

Salvador Martínez Espinosa



Desarrollador de Software.

(Java/JavaScript/TypeScript/Nodejs/React/Angular/VueJS/Nextjs/Nestjs/PHP/Python/Django/AWS/GoogleCloud)

Contacto.



México, Área Metropolitana.



797 100 94 88



55 42 55 92 10



cchhava@nube.unadmexico.mx



<http://salvadormartinez50.github.io/CV/>

Desarrollo Profesional

Como desarrollador de software he estado trabajando durante los últimos 4 años en diferentes industrias como lo son; el comercio minorista, la hostelería, las finanzas y el gobierno, desempeñándome tanto en Frontend como en Backend, contribuyendo en la creación y mantenimiento de aplicaciones web, por lo que me siento cómodo trabajando tanto en Backend, como en Frontend. Además de contar con la motivación para adquirir nuevas habilidades técnicas; que, mantengan actualizada mi competencia en cada uno de mis roles como desarrollador de software, mejorando día a día mi trabajo en equipo que contribuya al completo éxito de cada proyecto en el que participo.

Competencias Técnicas

Programación	Java, JavaScript, TypeScript, PHP, Python, Angular, React, Quarkus.
Tecnologías de desarrollo	Nodejs, Nestjs, Nextjs, Java Quarkus, Hibernate, DAO, DTO, Modelos, Oracle DB 12.
Bases de datos	MongoDB, MariaDB, Oracle, MySQL, SQL Server, PostgreSQL, Redis, DynamDB.
Ambientes	Windows, UNIX, Linux (RedHat, Ubuntu), Mac Os.

Otras Competencias	Inglés, Portugués, WebLogic 12c, seguimiento de errores en el registro, implementaciones automáticas y manuales. Instalación y preparación de entornos Java 11, 1.8 y 14, Diseño orientado a objetos (OOD), MVC, SOLID, JQuery, CSS3, HTML 5, Life ray (CMS), Drupal (CMS), AEM (CMS), Wordpress (CMS), Spring MVC, Spring Boot, Spring Security, Microservicios, XHTML, CSS3, FlexBox, SASS, JSON, JSP, JSF, AJAX, Laravel, Django, Vue.js, React.js, Node.js, Express.js, GraphQL, Prisma, Apollo Client, Angular, Bootstrap, Material UI, Vuetify, Gatsby.js, AWS (EC2, Amplificar, S3, RDS), Google Cloud, Kubernetes, Docker, React Native.
---------------------------	---

Experiencia Laboral

Ingeniero de Desarrollo de Software | Freelance (Nestjs/Nodejs/ Nextjs/Angular React/Java/ReactJS/VueJS/ /Python/Django/AWS/GoogleCloud)

Junio 2019 – a la fecha

Como desarrollador de software en mi trabajo actual, en Banorte; he sido responsable de la creación de Microservicios con tecnología Quarkus, exposición de servicios (endpoints) Rest tipo GET/POST/PUT, que integran la operación de agentes de seguros bancarios con AMIS. Además de proporcionar mantenimiento a la pila web de diferentes empresas, mis actividades incluyen solucionar problemas de CSS/HTML/Handlebars, así como de migrar la funcionalidad actual de servicios OSB a Microservicios Quarkus. Por lo que tengo experiencia en el uso de estas tecnologías centrales, el propósito de las correcciones ha sido mejorar los problemas de accesibilidad, como proporcionar mensajes adecuados solo para lectores de pantalla y habilitar la navegación adecuada. El proyecto implica la representación del lado del servidor utilizando la capa de Java, por lo que estoy familiarizado con la representación de marcado de archivos yaml y la lógica comercial de codificación utilizando Java y WebLogic.

