



**EDUCACIÓN**  
SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO®  
Campus Nogales



# DESARROLLO DE SOFTWARE BOC-DWT PARA REGISTRO DE TIEMPO CAÍDO UTILIZANDO EL ESTÁNDAR ISA-95

**TESIS**

QUE PARA OBTENER EL GRADO DE  
**MAESTRO EN SISTEMAS COMPUTACIONALES**

PRESENTA  
**JESÚS AARON CORTEZ JACOBO**

DIRECTOR  
**M.C. JESÚS RAÚL CRUZ RENTERÍA**

H. NOGALES, SONORA, MÉXICO.

JUNIO DE 2021.



Instituto Tecnológico de Nogales  
Departamento de Comunicación y Difusión

H. Nogales Son., 15/junio/2021  
Oficio No. DEPI/084/2021.

**REYNALDO GUTIÉRREZ GUTIÉRREZ**  
**JEFE DE LA DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN**  
**PRESENTE.**

Por este medio le comunicamos a Usted que el trabajo de Tesis denominado: **"DESARROLLO DE SOFTWARE BOC-DWT PARA REGISTRO DE TIEMPO CAÍDO UTILIZANDO EL ESTÁNDAR ISA-95"**, que presentó el alumno **JESÚS AARON CORTEZ JACOBO**, con número de control 17341001, candidato a obtener el grado de **Maestro en Sistemas Computacionales**, ha sido revisado por los miembros del Comité Tutorial y cubiertas las observaciones realizadas, se Autoriza su Impresión y se Acepta para su Evaluación en la presentación del Examen de Grado.

Agradeciendo de antemano el apoyo brindado al presente, le reitero mi consideración distinguida.

**ATENTAMENTE**

Excelencia en Educación Tecnológica.  
"La Ciencia y la Tecnología para la Liberación del Hombre".

**ZINDI SANCHEZ HERNANDEZ**  
**REVISOR**

**SIGIFREDO GARCÍA ALVA**  
**REVISOR**

**JESÚS RAÚL CRUZ RENTERÍA**  
**REVISOR**



RAA-079  
Fecha del certificado: 15/06/2021 10:18  
Fecha de certificación: 15/06/2021 10:18





Instituto Tecnológico de Nogales  
Departamento de Comunicación y Difusión

H. Nogales Sonora, 15/junio/2021  
Oficio No. DEPI/085/2021.

**JESÚS AARON CORTEZ JACOBO**  
**CANDIDATO A OBTENER EL GRADO DE MAESTRO EN SISTEMAS COMPUTACIONALES**  
**PRESENTE.**

*De acuerdo con el Reglamento de Titulación del Sistema Nacional de Institutos Tecnológicos de la Secretaría de Educación Pública y habiendo cumplido con todas las indicaciones que el Comité Tutorial realizó, con respecto a su Tesis titulada: "DESARROLLO DE SOFTWARE BOC-DWT PARA REGISTRO DE TIEMPO CAÍDO UTILIZANDO EL ESTÁNDAR ISA-95.", la División de Estudios de Posgrado e Investigación Autoriza su Impresión.*

*Agradeciendo de antemano el apoyo brindado al presente, le reitero mi consideración distinguida.*

**ATENTAMENTE**  
*Excelencia en Educación Tecnológica*  
*"La Ciencia y la Tecnología para la Liberación del Hombre"*

**REYNALDO GUTIÉRREZ GUTIÉRREZ**  
**JEFE DE LA DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN**



**INSTITUTO TECNOLÓGICO DE NOGALES**  
DIVISIÓN DE ESTUDIOS DE,  
POSGRADO E INVESTIGACIÓN

ccp. Archivo

RGG/cmmj



Foro 4-018  
Folio de verificación oficial 2021-06-08  
Folio de verificación oficial 2021-06-08



**DESARROLLO DE SOFTWARE BOC-DWT PARA REGISTRO DE  
TIEMPO CAIDO UTILIZANDO EL ESTÁNDAR ISA-95**

## RESUMEN

Con el paso del tiempo las empresas de manufactura agregan automatización por software de manera gradual a sus procesos, persiguiendo el objetivo de reducir costos utilizando la integración de aplicaciones que sustituyen procesos manuales, acelerando el tiempo de ejecución de las tareas y eliminando posibles errores humanos que pueden cometerse a la hora de trabajar de manera manual.

En la presente tesis se ejemplifica una buena práctica de desarrollo de software para el registro de mantenimiento correctivo utilizando ISA-95, que es el estándar internacional para la integración de sistemas en manufactura. El propósito del software *Bard Operación Center – Downtime (BOC-DWT)* o Sistema de registro de mantenimiento correctivo, es registrar información de mantenimiento correctivo generada por averías ocurridas en máquinas de producción localizadas en las diferentes áreas de producción de la empresa maquiladora BD Bard.

## **ABSTRACT**

Software systems are used in manufacturing facilities in order to automate their productive operations, following the objective of reducing costs by using the integration of applications that replace manual processes, accelerating the execution time of tasks and eliminating possible human errors that can be committed when working manually.

This thesis exemplifies a good software development practice for the corrective maintenance record using ISA-95, which is the international standard for the integration of systems in manufacturing. The purpose of the software Bard Operation Center - Downtime (BOC-DWT) or Corrective Maintenance Record System, is to record corrective maintenance information generated by breakdowns in production machines located in the different production areas of the BD Bard manufacturing company.

## **AGRADECIMIENTOS**

Agradecimientos:

Le agradezco a Dios por haberme acompañado y guiado a lo largo de mi carrera.

Le agradezco a la compañía BD Bard, compañeros de trabajo y amigos por todo su apoyo y por permitirme desarrollar mi proyecto de tesis en las instalaciones de la compañía.

Le agradezco a mis padres (por apoyarme y escucharme).

Le agradezco a mi director de tesis Jesús Raúl Cruz Rentería, maestros y compañeros de la Maestría en la DEPI (por su ayuda y apoyo).

Gracias . . .

## ÍNDICE DE CONTENIDO

RESUMEN .....	v
ABSTRACT.....	vi
AGRADECIMIENTOS.....	vii
ÍNDICE DE CONTENIDO.....	viii
ÍNDICE DE TABLAS.....	x
ÍNDICE DE FIGURAS .....	xi
INTRODUCCIÓN .....	1
CAPÍTULO 1: PLANTEAMIENTO DEL PROBLEMA .....	3
1.1. Antecedentes. ....	3
1.2. Estado del campo del arte.....	13
1.3. Estándar Internacional ISA-95 .....	18
1.4. Definición de la situación actual del problema .....	27
1.5. Definición del problema .....	29
1.6. Justificación .....	29
1.7. Solución propuesta.....	31
1.8. Requerimientos.....	31
1.9. Objetivos.....	33
1.10. Limitaciones y Delimitaciones .....	34
1.11. Método.....	36
1.12. Proceso de Desarrollo Unificado .....	37
CAPÍTULO 2: MARCO TEÓRICO.....	41
2.1. Estándar ISA-95.....	41
2.2. BOC-DWT.....	55
2.3. Metodología de Desarrollo Unificado.....	56
2.4. Iterativo e Incremental .....	59
2.5. Fases.....	60

2.6.	Fase de Inicio .....	60
2.7.	Fase de Elaboración .....	60
2.8.	Fase de Construcción.....	61
2.9.	Fase de Transición.....	61
2.10.	Disciplinas .....	62
2.11.	Modelado de Negocio .....	62
2.12.	Requerimientos .....	62
2.13.	Análisis y Diseño.....	62
2.14.	Implementación .....	63
2.15.	Pruebas .....	63
2.16.	Artefactos.....	63
2.17.	Requerimientos Funcionales.....	64
2.18.	Casos de Uso.....	64
2.19.	Diagrama de Clases .....	65
2.20.	KPIs .....	65
2.21.	Software de 3 capas .....	66
<b>CAPÍTULO 3: DESARROLLO DE BOC-DWT .....</b>		<b>68</b>
3.1.	Análisis y Diseño de BOC-DWT.....	68
3.2.	Fase de Inicio .....	69
3.3.	Fase de Elaboración .....	84
<b>CAPÍTULO 4: DESARROLLO DEL SOFTWARE BOC-DWT.....</b>		<b>93</b>
4.1.	Construcción.....	93
4.2.	Transición.....	102
<b>CAPÍTULO 5: Resultados y Conclusiones.....</b>		<b>107</b>
5.1.	Resultados.....	107
5.2.	Conclusiones y Aprendizajes .....	109
5.3.	Trabajos Futuros.....	109
<b>REFERENCIAS BIBLIOGRÁFICAS.....</b>		<b>111</b>

## ÍNDICE DE TABLAS

<b>Tabla 1.1 Máquinas y Personal por Unidad de Negocio (Bard, 1995).....</b>	<b>3</b>
<b>Tabla 1.2 Evolución de la tecnología en manufactura. ....</b>	<b>8</b>
<b>Tabla 3.1 Fases de Construcción del Proyecto .....</b>	<b>69</b>
<b>Tabla 3.2 Requerimientos funcionales de Equipment Specification. ....</b>	<b>73</b>
<b>Tabla 3.3 Requerimientos funcionales de Equipment Specification Property... </b>	<b>74</b>
<b>Tabla 3.4 Requerimientos funcionales de Personnel Specification Property....</b>	<b>74</b>
<b>Tabla 3.5 Requerimientos funcionales de Personnel Model. ....</b>	<b>75</b>
<b>Tabla 3.6 Requerimientos funcionales de Parameters Specification. ....</b>	<b>75</b>
<b>Tabla 3.7 Requerimientos funcionales de Equipment Model. ....</b>	<b>76</b>
<b>Tabla 3.8 Requerimientos funcionales de Personnel Specification. ....</b>	<b>77</b>
<b>Tabla 3.9 Casos de Uso de Alto Nivel (UC1) .....</b>	<b>77</b>
<b>Tabla 3.10 Casos de Uso de Alto Nivel (UC2) .....</b>	<b>78</b>
<b>Tabla 3.11 Work Definition Attributes.....</b>	<b>83</b>
<b>Tabla 3.12 Formato de caso de prueba de funcionamiento de prototipo.....</b>	<b>84</b>
<b>Tabla 3.13 Tabla de correspondencia del requerimiento R2. ....</b>	<b>90</b>
<b>Tabla 4.1 Fases de construcción del proyecto. ....</b>	<b>93</b>
<b>Tabla 4.2 Formato de Prueba de Captura de código de falla.....</b>	<b>103</b>
<b>Tabla 4.3 Formato de Prueba de Registro de Mantenimiento Correctivo.....</b>	<b>106</b>

## ÍNDICE DE FIGURAS

Figura 1.1 Entradas y salida típica de un proceso de registro de mantenimiento correctivo.....	6
Figura 1.2 Diagrama de Mantenimiento correctivo. ....	7
Figura 1.3 Etapas en la evolución de la integración <i>CIM</i> .....	11
Figura 1.4 Integración Empresarial como cuarto nivel de integración.....	13
Figura 1.5 Registro de tiempo caído a lo largo de los años en BD Bard.....	15
Figura 1.6 Diagrama de Causa y Efecto Para Proceso de Registro de Mantenimiento Correctivo. ....	27
Figura 1.7 Área de producción para la realización de la prueba de BOC-DWT. ....	35
Figura 1.8 Modelo de metodología del Proceso de Desarrollo Unificado para el desarrollo de <i>BOC-DWT</i> .....	37
Figura 2.1 Jerarquía funcional de los niveles de automatización de ISA-95.....	41
Figura 2.2 Jerarquía del Equipo de ISA-95.....	43
Figura 2.3 Modelo de actividades del manejo de operaciones de mantenimiento. ....	46
Figura 2.4 Metodología RUP.....	57
Figura 2.5 Arquitectura a 3 capas. ....	67
Figura 3.1 Análisis de Riesgo de Modelo de Negocio.....	70
Figura 3.2 Operations Definition Model.....	80
Figura 3.3 Definition Model Propuesto.....	81
Figura 3.4 Diagrama de clases (vista de análisis) de BOC-DWT.....	86
Figura 3.5 Diagrama de Clases (vista de diseño) de BOC-DWT.....	88
Figura 3.6 Diagrama de Colaboración. ....	89
Figura 3.7 Interfaz de Usuario de Página Principal. ....	91
Figura 3.8 Diagrama de Base de datos BOC-DWT. ....	92
Figura 4.1 Arquitectura de <i>BOC-DWT</i> .....	95
Figura 4.2 Diagrama de Base de datos BOC-DWT .....	96
Figura 4.3 Consulta de Datos de Máquina por Código de Falla .....	97
Figura 4.4 Consulta de Datos de Máquina por Área.....	97

<b>Figura 4.5 Diagrama de Clases (vista de diseño) de BOC-DWT .....</b>	<b>98</b>
<b>Figura 4.6 Código Fuente de Equipment Model .....</b>	<b>99</b>
<b>Figura 4.7 Conexión entre BOC-DWT y la Base de Datos.....</b>	<b>100</b>
<b>Figura 4.8 Pantalla Equipment Model.....</b>	<b>101</b>
<b>Figura 4.9 Registro de Código de Falla. ....</b>	<b>104</b>
<b>Figura 4.10 Registro de Área.....</b>	<b>104</b>
<b>Figura 4.11 Capturando Registro de Mantenimiento Correctivo.....</b>	<b>105</b>
<b>Figura 5.1 Historial de Fallas y Exportación de Datos a Excel.....</b>	<b>108</b>

## INTRODUCCIÓN

A lo largo de los años la automatización de los sistemas (de software) ha ido avanzando enormemente, haciendo procesos más fáciles, rápidos, con pocos errores y por lo tanto generando ahorros de tiempo y monetarios. Pero no todas las empresas cuentan con todos sus procesos automatizados.

En la presente tesis se trabaja sobre un software, *BOC-DWT (Bard Operation Center - Downtime)*, basado en el estándar internacional ISA-95, para el registro del mantenimiento correctivo de las averías de las máquinas de producción. Lo anterior se propone como una buena práctica de desarrollo de software para el registro del mantenimiento correctivo de manera que provea una estructura clave para soportar los cambios avanzados de automatización a los que se enfrenta BD Bard, una empresa dedicada a la manufactura de productos para el cuidado de la salud, ubicada en la ciudad de Nogales, Sonora, México.

El objetivo principal es desarrollar un software para el registro de mantenimiento correctivo basado en el estándar ISA-95, que sea capaz de registrar los eventos de mantenimiento correctivo generados a partir de las averías presentadas en las máquinas de manufactura de las áreas de producción.

Para mejorar el proceso manual que actualmente se utiliza, se procuraron los siguientes objetivos específicos para BOC-DWT:

- Desarrollar una interfaz fácil de utilizar para el usuario, lo cual es conveniente para hacer el proceso de registro de mantenimiento correctivo de manera rápida.

- Desarrollar una interfaz de usuario, para que las actualizaciones de los códigos y número total de registros se vean reflejadas en todos los tipos de usuarios del sistema.
- Desarrollar un software que permita registrar las fallas, códigos de falla, el número de máquina y el tiempo, en una base de datos.

La tesis se estructuró de la siguiente manera: en el primer capítulo se establecen los antecedentes del problema, así como las ventajas de registrar el mantenimiento correctivo en BOC-DWT y la situación actual.

En el segundo capítulo se presenta el marco teórico, que describe los conceptos necesarios para la comprensión del sistema BOC-DWT (*Bard Operations Center - Downtime*), los procesos en los que tendrá impacto y conceptos sobre el estándar ISA-95.

En el tercer capítulo se describe el desarrollo del prototipo de software *BOC-DWT (Bard Operation Center – Downtime)*.

En el cuarto capítulo se describe la arquitectura de los diferentes elementos del prototipo, su funcionamiento y las pruebas realizadas.

En el quinto capítulo se muestran los resultados y conclusiones y los trabajos futuros.

## CAPÍTULO 1: PLANTEAMIENTO DEL PROBLEMA

### 1.1. Antecedentes.

“BD BARD, Productos para el Cuidado de la Salud S.A. de C.V” es una empresa dedicada a la elaboración de productos médicos. La empresa se dedica a la manufactura y distribución de dispositivos médicos para drenado de orina y sangre, distribuidos a los diferentes hospitales alrededor del mundo (Bard, 1995). BD Bard hoy en día depende del uso de la información para responder rápidamente a los cambios y tomar decisiones oportunas (Deuel, 1994). BD Bard depende de su habilidad de comunicar y transferir la información adecuada en el momento oportuno a la gente apropiada (Nagalingam, 1999).

En la *Tabla 1.1* se muestran las diferentes unidades de negocio que se encuentran en BD Bard, en la cual se puede observar la cantidad de máquinas que hay en cada unidad de negocio, así como el producto manufacturado en cada una de ellas, número de Mecánicos por unidad de negocio y por último la cantidad de personas que laboran en cada unidad.

**Tabla 1.1 Máquinas y Personal por Unidad de Negocio (Bard, 1995)**

Unidad de Negocio	Número de Máquinas	Líneas de Producción	Producto Manufacturado	Número de Mecánicos	Personal por Unidad de Negocio
<b>Dielectric</b>	51	14	Bolsas Retenedoras de Orina	15	150
<b>Foley</b>	151	18	Kits Retenedores de Orina	6	1530
<b>PVC</b>	24	4	Catéteres para	2	36

			Infantes		
<b>Especialidades</b>	134	10	Instrumentos Quirúrgicos	8	400
<b>Silicon</b>	214	30	Catéteres Nasogástricos	6	650
<b>Medivance</b>	38	9	Parches Corporales	6	180

En el área de Dielectric, de acuerdo con la *Tabla 1.1*, hay un total de 51 máquinas donde se manufacturan bolsas retenedoras de orina desde el sellado del vinil para el formado de la bolsa y componentes que conforman las bolsas. En esta área hay 14 líneas de producción con 10 operadores de producción por línea dando un total de 140 personas, el personal restante consiste en personal de Calidad y Supervisor de Producción. También en Dielectric hay 15 Mecánicos los cuales son los encargados de la reparación y mantenimiento preventivo de la maquinaria de producción.

En la unidad de negocio Foley hay un total de 151 máquinas de acuerdo con lo mostrado en la *Tabla 1.1*, en donde se ensamblan componentes a los productos manufacturados en Dielectric, así como también se manufacturan productos de contenedores de residuos de fluidos corporales en general. Esta unidad de negocio se divide en 18 líneas de producción donde hay un total de 20 operadores de producción por línea y turno (4 turnos en total) dando un total de 360 personas por línea y el personal restante es personal de Calidad y Supervisores de Producción. También en Foley hay 6 Mecánicos los cuales son los encargados de la reparación y mantenimiento preventivo de la maquinaria de producción.

En la unidad de negocio PVC hay un total de 24 máquinas, de acuerdo con lo mostrado en *Tabla 1.1*, en donde se manufacturan catéteres para ser ensamblados en productos manufacturados en las unidades de negocio Foley, Dielectric y Especialidades. Esta unidad se divide en 4 líneas de producción con 3 operadores de producción por línea divididos en 2 turnos y el personal restante en personal de Calidad y supervisores de Producción. También en PVC hay 2 Mecánicos (1 por

turno), los cuales son los encargados de la reparación y mantenimiento preventivo de la maquinaria de producción.

En la unidad de negocios Especialidades hay un total de 134, de acuerdo con lo mostrado en la *Tabla 1.1*, en donde se manufacturan instrumentos desechables utilizados en quirófanos de los hospitales alrededor del mundo. Esta unidad de negocio se divide en 10 líneas de producción con un total de 10 operadores de producción por línea de producción divididas en 3 turnos y el resto del personal se divide en un cuarto turno de fin de semana. También en PVC hay 8 Mecánicos (2 por turno), los cuales son los encargados de la reparación y mantenimiento preventivo de la maquinaria de producción.

En la unidad de negocios Silicon hay un total de 214 máquinas, de acuerdo con lo mostrado en la *Tabla 1.1*, en donde se manufacturan catéteres de silicón para uso nasogástrico los cuales se usan en pacientes en coma que requieren ser alimentados o administrar algún tipo de medicamento por la vía nasogástrica. Esta unidad de negocio se divide en 30 líneas de producción con un total de 10 operadores de producción por línea, divididas en 2 turnos y el resto del personal se divide en personal de Calidad y supervisores de Producción. También en Silicon hay 6 Mecánicos (3 por turno), los cuales son los encargados de la reparación y mantenimiento preventivo de la maquinaria de producción.

En la unidad de negocios de Medivance hay un total de 134 máquinas, de acuerdo con lo mostrado en la *Tabla 1.1*, en donde se manufacturan parches corporales para infantes de 0 a 2 años que requieren un tratamiento especial como por ejemplo la enfermedad de Raynaud la cual es frío extremo en el cuerpo humano y estos parches ayudan a mantener la temperatura corporal controlada. Esa unidad de negocio se divide en 9 líneas de producción con un total de 10 operadores de producción por línea, dividida en 2 turnos y el resto del personal se divide en

personal de Calidad y Supervisores de Producción. También en Medivance hay 6 Mecánicos (3 por turno), los cuales son los encargados de la reparación y mantenimiento preventivo de la maquinaria de producción.

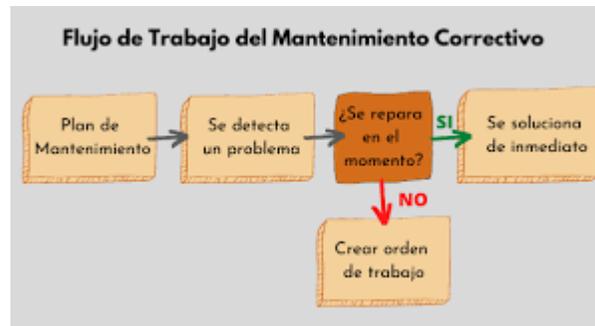
Un modelo del proceso de registro de mantenimiento correctivo de BD Bard puede ser visto en un diagrama (ver *Figura 1.1*) en el que ocurre una avería en una máquina como entrada para iniciar el proceso y da como salida un registro de mantenimiento correctivo. Los detalles internos del sistema de registro de mantenimiento correctivo dependen de procedimientos de operaciones estándar los cuales determinan los elementos involucrados para llevar este proceso a cabo, pero en múltiples procesos ya automatizados la función clave común a todas las organizaciones de manufactura es que el sistema procesa información (Slavendy, 2001).



**Figura 1.1 Entradas y salida típica de un proceso de registro de mantenimiento correctivo.**

La información del registro de mantenimiento correctivo es necesaria para la toma de decisiones e investigaciones por quejas del cliente y funciona como un sistema general de mantenimiento correctivo que puede ser separado en diferentes etapas (ver *Figura 1.2*). El mantenimiento correctivo se genera a partir del plan de mantenimiento preventivo ya existente, donde, si se ejecuta correctamente y durante el funcionamiento de la máquina se detecta un problema o falla, se analiza o evalúa si es una reparación rápida o no y en caso de no ser así, se considera un

mantenimiento correctivo y por ende se genera una orden de trabajo (Slavendy, 2001).



**Figura 1.2 Diagrama de Mantenimiento correctivo.**

Empresas maquiladoras alrededor del mundo han decidido implementar la automatización del registro de mantenimiento correctivo con herramientas como SAP (SAP, 2001), Easy Maint (Maint, 2000), Maximo (Maximo, 1980), entre otros, los cuales son software de escritorio y aplicaciones web.

En BD Bard se utilizó un software llamado Downtime, en el cual se registraba el mantenimiento correctivo por el Clerk de Producción, de acuerdo a lo que el personal de producción le mencionaba durante los diferentes turnos, esta herramienta se dejó de utilizar, ya que no cumplía con un requerimiento corporativo, el cual consiste que los softwares para registro de datos deben de utilizar una base de datos SQL server y este contaba con una base de datos realizada en Microsoft Access (Ruiz G. , 2015).

También, otra herramienta que se usa y es la que actualmente se utiliza, es un formato de registro de mantenimiento correctivo (formato impreso), en el cual se registra información básica de un mantenimiento correctivo y estos archivos son almacenados en un archivo de mantenimiento, siendo solamente utilizado para responder quejas de cliente y no para hacer un análisis para la toma de decisiones por parte de los Gerentes de Producción y Mantenimiento.

Por lo que el objetivo del software a desarrollar es contar con un archivo histórico por máquina, llevar un control del mantenimiento correctivo y preventivo e identificación de las fallas más comunes (Rabago, 2018).

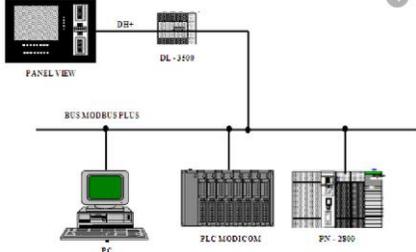
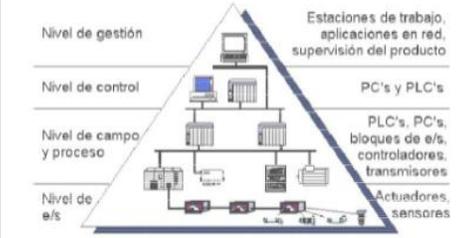
El uso de la computación y su integración en los procesos de manufactura es un fenómeno que tiene más de cinco décadas. Este fenómeno es conocido con las siglas en inglés *CIM* (*Computer Integrated Manufacturing*) ó Manufactura Integrada por Computadora y fue introducido en los años setenta. *CIM* no es un producto o una tecnología específica que pueda ser comprada (Doumeingts, 1995), es una filosofía y un acercamiento a una organización integrada de la fábrica y su administración. Involucra la integración y coordinación de diseño, manufactura y administración a través del uso de computadoras. Dicho de manera simple *CIM* es el uso de sistemas computacionales para integrar una empresa de manufactura (Masood, 2004).

