



TECNOLÓGICO DE ESTUDIOS SUPERIORES DE COACALCO

INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIONES

Informe técnico de residencias profesionales

Aplicativo industrial para el seguimiento y control de procesos de manufactura en líneas de ensamble de producto terminado

Presenta:

201921157 - Herrera del Castillo David Aaron

Asesores:

Externo: Ing. Luis Uriel García Sánchez

Interno: M. en C.C. José Alejandro Pineda Aguillón

Agradecimientos

Este trabajo es el resultado de un esfuerzo y dedicación que abarca no solo el período de residencias profesionales, sino también los últimos 4 años y medio de mi desarrollo profesional en el plantel del Tecnológico de Estudios Superiores de Coacalco. Quiero expresar mi sincero agradecimiento a aquellos que han sido parte fundamental de este trayecto.

Agradezco a mis padres, quienes me respaldaron de todas las formas posibles durante este camino. Me inculcaron valores sólidos, la importancia de la constancia, la responsabilidad y el hábito de concluir lo que se empieza. Por todo su amor y comprensión, les estoy enormemente agradecido.

Agradezco también a mis maestros y profesores que me acompañaron a lo largo de mi recorrido en mi carrera como Ingeniero en Tecnologías de la Información y Comunicaciones. Su orientación y conocimientos han sido fundamentales para mi crecimiento académico y profesional. La calidad de la educación recibida ha dejado una huella significativa en mi formación y en la realización de este proyecto.

Agradezco a todos mis compañeros y amigos que compartieron este camino conmigo, brindando apoyo, colaboración y momentos de camaradería que enriquecieron mi experiencia. Cada interacción contribuyó al aprendizaje mutuo y a la construcción de un ambiente propicio para el crecimiento en conjunto.

Por último, pero no menos importante quiero expresar mi más sincero agradecimiento a BAACC SYSTEMS S.A. de C.V., la empresa que me brindó la invaluable oportunidad de llevar a cabo mi proyecto de residencias profesionales. Su generosidad al proporcionarme este entorno de aprendizaje y su apoyo continuo han sido fundamentales para el éxito de este proyecto. Agradezco la confianza depositada en mí y por ofrecer todas las facilidades necesarias, lo que ha contribuido de manera significativa al desarrollo y conclusión exitosa de este trabajo.

“Las grandes oportunidades nacen de haber sabido aprovechar las pequeñas”

- Bill Gates

Resumen

En el presente trabajo, se detallan las iniciativas llevadas a cabo en la implementación de un aplicativo interactivo destinado a supervisar y gestionar las líneas de ensamble de productos terminados en la planta de Querétaro de INOAC Sistemas Exteriores S.A. de C.V. Desarrollada con tecnología 4.0 y servicios en la nube, la aplicación busca elevar la eficiencia y visibilidad de los procesos de producción. La adopción de prácticas innovadoras, como la automatización y la integración de datos en tiempo real, busca ofrecer un control más efectivo. Este enfoque refleja la estrategia de INOAC para avanzar significativamente hacia la industria 4.0, buscando continuamente mejorar sus operaciones mediante la implementación de soluciones tecnológicas avanzadas. La aplicación no solo representa un paso hacia la modernización dentro de la empresa, sino también un impulso para la competitividad y la agilidad operativa en el cambiante panorama industrial actual.

ÍNDICE

INDICE DE TABLAS	7
CAPÍTULO I: GENERALIDADES DEL PROYECTO	8
1.1 Introducción	8
1.2 Descripción de la empresa.....	9
1.3 Descripción del puesto	11
1.4 Problemáticas (ordenadas por prioridad).....	12
1.5 Justificación.....	13
1.6 Objetivos	15
1.6.1 Objetivo general	15
1.6.2 Objetivos específicos.....	15
CAPITULO II: MARCO TEÓRICO.....	16
2.1 Marco metodológico	16
2.1.1 Enfoque Metodológico	16
2.1.2 Proceso SCRUM.....	16
2.1.3 Herramientas y Tecnologías	17
2.1.4 Evaluación y Pruebas	18
2.1.5 Implementación y Seguimiento.....	18
2.2 Marco referencial	19
2.2.1 La industria 4.0.....	19
2.2.2 Cloud Computing	21
2.2.3 Sistema de información	24
2.2.4 Automatización de procesos.....	25
2.2.5 Arquitectura de capas	27
2.2.6 Programación Orientada a Objetos.....	29
2.3 Herramientas de desarrollo.....	31
2.3.1 Visual Studio.....	31
2.3.2 Microsoft SQL Server	32
2.2.3 .NET MAUI	33
2.2.4 Tkinter.....	34

CAPITULO III: DESARROLLO	35
3.1 Análisis.....	35
3.2 Diseño	37
3.3 Codificación	42
3.4 Pruebas	46
CAPITULO IV: RESULTADO	48
4.1 Historias de usuario	48
4.2 Release plan.....	48
4.3 Análisis, diseño y modelado de la estructura	49
4.3.1 Product Backlog.....	49
4.3.2 Sprint Backlog.....	49
4.3.3 Definition Of Done.....	50
4.4 Codificación del sistema automatizado de control de procesos	50
4.4.1 Product Backlog.....	50
4.4.2 Sprint Backlog.....	51
4.4.3 Diagrama de flujo	52
4.4.4 Caso de uso	52
4.4.5 Definition Of Done.....	53
4.5 Codificación del sistema integrado a multiplataforma y registro de paros.....	53
4.5.1 Product Backlog	53
4.5.2 Sprint Backlog.....	53
4.5.3 Diagrama de flujo	55
4.5.4 Caso de uso	56
4.5.5 Definition Of Done.....	56
4.6 Pruebas finales e implementación del aplicativo en producción	56
4.6.1 Product Backlog	56
4.6.2 Sprint Backlog.....	57
4.6.3 Definition Of Done.....	58
4.7 Burndown chart general del proyecto.....	59
CONCLUSIONES	60
COMPETENCIAS DESARROLLADAS	62

FUENTES DE INFORMACIÓN.....	65
ANEXOS	67

ÍNDICE DE ILUSTRACIONES

Ilustración 1. Organigrama de BAACC SYSTEMS S.A. de C.V	10
Ilustración 2. Croquis de BAACC SYSTEMS SA. de C.V.....	10
Ilustración 3. Tecnologías de la industria 4.0 - Fuente: IEBS Digital School	20
Ilustración 4. Arquitectura del Cloud Computing – Fuente: Quizlet.....	22
Ilustración 5. Diferencias entre servicios Cloud – Fuente: Red Hat.....	24
Ilustración 6. Esquema de la arquitectura de capas – Fuente: Platzi	27
Ilustración 7. Ejemplo de programación orientada a objetos – Fuente: Profile.....	30
Ilustración 8. Arquitectura de Visual Studio – Fuente: Microsoft.....	31
Ilustración 9. Arquitectura de .NET MAUI – Fuente: Microsoft	33
Ilustración 10. Wireframe dashboard principal, primera versión	37
Ilustración 11. Wireframe del sistema de registro de paros, visualización principal.....	38
Ilustración 12. Wireframe del registro de un paro recibiendo parámetros	39
Ilustración 13. Wireframe de pantalla de paro activo en producción	39
Ilustración 14. Wireframe nuevo diseño vertical del aplicativo sobre tótem interactivo.....	40
Ilustración 15. Wireframe de pantalla de autenticación para supervisores.....	41
Ilustración 16. Estructura del proyecto en .NET MAUI.....	44
Ilustración 17. Estructura básica de las tablas añadidas a la base de datos para el registro de paros	45
Ilustración 18. Diagrama de flujo de la codificación del sistema automatizado de control de procesos.....	52
Ilustración 19. Caso de uso de la codificación del sistema automatizado de control de procesos	52
Ilustración 20. Diagrama de flujo del sistema integrado a multiplataforma y registro de paros ...	55
Ilustración 21. Caso de uso del sistema integrado a multiplataforma y registro de paros	56
Ilustración 22. Burndown chart general del proyecto.....	59

INDICE DE TABLAS

Tabla 1. Comparación de Servicios Cloud	23
Tabla 2. Atributos de calidad para el aplicativo.	36
Tabla 3. Historias de usuario del aplicativo	48
Tabla 4. Release plan de versiones del aplicativo	48
Tabla 5. Sprint 1 del análisis, diseño y modelado de la estructura.....	49
Tabla 6. Sprint 6 del análisis, diseño y modelado de la estructura.....	50
Tabla 7. Sprint 3 de la codificación del sistema automatizado de control de procesos	51
Tabla 8. Sprint 4 de la codificación del sistema integrado a multiplataforma y registro de paros.....	54
Tabla 9. Sprint 5 de las pruebas finales e implementación del aplicativo en producción.....	57
Tabla 10. Sprint 6 de las pruebas finales e implementación del aplicativo en producción.....	57
Tabla 11. Competencias desarrolladas durante el desarrollo del proyecto	62

CAPÍTULO I: GENERALIDADES DEL PROYECTO

1.1 Introducción

En los últimos años, hemos sido testigos de un cambio significativo en el panorama empresarial, donde la digitalización en constante crecimiento a nivel mundial ha intensificado la demanda de tecnologías de la información y la comunicación. Según menciona ISEC (2022), un número creciente de empresas a comenzado a realizar en los últimos años inversiones significativas en herramientas y soluciones para mantenerse a la vanguardia en sus respectivas industrias y lo que es más importante, superar a la competencia. Este cambio en la adopción de estas tecnologías está transformando la forma en que las empresas operan y compiten en el mercado actual.

Específicamente en el ámbito de manufactura y ensamble a nivel industrial el uso de nuevas tecnologías es profundamente transformador. Estas tecnologías han allanado el camino hacia la automatización de procesos, la recopilación instantánea de datos y una mayor conectividad en la cadena de producción. Esto se traduce en una producción más eficiente, una gestión mejorada del inventario y la capacidad de realizar un seguimiento preciso de cada fase del proceso de ensamble.

INOAC, también conocida como 'Innovation & Action', es una destacada empresa internacional en la industria de polímeros que se dedica al desarrollo de materiales funcionales. Esta empresa opera una subsidiaria en México, INOAC Sistemas Exteriores S.A. de C.V. (IESM), que se especializa en la fabricación de pinturas y piezas para automóviles, que suministra a fabricantes OEM (Original Equipment Manufacturer) japoneses y estadounidenses. Ubicada en el estado de Querétaro, esta planta está explorando la implementación de soluciones basadas en la industria 4.0 con el objetivo de mejorar la eficiencia y productividad en su producción.

Este proyecto se enfoca en la optimización de la gestión y visualización de información en las líneas de ensamble de producto terminado en la planta de IESM. La automatización desempeñará un papel esencial al proporcionar una forma más eficiente y confiable de acceder y utilizar los datos que provienen de las líneas de producción. Esta iniciativa se alinea con la búsqueda de mejoras en la eficiencia y la competitividad de la empresa.

1.2 Descripción de la empresa

BAACC SYSTEMS S.A. de C.V. es una empresa 100% mexicana que se especializa en el desarrollo de soluciones tecnológicas innovadoras, manteniendo compromiso su con los más altos estándares de calidad de desarrollo. Su enfoque se centra en la simplificación de procesos, el aumento de la productividad organizacional y la reducción de costos para sus clientes. A lo largo de los años, se ha consolidado como una consultoría de tecnologías de la información líder, proporcionando servicios y soluciones en diversas áreas de las tecnologías de la información y comunicaciones (TIC).

Misión: Desarrollar bajo los más altos estándares de calidad soluciones tecnológicas e innovadoras, siempre enfocadas a simplificar procesos, aumentar la productividad organizacional y lograr una reducción de costos.

Visión: Llegar a ser una empresa con liderazgo y prestigio nacional, para generar con los mejores talentos y prácticas, soluciones tecnológicas que contribuyan al éxito de nuestros clientes, empleados y accionistas.

Valores:

- **Lealtad.** Mantener siempre una línea de confianza y compromiso con nuestros clientes
- **Disciplina.** Establecemos orden estricto a nuestras prioridades, con claro enfoque de servicio a nuestros clientes, buscando cumplir y superar sus expectativas
- **Confidencialidad.** Garantizar que el acceso a la información solo será permitido a personas autorizadas, manteniendo con ello la confianza de nuestros clientes, empleados y accionistas
- **Trabajo en equipo.** Plena integración entre los colaboradores de la empresa de tal forma que nos permita cumplir con objetivos en común

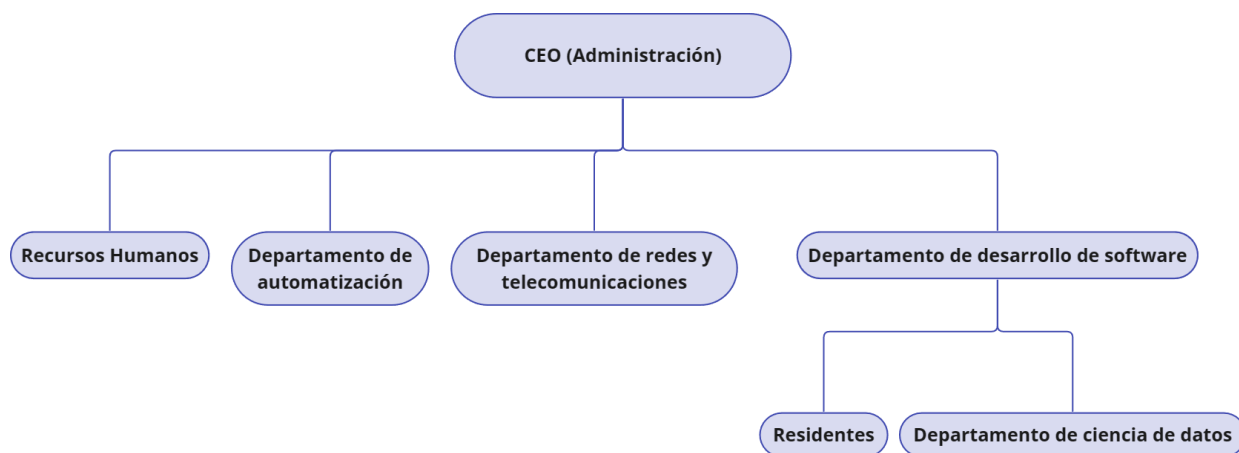


Ilustración 1. Organigrama de BAACC SYSTEMS S.A. de C.V.

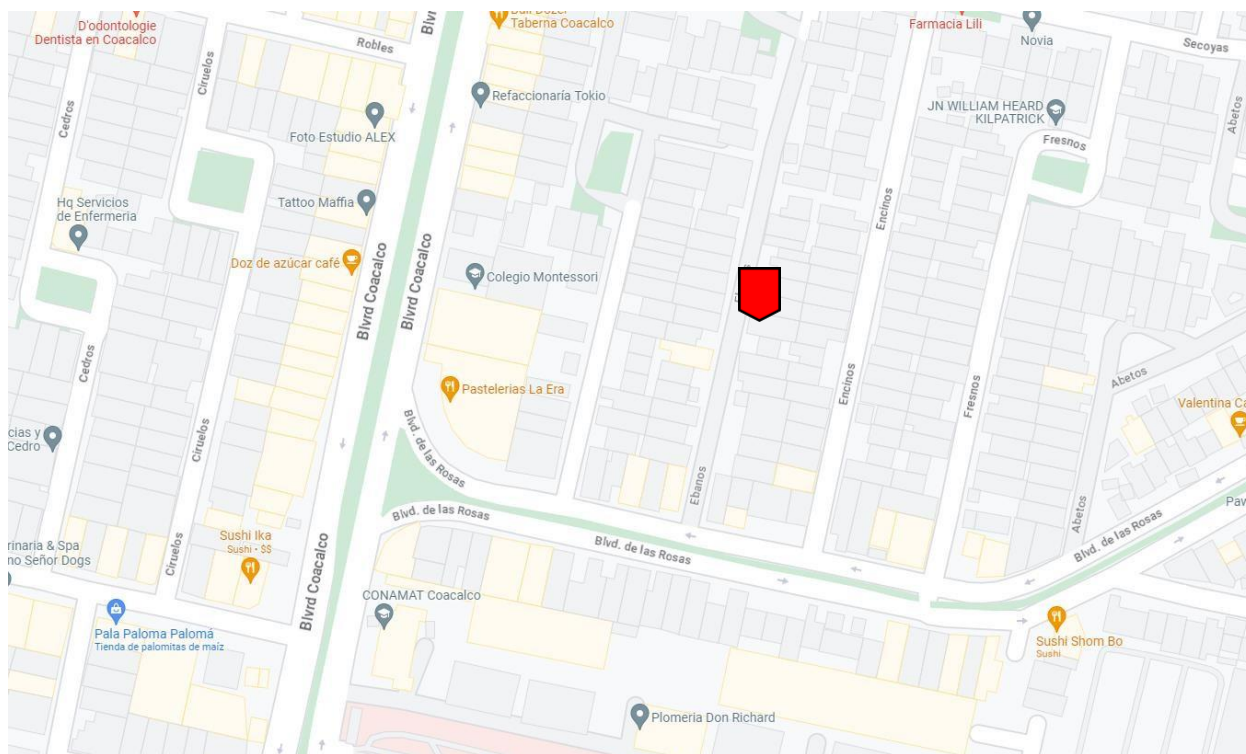


Ilustración 2. Croquis de BAACC SYSTEMS S.A. de C.V.

