**Universidad Tecnológica de La Habana José Antonio Echeverría**

**Facultad de Ingeniería Informática**



**Desarrollo de tecnología de monitoreo y diagnóstico industrial.**

***Informe de las Prácticas Profesionales 1***

**Autor: César Fernández García**

**Tutores: Dr. C. Juan C. Sepúlveda Peña**

**Ing. Juan Alejandro Baster Jiménez**

**La Habana, Julio 2023**

**Resumen**

Es lo último que se escribe, máximo debe tener 250 palabras, debe incluir problemática, objetivo, resultados y valor de los resultados.

**Palabras claves:**

**Abstract**

Resumen en inglés, máximo debe tener 250 palabras, debe incluir problemática, objetivo, resultados y valor de los resultados.

**Keywords:**

**Anexo A. Plan individual de tareas del estudiante**

|  |  |  |
| --- | --- | --- |
| **Tareas** | **Fecha de entrega** | **Rol(es) que desarrolla(n) con la tarea** |
| Reunión de inicio de la práctica | 12/6/2023 | - |
| Asimilación de la plataforma RAD Studio Versión 11.3 | 20/6/2023 | PG |
| Asimilación de la teoría sobre análisis por vibraciones y concepto de la (Transformada rápida de Fourier) FFT. Análisis en el dominio del tiempo y de la frecuencia. | 20/6/2023 | AS |
| Análisis y diseño de una plataforma escalable de captura y procesamiento de datos. | 23/6/2023 | AR |
| Programar una aplicación Multiplataforma (Windows, Linux, Android) que sea capaz de seleccionar el *driver* de captura de datos (.dll en Windows, .so en Linux y Android) y usarla para capturar datos, mostrar dichos datos gráficamente en el dominio del tiempo y de la frecuencia (usando la (Transformada rápida de Fourier) FFT) y almacenarlos en una base de datos para futuro post-procesamientos. | 30/6/23 | PG |
| Entrega primera versión del software para revisión | 30/6/23 |  |
| Entrega versión final del software | 15/7/23 |  |
| Elaborar informe de la práctica | 5/7/23 | EE |
| Entregar informe de la práctica al tutor | 5/7/23 | EE |
| Rectificar señalamientos del informe | 10/7/23 | EE |
| Entrega del informe final de la práctica | 17/7/23 | EE |
| Defensa de la práctica | 19-21/7/23 | Todos |

**Índice**

[Introducción 1](#_Toc140190901)

[Capítulo 1: Fundamentación teórica 4](#_Toc140190902)

[Capítulo 2: Solución propuesta 17](#_Toc140190903)

[Capítulo 3: Validación de la solución 20](#_Toc140190904)

[Conclusiones 21](#_Toc140190905)

[Recomendaciones 22](#_Toc140190906)

[Referencias bibliográficas 23](#_Toc140190907)

# Introducción

La Industria 4.0 es la materialización de la transformación digital del sector industrial, que ofrece toma de decisiones en tiempo real y una productividad, mantenibilidad, flexibilidad y agilidad mayores. La Industria 4.0 se caracteriza por la creciente automatización y el uso de máquinas y fábricas inteligentes que se integran con tecnologías como el internet de las cosas (IoT, por sus siglas en inglés), análisis y cloud computing, IA y machine learning. La Industria 4.0 también implica el desarrollo de fábricas inteligentes que proporcionan una oportunidad increíble a la industria manufacturera: les abre el camino a la cuarta revolución industrial (¿Qué es la Industria 4.0 y cómo funciona? | IBM, 2023).

La gran crisis económica mundial que se vive en la actualidad afecta a todos los aspectos de la sociedad, lo que se refleja en el deterioro de las maquinarias de transporte y de la tecnología en general. La obsolescencia de las maquinarias relacionadas con la producción de cualquier índole conlleva al desgaste de las piezas que las componen, lo que a la larga provoca roturas irreparables y afecta a los diferentes procesos de producción. Además, es muy difícil obtener nuevas máquinas y piezas debido al bloqueo económico-financiero impuesto por los Estados Unidos. Una de las causas del deterioro y rotura de las máquinas industriales es la falta de mantenimiento y reemplazo de las piezas en el momento adecuado. Debido a las causas antes mencionadas, estos elementos no son suficientes para todas las maquinarias que intervienen en los diferentes procesos de producción. Por lo tanto, es necesario detectar a tiempo aquellos equipos que necesitan más intervención preventiva. Esto es posible si se pueden detectar los indicios que provocan el desgaste y las roturas antes mencionadas, lo cual se puede lograr con el uso de un método capaz de monitorear los parámetros que caracterizan las vibraciones de dichas maquinarias. Sin embargo, el problema radica en que se carece de este método.

En la actualidad han surgido máquinas que utilizan componentes rotatorios que generan vibraciones, tales como; compresores de alta velocidad, turbinas de vapor y gas, generadores, bombas, etc. Aunque por lo general son resistentes y bien diseñados, los ejes en operación son muchas veces vulnerables a defectos que se desarrollan de manera imprevista. Dada la problemática de no poder detectar de forma prematura estas vibraciones (imperceptibles o no) tan peligrosas, en los equipos ocurren averías o roturas, ejemplo de estas son desalineamiento, desequilibrio, desgaste, y, por transitividad el paro de la producción (Zambrano-Reyes, Nossov, & Gómez-Mancilla) (Mohd Ghazali & Rahiman, 2021).

