

Desarrollo de una API Rest para la compañía CEA Systems

Juan Carlos Mimendi Díaz¹, Darío Cubillo Martínez ¹

¹*Instituto Tecnológico Superior de San Andrés Tuxtla, carlos.mimendi@outlook.com*

Área de participación: Sistemas Computacionales

Resumen

En el presente trabajo de investigación se aborda el desarrollo de una API Rest para la compañía CEA System. Donde el objetivo general es evitar la modificación de los sistemas informáticos actuales por cada cliente que solicite una interfaz para obtener los datos de manera externa.

Actualmente la compañía tiene en el mercado software industrial para el control y manejo de assets. *Pipe and Process* y *Electrican and Ingengineering*, ambos sistemas utilizados por diversos clientes en sus diferentes versiones, la implementación de una API Rest con el fin de acceder, modificar o eliminar información que los sistemas generan, creara una apertura en el mercado actual donde la información se prefiere almacenar en la nube considerando los diferentes modelos de distribución SaaS, PaaS, IaaS.

La metodología utilizada fue SCRUM, apoyando esta metodología se utilizó la herramienta Dev Ops de Microsoft.

Palabras clave: API, software, SCRUM, assets

Abstract

In the present research work, the development of an API Rest for the CEA System company is addressed. Where the general objective is to avoid the modification of the current computer systems by each client that requests an interface to obtain the data externally.

The company currently has industrial software on the market for the control and management of assets. Pipe and Process and Electrical and Engineering, both systems used by various clients in their different versions, the implementation of a Rest API in order to access, modify or delete information that the systems generate, will create an opening in the current market where the

information Cloud storage is preferred considering the different distribution models SaaS, PaaS, IaaS.

The methodology used was SCRUM, supporting this methodology the Microsoft Dev Ops tool was used.

Key words: API, software, SCRUM, assets

Introducción

Los sistemas informáticos hoy necesitan planear la demanda de datos que el usuario podría generar durante el uso, con sistemas robustos donde la información realiza operaciones o cálculos cada vez más precisos, incremento de seguridad de la información y garantía de acceso a la información, se generó un modelo de computación donde los datos son accesibles desde cualquier parte del mundo, la computación en la nube.

La transferencia de estado representacional (REST) y acceso a objetos simples (SOAP) son los dos principales servicios web populares utilizados hoy en día. REST está basado en un estilo arquitectónico, mientras que SOAP es un protocolo subyacente. Ambos servicios se utilizan para manejar la comunicación en la red mundial (www). Ambos tienen algunas ventajas e inconvenientes y es decisión del desarrollador decidir qué servicio es mejor usar de acuerdo con sus requisitos, el tiempo de respuesta de una aplicación SOAP toma entre 4 y 7 ms más que una REST, el uso de memoria también incrementa en SOAP conforme va adquiriendo más funcionalidades en comparación a REST. Muchos servicios web que afirman ser una API REST no están impulsados por hipertexto según lo prescrito por REST. Esta situación puede llevar a que las API de REST no sean tan escalables, extensibles e interoperables como promete REST.

Planteamiento del Problema

Las aplicaciones principales de la compañía CEA Systems, Plant 4D Pipe and Process (P&P) y Plant4D Instrument and Electrical (I&E) fueron desarrolladas con requerimientos y tecnologías diferentes.

Plant 4D P&P, es una aplicación de escritorio la cual realiza consultas y actualizaciones directamente a la base de datos, sin la necesidad de una aplicación middleware como una API o un Servicio Web.

Plant 4D I&E, al igual que P&P es una aplicación de escritorio, pero tiene como middleware un Servicio Web, el cual realiza las operaciones que la aplicación de escritorio necesita para su funcionamiento.

Ambas aplicaciones ya se encuentran en uso por clientes de la compañía, algunos de estos clientes solo utilizan una de ellas, esto ha generado que uno de los requerimientos más recibidos sea la posibilidad de realizar la consulta de la información a través de un software del cliente. Para satisfacer este requerimiento es necesario modificar ambas aplicaciones por cada cliente que lo solicite o compartir información confidencial de la lógica en las bases de datos. Lamentablemente realizar estas modificaciones a las aplicaciones crearían un gasto que impactaría en los costos de mantenimiento, generaría un tiempo de desarrollo mayor al tratarse de un requerimiento por cada usuario, esto significa especificaciones diferentes por cada cliente y al final el costo al usuario incrementaría.

Objetivo

Desarrollar una API REST la cual realizará las funciones de obtención y actualización de los datos utilizados por los sistemas I&E y P&P, esto evitará que ambas aplicaciones sean modificadas para la integración con sistemas externos a la compañía.

Revisión literaria.

La transferencia de estado representacional (REST) y acceso a objetos simples (SOAP) son los dos principales servicios web populares utilizados hoy en día. REST está basado en un estilo arquitectónico, mientras que SOAP es un protocolo subyacente. Ambos servicios se utilizan para manejar la comunicación en la red mundial (www). Ambos tienen algunas ventajas e inconvenientes y es decisión del desarrollador decidir qué servicio es mejor usar de acuerdo con sus requisitos, el tiempo de respuesta de una aplicación SOAP toma entre 4 y 7 ms más que una REST, el uso de memoria también incrementa en SOAP conforme va adquiriendo

más funcionalidades en comparación a REST [1].

Muchos servicios web que afirman ser una API REST no están impulsados por hipermedia según lo prescrito por REST. Esta situación puede llevar a que las API de REST no sean tan escalables, extensibles e interoperables como promete REST. La computación en la nube crea una visión innovadora de entorno basada en la red para los usuarios, lo que allana el camino para compartir cálculos y recursos independientemente de la ubicación. El Instituto Nacional de Estándares y Tecnología (NIST) define la computación en la nube como, "Una plantilla para proporcionar el acceso adecuado y cuando sea necesario a Internet, a un grupo colectivo de redes programables, almacenamiento, servidores, software y servicios que se puede emancipar rápidamente, con poca comunicación y supervisión por parte del proveedor" [2].

Multitenancy es una tecnología fundamental que permite que una instancia de aplicación sirva a varios clientes al mismo tiempo para compartir recursos de la nube y lograr una alta eficiencia operativa [3].

Se tienen cinco atributos importantes en la computación en la nube:

- **Escalabilidad masiva:** la computación en la nube brinda la capacidad de escalar a decenas de miles de sistemas, así como la capacidad de escalar masivamente el ancho de banda y el espacio de almacenamiento[4].
- **Elasticidad:** los usuarios pueden aumentar y disminuir rápidamente sus recursos informáticos según sea necesario, así como liberar recursos para otros usos cuando ya no sean necesarios. [2]
- **Pago por uso:** los usuarios pagan solo por los recursos que realmente usan y solo por el tiempo que los requieren[4].
- **Autoabastecimiento de recursos:** los usuarios auto aprovisionan recursos, como sistemas adicionales (capacidad de procesamiento, software y almacenamiento) y recursos de red (Mather, Kumaraswamy y Latif, 2009)[4].

Multitenancy (recursos compartidos): a diferencia de los modelos informáticos anteriores, que suponían recursos dedicados a un solo usuario o propietario, la informática en la nube se basa en un modelo de negocio en el que los recursos se comparten a nivel de red, host y aplicación [3].

En el desarrollo de un software en la nube el proceso fundamental es la especificación de su arquitectura, esta podrá definir qué tan mantenable y evolutivo es el proyecto.

De las arquitecturas disponibles para el desarrollo de una API Rest la más enfocada en un sistema distribuido es del desarrollo de Microservicios, Pahl y Pooyan Jamshidi et. al. [4] mencionan en sus conclusiones que la arquitectura de Microservicios es fácil de mantener, escalable, cada microservicio es independiente de la tecnología del otro y la estabilidad de cada uno no depende del otro.

Los microservicios son un enfoque para el desarrollo de una aplicación única como un conjunto de pequeños servicios y enfoque en los estándares actuales. [5], cada uno ejecutándose en su propio proceso y mecanismos ligeros de comunicación, a menudo un recurso de una interfaz de programación de aplicaciones (API) sobre protocolo de transferencia de hipertexto (HTTP).

Estos servicios están construidos alrededor de las capacidades del negocio y con independencia de despliegue e implementación totalmente automatizada. Existe un mínimo de gestión centralizada de estos servicios, los que pueden estar escritos en lenguajes de programación diferentes y utilizar diferentes tecnologías de almacenamiento de datos [6].

Una API Rest que se encuentra dentro de un microservicio tiene un solo funcionamiento, está desarrollada para un solo propósito. Los microservicios son autónomos y aislados. Autónomo, Existente o capaz de existir de forma independiente; Responder, reaccionar o desarrollarse independientemente del todo [7]. Aislado: Separado de los demás, sucediendo en diferentes lugares y en diferentes momentos [7]. “Eso significa que los microservicios son unidades autónomas de funcionalidades con dependencias poco acopladas de otros servicios y están diseñados, desarrollados, probados y lanzados de forma independiente” [7].

Dentro de las implicaciones más importantes es adaptar al equipo de desarrolladores a un esquema de trabajo donde cada microservicio debe ser independiente de otro, siguiendo esta línea, los entornos de desarrollo, prueba y producción deberán organizarse para respaldar a estos equipos, desarrollando e implementando sus productos de microservicio por separado [3]. Un beneficio clave de los microservicios es la facilidad de mantenimiento. La división de un sistema en servicios independientes y auto desplegables ayuda a los equipos de desarrolladores a probar sus servicios y realizar

cambios independientemente de otros desarrolladores, lo que simplifica el desarrollo distribuido. Por lo tanto, el pequeño tamaño de los microservicios contribuye a aumentar la comprensibilidad del código, mejorando también la mantenibilidad del código [3].

Las API Rest tienen principalmente un protocolo de comunicación http ampliamente difundido y menos restringido por firewall esto lo hace de igual forma vulnerable, dentro de las API Rest se tienen dos conceptos de seguridad, autenticación y autorización.

Autenticación: proceso que implica la verificación de un individuo quién dice ser que es.

Autorización: es un proceso que determina si el individuo tiene derechos, permisos o privilegios para realizar algún tipo de acción sobre algún recurso.

JWT (Json Web Tokens) es un estándar abierto basado en JSON propuesto por IETF (RFC 7519) estos tokens contienen la información de identidad y privilegios del usuario. Esta información se comparte en formato JSON de manera compacta y sencilla. La información es confiable ya que es firmada por la API con un algoritmo HMAC. Contiene 3 partes (Header, payload, signature) codificadas a Base64, separadas por un punto que es entregado a los clientes de una API como llave de acceso.

Este mecanismo le pide a un cliente que almacene el par de tokens públicos y privados calculados por el servidor. En cada comunicación, el cliente utiliza el token público almacenado, el token privado y la marca de tiempo actual para producir un token desecharable, que posteriormente recibe el servidor para su verificación. Con este mecanismo, cada comunicación será válida solo en un período fijo de tiempo, reduciendo así los riesgos de robo de identidad.

Los proveedores de servicios cloud han sido los encargados de implementar un sistema, que simplifica el uso del software, las aplicaciones, las plataformas y las infraestructuras para desarrollar la actividad de las empresas. Estos servicios se agrupan en tres modelos: el IaaS (Infrastructure as a Service), el PaaS (Platform as a Service) y el SaaS (Software as a Service). Cada uno de ellos hace referencia a un recurso puesto a disposición del usuario según un modelo bajo demanda, por lo que no es necesario invertir en una infraestructura para alojar y utilizar los servicios [8].

Software Como Servicio (Software as a Service, SaaS) está siendo utilizado con más frecuencia por las compañías el modelo se refiere a que ahora la Web API esta alojada con un proveedor el cual se encarga del mantenimiento de servidores y seguridad, donde ahora el pago es

por uso de la app o por número de usuarios [9]. Plataforma como servicio (Platform as a Service, PaaS) es una modalidad del cloud computing en la cual un tercero brinda el sistema de hardware y una plataforma de software de aplicaciones. La PaaS es ideal principalmente para los desarrolladores y los programadores, ya que permite que el usuario desarrolle, execute y gestione sus propias aplicaciones sin tener que diseñar ni mantener la infraestructura ni la plataforma que suelen estar relacionadas con el proceso.

Infraestructura como servicio (Infrastructure as a Service, IaaS) brinda la virtualización, el almacenamiento, la red y los servidores. De esta manera, el usuario no necesita tener un centro de datos on-premise ni debe preocuparse por actualizar o mantener físicamente estos elementos [9].

On-Premise se refiere a que la instalación del programa se ha realizado de manera local, en las instalaciones de la empresa y obligando a esta a crear una infraestructura informática compleja con servidores que requieren mantenimiento [9].

