

Diseño de un módulo de monitoreo con cámaras IP adaptado a un ERP Open Source

Design a monitored module for IP cameras adapted to an ERP Open-Source

José Miguel Zenteno Ríos¹, María Janai Sánchez Hernández¹, José Juan Hernández Mora¹, Jesús Adán Cortés Campech²

¹Tecnológico Nacional de México / Instituto Tecnológico de Apizaco, Fco I Madero s/n Barrio de San José. Apizaco, Tlaxcala, México. CP. 90300

²Microside Technology SA de CV. Calle Ignacio Allende #4B Col. Centro CP. 90000, Tlaxcala Centro

* Correo-e: pepe-zenteno@hotmail.com

PALABRAS CLAVE: RESUMEN

ERP (Planeador de Recursos Empresariales), MVC (Modelo vista controlador), cámaras IP (Cámara de red), Open Source (Código abierto).

This article presents a proposal of a monitored module for IP cameras adapted to an ERP Open-Source. The module is designed with a web interface for two types of users (Root, user) with basic functions of CRUD ("Create, Read, Update and Delete") separated by the established metrics for each determinate user. This proposal was implemented with MVC (Model View Controller) standard that divides the code structure into three parts to have the best distribution and access to the different functions of the module. The result obtained is the streaming that it's possible uploaded and downloaded in real time to gain more security from the company in handling information. This module will grant companies, which implement it, with the optimization of technological resources.

KEYWORDS: ABSTRACT

ERP (Enterprise Resource Planning), MVC (Model view controller), IP cameras (Network Camera), Open Source (Open Source).

En este artículo se presenta la propuesta de un módulo para monitoreo de cámaras IP adaptado a un ERP Open-Source (Planeador de Recursos Empresariales de código abierto). El módulo está diseñado con una interfaz web que maneja dos tipos de usuario (Root, usuario) con las funciones básicas de CRUD ("Crear, Leer, Actualizar y Borrar") separadas por las métricas establecidas para cada usuario determinado. Esta propuesta se implementó con el patrón MVC (Modelo vista controlador), que divide la estructura del código en tres partes para tener una mejor distribución y acceso a las diferentes funciones del módulo. Dentro de los resultados obtenidos se genera el streaming, que a su vez se almacena y descarga en tiempo real para brindar mayor seguridad a la empresa en el manejo y administración de su información. Este módulo permitirá a las empresas que lo implemente la optimización de recursos tecnológicos.

Recibido: 02 de enero de 2021 • **Aceptado:** 20 de mayo de 2021 • **Publicado en línea:** 4 de junio de 2021

I Introducción

Existe una extensa variedad de sistemas ERP (Planeador de Recurso Empresariales), originalmente conceptualizado como un software de escritorio caracterizados por ayudar a las empresas a predecir la demanda y estimar cantidades de materiales necesarias para la producción, pero a través del tiempo y por los avances tecnológicos, se impulsó a nuevas versiones en las que se van integrando cada vez más áreas organizativas como Marketing, Contabilidad, finanzas, Recursos Humanos, en una base de datos que permita acceder a información en menor tiempo, pero sobre todo que ayude a la toma de decisiones. [1]

Derivado de este impulso tecnológico surgen diferentes alternativas de ERP que actualmente se encuentran disponibles en la red, por ejemplo los ERP SaaS (Software como un servicio)[2], On Demand (Bajo demanda) [3], Open Source (Código abierto)[4], entre otros, y cada uno de estos ERP cuenta con funcionalidades únicas que se adaptan a las necesidades esenciales de las empresas que los implementan, sin embargo, surgen nuevas necesidades en el sector empresarial y no todos los ERP disponibles tienen todas las herramientas para satisfacerlas. Un ejemplo de esto suele ser la seguridad de la empresa que, actualmente está cubierta por sistemas de cámaras de video vigilancia. Estos sistemas son totalmente independientes de los sistemas de gestión empresarial. El interés en el desarrollo de este trabajo viene centrado principalmente en desarrollar sistemas que sean económicos y de fácil integración para las empresas, además que estos funcionen de forma eficaz y eficiente. Dando pie a la optimización de recursos.

