



UNIVERSIDAD DE ORIENTE
NÚCLEO DE SUCRE
ESCUELA DE CIENCIAS
DEPARTAMENTO DE MATEMÁTICAS
PROGRAMA DE LA LICENCIATURA EN INFORMÁTICA

APLICACIÓN WEB PARA LA GESTIÓN DEL SISTEMA DE
INTERRUPCIONES DE LA SIDERÚRGICA DEL ORINOCO ALFREDO
MANEIRO (SIDOR) (Modalidad: Pasantía)

JOSÉ LUIS CUMANA DOMINGUEZ

TRABAJO DE GRADO PRESENTADO COMO REQUISITO PARCIAL PARA
OPTAR AL TÍTULO DE LICENCIADO EN INFORMÁTICA

CUMANÁ, 2011

APLICACIÓN WEB PARA LA GESTIÓN DEL SISTEMA DE
INTERRUPCIONES DE LA SIDERÚRGICA DEL ORINOCO ALFREDO
MANEIRO (SIDOR)

APROBADO POR:

Prof. Carmen V. Romero
Asesor Académico

Prof. Dianelina Aguiar
Coasesor Académico

Ing. Carlos Agostini
Tutor industrial

(Jurado)

(Jurado)

ÍNDICE

	Pág.
DEDICATORIA.....	i
AGRADECIMIENTOS.....	ii
LISTA DE TABLAS.....	iii
LISTA DE FIGURAS.....	iv
LISTA DE ABREVIATURAS.....	vi
RESUMEN.....	vii
INTRODUCCIÓN.....	1
CAPÍTULO I.....	6
1.1 Planteamiento del problema	6
1.2 Alcance	8
1.3 Limitaciones	9
CAPÍTULO II.....	10
2.1 Marco teórico	10
2.1.1 Antecedentes de la investigación.....	10
2.1.2 Antecedentes de la organización.....	12
2.1.3 Área de estudio.....	14
2.1.4 Área de investigación.....	16
2.2 Marco metodológico	23
2.2.1 Metodología de la investigación.....	23
2.2.2 Metodología del área aplicada.....	24
CAPÍTULO III.....	29
3.1 FASE DE INICIO.....	29
3.1.1 Planificación de la fase de inicio	29
3.1.2 Documento visión	30
3.1.3 Estudio del contexto del sistema.	31
3.1.3.1 Modelado del negocio.....	32
3.1.3.2 Modelo de fines del negocio.....	32
3.1.3.3 Modelo de Michael Porter aplicado al negocio.....	33

3.1.3.4	Modelo de jerarquía de procesos aplicado a los procesos de más alto nivel del negocio.....	34
3.1.3.5	Notación de Eriksson y Penker aplicado a los procesos de alto nivel del negocio.....	37
3.1.3.6	Modelo del dominio	38
3.1.4	Riesgos del sistema	42
3.1.5	Flujo de trabajo (<i>workflow</i>) de requisitos	45
3.1.5.1	Requerimientos funcionales	45
3.1.5.2	Requerimientos no funcionales	46
3.1.5.3	Captura de requisitos como casos de uso	48
3.1.5.4	Modelo de casos de uso	50
3.1.6	Flujo de trabajo (<i>workflow</i>) de análisis	55
3.1.6.1	Modelo de análisis	56
3.1.6.2	Diagrama de clases de análisis.....	60
3.1.6.3	Diagrama de colaboración	63
3.1.6.4	Identificación de paquetes de análisis	75
3.1.7	Evaluación de la fase de inicio	76
3.2	FASE DE ELABORACIÓN	77
3.2.1	Planificación de la fase de elaboración.....	77
3.2.2	Flujo de trabajo (<i>workflow</i>) de requisitos	79
3.2.2.1	Requerimientos funcionales	79
3.2.2.2	Requerimientos no funcionales	80
3.2.2.3	Captura de requisitos como casos de uso	80
3.2.2.4	Modelo de casos de uso	80
3.2.2.5	Perfiles de usuarios	84
3.2.2.6	Prototipo de la interfaz de usuario	87
3.2.3	Flujo de trabajo (<i>workflow</i>) de análisis	89
3.2.3.1	Modelo de análisis	89
3.2.3.2	Diagrama de clases de análisis.....	91
3.2.3.3	Diagrama de colaboración	93
3.2.3.4	Identificación de paquetes de análisis	99
3.2.3.5	Diagrama de paquetes de análisis	100

3.2.4 Flujo de trabajo (<i>workflow</i>) de diseño	101
3.2.4.1 Diseño de la arquitectura.....	102
3.2.4.2 Modelo de diseño	103
3.2.4.3 Diagramas de clases del diseño.....	107
3.2.4.4 Diagrama de secuencia.....	111
3.2.4.5 Diseño de la base de datos local del sistema.....	111
3.2.5 Flujo de trabajo (<i>workflow</i>) de implementación	111
3.2.5.1 Códigos ejecutables para realizar los casos de uso	111
3.2.6 Flujo de trabajo (<i>workflow</i>) de pruebas	112
3.2.6.1 Partición equivalente.....	112
3.2.6.2 Aplicación de casos de prueba	113
3.2.7 Evaluación de la fase de elaboración	115
3.3 FASE DE CONSTRUCCIÓN	116
3.3.1 Planificación de la fase de construcción	117
3.3.2 Flujo de trabajo (<i>workflow</i>) de implementación	117
3.3.2.1 Códigos ejecutables para realizar los casos de uso	117
3.3.2.2 Documentación del sistema.....	120
3.3.3 Flujo de trabajo (<i>workflow</i>) de pruebas	120
3.3.3.1 Partición equivalente.....	120
3.3.3.2 Aplicación de casos de prueba	121
3.3.3.3 Casos de prueba basados en casos de uso	123
3.3.3.4 Pruebas de integración	124
3.3.4 Evaluación de la fase de construcción	124
CONCLUSIONES.....	125
RECOMENDACIONES	127
BIBLIOGRAFÍA	128
APÉNDICES	130
ANEXOS	174
Hoja de Metadatos	178

DEDICATORIA

Este trabajo está dedicado a:

Mi padre José Luis Cumana, que Dios lo tenga en su gloria y a mi madre Africa Domínguez de Cumana, a quienes debo todo, ellos con su amor, comprensión y paciencia me enseñaron las cosas buenas de la vida, y que con su esfuerzo y lucha constante han hecho posible el logro de mis metas, forjándome como un hombre de bien.

Mis hermanos, Anyela y Raúl quienes me han brindado su ayuda en todo momento durante mis estudios y siempre como hermanos unidos.

Mi abuela, Valeria Domínguez que desde muy niño me ha guiado por el buen camino, inculcándome valores como la honestidad, moral y buenas costumbres durante muchas etapas de mi vida, gracias por todos tus consejos.

Mi tío Luis Beltrán Cumana, que ha sido como un segundo padre y modelo a seguir.

Mis tías Olga y Marik quienes desde siempre han estado a mi lado apoyándome, luchado como mujeres emprendedoras para salir adelante frente a los problemas.

José Luis

AGRADECIMIENTOS

A:

La Universidad de Oriente, Núcleo de Sucre, casa de estudios que me permitió realizarme como profesional.

La Siderúrgica del Orinoco Alfredo Maneiro, por brindarme la oportunidad de realizar mi trabajo de grado en sus instalaciones.

El ingeniero Carlos Agostini (Coordinado de Nivel 2B en el Departamento de Automatización y Control) por haberme brindado sus conocimientos como profesional.

La profesora Carmen Victoria Romero y Dianelina Aguiar, por sus asesorías a nivel académico.

Todas las personas que laboran en el Departamento de Automatización y Control, especialmente a Claudia Rivas (Líder del Proyecto Interrupciones) quien con su colaboración, paciencia y tremenda calidad humana, hicieron posible la realización de este trabajo.

Mi tía Maritza Gonzales y primos quienes con cariño y amor me acogieron en su hogar el tiempo necesario para la realización de este trabajo.

La familia Jiménez Patiño que con su constante apoyo me ayudó en las buenas y en las malas, llenándome de ánimos para enfrentar las dificultades.

Mis amigas, Marci Smith, Karlys Noriega y Maribel Machado, que durante toda la carrera de estudios siempre estuvieron apoyándome, para lograr mis metas y sueños.

LISTA DE TABLAS

	Pág.
Tabla 1. Actividades y artefactos planificados para la fase de inicio.	29
Tabla 2. Roles necesarios para el desarrollo del proyecto.	31
Tabla 3. Glosario de términos del modelo del dominio del sistema propuesto.	41
Tabla 4. Plan de prevención y contingencia para los riesgos más predominantes en el desarrollo de la aplicación <i>Web</i>	44
Tabla 5. Requisitos funcionales en la fase de inicio.	45
Tabla 6. Requerimientos no funcionales en la fase de inicio.	46
Tabla 7. Casos de uso de la Aplicación <i>Web</i> para la Gestión del SI de SIDOR en la fase de inicio.	49
Tabla 8. Caso de Uso Clasificar Demoras.	51
Tabla 9. Caso de Uso Gestionar Productividad.	52
Tabla 10. Gestionar Catálogos.	53
Tabla 11. Caso de Uso Gestionar Usuarios.	55
Tabla 12. Clases de interfaz del sistema en la fase de inicio.	57
Tabla 13. Clases de control del sistema en la fase de inicio.	58
Tabla 14. Clases de entidad del sistema en la fase de inicio.	59
Tabla 15. Estatus de los artefactos planificados en la fase de inicio.	76
Tabla 16. Actividades y artefactos planificados para la fase de elaboración.	78
Tabla 17. Requerimientos funcionales en fase de elaboración.	79
Tabla 18. Casos de uso de la Aplicación <i>Web</i> para la Gestión del SI de SIDOR en la fase de elaboración.	80
Tabla 19. Caso de Uso Gestionar Patrones.	81
Tabla 20. Caso de Uso Generar Reportes.	83
Tabla 21. Clases de interfaz del sistema de la fase de elaboración.	89
Tabla 22. Clases de control del sistema de la fase de elaboración.	90
Tabla 23. Clases de entidad del sistema de la fase de elaboración.	90
Tabla 24. Aplicación de casos de prueba de la fase de elaboración.	113
Tabla 25. Estatus de los artefactos planificados en la fase de elaboración.	116
Tabla 26. Actividades y artefactos planificados para la fase de construcción.	117
Tabla 27. Aplicación de casos de prueba de la fase de construcción.	121

LISTA DE FIGURAS

	Pág.
Figura 1. Proceso Entrada Salida	24
Figura 2. Ciclos del Proceso Unificado de Desarrollo de Software.....	26
Figura 3. Fases del Proceso Unificado de Desarrollo de Software.....	26
Figura 4. Fases	27
Figura 5. Diagrama de Fines del Negocio.....	33
Figura 6. Cadena de valores de los procesos llevados a cabo en el Departamento de Automatización y Control.....	34
Figura 7. Diagrama de jerarquía de los procesos llevados a cabo en el en el Departamento de Automatización y Control.....	36
Figura 8. Diagrama de proceso de negocio Gestionar SI.....	37
Figura 9. Modelo del negocio del proceso Gestionar SI.....	38
Figura 10. Modelo de Dominio para el Sistema de Interrupciones de SIDOR.....	40
Figura 11. Diagrama de caso de uso de la Aplicación <i>Web</i> para la Gestión del SI de SIDOR de la fase de inicio.....	51
Figura 12. Diagrama de clases de análisis para el caso de uso Clasificar Demora de la fase de inicio.	61
Figura 13. Diagrama de clases de análisis para el caso de uso Gestionar Productividad de la fase de inicio.....	62
Figura 14. Diagrama de clases de análisis para el caso de uso Gestionar Catálogo de la fase de inicio.....	62
Figura 15. Diagrama de clases de análisis para el caso de uso Gestionar Usuarios de la fase de inicio.....	63
Figura 16. Diagrama de colaboración del caso de uso Clasificar Demora de la fase de inicio.....	64
Figura 17. Diagrama de colaboración del caso de uso Gestionar Productividad Programada (agregar) de la fase de inicio.....	65
Figura 18. Diagrama de colaboración del caso de uso Gestionar Productividad Programada (modificar) de la fase de inicio.	67
Figura 19. Diagrama de colaboración del caso de uso Gestionar Productividad Programada (eliminar) de la fase de inicio.	68
Figura 20. Diagrama de colaboración del caso de uso Gestionar Catálogo (agregar) de la fase de inicio.....	69
Figura 21. Diagrama de colaboración del caso de uso Gestionar Usuarios (agregar) de la fase de inicio.....	71
Figura 22. Diagrama de colaboración del caso de uso Gestionar Usuarios (modificar) de la fase de inicio.....	73
Figura 23. Diagrama de colaboración del caso de uso Gestionar Usuarios (eliminar) de la fase de inicio.....	75
Figura 24. Paquete de análisis de Catálogos.....	75

	Pág
Figura 25. Paquete de análisis de Productividad.....	75
Figura 26. Paquete de análisis de Demoras.....	76
Figura 27. Paquete de análisis de Usuarios.....	76
Figura 28. Diagrama de caso de uso de la Aplicación <i>Web</i> para la Gestión del SI de SIDOR de la fase de elaboración.....	81
Figura 29. Prototipo de interfaz principal de la aplicación.....	87
Figura 30. Prototipo de interfaz secundaria de la aplicación.....	88
Figura 31. Diagrama de clases de análisis para el caso de uso Gestionar Patrones de la fase de elaboración.....	91
Figura 32. Diagrama de clases de análisis para el caso de uso Generar Reportes de la fase de elaboración.....	92
Figura 33. Diagrama de clases de análisis para el caso de uso Gestionar Usuarios de la fase de elaboración.....	92
Figura 34. Diagrama de colaboración del caso de uso Gestionar Patrones (agregar) de la fase de elaboración.....	95
Figura 35. Diagrama de colaboración del caso de uso Gestionar Patrones (modificar) de la fase de elaboración.....	97
Figura 36. Diagrama de colaboración del caso de uso Gestionar Patrones (eliminar) de la fase de elaboración.....	98
Figura 37. Diagrama de colaboración del caso de uso Generar Reportes de la fase de elaboración.....	99
Figura 38. Paquete de análisis de Patrones.....	100
Figura 39. Paquete de análisis Generar Reportes.....	100
Figura 40. Diagrama de paquetes de análisis del sistema de la fase de elaboración.....	101
Figura 41. Capas de la arquitectura del sistema.....	103
Figura 42. Clases de diseño a partir de clases de interfaz.....	104
Figura 43. Clases de diseño a partir de clases de control.....	105
Figura 44. Clases de diseño a partir de clases de entidad.....	105
Figura 45. Clases detectadas.....	108
Figura 46. Diagrama de clases.....	110
Figura 47. Formulario Dinámico de carga de Catálogos.....	112
Figura 48. Formulario de carga de datos de demoras.....	118
Figura 49. Formulario de carga de datos de productividad programada.....	118
Figura 50. Formulario de carga de datos de Patrones.....	119

LISTA DE ABREVIATURAS

SIDOR: Siderúrgica del Orinoco Alfredo Maneiro

SI: Sistema de Interrupciones de SIDOR

UDO: Universidad de Oriente

UML: *Unified Model Language* (Modelado de Lenguaje Unificado)

HTML: *HyperText Markup Language* (Lenguaje de Marcas de Hipertexto)

HTTP: *Hypertext Transfer Protocol* (Protocolo de Transferencia de Hipertexto)

HTTPS: *Hypertext Transfer Protocol Security* (Versión Segura del Protocolo de Transferencia de Hipertexto)

TCP: *Transmission Control Protocol* (Protocolo de Control de Transmisión)

TCP/IP: *Transmission Control Protocol/Internet Protocol* (Protocolo de Control de Transmisión/Protocolo de Internet)

WWW: *World Wide Web*

IIS: *Internet Information Server* (Servidor *Web* de *Microsoft*)

SQL: *Structured Query Language* (Lenguaje Estructurado de Consultas)

SGBD: Sistema de Gestión de Base de Datos

ASP: *Active Server Pages*

IDE: Entorno de desarrollo integrado

RESUMEN

Se desarrolló una aplicación *Web* para la gestión del Sistema de Interrupciones (SI) de la Siderúrgica del Orinoco Alfredo Maneiro (SIDOR), que se enmarca dentro de la intención de la empresa de centralizar la información, debido a la necesidad de saber con precisión el estado de una línea de producción, además de las pérdidas sufridas en los respectivos turnos laborales en base a las metas propuestas por la Dirección de cada planta, permitiendo que el personal interesado pueda corroborar que la información suministrada de la línea de producción sea clara, exacta y veraz. Ante esto, la aplicación *Web* permite resolver los problemas relacionados con el registro, recuperación y organización de la información referente a los procesos de gestión de catálogos, metas de producción mensual de las plantas, clasificación de demoras, gestión de patrones de rotación de cuadrillas por turnos y de usuarios, brindando un medio para controlar funcionamiento del SI; lo que produce una toma de decisiones más acertada. La aplicación abarca todas las plantas que se encuentran funcionando bajo el SI, permitiendo el manejo de toda su información, apoyándose en varias bases de datos según sea la planta. Este desarrollo estuvo dirigido por el Proceso Unificado de Desarrollo de Software planteado por Jacobson, Booch y Rumbaugh (1999), y UML como lenguaje de modelado para la notación de los diagramas que permiten el entendimiento más detallado del sistema. En cuanto a la codificación y construcción del sistema, se utilizó el *Framework Visual Studio.NET 2003* específicamente el lenguaje de programación *Visual Basic.NET* para la creación de las páginas *Web* con tecnología *ASP.NET*, *SQL Server 2000* como manejador de base de datos, *Javascript* como lenguaje de programación interpretado y basado en objetos para la validación de formularios, *Internet Information Server (IIS)*, como servidor *Web* y *Windows Server 2003* como sistema operativo.

Palabras Claves: Aplicación Web, Proceso Unificado de Desarrollo de Software, UML, Sistema de Interrupciones, SIDOR, Línea de producción, Productividad programada, Turnos laborales, Patrones de rotación de cuadrillas, Catálogos, Justificación de demoras, Toma de decisiones, Visual Studio.NET, ASP.NET, SQL Server, Internet Information Server.

INTRODUCCIÓN

La información dentro de las organizaciones desempeña un importante papel, dado que alimenta los negocios y puede ser el factor crítico para la determinación del éxito o fracaso de éstos, por ello para maximizar su utilidad debe ser manejada correctamente tal como se manejan los demás recursos dentro de la organización [1].

Para el debido control de la información dentro de la empresa, éstas se han visto obligadas a desarrollar sistemas de información, que no son más que herramientas para mejorar y agilizar el proceso de toma de decisiones en forma oportuna, ya que brindan una panorámica global de la situación, contribuyen a detectar y fortalecer las ventajas competitivas de la empresa, así como también los puntos críticos que pueden afectar el logro de las metas; sus beneficios son de impacto directo en la calidad de decisión, mejora en la comunicación, reducción de costos, incrementos en productividad, manejo e integridad de la información, entre otros, combinando modelos y datos para resolver problemas no estructurados utilizando una interfaz amigable para el usuario [1].

Con el pasar del tiempo los sistemas han evolucionado de acuerdo a los avances de la tecnología, el creciente uso de Internet y las necesidades de las organizaciones de vincular los distintos departamentos que la conforman, surgiendo así los sistemas de información bajo ambiente *Web*, que mejoran el manejo de la información y el acceso a los datos o procesos desde lugares distantes. En el marco de estas necesidades surge el concepto de Intranet, que consta de una infraestructura de comunicación basada en los estándares de comunicación de Internet y en los de la *World Wide Web* (WWW) en combinación con los servicios de mensajería, acceso remoto, y toda una serie de facilidades Cliente/Servidor proporcionadas por el protocolo *TCP/IP*. Su propósito fundamental es optimizar el flujo de información dentro de una organización con el objeto de lograr una importante reducción de

costos en el manejo de documentos y de la comunicación interna. Propone usar un *browser* o navegador *Web* como interfaz universal de información, proporcionando un entorno común a los usuarios de la red de computadoras [2].

Las aplicaciones basadas en la *Web* (*WebApps*) son sistemas confiables, prácticos y adaptables que ofrecen un arreglo complejo de contenido y funcionalidad a una amplia población de usuarios finales, en la actualidad las *WebApps* han evolucionado en sofisticadas herramientas de computación, que no solo proporcionan función por sí misma al usuario final, sino que también se han integrado como bases de datos corporativas y aplicaciones de negocios [3].

En el país muchas empresas no escapan de esta realidad y en la búsqueda de una mejor calidad en sus servicios, así como la mejora de eficiencia en el trabajo de sus empleados, han venido usando sistemas bajo ambiente *Web*, debido a que esta es una plataforma con mucha accesibilidad. Tal es el caso de la Siderúrgica del Orinoco Alfredo Maneiro, C.A. (SIDOR) que se define como un complejo siderúrgico productor de acero, dedicado a la fabricación y comercialización de productos destinados tanto al mercado nacional como internacional [4]. SIDOR cuenta con una estructura distribuida en direcciones (anexo A), siendo de especial interés para este trabajo la Dirección de Producción Industrial (anexo B), que se encarga de coordinar las actividades productivas.

Para el logro de sus objetivos SIDOR emplea numerosas *WebApps* que están centralizados en una sola aplicación que las agrupa, empleando los principios de sistemas distribuidos, por lo que el acceso a la aplicación y a sus servicios es posible desde cualquier computador dentro de la Intranet de la organización.

En concordancia, SIDOR cuenta con una vasta plataforma tecnológica que necesita soporte técnico, para garantizar la estabilidad de una gran cantidad de herramientas *Web*, para ello cuenta con la Dirección de Ingeniería y Medio Ambiente,

específicamente el Departamento de Automatización y Control, una de sus dependencias, que cumple con la función de diseñar y desarrollar *software* así como también de analizar problemas operativos, de procesos, de mantenimiento y de seguridad en los sistemas de información de la empresa [4] (anexo C). Conceptualmente el Departamento de Automatización y Control de SIDOR, mantiene un esquema piramidal de funcionamiento dividido por niveles (anexo D), siendo el nivel 2B el encargado de la explotación de datos del sistema disponible en el nivel 2A para su presentación por medio de tecnología *Web* y para visualización de datos históricos de planta, así como también se encarga de publicar las demoras y pérdidas de productividad de cada área de SIDOR [4].

Dentro de las *WebApps* disponibles en la Intranet de SIDOR, a las cuales atiende el Departamento de Automatización y Control, se encuentra el Sistema de Interrupciones (SI), cuya función es la de informar en tiempo real a la gerencia de cada planta sobre el estado de las distintas líneas de producción que allí se encuentran, indicando si la línea está detenida (interrupción), normal (en funcionamiento), en bajo ritmo (pérdida de producción) o no disponible (PLC no emite señal).

Los reportes que genera el SI, para la empresa, son de carácter estratégico ya que permiten: medir los eventos que causan interrupciones en el proceso de producción, analizar en forma rápida y confiable los problemas que se estén presentando en la línea de producción, representa una base para la elaboración de proyecciones y facilita evaluar el impacto de las acciones de quienes gestionan las distintas plantas.

Debido a lo expuesto anteriormente, se propuso desarrollar una aplicación bajo ambiente *Web* para la gestión del Sistema de Interrupciones de la Siderúrgica del Orinoco Alfredo Maneiro, que permita consolidar el SI de SIDOR, en el cual se incluya: un método gestión de usuarios según sus labores y permisos, administrar las

metas de producción mensuales de las distintas plantas que componen la empresa, clasificación de demoras arrojadas por el SI, gestión de un catálogo de desglose de demoras y administración de los patrones de rotación por turnos a las cuadrillas que atienden las distintas líneas que integran cada planta, todo esto según las disposiciones del personal responsable encargado. Además, el desarrollo de este software permitirá la integración de los datos necesarios para producir indicadores que apoyen la toma de decisiones de manera precisa y confiable, obteniendo beneficios en cuanto a disposición y evaluación de datos reales, así como ahorro de recursos, entre otros, garantizando calidad de servicio y evitar el manejo inapropiado de la información produciendo respuestas rápidas.

El desarrollo de la aplicación *Web* para la gestión de SI tiene su origen en la iniciativa existente por parte de la empresa SIDOR, en mejorar y optimizar sus procesos de gestión a fin de mantener el alto nivel que le ha caracterizado por años; reduciendo los costos de comunicación y logrando ofrecer a los trabajadores afectados un ambiente laboral ligero de cargas y/o procedimientos engorrosos y obsoletos mediante la actualización de los mismos de acuerdo a los avances tecnológicos con los que se cuenta en la actualidad.

El siguiente trabajo está estructurado en cuatro (4) capítulos, como se especifica a continuación:

Capítulo I. Presentación

El cual comprende planteamiento del problema, describe el problema planteado y el propósito de la investigación. Luego el alcance y limitaciones de la investigación, establece lo que el sistema será capaz de hacer.

Capítulo II. Marco de referencia

Está conformado por dos secciones principales: el marco teórico, presenta los fundamentos teóricos necesarios para soportar la investigación, describiendo los antecedentes de la investigación y la organización, además del área de estudio e investigación, en el cual está enmarcado el trabajo propuesto. El marco metodológico, presenta la metodología aplicada para elaborar la solución al problema planteado.

Capítulo III. Desarrollo

Aquí se expone de forma detallada la aplicación de los procedimientos descritos en el marco metodológico para el logro de los objetivos planteados.

Finalmente, se presentan las conclusiones donde se muestra una apreciación global del trabajo realizado para la obtención de una posible solución, seguida de las recomendaciones sugeridas para las investigaciones posteriores.

CAPÍTULO I.

PRESENTACIÓN

1.1 Planteamiento del problema

Dentro de la empresa SIDOR se encuentran una serie de Plantas como lo son Laminación en Frío, Laminación en Caliente, Áreas Primarias, Acería de Palanquillas, Acería de Planchones, Planta de Cal y Barras y Alambrón, utilizadas para llevar a cabo todo el proceso de producción acero hasta llegar a sus distintas terminaciones. En cada una de estas plantas se hace uso de un Sistema de Interrupciones, que tiene como función monitorear y determinar el estado de funcionamiento o nivel de velocidad de las distintas líneas de producción con la finalidad de detectar fallas y pérdidas, así como mejorar la comunicación y la respuesta a los problemas que puedan presentarse. Para la realización de estas operaciones el Sistema de Interrupciones presenta una serie de debilidades, como lo son el retraso en el proceso de clasificación y justificación de las demoras que se detectan con relación al exceso de tiempo empleado, desorganización al momento de llevar el manejo de datos y diversidad de información errada.

En el transcurso del tiempo que lleva en funcionamiento el SI, se ha evidenciado por parte de los usuarios directos la falta de información en los reportes que genera; entre ellos: un catálogo amplio donde se pueda desglosar las características de una demora y un reporte de gestión que permita planificar la producción en base a información actualizada.

Los catálogos son arreglos que permiten que el usuario pueda identificar en el SI las características de una demora, como por ejemplo, la zona en que ocurrió, el evento que la originó, el sistema donde sucedió, el tipo de demora, la categoría a la que pertenece, entre otros. Representa información importante que alimenta al sistema

para la explotación de datos, pero su gestión debe seguir las políticas de la empresa y mantener un formato, además, sólo pueden ser manejados por las personas responsables en el momento que sea necesario, lo cual ha ocasionado la imposibilidad de realizar una gestión para la actualización del catálogo, ya que las plantas han cambiado su estructura o han crecido, donde obviamente las condiciones de trabajo o las labores realizadas no son las mismas, pues son lugares, circunstancias y características distintas, para lo cual no se tomó ninguna previsión.

Actualmente, los encargados de los requerimientos de las plantas tienen que especificar en hojas de cálculo los datos de los elementos que deben ser incluidos en el catálogo, para luego ser enviadas al Departamento de Automatización y Control donde se integran estos nuevos elementos al SI. Ante estas necesidades no hay una capacidad de repuesta, ya que muchas veces el departamento no atiende las solicitudes de los usuarios de manera inmediata, siendo esto un factor que aumenta la posibilidad de contar con datos incongruentes con respecto al tiempo, generando desconfianza en el sistema y nuevos problemas que deberán ser tratados con posterioridad.

Lo mismo ocurre para la gestión de la productividad programada, que marca las pautas que debe seguir el SI para determinar cuando ocurre una pérdida de productividad. La productividad programada está compuesta por datos que están en constante cambio, pues cada planta se traza metas de producción mensual y las pérdidas son calculadas en base a esta, es decir que una pérdida de productividad constituye la disminución de capacidad de producción contra la meta establecida. Por este motivo, los datos de productividad programada deben actualizarse a la mayor brevedad posible y con el menor margen de error, lo que constituye una carga masiva de datos. Todo esto dificulta las metas trazadas por la Dirección de Ingeniería y Medio Ambiente para el Departamento de Automatización y Control, ocasionando perjuicios en otros proyectos de desarrollo importantes que amerita la empresa, ya

que el grupo de trabajo tiene que dejarlos, total o parcialmente, para atender peticiones de los usuarios del SI, implicando un consumo de tiempo excesivo.

Las actividades antes mencionadas como la gestión de catálogos y de productividad programada solo puede ser realizada por usuarios pertenecientes al personal de la Dirección de Producción Industrial, que son los encargados de toda la gestión en las plantas, siendo ineludible establecer políticas de seguridad. También se hace necesario la implementación de una gestión de patrones, que comprende la rotación de cuadrillas que laboran en las plantas según el modelo de turnos que emplea la empresa para mantener sus niveles de producción. Por lo que se hace necesario que se realicen agrupaciones lógicas de los módulos de la aplicación *Web* de gestión, atendiendo a distintas funcionalidades del SI, según los perfiles de usuarios correspondientes.

Debido a la complejidad y relevancia de la gestión del SI, así como al constante aumento de las líneas automatizadas que pasan a formar parte del mismo, se propuso el desarrollo de una aplicación que permita a los usuarios finales autogestionar la administración de los catálogos, productividad programada, perfiles de usuarios, patrones y clasificación de demoras, permitiendo una actualización en mejor tiempo de la información necesaria para la toma de decisiones.

1.2 Alcance

El trabajo está enmarcado en el desarrollo de una aplicación basada en la *Web*, que apoye los servicios del Departamento de Automatización y Control vinculados a la gestión del SI en cada una de las plantas de SIDOR donde se encuentra instalado. Dicha aplicación permite la consolidación del SI mediante la aplicación de una autogestión por parte de los usuarios finales, por lo que, el software está en capacidad de realizar las siguientes funcionalidades:

Clasificar todas las causas y eventos por la cual sucedió una demora o la pérdida de producción.

Asignar responsables por cada demora clasificadas.

Administrar los catálogos de desglose de demoras, de acuerdo a las resoluciones emitidas por la Dirección de Producción Industrial.

Gestionar las metas mensuales de producción, programadas por el personal de la Dirección de Producción Industrial.

Administrar usuarios para mantener la seguridad y funcionamiento eficiente del SI de SIDOR.

Gestionar patrones de rotación de las cuadrillas de trabajo según el modelo de turnos planteado por SIDOR para mantener los niveles de producción.

1.3 Limitaciones

Dado que este proyecto fue planteado bajo la modalidad pasantía, el tiempo de desarrollo del mismo estuvo sujeto a la planificación del Departamento de Automatización y Control, lo que llevo a que se ejecutara según el Proceso Unificado de Desarrollo de Software, solo las fases de inicio, elaboración y construcción.

La planificación del desarrollo del proyecto debió ser reprogramado, puesto que se dedicó tiempo a inducción en las tecnologías que comprenden la plataforma de la empresa, ya que no se poseía conocimiento sobre las mismas.

Debido a las constantes manifestaciones con relación a la nacionalización de SIDOR y la absorción del personal de las contratistas por parte de la empresa, se demoró la construcción de la aplicación.

CAPÍTULO II.

MARCO DE REFERENCIA

2.1 Marco teórico

2.1.1 Antecedentes de la investigación

La elaboración de este proyecto se inició con la búsqueda de fuentes de información, efectuando revisiones bibliográficas con el fin de establecer un marco de referencia basado: primero, en los trabajos realizados en el área de Automatización y Control dentro de SIDOR; y segundo, en los conocimientos previos que deben tomarse en cuenta para desarrollar el proyecto.

El uso de aplicaciones basadas en la *Web* ha experimentado un gran crecimiento debido al auge que ha obtenido la Internet en la sociedad actual y a las diversas características que las definen como herramienta indispensable para el esquema de las organizaciones modernas.

En las empresas por ejemplo, el uso de estas aplicaciones ha sido de mucha importancia, ya que a través de sus sitios *Web* muestran al mundo información organizacional, servicios ofrecidos, ofertas, estrategias de mercadeo, entre otros.

En SIDOR, se han venido creando una serie de aplicaciones *Web* para el apoyo de las actividades realizadas en sus dependencias. Algunos de los trabajos de grado que han sido seleccionados para el marco de referencia se describen a continuación:

Gabriel Vergara en su trabajo de grado, titulado: Diseño e Implementación bajo Plataforma *Web* del Sistema de Interrupciones (SI) para la Planta de Briquetas (MATESI) de SIDOR C.A., da a conocer como el uso de estas tecnologías, facilitan a los gerentes, procesistas, inspectores, supervisores, operadores y jefes de planta, así

como al comité de calidad, visualizar los indicadores necesarios que muestren el estado en tiempo real de una línea de producción con el fin de determinar soluciones inmediatas en caso de fallas que influyan directamente en la disminución de la producción [5].

Ángel Sierralta (2005) en su trabajo de grado, titulado: Sistema *Web* para la Gestión de la Calidad en los Procesos de Laminación de SIDOR, expresa que el uso de las nuevas tecnologías que se aplican en SIDOR, facilitan la gestión de los procesos, mediante el desarrollo de aplicaciones *Web* basadas en tecnología de punta tal como *.NET*, ya que esta nueva herramienta, en conjunto con manejadores de bases de datos avanzados como *SQL Server*, permiten un amplio desarrollo de aplicaciones *Web* robustas y estables, las cuales están ligadas directamente a los procesos que ocurren en planta[6].

Lisbeth Ruiz (2005) en su trabajo de grado, titulado: Sistema de Información para la Gestión de Materiales Refractarios de SIDOR C.A., explica que para el desarrollo del proyecto se utilizó el Proceso Unificado de Desarrollo de Software, llevando a cabo un desarrollo iterativo en el cual se realiza análisis, diseño e implementación en cada una de las iteraciones, lo cual facilitó el análisis y diseño del Sistema de Refractarios y ocasionó que el mismo fuera un sistema bastante robusto, así como también su mantenimiento se hace bastante sencillo[7].

Rincón, Yessica (2006), en su Trabajo de Grado, titulado: Sistemas de Gestión en Tornería de Cilindros de Laminación en Frío de la Empresa SIDOR, explica la importancia del desarrollo y la implantación de los sistemas de gestión en las áreas productivas de la empresa SIDOR, específicamente en el área de laminación en frío, para el mejoramiento de los procesos, y la comunicación entre los diferentes niveles, para la toma de decisiones y aumento de la efectividad [8].

2.1.2 Antecedentes de la organización

La Siderúrgica del Orinoco Alfredo Maneiro, SIDOR es un complejo siderúrgico integrado que utiliza tecnologías de Reducción Directa y Hornos Eléctricos de Arco. Los procesos de esta siderúrgica se inician con la fabricación de Pellas y culminan con la entrega de productos finales Largos (Barras y Alambrón) y planos (Láminas en Caliente, Láminas en Frío y Recubiertos). Este complejo está ubicado en la zona industrial de Matanzas, estado Bolívar, región suroriental de Venezuela, sobre la margen derecha del río Orinoco, a 282 km de su desembocadura en el océano Atlántico.

Esta siderúrgica ubica a Venezuela en cuarto lugar como productor de acero integrado de América Latina y el principal de la región Andina, ha logrado colocar su nivel de producción en torno a los 4 millones de toneladas de acero líquido por año, con indicadores de productividad, rendimiento total de calidad, oportunidad en las entregas y satisfacción de sus clientes, comparables con las empresas más competitivas de Latinoamérica. Es reconocida además por ser el primer exportador no petrolero del país.

Desde el 12 de mayo del 2008, SIDOR es una empresa perteneciente al Estado Venezolano, luego de que el Presidente de la República Hugo Chávez Frías, decretará la nacionalización de la misma, la cual en 1997 había sido privatizada.

SIDOR cuenta actualmente con una Presidencia Ejecutiva, la cual está conformada por Direcciones Ejecutivas encargadas de realizar diferentes tareas administrativas o de gestión según sea el campo de aplicación o áreas de desempeño. La Presidencia Ejecutiva delega a las Direcciones, las funciones relacionadas con el ámbito interno de la empresa (Anexo A).

Una de las direcciones con mayor relevancia dentro de la empresa es la Dirección de Producción Industrial, que se encarga de coordinar las actividades productivas y tiene como objetivo primordial el fabricar productos siderúrgicos así como prestar los servicios industriales requeridos de manera competitiva y rentable. Esta Dirección está conformada por un conjunto de Gerencias, ejecutando cada una de sus funciones de acuerdo al área específica o campo de aplicación (Anexo B).

Las Gerencias representan de alguna forma los principales procesos productivos de la empresa, organizados de acuerdo a sus productos o servicios.

Los proyectos de automatización, aunque sean implementados en cualquier planta o gerencia de la empresa, son diseñados y elaborados en el Departamento de Automatización y Control, ubicada dentro de la Gerencia de Ingeniería y Medio Ambiente (Anexo C).

Para la automatización y control de procesos en SIDOR se ha desarrollado una plataforma que permite controlar todos los servicios, variables, comandos, eventos (alarmas) y visualizar en su totalidad el proceso productivo, la cual está conformada por 4 niveles (Anexo D), descritos a continuación:

- **Nivel 0**, se encuentran los elementos de campo (sensores, actuadores, etc.).
- **Nivel 1**, se encuentran los elementos de control (PLC's) y monitoreo de las variables de campo como por ejemplo la temperatura. El PLC recibe toda la información de los elementos de campo.
- **Nivel 2**, presenta divisiones:

El nivel 2A, es donde se realiza el monitoreo y control en tiempo real de los procesos. Es aquí donde intervienen los operadores de cabina a visualización y seguimiento en tiempo real de las actividades productivas.

El nivel 2B, está basado en tecnología *Weby* es donde se lleva a cabo el monitoreo de datos a través de los sitios de las distintas líneas de producción.

- *Nivel 3*, nivel Gerencial es donde se desarrolla la explotación de los datos, análisis estadístico del proceso y control de gestión. Intervienen las gerencias de las líneas.

2.1.3 Área de estudio

Está definida por las pautas del desarrollo de sistemas de información automatizados, puesto que la aplicación *Web* de gestión del SI de SIDOR hace uso del computador para la mejora de los procesos vinculados a la gestión de catálogos, productividad programada, administración de usuarios, patrones y clasificación de demoras.

- Sistemas de información: son una disposición de personas, actividades, datos, redes y tecnología integrados entre sí, con el propósito de apoyar y mejorar las operaciones cotidianas de una empresa, así como de satisfacer las necesidades de información para la resolución de problemas y la toma de decisiones por parte de los directivos de la empresa[9].
- Análisis y diseño de sistemas: el análisis y diseño de sistemas es un procedimiento para la resolución de problemas. Cuando se trata del diseño de sistemas de información, busca analizar sistemáticamente la entrada o flujo de datos, la transformación de los datos, el almacenamiento de datos y la salida de información en el contexto de una organización particular. También es usado para analizar, diseñar e implementar mejoras que puedan incorporarse a la organización y puedan ser alcanzadas al usar un sistema de información computarizado [10].
- Análisis y diseño de sistemas orientado a objetos: abordando el análisis y diseño desde el paradigma orientado a objetos, se describe el análisis al poner

énfasis en una investigación del problema y los requisitos, en vez de ponerle una solución. El análisis se debe calificar como análisis de requisitos (mediante un estudio de los requisitos) o análisis de objetos (estudiando los objetos del dominio). Por otro lado, el diseño pone énfasis en una solución conceptual que satisface los requisitos y prestando atención a la definición de los objetos de software y en cómo colaboran para satisfacer los requisitos, en vez de ponerlos en la implementación [11].

- Modelado de datos: los modelos de datos aportan la base conceptual para diseñar aplicaciones que hacen un uso intensivo de datos, así como la base formal para las herramientas y técnicas empleadas en el desarrollo y uso de sistemas de información. Con respecto al diseño de bases de datos, el modelado de datos puede ser descrito como: dados los requerimientos de información y proceso de una aplicación de uso intensivo de datos (por ejemplo, un sistema de información), construir una representación de la aplicación que capture las propiedades estáticas y dinámicas requeridas para dar soporte a los procesos deseados (por ejemplo, transacciones y consultas). Además de capturar las necesidades dadas en el momento de la etapa de diseño, la representación debe ser capaz de dar cabida a eventuales futuros requerimientos [12].
- Bases de datos: es un conjunto de datos relacionados entre sí. Por datos se entiende aquellos hechos conocidos que pueden registrarse y que tienen un significado implícito. Toda base de datos se puebla con datos para un propósito específico. Una base de datos es un conjunto de datos lógicamente coherente, con cierto significado inherente. Una colección aleatoria de datos no puede considerarse propiamente una base de datos [13].
- Sistemas cliente-servidor: en los sistemas cliente-servidor se distribuye el proceso de una aplicación entre múltiples ordenadores (de forma transparente

para el usuario) en una red LAN o WAN. En una aplicación cliente-servidor, los programas GUI (interfaz gráfica de usuario para entradas y salidas) funcionan normalmente en el propio PC o en la estación de trabajo inteligente del usuario, que recibe el nombre de cliente. Otras partes del mismo programa de aplicación (como las lecturas y escrituras en las bases de datos) se realizan en otros elementos informáticos denominados servidores [9].

- Lenguaje de Modelado Unificado (UML): es un lenguaje que permite modelar, construir y documentar los elementos que forman un sistema orientado a objetos. UML no define un proceso de desarrollo específico; es una notación que puede ser aplicada a diferentes tipos de sistemas informáticos o de otras ramas. Este lenguaje posibilita la captura, comunicación y nivelación de conocimiento estratégico, táctico y operacional para facilitar el incremento de valor, aumentando la calidad, reduciendo costos y reduciendo el tiempo de presentación al mercado; manejando riesgos y siendo proactivo para el posible aumento de complejidad o cambio [14].

2.1.4 Área de investigación

Esta investigación se enmarca en el área de los sistemas de información bajo ambiente *Web*, que está constituido por todo lo referente a tecnologías de automatización, informática y telecomunicaciones que apoyen al negocio.

Las páginas *Web* son documentos compuestos por texto, elementos multimedia (gráficos, sonido, vídeo digital) y vínculos [15].

Una aplicación *Web* es un conjunto de páginas *Web* que interactúan entre sí, con el usuario y con diversos recursos en un servidor *Web*, incluidas bases de datos [15].

Una aplicación *Web* contiene páginas almacenadas en un servidor *Web* con contenido parcial o totalmente indeterminado. El contenido final de una página se determina

sólo cuando el usuario solicita una página del servidor *Web*. Las aplicaciones *Web* se crean en respuesta a diversas necesidades o problemas [15].

Los términos involucrados en esta área son los siguientes:

- Fundamentos de la *Web*: los argumentos principales que dan sostén a la *Web* son el protocolo HTTP y el lenguaje HTML. Uno permite una implementación simple y sencilla de un sistema de comunicaciones que nos permite enviar cualquier tipo de ficheros de una forma fácil, simplificando el funcionamiento del servidor y permitiendo que servidores poco potentes atiendan miles de peticiones y reduzcan los costes de despliegue. El otro nos proporciona un mecanismo de composición de páginas enlazadas simple y fácil, altamente eficiente y de uso muy simple.
- El protocolo HTTP: el protocolo HTTP (*hypertext transfer protocol*) es el protocolo base de la WWW. De manera esquemática, el funcionamiento de HTTP es el siguiente: el cliente establece una conexión TCP hacia el servidor, hacia el puerto HTTP (o el indicado en la dirección de conexión), envía un comando HTTP de petición de un recurso (junto con algunas cabeceras informativas) y por la misma conexión el servidor responde con los datos solicitados y con algunas cabeceras informativas.
- El lenguaje HTML: el otro puntal del éxito del WWW ha sido el lenguaje HTML (*hypertext mark-up language*). Se trata de un lenguaje de marcas (se utiliza insertando marcas en el interior del texto) que nos permite representar de forma rica el contenido y también referenciar otros recursos, enlaces a otros documentos, mostrar formularios para posteriormente procesarlos, etc. El lenguaje HTML actualmente se encuentra en la versión 4.01 y empieza a proporcionar funcionalidades más avanzadas para crear páginas más ricas en contenido. Además se ha definido una especificación compatible con HTML, el

XHTML (*extensible hypertext markup language*) que se suele definir como una versión XML validable de HTML, proporcionándonos un XML *Schema* contra el que validar el documento para comprobar si está bien formado, etc.

- Servidor *Web*: un servidor *Web* es un programa que atiende y responde a las diversas peticiones de los navegadores, proporcionándoles los recursos que solicitan mediante el protocolo HTTP o el protocolo HTTPS (la versión segura, cifrada y autenticada de HTTP).
- Diseño de páginas *Web*: el lenguaje HTML se utiliza para crear documentos que muestren una estructura de hipertexto. Un documento de hipertexto es aquel que contiene información cruzada con otros documentos, lo cual nos permite pasar de un documento al referenciado desde la misma aplicación con la que lo estamos visualizando. HTML permite, además, crear documentos de tipo multimedia, es decir, que contengan información más allá de la simplemente textual, como por ejemplo: imágenes, sonido, video, subprogramas activos. El lenguaje HTML no es el único lenguaje existente para crear documentos hipertexto. Hay otros lenguajes anteriores o posteriores a HTML (SGML, XML, etc.), si bien HTML se ha convertido en el lenguaje estándar para la creación de contenido para Internet.
- *.NET Framework*: es una plataforma informática sobre la que se reúne todo un conjunto de lenguajes y servicios que simplifican enormemente el desarrollo de aplicaciones. esta herramienta ofrece un entorno de ejecución altamente distribuido, que permite crear aplicaciones robustas y escalables. El diseño de *.NET Framework* está enfocado a cumplir los objetivos siguientes:[16]
- Proporcionar un entorno coherente de programación orientada a objetos, en el que el código de los objetos se pueda almacenar y ejecutar de forma local, pero distribuida en Internet o ejecutar de forma remota.

Proporcionar un entorno de ejecución de código que reduzca lo máximo posible la implementación de software y los conflictos de versiones.

Ofrecer un entorno de ejecución de código que garantice la ejecución segura del mismo, incluso del creado por terceras personas desconocidas o que no son de plena confianza.

Proporcionar un entorno de ejecución de código que elimine los problemas de rendimiento de los entornos en los que se utilizan secuencias de comandos o intérpretes de comandos.

Ofrecer al programador una experiencia coherente entre tipos de aplicaciones muy diferentes, como las basadas en *Windows* o en el *Web*.

Basar toda la comunicación en estándares del sector para asegurar que el código de *.NET Framework* se puede integrar con otros tipos de código.