Esta problemática se ha intentado solucionar en el país de una manera a veces inadecuada, ya que estas soluciones no toman en cuenta las restricciones internacionales que presenta el país debido al embargo económico, por ejemplo: falta de presupuesto, falta de piezas de repuesto, la compra de piezas y máquinas a terceros países por un precio mayor, demora en la obtención de estos elementos y complementos. (Ministerio de Relaciones Exteriores de Cuba, 2023)

Teniendo en cuenta lo anterior descrito se puede identificar como problema:

¿Cómo desarrollar un software para el monitoreo y diagnóstico en tiempo real de máquinas rotatorias industriales?

El objeto de estudio está enmarcado en los diferentes aplicaciones existentes que establezcan el monitoreo sobre máquinas generadoras de vibraciones, así como la seguridad y almacenamiento de los datos procesados de señales y del software en sí, y que permitan un análisis de tendencia de los datos procesados, teniendo como campo de acción las aplicaciones de escritorio y móviles que permitan, el procesamiento de señales generadas por sensores de vibraciones en máquinas industriales.

Como objetivo general se plantea desarrollarun software multiplataforma para el monitoreo y diagnóstico industrial.

A partir de lo previamente mencionado se trazaron los siguientes objetivos específicos:

1. Investigar sobre otras aplicaciones que realicen el monitoreo y diagnóstico de procesos industriales.
2. Diseñar un software multiplataforma para el monitoreo de las vibraciones en máquinas rotatorias que realizan algún proceso industrial.
3. Implementar funcionalidad que permita la visualización en tiempo real de las señales capturadas tanto en el dominio del tiempo como en el de la frecuencia.
4. Implementar funcionalidad para almacenar en una base de datos las señales y los resultados provenientes de procesar cada señal.
5. Implementar mecanismos de seguridad que aseguren que el software no sea utilizado por personal no autorizado.
6. Validar la solución propuesta.

Para el cumplimiento de dichos objetivos se definieron las siguientes tareas:

1. Investigar y analizar sobre aplicaciones que permitan el monitoreo, visualización de señales, almacenamiento de su procesamiento y análisis de tendencia de esos procesamientos.
2. Asimilar la teoría sobre análisis por vibraciones y concepto de la transformada rápida de Fourier (FFT, por sus siglas en inglés).
3. Asimilar el lenguaje de programación Pascal, así como de la plataforma RAD Studio Versión 11.3 y el sistema gestor de bases de datos SQLite.
4. Analizar y diseñar una plataforma escalable de captura y procesamiento de datos.
5. Programar una aplicación multiplataforma que sea capaz de seleccionar el *driver* de captura de datos y usarla para capturar datos, mostrar dichos datos gráficamente en el dominio del tiempo y de la frecuencia, y almacenarlos en una base de datos para futuro post-procesamientos.
6. Realizar pruebas de software de caja negra y pruebas usabilidad.

**Valor práctico**

El valor práctico de este trabajo reside en el desarrollo de una aplicación multiplataforma que permita al usuario el monitoreo y almacenamiento del procesamiento de señales proveniente de máquinas industriales de cualquier tipo que produzcan vibraciones, permitiendo así, un diagnóstico más rápido y de mayor calidad por parte del usuario de las maquinarias involucradas.

# Capítulo 1: Fundamentación teórica

En este capítulo se presentan los principios teóricos fundamentales para comprender adecuadamente el trabajo realizado. Se abordan temas relacionados con el estado del arte de las aplicaciones para el monitoreo de máquinas industriales y el análisis de vibraciones en procesos industriales, específicamente en máquinas con ejes rotatorios. Además, se explican conceptos como el monitoreo y los sistemas de monitoreo. También se discute la Transformada Rápida de Fourier (FFT) y el mantenimiento predictivo. Finalmente, se hace un estudio y posterior selección de las tecnologías utilizadas para la confección de la propuesta.

* 1. **Estado del arte.**

Existen varias aplicaciones que permiten el monitoreo de señales provenientes de máquinas industriales y que también permiten el análisis espectral de las señales. Algunos ejemplos incluyen **DynaPredict** (Dynamox, 2023), **BK Connect** (Brüel & Kjær, 2023) y **PRTG** (PAESSLER, 2023). Estos software ofrecen soluciones para el monitoreo del estado de las máquinas y los componentes, y pueden ser utilizados para el mantenimiento predictivo.