En la *Tabla 1.2* se muestra como ha sido la evolución de la tecnología en manufactura y se puede apreciar la aparición del término *CIM* en la década de los 70s (Velderrain, 2015).

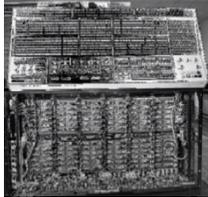
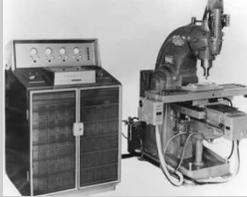
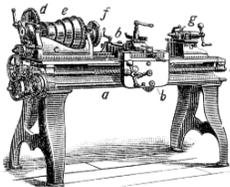
**Tabla 1.2 Evolución de la tecnología en manufactura.**

Actualidad	Uso de dispositivos inalámbricos para el control de sistemas automatizados	
------------	--	--

**CAPÍTULO 1: PLANTEAMIENTO DEL PROBLEMA**

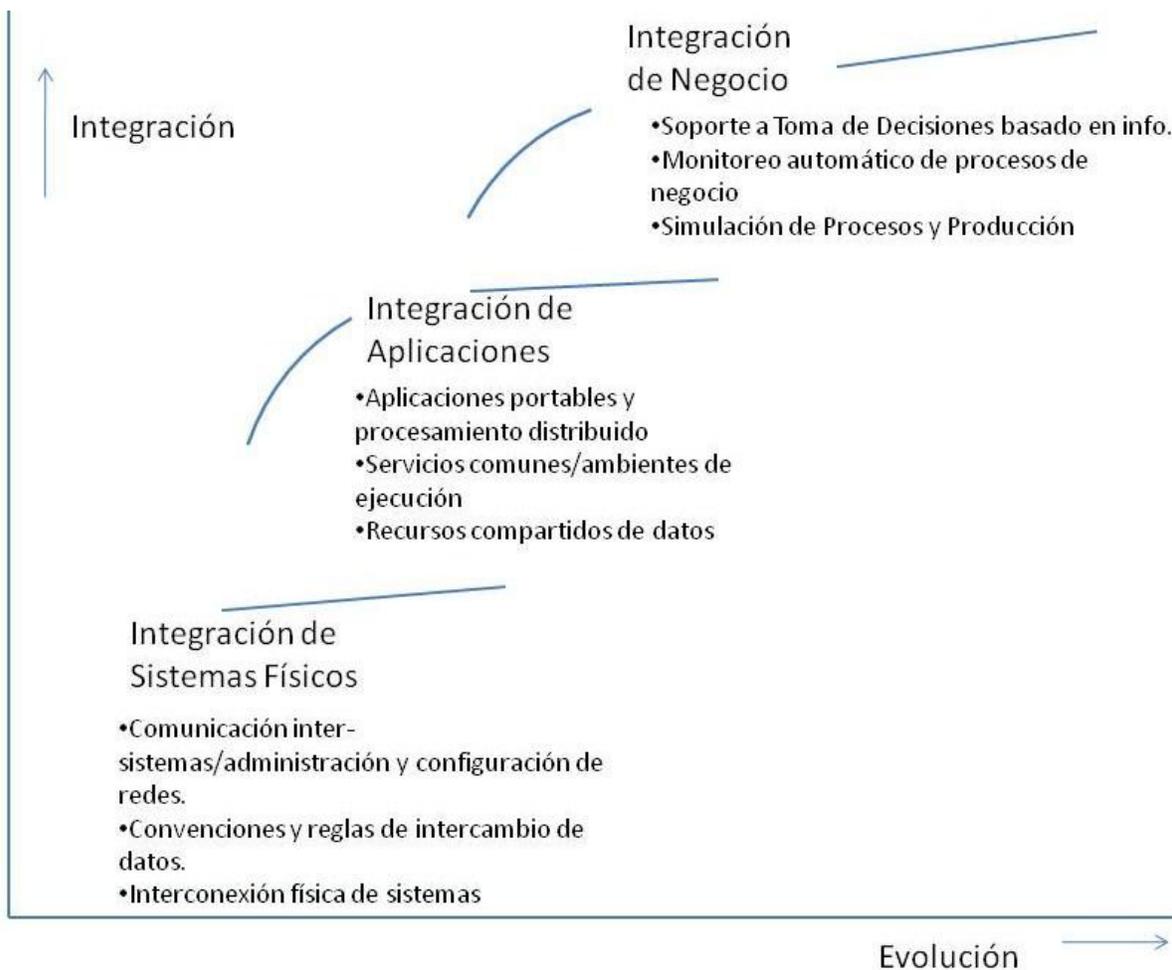
<p>2005</p>	<p>Desarrollo de conceptos de sistemas colaborativos CIM</p>	
<p>2000</p>	<p>Framework y Arquitectura de CIM Virtual</p>	
<p>1997</p>	<p>Introducción del concepto CIM virtual</p>	<p><b>CIM (Computer Integrated Manufacturing)</b></p> 
<p>1986</p>	<p>Sistemas avanzados CAM, CAPP, CAQC, ASR/RS, FMS etc.</p>	
<p>1980</p>	<p>Desarrollos en CAD, aplicación de sistemas basados en CAM</p>	

CAPÍTULO 1: PLANTEAMIENTO DEL PROBLEMA

<p>1970</p>	<p>Introducción del concepto CIM.</p>	
<p>1955</p>	<p>Comienzos de CAD y desarrollos en NC: tales como CNC, DNC.</p>	
<p>1950</p>	<p>Comienzos de CAD y desarrollos en NC: tales como CNC, DNC.</p>	
<p>1930</p>	<p>Grandes sistemas de producción en masa.</p>	
<p>1900</p>	<p>Sistemas de producción en masa.</p>	
<p>1750</p>	<p>Mecanización y primeros sistemas de fabricación.</p>	
	<p>Trabajo artesanal</p>	

1600	Labor manual y uso de animales.	
------	---------------------------------	--

Como se puede ver *CIM* es un término muy amplio pero su evolución se ha dado gradualmente en diferentes niveles. En la *Figura 1.3* se presenta la evolución de los niveles de integración en *CIM*.



**Figura 1.3** Etapas en la evolución de la integración *CIM*.

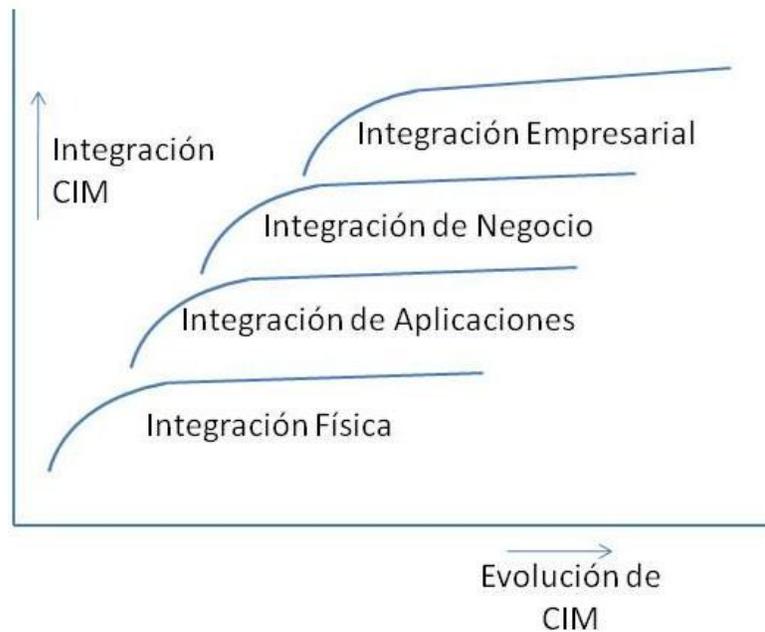
Primero apareció la integración de sistemas físicos, la cual se enfoca en la interconexión de la automatización de manufactura y las instalaciones de procesamiento de datos que permiten el intercambio de información (Slavendy, 2001). La integración de sistemas físicos involucra la interconexión de plantas de manufactura y el intercambio de datos entre varias unidades de redes computacionales basadas en protocolos estándar de comunicación. Esta integración involucra recursos de bajo nivel de manufactura (Nagalingam, 1999).

En segundo lugar, apareció la integración de aplicaciones. Integración a este nivel significa proveer información sin importar en dónde reside esa información (Slavendy, 2001). La integración de aplicaciones considera integración e interoperabilidad de sistemas en plataformas heterogéneas. Esta etapa involucra compartir datos e información entre todas las plantas, ambientes de procesamiento distribuido y servicios comunes para ambientes de ejecución. En la actualidad este aspecto de compartir incluye ambos; conexiones alámbricas e inalámbricas entre varios sistemas (Nagalingam, 1999).

En tercer lugar, apareció la integración de negocio, se enfoca en la integración de aquellas funciones que manejan, controlan y monitorean la ejecución diaria de las actividades en el nivel de aplicación (Slavendy, 2001). La integración de negocios concierne la integración de todas las funciones, procesos de negocio y sistemas a un nivel de empresa (dentro de una empresa y más allá a socios de negocios y clientes) que incluyen comercio electrónico, manejo de relación con clientes, y logística global y aplicaciones relacionadas y muchas más (Nagalingam, 1999).

En la actualidad se considera una cuarta etapa denominada Integración Empresarial que consiste en la integración global de plantas de manufactura de una

empresa incluyendo subcontratistas y proveedores, también es llamado CIM virtual (ver *Figura 1.4*).



**Figura 1.4 Integración Empresarial como cuarto nivel de integración.**

Algunos investigadores y consultores a menudo malinterpretan CIM enfocándose solo en la letra C de computadoras, sin embargo, CIM no se enfoca solo en computadoras sino también en las letras I y M de Manufactura Integrada (Nagalingam, 1999) por lo que el simple uso de computadoras no garantiza la solución de automatización, sino que los sistemas deben estar integrados.

## **1.2. Estado del campo del arte.**

En esta sección se describen una variedad de sistemas actuales de software que caen dentro de la categoría de CIM enfocados al departamento de mantenimiento:

- Maximo – Sistema de planeación y administración de mantenimiento preventivo y registro de mantenimiento correctivo (Maximo, 1980).
- Downtime – Sistema para el registro de mantenimiento correctivo (Ruiz E. , 2018).
- SAP – Sistema de múltiples módulos el cual aplicable a diferentes departamentos como control de documentos, compras, almacén, producción, calibración, mantenimiento entre otros (SAP, 2001).
- Easy Maint - Sistema de planeación y administración de mantenimiento preventivo y registro de mantenimiento correctivo (Maint, 2000).
- FaciliWorks – Sistema con múltiples módulos aplicable a diferentes áreas de producción como mantenimiento preventivo, refacciones, compras y calibración (Langhan, 2016).

Los sistemas anteriormente listados son algunos de los softwares más utilizados para el manejo de procesos de mantenimiento, pero además de ser costosos, éstos requieren un proceso de aprobación extenso por parte de BD Bard, el cual suele tardar de 1 a 2 años para aprobar el capital y realizar la compra; otro de los factores es que una vez comprado, se somete a un proceso de validación el cual tiene una duración promedio de hasta 3 años para poder ser implementado en el piso de producción, por estos motivos fue que se decidió desarrollarlo en casa, ya que al ser desarrollado por personal de BD Bard, no es necesario someter aprobación para la compra y el proceso de validación no es tan extenso como un software externo a la empresa (Ruiz G. , 2015). En BD Bard se utiliza FaciliWorks, el cual es un software que carece de un módulo para el registro de mantenimiento correctivo.

Algunos trabajos realizados en la empresa BD Bard para el registro de tiempo caído se muestra en la *Figura 1.5*.



**Figura 1.5 Registro de tiempo caído a lo largo de los años en BD Bard**

Como primer y último trabajo realizado en BD Bard se tiene la forma F76PM012 (Pineda, Registro y Manejo de Tiempo Caido, 2010), la cual es una forma que se usa para el registro de mantenimiento correctivo. Es un formato impreso donde se registran diferentes variables referentes al mantenimiento correctivo, como el tiempo, la falla, número de máquina, entre otros. Actualmente se utiliza este formato físico para el registro y se archiva por el departamento de mantenimiento.

En BD Bard, en el periodo de 1995-2010, el departamento de IT (Tecnologías de la Información) desarrolló un sistema para el registro de tiempo caído el cual cumple con el objetivo de registrar el mantenimiento correctivo, solo que éste cuenta con una base de datos realizada en Microsoft Access, por lo que al pasar de los años, cambiaron las regulaciones de BD Bard y no fue posible utilizarla más, volviendo a utilizar el formato anterior y por ende volviendo al proceso anterior de utilizar formatos físicos para el registro del mantenimiento correctivo, ocasionando dificultades al momento de analizar la información recabada para futuros análisis de fallas (Cardenas, 1990).

No obstante, a pesar de que la forma utilizada actualmente para el registro de mantenimiento correctivo cuenta con información de registro importante, sigue siendo insuficiente para tener un seguimiento operativo de la ejecución de las actividades de mantenimiento, debido a que el registro de mantenimiento correctivo es difícil de analizar, ya que se requiere capturar los datos del formato físico a un formato

electrónico. Hay también una carencia de sincronización oportuna de datos entre la realidad de planta y la captura de datos en un formato electrónico, la información que tiene el proceso de registro de mantenimiento correctivo llega incompleta, a destiempo y en ocasiones puede estar viciada de errores. La actualización de datos de mantenimiento correctivo ocurre en intervalos diarios, semanales o aún mensuales mientras que la planta funciona en tiempo real. En la mayoría de los casos, la información es cargada por algún operador al finalizar el turno, con lo cual se pierde la visión de lo que está ocurriendo en ese mismo momento. Es bastante común que los operadores incurran en errores de transcripción de los datos, registrando información que confunde y toma mucho tiempo en depurar (Mendoza, 2020).

La forma de registro de tiempo caído F76PM012 (Pineda, Registro y Manejo de Tiempo Caido, 2010) es una herramienta de gestión básica para el registro de mantenimiento correctivo utilizada en BD Bard y el mercado ofrece productos altamente competitivos, sin embargo, estos son costosos y al ser vendidos por un proveedor externo se tienen que someter a un proceso de validación, el cual suele durar de 2 a 3 años, ya que BD Bard una empresa dedicada a la manufactura de productos para el cuidado de la salud y sus protocolos regulatorios son más estrictos que los de otras empresas manufactureras, por lo tanto no es una opción muy conveniente (Cardenas, 1990). De igual forma, el formato para el registro de mantenimiento correctivo no está especialmente diseñado para satisfacer las necesidades de gestión en tiempo real en una planta productiva y no es capaz de ver y analizar los datos, ya que estos son capturados en formatos físicos por lo que BOC-DWT (desarrollado en la presente tesis) obtendrá la información de los mantenimientos correctivos al momento de presentarse el evento y quedarán almacenado en una base de datos desarrollada en SQL Server (Salazar, 2009).

Es común que las empresas que no se dedican a la manufactura de productos para el cuidado de la salud cuenten con sistemas robustos para el manejo de la totalidad de sus procesos (Refacciones, mantenimiento correctivo, mantenimiento preventivo entre otros) (Rabago, 2018). En la actualidad el departamento de IT de BD Bard, es el responsable de los sistemas relacionados a oficina como procesamiento de órdenes, manejo administrativo de materiales y contabilidad de costos, este departamento piensa en términos de lenguajes de programación como Java, .Net entre otros; por otro lado los sistemas de Control de Proceso de mantenimiento usualmente quedan a cargo del mismo departamento de Mantenimiento, ya que ellos hablan en términos del mundo relacionado en su totalidad a reparaciones y mantenimiento de equipos de manufactura y por esta razón suele ocurrir esto con frecuencia en las empresas maquiladoras, esto quiere decir que los software dedicados a los departamentos de mantenimiento son meramente técnicos y totalmente dedicados a los procesos de mantenimiento, el o los administradores del software suelen ser empleados del mismo departamento dejando fuera al departamento de Tecnologías de la información. (Excellence, 2018).

*BOC-DWT* se alimenta en tiempo real y en línea de datos provenientes directamente del personal de mantenimiento y almacenados en *SQL Server* y los convierte en información para la toma de decisiones. Algunos sistemas de control de proceso de mantenimiento entregan sus resultados también a otro software tales como sistemas conexos como *SAP*, *Baan*, *JDEdwards*, etc (Salazar, 2009).

La integración *BOC-DWT* provee fácil acceso a información en tiempo real de todos los niveles de la operación. Se usa para registrar, monitorear y controlar el proceso de mantenimiento correctivo (Deuel, 1994).

Los sistemas de ejecución de captura de mantenimiento correctivo, en este caso *BOC-DWT*, son sistemas que proporcionan la información necesaria para

optimizar los sistemas de mantenimiento preventivo desde el lanzamiento del evento de mantenimiento correctivo, aportando beneficios específicos en las actividades de mantenimiento (Salazar, 2009).

Los gerentes de mantenimiento a cargo del plan de mantenimiento preventivo tienen una multitud de preguntas que necesitan ser respondidas a diario. Ellos no pueden esperar a que los datos extraídos de las formas físicas de mantenimiento correctivo sean cargados a un formato electrónico y no pueden confiar en los datos de mantenimiento correctivo fuera de fecha. Los datos están típicamente disponibles en archivos históricos de los sistemas de control de mantenimiento. Los sistemas de Control de mantenimiento trabajan con parámetros de tiempo y están enfocados al equipo, consecuentemente los datos que necesita el gerente de mantenimiento no están en la forma correcta” (Deuel, 1994). Ejemplos de preguntas que pueden surgir en un área de mantenimiento son las siguientes:

- ¿Cuánto tiempo de mantenimiento correctivo hay en el último turno?
- ¿Cuánto tiempo de mantenimiento correctivo hay por área?
- ¿Cuál fue la falla con más tiempo caído?
- ¿Se necesitará más mano de obra para el siguiente turno?
- ¿Qué será mejor producir a continuación?
- ¿A qué velocidad conviene producir?

Por lo tanto, *BOC-DWT* provee una funcionalidad clave tanto para la integración de sistemas, así como para la realización de las funciones diarias en *BD Bard*.

### **1.3. Estándar Internacional ISA-95**

ISA-95 es el estándar internacional para integración de sistemas de manufactura de la *International Society for Automation (ISA)*, éste será usado para el desarrollo de la presente tesis.

Como se mencionó en un comienzo, han surgido algunas soluciones parciales que han resultado en sistemas desintegrados. En muchos casos debido al uso de diferentes tecnologías. Por esto se plantea adoptar el uso del estándar internacional ISA-95 (ISA95, 2011), que es un estándar vigente que consiste en siete partes.

### **1.3.1. Diferentes lenguajes**

En todo el mundo, cada compañía de manufactura está organizada de manera diferente y usa diferentes sistemas de automatización. No hay compañías con exactamente los mismos nombres para departamentos, actividades y funciones. La información que se intercambia también tiene diferentes nombres en diferentes compañías y sistemas de automatización, en conclusión todas las empresas difieren de acuerdo a su terminología en sus áreas y procesos pero cabe recalcar que estos son muy parecidos y por lo tanto el software a desarrollar para BD Bard se tendrán que adoptar algunos términos utilizados en la empresa para ser integrados en software BOC-DWT esto con el fin de que el usuario reconozca la terminología que utiliza en su día a día y esto por ende facilitará su uso al momento de interactuar con el nuevo software. (Excellence, 2018).

### **1.3.2. Intercambio de Información**

Aunque se hablan diferentes lenguajes, ambos niveles tienen que comunicarse uno con el otro. El intercambio de información es esencial para que el sistema funcione adecuadamente, pero para esto debe existir una comunicación, en la cual el personal de mantenimiento (Mecánicos) tendrá que proveer información acerca del registro de mantenimiento correctivo. El personal administrativo de

mantenimiento también tendrá que enviar información al gerente de mantenimiento. Por ejemplo, información acerca del tiempo total y número de órdenes acerca de todo lo referente a equipos de manufactura usados para el proceso de producción y demás. La información será compartida en distintas juntas con las diferentes áreas de producción para la toma de decisiones de acuerdo a la disponibilidad de las máquinas y el Planeador de Producción realice planes de producción correctamente evitando el paro de operadores de producción (Cardenas, 1990).

### **1.3.3. Interfaces Automatizadas**

Con la aparición de nuevas tecnologías, se está tornando más fácil automatizar el intercambio de información entre la oficina y el piso de producción. Una interfaz automatizada entre los sistemas empresariales y de control puede llevar a muchas ventajas. Información importante se convierte accesible al momento adecuado y en el lugar adecuado. La empresa tiene acceso a información de tiempo real tal como información acerca del tiempo total por orden de mantenimiento correctivo, lo que habilita el uso óptimo de la capacidad de almacenamiento, esto quiere decir que BOC-DWT, al contener la información de los mantenimientos correctivos en una base de datos, estará disponible para su consulta desde cualquier parte de la empresa, como por ejemplo en una presentación o sala de juntas facilitando a los Gerentes de las diferentes áreas la información de los mantenimientos correctivos para la toma de decisiones (Excellence, 2018).

### **1.3.4. La Solución**

El estándar internacional ISA-95 fue desarrollado para considerar los problemas encontrados durante el desarrollo de interfaces automatizadas entre

sistemas de control y empresariales. Éste estándar ha sido desarrollado para todo tipo de ambientes de manufactura y en todo el mundo. Puede ser aplicado a todas las industrias y sus diferentes procesos, como los procesos de mantenimiento.

### **1.3.5. Partes de ISA-95**

Existen 6 partes del estándar ISA-95. La parte 1 consiste de terminología y modelos de objeto que pueden ser usados para decidir qué información debe ser intercambiada (ANSI/ISA-95.00.01-2010, 2010). La parte 2 consiste de atributos para cada clase de objeto definido en la parte 1 (ANSI/ISA-95.00.01-2010, 2010). Los objetos y atributos de la parte 2 pueden ser usados para el intercambio de información entre diferentes sistemas y además estos objetos y atributos pueden también ser usados como base para bases de datos relacionales. La parte 3 se enfoca en las funciones y actividades de nivel 3 (ANSI/ISA-95.00.01-2013, 2013). Es una excelente guía para describir y comparar los niveles de producción de diferentes sitios de una manera estandarizada. Las partes 4 y 5 proporcionan, como sus títulos lo indican, los “Modelos de objetos y atributos de la gestión de las operaciones de manufactura” y la “Integración de la empresa y los sistemas de control” (ANSI/ISA-95.00.04-2012, 2012). En la parte 6 se describen el modelado de los servicios de mensajería usado para la interface de negocio y actividades de manufactura (ANSI/ISA 95-95.00.06, 2014).

### **1.3.6. Parte 1 de ISA-95**

La parte 1 de ISA-95 “*Enterprise Control System Integration 1: Models and Terminology*, Integración de Sistemas Empresariales y de Control: Modelos y Terminología” (ANSI/ISA-95.00.01-2010, 2010), provee modelos y terminología para

describir las interfaces entre los sistemas de negocio de una empresa, sus operaciones de manufactura y sistemas de control. Los modelos y terminología presentada en el estándar sirven para:

- Enfatizar las buenas prácticas de integración de sistemas de control con sistemas empresariales durante el ciclo de vida de los sistemas.
- Para mejorar las capacidades de integración de las operaciones de manufactura y sistemas de control con sistemas empresariales.
- Puede ser aplicado sin importar el grado de automatización.

Específicamente, éste estándar provee terminología estándar y un juego consistente de conceptos y modelos para integrar sistemas de control con sistemas empresariales, lo que mejorará la comunicación entre todas las partes involucradas. Algunos de los beneficios que se producirán será:

- El reducir el tiempo al usuario para lograr niveles de producción completos en nuevos productos.
- Habilitar a vendedores para entregar las herramientas apropiadas para implementar la integración de sistemas de control con sistemas empresariales.
- Habilitar a los usuarios para identificar sus necesidades de mejor manera.
- Reducir costos de automatización de procesos de manufactura.
- Optimizar las cadenas de suministros.
- Reducir el tiempo de ciclo de los esfuerzos de ingeniería.

Esta parte 1 del estándar está pensada para:

- Los involucrados en el diseño, construcción y operación de plantas de manufactura.

- Responsables de especificar interfaces entre manufactura y sistemas de control de procesos y otros sistemas de negocio.
- Los involucrados en el diseño creación, marketing e integración de productos de automatización usados para hacer la interface entre operaciones de manufactura y sistemas de negocios.
- Los involucrados en especificar, diseñar y manejar la creación de productos, movimiento y almacenamiento dentro de las empresas de manufactura.