La propuesta radica en el diseño e integración de un módulo de monitoreo/vigilancia integrado a un ERP de tipo OpenSource. Este módulo incluye generación de streaming, descarga y almacenamiento en tiempo real, para que permita a la empresa u organización que lo implemente un fácil y seguro manejo de su información. En este módulo se permite la conexión de cámaras de video por vía IP, por lo que se facilita su configuración. La arquitectura planeada para este módulo es el patrón MVC que nos permite una mejor distribución y manejo de datos. Al realizar el guardado de la información de video se da la oportunidad a tener un respaldo de la información que puede estar disponible para su revisión y análisis posterior.

2 Estado del arte

ERP de código abierto: implementación de ODOO en Micro, Pequeñas y medianas empresas

Natalia, L., Fredy, Jingga (2017), hablan sobre el nicho de mercado que es indonesia y se basa en las empresas

pequeñas y medianas. Estas empresas deben competir contra otras más grandes que tienen el capital suficiente para derrotarlos, por ello la necesidad de las pequeñas empresas de evolucionar y adaptarse para obtener las ventajas ante la globalización tecnológica.

Hay una gran cantidad de software ERP de código abierto disponible en el mercado como: webERP, Compiere, PostBooks, ERP5, Odoo, Front Accounting, etc. Cada software tiene sus propias debilidades y fortalezas. Después de este análisis los investigadores eligieron el ERP Odoo como la mejor opción, Odoo son ERP de código abierto que tienen funciones en la gestión de proyectos, gestión de relaciones con los clientes, recursos humanos, ventas. Contabilidad, fabricación e inventario.

1. El resultado de producción se registra en orden y se puede auditar. Un documento ordenado con la cantidad de documentos generados automáticamente por los sistemas.
2. Orden de compra registrada en sistemas y se pudo verificar la validez con buen recibo y factura. La integración entre orden de compra y factura disminuye la tasa de error en el procesamiento de pagos.
3. Entrada de diario para procesos de producción y compras. Informe financiero que puede ser utilizado por la gerencia para fines de evaluación de la empresa. [5]

Integración del marco de análisis de sentimientos Twitter y Odoo ERP

Hirikesh D., Zarine, C (2017), describen una forma de integrar al ERP Odoo con una aplicación de internet "Twitter", es un proyecto en el que consiste el análisis de sentimientos en tiempo real obtenidos de un microblogging involucrados en un proceso ERP. Dentro del marco propuesto, Odoo será responsable de la obtención de datos de los clientes, tales como, sus preferencias de compras, horarios de mayor audiencia, etc. y se emplean métodos estadísticos para extraer los datos más importantes y desechar la basura. El proyecto consiste en tres módulos: Filtrado, análisis y tablero. En el módulo filtrado toma tres entradas: tweets entrantes en tiempo real, una lista de eventos y la lista de productos de Odoo Erp. Cada tweet entrante se verifica con la lista de eventos y productos para verificar su relevancia. Se dice que un tweet es relevante sólo si la oración del tweet contiene el nombre del producto / evento (que contiene una o más palabras), enumerados en su orden respectivo. en este caso los emoticones y hashtag son características especiales del tweet. Este módulo se creó utilizando el framework Spark. El primer paso es traer el nombre de todos los productos y eventos de Odoo ERP para generar su hashtag correspondiente. El módulo de análisis es el responsable de analizar y extraer la opinión de los tweets

y de impulsar esos sentimientos en el ERP de Odoo. La entrada del módulo son todos los tweets obtenidos del módulo de filtrado. El módulo de filtrado procesa un lote de tweets entrantes en tiempo real cada dos segundos. Durante estos dos segundos, la cantidad de tweets entrantes puede ser 0 o mayor. Esto lleva a que se ingrese una cantidad variada de tweets en el módulo de análisis. En el módulo se muestra un tablero construido en Odoo ERP, es responsable de mostrar los sentimientos casi en tiempo real de los productos / eventos en gráficos que potencialmente contribuyen en el proceso de toma de decisiones de la organización. En conclusión, una gran variedad de datos no estructurados disponibles para ser analizados de los microblogging, esta información puede ser valiosa para la empresa y su toma de decisiones [6].