Acerca de DynaPredict:

DynaPredict es una solución de monitoreo de la condición de las máquinas desarrollada por Dynamox. Esta solución se basa en el uso de un DynaLogger, un data logger Bluetooth con sensores de vibración y temperatura para supervisar el estado de la máquina y realizar un análisis espectral triaxial. Los parámetros medidos se muestran instantáneamente en el smartphone y su historial de datos se almacena en la plataforma web para su análisis y toma de decisiones. (Dynamox, 2023)

DynaPredict permite a los usuarios supervisar el estado de sus máquinas en tiempo real y tomar medidas preventivas para evitar fallas y tiempos de inactividad. La recolección de datos del DynaLogger está automatizada por una pasarela, el DynaGateway, desarrollada por Dynamox. Esto permite una supervisión continua y sin interrupciones del estado de las máquinas. (Dynamox, 2023)

Acerca de BK Connect:

BK Connect es un software de análisis de sonido y vibraciones creado por la compañía Brüel & Kjær. Este software ofrece una amplia variedad de aplicaciones y opciones diseñadas para tareas generales de sonido y vibraciones. Constituye un conjunto de herramientas flexible, capaz de hacer grabaciones con un solo clic o de resolver problemas complejos, que permiten desde gestionar datos hasta elaborar informes (Brüel & Kjær, 2023).

Las aplicaciones esenciales de BK Connect tienen una estructura modular, orientada a los roles de los distintos usuarios, y forman la base de un sistema flexible de medida de sonido y vibraciones. Cada aplicación puede funcionar como un conjunto de herramientas autónomo o combinarse con otras aplicaciones. Es posible combinar las aplicaciones esenciales y sus opciones con otra funcionalidad avanzada, para crear soluciones optimizadas, a la medida de su flujo de trabajo (Brüel & Kjær, 2023).

Además, BK Connect ofrece applets diseñados para realizar tareas de ensayo muy específicas. Cada applet individual es un módulo de software autónomo, adaptado a las necesidades del usuario y concebido para realizar una tarea específica (Brüel & Kjær, 2023).

Acerca de PRTG:

PRTG es un software de monitoreo que incorpora elementos de todas las áreas, para que pueda supervisar la salud, el estado y la condición de máquinas, sistemas de control, dispositivos y más en entornos de Tecnología de la Información (TI, por sus siglas en inglés) y Tecnología de Operaciones (OT, por sus siglas en inglés). Este software es compatible de fábrica con estándares y protocolos industriales comunes como OPC UA, MQTT, Modbus TCP y más. Además, permite el monitoreo de dispositivos Ethernet industriales en su entorno de OT y la visualización de su entorno en paneles que incluyen elementos de TI, OT e Internet de las cosas industrial (IIoT, por sus siglas en inglés) (PAESSLER, 2023).

PRTG también ofrece la posibilidad de extender la funcionalidad de las soluciones comunes de la industria para proporcionar una funcionalidad ampliada. Proporciona sensores nativos para Modbus TCP y RTU y permite supervisar si los clientes pueden conectarse y suscribirse a su bróker de MQTT y publicar a través de él con el sensor ida y vuelta MQTT (PAESSLER, 2023).

* 1. **Análisis por vibraciones de procesos industriales.**

El diagnóstico de maquinarias por vibraciones, como una de las tantas técnicas involucradas en el mantenimiento predictivo, se basa en el monitorizado de las vibraciones producidas por las maquinarias en funcionamiento y, debido a las conveniencias de su aplicación, constituye uno de los más preferidos por los especialistas del mantenimiento. Las técnicas de análisis de vibraciones para el diagnóstico que más se aplican son extraídas fundamentalmente del campo del procesamiento de señales (Mohd Ghazali & Rahiman, 2021).

El análisis por vibraciones es una técnica que consiste en medir y analizar las oscilaciones o movimientos alternos de ciertos puntos de una máquina o estructura, para determinar su estado o condición, así como la fuente y la gravedad de posibles fallas. El análisis por vibraciones puede ayudar a prevenir y predecir problemas como desbalance, desgaste, desalineación, defectos en rodamientos, fricción, grietas en engranajes, etc. (Mohd Ghazali & Rahiman, 2021)

Para realizar el análisis por vibraciones se utilizan instrumentos como sensores, analizadores y colectores de datos, que captan y procesan las señales de vibración en el dominio del tiempo o de la frecuencia. Estas señales pueden ser representadas como formas de onda, espectros o diagramas que contienen información útil para el diagnóstico. Existen diferentes métodos y técnicas para extraer y clasificar las características de las señales de vibración, como el análisis estadístico, la transformada rápida de Fourier (FFT), el análisis de envolvente, el análisis espectral, la transformada wavelet (WT, por sus siglas en inglés), el análisis cepstral, la transformada de Hilbert-Huang (HHT, por sus siglas en inglés), entre otros. Cada método tiene sus ventajas y desventajas, y se aplica según el tipo de máquina, la frecuencia de vibración y el tipo de falla que se quiere detectar (Mohd Ghazali & Rahiman, 2021).

En el mantenimiento predictivo, el análisis por vibraciones es una técnica que permite supervisar y diagnosticar la maquinaria rotativa para predecir posibles fallas y planificar el mantenimiento preventivo.

Como resultado del análisis de una vibración en el dominio del tiempo, a partir de la cual se calculan parámetros como RMS o Valor efectivo, PICO y PICO-PICO. De ellos el más utilizado es el RMS que está asociado a la potencia de la vibración y este se determina de forma discreta de la siguiente forma: (Marín, 1997)

El valor PICO es el valor máximo de una magnitud (aceleración, velocidad, desplazamiento) que varía en un intervalo de tiempo (Marín, 1997). El valor PICO-PICO (de un evento oscilatorio) es la diferencia algebraica entre los valores extremos de una magnitud que varía en un intervalo de tiempo (Marín, 1997). En la [Figura 1](#Figura1) se observa la representación de los parámetros mencionados anteriormente en el dominio del tiempo (Marín, 1997)

Figura 1 Parámetros característicos de una señal.