- *ASP.NET*: está construida sobre las clases de programación del entorno *.NET Framework*, y proporciona un modelo para aplicaciones *Web* con un conjunto de controles y una infraestructura que simplifican la creación de aplicaciones *Web ASP*. *ASP.NET* incluye un conjunto de controles que encapsulan los elementos comunes de la interfaz de usuario de HTML, como cuadros de texto y menús desplegados. Sin embargo, dichos controles se ejecutan en el servidor *Web*, y envían al explorador *Web* su interfaz de usuario en forma de HTML. En el servidor, los controles exponen un modelo de programación orientado a objetos que ofrece al programador de *Web* la riqueza de la programación orientada a objetos.

ASP.NET proporciona también servicios de infraestructura, como administración de estado de sesión y reciclado de procesos, que reducen aún más la cantidad de código que debe escribir el programador, e incrementan la fiabilidad de las aplicaciones.

- *Visual Studio .NET*: es un conjunto de aplicaciones completo para la creación tanto de aplicaciones de escritorio como de aplicaciones *Web* de empresa para trabajo en equipo. Aparte de generar aplicaciones de escritorio de alto rendimiento, se pueden utilizar las eficaces herramientas de desarrollo basado en componentes y otras tecnologías de *Visual Studio* para simplificar el diseño, desarrollo e implementación en equipo de soluciones para empresa[16].

Visual Studio.NET comprende herramientas de desarrollo para la construcción de aplicaciones *Web* ASP, servicios *Web* XML, aplicaciones para escritorio y aplicaciones móviles. *Visual Basic.NET*, *Visual C++.NET*, *Visual C#.NET* y *Visual J#.NET* utilizan el mismo entorno de desarrollo integrado (IDE), que les permite compartir herramientas y facilita la creación de soluciones en varios lenguajes. Asimismo, dichos lenguajes aprovechan las funciones de *.NETFramework*, que ofrece acceso a tecnologías clave para simplificar el desarrollo de aplicaciones *Web* ASP y servicios *Web* XML.

- *Visual Basic.NET*: es un lenguaje rediseñado Orientado a Objetos, que mejora el Visual Basic tradicional e incluso rompe la compatibilidad con los programas Visual Basic clásicos existente. En resumen *Visual Basic.NET* es un lenguaje elegante, moderno e ideal para crear la siguiente generación de aplicaciones empresariales. En *Visual Basic .NET* a diferencia de lo que ocurría en las versiones anteriores de *Visual Basic*, sólo existe un tipo de fichero de código, el cual tiene la extensión *.vb*, en este tipo de fichero pueden coexistir distintos tipos de elementos, por ejemplo: un módulo de clase, un formulario, un módulo de código, un control, etc.; mientras que en las versiones anteriores de *Visual Basic*, cada uno de estos elementos tenían su propio tipo de fichero con su respectiva extensión. Por su parte El *ASP.NET* de *Microsoft* es una tecnología de scripts que corren en el servidor y pueden ser utilizados para crear aplicaciones dinámicas e interactivas en la *Web*. Una página *ASP.NET* es una

página de HTML que contiene scripts que son procesados por un servidor *Web* antes de ser enviados al navegador del usuario. El código *ASP.NET* es más “compacto” que el código ASP; los scripts requeridos para realizar una función dada son más cortos en *ASP.NET* que en ASP. Debido a que los scripts que corren del lado del servidor están contruidos en una página regular de HTML, pueden ser entregados en casi cualquier navegador, Un archivo *ASP.NET* puede ser creado utilizando cualquier herramienta de edición, como el *Notepad* [16].

- Servidor de aplicaciones: es un software que ayuda al servidor *Web* a procesar las páginas que contienen *scripts* o etiquetas del lado del servidor. Cuando se solicita al servidor una página de este tipo, el servidor *Web* pasa la página al servidor de aplicaciones para su procesamiento antes de enviarla al navegador [15].
- Acceso a una base de datos: un servidor de aplicaciones le permite trabajar con recursos del lado del servidor, como una base de datos. Por ejemplo, una página dinámica puede indicar al servidor de aplicaciones que extraiga datos de una base de datos y los inserte en el código HTML de la página [15].

La instrucción para extraer datos de una base de datos recibe el nombre de consulta de base de datos, la cual consta de criterios de búsqueda expresados en un lenguaje denominado SQL (*Structured Query Language*: lenguaje de consulta estructurado). La consulta SQL se escribe en los scripts o etiquetas del lado del servidor de la página [15].

Un servidor de aplicaciones no puede comunicarse directamente con una base de datos porque el formato propio de la base de datos hace que los datos sean ilegibles. El servidor de aplicaciones sólo se puede comunicar a través de un controlador de base de datos que actúa como intérprete entre la aplicación *Web* y la base de datos [15].

- Base de Datos: es un conjunto de datos estructurados, almacenados en algún soporte de almacenamiento de datos y se puede acceder a ella desde uno o varios programas. Antes de diseñar una base de datos se debe establecer un proceso partiendo del mundo real, de manera que sea posible plasmar éste mediante una serie de datos.

La imagen que se obtiene del mundo real se denomina modelo conceptual y consiste en una serie de elementos que definen lo que se quiere plasmar del mundo real en la base de datos [17].

- Sistema Gestor de Bases de Datos (SGBD): es un conjunto de programas, procedimientos y lenguajes que proporcionan a los usuarios las herramientas necesarias para operar con una base de datos.

Por tanto, el SGBD actúa como un intermediario entre los usuarios y los datos. Debe cumplir una serie de funciones como descripción de los datos, de manera que debe permitir definir los registros, sus campos, sus relaciones de autorización, etc. Debe manipular los datos permitiendo a los usuarios insertar, suprimir, modificar y consultar datos de la base de datos y por último, debe permitir usar la base de datos, dando una interfaz adecuada a cada tipo de usuario [17].

- Motor de base de datos: es el encargado de realizar las consultas, altas, bajas, modificaciones, procedimientos, etc. de forma transparente al usuario [17].
- *SQL Server*: es un sistema de gestión de base de datos relacionales (SGBD) cliente/servidor de alto rendimiento, que tiene como función, administrar el almacenamiento de datos para transacciones y análisis, responder a las solicitudes de aplicaciones cliente y utiliza *Transact-SQL* para enviar solicitudes entre un cliente y *SQL Server* [18].

2.2 Marco metodológico

2.2.1 Metodología de la investigación

- Forma de investigación: el objetivo de este proyecto es desarrollar una aplicación *Web* para la gestión del SI de SIDOR, partiendo de problemas existentes en condiciones y características precisas.

Por lo tanto, la forma de investigación es aplicada, porque comprende el estudio y aplicación de la investigación a problemas específicos, en circunstancias y características concretas[19].

- Tipo de investigación: se estudió la forma en que se gestiona el SI, así como el grupo de empleados que llevan a cabo esta tarea, con la finalidad de conocer las situaciones y comportamientos predominantes a través de la descripción exacta de las actividades, objetos, procesos y personas para fijar conclusiones precisas que reseñen la problemática existente.

En base a lo anterior, se puede decir que el tipo de la investigación es descriptiva, ya que comprende la descripción, registro, análisis e interpretación de la naturaleza actual, y la composición de los fenómenos [19].

- Técnicas de recolección de datos: los datos que ayudaron a determinar los requerimientos que cubrirá la aplicación *Web* de gestión del SI, fueron recolectados utilizando las entrevistas no estructuradas, los cuales se aplicaron a todo el personal involucrado en la administración del SI, así como a los usuarios finales, de igual manera se aplicó la técnica de observación directa, así como también la revisión documental, consultas en Internet y consultas bibliográficas.

2.2.2 Metodología del área aplicada

La aplicación *Web* para la gestión del SI de SIDOR, se desarrolló utilizando como metodología el Proceso Unificado de Desarrollo de Software planteado por Jacobson, Booch y Rumbaugh [20].

El proceso unificado es un proceso de desarrollo de software. Un proceso de desarrollo de software es el conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema software, como se muestra en la figura 1:

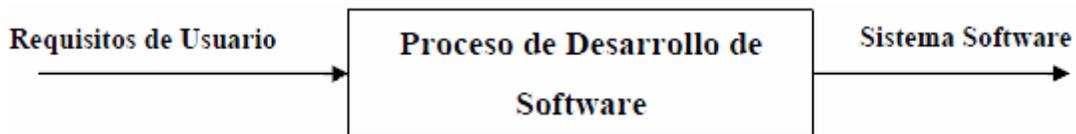


Figura 1. Proceso Entrada Salida

Sin embargo el proceso unificado de desarrollo de software es más que un simple proceso; es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud y diferentes tamaños de proyectos. El Proceso Unificado utiliza el Lenguaje Unificado de Modelado (Unified Modeling Language, UML) para preparar todos los esquemas de un sistema software. De hecho, UML es una parte esencial del proceso unificado. No obstante, los verdaderos aspectos definitorios del Proceso Unificado de Desarrollo de Software se resumen en frases clave; dirigido por casos de usos, centrado en la arquitectura, e iterativo e incremental.

Dirigido por casos de usos: cuando un usuario utiliza un sistema, se lleva a cabo una interacción que consiste en una secuencia de acciones (tanto por parte del usuario como del sistema) que proporcionan al usuario un resultado de valor importante para él. Un caso de uso es precisamente una interacción de ese tipo, que deriva en que el

sistema debe incluir un fragmento de funcionalidad para proporcionar al usuario el resultado de valor esperado.

Centrado en la arquitectura: la arquitectura del software sirve para poder contemplar el futuro sistema desde varios puntos de vista antes de que se construya y durante la construcción. El concepto de arquitectura software incluye los aspectos estáticos y dinámicos más significativos del sistema, es decir, especifica la forma, estructura y comportamiento del sistema desde diversos puntos de vista, todos ellos a un nivel de detalle que permita tener una idea global, clara, sin perderse en detalles insignificantes (que, precisamente, no influyen en la arquitectura del sistema). De esta forma quedan cubiertos dos aspectos fundamentales: ¿Qué hace el sistema para cada usuario? (casos de uso); ¿Cómo es el sistema y cómo funciona para realizar esos casos de uso? (arquitectura software).

Iterativo e incremental: para manejar el enorme esfuerzo necesario al ejecutar un proyecto, éste se divide en mini-proyectos llamados iteraciones. Cada iteración del proyecto (trabajo realizado) toma como entrada el producto resultado de la iteración anterior y genera como salida un producto incrementado, existiendo incrementos de dos tipos:

- Adición de nuevos artefactos al producto: es decir, creación o ampliación de los modelos (negocio, casos de uso, análisis, diseño o implementación). Se ha demostrado que los procesos en cascada no son efectivos a la hora de construir sistemas medianos/grandes, y que lo correcto es construir el sistema por partes, es decir, incrementalmente.
- Sustitución o modificación de algún artefacto del producto: es decir, modificación de alguno de los modelos creados anteriormente. Al final de una iteración, al realizar las pruebas, se pueden descubrir fallos producidos por artefactos defectuosos que deberán ser revisados en la siguiente iteración. En

general, es posible que un modelo no esté bien definido, siendo necesario revisar el trabajo hecho.

Ciclos de Proceso Unificado: el Proceso Unificado de Desarrollo de Software se repite a lo largo de una serie de ciclos que constituyen la vida de un sistema. Cada ciclo concluye con una versión del producto para los clientes.

A continuación, a través de las figuras 2 y 3, se pueden observar los ciclos del Proceso Unificado de Desarrollo de Software:

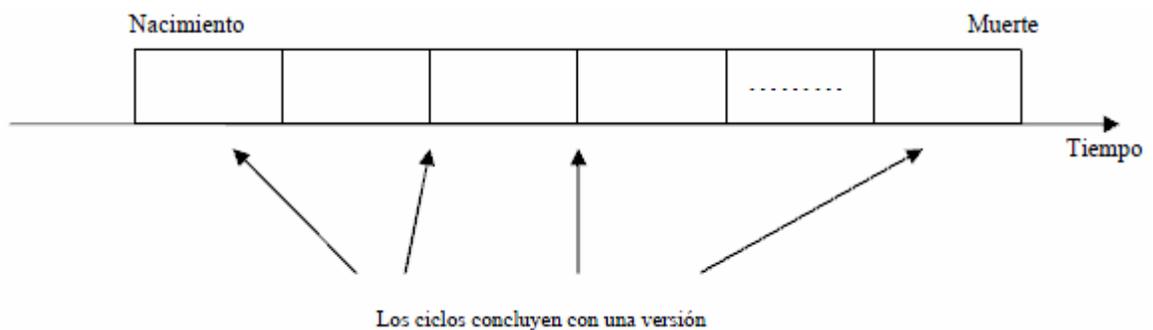


Figura 2. Ciclos del Proceso Unificado de Desarrollo de Software.

Cada ciclo consta de cuatro fases: inicio, elaboración, construcción y transición. Cada fase se subdivide a su vez en iteraciones, como se puede ver en la figura 3:

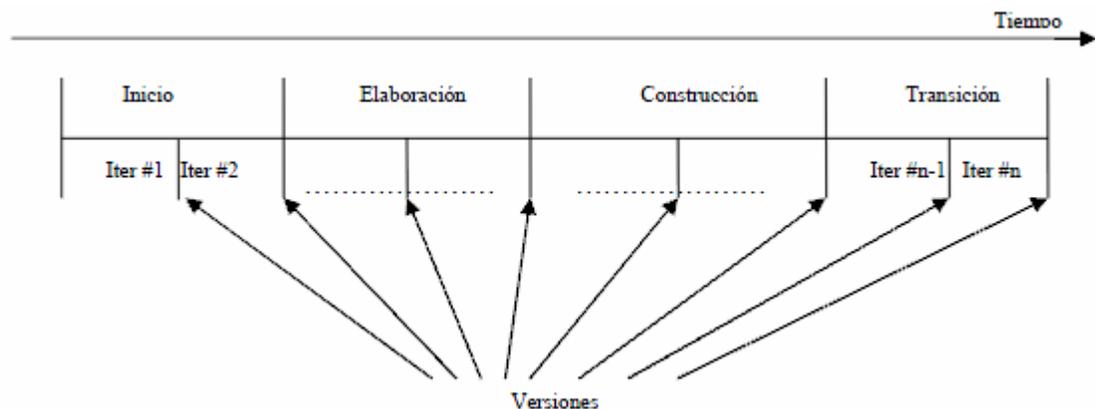


Figura 3. Fases del Proceso Unificado de Desarrollo de Software.

Fases dentro de un Ciclo: el Proceso Unificado de Desarrollo de Software, contempla que cada uno de sus ciclos se desarrollan a lo largo del tiempo.

A través de una secuencia de modelos es posible visualizar lo que esta sucediendo en cada una de las fases del proceso.

A continuación, en la figura 4 se puede observar como el tiempo en el Proceso Unificado de Desarrollo de Software, se divide en fases.

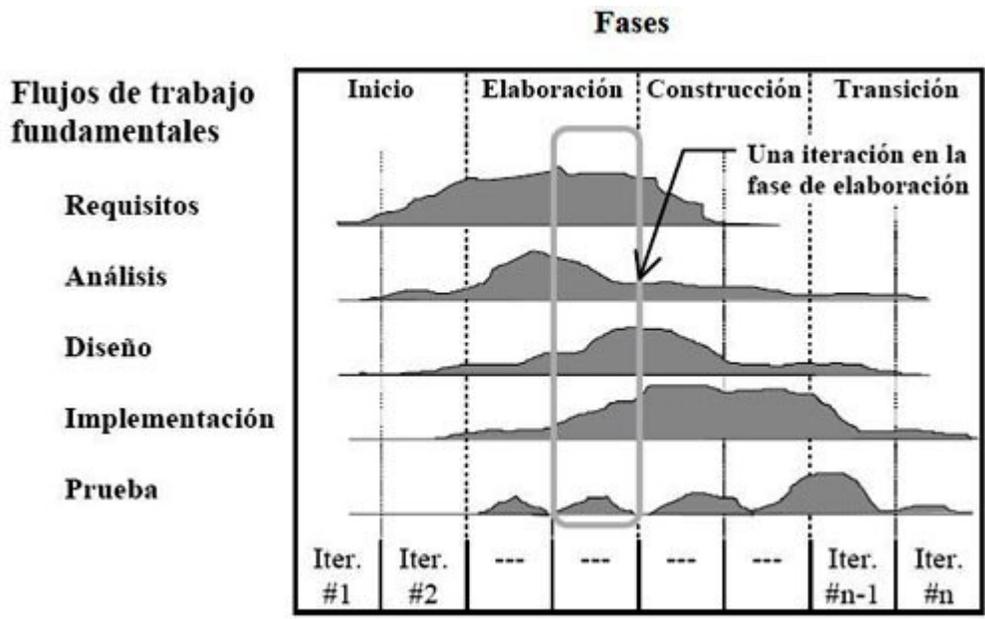


Figura 4. Fases

La figura 4 muestra en la columna izquierda los flujos de trabajo; requisitos, análisis, diseño, implementación y pruebas. Las curvas son una aproximación de hasta donde se llevan a cabo los flujos de trabajo en cada fase. Se debe recordar que cada fase se divide en iteraciones. Una iteración típica pasa por los cinco flujos de trabajo.

Durante la **Fase de Inicio**, se desarrolla una descripción del producto final a partir de una buena idea y se presenta el análisis de negocio para el producto. En esta fase se

responde a la pregunta; ¿Cuáles son las principales funciones del sistema para sus usuarios más importantes? La respuesta es que se encuentra en un modelo de casos de usos simplificados que contenga los casos de uso más críticos. En esta fase se identifican y priorizan los riesgos más importantes, se planifica en detalle la fase de elaboración, y se estima el proyecto de manera aproximada.

Durante la **fase de elaboración**, se especifica en detalle la mayoría de los casos de uso y se diseña la arquitectura del sistema. La relación entre la arquitectura del sistema y el propio sistema es primordial. Por lo tanto, la arquitectura se expresa en forma de vistas de todos los modelos del sistema, los cuales juntos representan el modelo entero. Esto implica que hay vistas arquitectónicas del modelo de casos de uso, del modelo de análisis, del modelo de diseño, del modelo de implementación y el modelo de despliegue.

Durante la **fase de construcción**, se crea el producto. En esta fase la línea base de la arquitectura crece hasta convertirse en el sistema completo. La descripción evoluciona hasta convertirse en un producto preparado para ser entregado a la comunidad de usuarios. Al final de esta fase, el producto contiene todos los casos de uso que se han acordado para el desarrollo de esta versión. Sin embargo puede que no esté completamente libre de defectos. Muchos de estos defectos se descubrirán durante la fase de transición.

La **fase de transición** cubre el periodo durante el cual el producto pasa de la versión beta a la versión final. Durante la versión beta un número reducido de usuarios prueba el sistema e informa de defectos y deficiencias. Esto hasta llegar a la versión final.

CAPÍTULO III.

DESARROLLO

3.1 FASE DE INICIO

En esta etapa, se expondrán los resultados obtenidos a través de los artefactos elaborados durante la primera fase del Proceso Unificado de Desarrollo de Software, donde se estableció un concepto inicial del sistema, esto se logra, creando un esquema general que permita la identificación de funcionalidades y requisitos para la elaboración de un conjunto de modelos preliminares que capturen la semántica y comportamiento del sistema, con el fin de lograr un entendimiento común entre los desarrolladores y los usuarios, así como planificar y poner en marcha el desarrollo del proyecto.

3.1.1 Planificación de la fase de inicio

Para esta fase, se realizó un análisis del negocio, con esto, se estudió inicialmente el contexto del sistema en relación a los procesos y actividades llevadas a cabo por las personas involucradas, obteniendo un modelo de negocio representado por medio de la notación de Eriksson y Penker. De igual modo se lograron obtener los casos de usos iniciales para la elaboración del modelo de casos de uso, se creó un modelo del dominio para comprender el problema planteado en su contexto. En la tabla 1 se presentan las actividades definidas y los artefactos obtenidos por cada una de ellas.

Tabla 1. Actividades y artefactos planificados para la fase de inicio.

Actividad	Artefactos
Contexto del sistema	Documento Visión. Estudio del contexto del sistema. Modelo del negocio.

Tabla 1. Continuación.

Actividad	Artefactos
	Modelo del dominio. Glosario de términos del modelo del dominio.
Riesgos del sistema	Lista de riesgos críticos.
Requisitos	Requisitos funcionales y no funcionales. Requisitos de software y hardware. Captura de requisitos como casos de usos Modelo de casos de uso. Descripción de casos de uso.
Análisis	Identificación de clases de análisis. Diagrama de clases de análisis. Diagrama de colaboración. Identificación de paquetes de análisis.

A continuación se describirán cada uno de los artefactos planificados para la fase de inicio.

3.1.2 Documento visión

Para tener una visión clara de las funciones básicas que tendría la aplicación *Web*, se estudió cual era el problema que se intentaría resolver, definiendo algunos términos que ayudaron a comprender su contexto así como detallar cuáles serían los roles y las responsabilidades necesarias para el desarrollo la aplicación. Un rol define el comportamiento y responsabilidades de un individuo, o de un grupo de individuos trabajando juntos como un equipo. Una persona puede desempeñar diversos roles, así como un mismo rol puede ser representado por varias personas. Las responsabilidades de un rol son tanto el llevar a cabo un conjunto de actividades como ser el dueño de un conjunto de artefactos.

En la tabla 2 se describen cada uno de los roles necesarios para el desarrollo del proyecto:

Tabla 2. Roles necesarios para el desarrollo del proyecto.

Rol	Responsabilidad
Analista de sistemas	Capturar, especificar y validar requisitos, interactuando con el cliente y los usuarios mediante entrevistas. Elaborar el Modelo de Análisis y Diseño. Colaborar en la elaboración de las pruebas funcionales y el modelo de datos. Controlar los requisitos y casos de uso para verificar la funcionalidad del sistema y sus limitaciones.
Analista de procesos de negocio	Es responsable de definir la arquitectura, definir los casos de uso, actores y la interacción entre ellos.
Programador	Construir el software. Colaborar en la elaboración de las pruebas funcionales, el modelo de datos y en las validaciones con el usuario.
Especialista en pruebas (<i>tester</i>)	Es responsable de realizar las pruebas del software cada vez que se realice una iteración y las pruebas finales anotando los resultados de esa comprobación.

3.1.3 Estudio del contexto del sistema.

En esta etapa se realizó un análisis detallado de las actividades más relevantes realizadas por el Departamento de Automatización y Control, el personal de Planta y de la Dirección de Producción Industrial con relación al Sistema de Interrupciones, lo cual ayudó a determinar cuáles eran los procesos más importantes que debía realizar el software, para satisfacer las necesidades de todo el personal involucrado y

establecer un conjunto de modelos que describen de la mejor manera el contexto en el que se desarrollará la solución de software.

3.1.3.1 Modelado del negocio

Para conseguir sus objetivos, una empresa organiza sus actividades por medio de un conjunto de procesos de negocio. Cada uno de ellos se caracteriza por una colección de datos que son producidos y manipulados mediante un conjunto de tareas, en las que ciertos agentes (por ejemplo, trabajadores o departamentos) participan de acuerdo a un flujo de trabajo determinado.

Este artefacto permitió representar todos los aspectos que describen dónde funcionará la Aplicación *Web* para la Gestión del SI dentro de la estructura organizacional de SIDOR y de las actividades habituales del personal. Además, se obtuvieron los procesos llevados a cabo en el Departamento de Automatización y Control, resaltando aquellos que involucren el objetivo del proyecto y detallándolos que resulten de más bajo nivel según la jerarquía de procesos.

3.1.3.2 Modelo de fines del negocio

El fin representa la más elevada expresión de lo que una empresa persigue, siendo genérico y, a veces, difuso.

Los objetivos son producto del análisis de la situación presente y futura de toda organización, constituyen los fines hacia los cuales se dirige una actividad y establecen resultados.

Para realizar el modelado de negocios de los procesos vinculador a la Gestión del Sistemas de Interrupciones de la Siderúrgica del Orinoco se debe realizar un estudio de la misión, propósitos y objetivos del Departamento de Automatización y Control, con la finalidad de capturar los procesos de negocio al más alto nivel.

A continuación en la figura 5 se describe los fines del Departamento de Automatización y Control de SIDOR:

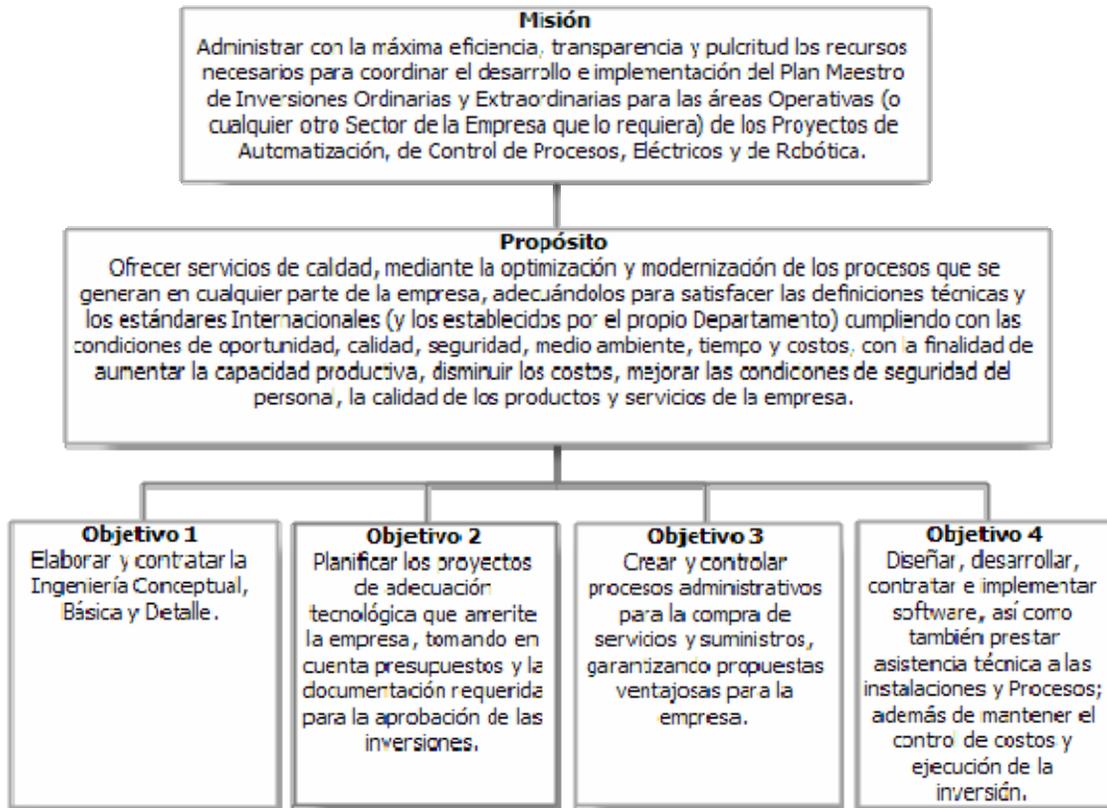


Figura5. Diagrama de Fines del Negocio

3.1.3.3 Modelo de Michael Porter aplicado al negocio

Para conseguir sus objetivos, una empresa organiza sus actividades por medio de un conjunto de procesos de negocio. Los procesos obtenidos dentro del Departamento de Automatización y Control son:

- 1) Elaborar y contratar la ingeniería conceptual, básica y detalle.
- 2) Planificar los proyectos de adecuación tecnológica que amerite la empresa, tomando en cuenta presupuestos y la documentación requerida para la

aprobación de las inversiones.

- 3) Crear y controlar procesos administrativos para la compra de servicios y suministros, garantizando propuestas ventajosas para la empresa.
- 4) Diseñar, desarrollar, contratar e implementar software, así como también prestar asistencia técnica a las instalaciones y procesos; además de mantener el control de costos y ejecución de la inversión.

En la figura 6, se muestra la cadena de valores aplicado a los procesos antes mencionados.

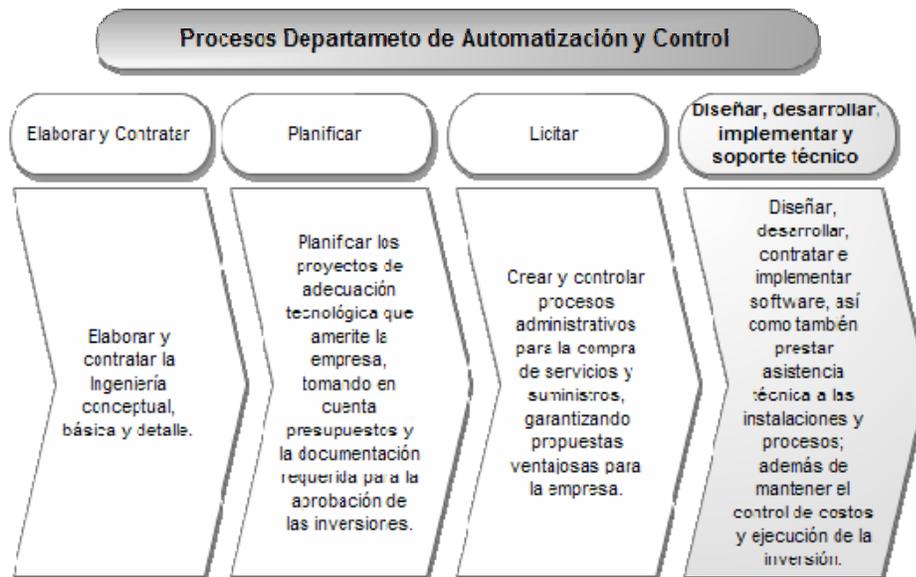


Figura6. Cadena de valores de los procesos llevados a cabo en el Departamento de Automatización y Control.

3.1.3.4 Modelo de jerarquía de procesos aplicado a los procesos de más alto nivel del negocio.

De los procesos de alto nivel llevados a cabo en el Departamento de Automatización y Control se tomará en cuenta el proceso: “Diseñar, desarrollar, contratar e implementar software, así como prestar asistencia técnica a las instalaciones y procesos”, ya que en él se encuentra inmersa la Gestión del SI y más específicamente

la Gestión de Catálogos, Productividad Programada, Clasificación de Demoras, de Usuarios y de rotación de cuadrillas por turnos, objeto de desarrollo en la aplicación *Web*. A este proceso se le aplico la regla de descomposición usando Diagramas de Jerarquía de Procesos de UML Business, con el fin de obtener los procesos de negocio de más bajo nivel:

PROCESO 1. Diseñar, desarrollar, contratar e implementar software, así como prestar asistencia técnica a las instalaciones y procesos, además de mantener el control de costos y ejecución de la inversión.

Proceso 1.1 Gestionar SI.

Proceso 1.1.1 Gestionar catálogo.

Proceso 1.1.1.1 Estudiar solicitud de industrial.

Proceso 1.1.1.2 Administrarcatálogo.

Proceso 1.1.2 Gestionar productividad programada.

Proceso 1.1.2.1 Estudiar solicitud de industrial.

Proceso 1.1.2.2 Administrarproductividad programada.

Proceso 1.1.3 Clasificar demoras.

Proceso 1.1.3.1 Estudiar solicitud de industrial.

Proceso 1.1.3.2 Desglosar y Justificar demora.

Proceso 1.1.4 Gestionarusuarios.

Proceso 1.1.4.1 Estudiar solicitud de industrial.

Proceso 1.1.4.2 Administrar usuarios.

Proceso 1.1.5 Gestionar patrones.

Proceso 1.1.5.1 Estudiar solicitud de industrial.

Proceso 1.1.5.2 Establecer cuadrillas y rotación mensual.

En la figura 7, presentada a continuación se muestra jerarquía de los procesos llevados a cabo en el Departamento de Automatización y Control que guardan relación con la Gestión del SI.

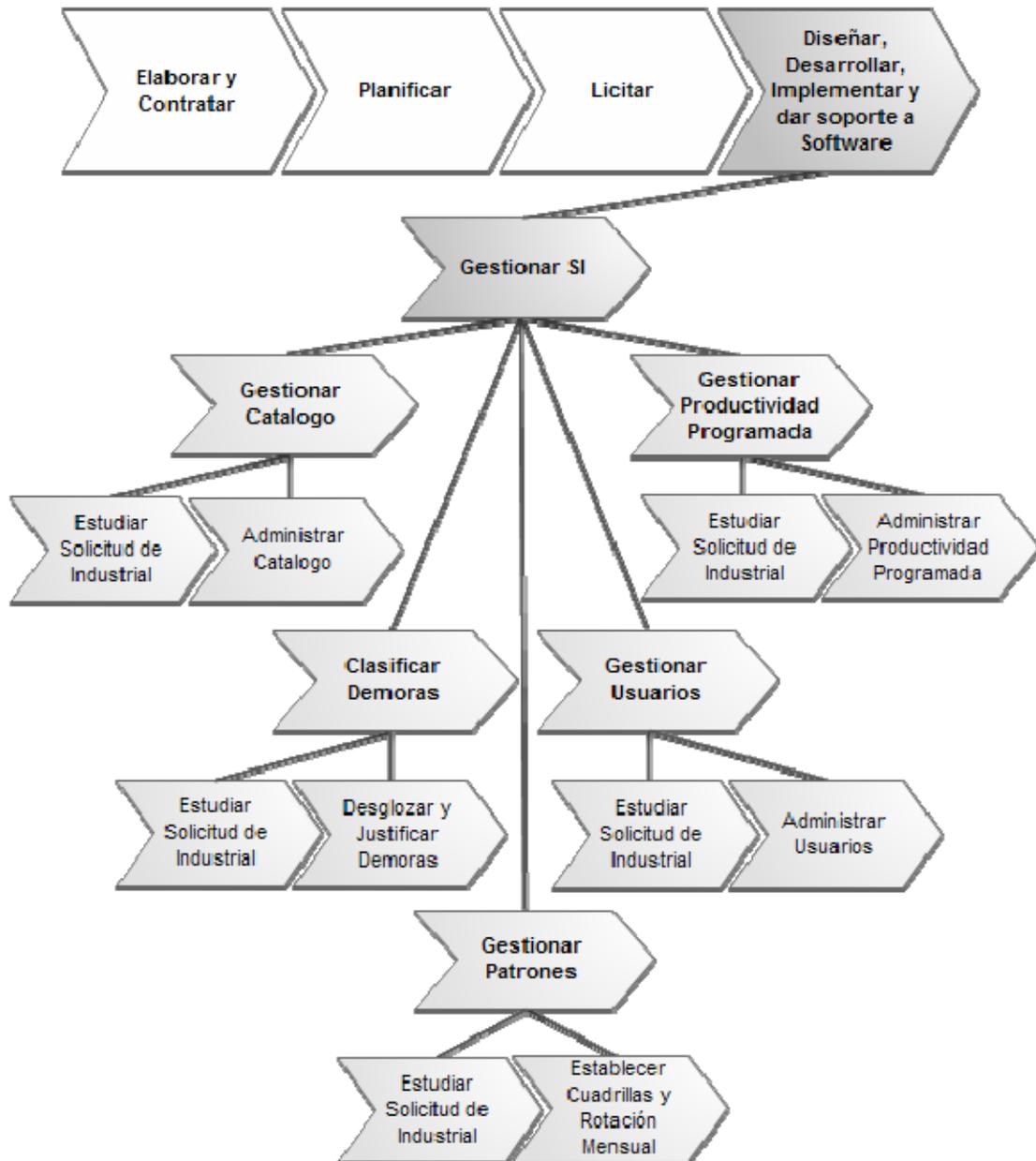


Figura7. Diagrama de jerarquía de los procesos llevados a cabo en el en el Departamento de Automatización y Control.

3.1.3.5 Notación de Eriksson y Penker aplicado a los procesos de alto nivel del negocio.

Para el modelado de Procesos de Negocio se tomaron en estudio el proceso Gestionar SI, que incluye Gestionar Catálogos, Gestionar Productividad Programada, Clasificar Demoras, Gestionar Usuario y Gestionar Patrones. Se le aplicó Notación de Eriksson y Penker para obtener el modelo de procesos de negocio.

El modelo preliminar del negocio permite capturar los eventos, las entradas, los recursos y las salidas más importantes vinculadas con el proceso de negocio, por tal motivo se presenta en la figura 8 el modelo del proceso de alto nivel Gestionar SI.

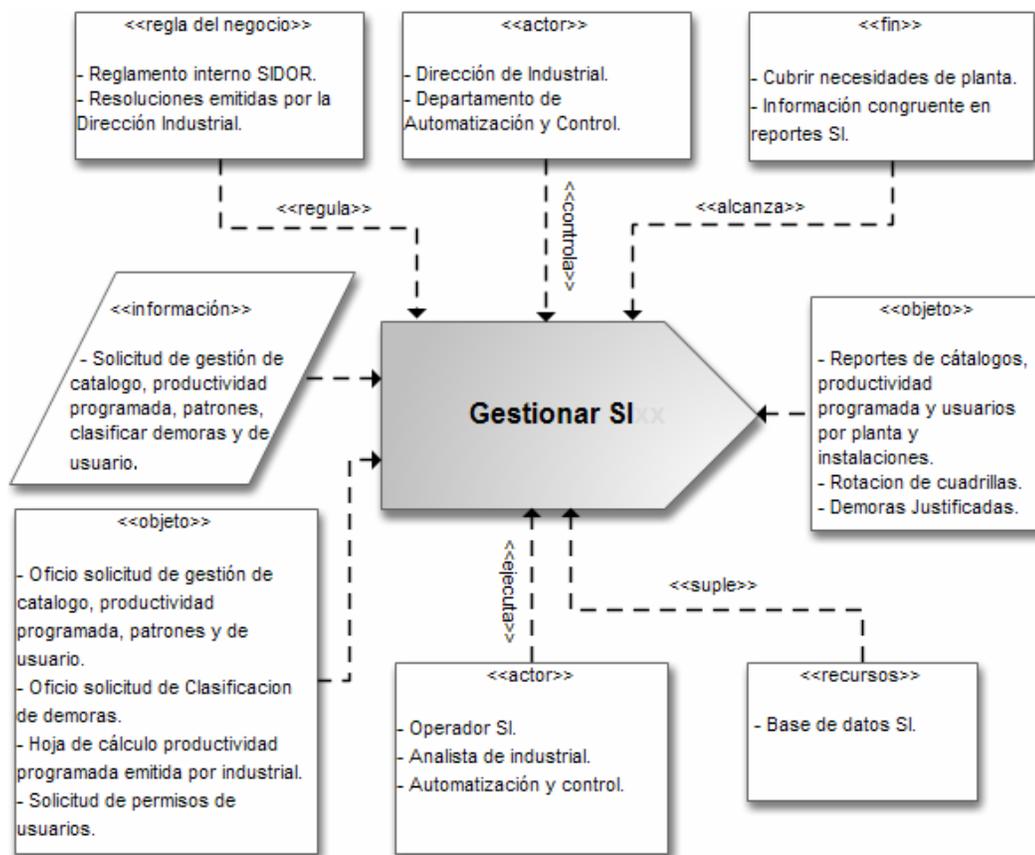


Figura8. Diagrama de proceso de negocio Gestionar SI.

A continuación en la figura 9, se presenta el modelo del negocio que describe los procesos asociados al Departamento de Automatización y Control así como a la Dirección de Producción Industrial en relación con el Sistema de Interrupciones de SIDOR. El modelo se representó por medio de una cadena de valor, identificándose las actividades principales, como Gestionar Catálogo, Productividad Programada, Clasificación de Demoras, Gestión de Usuarios y Gestión de Patrones. También la actividad secundaria de Planificación. El resultado de un proceso sirve de insumo al siguiente proceso en la realización de sus actividades.

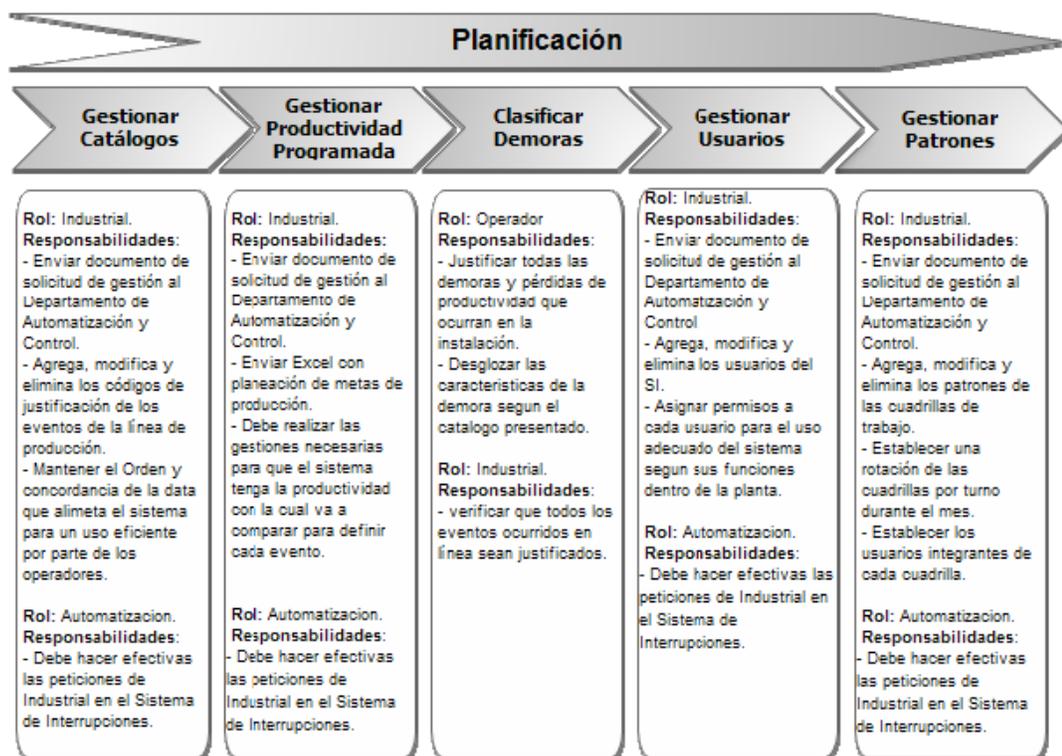


Figura9. Modelo del negocio del proceso Gestionar SI.

3.1.3.6 Modelo del dominio

Un modelo de dominio es un artefacto, presentado como uno o más diagramas de clases y que contiene, no conceptos propios de un sistema de software sino de la

propia realidad física, permitiendo capturar y expresar el entendimiento ganado en un área bajo análisis como paso previo al diseño de la aplicación para la gestión del SI.

En otras palabras, este modelo captura los tipos más importantes de clases u objetos en el contexto del sistema estudiado.

Los objetos del dominio o clases representan las cosas que existen o los eventos que suceden en el entorno en el que trabaja el SI, estas clases se relacionan a través de asociaciones, agregaciones y composiciones para modelar el desenvolvimiento de las actividades; ayudando a definir qué deberá hacer la aplicación para resolver el problema y no cómo lo hará.

En el dominio del Sistema de Interrupciones de SIDOR, se observa que existe un catálogo y que este a su vez tiene una estructura ya elaborada por el personal de Industrial, como lo son Tipo, Zona, Sistema, Subsistema, Evento y Subevento, donde se almacena la información básica para describir las características de un evento que ocurre en la planta, con el fin de justificar aquellos que sean demoras o pérdidas, en relación a una productividad programa que establece las metas de producción mensuales.

Los usuarios son todo aquel personal que labora en una determinada línea de producción y está encargado de ejercer alguna función dentro del SI según el perfil que mantenga, lo que está aunado a una serie de permisos que le permiten acceder solo a la información relevante de sus funciones, lo cual está estipulado en la estructura de funcionamiento para justificar los eventos de forma manual.

Los eventos de demoras son los que ocurren en una instalación de la planta y son atendidos por el personal perteneciente a la cuadrilla de guardia según los patrones de rotación de turnos establecidos mensualmente por Industrial, donde solo algunos miembros de esas cuadrillas tienen la labor de recaudar la información referente a los

contratiempos que surgen, a manera de llevar un control completo del personal y de todo lo que ocurre en las instalaciones.

A continuación en la figura 10 se muestra el modelo de dominio, donde se puede evidenciar lo anteriormente descrito.

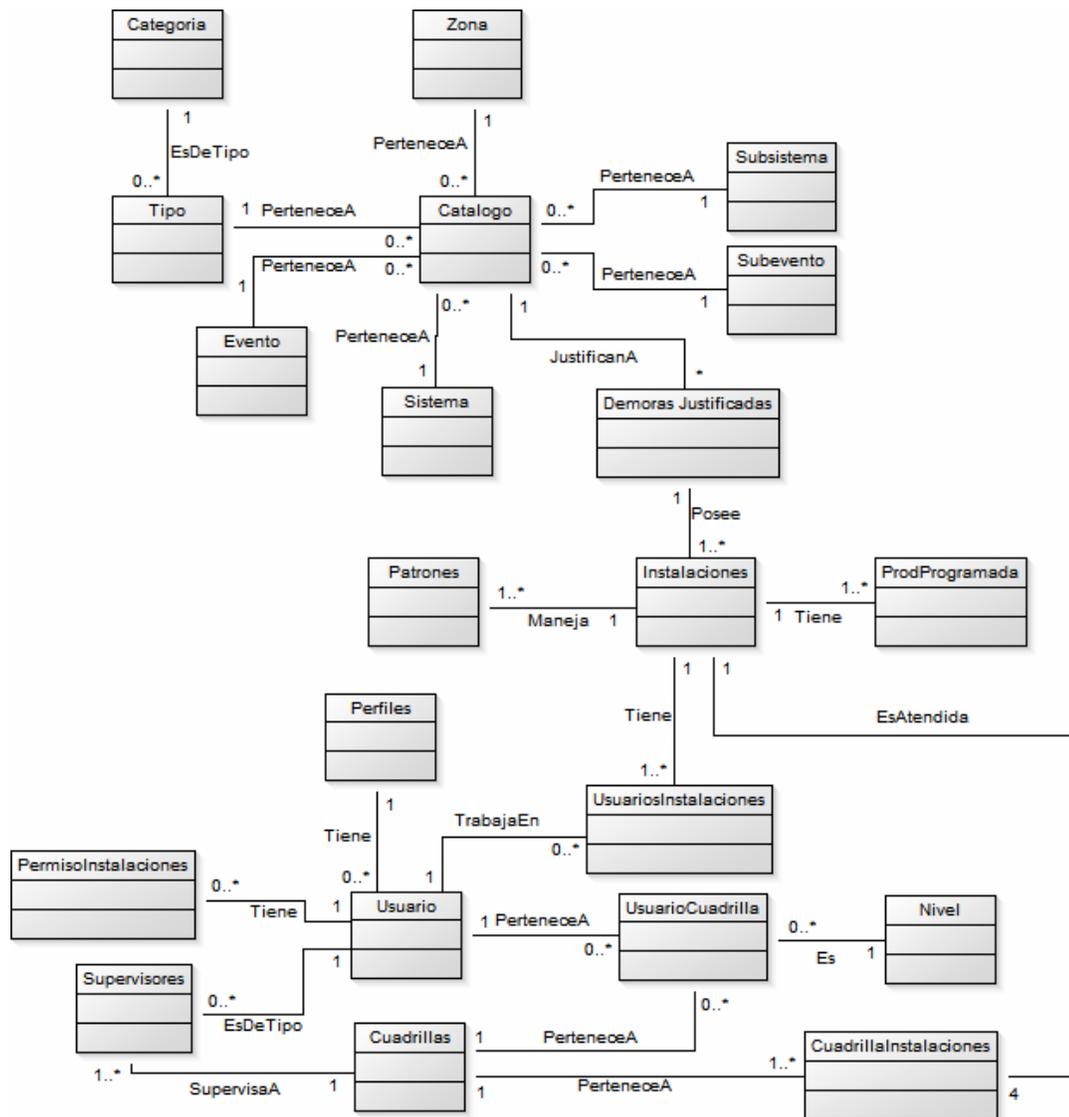


Figura 10. Modelo de Dominio para el Sistema de Interrupciones de SIDOR.