1.3 Descripción del puesto

Como residente dentro de la empresa BAACC SYSTEMS S.A. de C.V., participé en una variedad de actividades que contribuyeron al desarrollo y éxito del proyecto. Estas actividades incluyeron:

- Programación de aplicaciones con Python utilizando la librería de Tkinter
- Programación de aplicaciones con C# y haciendo uso del marco de trabajo (framework) multiplataforma con .NET MAUI
- Documentación técnica del desarrollo de los aplicativos desarrollados
- Manejo de versiones de código haciendo uso de Git, GitHub y Team Foundation Azure (TFVC)
- Manejo y administración de bases de datos con SQL Server
- Creación de procedimientos almacenados y esquemas para realizar CRUD en las bases de datos

1.4 Problemáticas (ordenadas por prioridad)

IESM cuenta con un sistema de control del proceso que se trabaja en todas sus líneas de ensamble completamente manual, en el que se registran detalles de interés durante el proceso, estos son registrados en unas pizarras en medio de todas las líneas de ensamble y siempre incluyendo los siguientes datos:

1. Cliente para el que se está ensamblando este producto
2. Nombre del producto que se está ensamblando
3. Conteo de piezas producidas por hora (rate) y por turno (8 horas en cada turno)
4. Motivos y duración de paro de producción (si es que hubo)
5. Momento del reinicio de la producción

Al finalizar cada turno, los datos son borrados sin guardar un historial más completo de los detalles de ese turno o esa pieza en especial, únicamente datos generales como el rate final, producto y fecha. Este enfoque manual ha dado lugar a una serie de problemáticas en la empresa, los cuales se detallan a continuación:

- a) **Ineficiencia en la gestión del tiempo:** El registro manual de eventos de paro y reinicio de producción consume tiempo valioso de los empleados en la línea de ensamble, generando demoras en la producción, ya que los empleados no pueden continuar con sus tareas y aumentando los costos operativos al requerir más mano de obra para tareas administrativas.
- b) **Seguridad y fiabilidad de los datos:** La recopilación manual de datos puede plantear preocupaciones sobre la seguridad de la información. Los datos críticos de producción están expuestos a la manipulación, lo que puede comprometer la confidencialidad de estos y la integridad del proceso de ensamble, esto porque cualquier persona en la línea podría alterar la información al ser un pizarrón libre.
- c) **Falta de datos precisos y medibles:** La carencia de un sistema preciso de control del rate de ensamble dificulta la capacidad de la empresa para evaluar si se están cumpliendo las metas de producción. La ausencia de datos medibles y confiables limita la capacidad de los supervisores para tomar decisiones informadas y ajustar la producción según sea necesario ya que el historial detallado de producción es borrado al terminar cada turno.
- d) **Limitaciones en la accesibilidad de datos:** El sistema de registro manual en un pizarrón puede presentar desafíos en cuanto a la accesibilidad de datos. Los equipos de gestión y supervisión pueden encontrar dificultades para acceder a la información de manera oportuna, lo que puede retrasar la toma de decisiones basadas en datos en tiempo real, lo que es crucial para una gestión eficaz.
- e) **Costos y desperdicio:** El proceso manual de registro implica gastos operativos adicionales, como la compra de papel y material de oficina. Además, es menos sostenible desde una perspectiva ambiental. La inversión en soluciones automatizadas y digitales puede ofrecer ahorros a largo plazo y una mayor eficiencia operativa.

1.5 Justificación

IESM, como cliente de BAACC SYSTEMS, ha reconocido la necesidad de modernizar sus operaciones, haciendo énfasis en el ámbito de las tecnologías 4.0. En un esfuerzo por mejorar sustancialmente los procesos de control de líneas de ensamble y aprovechando la experiencia y calidad de los desarrollos proporcionados por BAACC SYSTEMS en otras áreas, IESM ha optado por dar un giro radical a la forma en que lleva a cabo el control de este proceso. La propuesta se centra en la implementación de un sistema automatizado de control y registro de piezas en cada una de las líneas de ensamble, respaldado por una aplicación multiplataforma con enfoque móvil conectada a un sistema de trazabilidad de piezas previamente desarrollado por BAACC SYSTEMS, que permite controlar el ciclo de vida de los productos, marcando un paso significativo hacia la modernización y la mejora continua, sustituyendo la manera anterior en la que se trabajaba utilizando una pizarra.

En relación con la **ineficiencia en la gestión del tiempo**, derivada del registro manual de eventos de paro y reinicio de producción, impacta negativamente en la productividad de la línea de ensamble. La implementación de un sistema automatizado no solo eliminará los retrasos causados por la tarea manual, sino que también reducirá los costos operativos asociados al uso excesivo de mano de obra, haciendo uso del sistema de trazabilidad de piezas podemos controlar en qué momento se termina de ensamblar un producto, por lo que ya no será necesario que nadie registre manualmente datos, estos serán obtenidos de una base de datos conectada al sistema de trazabilidad permitiendo agilizar los tiempos muertos que generaba el anterior sistema.

Para la **seguridad y fiabilidad de los datos**, el cambio a un sistema automatizado y digital asegurará la integridad de la información, mitigando la posibilidad de manipulación indebida y garantizando la confidencialidad de los datos de producción, ya que al ser datos que nadie va a poder manipular directamente en el sistema ya que son obtenidos del sistema de trazabilidad nos permite tener una capa de seguridad para garantizar su integridad y fiabilidad.

En cuanto a la falta **de datos precisos y medibles**, la carencia de un sistema de control preciso dificulta la evaluación de metas de producción. Al implementar un sistema automatizado y conectado con trazabilidad, se garantiza la captura exacta y medible del rate de ensamble ya que son datos que no son registrados por cualquier persona en las líneas, proporcionando datos precisos que permitirán a los supervisores tomar decisiones informadas y ajustar la producción eficientemente.

En cuanto a **las limitaciones en la accesibilidad de datos**, el sistema manual actual presenta desafíos en la accesibilidad de datos para los equipos de gestión y supervisión. La transición a un sistema automatizado facilitará el acceso oportuno y eficiente a la información en tiempo real, mejorando la capacidad de toma de decisiones basada en datos, el sistema permitirá que únicamente supervisores de las líneas puedan hacer modificaciones a datos como el producto,

el cliente y la línea que se está trabajando, por medio de un sistema de autenticación que únicamente da estos permisos a los supervisores.

Y hablando de los **costos y desperdicio**, el proceso manual implica gastos adicionales y menos sostenibilidad. La inversión en soluciones automatizadas no solo reducirá los costos operativos a largo plazo, sino que también promoverá prácticas más sostenibles al eliminar el desperdicio de papel e insumos adicionales como marcadores y material para las pizarras que se utilizaban durante el proceso anterior.

1.6 Objetivos

1.6.1 Objetivo general

Implementar un aplicativo que muestre en tiempo real el seguimiento de la producción de las líneas de ensamble de producto terminado para la empresa INOAC Sistemas Exteriores SA. de C.V. con la finalidad de sustituir el antiguo sistema de registro manual por un tótem interactivo que automatice las tareas de captura y control de las líneas de ensamble realizando una conexión con el sistema de trazabilidad de BAACC Systems S.A. de C.V., empleando herramientas de desarrollo como .NET y arquitectura MVC.

1.6.2 Objetivos específicos

- a) Desarrollar un sistema de registro de paros que permita definir puntos de inicio, duración y término, facilitando la consulta de estos datos en todo momento.
- b) Desarrollar un sistema de automatización basado en el sistema de trazabilidad para obtener información sobre la finalización del ensamble de un producto y automatizar su registro.
- c) Implementar un sistema de autenticación para supervisores, otorgándoles privilegios de edición en puntos clave durante el proceso de ensamble.
- d) Integrar procedimientos almacenados necesarios para el aplicativo, consumiendo directamente de la base de datos del sistema de trazabilidad.
- e) Integrar una arquitectura de capas que garantice la comunicación segura entre el aplicativo y la base de datos, permitiendo cambios sin afectar las funcionalidades.

CAPITULO II: MARCO TEÓRICO

2.1 Marco metodológico

Dentro del contexto empresarial de INOAC Sistemas Exteriores S.A. de C.V., donde se enfrenta a las cambiantes oportunidades de mercado, fluctuaciones en las condiciones económicas y la introducción de nuevos productos y servicios competitivos, resulta imperativo que cambien los procesos en los que se manejan y se controlan sus procesos productivos de manera ágil y dinámica. La empresa busca adaptarse rápidamente al entorno empresarial en constante evolución para lograr sus objetivos y expandir su presencia en el mercado. En este escenario, la elección de una metodología de desarrollo ágil se presenta como la opción más idónea.

Se ha optado por hacer uso de SCRUM, una metodología reconocida, debido a la necesidad de entregar desarrollos de sistemas funcionales que posibiliten la integración inmediata. Esta elección permitirá observar el desempeño del sistema, realizar ajustes según sea necesario y agregar nuevas funcionalidades que sean relevantes para entregas futuras.

2.1.1 Enfoque Metodológico

“Scrum es un marco de administración que los equipos utilizan para organizarse por cuenta propia y trabajar en aras de alcanzar un objetivo común. Describe un conjunto de reuniones, herramientas y funciones para entregar proyectos de forma eficiente.” (Amazon Web Services, 2023)

Al igual que un equipo deportivo que se prepara para un encuentro crucial, las prácticas de Scrum permiten que los equipos de trabajo se autoadministren, aprendan de las experiencias y se adapten a las modificaciones en el menor tiempo posible. En este proyecto, se optará por el uso de esta metodología ágil, seleccionando una práctica que contribuya y beneficie al desarrollo de software.

2.1.2 Proceso SCRUM

Planificación del Sprint

Definición de Objetivos: En la reunión inicial, se establecerán los objetivos específicos del sprint, considerando los requerimientos de diseño de interfaz de usuario y la funcionalidad del aplicativo con respecto en lo que se espera como apartado funcional y no funcional, así mismo también las características adicionales que brinda tener un aplicativo en lugar de un pizarrón.

Asignación de Tareas: Para mantener el orden durante el desarrollo, se definirán tareas específicas para el diseño de la interfaz, desarrollo del sistema de registro de paros, implementación de la arquitectura de capas, creación de procedimientos almacenados y realización de pruebas, esto con la finalidad de tener claro lo que se tiene que trabajar y definir tiempos en un cronograma de actividades.

Desarrollo

Programación y Diseño: Se llevará a cabo la programación de la interfaz de usuario basada en los requerimientos establecidos, seguida del desarrollo del sistema de registro de paros y la implementación de la arquitectura de capas, haciendo uso de herramientas de ayuda visual como wireframes se logra obtener una mejor perspectiva de lo que se espera.

Revisión del Sprint

Evaluación de Resultados: Al finalizar cada uno de los sprint, se realizará una revisión detallada de los resultados obtenidos hasta el momento. El equipo colaborará para identificar posibles mejoras en el diseño, funcionalidad o rendimiento del aplicativo.

Retroalimentación

Feedback del Cliente: Se obtendrá feedback del cliente, en este caso, representado por los usuarios clave de IESM, para asegurar que el desarrollo se alinee con sus expectativas y requisitos.

Ajustes y Modificaciones: Basándose en la retroalimentación recibida ya sea por el asesor del proyecto o el cliente, se realizarán ajustes y modificaciones necesarios para optimizar el aplicativo según corresponda.

2.1.3 Herramientas y Tecnologías

Lenguajes de Programación: Python (Tkinter), C# (.NET MAUI)

Gestión de Versiones: Git, GitHub, Team Foundation Azure (TFVC)

Base de Datos: SQL Server para el almacenamiento seguro de los datos de producción.

2.1.4 Evaluación y Pruebas

Pruebas Unitarias: Se realizarán pruebas en cada componente del aplicativo para garantizar su funcionamiento independiente. La detección y corrección de errores en esta etapa garantizan la solidez de cada pieza antes de su integración.

Pruebas de Integración: Se verificará la correcta integración de todos los módulos y componentes del sistema. Este proceso asegura que el conjunto de elementos trabaje de manera cohesionada, cumpliendo con los requisitos de diseño y especificaciones.

Pruebas de Rendimiento: Se llevarán a cabo pruebas de rendimiento para asegurar que el aplicativo pueda manejar la carga prevista en un entorno de producción. El objetivo es descubrir posibles limitaciones y optimizar el rendimiento para manejar eficientemente las demandas del usuario final en situaciones del mundo real.

2.1.5 Implementación y Seguimiento

Implementación Gradual: Se planificará una implementación gradual del aplicativo, comenzando con un piloto en una línea de ensamble específica y expandiéndolo progresivamente al resto de las operaciones de las demás líneas de ensamble.

Monitoreo Continuo: Se llevará a cabo interacción con el cliente para hablar acerca del desempeño del aplicativo, esto para poder realizar un monitoreo continuo y evaluar la efectividad del aplicativo en tiempo real, posteriormente realizar ajustes según sea necesario.

2.2 Marco referencial

2.2.1 La industria 4.0

Según menciona IBM (2023) en su portal web, *“la Industria 4.0 es la realización de la transformación digital del campo, que brinda toma de decisiones en tiempo real, productividad mejorada, flexibilidad y agilidad”*. Se puede entender entonces que lo que han tenido en común todas las revoluciones industriales es que han sido impulsadas por tecnologías innovadoras y que además han generado un nuevo marco productivo más eficiente, eficaz y disruptivo. El autor De Los Ríos Sánchez (2023) menciona las innovaciones que nos han presentado cada una de estas:

- a) En torno al año 1784, se desata **la primera revolución industrial** con la aparición de la pionera máquina de vapor y su capacidad para generar energía mecánica, marcando un hito histórico.
- b) **La segunda revolución industrial** surge a finales del siglo XIX con la introducción de la electricidad y el petróleo como principales fuentes de energía. En este período, emerge la cadena de producción y se instaura el concepto de división del trabajo en tareas especializadas.
- c) Alrededor de 1970, se inicia un cambio significativo en las tecnologías de la información y la electrónica, propiciando la automatización de la producción y consolidando así **la tercera revolución industrial**.
- d) Un capítulo sin precedentes se anuncia en la historia con la llegada de **la cuarta revolución industrial**, también conocida como Industria 4.0. Este nuevo paradigma se materializa con la convergencia de diversas tecnologías que posibilitan la creación de procesos altamente automatizados, marcando una etapa innovadora en la evolución industrial.

En la era de la Industria 4.0, se ha desatado una revolución que transforma radicalmente la forma en que las empresas producen, mejoran y distribuyen sus productos. Los fabricantes están adoptando innovadoras tecnologías como el Internet de las Cosas (IoT), la computación en la nube, la inteligencia artificial y el aprendizaje automático (machine learning). En estas fábricas inteligentes, sensores avanzados, software integrado y robótica colaboran para recopilar y analizar datos, permitiendo una toma de decisiones más eficiente. Este enfoque integral redefine los estándares de eficiencia en la industria y marca una nueva era en la gestión de la producción según se menciona en el portal de IBM, (2023).

La digitalización impulsa la automatización, el mantenimiento predictivo y la optimización automática de procesos, según destaca IBM (2023). Estos avances se traducen en niveles de eficiencia y capacidad de respuesta a los clientes que anteriormente eran inalcanzables. Las fábricas inteligentes representan una extraordinaria oportunidad para que la industria manufacturera se sumerja en la cuarta revolución industrial. La principal particularidad de la Industria 4.0 es la fusión de tecnologías como herramienta para ahorrar tiempo, permitir ciertas

decisiones y reducir errores. Estas tecnologías, cumplen con una serie de características según menciona el autor De Los Ríos Sánchez (2023):

- a) **Descentralización:** En Industria 4.0, la descentralización significa que las máquinas no dependen de la interferencia humana para funcionar. Se ejecutan a través de sensores conectados a una red, capaces de tomar decisiones automatizadas basadas en datos de rendimiento
- b) **Interoperabilidad:** En la Industria 4.0, hace referencia a la capacidad de los diferentes sistemas, físicos, humanos e informáticos para comunicarse entre sí. Un ejemplo simple de esto son las páginas web, que funcionan en todos los navegadores a través de estándares abiertos que permiten el acceso de datos
- c) **Virtualización:** Permite distribuir las funciones de una máquina física entre varios usuarios o entornos, posibilitando el uso de toda la capacidad de la máquina
- d) **Respuesta en tiempo real:** Las tecnologías de la Industria 4.0 alimentan datos y algoritmos de forma instantánea. Estos brindan resultados inmediatos permitiendo gestionar cualquier fallo antes de que tengan consecuencias sobre el proceso productivo
- e) **Modularidad:** Gracias a la conexión y desconexión de diferentes módulos, las empresas pueden fabricar productos en secuencia sin tener que reconfigurar toda la línea de ensamble

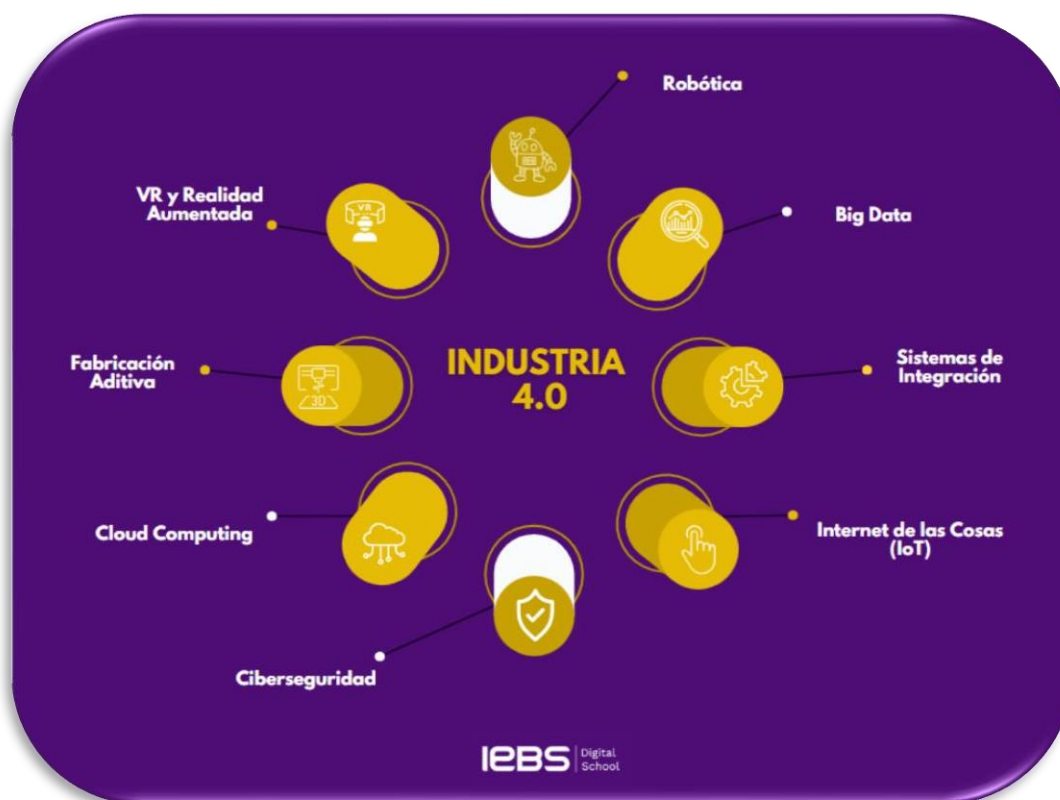


Ilustración 3. Tecnologías de la industria 4.0 - Fuente: IEBS Digital School

2.2.2 Cloud Computing

El Cloud Computing es una tecnología relativamente reciente y que llegó para quedarse, siendo un pilar esencial en el esquema de la industria 4.0, como se pudo observar en la Ilustración 3. Según se menciona en una nota de Forbes (2018), el **83% de la carga de trabajo de una empresa se almacenará en la nube**, esto hablando de finales del 2021. Esta es la realidad que hemos estado viviendo hoy en día y más aún con el acelerador tecnológico que significó la pandemia para muchas de las empresas alrededor del mundo.

La autora Silva (2021) define el Cloud Computing como el empleo de una red de servidores remotos conectados a Internet para almacenar, gestionar y procesar datos. En lugar de depender de un servidor físico ubicado en la oficina o en la computadora, se accede a una infraestructura donde el software y el hardware están virtualmente integrados.

Los modelos de servicio de computación en la nube se basan en la idea de compartir recursos informáticos, software e información a través de Internet según la demanda. Las empresas o individuos pagan por acceder a un conjunto virtual de recursos compartidos, que incluye servicios de computación, almacenamiento y redes. Estos recursos están alojados en servidores remotos propiedad de proveedores de servicios y gestionados por ellos.