Integración de internet de las cosas (IOT) con la aplicación de planificación de recursos empresariales en las industrias de fabricación

I.s. Stephan, T., Vinay, S., V.N. Sundharam, (2018), señalan la importancia del (IOT) internet de las cosas enfocado a las industrias, hoy en día la forma tradicional en la que se vende algún producto o servicios a cambiado ya que este método quedó obsoleto, esto ha orillado a las industrias a cambiar su forma del diseño o creación de los productos, así como su forma de los sistemas de manufactura para lograr una mayor eficiencia y ganancia. Esto se puede lograr mediante la IOT les dará nuevas oportunidades de crecimiento, ahorro dentro de las empresas con el fin de no quedar atrás de la globalización y caer en bancarrota. Los líderes empresariales entienden que el éxito a largo plazo en esta era de rápidos cambios tecnológicos y cambios económicos globales requiere una nueva forma de pensar y operar. De hecho, el manual corporativo se está reescribiendo y reemplazando por uno que lleva la agilidad empresarial a un nivel nunca antes visto. En este entorno agresivo, fomentar la innovación es la máxima prioridad estratégica para las industrias. El Internet de las cosas ha cambiado la función empresarial a nivel mundial de manera diferente y la industria manufacturera no es una excepción. Al combinar los mundos de producción real y virtual a través de Internet, IoT hace posible conectar todas las partes del proceso de producción: máquinas, material, hombre y método. Esto significa que las máquinas y los productos pueden comunicarse entre sí y administrarse por sí mismos. Las plataformas de servicios y sistemas basados en software desempeñarán un papel importante en la industria manufacturera del mañana. El documento tiene como objetivo construir un marco de IoT para la industria manufacturera. Vincula al ERP con dispositivos industriales y optimiza el funcionamiento en función de la entrada / salida. El documento tiene como objetivo construir un marco de IoT para la industria manufacturera. Vincula ERP con dispositivos

industriales y optimiza el funcionamiento en función de la entrada / salida. [7]

Herramienta de desarrollo para la plataforma Odoo

Shrif, H., Bashir, I., Amr, A., Mohammed, A, (2017), proponen la creación de una herramienta de desarrollo que ayude a los desarrolladores a crear módulos de forma rápida, fácil y flexible. Esta propuesta está centrada principalmente en el ERP Odoo ya que es considerada de tipo abierta y es una de las más utilizadas. Esta propuesta surge dado que empresas y desarrolladores están creando sistemas de este tipo y por lo tanto necesitan una herramienta que facilite el trabajo. Dentro de las herramientas utilizadas para el proyecto fueron HTML, CSS, JavaScript para algunas acciones del lado del cliente, cambios en la interfaz de usuario según las acciones de los usuarios, validación del lado de los clientes, jQuery para la interacción con HTML y JavaScript, Django como framework de alto nivel que permite un desarrollo limpio y ordenado además de ser libre. Django les permitió crear y administrar el servidor y crear una base de datos. Para usar el sistema el usuario debe tener una cuenta e iniciar sesión para crear, editar, compartir y descargar, si el usuario no tiene una cuenta debe crearla, en la página de inicio se muestran todos los módulos creados y el enlace para crear nuevo módulo y enlace a la comunidad, si el usuario elige crear un módulo desde cero, el editor se mostrará cuando el usuario ingrese la información del módulo, y si el usuario eligió crear módulo desde la base de datos, entonces el usuario debe ingresar toda la información de la base de datos antes de la visualización del editor. Ver **Figura 1**. Página de edición de módulos.

Al mismo tiempo, se generarán archivos de módulo, la ruta se agregará a la ruta del complemento odoo y el módulo será instalado en odoo.

La plataforma proporciona una comunidad para realizar la comunicación entre los desarrolladores de Odoo y otros. Además, la plataforma contiene un repositorio de módulos para controlar los cambios que realizó el desarrollador en los módulos y recupera los cambios en cualquier momento. La plataforma también proporciona ayuda y buena documentación que sirve como manual de usuario para el desarrollador para que puedan interactuar, intercambiar el conocimiento, las habilidades y la experiencia de los demás.

Una de las principales tendencias de la ingeniería de software es producir aplicaciones que ayuden a los desarrolladores a reducir el número de líneas de código que escribe el desarrollador hasta que llegan a la línea cero del punto de código, lo que lleva a reducir el tiempo y el esfuerzo. [8]

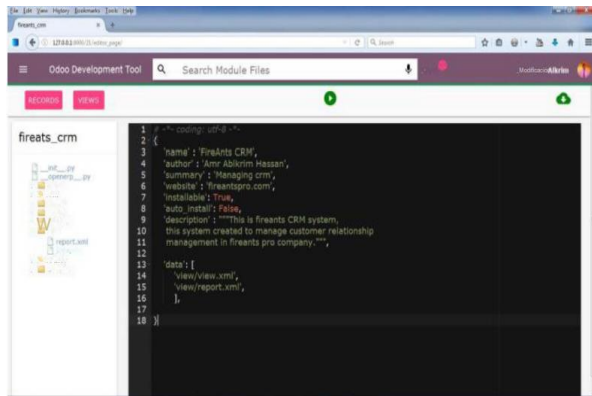


Figura 1. Página de edición de módulos [8]

