

Introducción al desarrollo Web con Ant Design

Formularios

Vamos a crear un formulario de login, por lo que voy a crear un componente llamado FormLogin, no olvides importar la hoja de estilos, y los elementos necesarios para el correcto funcionamiento de este componente.

Para mostrar nuestro formulario, vamos a usar el componente **Card** como se muestra en la imagen, no olvides importar estos componentes desde antd

```
<Card
  title="Bienvenido de nuevo!"
  bordered={false}
  className="responsive-card"
>
  /* Aquí tu formulario */
</Card>

4  .responsive-card{
5    text-align: center;
6    width: 80%;
7    max-width: 800px;
8    margin: 0 auto;
9    padding: 20px;
10
11 }
```

Para el Formulario, importamos los elementos de la librería de antd **import { Button, Form, Input, Card } from 'antd'**; al inicio del componente, crea la función para enviar los datos del formulario y la función en caso de error, por ahora solo responde con un console.log

```
//Función para enviar los datos del formulario
const onFinish = (values) => {
  console.log('Success:', values);
};
//Función para mostrar errores en el formulario
const onFinishFailed = (errorInfo) => {
  console.log('Failed:', errorInfo);
};
```

Para el formulario usaremos el componente Form que envuelve cada uno de los elementos que integran el formulario de login este recibe distintas propiedades, por ejemplo onFinish que ejecuta la función creada anteriormente, es importante resaltar que cada elemento del formulario debe de ir envuelto en la etiqueta <Form.Item> con su etiqueta de cierre, esta recibe parámetros como **“rules”** que integra distintas validaciones para cada uno de los campos, por ejemplo si el campo es obligatorio y la propiedad **“message”** que contiene el mensaje en caso de error.

Dentro del componente Input, y para este caso estamos agregando la propiedad **prefix** que renderiza un icono dentro del input, los iconos también es necesario importarnos **import { LockOutlined, UserOutlined } from '@ant-design/icons'**;

Para el botón, se considera de tipo primario y submit, igual agregamos una clase para personalizar, por último una etiqueta a para ingresar al registro del sistema.

```
<Form
  name="normal_login"
  className="login-form"
  initialValues={{
    remember: true,
  }}
  onFinish={onFinish}
>
  <Form.Item
    name="username"
    rules={[
      {
        required: true,
        message: 'Por favor ingrese su usuario',
      },
    ]}
  >
    <Input prefix={<UserOutlined className="site-form-item-icon" />} placeholder="Usuario" />
  </Form.Item>
  <Form.Item
    name="password"
    rules={[
      {
        required: true,
        message: 'Por favor ingrese su contraseña',
      },
    ]}
  >
    <Input
      prefix={<LockOutlined className="site-form-item-icon" />}
      type="password"
      placeholder="Contraseña"
    />
  </Form.Item>
  <Form.Item>
    <Button type="primary" htmlType="submit" className="login-form-button">
      Iniciar Sesión
    </Button>
  </Form.Item>
  <Form.Item>
    ¿Aún no tienes cuenta? <a href="">Regístrate</a>
  </Form>
```

Iconos

Para los iconos en necesario instalar la librería, ***npm install @ant-design/icons --save***