La detección del valor PICO-PICO se utiliza para mediciones de desplazamiento y los de PICO y RMS se usan para mediciones de velocidad y aceleración (Marín, 1997).

Los resultados del análisis por vibraciones se utilizan en el mantenimiento predictivo para monitorear el rendimiento y la condición de las máquinas y predecir posibles fallas. Al medir y analizar regularmente las señales de vibración de las máquinas, se pueden detectar cambios en su comportamiento que indiquen la presencia de una falla o un problema potencial. Esto permite a los técnicos de mantenimiento tomar medidas preventivas, como ajustar, reparar o reemplazar componentes antes de que fallen y causen daños mayores o interrupciones en la producción.

* 1. **Monitoreo.**

Monitoreo viene de la palabra monitor, es decir, una pantalla por la cual se pueden ver imágenes en tiempo real (Ferrer, 2023).

El monitoreo es el acto de supervisar los cambios en el estado y el flujo de datos en un sistema. Hay dos formas de hacerlo: proactiva y reactiva. La forma proactiva implica observar indicadores visuales, como series de tiempo y tableros de información, lo que se conoce como monitoreo. La forma reactiva implica enviar notificaciones automáticas a los operadores para informarles sobre cambios en el sistema, lo que se conoce como alerta (de Murga Aguiar, 2020).

Las alertas son la capacidad de un sistema de monitoreo para detectar y notificar a los operadores sobre un evento importante que provoca un cambio significativo en el estado. Estas notificaciones, conocidas como alertas, pueden adoptar diversas formas, como correo electrónico, SMS, mensajes instantáneos o llamadas telefónicas (de Murga Aguiar, 2020).

En general, el monitoreo consiste en la observación del curso de uno o más parámetros para detectar eventuales anomalías.

* 1. **Sistemas de monitoreo.**

Un sistema de monitoreo es responsable de hacer un seguimiento del estado del sistema completo, incluyendo la infraestructura y otros subsistemas, para asegurar la fiabilidad y estabilidad de los servicios que provee. Consta de una serie de componentes de software que realizan mediciones y recolectan, almacenan e interpretan los datos monitoreados. El sistema está optimizado para almacenar de manera eficiente y para generar métricas de monitoreo, que posteriormente son mostradas en series de tiempo y cuyos puntos de datos son analizados con la finalidad de emitir alertas. Observa y sigue en el tiempo las operaciones y actividades de los usuarios, aplicaciones y servicios de red, registrando todas las acciones y alertando si identifica alguna violación o brecha que resulte en un comportamiento anormal (de Murga Aguiar, 2020).

La mayoría de los sistemas de monitoreo tienen una estructura parecida. El procedimiento inicia con la recepción de datos, después el cliente recolecta y transmite los datos al sistema de monitoreo mediante una interfaz especializada. El sistema guarda estos datos en métricas y remite los nuevos puntos de datos para valorar las condiciones de los umbrales. Cuando se percibe que un umbral ha sido transgredido, se manda una notificación al operador acerca del fallo. La [Figura 2](#Figura2) ilustra las interacciones entre los componentes de un sistema de monitoreo (de Murga Aguiar, 2020).



Figura 2 Interacciones entre los componentes de la arquitectura de los sistemas de monitoreo.

* 1. **Transformada rápida de Fourier (FFT).**

Las funciones de análisis se pueden calcular mediante las integrales que la definen, pero el núcleo principal del análisis de señales y del análisis de sistemas se realiza por el análisis por Fourier sustentadas por las siguientes ecuaciones conocidas como transformada discreta de Fourier (DFT) (Marín, 1997):

(1.1)

(1.2)

Para n=0, 1,…, N-1; k=0, 1,…, N-1.

Siendo N el número de muestras discretas de un segmento de las señal vibroacústica. En el caso de las vibraciones como señales a procesar, y(n) es real y Y(k) es compleja. Al aplicar la transformada discreta de Fourier a un número N de muestras se muestra en la [Figura 3](#Figura3) (Marín, 1997).



Figura 3 Representación compleja de la transformada discreta de Fourier (DFT).

Hasta el año 1965 aproximadamente se usó la DFT en computadoras con programas que tenían que ejecutar operaciones de cálculo, siendo N el número de muestras discretas en un bloque de datos. Para ese mismo año se da a conocer la FFT, popularizada por James Willian Cooley de IBM y John W. Turkey de Bell Laboratories requiriendo solamente operaciones para transformar un bloque de N datos por mucha diferencia mejor que la DFT (Marín, 1997) (Davuluru, Hettiarachchi, & Balster, 2022).

En resumen la FFT es un algoritmo eficiente para calcular la DFT y su inversa. La DFT es una herramienta matemática para el análisis de señales en el dominio de la frecuencia, pero su cálculo directo puede ser muy costoso en términos de tiempo y recursos computacionales. La FFT reduce significativamente la complejidad computacional del cálculo de la DFT, lo que la hace más adecuada para aplicaciones en tiempo real.