No es la intención del estándar:

- Sugerir que solo existe una forma de implementar integración de sistemas de control a sistemas empresariales.
- Forzar a los usuarios a abandonar sus métodos actuales sobre el manejo de la integración.
- Restringir el desarrollo en el área de la integración de sistemas de control a sistemas empresariales.

Esta parte 1 del estándar discute la interface entre las funciones de manufactura y control y otras funciones empresariales. Basado en *Purdue Reference Model* (forma jerárquica) publicada por ISA. Éste estándar presenta un modelo parcial o referencia al modelo definido en ISO 15704 (ANSI/ISA-95.00.01-2010, 2010).

### **1.3.7. Parte 2 de ISA-95**

La parte 2 de ISA-95: “*Object Model Attributes* (Atributos del Modelo Objeto)” (ANSI/ISA-95.00.01-2010, 2010), define más allá de los objetos formales de intercambio de información descritos en la parte 1, usando modelos objeto UML, tablas de atributos y ejemplos.

Esta parte de ISA-95, en conjunto con la parte 1 del estándar, especifica una interface genérica de contenido entre las funciones de control de manufactura y otras funciones empresariales. La interface considerada está entre el nivel 3 de sistemas de manufactura y el nivel 4 de sistemas de negocio en la estructura jerárquica del modelo definido en la parte 1. El objetivo es reducir el riesgo, costo y errores asociados con la implementación de la interface (ISA95, 2011).

Ya que éste estándar cubre muchos dominios y hay muchos diferentes estándares en tales dominios, la semántica de este estándar es descrita al nivel intencionado para habilitar los otros estándares a ser mapeados en esta semántica. Para este fin éste estándar define un juego de elementos contenidos en la interface genérica, junto con un mecanismo para extender esos elementos para implementaciones.

El alcance de la Parte 2 está limitado a la definición de modelo objeto y los atributos de la información intercambiada definida en la Parte 1 (ANSI/ISA-95.00.01-2010, 2010).

La parte 2 del estándar no define atributos para representar la relación de objetos.

### **1.3.8. Parte 3 de ISA-95**

La parte 3 de ISA-95: “*Activity Models of Manufacturing Operations Management* (Modelos de Actividades de la Administración de las Operaciones de Manufactura)” (ANSI/ISA-95.00.01-2013, 2013), define las actividades de producción y flujos de información. Dentro de las áreas de producción varias actividades son ejecutadas y mucha información es intercambiada. ISA-95 parte 3 provee modelos de referencia para actividades de producción, calidad, mantenimiento e inventario. Con estos

modelos es posible hacer claro cuál es la situación real dentro de la compañía (ANSI/ISA-95.00.01-2013, 2013). Por ejemplo, las siguientes opciones pueden ser contestadas usando los modelos ISA-95:

- ¿Cuáles son las responsabilidades de los diferentes departamentos?
- ¿Cuáles son las responsabilidades de los diferentes sistemas de automatización?
- ¿Cuál es el enfoque de los múltiples proyectos dentro de la compañía y cuáles son las brechas y responsabilidades que se solapan?
- ¿Qué funcionalidad es representada en más de un sistema de información?
- ¿Cuáles son las fronteras entre nuestros sistemas de automatización?
- ¿Cuáles son los requerimientos de usuario para el nuevo sistema?
- ¿Qué producto es adecuado, basado en nuestros requerimientos?

ISA-95 parte 3 provee terminología y un conjunto de modelos consistentes para la definición de actividades que apuntan al intercambio y procesamiento de información de producción. El uso del estándar puede dar las siguientes ventajas:

- Los proveedores pueden desarrollar herramientas para las actividades de producción.
- Los usuarios finales pueden identificar sus necesidades más fácilmente.
- Los usuarios finales pueden comparar los productos de diferentes proveedores.
- Puede desarrollar una definición más completa de las actividades de producción y la información dentro de la compañía.
- Se puede hacer claro qué sistemas de automatización son usados dentro del área de producción y cuál es la responsabilidad exacta para cada sistema.
- Los usuarios finales pueden mejorar la comunicación con proveedores y con colegas.

### 1.3.9. Parte 4 de ISA-95

El comité SP95 está desarrollando la parte 4 de ISA-95, titulada “*Object Models and Attributes of Manufacturing Operations Management* (Modelos de Objetos y Atributos de la Administración de Operaciones de Manufactura)” (ANSI/ISA-95.00.04-2012, 2012). Esta especificación técnica define las interfaces entre las actividades de la empresa y el control de actividades. El objetivo de la parte 4 del estándar está limitado en la definición de modelos de objetos y atributos para la información definida en la parte 3. El objetivo es reducir el esfuerzo, costo y errores asociado con la implementación de interfaces de las partes 1 y 3. Los modelos y atributos de la parte 4 son la base para asegurar un intervalo flexible de cooperación e intercambio de información entre las diferentes actividades de Mantenimiento (ISA95, 2011).

El estándar ISA-95 parte 4, es una conjunción con ANSI/ISA-95.00.03-2005, Empresa – Control de integración de sistema - Parte 3: Modelos de actividad de manufactura para el manejo de operaciones, donde se especifica que son interfases cruzadas y con el nivel 3 de categorías del modelo de jerarquía definido en la parte 1 de ISA-95. De esta manera permitiendo reducir riesgos al momento de implementar la parte 1 y 3 del estándar ISA-95 (ISA95, 2011).

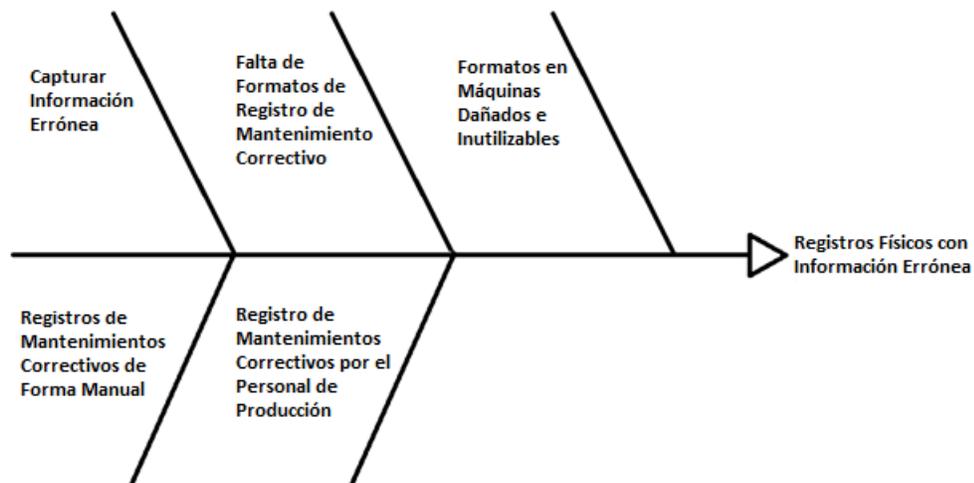
### 1.3.10. Parte 5 de ISA-95

La Parte 5 de ISA-95: “*Business to Manufacturing Transactions* (Transacciones de Negocio a Manufactura)” (ANSI/ISA-95.00.05-2013, 2013), está basada en el uso de los modelos abstractos previamente definidos en los estándares ISA-95 Parte 1 y Parte 2, combinados con verbos OAGi para definir modelos de transacción de

intercambio de información. Es reconocido que otros protocolos que no forman parte de ISA-95 Parte 5, son posibles de usar y no son considerados inválidos como resultado de este estándar. Las transacciones ocurren en todos los niveles dentro de la empresa y entre compañeros de negocio y están relacionadas con las actividades reales y las necesitadas, pero el enfoque de este estándar es la interface entre sistemas de negocio y sistemas de manufactura (ISA95, 2011).

#### 1.4. Definición de la situación actual del problema

En la *Figura 1.6* se muestra un diagrama de causa y efecto que determina cuáles son las causas del registro de mantenimiento correctivo en BD Bard y cuáles son sus efectos (Jeison, 2018):



**Figura 1.6 Diagrama de Causa y Efecto Para Proceso de Registro de Mantenimiento Correctivo.**

La falta de visión integradora en las implementaciones de software en los pisos de producción de BD Bard causa un serio problema en el registro de mantenimiento correctivo y ocurre cuando el registro de mantenimiento correctivo

depende totalmente del departamento de operaciones, produciendo una ineficiencia local y de la planta.

El software desarrollado por el departamento de IT (Tecnologías de la información) fue discontinuado por no cumplir con las regulaciones de la empresa al utilizar una base de datos desarrollada en Microsoft Access por lo que ya no fue posible utilizarlo y por ende se utiliza un formato físico de registro de mantenimiento correctivo el cual es llenado por operaciones y esto ocasiona el registro de información errónea, pérdida de información al encontrarse estos documentos archivados y no es posible realizar un análisis de información para investigaciones (Ruiz E. , 2018).

También existe la posibilidad de comprar uno en el mercado, pero al ser comprado por un proveedor externo y al mismo tiempo siendo BD Bard una empresa para la manufactura de productos para el cuidado de la salud y con estrictas regulaciones esto se vuelve complicado ya que se requiere una validación exhaustiva de 2 a 3 años siendo esto mucho tiempo y siendo un gran problema el proceso de registro de mantenimientos correctivos por que se optó por desarrollar *BOC-DWT* para evitar el proceso de validación ya que al ser desarrollado dentro de las instalaciones de la misma empresa y por empleados de la misma con los respectivos entrenamientos en estas regulaciones es posible omitir el paso de la extensa validación y solo someterse a pruebas funcionales lo cual resulta una buena opción del desarrollo de *BOC-DWT* (Ruiz E. , 2018).

Otra causa es la automatización por necesidad y no por planeación, que ocurre cuando a medida que transcurre el tiempo, en BD Bard automatizan parcialmente sus operaciones, conforme a las necesidades de eficiencia, seguridad, reducción de costos entre otros, todo esto sin una planeación que permita ir integrando estos incrementos graduales de automatización (Mendoza, 2020).

Estos factores son causa de porque es necesario y se requiere automatizar el proceso de registro de mantenimiento correctivo, por lo que en este momento el desarrollo de *BOC-DWT* es la mejor opción por motivos de tiempo y el problema presentado (Ruiz E. , 2018).

El problema más grande que se tiene en el departamento de mantenimiento de equipo es la forma física que realiza operaciones para el registro de mantenimiento correctivo el cual está lleno de errores ya que no es llenado por el personal indicado y al ser estos formas físicas estos tienden a ser extraviados o dañados por diferentes factores; a su vez también es difícil llevar una rastreabilidad de datos y hacer análisis de información para llevar a cabo investigaciones para mejora de procesos dentro del departamento de mantenimiento e incluso se dificulta dar respuesta a ciertos requerimientos de BD Bard (Pineda, Problemas DWT, 2017).

### **1.5. Definición del problema**

El registro físico de mantenimiento correctivo llenado por operaciones con información errónea y documentos extraviados o dañados en la empresa BD Bard, son los causantes de la generación de datos erróneos, esto quiere decir que la información registrada no es la correcta y por ende, ésta no es funcional para el análisis en la toma de decisiones. Al extraviarse ocasiona pérdida de información para el análisis y a su vez dificulta la respuesta a quejas del cliente por parte del departamento de Calidad.

### **1.6. Justificación**

El presente trabajo es un paso adelante en el esfuerzo de automatizar el proceso de registro de mantenimiento correctivo que se tiene en la empresa BD Bard que

ocasiona indisponibilidad de información para responder los problemas relacionados con la maquinaria en las áreas de producción. Dicha información es necesaria para responder de forma efectiva y rápida al personal directivo de la empresa, por lo que aumenta la visibilidad de lo que ocurre en la maquinaria de las áreas de producción, incrementado a su vez la capacidad de dar respuesta a contingencias.

El presente trabajo presenta una forma de estructurar los desarrollos de software en manufactura enfocado en el departamento de mantenimiento para integrar información de todo lo relacionado a las fallas de la maquinaria de las múltiples áreas, considerando que la solución de automatización se encuentra en un ambiente de cambios graduales en las necesidades de información de manufactura.

Además de lo anterior se pretende brindar los siguientes beneficios:

- Una guía de buenas prácticas en el desarrollo software para Mantenimiento de equipo.
- Conocer cómo evitar los errores comúnmente cometidos al desarrollar o adquirir soluciones específicas de software de automatización que solo resuelven problemas locales y a corto plazo, sin considerar su contexto y los problemas que trae en el mediano y largo plazo.
- Conocer las ventajas de la utilización de un estándar internacional para el desarrollo de software de manufactura.
- Conocer las ventajas de utilizar un software de registro de mantenimiento correctivo (*BOC-DWT*) y su diferencia con los más comúnmente encontrados en otras empresas maquiladoras.

- Conocer cómo *BOC-DWT* es clave en la integración vertical entre los sistemas de Mantenimiento Correctivo con los Sistemas de Control comúnmente encontrados en las empresas de manufactura.
- Conocer cómo crear un software robusto y estructurado utilizando el estándar ISA-95, utilizando los modelos de clases propuestos por el mismo estándar y también a identificar las clases necesarias para desarrollar el software de registro de mantenimiento correctivo BOC-DWT.

### **1.7. Solución propuesta.**

Se desarrollará un análisis, diseño y desarrollo de un software para el registro de mantenimiento correctivo *BOC-DWT*, que ayude al personal de mantenimiento de equipo a registrar el mantenimiento correctivo del equipo de las diferentes áreas de la empresa BD Bard, basado en el estándar ISA-95.

### **1.8. Requerimientos.**

Como parte de la funcionalidad del *Bard Operation Center (BOC-DWT)* y de la utilización del estándar ISA-95 para la realización de *BOC-DWT*, se automatizo la tarea de registrar el mantenimiento correctivo que presente la maquinaria de las diferentes áreas de producción:

- Falla en maquinaria.
- Número de Máquina.
- Tiempo de duración del mantenimiento correctivo.
- Código de falla de Máquina.

Para el desarrollo de esta tesis se utilizó la Parte 1 del estándar ISA-95, la cual brinda la terminología, el conjunto de conceptos y modelos para la integración de los sistemas de control, que a su vez se utilizó para mejorar la comunicación por parte del personal de mantenimiento a los demás departamentos. Ya que esta parte del estándar está limitado a describir en general las funciones relevantes en la empresa y siendo el departamento de mantenimiento una de estas, esto hace que sea ideal para ser utilizado en el presente proyecto de tesis ya que con esto se garantiza que el intercambio de información es robusto, seguro y nos ayudará a preservar la integridad completa de BOC-DWT.

Otra de las partes que se utilizó en este proyecto de tesis es la Parte 2 de ISA-95, la cual contiene todos los modelos de las clases que se implementaron en BOC-DWT y a su vez es un multicomponente del conjunto de los estándares que definen las interfaces entre las actividades de la empresa y el control de ellas. Al ser esta parte la que define los modelos de clases usando diagramas de UML, lo cual es ideal para ser implementado en el desarrollo de BOC-DWT.

Otra de las partes a utilizar en este proyecto de tesis es la parte 3 de ISA-95, la cual la convierte en una herramienta esencial, ya que contiene la sección de mantenimiento en la cual se describen los diferentes procesos que ocurren dentro del departamento por lo que solo se utilizará lo necesario o lo correspondiente al mantenimiento correctivo para el desarrollo de BOC-DWT. Otro de los aspectos que convierte en ideal a la Parte 3, es que tiene como objetivo reducir los riesgos y errores para implementar sistemas en las empresas de manufactura, por ende, ayudo con la implementación de BOC-DWT.

Además, el sistema tiene una interface de usuario para facilitar el registro de eventos de mantenimiento correctivo y también cuenta un módulo para el acceso de consulta de registros históricos y también de esta manera el sistema *BOC-DWT* permite estar concentrado en un servidor, evitando tener que modificar la base de

datos en cada una de las terminales cuando es necesario algún tipo actualización del software *BOC-DWT*.

## **1.9. Objetivos**

### **1.9.1. Objetivo general**

El objetivo general del presente trabajo de tesis consiste en desarrollar un software *BOC-DWT* para el registro de mantenimiento correctivo, basado en el estándar ISA-95, que integre al equipo de cómputo ya utilizado en las áreas de producción de las diferentes áreas de BD Bard.

### **1.9.2. Objetivos específicos.**

Además, se pretenden alcanzar los siguientes objetivos particulares:

- 1.- Hacer un estudio del estándar ISA-95 para determinar las partes del estándar a utilizar, así como también los modelos de clases a implementar en *BOC-DWT*.
- 2.- Realizar análisis y diseño orientado a objetos para el desarrollo de un software *BOC-DWT*.
- 3.- Utilizar el Proceso Unificado de Desarrollo (RUP) para el desarrollo del software.
- 4.- Exportar información de la base de datos SQL a Microsoft Excel como parte de la funcionalidad del sistema.
- 5.- Desarrollar el software con una arquitectura de 3 capas.
- 6.- El prototipo deberá tener una interfaz de usuario desarrollada en Visual Studio.
- 7.- El prototipo deberá programarse orientado a objetos en C#.

## **1.10. Limitaciones y Delimitaciones**

### **1.10.1. Limitaciones**

El software BOC-DWT fue desarrollado en la empresa BD Bard, por ende, al ser un software nuevo, las reseñas o comentarios de los usuarios acerca del uso y manejo del software y los requerimientos funcionales, permitirá hacer cambios graduales propuestos como trabajos futuros.

En el tiempo que se le dedica al desarrollo de BOC-DWT, se utilizaron las horas laborales en BD Bard, por lo que las actividades rutinarias en el puesto de jefe de Mantenimiento tendrán que reestructurarse para poder llevar a cabo el desarrollo del presente proyecto de tesis.

El presente trabajo se enfocará al registro de información básica para el mantenimiento correctivo, por lo que su funcionalidad será sobre un proceso de registro de mantenimiento correctivo en una máquina de producción y también se podrá exportar la información a un formato de Microsoft Excel para el análisis de información lo suficientemente válida.

### **1.10.2. Delimitaciones.**

La funcionalidad que tiene *BOC-DWT* es el registro de mantenimiento correctivo con información básica que consta de las siguientes variables: el No. Orden, Falla, No. Máquina, Tiempo, Código de Falla y el Área. Las variables registradas en *BOC-DWT* podrán ser exportadas a un formato de Microsoft Excel en el cual se podrá analizar la información para la toma de decisiones con respecto a la maquinaria.

La primera prueba consistirá en registrar un evento de mantenimiento correctivo, una vez que una máquina de una línea de producción presente una falla, en donde el Mecánico procederá a registrar el evento de mantenimiento correctivo en el sistema *BOC-DWT*. Cualquier máquina de esta línea modelo de producción puede presentar una falla la cual debe ser registrada en *BOC-DWT* por lo que esto aplica al interés para propósitos del registro de mantenimiento correctivo (ver *Figura 1.7*).



**Figura 1.7** Área de producción para la realización de la prueba de BOC-DWT.

Los datos sobre el registro de mantenimiento correctivo de maquinaria se encuentran en una base de datos. Los datos son utilizados para el análisis de la toma de decisiones en cuanto al plan de mantenimiento preventivo o dar estatus al personal directivo para la planeación de producción en caso de presentarse un evento de mantenimiento correctivo grave en alguna máquina o para el reacomodo de personal de producción o incluso considerar alternativas para manufacturar otros productos. Los datos están registrados por área, máquina y fallas (los cuales se

encuentran almacenados en el software *BOC-DWT* desarrollado para la presente tesis).

### **1.11. Método**

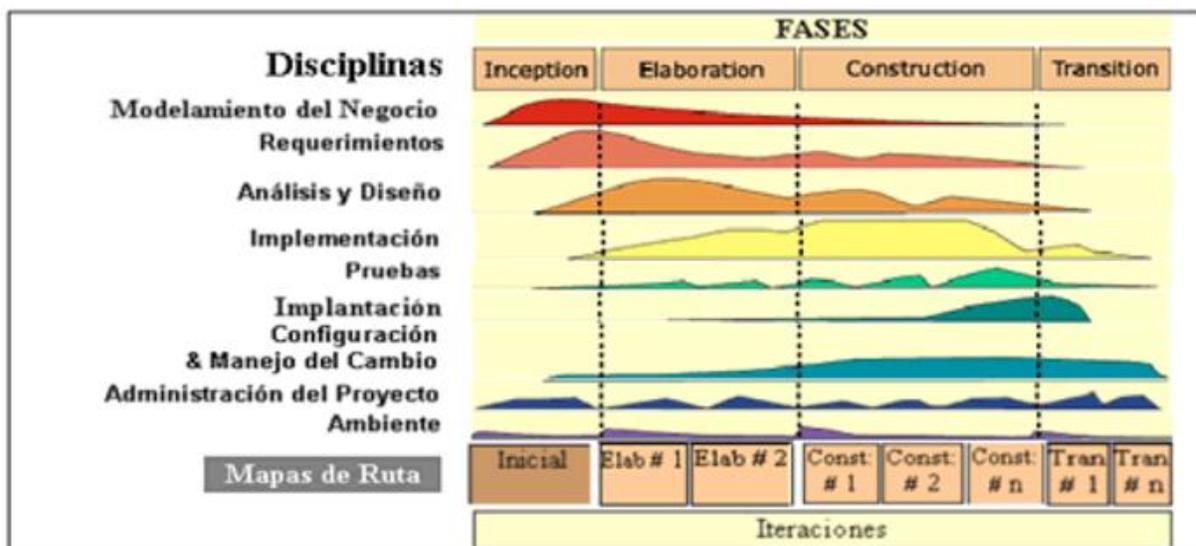
Para el cumplimiento de los objetivos del desarrollo del software *BOC-DWT* se utilizaron las herramientas de Visual Studio 2019 y SQL Server 2019. Utilizando el lenguaje de programación de C#, se utilizó algunas secciones del estándar ISA-95 para comprobar que se pueden implementar para el desarrollo de un sistema de registro de mantenimiento correctivo que propone el mismo estándar, así como también los diagramas de UML y la metodología RUP. La razón de utilizar estas herramientas es porque la empresa BD Bard ya cuenta con las licencias para la utilización de estas herramientas, por lo que no se tiene que invertir dinero en una herramienta de programación para el desarrollo de *BOC-DWT*.

#### **1.11.1. Productos de la tesis**

1. Artefactos UML de análisis y diseño orientado a objetos de *BOC-DWT* basado en ISA-95.
2. Interface de usuario en Visual Studio.
3. Código fuente en C# del prototipo de software basado en ISA-95 para el registro de mantenimiento correctivo.
4. Base de datos en SQL Server 2019.
5. Procedimiento de implementación del prototipo (modelo funcional).
6. Documentación generada durante el desarrollo de este proyecto basado en la metodología RUP.

### 1.12. Proceso de Desarrollo Unificado

Se utilizo la metodología del Proceso de Desarrollo Unificado para incrementar gradualmente la funcionalidad del prototipo (ver *Figura 1.8*) (Metodoss, 2018).



**Figura 1.8 Modelo de metodología del Proceso de Desarrollo Unificado para el desarrollo de BOC-DWT.**

De la *Figura 1.8*, las disciplinas que se eligieron para la estructura de este proyecto de investigación son requerimientos, análisis y diseño, implementación y pruebas. Se realizo por fases de acuerdo con las fases en cada disciplina para generar los artefactos de cada fase (Metodoss, 2018).

Los pasos a seguir para la metodología RUP para generar los productos de la presente tesis consisten en:

1. Definición de los requerimientos del prototipo de software. En esta parte se definirán las funcionalidades con las que debe contar el software, estas serán proporcionadas de acuerdo a las necesidades de BD Bard.

2. Análisis del problema a resolver. En esta parte se analizará la problemática principal presentada y por ende será la prioridad de acuerdo al desarrollo de BOC-DWT.
3. Diseño del software BOC-DWT prototipo basado en el estándar ISA-95 (UML). En esta parte se utilizará el diagrama de UML para el diseño de BOC-DWT la cual contendrá las clases a desarrollar.
4. Desarrollo del software BOC-DWT prototipo con lenguaje C# y SQL. En esta parte él se utilizarán las herramientas de Visual Studio y SQL para el desarrollo de BOC-DWT.
5. Diseño y realización de las pruebas de funcionamiento. En esta parte se realizarán las pruebas de funcionalidad para comprobar que los requerimientos funcionales fueron cumplidos.

- **Fase de inicio**

Las actividades desarrolladas en la fase de inicio de acuerdo con cada disciplina son las siguientes.

- **Requerimientos:** En esta disciplina se desarrolló una investigación estableciendo los requerimientos funcionales y no funcionales, así como también los modelos a utilizar del ISA-95.
- **Análisis y Diseño:** Se desarrollaron las clases, atributos y métodos a utilizar en el desarrollo del software. En esta iteración también se construye el caso del problema que BOC-DWT debe resolver.
- **Pruebas:** Se analizaron los diferentes requerimientos funcionales a realizar para que BOC-DWT sea funcional y cumpla con las expectativas de la empresa. El artefacto que se construyó en la primera iteración son los casos de prueba que se realizaron.