Una de las ventajas clave de la computación en la nube que se nos menciona en el portal de Google (2023) es su característico modelo de pago por uso. Esto permite que las organizaciones escalen de manera ágil y eficiente sin la necesidad de adquirir y mantener sus propios centros de datos físicos y servidores.

En términos más simples, la computación en la nube utiliza una red, comúnmente Internet, para conectar a los usuarios con una plataforma en la nube. Aquí, solicitan y acceden a servicios informáticos de alquiler. Un servidor central administra toda la comunicación entre dispositivos y servidores del cliente para facilitar el intercambio de datos. *“La seguridad y privacidad son funciones comunes que aseguran la protección de esta información”* (Google, 2023).

Cuando se trata de adoptar una arquitectura de computación en la nube, no hay una solución única. Lo que funcione para otra empresa puede no ser adecuado para ti y tus necesidades empresariales. De hecho, la flexibilidad y versatilidad son características distintivas de la nube que permiten a las empresas adaptarse rápidamente a los cambios en los mercados o en las métricas.

Según la información proporcionada por RedHat (2023) en su portal, se pueden identificar tres modelos distintos de implementación de computación en la nube: nube pública, nube privada y nube híbrida. Cada uno de estos modelos presenta características y beneficios específicos:

- a) **Nube Pública:** Este modelo consiste en un conjunto de recursos virtuales desarrollados a partir de hardware perteneciente a una empresa externa, que también se encarga de

su gestión. Estos recursos son compartidos entre varios clientes y se ponen a disposición a través de una interfaz de autoservicio de manera automática. La nube pública es especialmente útil para adaptar cargas de trabajo a variaciones inesperadas de la demanda.

- b) **Nube Privada:** En contraste con el modelo anterior, la nube privada es un entorno diseñado exclusivamente para el usuario final y por lo general, se encuentra dentro del firewall del usuario. Aunque tradicionalmente se ejecutaba localmente, actualmente las empresas también la diseñan en centros de datos alquilados fuera de las instalaciones. Este tipo de nube es considerada privada cuando la infraestructura de TI subyacente se destina a un solo cliente, ofreciendo un acceso completamente aislado
- c) **Nube Híbrida:** Por otra parte, la nube híbrida implica la combinación de al menos dos entornos informáticos que comparten información entre sí y ejecutan una serie uniforme de aplicaciones para una empresa o negocio. Estos entornos pueden incluir, entre otras opciones: al menos una nube privada y una pública, dos o más nubes privadas, dos o más nubes públicas, o un servidor dedicado, que puede ser tanto hardware físico como un entorno virtual conectado al menos a una nube.

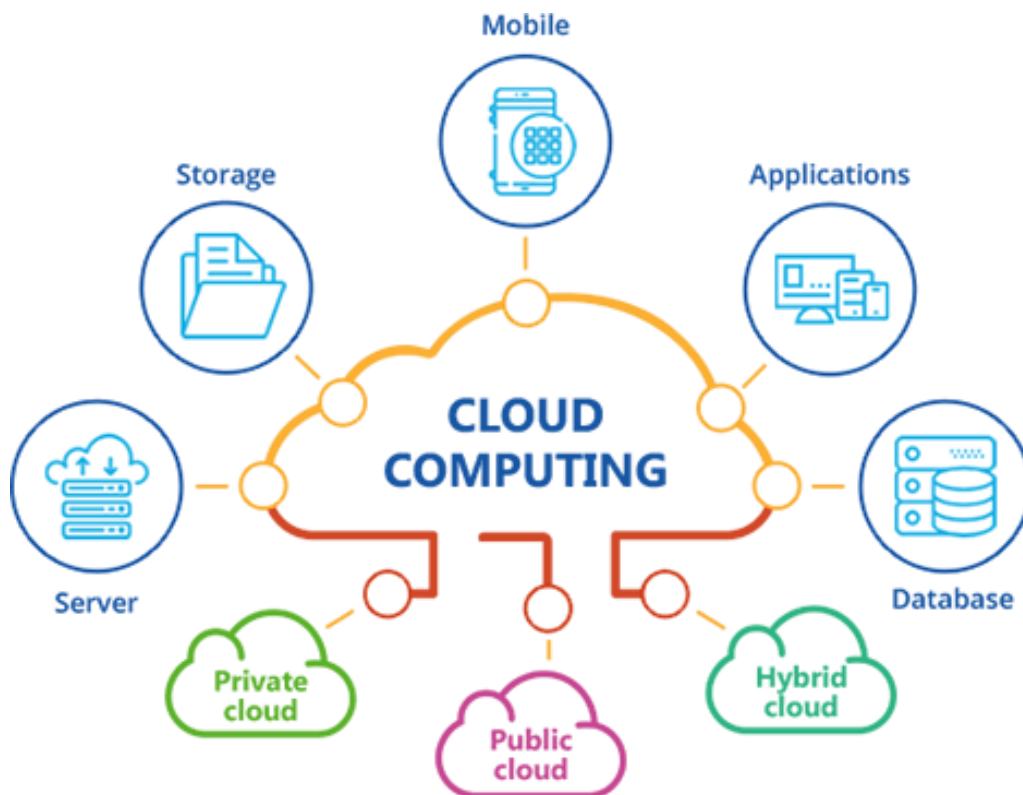


Ilustración 4. Arquitectura del Cloud Computing – Fuente: Quizlet

Entrando en el ámbito de los servicios de nube, es fundamental comprender los tres modelos principales que ofrece el cloud computing: IaaS (Infraestructura como Servicio), PaaS (Plataforma como Servicio) y SaaS (Software como Servicio). Estos términos, también conocidos como "servicios de nube" o categorías de cloud computing, delinean la forma en que una empresa utiliza la nube y el nivel de gestión que asume en sus entornos en la nube. Según la documentación oficial de Google (2023)

Tabla 1. Comparación de Servicios Cloud

Ventajas de IaaS	Inconvenientes de IaaS
Máximo nivel de control sobre la infraestructura. Escalabilidad bajo demanda. No hay un punto único de fallo para ofrecer una mayor fiabilidad. Reducción de las inversiones de capital por adelantado (por ejemplo, el pago por uso). Menos retrasos en el aprovisionamiento y malgasto de recursos. Mejora del desarrollo y tiempo de lanzamiento.	Eres responsable de la seguridad y recuperación de tus propios datos. Requiere configuración y mantenimiento prácticos. Dificulta la protección de aplicaciones antiguas en una infraestructura basada en la nube.
Ventajas de PaaS	Inconvenientes de PaaS
Acceso instantáneo a una plataforma de desarrollo completa y fácil de usar. El proveedor de servicios en la nube es responsable del mantenimiento y la seguridad de la infraestructura. Disponible a través de cualquier conexión a Internet en cualquier dispositivo. Escalabilidad bajo demanda.	La pila de aplicaciones se puede limitar a los componentes más relevantes. La dependencia de proveedores puede suponer un problema en función del proveedor de servicios en la nube. Menos control sobre las operaciones y la infraestructura general. Personalizaciones más limitadas.
Ventajas de SaaS	Inconvenientes de SaaS
Fácil de configurar y empezar a usar. El proveedor se encarga de gestionarlo y mantenerlo todo, desde el hardware hasta el software. Se puede acceder al software a través de cualquier conexión a Internet y en cualquier dispositivo.	No controlas la infraestructura ni los controles de seguridad. Problemas de integración con tus herramientas y aplicaciones actuales. La dependencia de proveedores puede suponer un problema en función del proveedor de servicios en la nube. Pocas opciones de personalización (o ninguna).

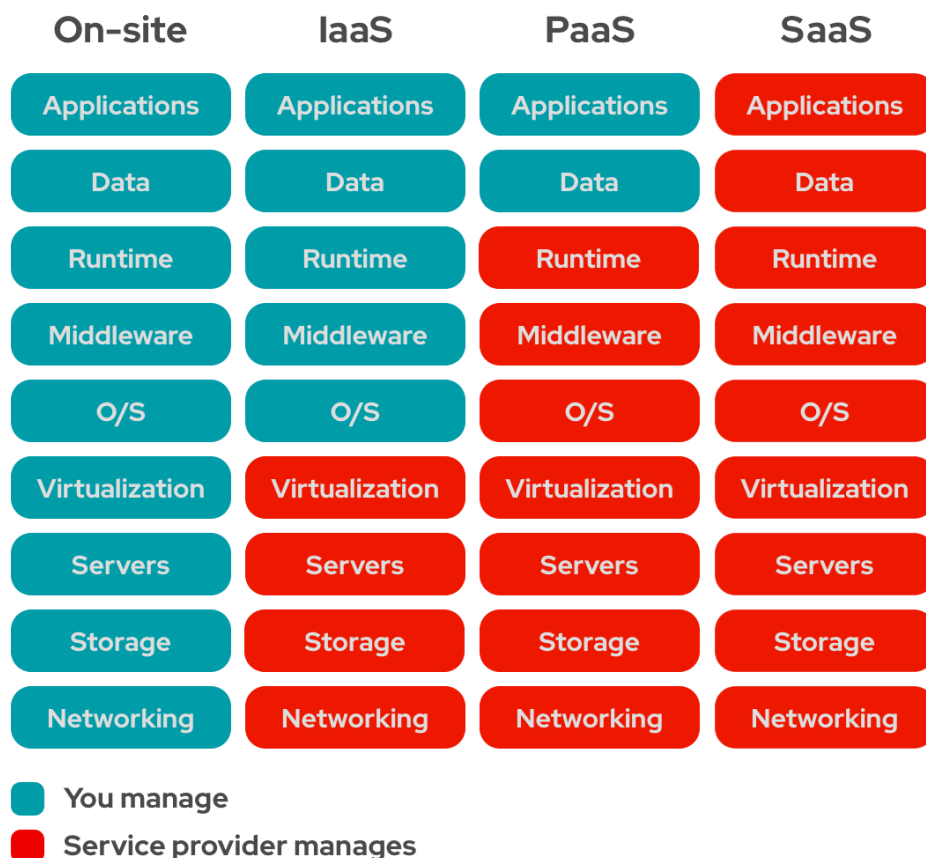


Ilustración 5. Diferencias entre servicios Cloud – Fuente: Red Hat

2.2.3 Sistema de información

Los sistemas de información, según la definición proporcionada por HubSpot (2023) en su plataforma, es que estos constituyen un conjunto integral de técnicas, herramientas y actores comprometidos en la administración y utilización de datos con el propósito de alcanzar metas empresariales. Estos sistemas desempeñan un papel crucial en la gestión de la información que genera y utiliza una organización, lo que permite contribuir significativamente a la mejora de procesos y operaciones.

Asimismo, la perspectiva de KIO (2023) sobre la tecnología de la información (TI) abarca todo lo relacionado con la conducción o facilitación del almacenamiento, procesamiento y flujo de comunicación dentro de una organización. Bajo el paraguas de la TI, se engloba la amplia gama de elementos computacionales, que incluyen software, redes, intranets, sitios web, servidores, bases de datos y telecomunicaciones.

En la actualidad, las organizaciones se ven fuertemente condicionadas por la imperante necesidad de depender de sistemas para gestionar diversas operaciones y tomar decisiones cruciales. Desde la administración del correo electrónico hasta el manejo de bases de datos y sitios web, estos sistemas han adquirido un carácter esencial, impulsando así la eficiencia y la competitividad empresarial. La interconexión eficaz de estos sistemas facilita la coordinación y la adquisición de información clave, elementos fundamentales para prosperar en entornos empresariales caracterizados por su dinamismo.

La información, según la perspectiva de KIO (2023), se origina como un flujo de datos sin procesar que representa eventos en organizaciones o en el entorno físico. Estos datos crudos son la materia prima para el procesamiento y abarcan hechos, eventos y transacciones. En este contexto se nos menciona en el portal de HubSpot (2023) que los sistemas de información se erigen con el propósito fundamental de transformar los recursos en bruto en información útil, facilitando así la toma de decisiones en una organización. Entre las funciones destacadas de un sistema de información se encuentran:

- a) **Gestionar y administrar** los datos e información que constituyen a una empresa.
- b) **Automatizar** procesos internos eliminando la necesidad de intermediarios para ejecutar operaciones específicas.
- c) **Unificar** la información empresarial a través de almacenes estandarizados para facilitar el uso y la comprensión de los datos generados.
- d) **Proporcionar** información actualizada en tiempo real y disponible para todos los colaboradores o aquellos encargados de las tomas de decisiones, agilizando y mejorando procesos y actividades de manera rápida.
- e) **Favorecer** un mejor aprovechamiento del tiempo de los empleados al posibilitar operaciones más automatizadas y sencillas, mejorando así su productividad.

En el portal de HubSpot (2023) se nos menciona que estas funciones evidencian la importancia de los sistemas de información y los alcances significativos que pueden tener al impulsar la transformación digital en las empresas. Lo destacado de los sistemas de información radica en su versatilidad, ya que no se limitan a una función única, sino que pueden ser beneficiosos para diversos sectores e industrias, adaptándose a sus necesidades y requerimientos específicos.

2.2.4 Automatización de procesos

"La automatización de procesos emplea tecnología para simplificar procesos comerciales complejos. En términos generales, se centra en la automatización de tareas, la centralización de información y la disminución de la necesidad de intervención humana" (TIBCO, 2023). Su diseño tiene como objetivo eliminar obstáculos, disminuir errores y pérdida de datos, al mismo tiempo que mejora la transparencia, la comunicación interdepartamental y la velocidad de procesamiento.

La automatización de procesos optimiza cualquier sistema al eliminar las intervenciones humanas, lo que disminuye los errores pues su principal objetivo es mejorar el proceso del flujo de trabajo, según menciona SLYDE (2022). Permitiendo acelerar la velocidad de entrega, mejorar la calidad, reducir los costos y simplificar el proceso comercial. La automatización de procesos engloba una variedad de tareas fundamentales:

- a) **Gestión y recopilación de archivos de datos:** Este aspecto se centra en la eficaz manipulación y recopilación de archivos de datos, asegurando que la información se gestione de manera óptima y esté disponible cuando sea necesaria.
- b) **Automatización de tareas repetitivas:** La automatización aborda específicamente las tareas repetitivas, liberando a los equipos de la carga de trabajos monótonos y propensos a errores, permitiendo una mayor concentración en actividades estratégicas y creativas.
- c) **Conexión e integración de fuentes de datos y servicios:** Un elemento crucial es la capacidad de conectar e integrar sin problemas diversas fuentes de datos y servicios. Esto facilita la sincronización y colaboración entre diferentes componentes del entorno empresarial, contribuyendo a un flujo de trabajo armonioso y eficiente.

2.2.5 Arquitectura de capas

Según la información proporcionada por IBM (2023), la arquitectura de tres niveles, también conocida como arquitectura de capas, se destaca como una estructura de aplicación de software ampliamente reconocida. Esta metodología organiza las aplicaciones en tres niveles lógicos y físicos distintos: el nivel de presentación o interfaz de usuario, el nivel de aplicación, donde se lleva a cabo el procesamiento de datos, y el nivel de datos, encargado de almacenar y gestionar los datos asociados con la aplicación.

El beneficio clave de esta arquitectura radica en su capacidad para permitir el desarrollo simultáneo de cada nivel por equipos de desarrolladores independientes. Además, la flexibilidad para actualizar o escalar cada nivel según sea necesario es una ventaja crucial, ya que no impacta en los demás niveles del sistema. Esto proporciona una estructura robusta y adaptable que se alinea con los requisitos cambiantes de las aplicaciones y entornos empresariales.

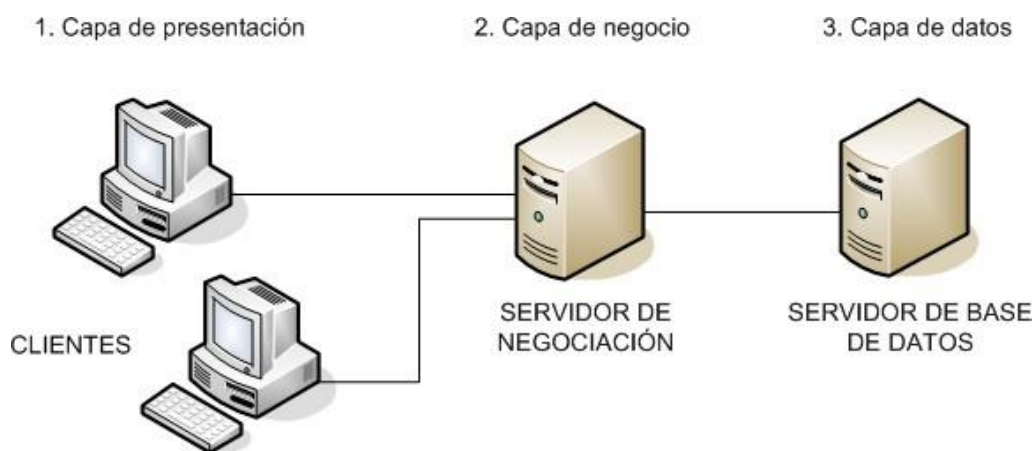


Ilustración 6. Esquema de la arquitectura de capas – Fuente: Platzi

Nivel de presentación

En concordancia con la información proporcionada por IBM (2023), el nivel o capa de presentación, detallado como la interfaz de usuario y comunicación de la aplicación, sirve como el punto de interacción entre el usuario final y la aplicación. Su propósito principal es mostrar información al usuario y capturar datos de este. Este nivel puede funcionar tanto en un navegador web como una aplicación de escritorio, presentando una interfaz gráfica de usuario. Las aplicaciones web de presentación suelen construirse con tecnologías como HTML, CSS y JavaScript, mientras que las aplicaciones de escritorio pueden emplear varios lenguajes según la plataforma o pudiendo ser también multiplataforma.

Nivel de aplicación

Por otro lado, el nivel o capa de aplicación, mencionado también por IBM (2023) como el nivel lógico o medio, representa el núcleo esencial de la aplicación. En este estrato, la información recopilada en el nivel de presentación se procesa, a veces en conjunto con otros datos del nivel de datos, mediante la aplicación de la lógica empresarial y un conjunto específico de normas empresariales. Además, el nivel de aplicación puede realizar acciones como agregar, eliminar o modificar datos en el nivel de datos. Por lo general, se desarrolla utilizando lenguajes como Python, Java, Perl, PHP o Ruby, y establece comunicación con el nivel de datos mediante llamadas API.

Nivel de datos

En la fase final, de acuerdo con la información brindada por IBM (2023), el nivel de datos, también conocido como nivel de base de datos, nivel de acceso a datos o backend, es la ubicación donde se almacena y administra la información procesada por la aplicación. Este puede adoptar la forma de un sistema relacional de gestión de bases de datos, como PostgreSQL, MySQL, MariaDB, Oracle, DB2, Informix o Microsoft SQL Server, o puede encontrarse en un servidor de bases de datos NoSQL, como Cassandra o MongoDB.