A continuación, en la tabla 3 se presenta una lista de términos empleados en el dominio del sistema propuesto, donde se definen cada una de las entidades involucradas en el modelo anteriormente mostrado.

Tabla 3. Glosario de términos del modelo del dominio del sistema propuesto.

Término	Definición
Catálogo	Son arreglos que permiten identificar en el SI las características de una demora.
Categoría	Clase en la que se encuentra la falla, como por ejemplo operativa o no operativa.
Tipo	Tipos con los que se pueden clasificar los registros.
Zona	Son las respectivas zonas de la planta.
Evento	Es una pequeña explicación sobre el suceso que ocurrió en la planta.
Sistema	Son los equipos de la línea de producción
Subsistema	Son cada una de las piezas que componen al equipo que este fallando.
Subevento	Es una información más detallada del evento que se generó en la planta.
Demoras Justificadas	Representa los detalles del porqué, como y donde ocurre, una pérdida o disminución de capacidad de producción contra la meta establecida.
Patrones	Establece la rotación de las cuadrillas de trabajo durante un lapso de tiempo.
Instalaciones	Líneas de Producción pertenecientes a una planta.
ProdProgramada	Se refiere a las metas de producción mensual trazadas por la gerencia de una planta.
Usuario	Personal encargado de mantener un control en las fallas de la línea de producción y de gestión del SI.

Tabla 3. Continuación.

Término	Definición
PermisoInstalaciones	Permisos de gestión dentro de una línea de producción.
UsuariosInstalaciones	Personal que laboran en una determinada instalación.
CuadrillaInstalaciones	Cuadrillas encargas de atender una línea de producción.
Cuadrillas	Grupo de trabajo encargado de la línea de producción durante un lapso de tiempo.
UsuarioCuadrilla	Personal perteneciente a una cuadrilla.
Supervisores	Personal encargado de inspeccionar el trabajo de las cuadrillas en determinada instalación.
Nivel	Nivel de jerarquía de un operador
Perfiles	Clasificación del personal que labora dentro de la planta y que indica sus responsabilidades para con la misma.

3.1.4Riesgos del sistema

Los riesgos dirigen la viabilidad del sistema en el Proceso Unificado de Desarrollo de Software; ya que existe la probabilidad de que el proyecto se vea afectado en su desarrollo o comportamiento futuro. En tal sentido, los riesgos deben ser ordenados por el nivel de relevancia o por su influencia en el desarrollo.

Una vez que se han identificado los riesgos, pueden ser tratados de diferentes maneras. Algunos pueden ser evitados llevando a cabo acciones como replanificar el proyecto, pueden ser también evitados haciendo cambios en los requisitos, otros podrían ser detectados y aislados de otras partes del proyecto, a manera de afectar solo una parte.

Se estableció un Plan de Administración de Riesgos que permite evitar o manejar de forma adecuada los riesgos que pueden presentarse a lo largo del desarrollo de la Aplicación *Web* para la Gestión del Sistema de Interrupciones de la Siderúrgica del Orinoco Alfredo Maneiro (SIDOR). Para ello se realizó un análisis de los riesgos asociados al proyecto y se determinaron estrategias para su prevención, análisis, jerarquización y mitigación.

Como se explicaba anteriormente, el estudio realizado arrojó una serie de riesgos críticos, que son de especial interés para el desarrollo de la Aplicación *Web*, los cuales se listan a continuación:

- Desconocimiento del ámbito del sistema: el no conocer el ámbito del sistema en base al cual se está trabajando, significa un riesgo crítico que debe ser mitigado en la fase de inicio. El diseñador debe tener conocimiento del contexto y entender sus aspectos fundamentales.
- Requisitos mal definidos: no disponer de requisitos claros y que abarquen todas las necesidades de los usuarios, puede significar un riesgo para el sistema. Se deben recolectar exhaustivamente todas las necesidades de los usuarios.
- Falta de robustez de la arquitectura: el sistema debe poseer una arquitectura robusta que le permita adaptarse a los cambios y al mantenimiento de manera elegante y poco traumática.
- Diseño no adecuado de la base de datos: el mal diseño de la base de datos es un riesgo crítico, la base de datos del sistema debe estar diseñada de tal manera que mantenga la integridad referencial, que no sea redundante y que garantice el acceso a los datos necesarios, permitiendo al sistema llevar a cabo sus funciones en un tiempo prudente.

Para estos riesgos, se planteó un plan de prevención y uno de contingencia como forma de evitar los posibles inconvenientes y de mitigar los que se pudieran presentar. El plan de riesgos establecido se ve enunciado a continuación en la tabla 4:

Tabla 4. Plan de prevención y contingencia para los riesgos más predominantes en el desarrollo de la aplicación *Web*.

Riesgos	Probabilidad	Impacto	Plan de Prevención	Plan de Contingencia
Desconocimiento del ámbito del sistema.	40%	Crítico	Aplicar un exhaustivo levantamiento de información y detallado modelado del sistema.	Chequear los diagramas realizados y aplicar las correcciones pertinentes.
Requisitos mal definidos.	30%	Crítico	Requisitos mal definidos.	Aplicar entrevistas no estructuradas para aclarar dudas sobre los requerimientos.
Falta de robustez de la arquitectura.	40%	Crítico	Realizar un diseño modular.	Revisar los artefactos generados en el proceso de desarrollo.
Diseño no adecuado de la base de datos.	20%	Crítico	Recaudar un alto porcentaje de información para el diseño de la misma.	Revisar la información recaudada y de los requerimientos de los usuarios.

3.1.5 Flujo de trabajo (*workflow*) de requisitos

Este artefacto permitió determinar los requerimientos que debe abarcar la Aplicación *Web* Para la Gestión del Sistema de Interrupciones de la Siderúrgica del Orinoco Alfredo Maneiro (SIDOR). Para ello se listan los requerimientos funcionales y no funcionales, se hace una breve descripción de cada uno de ellos y de los perfiles de usuario que manejará la aplicación, así como los requerimientos necesarios para determinar los casos de usos iniciales.

3.1.5.1 Requerimientos funcionales

Para capturar los requisitos de manera eficaz, se necesitó un artefacto para obtener una visión suficientemente buena del sistema, conocido como el modelo de casos de uso el cual se describirá más adelante. Por lo que se puede iniciar mencionado a continuación en la tabla 5 los requisitos funcionales para la aplicación:

Tabla 5. Requisitos funcionales en la fase de inicio.

Requerimiento	Necesidad	Estado
Gestionar los catálogos de desglose de demoras de acuerdo a las resoluciones emitidas por la Dirección de Producción Industrial.	E	A
Gestionar los planes mensuales de producción llevados a cabo por el personal de industrial.	E	A
Gestionar los trámites y autorizaciones de usuarios necesarias para la seguridad y mantenimiento eficiente del Sistema de Interrupciones de SIDOR.	E	A
Clasificar Demoras	E	A
Dinamizar el trabajo de gestión global del SI entre las distintas plantas e instalaciones.	E	A

En la tabla anterior la necesidad del requerimiento puede ser:

- E, cuando el requerimiento es esencial o indispensable para la aceptación del sistema;
- C, cuando el requerimiento se considera condicional o bien deseable pero no indispensable para la aceptación del sistema
- O, si el requerimiento es opcional sugerido por el usuario.

En cuanto al estado:

- C: Cerrado, cuando se considera que el requerimiento se encuentra lo suficientemente claro y definido, como para pasar a la siguiente etapa del análisis.
- A: Abierto, cuando se considera que el requerimiento está pendiente de mayor definición.

3.1.5.2 Requerimientos no funcionales

Los requisitos no funcionales se encuentran expresados a continuación en la tabla 6:

Tabla 6. Requerimientos no funcionales en la fase de inicio.

Requerimiento	Necesidad	Estado
Eficiencia en el tiempo de respuesta esperado para la ejecución de los procesos en el sistema.	E	A
Interfaz amigable para el usuario	E	A
Confianza en el sistema por parte del usuario.	E	A
Consistencia de la data.	E	A
El sistema debe identificar el perfil del usuario para asignarle los permisos correspondientes.	E	A

Tabla 6. Continuación.

Requerimiento	Necesidad	Estado
El desarrollo debe realizarse, basado en tecnología .NET, y bases de datos SQL Server 2000.	E	A
Acceso al sistema por medio de la Intranet de la empresa y conectividad para manejo de múltiples usuarios a la vez.	E	A
Utilizar metodologías y estándares para el análisis, diseño, desarrollo e implementación del software.	E	A
Desarrollar un manual de usuario que sirva de guía para usuarios con dificultades ante la aplicación.	E	A
Requisitos de Software		
<ul style="list-style-type: none"> - Navegador de Internet: <i>Internet Explorer 6.0</i>. - <i>Microsoft Visual Studio.NET</i> - Manejador de Base de Datos <i>MicrosoftSQLServer 2000</i>. - <i>Internet Information Server 5.1</i> como servidor <i>Web</i>. 	E	C
Requerimientos de Hardware		
Servidor:		
<ul style="list-style-type: none"> - Procesador x86 o equivalente a 1GHz o más. - 512 MB de memoria de acceso aleatorio (RAM). - Disco Duro de 20 Gb. - Monitor con una resolución máxima de 1280x1024. - Interfaz de red <i>Ethernet</i>. 	E	C
Clientes:		

-
- Procesador x86 o equivalente a 750 MHz o más.
 - 256 MB de memoria de acceso aleatorio (RAM).
-

Tabla 6. Continuación.

Requerimiento	Necesidad	Estado
<ul style="list-style-type: none"> - Monitor a color con una resolución máxima de 1024x768. - Interfaz de red <i>Ethernet</i>. 	E	C

3.1.5.3 Captura de requisitos como casos de uso

El modelo de casos de uso proporciona la entrada fundamental para el análisis, el diseño y las pruebas. Este modelo contiene ciertos artefactos que permiten describir el sistema, entre ellos se encuentran los actores, casos de uso y sus relaciones, a fin de presentar al cliente un modelo donde se represente los requerimientos del software.

Con la aplicación, interactuarán un conjunto de usuarios, y se pueden clasificar según las acciones o perfiles que puedan desempeñar dentro de la misma.

En los casos de usos uno de los elementos que los conforman son los actores y de las acciones que ejecuten los usuarios, se definirán estos:

Clasificador: usuario que tendrá la tarea de justificar todas las demoras y pérdidas de productividad que ocurran en las distintas líneas de producción.

Industrial: usuario que tendrá la responsabilidad de gestionar el sistema de interrupciones. Este se encargará de cargar los datos que el sistema necesite, tanto para la visualización de la cuadrilla que corresponde en el turno, como la carga de los catálogos de clasificación y de la productividad programada.

Automatización: actor encargado de brindar todo el apoyo y soporte técnico al resto de usuarios del Sistema de Interrupciones.

Son superusuarios en el SI por lo tanto no poseen ningún tipo de restricción a la hora de utilizarlo.

SI: actor que tendrá la responsabilidad de actualizar los datos del estado de la planta. Más que un usuario común y corriente estará relacionado con el tiempo, debido a que cada cierto tiempo estará actualizando la información de la velocidad de las líneas de producción con relación a la planta a la que pertenecen.

Luego de haber identificado los actores principales del sistema y de establecer los requisitos, se procede a identificar los casos de uso y las relaciones que existen entre ellos.

A continuación, en la tabla 7 se presentan los casos de usos identificados en esta fase para el desarrollo de la aplicación *Web* para la gestión del sistema interrupciones de SIDOR, acompañados por una breve descripción general para cada uno.

Tabla 7. Casos de uso de la Aplicación *Web* para la Gestión del SI de SIDOR en la fase de inicio.

Casos de usos	Descripción
Gestionar productividad	Agrega, modifica y elimina la productividad programada del mes.
Clasificar demoras	Permite justificar por medio de los catálogos, según los eventos ocurridos en la planta.
Gestionar catálogo	Agrega, modifica y elimina los códigos de justificación de los eventos de la línea de producción.
Gestionar usuarios	Agrega, modifica y elimina usuarios de la aplicación, así como asignar permisos de gestión a cada uno de ellos.

Los casos de uso guían la arquitectura del sistema y la arquitectura influye en la selección de los casos de uso. Por tanto, la arquitectura del sistema como los casos de uso maduran según avanza el ciclo de desarrollo.

3.1.5.4 Modelo de casos de uso

Un caso de uso es un fragmento de funcionalidad del sistema que proporciona al usuario un resultado importante, por lo que todos los casos de uso juntos constituyen el modelo que describe la funcionalidad total del sistema. En la figura 11, se puede observar que el actor Industrial inicia el sistema almacenando, eliminando o modificando información referente a catálogos, productividad y usuarios, la cual será suministrada a la base de datos para alimentar el sistema. Posteriormente el actor Clasificador podrá realizar las justificaciones en los eventos de demoras que lo requieran. Así como también se podrá consultar contenido de toda la información involucrada en este proceso por los usuarios identificados. A continuación, se muestra el diagrama de casos de uso resultante:

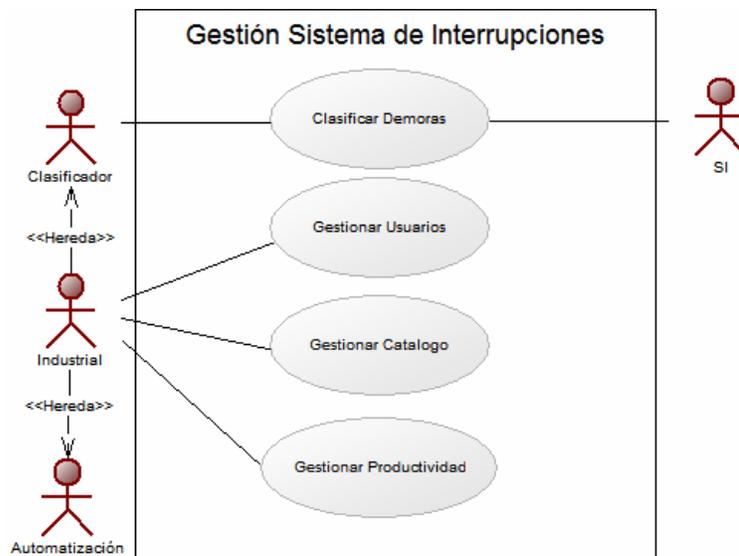


Figura 11. Diagrama de caso de uso de la Aplicación *Web* para la Gestión del SI de SIDOR de la fase de inicio.

A continuación, en las tablas 8, 9, 10 y 11, se presenta la descripción en detalle de los casos de uso más importantes:

Tabla 8. Caso de Uso Clasificar Demoras.

Clasificar Demoras	
Actor Principal:	Clasificador.
Actor Secundario:	SI.
Personal involucrado e intereses:	
Industrial:	Desea que todos los eventos de la línea sean justificadas.
Jefe de Planta:	Desea que todos los eventos de la línea sean justificadas.
Supervisor:	Desea que todos los eventos de la línea sean justificadas.
Operador:	Desea que todos los eventos de la línea sean justificadas.
Pre-Condiciones:	El usuario se identifica y autentica.
Post-Condiciones:	El sistema registra la información cargada.
Escenario principal de éxito:	
1.- El sistema consulta los eventos ocurridos en planta en la <i>DatawareHouse</i> del SI	
2.- El sistema lista los eventos para los parámetros de consulta.	
3.- El usuario selecciona el evento a clasificar.	
4.- El sistema muestra la información del evento y las características de la clasificación.	
5.- El usuario selecciona el tipo de evento que sucedió.	
6.- El usuario selecciona la zona donde ocurrió el evento.	
7.- El usuario selecciona el sistema que fallo.	
8.- El usuario selecciona el subsistema que fallo.	
9.- El usuario selecciona el evento que causó la falla.	
10.- El usuario decide guardar la justificación.	
11.- El sistema registra la justificación.	
12.- El sistema auto clasifica el módulo si es una pérdida.	
Flujos Alternativos:	

10.a.- El usuario decide particionar el evento.

1.- El evento es una demora.

Tabla 8. Continuación.

1.- El usuario reparte la duración de la demora.

2.- El usuario guarda la información.

3.- El sistema va al paso 11.

2.- El evento es una pérdida.

1.- El usuario reparte las toneladas pérdidas de la pérdida de productividad.

2.- El usuario guarda la información.

3.- El sistema va al paso 11.

10.-a El usuario ha escogido para clasificar el evento un código de "No Existe el Sistema", "No Existe el Subsistema", "No Existe el Evento" o "No Existe el Subevento".

1.- El sistema envía un correo al personal de industrial para que cree los nuevos códigos de clasificación.

2.- El sistema va al paso 11.

12.a.- El sistema detecta que se debe clasificar el módulo manualmente.

1.- El sistema va al paso 3.

Tabla 9. Caso de Uso Gestionar Productividad.

Gestionar Productividad

Actor Principal: Industrial

Personal involucrado e intereses:

Industrial: Desea que en el sistema tenga la productividad con la cual va a comparar para definir cada evento.

Jefe de Planta: Desea tener a la vista la producción de cada turno en el sistema.

Supervisor: Desea tener a la vista la producción de cada turno en el sistema.

Operador: Desea tener a la vista la producción de cada turno en el sistema.

Pre-Condiciones: El usuario se identifica y autentica.

Tabla 9. Continuación.

Post-Condiciones: Industrial ha gestionado satisfactoriamente la productividad programada.

Escenario principal de éxito:

1.- El sistema muestra todas las productividades vigentes y no vigentes.2.- El usuario selecciona insertar una productividad.

3.- El usuario selecciona el día y turno.

4.- El usuario carga la productividad y el tiempo efectivo del módulo.

5.- El usuario decide guardar la productividad.

6.- El sistema registra la productividad.

7.- El sistema revisa que ese día no se encuentre ya cargado.

8.- El sistema revisa que ese día no sea menor o igual que la fecha actual.

9.- El sistema va al paso 1.

Flujos Alternativos:

2.a.- El usuario selecciona modificar una productividad programada.

1.- El usuario decide cambiar el día programado.

1.- El usuario selecciona el nuevo día.

2.- El usuario decide guardar la modificación.

3.- El sistema va al paso 7.

2.- El usuario decide eliminar el día programado.

5.a.- El usuario decide cancelar la operación.

Tabla 10. Gestionar Catálogos.

Gestionar Catálogos

Actor Principal: Industrial

Actor Secundario: Jefe de Planta

Personal involucrado e intereses:

Industrial: Desea que se tengan los códigos y descripción completa.

Tabla 10. Continuación.

Jefe de Planta: Desea que en el sistema tenga los códigos y descripción completa con la cual el operador va a definir cada evento.

Operador: Desea tener a la vista todos los catálogos de desglose de demoras con información detallada para definir los eventos.

Pre-Condiciones: El usuario se identifica y autentica.

Post-Condiciones: Industrial ha gestionado satisfactoriamente los catálogos.

Escenario principal de éxito:

1.- El sistema muestra todos los catálogos vigentes y no vigentes.

2.- El usuario selecciona insertar un catálogo según la categoría deseada.

3.- El usuario selecciona la fecha de vigencia y fin de vigencia.

4.- El usuario carga los detalles del catálogo y el tiempo efectivo del módulo.

5.- El usuario decide guardar el catálogo.

6.- El sistema registra el catálogo.

7.- El sistema revisa que el código de catálogo no se encuentre ya cargado.

8.- El sistema revisa que ese día no sea menor o igual que la fecha actual.

9.- El sistema va al paso 1.

Flujos Alternativos:

2.a.- El usuario selecciona modificar un catálogo.

1.- El usuario decide cambiar el día programado.

1.- El usuario selecciona el nuevo día.

2.- El usuario decide guardar la modificación.

3.- El sistema va al paso 7.

2.- El usuario decide eliminar el catálogo.

5.a.- El usuario decide cancelar la operación.

Tabla 11. Caso de Uso Gestionar Usuarios.

Gestionar Usuarios	
Actor Principal:	Industrial
Personal involucrado e intereses:	
Industrial:	Desea tener los usuarios necesarios para la distribución de las labores, establecer responsables y permisos de seguridad.
Jefe de Planta:	Desea una cuenta que organice las funcionalidades que necesita.
Supervisor:	Desea una cuenta que organice las funcionalidades que necesita.
Operador:	Desea una cuenta que organice las funcionalidades que necesita.
Pre-Condiciones:	El usuario se identifica y autentica.
Post-Condiciones:	Industrial ha gestionado satisfactoriamente un usuario.
Escenario principal de éxito:	
1.- El sistema muestra todos los usuarios vigentes y no vigentes.	
2.- El usuario selecciona insertar un nuevo registro.	
3.- El usuario carga los datos personales, permisos, el periodo de vigencia y fin de vigencia.	
4.- El usuario decide guardar la gestión.	
5.- El sistema revisa que el usuario no se encuentre ya cargado y no posea históricos.	
6.- El sistema revisa que la fecha de vigencia no sea menor o igual que la fecha actual.	
7.- El sistema registra el nuevo usuario y va al paso 1.	
Flujos Alternativos:	
2.a.- El usuario selecciona modificar un registro.	
1.- El usuario decide cambiar los datos personal, fecha de inicio o fin de vigencia.	
1.- El usuario selecciona la nueva fecha.	
2.- El usuario decide guardar la modificación.	
3.- El sistema va al paso 6.	
2.- El usuario decide eliminar el registro.	
4.a.- El usuario decide cancelar la operación.	

3.1.6 Flujo de trabajo (*workflow*) de análisis

Durante este flujo se analizan los requisitos, depurándolos y estructurándolos en un modelo preliminar de clases de análisis, con el que se obtiene un esbozo o primera

impresión del modelo de diseño. El propósito fundamental del análisis es resolver los casos de uso analizando los requisitos con mayor profundidad.

3.1.6.1 Modelo de análisis

Un modelo de análisis tiene como propósito refinar los casos de uso con más detalle y establecer la asignación inicial de funcionalidad del sistema a un conjunto de objetos que proporcionan el comportamiento. El modelo de análisis permite a los desarrolladores comprender de manera más precisa los casos de uso descritos en el flujo de trabajo de los requisitos, refinándolos en forma de colaboraciones entre clasificadores conceptuales, que da como resultado una mejor forma de estructurar el funcionamiento interno del sistema.

Identificación de clases de análisis: en esta se especifican las clases de control, entidad, e interfaz necesarias para realizar los casos de uso y esbozar sus nombres, responsabilidades, atributos y relaciones. Las clases de análisis se identifican a través de tres estereotipos básicos: clases de interfaz, control y entidad, cada estereotipo implica una semántica específica, lo cual constituye un método potente y consistente de identificar y describir las clases de análisis y contribuye a la creación de un modelo de objetos y una arquitectura robusta.

Clases de interfaz: se utilizan para modelar la interacción entre el sistema y sus actores, esta interacción a menudo implica recibir información y peticiones de (y hacia) los usuarios y los sistemas externos. Las clases de interfaz permiten modelar las partes del sistema que dependen de sus actores, lo cual implica que clarifican y reúnen los requisitos en los límites del sistema. Por tanto, un cambio en un interfaz de usuario o en un interfaz de comunicaciones queda normalmente aislado en una o más clases de interfaz.

A continuación, en la tabla 12 se muestran las clases de interfaz del sistema.

Tabla 12. Clases de interfaz del sistema en la fase de inicio.

Clases	Definición
IU Principal SI	Clase de interfaz que permite a los usuarios interactuar con el sistema y llevar a cabo todas las operaciones asociados a la gestión del Sistema de Interrupciones.
IU Clasificación de Eventos	Se compone de varias zonas de entrada de datos en el cual el usuario seleccionara de una lista de códigos los más indicados y el cual describe el evento en la línea.
IU Productividad	Permite observar todas aquellas productividades que han sido cargados por el usuario, y cuales están aún vigentes o no, estos podrán modificar los datos para aquellos en donde se les permita.
IU Productividad Programada	Permite a los usuarios el registro, consulta y modificación de la productividad programada para una línea de producción en la cual labora y este asignado en el Sistema de Interrupciones.
UI Catálogos	El usuario podrá realizar diversas tareas como filtrar todos aquellos códigos que se desee buscar en los catálogos, así como también podrá registrar y modificar los detalles de los códigos de desglose de demoras, clasificados como categoría, tipo, zona, sistema, entre otros.
UI Usuarios	Clase que permite el registro, consulta y modificación de los usuarios y permisos necesarios para el correcto funcionamiento del sistema.

Clases de control: representan coordinación, secuencia, transacciones, y control de otros objetos y se usan con frecuencia para encapsular el control de un caso de uso en concreto. Las clases de control también se utilizan para representar derivaciones y

cálculos complejos, como la lógica del negocio, que no pueden asociarse con ninguna información concreta, de larga duración, almacenada por el sistema. A continuación, en la tabla 13 se muestran las clases de control del sistema.

Tabla 13. Clases de control del sistema en la fase de inicio.

Clases	Definición
Gestor de Catálogos	Clase encargada de controlar el ingreso de los datos que posteriormente serán necesarios para declaración de demoras en las líneas de producción, también controla los parámetros y datos para la modificación de los catálogos como categoría, zona, evento, tipo, sistema, subsistema y subevento, así como las consultas a las mismas.
Gestor de Productividad Programada	Indica la relación entre las toneladas del material producido, con relación al tiempo efectivo en el cual se produjo tal tonelaje, esto se realiza en base a las metas establecidas en la planta para una determinada línea de producción, esta clase se encarga de controlar los parámetros necesarios para el correcto funcionamiento del SIAL gestionar pérdidas de productividad.
Gestor de Demoras Clasificadas	Su función principal es la de manejar los procesos ligados a la declaración de un evento como una demora dentro de una línea de producción controlando el ingreso de los datos necesarios.
Gestor de Usuarios	Controla las labores que pueden ser llevados a cabo por los usuarios dentro del SI en lo que se refiere a la gestión del mismo, lo que conlleva a la utilización de ciertos permisos.

Clases de entidad: se utilizan para modelar información que posee una vida larga y que es a menudo persistente. A continuación, en la tabla 14 se muestran las clases de entidad del sistema.

Tabla 14. Clases de entidad del sistema en la fase de inicio.

Clases	Definición
Catálogo	Representa los datos relacionados con los arreglos de desglose de demoras.
Categoría	Representa los datos relacionados con la clase en la que se encuentra la una falla.
Tipo	Representa los datos relacionados tipos de fallas de una línea de producción.
Zona	Representa los datos de las zonas de una planta.
Evento	Representa los datos de la explicación sobre el suceso que ocurrió en la planta.
Sistema	Constituye los datos de los equipos de una línea de producción.
Subsistema	Compone los datos de cada una de las piezas que componen un equipo ha fallado.
Subevento	Establece información más detallada del evento que se generó en la planta.
Demora Justificada	Representa los datos de detalles del porqué, como y donde ocurre, una pérdida de producción.
Instalación	Datos de las líneas de producción pertenecientes a una planta.
Productividad Programada	Representa los datos de las metas de producción mensual trazadas por la gerencia de una planta.
Usuario	Compone los datos del personal encargado del control en las fallas de la línea de producción.

Tabla 14. Continuación.

Clases	Definición
Permiso Instalaciones	Permisos de gestión que posee un usuario dentro de

	una línea de producción.
Usuarios Instalaciones	Datos del personal que labora en una determinada instalación.
Usuario Cuadrilla	Datos del personal perteneciente a una cuadrilla.
Supervisores	Representa los datos del personal encargado de inspeccionar el trabajo de las cuadrillas en una línea de producción.
Nivel	Constituye los datos del nivel de jerarquía de un operador y la cantidad de días que este posee para reclasificar una demora.

3.1.6.2 Diagrama de clases de análisis

Luego de identificadas las clases de análisis se establece el diagrama de clases de análisis donde se plasman las posibles clases del diseño, encajándolas en la semántica de los tres estereotipos anteriormente explicados, que comprenden clases de interfaz, de control y de entidad.

Diagrama de clases de análisis para el caso de uso Clasificar Demoras: en el caso de uso Clasificar Demoras, se llevan a cabo funciones para ingresar, modificar y eliminar los datos referentes a las demoras que se presentan en la línea de producción con el fin de justificar un evento. En el diagrama de clases de análisis mostrado en la figura 12, se puede observar que el usuario Clasificador inicia la interacción con el sistema a través de la interfaz IU Principal SI y UI Clasificación de Eventos, para llevar a cabo las funciones anteriormente señaladas. Éstos son procesados en la clase de control denominada Gestor de Demoras Clasificadas, según sea el caso, para que el usuario obtenga la respuesta deseada. Los datos manejados en éste gestor serán almacenados en la clase entidad Demora Justificada.

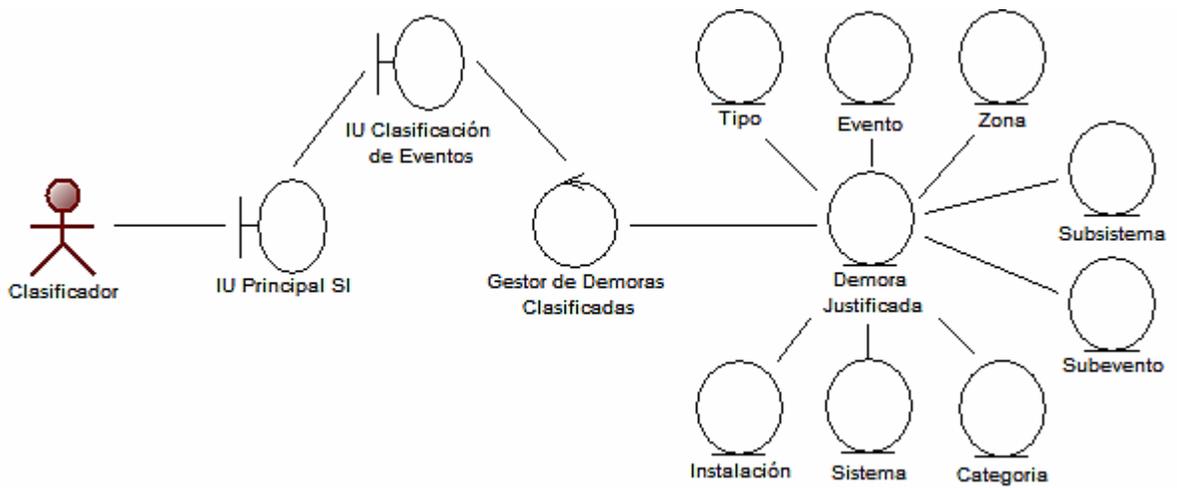


Figura 12. Diagrama de clases de análisis para el caso de uso Clasificar Demora de la fase de inicio.

Diagrama de clases de análisis para el caso de uso Gestionar Productividad: en este se llevan a cabo funciones para ingresar, modificar y eliminar los datos referentes a la productividad programada de una línea de producción en planta para un determinado mes.

En el diagrama de clases de análisis mostrado en la figura 13, se puede observar que el usuario Industrial inicia la interacción con el sistema a través de la interfaz IU Principal SI y IU Productividad Programada, para posteriormente llevar a cabo las funciones anteriormente señaladas mediante los procesos de la clase de control denominada Gestor de Productividad Programada, para que el usuario obtenga la respuesta deseada. Los datos manejados y generados en éste gestor serán almacenados en la clase entidad Productividad Programada.

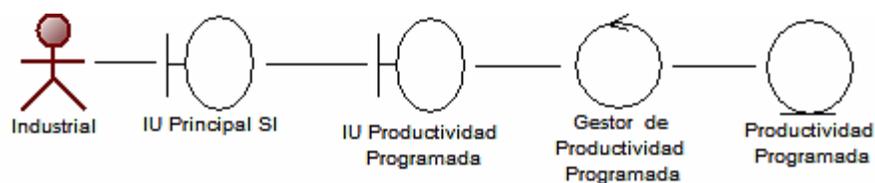


Figura 13. Diagrama de clases de análisis para el caso de uso Gestionar Productividad de la fase de inicio.

Diagrama de clases de análisis para el caso de uso Gestionar Catálogo: en este se llevan a cabo funciones para registrar, modificar y eliminar los datos referentes a los códigos y detalles de todas las conjeturas de una línea de producción que sean de interés a la hora de describir un determinado evento, como, el sistema, la zona, el evento, el tipo, categoría, subsistema y subevento, de una falla. En el diagrama de clases de análisis mostrado en la figura 14, se puede observar la interacción del usuario Industrial con el sistema por medio de la interfaz IU Principal SI y UI Catálogos, donde esta última interactúa con la clase de control denominada Gestor de Catálogos para que los datos manejados y generados en el sistema sean almacenados en la clase entidad Catálogo.

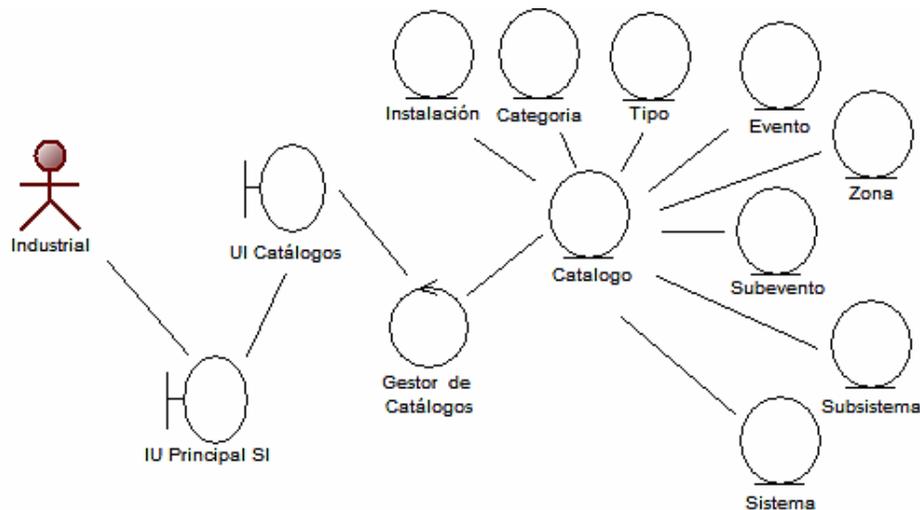


Figura 14. Diagrama de clases de análisis para el caso de uso Gestionar Catálogo de la fase de inicio.

Diagrama de clases de análisis para el caso de uso Gestionar Usuario: en este se llevan a cabo funciones para registrar, modificar y eliminar los datos referentes a los usuarios y detalles de los permisos de libertad que tienen sobre el Sistema de

Interrupciones. En el diagrama de clases de análisis mostrado en la figura 15, se puede observar la interacción del usuario Industrial con el sistema por medio de la interfaz IU Principal SI y UI Usuarios, donde esta última interactúa con la clase de control denominada Gestor de Usuarios para que los datos manejados y generados en el sean almacenados en la clase entidad Usuario.

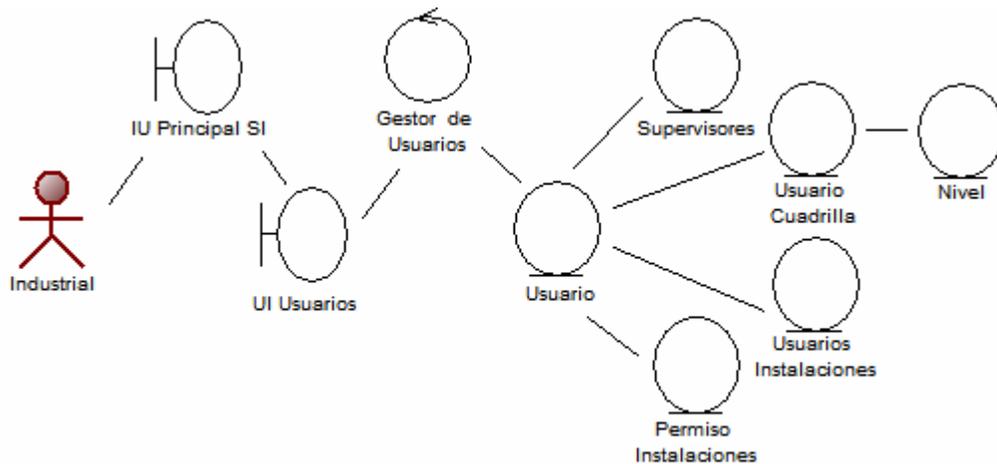


Figura 15. Diagrama de clases de análisis para el caso de uso Gestionar Usuarios de la fase de inicio.

3.1.6.3 Diagrama de colaboración

Cuando se tiene un esbozo de las clases necesarias para realizar un caso de uso, se debe describir cómo interactúan sus correspondientes objetos del análisis. Esto se hace mediante diagramas de colaboración que contienen las instancias de actores participantes, los objetos del análisis y sus enlaces. A continuación se realiza una descripción de los diagramas de colaboración de los casos de uso de la Aplicación para la Gestión del SI de SIDOR.

- Diagrama de colaboración para el caso de uso Clasificar Demora: en la realización del caso de uso se llevan a cabo las siguientes colaboraciones:

- 1) El usuario Operador, en la interfaz: IU Principal SI donde se visualiza el estado de las líneas de producción de la planta, por medio de la selección la opción Clasificar, que se encuentra habilitada para los eventos que el Sistema de Interrupciones considere una interrupción.
- 2) La clase interfaz: IU Clasificación de Eventos recibe la orden y los datos para crear o modificar una justificación del evento.
- 3) La clase de control: Gestor de Demoras Clasificadas capta los datos.
- 4) La clase de control: Gestor de Demoras Clasificadas crea una instancia de la clase entidad: Demora Justificada la cual prepara los datos y validaciones creando para ello instancias de las clases de entidad: Tipo, Evento, Zona, Sistema, Subsistema, Subevento, Instalación, y Categoría que le permiten cumplir las instrucciones.

En la figura 16 se puede detallar la descripción y el orden respectivo del diagrama de colaboración del caso de uso Clasificar Demora.

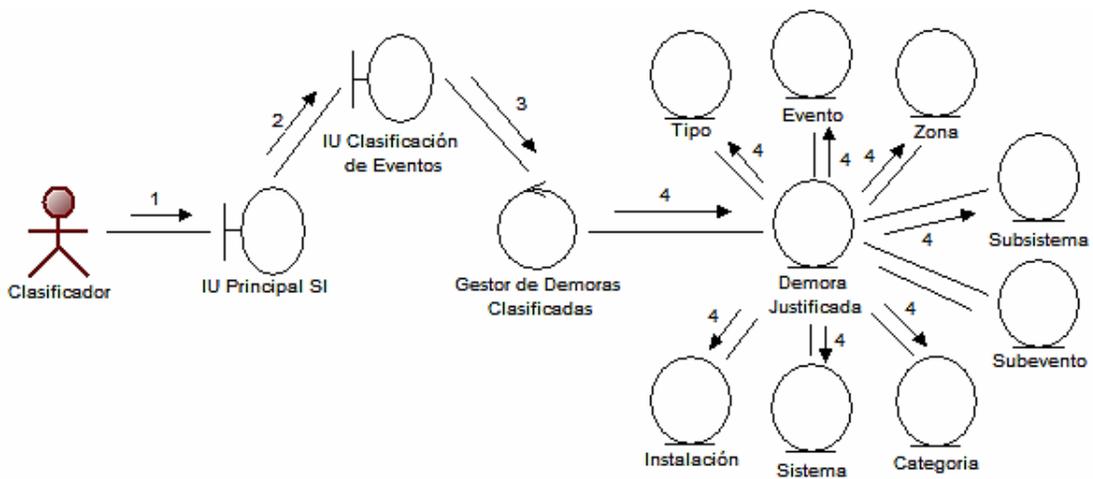


Figura 16. Diagrama de colaboración del caso de uso Clasificar Demora de la fase de inicio.

- Diagrama de colaboración para el caso de uso Gestionar Productividad: en la realización del caso de uso Gestionar Productividad se llevan a cabo las siguientes colaboraciones:

Agregar una productividad:

- 1) El usuario Industrial, en la interfaz: IU Principal SI, por medio de selección de la opción Productividad, hará posible agregar alguna Productividad Programada para la línea de producción en la que se encuentre.
- 2) El usuario Industrial provee los datos que describen la productividad programada mediante la clase interfaz: IU Productividad Programada que recibe la orden y los datos para agregar una nueva productividad.
- 3) La clase de control: Gestor de Productividad Programada capta los datos desde la interfaz.
- 4) La clase de control: Gestor de Productividad Programada crea una instancia de la clase de entidad: Productividad Programada enviando como parámetros los datos cargados por el usuario.

En la figura 17, se observa el diagrama de colaboración anteriormente descrito.

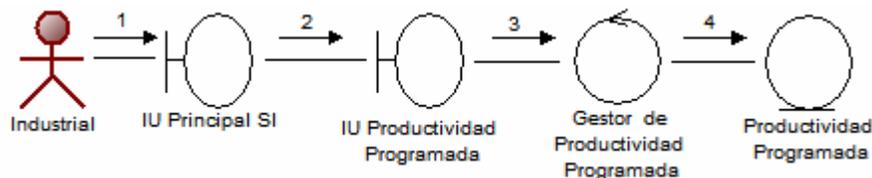


Figura 17. Diagrama de colaboración del caso de uso Gestionar Productividad Programada (agregar) de la fase de inicio

Modificar una productividad:

- 1) El usuario Industrial, en la interfaz: IU Principal SI, por medio de selección de la opción Productividad, enviará la orden para que se ejecute la interfaz: UI Productividad Programada.
- 2) La interfaz: UI Productividad Programada, recibe la orden y el usuario puede proveer los parámetros de filtrado que serán enviados para la consulta de la productividad programada en la línea.
- 3) La clase de control: Gestor de Productividad Programada, recibe los datos de filtrado para la consulta.
- 4) La clase de control: Gestor de Productividad Programada, invoca una instancia de la clase de entidad: Productividad Programada.
- 5) La clase de entidad: Productividad Programada, devuelve el objeto creado.
- 6) La clase de control: Gestor de Productividad Programada, con el objeto devuelto invoca al método de la instancia creada, que permita buscar las productividades programadas existentes que coincidan con el filtrado realizado por el usuario de Industrial.
- 7) El método devuelve las productividades programadas encontradas.
- 8) La clase de control: Gestor de Productividad Programada, envía a la clase de interfaz: UI Productividad Programada los detalles de las coincidencias para que sean listadas, siendo posible que el usuario modifique alguna de ellas.
- 9) El usuario de Industrial selecciona la productividad deseada y pulsa modificar, para que la clase de control: Gestor de Productividad Programada, capte la petición y capture los nuevos valores e invoque una función que permita realizar la modificación, función que es propia del objeto que fue instanciado en el paso 4.
- 10) La clase de control: Gestor de Productividad Programada sobrescribe los datos del objeto deseado. El proceso se puede observar en la figura siguiente:

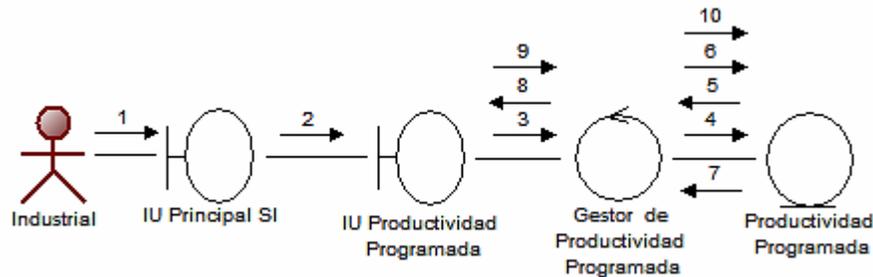


Figura 18. Diagrama de colaboración del caso de uso Gestionar Productividad Programada (modificar) de la fase de inicio.

Eliminar una productividad:

- 1) El usuario Industrial, en la interfaz: IU Principal SI, por medio de selección de la opción Productividad, enviará la orden para que se ejecute la interfaz: UI Productividad Programada.
- 2) La interfaz: UI Productividad Programada, recibe la orden y el usuario puede proveer los parámetros de filtrado que serán enviados para la consulta de la productividad programada en la línea.
- 3) La clase de control: Gestor de Productividad Programada, recibe los datos de filtrado para la consulta.
- 4) La clase de control: Gestor de Productividad Programada, invoca una instancia de la clase de entidad: Productividad Programada.
- 5) La clase de entidad: Productividad Programada, devuelve el objeto creado.
- 6) La clase de control: Gestor de Productividad Programada, con el objeto devuelto invoca al método de la instancia creada, que permita buscar las productividades programadas existentes que coincidan con el filtrado realizado por el usuario de Industrial.
- 7) El método devuelve las productividades programadas encontradas.

- 8) La clase de control: Gestor de Productividad Programada, envía a la clase de interfaz: UI Productividad Programada los detalles de las coincidencias para que sean listadas, siendo posible que el usuario elimine alguna de ellas.
- 9) El usuario de Industrial selecciona la productividad programada deseada y pulsa eliminar, para que la clase de control: Gestor de Productividad Programada, capte la petición realizada e invoque una función que permita realizar la supresión, función que es propia del objeto que fue instanciado en el paso 4.
- 10) La clase de control: Gestor de Productividad Programada elimina la Productividad Programada deseada.

En la figura 19, se observa el diagrama de colaboración anteriormente descrito.

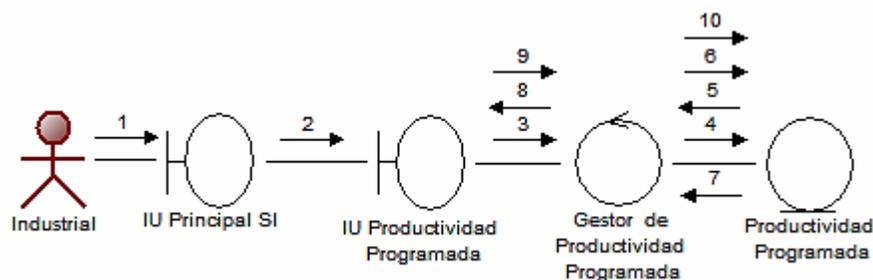


Figura 19. Diagrama de colaboración del caso de uso Gestionar Productividad Programada (eliminar) de la fase de inicio.

- Diagrama de colaboración para el caso de uso Gestionar Catálogo: en la realización del caso de uso Gestionar Catálogo se llevan a cabo las siguientes colaboraciones:

Agregar un catálogo:

- 1) El usuario Industrial, en la interfaz: IU Principal SI, por medio de selección de la opción Catálogo, hará posible agregar algún elemento para un catálogo en la línea de producción en la que se encuentre.
- 2) El usuario Industrial provee los datos que describen la productividad programada mediante la clase interfaz: IU Catálogos que recibe la orden y envía los datos para agregar un nuevo elemento a un catálogo.
- 3) La clase de control: Gestor de Catálogos capta los datos desde la interfaz.
- 4) La clase de control: Gestor de Catálogos crea una instancia de la clase de entidad: Catálogo, enviando los datos cargados por el usuario.
- 5) La clase de entidad: Catálogo, crea a su vez una instancia de la clase entidad: Instalación, así como también del tipo de elemento del que se tratase el nuevo catálogo: Categoría, Tipos, Evento, entre otros, preparando los datos que permiten cumplir con las instrucciones del.

