



# Roilan Lauzardo Sotolongo

## Desarrollo de Software y Aplicaciones Web

Graduado de Ingeniería Informática en La Universidad Agraria de la Habana "Fructuoso Rodríguez Pérez" en julio del 2015

## Perfil Profesional

Programador con conocimientos en HTML5, CSS3, JavaScript, TypeScript, Java, C#, SQL, NoSQL, Nodejs, React, React+Vite, Visual Code, Visual Studio. Me especializo en el desarrollo de software y aplicaciones web. Con un enfoque orientado a la solución de problemas y la eficiencia, he demostrado ser un desarrollador capaz de trabajar en equipos multidisciplinarios y de entregar resultados de alta calidad. Aprendí a programar en la Universidad Agraria de la Habana y he continuado mejorando mis habilidades a través de proyectos personales y de clase. Estoy dispuesto a colaborar en proyectos que me permitan utilizar y mejorar mis habilidades como programador.

## Experiencia Laboral

**Proyecto:** MirelisNailRol

**Tecnologías Utilizadas:** MongoDB, Node.js, Express, React, Material Design, JWT, Axios, Multer, Mongoose, CORS.

**Descripción del Proyecto:** Desarrollé y lancé con éxito un catálogo digital interactivo para MirelisNail, un negocio especializado en servicios de manicura. Este proyecto multifacético incluyó la creación de una robusta arquitectura de servidor y cliente, asegurando una experiencia de usuario fluida y una gestión eficiente del contenido.

### Responsabilidades y Logros:

► **Desarrollo de la Web Cliente:** Implementé una plataforma web orientada al cliente para promocionar los servicios de manicura ofrecidos por MirelisNail. La web incluye una galería de trabajos realizados, información detallada sobre el equipo y sus certificaciones, y un sistema para visualizar los horarios disponibles. Aseguré que la interfaz fuera intuitiva y accesible, mejorando la visibilidad y el alcance del negocio. 🔄

► **Construcción de la Web de Administración:** Diseñé y desarrollé una web de administración utilizando Material Design para permitir al propietario del negocio gestionar de manera autónoma el contenido del catálogo. Creé componentes visuales personalizados para React, como tablas, campos de entrada, avatares y selectores de archivos, que mejoraron la experiencia administrativa y la eficiencia operativa. 🔄

► **Integración de la API:** Desarrollé una API RESTful utilizando Node.js y Express, que sirve como puente entre la base de datos MongoDB y las interfaces de cliente y administración. Implementé la biblioteca Multer para manejar y redimensionar imágenes, mejorando la gestión de medios en la plataforma. 🔄

## Contacto

✉ roilanlauzardosotolongo@gmail.com

☎ +53 56431437

🌐 <https://github.com/rls1990>

## Tecnologías y Herramientas



NodeJS



React



React+Vite



MUI



Tailwind



Express



Visual Code



MongoDB



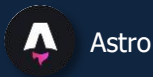
Visual Studio



XAML



Angular



Astro



Next

## Aptitudes y Competencia

- Compromiso con los proyectos
- Trabajo en equipo
- Desarrollo de prototipos
- Depuración de código
- Adaptabilidad
- Capacidad Analítica

► **Gestión de la Base de Datos:** Utilicé Mongoose para interactuar con MongoDB Atlas, asegurando un almacenamiento de datos eficiente y una recuperación rápida para la experiencia del usuario final.

► **Seguridad:** Implementé autenticación basada en tokens JWT para proteger la web de administración, garantizando la seguridad y la integridad de las operaciones del sitio.

► **Conectividad y Despliegue:** Aseguré una conexión segura y confiable entre la API y las interfaces de usuario mediante el uso de CORS. Desplugué la web de administración y el catálogo utilizando CloudFlare, y la API a través de Render, demostrando competencia en la gestión de servicios de hosting en la nube.