En un modelo de tres niveles, se destaca que toda la comunicación se canaliza a través del nivel de aplicación, y los niveles de presentación y datos no pueden interactuar directamente entre sí. Además, es importante señalar algunos beneficios clave que se mencionan en la documentación de IBM (2023) asociados con la implementación de esta arquitectura:

- a) **Desarrollo Rápido:** La capacidad para que diferentes equipos desarrollen simultáneamente cada nivel permite a las empresas llevar sus aplicaciones al mercado de manera más ágil, aprovechando los lenguajes y herramientas más recientes para cada nivel.
- b) **Escalabilidad Mejorada:** Cada nivel puede escalarse de forma independiente según las necesidades, brindando flexibilidad y eficiencia en términos de recursos.
- c) **Confianza Mejorada:** La independencia de cada nivel reduce la probabilidad de que una falla en un nivel afecte la disponibilidad o el rendimiento de los otros niveles.
- d) **Seguridad Mejorada:** La imposibilidad de comunicación directa entre los niveles de presentación y datos permite que un nivel de aplicación bien diseñado funcione como una especie de firewall interno, proporcionando una capa adicional de seguridad contra ataques maliciosos como inyecciones SQL y otras vulnerabilidades.

2.2.6 Programación Orientada a Objetos

Según la documentación de Mozilla Developer (2023), un paradigma de programación se define como un método o conjunto de principios en el desarrollo de software. Los paradigmas son conjuntos sistemáticos de métodos aplicables en todos los niveles del diseño de programas para abordar problemas computacionales.

En este sentido, la Universidad Europea (2022) explica que el paradigma de la programación orientada a objetos se basa en la creación de un modelo del problema en los programas, centrándose en disminuir errores y fomentar la reutilización del código. Se enfoca en los objetos que los programadores deben manipular, más que en la lógica necesaria para dicha manipulación, definiendo un objeto como un campo de datos con atributos y comportamientos únicos.

La característica principal, según la Universidad Europea (2022), es el soporte a objetos que tienen un tipo o clase asociado, pudiendo heredar atributos de una clase superior o superclase. Este enfoque se utiliza en programas grandes y complejos que requieren actualizaciones periódicas. Actualmente, existen diversos lenguajes de programación orientados a objetos como C++, Objective C, Java, Ruby, Visual Basic, Visual C Sharp, Simula, Perl, TypeScript, Smalltalk, PHP o Python. C++ y Java destacan como los más utilizados en este paradigma.

Las principales ventajas que nos ofrece el uso del paradigma de programación orientada a objetos son las siguientes:

- a) **Reutilización del código heredado:** Diseñar clases de manera adecuada permite la reutilización del código en distintas partes del programa y en proyectos diferentes. La herencia ahorra tiempo al crear una clase genérica y definir subclasses que heredarán sus rasgos, evitando la necesidad de reescribir funciones. Además, los cambios aplicados a la clase afectan automáticamente a todas las subclasses (Universidad Europea, 2022).
- b) **Mayor modificabilidad:** La programación orientada a objetos facilita la adición, modificación o eliminación de objetos o funciones, lo que simplifica las actualizaciones de los programas. Esta flexibilidad ahorra tiempo y esfuerzo para los programadores al realizar cambios en el código (Universidad Europea, 2022).
- c) **Facilidad para detectar errores en el código:** Gracias a la encapsulación, los objetos son autónomos, lo que simplifica la detección de errores. No es necesario revisar línea por línea del código, ya que los objetos pueden abstraer problemas y facilitar la identificación de errores cuando algo no funciona correctamente (Universidad Europea, 2022).
- d) **Modularidad:** La programación orientada a objetos fomenta la modularidad, permitiendo que un equipo de desarrollo trabaje en múltiples objetos simultáneamente. Esto minimiza las posibilidades de duplicar funcionalidades y facilita la división de

problemas en partes más pequeñas que pueden probarse de manera independiente (Universidad Europea, 2022).

- e) **Flexibilidad gracias al polimorfismo:** El polimorfismo en la programación orientada a objetos permite que una función cambie de forma para adaptarse a cualquier clase donde se encuentre. Esta característica ahorra tiempo de programación y aporta versatilidad al sistema (Universidad Europea, 2022).

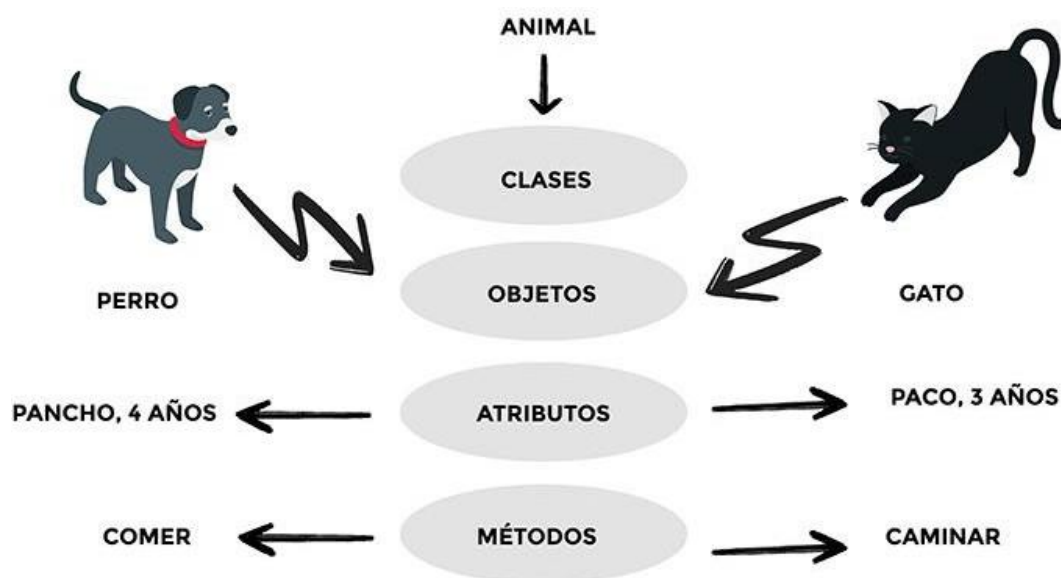


Ilustración 7. Ejemplo de programación orientada a objetos – Fuente: Profile

2.3 Herramientas de desarrollo

2.3.1 Visual Studio

La piedra principal herramienta de trabajo de este proyecto es Microsoft Visual Studio, una herramienta integral que despliega todo el ciclo de desarrollo en un solo lugar, según lo indica la documentación oficial de Microsoft (2023). Visual Studio se presenta como un Entorno de Desarrollo Integrado (IDE) completo que brinda la capacidad de escribir, editar, depurar y compilar el código, además de implementar la aplicación final. Más allá de las funciones básicas de edición y depuración, Visual Studio engloba una variedad de herramientas esenciales, como compiladores, funciones de finalización de código, control de código fuente, extensiones y otras características avanzadas, con el objetivo de potenciar cada fase del proceso de desarrollo de software.

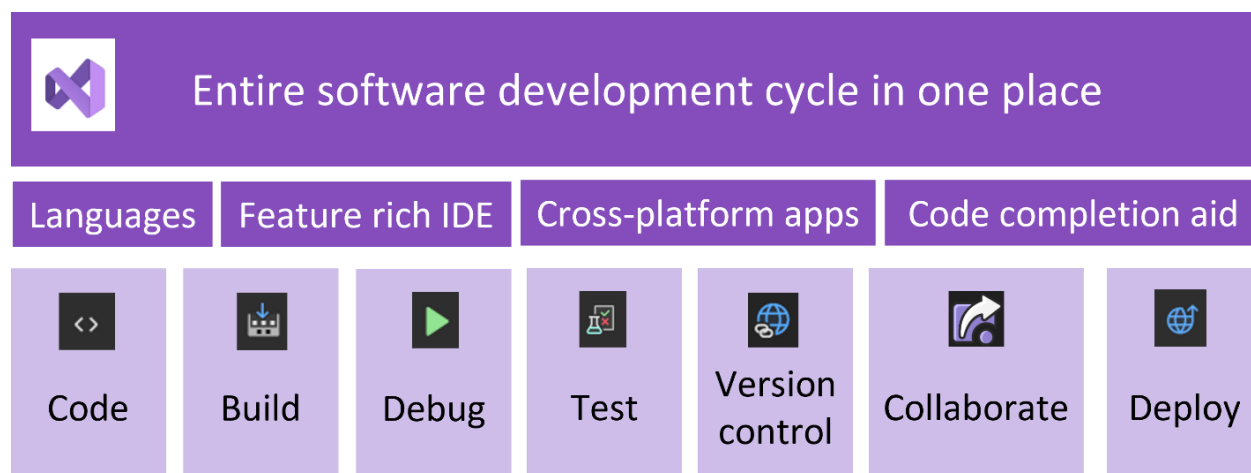


Ilustración 8. Arquitectura de Visual Studio – Fuente: Microsoft

Visual Studio proporciona a los desarrolladores un entorno de desarrollo enriquecido para desarrollar código de alta calidad de forma eficaz y colaborativa, y algunas de las principales ventajas que nos puede brindar hacer uso de esta herramienta son:

- Instalador basado en cargas de trabajo: instale solo lo que necesita.
- Herramientas y características de codificación eficaces: todo lo que necesita para compilar sus aplicaciones en un solo lugar.
- Compatibilidad con varios lenguajes: código en C++, C#, JavaScript, TypeScript, Python, etc.
- Desarrollo multiplataforma: compilación de aplicaciones para cualquier plataforma.
- Integración del control de versiones: colaboración en el código con compañeros de equipo.

- Desarrollo asistido por inteligencia artificial: escritura de código de forma más eficaz con ayuda de la inteligencia artificial

Según la documentación de Microsoft (2023), la versatilidad de Visual Studio se refleja en su capacidad para llevar a los desarrolladores desde la creación de su primer programa "Hola mundo" hasta el desarrollo e implementación de aplicaciones avanzadas. Entre las diversas características y lenguajes que admite, Visual Studio permite compilar, depurar y probar aplicaciones en .NET y C++, editar páginas de ASP.NET mediante la vista del diseñador web, desarrollar aplicaciones multiplataforma para móviles y escritorio con .NET, así como crear interfaces de usuario web responsivas en C#.

2.3.2 Microsoft SQL Server

En el ámbito de la gestión de datos y desarrollo de procedimientos almacenados para el aplicativo, se ha decidido emplear Microsoft SQL Server, un sistema de administración de bases de datos relacionales (RDBMS) que, según la documentación oficial de Microsoft (2023), se destaca por su confiabilidad y eficiencia. Las aplicaciones y herramientas se conectan a instancias o bases de datos de SQL Server y se comunican mediante Transact-SQL (T-SQL).

La elección de Microsoft SQL Server se fundamenta en su sólida reputación como sistema de administración de bases de datos. Como señala la documentación de Microsoft (2023), SQL Server ofrece características avanzadas de seguridad, rendimiento optimizado y una escalabilidad que se ajusta a las necesidades de crecimiento de las aplicaciones empresariales. La integración nativa con el lenguaje Transact-SQL proporciona un entorno robusto para la manipulación y administración de datos, facilitando la creación de procedimientos almacenados y asegurando una comunicación eficiente entre la aplicación y la base de datos.

Además, se hará uso de SQL Server Management Studio (SSMS), herramienta integrada que, según la documentación de Microsoft (2023), facilita la gestión de cualquier infraestructura de SQL, desde SQL Server hasta Azure SQL Database. SSMS ofrece herramientas para configurar, supervisar y administrar instancias de SQL Server y bases de datos. Su funcionalidad abarca desde la implementación hasta la actualización de los componentes de nivel de datos utilizados por las aplicaciones, incluyendo la capacidad de compilar consultas y scripts.

2.2.3 .NET MAUI

En el proceso de desarrollo y diseño del aplicativo, se optará por emplear el marco de trabajo proporcionado por Microsoft (2023), conocido como .NET Multi-platform App UI (.NET MAUI). Este marco o framework, según su documentación oficial, se presenta como una solución multiplataforma para la creación de aplicaciones móviles y de escritorio nativas utilizando C# y XAML. La versatilidad de .NET MAUI permite el desarrollo de aplicaciones que pueden ejecutarse en plataformas como Android, iOS, macOS y Windows, todo ello desde una única base de código compartida.

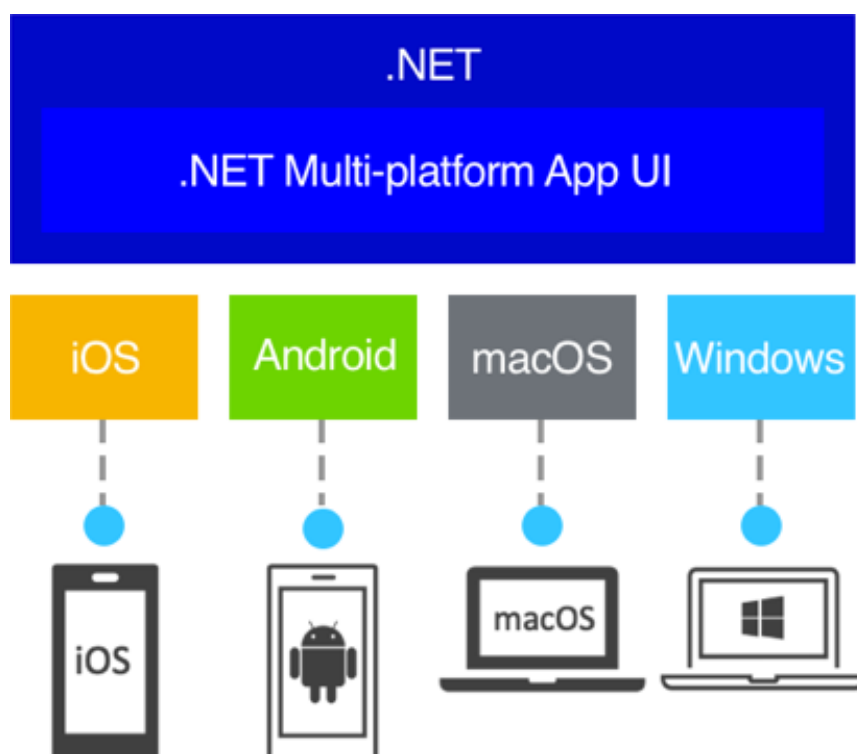


Ilustración 9. Arquitectura de .NET MAUI – Fuente: Microsoft

Con .NET MAUI, como destaca Microsoft (2023), se hace posible el desarrollo de aplicaciones multiplataforma a través de un único proyecto, ofreciendo la flexibilidad de agregar recursos y código fuente específicos de cada plataforma según las necesidades particulares. Uno de los objetivos fundamentales de .NET MAUI es permitir la implementación de gran parte de la lógica de la aplicación y el diseño de la interfaz de usuario en una única base de código, proporcionando eficiencia y coherencia al proceso de desarrollo.

La culminación de estas herramientas estratégicas se reflejará en el producto final: un tótem interactivo que transformará de manera significativa el proceso actual de registro manual en la línea de ensamble. La implementación de tecnologías clave como .NET, Visual Studio y SQL Server establece una base sólida y eficiente para el aplicativo. La infraestructura creada en Visual Studio, respaldada por el potente motor de Microsoft SQL Server, garantiza una gestión robusta y segura de los datos. La adopción de .NET MAUI amplía la versatilidad al posibilitar una experiencia multiplataforma, consolidando así el aplicativo en un único proyecto.

Este enfoque integral, respaldado por las tecnologías de vanguardia de Microsoft, redefine la gestión de la producción en la línea de ensamble, marcando un cambio sustancial hacia la eficiencia y la precisión mediante la sustitución física de la pizarra por un innovador tótem interactivo.

2.2.4 Tkinter

Según se menciona en el portal de KeepCoding (2023) Tkinter es una biblioteca de programación en Python diseñada para facilitar la creación y desarrollo de aplicaciones de escritorio con interfaces gráficas. Este paquete, que es el estándar de Python para interactuar con Tk, ofrece una herramienta única y versátil para el desarrollo de interfaces gráficas de usuario (GUI) que funciona de manera consistente en diversos sistemas operativos, como Windows, Mac OS y Linux, según la documentación oficial de Python.

De acuerdo con las especificaciones de Python (2024), Tk se autodenomina como el único kit de herramientas que abarca el desarrollo de GUI y es compatible con sistemas operativos múltiples. Este enfoque significa que los desarrolladores no necesitan sumergirse en las rutinas de bajo nivel de los sistemas operativos individuales, lo que simplifica considerablemente el control gráfico de las interfaces de usuario en diferentes plataformas.

Tkinter está especialmente diseñado y optimizado para su uso con lenguajes de alto nivel y dinámicos, como Tcl, Ruby o Perl, brindando flexibilidad y eficiencia en el desarrollo de aplicaciones. Al comprender y emplear Tkinter, los desarrolladores pueden beneficiarse de una mayor facilidad y coherencia en la creación de sus aplicaciones, ya que la biblioteca proporciona una capa de abstracción que simplifica la interacción con la interfaz gráfica, independientemente del sistema operativo subyacente lo que lo hace una biblioteca muy atractiva y popular hoy en día para el desarrollo de interfaces graficas con Python.

CAPITULO III: DESARROLLO

A continuación, se exponen los procedimientos planificados en el cronograma de actividades establecido al inicio del proyecto. Estas actividades están organizadas de manera cronológica y se documentan durante su ejecución para abordar todos los detalles pertinentes. Cada fase incluye una descripción detallada de su ejecución y las herramientas empleadas para alcanzar los objetivos establecidos.

3.1 Análisis

Durante la fase inicial del análisis para el desarrollo del aplicativo, se llevaron a cabo reuniones diarias, con una duración aproximada de una hora, con el propósito de comprender exhaustivamente los requisitos del proyecto. Durante este período, se establecieron directrices clave, se definieron los objetivos esenciales para iniciar el desarrollo y se delineó claramente lo necesario para satisfacer las necesidades de INOAC Sistemas Exteriores. En estas sesiones, se centró en la identificación detallada de los requisitos funcionales y no funcionales, así como en el establecimiento de metas cuantificables que guiarían el progreso a lo largo del tiempo.

Volviendo al objetivo inicial del aplicativo, está claro que el origen principal es empujar a la empresa de INOAC Sistemas Exteriores S.A. de C.V. a una nueva etapa de modernización en su planta retomando el concepto de que su planeación diaria de operación de las líneas de ensamble dependía al 100% de un pizarrón y plumones, por lo que el aplicativo debía cumplir con todos los aspectos que el cliente requiere para el funcionamiento correcto de la producción de ensamble de piezas.

En este contexto, el aplicativo busca sustituir eficazmente el uso de la pizarra tradicional, ubicada estratégicamente en el centro de las líneas de ensamble. La meta es transformar este método manual y limitado en un proceso digitalizado y eficiente. Al reemplazar la pizarra con un sistema interactivo y basado en tecnología, se busca eliminar las limitaciones asociadas con la gestión manual, brindando a los supervisores y operadores una herramienta integral que agilice la toma de decisiones y optimice la productividad en la planta. Este cambio de paradigma representa un paso audaz hacia la modernización en la gestión de la producción.

