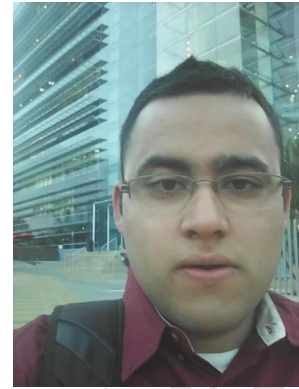


Datos personales

Nombre: Jesús Eduardo Navarro López
 Lugar de nacimiento: Puerto Vallarta, Jalisco
 Fecha de nacimiento: 31 de octubre de 1994
 Estado civil: Soltero
 Idiomas: Español, Ingles (técnico), Ruso (muy básico)
 Domicilio: Isla Pascua #2182
 Jardines de la cruz, C.P. 44950
 Guadalajara, Jalisco
 Teléfonos: Tel. 33 2000 8664
 Cel. 33 2130 7819
 Correos electrónicos: jednavlop@outlook.com
jednavlop@gmail.com
jednavlop@yahoo.com.mx



Formación académica

2012-2016 Ingeniería en Telemática
 Universidad de Guadalajara
 2009-2012 Técnico en informática
 CECYTEN

Cursos y logros

2017 Curso MongoDB: Conceptos básicos
 2015 Diplomado CISCO Telefonía IP
 2014 Concurso Semana de la Ingeniería con desarrollo FloodWarning Internet of Things
 2013 Curso de desarrollo de aplicaciones de red con Python
 2012 Curso de redes inalámbricas aplicadas a la teleradiología

Habilidades y conocimientos

Web	RESTful, AngularJS, MVC, HTML5, CSS3, Javascript, jQuery, Bootstrap3, AJAX
Arquitectura	MVC, Separation of Concerns, N-Tier Architecture
Metodologías	Scrum, Cascada
Microsoft	.NET C#, C/C++ WebAPI2 RESTful + OData, WCF (SOAP + REST), ASMX, MVC, Razor, ASPX, Entity Framework 6 (Database First + Code First + Model First), LINQ To Entities, LINQ to SQL, Sockets Async + Sync, Data Annotations, COM Interop, WinAPI, Multi-threading, Batch scripting, Windows Server 2003 y 2008; GPO, WDS, DNS, DHCP, IIS y AD
Bases de datos	Formas normales, ORM, Replication SQL; MSSQLServer, MySQL, OTL NoSQL; MongoDB (key-value, graphs, documents, wide-column)
VCS	SVN, GIT, TFS
Cisco	Academia CCNA; Routing (2950), Switching (2960), AP-350, VoIP CME 3
Borland y Embarcadero	Delphi 7, XE3 RTTI, BPL Dynamic plugins
Oracle	Java 8; Sockets, LLRP, Multi-threading
ASM	Nociones de Ensamblador (flat x86) y OllyDbg
GNU/Linux	ArchLinux, CentOS, RedHat, Arquitectura de Linux; LAMP, Sytemd, Xorg, Packages

Experiencia

Telemedia 2017

Proyecto	Telemedia Cabina de Sensores para la Salud
Descripción	Se terminó un proyecto inconcluso de cabinas para la medición de ciertos aspectos de la salud; presión arterial, niveles de oxígeno en la sangre, peso y altura. Mediante un dispositivo Arduino para la lectura serial de los sensores y una aplicación Java con capacidades multihilo para la sincronización del dispositivo usando semáforos, se obtenían las diferentes mediciones del usuario para ser registradas a través de un servicio web gubernamental.
Tecnologías empleadas	Java Swing, Java FX, Multi-threading

Benebión 2016

Proyecto	Benebión Web Site
Descripción	Se modernizó el sitio web empresarial para hacerlo responsivo. La migración por módulos desde el sitio ASPX Clásico hacia un nuevo sitio basado enteramente en el patrón de diseño MVC; arquitectura de 3-capas donde se utilizó Entity-Framework para representar el dominio de datos y sus reglas de negocio.
Tecnologías empleadas	.NET C#, Entity Framework 6 Database First, MVC, Javascript, jQuery, Bootstrap, CSS, AJAX

MédicoNet 2016 – Actualidad

Proyecto	TagHub
Descripción	Se creó una aplicación clásica de escritorio y un servicio de Windows para el registro de teclas en crudo (Raw Keys) usando funciones de la API de Windows en la lectura múltiple de códigos de citas provenientes de los cubículos de recepción de muestras en hospitales.
Tecnologías empleadas	.NET C#, WinAPI, Windows Service, Multi-threading

Proyecto	TFVC
Descripción	Se capacitó al personal de desarrollo para comenzar la portabilidad del código actual a un sistema de control de versiones. Se eligió TFS para la integración con Visual Studio y repositorios Git para la compatibilidad con IDEs antiguos. Se lideró la organización del código fuente en proyectos de equipo con un esquema basado en el mantenimiento post-liberación de cada versión.
Tecnologías empleadas	Team Foundation Services, Git

Proyecto	HematixGuard
Descripción	Se migró a la tecnología .NET acorde a los estándares de la empresa, añadiendo funcionalidad y modificando reglas de negocio para satisfacer las necesidades del cliente. El nuevo desarrollo se llevó a cabo usando una arquitectura de N-Niveles diseñando una API RESTful para exponer recursos mediante el protocolo OData 4 y una sitio web reactivo con Angular 2.
Tecnologías empleadas	Capa de datos: .NET C#, SQL, Entity Framework 6 Database First Capa de negocio: .NET C#, Web API 2 OData 4, LINQ To Entities Capa de presentación: HTML5, CSS3, Angular 2, Bootstrap, Cordova