* 1. **Mantenimiento predictivo.**

Para entender el concepto de mantenimiento predictivo, se puede consultar el diccionario Oxford, donde el término "Predictivo" se define como "relacionado con la capacidad de prever el futuro", mientras que "mantenimiento" se refiere a "la acción de mantener algo en buen estado mediante revisiones o reparaciones periódicas". Al combinar estos dos conceptos, podemos definir el mantenimiento predictivo como "la acción y capacidad de mantener algo en buenas condiciones para prever lo que sucederá en el futuro" (Ramos Cosio, 2020).

El uso del mantenimiento predictivo ofrece varias ventajas, como (Ramos Cosio, 2020):

1. Reducción de los tiempos de parada.
2. Prevención del funcionamiento de maquinarias en condiciones de riesgo.
3. Disminución de la interrupción del servicio en otros equipos en funcionamiento, evitando distorsiones en otras áreas.
4. Optimización de la gestión del área de mantenimiento.
5. Verificación y diagnóstico basados en los resultados de las mediciones del estado actual del equipo.
6. Determinación precisa del tiempo máximo de operación sin riesgo de fallas imprevistas.
7. Facilitación de la toma de decisiones sobre la parada de algún equipo en funcionamiento.
8. Ahorro significativo en tiempo y dinero al evitar paradas por mantenimiento preventivo en instalaciones en funcionamiento.
9. Aumento de la confiabilidad en equipos, materiales e instalaciones en general.

En resumen el mantenimiento predictivo da la capacidad de realizar mantenimiento donde puede haber un problema o falla en el futuro, para así obtener ventajas y evitar problemas de diferente índole.

* 1. **Análisis de las tecnologías.**

Después de decidir que se necesita crear un nuevo software o añadir una nueva funcionalidad, es esencial realizar un examen detallado de las tendencias y una investigación del estado actual de las distintas tecnologías relacionadas con el desarrollo de la propuesta. Este análisis debe ser exhaustivo, cubriendo todos los aspectos a considerar de componentes y tecnologías adecuadas para las tareas que demanda la aplicación.

* + 1. **Lenguaje de programación.**

Pascal es un lenguaje de programación creado por el profesor suizo Niklaus Wirth entre los años 1968 y 1969, y publicado en 1970. Su objetivo era crear un lenguaje que facilitara el aprendizaje de programación a sus alumnos, utilizando la programación estructurada y estructuración de datos (Cantú, 2003) (Wikipedia, 2023).

Algunas características de Pascal son (Wikipedia, 2023) (Cantú, 2003):

1. Es un lenguaje de programación fuertemente tipado, lo que significa que el tipo de dato de todas las variables debe ser declarado previamente para que su uso quede habilitado.
2. El código está dividido en porciones fácilmente legibles llamadas funciones o procedimientos, lo que facilita la utilización de la programación estructurada en oposición al antiguo estilo de la programación monolítica.
3. Utiliza el símbolo `:=` para la asignación en vez de `=`, lo que ayuda a prevenir errores comunes donde se utiliza el símbolo de igualdad para comparar valores en lugar del comparador `==`.
4. Sus programas tienen definidas dos partes: declarativa y ejecutiva. En la primera debe aparecer todo lo que se usará en la segunda, lo que ayuda a prevenir errores y mejorar la legibilidad del código.

Algunas de las ventajas de Pascal son (Lenguaje de programación Pascal actualizado a 2023, 2023):

1. Promueve un método disciplinado y elegante a la hora de programar, con programas bien organizados, claros y relativamente libres de errores.
2. Es un lenguaje orientado para cualquier tipo de ordenador, ya sea un gran sistema de cómputo o una simple computadora personal de sobremesa.
3. Es un lenguaje que casi se considera pseudo-código, lo que facilita la comprensión y explicación del código.
4. Fomenta los buenos principios de una buena práctica de programación, como la integridad, la simplicidad, la modularidad y la generalidad.

Para la solución del objetivo general se va a usar la evolución de Pascal, llamada Object Pascal, una versión que ofrece las facilidades del paradigma de la programación orientada a objeto y además que la solución del proyecto tiene antecedentes de código ya escrito y se va a poder reutilizar.

* + 1. **Entorno de desarrollo integrado (IDE).**

Para la creación de la aplicación se va a usar el IDE Embarcadero Delphi 11 versión 28.0.47991.2819, algunas características importantes son (Delphi Características, 2023) (Delphi 11.3, 2023):

1. Esta versión de Delphi incluye soporte para Markdown y un simulador de iOS en macOS ARM 64, lo que permite a los desarrolladores utilizar el lenguaje de marcado Markdown y probar aplicaciones para iOS en dispositivos macOS con procesadores ARM 64.
2. También migra el depurador de Delphi Linux a LLDB, un depurador de código abierto que ofrece características avanzadas y puede mejorar la experiencia de depuración para los desarrolladores que utilizan Delphi para crear aplicaciones para Linux.
3. Además, esta versión incluye mejoras en la usabilidad, el rendimiento y la estabilidad en varias áreas, como el IDE, los compiladores, las cadenas de herramientas, la UX, la RTL, la base de datos y las bibliotecas de Internet.
4. Permite la construcción rápida de aplicaciones multiplataforma con una base de código común para Windows, Android, iOS, macOS y Linux.
5. Ofrece Object Pascal moderno con bibliotecas de componentes y compiladores nativos para múltiples plataformas.