II Metodología

El modelo de Cascada también conocido como modelo tradicional ó como modelo lineal-secuencial [9], generalmente no es usado tal cual, debido a que esto implicaría que los requisitos deberán conocerse a la perfección ya que los cambios no son factibles, lo que significa diseños exactos y sin errores. Por lo cual surge una variante que es la cascada realimentada **Figura 2. Modelo de cascada realimentada**, esta adaptación permite que se produzca alguna realimentación entre etapas, esto da una mejor oportunidad al desarrollo de software en los cuales pueden surgir cambios. Este modelo se adapta perfectamente a las necesidades de desarrollo de este módulo. Las etapas son las mismas que la cascada tradicional, pero se permite retroceder a la etapa anterior e incluso se puede regresar a las varias anteriores si las necesidades del software así lo requieren.

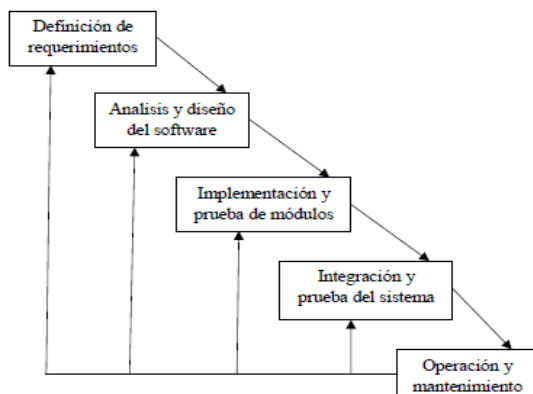


Figura 2. Modelo de cascada realimentada

A continuación, se describen las actividades realizadas en cada etapa.

1.- **Obtención de requerimientos.** La adquisición de los requisitos para el desarrollo de nuestro proyecto se llevó a cabo mediante pláticas con los usuarios del ERP dentro de una organización, se llevó a cabo de esta manera ya que se requería conocer las principales necesidades para definir de forma precisa los requisitos funcionales y no funcionales y de esta manera plantear algunos prototipos. Una de las actividades clave que se desarrolló en esta etapa, fue la definición del ecosistema de desarrollo. Para fines de nuestro desarrollo se hace la propuesta de un sistema de tipo código libre, como se puede ver en la **Figura 3. Ecosistema de desarrollo**, en la cual se representan todas las herramientas de desarrollo. Este ecosistema se diseñó en base al análisis de las necesidades, pero también en base a la compatibilidad del ERP al que se integra. La mayor parte son lenguajes de código libre. El ERP Open Source para el que se decidió generar el módulo de igual forma cumple con los requerimientos de desarrollo de Python.



Figura 3. Ecosistema de desarrollo

2.- **Análisis y diseño del software:** Una vez analizados los requerimientos, se comenzó con el diseño de una arquitectura funcional basada en el patrón MVC como se puede apreciar en la **Figura 4. Arquitectura del Patrón MVC**. Esta arquitectura permite que exista una separación en la estructura del modelo, es decir, permite que se realice el manejo de datos de acuerdo a las reglas establecidas, por ejemplo, hacer cada tarea de acuerdo a la información recibida, en la Vista están los perfiles de los usuarios puesto que en este caso se contará con dos, con lo cual también se define el esquema de seguridad del sistema, ya que sólo se permitirá el acceso a información a las personas asignadas a esta función. El Controlador se ejecutará directamente en el servidor del ERP open source y es ahí donde se desarrollan todas las tareas principales que se podrán ejecutar.

De igual forma se realizó la parte del modelado UML de los casos de uso para los usuarios definidos, ver en la **Figura 5. Casos de uso de usuarios finales** del módulo

de videovigilancia, en el cual se describen las funciones a que cada usuario tendrá acceso.