En la figura 20, se observa el diagrama de colaboración anteriormente descrito.

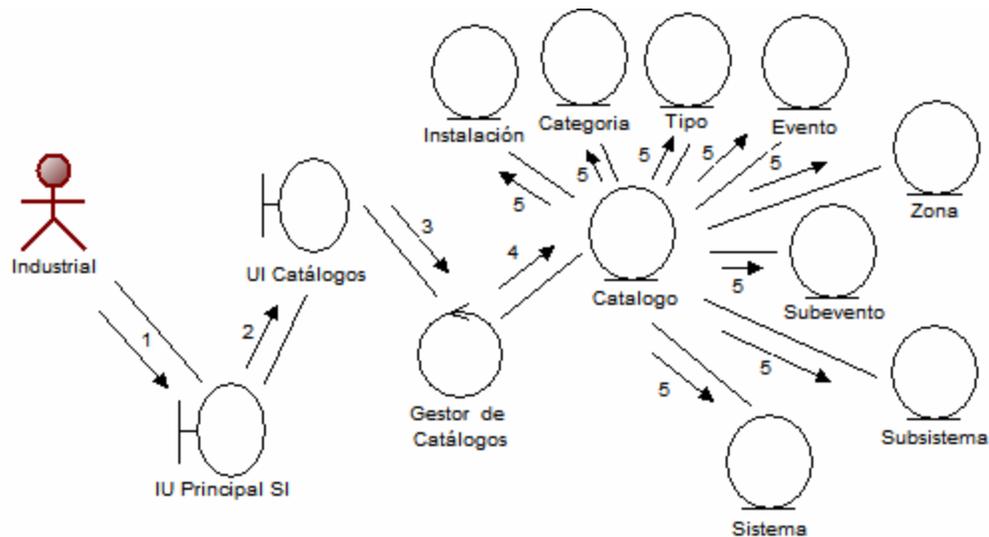


Figura 20. Diagrama de colaboración del caso de uso Gestionar Catálogo (agregar) de la fase de inicio

- Diagrama de colaboración para el caso de uso Gestionar Usuarios: en la realización del caso de uso Gestionar Usuarios se llevan a cabo las siguientes colaboraciones:

Agregar un usuario:

- 1) Industrial, en la interfaz: IU Principal SI, por medio de selección de la opción Usuarios, enviará la orden para que se ejecute la interfaz: UI Usuarios.
- 2) La interfaz: UI Usuarios, recibe la orden y envía una petición de consulta de los usuarios pertenecientes a la línea de producción.
- 3) La clase de control: Gestor de Usuarios, recibe los datos de filtrado por defecto.
- 4) Gestor de Usuarios, invoca una instancia de la entidad: Usuario.
- 5) La clase entidad: Usuario invoca instancias de las clases de entidad: Supervisores, Usuario Cuadrilla, Nivel, Permiso Instalaciones, entre otros.
- 6) Las clases de entidad: Supervisores, Usuario Cuadrilla, Nivel, Permiso Instalaciones y Usuario Instalaciones, devuelven los objetos creados.
- 7) La entidad: Usuario, devuelven los objetos creados a la clase de control.
- 8) La clase de control: Gestor de Usuarios, con los objetos devueltos invoca los métodos de las instancias creadas, que permitan buscar las relaciones entre las clases de entidad antes mencionadas.
- 9) Los métodos devuelven los usuarios encontrados.
- 10) Gestor de Usuarios, envía a UI Usuarios los detalles de las coincidencias para que sean listadas, siendo posible que el usuario Agregue alguno nuevo.

- 11) Industrial cargalos datos del nuevo usuario y pulsa aceptar, para que el Gestor de Usuarios, capte la petición, capture los nuevos valores e invoque la función que permita la agregación de los datos.
- 12) La clase de control: Gestor de Usuarios escribe los datos del nuevo objeto.

En la figura 21, se observa el diagrama de colaboración anteriormente descrito.

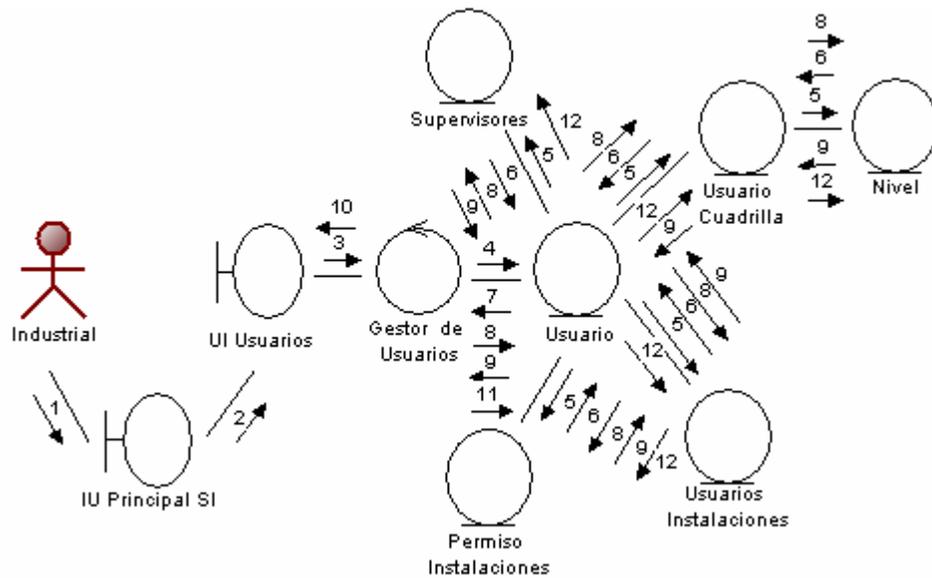


Figura 21. Diagrama de colaboración del caso de uso Gestionar Usuarios (agregar) de la fase de inicio

Modificar un usuario:

- 1) El usuario Industrial, en la interfaz: IU Principal SI, por medio de selección de la opción Usuarios, enviará la orden para que se ejecute la interfaz: UI Usuarios.
- 2) La interfaz: UI Usuarios, recibe la orden y envía una petición de consulta de los usuarios pertenecientes a la línea de producción.
- 3) La clase de control: Gestor de Usuarios, recibe los datos de filtrado por defecto para la consulta.

- 4) La clase de control: Gestor de Usuarios, invoca una instancia de la clase de entidad: Usuario.
- 5) La clase entidad: Usuario invoca instancias de las clases de entidad: Supervisores, Usuario Cuadrilla, Nivel, Permiso Instalaciones y Usuario Instalaciones.
- 6) Las clases de entidad: Supervisores, Usuario Cuadrilla, Nivel, Permiso Instalaciones y Usuario Instalaciones, devuelven los objetos creados.
- 7) Las clases de entidad: Usuario, devuelven los objetos creados a la clase de control.
- 8) La clase de control: Gestor de Usuarios, con los objetos devueltos invoca los métodos de las instancias creadas, que permitan buscar las relaciones entre las clases de entidad Supervisores, Usuario Cuadrilla, Nivel, Permiso Instalaciones y Usuario Instalaciones, donde dichas relaciones definen los usuarios que coinciden con el filtrado por defecto.
- 9) Los métodos devuelven los usuarios encontrados.
- 10) La clase de control: Gestor de Usuarios, envía a la clase de interfaz: UI Usuarios los detalles de las coincidencias para que sean listadas, siendo posible que el usuario modifique alguno.
- 11) Industrial selecciona los datos del usuario seleccionado y pulsa modificar, para que la clase de control: Gestor de Usuarios, capte la petición y capture los nuevos valores e invoque una función que permita realizar la operación.
- 12) La clase de control: Gestor de Usuarios sobrescribe los datos del nuevo objeto.

En la figura 22, se observa el diagrama de colaboración anteriormente descrito.

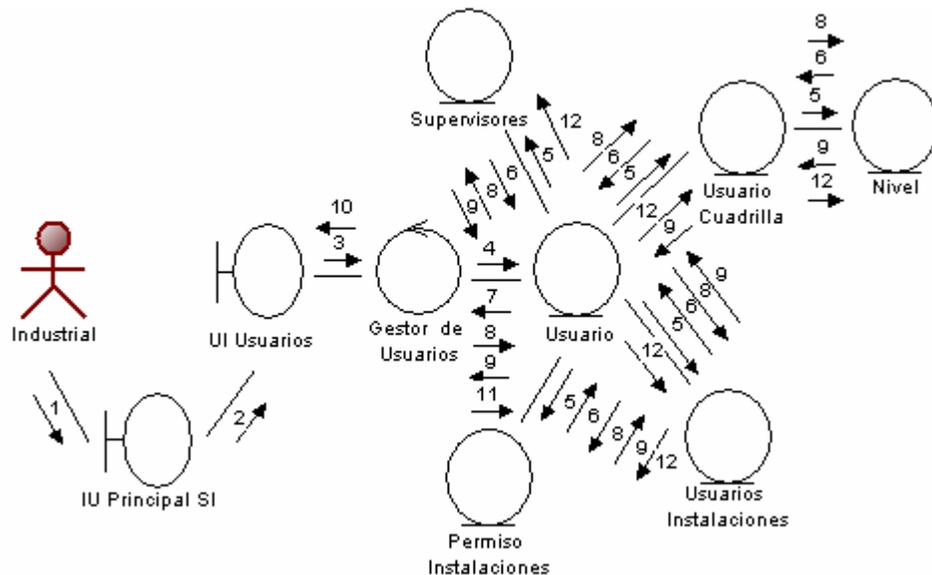


Figura 22. Diagrama de colaboración del caso de uso Gestionar Usuarios (modificar) de la fase de inicio

Eliminar un usuario:

- 1) El usuario Industrial, en la interfaz: IU Principal SI, por medio de selección de la opción Usuarios, da la orden para que se ejecute la interfaz: UI Usuarios.
- 2) La interfaz: UI Usuarios, recibe la orden y envía una petición de consulta de los usuarios pertenecientes a la línea de producción.
- 3) La clase de control: Gestor de Usuarios, recibe los datos de filtrado.
- 4) La clase de control: Gestor de Usuarios, invoca una instancia de la clase de entidad: Usuario.
- 5) Usuario, invoca instancias de las clases de entidad: Supervisores, Usuario Cuadrilla, Nivel, Permiso Instalaciones y Usuario Instalaciones.
- 6) Las clases de entidad: Supervisores, Usuario Cuadrilla, Nivel, Permiso Instalaciones y Usuario Instalaciones, devuelven los objetos creados.

- 7) Las clases de entidad: Usuario, devuelven los objetos creados.
- 8) La clase de control: Gestor de Usuarios, con los objetos devueltos invoca los métodos de las instancias creadas, que permitan buscar las relaciones entre las clases de entidad Supervisores, Usuario Cuadrilla, Nivel, Permiso Instalaciones y Usuario Instalaciones, donde dichas relaciones definen los usuarios que coinciden con el filtrado por defecto.
- 9) Los métodos devuelven los usuarios encontrados.
- 10) La clase de control: Gestor de Usuarios, envía a la clase de interfaz: UI Usuarios los detalles de las coincidencias para que sean listadas, siendo posible que el usuario elimine alguno.
- 11) Industrial selecciona el usuario y pulsa eliminar, para que Gestor de Usuarios, capte la petición e invoque la función para realizar la supresión.
- 12) La clase de control: Gestor de Usuarios suprime el registro.

En la figura 23, se observa el diagrama de colaboración anteriormente descrito.

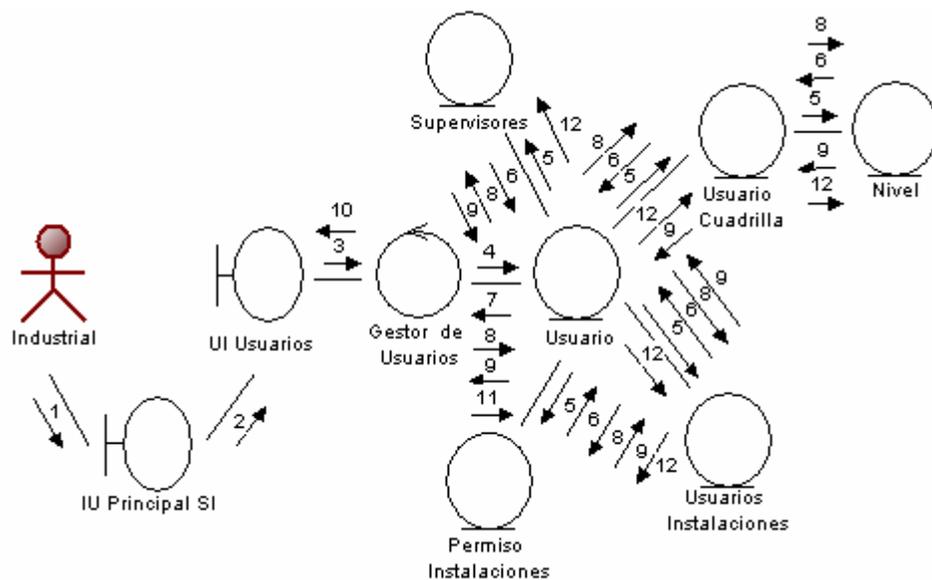


Figura 23. Diagrama de colaboración del caso de uso Gestionar Usuarios (eliminar) de la fase de inicio

3.1.6.4 Identificación de paquetes de análisis

Los paquetes de análisis constituyen una forma de organizar los artefactos del modelo de análisis y realizaciones de casos de uso en partes más manejables que representan abstracciones de subsistemas y posiblemente capas completas del diseño del sistema. Los paquetes de análisis deberán ser cohesivos, y deberán ser débilmente acoplados (es decir, sus dependencias unos de otros deberán minimizarse). A continuación, en las figuras 24, 25, 26 y 27 se presentan los paquetes de análisis identificados a partir de los casos de usos más relevantes del software:

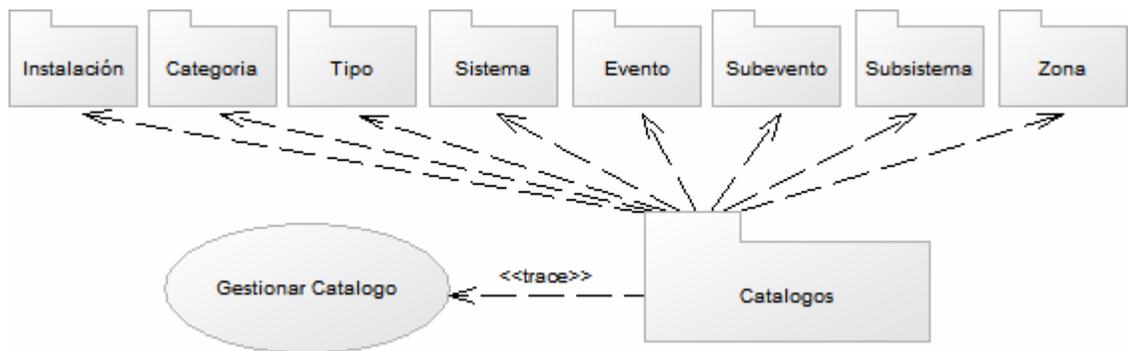


Figura 24. Paquete de análisis de Catálogos.

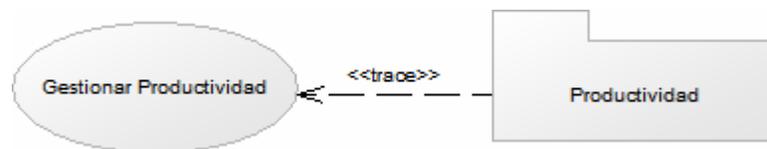


Figura 25. Paquete de análisis de Productividad.

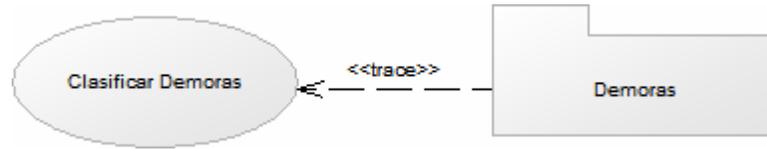


Figura 26. Paquete de análisis de Demoras.

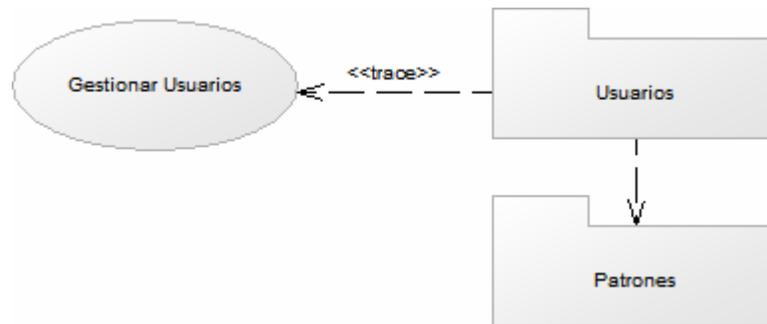


Figura 27. Paquete de análisis de Usuarios.

3.1.7 Evaluación de la fase de inicio

En esta fase, se analizó el sistema actual para capturar los objetos más importantes dentro del contexto del SI, se listaron de requisitos funcionales, un modelo del negocio y un modelo de dominio. Luego se construyó el modelo de casos de uso, también se determinó un modelo de análisis, diagramas de colaboración y por último los paquetes de análisis. En la tabla 15, se presenta el estatus de los artefactos:

Tabla 15. Estatus de los artefactos planificados en la fase de inicio.

Artefacto	Estatus
Modelo del negocio	Culminado
Modelo de dominio	Culminado
Lista de riesgos del sistema	Culminado
Requisitos funcionales	En desarrollo
Requisitos no funcionales	Culminado

Requisitos de software y hardware	Culminado
Modelo de casos de uso	En desarrollo
Descripción textual de casos de uso	En desarrollo
Modelo de análisis	En desarrollo
Diagrama de clases de análisis	En desarrollo
Identificación de paquetes de análisis	En desarrollo

Después de examinar los objetivos de la fase de inicio, el ámbito del sistema, los riesgos críticos y la arquitectura candidata, se logró establecer que el proyecto es viable y por tanto se decidió continuar con el desarrollo del sistema planteado.

3.2 FASE DE ELABORACIÓN

En esta sección se establece una línea base de la arquitectura que pueda guiar el desarrollo del sistema en su totalidad, así como la adición de mejoras y nuevas funcionalidades. Se recopila la mayoría de los requisitos que aún quedan pendientes. Además se transforman y refinan los modelos de la fase de inicio en otra serie de modelos que perfilen una solución más cercana a los requerimientos de los usuarios.

3.2.1 Planificación de la fase de elaboración.

En esta fase se desarrollaron los flujos de trabajos requisitos, análisis, diseño, implementación y prueba, tomando como guía los modelos construidos en la fase de inicio. Se llevó a cabo el flujo de trabajo requisitos, en donde se identificaron nuevos requerimientos, así como también se evaluaron y refinaron los de la fase de inicio. Además se adaptó el modelo de casos de usos a la aparición de nuevos casos de uso. Se desarrolló el flujo de trabajo análisis, para esto se elaboraron los diagramas de clases de análisis como una especificación detallada de los requisitos y como primera aproximación al modelo de diseño; luego se realizaron los diagramas de colaboración para representar las interacciones entre los objetos del sistema y se construyó el

diagrama de paquetes de análisis para encapsular los casos de usos que fueron definidos al realizar el análisis del sistema. En el flujo de trabajo diseño, se construyó el diagrama de clases de diseño para mostrar la estructura del sistema, se elaboraron los diagramas de secuencia en donde se visualiza el orden de interacción entre los objetos del sistema, se diseñó la base de datos para manejar y almacenar toda la información necesaria, se creó la interfaz de usuario. Finalmente se llevó a cabo el flujo de trabajo implementación, donde se procedió a transformar un modelo de diseño en código ejecutable. Para este proyecto, en esta fase se realizó una única iteración conformada por los flujos de trabajo mencionados. A continuación, en la tabla 16 se muestra lo expuesto anteriormente.

Tabla 16. Actividades y artefactos planificados para la fase de elaboración.

Actividades	Artefactos
Requisitos	Requisitos funcionales y no funcionales Requisitos de software y hardware Captura de requisitos como casos de usos: identificación de actores y casos de uso Descripción de casos de uso. Modelo de casos de uso Prototipos de interfaz de usuario
Análisis	Modelo de análisis Diagrama de clase de análisis Diagrama de colaboración Identificación de paquetes de análisis Diagramas de paquetes de análisis
Diseño	Diseño de la arquitectura Modelo de diseño Diagrama de clase de diseños y glosario de métodos

	Diagrama de secuencia
	Diseño físico de la base de datos
Implementación	Código fuente para la realización de los casos de usos
Pruebas	Casos de prueba

A continuación se procede a describir cada uno de los artefactos planificados para la fase de elaboración:

3.2.2 Flujo de trabajo (*workflow*) de requisitos

La captura de requisitos en esta fase se lleva a cabo con la finalidad de completar el trabajo realizado en la fase anterior, contemplando nuevos requisitos que habían sido obviados.

3.2.2.1 Requerimientos funcionales

Para la elaboración de esta fase, surgieron los requisitos por parte de los usuarios del sistema, los cuales están expresados a continuación en la tabla 17:

Tabla 17. Requerimientos funcionales en fase de elaboración.

Requerimiento	Necesidad	Estado
Gestionar patrones de rotación de las cuadrillas de trabajo.	E	A
Generar reportes que permitan el control de los procesos de las plantas de SIDOR.	E	A
Establecer perfiles de usuarios con agrupaciones lógicas de las funcionalidades de la aplicación.	E	A

3.2.2.2 Requerimientos no funcionales

Para esta fase no se encontraron requisitos no funcionales debido a que los requerimientos necesarios ya fueron identificados en la fase de inicio.

3.2.2.3 Captura de requisitos como casos de uso

En vista de que han surgido nuevos requisitos a los ya capturados en la fase anterior, se han identificado nuevos casos de uso y se modificará levemente alguno ya existente, para completar la descripción del sistema en cuanto a operatividad. A continuación en la tabla 18, se presentan los casos de usos identificados.

Tabla 18. Casos de uso de la Aplicación *Web* para la Gestión del SI de SIDOR en la fase de elaboración.

Casos de usos	Descripción
Gestionar patrones	Agrega, modifica y elimina los patrones de cuadrillas.
Generar reportes	Genera reportes impresos y electrónicos referentes a las consultas de usuarios.

3.2.2.4 Modelo de casos de uso

En la figura 28, se puede observar la inclusión de los nuevos casos de uso.

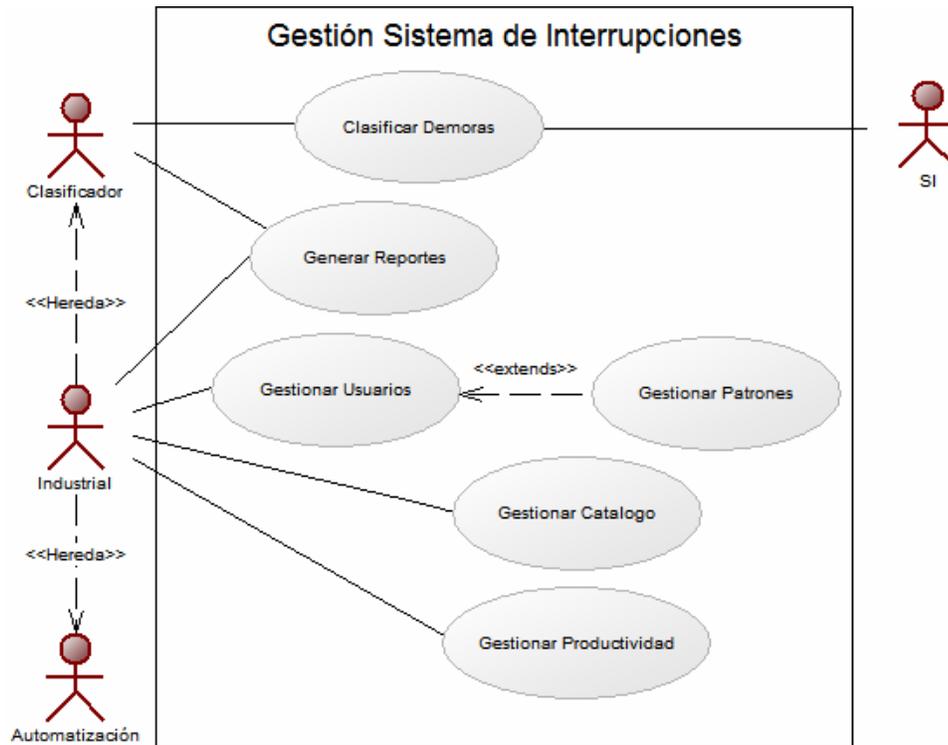


Figura 28. Diagrama de caso de uso de la Aplicación *Web* para la Gestión del SI de SIDOR de la fase de elaboración.

A continuación se presentan una descripción en detalle de los nuevos casos de determinados para el sistema así como otras necesidades de los usuarios:

Tabla 19. Caso de Uso Gestionar Patrones.

Gestionar Patrones	
Actor Principal:	Industrial
Personal involucrado e intereses:	
Industrial:	Cargar la rotación de las cuadrillas por turno durante el mes.
Jefe de Planta:	Desea que el sistema muestre la información de la cuadrilla que se encuentra laborando en ese turno.
Supervisor:	Desea que el sistema muestre la información de la cuadrilla que se encuentra laborando en ese turno.

Operador: Desea que el sistema muestre la información de la cuadrilla que se encuentra laborando en ese turno.

Pre-Condiciones: El usuario se identifica y autentica.

Post-Condiciones: El sistema carga el patrón según la planeación del usuario.

Escenario principal de éxito:

1.- El sistema muestra todos los patrones vigentes y no vigentes.

2.- El usuario selecciona insertar un patrón.

3.- El usuario selecciona con cuantas cuadrillas se va a trabajar en la planta.

4.- El usuario selecciona la fecha de inicio de vigencia del patrón.

5.- El usuario decide si agregar una fecha fin de vigencia.

6.- El sistema muestra en una matriz los días y turnos para asignar a las cuadrillas.

7.- El usuario va seleccionando para cada turno de cada día la cuadrilla correspondiente.

8.- El usuario decide guardar el patrón.

9.- El sistema revisa que el patrón no se solape con un patrón vigente.

10.- El sistema revisa que una cuadrilla no trabaje 2 turnos seguidos.

11.- El sistema revisa que una cuadrilla no trabaje 2 turnos en un mismo día.

Tabla 19. Continuación.

12.- El sistema registra el patrón.

13.- El sistema va al paso 1.

Flujos Alternativos:

2.a.- El usuario selecciona modificar un patrón vigente.

1.- Extender el patrón.

1.- el usuario decide cambiar la fecha fin de vigencia.

2.- el usuario guarda la modificación.

3.- el sistema va al paso 9.

2.- Eliminar patrón.

2.b.- El usuario selecciona modificar un patrón no vigente.

1.- el usuario decide copiar el patrón no vigente.

2.- el usuario cambia las fechas de vigencias.

3.- el usuario guarda el patrón.

4.- el sistema va al paso 9.

8.a.- El usuario decide cancelar la operación.

Tabla 20.Caso de Uso Generar Reportes.

Generar Reportes

Actor Principal: Industrial

Personal involucrado e intereses:

Industrial: Desea tener un reporte detallado de productividad programada, catálogos, usuarios y patrones a través de opciones de filtrado para obtener información de interés.

Jefe de Planta: Desea que el SI no tenga incongruencias en los datos.

Supervisor: Desea que el SI no tenga incongruencias en los datos.

Operador: Desea que el SI no tenga incongruencias en los datos.

Pre-Condiciones: El usuario se identifica y autentica.

Tabla 20. Continuación.

Post-Condiciones: El sistema genera un reporte en un formato portable, organizado de acuerdo a las necesidades del solicitante.

Escenario principal de éxito:

1.- El usuario selecciona una línea de la cual quiere revisar la información contenida en los módulos de productividad programada, catálogos, usuarios o patrones.

2.- El usuario usa las herramientas de filtrado que proporciona la aplicación para buscar la información de su interés.

3.- El sistema revisa la petición del usuario.

4.-El sistema busca la información que corresponde al módulo de acuerdo a como lo deseó el usuario.

5.- El reporte es listado en pantalla con más opciones para organizar o filtrar.

6.- El usuario decide exportar el reporte a un formato PDF o Excel para su portabilidad.

Flujos Alternativos:

*a.- En cualquier momento el sistema falla:

1.- Se debe emitir un mensaje al usuario, de manera tal que reporte el fallo.

5.a.- No se visualiza el listado.

1.- Mostrar un mensaje indicando que no existen datos para la consulta realizada.

2.- Se deshabilitan las opciones de exportación.

Para el caso de uso Gestionar Usuarios surgió una leve modificación en donde el Actor de Industrial al cargar o modificar los datos de un usuario, estará en el deber de seleccionar un determinado perfil para el usuario en gestión.

3.2.2.5 Perfiles de usuarios

En los requerimientos funcionales, también se determinó la necesidad de distribuir las funcionalidades del sistema en agrupaciones lógicas que permitan abarcar las necesidades de información y de acceso dentro de la aplicación, con la finalidad de proporcionar solo lo que necesitan los usuarios para llevar a cabo sus actividades con el mayor éxito y simpleza posible. Esto se logra mediante la inclusión de perfiles de usuarios, lo cual se plantea debidamente a continuación.

En SIDOR C.A., existe una gran variedad de personas que desempeñan diversas actividades dentro de ella. Según la gerencia en la cual estén laborando, tendrán un determinado perfil.

El software a desarrollar podrá ser utilizado por una gran cantidad de usuarios, ya que se adaptará a las necesidades de las diferentes plantas o líneas de producción de SIDOR. Para esto será necesario contar con cinco perfiles de usuarios: perfil de operadores, perfil de supervisores, perfil jefe de planta, perfil de industrial y perfil de automatización.

En las Gerencias de Laminación en Caliente, Laminación en Frio, Productos Primarios, Acería de Planchones, Acería de Palanquillas, Barras y Alambrón y Planta de Cal, el personal que más interactuará con el sistema, serán los operadores de la

línea, los cuales estarán encargados de justificar cada evento que refleje el SI (Interrupciones, Pérdidas de Productividad), y de ejercer a su vez labores que corresponde de la línea. Además de los operadores, los supervisores y jefe de planta serán los otros usuarios que interactuarán con el sistema, revisando o modificando las justificaciones hechas por el operador, las cuales consideren ellos que no fuesen bien declaradas.

SIDOR es una empresa que en teoría nunca para su producción y que trabaja las 24 horas de los 7 días de la semana, para ello establecieron 3 turnos de trabajos, los cuales comprenden desde las 23 hasta las 07 del otro día, de las 07 a las 15 y de las 15 a las 23, siendo estos manejados por cuadrillas. Las cuadrillas están conformadas por un Jefe de planta, un supervisor y cuatro operadores. Dentro de ella se trabajan con cuatro cuadrillas de las cuales siempre una va estar libre mientras las otras cumplen su turno asignado, y se irán rotando el día libre entre ellas.

A partir de estos puntos fueron tomados en cuenta los siguientes usuarios para el sistema:

- **Perfil de operadores:** muy pocos de ellos se sienten cómodos con el manejo de la computadora y no están familiarizados con ningún sistema computarizado, son muy inexpertos y un poco precavidos en su uso, muchos no han manejado una PC, la mayoría comprenden edades entre 30 y 40 años de edad, y vienen trabajando en la empresa por más de 10 años, así que su conocimiento sobre el funcionamiento de la línea de producción es bien entendida por ellos, y lograda por años de experiencia. Su nivel de educación no es muy elevado, y la mayoría no han terminado el bachillerato, a su vez son muy reacios al cambio, todos pertenecen a una nómina colectiva, en donde los convenios y aumento de sueldo son manejados por el sindicato de trabajadores.

- **Perfil de supervisor:** son personas con un nivel educativo más alto que el de los operadores, capacitados en diferentes ramas de la ingeniería, su manejo de la computadora y de sistemas computarizado es más versátil, están acostumbrados a interactuar con ellas más seguido que los operadores, así que la incorporación de una PC en la línea de producción para el manejo del sistema de interrupción no será inconveniente para ellos. Así como los operadores estos están también en otra nómina colectiva, pero diferente a la de los operadores. Son considerados “Personal de Confianza” y están disponibles a cualquier hora del día, siempre que la planta los necesite, sin recibir por ellos un pago de horas extras trabajadas. En el Sistema de Interrupciones se tiene definido que monitoree las declaraciones de los operadores, y ante cualquier inconformidad podrá cambiar la declaración hecha por el operador. Este tiene derecho a clasificar o modificar en todos los turnos hasta cinco (5) días atrás.

- **Perfil jefe de planta:** Deben coordinar a los operadores y supervisores en las tareas diarias de la planta, a su vez, tienen la responsabilidad de que todo trabajo que se vaya hacer en la línea esté debidamente calculado, planeado y que también se hayan tomado todas las medidas de seguridad antes de poder aprobarlo. En el sistema de interrupciones debe revisar diariamente las imputaciones que ha colocado el operador en la línea, validando la información que el sistema muestra; en caso de error o no clasificación de algún evento, el jefe de planta debe corregir. Tiene derecho a clasificar o modificar en todos los turnos hasta cinco (5) días atrás.

- **Perfil de industrial:** son personas que están bajo la coordinación de la Gerencia de Industrial, los cuales son los encargados de toda la gestión en la planta. Se encargan de elaborar y programar la producción del mes, y de coordinar y estudiar todos los eventos relacionados con ella. Estos serán los encargados de gestionar el sistema de interrupciones y deberán asegurarse que todos los demás

usuarios del sistema estén ejerciendo su labor en el mismo. Tienen derecho a clasificar o modificar en todos los turnos hasta cuarenta y cinco (45) días atrás.

- **Perfil de automatización:**son personas que están bajo la coordinación del Departamento de Automatización y Control, los cuales son los encargados de brindar todo el apoyo y soporte técnico a los usuarios del Sistema de Interrupciones. Son superusuarios en el SI por lo tanto no poseen ningún tipo de restricción a la hora de utilizarlo.

3.2.2.6 Prototipo de la interfaz de usuario

Los prototipos de interfaz de usuario ayudan a comprender y especificar las interacciones entre actores humanos y el sistema durante la captura de requisitos. No sólo nos ayuda a desarrollar una interfaz gráfica mejor, sino también a comprender mejor los casos de uso. En vista de que han sido capturados la mayoría de los requerimientos del sistema y se han definido todos los casos de uso, es conveniente el modelado de interfaz de usuario en este flujo de trabajo. A continuación en las figuras 29 y 30 se puede observar un primer bosquejo de las interfaces principales que se diseñaron:

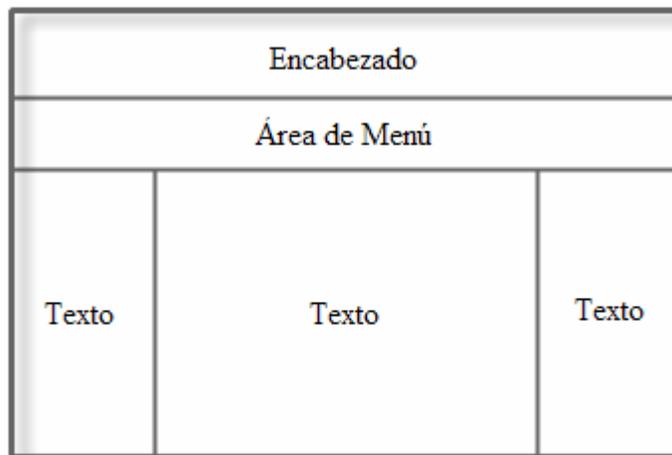


Figura 29. Prototipo de interfaz principal de la aplicación.

En el prototipo anterior se ubican los bloques principales que van a componer la interfaz principal del sistema.

La finalidad de esta interfaz es enlazar todas las demás interfaces que se diseñarán para representar la actividad de los casos de uso, siguiendo los lineamientos de interfaz de la empresa.

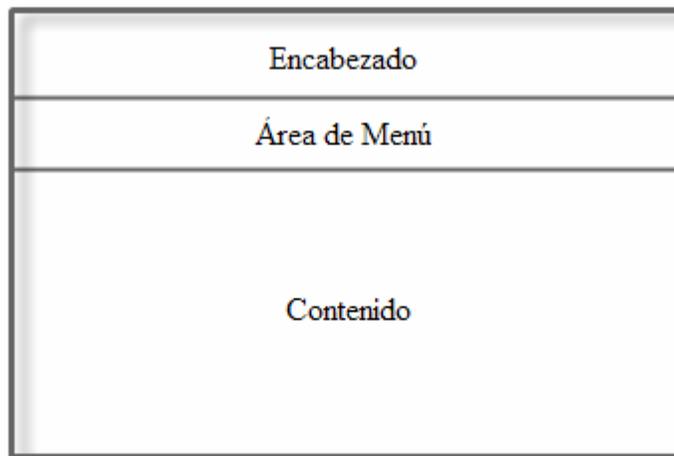


Figura 30. Prototipo de interfaz secundaria de la aplicación.

En principio para el diseño de las pantallas, se deberá mantener el estándar de la empresa de un menú el cual es creado en tiempo de ejecución mediante un control de usuario y utilizando un archivo XML.

Los colores a utilizar también entran en los estándares de la empresa y ya se tienen definidos los estilos de presentación, las demás áreas de las pantallas a diseñar fueron acordadas con el cliente por medio de múltiples reuniones, de las cuales se obtuvieron la información que querían ellos visualizar en el sistema.

3.2.3 Flujo de trabajo (*workflow*) de análisis

En la fase de inicio se analizó un pequeño porcentaje del sistema, por lo que en esta fase de elaboración se profundizará y se analizarán los nuevos casos de uso identificados en este flujo de trabajo.

3.2.3.1 Modelo de análisis

En vista de que han sido identificados nuevos casos de usos, es necesario refinar el modelo de análisis de la Aplicación *Web* para la Gestión del SI de SIDOR. En tal sentido, se realizarán modificaciones a los diagramas mostrados en la fase anterior, y se detallará la realización de los otros casos de uso del sistema.

Identificación de clases de análisis: en la fase de inicio, fueron identificadas las clases de análisis para la realización de los casos de uso Clasificar Demoras, Gestionar Catálogos y Gestionar Productividad, mostradas en las tablas 12, 13 y 14; dichas clases no han sido modificadas y se seguirán considerando para la fase de elaboración. Sin embargo las clases de análisis para la realización del caso de uso Gestionar Usuarios, sufrieron cambios debido a nuevas clases identificadas. De igual forma como en esta fase se encontraron nuevos casos de uso, como Gestionar Patrones y Generar Reportes, de estos se derivan un nuevo grupo de clases de análisis.

A continuación en las tablas 21, 22 y 23 serán presentadas las nuevas clases de análisis identificadas:

Tabla 21. Clases de interfaz del sistema de la fase de elaboración.

Clases	Definición
--------	------------

UI Patrones	En esta el usuario seleccionara la cantidad de cuadrillas con la que se trabajará en la línea, donde se generará una tabla de datos con la cantidad de semanas y la fecha de inicio de vigencia y fin de vigencia del patrón. Conteniendo información importante por medio de la cual Industrial seleccionará para cada cuadrilla el turno que le corresponde trabajar en el día.
UI Reportes	Se integrara a cada uno de los módulos de la aplicación donde el usuario tenga la necesidad de exportar la información, como por ejemplo, catálogos, productividad programada y usuarios.

Tabla 22. Clases de control del sistema de la fase de elaboración.

Clases	Definición
Gestor de Patrones	Controla la rotación de las cuadrillas por las líneas de producción permitiendo la distribución de las labores de los empleados de dichas cuadrillas según los turnos en que a estas les corresponda trabajar.
Gestor de Configuración de Reportes	Esta clase posee los métodos y los atributos necesarios para establecer la configuración de los reportes.

Tabla 23. Clases de entidad del sistema de la fase de elaboración.

Clases	Definición
Cuadrilla Instalaciones	Datos de las cuadrillas encargas de atender una línea.
Cuadrillas	Permite establecer la configuración de reportes, invocando la creación de un objeto que contiene los métodos necesarios realizar dicha tarea.
Patrón	Constituye los datos de la rotación de las cuadrillas de

	trabajo durante un lapso de tiempo.
Perfiles	Datos de la clasificación del personal que labora dentro de la planta.
Parámetros de Reportes	Datos del grupo de trabajo encargado de la línea de producción durante un lapso de tiempo.

3.2.3.2 Diagrama de clases de análisis

Diagrama de clases de análisis para el caso de uso Gestionar Patrones: en este se llevan a cabo funciones para registrar, modificar y eliminar el esquema de rotación mensual de las cuadrillas que atienden una línea durante tres turnos. En la figura 31, se puede observar la interacción del usuario Industrial con el sistema por medio de la interfaz IU Principal SI y UI Patrones, donde esta última interactúa con la clase de control denominada Gestor de Patrón para que los datos manejados y generados en el sean almacenados en la clase entidad Patrón.

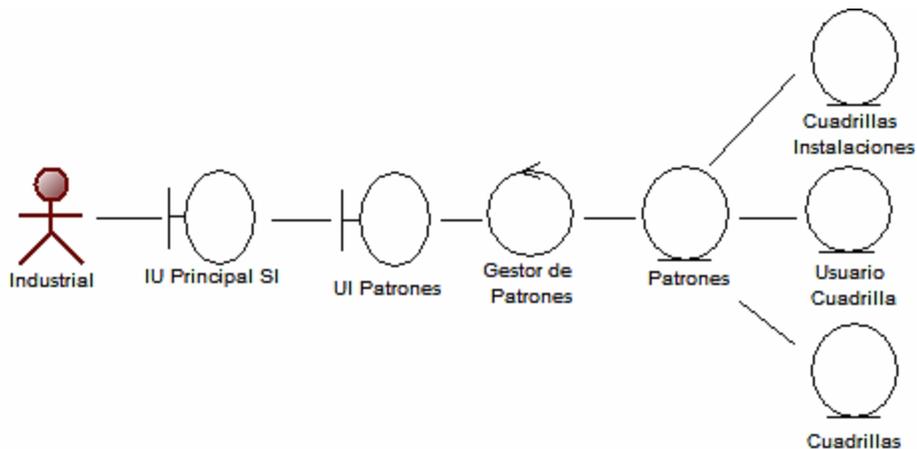


Figura 31. Diagrama de clases de análisis para el caso de uso Gestionar Patrones de la fase de elaboración.

Diagrama de clases de análisis para el caso de uso Generar Reportes: en la clase de análisis de la figura 32, el usuario inicia la interacción con el sistema mediante la

clase de interfaz IU Principal SI y UI Reportes, esta última muestra una serie de opciones que le permiten establecer las configuraciones de los reportes. El usuario establece la configuración, ésta es captada desde la interfaz por la clase de control Gestor de Configuración de Reportes. Esta clase de control invoca la creación de un objeto Parámetros de Reportes, el objeto es creado y es devuelto a la clase de control, posteriormente la clase de control realiza un llamado a la función establecer parámetros, función propia del objeto Parámetros de Reporte.

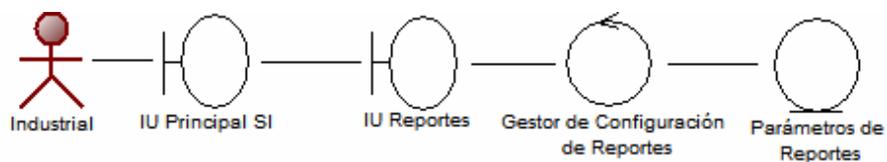


Figura 32. Diagrama de clases de análisis para el caso de uso Generar Reportes de la fase de elaboración.

Diagrama de clases de análisis para el caso de uso Gestionar Usuarios:el diagrama en la figura 15 de la fase de inicio, sufrió un ligero cambio para la inclusión de perfiles de usuario, con el fin de agrupar funcionalidades y asegurar una gestión más simple. El diagrama sigue siendo el mismo solose agrególa nueva clase de entidad Perfiles. En la figura 33se puede apreciar la modificación:

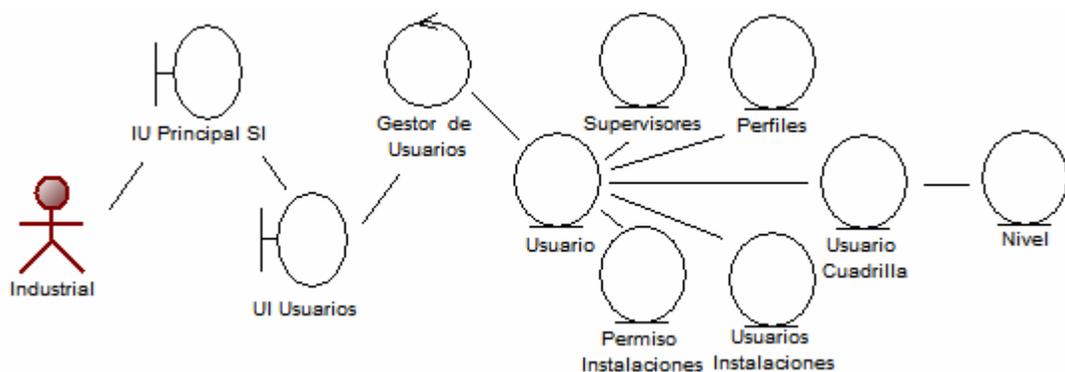


Figura 33. Diagrama de clases de análisis para el caso de uso Gestionar Usuarios de la fase de elaboración.

3.2.3.3 Diagrama de colaboración

En esta fase se explicarán los diagramas de colaboración de los casos de uso que han sido más resaltantes en el flujo de trabajo de requisitos, omitiendo así los diagramas de colaboración cuyo funcionamiento sean semejantes

- Diagrama de colaboración para el caso de uso Gestionar Patrones: en la realización del caso de uso Gestionar Patrones se llevan a cabo las siguientes colaboraciones:

Agregar un patrón:

- 1) El usuario Industrial, en la interfaz: IU Principal SI, por medio de selección de la opción Patrones, enviará la orden para que se ejecute la interfaz: IUPatrones.
- 2) La interfaz: IUPatrones, recibe la orden y envía una petición de consulta a la clase de control Gestor de Patrones con los parámetros de filtrado saeteados por defecto.
- 3) La clase de control: Gestor de Patrones, recibe los datos de filtrado para la consulta.
- 4) La clase de control: Gestor de Patrones, invoca una instancia de la clase de entidad: Patrones.
- 5) La clase de entidad: Patrones, invoca instancias de las clases de entidad: Cuadrillas, Cuadrillas Instalaciones y Usuario Cuadrilla.
- 6) Las clases de entidad: Cuadrillas, Cuadrillas Instalaciones y Usuario Cuadrilla, devuelve los objetos creados.
- 7) La clase de entidad: Patrones, recibe los objetos y con ellos los métodos de cada uno, necesarios para ser devueltos a la clase de control Gestor de Patrones y llevar a cabo cualquier petición del usuario.