- **Fase de Elaboración**
  - En esta fase, ya se tiene el conocimiento de lo que se desarrolló, es necesario realizar los diagramas. Las actividades de esta fase son las siguientes:
  - **Análisis y Diseño:** Se desarrollo el análisis y diseño de los diagramas de casos de uso del sistema, diagrama de secuencia y diagrama de clases.
  - **Implementación:** Se desarrollo el bosquejo de las pantallas de BOC-DWT, así como también la tabla de correspondencia de las clases de ISA-95 con las utilizadas en BOC-DWT.
  - **Pruebas:** Se creó la librería de clases y a su vez el diagrama de la base de datos.
  
- **Fase de Construcción**
  - En la fase de Construcción se desarrolló la sintaxis de la base de datos para la realización de consultas, interfaz de usuario y se creó el código fuente para la generación de clases de acuerdo con el estándar de ISA-95.
  - **Análisis y diseño:** Se desarrollo la sintaxis para las consultas y actualización de datos de la base de datos.
  - **Implementación:** En esta fase se desarrolló la interfaz de usuario con la cual los usuarios de la aplicación tienen interacción en su uso.

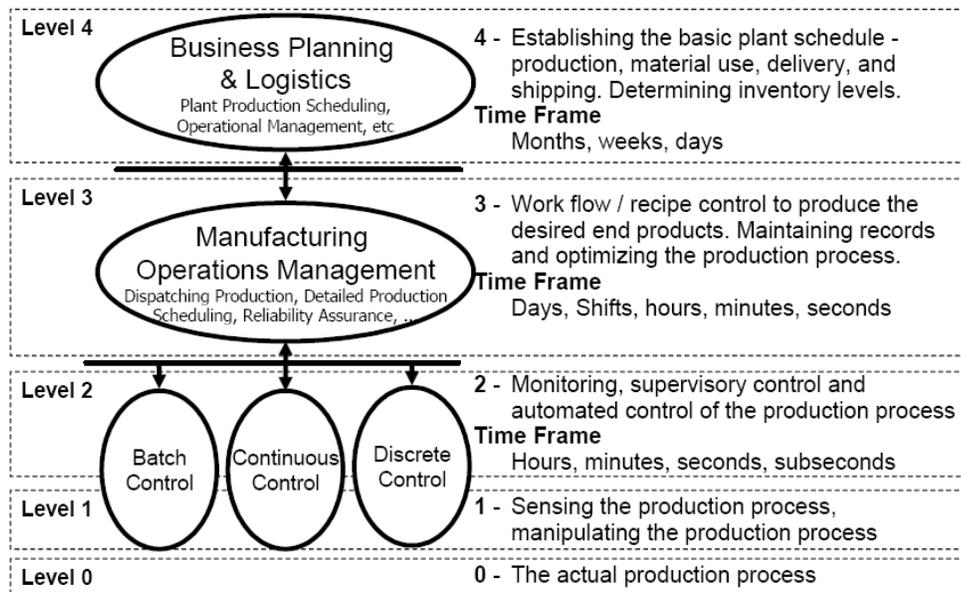
- **Pruebas:** En esta fase se creó el código fuente para generar las clases que contiene BOC-DWT.
  
- **Fase de Construcción**
  - Para la última fase se revisó que el análisis y diseño que se planteó cumple con los requerimientos funcionales. En esta iteración solo se contemplan las siguientes disciplinas las cuales se realizarán en una sola iteración.
  
  - **Pruebas:** Se realizaron las pruebas unitarias de nuevo para asegurar que los requerimientos funcionales cumplen con el objetivo del proyecto, además que los modelos de clases realizados estén implementados en BOC-DWT.

## CAPÍTULO 2: MARCO TEÓRICO

### 2.1. Estándar ISA-95

El propósito del estándar ISA-95 es definir la interface entre las funciones de control y las funciones empresariales, basado en el modelo *Purdue Reference Model for CIM* (forma jerárquica) publicado por ISA. El objetivo es reducir el riesgo, costo y errores asociados con la implementación de estas interfaces. El estándar define el intercambio de información para que sea robusto, seguro y efectivo en costo. El mecanismo de intercambio debe preservar la integridad de cada sistema y sus funciones de control (ISA-95, 2011).

ISA-95 considera la jerarquía de tres niveles básicos de funciones en la automatización, como lo muestra el diagrama de la *Figura 2.1*, extraído del estándar.



**Figura 2.1 Jerarquía funcional de los niveles de automatización de ISA-95.**

Éste diagrama comienza con el nivel 4 en la cima que corresponde al nivel de planeación de negocios y logística, aquí es donde se establece el plan de producción básico de la planta, uso de materiales, entregas y embarques. Se manejan márgenes de tiempo de meses, semanas y días (ANSI/ISA-95.00.01-2010, 2010).

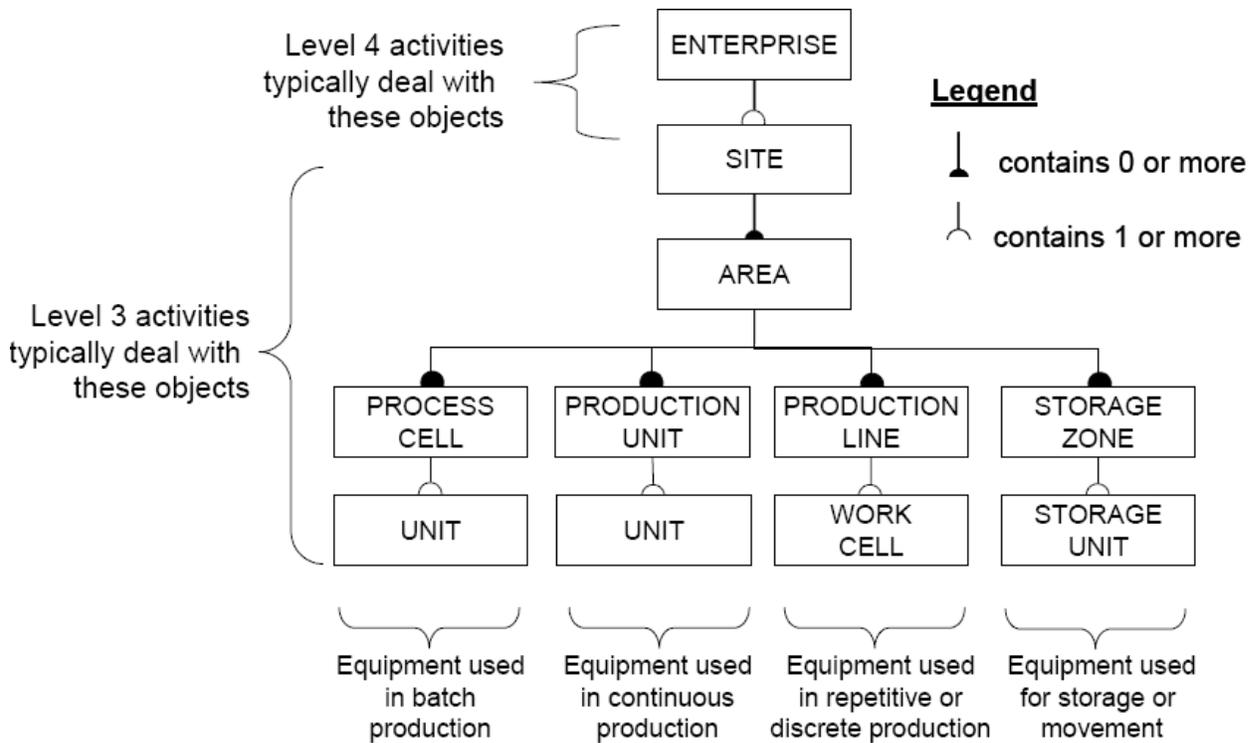
Después se encuentra el Nivel 3 que corresponde al manejo de Operaciones de Manufactura donde se realizan los flujos de trabajo e instrucciones de trabajo para producir los productos deseados, almacenamiento de registros y optimización de la producción del proceso. Aquí se manejan márgenes de tiempo de días, turnos, horas, minutos y segundos (ANSI/ISA-95.00.01-2010, 2010).

El nivel 2 corresponde al monitoreo, supervisión y control, así como el control automático del proceso de producción (nivel de control de proceso). Los márgenes de tiempo aquí son horas, minutos, segundos y fracciones de segundos (ANSI/ISA-95.00.01-2010, 2010).

El nivel 1 es el de censado y manipulación del proceso de producción.

Por último, el nivel 0 corresponde al proceso real.

El estándar también muestra la jerarquía de automatización desde la perspectiva física del equipo (ver *Figura 2.2*).



**Figura 2.2 Jerarquía del Equipo de ISA-95.**

En el nivel superior de la jerarquía de equipo está el nivel “Empresa” (*Enterprise*), que a su vez se compone de uno o más “Sitios” (*Site*), los Sitios a su vez se componen de cero o más de un “Áreas” (*Area*), y las Áreas se componen de cualquier combinación de cero o más “Celdas de Proceso” (*Process Cell*) y/o “Unidades de Producción” (*Production Unit*) y/o Líneas de Producción (*Production Line*) y/o “Zonas de Almacenamiento” (*Storage Zone*) (ANSI/ISA-95.00.01-2010, 2010).

Cada una de las Celdas de Proceso y Unidades de Producción tienen una o más “Unidades” (*Unit*), las Líneas de Producción tienen una o más “Celdas de Trabajo” (*Work Cell*) y por último las “Zonas de Almacenamiento” tienen una o más “Unidades de Almacenamiento” (*Storage Unit*) (ANSI/ISA-95.00.01-2010, 2010).

Además de las jerarquías anteriormente mencionadas, el estándar ISA-95 contiene un modelo en la parte 3 para el manejo de operaciones de mantenimiento la cual es definida como un conjunto de actividades que coordinan, dirigen y dan seguimiento a las funciones que mantiene al equipo, las herramientas y los activos relacionados para asegurar su disponibilidad para la fabricación y asegurar la programación del mantenimiento reactivo, periódico, preventivo o proactivo. La gestión de operaciones de mantenimiento admite las siguientes cuatro categorías principales de mantenimiento (ANSI/ISA-95.00.01-2013, 2013). Actividades incluidas:

La primera actividad es proporcionar respuestas de mantenimiento a los problemas del equipo (o también llamado mantenimiento correctivo) (ANSI/ISA-95.00.01-2013, 2013).

La segunda actividad consiste en programar y realizar el mantenimiento en un ciclo periódico basado en el tiempo o ciclos (o también llamado mantenimiento preventivo) (ANSI/ISA-95.00.01-2013, 2013).

La tercera actividad proporciona una condición basada en el mantenimiento derivado para obtener información del equipo o deducida del mismo, aquí en esta actividad se incluye el mantenimiento predictivo basado en el pronóstico de una futura falla (ANSI/ISA-95.00.01-2013, 2013).

La cuarta actividad optimiza el recurso del rendimiento y la eficiencia operativa de los recursos, esto es considerado como parte del análisis de producción y proceso (ANSI/ISA-95.00.01-2013, 2013).

El manejo de las operaciones de mantenimiento puede incluir:

- Proveer todas las actividades de manejo de mantenimiento correctivo, preventivo y predictivo.
- Proveer monitoreo de las actividades del equipo anticipando fallas futuras, incluyendo equipos con autodiagnóstico y desarrollar actividades de diagnóstico para el personal.
- Desarrollar costos del mantenimiento y reportes de eficiencia.
- Coordinar y monitorear el trabajo contratado (Proveedores externos).
- Supervisión del mantenimiento contratado.
- Reportes sobre el mantenimiento realizado, incluidas las piezas de repuesto utilizadas, la mano de obra de mantenimiento y los costos de mantenimiento.
- Coordinar el trabajo planificado con los operadores y la supervisión de la planta.
- Realizar verificaciones de rendimiento de los equipos de producción.
- Brindar soporte con las necesidades de cambio de producto que involucran cambios de equipo (Setup).
- Monitorear y actualizar los archivos del historial de mantenimiento.

Una vez conocidas las actividades y manejo de operaciones de mantenimiento que *BOC-DWT* utilizará para el registro de mantenimiento correctivo, el estándar define a un nivel mayor de detalle la estructura y las funciones internas de un sistema de mantenimiento (ver *Figura 2.3*) (ANSI/ISA-95.00.01-2013, 2013):



diseños (dibujos), documentación de ingeniería, especificaciones, manuales, diagnósticos y procedimientos de mantenimiento preventivo. La documentación de la maquinaria es utilizada para instruir o entrenar al personal de mantenimiento para el desarrollo de actividades específicas de mantenimiento, pero también se incluyen documentos para la confirmación del entrenamiento, estos son firmados por el personal una vez recibido dicho entrenamiento. En la definición del mantenimiento también se incluye los KPIs (Key performance indicators en inglés) de mantenimiento (ANSI/ISA-95.00.01-2013, 2013).

- Capacidad de Mantenimiento, es el conjunto de la capacidad futura prevista, comprometida e inalcanzable de los recursos utilizados en las actividades de mantenimiento. La capacidad de mantenimiento incluye los recursos a utilizar y está basada en (ANSI/ISA-95.00.01-2013, 2013):
  - Personal, basado en una calificación, ésta puede ser entrenamiento, experiencia y disciplina en un dispositivo o máquina.
  - Máquina, basado en la calibración, transporte o el uso de una herramienta especial.
  - Material, consumibles para un mantenimiento preventivo, los cuales pueden ser refacciones.
  
- Solicitudes de mantenimiento, son solicitudes de servicios de mantenimiento. Las soluciones de mantenimiento pueden ser para mantenimiento correctivo, preventivo, proactivo y basado en la condición. Las solicitudes de mantenimiento se pueden generar a partir de las actividades de nivel 3, 4 o de nivel inferior, sobre los procesos de negocio y operaciones implementadas. Los instrumentos y controladores inteligentes en el nivel 1 y los sistemas de control de nivel 2 pueden generar

automáticamente solicitudes de servicios de mantenimiento basadas en la condición. Además, puede haber solicitudes de servicios de "mejora", cambio de producción o asistencia en problemas de rendimiento de la producción. Esto a menudo requiere una coordinación significativa con las actividades de análisis de procesos y producción para realizar pruebas e implementar mejoras o cambios (ANSI/ISA-95.00.01-2013, 2013).

- Respuesta de mantenimiento, es información documentada en el mantenimiento correctivo o acción de mejora como es especificada en el registro de requisición (ANSI/ISA-95.00.01-2013, 2013).
- Procedimientos de mantenimiento a equipo específico, son instrucciones para una máquina que es enviada al nivel 2 basada en tareas específicas asignadas. El mantenimiento puede extenderse más allá del equipo como condiciones ambientales necesarias para el ambiente (ANSI/ISA-95.00.01-2013, 2013).
- Resultados de mantenimiento, es la información recibida del nivel 2 en respuesta a los comandos de mantenimiento y procedimientos. Los resultados del mantenimiento típicamente corresponden a los mantenimientos completados. Esto puede incluir información detallada de las actividades del mantenimiento recopilada durante la ejecución del mantenimiento (ANSI/ISA-95.00.01-2013, 2013).
- Datos del estado de salud del equipo, es información recibida como resultado del nivel 2 o nivel 1 que indica el estado del equipo. Los datos pueden representar condiciones pasadas, presentes o futuras. Los datos sobre el estado de salud del equipo no suelen estar asociados con un

comando o procedimiento de mantenimiento (ANSI/ISA-95.00.01-2013, 2013).

- Definición del manejo del mantenimiento, es el conjunto de actividades que definen el manejo y recopilación de información e instrucciones necesarias para completar las tareas de mantenimiento. La definición del manejo del mantenimiento incluye (ANSI/ISA-95.00.01-2013, 2013):
  - Manejo de documentos como instrucciones de mantenimiento o de trabajo, documentación del proveedor, diseños CAD, registros de una base de datos y herramientas.
  - Derivar y gestionar un conjunto de definiciones de mantenimiento.
  - Gestionar los cambios en las definiciones de mantenimiento. Esto puede incluir la capacidad de encaminar cambios a través de un proceso de aprobación apropiado, gestión de versiones de definiciones, seguimiento de modificaciones y control de seguridad de las definiciones.
  - Brindar definiciones de mantenimiento de otras aplicaciones, equipo, personal o actividades.
  - Proveer definiciones de mantenimiento para otras aplicaciones, equipo, personal y actividades.
  - Gestionar el intercambio de definiciones para mantenimiento con las funciones del nivel 4, como el nivel de detalle requerido para las operaciones del negocio.
  - Optimizar las definiciones de mantenimiento basadas en el análisis del rendimiento del proceso y del mantenimiento.
  - Generar y mantener definiciones de mantenimiento no relacionadas con el equipo de producción, como para el mantenimiento del equipo de mantenimiento y la validación del equipo de mantenimiento.

- Gestionar las definiciones KPI asociadas con mantenimiento.
  - Gestionar las definiciones de mantenimiento relacionadas para la seguridad y procedimientos ambientales.
  - La gestión de definiciones de mantenimiento incluye la gestión de la distribución de definiciones de mantenimiento. Pueden existir algunas definiciones de mantenimiento para equipos de Nivel 2 y Nivel 1. Cuando ese sea el caso, las descargas de esta sala de información se coordinarán con otras funciones de gestión de operaciones de fabricación para reducir el impacto en la producción. Esta información puede incluirse como parte de los comandos y procedimientos de mantenimiento cuando la descarga se inicia como parte de una actividad de gestión de ejecución de mantenimiento.
- 
- Gestión de recursos de mantenimiento, son la colección de actividades que gestionan la información sobre el estado de los recursos y las relaciones entre los recursos utilizados dentro del dominio de control de mantenimiento. Los recursos gestionados pueden incluir equipos de mantenimiento, herramientas de mantenimiento, personal (con habilidades), documentación y material y energía utilizados en el mantenimiento. El recurso de mantenimiento puede incluir (ANSI/ISA-95.00.01-2013, 2013):
    - Mantener información sobre el personal de mantenimiento, incluida la información de calificación, como el estado de calificación y los resultados de las pruebas de calificación, como se define en el modelo de personal ISA-95 Parte 1 e ISA-95 Parte 2.

- Mantener información sobre el equipo utilizado en el mantenimiento y las pruebas de capacidad del equipo, como se define en el modelo de equipo ISA-95 Parte 1.
- Mantener información sobre suministros de mantenimiento, definidos como materiales consumibles, como se describe en el modelo de material ISA-95 Parte 1.
- Mantener información sobre la salud y el estado, la asignación y el estado de disponibilidad de los recursos que se usarán y se usarán en todas las actividades de mantenimiento de Nivel 3.
- Coordinar y monitorear el trabajo contratado.
- Supervisar el mantenimiento requerido.

El propósito de la gestión de recursos de mantenimiento es aumentar de forma segura la producción total de una planta a un costo de mantenimiento reducido por unidad de producción. Lo logra proporcionando información oportuna para que el personal de operaciones de fabricación tome decisiones óptimas con respecto a las operaciones del proceso y el mantenimiento de los equipos (ANSI/ISA-95.00.01-2013, 2013).

- Programación del mantenimiento detallado, es un conjunto de actividades que generan un programa para mantenimiento. Las tareas detalladas del programa de mantenimiento pueden incluir (ANSI/ISA-95.00.01-2013, 2013):
  - Revisión de las requisiciones de mantenimiento.
  - Confirmación o negación de la requisición del mantenimiento.
  - Determinar la prioridad de la requisición, el nivel de esfuerzo y la disponibilidad de todos los recursos.

- Programar las requisiciones de mantenimiento para ser realizadas dentro de un programa de una o más órdenes de trabajo.
  - Coordinar el plan de trabajo con los operadores y supervisión de la planta.
  - Un programa de mantenimiento puede ser generado por cada sitio o área, basado en órdenes de trabajo requeridas y recursos disponibles (personal, equipo, máquinas disponibles y materiales). La programación del mantenimiento detallado mantiene los requisitos y a su vez desarrolla la organización necesaria de las órdenes de trabajo. Las solicitudes de mantenimiento pueden originarse en una o más funciones de nivel superior, desde otras actividades de nivel 3 o incluso directamente desde equipos inteligentes.
- Despacho de mantenimiento, es el conjunto de actividades que asignan y envían órdenes de trabajo a los recursos de mantenimiento apropiados según lo identificado por el programa de trabajo para las definiciones de mantenimiento. El envío comunica la tarea a realizar y los recursos que se utilizarán y puede implicar el envío del trabajo a los empleados o contratistas para realizar el trabajo.

Los recursos no asignados como parte del programa de trabajo pueden ser asignados por la actividad de despacho de mantenimiento (ANSI/ISA-95.00.01-2013, 2013).
  - Manejo de la ejecución del mantenimiento, es un conjunto de actividades que son directamente desarrolladas por el departamento de mantenimiento. El manejo de la ejecución del mantenimiento puede tener la responsabilidad de (ANSI/ISA-95.00.01-2013, 2013):

- Asegurarse de que los procedimientos de mantenimiento y regulaciones son seguidas durante las actividades de mantenimiento.
  - Documentar el estatus y resultados del trabajo desarrollado.
  - Informar al despacho de mantenimiento y/o la programación detallada del mantenimiento cuando eventos imprevistos provoquen la incapacidad de cumplir con los requisitos de trabajo.
  - Confirmar que el trabajo fue realizado para ser aceptado por los estándares de calidad.
  - Asegurarse que los recursos son los indicados para el mantenimiento.
  - Verificar que el equipo y las certificaciones del personal son válidas para las tareas asignadas.
  - Ayudar con las necesidades de cambio de producto que involucran cambios de equipo.
- 
- Recopilación de datos de mantenimiento, son el conjunto de actividades que resume e informa sobre la información y los eventos relacionados con la disposición de la orden de trabajo. La información puede incluir el estado actual, el tiempo requerido, el tiempo de inicio, el tiempo actual, el tiempo estimado de finalización, el tiempo real, los recursos utilizados e información adicional para presentar un historial de mantenimiento completo para las órdenes de trabajo existentes y las órdenes de trabajo anteriores (ANSI/ISA-95.00.01-2013, 2013).
  - Seguimiento de mantenimiento, es la recopilación de actividades que gestionan la información sobre la utilización de recursos para realizar actividades de mantenimiento y la efectividad relativa de los resultados de la actividad de mantenimiento (ANSI/ISA-95.00.01-2013, 2013).

El seguimiento de mantenimiento incluye la actividad de generar o actualizar registros relacionados con el estado y la usabilidad del equipo mantenido. Esto puede incluir los registros necesarios para fines normativos o de gestión de la calidad (ANSI/ISA-95.00.01-2013, 2013).

- Análisis de rendimiento de mantenimiento, es el conjunto de actividades que examinan al personal, al equipo, los activos físicos y el historial de materiales para identificar áreas problemáticas o áreas de mejora (ANSI/ISA-95.00.01-2013, 2013).

El análisis de las funciones de rendimiento puede incluir las siguientes condiciones:

- ¿Qué equipo puede fallar si no recibe la intervención de mantenimiento?
- ¿Qué intervención se debe realizar y en qué plazo?
- ¿Dónde se pueden reducir las actividades de mantenimiento preventivo de rutina?
- ¿Dónde se pueden concentrar los esfuerzos para mejorar el rendimiento de los activos mediante la eliminación de fallas costosas o repetitivas?

El análisis del rendimiento del mantenimiento también puede ayudar a la planificación de operaciones y en producción identificando condiciones tales como (ANSI/ISA-95.00.01-2013, 2013):

- ¿Debería hacerse algún ajuste en el proceso para prolongar la vida útil de los activos críticos de la planta?
- ¿A qué nivel puede continuar la producción sin incurrir en un riesgo inaceptablemente alto de desaceleración del proceso, tiempo de inactividad, problemas de calidad o paradas de seguridad?
- ¿Cuál es la probabilidad de producir con éxito una cantidad específica de producto en un período de tiempo?

El análisis del rendimiento del mantenimiento también puede incluir un análisis de trazabilidad de recursos, que rastrea el historial de todos los recursos en términos de las acciones de mantenimiento y los eventos que se ocuparon de los recursos. Esto puede incluir información como (ANSI/ISA-95.00.01-2013, 2013):

- ¿Qué materiales se utilizaron en las actividades de mantenimiento?
- ¿Qué herramientas se utilizaron en las actividades de mantenimiento y qué equipo se mantuvo?
- ¿Qué personal participó en las actividades de mantenimiento?
- Desarrollar informes de rendimiento y costos de mantenimiento.
- Informar sobre el mantenimiento realizado, incluidas las piezas de repuesto utilizadas, el trabajo de mantenimiento y los costes de mantenimiento.

Esta información sobre el mantenimiento que proporciona resúmenes del desempeño pasado e indicaciones del desempeño futuro o posibles problemas futuros. En conjunto, esta información se define como "indicadores de mantenimiento". Una de las actividades dentro del análisis es la generación de indicadores de mantenimiento. Esta información se puede utilizar internamente dentro de las operaciones de fabricación para mejoras y optimización, o, si hay un proceso comercial de recepción que requiere la información, se puede enviar a procesos comerciales de nivel superior para análisis y decisiones adicionales (ANSI/ISA-95.00.01-2013, 2013).