Proyecto	HL7
Descripción	Se implementó el estándar internacional HL7 para el reporte de órdenes de laboratorio y sus resultados a los servidores centrales del IMSS en tiempo real, así como la consulta de pacientes derechohabientes y sus datos demográficos mediante el desarrollo de una librería en .NET C# con una interfaz expuesta a COM de modo que pueda ser llamada a través de Interop. La librería consumía el servicio SOAP mediante un contrato de Windows Communication Foundation con operaciones según el estándar para el envío y recepción de datos usando clases serializables a XML (XSD First). Con una necesidad posterior al desarrollo de la librería, se desarrolló también el endpoint-server que actuara como los servidores del IMSS manteniendo así la misma estructura de mensajes y logrando la recepción de datos de laboratorios y hospitales menores.
Tecnologías empleadas	.NET C#, SOAP, WCF, XML, XSD, MySQL, Windows Service, COM Interop, Multi-threading

Proyecto	CustomerCare
Descripción	Se dió mantenimiento al sistema Web para atención a clientes a través de tickets corrigiendo bugs y migrando en etapas la funcionalidad del sitio ASP.NET Classic a servicios Web para lograr una segmentación en capas (3-Tier Architecture).
Tecnologías empleadas	.NET C#, ASP.NET Classic, SQL, HTML5, CSS3, Javascript

Proyecto	Capturador por lotes
Descripción	Se desarrolló una aplicación de escritorio clásico para atender las necesidades urgentes de captura y procesamiento de lotes de órdenes de laboratorio (tubos de ensayo) en los importantes laboratorios maquiladores IMSS y privados a nivel internacional. Se creó un dominio de datos a través de un ORM a la base de datos principal de los laboratorios de la empresa, para encapsular las reglas de negocio tal y como indica la arquitectura N-Tier. Se hizo un uso intensivo de LINQ To Entities para agilizar las operaciones CRUD. Se crearon módulos diversos de utilidad para importar y exportar datos de órdenes de laboratorio de sistemas de terceros mediante el procesamiento de Schemas XML estándares y tareas multihilo.
Tecnologías empleadas	.NET C#, Multi-threading, XML, XSD, LINQ, MySQL, ORM

Proyecto	ResolutionMD
Descripción	Se implementó un conjunto de servidores Linux para la disposición de tecnología de imagenología médica en tres dimensiones; almacenamiento de series de imágenes de cortes, documentos adjuntos y otros archivos multimedia en la plataforma DCM4CHEE en cumplimiento del estándar DICOM. Se desplegó también el sitio Web ResolutionMD para la visualización y consulta del repositorio de archivos médicos.
Tecnologías empleadas	Linux RedHat, ResolutionMD, DCM4CHEE

Proyecto	Visor de buzones
Descripción	Se creó un sitio Web para la consulta y visualización del estado de envío de órdenes de laboratorio entre hospitales y laboratorios maquiladores. Mediante consultas AJAX y CORS habilitado se realizaba la consulta de datos de diversas fuentes para la recopilación, comparativa y visualización de los mismos en un dashboard centralizado.
Tecnologías empleadas	.NET C#, ASP.NET Classic, XML, XSD, HTML5, CSS3, Bootstrap, Javascript, jQuery, Ajax, CORS

Proyecto	MiniHematix
Descripción	Se creó un sitio Web para el mantenimiento y visualización de los datos históricos de sistemas de banco de sangre de terceros, como parte de los requerimientos de conservación de datos por licitación. La arquitectura del sitio N-Tier demandó la segmentación en capas; SQL y Entity Framework 6 para la capa de datos, MVC y LINQ To Entities para la capa de negocio y la capa de aplicación, ésta última apoyándose de las nuevas tecnologías Web; Razor, HTML5, CSS3, jQuery y Bootstrap.
Tecnologías empleadas	.NET C#, ASP.NET MVC, Razor, Entity Framework 6 Database First, LINQ To Entities, MS SQL, HTML5, CSS3, Javascript, jQuery, Ajax, Bootstrap

Proyecto	HematixGuard
Descripción	Se creó un sistema para la vigilancia y ubicación de unidades de sangre de múltiples bancos a través de un inventario en tiempo real usando antenas y etiquetas RFID. El programa principal, codificado en Java, encendía antenas para leer los datos de las etiquetas RFID de cada bolsa de sangre usando el protocolo LLRP para comunicarse con los lectores de radiofrecuencia. Así mismo, hacía uso de un mapa hash para ubicar cada bolsa y construir el inventario. Se creó un sitio web dinámico con una vista cúbica de los datos del inventario con la funcionalidad adicional de poder encontrar una unidad de sangre con un grupo sanguíneo y un hemocomponente específico en cuestión de segundos, desplegando las unidades en existencia y la ubicación geográfica de sus bancos de sangre en un mapa de Google.
Tecnologías empleadas	Java, Multi-threading, LLRP, PHP, HTML5, CSS3, Bootstrap, Javascript, jQuery, Ajax, Google API, MySQL, Linux CentOS