Las implementaciones de IaaS se consideraron la mejor opción cuando se trata de costos iniciales y recurrentes, mientras que las implementaciones en las instalaciones se consideraron la mejor opción en términos de seguridad y privacidad. SaaS dominó todos los demás factores y se percibe como la opción preferida por su disponibilidad y confiabilidad, facilidad de implementación, rendimiento y escalabilidad, y el hecho de que es fácil de actualizar [9].

Dentro de SaaS tenemos un enfoque más específico Multitenancy esta es una de las características más recientes en la computación enfocada en la nube, esta característica se refiere a tener un sistema de base de datos con múltiples clientes, es decir una base de datos general donde el esquema de la base de datos esté relacionado al cliente [10]. Esto proporciona escalabilidad y beneficio económico tanto para el proveedor de servicios en la nube como para los usuarios finales [11].

Metodología

Para lograr el desarrollo de la API se utilizó la metodología Scrum, esta está fomentada en el trabajo en equipo donde el principal objetivo es realizar un conjunto de tareas de forma regular y con colaboración con los demás miembros del equipo.

Con esta metodología se pretende llegar a un resultado óptimo. Las actividades que se realizan

están potencialmente creadas para ser competitivos con los tiempos de cada desarrollador.

En Scrum se van realizando entregas regulares y parciales del trabajo final, de manera prioritaria y en función del beneficio que aportan dichas entregas a los receptores del proyecto. Por este motivo, es una metodología especialmente indicada para proyectos complejos, con requisitos cambiantes y en los que la innovación y la flexibilidad son protagonistas [12].

Los hitos de la Metodología de trabajo Scrum son:

- **Sprint**

Es el tiempo que se asigna para realizar tareas relacionadas al proyecto, el tiempo no debe ser mayor a un mes.

- **Sprint planning**

En esta reunión todo el equipo Scrum define qué tareas se van a abordar y cuál será el objetivo del sprint.

- **Daily meeting**

Es una reunión diaria dentro del sprint que tiene como máximo 15 minutos de duración. En ella debe participar, sí o sí, el equipo de desarrollo y el Scrum Master.

En esta reunión diaria el equipo de desarrollo hace las siguientes tres preguntas:

¿Qué hice ayer?

¿Qué voy a hacer hoy?

¿Tengo algún impedimento que necesito que me solucionen?

Esta reunión es la más oportuna para poder inspeccionar el trabajo y poder adaptarse en caso de que haya cambio de tareas dentro de un sprint.

- **Sprint review**

Su duración es de 4 horas para sprints de un mes. En ella el Product Owner presenta lo desarrollado al cliente y el equipo de desarrollo muestra su funcionamiento. El cliente valida los cambios realizados y además brinda feedback sobre nuevas tareas que el Product Owner tendrá que agregar al Product backlog.

- **Sprint retrospective**

La retrospectiva es el último evento de Scrum, tiene una duración mayor a las juntas diarias por Sprint de un mes, y es la reunión del equipo en la que se hace una evaluación de cómo se ha implementado la metodología Scrum en el último sprint.

Es una gran oportunidad para el equipo Scrum de

inspeccionarse a sí mismo, proponiendo mejoras para el siguiente *sprint*.

Azure – DevOps

Como complemento de la metodología *Scrum*, se utilizó la herramienta *Azure Dev Ops* de Microsoft

[13], contiene servicios que apoyan en la administración de las actividades y tiempos en los proyectos en desarrollo de software.

Al utilizar estas herramientas con la metodología *Scrum*, el avance de los objetivos del *sprint* se pueden visualizar gráficamente.

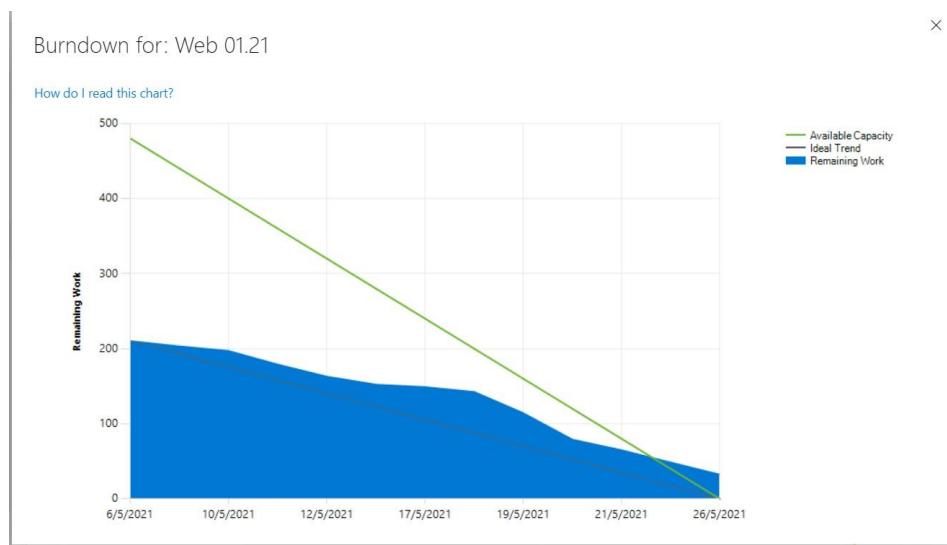


Figura 1. Objetivos y avances de un periodo dentro del sprint.

Materiales

Para el desarrollo de la API se utilizó el lenguaje de programación C#, la plataforma de desarrollo .NET Core y el *framework* DevOps.

C#

C# (leído en inglés "C Sharp") es un lenguaje de propósito general diseñado por Microsoft para su plataforma .NET.

Aunque es posible escribir código para la plataforma .NET en muchos otros lenguajes, C# es el único que ha sido diseñado específicamente para ser utilizado en ella, por lo que programarla usando C# es mucho más sencillo e intuitivo que hacerlo con cualquiera de los otros lenguajes ya que C# carece de elementos heredados innecesarios en .NET. Por esta razón, se suele decir que C# es el lenguaje nativo de .NET [14]

.Net Core

.NET podría considerarse una respuesta de Microsoft al creciente mercado de los negocios en entornos Web, haciendo competencia a las diversas plataformas ya existentes como Java de Oracle o PHP. Su propuesta es ofrecer una

manera rápida y económica, a la vez que segura y robusta, de desarrollar aplicaciones o como la misma plataforma las denomina, soluciones— permitiendo una integración más rápida y ágil entre empresas y un acceso más simple y universal a todo tipo de información desde cualquier tipo de dispositivo [15].

Dev-Ops

Es un conjunto de prácticas que agrupan el desarrollo de software (Dev) y las operaciones de TI (Ops). Su objetivo es hacer más rápido el ciclo de vida del desarrollo de software y proporcionar una entrega continua de alta calidad.

Como DevOps pretende ser un modo de trabajo interfuncional, en lugar de una sola herramienta de DevOps existen conjuntos de múltiples herramientas. Se espera que tales herramientas de DevOps encajen en una o más de estas categorías, que reflejen los aspectos clave del proceso de desarrollo y entrega [16]:

1. Código: desarrollo y revisión de código, herramientas de administración de código fuente,

- 2. Construcción: herramientas de integración continua, estado de compilación
- 3. Prueba: herramientas de prueba continuas que brindan retroalimentación sobre los riesgos comerciales
- 4. Paquete: repositorio de artefactos, distribución previa a la implementación de la aplicación
- 5. Lanzamiento - gestión de cambios, aprobaciones de versiones, automatización de versiones
- 6. Configurar - configuración y gestión de la infraestructura, Infraestructura como código

- 7. Monitor: monitoreo del rendimiento de las aplicaciones, experiencia del usuario final

Algunas categorías son más esenciales en una cadena de herramientas DevOps que otras; especialmente la integración continua y la infraestructura como código.

Desarrollo

Al desarrollar la aplicación API los sistemas internos de la empresa se mantienen sin modificaciones para la obtención de datos por sistemas externos, los periodos de tiempo especificados para dichos sistemas se han enfocado en la corrección de errores y no en modificaciones.

8/04/2021 - 21/04/2021 Corrección de Errores P4D aplicación de Escritorio

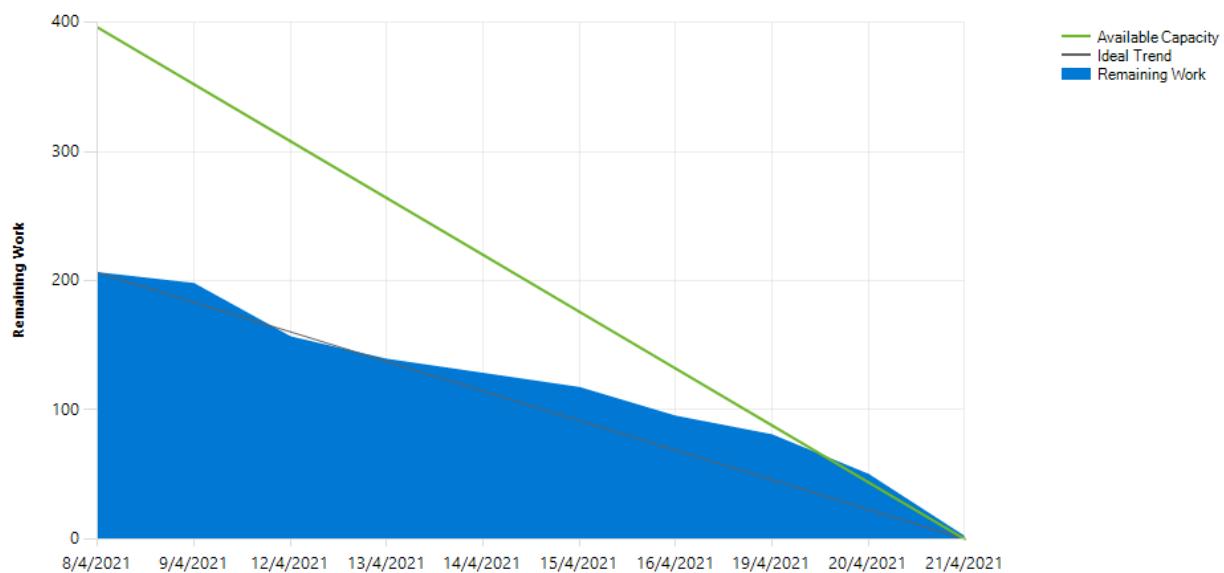


Figura 2. Actividades de corrección de errores.

22/04/2021 – 05/05/2021 Corrección de Errores P4D aplicación de Escritorio

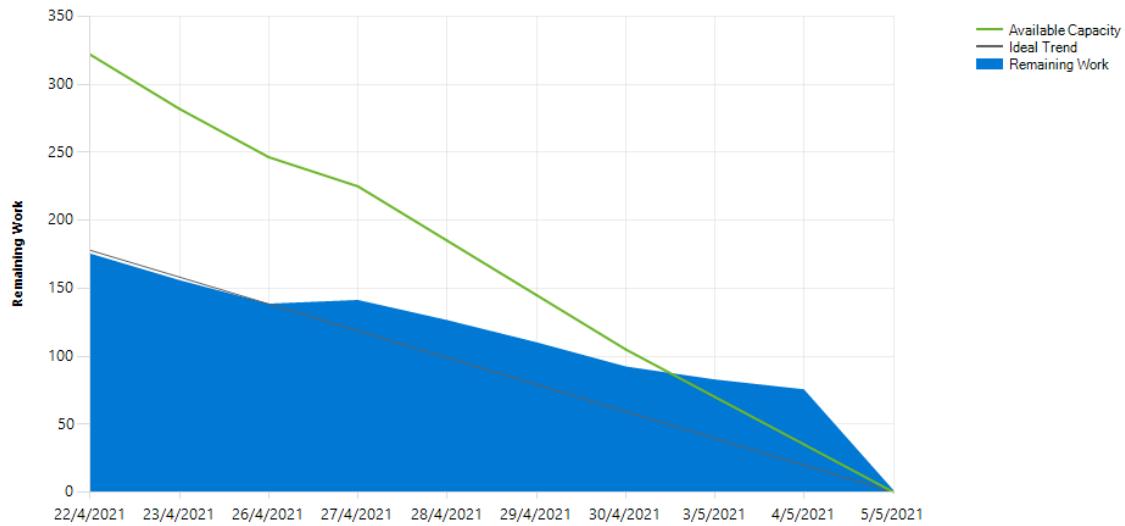


Figura 3. Decremento de actividades de corrección de Errores P4D.

En los dos periodos de muestreo se puede observar en la gráfica como las tareas pendientes sobre las correcciones y mantenimiento se cierran más eficaz y de acuerdo a lo planeado.