A lo largo del desarrollo continuo del aplicativo, se llevaron a cabo reuniones diarias más breves para revisar el estado actual del trabajo, abordar dudas y, ocasionalmente, recibir sugerencias del cliente (generalmente en intervalos mensuales o después de revisiones con el cliente). Estas sesiones sirvieron para proporcionar una visión clara del progreso, establecer referencias sobre el estado actual del proyecto, definir los objetivos siguientes y estimar el tiempo necesario para alcanzarlos.

La siguiente tabla presenta una lista detallada de atributos de calidad, también conocidos como requisitos no funcionales, que se incorporaron en el diseño y desarrollo del aplicativo:

Tabla 2. Atributos de calidad para el aplicativo.

Atributo o requerimiento	Descripción	Cantidad
Desplegabilidad	El aplicativo debe ser capaz de desplegarse de manera sencilla en un entorno incluso durante producción.	Instalación de nuevas versiones de manera rápida (menos de 1 hora).
Disponibilidad y tiempo de respuesta	El aplicativo debe ser capaz de trabajar incluso en operaciones de alta demanda y presentar información en tiempo real.	Cada segundo.
Plataforma	El aplicativo debe ser compatible con Android al menos en la primera versión.	Android 8 o superior, al ser desarrollada en .NET MAUI tenemos la flexibilidad de trabajar para iOS.
Usabilidad y accesibilidad	El aplicativo debe contar una interfaz de usuario intuitiva y funcionalmente táctil.	El aplicativo se implementará en un tótem interactivo por lo que debe ser fácil de usar.
Seguridad	El aplicativo debe contar con medidas de seguridad sólidas, protegiendo información confidencial de los clientes.	El aplicativo estará trabajando con datos delicados del sistema de trazabilidad por lo que debe ser seguro.
Escalabilidad	El aplicativo debe ser capaz de escalar en cuanto a funcionalidades futuras que requiera el cliente.	Siendo un sistema que potencialmente puede crecer en cuando a lógica y seguridad.

3.2 Diseño

Durante la fase de diseño, especialmente en lo relacionado con la interfaz de usuario (UI) y la experiencia del usuario (UX), se emplearon herramientas específicas como Excalidraw y Miro. Estas plataformas facilitaron la creación de wireframes y diagramas de flujo, permitiendo una representación visual clara de las expectativas visuales del aplicativo y la definición de su funcionalidad. Ambas aplicaciones ofrecen capacidades para:

- Diagramas UML y de flujo
- Wireframes o interfaces de usuario de modelado ágil
- Modelos y esquemas

La siguiente figura presenta un boceto inicial que ilustra las expectativas del aplicativo en términos de los requisitos básicos funcionales discutidos desde el inicio del proyecto. Este boceto se generó mediante una lluvia de ideas y teniendo en cuenta que la aplicación al inicio iba a trabajar y desarrollarse usando la biblioteca para generar interfaces graficas de Python llamada Tkinter, dando forma a una propuesta de diseño:

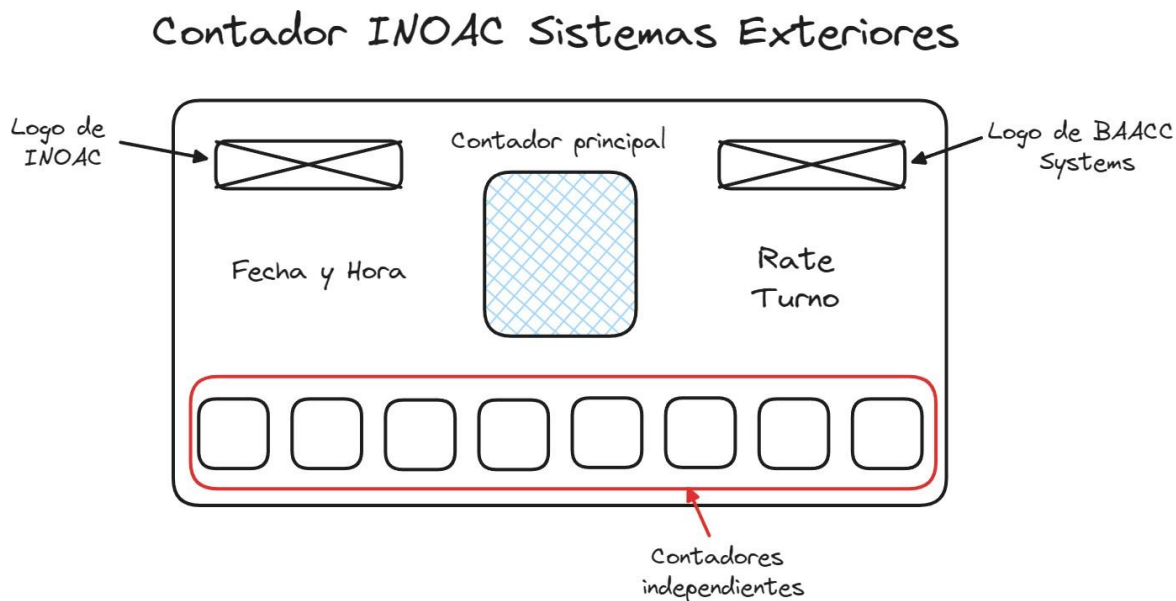


Ilustración 10. Wireframe dashboard principal, primera versión

En el wireframe anterior se puede observar que se esperaba que la aplicación fuera al inicio una aplicación informativa y meramente visual, con la funcionalidad de mostrar a los operadores de las líneas de ensamble la producción que se está trabajando en ese momento, los contadores independientes (8) hacen referencia a las horas que contiene cada turno en la planta de producción y el contador principal se vincula con el contador de la hora que se trabaja en ese mismo momento.

Conforme fueron avanzando los días la lógica de la aplicación fue cambiando y aumentando, pasando de una aplicación meramente informativa a una plataforma o sistema completo en donde se definió lo que se esperaba visualmente del sistema de registro de paros en producción.

Teniendo en cuenta esto, una producción puede entrar en paro si es que existen ciertos motivos que lo avalen, como que se haya terminado material para continuar con la producción, falla en alguna máquina, falta de personal, etc. Esto implica que se tenga que conocer todos estos datos, por lo que también fue necesario diseñar desde cero un sistema que pueda registrar los paros en las líneas de ensamble.

En la siguiente figura se puede observar la primera propuesta de diseño para el sistema de registro de paros en la línea de ensamble, como solución se propuso generar una ventana modal en el aplicativo, la cual tenía como presentación una bitácora de los paros anteriores que ya se habían registrado en la línea de ensamble:

The wireframe illustrates a modal window titled "REGISTRO DE PAROS" (Stop Registration). The window contains a table with two columns: "Paro por fallo en maquina" (Stop due to machine failure) and "11:00am - 12:00pm Finalizado" (11:00am - 12:00pm Completed). Below the table is a "NUEVO" (New) button.

Paro por fallo en maquina	11:00am - 12:00pm Finalizado

NUEVO

Ilustración 11. Wireframe del sistema de registro de paros, visualización principal

La pantalla anterior cuenta con un botón para poder generar un nuevo paro en la línea, en la siguiente figura se muestra la solución para generar esta pantalla, la cual fue una ventana modal que permita recibir valores como el área solicitada, motivo y tiempo que va a durar el paro.

REGISTRO DE PAROS

ÁREA SOLICITADA ▼

MOTIVO ▼

TIEMPO

COMENZAR

Ilustración 12. Wireframe del registro de un paro recibiendo parámetros

La figura anterior abre paso a una siguiente pantalla estática que solo muestra información de que un paro está en ejecución, mostrando el tiempo que lleva este paro, indispensable para los supervisores, el motivo por el cual se inició este paro y el área que se solicitó para poder solucionar este paro, como se ve en la siguiente figura:

REGISTRO DE PAROS

LA LINE LLEVA 00:00:05 EN PARO

EL MOTIVO ES: _____

ÁREA SOLICITADA: _____

DETERENER

Ilustración 13. Wireframe de pantalla de paro activo en producción

Todas las figuras anteriores representaron el primer boceto general de cómo debería funcionar la aplicación de registro de paros, una vez terminado el producto se realizó una primera versión la cual fue revisada junto con el cliente para poder recibir feedback de mejoras. El cliente fue directo con nosotros indicando que el primer boceto o wireframes de la aplicación estaba bien, pero e quería sustituir unos pizarrones enormes en medio de las líneas de producción por lo que el nuevo diseño no cumplía demasiado con estas características por lo que el objetivo del diseño tuvo que cambiar un poco.

Trabajando con el equipo de desarrollo y el asesor del proyecto se definió que el proyecto debía tener un giro multiplataforma y móvil principalmente ya que se optó por incorporar un tótem interactivo y estilo publicitario para poder sustituir los pizarrones que usan en las líneas de ensamble, cambiando también un poco el diseño pasando de ser una pantalla horizontal a una pantalla en formato vertical, como se puede observar en la siguiente figura la cual representa este nuevo rediseño.

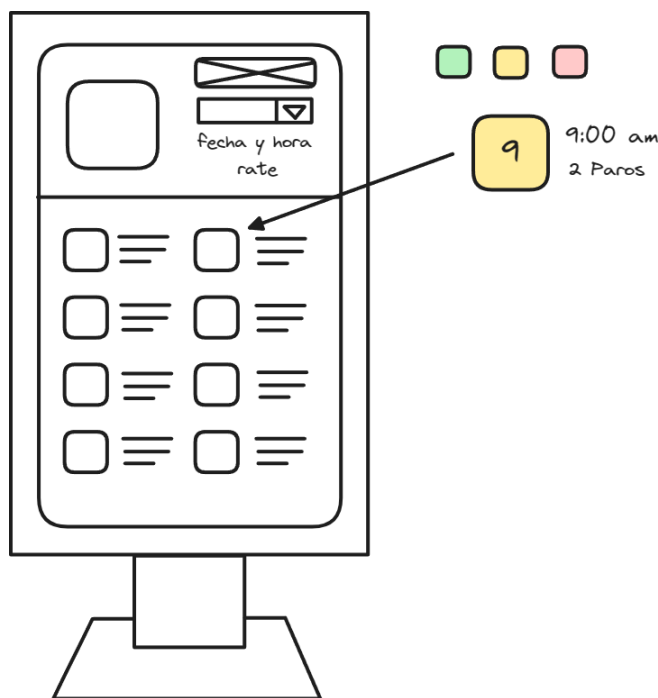


Ilustración 14. Wireframe nuevo diseño vertical del aplicativo sobre tótem interactivo

Las diferencias más notables en cuanto al diseño anterior fue el combo box que se colocó en la parte superior derecha permitiendo que el tótem pueda servir como una estación móvil para todas las líneas de ensamble, esto únicamente cambiando entre líneas con un combo box, si se quieren mover de línea 3 a línea 4, lo pueden hacer moviendo solo un combo box y los contadores

individuales pasaran a indicar el nivel de producción de manera visual, si se está muy por debajo de la meta, justo en la meta o por encima de la meta de ensamble usando indicadores de colores.

El apartado de registro de paros queda de la misma manera que en el diseño anterior pero ahora solo los paros van a poder ser generados por los supervisores y también si se quiere detener un paro únicamente podrá hacerlo un supervisor, la siguiente figura muestra como interactúa este sistema de autenticación.



Ilustración 15. Wireframe de pantalla de autenticación para supervisores

En la anterior figura se muestra una pantalla de autenticación para los supervisores, requiriendo que se ingresen dos datos importantes, el IDR (Identificador De Responsable) y su contraseña, estos datos vienen del sistema de trazabilidad que ya trabajaba anteriormente en INOAC Sistemas Exteriores, este sistema desarrollado por BAACC Systems S.A. de C.V. es una aplicación web desarrollada tiene la interconexión con el aplicativo de trazabilidad que se trabajó.

Con este apartado se finalizó la etapa de diseño del aplicativo tomando en cuenta todos los requerimientos y necesidades del cliente para cubrir con los aspectos de generar un sistema seguro, intuitivo y completo para poder estar presente durante el proceso de ensamble en las líneas de producto terminado.

3.3 Codificación

Durante la fase de desarrollo o codificación se estuvo trabajando de primera mano con el lenguaje de programación Python realizando código para poder generar los frames o ventanas desde cero, se estuvo investigando a cerca de las bibliotecas de Python para poder generar interfaces gráficas y entre opciones como PyQt, Kivy y demás, se decidió por usar la más popular de todas, Tkinter. Tkinter trabaja generando componentes desde cero como ventanas, diálogos, botones, etc.

Se trabajo de primera instancia con el dashboard principal, generando los ocho contadores individuales, el contador principal y dándole forma a los demás componentes que componen esta pantalla, el principal reto al que se enfrentó en cuanto al apartado del código fue entender que está misma tenía que refrescarse o actualizarse cada cierto tiempo, que al final termino definiéndose que sea en tiempo real o cada segundo, permitiendo una mayor precisión a la hora de visualizar los datos.

En el contexto del desarrollo de aplicaciones, es esencial comprender el papel del hilo principal. Este hilo es como el director principal de una obra teatral, encargado de coordinar las tareas fundamentales de la aplicación y de manejar la interacción con la interfaz de usuario. Cualquier aplicación se ejecuta a través de este hilo principal, que asegura la ejecución suave de tareas esenciales.

Cuando consideramos la necesidad de actualizar periódicamente la interfaz de usuario, como en el caso de refrescar la ventana cada segundo, es crucial evitar bloquear el hilo principal. Aquí es donde entran en juego los métodos asíncronos. Al utilizar métodos asíncronos, podemos realizar operaciones que pueden llevar tiempo, como actualizaciones periódicas, sin detener el flujo principal de la aplicación. Esto permite mantener una interfaz de usuario receptiva y ágil, ya que las tareas asíncronas se ejecutan de manera independiente al hilo principal. Así, logramos una experiencia de usuario más fluida y evitamos que la aplicación parezca congelada durante operaciones que podrían ser intensivas en recursos.

Tras la configuración de la interfaz gráfica para el dashboard principal, se procedió al desarrollo del procedimiento almacenado esencial para el funcionamiento del sistema. Este procedimiento almacenado, diseñado para trabajar en conjunción con Microsoft SQL Server y SQL Server Management Studio (SSMS), desempeña un papel crucial en la manipulación eficiente de datos.

En el proceso de desarrollo, se llevaron a cabo diversas consultas SQL para definir las operaciones específicas que el procedimiento almacenado debe realizar. Esto incluyó consultas para la extracción de datos relevantes, la organización de la información de acuerdo con los requisitos del dashboard y la gestión de la visualización de los datos en la interfaz gráfica, algunos datos relevantes que se pueden mencionar son el conteo de piezas por hora y por turno, todo esto

siendo filtrado por la línea de ensamble, que es un parámetro que tiene que recibir el procedimiento almacenado desde el mismo aplicativo.

Todos los datos que genera el procedimiento almacenado son consumidos por el aplicativo haciendo uso de un ODBC (Open Database Connectivity). ODBC actúa como un puente esencial que facilita la comunicación entre la aplicación y la base de datos subyacente. Gracias a ODBC, el aplicativo puede establecer una conexión eficiente con la base de datos, ejecutar el procedimiento almacenado y recuperar los resultados de manera estructurada.

La ventaja clave de utilizar ODBC radica en su capacidad para proporcionar una interfaz estándar de acceso a bases de datos, permitiendo que la aplicación sea independiente del sistema de gestión de bases de datos subyacente. Esto significa que el aplicativo puede mantenerse flexible y compatible con diferentes bases de datos sin requerir modificaciones extensas en su lógica interna. Además, ODBC facilita la gestión de la conectividad, permitiendo la configuración de parámetros esenciales, como la cadena de conexión y las credenciales de acceso, de manera centralizada. Esto simplifica el proceso de mantenimiento y garantiza una mayor seguridad en el acceso a los datos.

Trabajando con el asesor del proyecto y una vez conectado el aplicativo con el dashboard principal y tomando en cuenta que para este momento la decisión era cambiar de giro el desarrollo de la aplicación empezó a trabajarse una nueva interfaz haciendo uso de .NET MAUI, por lo que hasta este punto empezó a usarse en entorno de desarrollo integrado Visual Studio, de primera instancia creando un proyecto de tipo aplicación multiplataforma con MAUI.

La estructura de carpetas dentro del proyecto se empezó a trabajar haciendo uso de la arquitectura de tres capas que se decidió trabajar durante la fase de análisis, la estructura del aplicativo se muestra en la siguiente figura.

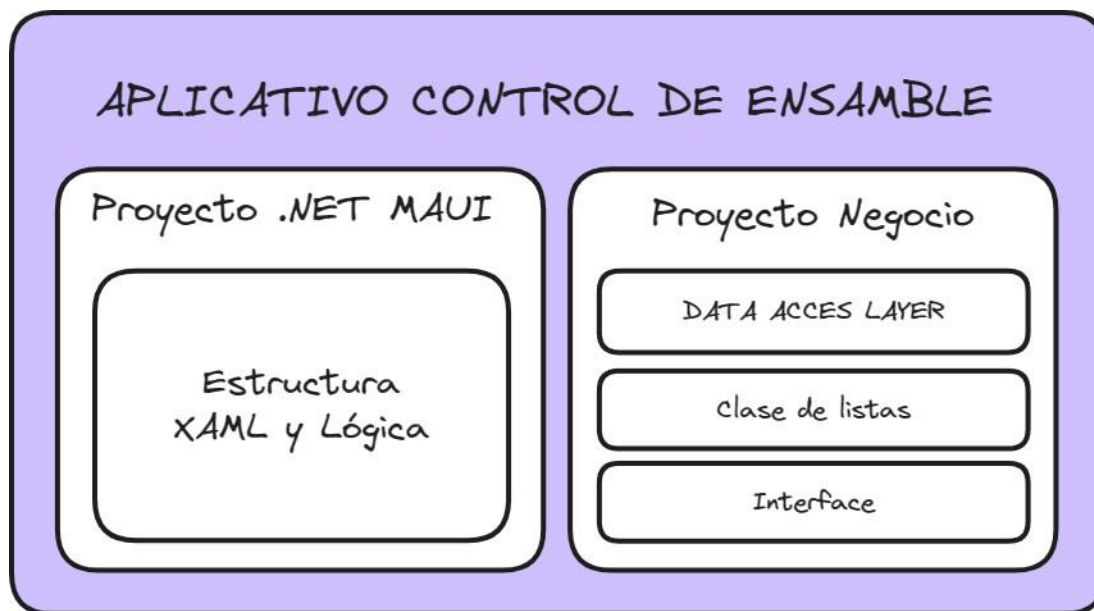


Ilustración 16. Estructura del proyecto en .NET MAUI

En la figura anterior, se presenta una arquitectura de aplicación dividida en dos proyectos principales. El primero, que alberga la estructura XAML y la lógica de programación, es fundamental para el desarrollo de la interfaz de usuario y está diseñado para trabajar con .NET MAUI. La estructura XAML (Extensible Application Markup Language) es un lenguaje declarativo que define la apariencia y comportamiento de la interfaz de usuario. Es particularmente relevante en el contexto de .NET MAUI, que es un framework multiplataforma para la creación de aplicaciones móviles modernas.

