
if (validator.isEmpty(username.trim()) ||  
    validator.isEmpty(password.trim())) {  
    return res  
        .status(400)  
        .json({ message: "Los campos son requeridos", code: 1 });  
}  
  
const lstUsers = await model.getUserByUsername(username);  
if (lstUsers.length <= 0) {  
    return res.status(404).json({ message: "El usuario y/o contraseña es incorrecto", code: 1 });  
}  
  
console.log(lstUsers[0].username, lstUsers[0].password);  
if (await utils.checkPassword(password, lstUsers[0].password)) {  
    const newUser = {  
        username: lstUsers[0].username,  
        password: lstUsers[0].password,  
        role: lstUsers[0].role,  
        nombre: lstUsers[0].nombre,  
        apellido: lstUsers[0].apellido  
    };  
  
    let token = jwt.sign(newUser, db.keys.secret, { expiresIn: '1h' });  
    return res.json({ message: "Autenticación correcta", token, code: 0 });  
}  
  
return res.json({ message: "Autenticación correcta", code: 0 });  
  
} catch (error: any) {  
    return res.status(500).json({ message: `${error.message}` });  
}
```


The terminal shows the following output:

```
[1] Gtumabdo $2a$10$5L0DuV5WfFmPg2oDgWT8Lh./9LF5/NPfuCCKAeVEnDD67G  
[1] POST / 200 87.483 ms - 382  
[1] OPTIONS / 204 0.263 ms - 0  
[1] Gtumabdo $2a$10$5L0DuV5WfFmPg2oDgWT8Lh./9LF5/NPfuCCKAeVEnDD67G  
[1] POST / 200 118.928 ms - 302
```




localhost / 127.0.0.1 / web\_integrat / i | Creación de backEnd.docx - Docume | JSON Web Tokens - jwt.io | Classroom

jwt.io



Debugger Libraries Introduction Ask

Crafted by  auth0  
by Data

Encoded PASTE A TOKEN HERE

```
eyJ1c2VybmFtZSI6Ikd0dW1hYmRvIiwicGFzc3dvcml0i0iIkMmEkmTAkNUxPRHV2NVdqRnJNUGdxMm9EZ1dULjhMaC4vOUxGNWp0UGZ1QkN4NUtBaWV3U0VtREQ2N0ciLCJyb2x1IjoIYWRTaW4iLCJub21icmUiOiJKb3JnZSI6ImFwZWxsaWRvIjoIUm1jbyIsIm1hdCI6MTY4ODU5NjM0MCwiZXhwIjoxNjg4NTk5OTQwfmQ
```

Decoded EDIT THE PAYLOAD AND SECRET

HEADER: ALGORITHM & TOKEN TYPE


```
{  "username": "Gtumabdo",  "password":  "$2a$10$5L0Duv5WjFrMPgq2oDgWT.8Lh./9LF5jNPfuBCx5KA1ewSEmDD676",  "role": "admin",  "nombre": "Jorge",  "apellido": "Rico",  "iat": 1688596340,  "exp": 1688599940}
```


PAYLOAD: DATA


```
{}
```

VERIFY SIGNATURE

```
HMACSHA256(  base64urlEncode(header) + ".",  base64urlEncode(payload),  secret-key)
```

 31°C  
Mayorm, soleado

 Búsqueda



16:36  
05/07/2023