Actualmente los reportes de errores sobre los sistemas de escritorio son mínimos, dejando así un tiempo más amplio para el proyecto de Rest API.

Resultados y discusión

Primeros Resultados

La aplicación Rest API ya se encuentra en su primera versión, la cual consta con los mismos módulos que las aplicaciones de escritorio, esto genera un mejor entendimiento sobre su funcionamiento y garantiza un uso óptimo

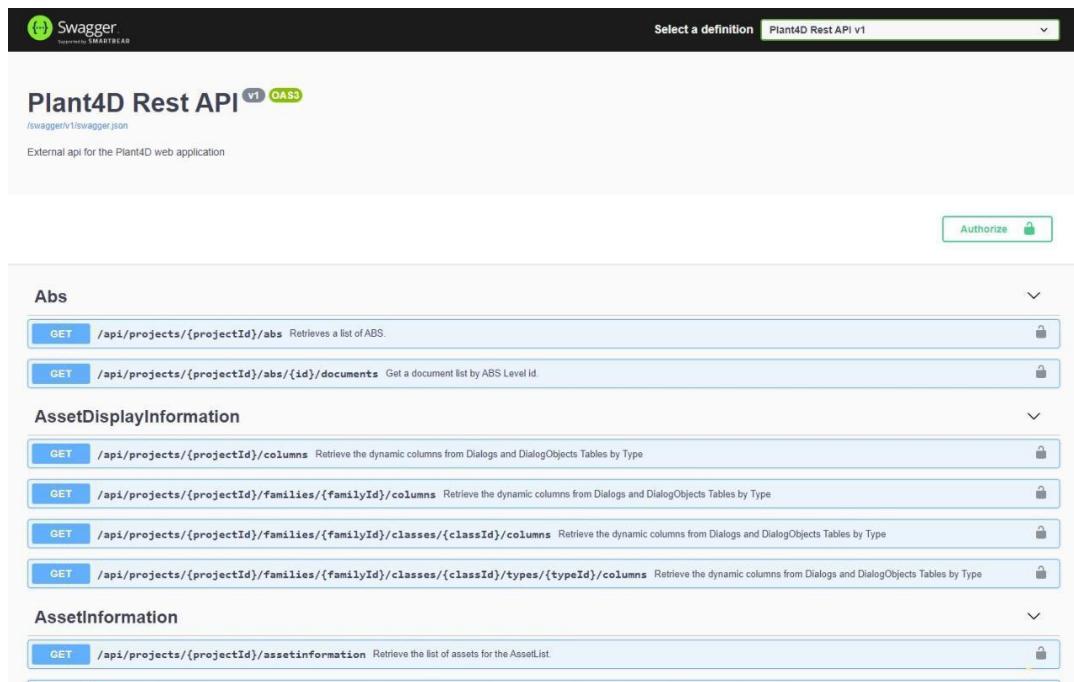


Figura 4. API Rest CEA System.

Seguridad

En la parte de seguridad se integró a la API Rest la autenticación por Token, cada petición a la API lleva en la cabecera una cadena de caracteres la cual al ser procesado y descifrado contiene la información del usuario, nivel de acceso, correo electrónico y la caducidad del token. Cada llamada a la API se valida que el usuario tenga acceso al proyecto solicitado, con los

niveles necesarios para generar la respuesta.

Contratiempos

Durante el desarrollo de los módulos, se presentaron retrasos, los problemas principales fueron la generación del módulo de Documentos y los tiempos disponibles para el desarrollo de la API debido a las actividades pendientes para las aplicaciones de escritorio.

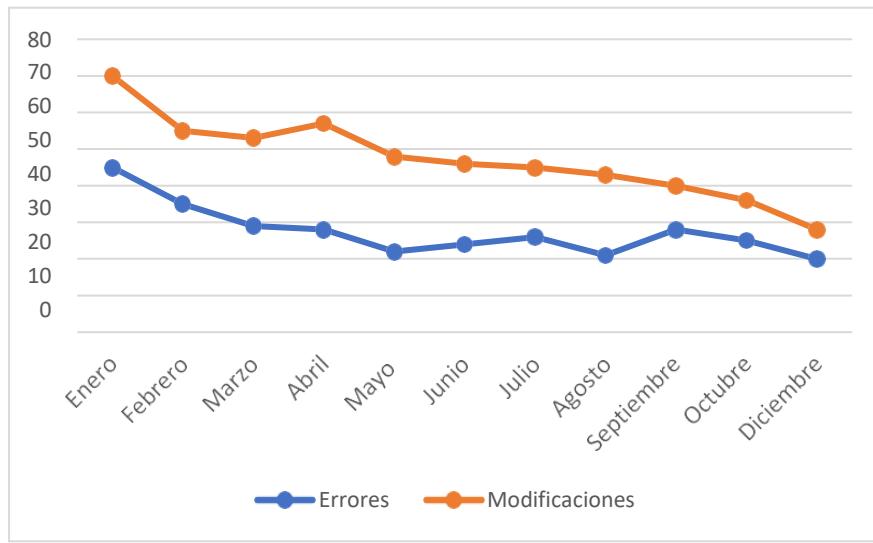


Figura 5. Decremento en las actividades asignadas en las aplicaciones de escritorio.

Como se puede observar en la gráfica las actividades para las aplicaciones de escritorio fueron disminuyendo conforme los módulos de la API Rest fueron completados, de esta forma las actividades se fueron concentrando en el proyecto API.

Tiempo de Integración

Con la API Rest ya en funcionamiento las integraciones con sistemas informáticos de empresas externas, solo dependerá de los tiempos que ellos asignen en modificar o actualizar las aplicaciones que consumirán la API y sus diferentes funciones. Haciendo esto se mantienen las aplicaciones de escritorio sin modificaciones y solo la estabilidad de estas.

Conclusiones

Como se evidenciado en el desarrollo de esta API, dada la necesidad de la empresa CEA Systems para tener una forma más efectiva de permitir que la información de sus aplicaciones sean accesibles desde otra plataforma, se justificó el desarrollo del proyecto de creación de una API Rest para la actualización, generación y

eliminación de datos.

Por otra parte, es necesario mencionar que se cumplieron los siguientes objetivos específicos:

- Se definió el lenguaje de programación C# .Net Core como el lenguaje base para el desarrollo del proyecto, esto de acuerdo con la experiencia del equipo de programadores, facilitó el desarrollo del proyecto al ser un lenguaje conocido por los mismos.
- Se analizó que la base de datos no necesitaba un cambio de estructura, manteniendo así la integridad y el funcionamiento de las aplicaciones de escritorio.
- Se determinaron los modelos de datos para la API Rest estos se realizaron tomando como base los modelos en las aplicaciones de escritorio.
- Se estableció el tipo de seguridad a utilizar en la API Rest como Json Web Token, este token contendría de manera encriptada la información del usuario, nivel de acceso y la fecha de creación de este.

- Se específico como estructura del proyecto MVC (Modelo Vista Controlador), es una estructura con la

que el equipo está acostumbrado, obteniendo de esta forma una mejor comprensión de la estructura.

Referencias

[1] «RedHat,» RedHat, 2019. [En línea]. Available: <https://www.redhat.com/es/topics/integration/what-is-the-difference-between-soap-and-rest>. [Último acceso: 2020].

[2] A. J. Nalini Subramanian(Research Scholar), «Recent security challenges in cloud computing,» *Computers & Electrical Engineering*, vol. 71, nº 71, pp. 28-42, 2018.

[3] K. Curran, *Pervasive and Ubiquitous Technology Innovations for Ambient Intelligence Environments*, Hershey, PA: IGI Global, 2013.

[4] C. P. y. P. Jamshidi, *Microservices: A Systematic Mapping Study*, Roma, Italia: Springer, 2016.

[5] L. D. Lauretis, «From Monolithic Architecture to Microservices Architecture,» *IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)*, p. 93–96, 2019.

[6] E. M. Daniel López, «Arquitectura de Software basada en Microservicios para Desarrollo de Aplicaciones Web,» *Séptima Conferencia de Directores de Tecnología de Información, TICAL 2017 Gestión de*, 2018.

[7] E. M. S. González, «Aplicando seguridad a una API REST con JSON Web Tokens,» p. 80, 2020.

[8] C. M. Mohammed y S. R. M. Zeebaree, «Sufficient Comparison Among Cloud Computing Services: IaaS, PaaS, and SaaS: A Review,» *International Journal of Science and Business*, vol. 5, nº 2, pp. 17-30, 2021.

[9] A. B. y. S. Nogara, «SaaS vs. IaaS vs. On-premise – Survey on Adoption Factors,» 2021.

[10] J. P. Pallavi Gb, «Secure and efficient multi-tenant database management system for cloud computing environment,» *Researchgate*, 2020.

[11] F. J. A.-M. y. M. A.-C. P. R. Palos-Sánchez, «Cloud Computing (SaaS) Adoption as a Strategic Technology: Results of an Empirical Study,» *Mobile Information Systems*, vol. 2017, nº 1, p. 1–20, 2017.

[12] K. Schwaber, «scrum.org,» Scrum.org, 2021. [En línea]. Available: <https://www.scrum.org/resources/what-is-sprint-planning>. [Último acceso: Octubre 2021].

[13] Microsoft, «Azure DevOps,» Microsoft, [En línea]. Available: <https://azure.microsoft.com/es-es/services/devops/>. [Último acceso: Agosto 2020].

[14] F. B. Galiano, «<https://elvex.ugr.es/>,» [En línea]. Available: <http://elvex.ugr.es/decsai/csharp/language/intro.xml>. [Último acceso: 2021].

[15] Microsoft, «.NET documentation,» Microsoft, [En línea]. Available:

Integration of an industrial control to a digital twin at the industrial level

Veronica Guerrero-Hernandez
Division Ingenieria Informatica
Tecnologico Nacional de Mexico/
Instituto Tecnologico Superior de
San Andres
San Andres Tuxtla, Veracruz,
Mexico
veronicaguerrero@itssat.edu.mx

Guillermo Reyes-Morales
Division Ingenieria Mecatronica
Tecnologico Nacional de Mexico/
Instituto Tecnologico Superior de
San Andres)
San Andres Tuxtla, Veracruz,
Mexico
guillermoreyes@itssat.edu.mx

Farid-Armando Matacapan-Toto
Mantenimiento Mecanico
Ingenio Presidente Benito Juarez
Cardenas Tabasco, Mexico
faridmatacapan@gmail.com

Pedro Jacome-Onofre
Division Ingenieria Industrial
Tecnologico Nacional de Mexico/
Instituto Tecnologico Superior de
San Andres
San Andres Tuxtla, Veracruz,
Mexico
pedrojacome@itssat.edu.mx

Jorge Alberto Ortega Moody
Ingenieria Mecatronica
Morehead State University
Morehead KY, US
jorgemoody@gmail.com

Maria Angelica Martinez Herrera
Deptamaneto de Sistemas y
Computacion
Tecnologico Nacional de Mexico/
Instituto Tecnologico de Orizaba
Orizaba, Veracruz, Mexico
maria.mh@orizaba.tecnm.mx

Abstract— The objective of this research work was to demonstrate the operation of a dynamic system in a virtual reality or digital twin environment at an industrial level, establishing automation with an industrial PLC control (Programmable Logic Control). The methodology used follows the development objectives of the "Serious Games", which apply physical or cyberphysical engines whose process to be virtualized or case study was under the series of physical specifications of the process of "an automated industrial plant". As a result, a mathematical model of differential equations with discrete-time response was obtained to define the dynamical system. This mathematical model was treated using the Minitab 2019 and Matlab R2015 programs, managing to demonstrate its relevance to be implemented in the cyber-physical system stage. The above was validated with the implementation of the PLC industrial control for real-time training in the digital twin at the industrial level.

Keywords— automation, virtual reality, cyber-physical systems, processes, programmable logic control, digital twins

I. INTRODUCTION (HEADING I)

To date, due to the installation of various apparatus in process industries, both factors of complex structures and severe operating conditions could result in higher accident frequencies and maintenance challenges [1]. They must take into account that all the equipment that performs the system process must be assembled as a whole. Thanks to the successful implementation of digital manufacturing, design experimentation and validation becomes feasible [2]. As well as the configuration of products, procedures and systems [3], without assembling any physical component.

Of the above, a large representation of design drawbacks and flaws remain undetectable, even until any system is tested, leading to project delivery delays, rework, and financial loss. According to Zäh and Wünsch (2005), commissioning time consumes up to 15-20% of the engineering design and

construction time of an automated system, and more than 70% of this time is wasted correcting software errors. These software are of the control that allows the interaction between the PLC and each of the components of the system [4].