En este proyecto, la lógica de programación se encarga de la manipulación de datos y la interacción con la interfaz de usuario. Aquí es donde se implementan los eventos, la lógica de negocio y la manipulación de la interfaz para proporcionar una experiencia de usuario coherente, incluyendo el manejo del método asíncrono de la ventana la cual se sigue manteniendo conforme a la esencia de la primera versión del aplicativo.

Por otro lado, se presenta un segundo proyecto que constituye la capa de Negocio. Este proyecto actúa como intermediario entre la aplicación y la base de datos. Su estructura incluye:

Clase de Lista: Esta clase sirve como una estructura de datos que contiene getters y setters para los campos relevantes. Es utilizada para almacenar y transportar datos entre las capas de la aplicación. Se llena conforme a la información recuperada del procedimiento almacenado.

Data Access Layer (DAL): El Data Access Layer es responsable de la conexión con la base de datos. Utiliza el procedimiento almacenado para recuperar datos y llena la Clase de Lista

con la información obtenida. Esta capa se encarga de la gestión eficiente de la conexión y la ejecución de consultas en la base de datos, también encargándose de capturar los errores en cuanto a la conexión con la base de datos.

Interface: La interfaz actúa como un punto de conexión entre el DAL y el proyecto de .NET MAUI. Al crear una instancia de un objeto a través de una interfaz, se establece una capa de abstracción que facilita la modularidad y el intercambio de implementaciones. Este enfoque permite que el proyecto de .NET MAUI interactúe con el DAL sin depender directamente de sus detalles internos.

Esta arquitectura sigue un diseño de capas que separa claramente las responsabilidades entre la interfaz de usuario, la lógica de programación, la capa de negocio y la interacción con la base de datos. La estructura XAML y .NET MAUI se encargan de la experiencia del usuario, mientras que el proyecto de la capa de Negocio gestiona la conexión a la base de datos y la manipulación de datos a través de procedimientos almacenados. La interfaz proporciona un nivel de abstracción para facilitar la integración entre estas capas.

Pasando al desarrollo del sistema de trazabilidad, en la siguiente figura se muestra un esquema básico de las tablas en la base de datos de ensamble de INOAC Sistemas Exteriores y que por motivos de privacidad de la información a la empresa solo se muestra una referencia básica. La base de datos de ensamble tiene una tabla principal o maestra que hace referencia a una producción de ensamble, a esta producción se le agrego el sistema de paros por medio de una tabla para poder registrarlos, teniendo como referencia para un registro la misma producción y un motivo que se sabrá de un catálogo de motivos de paro que tiene que funcionar solo para consulta de los motivos.

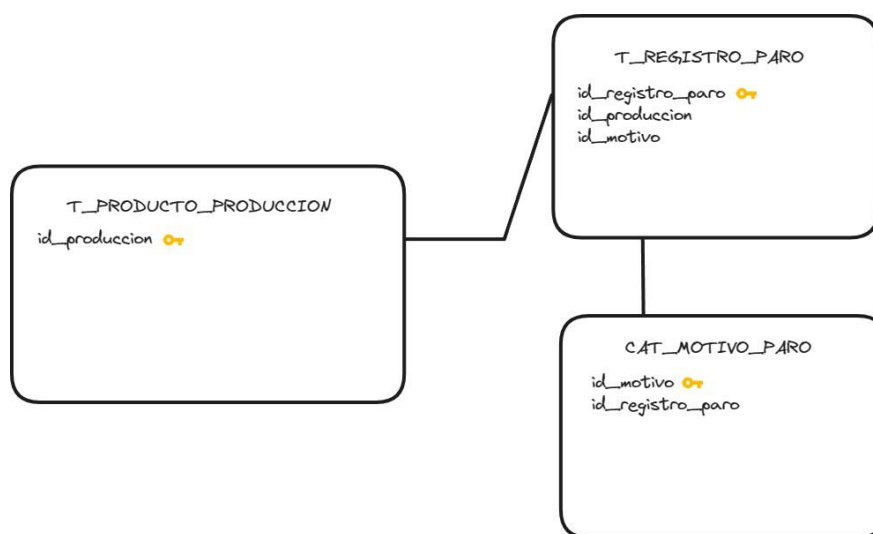


Ilustración 17. Estructura básica de las tablas añadidas a la base de datos para el registro de paros

En el sistema de registro de paros, se implementó una interfaz de usuario que sigue las pautas establecidas durante la fase de diseño. Al interactuar con un contador individual asociado a la hora en turno, se activa una ventana emergente al presionar o hacer clic. Esta ventana muestra una bitácora general de los paros registrados en esa hora específica. En caso de no existir información, se presenta un mensaje indicando la ausencia de paros en ese periodo.

Dentro de la primera ventana, se incorpora un botón destinado a generar un nuevo paro. Al hacer clic en este botón, se despliega un formulario diseñado para autenticar al usuario, asegurando que solo un supervisor de producción autorizado pueda iniciar un paro en las líneas de ensamble. Este formulario incluye campos específicos para la autenticación, como nombre de usuario y contraseña los cuales están conectados al sistema de trazabilidad de BAACC Systems por lo que los usuarios registrados como supervisores en el sistema solo pueden autenticarse en este caso con su IDR (Identificador De Responsable).

Si la autenticación es exitosa, se permite el acceso al siguiente formulario. En este formulario, el supervisor tiene la capacidad de registrar detalles relacionados con el paro. Los campos incluyen información como el motivo del paro, el área que necesita intervención para resolver el problema y, en caso necesario, un campo de tiempo establecido como referencia. Estos datos proporcionan un registro detallado del paro y facilitan la toma de decisiones para resolver los problemas identificados.

Finalmente, después de registrar el paro, se presenta una última ventana emergente informativa. Esta ventana notifica que la máquina está en estado de paro, y proporciona información adicional, como el área notificada para intervenir y el tiempo transcurrido desde el inicio del paro. Este enfoque ofrece una visión clara y detallada del estado actual de la producción.

3.4 Pruebas

Durante la fase de pruebas, se estableció una estrecha colaboración con el cliente, INOAC Sistemas Exteriores, presentando versiones iterativas del aplicativo a medida que se realizaban cambios significativos. La primera versión del sistema, desarrollada en Python, fue probada en dispositivos Raspberry Pi en la planta de Querétaro. Estas pruebas iniciales permitieron recibir valiosos comentarios del cliente, especialmente en cuanto al aspecto visual y la funcionalidad del sistema. Se realizaron ajustes, como la introducción de indicadores más visuales y la adaptación de la interfaz para garantizar una experiencia óptima en dispositivos con sistemas operativos distintos.

En reuniones subsiguientes con el cliente, se consideró una transición a una arquitectura multiplataforma. En este contexto, se decidió desarrollar una aplicación que pudiera reemplazar las pizarras utilizadas en las líneas de ensamble. Como parte de esta evolución, se exploró la implementación de un tótem interactivo publicitario, que consistiría en una pantalla de mayor tamaño que trabaja con sistema operativo Android. Esta decisión estratégica abrió la puerta para desarrollar un aplicativo multiplataforma utilizando .NET MAUI.

El cambio hacia .NET MAUI, un marco de trabajo desarrollado por Microsoft permitió crear una aplicación que puede ejecutarse en diferentes plataformas, como Android, iOS, macOS y Windows, desde una única base de código compartida pero que en primera instancia solo se trabajó pensando en Android. Esta elección no solo proporciona versatilidad y coherencia en el desarrollo, sino que también aprovecha las características avanzadas de .NET MAUI para ofrecer una experiencia de usuario fluida y optimizada en diversas pantallas y dispositivos.

Con el desarrollo concluido, el aplicativo ahora consumiendo eficientemente procedimientos almacenados desde la base de datos de ensamble, interactuando con las tablas creadas para gestionar y registrar los paros en la línea de producción. La implementación final incluyó la última prueba en la planta de Querétaro, donde se pudo validar la funcionalidad en un entorno operativo real. Durante esta fase de prueba, se verificó la precisión de la captura de paros, la autenticación de supervisores y la generación de informes detallados en la base de datos. La aplicación, ejecutándose en el tótem interactivo con Android, demostró su capacidad para reemplazar eficazmente las pizarras tradicionales en las líneas de ensamble. Este hito representa no solo la culminación de un proyecto tecnológico avanzado, sino también una mejora significativa en la eficiencia y la gestión de la producción para INOAC Sistemas Exteriores.

CAPITULO IV: RESULTADO

A lo largo de este segmento, se detalla la aplicación práctica de la metodología Scrum en el desarrollo del proyecto. Se desglosan las diferentes etapas, destacando tablas, diagramas, interfaces de usuario y seguimientos de cada requisito. Este enfoque se alinea con la metodología de desarrollo y los plazos establecidos en el cronograma de actividades, proporcionando una visión exhaustiva de la ejecución del proyecto bajo el marco metodológico propuesto.

4.1 Historias de usuario

A continuación, se describe de manera detallada el requerimiento completo de historias de usuario que fueron desarrolladas encapsuladas por grupos y definiendo su complejidad independiente.

Tabla 3. Historias de usuario del aplicativo

Grupo de historias del usuario	Historia del usuario	Complejidad de la historia
Home o dashboard principal	Pantalla principal	8
	Sistema de trazabilidad	9
Sistema de registro de paros	Autenticación de supervisor	9
	Registro de paro	7
	Finalizar paro	8

4.2 Release plan

En la siguiente tabla se detalla el historial de cada una de las versiones (release plan) desarrolladas y revisadas por el asesor de proyecto, para determinar los puntos de estado en cada fase del diseño, modelado y desarrollo, considerando la versión y las fechas de cada una.

Tabla 4. Release plan de versiones del aplicativo

Versión	Descripción	Fecha
1.0	Primera versión de planes de diseño y diagramas de flujo de la funcionalidad del aplicativo (wireframes y diagramas)	04/10/2023
1.0	Primera versión del aplicativo o dashboard principal, junto con la funcionalidad de conexión al sistema de trazabilidad	06/10/2023
2.0	Segunda versión del aplicativo o dashboard principal	20/10/2023
1.0	Primera versión del sistema de registros de paros para la producción de líneas ensamble de producto terminado	18/11/2023

2.0	Segunda versión del sistema de registro de paro incluyendo la autenticación de supervisores	16/12/2023
-----	---	------------

4.3 Análisis, diseño y modelado de la estructura

4.3.1 Product Backlog

Se lleva a cabo una investigación y se aterrizan los puntos principales y necesidades que deben quedar cubiertas por el aplicativo, junto con el equipo de desarrollo se fue aterrizando las necesidades clave de cada uno de los implicados en las líneas de ensamble de producto terminado:

- Como operadores de las líneas es necesario que tengan información visible en todo momento y que sea fácil de comprender a simple vista
- Como supervisores es indispensable mantener la integridad de los datos, que no sean registrados de manera manual y tener una legibilidad clara de la producción y su rate

4.3.2 Sprint Backlog

Una vez aterrizados los puntos clave e indispensables, se obtiene un listado de las actividades a desarrollar, incluidos los requerimientos del sistema a nivel funcional asignando tiempos y prioridades a realizar durante cada sprint.

Tabla 5. Sprint 1 del análisis, diseño y modelado de la estructura

Sprint 1				
Objetivo. Comprender las necesidades básicas del aplicativo y plasmar funciones básicas mediante un análisis.				
No. Tarea	Requerimiento	Prioridad	Responsable	Tiempo
1.1	Junta con el cliente y el asesor del proyecto para conocer los requerimientos del proyecto, alcance y limitaciones.	Alta	Luis Uriel García Sánchez	1 día
1.2	Reunión con el asesor del proyecto para aterrizar toda la lógica de negocio y funcionalidades.	Alta	Luis Uriel García Sánchez y David Aaron Herrera del Castillo	1 día

Tabla 6. Sprint 6 del análisis, diseño y modelado de la estructura

Sprint 2				
Objetivo. Comprender los casos de uso, diseñar los flujos y secuencia de las interfaces del aplicativo tomando en cuenta los principios de la arquitectura de tres capas.				
No. Tarea	Requerimiento	Prioridad	Responsable	Tiempo
2.1	Diagramas de secuencia del aplicativo.	Alta	Luis Uriel García Sánchez	1 día
2.2	Definición de wireframes para las interfaces de usuario, flujos normales y alternos	Alta	Luis Uriel García Sánchez y David Aaron Herrera del Castillo	2 días
2.3	Creación del proyecto y definición de estructura de carpetas y archivos en Python		David Aaron Herrera del Castillo	2 días

4.3.3 Definition Of Done

Se realizaron las definiciones del sistema al ser implementado, indicando que, si se cumplieron los criterios, o si se cuentan con observaciones obtenidas durante el desarrollo.

- El diseño fue revisado por el asesor del proyecto y aprobado en cuanto a los requerimientos funcionales definidos durante el sprint 1
- La creación del proyecto y definición de las carpetas tomando en cuenta la arquitectura de tres capas fue aprobada de igual manera por el asesor del proyecto, teniendo en cuenta el desarrollo en Python - Tkinter

4.4 Codificación del sistema automatizado de control de procesos

4.4.1 Product Backlog

Con base en los requerimientos estructurales vistos anteriormente, se pudo apreciar con más claridad las necesidades funcionales que necesita tener el aplicativo por lo que se pudo aterrizar de mejor manera a la fase de codificación y desarrollo del sistema automatizado para el conteo de piezas producidas en las líneas de ensamble.

- Como operadores en las líneas de ensamble es necesario que la información pueda ser entendible y clara a primera vista ya que también lo pueden manejar como un panel informativo, por lo que la interfaz de usuario debe ser clara y limpia

4.4.2 Sprint Backlog

Se proceden a enlistar las actividades a realizar comparando los requerimientos del sistema con base en los resultados del cliente, asignando tiempos y prioridades a realizar durante cada sprint.

Tabla 7. Sprint 3 de la codificación del sistema automatizado de control de procesos

Sprint 3				
Objetivo. Desarrollar la primera versión del aplicativo o dashboard principal conectando por completo el sistema de trazabilidad				
No. Tarea	Requerimiento	Prioridad	Responsable	Tiempo
3.1	Realización de dashboard principal haciendo uso de la librería de Python - Tkinter	Alta	David Herrera del Castillo	3 días
3.2	Generación del procedimiento almacenado para consumir datos del sistema de trazabilidad	Alta	David Herrera del Castillo y Luis Uriel García Sánchez	1 día
3.3	Implementación del procedimiento almacenado con el aplicativo mediante una conexión a la base de datos haciendo uso de un ODBC y respetando la arquitectura definida de tres capas	Alta	David Herrera del Castillo	5 días

4.4.3 Diagrama de flujo

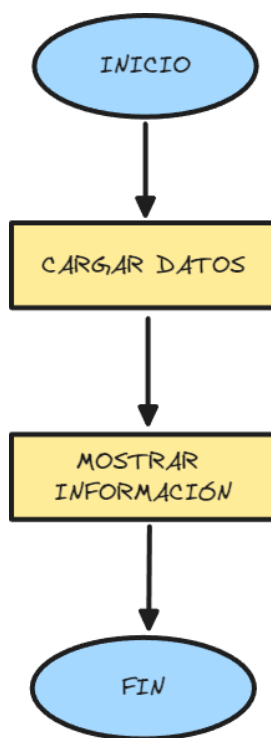


Ilustración 18. Diagrama de flujo de la codificación del sistema automatizado de control de procesos

4.4.4 Caso de uso

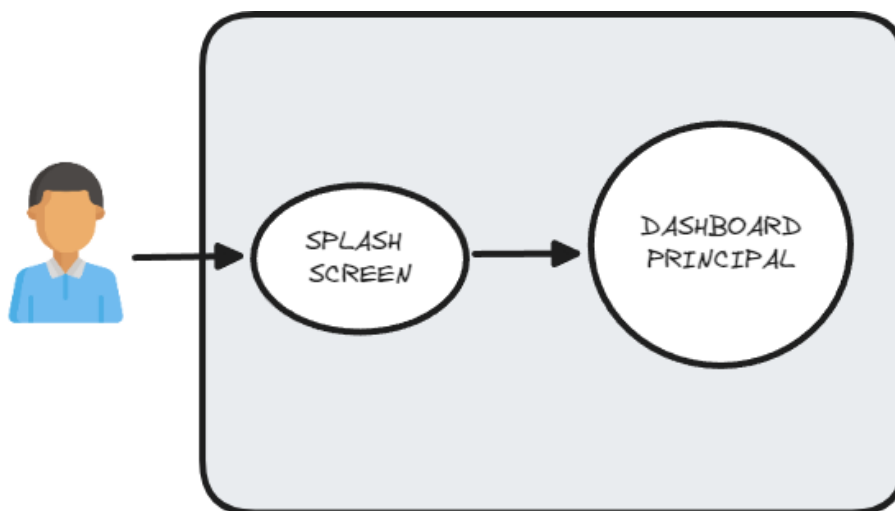


Ilustración 19. Caso de uso de la codificación del sistema automatizado de control de procesos

4.4.5 Definition Of Done

Se realizaron las definiciones del sistema hasta el punto exacto de revisión, detallando, si se cumplieron los criterios, o si se cuentan con observaciones obtenidas durante el desarrollo de la primera versión del aplicativo.

- El desarrollo fue revisado por el asesor del proyecto y corregido en aspectos de rendimiento y mejora de buenas prácticas de código
- El aplicativo fue probado y revisado por el cliente, obteniendo así retroalimentación del que sería una nueva versión del aplicativo haciendo uso de estrategias y tecnologías diferentes multiplataforma

4.5 Codificación del sistema integrado a multiplataforma y registro de paros

4.5.1 Product Backlog

Durante esta fase, se profundiza en los requisitos específicos para el desarrollo del sistema integrado a multiplataforma y el módulo de registro de paros. Se destaca la necesidad de una interfaz adaptable y eficiente que funcione de manera consistente en diferentes plataformas, haciendo énfasis en la experiencia del usuario. Además, se establece la importancia de la funcionalidad de registro de paros, detallando los requisitos para garantizar que los supervisores de producción puedan iniciar y gestionar paros de manera eficaz, incluyendo la autenticación segura, la captura de datos esenciales como motivo y área solicitada.

- Como supervisores que administran el funcionamiento de las líneas de ensamble es importante conocer en todo momento el trabajo de los operadores y la producción en curso, incluida la visualización de tiempos muertos (paros de producción) en las líneas de ensamble

4.5.2 Sprint Backlog

Se proceden a enlistar las actividades a realizar comparando los requerimientos del sistema con base en los resultados del cliente, asignando tiempos y prioridades a realizar durante cada sprint.