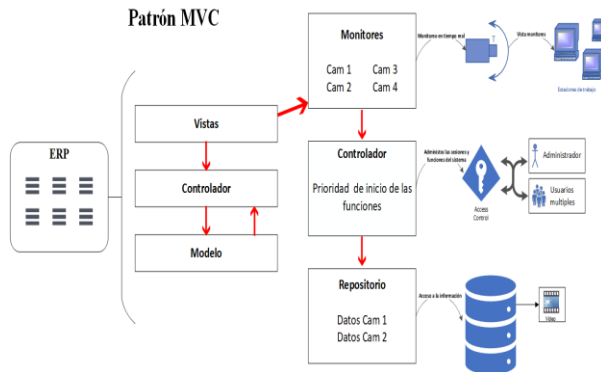


Figura 4. Arquitectura del Patrón MVC



Figura 5. Casos de uso de usuarios finales

3.- **Implementación y pruebas de utilidades.** Se desarrollaron diversas pruebas al sistema para identificar algunos errores como la conectividad entre dispositivos, al término de las pruebas regresamos a una etapa anterior con la finalidad de redefinir algunos aspectos del sistema donde se encontraron fallos importantes.

4.- **Integración y pruebas del sistema:** En esta etapa el módulo ya estaba integrado al ERP y se corrigieron detalles importantes. Se diseñaron pruebas en la integración para verificar un correcto funcionamiento en tiempo real, al término de estas pruebas se descubren nuevos errores y se desarrollan mejoras del módulo.

5.- **Operación y mantenimiento:** Se confirmó la ejecución dentro del ERP.

III Resultados

Se implementó el módulo de videovigilancia en la suite de Anaconda [10], utilizando su gestor de programación Spyder en el lenguaje interpretado Python con la librería Opencv[11], tal herramienta de cómputo es una de las más utilizadas para el manejo de video en el ámbito de la programación. En cuanto al procesamiento de imágenes y video en tiempo real según el modelo, tipo de cámara, se analizó el comportamiento del sistema en cuanto a funcionalidad y uso. Se realizaron diversas pruebas de acceso, verificando los inicios de sesión de acuerdo a los usuarios determinados.

3.1 Pruebas con herramientas de testing Apache JMeter

Se realizaron pruebas con “Apache JMeter” [12] se tratan de pruebas de velocidad y de la validación del envío de formularios, ya que son la parte primordial en el módulo de videovigilancia si estos no funcionan como debe ser el video de las videocámaras no se reproducirá de manera correcta.

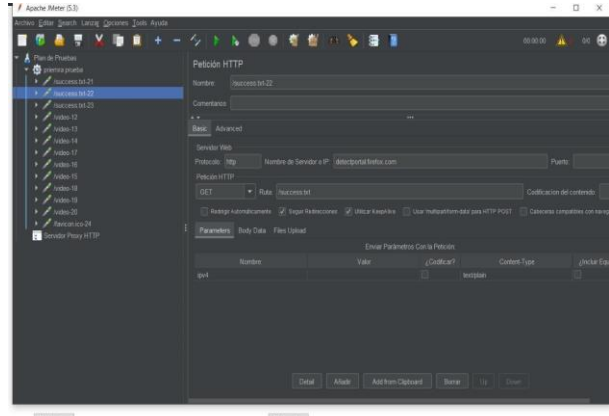


Figura 6. Análisis del envío de los datos

Como se muestra en la **Figura 6. Análisis del envío de los datos**, el envío se realiza mediante hilos para evaluar su correcto funcionamiento. Cada acción en este caso se denomina hilo, el usuario inserta los datos del formulario para la instalación de una nueva videocámara, la dirección se envía mediante variables independientes que se guardan para poder editarlas en algún futuro en ajustes.

3.2. El procesamiento de las imágenes

El procesamiento de las imágenes depende en gran medida de los parámetros de entrada asignados para cada imagen como la cantidad de frames capturados y de igual manera su peso, que es parte esencial para un procesamiento rápido. En la primera prueba se modificaron la cantidad de frames obtenidos por prueba. Como se aprecia en la **Figura 7. Imagen ejemplo, verificación de tamaño y peso.** Se selecciona una imagen ejemplo y en propiedades se realizó la

comprobación del tamaño y peso para facilitar el procesamiento. Los mejores resultados fueron 10 frames por segundo a una escala de ancho = 1280 px y altura 720 px para una mejor calidad y rapidez en el procesamiento.