Precisely, to avoid losses in the commissioning of an industrial plant, use is made of emerging technologies that can recreate plants that are just going to be put into operation with virtual reality and connect it to the PLC in real time, thus avoiding losses. With this, the loss of time in programming in the control software that allows the interaction between the PLC and each one of the components of the system is avoided.

At present, the meaning of virtual commissioning has incorporated a sovereignly viable solution to solve the aforementioned drawbacks. The main idea is to assemble a virtual factory to a real controller, so that engineers from different areas (processes and automation, for example) can operate together as a common prototype.

In this way, the PLC program can be tested without having to wait for the physical implementation of the process, and the general functionality of the plant can be validated [5][6][7]. Studies have shown the positive effects of working with virtual commissioning, reducing costs and adding value [8][9]. Likewise, its scalability has been studied, which allows the estimation of the economic and ecological benefits of the using a digital twin for a company. [10] [11].

A wide variety of works are available in the literature that represent important efforts as emerging technology, its widespread implementation is increasing in several domains, such as industrial, automotive, medicine, smart cities, among others [12][13][14][15]. Each of these works show great advances in this subject, and all of them reveal certain aspects or inconveniences in which it is inevitable to work. These include: 1.- The generation of virtual reality scenarios given the need for

highly specialized software and 2.- The adequate modeling of the physical behavior of the components involved in the 3D scene, specifically highly specialized devices such as robots, thermal, hydraulic, pneumatic, chemical systems, among others [16].

Similarly, the possibility of carrying out a communication protocol between the virtual scenario and any PLC is admitted, but in order to control any dynamic system, the embedded system must be modeled on stage [17].

II. STATE OF THE ART REVIEW

A review was carried out based on academic publications, verified by peers in Scopus databases, using virtual laboratories

applied to simulation in areas related to the research topic. In this way, the importance of immersive simulation in training applications is increasing in different areas, such as industrial, medical, aerospace, among others, as evidenced by the exponential increase in publications on the subject.

For this reason, simulation is an increasingly significant part with the potential to support aspects such as saving time and costs, mainly, such as those shown in table I. Where it is observed that in these laboratory applications they do not carry out the application of external industrial controls such as the PLC.

TABLE I. SYSTEMATIC REVIEW OF VIRTUAL LABORATORIES

Authors	Article title	Technique	Control system	Platform	Advantage	Disadvantages
A. Burghardt, D. Szybicki, P. Gierlak, K. Kurc, P. Pietrus, and R. Cygan	Programming of Industrial Robots Using Virtual Reality and Digital Twins	Framework of simulation system[17]	Virtual	RobotStudio	Control the modeling of a robot	Totally virtual
Sánchez-Alonso, R. E., Ortega-Moody, J., González-Barbosa, J. J., and Reyes-Morales, G.	Use of Platforms for the Development of Virtual Applications in the Modeling of Robot Manipulators	3D modeling in robotics [18]	Virtual	Unity 3D	Parallel robot modeling	Totally virtual
Al Mashhadany, Y., Gaeid, K. S., and Awsaj, M. K.	Intelligent Controller for 7-DOF Manipulator Based upon Virtual Reality Model	ANFIS [19]	Virtual	Matlab	7-DOF manipulator modeling	Totally virtual
Flavia Pires, Ana Cachada, José Barbosa, Antonio Paulo Moreira, Paulo Leitao	Digital Twin in Industry 4.0: Technologies, Applications and Challenges	3D modeling in robotics[20]	Virtual	V-REP	Monitoring, optimisation and decision support system	Totally virtual
Uk, M. E., Sajjad, F. B., Soyaslan, M., and Erdogan, O.	Modeling, control, and simulation of a SCARA PRR-type robot manipulator	PRR (Prismatic-Revolute-Revolute) [21]	Virtual	Matlab VRML	SCARA Manipulator Modeling, Major Torque	Difficulty in control
Vladimir KUTS, Tauno OTTO, Toivo TÄHEMAA, Yevhen BONDARENKO	digital twin based synchronised control and simulation of the industrial robotic cell using virtual reality	synchro-nisation model [22]	Virtual	Unity 3D	synchro-nisation model of real and virtual industrial robots and experimental	Totally virtual
T. R. Wanasinghe <i>et al</i>	Digital Twin for the Oil and Gas Industry: Overview, Research Trends, Opportunities, and Challenges	Framework of simulation system [23]	Virtual	Plataforma 3D	Digital Twin Framework proposal with five components	Totally virtual
Marayong, P., Shankar, P. Wei, J. Nguyen, H., Strybel, T. Z., and Battiste, V.	Urban Air Mobility System Testbed using CAVE Virtual Reality Environment	CAVE VR [24]	Joystick	Unity 3D	flight control	Manual control
Bruguera, M. B., Ilk, V., Ruber, S. and Ewald, R.	Use of Virtual Reality for astronaut training in future space missions - Spacecraft piloting for the Lunar Orbital Platform - Gateway (LOP-G)	Lunar Orbiting Platform - Gateway (LOP-G) [25]	Virtual	Unity 3D y Orbiter Space Flight Simulator	Space flight simulator	Totally virtual

Perez-Ramirez, M., Arroyo-Figueroa, G., and Ayala, A	The use of a virtual reality training system to improve technical skill in the maintenance of live-line power distribution networks	Maintenance of power distribution networks with live lines [26].	Virtual	Plataforma 3D	Airline operator training	Totally virtual
S. Piechowski <i>et al</i>	Virtual reality as training aid for manual spacecraft docking	Artificial gravity bed with ESA (AGBRESA) [27]	Manual control	Unity 3D y el software 6df	Spacecraft Control Training	You don't have all the controls.
Feng, Z., González, V.A., Trotter, M., Spearpoint, M., Thomas, J., Ellis, D., and Lovreglio, R	How people make decisions during earthquakes and post-earthquake evacuation: Using Verbal Protocol Analysis in Immersive Virtual Reality	Verbal Protocol Analysis (VPA) [28]	Does not apply	Unity 3D	Decision making of people in earthquakes and evacuation	Difficulty in obtaining data in real earthquakes
Gace, I., Jaksic, L., Murati, I., Topolovac, I., Zilak, M., and Car, Z.	Virtual Reality Serious Game Prototype for Presenting Military Units	LoPoly Rigged Hand y RigidHand.[29]	Manual	Unity 3D	Military unit selection	Totally virtual
Zheng, J., Shi, P., and Yu, H.	A Virtual Reality Rehabilitation Training System Based on Upper Limb Exoskeleton Robot	VR Rehabilitation Training Game Design [30]	Virtual	Unity 3D	Cerebrovascular rehabilitation	Totally virtual
Liou, W. K., and Chang, C. Y.	Virtual Reality Classroom Applied to Science Education	HTC Vive VR [31]	Manual	3d modeling	Teaching chemical elements and human anatomy experiment	Totally virtual

According to what is presented in Table I, it can be clearly seen that specific publications on virtual reality are marked as such, in contrast to purely theoretical contributions.

III. METHODOLOGY

In this research work, a methodology is proposed to determine if the design of the dynamic systems model, applied in virtual reality scenarios, is functional with the implementation of an industrial control such as the PLC.

The methodology follows the development objectives in the use of "Serious Games", a guideline that uses videogame or cyberphysical engines and other highly available tools (software and hardware) in the development of virtual scenarios of the industrial plant. Which is contemplated to make entertainment more realistic.

The methodology used in the design is shown in Figure 1. It begins with a series of specifications of the process to be virtualized. Once the physical characteristics are obtained from the process, as well as the dynamic behavior of each element (mechanisms, sensors and actuators) and its computer-aided design (CAD). The third step is to insert the CAD designs into the physics engine to assign the dynamic behavior of each element. Finally, the performance of each element is tested to ensure that its behavior is as close to reality as possible.

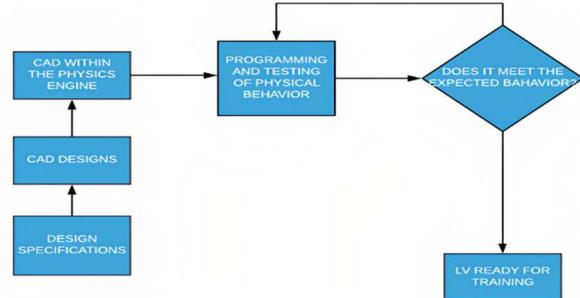


Fig. 1. Methodology

The modeling stage begins with the characterization of all the objects that will be visually available in the scene: buttons, tanks, filling valves, conveyor belts, bottles, among others. Subsequently, we proceed to design with the manufacturer's specifications, or starting at zero each of these components. As for all the manipulated CAD models, these were developed to real scale in SolidWorks®, and later they were sent in STL (Stereo Lithography) format to 3DS MAX, where they worked on the reduction of the polygons that constitute the meshing of each 3D model. The intention of this treatment is to reduce the computational cost of the graphical development of the virtual application. Next, each 3D model is sent in FBX (FilmBox) format to Unity 3D.

A. Mathematical modeling of the dynamical system of technological systems

The technological equipment fixed in the virtual scenario are categorized into two classes: those whose stage in time

manages to take any value within a category and those whose status is incorporated into a Boolean variable.

The tank where the product to be packaged is mixed and heated must be modeled as a dynamic system, which in turn responds to a thermal system, in which the temperature value is variable over time until it reaches an expected temperature. Thus, the time and the way in which the expected temperature is achieved depend on the physical characteristics of the elements involved in the system.

In a general way, it was possible to express that this system is characterized by the heat transfer function of a fluid inside a tank, with the analysis of the energy balance equation. Said analysis considers that the accumulation of the heat energy of the fluid inside the tank is equal to the heat energy of the fluid, based on the temperature differences plus the heat energy that is applied to the fluid inside the tank, minus the heat energy by conduction. For the last energy change, it is considered as 0 because the tank is well insulated, and it is represented in the following equation (1):

$$VC_p\rho \frac{dT}{dt} = \rho C_p F [T_e - T(t)] + Q - \frac{T(t) - T_a(t)}{R} \quad (1)$$

Where $T(t)$ is the temperature of the fluid at the tank outlet, F the mass flow, V the volume, T_e the initial or inlet temperature, Q the heat flow applied to the fluid inside the tank, $T_a(t)$ the ambient temperature outside the tank, R the resistance of the thermal material adhered to the tank, ρ the density of the fluid and C_p the specific heat. All numbers and brackets in the text and formulas are to be vertical.

According to classical control theory, the transfer equation of the plant (tank) is obtained with the variable that represents the output divided by the variable that represents the input, so that the dynamic transfer equation in the Laplace domain (2) would be as follows:

$$\frac{\Delta T}{Q} = \frac{T(t) - T_a(t)}{K_1 + K_2} \quad (2)$$

The equation corresponding to the Laplace domain, which represents the plant (tank) for the system to recognize in real time the operation of the automated bottle filling plant in virtual reality, must be obtained in discrete time to be able to implement it in Unity 3D. On the other hand, regarding the investigations of the state of the art of the application in digital systems, the most recommended method is the bilinear transform (Tustin method).

Applying the bilinear transform process of equation 3, where $Z-1$ is expressed in the complex domain of S , and considering a period less than 0.05, equation (3) is shown below:

$$T(Z) = \frac{Q(Z)}{K_1 \frac{Z^2 - Z^{-1}}{T_1 + Z^{-1}} + K_2} \quad (3)$$

On the other hand, K_1 and K_2 are considered as constants of the fluid and Q as the thermal constant, obtaining equation 4 in the digital domain:

$$T(K) = \frac{TQ(K-1)}{T_p C_p F + 2V C_p \rho} + \frac{TQ(K)}{T_p C_p F + 2V C_p \rho} - \frac{[T_p C_p F - 2V C_p \rho]T(K-1)}{T_p C_p F + 2V C_p \rho} \quad (4)$$

According to the above developed, equation 4 is the one that is applied in Unity 3D to establish the dynamic system in digital form of the plant (tank).

A fully proportioned mixing and heating tank is shown in Figure 2. You can see one of the access pipes, the outlet pipe, the mixer blades and the heater inside the tank, as well as the mixer motor and the temperature and level sensors. The bottling factory not only has the mixing and heating process, but incorporates a series of devices (sensors and actuators) that allow the entire process to be automated.