- 8) La clase de control: Gestor de Patrones, con los objetos devueltos invoca los métodos de las instancias creadas, que permitan buscar los patrones existentes que coincidan con los de filtrado por defecto existentes en el sistema.
- 9) La clase entidad: Patrones ejecuta los métodos instanciados por los objetos obtenidos de las otras clases entidad.
- 10) Los métodos de las clases de entidad devuelven las peticiones.
- 11) La clase entidad: Patrones devuelve los resultados a la clase de control Gestor de Patrones.
- 12) La clase de control: Gestor de Patrones, devuelve los resultados a la clase de interfaz: UI Patrones, para listar los patrones obtenidos.
- 13) El usuario Industrial provee los datos del personal y describe la rotación de las cuatro cuadrillas a través de tres turnos laborales, mediante la clase interfaz: IU Patrones, que envía la orden y los datos para agregar el nuevo patrón mediante la clase de control: Gestor de Patrones, que capta los datos desde la interfaz.
- 14) La clase de control: Gestor de Patrones hace referencia a la instancia de la clase de entidad: Patrones enviando como parámetros los datos cargados por el usuario.
- 15) La clase de entidad: Patrones, hace referencia a las instancia de clase entidad: Cuadrillas, Cuadrillas Instalaciones y Usuario Cuadrilla, preparando los datos y validaciones para permitir agregar el nuevo patrón. (Véase figura 34).

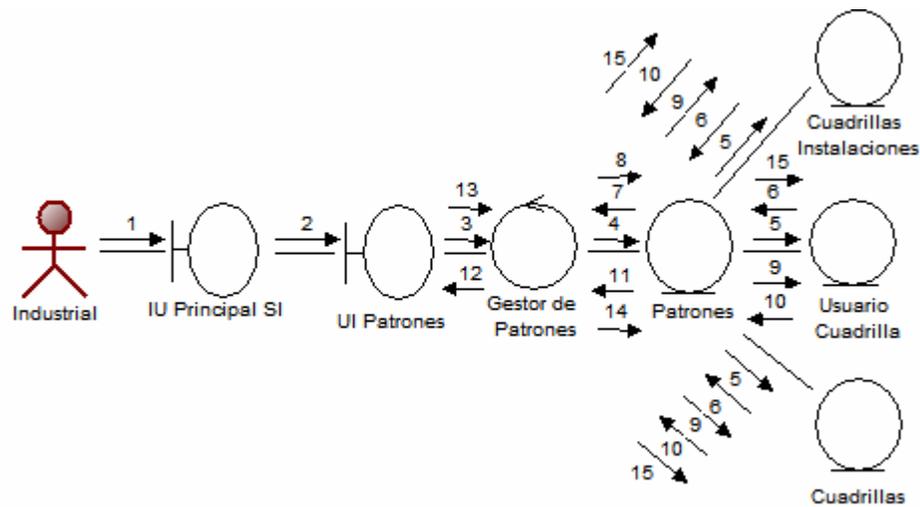


Figura34. Diagrama de colaboración del caso de uso Gestionar Patrones (agregar) de la fase de elaboración.

Modificar un patrón:

- 1) El usuario Industrial, en la interfaz: IU Principal SI, por medio de selección de la opción Patrones, enviará la orden para que se ejecute la interfaz: UI Patrones.
- 2) La interfaz: UI Patrones, recibe la orden y el usuario puede proveer los parámetros de filtrado que serán enviados para la consulta de los patrones programados en los turnos de trabajo de la línea.
- 3) La clase de control: Gestor de Patrones, recibe los datos de filtrado para la consulta.
- 4) La clase de control: Gestor de Patrones, invoca una instancia de la clase de entidad: Patrones.
- 5) La clase entidad Patrones invoca instancias de las clases de entidad: Cuadrillas, Cuadrillas Instalaciones y Usuario Cuadrilla.
- 6) Las clases de entidad: Cuadrillas, Cuadrillas Instalaciones y Usuario Cuadrilla, devuelven los objetos creados.

- 7) Las clases de entidad: Patrones, devuelven los objetos creados a la clase de control.
- 8) La clase de control: Gestor de Patrones, con los objetos devueltos invoca los métodos de las instancias creadas, que permitan buscar las relaciones entre las clases de entidad Cuadrillas, Cuadrillas Instalaciones y Usuario Cuadrilla, donde dichas relaciones definen la programación de los patrones que coincidan con el filtrado realizado por el usuario de Industrial.
- 9) Los métodos devuelven los patrones encontrados.
- 10) La clase de control: Gestor de Patrones, envía a la clase de interfaz: UI Patrones los detalles de las coincidencias para que sean listadas, siendo posible que el usuario modifique alguna de ellas.
- 11) El usuario de Industrial selecciona el patrón deseado y pulsa modificar, para que la clase de control: Gestor de Patrones, capte la petición y capture los nuevos valores e invoque una función que permita realizar la modificación, función que es propia de los objetos que fueron instanciados en los pasos 4 y 5.
- 12) La clase de control: Gestor de Patrones sobrescribe los datos del objeto deseado. (Véase figura 35)

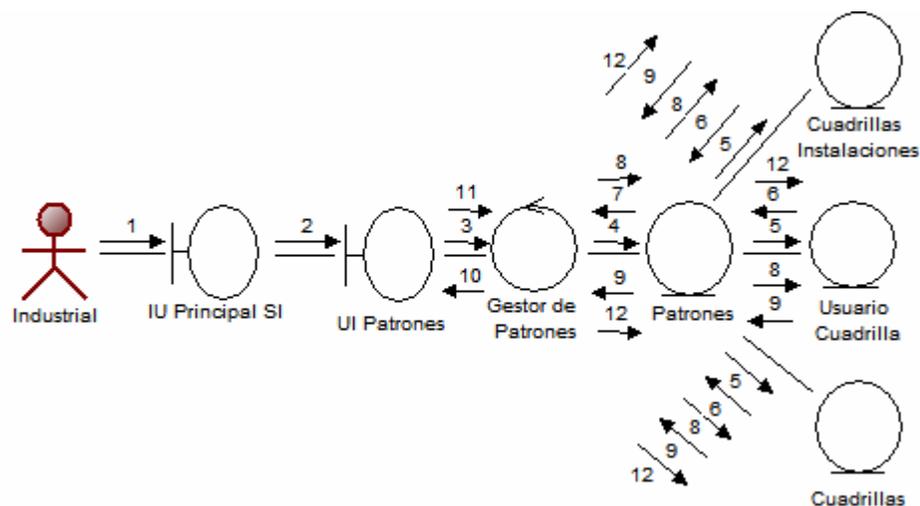


Figura35. Diagrama de colaboración del caso de uso Gestionar Patrones (modificar) de la fase de elaboración.

Eliminar un patrón:

- 1) El usuario Industrial, en la interfaz: IU Principal SI, por medio de selección de la opción Patrones, enviará la orden para que se ejecute la interfaz: UI Patrones.
- 2) La interfaz: UI Patrones, recibe la orden y el usuario puede proveer los parámetros de filtrado que serán enviados para la consulta de los patrones programados en los turnos de trabajo de la línea.
- 3) La clase de control: Gestor de Patrones, recibe los datos de filtrado para la consulta.
- 4) La clase de control: Gestor de Patrones, invoca una instancia de la clase de entidad: Patrones.
- 5) La clase entidad Patrones invoca instancias de las clases de entidad: Cuadrillas, Cuadrillas Instalaciones y Usuario Cuadrilla.
- 6) Las clases de entidad: Cuadrillas, Cuadrillas Instalaciones y Usuario Cuadrilla, devuelven los objetos creados.
- 7) Las clases de entidad: Patrones, devuelven los objetos creados a la clase de control.
- 8) La clase de control: Gestor de Patrones, con los objetos devueltos invoca los métodos de las instancias creadas, que permitan buscar las relaciones entre las clases de entidad Cuadrillas, Cuadrillas Instalaciones y Usuario Cuadrilla, donde dichas relaciones definen los patrones que coinciden con el filtrado.
- 9) Los métodos devuelven los patrones encontrados.
- 10) La clase de control: Gestor de Patrones, envía a la clase de interfaz: UI Patrones los detalles de las coincidencias para que sean listadas.

- 11) El usuario de Industrial selecciona el patrón deseado y pulsa eliminar, para que la clase de control: Gestor de Patrones, capte la petición e invoque una función que permita realizar las validaciones necesarias para la supresión.
- 12) La clase de control: Gestor de Patrones elimina los datos del objeto deseado mientras el patrón no esté en fecha de vigencia. (Véase figura 36)

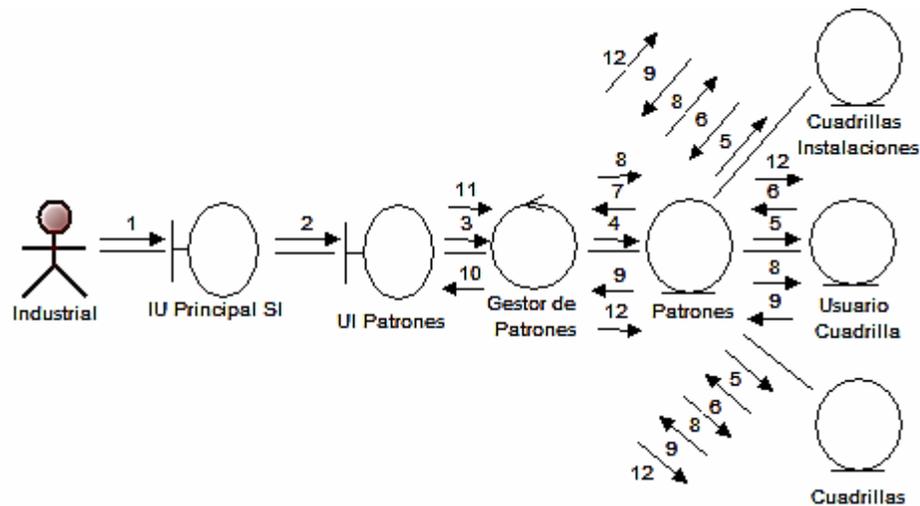


Figura36. Diagrama de colaboración del caso de uso Gestionar Patrones (eliminar) de la fase de elaboración.

- Diagrama de colaboración para el caso de uso Generar Reportes: en la realización del caso de uso Generar Reportes se llevan a cabo las siguientes colaboraciones:
 - 1) El usuario inicia la interacción con el sistema mediante la clase de interfaz: IU Principal SI, que se comunica con la clase interfaz: UI Reportes.
 - 2) La interfaz: UI Reportes, recibe la orden y provee al usuario las opciones que permitan establecer la configuración del reporte solicitado.
 - 3) El usuario establece la configuración, la cual es captada desde la interfaz por la clase de control: Gestor de Configuración de Reportes.

- 4) La clase de control invoca la creación de un objeto: Parámetros de Reportes, el objeto es creado.
- 5) El objeto es devuelto a la clase de control.
- 6) Posteriormente la clase de control realiza un llamado a la función establecer parámetros, función propia del objeto Parámetros de Reportes.
- 7) La clase entidad: Parámetros de Reportes, devuelve el método a la clase de control, para que sea generado el reporte.
- 8) La clase de control: Gestor de Configuración de Reportes, devuelve los datos del reporte a la clase de interfaz: UI Reportes.

A continuación se puede apreciar lo anteriormente descrito mediante el diagrama de colaboración en la figura 37.

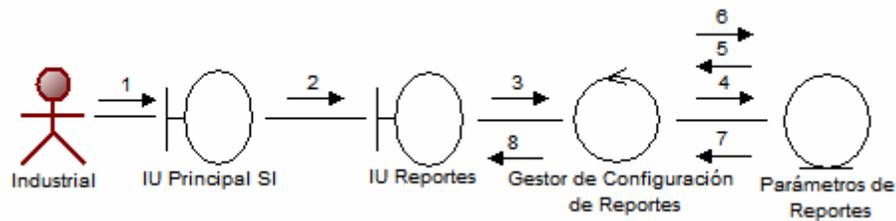


Figura37. Diagrama de colaboración del caso de uso Generar Reportes de la fase de elaboración.

3.2.3.4 Identificación de paquetes de análisis

En la fase de inicio, fueron definidos los paquetes de análisis Catálogos (Figura 24), Productividad (Figura 25), Demoras (Figura 26) y Usuarios (Figura 27); dichos paquetes no han sido modificados y se seguirán considerando para la fase de elaboración.

En las figuras 38 y 39, se presentan los paquetes de análisis identificados a partir de la vista de la arquitectura de los casos de uso, que han sido incluidos para esta fase. Se observa que en estos paquetes se incluyen los procesos relacionados con la

realización de consultas y reportes, manteniendo una relación directa con el caso de uso Generar Reportes.

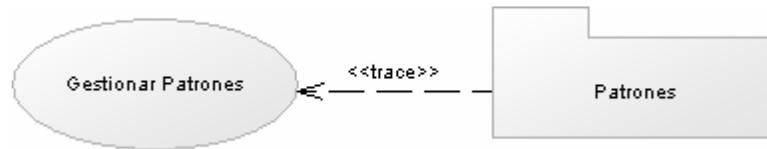


Figura 38. Paquete de análisis de Patrones.

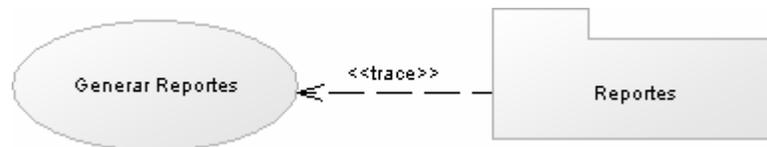


Figura 39. Paquete de análisis Generar Reportes.

3.2.3.5 Diagrama de paquetes de análisis

En este se representan las relaciones de todos los paquetes identificados para el sistema; garantizando como sea posible la independencia de otros paquetes y cumpliendo con el objetivo de realizar algunas clases del dominio o casos de uso. Dicho análisis pretende escribir las dependencias de forma que pueda estimarse el efecto de los cambios futuros. Al relacionar varios paquetes de análisis, se debe definir y mantener las dependencias del paquete con otros paquetes cuyas clases contenidas estén asociadas a él, e intentar hacer cohesivo el paquete incluyendo sólo objetos relacionados funcionalmente. Sin embargo, es necesario limitar la dependencia con otros paquetes, considerando la reubicación de aquellas clases contenidas en paquetes que son demasiado dependientes de otros paquetes. Una clase de un paquete puede aparecer en otro paquete por la importación a través de una relación de dependencia entre paquetes. Todas las clases no son necesariamente visibles desde el exterior del paquete, es decir, un paquete encapsula a la vez que agrupa. En la figura 40, se muestra para la aplicación de gestión del SI de SIDOR, los

paquetes que encapsulan los diferentes casos de uso que fueron definidos al realizar el análisis del sistema, los cuales fueron agrupados y relacionados, obteniéndose de esta manera el diagrama de paquetes de análisis que contiene los requisitos del sistema: Gestionar Catálogos, Gestionar Productividad, Gestionar Usuarios, Clasificar Demoras, Gestionar Patrones y Generar Reportes.

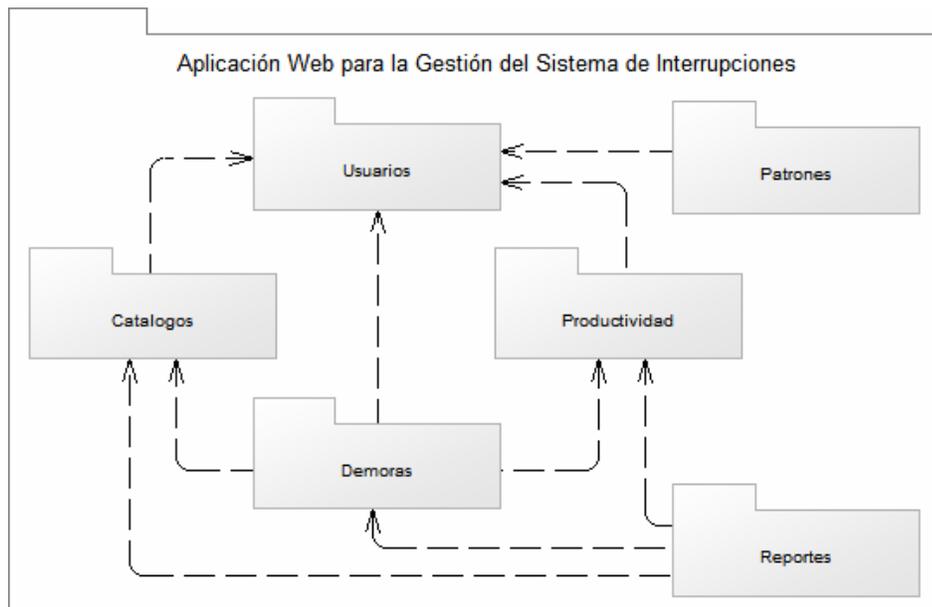


Figura 40. Diagrama de paquetes de análisis del sistema de la fase de elaboración.

3.2.4 Flujo de trabajo (*workflow*) de diseño

El diseño es el resultado del análisis completo del sistema. Mediante éste se adquiere una comprensión detallada de los aspectos relacionados con todos los requisitos, componentes reutilizables, tecnologías de interfaz de usuario, etc. Asimismo, se crea una entrada apropiada para actividades de implementación subsiguientes, capturando los requisitos individuales, interfaces y clases.

El diseño también permite la capacidad de descomponer los trabajos de implementación en partes más manejables, que puedan ser llevadas a cabo por diferentes equipos de desarrollo.

En esta fase se tiene como objetivo obtener el diseño base del software a través del modelo de diseño, para esto se toma como entrada la vista de la arquitectura del modelo de análisis, descrita anteriormente.

3.2.4.1 Diseño de la arquitectura

El diseño de la arquitectura está conformado por tres (3) capas, cada una de ellas formadas por los paquetes del análisis, que representan los subsistemas que se identifican a continuación.

- 1) La capa de aplicación está dividida en una capa específica y una capa general, las cuales están conformadas por los paquetes de análisis del *workflow* de análisis.
- 2) La capa intermedia que está integrada por: el subsistema navegador *Web* que permite cargar las páginas *Web*, incorporando en ellas el código del cliente; el lenguaje de programación *ASP.NET* que es el motor de *script* para desarrollar la aplicación y ejecutarla en el servidor *Web*; *IIS*, que es el servidor que permite la administración de los sitios *Web*.
- 3) La capa de software del sistema, representada por: el subsistema *TCP/IP*, que es el protocolo mediante el cual los datos se envían por la red que conecta al usuario con el servidor. El sistema operativo, que es el software encargado de coordinar, administrar y gestionar todos los recursos del sistema, para así dar soporte a las actividades realizadas por los demás subsistemas. El sistema manejador de base de datos *Microsoft SQL Server 2000*, que es el encargado de

manejar o atender de forma transparente las solicitudes realizadas sobre la base de datos.

En la figura 41, se muestra la dependencia o relación entre las capas anteriormente descritas.

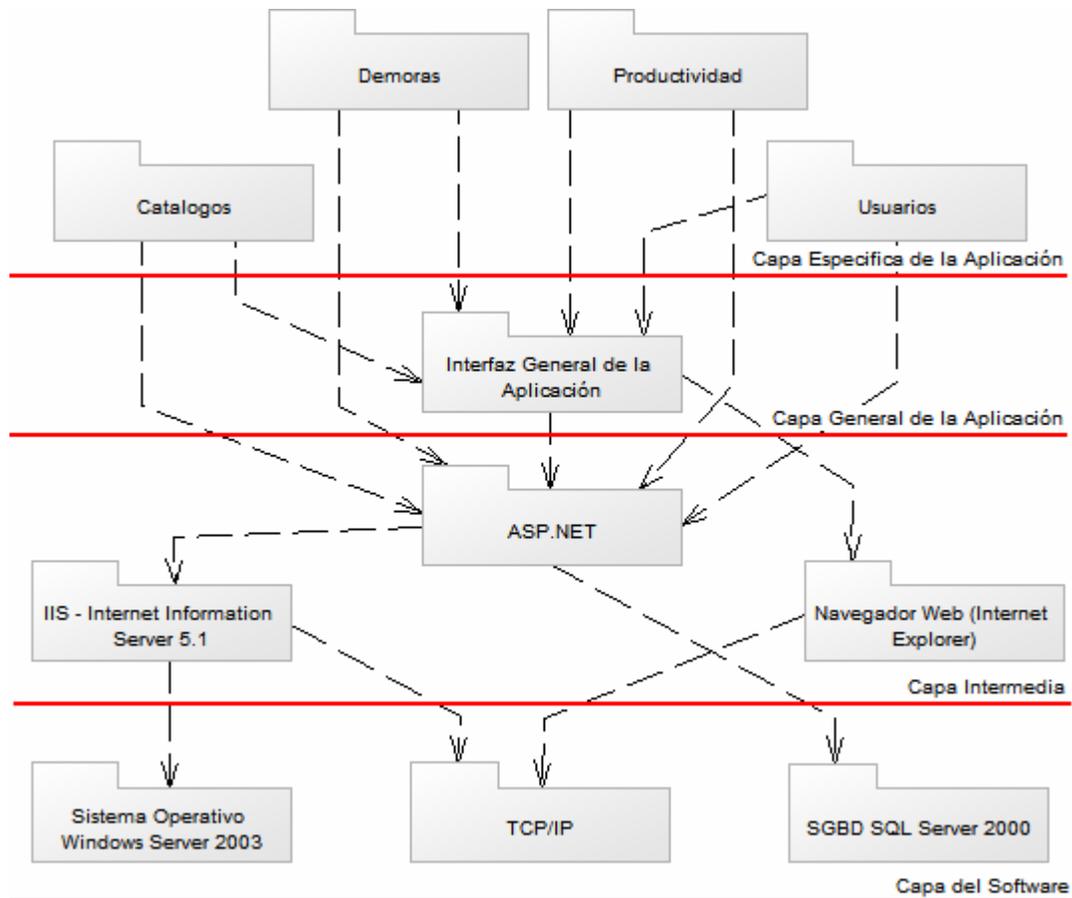


Figura 41. Capas de la arquitectura del sistema.

3.2.4.2 Modelo de diseño

El modelo de diseño es un modelo de objetos que describe la realización física de los casos de usos, centrándose en cómo los requisitos funcionales y no funcionales tienen impacto en el sistema en estudio. Las clases del diseño se identifican a partir de las clases del análisis interfaz (tablas 12 y 21), control (tablas 13 y 22) y entidad (tablas

14 y 23) de las realizaciones de los casos de uso más relevantes para la arquitectura.
En las figuras 42, 43 y 44, se muestran las clases del diseño identificadas:

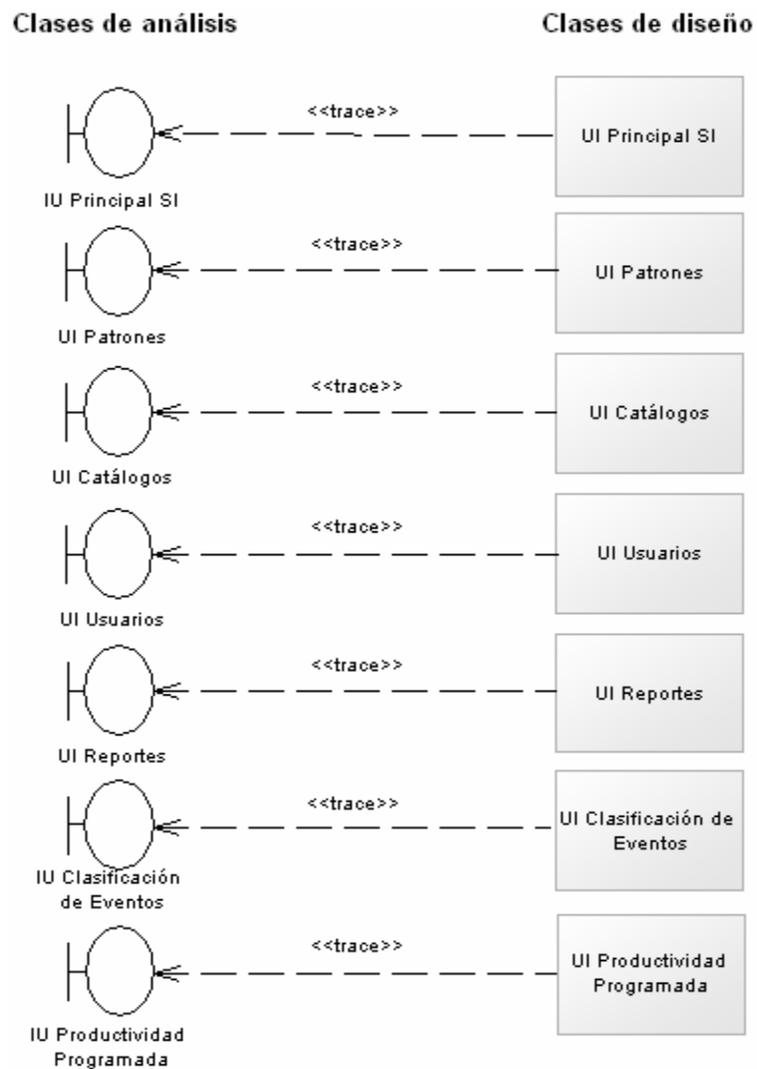


Figura 42. Clases de diseño a partir de clases de interfaz.

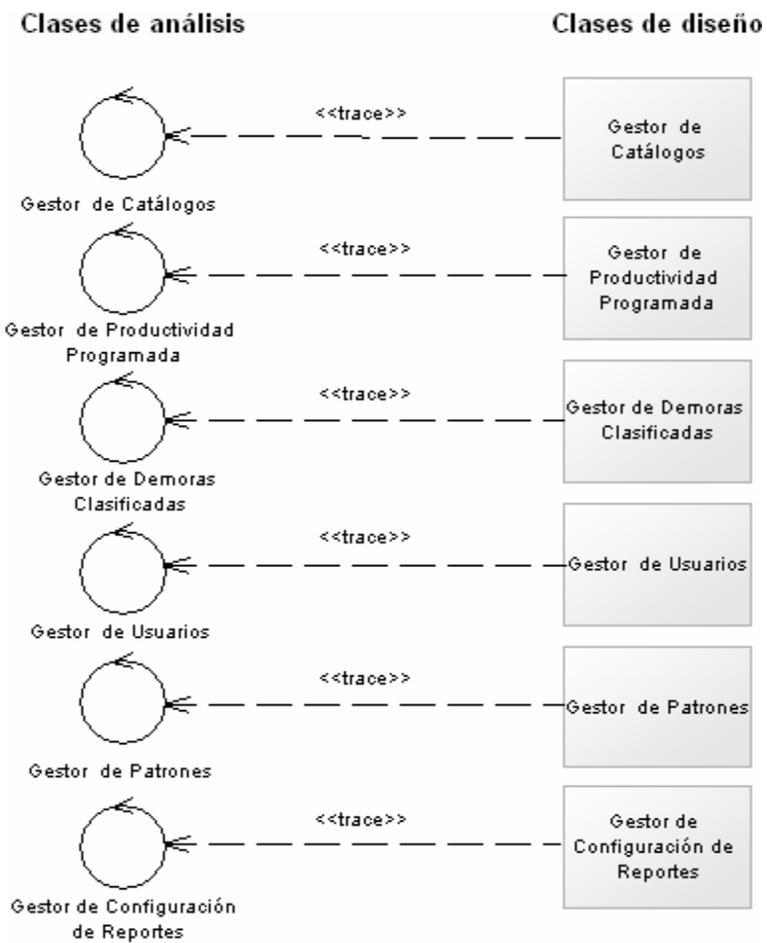


Figura 43. Clases de diseño a partir de clases de control.

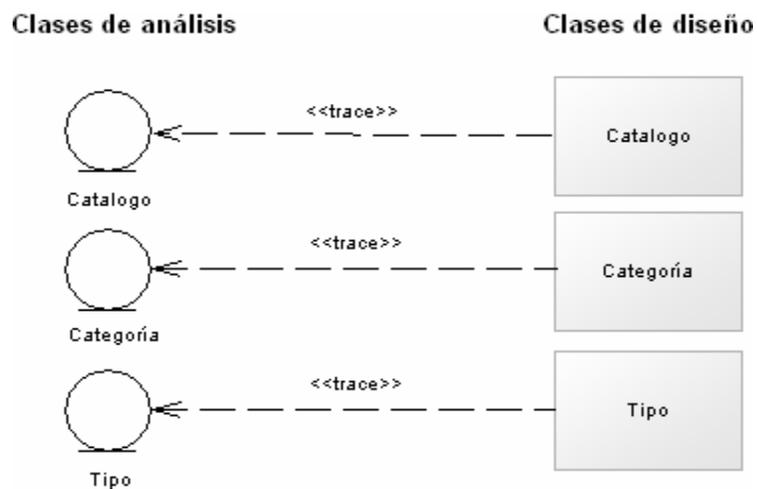


Figura 44. Clases de diseño a partir de clases de entidad.

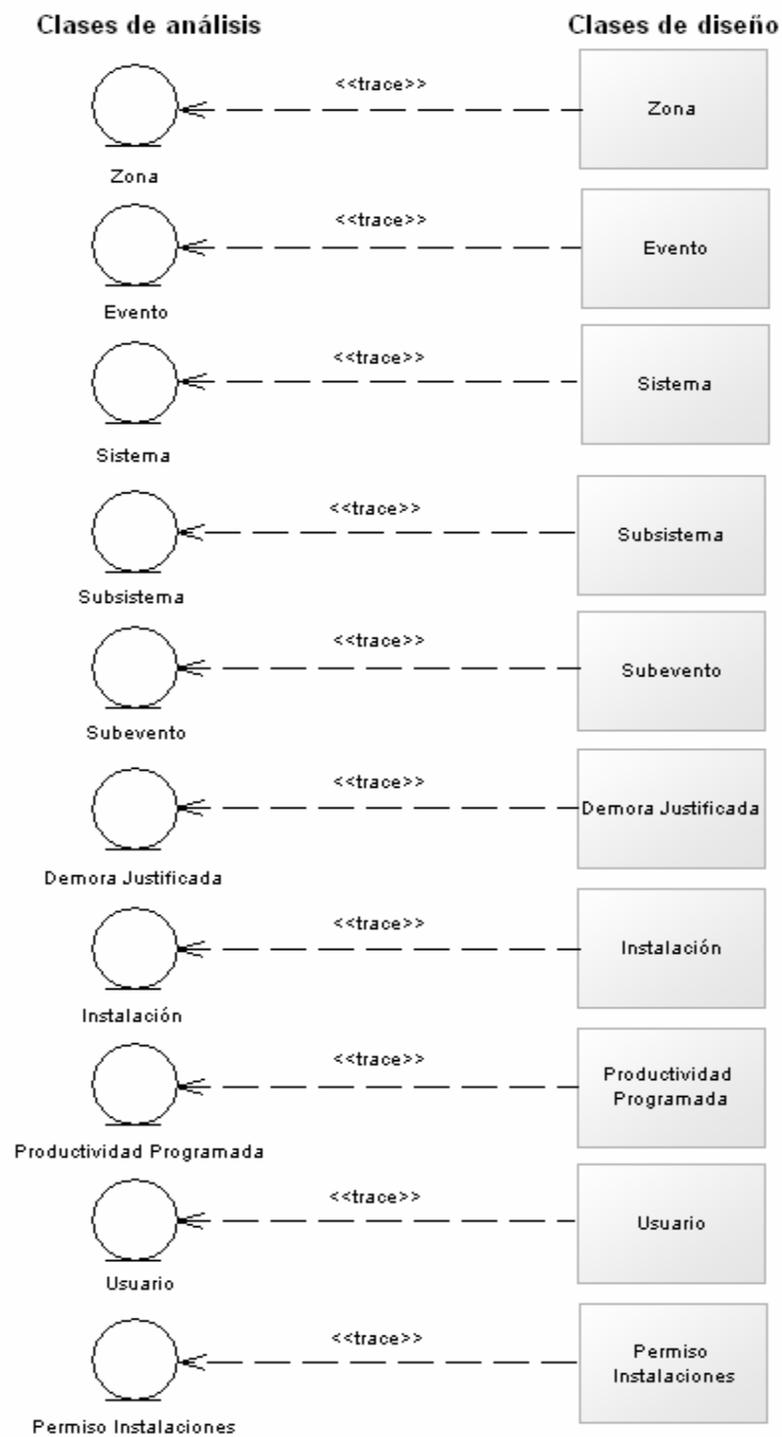


Figura 44. Continuación

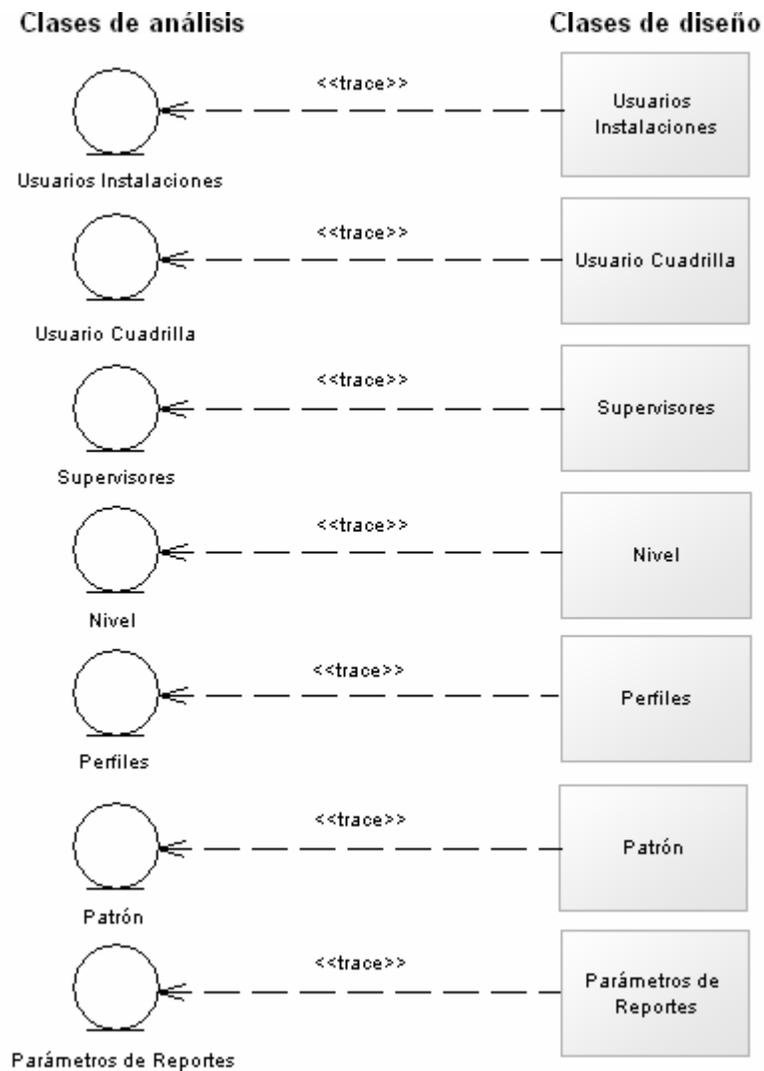


Figura 44. Continuación

3.2.4.3 Diagramas de clases del diseño

Este diagrama permite visualizar las clases que componen el sistema, y la estructura estática de los casos de uso, reflejando las relaciones de generalización, agregación y composición entre clases; y nos muestra lo que el sistema puede hacer y cómo puede ser construido. A continuación, en la figura 45 se presentan las clases críticas detectadas para el desarrollo y construcción de la aplicación.



Figura 45. Clases detectadas

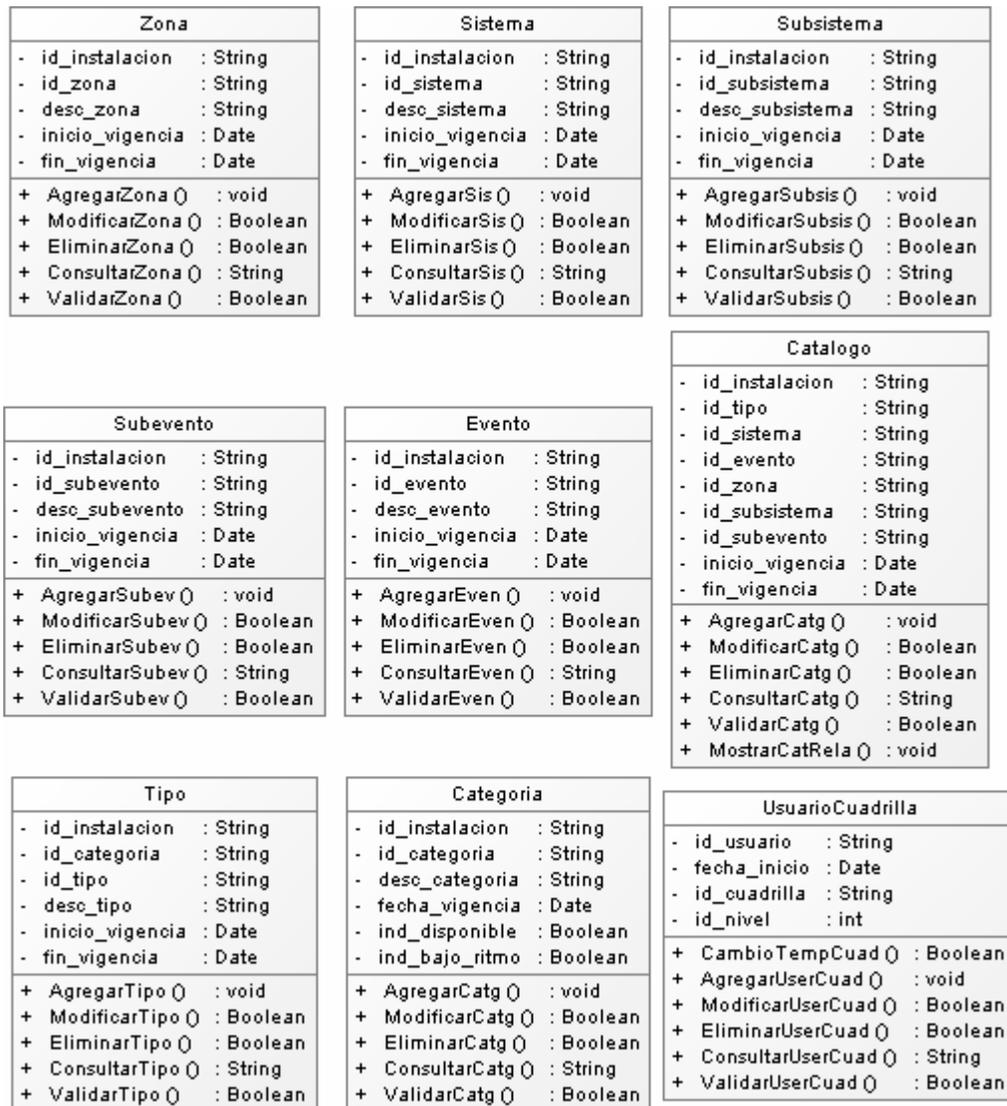


Figura 45. Continuación

Para la realización del diagrama de clases de diseño del sistema, se tomó como base las clases de análisis del modelo de análisis, además de nuevas clases que no habían sido consideradas; todas estas clases fueron modeladas como se muestra en las figuras anteriores, para así representar completamente la estructura del software. A Continuación en la Figura 46 se muestra el diagrama de clases de diseño para la Aplicación Web para la Gestión del Sistema de Interrupciones de SIDOR.

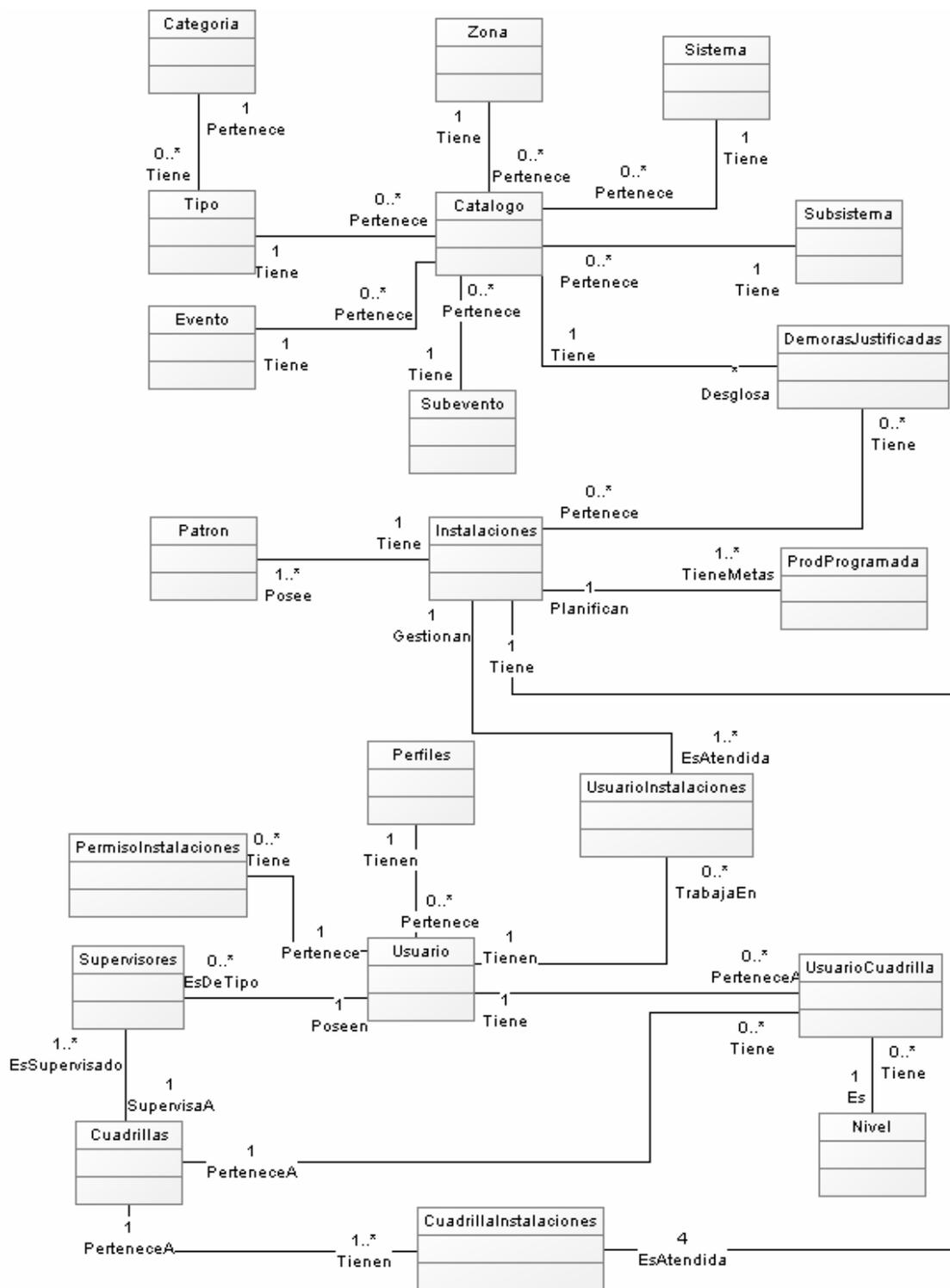


Figura 46. Diagrama de clases

3.2.4.4 Diagrama de secuencia

Son diagramas dinámicos que representan de forma jerárquica el orden y el tiempo en el cual se activan los objetos mediante los mensajes. En el apéndice A, se muestran los diagramas de secuencias correspondientes a los casos de uso críticos en la aplicación.

3.2.4.5 Diseño de la base de datos local del sistema

Para el diseño de la base de datos se creó un esquema conceptual mediante un modelo de datos de alto nivel. Este esquema es una descripción concisa de la información según los requerimientos previamente estudiados. Se determinaron las posibles estructuras de las tablas que requiere el sistema, tipos de datos, vínculos y restricciones. En el apéndice B, se muestra el diseño físico de la base de datos.

3.2.5 Flujo de trabajo (*workflow*) de implementación

En este flujo de trabajo se implementaron los archivos de código fuente, ficheros de código binario, *scripts* y ejecutables a partir de los elementos de diseño más importantes para la arquitectura. Las páginas se realizaron haciendo uso del lenguaje *ASP.NET* y los *scripts* ejecutados en el cliente mediante *JavaScript*.

3.2.5.1 Códigos ejecutables para realizar los casos de uso

- Interfaz gráfica y formulario dinámico de carga: en la figura 47, se muestra uno de los formularios de carga de los distintos catálogos en este caso el de Tipos.
- Caso de uso relacionado: Gestión de Catálogos.
- Nombre de archivos involucrados: `gestion_catalogo_dataentry.aspx`, `gestion_catalogo_dataentry.js`, `gestion_catalogo.aspx`.
- Código fuente: apéndice C.

Figura 47. Formulario Dinámico de carga de Catálogos.

3.2.6 Flujo de trabajo (*workflow*) de pruebas

El objetivo principal de llevar a cabo un flujo de trabajo de pruebas es asegurarse de que las capas específicas de la aplicación funcionen y no presenten ningún problema.

3.2.6.1 Partición equivalente

Una partición equivalente es un método de prueba de caja negra que divide el dominio de entrada de datos de un programa en clases de datos.

- Identificación de las clases de equivalencia: las clases de equivalencia a estudiar, se encuentran divididas en las siguientes clases de datos:
 - 1) Sólo números.
 - 2) Sólo caracteres.
 - 3) Caracteres y números.
 - 4) Ningún carácter.
- Grupo de tipos de entrada de datos:

Código (1): identificación única del Tipo de evento.

Fecha Inicio (2): fecha inicio de vigencia del catálogo gestionado.

Fecha Fin (3): fecha fin de vigencia del catálogo gestionado.

Líneas (4): líneas múltiples a las que se añadirá la gestión.

Categorías (5): categoría de la cual pertenece el evento que esta en gestión.

Descripción (6): cualquier información que describa de manera clara el evento que se está gestionando y que especifique de que se trata.

Usuario (7): usuario del sistema encargado de registrar el catálogo así como mantener y ejecutar modificaciones sobre el catálogo.

Planta (8): planta a la que pertenece el evento que se describe.

Instalación (9): instalación donde se ubica exactamente el evento.

Indicador de Status *DatawareHouse*(10): indicador de actividad del registro en la base de datos.

3.2.6.2 Aplicación de casos de prueba

En la tabla 24, se detallan los casos de pruebas aplicados a los grupos seleccionados.

Tabla 24. Aplicación de casos de prueba de la fase de elaboración.

Grupo	Caso de prueba	Válido	No válido	Clase de equivalencia
1	1345980		x	1
1	ABcd/%#		x	2
1	66AMQ50er3	x		3
1	“ ”		x	4
2	1345980		x	1
2	ABcd/%#		x	2
2	23/11/2010	x		3
2	“ ”	x		4
3	1345980		x	1

Tabla 24. Continuación.

Grupo	Caso de prueba	Válido	No válido	Clase de equivalencia
3	ABcd/%#		x	2
3	23/11/2010	x		3
3	“ ”	x		4
4	1345980		x	1
4	ABcd/%#		x	2
4	66AMQ50er3	x		3
4	“ ”		x	4
5	1345980		x	1
5	ABcd/%#		x	2
5	66AMQ50er3	x		3
5	“ ”		x	4
6	1345980	x		1
6	ABcd/%#	x		2
6	66AMQ50er3	x		3
6	“ ”		x	4
7	1345980		x	1
7	ABcd/%#		x	2
7	66AMQ50er3	x		3
7	“ ”		x	4
8	1345980	x		1
8	ABcd/%#	x		2
8	66AMQ50er3	x		3
8	“ ”		x	4
9	1345980	x		1
9	ABcd/%#	x		2
9	66AMQ50er3	x		3

Tabla 24. Continuación.