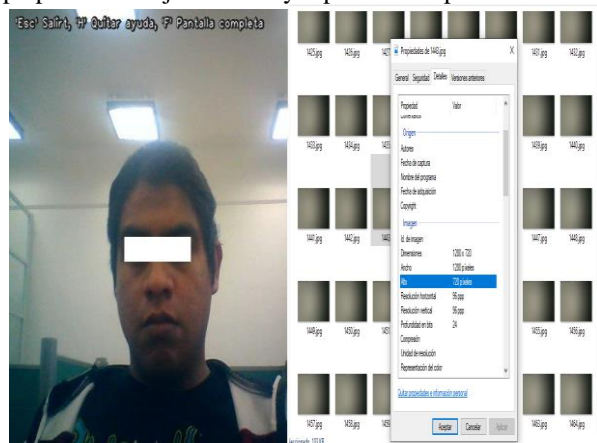


Figura 7. Imagen ejemplo, verificación de tamaño y peso

3.3. El procesamiento de vídeo

Como se comentó anteriormente, el tamaño del archivo influye en la velocidad de procesamiento. El objetivo del experimento es encontrar la mejor manera de obtener el archivo de vídeo con la mejor calidad y tamaño. Como se puede ver en la **Figura 8. Vista del video**, el formato del video es .avi que permite la ejecución en cualquier reproductor de video.

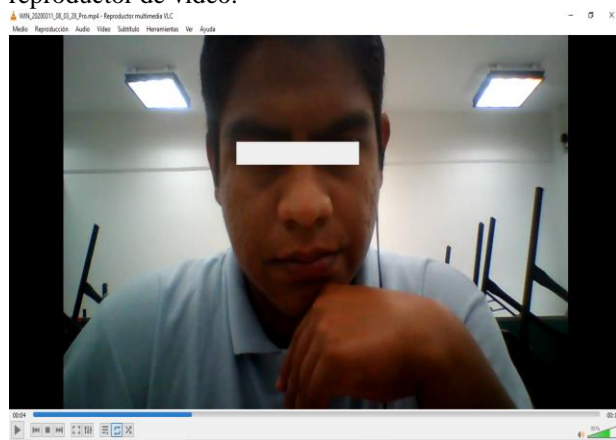


Figura 8. Vista del video

3.4. Comportamiento general del sistema

Se realizó el análisis de las interfaces y se confirmó su ejecución en tiempo real, considerando las funciones de un usuario, para iniciar grabación y descargar. En la **Figura 9. Ejecución de video y descarga**, se muestra la interfaz de la pantalla principal del centro de monitoreo, en dos salas en las que se pueden apreciar datos que el usuario puede capturar, como nombre, definición de características, iniciar grabación y dirección a la cual puede ser guardado el archivo.

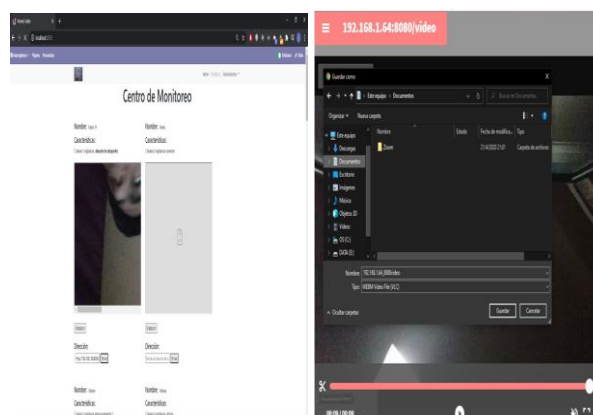


Figura 9. Ejecución de Video y Descarga

3.5. Seguridad en la autenticación de usuarios

De acuerdo a la guía de pruebas OWASP de intrusión de aplicaciones web se desarrolló la prueba de modo pasivo. [13] El módulo de monitoreo tiene dos usuarios definidos, cada uno de ellos lleva acciones particulares en el sistema. En el momento de estar llevando a cabo un proceso de monitoreo de un área en particular y al contener información sensible en cuanto a los procesos de la empresa, se requiere de seguridad en el acceso de información, por lo tanto, se realizaron pruebas de autenticación, la cual consistió en la captura de los campos correo electrónico y contraseña. En caso de realizar una captura incorrecta, ya sea del correo electrónico o contraseña el sistema enviará un mensaje de usuario o contraseña incorrecta. Lo que indicará que el usuario nuevamente deberá volver a realizar la captura de estos datos si es que sólo se equivocó en la captura, como se puede observar en la **Figura 10. Pantalla de autenticación**. Si la información capturada de usuario fue correcta dará acceso a la pantalla principal del ERP, **Figura 11. Interfaz del sistema principal del ERP**, en la que se puede observar del lado superior izquierdo la opción de videovigilancia y podrá realizar las operaciones según los permisos asignados al tipo de usuario.