**Proyecto:** ChatReact

**Tecnologías Utilizadas:** MongoDB, Node.js, Express, WebSocket, Socket.io, JWT, Material Design, React, CSS Frameworks

**Descripción del Proyecto:** Desarrollé con éxito una aplicación web de chat en tiempo real para ChatReact, que permite a múltiples usuarios interactuar y comunicarse en tiempo real. Este proyecto involucró la creación de una arquitectura de servidor y cliente, asegurando una experiencia de usuario fluida y una gestión eficiente de credenciales y datos.

#### **Responsabilidades y Logros:**

► **Desarrollo de la Web:** Diseñé y desarrollé una aplicación web de chat que ofrece a los usuarios la funcionalidad de comunicarse en tiempo real. Implementé la seguridad de los datos utilizando JWT, con un token de acceso y un token de refresco para garantizar la privacidad y la integridad de las operaciones del sitio. 🔄

► **Integración de la API:** Utilicé Node.js y Express como base para los endpoints del proyecto, lo que permitió establecer una conexión bidireccional en tiempo real entre el cliente y el servidor. Implementé WebSocket mediante la utilización de socket.io en la API y socket.io-client en el front, lo que resultó en una comunicación rápida y eficiente entre los usuarios. 🔄

► **Gestión de la Base de Datos:** Utilicé MongoDB como servidor de base de datos, lo que permitió almacenar eficazmente las credenciales de los usuarios y gestionar la información del chat. Funcionalidad de Progressive Web App: Aseguré que la aplicación web funcionara como una Progressive Web App, lo que mejora la experiencia del usuario y la accesibilidad de la aplicación.

► **Diseño y Desarrollo Frontend:** Utilicé Material Design como base para el diseño de la interfaz, creando componentes personalizados en React como un chat completo, inputs personalizados y componentes adicionales.

**Proyecto:** Formulario Médico 🔄

**Tecnologías Utilizadas:** Node.js, React Vite, JWT, react-router-dom, Context API, Material-UI, Formik, js-cookie, Yup.

**Descripción del Proyecto:** Desarrollé el front-end en React de una aplicación web destinada al registro de la historia clínica de diferentes pacientes. Esta plataforma permite a los profesionales de la salud gestionar de manera eficiente la información médica de sus pacientes.

### Responsabilidades y Logros

► **Desarrollo de la Web Cliente:** Implementé la plataforma web del cliente para registrar la historia clínica de diferentes pacientes. Utilicé Material-UI para mejorar la apariencia visual, combinándolo con Formik y Yup para validar los formularios de manera efectiva. Además, generé reportes de las historias clínicas, permitiendo la creación, edición y eliminación de registros.

► **Integración de la API:** Utilicé Axios para la integración con la API, facilitando la obtención de respuestas de los diferentes endpoints y asegurando una comunicación fluida entre el front-end y el back-end.

► **Seguridad:** Implementé js-cookie para gestionar las cookies que almacenan los tokens de acceso (AT) y de refresco (RF), garantizando la seguridad de la aplicación mediante la implementación de JWT en el servidor.

**Proyecto:** Proyectos de Práctica

**Tecnologías Utilizadas:** Node.js, React, JWT, react-router-dom, Context, Reducers, Redux, Material-UI, Google APIs

**Descripción del Proyecto:** Desarrollé proyectos de ejemplos que demuestran una variedad de habilidades y conocimientos como autenticación, enrutamiento, contexto, reducers, redux, memorización y la creación de APIs. Estos proyectos involucraron la creación de una arquitectura de servidor y cliente robusta, asegurando una experiencia de usuario fluida y una gestión eficiente de datos y estados complejos.


### Experiencia Adquirida:


**Autenticación:** Implementé autenticación con JWT, lo que permitió a los usuarios iniciar sesión y acceder a funciones específicas basadas en sus roles y permisos. 🔄 🔄


**Enrutamiento:** Utilicé react-router-dom para establecer rutas y navegación en la aplicación, lo que mejoró la navegabilidad y la accesibilidad de la aplicación. 🔄

**Contexto:** Creé un contexto global utilizando Context, lo que permitió a los componentes compartir información común sin la necesidad de pasar parámetros excesivos. 🔄


## Experiencia Adquirida:

**Reducers:** Utilicé reducers para establecer variables de estado complejas y lógica compleja en los componentes, lo que mejoró la gestión de estados y la escalabilidad de la aplicación. 