En resumen significa que esta versión de Delphi presenta la posibilidad de trabajo con el motor base de datos SQLite, permite que la aplicación sea multiplataforma y llegue a la mayor cantidad posible de usuarios finales. Además, Delphi proporciona una gran cantidad de componentes visuales y no visuales disponible para el desarrollo de aplicaciones.

* + 1. **Motor de base de datos.**

Se ha elegido a SQLite debido a que, este es una biblioteca escrita en lenguaje C que implementa un motor de base de datos SQL pequeño, rápido, autónomo, confiable y con muchas funciones. Es de los motores de bases de datos más usados en el mundo y está integrado en todos los teléfonos móviles y la mayoría de las computadoras, además de venir incluido en innumerables otras aplicaciones que las personas usan todos los días. La versión a usar es la más reciente de SQLite en el momento la 3.42.0. (SQLite, What Is SQLite?, 2023).

SQLite puede ejecutarse en múltiples plataformas. Almacena toda la base de datos (definiciones, tablas, índices y los datos en sí) como un solo archivo multiplataforma en una máquina host. El formato de archivo de la base de datos es multiplataforma, lo que significa que puedes copiar libremente una base de datos entre sistemas de 32 y 64 bits o entre arquitecturas big-endian y little-endian. Esto hace que SQLite sea una opción conveniente para el desarrollo de aplicaciones multiplataforma (SQLite, Single-file Cross-platform Database, 2023) (SQLite, About SQLite, 2023) (SQLite, System Requirements For SQLite, 2023).

SQLite no es directamente comparable con motores de bases de datos SQL cliente/servidor como MySQL, Oracle, PostgreSQL o SQL Server, ya que SQLite intenta resolver un problema diferente. Los motores de bases de datos SQL cliente/servidor se esfuerzan por implementar un repositorio compartido de datos empresariales. Hacen hincapié en la escalabilidad, la concurrencia, la centralización y el control. SQLite, por otro lado, se esfuerza por proporcionar almacenamiento de datos local para aplicaciones y dispositivos individuales. SQLite hace hincapié en la economía, la eficiencia, la fiabilidad, la independencia y la simplicidad. SQLite no compite con las bases de datos cliente/servidor (SQLite, Appropriate Uses For SQLite, 2023).

En conclusión se puede afirmar que SQLite cumple las características de ser rápido, autónomo, eficiente, compacto y permite que la gestión de bases de datos en múltiples plataformas.

**Conclusiones parciales**

Con la culminación de este capítulo se puede arribar a las siguientes conclusiones:

1. Los resultados obtenidos en el análisis por vibraciones al ser usados en el mantenimiento predictivo previene fallas en las maquinarias industriales.
2. Al realizar el análisis por vibraciones mediante la FFT se realizan menos operaciones de cálculo por parte del software, trayendo consigo mayor velocidad.
3. Los sistemas de monitoreo son utilizados para la compilación de datos, su procesamiento y presentación de los mismos.
4. Utilizar las tecnologías presentadas hacen que se cumplan los requisitos que tiene que cumplir la aplicación con eficiencia, rapidez y compatibilidad.

# Capítulo 2: Solución propuesta

En este capítulo se aborda el diseño que da solución al problema antes mencionado en la introducción. Para ello se hace uso de diagramas UML y se realiza la modelación del negocio, captura de requisitos y diseño del sistema. En la modelación del negocio se exponen conceptos y entidades que serán presentados en una representación del dominio. A continuación, se realiza un análisis de los requerimientos que debe cumplir el sistema y se presentan los casos de usos que dan solución a los requisitos antes mencionados. Además, se detalla el diseño de la arquitectura para la implementación de la solución, haciendo énfasis en el diagrama de clases, diseño de la base de datos y diagrama de despliegue.

2.1 Reglas del negocio

Una regla de negocio es una afirmación que establece o restringe ciertas partes de un negocio. Por lo tanto, para que el sistema funcione adecuadamente, es esencial establecer un conjunto de reglas y limitaciones sobre cómo se utiliza el software (Ross, 2013). Las reglas del negocio fundamentales se muestran a continuación:

* El receptor crea la señal digital a partir de ser capturada la señal analógica.
* El receptor transmite la señal digital en tiempo real a la aplicación.

2.2 Requisitos.

La recopilación de requisitos permite documentar las demandas del cliente y convertirlas en un modelo que describe las funciones del sistema. A continuación, se presentan los requisitos funcionales y no funcionales del sistema (de Murga Aguiar, 2020).

2.2.1 Requisitos funcionales.

Los requisitos funcionales son las características que definen los comportamientos, capacidades y elementos esenciales que forman parte de un sistema (Pressman, 2010). A continuación, se detallan los requisitos funcionales definidos:

* RF1: Mostrar la señal proveniente del equipo monitoreado.
* RF2: Mostrar el espectro de la señal.
* RF3: Mostrar el análisis de tendencia de los parámetros característicos.
* RF4: Almacenar la señal con sus parámetros característicos.
* RF5: Gestionar las rutas, máquinas y usuarios de la aplicación.