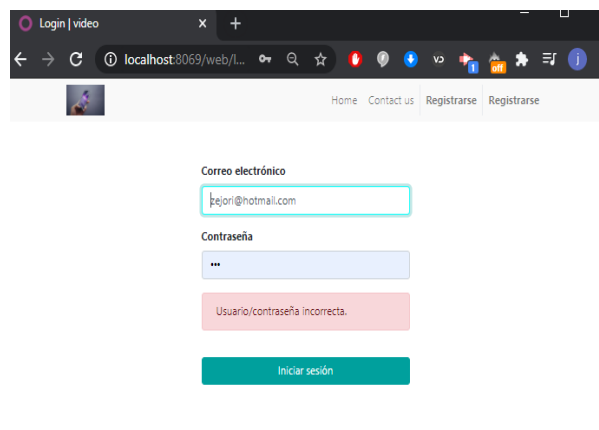


Figura 10. Pantalla de autenticación

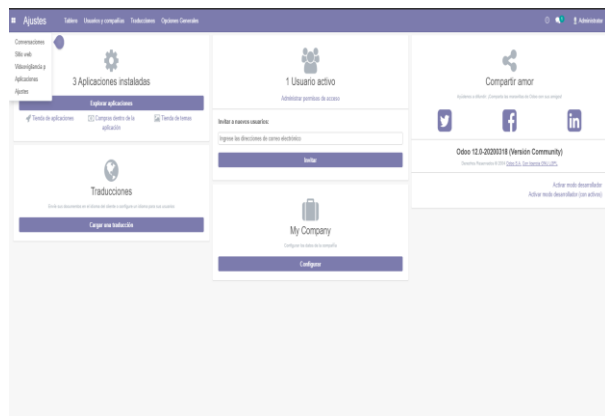


Figura 11. Interfaz del sistema principal del ERP

IV Conclusiones

Al concluir el desarrollo del módulo de videovigilancia el cual denominamos “Control de videovigilancia”, podemos determinar que el comportamiento del módulo desarrollado depende mucho del procesamiento de los datos, en este caso el vídeo en formato .avi es el que permite un mejor desempeño. En las pruebas realizadas los mejores resultados fueron con 10 frames por segundo a una escala de ancho = 1280 px y altura 720 px para una mejor calidad y rapidez en el procesamiento. En cuanto a la configuración de cámaras pueden configurarse sin problema todas las cámaras del tipo IP.

La arquitectura del modelo nos permitió dividir la lógica y realizar una primera versión que puede hacer que este proyecto sea escalable. El uso de OpenCv y Python [14] permiten que en un futuro se puedan implementar algunos algoritmos de visión artificial para poder así identificar objetos clave o acciones específicas en áreas en donde sean instaladas las cámaras, entre otros.

La herramienta de prueba “Apache JMeter” fue eficiente para medir el rendimiento de carga de los datos usados en los formularios para los dispositivos (cámaras) y está enfocado en su totalidad a aplicaciones web. Algunas pruebas se realizaron en conjunto con el navegador web “Mozilla Firefox” ya que es de software libre y es compatible con nuestra herramienta de testing, los resultados fueron satisfactorios ya que en las pruebas realizadas se puede notar que el envío de datos, hablando de los formularios cumplen su función de registrar los datos del dispositivos para guardarlos en unas variables y servir como intermediario en la comunicación con la cámara deseada, otros aspectos de las pruebas es para validar la seguridad en los datos ya que estos están protegidos para evitar accesos no deseados mediante login y valores asignados.

Podemos concluir que el uso de la metodología cascada realimentada nos permitió llevar un proceso de desarrollo de forma ordenada, pero sobre todo nos ayudó a realimentar etapas en las que surgieron nuevos requerimientos o bien donde se detectaron errores y por lo tanto fue posible corregirlos. La propuesta de este módulo cumple con los requisitos de funcionalidad establecidos, y por lo tanto es una aplicación funcional.