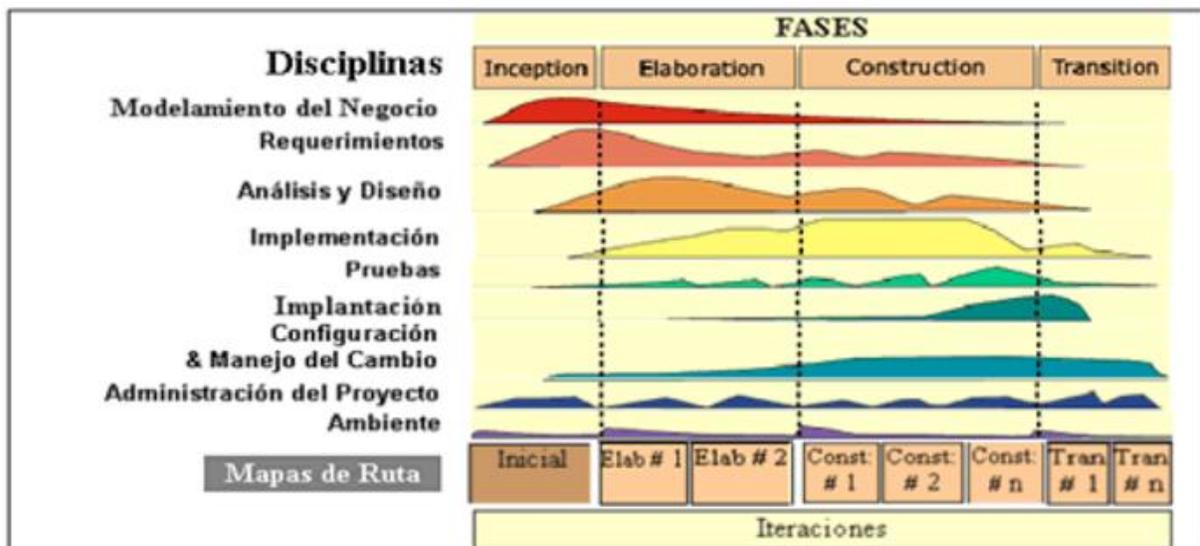
## 2.2. BOC-DWT

El sistema *BOC-DWT* (*Bard Operation Center - Downtime*) es definido como el sistema que asegura el control de los procesos de registro de mantenimiento

correctivo. Ya que guarda registros en una base de datos, la cual es excelente para la toma de decisiones y es uno de los actores de la integración empresarial. *BOC-DWT* es un sistema específico para la empresa BD Bard y es definido como la interface entre negocios y manufactura. Materializa la frontera entre los planes provisionales y los hechos. *BOC-DWT* se encarga del registro de las actividades del mantenimiento correctivo, de la generación de reportes de mantenimiento correctivo, lo cual es ideal para hacer ajustes a planes y su seguimiento a planes de producción y mantenimiento preventivo (Masood, 2004).

### 2.3. Metodología de Desarrollo Unificado

La metodología RUP, abreviatura de Proceso de Desarrollo Unificado, es un proceso de desarrollo propio de la ingeniería de software, creado por Rational Software, adquirida por IBM, por lo que es una marca en el área de software. Proporciona técnicas que deben seguir los miembros del equipo de desarrollo de software con el fin de aumentar su productividad en el proceso de desarrollo. Una representación típica de esta estructura se muestra en la *Figura 2.4* (Torossi, 2011).



**Figura 2.4 Metodología RUP.**

Al terminar una iteración se comprueba que lo que se ha hecho efectivamente cumple con los requisitos establecidos, también se verifica que funciona correctamente. El propio cliente evalúa el producto. Cuando hay que hacer un cambio, este puede consistir en un nuevo ciclo.

### 2.3.1. Ventajas

- Es el proceso de desarrollo más general de los existentes actualmente. Es decir, este proceso es de los más utilizados para el desarrollo del software por la mayoría de las empresas, pues su enfoque es bastante óptimo y tiende a ser una metodología viable para la mayoría de éstas.
- Es una forma disciplinada de asignar tareas y responsabilidades en una empresa de desarrollo, pues los roles están muy bien definidos y dictan quien realiza cada actividad, dependiendo del área en el que se desarrollará, de esta manera es bastante útil para definir roles en los proyectos.
- Mantenimiento más sencillo y modificaciones locales. Esta es una ventaja muy importante, pues si el proceso así lo permite, es bastante más fácil poder realizar un cambio al proyecto en un futuro, sin generar pérdidas o retrasos tan notorios o sobresalientes.
- Reutilización. Los roles pueden ser reutilizados en proyectos futuros, dando como resultado una mejor organización al proyecto y menos utilización de recursos o tiempo, aspectos que se pueden emplear directamente en el proyecto.
- Un proceso de software hecho a la medida para ser publicado y hacerlo accesible para todo el equipo del proyecto. Esto quiere decir que cualquiera que se encuentre trabajando en el proyecto pueda acceder a este con más facilidad, evitando problemas relacionados a este tipo de cuestiones.

- Ofrece a cada usuario, un filtrado personalizado de la definición del proceso publicado, acorde con su rol dentro del proyecto.

### **2.3.2. Inconvenientes**

- Por el grado de complejidad puede ser no muy adecuado. Debido a que es un proceso bastante grande y complejo es muy común que no sea el adecuado para cualquier proyecto pequeño (cosa que se explicará en la siguiente desventaja), es por eso que a veces no puede ser el adecuado.
- En proyectos pequeños, es posible que no se puedan cubrir los costos de dedicación del equipo de profesionales necesarios. Al ser una metodología bastante cara y con bastantes requerimientos en cuanto a roles (personales), a veces los costos son muy elevados, dando como resultado una imposibilidad por costear el proyecto.
- Método pesado. En muchos aspectos tiende a ser muy pesado, pues como se explicaba en los puntos anteriores, la complejidad es alta.

### **2.4. Iterativo e Incremental**

Las iteraciones deben estar controladas. Esto significa que deben seleccionarse y ejecutarse de forma planificada. Los desarrolladores se basan en la selección de lo que implementará en cada iteración en dos cosas: el conjunto de casos de uso que amplían la funcionalidad y en los riesgos más importantes que deben mitigarse.

En cada iteración los desarrolladores identifican y especifican los casos de uso relevantes, crean un diseño utilizando la arquitectura seleccionada como guía,

para implementar dichos casos de uso. Si la iteración cumple sus objetivos, se continúa con la próxima. Si no, deben revisarse las decisiones previas y probar un nuevo enfoque (Torossi, 2011).

## **2.5. Fases**

El proceso unificado se repite a lo largo de una serie de ciclos que constituye en la vida de un sistema. Cada ciclo constituye una versión del sistema. Cada ciclo constas de cuatro fases: inicio, elaboración, construcción y transición mostrado en la *Figura 2.4* (Metodoss, 2018).

## **2.6. Fase de Inicio**

La fase de inicio contiene flujos de trabajo necesarios para el acuerdo de las partes interesadas con los objetivos, la arquitectura y la planificación del proyecto (Metodoss, 2018). Si estos actores tienen un buen conocimiento, no será necesario analizar. De lo contrario, se requiere un análisis más elaborado.

Los artefactos que típicamente sobreviven a esta fase son (Torossi, 2011):

- Un enunciado de los mayores requerimientos planteados generalmente como casos de uso.
- Un boceto inicial de la arquitectura.
- Una descripción de los objetivos del proyecto.
- Una versión muy preliminar del plan del proyecto.
- Un modelo de negocio.

## **2.7. Fase de Elaboración**

La preparación será para el diseño del sistema, como complemento de la encuesta y/o documentación de casos de uso, frente a la arquitectura del sistema,

revisar el modelo de negocio para el proyecto e iniciar la versión del manual del usuario (Metodoss, 2018).

En esta fase se construyen típicamente los siguientes artefactos (Torossi, 2011):

- El cuerpo básico del software en la forma de un prototipo arquitectural.
- Casos de prueba.
- La mayoría de los casos de uso (80%) que describen la funcionalidad del sistema.
- Un plan detallado para las siguientes iteraciones.

## **2.8. Fase de Construcción**

En la fase de construcción, el desarrollo físico del software se inicia, códigos de producción, pruebas alfa, pruebas beta se llevaron a cabo al inicio de la fase de transición. Se debe aceptar las pruebas, procesos estables y de prueba (Metodoss, 2018).

Los artefactos producidos durante esta fase son (Torossi, 2011):

- El Sistema de software.
- Los casos de prueba.
- Los manuales de usuario.

## **2.9. Fase de Transición**

En esta fase es la entrega de software, se lleva a cabo el plan de despliegue y entrega, el seguimiento y la calidad del software. Se lanzan versiones de productos que se van a entregar y se evalúa la satisfacción del cliente (Metodoss, 2018).

Los artefactos construidos en esta fase son los mismos que en la fase de construcción. El equipo se encuentra ocupado fundamentalmente en corregir y extender la funcionalidad del sistema desarrollado en la fase anterior (Torossi, 2011).

### **2.10. Disciplinas**

Cada fase se subdivide en iteraciones. En cada iteración se desarrolla en secuencia un conjunto de disciplinas o flujos de trabajos. A continuación, se describen las disciplinas que intervienen en las iteraciones de la metodología RUP (ver *Figura 2.4*).

### **2.11. Modelado de Negocio**

El modelado de negocios explica cómo describir la visión de una organización en la que se implementará el sistema y cómo utilizar esta visión como base para describir los procesos, funciones y responsabilidades (Metodoss, 2018).

### **2.12. Requerimientos**

Por completo se entiende que todos los servicios requeridos por el usuario deben especificarse y la consistencia significa que ninguna definición de requisitos debe contradecir a otra. En la práctica, para sistemas grandes y complejos, es casi imposible lograr que los requisitos sean consistentes y completos en la versión inicial del documento (Metodoss, 2018).

### **2.13. Análisis y Diseño**

El modelo de diseño consta de clases de diseño estructurados en paquetes y subsistemas con interfaces bien definidas, en representación de lo que se convertirá

en componentes de la aplicación. También contiene descripciones de cómo los objetos de estas clases colaboran para llevar a cabo el diseño de casos de uso (Metodoss, 2018).

#### **2.14. Implementación**

Los efectos de la aplicación son (Metodoss, 2018):

- Para configurar el código de la organización en términos de subsistemas de aplicación organizados en capas.
- Para llevar a cabo las clases y objetos en términos de componentes (archivos de código fuente, binarios, ejecutables, etc.).
- Para probar los componentes desarrollados como unidades.
- Incorporar los resultados producidos por los ejecutores individuales (o equipos) en un sistema ejecutable.

#### **2.15. Pruebas**

Los fines de la disciplina de Pruebas son (Metodoss, 2018):

- Comprobar la interacción entre los objetos.
- Comprobar la correcta integración de todos los componentes de software.
- Comprobar que todos los requisitos han sido ejecutados correctamente.
- Identificar y asegurar que los defectos se tratan antes de la implementación de software.
- Asegurar de que todos los defectos son corregidos, revisados y cerrados.

#### **2.16. Artefactos**

En RUP en cada una de sus fases se realizan una serie de artefactos para conocer la función y estructura de un programa.

Un artefacto puede ser:

- Un documento: como un Caso de Negocio o un documento de la arquitectura del Software.
- Un modelo: como un modelo de caso de uso.
- Un elemento de un modelo: como una sola clase de todo el Diagrama de Clases.

Los artefactos que se utilizan en esta investigación se definen en las subsecciones siguientes.

### **2.17. Requerimientos Funcionales**

Los requerimientos funcionales hacen referencia a la descripción de las actividades y servicios que un sistema debe proveer. Normalmente este tipo de requerimientos están vinculados con las entradas, las salidas, los procesos y los datos a almacenar en el sistema (Alarcon, 2006).

Las funciones del sistema son lo que éste habrá de hacer; por ejemplo, agregar diferentes requisiciones de producción. Hay que identificar las funciones y listarlas en grupos cohesivos y lógicos para poder tener todos los requerimientos funcionales (Larman, 1999).

### **2.18. Casos de Uso**

Un diagrama de casos de uso representa las interacciones entre el sistema y los sistemas externos y los usuarios. En otras palabras, describe gráficamente quien utiliza el sistema y la forma en que los usuarios esperan interactuar con el Sistema. Como menciona Alarcón, “un caso de uso es la descripción paso a paso de las actividades que forman el caso de uso” (Alarcon, 2006).

## 2.19. Diagrama de Clases

Los diagramas de clase describen los tipos de objetos de un sistema, así como los distintos tipos de relaciones que pueden existir entre ellos. Los diagramas de clase se convierten así en la técnica más potente para el modelado conceptual de un sistema software, la cual suele recoger los conceptos clave del modelo de objetos subyacente al método orientado a objetos que la incorpora (Pardo, 1988).

### 2.19.1. Diagrama de Clases (Vista de Análisis)

El diagrama de clase representa los conceptos en el dominio del problema que se está estudiando. Este modelo debe crearse con la mayor independencia posible de la implementación final del sistema (Pardo, 1988).

### 2.19.2. Diagrama de Clases (Vista de Diseño)

El diagrama de clases refleja las interfaces de las clases, pero no su implementación. Aquí las clases aparecen más cercanas a los tipos de datos, ya que un tipo representa una interfaz que puede tener muchas implementaciones diferentes.

## 2.20. KPIs

*Key Performance Indicators KPIs* (ó Indicadores Clave de Desempeño). También conocidos como *Key Success Indicators (KSI)* (Indicadores Clave de Éxito), ayudan a una organización a definir y medir el progreso hacia los objetivos organizacionales (F., 2011).

Una vez que una organización ha analizado su misión, identificado sus partes interesadas y ha definido sus objetivos, necesita una manera de medir el progreso

hacia esos objetivos. Los indicadores clave de desempeño son esas medidas (F., 2011).

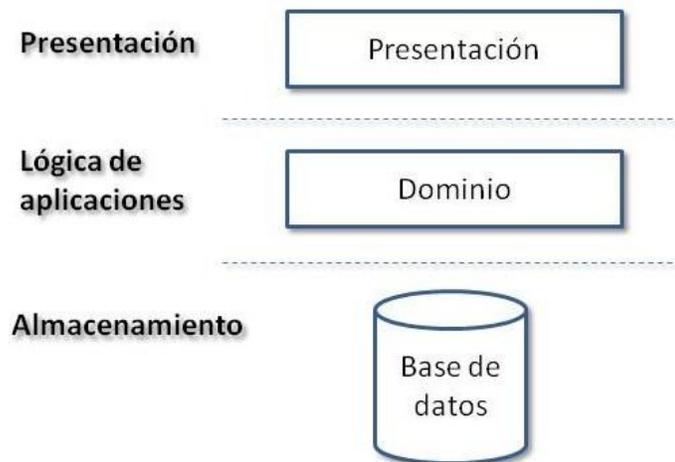
*KPIs* son medidas cuantificables, previamente acordadas, que reflejan el éxito de una organización. Difieren dependiendo de la organización (F., 2011).

Una vez que se tienen definidos, se utilizan como una herramienta administrativa y para dar una imagen clara de lo que se necesita que suceda, para hacer que todos los involucrados se enfoquen en cumplir o exceder esos *KPIs* (F., 2011).

### **2.21. Software de 3 capas**

En esta sección se analiza la arquitectura de 3 capas (Lairman, 1999). La cualidad tan especial de esta arquitectura consiste en aislar la lógica de aplicación y convertirla en una capa intermedia bien definida y lógica del software (ver *Figura 2.5*). En la capa de presentación se realiza relativamente poco procesamiento de la aplicación; las ventanas envían a la capa intermedia peticiones de trabajo y ésta se comunica con la capa de almacenamiento del extremo posterior. Cada capa contiene lo siguiente:

- **La capa de Presentación:** Incluye ventanas, reportes, etcétera.
- **La lógica de aplicaciones:** Incluye tareas y reglas que rigen el proceso.
- **Almacenamiento:** Mecanismo de almacenamiento persistente.



**Figura 2.5 Arquitectura a 3 capas.**

Esta arquitectura contrasta con el diseño de dos capas, donde – por ejemplo – es colocada la lógica de aplicaciones dentro de las definiciones de ventana, que leen y escriben directamente en una base de datos; no hay una capa intermedia que separe la lógica. Una de sus desventajas es la imposibilidad de representar la lógica en componentes aislados, lo cual impide reutilizar el software. No es posible distribuir la lógica de aplicaciones en una computadora diferente (Lairman, 1999).

La arquitectura multicapas se adecúa a los sistemas de información orientados a objetos, incluye la división de las responsabilidades que encontramos en la arquitectura clásica de tres capas. Las responsabilidades se asignan a los objetos del software (Lairman, 1999).

## CAPÍTULO 3: DESARROLLO DE BOC-DWT

### 3.1. Análisis y Diseño de BOC-DWT

Para desarrollar el software BOC-DWT fue necesario analizar las Partes 1, 2, 3 y 4 del Estándar ISA-95 que se mencionaron a lo largo de este documento, así como también se utilizó la metodología RUP para la construcción del mismo.

En las siguientes secciones se describen los pasos que se realizaron para la construcción del software, incluyendo las actividades realizadas y artefactos obtenidos en las disciplinas de las fases de inicio y elaboración que propone RUP.

Los artefactos que se desarrollan en estas fases se pueden observar en la *Tabla 3.1*. En este capítulo solo se describirán los artefactos construidos en las disciplinas para las fases de inicio y elaboración.

**Tabla 3.1 Fases de Construcción del Proyecto**

Disciplinas	Inicio	Elaboración	Construcción	Transición
Requerimientos	Protocolo de investigación.			
	Definir requerimientos funcionales.			
	Definir requerimientos no funcionales.			
	Seleccionar modelos del estandar ISA-95 que se utilizarán.			
Análisis y diseño	Seleccionar las clases y atributos de los modelos del estandar ISA-95.			
	Definir casos de un alto nivel.			
	Establecer el caso del problema a resolver.			
	Realizar diagramas de clases.			
	Elaborar diagramas de secuencia.			
Modelo de análisis.				
Realizar diagrama de la base de datos.				
Implementación		Realizar bosquejo de las pantallas del software.	Crear proyecto para librería de clases.	
		Definir la tabla de correspondencia de las clases de ISA-95 con las utilizadas en el Software.	Crear código fuente para generar clases.	
			Elaborar sintaxis para hacer consultas y actualización de datos de BD.	
Pruebas	Definir formato de caso de prueba.		Diseñar interfaz de usuario.	
			Verificar que se cumplieron los requerimientos iniciales.	
				Realizar pruebas unitarias en el sistema.

### 3.2. Fase de Inicio

Para planear la inversión de adquirir un software tecnológico como lo es un sistema para el registro de mantenimiento correctivo, es necesario crear un modelo de negocio para evaluar los riesgos y las contribuciones que aportan a la empresa. Dentro del modelo de negocio se analizaron las ventajas y desventajas de implementar un sistema para el registro de mantenimiento correctivo en el cual se concluyó que efectivamente automatizar el proceso de un sistema manual a un formato electrónico trae consigo muchas más ventajas que desventajas. Es necesario determinar de manera precisa el reto empresarial que debe abordarse con la inversión de un software de mantenimiento correctivo.

El análisis de riesgo que se implementó en el modelo de negocio se puede observar en la *Figura 3.1*.

	<b>Risk Management Evaluation for Process Change Worksheet</b> (This is used to support a CR packet)	F76MC023 Revision 4 Page 1 of 1 Refer to S76QA051
---	---	--

<b>CR Originator:</b>	<b>Date:</b>	<b>CR Number:</b>
Aaron Cortez	May 25, 2021	NG-COM-25758

Document number (Infocard) and title/description	Explain: What is the process affected with the change requested?	What is changing? Change description	Adverse effect evaluation/ Rationale (This section is required for product impact)
S76PM010 Rev.5 (Maintenance Corrective Procedure)	This process will be impact into Preventive maintenance general procedure. This procedure is the general procedure for Maintenance Department.	Added a new process to create electronic records into new system BOC-DWT.	1. Server problems. 2. Create a method to create a training to personnel involved. 3. Create a culture to fill out the Downtime records into BOC-DWT by maintenance personnel. 4. Fill out the downtime records with incorrect information.

- Product/Process control impact (Fill out Table A)
- Quality System impact (Fill out Table B)

**Table A**

Current information documented within PFMEA

New failure mode(s)  Refers to S76QA051/RA0301263/DFMEA for severity rating

PFMEA product/process	Failure Mode	Severity rating	Occ./ rating	Risk Level Q1, Q2, Q3, and Q4	Does CR impact the current risk? Reduced/Increased no impact	For new failure mode and/or document updates Tracking # or CR# is required
N/A	N/A	N/A	N/A	N/A	N/A	N/A

**Table B**

Failure Rate History (24 months)	Severity rating	Occurrence/ Probability of Risk	Risk Index <small>Sev x Occ</small> Low/ Medium/ High	Reason/Purpose for CR
10	10 Incidents In 2020-2021	1 <75%	1x3=3	To improve downtime records process.

**Figura 3.1 Análisis de Riesgo de Modelo de Negocio**

En la *Figura 3.1* se describe el proceso del análisis de riesgo que se realizó para determinar los riesgos que implica desarrollar el cambio de proceso de un formato impreso a uno digital.

Uno de los factores que influyen en la adquisición del software es el uso de metodologías de implementación fáciles para los usuarios y sobre todo que exista documentación de esta implementación fácil para los usuarios y sobre todo que

exista la documentación de esta implementación, los hitos y los entregables. Con esto se le brinda a la empresa la confianza de que el proceso ha sido analizado correctamente y que el sistema está preparado para adaptarse a las contingencias que se presenten.

Durante la fase de inicio se desarrolla una descripción del producto final, se identifican y priorizan los riesgos más importantes (Torossi, 2011). Para esta iteración se han elegido las disciplinas de requerimientos, análisis y diseño, implementación y pruebas que van a ser desarrolladas en las siguientes subsecciones.

### **3.2.1. Caso del Problema**

Para describir el proceso que debe resolverse se describirá el proceso que actualmente se sigue para el registro del mantenimiento preventivo en la empresa BD Bard. En esta empresa usualmente cuando ocurre una falla o avería que incapacite la máquina en su operación, el personal de producción junto al personal de mantenimiento de equipo registra en un formato físico el evento de mantenimiento correctivo. La funcionalidad se limita a resolver el siguiente problema:

El registro de mantenimiento correctivo en la empresa BD Bard ocasiona problemas de pérdida de registros ya que los eventos son registrados en formatos físicos los cuales suelen extraviarse, otro de los problemas que esto ocasiona también es la pérdida de información ya que esta información tampoco se transfiere a una base de datos por lo que tampoco se puede utilizar la información para el análisis y tomar decisiones esenciales para el negocio. Se pretende automatizar el proceso de registro de mantenimiento correctivo que se realiza en formatos físicos impresos, se pretende que ahora sea registrado en un formato electrónico.

### 3.2.2. Requerimientos

Los requerimientos de *BOC-DWT* fueron definidos de la siguiente manera:

- R1. Es necesario que el administrador registre a los usuarios y asigne un rol dependiendo de la actividad a desempeñar en BOC-DWT.
- R2. El sistema permitirá a los usuarios crear registros de mantenimiento correctivo.
- R3. El sistema será capaz de registrar áreas nuevas.
- R4. El sistema será capaz de registrar nuevos códigos de falla.
- R5. El sistema será capaz de mostrar el historial de registros de mantenimiento correctivo y exportarlos a Microsoft Excel.

Para describir las actividades que realiza *BOC-DWT* se han elaborado los requerimientos funcionales del sistema, con esto se establecen las funciones que el sistema debe realizar y los procesos ocultos que deben ejecutarse para lograr estos requerimientos.

### 3.2.3. Requerimientos Funcionales

El software *BOC-DWT* pretende resolver el problema que se presenta en la sección 3.2.1 de este capítulo, para lograr este fin es necesario identificar los requerimientos funcionales que van a ser implementados en él. Están incluidos los módulos que son necesarios para resolver este problema, dejando por un lado los modelos de clases que no son indispensables para esta investigación.

Los requerimientos funcionales del prototipo de software *BOC-DWT* que se definieron están divididos en 7 módulos, entre los cuales tenemos a Equipment Specification que se encargará de registrar, modificar o eliminar nuevas áreas, Equipment Specification Property se encargará de registrar, modificar o eliminar

nuevos códigos de falla, Personnel Specification Property se encargará de validar usuario y contraseña de los usuarios registrados, Personnel Model se encargará de modificar la contraseña de los usuarios en caso de que se requiera, Equipment Model se encargará de registrar los eventos de mantenimiento correctivo, Parameter Specification se encargará de mostrar el historial de los eventos de mantenimiento correctivo y Personnel Specification se encargará de registrar, modificar o borrar usuarios.

A continuación, se describen a detalle los requerimientos funcionales que se seleccionaron para cada módulo que contiene el software *BOC-DWT*.

- Módulo Equipment Specification:

En este módulo se dan de alta, modifican o borran las áreas de producción en la cual se asigna un código por área y el nombre asignado por el usuario siendo de esta manera en la que el administrador hace los cambios que sean necesarios.