Tabla 8. Sprint 4 de la codificación del sistema integrado a multiplataforma y registro de paros

Sprint 4				
Objetivo. Desarrollar la segunda versión del aplicativo haciendo uso del enfoque multiplataforma que se definió conforme al feedback del cliente e implementar un sistema de registro de paros conforme a cada una de las líneas de ensamble				
No. Tarea	Requerimiento	Prioridad	Responsable	Tiempo
4.1	Realización de dashboard principal mediante un lenguaje de marcado (XAML)	Alta	David Herrera del Castillo	3 días
4.3	Estructuración del programa implementando la arquitectura de tres capas en el proyecto .NET MAUI	Alta	David Herrera del Castillo	1 día
4.3	Implementación del procedimiento almacenado de trazabilidad al aplicativo desarrollado con .NET MAUI respetando la arquitectura definida de tres capas	Alta	David Herrera del Castillo	2 días
4.4	Desarrollo del sistema de autenticación de supervisores con base en el sistema de trazabilidad de BAACC Systems	Alta	David Herrera del Castillo	3 días
4.5	Desarrollo de las tablas complementarias a la base de datos del sistema de paros	Alta	David Herrera del Castillo	1 día
4.6	Desarrollo del sistema de registro de paros definiendo los motivos y áreas solicitadas que pueden ocurrir en cada uno de estos	Alta	David Herrera del Castillo	7 días

4.5.3 Diagrama de flujo

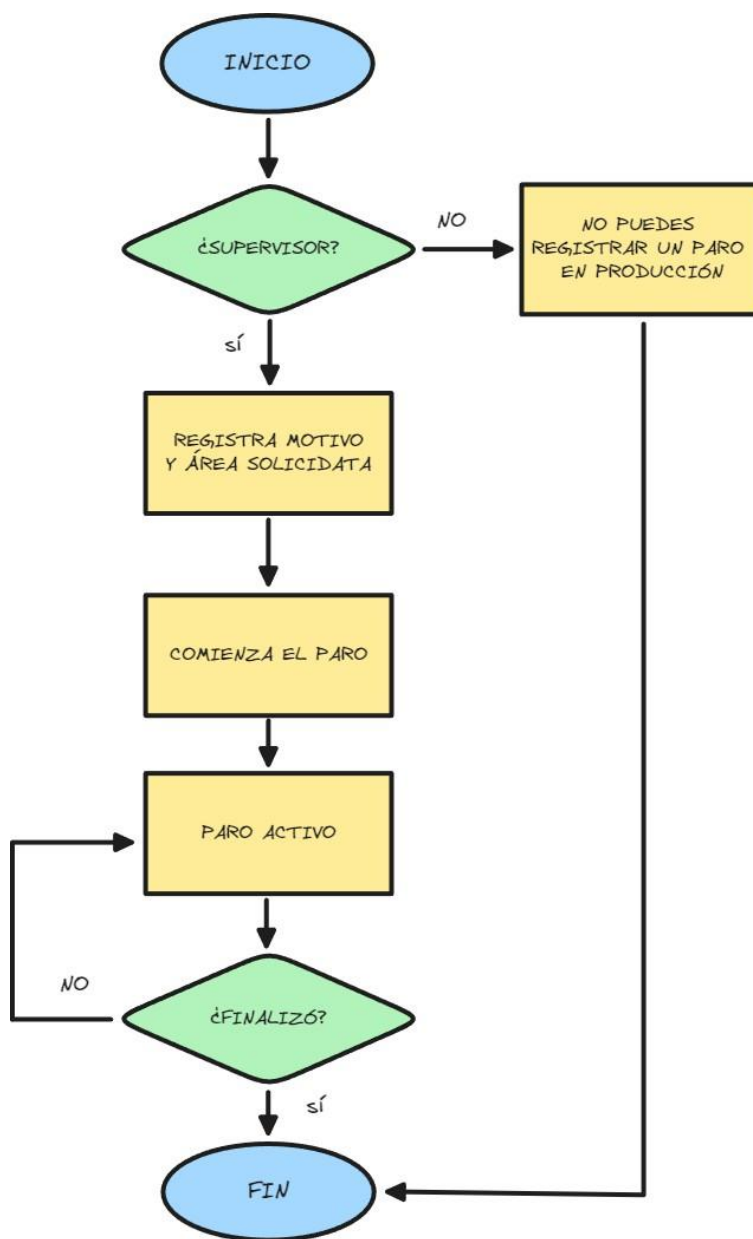


Ilustración 20. Diagrama de flujo del sistema integrado a multiplataforma y registro de paros

4.5.4 Caso de uso

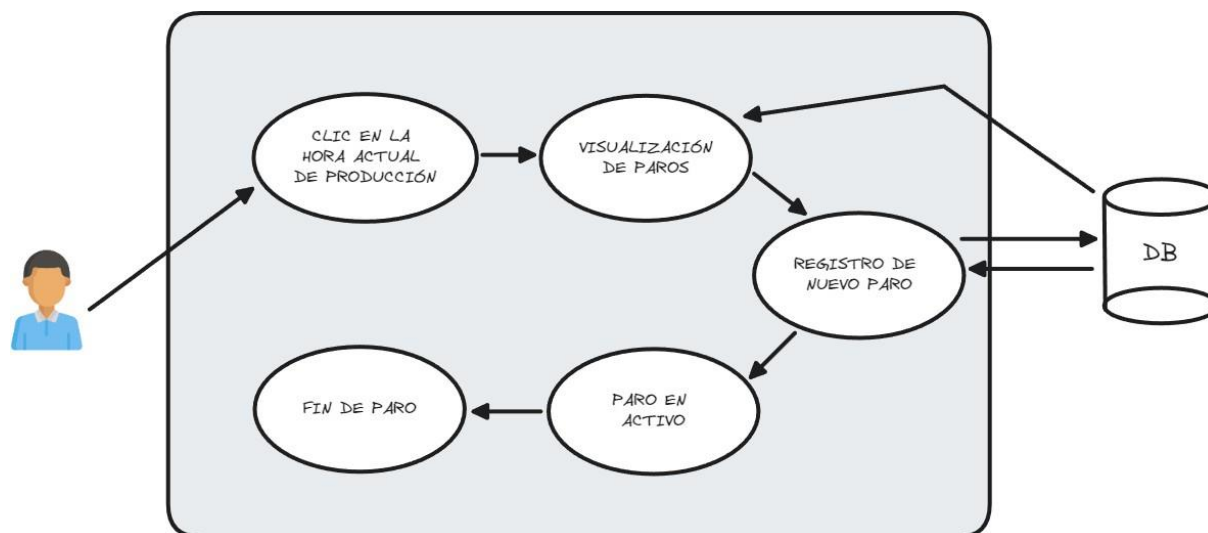


Ilustración 21. Caso de uso del sistema integrado a multiplataforma y registro de paros

4.5.5 Definition Of Done

Se realiza las definiciones del sistema hasta el punto exacto de revisión, detallando, si se cumplieron los criterios, o si se cuentan con observaciones obtenidas durante el desarrollo de la segunda versión del aplicativo y el desarrollo/funcionamiento del sistema de registro de paros y autenticación de los supervisores.

- El desarrollo fue revisado por el asesor del proyecto y corregido en aspectos de rendimiento y mejora de buenas prácticas de código
- El cliente evaluó el aplicativo durante las pruebas, proporcionando feedback detallado que se tradujo principalmente en mejoras visuales. Este proceso de revisión garantizó que el producto final cumpliera con las expectativas y requisitos del cliente.

4.6 Pruebas finales e implementación del aplicativo en producción

4.6.1 Product Backlog

Durante la fase de pruebas finales e implementación, se concentra en la verificación exhaustiva del aplicativo antes de su despliegue definitivo. Se ejecutan pruebas integrales para garantizar la funcionalidad y estabilidad del sistema integrado a multiplataforma y el módulo de registro de paros.

El proceso incluye la revisión minuciosa de la interfaz de usuario. Además, se verifica la correcta ejecución del módulo de registro de paros, confirmando que los supervisores puedan autenticarse y gestionar eficientemente los paros de producción.

Con el objetivo de asegurar la calidad y satisfacción del cliente, se implementa un ciclo de retroalimentación continua. Se incorporan mejoras basadas en el feedback recibido, centrándose especialmente en aspectos visuales y de usabilidad señalados por el cliente. La fase concluye con la implementación exitosa del sistema, listo para su operación plena en el entorno de producción.

4.6.2 Sprint Backlog

Se proceden a enlistar las actividades a realizar comparando los requerimientos del sistema con base en los resultados del cliente, asignando tiempos y prioridades a realizar durante cada sprint.

Tabla 9. Sprint 5 de las pruebas finales e implementación del aplicativo en producción

Sprint 5				
Objetivo. Verificar el funcionamiento completo del aplicativo y los módulos desarrollados en el entorno de producción del cliente final.				
No. Tarea	Requerimiento	Prioridad	Responsable	Tiempo
5.1	Realizar pruebas del sistema automatizado de conteo de piezas por cada una de las líneas de ensamble	Alta	Luis Uriel García Sánchez	1 días
5.2	Realizar pruebas del sistema de autenticación de supervisores y el registro y visualización de paros en cada una	Alta	Luis Uriel García Sánchez	1 día

Tabla 10. Sprint 6 de las pruebas finales e implementación del aplicativo en producción

Sprint 6				
Objetivo. Asegurar la calidad y satisfacción del cliente implementando un ciclo de retroalimentación del sistema de seguimiento y control de las líneas de ensamble de producto terminado				
No. Tarea	Requerimiento	Prioridad	Responsable	Tiempo
6.1	Capacitar a los supervisores y operadores de las líneas de	Alta	David Aaron Herrera del	1 día

	ensamble para poder atender y controlar el aplicativo de control y seguimiento de piezas		Castillo y Luis Uriel García Sánchez	
6.2	Entregar documentación final del sistema de control y seguimiento de piezas	Alta	David Aaron Herrera del Castillo	1 día

4.6.3 Definition Of Done

Se realiza las definiciones del sistema hasta el punto exacto de revisión, detallando, si se cumplieron los criterios, o si se cuentan con observaciones obtenidas durante la fase final de este proceso que comprende principalmente el despliegue completo del aplicativo en la planta de INOAC Sistemas Exteriores S.A. de C.V., la capacitación del personal y la entrega de la documentación.

- El aplicativo se desplegó con éxito en la planta de Querétaro, obteniendo feedback positivo del cliente
- Se llevó a cabo una sesión integral de capacitación con el cliente para garantizar una comprensión completa de las funciones y características del aplicativo
- Se proporcionó al cliente una documentación detallada, abarcando manuales de usuario y descripciones técnicas

4.7 Burndown chart general del proyecto

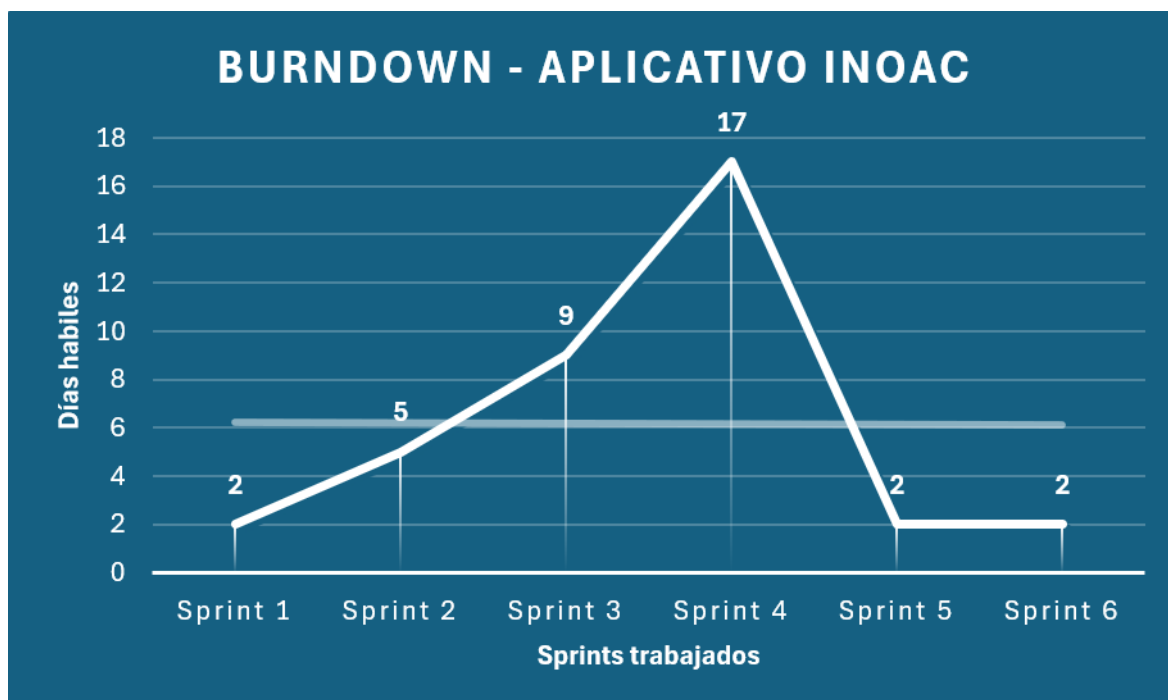


Ilustración 22. Burndown chart general del proyecto

El gráfico burndown es una representación visual del progreso del proyecto a lo largo de los sprints. Muestra la cantidad de trabajo pendiente en función del tiempo. La línea principal refleja la disminución o aumento del trabajo restante a medida que avanzan los sprints, lo que proporciona una rápida visión general del rendimiento durante el desarrollo.

CONCLUSIONES

En cuanto al **desarrollo del sistema de registro de paros**, se logró implementar un sistema eficiente que permite la definición y consulta de paros en las líneas de ensamble. La aplicación satisface las necesidades del cliente al facilitar el seguimiento y control de los paros durante el proceso de producción.

En cuanto a la **automatización basada en el sistema de trazabilidad**, se alcanzó con éxito la integración del sistema con la trazabilidad de BAACC SYSTEMS, gracias a este se pudo definir un aplicativo totalmente conectado con la trazabilidad y seguimiento de piezas en producción. El aplicativo automatiza el registro de la finalización del ensamble de productos, mejorando significativamente la eficiencia y precisión en el control de la producción.

Durante la **implementación de sistema de autenticación para supervisores**, la aplicación incluye ahora un sistema de autenticación seguro que concede a los supervisores los privilegios necesarios para realizar ediciones en puntos clave del proceso de ensamble, haciendo énfasis al sistema de registro de paros. Esto garantiza un control adecuado sobre las operaciones críticas del sistema.

En el apartado de la **integración de procedimientos almacenados**, se logró una integración exitosa de los procedimientos almacenados, consumiendo datos directamente de la base de datos del sistema de trazabilidad y tablas generadas para el funcionamiento completo del aplicativo. Proporcionando así una conexión sólida y directa con la información necesaria para el funcionamiento del aplicativo.

En cuanto a la **implementación de arquitectura de capas**, la aplicación se diseñó con una arquitectura de capas que asegura una comunicación segura entre el aplicativo y la base de datos. Este enfoque permite realizar cambios sin afectar las funcionalidades existentes, asegurando la flexibilidad y mantenimiento a largo plazo.

Conclusión general del proyecto

En términos generales, el proyecto logró cumplir con su objetivo principal de modernizar el sistema de control de producción en las líneas de ensamble de INOAC Sistemas Exteriores. La implementación del aplicativo principalmente Android y la transición hacia un enfoque multiplataforma con .NET MAUI representan un avance significativo hacia la mejora de la eficiencia y precisión en la gestión de la producción para la planta ubicada en Querétaro.

Sugerencias o propuestas para futuro

Para futuras iteraciones, se sugiere considerar la expansión del aplicativo para incluir características adicionales, como la generación de informes detallados, análisis predictivo de

producción y la implementación de tecnologías emergentes como la inteligencia artificial. Estas adiciones no solo enriquecerían la experiencia del usuario sino también potenciarían la eficiencia operativa.

Adicionalmente, debido a las limitaciones económicas que surgieron durante la implementación, se recomienda explorar en futuras versiones la optimización del aplicativo para adaptarse a pantallas más pequeñas, como las utilizadas en lugar del tótem interactivo inicialmente planeados. Esto podría implicar ajustes en la interfaz de usuario y la disposición de la información para garantizar la legibilidad y la eficacia del aplicativo en estos nuevos dispositivos.

Apreciación/experiencia profesional

Este proyecto ha sido un pilar fundamental en mi desarrollo profesional. La inmersión en tecnologías avanzadas como .NET y SQL Server ha sido esencial para la adquisición de habilidades prácticas sólidas. La interacción constante con sistemas ya establecidos en el entorno de INOAC Sistemas Exteriores ha proporcionado valiosas lecciones sobre la integración efectiva de soluciones.

A lo largo de este proceso, la colaboración estrecha con el cliente ha sido crucial, brindándome una perspectiva única sobre las necesidades específicas de la industria y perfeccionando mis habilidades para adaptarme a cambios inesperados. La resolución de desafíos técnicos ha sido una oportunidad para poner a prueba mis conocimientos y fortalecer mi capacidad para encontrar soluciones prácticas.

Esta experiencia no solo ha contribuido a mi formación como ingeniero, sino que también ha ampliado mi comprensión sobre la importancia de la tecnología en entornos industriales. Me siento preparado y motivado para abordar futuros proyectos con la misma dedicación y entusiasmo, aprovechando los valiosos conocimientos adquiridos en este trayecto de desarrollo de software.

