

# PRÁCTICA: Digitalización de albaranes

---

## Descripción:

Desarrolla una aplicación web con Next.js que permita la gestión de albaranes (partes de horas o materiales) entre clientes y proveedores. La aplicación debe permitir a los usuarios lo siguiente:

Cread las siguientes páginas realizando las peticiones necesarias al back-end, ya provisto, en <https://bildy-rpmaya.koyeb.app/api-docs/> (puedes usar fetch o axios para las llamadas)

- **OnBoarding**

- Crear una cuenta de usuario: POST /api/user/register

## Create your ZoSale ID

First name

Last name

Email

Password

☐ By proceeding, you agree to the [Terms and Conditions](#)

Sign up with email

NOTA: Al crear la cuenta (o al logarte más adelante), recibirás un token JWT desde el back-end, este deberás almacenarlo en, por ejemplo, la caché local del navegador:

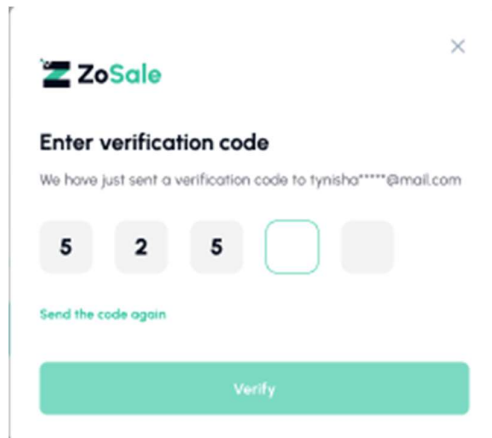
```
//Guardar datos en localStorage: localStorage.setItem('clave', 'valor')
```

```
localStorage.setItem('jwt', token)
```

```
//Leer datos de localStorage: const valor = localStorage('clave')
```

```
const token = localStorage('jwt')
```

- Validar el correo a través de un código recibido en el mismo:  
PUT /api/user/validation

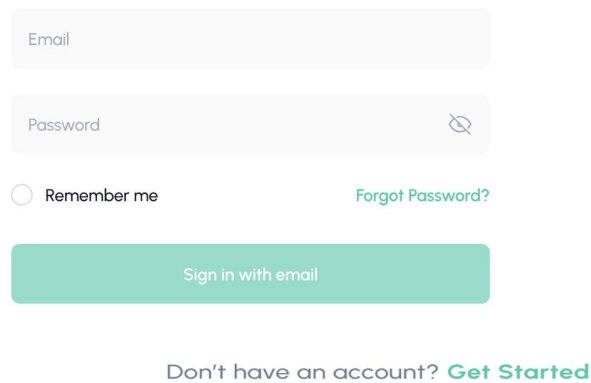
A screenshot of a web application interface for ZoSale. At the top left is the ZoSale logo, and at the top right is a close button (X). The main heading is "Enter verification code". Below it, a message states: "We have just sent a verification code to tynisha\*\*\*\*\*@mail.com". There are five input fields for the code; the first three contain the numbers "5", "2", and "5", while the fourth and fifth are empty. Below the input fields is a link that says "Send the code again". At the bottom is a large green button labeled "Verify".

NOTA: Como la petición requiere token de sesión, guardado previamente en localStorage, enviaremos la petición con las cabeceras:

```
headers: { 'Content-Type': 'application/json', 'Authorization': `Bearer ${token}` }
```