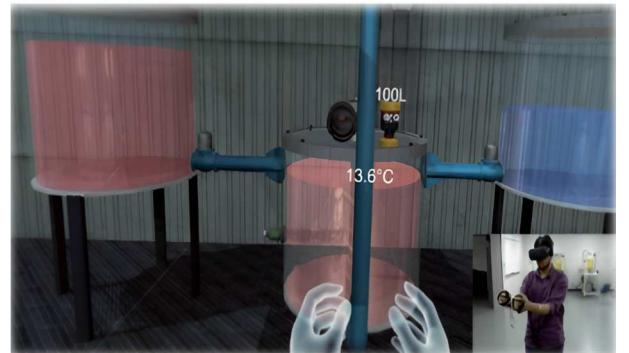


Fig. 2. Interaction on the virtual stage by means of 'oculus rift'

The architecture of the system is shown in figure 3, as the main base it makes use of the cyber-physical engines that are represented in the starting blocks of the figure, in which the virtual reality scenarios of the automated plant are implemented, which are communicate with external controllers via serial communication protocol.

The simulation of the physical behavior of the objects is carried out by means of a physics engine independent of the communication card of the industrial control with the PLC, which helps the system not to become slow due to excessive calculations and frees it to concentrate in the communication with the client, as shown in the following figures.

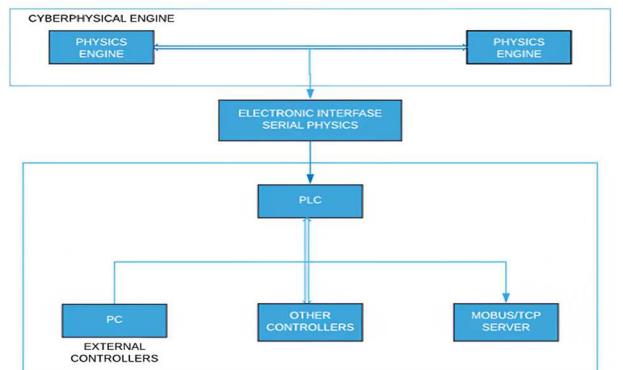


Fig. 3. System architecture

As shown below, in the following figures 4 and 5, the Validation of the implementation of industrial control is presented in a virtual reality scenario in real time.



Fig. 4. Allen Bradley Compact Logic 5370 L1 PLC.



Fig. 5. Operator touching full bottle.

B. Industrial digital twin implementing an industrial control

The great totality of the available platforms for the improvement of virtual applications supports the introduction of programming code in high-level languages.

This facilitates the programming of the dynamic modeling of the physical systems embedded in the scene and makes it possible to endorse a simulation coupled to reality as shown in figure 6.

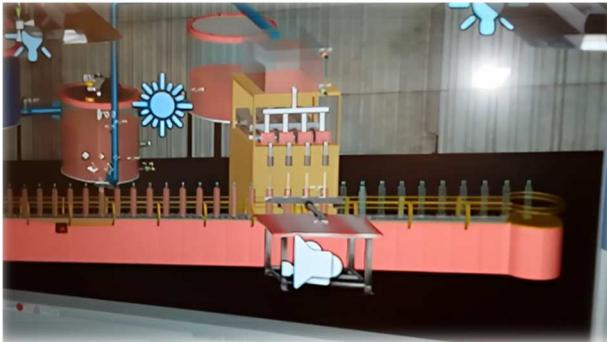


Fig. 6. Virtual bottling plant perspective.

Similarly, the possibility of carrying out a communication protocol incorporating the virtual scenario and any brand or model of PLC is allowed, showing the programming in figure 7.

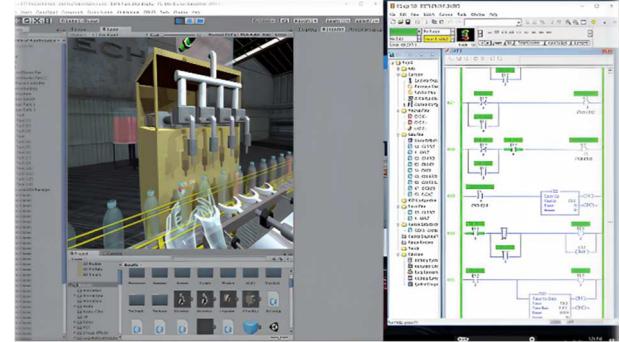


Fig. 7. Perspective of control program execution in RS Logix 500.

For its part, the virtual bottle-filling process is controlled by the Allen Bradley family's compaclogic 5370 L1 PLC, which is programmed with RSLogix 500 software (figure 4). In conclusion, the PLC handles: 12 digital outputs to control the solenoid valves, the conveyor belt motor, the filling injectors and other actuators; 14 digital inputs to monitor the level of the deposits, the count of the bottles, among others; 1 built-in analog output to time control signal going to heater; and 1 analog input associated with the time record ordered by the temperature sensor.

IV. RESULTS AND DISCUSSION

This section details the results that were achieved in the research of the project; the experiments are related to the discrete-time response of the bottling plant operation. First, the experimental determination of the transfer function was carried out with the Minitab 2019 software. Afterwards, the mathematical analysis was carried out to demonstrate the feasibility of the project in real time with the MATLAB R2015 software, a stage in which obtaining the response in continuous time with the transfer function obtained from the plant was a priority. Subsequently, the discrete-time response was achieved by applying the Tustín function. Finally, the PLC is physically connected to the cyber-physical system to control the virtual reality scenario, to check the operation of the automated industrial plant.

A. Experimental Determination of the Transfer Function with Minitab 2019

To be able to know that the main signal processing is the plant, it is necessary to reconstruct the signal in its digital form, the one used in the cyber-physical system. The first step in the conversion of the analog signal to digital is sampling, since it is about recovering a signal, according to the theory on signal sampling developed by H. Nyquist, known as the sampling theorem. It establishes the conditions to sample a signal without loss of information.

For this reason, in order to carry out an analysis of the signal under the consideration of the transfer equation obtained for the plant, whose directed or intentional sampling is carried out by applying a sampling frequency of 50 Hz, and since these samples have to be equispaced. Periodic sampling every $T_s = 0.02$ sec has to be considered. In this way, 3,999

values that make up the sample are obtained for the case study, on which the analysis is developed in the Minitab version 2019 software to demonstrate the behavior of the dynamic system, obtaining the following results in figure 8.

Firstly, to carry out a good statistical analysis of the samples obtained under the mathematical model of the dynamic system, the normality test must be carried out to demonstrate the proposed hypothesis. In basic statistics, the test for normality or linear relationships is performed to confirm that the data generated are parametric and follow a normal distribution in the use of continuous measurements. Likewise, the null hypothesis $H_0: p \geq 0.5$ or the research or alternative hypothesis $H_A: p \leq 0.5$ is considered.

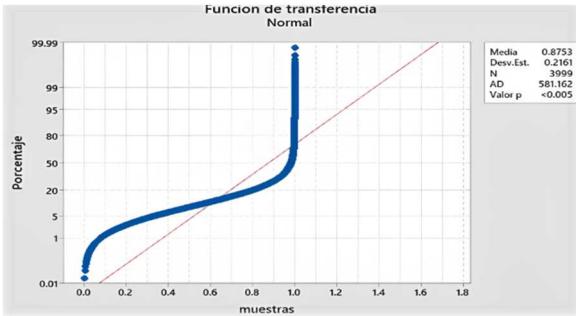


Fig. 8. Test of normality in the transfer function.

As a second step, the samples were analyzed using the Minitab software, with which the statistical tool for quality of non-normal parameters was applied under the analysis of sample process capability. The foregoing determines that the output of the process satisfies the requirements of the implementation of the transfer function, since its data does not follow a normal distribution and a more consistent response is obtained, as shown in figure 9.

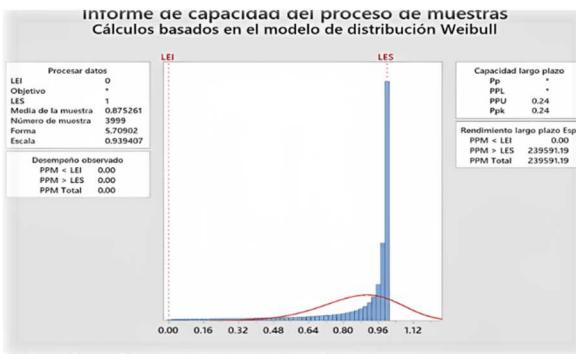


Fig. 9. Test of non-normality in the transfer function.

As can be seen in figure 10, the sampling of the data to obtain the quantification of the proposed system is from 0.90 to 1.00; the variation remains constant or is normalized, resulting in the behavior of the mathematical model with a stable response. While from 0.00 to 0.90, the system response behaves with a non-linear variation, and the dynamic part of the transfer function is represented. Thus, in Figure 20 the normal distribution is shown starting at 0.87.

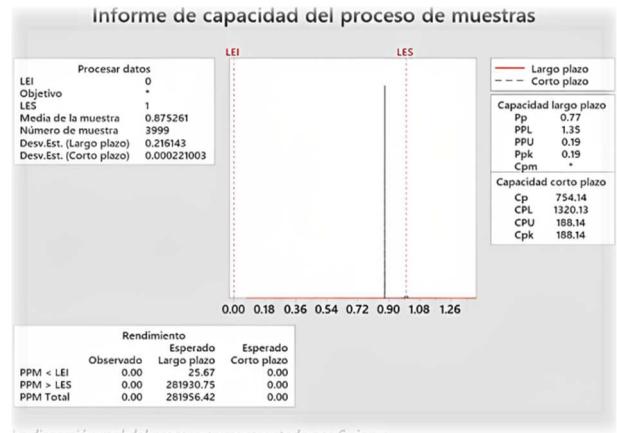


Fig. 10. Normal distribution from 0.85 of the transfer function.

With what has been explained and shown in the previous figures, it was possible to verify that the most convincing and adequate hypothesis for this work is alternative or research $H_A: p < 0.5$. Resulting in a nonlinear function, which is a mathematical model adjusted to data from the real world or that describes them; and it is that, in itself, the main objective is to briefly describe how to use technology to obtain a good mathematical model. In this way, and in accordance with what is represented in Figure 13, it is stated that the dynamic system obtained in the transfer function of the automated bottling plant is validated by its 95% confidence level.

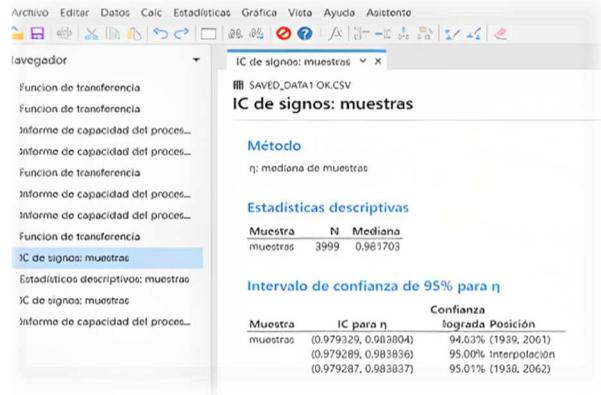


Fig. 11. Confidence level of 95% of the transfer function.

B. Result of the Behavior of the Plant Applying Matlab R2015

To analyze the behavior of the plant or the transfer function of the system, the MATLAB R2015 software was used, the equation was obtained from the energy balance with the system of equations and the Laplace transform, obtaining the response in continuous time. Afterwards, the answer of the same transfer equation was obtained, but discretized with the Tustin function that MATLAB R2015 has. Achieving "characterize the dynamic model in discrete time of the industrial system to be implemented, in order to discretize the dynamic behavior of the physical system embedded in the environment" of the virtual reality laboratory for PLC programming training, which is shown in figure 12.

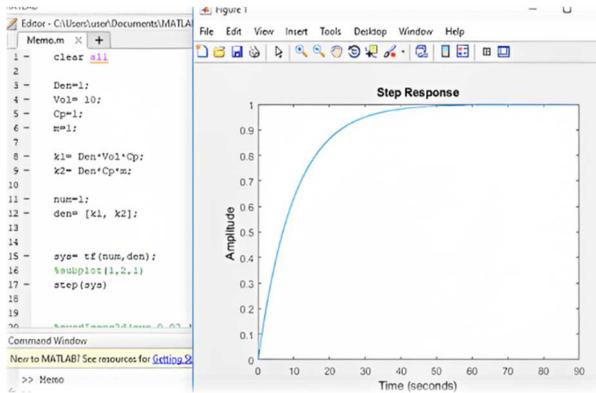


Fig. 12. Representation of the plant in continuous time.

Similarly, considering water as a liquid, taking its density, specific weight, mass flow and specific heat, the simulation was performed again in MATLAB R2015 to see how the system of equations behaved, applying the Tustin method with a sampling of 0.02 and obtaining as a result a similarity with the simulation that was represented in continuous time. Figure 14 shows the times and the amplitude of the function for the plant.