Grupo	Caso de prueba	Válido	No válido	Clase de equivalencia
9	“ ”		x	4
10	1	x		1
10	ABcd/%#		x	2
10	66AMQ50er3		x	3
10	“ ”		x	4

3.2.7 Evaluación de la fase de elaboración

Al finalizar esta fase, el resultado principal es una arquitectura estable para guiar el sistema a lo largo de su vida, comprobándose que efectivamente la línea base desarrollada representa una arquitectura capaz de llevar a cabo los objetivos iniciales del proyecto y mitigar los riesgos. Se considera que fueron determinados la mayoría de los requisitos mediante el modelo de casos de uso.

Con el fin de visualizar la estructura estática del sistema y las clases que lo componen, se realizó el diagrama de clases de diseño, mostrando las relaciones de dichas clases. Y los diagramas de colaboración, así como los diagramas de secuencia, indicando la sucesión de acciones que se llevan a cabo y las interacciones reales entre las clases y los objetos del sistema.

Una vez obtenidas las vistas arquitectónicas de los modelos de caso de uso, análisis y diseño; se realizó el modelo físico de la base de datos local del sistema, describiéndose las tablas correspondientes así como sus atributos.

Se presentó, el código ejecutable responsable de realizar los casos de usos considerados para esta fase. Como producto final de la fase de elaboración se obtuvo la línea base de la arquitectura de la Aplicación *Web* para la Gestión del Sistema de Interrupciones de la Siderúrgica del Orinoco Alfredo Maneiro.

En la tabla 25, se presenta el estatus de todos los artefactos que fueron planificados para esta fase.

Tabla 25. Estatus de los artefactos planificados en la fase de elaboración.

Artefactos	Estatus
Requisitos funcionales y no funcionales	Culminado
Requisitos de software y hardware	Culminado
Modelo de casos de uso	Culminado
Descripción de casos de uso	Culminado
Diagrama de clases de análisis	Culminado
Diagrama de colaboración	Culminado
Identificación de paquetes de análisis	Culminado
Diseño de la arquitectura	Culminado
Diagrama de clases de diseño	Culminado
Diagrama de secuencia	Culminado
Diseño físico de la base de datos	Culminado
Código fuente para la realización de los casos de uso	En desarrollo
Aplicación de casos de pruebas	En desarrollo

3.3 FASE DE CONSTRUCCIÓN

En la fase anterior se establecieron los fundamentos de los elementos importantes en cuanto a arquitectura se refiere, así como también los casos de uso significativos.

En la fase de construcción se realizó la codificación de las páginas *Web* que se diseñaron en el modelo de navegación y gestión de contenidos.

3.3.1 Planificación de la fase de construcción

Cumpliendo con el ciclo iterativo del Proceso Unificado de Desarrollo de Software, en esta fase se desarrollaron los flujos de trabajos de implementación y pruebas, tomando como guía los modelos construidos en la fase de elaboración.

Los artefactos desarrollados se describen a continuación en la tabla 26:

Tabla 26. Actividades y artefactos planificados para la fase de construcción.

Actividad	Artefacto
Implementación	Código fuente para la realización de los casos de uso.
Pruebas	Casos de pruebas a casos de usos.

3.3.2 Flujo de trabajo (*workflow*) de implementación

En este flujo de trabajo se implementó los archivos de código fuente, ficheros de código binario, scripts y ejecutables correspondientes a los modelos de navegación y gestión de contenidos que no fueron implementados en la fase anterior con la finalidad de obtener la versión beta de la aplicación *Web* para la gestión del SI.

Con la finalidad de mostrar un poco el código fuente que se usó para la construcción de la aplicación se mostrará un fragmento perteneciente al formulario de carga de datos de productividad programada en el Apéndice C.

3.3.2.1 Códigos ejecutables para realizar los casos de uso

- Interfaz gráfica y formulario de carga de justificación de demoras: a continuación, en la figura 48, se muestra el formulario de carga de datos de justificación de demoras.

DATOS DE LA INTERRUPCIÓN

Bobina: 0 Inicio:07/10/2008 07:00:00 Fin:07/10/2008 08:51:01 Durac. total:01:51:01 Cambio bob.:No

PARTICIÓN 1

Tipo: [dropdown]

Zona: [dropdown]

Sistema: [dropdown]

Subsistema: [dropdown]

Evento: [dropdown]

Subevento: [dropdown]

Duración: [hh:mm:ss] 01:51:01

Justificación: [text area]

[Crear Partición] [Aceptar] [Cancelar]

Figura 48. Formulario de carga de datos de demoras.

Interfaz gráfica y formulario de carga de justificación de demoras.

Caso de uso relacionado: Clasificar Demoras.

Nombre de algunos archivos involucrados: Clasificacion_Demora.aspx, Principal_Eventos.aspx

- Interfaz gráfica y formulario de carga de productividad programada: en la figura 49, se muestra el formulario de carga de datos referentes a las metas de producción programadas por el personal de Industrial para las líneas en que laboran.

Agregar Productividad Programada

Agregar Productividad	
Inicio Vigencia	[calendar icon]
Turno	1 [dropdown arrow]
Prod. Program.	[text input]
Tiempo Efectivo	[text input]
Efectividad	[text input]

[Guardar] [Cerrar]

Figura 49. Formulario de carga de datos de productividad programada.

Interfaz gráfica y formulario de carga de productividad programada.

Caso de uso relacionado: Gestionar Productividad Programada.

Nombre de archivos involucrados: produccion_programada_ABM.aspx, produccion_programada.aspx.

- Interfaz gráfica y formulario de carga de patrones: a continuación en la figura 50, se observar la interfaz del formulario de carga de datos relacionado a la rotación de cuadrillas por turnos establecidos en la empresa para mantener sus niveles de producción, cuyo formulario pertenece al caso de uso Gestión de Patrones.

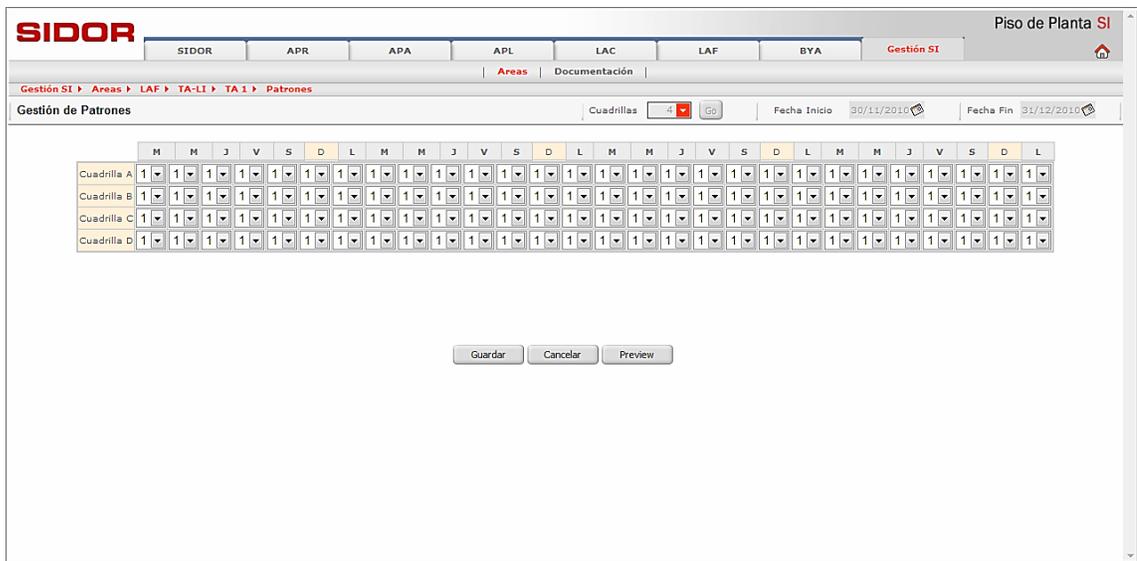


Figura 50. Formulario de carga de datos de Patrones.

Interfaz gráfica y formulario de carga de Patrones.

Caso de uso relacionado: Gestionar Patrones.

Nombre de algunos archivos involucrados: gestión_patrones_ABM.aspx, gestión_patrones.aspx.

3.3.2.2 Documentación del sistema

Incluye la realización de un manual lo suficientemente detallado como para guiar a los usuarios en el uso de la versión beta de la aplicación. Los casos de uso también se utilizaron como punto de partida para escribir este manual de usuario. Ya que cada caso de uso poseen la lógica ideal que describe la forma de interacción entre el usuario y la aplicación. En el apéndice D, se puede detallar dicho manual.

3.3.3 Flujo de trabajo (*workflow*) de pruebas

En este flujo se prueban los escenarios de todos los casos de uso restantes de la aplicación, incluyendo la entrada y las condiciones bajo las cuales han de probarse. A continuación se mostraran algunas de estas pruebas:

3.3.3.1 Partición equivalente

En esta fase se llevó a cabo el método de prueba de caja negra de particiones equivalentes para todos aquellos campos de entrada y clases de equivalencias que no fueron llevadas a cabo en la fase anterior por no disponer de un componente codificado para probar.

- La identificación de las clases de equivalencia: las clases de equivalencia a estudiar, se encuentran divididas en las siguientes clases de datos:
 - 1) Sólo números.
 - 2) Sólo caracteres.
 - 3) Caracteres y números.
 - 4) Ningún carácter.

- Grupo de tipos de entrada de datos para la interfaz de clasificación de demoras:

Inicio (1): fecha y hora de inicio del evento.

Fin (2): fecha y hora de fin del evento.

Duración total (3): tiempo de duración del evento.

Tipo (4): tipos con los que se pueden clasificar los registros.

Zona (5): son respectivas zonas de la planta.

Sistema (6): comprende todos los equipos de la línea de producción.

Subsistema (7): es cada una de las piezas que componen al equipo que este fallando.

Evento (8): explicación sobre el suceso que ocurrió en la planta.

Subevento (9): información más detallada del evento que se generó en la planta.

Justificación (10): acotación adicional que sea importante y que no se ve claramente reflejada en la selección de los códigos anteriores.

Partición (11): cuando las causas de la demora sean más de una, será necesario particionar la demora para crear un nuevo conjunto de campos con el fin asignar la codificación correspondiente a la segunda causa de la demora. Las demoras poseen un número que indica la cantidad de particiones de la misma.

3.3.3.2 Aplicación de casos de prueba

En la tabla 27, se detallan los casos de pruebas aplicados a los grupos seleccionados.

Tabla 27. Aplicación de casos de prueba de la fase de construcción.

Grupo	Caso de prueba	Válido	No válido	Clase de equivalencia
1	1345980		x	1
1	ABcd/%#		x	2
1	23/11/2010 02:50:34 p.m.	x		3
1	“ ”		x	4
2	1345980		x	1
2	ABcd/%#		x	2

Tabla 27. Continuación.

Grupo	Caso de prueba	Válido	No válido	Clase de equivalencia
2	23/11/2010 02:52:12 p.m.	x		3
2	“ ”		x	4
3	1345980		x	1
3	ABcd/%#		x	2
3	00:02:38	x		3
3	“ ”		x	4
4	1345980		x	1
4	ABcd/%#		x	2
4	abCD01	x		3
4	“ ”		x	4
5	1345980		x	1
5	ABcd/%#		x	2
5	abCD01	x		3
5	“ ”		x	4
6	1345980		x	1
6	ABcd/%#		x	2
6	abCD01	x		3
6	“ ”		x	4
7	1345980		x	1
7	ABcd/%#		x	2
7	abCD01	x		3
7	“ ”		x	4
8	1345980		x	1
8	ABcd/%#		x	2
8	abCD01	x		3
8	“ ”		x	4

Tabla 27. Continuación.

Grupo	Caso de prueba	Válido	No válido	Clase de equivalencia
9	1345980		x	1
9	ABcd/%#		x	2
9	abCD01	x		3
9	“ ”		x	4
10	1345980		x	1
10	ABcd/%#		x	2
10	abCD01	x		3
10	“ ”		x	4
11	1	x		1
11	ABcd/%#		x	2
11	abCD01		x	3
11	“ ”		x	4

3.3.3.3 Casos de prueba basados en casos de uso

A continuación se mostrarán las entradas de datos válidas para la correcta ejecución de los casos de usos estudiados en esta fase de desarrollo.

- Identificación de casos de prueba:

Caso de prueba para el caso de uso Clasificar Demoras.

Entrada:

Inicio: 31/03/2008 09:40:56 a.m.

Fin: 31/03/2008 14:24:18 p.m.

Duración total:02:23:22

Tipo: Mecánica

Zona: Sistema de Colección de Polvos Línea A

Sistema: Cinta Transporte Polvo AH1011

Subsistema: Cinta Transporte Polvo AH1011

Evento: Daño en Acople

Subevento: Daño en Acople

Justificación: Por Contaminación ambiental. Sistema del AH-1011 parado.

Partición: 3

Resultado: los datos son validados e ingresados en la base de datos local.

Condiciones: el usuario debe estar autenticado como operador del sistema y debe tener permisos en la instalación donde esta realizando la clasificación, así como también pertenecer a la cuadrilla que se encuentra de turno para el momento de desglosar la demora.

Procedimiento: presionar botón aceptar para justificarla demora.

3.3.3.4 Pruebas de integración

En esta fase se procedió de la misma manera que en la fase anterior. Luego de realizar las pruebas por separado de las páginas que conforman los casos de uso restantes, se procedió a la integración de cada una de las páginas para probar su forma de interacción del sistema en su totalidad, solventándose las fallas presentadas. En el apéndice E, se muestra la aplicación de esta prueba para la aplicación.

3.3.4 Evaluación de la fase de construcción

La fase de construcción fue ejecutada de manera satisfactoria, abarcando sólo los flujos de trabajo de implementación y pruebas. Durante la implementación se realizó la codificación efectiva de la aplicación *Web* y los distintos componentes que conforman el software. En el flujo de trabajo de prueba fueron validadas y solventadas las fallas que surgieron, con el método de particiones equivalentes. Este proceso tuvo como resultado la culminación de los artefactos planificados para el desarrollo de la aplicación, obteniéndose finalmente la una versión beta.

CONCLUSIONES

Luego de aplicar el Proceso Unificado de Desarrollo de Software, para la realización de la aplicación *Web* de Gestión del SI, se puede decir que éste proporcionó un soporte teórico robusto y bien definido, que facilitó y garantizó su consolidación.

Debido a que el Proceso Unificado de Desarrollo de Software es una metodología iterativa, fue posible avanzar en base a retroalimentaciones de cada una de las iteraciones, ayudando al refinamiento progresivo e incremental del sistema, prueba de ello fue la identificación de nuevos requerimientos en la fase de elaboración que inicialmente fueron pasados por alto en la fase anterior.

Desde el punto de vista de construcción y mantenimiento de sistemas, el haber utilizado UML, fue de gran ayuda, ya que los modelos y diagramas obtenidos durante las etapas de Análisis y Diseño ayudaron al entendimiento claro y conciso de lo que se quería que hiciera la aplicación y cómo hacerlo. Además de ello, UML facilita la ampliación y mejora de la aplicación ya que dejó una arquitectura bien definida, puesto que es relativamente sencillo comprender su funcionamiento.

La implementación de un modelo de dominio y modelo de diseño en el desarrollo de un proyecto de esta índole, orientó el diseño de la base de datos de la aplicación *Web* facilitando su construcción, ya que pudieron transformarse las clases obtenidas del diagrama de clases de diseño en tablas del modelo físico de la base de datos.

El uso de Diagramas de Secuencia para el desarrollo de la aplicación, permitió definir su funcionamiento desde un punto de vista dinámico, ya que este muestra la interacción entre el usuario, las clases y la *Datawarehouse* del SI, modelando su comportamiento en relación al tiempo y no estáticamente como se hace por ejemplo en el diagrama de clases.

Las características del lenguaje de programación *Visual Basic.NET* de la mano de *ASP.NET*, hizo que la construcción de la aplicación de Gestión del SI, fuera sencilla y clara, ya que por ser un lenguaje orientado a objetos, la implementación de los diseños se hizo más cónsona con la metodología aplicada. Además de esto, el entorno gráfico para la implementación de pantallas *Web* es bastante elegante, lo que permitió la creación de interfaces claras y bien entendibles.

En el proceso de codificación de las interfaces usuario, se mejoró el uso de los estilos y colores que se encontraban ya definidos por SIDOR, haciendo el mejor uso de ellos, en la manera como se muestra la información y las diversas acciones que la aplicación permite realizar a los usuarios, como por ejemplo, se mejoró la manera en que se presentan las tablas de datos, la navegación por encabezados de menús y las opciones de filtros a los que ya se estaba acostumbrado para las aplicaciones en la empresa, además de que se estructuraron las pantallas de manera que se localizara de forma rápida y casi intuitiva la información, reduciendo así el tiempo en su búsqueda.

RECOMENDACIONES

Realizar una revisión de los desarrollos hechos utilizando el Proceso Unificado de Desarrollo, con el fin de crear un catálogo de sugerencias que permitan al estudiante hacer un mejor uso de la metodología.

En cuanto a la pasantía la UDO debe solicitar a la empresa permitir dedicar más tiempo a la fase de inicio, para así poder obtener la mayor información posible de los usuarios finales sobre el sistema que se quiere implantar.

Crear un plan de adiestramiento con el fin de capacitar al personal que manejará la aplicación, a modo de asegurar el buen uso e integridad de la información que esté contenida.

Establecer un plan de mantenimiento para la aplicación, con la finalidad de precisar posibles mejoras.

Tomar en cuenta el tiempo de capacitación en las herramientas de desarrollo, en los casos que sea necesario.

Incorporar a la aplicación la opción de visualizar gráficos estadísticos de las principales fallas en las líneas de producción.

BIBLIOGRAFÍA

1. Kendall, K. y Kendall, J. 1997. *Análisis y diseño de sistemas*. Tercera edición. Prentice Hall. España.
2. Stallings, W. 2000. *Comunicaciones y redes de computadoras*. Sexta edición. Prentice Hall. España.
3. Pressman, R. 2002. *Ingeniería del Software. Un enfoque práctico*. Quinta edición. McGraw-Hill/Interamericana Editores S.A. de C.V., México.
4. Sidor. 2009. *Intranet de la Siderúrgica del Orinoco Alfredo Maneiro*. "Sidornet" <<http://sidornet>>(10/11/2008).
5. Vergara, G. 2006. *Diseño e Implementación bajo Plataforma Web del Sistema de Interrupciones (SI) para la Planta de Briquetas (MATESI) de SIDOR C.A.* Trabajo de pregrado. Escuela de Ingeniería de Sistemas, Universidad de los Andes, Mérida.
6. Sierraalta, A. 2005. *Sistema Web para la Gestión de la Calidad en los Procesos de Laminación de SIDOR*. Trabajo de pregrado. Escuela de Ingeniería de Sistemas, Universidad de los Andes, Mérida.
7. Ruiz, L. 2005. *Sistema de Información para la Gestión de Materiales Refractarios de SIDOR C.A.* Trabajo de pregrado. Escuela de Ingeniería de Sistemas, Universidad de los Andes, Mérida.
8. Rincón, Y. 2005. *Sistemas de Gestión en Tornería de Cilindros de Laminación en Frío de la Empresa SIDOR*. Trabajo de pregrado. Escuela de Ingeniería de Sistemas, Universidad de los Andes, Mérida.
9. Whitten, J., Bentley, L. y Barlow, V. 1996. *Análisis y diseño de sistemas de información*. Tercera edición. McGraw-Hill. España.
10. Senn, J. 1992. *Análisis y diseño de sistemas de información*. Segunda edición. McGraw Hill. México.
11. Larman, C. 2003. *UML y patrones*. Segunda edición. Pearson Educación, S.A. España.
12. Moreno, A. 2000. *Bases de datos: Modelo de datos*. "elies.rediris". <<http://elies.rediris.es/elies9/4-2.htm>> (15/10/2010).

13. Elmasri, R. y Navathe, S. 1997. *Sistemas de bases de datos*. Segunda edición. Addison Wesley. México.
14. Schmuller, J. (2003). *Aprendiendo UML en 24 Horas*. Segunda edición. Prentice Hall Hispanoamericana S. A. México.
15. Mateu, C. 2004. *Desarrollo de aplicaciones Web*. Ediciones Eureka Media, SL. España.
16. MSDN. 2010. *Arquitectura de la tecnología .NET*. “msdn.microsoft”. <<http://msdn.microsoft.com>>(15/10/2010).
17. Silberschatz, A., Korth, H. y Sudarshan, S. 2006. *Fundamentos de Bases de Datos*. Quinta edición. McGraw Hill. México.
18. Fazio, E. 2005. *Microsoft SQL Server 2000*. “Microsoft”. <<http://www.microsoft.com/argentina/educacion/universidades/workshop4.asp>> (15/10/2010).
19. Tamayo y Tamayo, M. 2001. *El proceso de la investigación científica*. Cuarta edición. Limusa. México.
20. Jacobson, I., Booch, G. y Rumbaugh, J. 1999. *El Proceso Unificado de Desarrollo de Software*. Pearson – Addison Wesley. España.

APÉNDICES

Apéndice A. Diagramas de secuencia

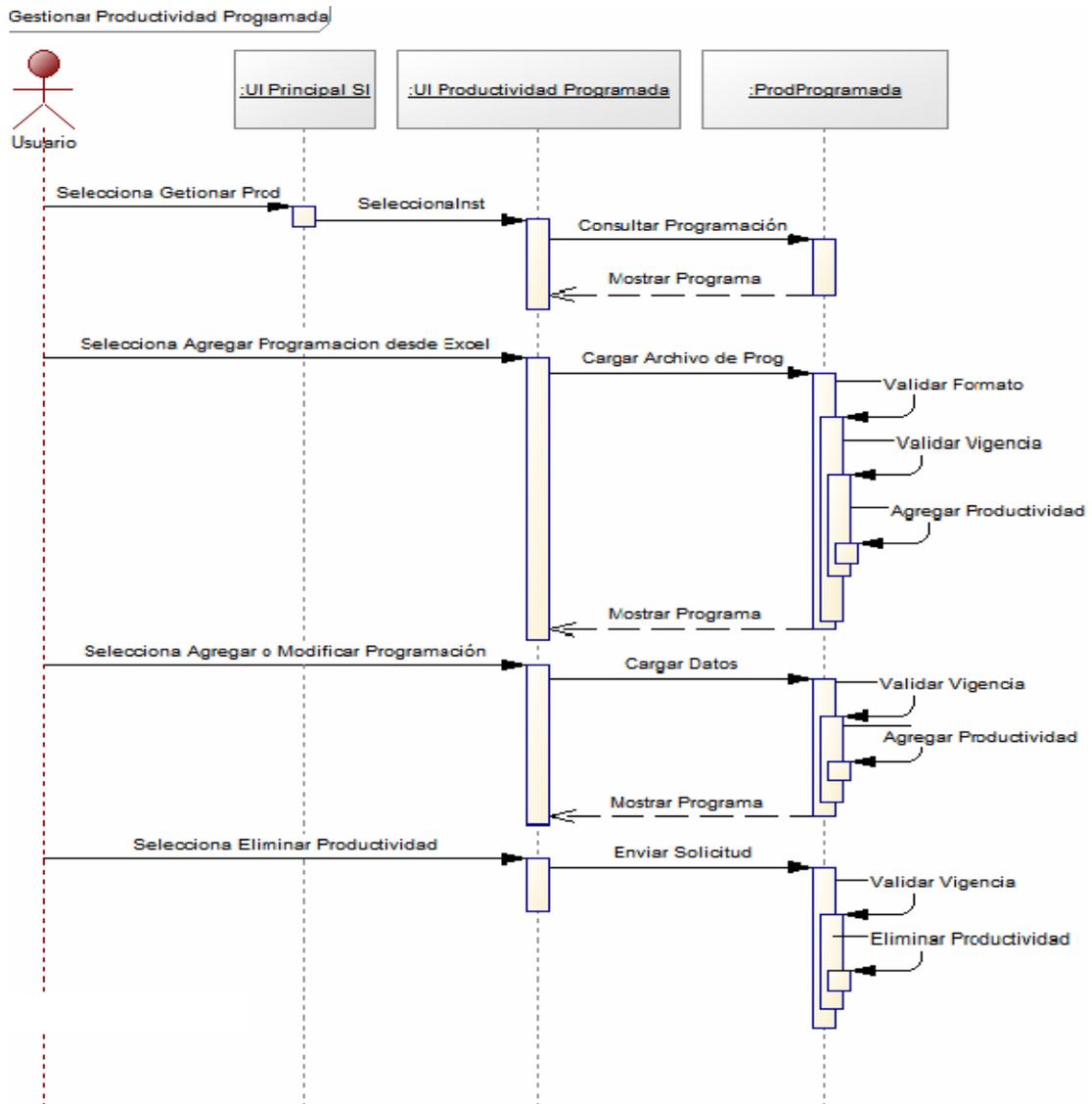


Figura A1. Diagrama de secuencia del caso de uso Gestionar Productividad Programada.

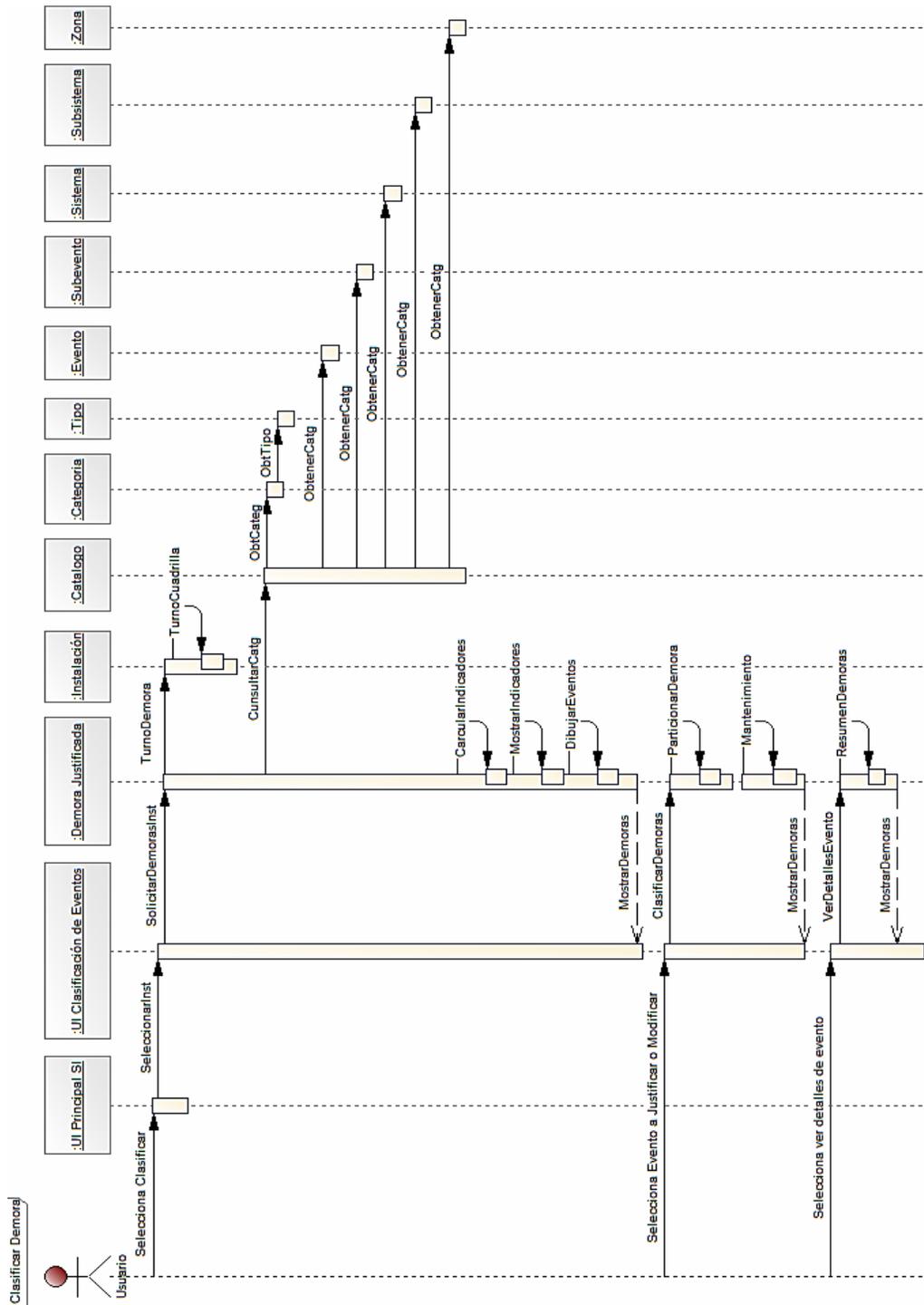


Figura A2. Diagrama de secuencia del caso de uso Clasificar Demoras.

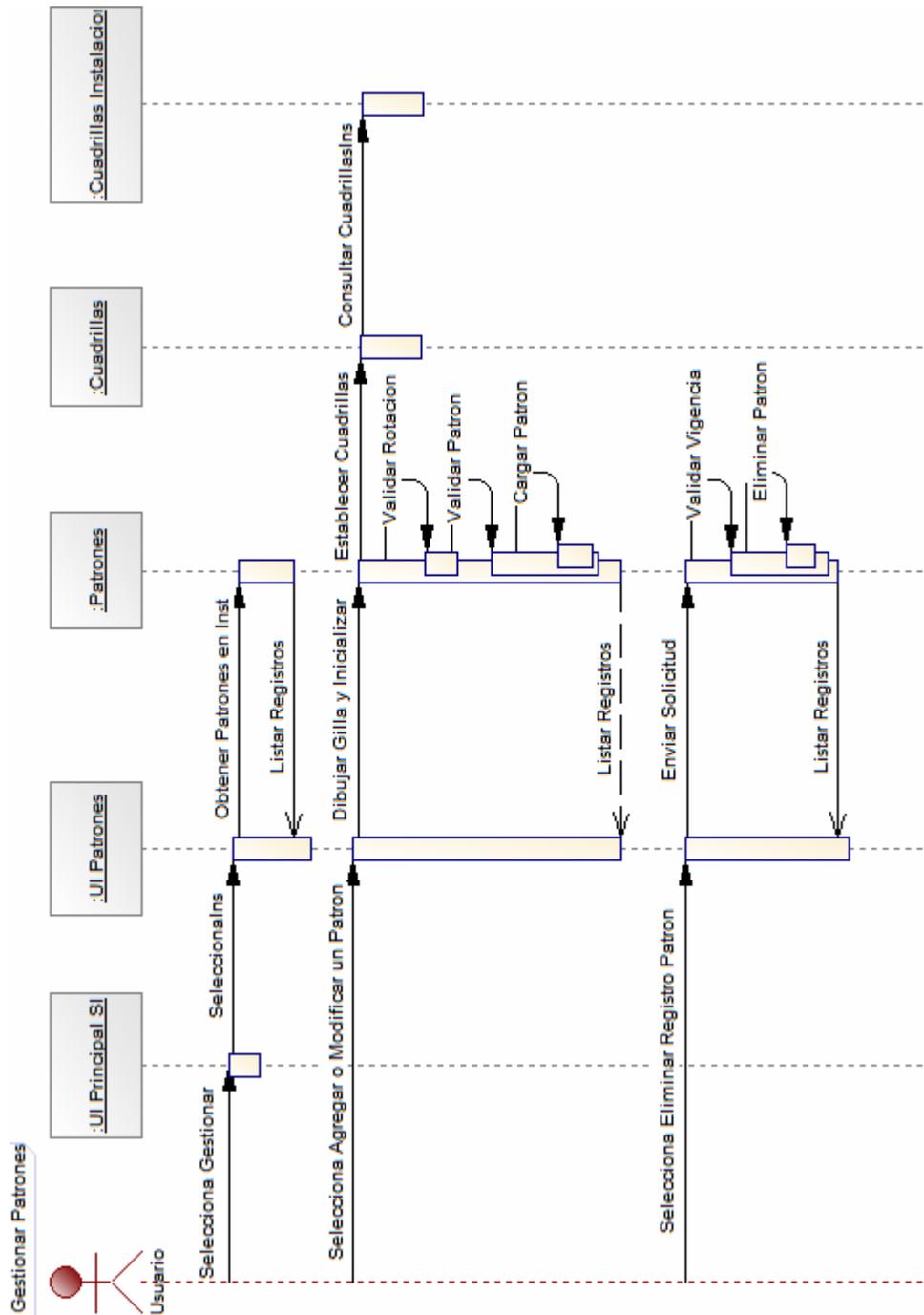


Figura A3. Diagrama de secuencia del caso de uso Gestionar Patrones.

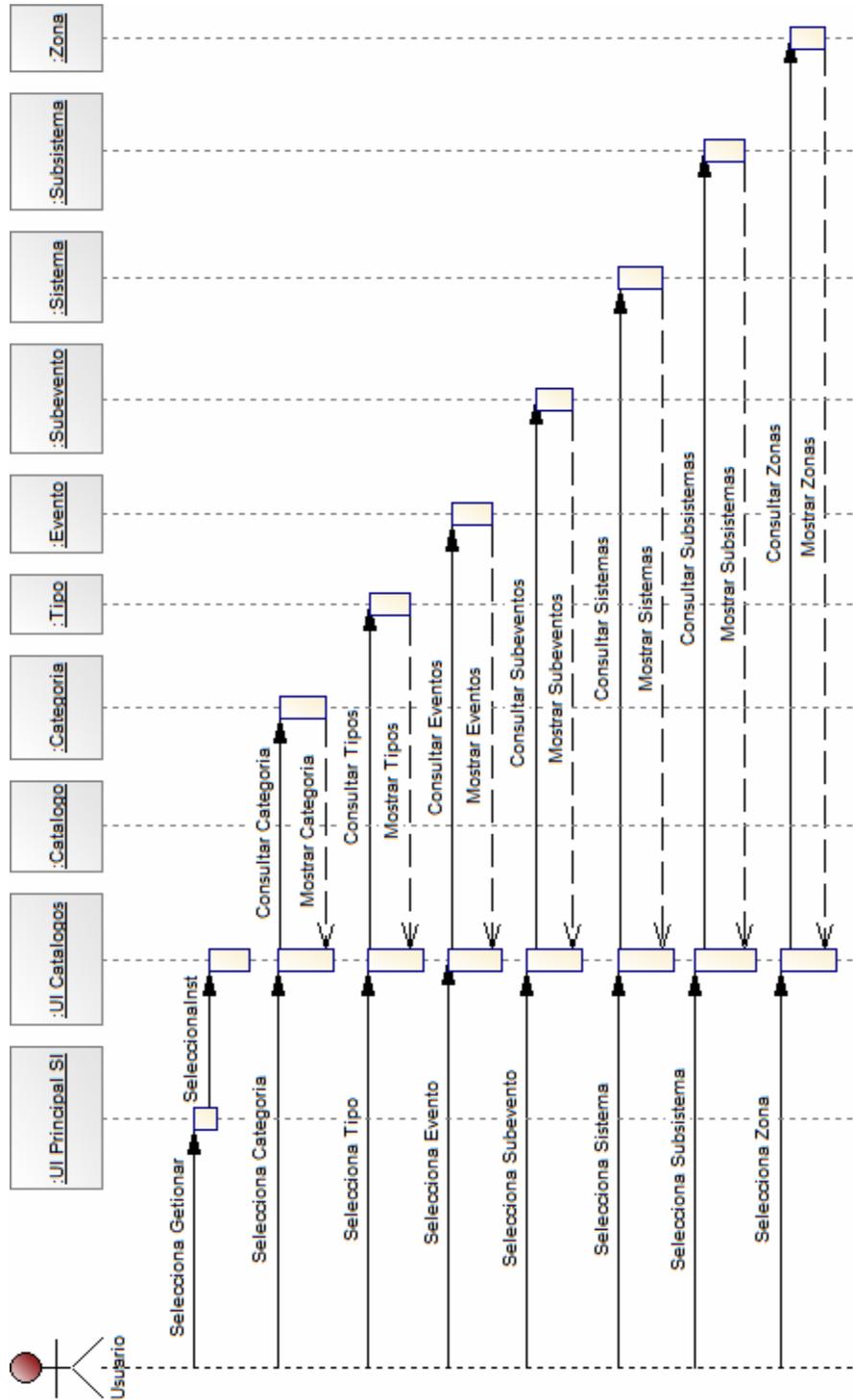


Figura A4. Diagrama de secuencia del caso de uso Gestionar Catálogos.

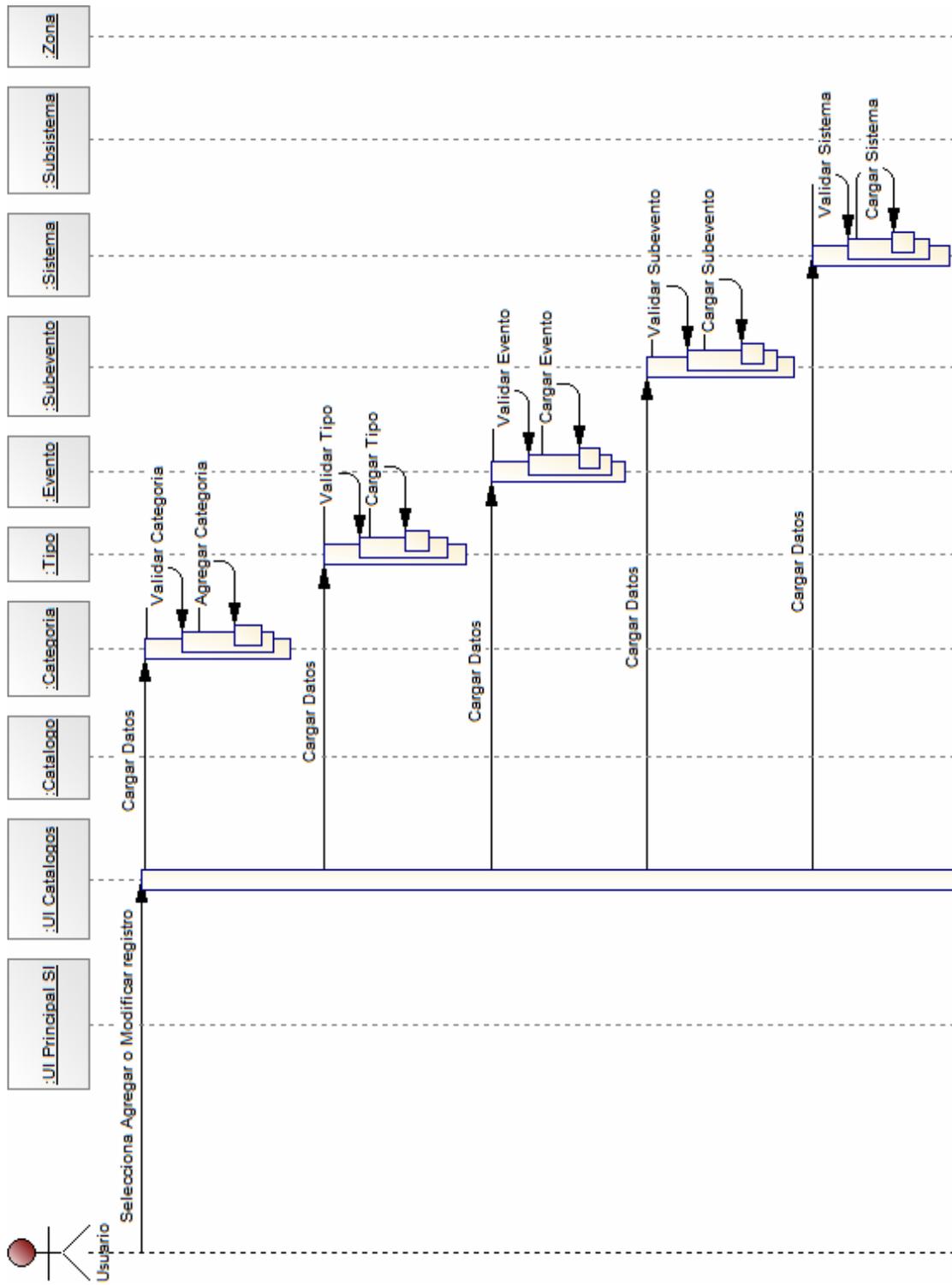


Figura A4. Continuación.

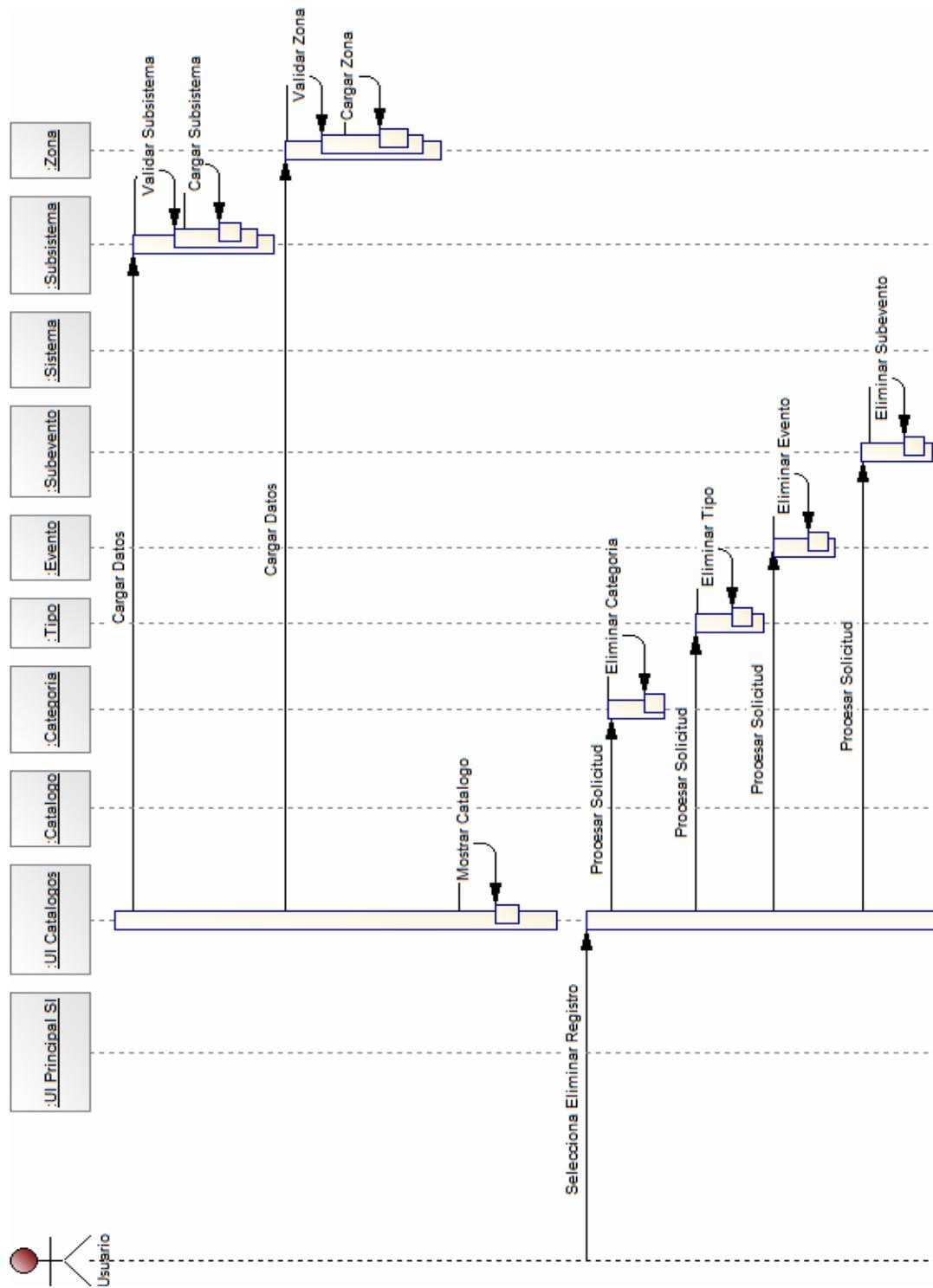


Figura A4. Continuación.

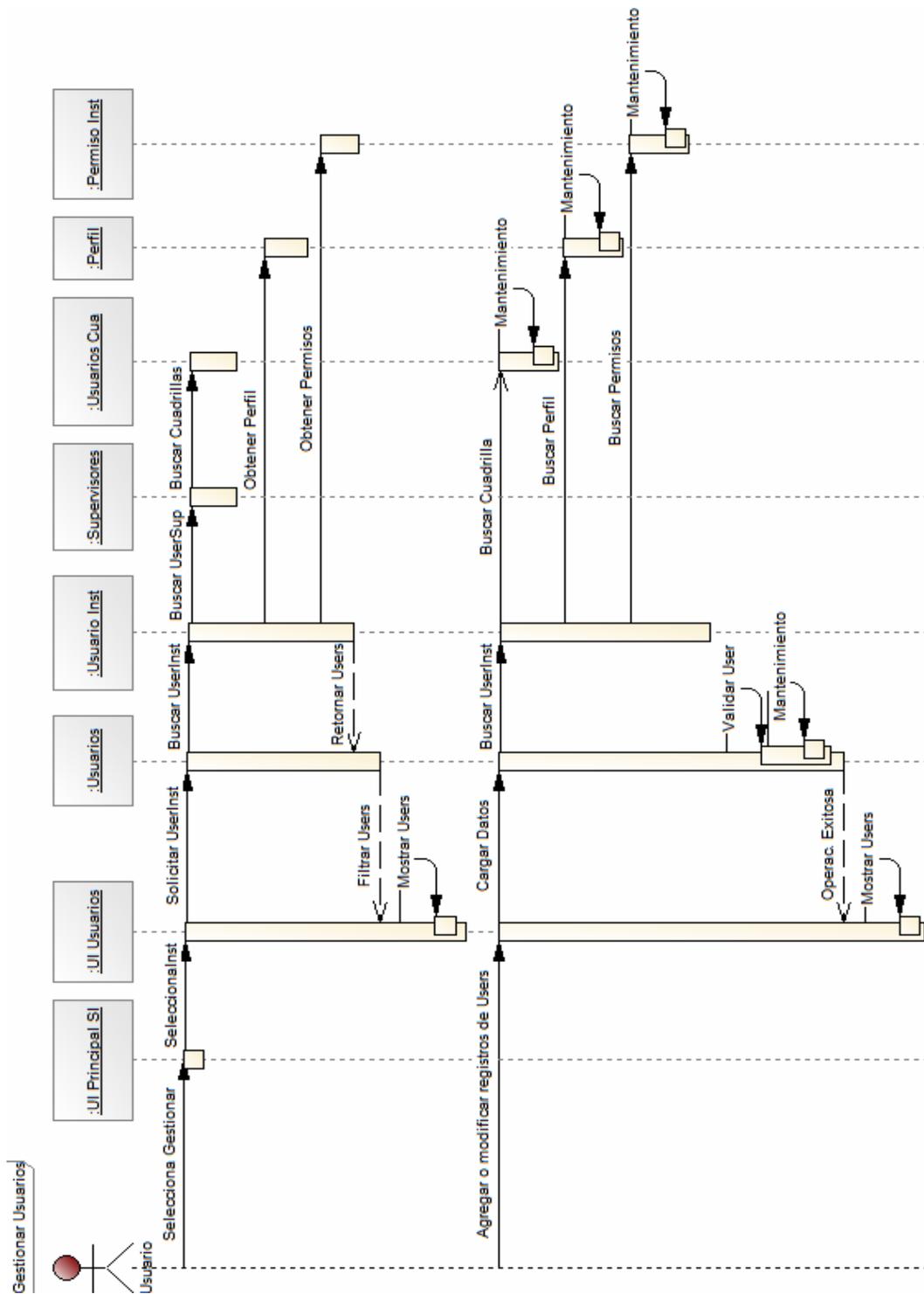


Figura A5. Diagrama de secuencia del caso de uso Gestión de Usuarios.