BIBLIOGRAFÍA

- [1] Molina, G. J., & Pérez, B. E. (2007). *Sistemas de Información Integrados [ERP]*. Madrid, España: Asociación Española de Contabilidad y Administración de Empresas.
- [2]Microsoft. ¿Qué es SaaS?. Recuperado el 20 de septiembre de 2020. <https://azure.microsoft.com/es-mx/overview/what-is-saas/>
- [3]Velosio. ERP On-demand Vs. On-premise. Read more at: <https://erpsoftwareblog.com/cloud/2015/11/erp-on-demand-vs-on-premise/> | ERP Cloud Blog Recuperado el 20 de septiembre de 2020. <https://erpsoftwareblog.com/cloud/2015/11/erp-on-demand-vs-on-premise/>. Noviembre 2015
- [4]Copyright ©2020 Red Hat, Inc.¿Qué es el open source o el software open source (OSS)?. Recuperado el 23 de septiembre del 2020 <https://www.redhat.com/es/topics/open-source/what-is-open-source>
- [5]Natalia, L., Fredy, Jingga. Open Source ERP: ODOO Implementation at Micro Small Medium Enterprises. <https://ieeexplore.ieee.org/document/8273562>. 2017. 1-5
- [6]Hirikesh D., Zarine, C. Sentiment Analytics Framework Integrating Twitter and Odoo ERP. <https://ieeexplore.ieee.org/document/8285994>. 2017, 1-7.
- [7]I.s. Stephan, T., Vinay, S., V.N. Sundharam. Internet Of Things (Iot) Integration With Enterprise Resource Planning Application In Manufacturing Industries. http://www.iaeme.com/MasterAdmin/UploadFolder/IJMET_09_07_095/IJMET_09_07_095.pdf. Julio 2018. 1-8.
- [8]Shrif, H., Bashir, I., Amr, A., Mohammed, A. Developing Tool For Odoo Platform. <https://ieeexplore.ieee.org/document/7867677>. 2017, 1-7.
- [9]Ingeniería del Software, un enfoque Práctico, Quinta edición, Capítulo: El producto. Autor: Roger S. Pressman, Editorial: McGraw Hill, Año: 2003.
- [10]©2020 Anaconda Inc. Recuperado el 20 de septiembre de 2020. <https://www.anaconda.com/>
- [11]©Copyright 2020, OpenCV team. Recuperado el 20 de septiembre de 2020. <https://opencv.org/>.
- [12]Copyright © 1999-2020, Apache Software Foundation. Recuperado el 20 de septiembre del 2020. <https://jmeter.apache.org/>
- [13]©2002-2008 OWASP Foundation. Recuperado el 20 de septiembre de 2020. https://owasp.org/www-pdf-archive/Gu%C3%ADa_de_pruebas_de_OWASP_ver_3.0.pdf
- [14]Copyright ©2001-2020. Python Software Foundation. Recuperado el 20 de septiembre del 2020. Python <https://www.python.org/>

Acerca de los autores



José Miguel Zenteno Ríos. Ingeniero en Tecnologías de la Información, Universidad Politécnica de Tlaxcala, 2017. Actualmente estudiante de la Maestría en Sistemas Computacionales, Instituto Tecnológico de Apizaco, sus temas de interés son el procesamiento de datos y aplicaciones web.



María Janaí Sánchez Hernández. Licenciada en Informática, Instituto Tecnológico de Apizaco, 2001. Maestra en Ciencias en Ciencias de la Computación, Instituto Tecnológico de Apizaco, 2006. Docente del Departamento de Sistemas y Computación, 2006. Profesora de la Maestría en Sistemas Computacionales, del Instituto Tecnológico de Apizaco del TecNM.



José Juan Hernández Mora. Ingeniero en Computación, Universidad Autónoma de Tlaxcala, 1994. Maestro en Ciencias en Ciencias Computacionales por el Centro Nacional de Investigación y Desarrollo Tecnológico del TecNM, 2003. Profesor Investigador del Tecnológico de Apizaco del TecNM. Docente de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Apizaco.



Jesús Adán Cortés Campech. Ingeniero en electrónica, Instituto Tecnológico de Puebla, 2002. Maestro en Desarrollo de Software por el IT University Puebla, 2012. Fundador y Director Microside Technology, Presidente de Innovación y Tecnología Coparmex, empresa especialista en la generación de nueva tecnología, desde el diseño, integración, capacitación y comercialización de soluciones basadas en dispositivos electrónicos. Experto en integración de plataformas de hardware con sistemas embebidos y diseño industrial, ofrece soluciones a medida basadas en el diseño de dispositivos electrónicos portátiles y fijos.