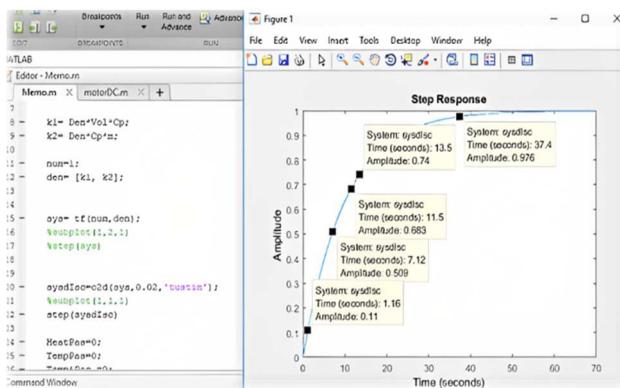


Fig. 13. Graph of the plant using the Tustin function in MATLAB R2015.

C. Discussion

In our work, final considerations are established from the results obtained from data processing and statistical analysis, to demonstrate the feasibility and validation of the dynamic system in the digital twin (bottling plant). Taking the results obtained from the research as a starting point, the theoretical discussions of the literature reviews that were shown in table I of section II are presented. In contrast to the work presented in this document, it explores the implementation of an industrial control (PLC) to control the digital twin (bottling plant) in real time as a contribution to science.

We can mention [32], who worked on his research with a security architecture for industrial automation and control systems based on OpenPLC software that uses the framework in its industrial cybersecurity research. In the same way, [33] tells us that most of the selected publications have focused on the development of the framework and the methodology.

According to [34] review, in his research he found representations of established Digital Twin models in the most relevant literature, as well as derivative purposes of Digital Twins. Similarly [35] based on the definitions of SMSD (smart manufacturing design) and the advantage of DT-SMSD, a framework of Function-Structure-Behavior-Control-Intelligence-Performance (FSBCIP) is proposed to review the main steps of DT -SMSD.

According to [36], among the principles of Smart Manufacturing (SM) or industry 4.0 (I4.0), for the application of digital twins, it is required as a requirement for its development of the Framework, which has been cited as the greatest technological trend that alters engineering, and design today.

Our work presented in this document in counterpart to the ones already mentioned explores the possibilities, challenges and limitations for the implementation of digital twins and their use in different domains. By understanding how digital twin applications may differ in terms of their requirements, by applying free software development. It brings updates to industries as an advantage, preventing their industrial applications with digital twins from becoming obsolete due to high licensing software costs.

ACKNOWLEDGMENT

This work has been realized thanks to the support of the CONACYT, the CIATEQ, TecNM/ITS de San Andres Tuxtla and Morehead State University.

REFERENCES

- [1] R. He, G. Chen, C. Dong, S. Sun, and X. Shen, "Data-driven digital twin technology for optimized control in process systems," *ISA Trans.*, vol. 95, no. May, pp. 221–234, 2019, doi: 10.1016/j.isatra.2019.05.011.
- [2] A. Löcklin, M. Müller, T. Jung, N. Jazdi, D. White, and M. Weyrich, "Digital Twin for Verification and Validation of Industrial Automation Systems - A Survey," *IEEE Int. Conf. Emerg. Technol. Fact. Autom. ETFA*, vol. 2020-Septe, pp. 851–858, 2020, doi: 10.1109/ETFA46521.2020.9212051.
- [3] D. Mourtzis, "Simulation in the design and operation of manufacturing systems: state of the art and new trends," *Int. J. Prod. Res.*, vol. 58, no. 7, pp. 1927–1949, 2020, doi: 10.1080/00207543.2019.1636321.

[4] A. A. A. Rahman and N. R. Mohamad, "Software-in-the-Loop technique: An approach to support reconfiguration of manufacturing system," *ARPN J. Eng. Appl. Sci.*, vol. 11, no. 16, pp. 9789–9795, 2016.

[5] T. Lechner, E. Fischer, M. Metzner, A. Mayr, and J. Franke, "Virtual commissioning – Scientific review and exploratory use cases in advanced production systems," *Procedia CIRP*, vol. 81, pp. 1125–1130, 2019, doi: 10.1016/j.procir.2019.03.278.

[6] E. Eros, M. Dahl, A. Hanna, A. Albo, P. Falkman, and K. Bengtsson, "Integrated virtual commissioning of a ROS2-based collaborative and intelligent automation system," *IEEE Int. Conf. Emerg. Technol. Fact. Autom. ETFA*, vol. 2019-Septe, pp. 407–413, 2019, doi: 10.1109/ETFA.2019.8869444.

[7] A. Hanna, K. Bengtsson, M. Dahl, E. Eros, P. L. Götvall, and M. Ekström, "Industrial Challenges when Planning and Preparing Collaborative and Intelligent Automation Systems for Final Assembly Stations," *IEEE Int. Conf. Emerg. Technol. Fact. Autom. ETFA*, vol. 2019-Septe, pp. 400–406, 2019, doi: 10.1109/ETFA.2019.8869014.

[8] D. R. Delgado Sobrino, R. Ružarovský, R. Holubek, and K. Velišek, "Into the early steps of Virtual Commissioning in Tecnomatix Plant Simulation using S7-PLCSIM Advanced and STEP 7 TIA Portal," *MATEC Web Conf.*, vol. 299, p. 02005, 2019, doi: 10.1051/matecconf/201929902005.

[9] I. A. Fernández, M. A. Egúia, and L. E. Echeverría, "Virtual commissioning of a robotic cell: An educational case study," *IEEE Int. Conf. Emerg. Technol. Fact. Autom. ETFA*, vol. 2019-Septe, pp. 820–825, 2019, doi: 10.1109/ETFA.2019.8869373.

[10] F. Oettl, J. Schoeler, and J. Schilp, "Development of a method for evaluating the benefits of using a digital twin," *Int. Conf. Electr. Comput. Commun. Mechatronics Eng. ICECCMEE 2022*, 2022, doi: 10.1109/ICECCMEE55909.2022.9988218.

[11] M. Ahrens, C. Richter, P. Hehenberger, and G. Reinhart, "Novel approach to establish model-based development and virtual commissioning in practice," *Eng. Comput.*, vol. 35, no. 3, pp. 741–754, 2019, doi: 10.1007/s00366-018-0622-6.

[12] D. M. Botín-Sanabria, S. Mihaita, R. E. Peimbert-García, M. A. Ramírez-Moreno, R. A. Ramírez-Mendoza, and J. de J. Lozoya-Santos, "Digital Twin Technology Challenges and Applications: A Comprehensive Review," *Remote Sens.*, vol. 14, no. 6, pp. 1–25, 2022, doi: 10.3390/rs14061335.

[13] Á. Bárkányi, T. Chován, S. Németh, and J. Abonyi, "Modelling for digital twins—potential role of surrogate models," *Processes*, vol. 9, no. 3, 2021, doi: 10.3390/pr9030476.

[14] E. M. Martinez, P. Ponce, I. Macias, and A. Molina, "Automation pyramid as constructor for a complete digital twin, case study: A didactic manufacturing system," *Sensors*, vol. 21, no. 14, 2021, doi: 10.3390/s21144656.

[15] T. Bikmukhametov and J. Jäschke, "First Principles and Machine Learning Virtual Flow Metering: A Literature Review," *J. Pet. Sci. Eng.*, vol. 184, no. March 2019, p. 106487, 2020, doi: 10.1016/j.petrol.2019.106487.

[16] J.-A. Ortega-Moody et al., "VIRTUAL LABORATORY OF INDUSTRIAL SCENARIOS FOR TRAINING IN THE AREAS OF AUTOMATION AND CONTROL," *DYNA Ing. e Ind.*, vol. 92, no. 3, pp. 285–287, 2017, doi: <http://dx.doi.org/10.6036/8051.1>.

[17] A. Burghardt, D. Szybicki, P. Gierlak, K. Kurc, P. Pietruś, and R. Cygan, "Programming of industrial robots using virtual reality and digital twins," *Appl. Sci.*, vol. 10, no. 2, 2020, doi: 10.3390/app10020486.

[18] R. E. Sánchez-Alonso, J. Ortega-Moody, J. J. González-Barbosa, and G. Reyes-Morales, "Uso de Plataformas para el Desarrollo de Aplicaciones Virtuales en el Modelado de Robot Manipuladores," *RIAI - Rev. Iberoam. Autom. e Inform. Ind.*, vol. 14, no. 3, pp. 279–287, 2017, doi: 10.1016/j.riai.2017.04.001.

[19] Y. Al Mashhadany, K. S. Gaeid, and M. K. Awsaj, "Intelligent controller for 7-DOF manipulator based upon virtual reality model," *Proc. - Int. Conf. Dev. eSystems Eng. DeSE*, vol. October-20, pp. 687–692, 2019, doi: 10.1109/DeSE.2019.00128.

[20] F. Pires, A. Cachada, J. Barbosa, A. P. Moreira, and P. Leitao, "Digital twin in industry 4.0: Technologies, applications and challenges," *IEEE Int. Conf. Ind. Informatics*, vol. 2019-July, pp. 721–726, 2019, doi: 10.1109/INDIN41052.2019.8972134.

[21] M. E. Uk, F. B. Sajjad, M. Soysal, and O. Erdogan, "Modeling, control, and simulation of a SCARA PRR-type robot manipulator," *Sci. Iran.*, vol. 27, no. 1, pp. 330–340, 2020, doi: 10.24200/sci.2018.51214.2065.

[22] V. Kuts, T. Otto, T. Tähemaa, and Y. Bondarenko, "Digital twin based synchronised control and simulation of the industrial robotic cell using virtual reality," *J. Mach. Eng.*, vol. 19, no. 1, pp. 128–144, 2019, doi: 10.5604/01.3001.0013.0464.

[23] T. R. Wanasinghe et al., "Digital Twin for the Oil and Gas Industry: Overview, Research Trends, Opportunities, and Challenges," *IEEE Access*, vol. 8, pp. 104175–104197, 2020, doi: 10.1109/ACCESS.2020.2998723.

[24] P. Marayong, P. Shankar, J. Wei, H. Nguyen, T. Z. Strybel, and V. Battiste, "Urban Air Mobility System Testbed using CAVE Virtual Reality Environment," *IEEE Aerosp. Conf. Proc.*, pp. 1–7, 2020, doi: 10.1109/AERO47225.2020.9172534.

[25] M. B. Bruguera, V. Ilk, S. Ruber, and R. Ewald, "Use of Virtual Reality for astronaut training in future space missions - Spacecraft piloting for the Lunar Orbital Platform - Gateway (LOP-G)," *Proc. Int. Astronaut. Congr. IAC*, vol. 2019-Octob, no. October, pp. 1–10, 2019.

[26] M. Perez-Ramirez, G. Arroyo-Figueroa, and A. Ayala, "The use of a virtual reality training system to improve technical skill in the maintenance of live-line power distribution networks," *Interact. Learn. Environ.*, vol. 0, no. 0, pp. 1–18, 2019, doi: 10.1080/10494820.2019.1587636.

[27] S. Piechowski et al., "Virtual reality as training aid for manual spacecraft docking," *Acta Astronaut.*, vol. 177, no. April, pp. 731–736, 2020, doi: 10.1016/j.actaastro.2020.08.017.

[28] Z. Feng et al., "How people make decisions during earthquakes and post-earthquake evacuation: Using Verbal Protocol Analysis in Immersive Virtual Reality," *Saf. Sci.*, vol. 129, no. October 2019, p. 104837, 2020, doi: 10.1016/j.ssci.2020.104837.

[29] I. Gace, L. Jaksic, I. Murati, I. Topolovac, M. Zilak, and Z. Car, "Virtual reality serious game prototype for presenting military units," *ConTEL 2019 - 15th Int. Conf. Telecommun. Proc.*, pp. 1–6, 2019, doi: 10.1109/ConTEL.2019.8848505.

[30] J. Zheng, P. Shi, and H. Yu, "A Virtual Reality Rehabilitation Training System Based on Upper Limb Exoskeleton Robot," *Proc. - 2018 10th Int. Conf. Intell. Human-Machine Syst. Cybern. IHMSC 2018*, vol. 1, pp. 220–223, 2018, doi: 10.1109/IHMSC.2018.00058.

[31] W. K. Liou and C. Y. Chang, "Virtual reality classroom applied to science education," *2018 23rd Int. Sci. Conf. Inf. Technol. IT 2018*, pp. 1–4, 2018, doi: 10.1109/SPIT.2018.8350861.

[32] C. Gehrmann and M. Gunnarsson, "A digital twin based industrial automation and control system security architecture," *IEEE Trans. Ind. Informatics*, vol. 16, no. 1, pp. 669–680, 2020, doi: 10.1109/TII.2019.2938885.