Otro aspecto importante de mi trabajo ha sido el desarrollo de microservicios web REST utilizando NodeJS, TypeScript, ExpressJS y Sequelize (como ORM para interactuar con la base de datos Postgres actual). Con frecuencia, mi responsabilidad ha sido migrar la funcionalidad de la antigua API monolítica integrada en Java Spring a los microservicios creados con Node. En ese sentido, estoy familiarizado con la escritura de capas de autorización/autenticación usando JWT y cookies, escribiendo operaciones CRUD, interactuando con bases de datos SQL, escribiendo Scripts SQL, escritura de pruebas unitarias para la implementación de servicios usando Jest, Chai, Sinon y Mocha, escritura de lógica de negocios en TS, implementación de arquitectura básica en capas Controller-Service-Model, middleware de validación desde cero y manejo de errores generados por Bad Requests o problemas internos para la ejecución en el servicio web, tratando y manejando las colas de mensajes del intermediario de mensajes implementado con NATS y, en general, tratando de mejorar la calidad del código escribiendo un código eficaz que se ejecuta más rápido al experimentar con diferentes algoritmos que encuentro en los enfoques de Python. En este proyecto, estuve expuesto a Docker y Kubernetes para crear, implementar y orquestar mis microservicios y con Google Cloud para la implementación y administración de la infraestructura de Kubernetes, por lo que estoy familiarizado con la definición de archivos dockerfile y yaml para la administración de la infraestructura.

Además, como Desarrollador Full Stack Freelance me he concentrado en desarrollar aplicaciones web full stack de comercio electrónico para una variedad de clientes, todos ellos

pequeñas empresas que venden una variedad de productos físicos en todo México. En estos proyectos desarrollo toda la aplicación por lo que para la parte frontend hago uso de HTML, CSS, SASS y Material UI para la UI, ReactJS para el desarrollo de lógica frontend e integración de endpoints desde el backend, uso componentes funcionales y reacciono engancha la lógica de frontend, axios para consumir servicios web REST, React-Testing-Library para pruebas unitarias, React-Redux para gestión de estado y Styleguidist para documentación de componentes. En lo que respecta al desarrollo de back-end, trabajo con TypeScript, Node/Express y Python/Django para el desarrollo de API REST, express-validator para middleware de validación y sanitización, Jest, Mocha, Chai y Sinon para pruebas unitarias y de integración, Swagger para Documentación API, TSDoc para documentación de código, JWT, bcryptjs y Cookies para estrategias de autenticación, autorización y encriptación, dependiendo de la base de datos con la que elijo trabajar utilizo un ORM específico como Sequelize para bases de datos SQL (he trabajado con MySQL, SQL Server, Postgres, AWS DynamoDB, SQLite), Prisma cada vez que trabajo con una API GraphQL (y Apollo Client para consumir mi API GraphQL) o Mongoose cuando trabajo con MongoDB, Stripe para manejar pagos y para los backends monolíticos que uso AWS para la implementación, algunos servicios que uso normalmente son: AWS Amplify, AWS S3, AWS EC2, AWS Route 53, AWS lambdas. También trabajé en un enfoque de microservicios, por lo que uso Docker, Kubernetes, Skaffold y Google Cloud para la infraestructura y la implementación, GitHub Actions para CI/CD, NATS Streaming Server como mi cola de mensajes para mi arquitectura de eventos y Next.js para SSR en la interfaz.

Desarrollador Full Stack (React/Node/Vue/Java/Ionic/Angular/Spring) | CIAT (México, Consultoría Innovadora de Aplicaciones Tecnológicas S. de RL.)

Abril 2017 – junio 2019

Como Desarrollador Full Stack en CIAT, fui responsable del desarrollo de nuevas funciones de backend y del mantenimiento de las ya integradas mediante el desarrollo de nuevos servicios web REST y el ajuste de la lógica comercial actual en las API ya existentes en algunos casos y el desarrollo de otros desde cero, de acuerdo a la especificación y necesidad en Walmart Centroamérica usando TypeScript, Node, Express, también construí middlewares de validación y sanitización haciendo uso de Express-Validator para todos y cada uno de los endpoints que expuse al cliente. Durante el tiempo que trabajé allí, utilicé buenas prácticas de desarrollo al hacer que mi código fuera reutilizable, comprensible, fácil de leer, comprobable (implementando pruebas unitarias y de integración con Jest, Mocha, Chai y Sinon) (documentando mi API con puntos finales usando swagger y TSDoc). La base de datos que usé fue SQL Server, así que usé Sequelize como un ORM para interactuar con la base de datos desde mi código API. También dediqué suficiente tiempo al asegurarme de que mis servicios web fueran eficientes al garantizar un alto rendimiento y capacidad de respuesta a las solicitudes del front-end mediante el uso de algoritmos eficientes para diferentes necesidades. Como parte de mis tareas de frontend, trabajé principalmente en la integración de Endpoints de backend en la aplicación React.JS y en el desarrollo de la lógica de frontend con los datos de backend, por lo que usé las funciones principales de React, como componentes funcionales, ganchos de reacción y react-redux para la gestión de estado, y axios como cliente REST para consumir Endpoints de API y probando mi código con el uso de React-Testing-Library. También he ayudado en un título secundario en el desarrollo y estilo de la interfaz de usuario con Material UI, SASS, HTML y CSS. En este proyecto trabajé en un equipo de 6 desarrolladores más, un Scrum Master y 3 ingenieros de QA, estábamos utilizando una metodología ágil para el ciclo de desarrollo y