2.2.2 Requisitos no funcionales.

Un requisito no funcional, también llamado atributo de calidad, establece los criterios que se utilizan para evaluar el funcionamiento de un sistema, en lugar de sus comportamientos particulares. En esencia, se refiere a todos los requisitos que no describen la información que se debe almacenar ni las funciones comerciales que se deben llevar a cabo (I. Jacobson & Rumbauch, 2000). A continuación, se detallan los requisitos no funcionales definidos:

**Seguridad**

Los usuarios deben estar autenticados para poder utilizar la herramienta. Esta autenticación será realizada a través de un nombre de usuario y contraseña creados a partir de la misma.

**Usabilidad**

La usabilidad se refiere a cómo el diseño sirve como comunicación entre el objeto y el usuario y cómo optimizar ese conducto de comunicación para hacer que la experiencia de usar el objeto sea placentera. Las reglas son simples: hacer las cosas visibles, explotar las relaciones naturales que acoplan la función y el control, y hacer un uso inteligente de las restricciones. El objetivo es guiar al usuario sin esfuerzo hacia la acción correcta en el control correcto en el momento adecuado (Norman, 2018).

La usabilidad se define por 5 componentes, los cuales son mencionados a continuación, y argumentados para el caso específico del proyecto (Nielsen, 2012):

* Aprendizaje: El especialista debe ser capaz de usar el sistema después de una explicación y práctica de aproximadamente una 45 minutos.
* Eficiencia: Un buen diseño de interfaces permite a los usuarios realizar más tareas en menos tiempo, aumentando así su eficiencia. Para lograr esto:
  + Se reducirá al máximo posible la cantidad de pasos que el usuario deba realizar para llevar a cabo una tarea en el sistema.
  + Se añadirán mecanismos para que el usuario sepa en todo momento donde se encuentra y que está ocurriendo en la aplicación.
  + Se ofrecerán atajos o comandos rápidos para los usuarios expertos que quieran ahorrar tiempo.
  + Se informará en todo momento al usuario cada acción que esté haciendo, para que este sepa donde se encuentra y que está pasando en la aplicación.
* Memorabilidad (Facilidad de recordar): Después de un tiempo que el usuario haya dejado de usar la aplicación este debe de recordar cómo usarlo sin mucha demora, para lograr esto se implementó lo siguiente:
  + Se logrará la consistencia en cuanto a patrones y estilos familiares, para que el usuario asocie comportamientos similares en distintas operaciones.
  + Se utilizarán íconos sobre los controles.
  + Se proporcionará ayuda clara y accesible.
  + Se evitará cambiar las funciones o la interfaz sin avisar a los usuarios.
* Eficacia: Se debe menguar el número y la gravedad de los errores que el usuario pueda hacer:
  + Validando la entrada de datos mediante la interfaz.
  + La selección se logra mediante listas desplegables.
  + Se informa al usuario con mensajes claros en los cuales se pide confirmación, para evitar acciones de error e inconsistencia de los datos.
  + Se permiten las acciones de deshacer y rehacer.
  + Se proporcionan mensajes de error claro y constructivo que indiquen como solucionar los problemas.
* Satisfacción: Se debe lograr que el usuario sienta gusto de usar la aplicación, esto se logra mediante:
  + El diseño de las pantallas son fáciles de aprender a usar, atractivas e intuitivas, cumpliendo con las expectativas y objetivos de uso del usuario.
  + Antes y durante del diseño de la aplicación se tuvo en cuenta las opiniones de los futuros usuarios.
  + Antes de realizar actualizaciones se van a realizar encuestas, entrevistas o cuestionarios a los usuarios.

2.3 Modelo del Dominio.

Un Modelo del dominio es un modelo conceptual de alto nivel que define objetos físicos y abstractos en un área de interés para el proyecto. Se puede usar para documentar relaciones entre ellos y responsabilidades de clases conceptuales (Systems, 2010).

2.3.1 Definición de los conceptos clave.

Se presentan y describen a continuación, los conceptos clave que explican el contexto en el que se utiliza la herramienta:

**Tabla 1 Definición de las entidades.**

|  |  |
| --- | --- |
| Entidad del negocio | Descripción |
| Equipo | Equipo que contiene el software de monitoreo. |
| Receptor del Equipo | Dispositivo que captura la señal analógica, realiza la conversión analógica-digital y se encarga de transmitirla al Equipo. |
| Receptor | Controla el Receptor del Equipo a través de un Driver. |
| Driver | Entidad encargada de controlar el flujo de información entre el Receptor y el Receptor del Equipo. |
| Señal digital | Señal enviada por el Receptor del Equipo. |
| Señal analógica | Señal que se encuentra en el espectro, la cual es capturada por el Receptor del Equipo. |
| Gráfico | Elemento en el que se representa la señal digital. |
| Ruta | Camino por el cual se realizan las mediciones para ejecutar eficiente y rápidamente el monitoreo de una línea de máquinas. |
| Máquina | Máquina que realiza una operación y genera señales vibratorias. |

2.3.2 Representación del modelo del dominio.

# Capítulo 3: Validación de la solución

# Conclusiones

# Recomendaciones

# Referencias bibliográficas

¿Qué es la Industria 4.0 y cómo funciona? | IBM. (July de 2023). *¿Qué es la Industria 4.0 y cómo funciona? | IBM*. Obtenido de https://www.ibm.com/es-es/topics/industry-4-0

Brüel & Kjær. (July de 2023). Software de análisis de señales. *Software de análisis de señales*. Obtenido de https://www.bksv.com/es/analysis-software/data-acquisition-software/bk-connect/signal-analysis

Cantú, M. (July de 2003). *Essential Pascal.* (M. C. Books, Ed.) Marco Cantú Books.