**Tabla 3.2 Requerimientos funcionales de Equipment Specification.**

Identificador (ID)	Función	Categoría
R.3.1	Inserta, modificar y borra áreas de producción.	Evidente
R.3.2	El sistema inserta código de área.	Oculto
R.3.3	El sistema muestra el cambio realizado del área.	Evidente

- Módulo Equipment Specification Property:

En este módulo se dan de alta, modifican o borran los códigos de falla, se asigna un código por falla y el nombre asignado por el usuario, siendo de esta manera en la que el administrador hace los cambios que sean necesarios para el registro de fallas de maquinaria.

**Tabla 3.3 Requerimientos funcionales de Equipment Specification Property.**

<b>Identificador (ID)</b>	<b>Función</b>	<b>Categoría</b>
R.4.1	Inserta, modificar y borra códigos de falla.	Evidente
R.4.2	El sistema inserta código al código de falla.	Oculto
R.4.3	El sistema muestra el cambio realizado del código de falla.	Evidente

- Módulo Personnel Specification Property:

Este módulo se encarga de validar el usuario y contraseña ingresados por el usuario que intente ingresar a BOC-DWT, de ser incorrecto alguno de los campos (usuario o contraseña), éste mostrará un mensaje indicando al usuario que hay un error.

**Tabla 3.4 Requerimientos funcionales de Personnel Specification Property.**

<b>Identificador (ID)</b>	<b>Función</b>	<b>Categoría</b>
R.1.1	Se encarga de validar usuario y contraseña del usuario.	Evidente
R.1.2	El sistema envía una notificación de bienvenida si el usuario y contraseña son correctos.	Evidente
R.1.3	El sistema envía una notificación de usuario o contraseña incorrecta en caso de no coincidir la evaluación de los campos.	Evidente

- Módulo Personnel Model:

Este módulo se encarga de cambiar la contraseña de usuario en caso de que el usuario lo requiera.

**Tabla 3.5 Requerimientos funcionales de Personnel Model.**

<b>Identificador</b>	<b>Función</b>	<b>Categoría</b>
R.1.4	Se encarga de cambiar la contraseña en caso de ser requerido por el usuario.	Evidente
R.1.5	El sistema inserta el cambio de contraseña en la base de datos.	Oculto
R.1.6	El sistema envía una notificación de contraseña modificada cuando se ingrese la nueva contraseña.	Evidente

- Módulo Parameters Specification:

Este módulo se encarga de registrar los eventos de mantenimiento correctivo por parte del usuario mecánico, se indicará el número de orden, falla, número de máquina, tiempo, código de falla y área. Aquí es donde se procesa el evento de mantenimiento correctivo en el cual se insertan o guardan los datos en la base de datos de SQL.

**Tabla 3.6 Requerimientos funcionales de Parameters Specification.**

<b>Identificador</b>	<b>Función</b>	<b>Categoría</b>
R.5.1	Muestra el historial de eventos de mantenimiento correctivo.	Evidente
R.5.2	El sistema permite filtrar por área el historial de mantenimientos correctivos.	Evidente

R.5.3	El sistema permite filtrar por falla el historial de mantenimientos correctivos.	Evidente
-------	--	----------

- Módulo Equipment Model:

En este módulo se pueden visualizar los registros o historial de eventos de mantenimiento correctivo, el cual puede ser filtrado por área o falla y estos datos se pueden exportar a un formato Excel para ser utilizados en el análisis y toma de decisiones.

**Tabla 3.7 Requerimientos funcionales de Equipment Model.**

Identificador	Función	Categoría
R.2.1	Captura el registro de mantenimiento correctivo de una máquina de producción.	Evidente
R.2.3	El sistema inserta No. de orden.	Evidente
R.2.4	El sistema inserta Falla.	Evidente
R.2.5	El sistema registra No. Máquina.	Evidente
R.2.6	El sistema registra Tiempo (Minutos).	Evidente
R.2.7	El sistema registra Cod. Falla.	Evidente
R.2.8	El sistema registra Área.	Evidente
R.2.9	El sistema puede buscar fallas mediante el buscador de fallas.	Evidente
R.2.10	El sistema muestra el historial de fallas.	Evidente

- Módulo Personnel Specification:

En este módulo se dan de alta, modifican o borran los usuarios del personal de mantenimiento de equipo, se inserta el usuario, contraseña y nombre, siendo de esta manera en la que el administrador hace los cambios que sean necesarios.

**Tabla 3.8 Requerimientos funcionales de Personnel Specification.**

<b>Identificador</b>	<b>Función</b>	<b>Categoría</b>
R.1.7	Inserta, modificar y borra usuarios de mantenimiento de equipo.	Evidente
R.1.8	El sistema inserta Nombre de usuario.	Evidente
R.1.9	El sistema inserta Nombre de personal de mantenimiento.	Evidente
R.1.10	El sistema inserta contraseña de usuario.	Evidente

### 3.2.4. Casos de Uso de Alto Nivel

Para describir la interacción de los usuarios con los procesos que va a realizar *BOC-DWT* se han diseñado los casos de uso de alto nivel, en los cuales se describen las acciones y los actores involucrados en cada procedimiento.

Los actores que intervienen en las funciones de *BOC-DWT* son 3, en la tabla 3.9 se describen las funciones de cada actor.

**Tabla 3.9 Casos de Uso de Alto Nivel (UC1)**

<b>Caso de Uso</b>	UC1. Registro de un nuevo Usuario
<b>Actor</b>	Administrador
<b>Tipo</b>	Primario

<b>Descripción</b>	Al ingresar al sistema el administrador será capaz de registrar al nuevo usuario con datos personales (Administrador, Mecánico o Supervisor).
--------------------	---

**Tabla 3.10 Casos de Uso de Alto Nivel (UC2)**

<b>Caso de Uso</b>	<b>UC2. Registro de Mantenimiento correctivo</b>
<b>Actor</b>	Mecánico - Supervisor
<b>Tipo</b>	Secundario
<b>Descripción</b>	El ticket se emitirá a través del personal encargado de reparar la máquina que presente falla, donde se registrará la información necesaria para crear el registro. Una vez que el personal asignado que reparó la máquina genere un ticket mediante el sistema especificando la falla, el Supervisor podrá consultar el evento de mantenimiento correctivo. El sistema convertirá la información del ticket en archivo Excel.

Para fines de demostración, se incluyó un caso de uso de alto nivel para el registro de Equipment Model. En la *Tabla 3.9* se puede observar el usuario administrador que interactúa en el proceso. Los demás casos de uso de alto nivel se incluyen en el CD de anexos de la tesis.

### **3.2.5. Selección de Modelos del Estándar ISA-95 Implementados**

Para esta investigación se ha tomado el modelo de Operations Definition Model, el cual se implementó en el software *BOC-DWT* con las clases y atributos principales. Estos modelos fueron seleccionados después de revisar la parte 1 y 2 del estándar ISA-95.

Al intentar resolver el problema de la empresa BD Bard y teniendo la necesidad de registrar los eventos de mantenimiento correctivo y obtener la información para la toma de decisiones, se decidieron seleccionar algunas clases de los modelos ya mencionados anteriormente. Con estas clases y atributos se intentará resolver el problema del registro de eventos de mantenimiento correctivo en un sistema electrónico (*BOC-DWT*).

El modelo seleccionado para *BOC-DWT* es Operations Definition Model, el cual contiene la definición de las clases que permite crear una orden de mantenimiento para un equipo. En la *Figura 3.2* se puede observar el diagrama de clases que brinda la Parte 2 del estándar ISA-95 para este modelo (ANSI/ISA-95.00.01-2010, 2010).

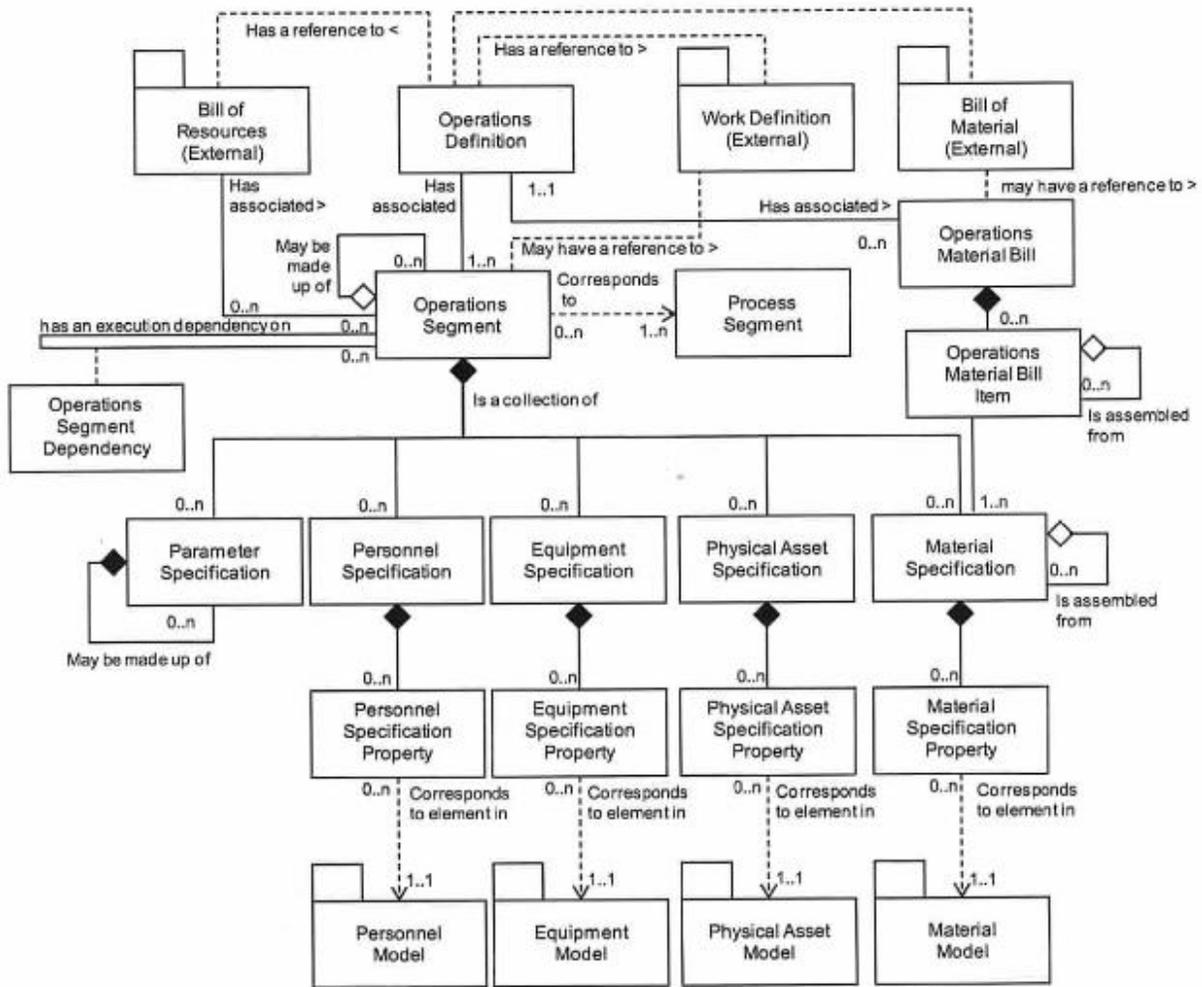


Figura 3.2 Operations Definition Model.

El modelo que presenta el estándar ISA-95 contiene las clases de Especificación del Equipo y Especificación de la propiedad del equipo y el submodelo del equipo. Para fines de este proyecto de tesis, solo se utilizaron algunas clases. En la *Figura 3.3* se observa el modelo de clases propuesto para implementarlo en el software *BOC-DWT*.

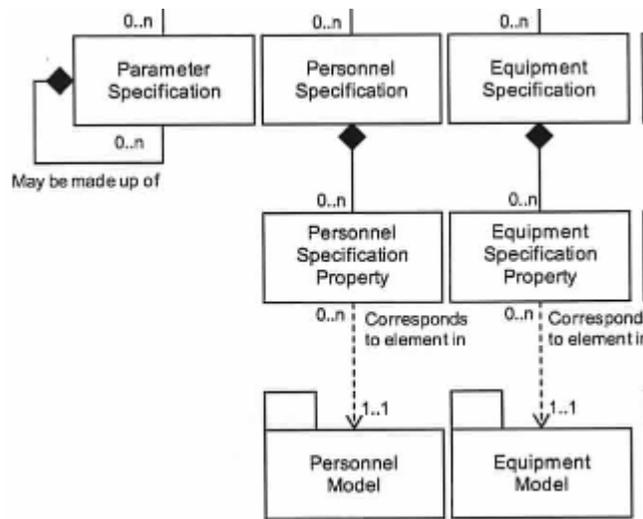


Figura 3.3 Definition Model Propuesto.

Solo se utilizaron las clases y modelos que se presentan en el diagrama ya que cumplen con los requerimientos básicos para resolver el problema que se presenta y que se pretende implementar en *BOC-DWT*, no se utilizaron todas las clases de este modelo ya que el objetivo de esta investigación no contempla esos modelos.

### 3.2.6. Análisis de Atributos de las Clases Seleccionadas

El software *BOC-DWT* que se realiza, solo utiliza algunos modelos de clases del estándar ISA-95 para resolver el problema presentado en este documento, por lo que el modelo de Operations Definition Model será el único modelo a utilizar, ya que cuenta con las clases necesarias para llevar a cabo el desarrollo de BOC-DWT. En esta subsección se presentan las clases y atributos que se seleccionaron para implementarlos en el sistema.

- **Operations Definition Model**

Las clases seleccionadas del modelo Operations Definition Model son las que se presentan en esta sección. En la *Tabla 3.11* (ANSI/ISA-95.00.01-2010, 2010), se

presentan los atributos de la clase Equipment Model que se implementó en el software.

Los atributos mostrados en la *Tabla 3.11* se implementaron con los mismos nombres en el software *BOC-DWT* que se desarrolló, con el fin de conservar la integridad con estándar.

Tabla 3.11 Work Definition Attributes.

Attribute Name	Description	Production Examples	Maintenance Examples	Quality Examples	Inventory Examples
ID	Uniquely identifies the operations definition.  The ID shall be used in other parts of the model when the <i>Operations definition</i> needs to be identified	Export Quality Widget	Medium Size AC Motor Overhaul	Potency Test Procedure	Tank Transfer Procedure
Version	An identification of the version of the <i>Operations definition</i> .  In cases where there are multiple versions of an <i>Operations definition</i> , then the version attribute shall contain the additional identification information to differentiate each version.	1.0	1.4	1.1	1.1
Description	Contains additional information and descriptions of the <i>Operations definition</i>	Information defining resources required for production of a single 'Export Quality Widget'.	For overhauls of motors less than 200 HP.	Test for potency of product	Movement of material from one tank to another
Operations type	Describes the category of operation  Required attribute  Defined values are: Production, Maintenance, Quality, Inventory, or Mixed.  "Mixed" shall be used when the operations definition contains several types of operations requests and/or segment requirements	Production	Maintenance	Quality	Inventory
Hierarchy Scope	Identifies where the exchanged information fits within the role based equipment hierarchy.	East Wing(AREA)/ Manufacturing Line #2(WORK CENTER)	CNC Machine Asset ID 13465	Test Cell 4 Receiving	Warehouse B
Bill of Material ID	Identification of the external Bill Of Material associated with this Operation Definition	BOM9929	BOM9928	BOM9927	BOM9926
Work Definition ID	Identification of the external Work Definition associated with this Operations Definition	WD009 V0.23	WD008 V03	WD007 V1.3	WD006
Bill of Resource ID	Identification of the external Bill Of Resource associated with this Operation Definition	BOR77782 V01	BOR77783	BOR77784 V11	BOR77785 V3.45

Con el resto de las clases seleccionadas del modelo "Operation Definition Model", se realizó el mismo procedimiento de selección, se utilizaron los mismos atributos que se muestran en la parte 2 del estándar ISA-95. La definición de los atributos que se utilizaron aparece en la sección 6.1.2 de la parte 2 del estándar en la tabla 43 (ANSI/ISA-95.00.01-2010, 2010).

### 3.2.7. Análisis y Diseño

En la fase de inicio, en la disciplina de análisis y diseño, se hizo el análisis del problema que se va a resolver por medio del software y en esta sección se describen y presentan los artefactos que se generaron para satisfacer los requerimientos iniciales.

De la misma manera, se diseñaron los casos de uso que describen el funcionamiento del sistema, estos casos de uso serán utilizados para desarrollar los diagramas de la siguiente iteración de la fase de elaboración.

### 3.2.8. Pruebas

Para realizar las pruebas de funcionamiento de *BOC-DWT*, se ha realizado el formato de un caso de prueba para ser utilizado en la fase de transición que permita probar el sistema y comprobar resultados (ver *Tabla 3.12*).

**Tabla 3.12 Formato de caso de prueba de funcionamiento de prototipo**

Nombre del Proyecto:		Caso No.	
No. Prueba:		Nombre de Caso:	
Fecha:		Escrito por:	
Descripción del Caso de Prueba:			
Resultados Esperados:			
Resultados Obtenidos:			

### 3.3. Fase de Elaboración

Una vez finalizada la fase de Inicio, se procede con las disciplinas incluidas en la iteración de Elaboración, que son análisis y diseño e implementación. Los artefactos construidos en estas disciplinas son los diagramas de clases (vista de

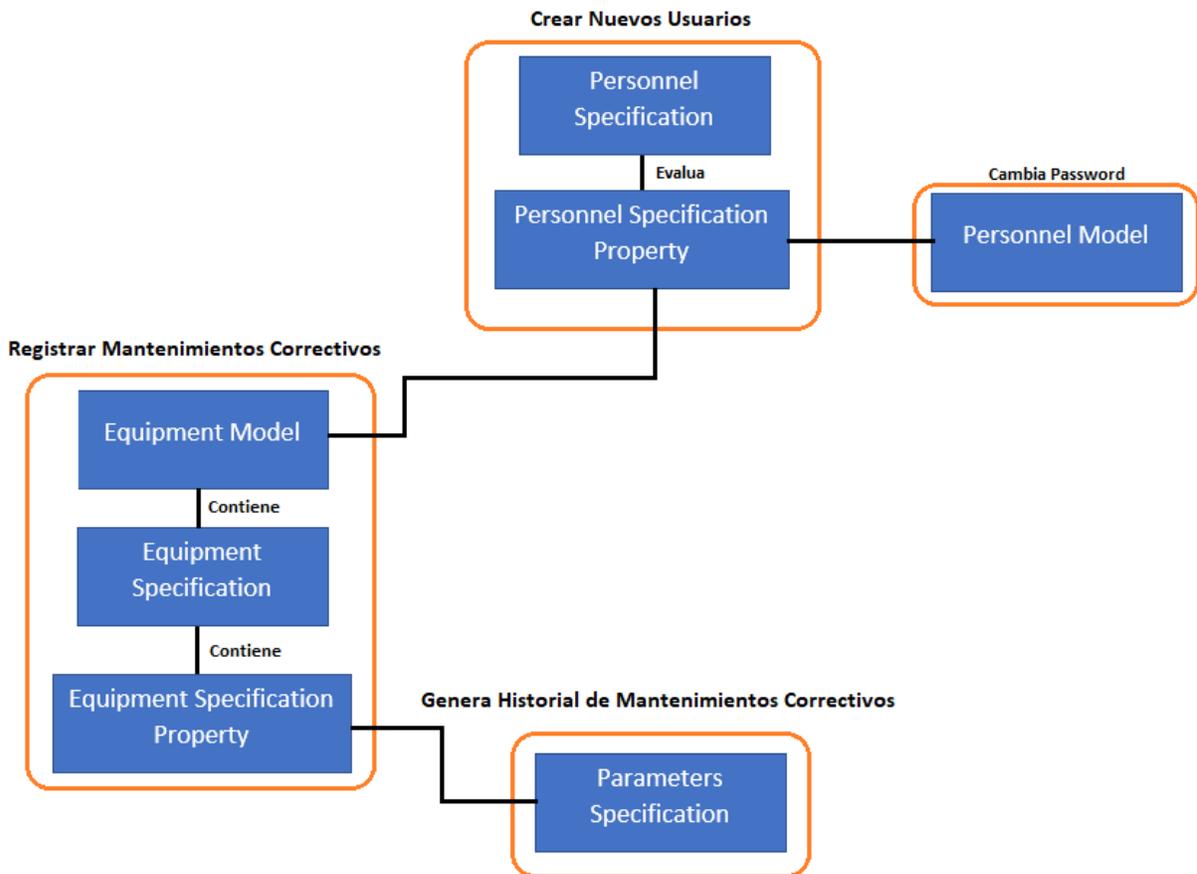
análisis) y diagramas de clases (vista de diseño), basados en los modelos de clases de estándar ISA-95, los diagramas de secuencia, el diagrama de base de datos y las tablas de correspondencia. El caso de prueba que se realizará para comprobar el funcionamiento de BOC-DWT será la creación de un registro de mantenimiento correctivo y consultar el registro en la base de datos para verificar que realmente cumple con los requerimientos funcionales.

### **3.3.1. Análisis y Diseño**

En la disciplina de análisis y diseño de la fase de elaboración se realizaron el diagrama de clases (vista de análisis), el diagrama de clases (vista de diseño), los diagramas de secuencia y la tabla de correspondencia de los modelos utilizados del estándar ISA-95. Estos artefactos brindan el soporte visual para explicar mejor el flujo que realiza el software al tratar de resolver el problema que se tiene. A continuación, describiremos los artefactos generados en esta iteración.

### **3.3.2. Diagrama de Clases (Vista de Análisis)**

Para proporcionar un marco de referencia, estructurado y claramente definido se ha realizado el diagrama de clases (vista de análisis) del sistema.



**Figura 3.4 Diagrama de clases (vista de análisis) de BOC-DWT.**

En la *Figura 3.4* se pueden observar los conceptos del modelo Operations Definition Model que pertenecen al estándar ISA-95 y han sido implementadas en el software BOC-DWT.

Se agruparon las clases para una mejor comprensión del diagrama y se señalaron la relación que contiene cada una de las clases.

### 3.3.3. Diagramas de Clases (Vista de Diseño)

Al analizar las clases seleccionadas para implementar en el software, se desarrolló el diagrama de clases (vista de diseño) con los atributos seleccionados en la sección.

Los métodos que aparecen en las clases fueron definidos de acuerdo a los requerimientos funcionales presentados en la sección 3.2.3 de este capítulo y las descripciones de las clases que aparecen en la sección 6.1.2 de la parte 2 del estándar. Las clases y multiplicidades se establecieron de acuerdo al diagrama de clases propuesto por ISA-95.

En la *Figura 3.5* se muestra el diagrama de clases realizado para la implementación del software *BOC-DWT*, de acuerdo con el modelo “Operations Definition Model” y las clases seleccionadas en la sección 3.2.5 de este capítulo (ver *Figura 3.5*).

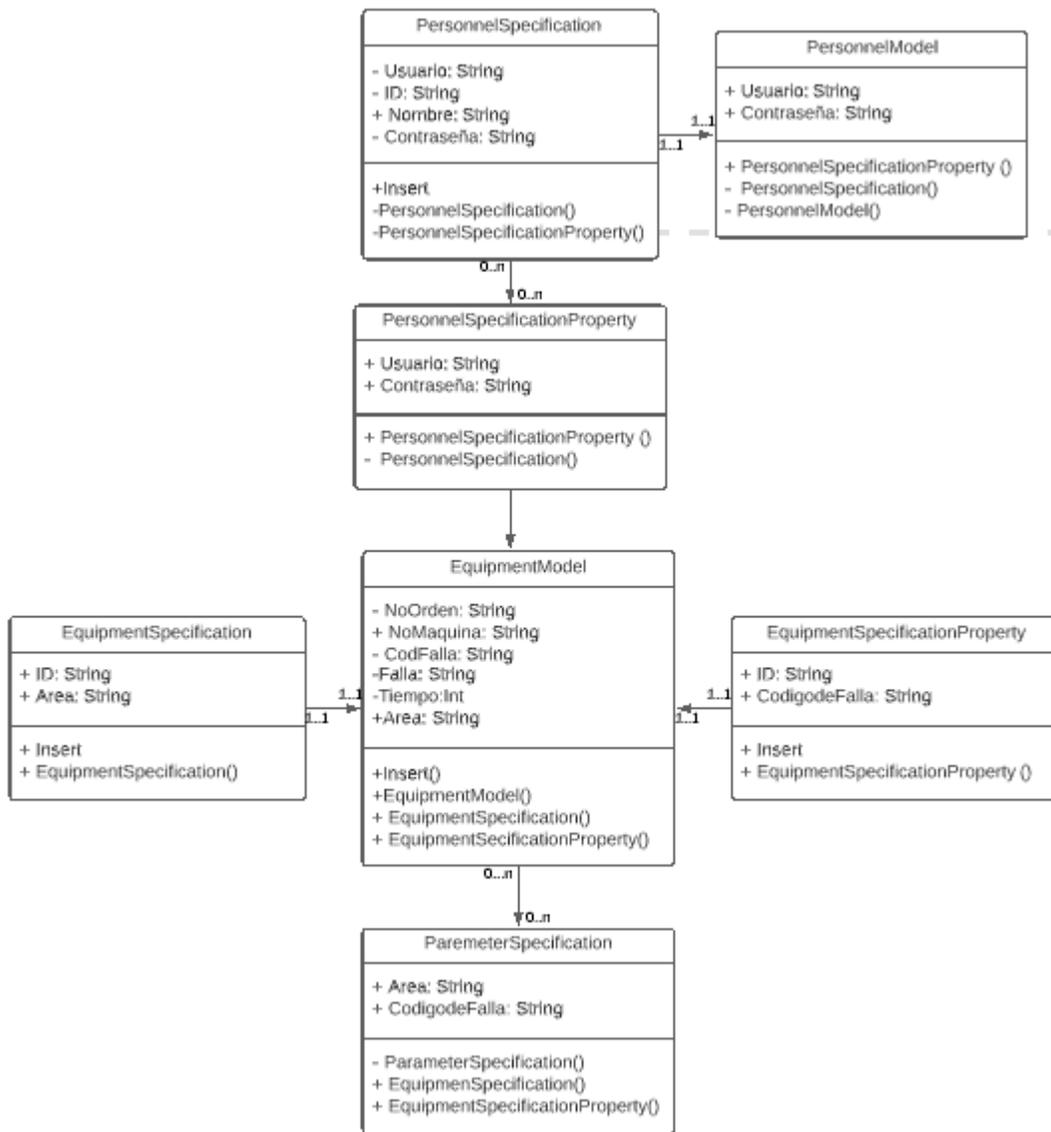


Figura 3.5 Diagrama de Clases (vista de diseño) de BOC-DWT.