[33] T. Y. Melesse, V. Di Pasquale, and S. Riemma, "Digital Twin models in industrial operations: State-of-the-art and future research directions," *IET Collab. Intell. Manuf.*, vol. 3, no. 1, pp. 37–47, 2021, doi: 10.1049/cim2.12010.

[34] M. Sjarov et al., "The Digital Twin Concept in Industry - A Review and Systematization," *IEEE Int. Conf. Emerg. Technol. Fact. Autom. ETFA*, vol. 2020-September, pp. 1789–1796, 2020, doi: 10.1109/ETFA46521.2020.9212089.

[35] J. Leng, D. Wang, W. Shen, X. Li, Q. Liu, and X. Chen, "Digital twins-based smart manufacturing system design in Industry 4.0: A review," *J. Manuf. Syst.*, vol. 60, no. July, pp. 119–137, 2021, doi: 10.1016/j.jmsy.2021.05.011.

[36] J. Moyne et al., "A Requirements Driven Digital Twin Framework: Specification and Opportunities," *IEEE Access*, vol. 8, pp. 107781–107801, 2020, doi: 10.1109/ACCESS.2020.3000437.

Planificación para la Implementación de un Web Framework con Visualización de Modelos Tipo CAD

Darío Cubillos Martínez (ITSSAT, 201u0536@alumno.itssat.edu.mx)

Juan Carlos Mimendi Diaz (ITSSAT, 201u0533@alumno.itssat.edu.mx)

Resumen—Actualmente las tecnologías web y representación gráfica por computadora está muy avanzada en comparación a los primeros años de la informática por esa razón muchas empresas utilizan distintos tipos de tecnologías de visualización de modelos de tipo CAD, para realizar esto muchas se auxilian de motores de renderizado 3D comerciales que les permiten integrar esta funcionalidad a sus sistemas existentes, esto ha sido muy común durante unas cuantas décadas atrás, pero con la llegada de la web y sus enormes ventajas y aun grandes desafíos en esta área por lo que esta publicación plantea la planificación e implementación de estas nuevas tecnologías web para la visualización de modelos y gráficos del tipo CAD.

Abstract-- Currently, web technologies and computer graphics are very advanced compared to the early years of computing, for that reason, many companies use different types of visualization technologies CAD models to do this many are aided by commercial 3D rendering engines that allow them to integrate this functionality to their existing systems, this has been very common a few decades ago, but with the advent of the web and its enormous advantages and even great challenges in this area so this publication raises the investigation of an implementation of these new web technologies for the visualization of CAD type models and graphics.

I. INTRODUCCIÓN.

La publicación hace una investigación de la planificación e implementación de un web framework para la visualización de modelos de tipo CAD la planificación como y la tecnología están destinadas a la industria, debido que las representaciones CAD (*computer-aided design*) o diseño asistido por computadora generalmente tienen por objetivo un contexto profesional.

Este estudio y la información resultante de la investigación proporciona una lectura de valor para hacer una implementación exitosa en un contexto profesional y principalmente en las áreas donde la graficación es el eje central de la tecnología web.

II. ANTECEDENTES TECNOLÓGICOS.

En el año 1961 a William Fetter uso el término “gráficos

porcomputadora” en 1961 para describir su trabajo en Boeing, fue una de las primeras interpretaciones, pero la historia de los gráficos por computadora empieza en 1950 cuando un matemático de Iowa llamado Ben Laposky creó las primeras imágenes generadas por computadora estas imágenes fueron figuras de Lissajous que mostró en un osciloscopio, este acontecimiento fue muy temprano si consideramos que la historia de la electrónica digital moderna comenzó en 1940. [1]

En 1959, con trabajo conjunto General Motors e IBM produjeron la primera pieza de software de dibujo, el DAC-1 (Diseño aumentado por computadoras). Los usuarios pudieron ver el automóvil en perspectiva y rotarlo. Los inicios de los gráficos por computadora para el público fue en la década de 1960 pero aún estaban fuera del alcance de la mayoría de los usuarios porque el hardware gráfico especial era caro.

Los inicios de los gráficos por computadora para el público fue en la década de 1960 pero aún estaban fuera del alcance de la mayoría de los usuarios porque el hardware gráfico especial estaba fuera del alcance de las personas y solamente empresas de gran poder adquisitivo o gobiernos podían tener acceso a este. [1]



Fig. 1. El primer sistema de dibujo por computadora, la DAC-1 (Design Augmented by Computers) Fue creado por General Motors e IBM.

En aquellos años los usuarios tenían que pagar por el tiempo del mainframe por segundo o comprar caros miniordenadores. El resultado fue que pocos profesionales de la computación pudieron desarrollar técnicas de gráficos por computadora.

Gracias a estos avances en el terreno de los gráficos por computadora comenzaron a desarrollarse diferentes tipos de software enfocados a facilitar al usuario la creación y visualización de gráficos en los computadores.



Fig. 2. Demostración grafica de la Commodore 64 en 1984.

Entre los años 1970 a 1980 los gráficos por computadora evolucionaron hasta llegar al mundo del entretenimiento, gracias a consolas de videojuegos como el Magabox Odisey o los primeros arcades de Atari, además de los primeros micro ordenadores en comercializarse por esos años de las marcas Comodore y la Sinclair, fue como se las capacidades graficas de las computadoras se definieron por representaciones planas de *Pixeles* (acrónimo del inglés picture element, 'elemento de imagen', es la menor unidad homogénea en color que forma parte de una imagen digital).

III. ESTÁNDAR ACTUAL.

Gracias a este estallido en la demanda de más software y hardware para la graficación en computadoras, inicio una guerra por estandarizar la tecnología de creación de gráficos con la creación de APIs, (acrónimo del inglés interfaz de programación de aplicaciones, conocida también por la sigla API, en inglés, *application programming interface*) por diferentes empresas dedicadas a crear software para el procesamiento de gráficos 3D como Microsoft que forzó su API (DirectX) directa a usarse en sus sistemas operativos, además de Microsoft, otras empresas de la industria informática (Sun Microsystems, Hewlett-Packard e IBM, entre otros) desarrollaron sus APIs para gráficos para competir pero al final de la década el resultado de estas tecnologías fue la creación de OpenGL. [2]

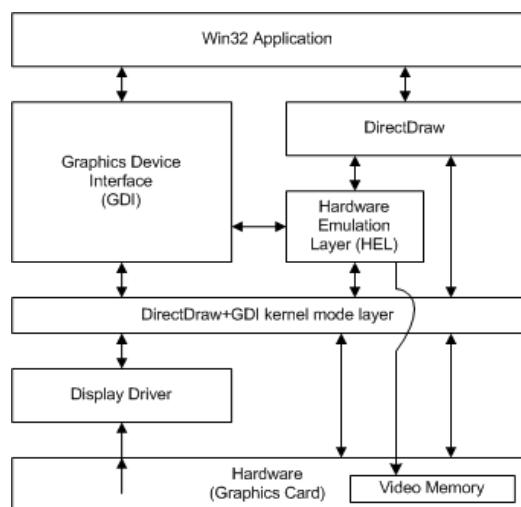


Fig. 3. Diagrama que explica el controlador DirectDraw base de la API DirectX base de la comunicación tecnológica entre el software y hardware.

En 2000, Microsoft lanzó una nueva gama de dispositivos y acuñó el término Microsoft Tablet PC para las tabletas construidas según la especificación de Microsoft. Los Tablet PC, basados en la arquitectura de CPU x86, son PC completamente funcionales que admiten pantalla táctil, en lugar de una pantalla, un mouse y un teclado tradicionales, este sumado a los nuevos dispositivos de inicios del milenio permitió que los gráficos en 3D fueran ahora portables para dispositivos tanto profesionales como de entretenimiento, durante este último siglo lo que ha significado un gran avance tanto en la cantidad de polígonos como en la capacidad de verse en dispositivos cada vez más pequeño ha sido un significativo avance en una industria como poco más de 50 años.[2]

IV. APLICACIÓN Y RESULTADOS EN LA INDUSTRIA.

Podemos encontrar muchas áreas de aplicación especialmente en la ingeniería donde gobiernos o la industria privada han tenido excelentes resultados en emplear tecnología de este tipo como, en dos ciudades muy grandes como Berlín Alemania y Rotterdam, Holanda que lograron representar ambas ciudades desde una web donde el reto está en la gran cantidad de información y modelos a manipular. [3]



Fig. 4. Modelo 3D de la ciudad de Rotterdam representación del Proyecto CityGML

Dentro de la ingeniería civil otro proyecto muy puntero es el de la universidad de Las Palmas de Gran Canaria en Las Palmas, España donde gracias a la gran cantidad de datos abiertos se plantea la generación de modelos para simulaciones y predicciones que busca un mejor enfoque de la planificación urbana y espacios adaptados a las necesidades de los ciudadanos.[4]



Fig. 5. Mapa que muestra una nube de puntos para ayudar en la planificación urbana en el Proyecto *Generating 3D City Models from Open LiDAR*.

Muy cerca en Portugal en el departamento de Ingeniería Civil, Universidad de Aveiro, en ese departamento tienen comoprioridad desarrollar los conocimientos prácticos que permiten la aceleración de la innovación en industrias como la construcción, ayudando a futuras investigaciones y su acción para madurar el modelo de esa universidad y su investigación de los datos relacionados a la construcción.

El modelado de información de edificios (BIM), como metodología ha venido siendo esencial; apoyando el desarrollo de la industria de la construcción, incluyendo el concepto/diseño, la planificación/programación de los trabajos de construcción o el mantenimiento/gestión de edificios. En el sector de la construcción, BIM ha automatizado los procesos ofreciendo una gran ventaja al disminuir las discrepancias entre el modelos planificados y construidos, ahorrando dinero y tiempo y contribuyendo a evitar errores, limitar los riesgos y reducir las divergencias de costes.[5]

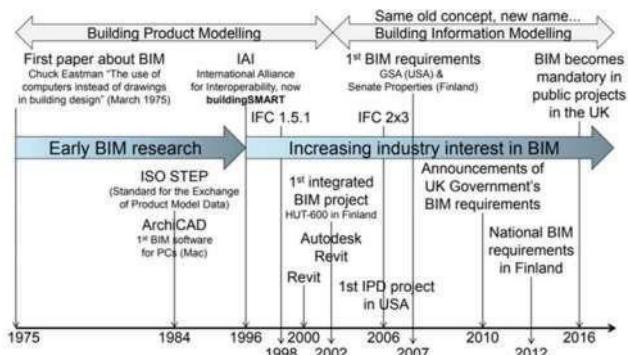


Fig. 6. Evolución de la metodología BIM.

En los usos muy particulares relacionados a la representación de modelos en 3d en el mundo como, por ejemplo, en el *Tokyo Research Laboratory*, *IBM Japan* donde ya poseen un *framework* para el procesamiento de información y visualización de 3D para la industria. En diversas áreas de la industria, la acumulación y la comunicación del conocimiento se está volviendo más importante que nunca.

Especialmente en las áreas de fabricación, este ciclo de conocimiento, que se puede aplicar a través de la discusión, la

toma de decisiones, la planificación y la producción procesos, se ha acortado recientemente, y por lo tanto su la eficiencia es más importante que nunca.

Los contenidos se recogen a través de muchas discusiones, los participantes en otra discusión pueden recuperar varios objetos del contenido recopilado especificando un objeto clave. Nosotros Considera que el primero de estos dos procesos cae en la categoría de colaboración web para múltiples usuarios comunicación, mientras que el segundo cae en el de búsqueda de geometría para la sistematización de objetos 3D. [6]

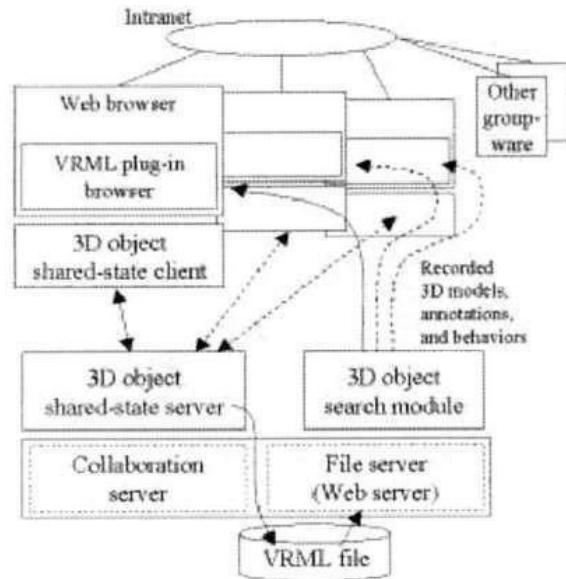


Fig. 7. Estructura del *framework* desarrollado por IBM para el manejo de modelos en 3D.

