

# Desafío Profesional Full Stack

## Sprint 5 | Middlewares + Auth

En este sprint ya debería estar resuelta la parte visual del sitio. Ya lo podemos navegar, pero todavía no lo podemos usar...

¡Muy bien! Es momento ahora de lograr que el sitio comience a cobrar vida haciendo que todos esos formularios y acciones trabajen con productos y usuarios reales almacenados en formato JSON.

### Requisitos previos

#### Fuente de datos de usuarios

Indispensable para darle vida a los formularios de registro, login y a la página de perfil. Deberás tener una buena idea de los campos que necesitan guardar para cada usuario. **Recuerda que en el sprint anterior tienes una sugerencia de los campos mínimos que deberían tener.**

### Objetivo

- Durante esta iteración tu foco será el de modificar el sitio para que:
- Permita el flujo de registro, login y logout de usuarios.
- Permita recordar al usuario para que pueda ingresar sin volverse a loguear.
- Tenga rutas accesibles solo por huéspedes (visitantes sin login).
- Tenga rutas accesibles sólo por usuarios (que hicieron login).

### Consignas

#### 1. Implementar la entidad de usuarios

Tomando como ejemplo lo realizado para productos, replicar la estructura de archivos y directorios que necesitarás para implementar las funcionalidades.

Te sugerimos esta estructura, aunque si trabajaste de otra manera es válido igualmente.



- Rutas: src/routes/users.js
- Controlador: src/controllers/usersController.js
- Vistas: src/views/users/
- Directorio para imágenes: public/images/users/
- Colección: src/data/users.json

#### Entregable

- Estructura de archivos y directorios de usuarios.

## 2. Implementar el registro de usuarios

Tomando como referencia el formulario de creación de productos, implementar el formulario de registro de usuarios.

- Deberá incluir los campos mínimos mencionados en el sprint anterior.
- Deberá permitir la subida de una imagen de perfil (con [Multer](#)).
- Deberá encriptar la contraseña ingresada por el usuario (con [bcrypt.js](#)).
- Deberá guardar los datos enviados en el archivo JSON de usuarios.

#### Entregable

- Formulario funcional de creación de usuarios.

## 3. Implementar el login de usuarios

Ahora, es momento de poner en práctica middlewares, sesiones y cookies. Deberán implementar un formulario de login que:

- Incluya los campos de email y password.
- Verifique la información enviada por el usuario y según el caso:
  - Redirija a la home o a la página de perfil en caso de éxito y muestre los datos del usuario en algún lugar del sitio, como el header.
  - Redirija nuevamente al login en caso de error.

## Entregable

- Formulario funcional de login.

### 4. Implementar la función de recordar al usuario (Opcional)

Deberán agregarle al formulario de login la posibilidad de que se recuerde al usuario (checkbox). En caso de que el usuario decida ser recordado:

- Utilizar cookies para guardar esa información en el navegador.
- Implementar un middleware de aplicación que busque la cookie y loguee al usuario en caso de que exista y sus datos sean correctos.

### 5. Implementar la función de recordar al usuario (Opcional)

Ahora que tienen un login funcionando, su próximo desafío será el de separar las rutas que se pueden acceder en cualquier momento, de las que se puede acceder solo si uno no está logueado y, por último, de las que requieren estar logueado.

En cada caso deberán implementar el comportamiento que corresponda:

- Rutas accesibles por cualquiera → sin cambios
- Rutas accesibles solo sin login → redirigen al perfil
- Rutas accesibles solo con login → redireccionan al login

## Conclusión

Ya habíamos mencionado que en programación siempre buscamos no repetirnos, así que cuando una tarea la estamos haciendo muchas veces en diferentes partes de nuestro código, la abstraemos. Es decir, que aislamos ese comportamiento en un bloque de código reutilizable. Los middlewares te permiten justamente eso. Un middleware de aplicación se ejecutará en cada pedido del cliente, mientras que un middleware de ruta se ejecutará cada vez que se pida esa ruta particular.

Con estas dos herramientas se pueden hacer cosas muy poderosas. De hecho, cada uno de los paquetes que seguramente hayas instalado (Express, EJS, method-override, Multer, etc.) están contruidos como middlewares.

Por otro lado, las sesiones y las cookies son parte del funcionamiento normal de la mayoría de

los sitios que son visitados a diario. Es otra parte del desarrollo web que estás dominando e implementando.

El mundo digital que te rodea está escrito utilizando este tipo de tecnologías. A medida que avances en tu práctica y dedicación te va a resultar más fácil entender cómo funciona todo este ecosistema de webs y aplicaciones.