### 3.3.4. Diagramas de Colaboración

Para modelar la interacción de los objetos en el sistema, se han creado los diagramas de secuencia de los módulos que contiene el software.

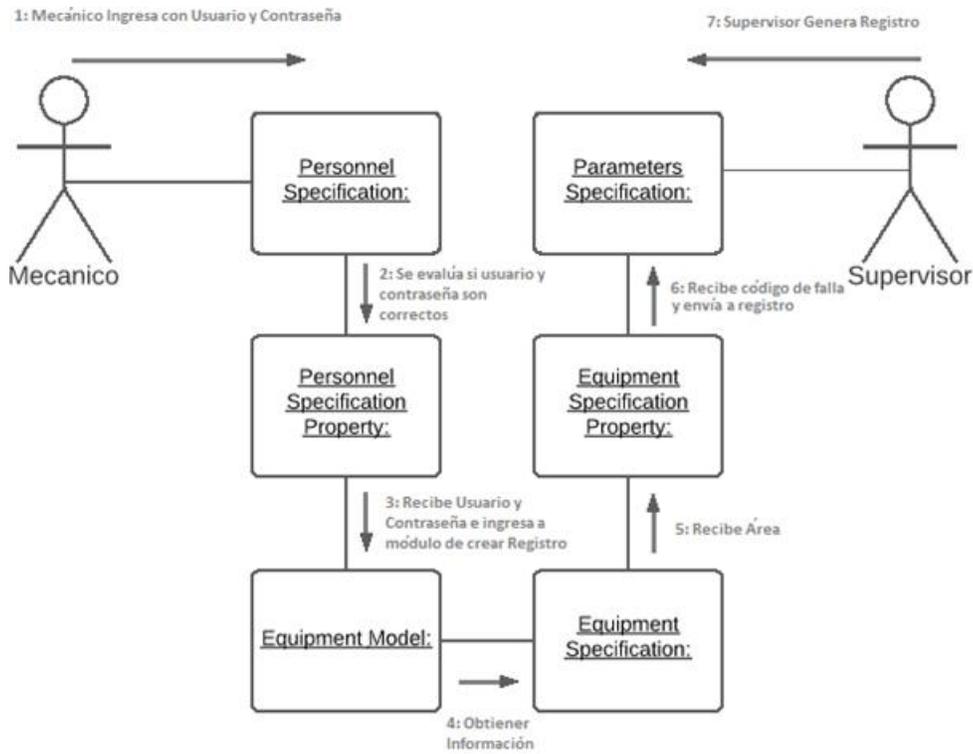


Figura 3.6 Diagrama de Colaboración.

En la *Figura 3.6* se puede observar las clases de la capa del dominio que intervienen en el proceso del registro de mantenimiento correctivo, así como la manera en que tienen interacción la interfaz de usuario con ellas. Este proceso consta de un proceso de registro de mantenimiento correctivo el cual es iniciado por un Mecánico al reparar una máquina ingresando al sistema con su respectivo usuario y contraseña, después este creará un registro de mantenimiento correctivo indicando la máquina, código de falla y área. Por último, solo guardará el registro y este será almacenado en la base de datos y podrá ser consultado por el Supervisor.

### 3.3.5. Tablas de Correspondencia

Para esta investigación se realizaron las tablas de correspondencia de los conceptos y modelos de ISA-95 implementados en el software BOC-DWT. La *Tabla 3.13* muestra cómo se corresponden el requerimiento “R2. El sistema permitirá a los

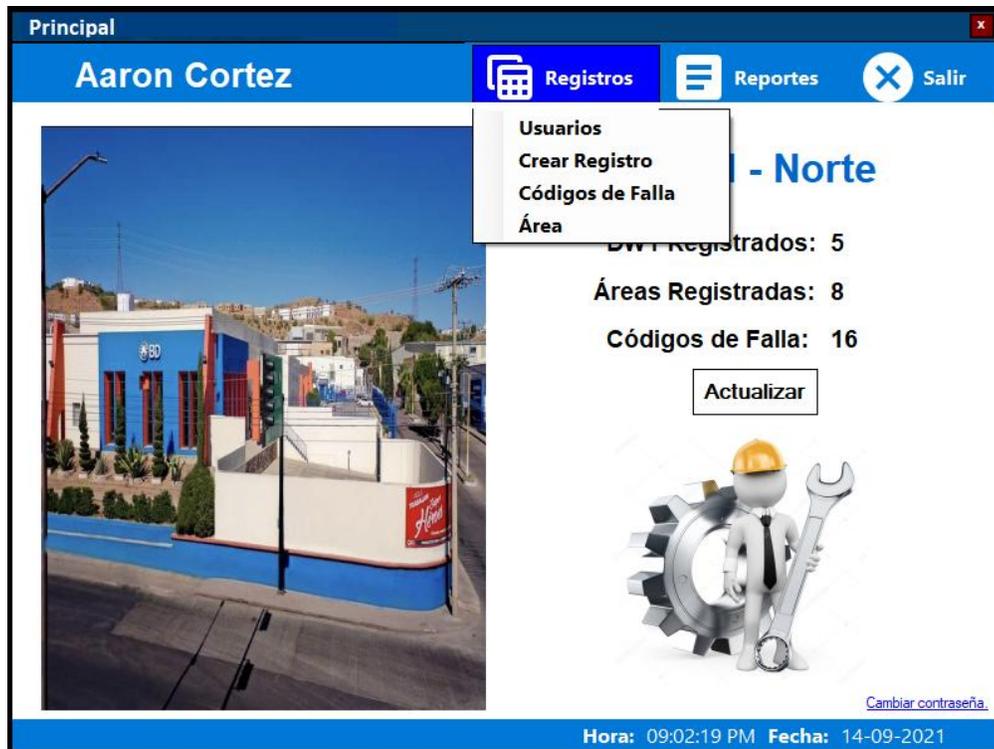
usuarios crear registros de mantenimiento correctivo.” con las diferentes secciones, tablas y figuras de ISA-95 que se utilizaron en los diferentes artefactos durante el desarrollo del software BOC-DWT, que logran cumplir el requerimiento R2.

**Tabla 3.13 Tabla de correspondencia del requerimiento R2.**

<b>Requerimiento “R2. El sistema permitirá a los usuarios crear registros de mantenimiento correctivo”.</b>		
<b>Artefacto de BOC-DWT</b>	<b>Parte 1, ISA-95</b>	<b>Parte 2, ISA-95</b>
<b>Diagrama de clases (Vista de análisis)</b>	3.1.1. Área, Página 12.	6.1.2. Operations Definition, Página 60
	5.2.4. Level 3 Activities, Página 20.	Tabla 43, Attributes of operations definition, Página 71.
	5.2.4.12. Maintenance operations management, Página 23.	6.1.7. Personnel Specification, Página 75.
	6.4.10 Maintenance Management, Página 37.	6.1.8. Personnel Specification Property, Página 73.
		6.1.9. Equipment Specification, Página 74.
		6.1.10. Equipment Specification. Property 6.1.10.
		Tabla 12, Attributes of Equipment, Página 36.
		Figura 5 Personnel Model, Página 27.

### 3.3.6. Interfaz de Usuario

La interfaz del usuario del software *BOC-DWT* está diseñado con la herramienta Visual Studio 2019. En la *Figura 3.7* se puede observar la interfaz de usuario que se utilizó para el diseño del software *BOC-DWT*.



**Figura 3.7 Interfaz de Usuario de Página Principal.**

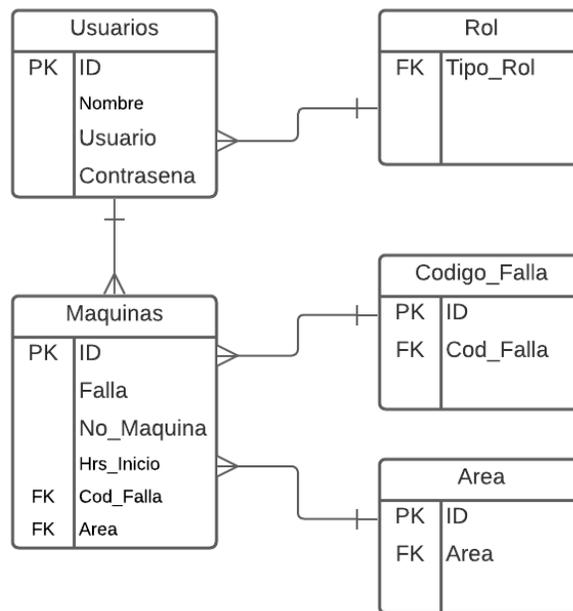
1. **Menú.** Se ha elegido un menú horizontal que contiene las secciones de los módulos realizados. Se agruparon por modelos para dar una mejor distribución de la página y que el usuario pueda ubicar correctamente el módulo que se necesita utilizar.
2. **Submenú.** Se han agregado las opciones relacionadas al menú en un segundo nivel al cual se denominó submenú. Estas opciones despliegan la información de cada módulo seleccionado.

### 3.3.7. Implementación

Para la disciplina de implementación en la fase de elaboración se ha desarrollado el diagrama de base de datos que se va a implementar en el software *BOC-DWT*.

### 3.3.8. Diagrama de Base de Datos

La base de datos se creó con el nombre de DBBard, está creada en el gestor de base de datos SQL Server 2019, que contiene las entidades y propiedades necesarias para utilizarlas en el software. El diagrama de base de datos que se generó se muestra la *Figura 3.8*.



**Figura 3.8 Diagrama de Base de datos BOC-DWT.**

CAPÍTULO 4: DESARROLLO DEL SOFTWARE BOC-DWT

En las siguientes secciones de este capítulo se describen los artefactos del código fuente, diseño de interfaz de usuario, consultas de base de datos e implementación de librerías del software BOC-DWT que se desarrollaron en las disciplinas de análisis y diseño e implementación de las fases de construcción y transición, tal como se puede observar en la *Tabla 4.1*.

Disciplinas	Inicio	Elaboración	Construcción	Transición
Requerimientos	Protocolo de investigación.			
	Definir requerimientos funcionales.			
	Definir requerimientos no funcionales.			
	Seleccionar modelos del estandar ISA-95 que se utilizarán.			
Análisis y diseño	Seleccionar las clases y atributos de los modelos del estandar ISA-95.			
	Definir casos de un alto nivel.			
	Establecer el caso del problema a resolver.			
	Realizar diagramas de clases.			
	Elaborar diagramas de secuencia.			
Implementación		Realizar diagrama de la base de datos.		
		Realizar bosquejo de las pantallas del software.		
		Definir la tabla de correspondencia de las clases de ISA-95 con las utilizadas en el Software.		
		Realizar diagrama de la base de datos.		
Pruebas	Definir formato de caso de prueba.			

Tabla 4.1 Fases de construcción del proyecto.

4.1. Construcción

Habiéndose analizado las secciones para elaborar los diagramas de interacción y los requerimientos del sistema, se inicia la fase de construcción. En esta fase solo se describirán los artefactos construidos en las disciplinas de Análisis y diseño, Implementación y Pruebas.

#### 4.1.1. Análisis y Diseño

En la disciplina de análisis y diseño para la fase de construcción se realizaron los artefactos de la generación de código fuente para la librería que contiene las clases, se desarrollaron los métodos que contiene cada clase y que se utilizan para ejecutar las acciones necesarias en cada módulo, también se desarrollaron las consultas y sintaxis necesarias para hacer actualizaciones en la base de datos, así como también la interfaz de usuario para cada módulo que contiene el software *BOC-DWT*.

A continuación, se describe a detalle el proceso que se realizó para construir los artefactos en la disciplina de análisis y diseño.

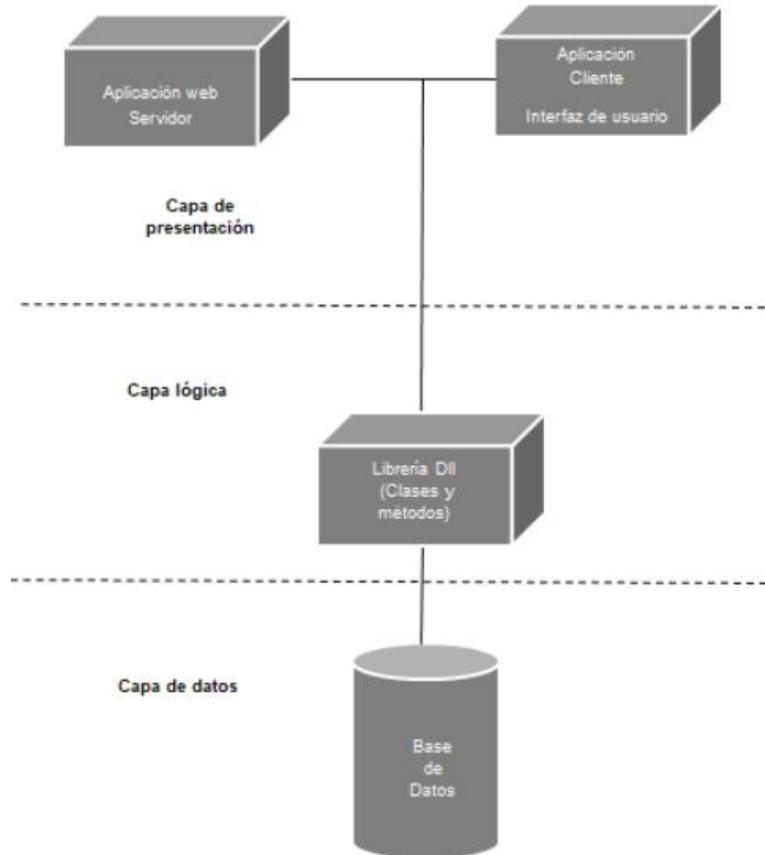
#### 4.1.2. Arquitectura del Software BOC-DWT

El software BOC-DWT fue desarrollado con una arquitectura de 3 capas, los componentes que presenta son la librería de clases (capa lógica), la aplicación (capa de presentación) y base de datos (capa de datos) (LEA, 2005).

En la *Figura 4.1* se observa la arquitectura de 3 capas con la que fue construido el prototipo y la interacción que tienen las capas. La librería generada con las clases y métodos propuestos por el estándar ISA-95 y que han sido adaptados para cumplir el objetivo de esta investigación hace las peticiones de información a la base de datos, esta a su vez responde con la información solicitada para transformarla de la manera en que el usuario la desee ver. Los objetos de las clases devuelven la información y se plasman en la interfaz del usuario.

La interfaz del usuario en la capa de presentación hace las peticiones a la librería para realizar alguna acción como insertar o consultar algún registro, la base de datos es la que se encarga de realizar las transacciones para consulta e inserción

de información, mientras tanto la librería de clases se encarga de realizar las peticiones a la base de datos por medio de consultas.

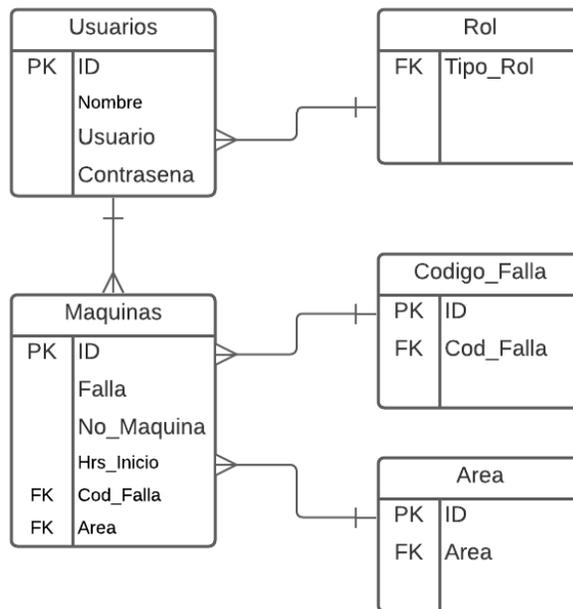


**Figura 4.1** Arquitectura de *BOC-DWT*

La capa de lógica o capa del dominio del software *BOC-DWT* está programado en la herramienta de desarrollo visual en su versión 2019 y para desarrollar la librería de clases y atributos se utilizó el lenguaje de programación C#.

Se decidió implementar las clases en una aplicación de escritorio, ya que se decidió utilizar las herramientas disponibles que actualmente se utilizan en la empresa BD Bard, de esta manera no fue necesario comprar una herramienta para el desarrollo de esta aplicación de escritorio y por lo tanto evitar comprar un software nuevo.

El gestor para la base de datos que se utilizó en la capa de datos es Microsoft SQL Server 2019, debido a que también es una herramienta con la que ya se cuenta en BD Bard y a su vez es una herramienta que soporta alta demanda de transacciones, es eficaz, confiable y ofrece un almacén de datos completo y confiable. La base de datos lleva por nombre DBBard.mdf, está formada por 5 tablas como lo muestra el diagrama de base de datos de la *Figura 4.2*.



**Figura 4.2 Diagrama de Base de datos BOC-DWT**

#### 4.1.3. Generación de Consultas de SQL Server

Las tablas y propiedades que se crearon para la base de datos DBBard contienen las llaves primarias y foráneas necesarias para relacionarlas entre sí. Además de la creación de tablas y sus llaves, se realizaron las consultas necesarias para implementarlos en los métodos de las clases del software BOC-DWT.

```

public DataTable D_buscarmaquinaXcodigodefalla(ClassEntidad obje)
{
    SqlCommand cmd = new SqlCommand("sp_buscar_maquina_fallacod", cn);
    cmd.CommandType = CommandType.StoredProcedure;
    cmd.Parameters.AddWithValue("@id_codfalla", obje.id_cod);
    SqlDataAdapter da = new SqlDataAdapter(cmd);
    DataTable dt = new DataTable();
    da.Fill(dt);
    return dt;
}

```

Figura 4.3 Consulta de Datos de Máquina por Código de Falla

```

public DataTable D_buscarmaquinaXarea(ClassEntidad obje)
{
    SqlCommand cmd = new SqlCommand("sp_buscar_maquina_area", cn);
    cmd.CommandType = CommandType.StoredProcedure;
    cmd.Parameters.AddWithValue("@id_area", obje.id_cod);
    SqlDataAdapter da = new SqlDataAdapter(cmd);
    DataTable dt = new DataTable();
    da.Fill(dt);
    return dt;
}

```

Figura 4.4 Consulta de Datos de Máquina por Área

Las 2 consultas de datos que se realizan en BOC-DWT se muestran en la *Figura 4.3*, donde se consulta la información de las máquinas por códigos de falla y en la *Figura 4.4*, donde se consulta la información de las máquinas por área asignada.

#### 4.1.4. Desarrollo de Clases del Estándar ISA-95

Para desarrollar las clases con sus atributos y métodos que propone el estándar, se hizo un análisis de la información que se necesita almacenar y la manera en la que están relacionados los atributos.

En la *Figura 4.5* se muestra el diagrama de clases que se generó para las clases que propone el estándar ISA-95, dentro del cual se encuentra la clase “Equipment Model”. Se han implementado sus atributos con los mismos nombres que define el estándar y el mismo nombre que se le asignó a la clase en software, que es “Equipment Model”.

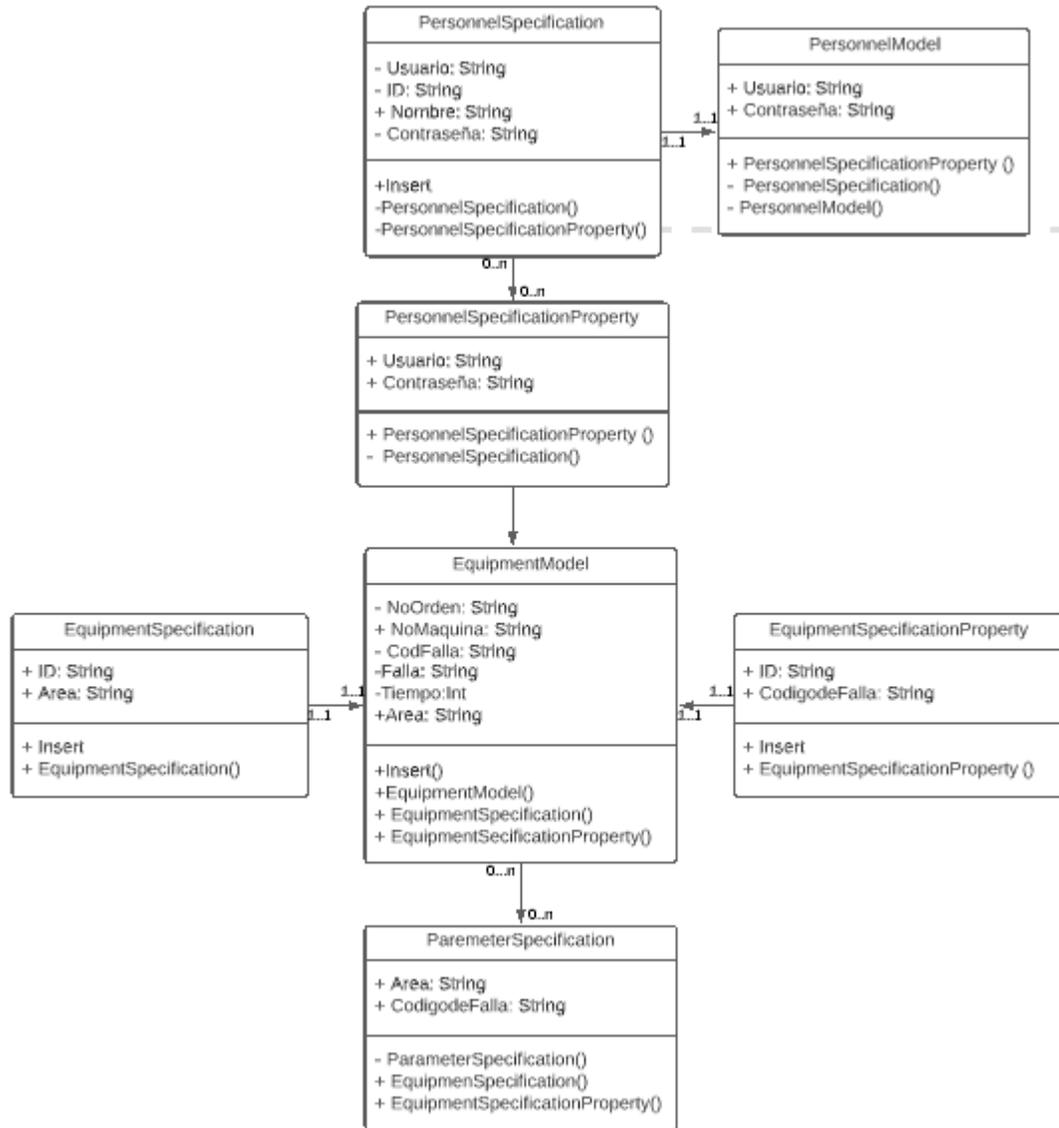


Figura 4.5 Diagrama de Clases (vista de diseño) de BOC-DWT

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Data;
using System.Data.SqlClient;
using System.Configuration;
using Capa_Entidad;

namespace Capa_Datos
{
    0 references
    public class Equipment_Model
    {
        public String D_mantenimientoalumno(ClassEntidad obje)
        {
            String accion = "";
            SqlCommand cmd = new SqlCommand("sp_mantenimiento_maquina", cn);
            cmd.CommandType = CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("@id_maquina", obje.id_maquina);
            cmd.Parameters.AddWithValue("@falla", obje.falla);
            cmd.Parameters.AddWithValue("@Nomaquina", obje.Nomaquina);
            cmd.Parameters.AddWithValue("@id_codfalla", obje.id_codfalla);
            cmd.Parameters.AddWithValue("@id_area", obje.id_area);
            cmd.Parameters.Add("@accion", SqlDbType.VarChar, 50).Value = obje.accion;
            cmd.Parameters["@accion"].Direction = ParameterDirection.InputOutput;
            if (cn.State == ConnectionState.Open) cn.Close();
            cn.Open();
            _ = cmd.ExecuteNonQuery();
            accion = cmd.Parameters["@accion"].Value.ToString();
            cn.Close();
            return accion;
        }
    }
}

```

Figura 4.6 Código Fuente de Equipment Model

En esta clase se insertan los datos para el registro de mantenimiento correctivo, aquí los datos necesarios para llevar a cabo el registro se insertan directamente en la base de datos DBBard.mdf, para inicializar la inserción de datos (ver la *Figura 4.6*), se utilizó el comando `SqlCommand`, el cual permite consultar y enviar comandos a una base de datos. Para llamar al procedimiento almacenado se

utiliza `cmd.CommandType = CommandType.StoredProcedure` y para agregar los datos, se utilizó el comando `cmd.parameters.AddWithValue`.

Para los demás métodos de búsqueda e inserción que se generaron en el diseño de clases del software *BOC-DWT*, se realizó el mismo proceso descrito.

La conexión entre el software desarrollado en Visual Studio con el lenguaje C# y SQL (Base de datos DBBard) se realizó de la siguiente manera (ver la *Figura 4.7*), la conexión que se utilizó fue usando el comando `connectionString`, el cual es una cadena de conexión que especifica información sobre la fuente de datos (DBbard.mdf) y los medios para conectarse a ella. Así se pasa en código a un controlador o proveedor subyacente para iniciar la conexión entre la base de datos SQL y Visual Studio.