**Redux:** Implementé Redux para establecer un estado global de lógica compleja para proyectos escalables, lo que permitió a los componentes acceder a datos y estados compartidos. 

**Memorización:** Utilicé memorización para memorizar componentes y funciones, lo que redujo el número de re-renderizados y mejoró la eficiencia de la aplicación. 

**Componentes visuales:** Utilicé Material-UI como biblioteca de componentes visuales, lo que permitió crear una interfaz atractiva y accesible para los usuarios.

**APIs:** Creé una API en Node.js con Express para enviar archivos a Google Drive utilizando la librería de googleapis. 


**Proyecto:** Ichimoku Kumo Multiframe 

**Tecnologías Utilizadas:** MQL4, MetaTrader 4, MetaEditor, indicador técnico Ichimoku Kinko Hyo.

**Descripción del Proyecto:** Desarrollé un indicador técnico en MQL4 que se basa en Ichimoku, permitiendo visualizar el Kumo de los dos periodos anteriores en el período actual. El indicador ajusta automáticamente los valores de los periodos superiores al actual, requiriendo solo la introducción de los parámetros estándar del Ichimoku principal (9, 26, 52) y calcula las otras dos nubes o Kumo en base a esos parámetros.

**Responsabilidades:** Diseñé y desarrollé el indicador Ichimoku\_multiframe, integrando el algoritmo de cálculo del Ichimoku para obtener los valores de los dos periodos superiores. Implementé la lógica para ajustar automáticamente los valores de los periodos superiores al actual, asegurando la precisión y la eficiencia del indicador. Realicé pruebas exhaustivas para garantizar la corrección y la estabilidad del indicador en diferentes condiciones de mercado.

**Logros:** El indicador Ichimoku\_multiframe es una herramienta útil para los traders y analistas, permitiéndoles visualizar la tendencia y la dinámica del mercado de manera más detallada y precisa. El indicador se caracteriza por su capacidad para adaptarse a diferentes periodos de tiempo y condiciones de mercado, lo que lo hace especialmente útil para análisis técnicos y estrategias de trading basados en tendencias.

**Título:** Ichimoku Panel 

**Tecnologías Utilizadas:** MQL4, MetaTrader 4, MetaEditor, indicador técnico Ichimoku Kinko Hyo.

**Descripción del Proyecto:** El indicador Ichimoku Panel es un programa desarrollado en MQL4 que se encarga de mostrar el comportamiento de diferentes señales del sistema Ichimoku en todas las temporalidades. Este indicador es una herramienta valiosa para los traders que buscan obtener una visión general de la tendencia del mercado y detectar posibles oportunidades de trading.

**Responsabilidades:**

Desarrollar un indicador que muestre el comportamiento de las señales del Ichimoku en diferentes temporalidades. Implementar las líneas clave del sistema Ichimoku, incluyendo Tenkan Sen, Kijun Sen, Senkou Span A, Senkou Span B y Chikou Span. Agregar funcionalidades para mostrar el cruce de las medias móviles Tenkan Sen y Kijun Sen, la posición de las medias móviles con respecto al kumo, el comportamiento angular de la media móvil Kijun Sen y la posición del Chikou Span con respecto al precio. Incorporar colores para indicar tendencias alcistas, bajistas y neutrales.

**Logros:**

Este indicador es una herramienta valiosa para cualquier trader que busque obtener una visión general de la tendencia del mercado y detectar posibles oportunidades de trading. Su capacidad para mostrar el comportamiento de las señales en diferentes temporalidades lo hace especialmente útil para traders que buscan obtener una perspectiva más amplia de la tendencia del mercado.

# Galería