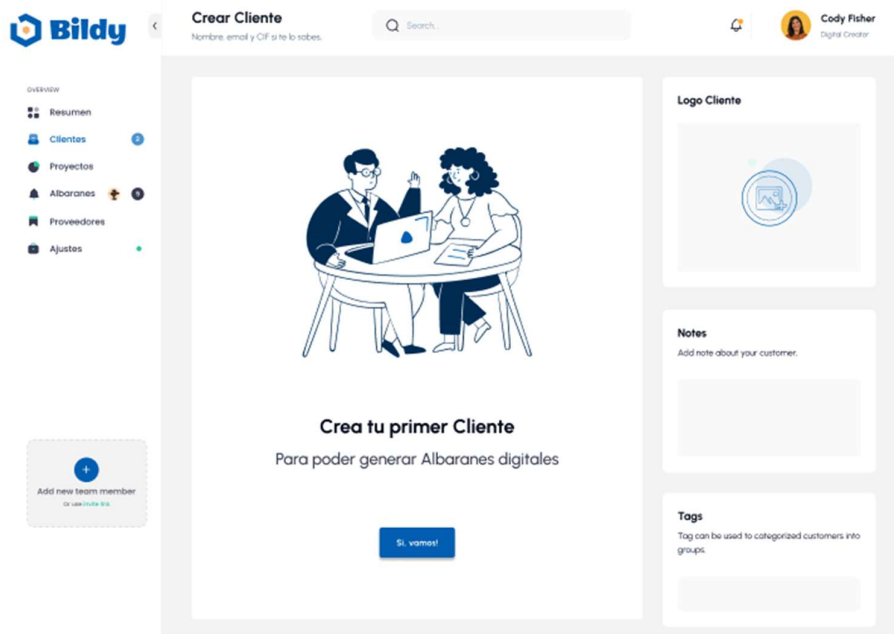
- Login en la aplicación (devuelve un token de sesión): POST /api/user/login

## Login to your account

A screenshot of a login form titled "Login to your account". It features two input fields: "Email" and "Password". The "Password" field has a toggle icon for visibility. Below the fields are two options: a radio button for "Remember me" and a link for "Forgot Password?". A large green button labeled "Sign in with email" is positioned below these options. At the bottom, there is a link "Don't have an account?" followed by a green button labeled "Get Started".

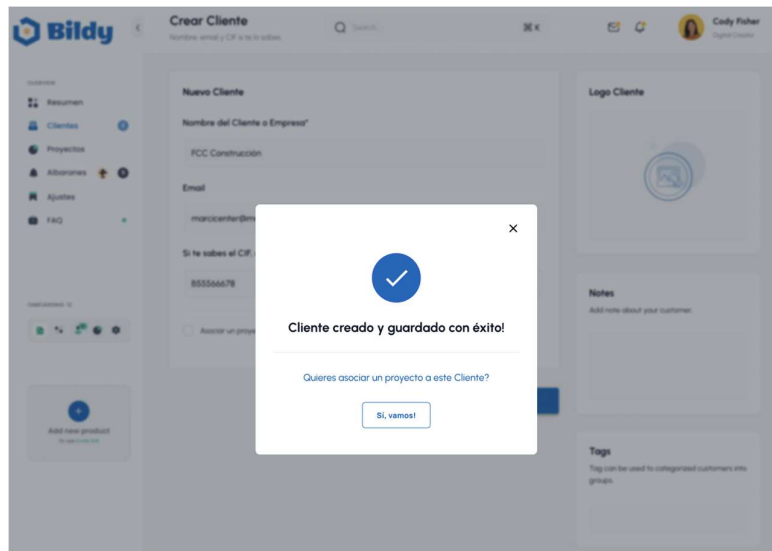
- **Crear un cliente:**

- Si todavía no hay ninguno, primero mostrar esta pantalla:

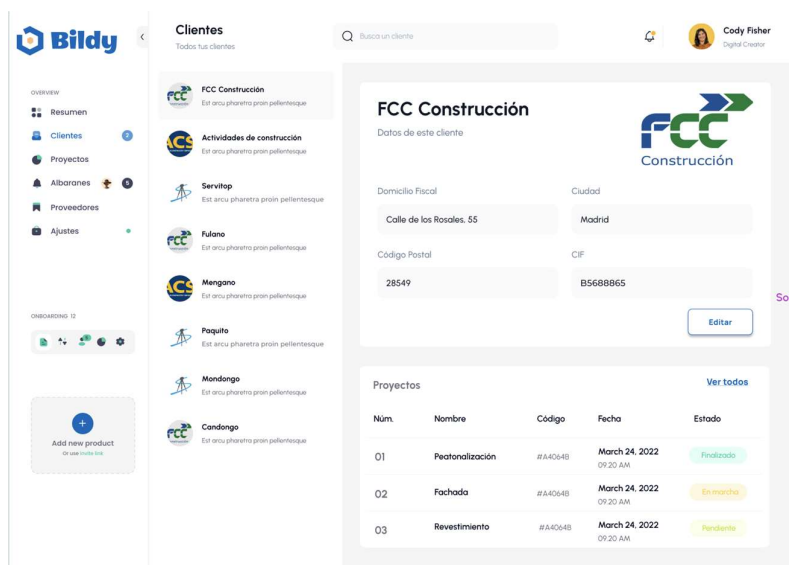


- Crear el cliente: POST /api/client

- Confirmar la creación en ventana emergente

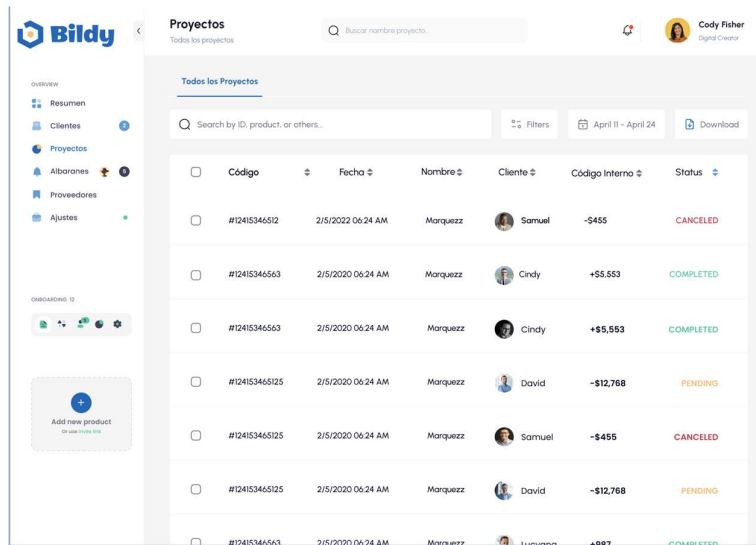


- Si ya hay clientes, mostrarlos todos: GET /api/client
- y uno en concreto: GET /api/client/{id}

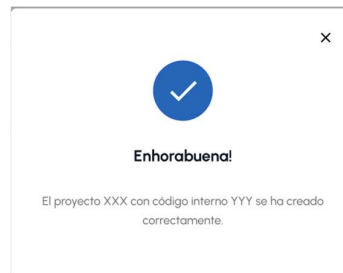


- **Proyectos:**

- Vista de proyectos en general de un cliente: GET /api/project



- Crear un proyecto asociado a un cliente: POST /api/project  
Además, si quiero actualizarlo: PUT /api/project/{projectId}



- Vista de un proyecto específico: GET /api/project/one/{id}

**Proyectos**  
Detalle de Proyecto

Buscar nombre proyecto...

### Peatonalización

Datos de este Proyecto

Código Interno: UCO23654-32      Ubicación: Calle de la Bamba, 68. Madrid. 28080.

[Editar](#)

**Albaranes** [Ver todos](#)

Núm.	Nombre	Código	Fecha	Estado
01	Peatonalización	#A40648	March 24, 2022 09:20 AM	Finalizado
02	Fachada	#A40648	March 24, 2022 09:20 AM	En progreso
03	Revestimiento	#A40648	March 24, 2022 09:20 AM	Pendiente
01	Peatonalización	#A40648	March 24, 2022 09:20 AM	Finalizado
02	Fachada	#A40648	March 24, 2022 09:20 AM	En progreso
03	Revestimiento	#A40648	March 24, 2022 09:20 AM	Pendiente

**Cliente**

Domicilio Fiscal  
Calle de los Rosales, 55.  
28054, Madrid

CIF  
B55566678

**Notas**  
Add note about your customer.

- **Albarán:**

- Crear albarán de un proyecto en concreto: POST /api/deliverynote

**Proyectos**  
Detalle de Proyecto

Buscar nombre proyecto...

### Peatonalización

Datos de este Proyecto

Código Interno: UCO23654-32      Ubicación: Calle de la Bamba, 68. Madrid. 28080.