debido a que el proyecto se desarrollaba geográficamente en Centroamérica, trabajé de forma remota. de principio a fin.

Así como también he sido responsable del desarrollo de nuevas funciones de Frontend para los sitios de comercio electrónico de Telcel, las actividades representativas están creando plantillas, componentes, páginas, estableciendo rutas y configurando servicios OSGi con el objetivo final de permitir a los usuarios buscar los servicios y productos en línea. y durante este trabajo estuve usando tecnologías como HTML5, HTL, CSS3, Sightly, JavaScript, JSP, Java, Servlets.

En el CIAT también brindé apoyo en un proyecto empresarial de Gran Escala para Grupo Posadas Corporativo. Proyecto que ha consistido en la migración de su sitio de servicios hoteleros (login, problemas de acceso, recuperación de contraseña, búsqueda de disponibilidad de reservas, registro de reservas, pago de reservas, consulta de reservas, modificación de reservas, eliminación de reservas, envío de notificaciones por mail, creación de perfiles de usuarios, creación de usuarios, panel de administración, sesiones basadas en roles, analítica, SEO, búsqueda avanzada de reservas) migradas y desarrolladas desde solo tecnologías Java a SPA JavaScript/Vue/ React en el Front End y Microservicios Java/Spring Boot en el Back End. Además de funcionalidades adicionales del original como implementación de roles/privilegios de usuario, integración de LifeRay para activos digitales, análisis digital y SEO con API de Google.

Este ha sido el proyecto de modalidad a distancia más grande en el que he participado, he aprendido a trabajar en un equipo multidisciplinar con profesionales de diferentes formaciones, QA's, BA's,

Diseñadores, PM y compañeros desarrolladores que trabajan bajo el marco Scrum. Mi participación en este proyecto ha sido desde el inicio del mismo, desde el inicio del primer Sprint, mis actividades como desarrollador han sido como desarrollador Back y Front End, en cuanto a desarrollo Front-End se refiere he estado desarrollando el correspondiente SPA del proyecto donde he aplicado buenas prácticas de desarrollo, diseñando y construyendo interfaces de usuario receptivas, móviles primero e implementando la funcionalidad del usuario. Siempre con el objetivo de crear código modular donde cada componente sea altamente reutilizable y con un propósito definido, asegurando un rendimiento óptimo, evitando la duplicación de código aplicando los principios S.O.L.ID y TDD para la suite de codificación y pruebas unitarias (usando Jest.js). Codificación utilizando funciones modernas de ES6 con el uso de Babel para compatibilidad con diferentes tipos de navegadores. Uso de paquetes NPM para diferentes propósitos como validación de formularios, documentación de componentes, internacionalización, axios para consumo de servicios web tipo REST, encriptación, uso de localStorage con sus respectivos polyfills para compatibilidad con navegadores, desarrollo completamente en JavaScript y utilizando los beneficios de Vue .js/React.js como Vuex/Redux/ Flux para la gestión de estado y vue-router/react-router para el enrutamiento de componentes. Otro punto interesante fue mi experiencia con un alto nivel de concurrencia donde la programación asincrónica era

necesario, sin embargo, JavaScript y sus propiedades de promesas y async / await fueron muy útiles para poder procesar correctamente los datos consumidos desde la API, evitando errores de tiempo de ejecución. Debido a que en este proyecto se utilizaron más de cien componentes y se implementó una alta reactividad del contenido consumido a partir de respuestas JSON complejas de servicios web tranquilos con cientos de listas anidadas por acciones de los usuarios, fue necesario un buen análisis del flujo de datos. mantener la persistencia de los mismos independientemente de la jerarquía y asegurando que los datos son constantes y esperados utilizando localStorage y al mismo tiempo respetando los factores de autorización y autenticación a través de la gestión de los JWT consumidos desde la API, respetando el tiempo de vida del token y inactividad del usuario en la aplicación para cerrar sesión y borrar el caché