```
?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <startup>
    <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.7.2" />
  </startup>
  <connectionStrings>
    <add name="sql" connectionString="server=DESKTOP-ILSNQP1;integrated security=true;database=DBBard"/>
  </connectionStrings>
</configuration>
```

**Figura 4.7 Conexión entre BOC-DWT y la Base de Datos**

#### 4.1.5. Implementación

En la fase de elaboración se diseñó la interfaz de usuario de la página principal de la aplicación de escritorio (*BOC-DWT*). A partir de ahí se diseñó la interfaz de usuario de cada uno de los módulos que contiene la aplicación, la intención es darle uniformidad a las pantallas para que el usuario se sienta más cómodo utilizando el mismo diseño en cada una de las opciones.

A continuación, se presenta la distribución de información en la interfaz de usuario del módulo “Equipment Model” y se describe a detalle sus elementos. Sirve como modelo para ejemplificar el resto de las interfaces de usuario desarrolladas para *BOC-DWT*.

No. Orden	Falla	No. Máquina	Tiempo (Min)	Código Falla	Área
A0005	Falla en P...	500	180	Falla Elec...	Foley
A0006	Falla en s...	100	25	Falta de ...	Extrusion
A0007	Falla en p...	1200	300	Falla Neu...	SBU2
A0008	Falla en b...	150	45	Falla Elec...	Silicone
A0009	Falla en n...	456	400	Problema ...	Silicone

Figura 4.8 Pantalla Equipment Model

En la *Figura 4.8* se muestran los controles que se utilizaron para el diseño de la interfaz, están enumerados para dar una mejor visión de la distribución. Las secciones que contiene son las siguientes:

1. **No. Orden.** Este campo se llena en automático por el mismo sistema asignando un número secuencial (Ejemplo A0005, A0006, A0007, Etc.).

2. **Falla.** Este campo se llena con la falla que se haya presentado en la máquina.
3. **No. Máquina.** Este campo se llena con el número de máquina que presente la falla.
4. **Tiempo (Min).** Este campo se llena con el tiempo en minutos del total del tiempo que le tomó al mecánico reparar la máquina.
5. **Cod. Falla.** En este campo se selecciona el código de falla de la lista desplegable.
6. **Área.** En este campo se selecciona el área al que pertenece la máquina que presente la falla de la lista desplegable.
7. **Buscar por Falla.** En este campo se debe ingresar alguna palabra clave y mostrará fallas relacionadas con la palabra ingresada.

## 4.2. Transición

En la fase de transición se analizaron las pruebas realizadas a los módulos del software para evaluar si cumplieron con los requerimientos funcionales y si es necesario realizar ajustes.

### 4.2.1. Pruebas

Para la disciplina de pruebas en la fase de transición se realizaron las pruebas unitarias de cada módulo utilizando el formato de caso de prueba descrito en la sección 3.2.8 del capítulo anterior.

De acuerdo a los requerimientos funcionales se hicieron las pruebas correspondientes, verificando que el software *BOC-DWT* que se construyó puede resolver el problema presentado en esta investigación. A continuación, se describe el procedimiento para realizar las pruebas en cada sección del prototipo.

#### 4.2.2. Caso de Prueba

Lo primero que se hizo fue capturar un código de falla como se puede apreciar en la *Tabla 4.2*, en donde se hizo una prueba para el registro de un código de falla y un área. Para la demostración del procedimiento se tomó como ejemplo la inserción del código de falla “Falla Eléctrica” y el área APON. La *Figura 4.9* y *4.10* muestran la captura de los datos mencionados en BOC-DWT y los resultados obtenidos se muestran en la *Tabla 4.2* los cuales dieron resultado satisfactorio.

**Tabla 4.2 Formato de Prueba de Captura de código de falla.**

Nombre del Proyecto:	BOC-DWT	Caso No.	1
No. Prueba:	001	Nombre de Caso:	Captura de código de falla y área.
Fecha:	Jun/02/2021	Escrito por:	Aaron Cortez
Descripción del Caso de Prueba:	Registrar código de falla y área para el registro del evento de mantenimiento correctivo.		
Resultados Esperados:	El código de falla y área es insertado correctamente.		
Resultados Obtenidos:	El código de falla y área fue insertado correctamente. La prueba es la número 001 y por los resultados obtenidos no es necesario realizar ningún tipo de ajuste ni repetir la prueba.		

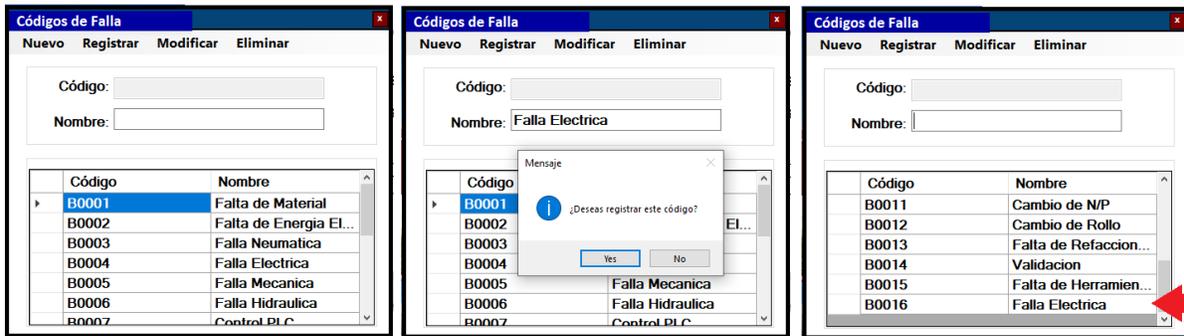


Figura 4.9 Registro de Código de Falla.

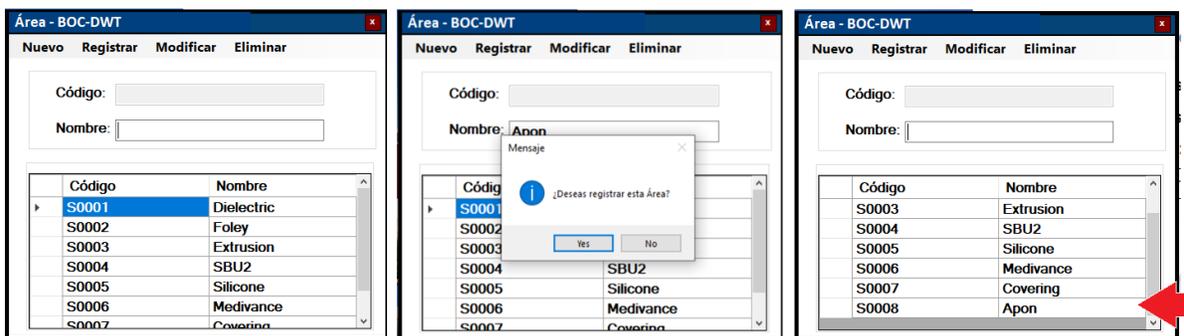


Figura 4.10 Registro de Área.

Una vez capturado el código de falla y área, se creó el registro de mantenimiento correctivo como en la *Figura 4.10*.

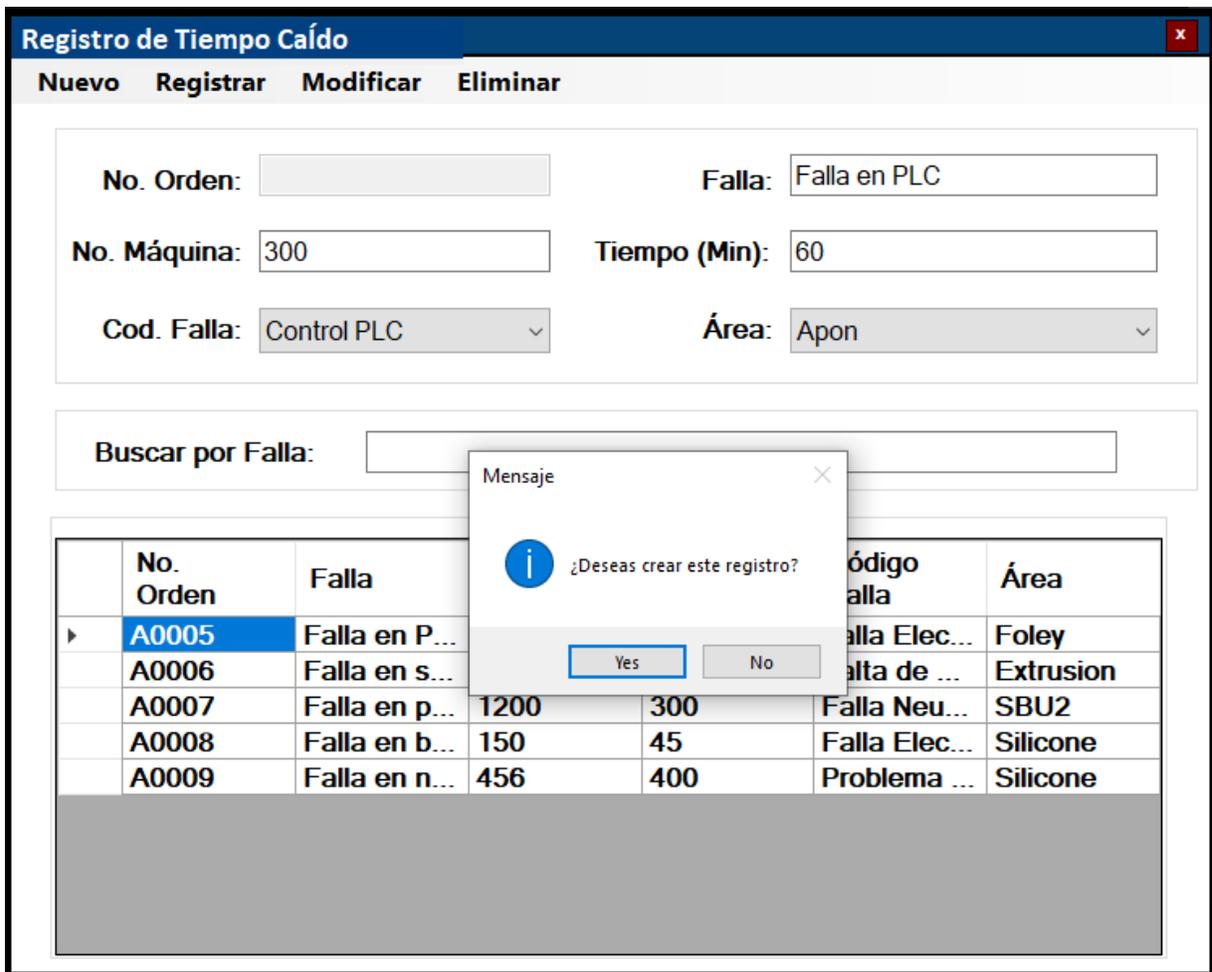


Figura 4.11 Capturando Registro de Mantenimiento Correctivo.

Una vez capturado el registro de mantenimiento correctivo con la información necesaria, el registro se considera completado por lo que los resultados se documentaron en la *Tabla 4.3* para comprobar que los resultados son los que se esperaban.

**Tabla 4.3 Formato de Prueba de Registro de Mantenimiento Correctivo.**

Nombre del Proyecto:	BOC-DWT	Caso No.	1
No. Prueba:	001	Nombre de Caso:	Registro de Mantenimiento Correctivo
Fecha:	Jun/02/2021	Escrito por:	Aaron Cortez
Descripción del Caso de Prueba:	Registrar mantenimiento correctivo con información requerida como falla, número de máquina, tiempo, código de falla y área.		
Resultados Esperados:	El registro es insertado correctamente.		
Resultados Obtenidos:	El registro para la máquina 300 fue insertado correctamente. La prueba es la número 001 y por los resultados obtenidos no es necesario realizar ningún tipo de ajuste ni repetir la prueba.		

La *Tabla 4.3* muestra el formato de prueba aplicado para la sección de Equipment Model en el caso de insertar el registro de mantenimiento correctivo para la máquina 300. La prueba es la número 001 y por los resultados obtenidos no es necesario realizar ningún ajuste ni repetir la prueba.

## CAPÍTULO 5: Resultados y Conclusiones

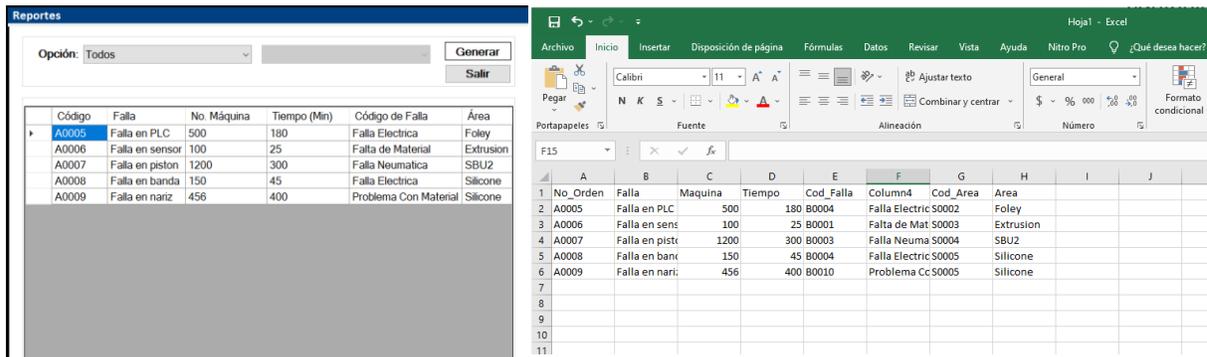
Una vez concluido el desarrollo del software *BOC-DWT* basado en los modelos de clases propuestos por el estándar ISA-95, se presenta el análisis de resultados de los objetivos y requerimientos planteados en un inicio y que se han implementado en esta investigación.

### 5.1. Resultados

El estudio del estándar ISA-95 parte 1, 2 y 3 permitió generar tablas de correspondencia (sección 3.3.5), en donde se muestran los requerimientos funcionales del software *BOC-DWT* y las secciones del estándar que se utilizaron para cumplir con dicho requerimiento, que dieron como resultado las clases, relaciones y métodos para el modelo utilizado en esta investigación, por lo que facilitó la implementación en la aplicación de escritorio *BOC-DWT*.

Fue posible resolver el problema con el prototipo creado, se realizó el proceso completo para registrar eventos de mantenimiento correctivo con la información contenida en el problema a resolver y como resultado, el software *BOC-DWT* generó la información del problema, el cual sirvió como apoyo para visualizar los datos relacionados, dando como resultado resolver el objetivo principal de registrar el mantenimiento correctivo.

En la *Figura 5.1* se puede observar la información contenida para el registro de mantenimiento correctivo. Esta información puede ser exportada al formato de Excel para ser analizada.



**Figura 5.1 Historial de Fallas y Exportación de Datos a Excel**

En la *Figura 5.1* se muestra la información capturada de los registros de mantenimiento correctivo. Con este proceso se cumplieron los objetivos específicos de la tesis:

- Se desarrolló una interfaz fácil de utilizar para los usuarios del sistema BOC-DWT, lo cual permitió hacer el proceso de registro de mantenimiento correctivo de manera más rápida.
- BOC-DWT permite que las actualizaciones de los códigos y número total de registros se vean reflejadas en todos los tipos de usuarios del sistema.
- BOC-DWT permite registrar las fallas, códigos de falla, el número de máquina y el tiempo, en una base de datos.

Al cumplirse los objetivos de esta investigación, BOC-DWT pudo registrar en una base de datos en SQL Server, los casos de mantenimiento correctivo que ocurrieron en BD Bard.

## 5.2. Conclusiones y Aprendizajes

Al realizar esta investigación se tuvieron que estudiar las partes 1, 2 y 3 del estándar ISA-95, los modelos de clases que propone y sus atributos. Con este aprendizaje se llegó a las siguientes conclusiones:

1. Es posible implementar los modelos de clases en un sistema de software de una industria maquiladora para resolver los procesos de mantenimiento correctivo, aunque este sistema solo incluye el registro de mantenimiento correctivo, se concluye que es posible implementar todos los modelos del estándar ISA-95 y crear un sistema más robusto.
2. Para registrar un evento de mantenimiento correctivo fue necesario utilizar las clases Personnel Specification, Equipment Specification, Equipment Specification Property, Personnel Specification Property, Personnel Model, Equipment Model y Parameter Specification del estándar ISA-95.
3. El costo de realizar un software para el registro de mantenimiento correctivo fue mucho menor que si se hubiera comprado uno comercial, ya que solo se requirió horas hombre para el desarrollo de la aplicación ya que en la empresa BD Bard contaba con el software Visual Studio 2019 y SQL Server 2019, así como también con el equipo de cómputo a ser utilizado por los diferentes usuarios dónde se instalará el software.

## 5.3. Trabajos Futuros

Al término de esta investigación se vio el potencial de la utilización del estándar ISA-95 para el desarrollo de software en las empresas a un costo mucho menor que comprar uno en el mercado.

También para las siguientes mejoras o nuevos requerimientos se recomienda revisar los otros modelos de clases de ISA-95, esto con el fin de que, al ser ISA-95 un estándar internacional, se siga manteniendo el reconocimiento de cumplimiento del software *BOC-DWT* Con el estándar. Algunas de estas mejoras serán las siguientes:

1. El sistema BOC-DWT será complementado para poder ser utilizado por el personal de producción de BD Bard.
2. El sistema BOC-DWT será complementado para incluir registros oficiales por máquina para auditorías internas y externas.
3. El sistema BOC-DWT será complementado para contar con un sistema de entrenamiento en mantenimiento preventivo para Mecánicos.
4. Una vez implementado el sistema BOC-DWT, será evaluado con el fin de conocer las nuevas necesidades que surjan y ver la viabilidad de incorporarlas al sistema.
5. Abordar otras partes del estándar ISA-95. En este trabajo solo se estudiaron las partes 1, 2 y 3, pero aún hay más partes, con esto se podrían desarrollar más y mejores sistemas de control basados en el mismo estándar.

## REFERENCIAS BIBLIOGRÁFICAS

- Alarcon. (2006). *Desarrollo de Sistemas de Información una Metodología Basada en el modelado*. Catalunya: UPC.
- ANSI/ISA 95-95.00.06. (2014). *Control System Integration Part 6: Messaging Service Model*. N/A: ISA.
- ANSI/ISA-95.00.01-2010. (2010). *Control System Integration Part1: Model Terminology*. N/A: ISA-Instrumentation.
- ANSI/ISA-95.00.01-2010. (2010). *Control System Integration Part2: Object Models and Attributes*. N/A: ISA-The Instrumentation.
- ANSI/ISA-95.00.01-2013. (2013). *Control System Integration Part3: Activity Models of Manufacturing Operaton Managment*. N/A: ISA-The Instrumentation.
- ANSI/ISA-95.00.04-2012. (2012). *Control System Integration Part 4: Objects and attributes for manufacturing operations management integration*. N/A: ISA.
- ANSI/ISA-95.00.05-2013. (2013). *Control System INtegration Part 5: Bussiness to manufacturing Transactions*. N/A: ISA.
- Bard. (1995, April 2). *Bard Nogales South*. Retrieved from [www.bardnogalessouth.com](http://www.bardnogalessouth.com)
- Cardenas, A. (1990). *Procedimiento para el registro de tiempo caído*. Nogales: Bard.
- Deuel, A. (1994, N/A N/A). *Isa Transactions*. Retrieved from <http://www.sciencedirect.com/science/article/pii/0019057894900426>

Doumeingts. (1995, N/A N/A). *Methodologies for designing CIM systems: A survey*.

Retrieved from

<<http://www.sciencedirect.com/science/article/pii/016636159400036P>>

Excellence, I. T. (2018, Marzo 28). *Gestión por procesos*. Retrieved from

<https://www.isotools.org/2018/03/28/por-que-automatizar-los-procesos-en-tu-organizacion/>

F., R. J. (2011, Mayo 11). *Key Performance Indicators*. Retrieved from

(<http://management.about.com/cs/generalmanagement/a/keyperfindic.htm>)

ISA-95. (2011, N/A N/A). *ISA*. Retrieved from

<<http://www.isa.org/MSTemplate.cfm?MicrositeID=285&CommitteeID=4747>>

ISA95. (2011, N/A N/A). *ISA95*. Retrieved from <<http://www.isa-95.com/index.php>>

Jeison. (2018, Junio 12). *Qualiex*. Retrieved from

<https://blogdelacalidad.com/diagrama-de-ishikawa/>

Lairman. (1999). *UML y Patrones Introducción al análisis y diseño orientado a*

*objetos*. Mexico: Pretince Hall.

Langhan, M. (2016, June 5). *Faciliworks*. Retrieved from Cybermetrics:

<https://faciliworks.com/support/>

Larman. (1999). *introducción al análisis y diseño orientado a*. N/A: Pearson.

LEA, B.-R. (2005, N/A N/A). *A prototype multi-agent ERP system: an integrated architecture and a conceptual framework*. Retrieved from

<<http://www.sciencedirect.com/science/article/pii/S0166497203001536>>

Maint, E. (2000, N/A N/A). *Easy Maint*. Retrieved from <http://easymaint.net/>

- Masood. (2004, Junio 29). *Productivity Improvement Through Computer Integrated Manufacturing in Post*. Retrieved from  
<<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.127.9275&rep=rep1&type=pdf> >
- Maximo. (1980, N/A N/A). *IBM*. Retrieved from <https://www-01.ibm.com/software/mx/tivoli/products/maximo-asset-mgmt/>
- Mendoza, T. (2020, Diciembre 16). Mantenimiento Correctivo. (A. Cortez, Interviewer)
- Metodoss. (2018). *Metodoss*. Retrieved from [www.metodoss.com/metodologia-rup/#Fase\\_de\\_diseno](http://www.metodoss.com/metodologia-rup/#Fase_de_diseno)
- Nagalingam. (1999, N/A N/A). *Latest developments*. Retrieved from  
<<http://www.sciencedirect.com/science/article/pii/S0736584599000368>>
- Pardo, G. &. (1988). *Diagrama de Clases*. N/A: N/A.
- Pineda, R. (2010). *Registro y Manejo de Tiempo Caído*. Nogales: Bard.
- Pineda, R. (2017, Abril 7). Problemas DWT. (A. Cortez, Interviewer)
- Rabago, J. (2018, Junio 10). Mantenimiento Correctivo. (A. Cortez, Interviewer)
- Ruiz, E. (2018, Junio 20). Software Tiempo Caído. (A. Cortez, Interviewer)
- Ruiz, G. (2015). *Compliance and Software Requirements*. Nogales: Bard.
- Salazar, V. (2009, N/A N/A). *Análisis de la integración de los sistemas MES-ERP en industrias de manufactura*. Retrieved from  
<<http://www.laccei.org/LACCEI2009-Venezuela/p207.pdf>>
- SAP. (2001, N/A N/A). *SAP Latinoamérica*. Retrieved from  
<https://www.sap.com/latinamerica/index.html>

Slavendy. (2001). *Handbook of industrial engineering: technology and operations management*. US: Institute of industrial Engineers.

Torossi, G. (2011). *El Proceso Unificado de Desarrollo de Software*.

Velderrain, H. (2015, September 8). *Informática: Línea del Tiempo Computadora*. Retrieved from Prezi: <https://prezi.com/7f57vqbx4qka/informatica-linea-del-tiempo-computadora/>