COMPETENCIAS DESARROLLADAS

Tabla 11. Competencias desarrolladas durante el desarrollo del proyecto

Asignatura	Competencia
Fundamentos de programación	Desarrolla algoritmos y utiliza lenguajes de programación para resolver problemas de manera eficiente, aplicando conceptos fundamentales en el diseño y la implementación de soluciones.
Fundamentos de investigación	Aplica métodos de investigación para recolectar, analizar y presentar información de manera rigurosa y ética, contribuyendo al avance del conocimiento en su área de estudio.
Taller de ética	Analiza y reflexiona sobre dilemas éticos relacionados con la tecnología y la informática, proponiendo soluciones éticas y tomando decisiones informadas en el ejercicio profesional.
Programación orientada a objetos	Diseña, implementa y gestiona sistemas de software utilizando los principios y conceptos de la programación orientada a objetos, promoviendo la reutilización del código y la creación de soluciones modulares.
Fundamentos de bases de datos	Diseña y desarrolla bases de datos eficientes, aplicando conocimientos fundamentales sobre modelado de datos, normalización y consultas para gestionar la información de manera efectiva.
Taller de base de datos	Implementa soluciones prácticas en entornos de bases de datos, abordando problemas específicos, realizando consultas avanzadas y administrando eficientemente la información.
Ingeniería de software	Aplica principios de ingeniería de software en la planificación, diseño, implementación y mantenimiento de sistemas, asegurando la calidad y la eficiencia en el ciclo de vida del software.
Taller de ingeniería de software	Gestiona proyectos de desarrollo de software utilizando metodologías y técnicas de ingeniería de software, liderando equipos y

	aplicando buenas prácticas para lograr resultados exitosos.
Taller de investigación I	Desarrolla habilidades de investigación avanzada, identificando problemas, formulando preguntas de investigación y aplicando metodologías adecuadas para generar conocimiento significativo.
Sistemas operativos I	Comprende, configura y administra sistemas operativos, optimizando el rendimiento y asegurando la estabilidad de los entornos computacionales.
Arquitectura de computadora	Analiza y comprende la arquitectura de los sistemas computacionales, desde el nivel de hardware hasta el nivel de software, para diseñar soluciones eficientes y tomar decisiones informadas en la selección de componentes.
Taller de investigación II	Desarrolla habilidades avanzadas de investigación, diseñando y ejecutando proyectos de investigación independientes, contribuyendo al conocimiento en el área de estudio y aplicando metodologías apropiadas.
Sistemas operativos II	Administra sistemas operativos en entornos más complejos, implementando soluciones avanzadas para la gestión de recursos, la seguridad y la integración de servicios en redes.
Auditoría en tecnologías de la información	Evalúa y audita sistemas y procesos de tecnologías de la información, identificando riesgos, implementando controles de seguridad y asegurando el cumplimiento de normativas y mejores prácticas.
Fundamentos de Android	Desarrolla aplicaciones para dispositivos móviles utilizando el sistema operativo Android. Aplica los principios fundamentales de programación para crear soluciones eficientes, comprendiendo los conceptos clave de Android, como actividades, fragmentos e interfaz de usuario.

Android avanzado	Diseña y desarrolla aplicaciones Android avanzadas, aplicando conceptos de programación orientada a objetos y técnicas avanzadas de Android. Utiliza patrones de diseño y optimiza el rendimiento de las aplicaciones.
Servicios de nube	Aplica la arquitectura de tres capas y soluciones en la nube para el desarrollo de aplicaciones. Aprovecha eficientemente estas estructuras para mejorar la funcionalidad de las aplicaciones, optimizando la gestión de datos en la nube.
Fundamentos de iOS	Desarrolla aplicaciones nativas para dispositivos iOS, utilizando el entorno de desarrollo de Apple. Aplica los principios de programación orientada a objetos en Swift, comprende la interfaz de usuario.

FUENTES DE INFORMACIÓN

Amazon Web Services. (2023). *Amazon Web Services*. Obtenido de <https://aws.amazon.com/es/what-is/scrum/>

De Los Ríos Sánchez, J. L. (27 de Febrero de 2023). *IEBSchool*. Obtenido de <https://www.iebschool.com/blog/industria-cuarta-revolucion-industrial-business-tech-logistica/>

Forbes. (7 de Enero de 2018). *Forbes*. Obtenido de <https://www.forbes.com/sites/louiscolumbus/2018/01/07/83-of-enterprise-workloads-will-be-in-the-cloud-by-2020/?sh=213347de6261#398420276261>

Google LLC. (2023). *Google Cloud*. Obtenido de <https://cloud.google.com/learn/paas-vs-iaas-vs-saas?hl=es>

Google LLC. (2023). *Google Cloud*. Obtenido de <https://cloud.google.com/learn/what-is-cloud-computing?hl=es>

HubSpot. (2023). *HubSpot*. Obtenido de <https://blog.hubspot.es/marketing/sistema-informacion#que>

IBM. (2023). Obtenido de <https://www.ibm.com/mx-es/topics/industry-4-0>

IBM. (2023). *IBM*. Obtenido de <https://www.ibm.com/mx-es/topics/three-tier-architecture>

ISEC, U. d. (19 de Julio de 2022). *UNEG Web Site*. Obtenido de <https://uneg.edu.mx/como-se-usan-las-tic-en-la-sociedad/>

KeepCoding. (4 de Diciembre de 2023). Obtenido de KeepCoding: <https://keepcoding.io/blog/que-es-tkinter/#:~:text=Tkinter%20es%20una%20librer%C3%ADa%20del,gr%C3%A1fica%20de%20escritorio%20con%20Python.>

KIO. (2023). *KIO*. Obtenido de <https://www.kio.tech/blog/data-center/los-sistemas-de-informacion-de-una-empresa>

Martínez Canelo, M. (9 de Junio de 2020). *Profile*. Obtenido de <https://profile.es/blog/que-son-los-paradigmas-de-programacion/>

Microsoft. (2023). *Microsoft*. Obtenido de <https://learn.microsoft.com/es-es/visualstudio/get-started/visual-studio-ide?view=vs-2022>

Microsoft. (2023). *Microsoft*. Obtenido de <https://learn.microsoft.com/es-es/sql/sql-server/what-is-sql-server?view=sql-server-ver16>

Microsoft. (2023). *Microsoft*. Obtenido de <https://learn.microsoft.com/es-es/dotnet/maui/what-is-maui?view=net-maui-8.0>

Microsoft. (2023). *Microsoft*. Obtenido de <https://learn.microsoft.com/es-mx/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver16>

Mozilla Developer. (2023). *Mozilla Developer*. Obtenido de <https://developer.mozilla.org/es/docs/Glossary/OOP>

Python. (2024). Obtenido de Python: <https://docs.python.org/es/3/library/tkinter.html>

RedHat. (5 de Septiembre de 2023). *RedHat*. Obtenido de <https://www.redhat.com/es/topics/cloud-computing/what-is-private-cloud>

RedHat. (17 de Marzo de 2023). *RedHat*. Obtenido de <https://www.redhat.com/es/topics/cloud-computing/what-is-hybrid-cloud>

RedHat. (3 de Enero de 2023). *RedHat*. Obtenido de <https://www.redhat.com/es/topics/cloud-computing/what-is-public-cloud>

Silva, D. d. (8 de Junio de 2021). *Zendesk*. Obtenido de <https://www.zendesk.com.mx/blog/cloud-computing-definicion/>

SLYDE. (26 de Septiembre de 2022). *SLYDE*. Obtenido de <https://www.slyde.com/es/blog/automatizacion-de-procesos-6070ae4c9b901904c4349dcb>

TIBCO. (2023). *TIBCO*. Obtenido de <https://www.tibco.com/es/reference-center/what-is-process-automation>

Universidad Europea. (24 de Agosto de 2022). *Universidad Europea*. Obtenido de <https://universidadeuropea.com/blog/programacion-orientada-objetos/>

ANEXOS

Contrato de confidencialidad con BAACC SYSTEMS S.A. de C.V.



BAACC SYSTEMS S.A. DE C.V.

CONTRATO DE CESION DE DERECHOS Y CONFIDENCIALIDAD QUE CELEBRAN EL C. DAVID AARON HERRERA DEL CASTILLO, A QUIEN EN LO SUCESIVO SE LE DESIGNARA EL "CEDENTE", Y POR LA OTRA PARTE; BAACC SYSTEMS S.A. DE C.V. A LA QUE EN LO SUCESIVO SE LE DENOMINARA EL "CESIONARIO", REPRESENTADA EN ESTE ACTO POR EL C. ALEJANDRO MONTEERRUBIO HERNÁNDEZ, AL TENOR DE LAS DECLARACIONES Y CLAUSULAS SIGUIENTES:

DECLARACIONES

I. DECLARA EL "CEDENTE":

1. Es una persona física de nacionalidad MEXICANA en pleno uso de su capacidad legal.
2. Acredita su personalidad mediante INE, que se adjunta como Anexo 1 al presente contrato.
3. Acredita no ser propietario de ningún producto de software (programas, sistemas, etc.) desarrollados por el CEDENTE durante las fechas que van del **01 de Enero de 2024 al 31 de Diciembre de 2024**, y ceder los Derechos de Autor que posea sobre cualquier software desarrollado en esas fechas, por estar comprometida su total disponibilidad en exclusividad con el CESIONARIO, debido al tipo de actividades necesarias para el desarrollo de software y actos relacionados con ello, para con el CESIONARIO y para con sus diferentes clientes en esas fechas, durante las cuales hizo uso de un cubículo e instalaciones eléctricas, de red alámbrica e inalámbrica y telefónicas, así como implementos de oficina, software, etc. con conexiones a Internet para el buen cumplimiento de su trabajo, pagados por el CESIONARIO, en las oficinas del CESIONARIO, situadas en la dirección: **Ébanos 107 Col. Villa de las Flores, Coacalco de Berriozábal, Estado de México, C.P. 55710**.
4. Acuerda entregar cualquier copia u original de documentos en cualquier formato sea este electrónico, escrito, grabado, etc. que tengan alguna relación con el CESIONARIO o con alguno de los clientes del CESIONARIO.
5. Se compromete a no divulgar de ninguna forma hablada, escrita, electrónica, etc. los datos, procedimientos, acuerdos, contactos del CESIONARIO o de alguno de sus CLIENTES.
6. Es su deseo celebrar el presente contrato de Cesión y Confidencialidad libre de vicio alguno.
7. La Cesión de Derechos y Confidencialidad se realiza con motivo de que libremente se incorporó a los proyectos del CESIONARIO y que estuvo presente en reuniones con clientes del CESIONARIO y en reuniones con el CESIONARIO donde se detallaron procesos, procedimientos, datos, nombres de usuarios, contraseñas, etc. de la operación de los clientes del CESIONARIO y del CESIONARIO, y que leyó y tuvo en su poder los programas de software, scripts de bases de datos en lenguaje SQL, diseño de bases de datos, bases de datos con estructuras y datos, propiedad y con registro de derechos de autor del CESIONARIO, del **01 de Enero de 2024 al 31 de Diciembre de 2024**.
8. Que se compromete a no tocar, ver, copiar, divulgar, prestar, comentar o usar ninguno de los documentos, procedimientos, datos, programas, etc. no importando su formato, ya sea escrito, electrónico, grabado, etc. que se encuentren en las instalaciones, computadoras, servidores, etc. del CESIONARIO o de alguno de los clientes del CESIONARIO, o que sean producto de las reuniones con los clientes del CESIONARIO o de las reuniones internas con el CESIONARIO, ni firmarse o introducirse en ninguna de sus instalaciones, computadoras, servidores, etc. del CESIONARIO o de alguno de sus CLIENTES a partir e incluido el día **01 de Enero de 2024**.
9. Que, para los efectos legales derivados de este contrato, señala como su domicilio legal: **LINOS 617, COL. VILLA DE LAS FLORES, COACALCO DE BERRIOZABAL ESTADO DE MÉXICO C.P. 55710**

Camelia 64, Col. Bosque de los Remedios, Naucalpan de Juárez, Estado de México, C.P. 53458

Oficina **(55) 8624 7021** y **(55) 7577 8805** * WhatsApp Business **(55) 6263 7194**

contacto@baacc-systems.mx * www.baacc.mx



BAACC SYSTEMS S.A. DE C.V.

II. DECLARA EL "CESIONARIO":

1. Es una SOCIEDAD ANONIMA DE CAPITAL VARIABLE constituida de conformidad con las leyes de los Estados Unidos Mexicanos.
2. Tiene como objeto:
 - i. Análisis de requerimiento de negocios para diseñar, desarrollar e implementar soluciones de infraestructura de redes, comunicaciones, sistemas operativos, aplicaciones y bases de datos.
 - ii. Diseñar, desarrollar e implementar soluciones de software centralizadas, distribuidas, cliente servidor y en la web.
 - iii. Diseñar, desarrollar e implementar modelos de gestión, así como lineamientos, normas, procesos, procedimientos y estándares en tecnologías de la información y de las comunicaciones.
 - iv. Servicios de posicionamiento en buscadores de páginas web. Compra y venta al mayor y menor de dominios, espacio de almacenamiento en la web, banda ancha y certificados de autenticidad.
 - v. Instalaciones de redes integradas de cómputo con cableado estructurado, cableado de fibra óptica, cableado coaxial.
 - vi. Venta y arrendamiento de equipos de cómputo, dispositivos periféricos, gabinetes para concentradores y ruteadores. Mantenimiento preventivo y correctivo a redes, equipos en red, equipos de cómputo, dispositivos periféricos.
 - vii. Prestación, contratación, subcontratación de todo tipo de servicios informáticos, de telecomunicaciones y de consultoría en tecnologías de la información y de las comunicaciones.
 - viii. Elaboración, edición, producción, coproducción, rodaje, publicación y comercialización de toda clase de obras audiovisuales, incluyendo spots publicitarios; doblaje, sonorización, montaje, edición y postproducción de obras audiovisuales.
 - ix. Servicios de diseño gráfico, realización de proyectos editoriales, de artes gráficas o de cualquier otra naturaleza. Publicación electrónica, imagen digital, impresión, servicios publicitarios, agencia de publicidad. Gestión, realización, comercialización y distribución de servicios de publicidad a través de uno o varios medios de difusión.
 - x. Ejecutar, celebrar y cumplir con toda clase de actos mercantiles, civiles, laborales, administrativos, y de cualquier otra naturaleza relacionado con las actividades de la sociedad en forma directa o indirecta por lo que así mismo, podrán adquirir, poseer, arrendar usar, enajenar, y en general ser parte de cualquier contrato relativo a la posesión, arrendamiento, uso y enajenación de los bienes muebles.
 - xi. Solicitar y obtener préstamos, suscribir y endosar, aceptar, toda clase de títulos de crédito y garantizar obligaciones de la sociedad, con la intervención de las instituciones específicas por la ley cuando ello sea necesario, directamente relacionados con los objetos de la sociedad.
 - xii. Comercializar, crear, ampliar, fusionar, enajenar, operar y administrar empresas dedicadas a cualquier actividad anexa o conexa, establecer, arrendar, subarrendar y negociar en todas sus formas con fábricas, plantas industriales, almacenes, tiendas, oficinas y en general con todo tipo de personas físicas.
 - xiii. Manufacturar, importar, exportar, reconstruir, distribuir, ensamblar y en general negociar en cualquier forma con los accesorios, artículos, refacciones, suministros, materia prima, productos terminados o semiterminados y con aquellos bienes relacionados directa o indirectamente con los de esta sociedad.
 - xiv. Establecer oficinas, agencias, representaciones y sucursales en la República Mexicana y en el extranjero para el desarrollo de las actividades de la empresa.
 - xv. La adquisición por cualquier medio legal de acciones, partes sociales y en general participaciones y/o intereses en otras sociedades de objeto análogo, similar o vinculado directa o indirectamente con los de esta sociedad.
 - xvi. Representar o ser agente en los Estados Unidos Mexicanos de empresas industriales y comerciales, nacionales o extranjeras.

Camelia 64, Col. Bosque de los Remedios, Naucalpan de Juárez, Estado de México, C.P. 53458

Oficina **(55) 8624 7021** y **(55) 7577 8805** * WhatsApp Business **(55) 6263 7194**

contacto@baacc-systems.mx * www.baacc.mx



BAACC SYSTEMS S.A. DE C.V.

- xvii. Comprar, vender, arrendar, hipotecar y en general comercializar en cualquier forma legal con bienes de muebles e inmuebles.
 - xviii. Establecer sucursales y subsidiarias en la República Mexicana y/o en el extranjero.
 - xix. Solicitar, obtener, comprar, arrendar, ceder, o de cualquier otra forma, disponer y adquirir patentes de invención, registro de modelos de utilidad, diseños industriales, marcas, avisos y nombres comerciales, licencias de explotación y de uso de estos: declaraciones de protección de denominaciones de origen y autorizaciones para uso y derechos de autor ya sean en los Estados Unidos Mexicanos o en el extranjero.
 - xx. Dar y tomar dinero en préstamo con y sin garantía; emitir bonos y cédulas; suscribir, aceptar, endosar y avalar títulos de crédito; garantizar obligaciones propias y de terceros, con o sin garantías específicas; emitir obligaciones y toda clase de valores con intervención en su caso, de las instituciones requeridas por la Ley.
 - xxi. Otorgar garantías propias a favor de terceros, fianzas, confianzas, así como obligarse solidariamente ante instituciones de crédito y otras entidades financieras y particulares a favor de las sociedades y otras personas morales y físicas aun cuando se trate de personas extrañas a la sociedad.
 - xxii. En general realizar toda clase de actos comerciales, industriales y celebrar toda clase de contratos, convenios y operaciones, ya sean civiles o mercantiles permitidos por la Ley y llevar a cabo todos los actos legales necesarios para el desarrollo de su objeto social.
3. El C. ALEJANDRO MONTEERRUBIO HERNÁNDEZ tiene poderes y facultades suficientes para celebrar el presente contrato, mismas que hasta la fecha no le han sido revocadas, modificadas y/o limitadas.
 4. Es su deseo celebrar el presente contrato de Cesión de Derechos y Confidencialidad, libre de vicio alguno.
 5. Que, para los efectos legales derivados de este contrato, señala como su domicilio legal: **CALLE SAN FRANCISCO, MANZANA 17 LOTE 13 CASA 1, COL. LOS HEROES, COACALCO DE BERRIOZABAL, ESTADO DE MÉXICO, C.P. 55712.**

CLAUSULAS

PRIMERA. En caso de cualquier controversia que pudiera suscitarse respecto al cumplimiento o interpretación del presente instrumento, las partes acuerdan someterse a las leyes aplicables, competencia y jurisdicción de los tribunales del Estado de México, por lo que a partir de este momento renuncia a cualquier otra jurisdicción que por virtud del domicilio presente o futuro o por cualquier otra circunstancia pudiese corresponderles.

Leído el presente contrato y enteradas las partes de su valor y contenido legal, lo firman al calce de la última hoja y al margen de las anteriores, en Coacalco de Berriozábal, Estado de México, a PRIMERO del mes de SEPTIEMBRE del 2023.

POR EL "CEDENTE"

DAVID AARON HERRERA DEL CASTILLO

POR EL "CESIONARIO"

ALEJANDRO MONTEERRUBIO HERNÁNDEZ

**MÉXICO INSTITUTO NACIONAL ELECTORAL
CREDENCIAL PARA VOTAR**

NOMBRE
HERRERA
DEL CASTILLO
DAVID AARON

DOMICILIO
C SAN FRANCISCO MZA 17 LT 13 CS 1
FRACC LOS HEROEES COACALCO 55712
COACALCO DE BERRIOZABAL, MEX

FECHA DE NACIMIENTO
07/09/2000

SEXO H

CLAVE DE ELECTOR HRCSDV00090709HT00

CURP HECDD000907HDFRSV/A8

ANO DE REGISTRO 2018 00

ESTADO 15

MUNICIPIO 020

SECCION 0558

LOCALIDAD 0001

EMISION 2018

VIGENCIA 2028

IDMEX 1780869869<<0558121924302
0009070H2812313MEX<00<<20331<7
HERRERA<DEL<CASTI<<DAVID<AARON

INE



BAACC Systems, S.A. de C.V.
BSY - 160628 - Jr. 5

Camelia 64
Col. Bosque de los Remedios
Naucalpan de Juárez
Estado de México
C.P. 53458

www.baacc-systems.mx