**Borrador Albarán**  
Salesline and your customers will use this information to contact you.

Store name: Makostore      Industry: Clothing

Store currency: US Dollars (USD)      Timezone: US & Canada      Weight unit: Kilogram (kg)

**Store Address**  
This address will appear on your invoices.

Legal name of company: Makostore

Apartment, suite, or etc.: Makostore      Address: 4517 Washington Ave, Manchester

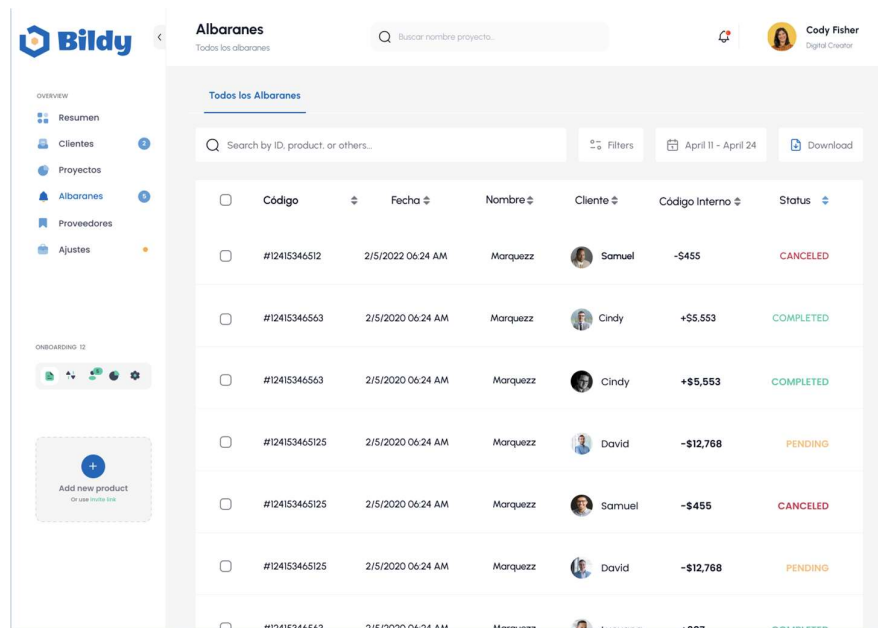
**Cliente**

Domicilio Fiscal  
Calle de los Rosales, 55.  
28054, Madrid

CIF  
B55566678

**Notas**  
Add note about your customer.

- Listar albaranes: GET /api/deliverynote



- Descargar albarán en pdf: GET /api/deliverynote/pdf/{id}

## Objetivos:

- Practicar todo lo aprendido realizando una aplicación con Next.js profesional..
- Implementar los conceptos básicos de React:
  - Creación de componentes
  - Props
  - useState hook
  - Manejo de eventos
  - Comunicación hijo-padre
  - Condicionales y listas
  - Aplicación de estilos
  - Asignación de estilos y clases dinámicos
  - Styled components y CSS modules
- Integrar librerías:
  - Formik o React Hook Form (opcional)
  - Peticiones HTTP (axios o fetch) a un API real.

## Guía de resolución:

### 1. Planificación:

- Diseñar la estructura de la aplicación:
  - Páginas/rutas principales
  - Componentes
  - Flujo de navegación
- Definir el estado y las props de los componentes.
- Elegir las librerías a utilizar (opcional).

### 2. Implementación:

#### 2.1 Routing:

- Configurar las rutas de la aplicación (árbol de directorios) con NextJS.
- Crear page.js y sus componentes para cada página.

#### 2.2 Componentes:

- Desarrollar los componentes de la interfaz cuando aplique:
  - Header
  - Footer
  - Barra lateral
  - Menú de navegación
  - Página de inicio
  - Página de clientes
  - Página de proyectos
  - Página de albaranes
  - Formularios
  - Descarga de albaranes

#### 2.3 Funcionalidades:

- Implementar la lógica de cada componente:
  - Filtrado de clientes, proyectos y albaranes
  - Manejo de formularios
  - Envío de datos (POST y PUT)

#### 2.4 Estilos:

- Aplicar estilos a los componentes con CSS.
- Usar tailwind (opcional).

### 3. Integración de librerías (opcional):

- Implementar Formik o React Hook Form para los formularios (opcional).
- Realizar peticiones HTTP para obtener datos (axios o fetch).



#### 4. Pruebas y depuración:

- Probar la aplicación en diferentes navegadores y dispositivos.
- Corregir errores y optimizar el rendimiento.

#### Navegación:

- Usar el componente Link para navegar a otras páginas:

```
<Link to="/clients">Clientes</Link>
```

- Usar useRouter para navegar ejecutando acciones cuando aplique.

#### Consideraciones adicionales:

- Redireccionar a los usuarios a la página 404 creada por nosotros si intentan acceder a una ruta inexistente.
- Mostrar mensajes de error personalizados si hay problemas.