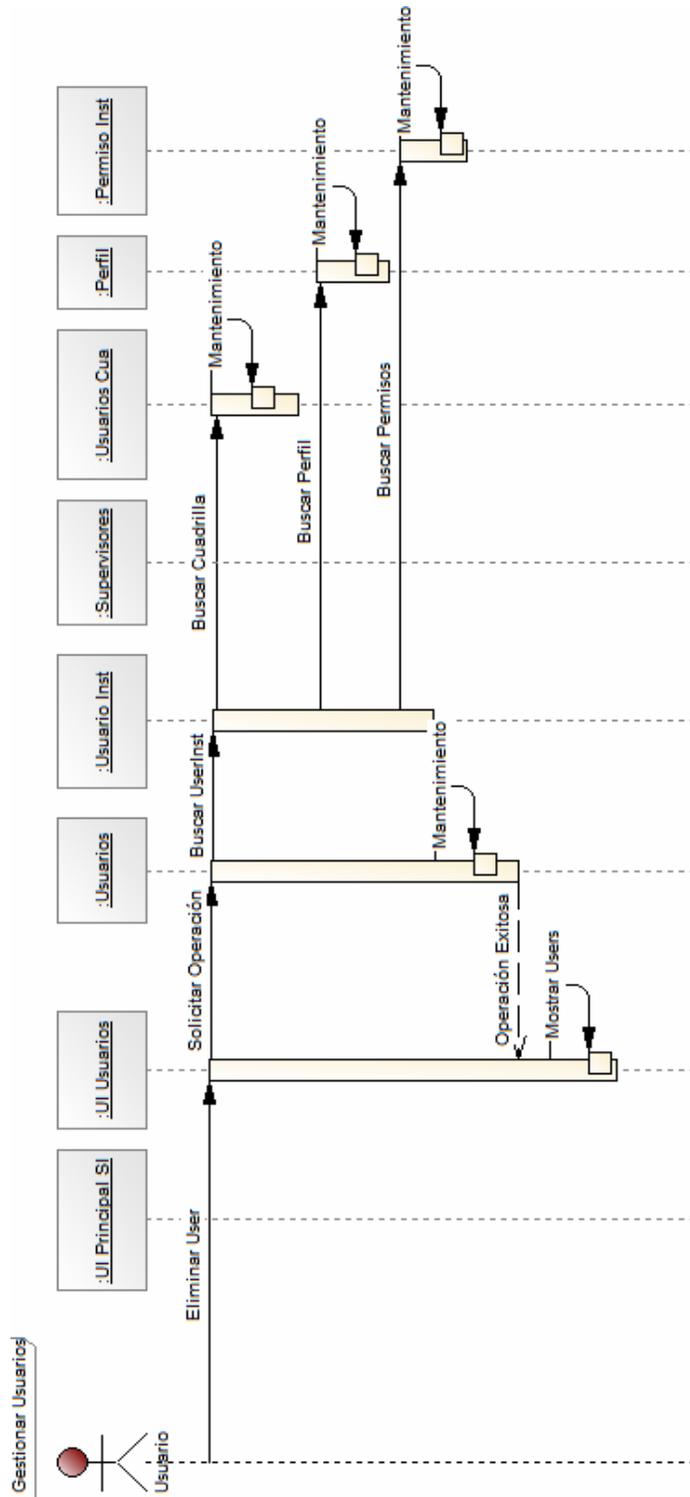


Figura A5. Continuación.

Apéndice B. Modelo Físico de la Base de Datos

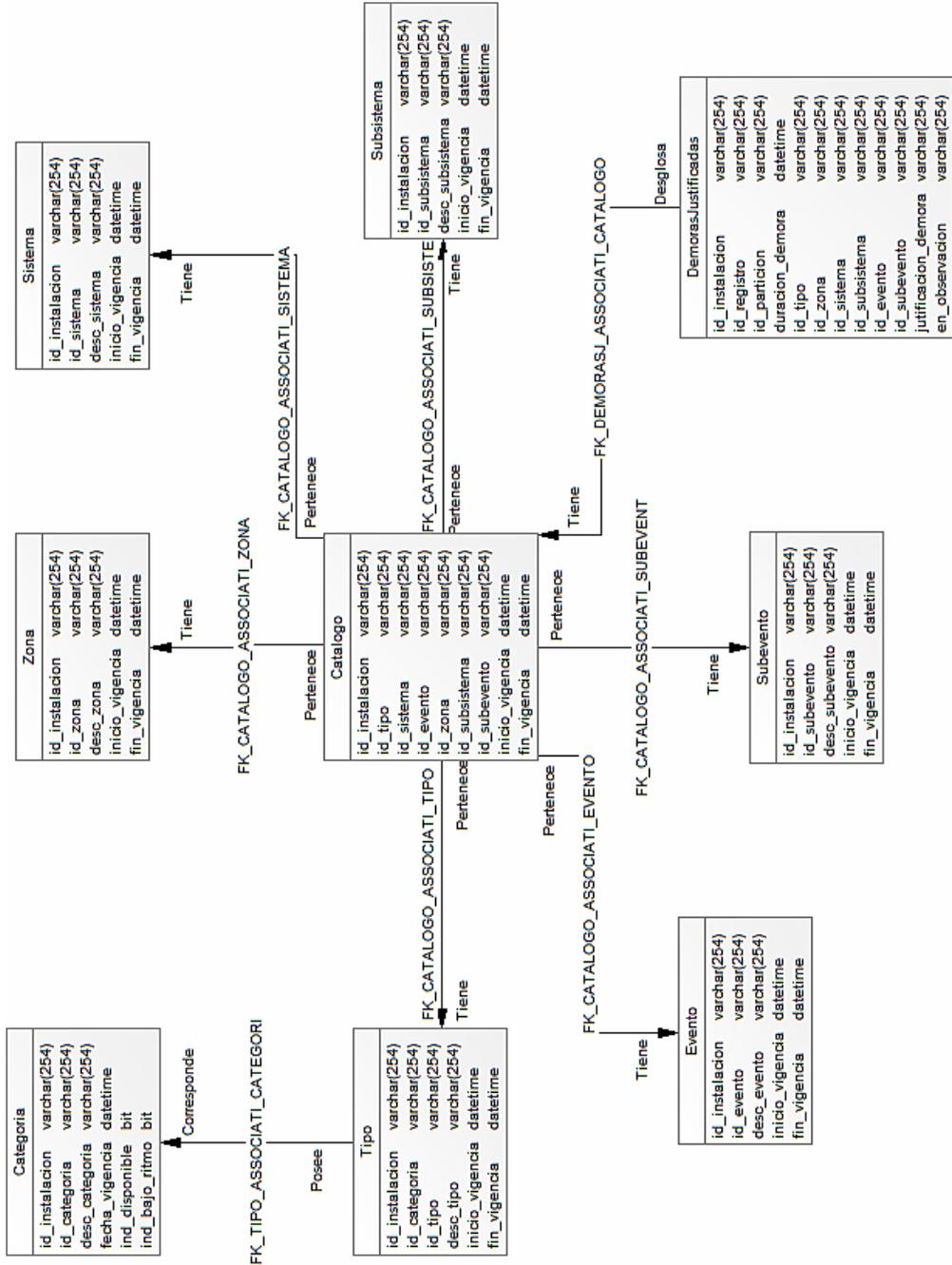


Figura B1. Diseño Físico de la Base de Datos.

Apéndice C. Código fuente

Código fuente del formulario de carga de Productividad Programada

Caso de uso asociado: Gestionar Productividad.

Nombre del archivo: produccion_programada_ABM.aspx.

Fuente:

```
<%@ Register TagPrefix="uc1" TagName="nav_bar" Src="sgl_shared/sgl_bar/nav_bar.ascx" %>
<%@ Register TagPrefix="uc1" TagName="sgl_combo" Src="sgl_shared/sgl_combo/sgl_combo.ascx" %>
<%@ Page Language="vb" AutoEventWireup="false" Codebehind="produccion_programada_ABM.aspx.vb"
Inherits="si_apr.produccion_programada_ABM" %>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
  <HEAD>
    <title>Productividad Programada</title>
    <meta content="Microsoft Visual Studio .NET 7.1" name="GENERATOR">
    <meta content="Visual Basic .NET 7.1" name="CODE_LANGUAGE">
    <meta content="JavaScript" name="vs_defaultClientScript">
    <meta content="http://schemas.microsoft.com/intellisense/ie5" name="vs_targetSchema">
    <LINK href="/sgl/css/pp_css.css" type="text/css" rel="stylesheet">
    <LINK href="/sgl/css/calendario.css" type="text/css" rel="stylesheet">
    <script src="/sgl_shared/calendario/js/calendar-stripped.js" type="text/javascript"></script>
    <script src="/sgl_shared/calendario/js/calendar-es.js" type="text/javascript"></script>
    <script src="/sgl_shared/calendario/js/script.js" type="text/javascript"></script>
    <script src="/sgl_shared/js/calendar-stripped.js" type="text/javascript"></script>
    <script src="/sgl_shared/js/calendar-es.js" type="text/javascript"></script>
    <script src="/sgl_shared/js/script.js" type="text/javascript"></script>
    <script>
      flagcal=false;
      //-----
      function valida_decimal()
      {
        var key = window.event.keyCode;
        var txt = window.event.srcElement;
        if (key == 44)
          if (txt.value.indexOf(",") < 0 && txt.value != "")
            if ((txt.id == "txefectividad") && (txt.value +
String.fromCharCode(key) > 100))
              return false;
            return true;
          }
          else
            return false;
        else
          if (key > 47 && key < 58 )
            {
              if ((txt.id == "txefectividad") && (txt.value +
String.fromCharCode(key) > 100))
                return false;
              return true;
            }
            else
              return false;
          }
      }
      //-----
    </script>
  </HEAD>
  <BODY>
    <div id="Form1" style="width: 100%; text-align: center; border: 1px solid black; padding: 5px;">
      <table border="1" style="width: 100%; border-collapse: collapse;">
        <tr>
          <td style="width: 50%; padding: 5px;">
            <uc1:nav_bar id="nav_bar" />
          </td>
          <td style="width: 50%; padding: 5px;">
            <uc1:sgl_combo id="sgl_combo" />
          </td>
        </tr>
      </table>
    </div>
  </BODY>
</HTML>
```

```

function validaciones()
{
    var mensaje = "Se encontraron los siguientes errores:";
    var bandera = 0;
    var fecha_seleccionada = new Array();
    var array = new Array();
    var today = new Date();
    var mifecha=new Date();
    var txfecha = document.getElementById('txfecha');
    today.setFullYear(today.getFullYear(),today.getMonth(),today.getDate());
    fecha_seleccionada = txfecha.value.split("/");
    mifecha.setFullYear(fecha_seleccionada[2],fecha_seleccionada[1]-
1,fecha_seleccionada[0]);

    if ((mifecha < today) && (txfecha.value != ""))
    {
        bandera = 1;
        txfecha.style.border = '#666666 2px solid';
        mensaje = mensaje + "\r- La Fecha Inicio de Vigencia no puede ser
anterior a la fecha actual.";
    }
    if (txfecha.value == "")
    {
        bandera = 1;
        txfecha.style.border = '#666666 2px solid';
        mensaje = mensaje + "\r- El campo Fecha esta vacío."
    }
    var txvalor = document.getElementById('txvalor');
    if (txvalor.value == "")
    {
        bandera = 1;
        txvalor.className = 'data_entries_resalte_p'
        mensaje = mensaje + "\r- El campo Producción Programada esta vacío.";
    }
    var txtiempoefectivo = document.getElementById('txtiempoefectivo');
    if (txtiempoefectivo.value == "")
    {
        bandera = 1;
        txtiempoefectivo.className = 'data_entries_resalte_p'
        mensaje = mensaje + "\r- El campo Tiempo Efectivo esta vacío.";
    }
    var txefectividad = document.getElementById('txefectividad');
    if (txefectividad.value == "")
    {
        bandera = 1;
        txefectividad.className = 'data_entries_resalte_p'
        mensaje = mensaje + "\r- El campo Efectividad esta vacío.";
    }
    if (bandera == 0)
        return true;
    else
    {
        alert(mensaje);
        return false;
    }
}

//-----
function click_on_date(id)
{
    try{
        return mostrarCalendario(id);
    }catch(e){}
}

```

```

//-----
function CerrarActualizando()
{
    window.close();
    window.opener.location.reload();
}
function SimularPopUpModal()
{
    if (window.opener.focus)
    {
        window.opener.focus = false;
        self.focus();
    }
}
</script>
</HEAD>
<body MS_POSITIONING="GridLayout" onBlur="SimularPopUpModal();">
<form id="Form1" method="post" runat="server">
<uc1:nav_bar id="Nav_bar1" runat="server"></uc1:nav_bar>
<table class="tabla_filtro" width="100%">
<tr>
<td class="separador"></td>
<td class="titulo" width="15%" runat="server" id="titulo"></td>
<td width="80%" align="right"></td>
<td class="separador"></td>
</tr>
</table>
<br>
<!------->
<table class="tabla_titulo" width="75%" align="center">
<tr>
<td id="encabezado" align="center" runat="server"></td>
</tr>
</table>
<!------->
<TABLE id="Table1" width="75%" align="center" class="tabla_reporte" cellSpacing="0"
cellPadding="0">
<TR>
<td class="encabezado" width="30%">Inicio Vigencia</td>
<td width="70%"><asp:textbox id="txfecha"
onclick="click_on_date(this.id)" CssClass="txt_fecha" ReadOnly="true"
Runat="server"></asp:textbox></td>
</TR>
<tr>
<td class="encabezado">Turno</td>
<td align="center"><uc1:sgl_combo runat="server" ID="cbo_turno"
Width="115px"></uc1:sgl_combo></td>
</tr>
<TR>
<td class="encabezado">Prod. Program.</td>
<td><asp:textbox id="txvalor" MaxLength="10" CssClass="txt_input"
Runat="server" onkeypress="return valida_decimal();"></asp:textbox></td>
</TR>
<TR>
<td class="encabezado">Tiempo Efectivo</td>
<td><asp:textbox id="txtiempoefectivo" MaxLength="10"
CssClass="txt_input" Runat="server" onkeypress="return valida_decimal();"></asp:textbox></td>
</TR>
<TR>
<td class="encabezado">Efectividad</td>
<td><asp:textbox id="txefectividad" MaxLength="10"
CssClass="txt_input" Runat="server" onkeypress="return valida_decimal();"></asp:textbox></td>
</TR>
</TABLE>

```

```

<br>
<!------->
<TABLE align="center" width=10%>
  <TR>
    <TD align="center">
      <asp:button id="btaceptar" Runat="server" Text="Guardar"
cssclass="boton_filtro_normal" onmouseover="this.className='boton_filtro_highlite';"
      onmouseout="this.className='boton_filtro_normal';"></asp:button>
    </TD>
    <td width=2%>&nbsp;</td>
    <TD align="center">
      <asp:button id="bteliminar" Runat="server" Text="Eliminar"
cssclass="boton_filtro_normal" onmouseover="this.className='boton_filtro_highlite';"
      onmouseout="this.className='boton_filtro_normal';" CausesValidation="False"></asp:button></TD>
    <td width=2%>&nbsp;</td>
    <TD align="center">
      <asp:button id="btcancelar" Runat="server" Text="Cerrar"
cssclass="boton_filtro_normal" onmouseover="this.className='boton_filtro_highlite';"
      onmouseout="this.className='boton_filtro_normal';" CausesValidation="False"></asp:button>
    </TD>
  </TR>
</TABLE>
<!------->
<a id="Script" runat="server"></a>
<!------->
<asp:validationsummary id="ValidationSummary1" runat="server" ShowSummary="False"
ShowMessageBox="True"></asp:validationsummary>
  <asp:requiredfieldvalidator id="llenar_valor" runat="server" ControlToValidate="txvalor"
    ErrorMessage="Llene el Campo Prod. Program."
Display="None"></asp:requiredfieldvalidator>
  <asp:requiredfieldvalidator id="llena_fecha" runat="server" ControlToValidate="txfecha"
    ErrorMessage="Llene el Campo Inicio Vigencia"
Display="None"></asp:requiredfieldvalidator>
  <asp:textbox id="id_prod_prog" runat="server" Width="0px"></asp:textbox>
<!------->
</form>
</body>
</HTML>

```

Nombre del archivo: producción_programada_ABM.aspx.vb

Fuente:

```
Imports System.Net.Sockets
Imports System.Net
```

```
Public Class produccion_programada_ABM
  Inherits System.Web.UI.Page
  Private mConexion As System.Data.SqlClient.SqlConnection
```

```
#Region " Web Form Designer Generated Code "
```

```
  'This call is required by the Web Form Designer.
  <System.Diagnostics.DebuggerStepThrough()> Private Sub InitializeComponent()
```

```
End Sub
  Protected WithEvents cbo_turno As sgl_shared.sgl_combo
  Protected WithEvents Nav_bar1 As sgl_shared.barra_menu
```

```

Protected WithEvents txfecha As System.Web.UI.WebControls.TextBox
Protected WithEvents txvalor As System.Web.UI.WebControls.TextBox
Protected WithEvents txcap As System.Web.UI.WebControls.TextBox
Protected WithEvents btaceptar As System.Web.UI.WebControls.Button
Protected WithEvents btcancelar As System.Web.UI.WebControls.Button
Protected WithEvents bteliminar As System.Web.UI.WebControls.Button
Protected WithEvents ValidationSummary1 As System.Web.UI.WebControls.ValidationSummary
Protected WithEvents llenar_valor As System.Web.UI.WebControls.RequiredFieldValidator
Protected WithEvents llenar_fecha As System.Web.UI.WebControls.RequiredFieldValidator
Protected WithEvents id_prod_prog As System.Web.UI.WebControls.TextBox
Protected WithEvents encabezado As System.Web.UI.HtmlControls.HtmlTableCell
Protected WithEvents tblBotonEliminar As System.Web.UI.HtmlControls.HtmlTable
Protected WithEvents rdbTurno As System.Web.UI.WebControls.RadioButtonList
Protected WithEvents txtiempoefectivo As System.Web.UI.WebControls.TextBox
Protected WithEvents Script As System.Web.UI.HtmlControls.HtmlAnchor
Protected WithEvents llenar_cap_inst As System.Web.UI.WebControls.RequiredFieldValidator
Protected WithEvents titulo As System.Web.UI.HtmlControls.HtmlTableCell
Protected WithEvents txefectividad As System.Web.UI.WebControls.TextBox

```

'NOTE: The following placeholder declaration is required by the Web Form Designer.

'Do not delete or move it.

```
Private designerPlaceholderDeclaration As System.Object
```

```
Private Sub Page_Init(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Init
```

```
'CODEGEN: This method call is required by the Web Form Designer
```

```
'Do not modify it using the code editor.
```

```
InitializeComponent()
```

```
End Sub
```

```
#End Region
```

```
Private Sub Page_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
```

```
Try
```

```
mConexion = New System.Data.SqlClient.SqlConnection(Conexion)
```

```
Me.id_prod_prog.Text = Request("id_instalacion")
```

```
mConexion.Open()
```

```
If Not Me.IsPostBack Then
```

```
CargarComboTurno()
```

```
If Request("fecha_vigencia") <> "" Then
```

```
Me.titulo.InnerText = "Modificar Productividad Programada"
```

```
Me.encabezado.InnerText = "Modificar Productividad"
```

```
modificar()
```

```
Else
```

```
Me.titulo.InnerText = "Agregar Productividad Programada"
```

```
Me.encabezado.InnerText = "Agregar Productividad"
```

```
Me.bteliminar.Visible = False
```

```
'Me.tblBotonEliminar.Visible = False
```

```
End If
```

```
End If
```

```
Catch ex As Exception
```

```
Dim err As New ManejoError.EnviarError(Me.Page, ex)
```

```
End Try
```

```
End Sub
```

```
Private Sub CargarComboTurno()
```

```
cbo_turno.Items.Add(New ListItem("1", "1"))
```

```
cbo_turno.Items.Add(New ListItem("2", "2"))
```

```
cbo_turno.Items.Add(New ListItem("3", "3"))
```

```
cbo_turno.SelectedValue = "1"
```

```
End Sub
```

```
Sub modificar()
```

```

Dim dr As SqlConnection.SqlDataReader
si_apr.da.web_apr_si_obtener_prod_programadas(mConexion, dr, Request("id_instalacion"), "", "", Request("fecha_vigencia"),
1)
    While dr.Read
        Me.txfecha.Text = Format(FechaSidor(dr.Item("fecha_hora_inicio_vigencia"), True), "dd/MM/yyyy")
Me.txvalor.Text = Replace(dr.Item("prod_programada"), ",", ".")
Me.cbo_turno.SelectedValue = ObtenerTurnoDadaFechaHora(dr.Item("fecha_hora_inicio_vigencia"))
        Me.txtiempoefectivo.Text = Replace(dr.Item("tiempo_efectivo"), ".", ",")
Me.txefectividad.Text = Replace(dr.Item("efectividad"), ".", ",")
    End While
    dr.Close()
End Sub

Private Sub btaceptar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btaceptar.Click
    Dim dr As SqlConnection.SqlDataReader
    If si_apr.da.web_apr_si_actualizar_prod_programada(mConexion, Request("fecha_vigencia"), Request("id_instalacion"),
Funciones.ArmarHoraInicioTurno(New Date(Me.txfecha.Text.Substring(6, 4), Me.txfecha.Text.Substring(3, 2),
Me.txfecha.Text.Substring(0, 2)), Me.cbo_turno.SelectedValue), CSng(Replace(Me.txvalor.Text, ".", ",")),
CSng(Replace(Me.txtiempoefectivo.Text, ".", ",")), Session("id_usuario"), CSng(Replace(Me.txefectividad.Text, ".", ","))) Then
        Me.Response.Write("<script language='javascript'>alert('NO SE PUDO INSERTAR, YA ESTA PROGRAMADO ESE
DIA')</script>")
    Else
        Me.Script.InnerHtml = "<script language='javascript'>CerrarActualizando()</script>"
    End If
    Try

        Catch ex As Exception
            Dim err As New ManejoError.EnviarError(Me.Page, ex)
        End Try
    End Sub

Private Sub Page_Unload(ByVal sender As Object, ByVal e As System.EventArgs) Handles MyBase.Unload
    Try
        mConexion.Close()
        mConexion.Dispose()
    Catch ex As Exception
        Dim err As New ManejoError.EnviarError(Me.Page, ex)
    End Try
End Sub

Private Sub bteliminar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles bteliminar.Click
    Try
        si_apr.da.web_apr_si_eliminar_prod_programada(mConexion, Request("id_instalacion"),
Funciones.ArmarHoraInicioTurno(New Date(Me.txfecha.Text.Substring(6, 4), Me.txfecha.Text.Substring(3, 2),
Me.txfecha.Text.Substring(0, 2)), Me.cbo_turno.SelectedValue), Session("id_usuario"))
        Me.Script.InnerHtml = "<script language='javascript'>CerrarActualizando()</script>"
    Catch ex As Exception
        Dim err As New ManejoError.EnviarError(Me.Page, ex)
    End Try
End Sub

Private Sub Page_PreRender(ByVal sender As Object, ByVal e As System.EventArgs) Handles MyBase.PreRender
    Try
        Me.btaceptar.Attributes.Add("onclick", "return validaciones();")
        Me.btcancelar.Attributes.Add("onclick", "window.close();")
        Me.Nav_bar1.Visible = False
    Catch ex As Exception
        Dim err As New ManejoError.EnviarError(Me.Page, ex)
    End Try
End Sub
End Class

```

Apéndice D. Manual de usuario de la aplicación

MANUAL DE USUARIO DE LA APLICACIÓN *WEB* DE GESTIÓN DEL SI

Antes de iniciar con la explicación del funcionamiento de la aplicación se expondrán los principales términos y conceptos manejados, así como los indicadores y interfaz de controles que la aplicación contempla

ESTRUCTURA DE TIEMPOS EN EL SISTEMA DE INTERRUPCIONES

Tiempo Calendario: Son las 24 horas de cada día, o las 8 horas de cada turno.

Tiempo Efectivo: Tiempo en el que la línea está produciendo.

Tiempo Disponible: Es el tiempo que disponen los “gestionadores” de la línea para ejercer su función. Es la suma del Tiempo Efectivo + Demoras.

Demoras: Son las detenciones imprevistas de la línea. Las demoras pueden ser de dos categorías:

- **Operativas:** Son aquellas que se generan como consecuencia de fallas en el proceso productivo, tales como: repetir soldadura, limpieza de rodillos, cambiando o calibrando cabezales, inspección o descarte de material, cambios de cilindros, entre otros.
- **No Operativas:** Son aquellas que se generan y que no son propias del proceso productivo, estas son:
 - Mecánicas
 - Eléctricas
 - Grúas
 - Externas

Tiempo No Disponible (Paradas): Son las detenciones previsibles de la línea, su frecuencia y duración dependen de políticas de la empresa o de limitaciones de los equipos, entre estas están:

- Feriados
- Paros Sindicales
- Cambios de medida
- Paradas programadas de Mant.
- Falta de material/ demanda
- Régimen de turnos
- Reparaciones extraordinarias

La estructura de tiempos que maneja el SI estará establecida, como se indica en la siguiente Figura D1:

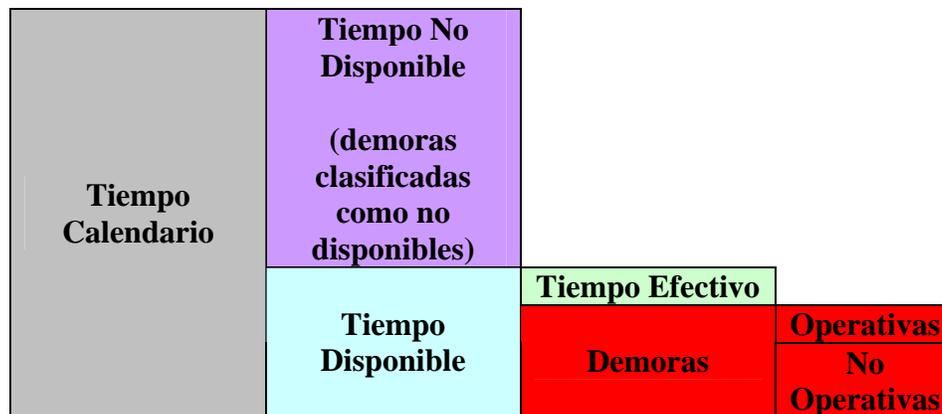


Figura D1. Interpretación de tiempos

INDICADORES DELA APLICACIÓN

La aplicación de gestión maneja una serie de indicadores que representan los estados en la cual se encuentra la línea de producción durante un turno de trabajo mediante una barra que lo especifica respecto al horario.

En la tabla D1 siguientes se muestra una leyenda que le ayudara a identificar estos estados en la aplicación.

Tabla D1. Estados de la Barra de la aplicación *Web*.

Colores	Indicativo
	Interrupción.
	Tiempo no disponible.
	Estado normal, línea trabajando.
	Perdida de productividad.

Indicador de Disponibilidad: Indica el efecto de los factores exógenos a la línea sobre su estructura de tiempos.

Indicador de Efectividad: Indica el aprovechamiento del tiempo disponible por parte de los “gestionadores” de las líneas.

Indicador de Productividad: indica la relación entre las toneladas del material producido, con relación al tiempo efectivo en el cual se produjo tal tonelaje.

DEFINICIÓN DE TÉRMINOS MANEJADOS EN LA APLICACIÓN

Una interrupción: representa el momento en el tiempo en el cual la capacidad se hace cero, por efecto de una falla de un equipo, de un evento programado u originados por terceros

Una pérdida de productividad: representa la pérdida o disminución de capacidad (en tn/hr) contra la meta establecida, puede deberse a la falla de un equipo que merma capacidad, a las condiciones de operación (cambio o regulación en variables de proceso), a la materia prima o a la calidad o cantidad de los insumos operativos. Normalmente, antes y después de un evento que detenga la producción se origina un bajo ritmo por el evento en cuestión.

La duración de la interrupción: se mide desde que la producción de las líneas es igual a cero hasta que comienza a producir otra vez.

El tiempo con pérdida de producción (tpp): cuando la línea permanece más de este tiempo en una velocidad menor a la nominal, se registra esta situación.

DEFINICION DE LOS PRINCIPALES CONTROLES EMPLEADOS EN LA APLICACIÓN.

En SIDOR es muy común la búsqueda de datos a través de fechas y turnos, y debido a esto usted podrá usar controles para esto, la incorporación de un calendario a la aplicación, lo ayuda a que se encuentre más familiarizado y más confiado a la hora de seleccionar una fecha, y no que tenga que introducir la fecha manualmente, lo que podría generar que introduzca un formato de fecha erróneo.



Figura D2. Calendario

Otros de los controles o estilos de interacción que se utilizarán para el ingreso de datos son los combos o listas así como los checkcombos, con los cuales se quiere que usted solo seleccione uno o varios ítem de la lista de valores.

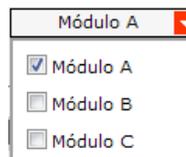


Figura D3. Checkcombo

Con la implementación de la aplicación, nació un control nuevo, el cual formara parte de ahora en adelante de los estilos definidos en SIDOR. Para la visualización de los eventos de la planta, se quiso, que de una manera rápida y veraz usted pudiera observar su estado en todo el turno, debido a esto se creó una barra de estado en el cual se visualizan los eventos de la planta distinguiéndolos por colores y que a su vez tendría la escala por horas desde el inicio de turno hasta su finalización.

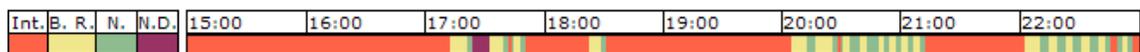


Figura D4. Barra de estado

En varias de las interfaces de la aplicación se podrá ver que el encabezado de filtros, consta de botones y calendarios, los cuales le permiten elegir la fecha y el turno de consulta deseados así como rangos de los mismos. Inicialmente estos controles, muestran los datos correspondientes a la fecha y turno actuales.



Figura D5. Control de turnos

Si desea visualizar un turno anterior o posterior a la fecha mostrada lo podrá hacer presionando las flechas que se encuentran en los laterales de la misma.

Otra forma hacer este cambio, consta de hacer clic sobre el campo fecha, con el cual se desplegara el calendario mostrado anteriormente, que le permitirá escoger la fecha deseada. Siguiendo la misma lógica, podrá cambiar el turno de consulta, bien sea presionando las flechas anterior/posterior, o clickeando sobre los botones correspondientes que indican el número del turno a consultar. También podrá elegir ver los datos del turno actual, al presionar el botón correspondiente.

La aplicación, posee otro control de este tipo y que maneja la misma lógica para la realización de consultas pero con la particularidad de que en este se puede especificar un rango de tiempo.



Figura D6. Control de fechas y turnos

Al presionar cualquiera de los botones pertenecientes a este control este dinámicamente cambiará su aspecto para adaptarse a sus necesidades y hacer más cómoda la selección de los filtros.

También existen una serie de botones y link identificados con imágenes, considerados especiales ya que poseen una funcionalidad bien definida, estos se muestran a continuación en la siguiente tabla:

Tabla E2. Botones y enlaces de interés

Botón	Nombre	Función
	Home	Enlace a la página principal.
	Pdf	Genera reporte en formato Pdf.
	Excel	Genera reporte en formato Excel.
	Agrupar	Agrupar por fechas.
	Imprimir	Imprimir interrupciones.
	Detalles demora	Enlace que indica si una demora está clasificada y muestra sus detalles.
	Editar	Modificar un registro.
	Eliminar	Eliminar un registro.
	Carpeta de archivos	Carpeta contenedora de documentos.
	Guardar documento	Descarga documentos.
	Ayuda	Manual de usuario de la aplicación.

REQUERIMIENTOS MÍNIMOS PARA UTILIZAR EL SISTEMA

Requisitos de software

Servidor:

- Navegador de Internet: Microsoft Internet Explorer 6 o superior.
- *Microsoft Visual Studio 2003.*
- Manejador de Base de Datos *SQL Server 2000.*
- Servidor de Reportes *Microsoft SQLServerReporting Services2000.*
- Sistema operativo *Microsoft WindowsServer 2003*, como plataforma de soporte para las aplicaciones que permiten el desarrollo.
- IIS (*Internet Information Server*) 6.0.

Cliente:

- Sistema operativo *Microsoft Windows XP Professional Service Pack 2*
- Navegador de Internet: *Microsoft Internet Explorer 6* o superior.

Requisitos de la plataforma hardware

Servidor:

- Equipos *Advantech*. Con procesador Pentium 4 de 2.8 Ghz con un 1 Gb de RAM. Con una arquitectura de clúster pasivo/activo con storage de capacidad de 550Gb con RAID de 5+1.
- Interfaz de red *Ethernet*.
- Monitor a color con una resolución máxima de 1280 x 1024.

Cliente:

- Procesador Pentium 2 o superior.
- 128 MB de memoria de acceso aleatorio (RAM).
- Monitor a color de 17 pulgadas con resolución de 1024x768 píxeles como mínimo.
- Disponibilidad para conexión a red/ Internet mediante interfaz de red *Ethernet*.

Parámetros de instalación

Para la instalación del sistema y su correcto desempeño, la empresa debe disponer de los equipos de computación y el software indispensable para el funcionamiento completo de la aplicación desarrollada. Además se requiere de un usuario disponible y que este directamente relacionado con los procesos automatizados por el sistema.

Pantalla principal de la Aplicación de Gestión del SI

A continuación observara la pantalla principal del Sistema de Gestión del Si, en la cual podrá observar el estado actual de las líneas que conforman las plantas de la empresa. Inicialmente al entrar a la página, se muestran los datos correspondientes a la fecha, usuario en dominio SIDOR y el estado en tiempo real de las principales líneas de producción.

The screenshot displays the main interface of the SIDOR SI Management System. At the top, there is a navigation bar with the SIDOR logo and several menu items: SIDOR, Clasf. APR, Clasf. LAC, Clasf. LAF, and Gestión SI. Below this, a secondary navigation bar shows 'SIDOR > HOME' and 'HOME | Documentación'. The main content area is divided into several sections. On the left, there is a sidebar menu with expandable sections: 'Áreas Primarias' (listing lines like Pellas A, HYL2, MID1, M2A, M2B, M2C, Matesi M1, Matesi M2), 'Acera de Palanquillas' (listing lines HE 1, HE 3, HC 2, HC 4, MCC2, MCC3), 'Barras y Alambrón' (listing lines Barras, Alambrón), and 'Servicios' (listing line Oxígeno 3). The central part of the screen features a large banner with the text '¡Fíjate en los colores!' and 'Deposita los materiales recuperables y desechos en sus respectivos contenedores'. Below the banner, there are five colored boxes representing different material categories: Similar a Doméstico, Compactables; Similar a Doméstico, No Compactables; Chatarra Ferrosa; Chatarra No Ferrosa; and Material cosmético con Aceite o Grasa. Below these boxes, there is a section titled 'Clasifica cada material en su contenedor' with a list of instructions. On the right side, there are several tables showing the status of production lines, including 'Acera de Planchones', 'Laminación en Caliente', and 'Laminación en Frío'. Each table has columns for 'Línea', 'No Disp.', 'Demoras', and 'Efect.'. The top right corner of the interface shows 'Piso de Planta SI' and a home icon.

Figura D7. Pantalla Principal Gestión del SI

En la figura anterior podemos detallar varias partes de la aplicación que merece la pena explicar progresivamente. En principio podemos ver que el encabezado de

menú consta de pestañas que vemos en la Figura D8, las cuales le permiten al usuario elegir las acciones que quiere ejecutar sobre la aplicación.



Figura D8. Encabezado de Menú

El menú muestra primordialmente las opciones de clasificar demoras en las plantas correspondientes, desplegando un submenú donde se pueden seleccionar la línea que se quiere consultar, de igual manera la pestaña Gestión SI nos conduce a acciones de gestión sobre una planta en una línea específica cómo gestionar productividad programa de la línea o los catálogos que maneja, entre otros.

Pantalla Clasificación de Demoras

Para acceder a esta pantalla es necesario escoger una de las opciones en el menú que indica Clasf. LAF, Clasf. APR, Clasf. LAC y en el submenú emergente seleccionar instalación y la línea que se quiere consultar para luego clasificar las demoras que lo ameriten. Esto se Muestra en la Figura Siguiente.

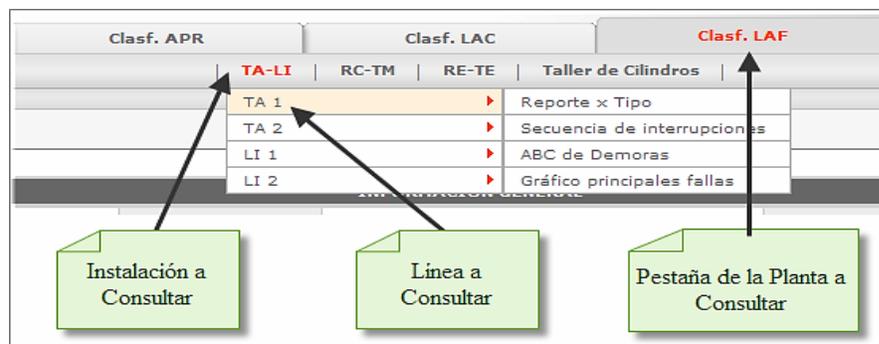


Figura D9. Pestañas de clasificación de eventos.

Seguidamente se mostrará la pantalla siguiente en la Figura D10, seteando los datos de la consulta como los actuales.

The screenshot shows the SIDOR software interface. At the top, there are navigation tabs for 'SIDOR', 'Clasf. APR', 'Clasf. LAC', 'Clasf. LAF', and 'Gestión SI'. Below this, there are buttons for 'TA-LI', 'RC-TM', 'RE-TE', and 'Taller de Cilindros'. The main header displays 'Clasf. LAF > TA-LI > TA 1' and 'Tandem 1'. A date and time selector shows '13/09/2008' and 'Turno Actual'. A 'Refrescar' button is also present.

INFORMACIÓN GENERAL

Línea: Tandem 1 Usuario: Administrador Fecha y Hora Actual: 30/11/2010 20:22:13

DATOS TURNO SELECCIONADO

Turno:	3	Inicio:	13/09/2008 15:00:00	Fin:	13/09/2008 23:00:00	Cuadrilla:	D
Tiempo No Disponible [h:ms]:	00:06:59	Utilización Neta (T. Neto / T. Disp.) * 100:	32.69 %	Cantidad de Interrupciones Detectadas:	9	Cantidad de Interrupciones Clasificadas:	9
Tiempo Disponible [h:ms]:	07:51:01	Productividad Neta:		Total Micro Demoras Positivas [h:ms]:	00:06:57	Total Micro Demoras Negativas [h:ms]:	00:00:59
Total Demoras [h:ms]:	05:17:02	Total de Bobinas Procesadas [#]:		Total Toneladas Procesadas [T]:			
Tiempo Neto [h:ms]:	02:33:59						

INTERRUPCIONES

Detalle	Hora Inicio	Hora Fin	Duración	Velocidad	C. Bob.	F. Bob.	Tipo	Zona	Sistema	Subsistema	Evento	Subevento	Observaciones
	22:56:16	23:00:00	00:03:44	Detenido	✓			Línea operativa	Gestion operativa	Falta de personal	Falta de operad...	Cambio de turno	n#18-1673431.B...
	22:52:05	22:53:21	00:01:16	78% - 95%			Perdida de prod...	Velocidad Están...	No aplica	No aplica	No aplica	No aplica	n#18-1673431.B...
	22:51:03	22:52:05	00:01:02	60% - 78%			Perdida de prod...	Velocidad Están...	No aplica	No aplica	No aplica	No aplica	n#18-1673431.B...
	22:46:06	22:48:42	00:02:36	Detenido	✓			Línea operativa	Material	Inspeccion de m...	Por defectos	Otros	n#9-18-1673431-...
	22:39:22	22:43:05	00:03:43	78% - 95%			Perdida de prod...	Velocidad Están...	No aplica	No aplica	No aplica	No aplica	n#17-1673424-12...
	22:31:17	22:35:17	00:04:00	78% - 95%			Perdida de prod...	Velocidad Están...	No aplica	No aplica	No aplica	No aplica	n#16-1673435-12...
	22:23:33	22:26:51	00:03:18	78% - 95%			Perdida de prod...	Velocidad Están...	No aplica	No aplica	No aplica	No aplica	n#15-1673436-12...
	22:14:42	22:19:07	00:04:25	60% - 78%			Perdida de prod...	Línea perd. pro...	1, 2 o 3 bobina...	No aplica	No aplica	No aplica	n#14-1673437-12...
	22:03:22	22:10:21	00:06:59	42% - 60%			Perdida de prod...	Línea perd. pro...	1, 2 o 3 bobina...	No aplica	No aplica	No aplica	n#13-1673438-12...
	21:13:11	22:03:11	00:50:00	Detenido	✓		Taller de cilin...	Línea taller de...	Operativa	Cambio de cil. ...	Exfoliacion	No aplica	n#13-1673438-12...

[Ver Todos] >> Siguiente

Figura D10. Pantalla Clasificar Demoras.

En la pantalla anterior se muestran algunos de los controles planteados en la primera parte de este manual. Pero no esta demás explicar nuevamente su funcionamiento.

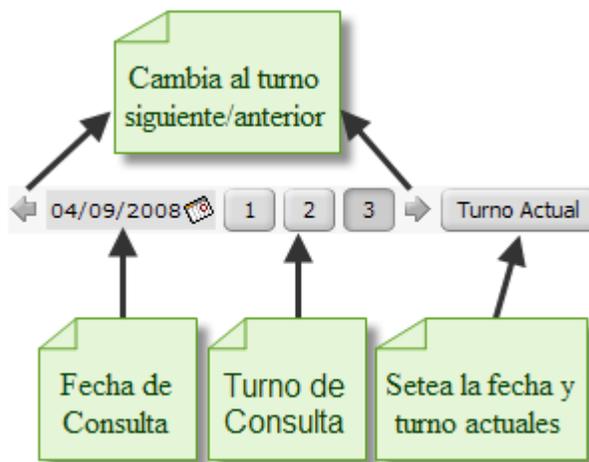


Figura D11. Control de Turnos

Si el usuario desea imprimir la información mostrada, lo podrá hacer al presionar el botón imprimir, con el cual se abrirá un reporte con formato de impresión con el contenido de los datos mostrados.

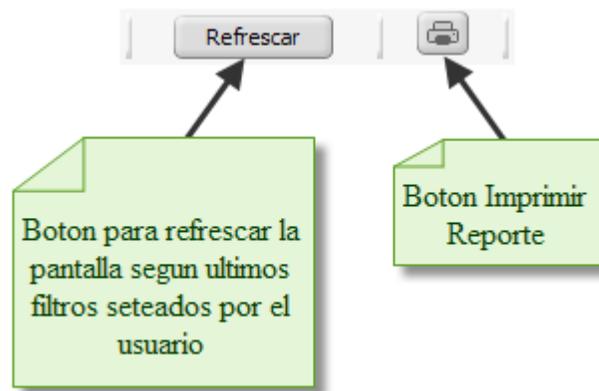


Figura D12. Botones refrescar e imprimir.

Para el turno seleccionado se muestra la información general la cual contiene el nombre de la línea donde se está accediendo, el usuario logeado, y la fecha y hora actual, como se muestra en la Figura D13.

INFORMACIÓN GENERAL					
Línea:	Tandem 1	Usuario:	Administrador	Fecha y Hora Actual:	30/11/2010 20:22:13

Link de cambio temporal de personal entre cuadrillas

Figura D13. Información general.

Luego vemos los datos del turno seleccionado, en el cual se visualizan algunos indicadores de cómo estuvo la línea en el turno que se esté consultando:

DATOS TURNO SELECCIONADO							
Turno:	3	Inicio:	13/09/2008 15:00:00	Fin:	13/09/2008 23:00:00	Cuadrilla:	D
Tiempo No Disponible [h:m:s]:	00:00:00	Utilización Neta [(T. Neto / T. Disp.) * 100]:	100%	Pérdida de Producción del turno [T]:	436.90		
Tiempo Disponible [h:m:s]:	08:00:00	Productividad Neta [T/H]:	466.71	Ganancia de Producción del turno [T]:	1040.74		
Total Demoras [h:m:s]:	00:00:00	Total Toneladas Procesadas [T]:	3733.70	Cantidad de Interrupciones Detectadas:	0		
Tiempo Neto [h:m:s]:	08:00:00	Producción Programada para el turno [T]:	3129.87	Cantidad de Interrupciones Clasificadas	0		

Figura D14. Datos turno seleccionado

- Turno.
- Fecha hora inicio del turno.
- Fecha hora fin del turno.
- Cuadrilla correspondiente al turno.
- Tiempo no disponible: corresponderá al tiempo de las demoras que sean clasificadas por los operadores como no disponible.
- Tiempo disponible: es la diferencia de tiempo calendario y el tiempo no disponible.
- Tiempo demoras: corresponde al tiempo en el cual ocurrieron las demoras en la línea.
- Tiempo efectivo: es la diferencia de tiempo disponible y las demoras ocurridas durante ese tiempo.
- Efectividad: Corresponde a la relación porcentual entre el tiempo efectivo y el tiempo disponible, lo cual permite hacer comparación con otras líneas aunque el proceso realizado sea totalmente diferente.
- Cantidad de interrupciones detectadas.
- Productividad Efectiva: indica la relación de toneladas procesadas entre el tiempo en que estuvo efectiva la línea.
- Productividad Promedio
- Total Toneladas Procesadas [T.]: indica la cantidad de toneladas de briquetas que fueron procesadas en el turno consultado.

- Producción Programada para el Turno.
- Pérdida de Producción del Turno: indica las toneladas que se perdieron por estar produciendo por debajo de la producción programada.
- Ganancia de Producción para el Turno: indica las toneladas que se ganaron por estar produciendo por encima de la producción programada.

Seguido de estos datos, observamos la barra de estado de la línea de producción consultada. La cual consta de tres estados, que ya hemos descrito, los cuales corresponden a interrupción (color rojo), tiempo no disponible (color morado), estado normal (color verde), y perdida de producción (color amarillo) (ver **¡Error! No se encuentra el origen de la referencia.**).