por completo. En cuanto al diseño, también agregué mejoras a los componentes entregados por el equipo de UX/UI, estas mejoras se realizaron con HTML, CSS, SASS y Bootstrap/Material UI/Vuetify. Finalmente, la documentación del componente se implementa con Vuese.js/Styleguidist para la generación automática de documentación por componente. Para el lanzamiento del código a producción, utilicé webpack para generar los archivos optimizados que se implementarían en un servidor Tomcat. Otro punto importante que aprendí es a tener siempre un flujo de trabajo claro en git, ya que trabajamos en un equipo de varios desarrolladores, es mejor usar compromisos descriptivos y documentar el código claramente para evitar conflictos de fusión de git. En cuanto a mi participación como Back-End desarrollador, he contribuido desarrollando servicios web tipo restful bajo una arquitectura de microservicios utilizando Java/Spring Boot/Data/Seguridad y consumo de servicios web tipo SOAP. Estos servicios web ejecutan las operaciones CRUD básicas pero siempre cuidando de optimizar el rendimiento de la respuesta. En este proyecto tuve la oportunidad de explorar Liferay CMS para consumo de archivos, Swagger para documentación de endpoints y JUnit para pruebas unitarias, trabajando con bases de datos relacionales como Oracle y Postgres y configurando servidor WebLogic y Apache.

Como Desarrollador de Aplicaciones Móviles en el CIAT; Desarrollé la funcionalidad de una aplicación móvil para krispy kreme en colaboración con un equipo de 3 desarrolladores más. Mis principales responsabilidades y actividades consistían en hacer que los componentes de la interfaz de usuario fueran funcionales al integrar la interfaz de usuario con la API RESTful utilizada para esa aplicación, por lo tanto, tales actividades fueron el consumo de servicios web RESTful, la gestión del estado dentro de la interfaz de usuario y la representación de la interfaz de usuario adecuada con el escenario específico también. ya la inversa, capturando la entrada del usuario y enviándola a la API y procesando las respuestas de esos flujos de trabajo, la autenticación y autorización de middleware también formaban parte de mis responsabilidades en lo que respecta a la interfaz de usuario. Usé Ionic y React Native para el desarrollo de esta aplicación móvil.

Durante estos años en el CIAT, trabajé como desarrollador FullStack desarrollando la funcionalidad central de la interfaz de usuario y la integración de la API utilizando el marco basado en TypeScript Angular 2+ y desarrollando servicios web RESTful utilizando el marco Java Spring Boot bajo una arquitectura de microservicios e implementando AWS S3 para algunos de estos. microservicios. Tuve la oportunidad de trabajar solo con desarrolladores senior, siendo yo el único junior en ese momento, lo que me dio el beneficio de aprender algunas buenas prácticas de codificación y patrones de diseño como MVC y principios como SOLID, así como mi primer proyecto bajo un régimen de Scrum, lecciones que me ayudaron mucho a lo largo de mi carrera como Desarrollador desde entonces.

Ingeniero de mantenimiento en mecatrónica. | Coin Acceptors Co. (CoinCo. USA)

(Programación del Sistema Bios de los microprocesadores embebidos de las tarjetas electrónicas de los aceptadores de monedas y billetes)

Marzo 2015 – abril 2017

Mi responsabilidad en CoinCo consistió en programar el microprocesador CoinCo 67095 basado en el microprocesador motorola 68HC09, para el correcto control de los sensores involucrados en la aceptación de diferentes tipos de monedas en Latinoamérica, así como controlar su movimiento mecánico por medio de actuadores electrónicos.

Otras herramientas relevantes

- Angular, React Native, Laravel, Django


Educación

- LinkedIn Learning.com



2014 – a la fecha.


¡Para el desarrollo Web! Lo que necesito la mayoría de las veces lo encuentro aquí.

-  Instituto Politécnico Nacional IPN, CDMX, México.



2006 – 2009.

Maestría en Ciencias Exactas.

-  Instituto Politécnico Nacional IPN, CDMX, México.



1989 – 1994.

Ingeniería en Comunicaciones y Electrónica.