Davuluru, V. S., Hettiarachchi, D. L., & Balster, E. (2022). Performance Analysis of DFT and FFT Algorithms on Modern GPUs.

de Murga Aguiar, G. A. (September de 2020). *Herramienta de monitoreo y control en tiempo real de los nodos HEATS-RT.* candthesis, Universidad Tecnológica de La Habana José Antonio Echeverría, La.

Delphi 11.3. (July de 2023). *Delphi 11.3*. Obtenido de https://lenguajesdeprogramacion.net/pascal-y-delphi/

Delphi Características. (July de 2023). *Delphi Características*. Obtenido de https://www.embarcadero.com/es/products/delphi/features/delphi

Dynamox. (July de 2023). Aumente la disponibilidad con Dynamox. *Aumente la disponibilidad con Dynamox*. Obtenido de https://dynamox.net/es

Ferrer, J. (July de 2023). Definición de Monitoreo. *Definición de Monitoreo*. Obtenido de https://enciclopedia.net/monitoreo/

I. Jacobson, & Rumbauch, J. (2000). *The Unified Software Development Process.* Madrid, España: Pearson Educación.

Lenguaje de programación Pascal actualizado a 2023. (July de 2023). *Lenguaje de programación Pascal actualizado a 2023*. Obtenido de https://lenguajesdeprogramacion.net/pascal-y-delphi/

Marín, E. P. (1997). *La medición y el análisis de vibraciones en el diagnóstico de máquinas rotatorias.* (D. de Ingeniería de las vibraciones y diagnóstico, Ed.) Centro de Estudios Innovación y Mantenimiento.

Ministerio de Relaciones Exteriores de Cuba. (July de 2023). Informe de Cuba en virtud de la resolución 75/289 de la Asamblea General de las Naciones Unidas, titulada “Necesidad de poner fin al bloqueo económico, comercial y financiero impuesto por los Estados Unidos de América contra Cuba”. *Informe de Cuba en virtud de la resolución 75/289 de la Asamblea General de las Naciones Unidas, titulada “Necesidad de poner fin al bloqueo económico, comercial y financiero impuesto por los Estados Unidos de América contra Cuba”*. Calle Calzada, No. 360, Vedado. Plaza de la Revolución. La Habana, Cuba. Obtenido de https://cubaminrex.cu/es/informe-de-cuba-en-virtud-de-la-resolucion-75289-de-la-asamblea-general-de-las-naciones-unidas

Mohd Ghazali, M. H., & Rahiman, W. (2021). Vibration analysis for machine monitoring and diagnosis: a systematic review. *Shock and Vibration, 2021*, 1–25.

Norman, D. (2018). *The Design of Everyday Things.*

PAESSLER. (July de 2023). Supervisión de procesos industriales. *Supervisión de procesos industriales*. Obtenido de https://www.paessler.com/es/industrial-it-monitoring

Pressman, R. S. (2010). *Ingeniería del Software. Un enfoque práctico* (7 ed.).

Ramos Cosio, F. J. (2020). Revisión de los beneficios con la aplicación del mantenimiento predictivo para la mejora en la confiabilidad de las máquinas industriales en el sector industrial en los últimos diez años.

Ross, R. G. (April de 2013). *Business rule concepts: Getting to the point of knowledge* (4th ed.). Business Rule Solutions, LLC.

SQLite. (July de 2023). About SQLite. *About SQLite*. Obtenido de https://www.sqlite.org/about.html

SQLite. (July de 2023). Appropriate Uses For SQLite. *Appropriate Uses For SQLite*. Obtenido de https://www.sqlite.org/whentouse.html

SQLite. (July de 2023). Single-file Cross-platform Database. *Single-file Cross-platform Database*. Obtenido de https://www.sqlite.org/onefile.html

SQLite. (July de 2023). System Requirements For SQLite. *System Requirements For SQLite*. Obtenido de https://www.sqlite.org/draft/sysreq.html

SQLite. (July de 2023). What Is SQLite? *What Is SQLite?* Obtenido de https://www.sqlite.org/index.html

Wikipedia. (July de 2023). Pascal (lenguaje de programación) — Wikipedia, La enciclopedia libre. *Pascal (lenguaje de programación) — Wikipedia, La enciclopedia libre*. Obtenido de https://es.wikipedia.org/w/index.php?title=Pascal\_(lenguaje\_de\_programaci%C3%B3n)&oldid=151221731

Zambrano-Reyes, A., Nossov, V. R., & Gómez-Mancilla, J. C. (s.f.). Análisis de la Respuesta Vibratoria de Ejes Fisurados sobre Chumaceras Lubricadas para Control y Atenuación de Vibraciones en Máquinas Rotatorias.