Otro Proyecto muy significativo que aporta la flexibilidad de visualización en estos modelos Podemos encontrarlo en Korea del sur donde la escuela de ciencias computaciones en *Kyungpook*, *Daegu* donde se busca emular el motor *OpenGL* en plataformas no diseñadas para ese trabajo lo que permitiría la flexibilidad de la representación de la información en general relacionada a los modelos 3D. [7]

Un uso muy novedoso e interesante es tomado por la sociedad internacional de cirugía plástica donde gracias al modelado en 3D se puede auxiliar mucho la práctica de la reconstrucción de glándulas mamarias en pacientes que hayan sufrido algún tipo de cáncer.[8]

Estos muchos avances en el mundo también hay llevado a reconsiderar otros aspectos no directamente relacionados al modelado pero si al transporte y representación de estos datos como en la conferencia de *Holography: Origin, Rediscovery, Development* en Shanghai China, donde el envío de hologramas por medio de fibra óptica aprovechando fenómenos físicos en el comportamiento de la transmisión permite la transmisión de estos datos de una forma sorprendentemente rápida y finalmente en *Technische Universität Darmstadt* en Alemania, donde se estudia la creación y el software disponible que dependiendo de las necesidades del proyecto en la universidad estos pueden ser una buena opción para el uso y desarrollo de la tecnología en aquel país. [9]

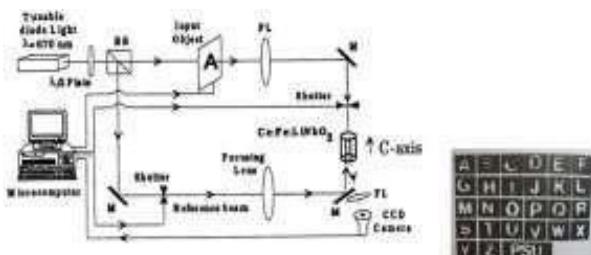


Fig. 8. Configuración experimental para la construcción de hologramas de fibra PR.

Esta gran cantidad de innovaciones y proyectos interesantes determinan que existe un gran futuro en la representación y procesamiento de datos en el manejo de modelos en 3D, para planificación urbana de universidades o grandes ciudades incluso auxiliares en disciplinas muy diferentes, como la cirugía plástica, la aeronáutica, química y geografía se puede deducir que actualmente está disciplina es muy socorrida para la toma de decisiones en todas estas disciplinas por tal motivo el manejo y representación de estos datos se ha convertido en prioridad para muchas organizaciones, gobiernos y universidades del mundo.

V. REPRESENTACIÓN 3D EN LA WEB.

Hoy en día, rápido desarrollo tecnológico para adquirir información; unido a las capacidades para procesar estos datos en un período relativamente corto de tiempo, permite la generación de modelos detallados que se convertirán en una parte esencial de la infraestructura de información moderna y se puede utilizar para integrar diversos datos de diferentes fuentes para visualización accesible al público y muchas otras aplicaciones.

Estos modelos son representados en su mayoría gracias a el uso de motores gráficos, en la actualidad existen muchos para diferentes necesidades que tengan los proyectos a desarrollar estos motores ya que su función principal es realizar el proceso de creación de dichos gráficos, mediante cálculos y técnicas propias del motor.

Pero para crear esas aplicaciones se necesita una base robusta que pueda, gestionar los recursos informáticos y la presentación de esta información, por lo que recientemente es pertinente el uso de un *framework* web.

Un *framework* para aplicaciones web es un marco de trabajo diseñado para apoyar el desarrollo de sitios web dinámicos, aplicaciones web y servicios web. Este tipo de *frameworks* intenta aliviar el exceso de carga asociado con actividades comunes usadas en desarrollos web. Por ejemplo, muchos *frameworks* proporcionan bibliotecas para acceder a bases de datos, estructuras para plantillas y gestión de sesiones, y con frecuencia facilitan la reutilización de código.[10]

VI. METODOLOGÍA PARA EL DESARROLLO.

Actualmente, las metodologías en el desarrollo se plantean de acuerdo a el tamaño del proyecto y a los recursos disponibles, en este caso particular se optará por la metodología scrum para esta planeación, *Scrum* es una Metodología Ágil de Gestión de Proyectos que se basa en la adaptación continua a las circunstancias evolutivas del Proyecto apoyándose en iteraciones cortas conocidas como *Sprints* a través del siguiente ciclo:

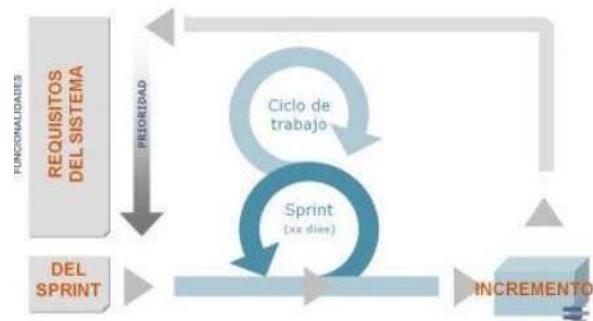


Fig. 9. Diagrama que explica los ciclos de trabajo e incrementos en el software completo para un proyecto usando Scrum.

El funcionamiento y seguimiento adecuado de esta metodología está marcado por sus roles ya que son las personas encargadas del software que se está desarrollando.

Roles/Responsabilidades:

Comprometidos:

- Propietario del Producto (*Product Owner*): responsable de lograr el mayor valor del producto desarrollado.
- Equipo de Desarrollo (*Team*): responsable de desarrollar el producto.

Implicados:

- Otros interesados (*Stakeholders*): intervienen de manera indirecta o tienen intereses en el proyecto.
- *Scrum Master*: responsable del funcionamiento de *Scrum*.

El control de este personal durante los *Sprints* se puede producir paulatinamente por reuniones de diferentes tipos que llevarán el seguimiento del software que se está desarrollando.

Reuniones:

- Planificación del *Sprint*: reunión donde el Equipo define la Pila del Sprint a partir de la explicación de la Pila del Producto por parte del Propietario del Producto.
- Seguimiento del *Sprint*: reunión rápida diaria donde el Equipo revisa las tareas de la Pila del Sprint que ha realizado el día anterior, las que hará en el día y las posibles necesidades y impedimentos que tenga para continuar el trabajo.
- Revisión del *Sprint*: reunión informativa donde el Equipo presenta al Propietario del Producto el Incremento.
- Retrospectiva: reunión de, “mejora continua” donde el Equipo analizará los diferentes problemas encontrados y los aspectos mejorables de la aplicación de *Scrum*.

Equipo de Desarrollo:

Para *Scrum* el equipo es el grupo multidisciplinar y autoorganizado cuyo trabajo se realiza de forma cohesionada en aras de completar las tareas de los diferentes *sprints* que se generan a partir de la pila del producto. Frente al concepto de grupo de trabajo, el equipo, respondiendo en su conjunto y sin la necesidad de un gestor que marque su ritmo, trabaja de manera colaborativa en la búsqueda de proporcionar el mayor valor posible al cliente.

En un entorno distribuido los miembros del equipo deben mostrar un mayor compromiso y una mayor disciplina en la gestión de sus quehaceres personales, así como con el uso de las herramientas que facilitan la interacción ya que, en tales ambientes, es mucho más fácil que estos se vean comprometidos.

Esto implica, por tanto, una mayor responsabilidad de interacción por parte del equipo, que no deberá dejar caer sobre el *Scrum Manager* la responsabilidad de perseguir, a través de los diferentes mecanismos y herramientas de comunicación establecidos, a los miembros del equipo en busca de, entre otros, la visibilidad de los avances y de los impedimentos que requiere esta metodología.

En definitiva, al trabajar en un *Scrum* Distribuido los miembros del equipo deberían exigirse una mayor disciplina personal, viendo las herramientas como extensiones o ayudas que, en estos contextos, facilitarán la interacción y la autoorganización del equipo.

VII. SERVIDOR PARA EL CONTROL DE VERSIONES Y GESTIÓN DEL PROYECTO.

Para el control de todo el software producido en este caso utilizaremos *Azure DevOps Server* es un producto de Microsoft que proporciona control de versiones, informes, gestión de requisitos, gestión de proyectos, capacidades de gestión de versiones, pruebas y compilaciones automatizadas. Cubre todo el ciclo de vida de la aplicación y habilita las capacidades de *DevOps*. *Azure DevOps* se puede utilizar como *back-end* para numerosos entornos de desarrollo integrados (*IDE*), pero es importante tener en cuenta que está específicamente diseñado para *Microsoft Visual Studio*, *Eclipse* y todas las tecnologías enfocadas a ecosistemas en Microsoft.

VIII. PLAN DE RECOLECCIÓN DE LA INFORMACIÓN PARA EL TRABAJO DE CAMPO.

Gracias a él instrumento antes mencionado se puede directamente con desarrolladores y los informes de *Azure DevOps* tener un panorama general antes, durante y después de la información del desarrollo.

Fuentes para la obtención de los datos en el trabajo de campo:

- *Azure evOps*.
 - Código.
 - Graficas de desempeño.
 - Graficas de performance de software.
 - Reporte de errores.
 - Planes de pruebas.
 - Graficas de performance de software.
- *Desarrolladores*.
 - Informes de código.
 - Informes de desarrollo.
 - Informes de scrum.
 - Retrospectivas.

IX. CONCLUSIONES.

El enfoque de las tecnologías web, tiene una visión de las necesidades que requiere la industria para la administración de recursos e información según los más altos estándares de calidad y respuesta como menciona Ryo Yoshida de IBM en su publicación *3D web environment for knowledge management* cito: "Hoy, el objetivo general de la gestión del conocimiento es poner activos intelectuales en un ciclo de conocimiento, es decir, almacenar, recopilar, reutilizar, compartir y actualizar valiosa experiencia, sabiduría y conocimiento de las personas que pertenecen a un equipo de trabajo de manera eficiente y de forma visible, con el objetivo de mejorar la capacidad de respuesta, el tiempo del ciclo de desarrollo, la productividad y competencia." [11]

Gracias a los ciclos de desarrollo y un enfoque en la productividad en el software a desarrollar de nuestra planificación e implementación serán los óptimos siguiendo las nuevas las metodologías de desarrollo ágil y planificación además teniendo las herramientas adecuadas y con el contexto antes mencionado será posible implementar nuestro *framework* web en un entorno profesional y sin complicaciones.

REFERENCIAS:

- [1] D. Salomon (emeritus), «Historical Notes», en *The Computer Graphics Manual*, London: Springer London, 2011, pp. 9-27.
- [2] J. Peddie, *The History of Visual Magic in Computers*. London: Springer London, 2013, pp 199-200.
- [3] F. Prandi, F. Devigili, M. Soave, U. Di Staso, y R. De Amicis, «3D web visualization of huge CityGML models», *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, vol. XL-3/W3, pp. 601-605, ago. 2015.
- [4] S. Ortega, J. M. Santana, J. Wendel, A. Trujillo, y S. M. Murshed, «Generating 3D City Models from Open LiDAR Point Clouds: Advancing Towards Smart City Applications», en *Open Source Geospatial Science for Urban Studies*, A. Mobasher, Ed. Cham: Springer International Publishing, 2021, pp. 97-116.
- [5] H. Rodrigues, F. Gaspar, P. Fernandes, y A. Mateus, Eds., *Sustainability and Automation in Smart Constructions: Proceedings of the International Conference on Automation Innovation in Construction (CIAC-2019)*, Leiria, Portugal. Cham: Springer International Publishing, 2021.
- [6] R. Yoshida, T. Murao, y T. Miyazawa, «3D web environment for knowledge management», *Future Generation Computer Systems*, vol. 17, n.º 1, pp. 73-78, sep. 2000.

- [7] N. Baek, «An emulation scheme for OpenGL SC 2.0 over OpenGL», *Supercomput*, vol. 76, n.º 10, pp. 7951-7960, oct. 2020.
- [8] G. S. Hura, A. K. Singh, y L. Siong Hoe, Eds., *Advances in Communication and Computational Technology: Select Proceedings of ICACCT 2019*, vol. 668. Singapore: Springer Singapore, 2021, pp 1153.
- [9] F. T. S. Yu, «Holography: Origin, Rediscovery, Development and Beyond», p. 12, 2015.
- [10] 15 Ene 2010. Web application framework. Disponible:
https://web.archive.org/web/20150723163302/http://docforge.com/wiki/Web_application_framework. [Consultado Jun. 22, 2021].
- [11] R. Yoshida, T. Murao, y T. Miyazawa, «3D web environment for knowledge management», *Future Generation Computer Systems*, vol. 17, n.º 1, pp. 73-78, sep. 2000.