

Evaluación

Proyecto Práctico Integrador

En el transcurso de este módulo conocerás las herramientas fundamentales de React y Node.js, por lo cual te presentamos el proyecto integrador, en el cual debes aplicar todo lo aprendido en el transcurso de este.

¡Presta mucha atención!

A decorative graphic in the bottom left corner consisting of several overlapping, semi-transparent triangles in shades of purple and pink.

PROYECTO INTEGRADOR FULLSTACK – MÓDULO 3. ÉNFASIS REACT+NODE.JS

¡Enyoier!

En el transcurso de este módulo conocerás las herramientas fundamentales de React y Node.js por lo cual, te presentamos el proyecto integrador, en el cual debes aplicar todo lo aprendido en el transcurso de este.

¡Presta mucha atención!

Contexto general

La empresa **ZTechnology** desea implementar un CRM a la medida y te ha seleccionado a ti para llevar a cabo el proyecto. Será una plataforma en la nube que permita gestionar usuarios, clientes y cotizaciones.

Requisitos de negocio

La empresa tiene algunos requerimientos que un analista se ha encargado de recolectar y estructurar. Si consideras que algún requerimiento es ambiguo recuerda no asumir y preguntar a tu tutor.

- La plataforma debe permitir el inicio de sesión de los miembros mediante un usuario y una contraseña.
- Existirán dos tipos de usuario:
 - **Administrador:** tiene todas las capacidades en la plataforma.
 - **Gestor:** tiene todas las capacidades de la plataforma excepto la creación de usuarios y el cambio de información de los usuarios.
- Un usuario debe poder cambiar su propia contraseña. base de datos usuarios ✓
- Un administrador debe poder crear usuarios.
- Un administrador debe poder asignar un rol a otro usuario. ✓



- Un administrador debe poder modificar los datos (incluida la contraseña) de usuarios gestores.
- Un administrador debe poder modificar los datos de otros administradores.
- Si un usuario falla en su inicio de sesión 3 veces consecutivas, debe bloquearse la cuenta por 2 horas.
- Un usuario debe poder cerrar sesión.
- Un usuario debe poder crear cuentas (clientes). [base de datos clientes](#) ✓
- Un usuario debe poder modificar cuentas.
- Un usuario debe poder ver un listado de clientes.
- Un usuario debe poder crear productos. [base de datos productos](#) ✓
- Un usuario debe poder modificar productos.
- Un usuario debe poder listar productos.
- Un usuario debe poder crear cotizaciones. [base datos cotizaciones](#) ✓
 - Las cotizaciones deben contener una numeración automática pero modificable.
 - Las cotizaciones deben contener productos (cada uno con cantidad, detalle y precio).
 - Las cotizaciones deben estar asociadas a una cuenta y a un usuario. o Las cotizaciones deben tener un precio de envío.
 - Una cotización puede tener un descuento (se debe dar opción de hacerlo porcentual o monto fijo).
 - Las cotizaciones deben tener un subtotal y un total (antes y después de envío).
- Un usuario debe poder modificar una cotización.
- Se debe poder enviar cada cotización al correo electrónico del cliente asociado (en formato HTML dentro del cuerpo del correo).
- Se debe mostrar en el escritorio (la pantalla inicial después de iniciar sesión) una gráfica totalizando los valores cotizados por cada uno de los días de la última semana.



Requisitos adicionales si se entrega en pareja

Si deseas entregar tu proyecto en pareja, se agregarán algunos requerimientos técnicos adicionales, por favor léelos con detenimiento:

- Se debe poder enviar cada cotización al correo electrónico del cliente asociado en formato PDF; por tanto la plataforma debe generar el PDF y enviarlo por correo.
- Se debe tener un registro de los correos electrónicos enviados.
- Se debe conectar por medio de IMAP a un proveedor de correo electrónico para visualizar desde la plataforma la bandeja de entrada de esa cuenta de correo (no es necesario ingresar a cada uno de los mensajes pero sí ver el asunto y el remitente).
- Se debe contar con una API para ejecutar dos funciones principales: actualizar los datos de un cliente y obtener los datos de un cliente.

Requisitos técnicos

- El proyecto debe funcionar empleando React JS, una base de datos MySQL (PostgreSQL o MariaDB) y ExpressJS.

Entregable

Se debe compartir un repositorio de Github o Bitbucket que cuente con el código fuente del proyecto y que incluya una carpeta llamada "DOCS" donde se encuentren los siguientes elementos:

- SQL de la base de datos generada.
- Documentación necesaria para el despliegue (una guía de no más de una hoja).
- Historias de usuario de al menos 4 requerimientos.
- Un archivo llamado integrantes.txt donde coloques el nombre y correo electrónico de quienes participaron en el proyecto.

Adicionalmente, es necesario desplegar el proyecto en un ambiente web al que los revisores puedan acceder fácilmente.



¡Te recomendamos!

- El cliente no maneja impuestos así que no es necesario incluirlos.
- Antes de codificar, tómate un tiempo para hacer la estructura del proceso en la herramienta que consideres necesaria (incluso en hojas) para que puedas tener claridad.
- Recuerda que para la mayoría de las personas las decisiones entran por los ojos así que si bien es fundamental la parte funcional del proyecto, no descuides la parte estética.