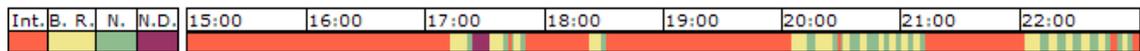


Figura D15. Barras de estado

Luego encontramos la lista de las Interrupciones y pérdidas de productividad detectados por el sistema como se muestra en la Figura D16, que deben ser clasificadas en los turnos correspondientes. Se pueden diferenciar cada registro por el color de la fila, para las perdidas su color será de naranja en cambio para las demoras se visualizaran de color blanco.

Link ver detalles Link Clasificar Demora Link Personalizar Tabla

INTERRUPCIONES													
Detalle	Hora Inicio	Hora Fin	Duración	Velocidad	C. Bob.	F. Bob.	Tipo	Zona	Sistema	Subsistema	Evento	Subevento	Observaciones
	22:56:16	23:00:00	00:03:44	Detenido	√			Linea operativa	Gestion operativa	Falta de personal	Falta de operad...	Cambio de turno	n#18. 1673431.B...
	22:52:05	22:53:21	00:01:16	78% - 95%			Perdida de prod...	Velocidad Están...	No aplica	No aplica	No aplica	No aplica	n#18. 1673431.B...
	22:51:03	22:52:05	00:01:02	60% - 78%			Perdida de prod...	Velocidad Están...	No aplica	No aplica	No aplica	No aplica	n#18. 1673431.B...
	22:46:06	22:48:42	00:02:36	Detenido	√			Linea operativa	Material	Inspeccion de m...	Por defectos	Otros	n#3-18-1673431-...
	22:39:22	22:43:05	00:03:43	78% - 95%			Perdida de prod...	Velocidad Están...	No aplica	No aplica	No aplica	No aplica	n#17-1673434-12...

Figura D16. Datos interrupciones

En la primera columna aparece un icono que cuando se hace clic en él, se presenta una página con la información de detalle de la demora (duración, clasificación, usuarios que la modificaron, etc.). La segunda columna es la hora de inicio real en que comenzó la demora en horas, minutos y segundos. La tercera columna representa la hora de fin de la demora en horas, minutos y segundos. La cuarta columna representa la duración de la demora. De la octava a la décima cuarta columna se muestra la clasificación que los usuarios hayan hecho de la demora o pérdida de productividad.

Clasificación de Interrupciones

Para realizar la clasificación de las interrupciones se debe hacer un clic en el link que se encuentra en la columna **Duración**. Este link se encuentra habilitado sólo para los usuarios con los permisos para clasificar y dentro del período permitido de acuerdo al perfil del usuario. Al hacer clic sobre el link se abrirá una pantalla como la que observamos en la Figura D18.

DATOS DE LA INTERRUPCIÓN	
Bobina:	1673431
Inicio:	13/09/2008 22:46:06
Fin:	13/09/2008 22:48:42
Durac. total:	00:02:36
Cambio bob.:	Sí

PARTICIÓN 1	
Tipo:	Electricas
Zona:	
Sistema:	
Subsistema:	
Evento:	
Subevento:	
Duración: [hh:mm:ss]	00:02:36
Justificación:	

Figura D18. Pantalla para clasificación de Interrupciones.

Como nota en la Figura D18, se muestran los datos correspondientes a la interrupción, la fecha y hora de inicio, la duración de la misma y otros datos de interés dependiendo de la planta y línea de producción. Bajo estos datos, notamos que se encuentra en blanco los campos que el usuario debe completar para la clasificación respectiva de la interrupción. Estos campos corresponden al tipo, zona, sistema, subsistema, evento, subevento y justificación de la interrupción. Para realizar la clasificación de las interrupciones hay que comenzar seleccionando en el campo Tipo, quién fue el responsable del evento. Pueden encontrarse causas Eléctricas, Mecánicas, Operativas, etc. Para el caso particular de paradas programadas, paros sindicales y otras causas que reducen el tiempo disponible de la planta, deberá seleccionarse "No disponibles". Es de notar, que se debe hacer la clasificación de forma progresiva, es decir el usuario debe seleccionar el tipo antes de seleccionar la zona, ya que los datos siguientes solo se cargan una vez que el usuario ha hecho su selección inicial. En el campo Justificación es posible ingresar un texto descriptivo que amplíe la información relacionada con el registro. A continuación vemos un ejemplo de una demora clasificada:

DATOS DE LA INTERRUPCIÓN	
Bobina:	1673446
Inicio:	13/09/2008 20:28:43
Fin:	13/09/2008 20:30:44
Durac. total:	00:02:01
Cambio bob.:	Sí

PARTICIÓN 1	
Tipo:	Eléctricas
Zona:	Sistema auxiliares
Sistema:	Sistema ventilacion de transformadores
Subsistema:	No aplica
Evento:	No aplica
Subevento:	No aplica
Duración: [hh:mm:ss]	00:02:01
Justificación:	n#7-1673446-1225x2.50x0.45mm- se inspecciona el material en la salida del 5to bast.

Figura D19. Demora Clasificada

En los casos donde las causas de la demora sean más de una, será necesario particionar la demora.

Al presionar este botón se creará un nuevo conjunto de campos para poder asignar la codificación correspondiente a la segunda causa de la demora.

En el caso de Demoras deberá imputarse a cada partición la duración que le corresponda, editando el valor en el campo "Duración:", ingresando el tiempo expresado en horas, minutos y segundos.

Al modificar la duración de una partición, el sistema recalcula automáticamente la duración de las demás particiones para ajustarlas a la duración total de la demora. Se pueden realizar un máximo de cuatro particiones para cada interrupción.

DATOS DE LA INTERRUPCIÓN			
Bobina:	1673371	Inicio:	13/09/2008 15:00:00
Fin:	13/09/2008 17:12:14	Durac. total:	02:12:14
Cambio bob.:	No		

PARTICIÓN 1		PARTICIÓN 2	
Tipo:	Electricas	Tipo:	Instrumentación/ee
Zona:	Bastidores	Zona:	Bastidores
Sistema:	Bastidor nº 5	Sistema:	Bastidor nº 1
Subsistema:	No aplica	Subsistema:	No aplica
Evento:	No aplica	Evento:	No aplica
Subevento:	No aplica	Subevento:	No aplica
Duración: [hh:mm:ss]	01:55:01	Duración: [hh:mm:ss]	00:17:13
Justificación:	se recibe la línea parada por estar saliendo el material manchado el personal mecanico detecta que la goma lado motor estaba despegada se agrego silicon para pegarla	Justificación:	falla el medidor de espesor del H1 por filtro sucio
Eliminar		Eliminar	

Figura D20. Particionar demora.

En la Figura D20, podemos ver que se han creado dos particiones para una demora de duración 02:12 minutos, y se ha repartido dicha duración en ambas particiones.

Si se ha creado una partición por error, y necesita ser eliminada, esto se hace presionando el botón "Eliminar" correspondiente a la partición que se quiera quitar, note que este botón estará disponible para cada conjunto de particiones que se ha creado. Al eliminar una partición el sistema colocará la duración que contenía la partición eliminada a la última partición creada.

Una vez escogidos todos los datos deseados, el usuario puede presionar el botón aceptar con el cual se guardarán los cambios efectuados, y se actualizará en la página principal, cerrándose la pantalla de la Figura D20.

Note que si el usuario no selecciona algunos de los campos para la clasificación entonces el sistema envía un mensaje de error indicando que debe escoger el campo faltante, y no le permitirá actualizar los datos hasta que no realice la selección adecuada, como se muestra en la Figura D21.

DATOS DE LA INTERRUPCIÓN					
Bobina:	1673430	Inicio:	13/09/2008 22:56:16	Fin:	13/09/2008 23:00:00
Durac. total:	00:03:44	Cambio bob.:	Sí		
Debe seleccionar la Zona					

PARTICIÓN 1	
Tipo:	Electricas
Zona:	
Sistema:	
Subsistema:	
Evento:	
Subevento:	
Duración: [hh:mm:ss]	00:03:44
Justificación:	

Figura 21. Mensaje de Error al Guardar Clasificación

Pantalla de detalles

Otra funcionalidad de las pantallas es la visualización de detalles, a la cual se tiene acceso al presionar el icono de la primera columna del listado de interrupciones.

En esta página se muestran a detalle los datos que corresponden a la demora elegida, en caso de que se hayan realizado varias particiones de la interrupción se pueden ver los datos de cada partición movilizándose a través de cada partición por las flechas que se encuentran en el área inferior de la página.

También puede ver el registro de las modificaciones que se han hecho sobre la interrupción dada, en esta área de registros se lista la fecha y el nombre del usuario que ha realizado la modificación respectiva.

Pantallas de reportes de demoras por tipos y secuencia

Para acceder a estos reportes se realiza del mismo modo que para clasificar una demora solo que en lugar de seleccionar directamente la instalación a consultar en el submenú que está desplegado se encuentran las opciones de Reporte por Tipo y Secuencia de Interrupciones, como se muestra en la Figura D9.

Al acceder a cualquiera de estas opciones se mostrará la página concerniente al reporte con los filtros necesarios para realizar la consulta. En la Figura D23 se muestra el Reporte por Tipo antes mencionado, Los controles de vital importancia en esta página son los alusivos a la instalación a consultar y los datos que se quieren consultar como el tipo de reporte y filtro los registros según el valor del Tipo, además se dispone de algunos de los botones mencionados en la introducción del manual como Agrupar, Generar reporte Pdf o Excel, por lo que el uso no representa mayor complejidad.

SIDOR Piso de Planta SI

SIDOR Clasf. APR Clasf. LAC Clasf. LAF Gestión SI

Clasf. APR Pellas Manejo Reporte x Tipo

General, Manejo Inst.: PELL Manejo Reporte: General Filtro: Todos Consultar

Año: 2008 Diario Mensual Anual Rango Turno

MMPS

No Clasificadas

Fecha	T	C	Sistema	Descripción de Evento	Justificación	Pérdida(Tn)	Usuario	Duración(Hrs.)
14/04/2008	3	A	No Clasificadas	No Clasificadas		1654.17		06:30:19
04/04/2008	3	A	No Clasificadas	No Clasificadas		1202.34		01:06:02
04/04/2008	2	B	No Clasificadas	No Clasificadas		6588.83		03:49:40
04/04/2008	1	C	No Clasificadas	No Clasificadas		9592.66		07:39:59
03/04/2008	3	A	No Clasificadas	No Clasificadas		6311.45		05:15:50
03/04/2008	2	B	No Clasificadas	No Clasificadas		6800.73		05:20:18
03/04/2008	1	D	No Clasificadas	No Clasificadas		1599.32		00:40:03
02/04/2008	3	A	No Clasificadas	No Clasificadas		2168.28		00:54:59
02/04/2008	2	C	No Clasificadas	No Clasificadas		4830.10		02:01:42
02/04/2008	1	D	No Clasificadas	No Clasificadas		2711.74		01:15:45
01/04/2008	3	B	No Clasificadas	No Clasificadas		2725.01		01:15:25
01/04/2008	2	C	No Clasificadas	No Clasificadas		5848.19		03:45:07

Figura D22. Reporte por tipos de demoras.

A continuación la descripción del Reporte Secuencia de Interrupciones en la Figura D23.

SIDOR Piso de Planta SI

SIDOR Clasf. APR Clasf. LAC Clasf. LAF Gestión SI

Clasf. APR Pellas Manejo Secuencia de interrupciones

Secuencia de Interrupciones Instalación: PELL Línea: Línea A Registro: Todos Tipo: Todos Consultar

Día: 08/09/2008 Diario Mensual Rango Turno

INTERRUPCIONES

Fecha	Hora Inicio	Hora Fin	Duración	Estado	Ton. Perd.	Tipo	Zona	Sistema	Subsistema	Evento	Subevento	Justificación
08/09/2008	18:57:40	19:04:24	00:06:44	Detenido	28.8439826965332							
08/09/2008	15:56:01	16:19:00	00:22:59	Detenido	165.647430419922	Operativa propia	Sistema De Aire Comprimido	Flujo Aire Comprimido A Planta Fr-1084	Flujo Aire Comprimido A Planta Fr-1084	No Existe el Evento	No Existe el Evento	
08/09/2008	12:22:43	15:00:00	02:37:17	Bajo Ritmo	145.511016845703							
08/09/2008	12:17:03	12:22:43	00:05:40	Detenido	40.8213577270508							
08/09/2008	07:00:00	12:17:03	05:17:03	Bajo Ritmo	328.744323730469							
			08:29:43		710							

INDICADORES

Tiempo no Disponible (H:M:S):	00:00:00	Utilización Neta (T.Net/T.Disp.)*100:	97.54
Tiempo Disponible (H:M:S):	24:00:00	Cantidad de Interrupciones Detectadas:	3
Tiempo Demoras (H:M:S):	00:35:23	Total Pérdida de Producción:	474.26
Total Tiempo Neto (H:M:S):	23:24:37	Total Ganancia de Producción:	1795.75

Figura D23. Secuencia de Interrupciones

En este reporte se muestran las interrupciones secuencialmente fueron ocurriendo en la línea de producción para una fecha y turno específico, pudiendo filtrar solo las interrupciones de Tipo en la que se tenga interés además del estatus de la interrupción en el sistema.

Pantallas de Gestión de Catálogos

Para entrar en la pantalla de gestión de catálogos se debe acceder por medio del encabezado de menú en la pestaña Gestión SI, la planta y luego en el submenú la instalación donde se quiere realizar las acciones, en el caso necesario si la instalación tiene líneas tendrá que navegar por ellas hasta encontrar la deseada, seleccionando la opción Gestión de Catálogos que se presenta, en la Figura D24 se puede observar un ejemplo.

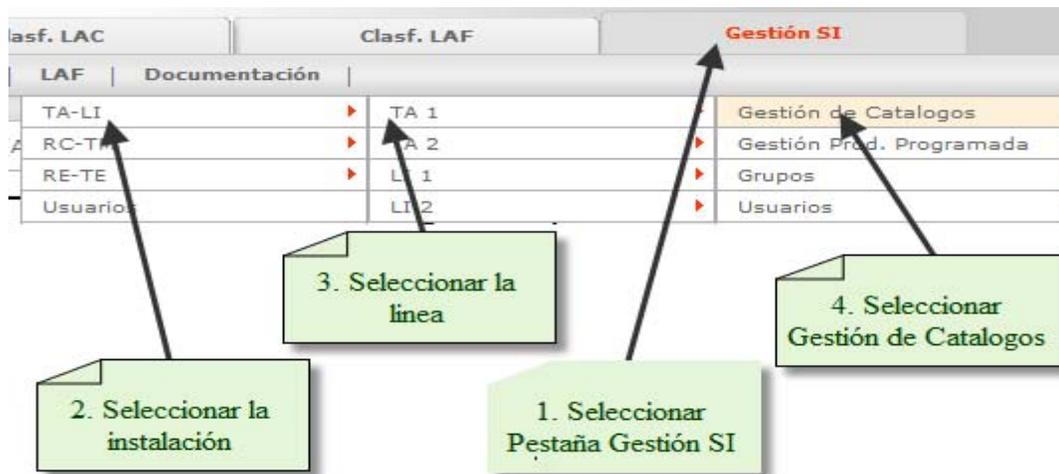


Figura D24. Acceder a Gestionar Catálogos.

Luego se mostrará la pantalla de interfaz de Gestión de catálogos para la instalación o línea seleccionada. Como se muestra en la siguiente figura:

Línea	ID	Descripción	Inicio Vigencia	Fin Vigencia	Editar	Eliminar
TA 1	1	1, 2 o 3 bobina cambio de cilindro bastidor 5				
TA 1	AGUAS	Agua				
TA 1	A1AC	Aire acondicionado				
TA 1	A1CO	Aire comprimido				
TA 1	ALTE	Alta tension				
TA 1	AUTO	Automacion				
TA 1	21	Bajo Ritmo por Problemas Sindicales	30/06/2006			
TA 1	BAS1	Bastidor nº 1				
TA 1	BAS2	Bastidor nº 2				
TA 1	BAS3	Bastidor nº 3				
TA 1	BAS4	Bastidor nº 4				
TA 1	BAS5	Bastidor nº 5				
TA 1	14	Bobina floja entrada				
TA 1	CONJ	Conjunto de salida				
TA 1	CDP	Control de flujo				

Figura D25. Pantalla de Gestión de Catálogos.

Los usuarios pueden agregar, modificar y eliminar registros solo cuando poseen la permisología necesaria.

Pantalla de Gestión de Productividad

De la misma forma se puede acceder a la programación de las metas para una determinada línea de producción.

Inicio Vigencia	Turno	Productividad [t/h]	Tiempo Efectivo	Efectividad	Capacidad Instalada	Usuario	Modificar	Eliminar
29/12/2010	1	90.57224	7.908	98.85	133	Jose Luis Cumana		
28/12/2010	1	83.71483	7.908	98.85	133	Jose Luis Cumana		
27/12/2010	1	90.57224	0	98.85	133	Jose Luis Cumana		
11/12/2010	1	90.57224	7.908	98.85	133	Jose Luis Cumana		
10/12/2010	1	90.57224	7.858575	98.85	133	Jose Luis Cumana		
03/12/2010	1	90.57224	7.908	98.85	133	Jose Luis Cumana		
02/12/2010	1	83.71483	7.908	98.85	133	Jose Luis Cumana		
01/12/2010	1	90.57224	0	98.85	133	Jose Luis Cumana		

Figura D26. Pantalla de Gestión de Productividad.

Los datos de producción solo podrán ser modificados o eliminados mientras no se encuentren vigentes de no ser así los link de formularios estarán deshabilitados.

Se pueden realizar consultas para los últimos 3 años, y se realizan mensualmente ya que las metas las instalaciones las establecen cada mes.

Para cargar la productividad desde un documento Excel este debe cumplir con el formato establecido por la Dirección de Producción Industrial.

Se debe presionar el botón Excel y se mostrara una ventana emergente donde podemos buscar la dirección del sitio donde se encuentra alojado el documento. Una vez realizado esto presionar aceptar.

Cabe destacar que si los datos ya se encuentran cargados en el sistema recibirá un mensaje advirtiéndolo sucedido, con la finalidad de evitar solapamiento.

Pantalla de Gestión de Patrones.

A través de la pantalla principal de gestión de patrones podemos ver el historial de patrones de rotación existente en la instalación hasta el que se encuentra vigente para el momento de consulta, brindando la posibilidad de tomar como plantilla alguno de los anteriores para establecer la misma rotación nuevamente.

Mientras un patrón no se encuentre en estado de vigencia vencida o activo no se podrá modificar ni eliminar, estas acciones solo podrán ser posibles cuando los registros de patrones pertinentes aun no hayan entrado en fecha de vigencia.

A continuación la pantalla principal de gestión de patrones:

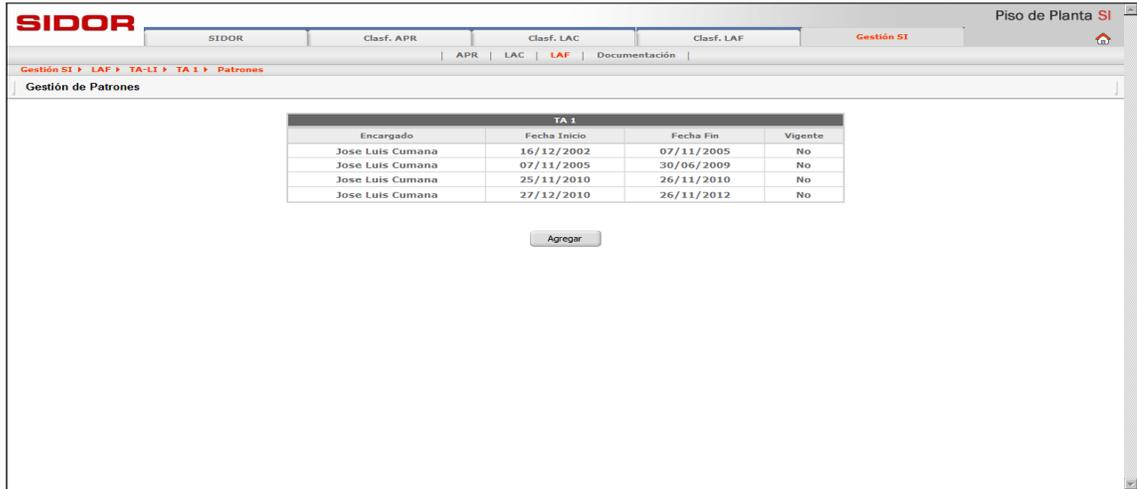


Figura D27. Pantalla Principal de Gestión de Patrones.

Si seleccionamos alguno de los registros lo podremos copiar o solo observar si ya está vencido o activo. Si presionamos el botón agregar se mostrara una nueva pantalla con un formulario donde se podrá establecer la nueva programación, seteando primero el número de cuadrillas que rotaran y la fecha de inicio y fin del patrón. Luego se presiona el Botón Go, para que se presente la matriz de programación, como se muestra en la figura siguiente:

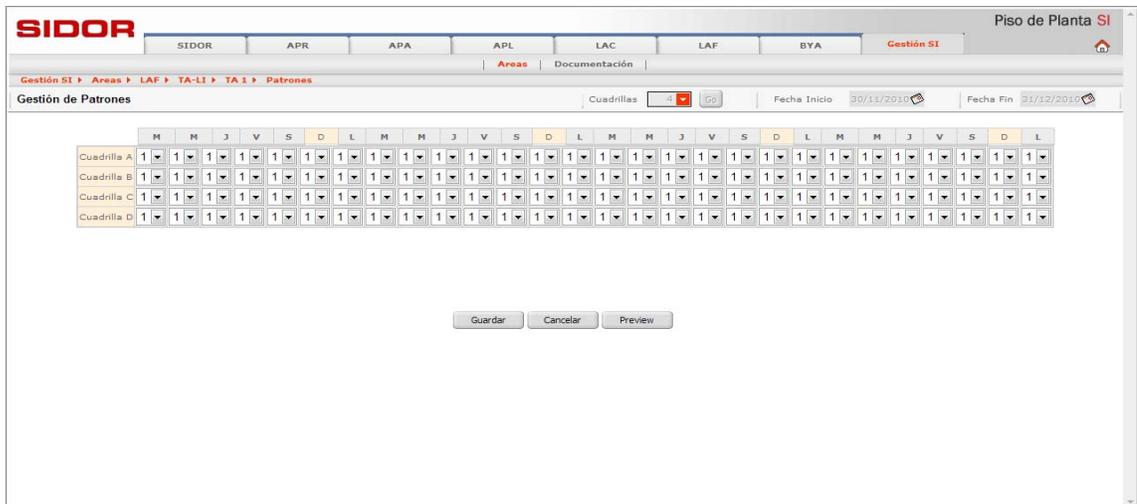


Figura D28. Pantalla de Carga de Patrón de rotación

Pantalla de gestión de Usuarios.

La interfaz en este caso es totalmente similar a las experiencias que se han tenido anteriormente con las pantallas anteriores.

Los registros mostrados pueden ser editados o modificados y las pagina siguiente se setea dependiendo del perfil de usuario que se esté gestionando.

A continuación se muestra la pantalla de edición de un Usuario de tipo Operador en la Figura D29.

The screenshot shows a web application window titled "Mantenimiento de Usuarios". It contains two main sections: "Modificación de Usuario" and "Asociación a Cuadrillas Globales".

Modificación de Usuario

Login:	<input type="text" value="sirezr"/>
CI:	<input type="text"/>
Nombre:	<input type="text" value="RODRIGUEZ"/>
Apellido:	<input type="text" value="Alfredo"/>
Super usuario:	<input type="checkbox"/>
Perfil:	<input type="text" value="Operador"/>
Fecha deshabilitación:	<input type="text"/> [Limpiar]

Asociación a Cuadrillas Globales

Cuadrilla global:	<input type="text" value="Cuadrilla C / LI 1-2"/>	
Nivel:	<input type="text" value="Operador"/>	
Fecha Inicio	Cuad. Global	Nivel
20/01/2009	Cuadrilla C/LI 2	Operador

Buttons: Guardar, Eliminar, Cancelar

Figura D29. Pantalla de Carga de Permisos y Datos de Usuario.

Apéndice E. Pruebas de Integración

PRUEBAS DE INTEGRACIÓN DE LOS MODULOS DE LA APLICACIÓN



Figura E1. Resultado de la integración del modelo de navegación y el módulo Gestión de Productividad Programada de la fase de elaboración.



Figura E2. Resultado de la integración del modelo de navegación y el módulo Gestión de Catálogos de la fase de construcción.

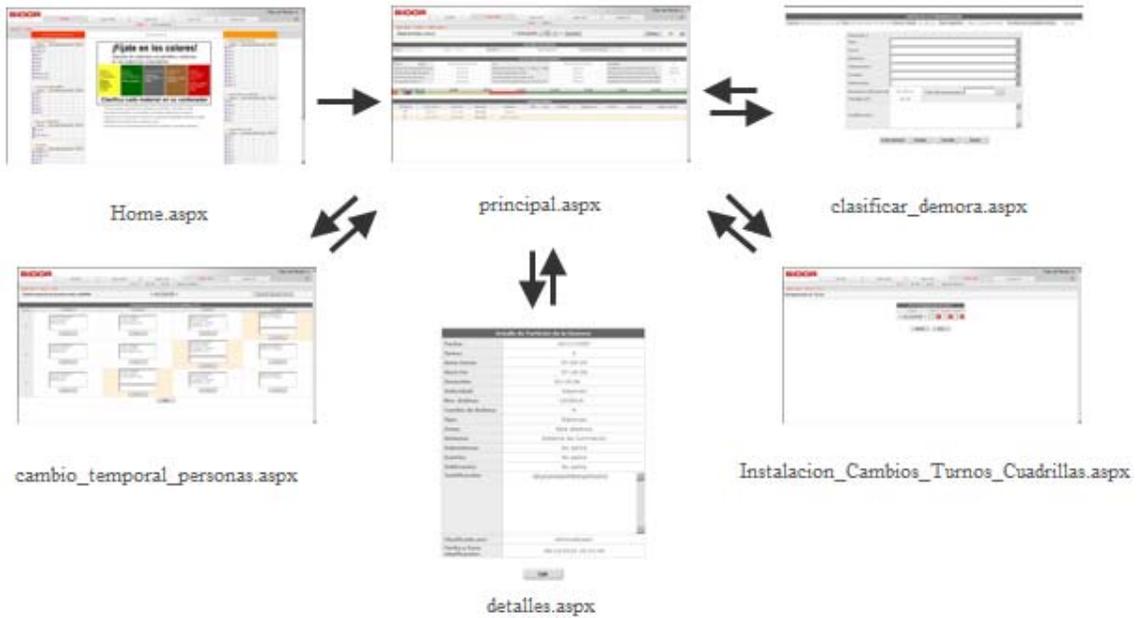


Figura E3. Resultado de la integración del modelo de navegación y el módulo Gestión de Clasificación de demoras de la fase de construcción.

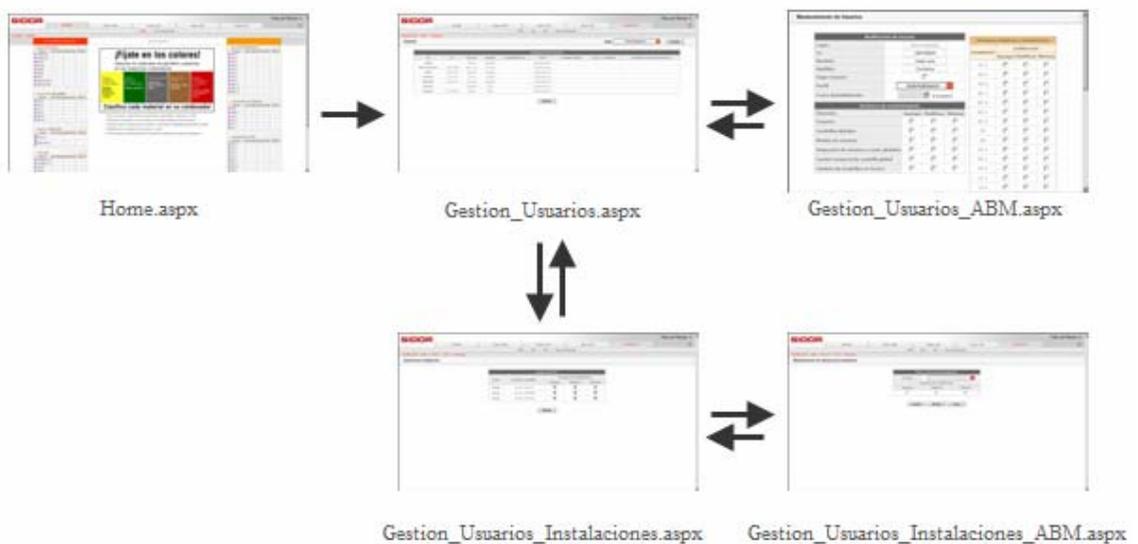


Figura E4. Resultado de la integración del modelo de navegación y el módulo Gestión de Catálogos de la fase de construcción.



Figura E5. Resultado de la integración del modelo de navegación y el módulo Gestión de Patrones de la fase de construcción.

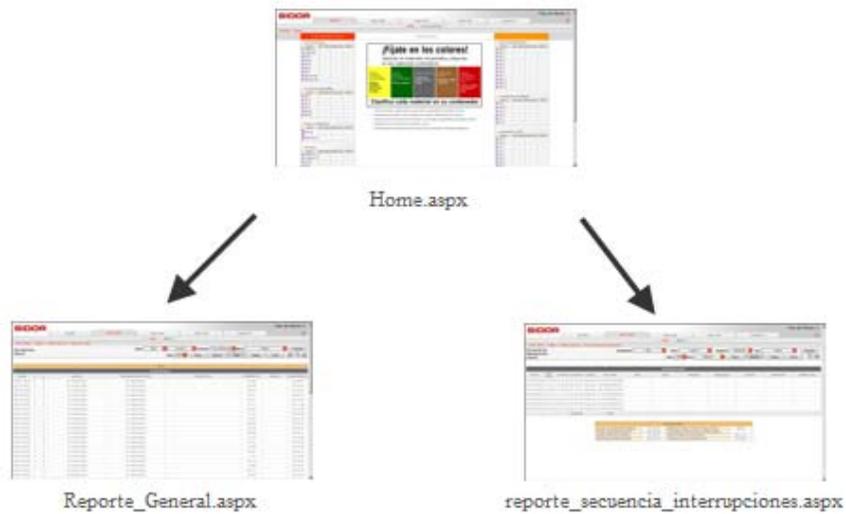
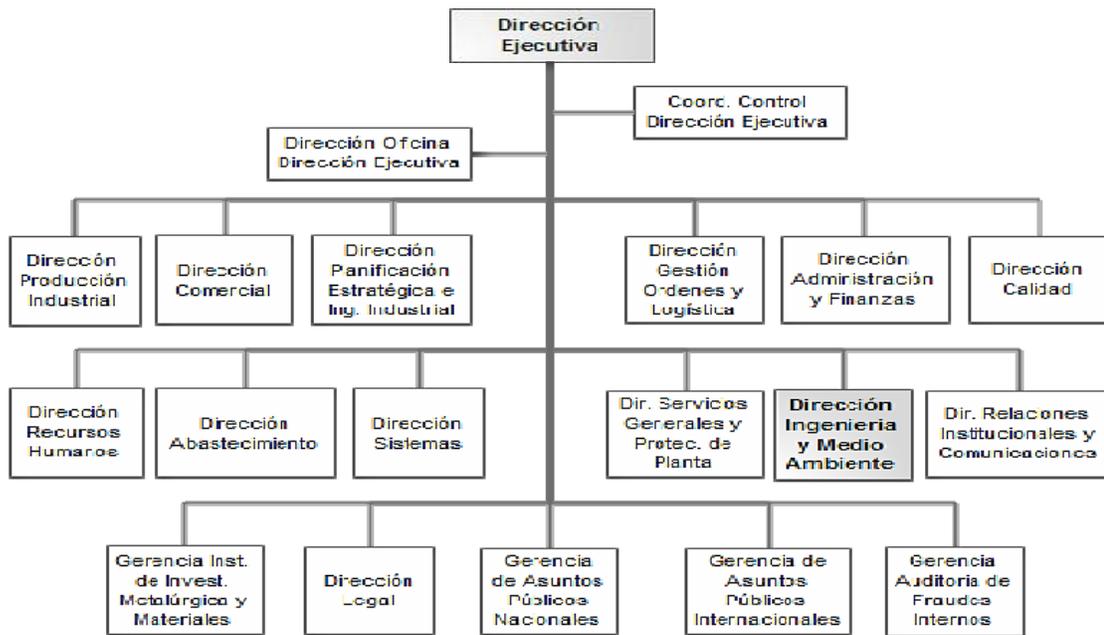


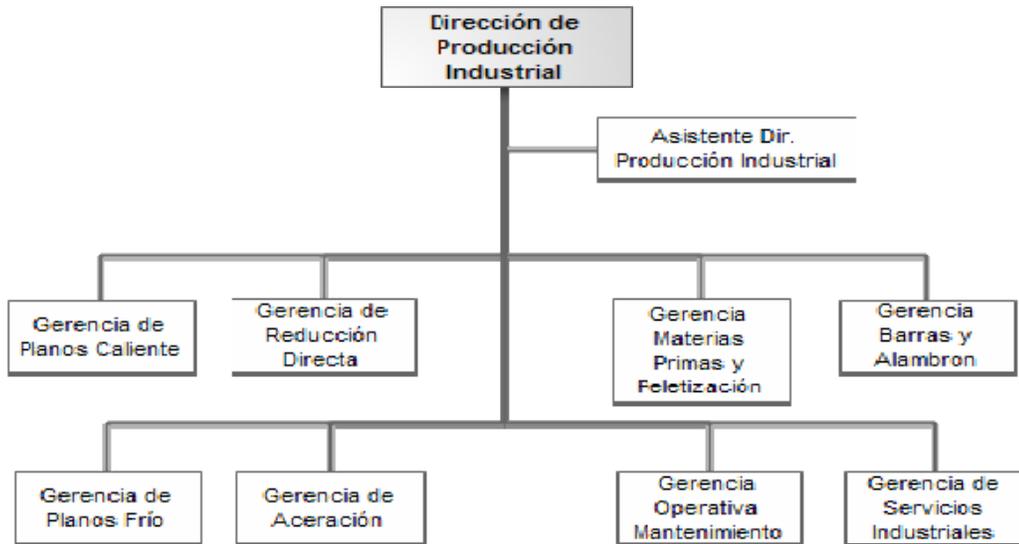
Figura E5. Resultado de la integración del modelo de navegación y el módulo de Reportes de la fase de construcción.

ANEXOS

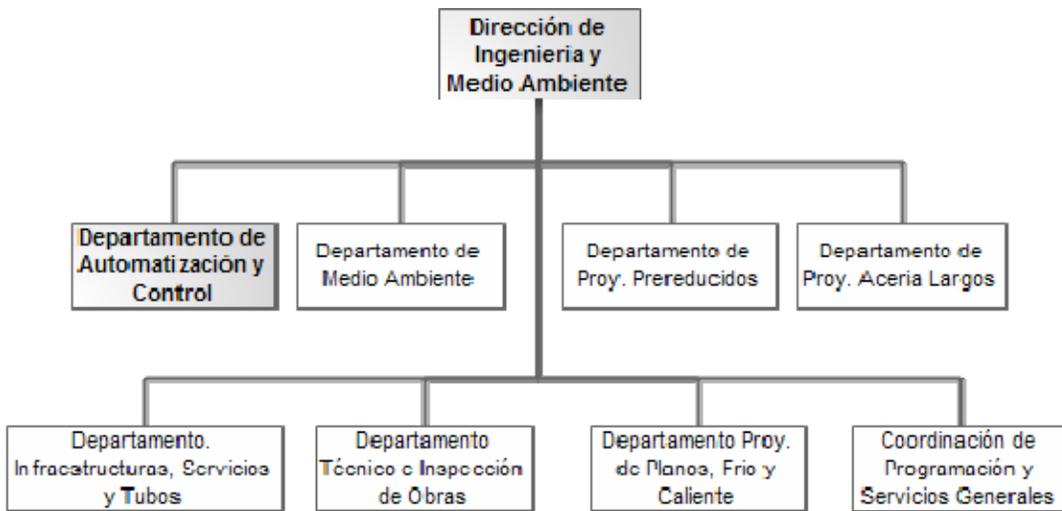
Anexo A. Estructura organizativa de SIDOR.



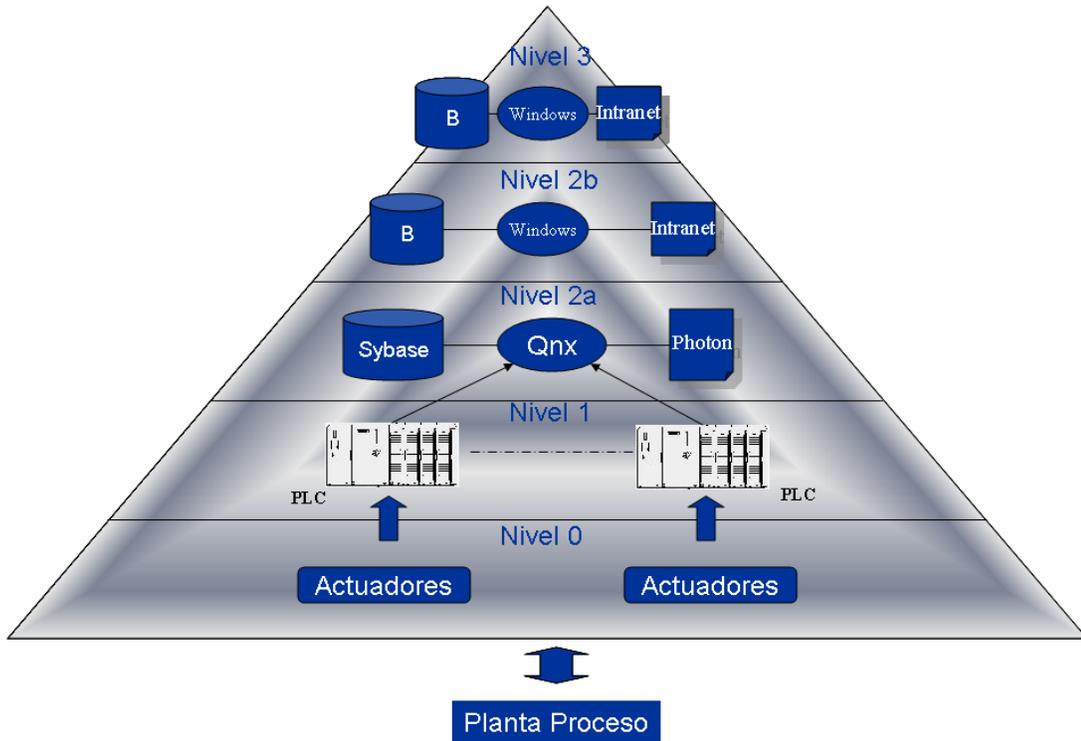
Anexo B. Dirección de Producción Industrial



Anexo C. Estructura Organizativa de la Dirección de Ingeniería y Medio Ambiente.



Anexo D. Esquema piramidal del Departamento de Automatización y Control.



Nivel 0: Se encuentran los elementos de campo (sensores, actuadores, etc.).

Nivel 1: Se encuentran los elementos de control (PLC's) y monitoreo de las variables de campo como por ejemplo la temperatura. El PLC recibe toda la información de los elementos de campo.

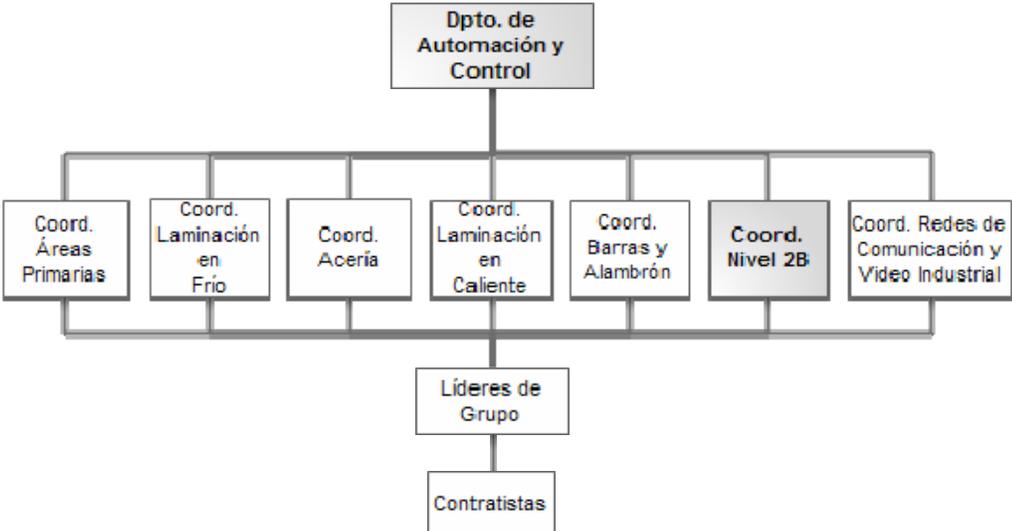
Nivel 2: Presenta divisiones:

El **nivel 2A**, en donde se realiza el monitoreo y control en tiempo real del proceso. Es aquí donde intervienen los operadores de cabina a visualización y seguimiento en tiempo real del proceso.

En el **nivel 2B**, basado en tecnología Web se lleva a cabo el monitoreo y presentación Web de datos a través de los sitios de las distintas líneas de producción.

Nivel 3: Nivel Gerencial es donde se desarrolla la explotación de los datos, análisis estadístico del proceso y control de gestión. Intervienen las gerencias de las líneas.

Anexo E. Estructura Organizativa del Departamento de Automatización y Control.



HOJA DE METADATOS

Hoja de Metadatos para Tesis y Trabajos de Ascenso – 1/5

Título	APLICACIÓN WEB PARA LA GESTIÓN DEL SISTEMA DE INTERRUPTORES DE LA SIDERÚRGICA DEL ORINOCO ALFREDO MANEIRO (SIDOR)
Subtítulo	

Autor(es)

Apellidos y Nombres	Código CVLAC / e-mail	
Cumana Dominguez, José Luis	CVLAC	18073924
	e-mail	joseluiscumana@gmail.com
	e-mail	sk8globe_01@hotmail.com

Palabras o frases claves:

Aplicación Web
Proceso Unificado de Desarrollo de Software
UML
Sistema de Interrupciones
SIDOR
Línea de producción
Productividad programada
Turnos laborales
Patrones de rotación de cuadrillas
Catálogos
Justificación de demoras
Toma de decisiones
Visual Studio.NET
ASP.NET
SQL Server
Internet Information Server

Hoja de Metadatos para Tesis y Trabajos de Ascenso – 2/5

Líneas y sublíneas de investigación:

Área	Subárea
CIENCIAS	Informática

Resumen (abstract):

Se desarrolló una aplicación *Web* para la gestión del Sistema de Interrupciones (SI) de la Siderúrgica del Orinoco Alfredo Maneiro (SIDOR), que se enmarca dentro de la intención de la empresa de centralizar la información, debido a la necesidad de saber con precisión el estado de una línea de producción, además de las pérdidas sufridas en los respectivos turnos laborales en base a las metas propuestas por la Dirección de cada planta, permitiendo que el personal interesado pueda corroborar que la información suministrada de la línea de producción sea clara, exacta y veraz. Ante esto, la aplicación *Web* permite resolver los problemas relacionados con el registro, recuperación y organización de la información referente a los procesos de gestión de catálogos, metas de producción mensual de las plantas, clasificación de demoras, gestión de patrones de rotación de cuadrillas por turnos y de usuarios, brindando un medio para controlar funcionamiento del SI; lo que produce una toma de decisiones más acertada. La aplicación abarca todas las plantas que se encuentran funcionando bajo el SI, permitiendo el manejo de toda su información, apoyándose en varias bases de datos según sea la planta. Este desarrollo estuvo dirigido por el Proceso Unificado de Desarrollo de Software planteado por Jacobson, Booch y Rambaugh (1999), y UML como lenguaje de modelado para la notación de los diagramas que permiten el entendimiento más detallado del sistema. En cuanto a la codificación y construcción del sistema, se utilizó el *Framework Visual Studio.NET 2003* específicamente el lenguaje de programación *Visual Basic.NET* para la creación de las páginas *Web* con tecnología *ASP.NET*, *SQL Server 2000* como manejador de base de datos, *Javascript* como lenguaje de programación interpretado y basado en objetos para la validación de formularios, *Internet Information Server (IIS)*, como servidor *Web* y *Windows Server 2003* como sistema operativo.

Hoja de Metadatos para Tesis y Trabajos de Ascenso – 3/5

Contribuidores:

Apellidos y Nombres	ROL / Código CVLAC / e-mail	
Romero, Carmen Victoria	ROL	CA <input type="checkbox"/> AS <input checked="" type="checkbox"/> TU <input type="checkbox"/> JU <input type="checkbox"/>
	CVLAC	10947403
	e-mail	cvromerob@gmail.com
Aguiar, Danelina	ROL	CA <input checked="" type="checkbox"/> AS <input type="checkbox"/> TU <input type="checkbox"/> JU <input type="checkbox"/>
	CVLAC	11831828
	e-mail	danelina@cantv.net
Carmona, Claudia Patricia	ROL	CA <input type="checkbox"/> AS <input type="checkbox"/> TU <input type="checkbox"/> JU <input checked="" type="checkbox"/>
	CVLAC	13275156
	e-mail	ccarmona@udo.edu.ve
Fuentes, Ana Teresa	ROL	CA <input type="checkbox"/> AS <input type="checkbox"/> TU <input type="checkbox"/> JU <input checked="" type="checkbox"/>
	CVLAC	12666425
	e-mail	afuentes_marquez@hotmail.com
Agostini, Carlos	ROL	CA <input type="checkbox"/> AS <input checked="" type="checkbox"/> TU <input type="checkbox"/> JU <input type="checkbox"/>
	CVLAC	12013608
	e-mail	cagostini@sidor.com

Fecha de discusión y aprobación:

Año	Mes	Día
2011	02	17

Lenguaje: SPA

Hoja de Metadatos para Tesis y Trabajos de Ascenso – 4/5

Archivo(s):

Nombre de archivo	Tipo MIME
TESISCUMANAJ.doc	Aplication/word

Alcance:

Espacial: NACIONAL(Opcional)

Temporal: INTEMPORAL (Opcional)

Título o Grado asociado con el trabajo: Licenciatura en Informática

Nivel Asociado con el Trabajo: Licenciado

Área de Estudio: Sistemas de Información

Institución(es) que garantiza(n) el Título o grado:

UNIVERSIDAD DE ORIENTE (UDO)

Hoja de Metadatos para Tesis y Trabajos de Ascenso – 5/5

Derechos:

Yo, José Luis Cumana Dominguez, titular de la Cédula de Identidad número: V-18.073.924, como autor del trabajo de grado presentado en este documento, autorizo a la Universidad de Oriente en la publicación y uso de la información contenida en éste, siempre y cuando se respeten los derechos de autor.



AUTOR
José Luis Cumana



ASESOR
Carmen Romero



ASESOR INDUSTRIAL
Carlos Agostini



COASESOR
Dianelia Aguiar



JURADO
Claudia Carmona



JURADO
Ana Fuentes

POR LA COMISIÓN DE TESIS:

