

Universidad de las Ciencias Informáticas

Facultad 3



**Componente para la extracción de registros de eventos
en formato XES del sistema ZUN Suite.**

Trabajo de Diploma para optar por el Título de
Ingeniero en Ciencias Informáticas

Autores:

Aliuska Batista Corella

Yordany Aguilera Martínez

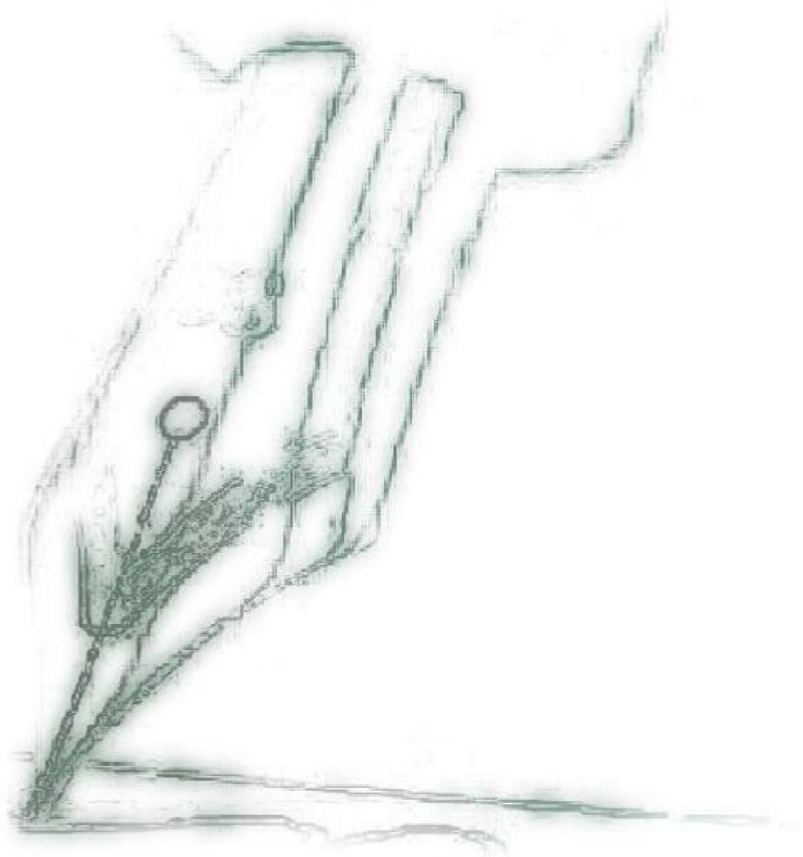
Tutor: MSc. Damián Pérez Alfonso

Co-Tutores: Ing. Osiel Fundora Ramírez

Ing. Eudel Pupo Hernández

La Habana, Junio de 2015

“Año 57 de la Revolución”



"Existe una fuerza motriz más poderosa que el vapor, la electricidad y la energía atómica: la voluntad."

Albert Einstein

DECLARACIÓN DE AUTORÍA

Declaramos ser los únicos autores del presente trabajo que lleva como título: “Componente para la extracción de registros de eventos en formato XES del sistema ZUN Suite” y autorizamos a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmamos la presente a los ____ días del mes _____ del año _____.

Aliuska Batista Corella

Autor

Yordany Aguilera Martínez

Autor

MSc. Damián Pérez Alfonso

Tutor

De Aliuska:

*A mis padres por todo el amor y el cariño que me han dado y ser mi punto de apoyo
para tantas cosas.*

A mi hermana por ser esa otra parte de mí que me motiva a seguir.

Los amo a todos.

De Yordany:

*A mis padres y mis hermanos por ser las personas más importantes en mi vida,
por toda la confianza y el amor de tantos años.*

De Aliuska:

A mi mamita, por darme la vida, ser mi guía, mi amiga, mi todo, por enseñarme siempre a ser mejor persona y sacar lo mejor de mí, te adoro. A mi papito, que sin ti no sería lo que soy hoy, me haz enseñado tantas cosas que no me darían las hojas para escribirlas, gracias por apoyarme en mis decisiones y estar de mi lado siempre, te adoro. A mi hermanita, que también me impulsas a seguir adelante, para que veas un ejemplo en mí, en tu hermana que te quiere con la vida. A mi abuelos, por su preocupación por mi bienestar y darme tanto cariño, aunque a algunos ya no los tengo conmigo siempre los tengo presentes. En general a toda mi familia, a mis tíos, mis primos en especial a mi tío René, a la Gorda y a Taty por siempre estar al pendiente de mí y quererme mucho. A mi prima Liuba que sin su ayuda no hubiera sido posible la realización de esta tesis, por todo su apoyo y dedicación a pesar de su limitado tiempo. A todo el personal del hotel Club Amigo Atlántico Guardalavaca de Holguín que igualmente nos ayudaron muchísimo en la realización de la tesis, gracias a todos.

A mi Tito, que ha sido más que un compañero de tesis para mí, mi pareja, mi consejero, mi amigo, mi vida, aunque hubo veces que no te soporté en todo este tiempo de la tesis por ponerte tus espejuelos de doctor, quiero decirte que te amo mucho y que sin ti no hubiera hecho todo esto posible. A mis suegros, que desde el primer momento que los conocí siempre me han apoyado y me han dado todo su cariño, son como unos segundos padres para mí, los quiero mucho.

A Damián, que ha sido más que un tutor para mí, se ha convertido en un buen amigo, regañón en muchas ocasiones pero que ha sabido guiarnos para hacernos los profesionales que somos hoy. A Pupo y a Osiel, por también ser parte de esta nueva amistad y apoyarnos y ayudarnos en todo momento, muchas gracias.

Al grupo de investigación de minería de procesos por sus recomendaciones y correcciones que contribuyeron a perfeccionar nuestra tesis, a todos muchas gracias.

A mis amigos, a los que he tenido desde la primaria hasta los que he hecho en estos 5 años, gracias a todos por haber formado parte de mi vida y haber compartido con ustedes momentos inolvidables. A mis compañeros de aula del 3502 y del 3503, que me recibieron con los brazos abiertos y supieron ganarse mi cariño, los voy a extrañar mucho.

A todas las personas que se me quedan por mencionar pero que de alguna manera u otra han tocado mi vida y me han hecho lo que soy hoy.

De Yordany:

A mi mamá porque la palabra madre le queda pequeña, has sido para mí esa virgencita que va conmigo a todas partes, por demostrarme esa confianza incondicional en todo momento y por guiarme por el buen camino de la vida. A mi papá por inculcarme con su ejemplo los verdaderos valores que hacen grande a un hombre; la honestidad, la solidaridad, la sencillez y el carácter. A mi hermano Yosvany por ser mi guía y mi modelo a seguir, por demostrarme que para lograr grandes cosas hay que hacer grandes sacrificios y porque aún en la distancia has sido más que un hermano. A mi hermano Onilber por los consejos, por el apoyo durante toda mi carrera y por estar en los momentos difíciles. A mis abuelos por todo su apoyo y por todas las cosas buenas que hemos pasado juntos. A todos mis tíos, especialmente a Neysa, Onelio y Redel, los considero mis otros padres porque así se han portado conmigo, han estado presentes en todo momento y eso lo llevo por dentro. De forma general a toda mi familia por la dedicación y por inspirarme a realizar este sueño.

A mi compañera de tesis Tita como todos la conocen, muchas gracias por ser mi sustento, por ser más que una compañera, por ser mi amiga y por ayudarme a ser mejor persona en todos los aspectos de la vida, te debo mucho y espero podamos convertir más sueños en realidad. A mis suegros por la confianza, por la constancia y por contribuir a que hoy estemos juntos logrando este sueño.

A mi tutor, gracias por las enseñanzas, por los consejos, por los regaños y muy importante gracias por convertirnos en Ingenieros, eres el ejemplo perfecto de cómo ser un buen profesional, has sido profesor, tutor, amigo e incluso árbitro cuando había desacuerdo entre nosotros. A nuestros cotutores Osiel y Pupo, ambos han sido el

complemento perfecto para este equipo de trabajo y en general han aportado mucho a nuestra tesis desde un inicio hasta el final.

Quiero agradecer a todas mis amistades, a los que vienen conmigo desde la secundaria, el preuniversitario y a los muchos que he encontrado en esta universidad, a todos muchas gracias por ser tan especiales conmigo. Agradecimientos especiales para Ronald, has sido un hermano para mi desde que éramos niños y en un día como hoy te agradezco por todos los consejos y por tu amistad incondicional. A todos mis compañeros del grupo por la experiencia vivida en estos 5 largos años, a los integrantes del Grupo de Investigación de Minería de Procesos gracias por todas las recomendaciones, con todos ustedes he aprendido muchas cosas buenas.

A mis vecinos por la preocupación y el cariño y a todos aquellos que no he mencionado pero que saben lo mucho que les estoy y estaré agradecido.

De manera general agradezco a todas las personas que de buena fe han contribuido de una forma u otra a mi formación como profesional, muchas gracias.

Resumen

La minería de procesos es una disciplina de investigación que proporciona técnicas para descubrir, monitorear y mejorar los procesos de negocio en las organizaciones. Las mismas dedican tiempo y esfuerzo para analizar sus procesos con el objetivo de mejorar su desempeño organizacional y operacional. En el sector del turismo en Cuba, uno de los sistemas de información que se utiliza es el ZUN Suite, el cual registra los datos asociados a la estancia de los clientes en el hotel. El análisis de los procesos con técnicas tradicionales ocasiona, que no se posea una visión clara de cómo se ejecutan realmente dichos procesos en las instituciones hoteleras. Por ello la investigación se enfoca en el desarrollo de un componente que permita la extracción de registros de eventos a partir de las bases de datos de estas instituciones, que contribuya a disminuir el tiempo requerido para el análisis de los procesos de negocio donde se utiliza el sistema de información ZUN Suite. Como resultado se obtiene una descripción de los procesos de gestión hotelera cuyas ejecuciones son registradas en el sistema ZUN Suite. Además, un componente para el marco de trabajo de la minería de procesos ProM, que permita la extracción de los registros de eventos. La solución propuesta disminuye el tiempo en el análisis de los procesos en las instalaciones hoteleras, lo cual puede mejorar la atención a los clientes.

Palabras claves: Instituciones hoteleras, minería de procesos, registros de eventos, ZUN Suite.

Abstract

Process mining is a research discipline that provides techniques to discover, monitor and improve processes in organizations. They devote time and effort to analyze their processes in order to improve its organizational and operational performance. In the tourism sector in Cuba one of the information systems used is ZUN Suite, which records data associated with guests' stay at the hotel. The analysis of the processes with traditional techniques that causes a clear vision of how not possess are actually running on the hotel institutions. The research focuses on the analysis of business processes, applying mining processes from event logs extracted ZUN Suite system. As a result a description of the processes of hotel management whose executions are recorded in the system is obtained ZUN Suite. In addition a component framework for mining ProM process that allows the extraction of event logs. The proposed solution reduces the time to analyze processes in the hotel facilities, which may result in better customer service.

Keywords: *Event log, hotel institutions, process mining, ZUN Suite.*

Índice

Introducción	13
Capítulo 1: Fundamentación teórica.....	18
1.1 Introducción	18
1.2 Minería de procesos.....	18
1.3 Técnicas de minería de procesos	19
1.4 Registro de eventos.....	20
1.5 Formato XES	23
1.6 Marco de trabajo para la minería de procesos ProM.....	24
1.7 Algoritmos de descubrimiento.....	24
1.8 Algoritmos de conformidad.....	27
1.9 Gestión de Procesos	29
1.10 Gestión Hotelera	30
1.11 ZUN Suite.....	31
1.12 Tecnologías para el desarrollo de la solución	33
1.13 Metodología para el desarrollo de la solución.....	34
1.14 Conclusiones parciales	36
Capítulo 2: Análisis y diseño de la solución	37
2.1 Introducción	37
2.2 Descripción del proceso.....	37
2.3 Conceptos asociados a la investigación	39
2.4 Reglas del negocio	40
2.5 Propuesta de solución.....	41
2.5.1 Diseño del algoritmo.....	43
2.6 Fase de planificación.....	46
2.6.1 Historias de usuario.....	46
2.6.2 Plan de iteraciones	48
2.7 Requisitos del sistema	48
2.7.1 Requisitos funcionales del software.....	49

2.7.2 Métricas para la validación de requisitos	49
2.7.3 Requisitos no funcionales del software	50
2.8 Conclusiones parciales	52
Capítulo 3: Implementación y validación de la solución	53
3.1 Introducción	53
3.2 Fase de desarrollo	53
3.2.1 Estándares de codificación	53
3.2.2 Diagrama de Componentes	54
3.3 Fase de pruebas	55
3.2.1 Pruebas de aceptación	56
3.4 Descripción del análisis de los registros de eventos.....	58
3.5 Extracción de los registros de eventos.....	59
3.6 Caso de estudio	64
Conclusiones generales	75
Recomendaciones	76
Referencias bibliográficas	77

Introducción

Las organizaciones, inmersas en entornos y mercados competitivos y globalizados necesitan gestionar sus actividades y recursos eficientemente, esto ha generado la adopción de herramientas y metodologías para la gestión de procesos de negocio. Para monitorear y gestionar un proceso de negocio, se requiere una solución que utilice el enfoque de Gestión de Procesos de Negocio (BPM¹). El objetivo de esta metodología es crear una mejor perspectiva de los procesos de negocio para poder optimizarlos y automatizarlos en el mayor grado posible mediante el uso de aplicaciones informáticas.

La Gestión de Procesos de Negocio es una metodología corporativa cuyo objetivo es mejorar la eficiencia en los procesos de una organización. La misma utiliza métodos, técnicas y software para diseñar, ejecutar, controlar y analizar procesos de negocio que involucran personas, organizaciones, aplicaciones, documentos y otras fuentes de información (Weske 2007). La Gestión de Procesos de Negocio incluye conceptos, métodos y un conjunto de técnicas que se han desarrollado para apoyar el diseño, administración, configuración, aprobación y el análisis de los procesos de negocio.

El ciclo de vida de BPM, se compone de siete fases que están relacionadas entre sí. Las fases se organizan en una estructura cíclica mostrando sus dependencias lógicas. Estas dependencias no implican un estricto orden temporal en el que las fases tienen que ser ejecutadas (Weske 2007).

Una de las fases del ciclo de vida BPM es la evaluación, la cual utiliza la información disponible para evaluar y mejorar los modelos de procesos de negocio y sus implementaciones. Una alternativa para la mejora de estos procesos es la utilización de técnicas de análisis automático que parten de los registros de ejecución de los procesos en formato XES², como el monitoreo de actividades del negocio y las técnicas de minería de procesos, tratando de identificar la calidad del modelo de proceso del negocio y su adecuación al ambiente de ejecución.

Consideraciones similares se aplican a la minería procesos, la cual es una disciplina de investigación relativamente joven (IEEE Task Force on Process Mining 2011). Se entiende por minería de procesos a la disciplina de investigación que permite descubrir, monitorear y mejorar los procesos reales a través

¹Traducido de las siglas en inglés BPM: Business Process Management.

²XES: Flujo Extensible de Eventos (eXtensible Event Stream, XES por sus siglas en inglés).

de la extracción de conocimiento de los registros de eventos disponibles en los actuales sistemas de información (IEEE Task Force on Process Mining 2011).

La minería de procesos ha llamado la atención dentro de la Gestión de Procesos de Negocio ya que es una disciplina que desarrolla técnicas y herramientas que permiten analizar los procesos, extraer información a partir de los registros de eventos y presentar de forma explícita el conocimiento que contienen (AALST W. M. P., 2011). Además, permite el soporte operacional y brinda apoyo a la toma de decisiones, partiendo de los registros de eventos.

El enfoque de la Gestión de Procesos de Negocio se ha utilizado para analizar los procesos de negocio en el área de la gestión hotelera. La gestión de procesos en las entidades hoteleras cubanas se basa en los procesos fundamentales que existen en estas organizaciones (alojamiento, animación, alimentos y bebidas y servicios técnicos) (Pérez Almaguer Y., Otros, 2012).

El tiempo utilizado en cada uno de estos procesos limita la capacidad de satisfacer a los clientes, afecta el nivel de servicio percibido, puede afectar la eficiencia del uso óptimo de los recursos de acuerdo con la demanda máxima de los clientes y la capacidad máxima disponible de recursos (Pérez Pravia, 2010). El sector del turismo en Cuba no está ajeno a esta realidad, ya que para lograr su desarrollo se hace necesario diversificar su oferta, mejorar la orientación al cliente, los resultados de las instalaciones hoteleras y los procesos de negocio en las mismas.

En las instituciones hoteleras donde se ejecutan estos procesos se utiliza como sistema de información el ZUN Suite. Este software de gestión está basado en estándares internacionales de la industria hotelera, integrado por módulos adaptables a las particularidades de cualquier hotel, independientemente de la cantidad de habitaciones que posea o la complejidad de sus operaciones.

Aunque este sistema permite la informatización de los procesos, no brinda información para realizar análisis de los procesos que soporta. No contiene ninguna herramienta que facilite un análisis de estos procesos, por lo que ocasiona que no se posea una visión clara de cómo se ejecutan realmente los procesos de negocio en estas instituciones. En su lugar hay anécdotas, intuición y muchas opiniones subjetivas que deben reconciliarse, lo que proporciona falta de comunicación entre las partes (clientes y empleados). El descubrimiento manual del proceso a través de talleres y entrevistas es costoso en tiempo y dinero, además de incompleto y subjetivo. Existen quejas de los clientes derivadas de

ineficiencias y tiempos de ejecución demasiado largos en los procesos de gestión hotelera, lo que trae consigo la poca satisfacción y el descenso de clientes.

A partir de la problemática antes descrita se define el siguiente **problema a resolver**:

¿Cómo disminuir el tiempo requerido para el análisis de los procesos de negocio en entidades hoteleras donde se utiliza el sistema de información ZUN Suite?

A partir del problema a resolver planteado, se define como **objeto de estudio** la minería de procesos.

Para dar solución al problema a resolver se plantea el siguiente **objetivo general**: Desarrollar un componente que permita la extracción de registros de eventos en formato XES para la aplicación de técnicas de minería de procesos, que contribuya a disminuir el tiempo requerido para el análisis de procesos de negocio, en entidades hoteleras donde se utiliza el sistema ZUN Suite.

Por tanto el **campo de acción** se basa en la extracción de registros de eventos en formato XES.

Del objetivo general antes expuesto se desglosan los siguientes **objetivos específicos**:

1. Definir los datos necesarios para aplicar técnicas de minería de procesos sobre los procesos de gestión hotelera, a partir de las características del negocio y la definición del estándar XES.
2. Diseñar un algoritmo para la extracción de los datos asociados a los procesos de gestión hotelera registrados en el sistema ZUN Suite y su transformación hacia el formato XES.
3. Desarrollar un componente para el marco de trabajo de minería de procesos ProM que implemente el algoritmo diseñado.
4. Aplicar técnicas de minería de procesos a registros de eventos extraídos con el componente desarrollado para validar el cumplimiento del objetivo de la investigación.

Posibles resultados:

- Descripción de los procesos de gestión hotelera cuyas ejecuciones son registrados en el sistema ZUN Suite.
- Un componente para el marco de trabajo de minería de procesos ProM que permita la extracción de registros de eventos en formato XES del sistema ZUN Suite.
- Análisis de procesos de gestión hotelera utilizando minería de procesos.

Para dar solución a los objetivos planteados se establecen las siguientes **tareas de la investigación**:

- Análisis de los principales conceptos y trabajos relacionados con la definición de XES, sus extensiones y análisis de minería de procesos.
- Descripción de los procesos de gestión hotelera cuyas ejecuciones son registrados en el sistema ZUN Suite.
- Diseño de un algoritmo para la extracción de los datos asociados a los procesos de gestión hotelera cuyas ejecuciones son registrados en el sistema ZUN Suite.
- Diseño de un algoritmo para la transformación de los datos extraídos hacia trazas que cumplan con el formato XES.
- Implementación del algoritmo diseñado permitiendo su integración en el marco de trabajo de minería de procesos ProM.
- Realización de pruebas aceptación, propuestas por la metodología seleccionada.
- Extracción de registros de eventos con el componente desarrollado.
- Análisis de procesos a partir de los registros de eventos extraídos utilizando técnicas de minería de procesos.

Para el desarrollo del **diseño metodológico** de la investigación se utilizaron los siguientes **métodos científicos**:

Métodos Teóricos:

- Analítico-Sintético: permitió realizar el estudio teórico de la investigación, facilitando el análisis de documentos y la extracción de los elementos fundamentales acerca de la utilización de la minería de procesos en entornos reales.
- Histórico-Lógico: permitió realizar un estudio sobre las principales herramientas empleadas en el desarrollo de la minería de procesos, las tendencias del uso actual de las mismas en el mundo y en proyectos de la universidad, con el fin de seleccionar las más apropiadas para darle cumplimiento al objetivo general de la presente investigación.

- Hipotético-deductivo: se utiliza para guiar la investigación desde el mantenimiento del problema hasta la verificación de la solución a partir de las validaciones, orientando la secuencia lógica de las tareas que se realizan.

Métodos Empíricos:

- Caso de estudio: se utiliza para validar el enfoque y los objetivos propuestos en un entorno real.

En el presente documento puede encontrarse un resumen, una introducción, tres capítulos y conclusiones. A continuación se resume el contenido de los capítulos:

Capítulo 1, “Fundamentación teórica”, se introducirán los principales conceptos relacionados con la gestión de procesos de negocio, la minería de procesos, los conceptos asociados al objeto de estudio, se brinda una descripción del entorno en el cual se desarrolla la problemática y se analizan las posibles soluciones. Se realiza un estudio crítico de las técnicas de diagnóstico de proceso existentes, sus fortalezas y debilidades. Se presentan las principales características del marco de trabajo para la minería de procesos ProM.

Capítulo 2, “Análisis y diseño de la solución”, se describen los requisitos funcionales y no funcionales de la solución desarrollada. Se hace uso de la metodología XP para el desarrollo del software. Se explica la conformación del registro de eventos a través de las bases de datos Front e IHSecurity.

Capítulo 3, “Implementación y validación de la solución”, se describe la implementación del complemento, la cual se valida mediante los resultados obtenidos a partir de los análisis realizados. Se utiliza la técnica desarrollada en la extracción de registros de eventos provenientes de entornos reales. Además se valida la solución mediante la aplicación de pruebas de aceptación propuestas por la metodología utilizada.

Capítulo 1: Fundamentación teórica

1.1 Introducción

En el presente capítulo se describen los conceptos asociados a la minería de procesos con la gestión de procesos de negocio, la minería de procesos, la gestión hotelera y los conceptos asociados al objeto de estudio. Se brinda una descripción del entorno en el cual se desarrolla la problemática y se analizan las posibles soluciones. Se realiza un estudio crítico de las técnicas de minería de procesos, diagnóstico, conformidad y mejoramiento de los procesos existentes y los algoritmos asociados. Se presentan las principales características del marco de trabajo para la minería de procesos ProM.

1.2 Minería de procesos

El punto de partida de esta tecnología es BPM, que es la base para los Sistemas de Administración de Procesos de Negocios que generan la información sobre la que trabaja la minería de procesos. La información de estos sistemas se almacenan en registros de trazas y un correcto análisis de estos datos puede ayudar a mejorar la forma en que se ejecutan y gestionan los procesos en las organizaciones.

La Gestión de Procesos de Negocio permite, utilizando métodos, técnicas y software; diseñar, ejecutar, controlar y analizar procesos operacionales que involucran personas, organizaciones, aplicaciones, documentos y otras fuentes de información (van der Aalst, Ter Hofstede & Weske, 2003). También es conocida como una gestión integral que promueve la eficacia empresarial y la eficiencia mientras se esfuerza por la innovación, flexibilidad e integración con la tecnología (Gao, 2013). Además está creciendo como una disciplina, donde las nuevas tecnologías están emergiendo rápidamente, manteniendo el centro del escenario BPM en los dominios de negocio y tecnología.

Los sistemas de información registran enormes cantidades de eventos, de los cuales muchos son guardados de forma no estructurada, lo que ocasiona pérdida de tiempo y esfuerzo en la extracción de los mismos (AALST W. M. P., 2012). La minería de procesos está compuesta por tres tipos de técnicas, el descubrimiento de procesos, la verificación de conformidad y el mejoramiento de modelos, como muestra la Figura 1.1.



Figura 1.1: Representación de los tres tipos de técnicas de minería de procesos: descubrimiento, conformidad y mejoramiento (AALST W. M. P., 2011).

1.3 Técnicas de minería de procesos

Las técnicas de minería de procesos, permiten extraer información no trivial y útil de los registros de trazas almacenados por sistemas de información (Yzquierdo, 2012). El descubrimiento de proceso consiste en determinar, a partir de un registro de eventos, un modelo de proceso que explique el comportamiento presente en el mismo (AALST W. M. P., 2011). Las diferentes técnicas de minería de procesos permiten el descubrimiento de información a partir de los registros de eventos. De esta forma permiten conocer el flujo de ejecución de un proceso, las redes sociales y las métricas de desempeño (Aalst 2012). También permiten identificar cuellos de botella, prever problemas, registrar violaciones de políticas, recomendar contra medidas y simplificar procesos, además de disminuir los tiempos de diseño y con ello sus costos (Herrera, Castro, Cortés, Graña 2012).

El proceso de descubrimiento es una de las tareas más complejas y atendidas en la minería de procesos (van der Aalst 2012). Su objetivo es la construcción de un modelo de proceso a partir de la información obtenida de un registro de eventos (Aalst 2011). El descubrimiento de procesos es la técnica de minería de procesos más destacada (AALST W.M.P., 2011). Las técnicas existentes son capaces de descubrir los procesos reales basándose en las muestras de ejecución de los registros de eventos. Los modelos descubiertos pueden ser representados utilizando múltiples notaciones, dentro de las que se encuentran redes de petri, redes de flujo de trabajo, redes causales, redes difusas, entre otras (AALST W. M. P., 2013). El descubrimiento de un modelo de proceso requiere que el registro de eventos contenga información suficiente, es decir, posea un nivel de completitud tal que sus trazas

sean representativas del comportamiento del proceso. El grado y tipo de completitud necesario para realizar el proceso de descubrimiento varía de un algoritmo a otro.

El segundo tipo de técnica de minería de procesos es la verificación de conformidad. Esta técnica consiste en la comparación de un modelo de proceso existente con un registro de eventos del mismo proceso. La verificación de conformidad puede utilizarse para comprobar si las restricciones del modelo que lo originó se cumplen en la ejecución real del proceso (IEEE Task Force on Process Mining 2011). Su objetivo es detectar las discrepancias y medir su gravedad.

El tercer tipo de técnica es el mejoramiento del modelo es utilizado para extender o mejorar un modelo de procesos a través de información almacenada en el registro de eventos. Esto propicia que el modelo refleje un mayor entendimiento del mismo, con el fin de ajustarlo o extenderlo de modo que refleje mejor lo que ocurre en la realidad de su ejecución. Por ejemplo, se pueden identificar cuellos de botella reproduciendo un registro de eventos en un modelo de proceso, mientras se examinan las marcas de tiempo (IEEE Task Force on Process Mining 2011).

La minería de procesos es una disciplina impulsora del desarrollo de técnicas y herramientas para el análisis de los procesos de negocio que forman parte de los registros de eventos. Estos análisis se centran en brindar una mejor comprensión de los procesos para mejorar su rendimiento (AALST W. M. P., 2013).

1.4 Registro de eventos

La mayoría de los sistemas de información modernos utilizan mecanismos para registrar la ejecución real de los procesos, debido a que estos poseen un registro de trazas. Las técnicas de minería de procesos asumen que es posible registrar eventos secuencialmente (van der Aalst et al., 2011), donde cada proceso está compuesto por casos que no son más que instancias del mismo (Figura 1.2). Cada caso se compone de eventos que representan pasos bien definidos dentro del proceso; los casos contienen atributos o propiedades, donde los más usuales son la actividad que representan, la fecha y el usuario.

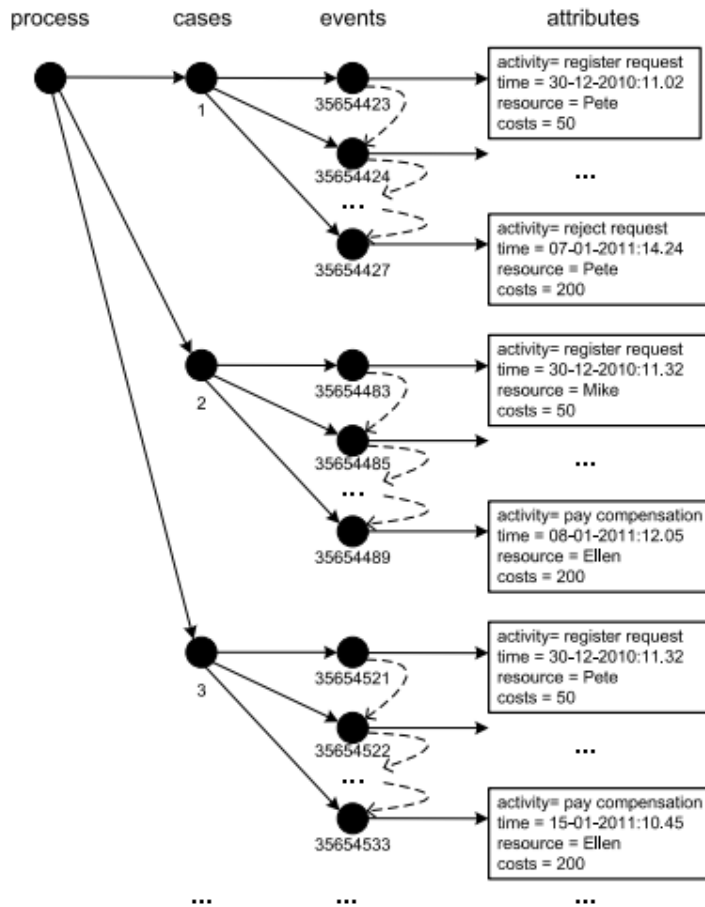


Figura 1.2: Resumen de un registro de eventos (van der Aalst, 2011).

Los casos y cada evento en un registro utilizan un identificador, mientras cada atributo de los eventos puede ayudar a extender el modelo con información extra. Dependiendo de la cantidad o tipo de información disponible será la perspectiva que se podrá asumir y la información que se podrá extraer. En la Figura 1.3 se expone un ejemplo de un registro de eventos donde se puede observar los identificadores y los atributos de los eventos.

Case id	Event id	Properties			
		Timestamp	Activity	Resource	Cost
1	35654423	30-12-2010:11.02	Register request	Pete	50
	35654424	31-12-2010:10.06	Examine thoroughly	Sue	400
	35654425	05-01-2011:15.12	Check ticket	Mike	100
	35654426	06-01-2011:11.18	Decide	Sara	200
	35654427	07-01-2011:14.24	Reject request	Pete	200
2	35654483	30-12-2010:11.32	Register request	Mike	50
	35654485	30-12-2010:12.12	Check ticket	Mike	100
	35654487	30-12-2010:14.16	Examine casually	Pete	400
	35654488	05-01-2011:11.22	Decide	Sara	200
	35654489	08-01-2011:12.05	Pay compensation	Ellen	200
3	35654521	30-12-2010:14.32	Register request	Pete	50
	35654522	30-12-2010:15.06	Examine casually	Mike	400
	35654524	30-12-2010:16.34	Check ticket	Ellen	100
	35654525	06-01-2011:09.18	Decide	Sara	200
	35654526	06-01-2011:12.18	Reinitiate request	Sara	200

Figura 1.3: Ejemplo de un registro de eventos (van der Aalst, 2011).

El punto de partida para aplicar cualquier tipo de técnica de minería de procesos es un registro de eventos (AALST W.M.P., 2011). Una instancia de un proceso, en el ámbito de la minería de procesos, hace referencia a la entidad siendo ejecutada por el proceso que es analizado. Los eventos se refieren a instancias del proceso. A su vez, la expresión del estado de una actividad para una instancia particular de un proceso se denomina evento, quien es el que contiene a todas las actividades que ocurren dentro de él y se ejecutan en un determinado tiempo, por ejemplo, el inicio, conclusión o cancelación de una actividad (IEEE Task Force on Process Mining 2011). El registro de eventos es la colección de eventos utilizados como entrada para aplicar las técnicas de minería de procesos (J.C.A.M. Buijs 2010).

Para asegurar un análisis de minería de procesos exitoso, además del formato de almacenamiento del registro de eventos se debe garantizar su calidad. La misma se define a partir de tres aspectos fundamentales: confiabilidad, completitud y seguridad. La confiabilidad consiste en que los eventos deben ser confiables, es decir, debería ser seguro asumir que los eventos registrados realmente ocurrieron y que los atributos de los eventos son correctos. La completitud se relaciona a que los registros de eventos deberían ser completos, dado un determinado contexto, no puede faltar ningún evento. Además, cualquier evento registrado debe tener una semántica bien definida, la cual se logra

asociando a cada evento del registro de eventos el nombre de la actividad a la que hace referencia en el proceso de negocio. Por otra parte los datos de eventos son seguros si se tienen en cuenta consideraciones de privacidad y seguridad al registrar los eventos (IEEE Task Force on Process Mining, 2011).

Niveles de madurez

La combinación de los tres aspectos de calidad se refleja en los niveles de madurez definidos por la IEEE para los registros de eventos (IEEE Task Force on Process Mining, 2011):

Nivel-1: Los eventos se registran manualmente, los eventos registrados podrían no corresponder a la realidad y podrían faltar eventos.

Nivel-2: Los eventos se registran automáticamente, no se sigue un enfoque sistemático para decidir qué eventos se registran, podrían faltar eventos o estos podrían no registrarse correctamente.

Nivel-3: Los eventos se registran automáticamente, pero no se sigue un enfoque sistemático para registrar los eventos. Hay un nivel de garantía de que los eventos registrados coinciden con la realidad (el registro de eventos es confiable pero no necesariamente completo).

Nivel-4: Los eventos se registran automáticamente y de manera sistemática y confiable, a diferencia del nivel 3 da soporte de manera explícita a nociones tales como instancia de proceso (caso) y actividad.

Nivel-5: El registro de eventos es de excelente calidad (confiable y completo) y los eventos están bien definidos. Los eventos se registran de manera automática, sistemática y segura. Se toman en cuenta adecuadamente consideraciones acerca de la privacidad y la seguridad.

Los eventos no necesitan ser almacenados en un archivo de registro por separado (por ejemplo, los eventos pueden estar dispersos en diferentes tablas de bases de datos). Los registros de eventos están definidos por diferentes formatos, uno de estos formatos es el XES.

1.5 Formato XES

Para formalizar la estructura de los registros de eventos a utilizar en la minería de procesos se han definido dos estándares: MXML y XES. MXML es un formato basado en XML para el intercambio de registros de eventos. MXML fue el primer estándar que surgió en el 2003 y fue adoptado por la herramienta de minería de procesos ProM. MXML establece una notación estándar para almacenar

fechas, recursos y tipos de transacciones. XES en el 2010 reemplaza a MXML como el nuevo formato para minería de procesos independiente de la herramienta (IEEE Task Force on Process Mining, 2011).

El formato XES es un estándar XML para los registros de eventos. El estándar ha sido adoptado por la IEEE Fuerza de Trabajo de la Minería de procesos, como el formato de intercambio de registros de eventos por defecto.

Una vez obtenido el registro de eventos en formato XES, se debería ser capaz de analizar este registro de todas las maneras posibles en el ámbito de la minería de procesos (J.C.A.M. BUIJS, 2010). Los datos requeridos por el formato XES son, la marca de tiempo, identificador, nombre de la actividad y recurso.

1.6 Marco de trabajo para la minería de procesos ProM

ProM es un marco de trabajo para el desarrollo de herramientas de minería de procesos en un ambiente estandarizado (YANG H., 2012). Está desarrollado en Java y se encuentra disponible bajo licencia GPL1.0. ProM está concebido para admitir la adición de complementos y de esta manera posibilitar el desarrollo de nuevos algoritmos y técnicas en el campo de la minería de procesos. Los complementos necesitan determinada cantidad de parámetros de entrada y producen uno o varios objetos de salida. Los parámetros de entrada pueden ser registros de eventos u objetos obtenidos a partir del procesamiento realizado por otros complementos. Mientras que los objetos de salida obtenidos pueden ser empleados como parámetros de entrada de otros complementos. Este marco de trabajo cuenta con más de 550 complementos, cada uno de los cuales posibilita realizar diferentes análisis (AALST W. M. P., 2013). Las herramientas desarrolladas en ProM han sido empleadas en el análisis de procesos provenientes de diferentes dominios entre los que se encuentran gubernamental, hospitalario y sistemas ERP (AALST W. M. P., 2011).

1.7 Algoritmos de descubrimiento

Se denomina algoritmo de descubrimiento a la función que mapea un registro de eventos hacia un modelo de proceso (AALSTW.M.P., 2011). Existen varios algoritmos desarrollados entre los que se pueden mencionar: Alpha (AALSTW.M.P., 2004), Heuristic Miner (WEIJTERS A.J.M.M., 2003), Transition System Miner o Fuzzy Miner (GÜNTHER C. W., 2007).

Según (Fundora-Ramírez Osiel 2013) se hizo un estudio de los principales algoritmos y según las características de los registro de eventos provenientes de entornos reales, los más cualificados para utilizarse son los que se muestran en la siguiente (Tabla 1.1):

Tabla 1.1: Impacto de las características de los registros de eventos en los algoritmos de descubrimiento (Fundora-Ramírez Osiel 2013).

Nombre	Ruido	Tareas ocultas	Tareas duplicadas	Alternativas no libres	Lazos
EnhancedWfMiner	+	+	+	+	+
AGNEsMiner	+	+	+	+	+
DT Genetic Miner	+	+	+	+	+
HeuristicsMiner	+	+	-	+	+
FuzzyMiner	+	-	-	-	+
FSM Miner/Petrify	+	+	+	+	+
GeneticMiner	+	+	-	+	+

Alpha Miner

Fue el primer algoritmo disponible para descubrir concurrencias. Aunque su uso es mayormente para describir o reconstruir un conjunto de secuencias de eventos, permite observar la existencia de lazos, partes paralelas, entre otras, dentro de la ejecución del registro de eventos, garantizando propiedades certeras (AALSTW.M.P., 2004).

Este algoritmo recibe como entrada un registro de eventos y genera como salida un modelo en la notación red de petri. Se centra en el control del flujo ignorando los recursos y otros datos como el identificador, el nombre de la actividad, la marca de tiempo, entre otras. Presta atención al orden de las actividades sin caer en casos particulares.

Heuristic Miner

El algoritmo Heuristic Miner extrae la perspectiva de control de flujo de un modelo de proceso. Puede lidiar con el ruido y las excepciones, además de permitirles a los usuarios enfocarse en los principales

descubrimiento, basados en la correlación entre las tareas y la importancia de una secuencia. La siguiente figura (Figura 1.6) muestra un ejemplo del algoritmo.

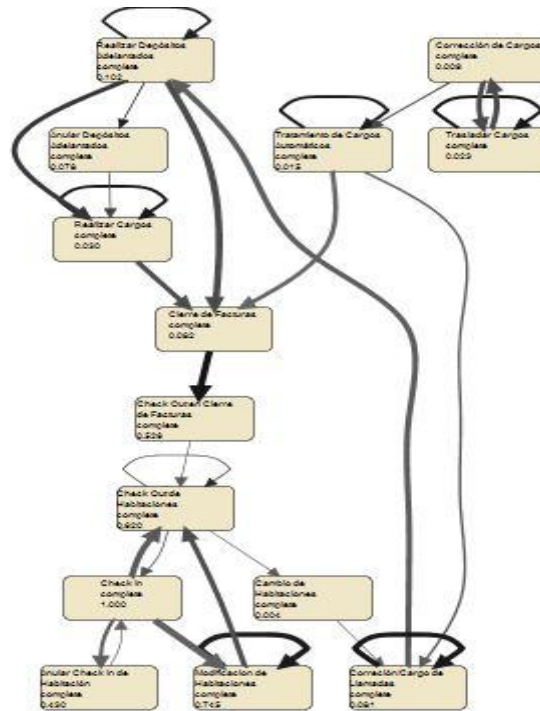


Figura 1.6: Algoritmo Fuzzy Miner.

1.8 Algoritmos de conformidad

Existen técnicas de minería de procesos que son capaces de determinar a partir de un modelo de proceso y un registro de eventos de dicho proceso, las diferencias entre el modelo concebido o esperado y la ejecución real de dicho proceso. A estas técnicas se les denomina verificación de conformidad.

ETConformance

El chequeo de conformidad resulta especialmente útil para comprobar el cumplimiento real de determinadas reglas definidas para un proceso determinado, como puede ser el principio de separación de responsabilidades. La conformidad puede ser aplicada a modelos procedurales, modelos organizacionales, modelos de procesos declarativos, políticas/reglas de negocio, regulaciones, entre otros. Permite por lo tanto detectar, localizar y explicar desviaciones, así como la severidad de dichas

desviaciones (Rozinat and Aalst, 2008). Es de especial utilidad en el descubrimiento de casos potenciales de fraude.

Esta técnica puede ser aplicada a la mayoría de los sistemas de información que generen registro de eventos, se utilizan principalmente para auditar y controlar procesos (Song & van der Aalst, 2007). Al analizarse adecuadamente un registro con esta técnica, se puede obtener un gran conocimiento lo que ayuda a las organizaciones a mejorar considerablemente la calidad de sus servicios.

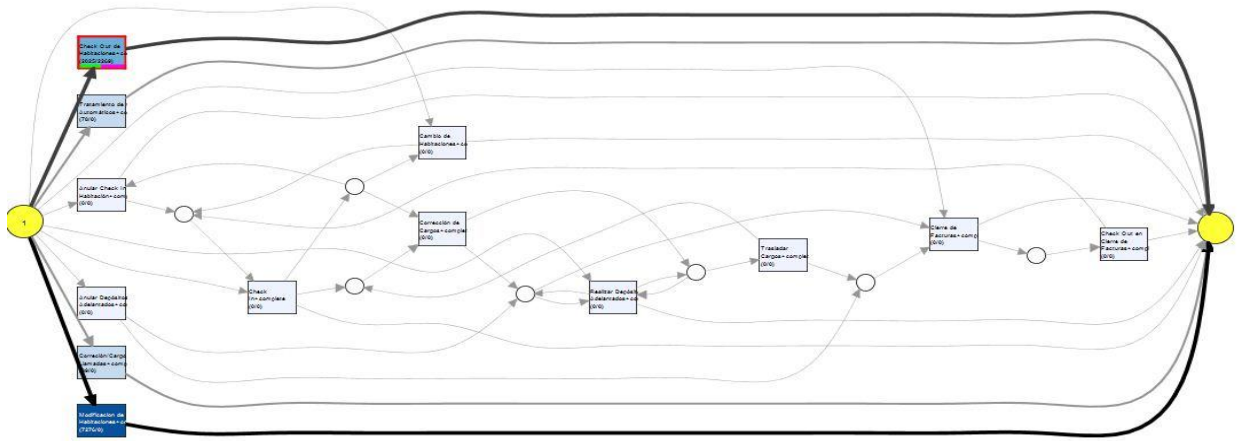


Figura 1.7: Algoritmo Chequeo de Conformidad.

Dotted Chart Analyzer

El Dotted Chart o Gráfico de Puntos es un gráfico similar a un diagrama de Gannt. Muestra una propagación de eventos de un registro de eventos en el tiempo. La idea básica de este algoritmo es trazar los puntos de acuerdo con el tiempo.

Un punto en el gráfico representa un único evento en el registro de eventos, la gráfica tiene dos dimensiones ortogonales: el tiempo y tipos de componentes. El tiempo se mide a lo largo del eje horizontal del gráfico. A lo largo del eje vertical, se muestran los tipos de componentes tales como, el originador, la tarea, tipo de evento o elementos de datos.

Sobre la base de los tipos de componentes, los eventos se reorganizan. Hay cinco tipos que se mencionaron anteriormente, estos tipos de componentes ayudan a los usuarios a centrarse en un aspecto en particular del registro de eventos. Si la instancia se utiliza como un tipo de componente, la

propagación de eventos dentro de cada instancia puede ser identificada. Luego los usuarios pueden identificar fácilmente qué instancia lleva más tiempo, la instancia que tenga muchos eventos.

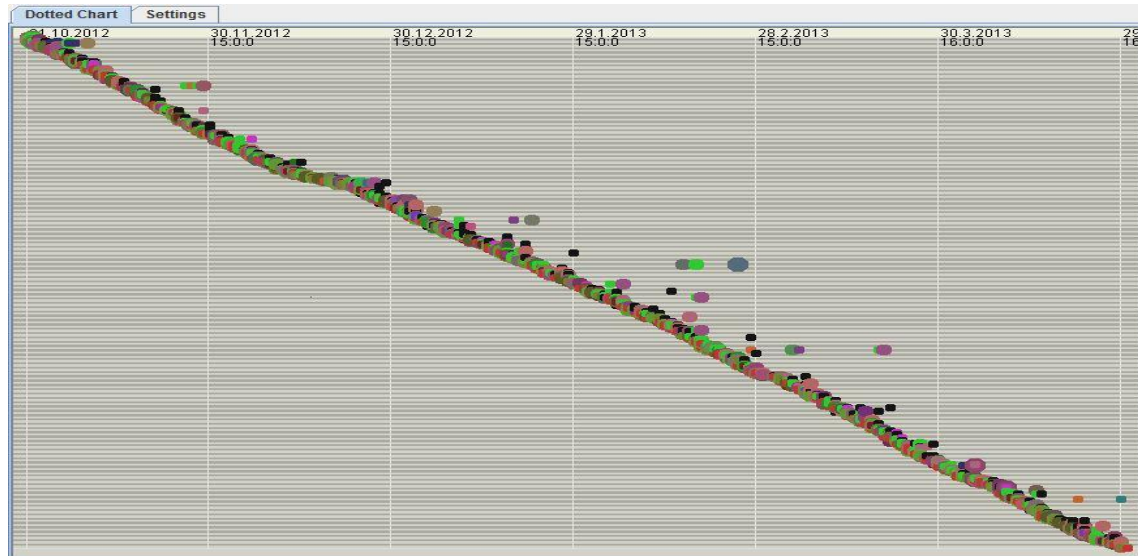


Figura 1.8: Algoritmo Dotted Chart Analyzer.

LTL Checker

Esta técnica permite al usuario comprobar el registro con un modelo LTL determinado y especificando una serie de ajustes previamente, se utiliza mayormente para realizar auditorías. Existen dos versiones, la primera el comprobador LTL, la cual requiere un modelo LTL como entrada; la segunda es el comprobador LTL predeterminado (LTL Checker Default), esta utiliza un modelo LTL predefinido y no requiere a ningún otro modelo LTL como entrada.

1.9 Gestión de Procesos

El ciclo de vida de BPM (Figura 1.10), se compone de fases que están relacionadas entre sí. Las fases se organizan en una estructura cíclica, mostrando sus dependencias lógicas. Estas dependencias no implican un estricto orden temporal en el que las fases tienen que ser ejecutadas (Weske 2007).

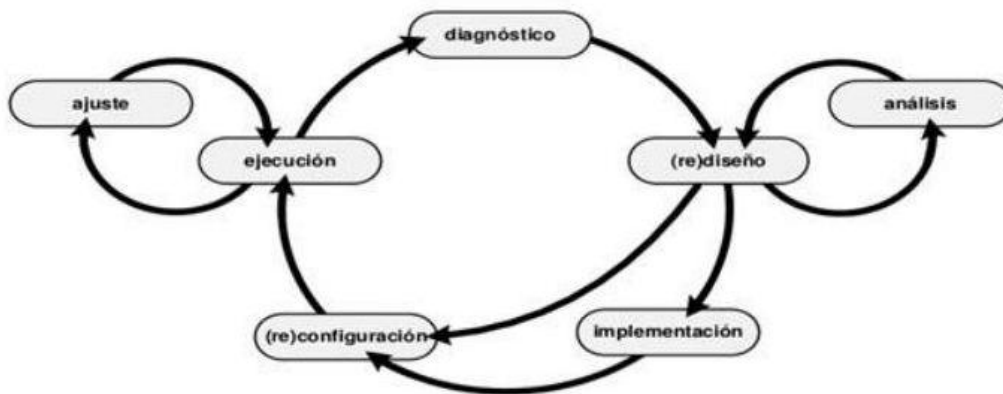


Figura 1.10: Ciclo de vida BPM. (Weske 2007).

El ciclo de vida BPM está compuesto por siete fases que se clasifican en Diseño y Análisis; Configuración; Ejecución y Ajuste; Evaluación. La Implementación y los involucrados están presentes en todas las fases. En la fase de Diseño y Análisis los procesos de negocio son identificados, revisados, validados y representados mediante modelos de procesos de negocio. Una vez que este ha sido diseñado y modificado, necesita ser implementado, la plataforma de implementación se elige durante la fase de Configuración. Esta configuración incluye las interacciones de los empleados con el sistema y la integración de los sistemas de software existentes. En la fase de Ejecución se ejecuta el modelo diseñado, el proceso es monitoreado durante esta fase. Además, se podrán realizar pequeños ajustes sin rediseñar el proceso (fase de Ajuste). En la fase de Evaluación se analiza el proceso ejecutado y la salida de esta fase podría generar una nueva fase de diseño del proceso (Weske 2007).

1.10 Gestión Hotelera

En la gestión hotelera constituye un reto desarrollar una gestión que asegure que los servicios sean percibidos por todos los clientes conforme a que los puedan satisfacer. Esto también es aplicable a las instituciones hoteleras en cuanto a asegurar el confort de sus clientes. El objetivo de toda gestión hotelera es, cumplir con la exigencia de la mayoría de los clientes, por lo que necesitan un modelo para poder llevar a cabo la persecución a este objetivo fundamental.

El sistema de información donde se gestionan todos los procesos en las instituciones hoteleras en Cuba, es el ZUN Suite.

1.11 ZUN Suite

Este software de gestión está compuesto por ocho módulos como muestra la Figura 1.11. A continuación se especifican cada uno de los módulos que componen el sistema ZUN Suite, enunciando las principales características y funcionalidades de los mismos.

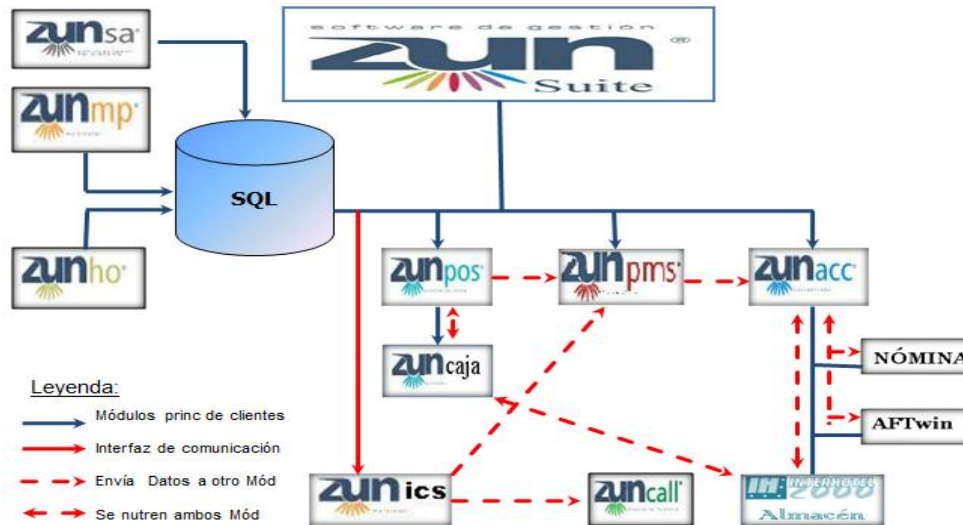


Figura 1.11: Módulos del sistema ZUN Suite.

System administration / Administración del Sistema

Es la herramienta que ofrece la suite ZUN al informático o administrador del sistema para gestionar la configuración de la entidad en cuanto a los módulos utilizados, los juegos de datos y bases de datos que los componen, sus enlaces y todo lo relacionado con la definición de las terminales, operadores y sus accesos.

Property Management System / Carpeta / Front Office

Sistema de Gestión Hotelera que a través de una comunicación sencilla y agradable con el operador:

- Posibilita el registro en línea de las características, clientes y operaciones del hotel, permitiendo así conocer en tiempo real la ocupación, la disponibilidad de alojamiento, previsiones y otros datos.
- Soporta el proceso de facturación.
- Guarda la historia de todas las operaciones que se producen en el hotel.

- Debido a su diseño, brinda seguridad, tanto desde el punto de vista del acceso de los operadores a las diferentes funciones, como de la integridad de la información.
- Ofrece información para la toma de decisiones.

En este módulo se centra el desarrollo de la investigación para el análisis de los procesos, ya que el resto de los módulos controlan fundamentalmente datos estáticos que no están orientados a un proceso. En este módulo existen un conjunto de pasos e interacciones con el cliente desde que llega al hotel hasta que se marcha del mismo, por lo que se puede seguir la trazabilidad del proceso.

Accounting / Contabilidad

Es un sistema de gestión empresarial que se encarga de todos los procesos para la organización eficiente de la empresa y genera la solución contable total para la entidad. Es un sistema multiempresa y multidivisa con plan de cuentas controladas según estructura, donde se definen todos los conceptos y la filosofía de trabajo en relación con determinados procesos:

- Creación de registros contables.
- Balance de comprobación.
- Estados de resultados.
- Traslado entre cuentas.
- Flujo de caja.
- Cuentas por edades.
- Costo por peso.
- Estados financieros.
- Otras operaciones contables.

Communication Interface / Interfaces de Comunicación

Este módulo permite enlazarse con cualquier tipo de central telefónica, cargo automático según habitación, extensión o código de acceso, realiza el cálculo del importe por pulso o tiempo según área geográfica, opción para la operadora, configuración de tarifas telefónicas, áreas geográficas, plantas, troncos, extensiones y códigos de acceso, cargos por los servicios prestados a los clientes a su cuenta en ZUNpms.

En la red extra hotelera se enlaza con ZUNcall (Sistema de telefonía), que permite registrar las llamadas telefónicas salientes en una Base de Datos reflejando el costo de las mismas.

Points of Sales / Puntos de Venta

Es un Sistema de Gestión para Puntos de Ventas Hoteleros y Extra hoteleros:

- Integrado y basado en modernas tecnologías en el campo de la informática y la hotelería.
- Mantiene una comunicación sencilla y agradable con el operador.
- Realiza las funciones de definición y control que requiere el módulo ZUNcaja.
- Guarda la historia de todas las operaciones que se producen en el Punto de Venta.
- Ofrece información para la toma de decisiones.

Head Office / Central de Cadenas

Módulo que permite la consolidación mensual de los datos contables de los hoteles, brindándose estados financieros a nivel central que reflejen el estado de toda la cadena. Posibilita que la cadena defina un plan de detalle propio a partir de las cuentas terminales de su estructura, donde se acumularán los movimientos enviados en las cuentas de detalle del hotel.

Multi Property / Multihotel

Este módulo permite una eficiente gestión de varios hoteles con una administración común, garantizando las operaciones propias de este tipo de estructura tales como: cambios de habitaciones, traslados de reservas y rooming list, cargos de teléfono y puntos de venta, entre los hoteles de la multipropiedad. Brinda posibilidad de consolidación de la información de cualquier combinación de los hoteles de la multipropiedad con las opciones: Listados de clientes, previsión de llegadas y salidas, disponibilidad de alojamiento y pronósticos, históricos de habitaciones, huéspedes, reservas y rooming list, así como listados estadísticos y composiciones estadísticas.

1.12 Tecnologías para el desarrollo de la solución

A continuación se describen las herramientas y tecnologías escogidas para el desarrollo de la solución que permitan la experimentación y el desarrollo del complemento.

ProM

Como se ha mencionado con anterioridad en el desarrollo de este capítulo, se utiliza la herramienta ProM en su versión 6.4 como marco de trabajo para la minería de procesos.

Netbeans

Netbeans en su versión 8.0, es un IDE (traducido al español Entorno de Desarrollo Integrado) libre y de código abierto, brinda las herramientas necesarias para el desarrollo de aplicaciones profesionales de escritorio, web y aplicaciones móviles con la plataforma Java, así como con PHP y otras (Oracle Corporation and its affiliates, 2012).

Visual Paradigm

Visual Paradigm en su versión 8.0, es una herramienta CASE (traducido al español Ingeniería de Software Asistida por Computadora) que utilizando UML (traducido al español Lenguaje de Modelado Unificado) como lenguaje de modelado, permite la captura, diseño, gestión y documentación de los artefactos generados durante el proceso de desarrollo de software (Visual Paradigm, 2011).

SQL Server 2005

Es un sistema RDBMS (Sistema Gestor de Bases de Datos Relacionales) basado en SQL Server 2000. Además de un servidor de datos, ofrece herramientas de análisis y gestión de almacenes de datos, servicios de generación de informes y notificación.

1.13 Metodología para el desarrollo de la solución

Según lo expuesto por (Abrahamsson, y otros, 2002), desarrollar un buen software depende de un elevado número de actividades y etapas, donde el impacto de elegir la mejor metodología para un equipo, en un determinado proyecto es trascendental para el éxito del producto. El desarrollo de software no es una tarea fácil, prueba de ello es que existen numerosas propuestas metodológicas tanto ágiles como tradicionales que inciden en distintas dimensiones del proceso de desarrollo.

Existen dos clasificaciones de metodologías que llevan a los equipos de desarrollo a obtener una mejor documentación de todo el proceso o a sacrificar esta por un proceso más ágil, llamadas Metodologías Ágiles y Metodologías Pesadas. Dada la necesidad de llevar a cabo el desarrollo de la propuesta de solución en un corto periodo de tiempo, manteniendo la colaboración constante con el cliente y la necesidad de garantizar la flexibilidad necesaria frente a la variación de los requisitos, se hace

necesario optar por un enfoque ágil de desarrollo de software en lugar de un enfoque pesado (Figueroa, y otros, 2009).

Según lo planteado por (Beck, 1999), (Jeffries, y otros, 2001) y (Abrahamsson, y otros, 2002) entre las principales metodologías ágiles se encuentran Programación Extrema (XP³), Iconix, SCRUM, Cristal Methods, Proceso Unificado Ágil (AUP⁴) y otras. Estas metodologías sobreponen la capacidad de respuesta a un cambio sobre el seguimiento estricto de un plan, dado el interés que supone para muchos clientes esta elasticidad.

eXtreme Programming

“Más que una metodología, XP se considera una disciplina, la cual está sostenida por valores y principios propios de las metodologías ágiles. Existen 4 valores que sirven como pilares en el desarrollo de estas metodologías” (Echeverry Tobón, y otros, 2007).

- La comunicación: en XP la relación con el cliente es tan estrecha, que es considerado parte del equipo de desarrollo.
- La simplicidad: no utiliza recursos para la realización de actividades complejas, solo se desarrolla lo que el cliente demanda, de la forma más sencilla.
- La retroalimentación: está presente desde el principio del proyecto, ayuda a encaminarlo y darle forma.
- El coraje: el equipo de desarrollo debe estar preparado para enfrentarse a los continuos cambios que se presentarán en el transcurso de la actividad.

Además de estos valores XP centra su desarrollo en 12 prácticas específicas que la caracterizan. Estas son (Echeverry Tobón, y otros, 2007):

- El juego de la planificación.
- Entregas pequeñas.
- Utilización de metáforas del sistema.
- Diseño simple.
- Pruebas.

³ Del inglés eXtreme Programming.

⁴ Del inglés: Agile Unified Process.

- Refactorización.
- Programación por parejas.
- Propiedad colectiva del código.
- Integración continua del código.
- 40 horas por semana.
- Cliente in-situ.
- Estándares de programación.

1.14 Conclusiones parciales

El estudio de los objetivos de la minería de procesos y las técnicas que involucra, además de su aplicación en entornos reales se enfocan en la extracción de información a partir de los registros de eventos. Se considera que la aplicación de estas técnicas, contribuye a la solución del problema planteado en las instituciones donde el sistema de información que se utiliza es el ZUN Suite.

A partir de los análisis realizados sobre las técnicas de minería de procesos y el estándar XES, se identificó que es necesario recopilar para los registros de eventos los datos referentes al formato XES. Considerando las características e información relevante para la gestión hotelera, se decidió agregar al registro de eventos los siguientes campos: nacionalidad, tipo de habitación y habitación.

Considerando que ProM es el marco de trabajo para la minería de procesos y una herramienta de software libre que soporta la adición de nuevos componentes, es la más adecuada para la ejecución del complemento que dará solución al problema planteado.

Capítulo 2: Análisis y diseño de la solución

2.1 Introducción

En este capítulo se describe la propuesta de solución denominada Componente para la extracción de registros de eventos en formato XES del sistema ZUN Suite. Para un mejor entendimiento de la propuesta de solución, en este capítulo se realiza una descripción del proceso involucrado. Además se presentan los diagramas de modelado de procesos de negocio y el mapa conceptual de la base de datos.

2.2 Descripción del proceso

Se escogió el módulo de Front Office o Carpeta para el análisis de los procesos ya que es el único que maneja datos a través de los cuales se puede seguir la trazabilidad de los procesos. A continuación se muestra el diagrama BPMN del proceso mencionado para un mejor entendimiento del mismo, así como su estructura y el orden de las actividades (Figura 2.1):

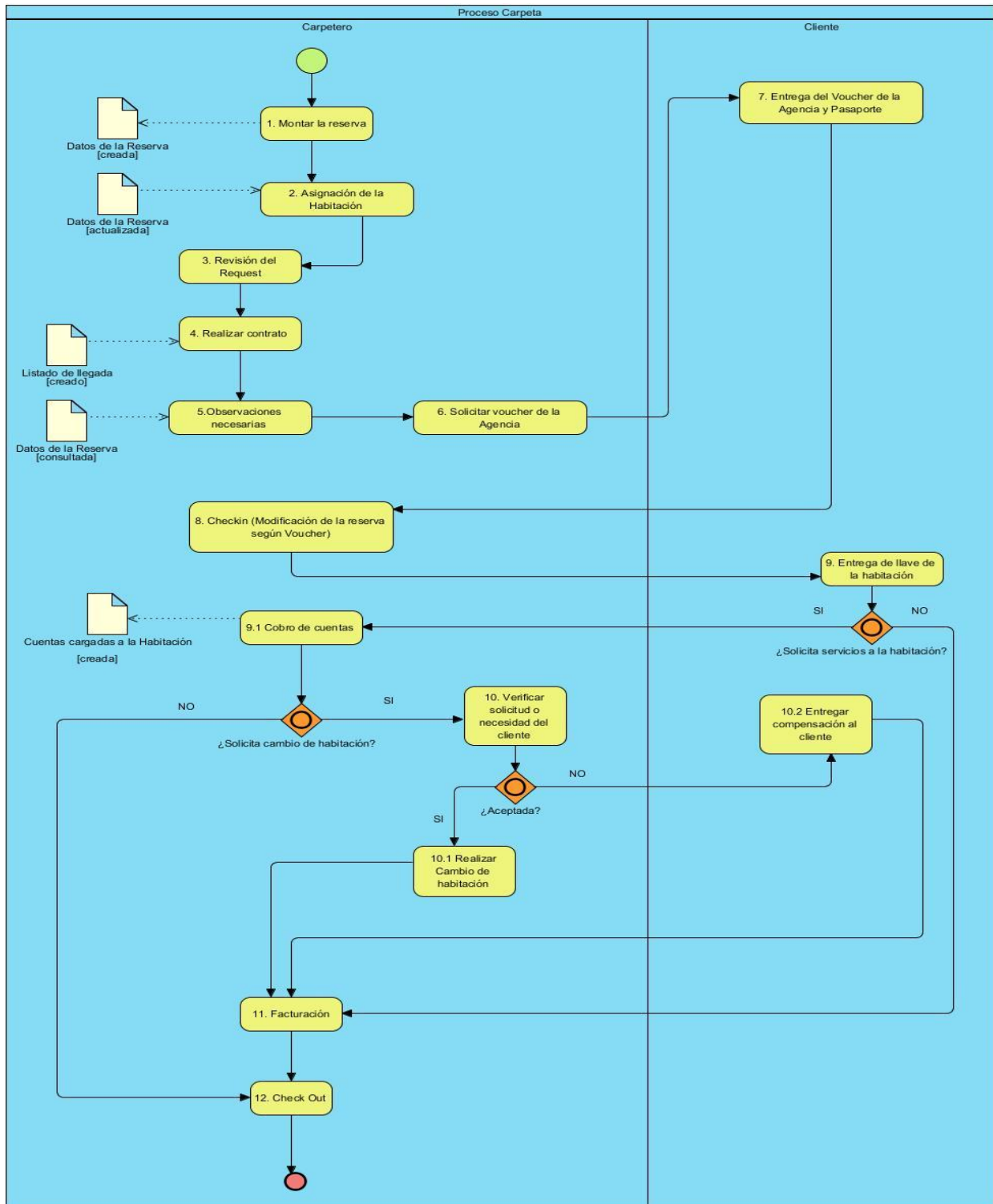


Figura 2.1: Diagrama BPMN del proceso Front Office o Carpeta.

Luego de un análisis previo de los campos referentes al registro eventos y a las base de datos referentes al proceso Carpeta, se identificaron como datos relevantes del proceso, la fecha de inicio y la fecha fin de llegada y salida de un cliente en el hotel, la nacionalidad de dicho cliente, la habitación, la observación y el id_face que se refiere al identificador de cada actividad.

2.3 Conceptos asociados a la investigación

Para posibilitar un mejor entendimiento del proceso es necesario formalizar un conjunto de definiciones relacionadas al mismo.

Concepto 1. *Check In:* El Check In es el inicio de la estancia del cliente. Todo sucede dentro de ese período, es decir, todas las actividades que realice el cliente dentro de su estancia en el hotel queda registrada en el sistema.

Concepto 2. *Check Out:* El Check Out se realiza cuando el cliente ha concluido su estancia en el hotel.

Concepto 3. *Modificación de Habitaciones:* La modificación de la habitación se realiza con la información correcta, es decir, el cliente debe presentar el voucher de la agencia y su pasaporte, para que con los datos reales del cliente se modifique todo lo que no esté, el nombre, categoría, cantidad de personas, entre otros datos.

Concepto 4. *Cierre de facturas:* Se realiza la actividad de cierre de factura cada vez que se haga un cargo por un concepto (mini bar, caja de seguridad, peluquería, masaje, entre otros) y el cliente efectúa el pago, ya sea en efectivo o con tarjeta, se cierra la factura y solo quedaría dar baja en la salida del hotel.

Concepto 5. *Realizar cargos:* Esta actividad se refiere a los cargos manuales, o sea, son todos aquellos pedidos que realiza el cliente, como pueden ser vinos a la carta, misceláneas (flores, toallas perdidas, roturas, entre otros), bodas, tarjeta de internet, entre otros.

Concepto 6. *Tratamientos de cargos automáticos:* Los cargos automáticos son los que se hacen desde el día de la llegada del cliente o al inicio del mismo. Dentro de los tratamientos de cargos automáticos se encuentran los cambios de sección, caja de seguridad, mini bar, entre otros. Se efectúa un arqueo de caja, para por si por error se para el cargo, se puede continuar, y se terminaría en la fecha de la salida. Cada cargo sale registrado en el cierre diario.

Concepto 7. Realizar depósitos adelantados: Esta actividad se refiere a si el cliente decide crear una cuenta con dinero en el hotel para a partir de ella ir liquidando sus cuentas.

Concepto 8. Cambio de habitaciones: La actividad de cambio de habitación está dada por solicitud o por necesidad del cliente. También está asociado a procesos de quejas y en determinados momentos se realiza el cambio por compensación.

Concepto 9. Corrección/Cargos de llamadas: Se realiza cuando se hacen devoluciones de servicios solicitados por el cliente. Para ello se requiere de la autorización de la dirección y se pueden realizar en cash o tarjeta de crédito.

Concepto 10. Corrección de cargos: La actividad se realiza para ajustar errores del carpetero en caso de no activar los Cargos Automáticos o también para hacer devoluciones de servicios solicitados por el cliente.

Concepto 11. Trasladar cargos: Se realiza cuando las cuentas de un cliente están mal cargadas, o sea, se cargan a la habitación incorrecta, entonces se realiza el traslado de cargos a la habitación correcta.

Concepto 12. Anular Check In de habitaciones: Se realiza cuando existen errores en la reserva, como puede ser que se le realice el Check In a un rooming list y existan clientes que no pertenezcan a ese vuelo.

Concepto 13. Anular depósitos adelantados: Esta actividad se realiza al comparar los depósitos adelantados contra los pagos, si se realiza la anulación es porque habrá una devolución al cliente o por error del carpetero.

Concepto 14. Check Out en cierre de facturas: Se realiza automáticamente cuando se realiza el Check Out del cliente o de la habitación.

2.4 Reglas del negocio

Para posibilitar un mejor entendimiento del proceso es necesario formalizar un conjunto de reglas relacionadas al mismo.

Regla 1. Check Out: Existen otros casos en los que puede ocurrir un Check Out, por ejemplo cuando existe algún error del recepcionista a la hora de realizar algún cambio en el sistema, o también porque existen casos de clientes que viajan con reservaciones divididas en dos números de booking

(alojamiento), por lo que será necesario realizar un Check Out, luego un Check In y otro Check Out para el otro número de booking.

Regla 2. Modificación de Habitaciones: Cada vez que al cliente se le agregue un cambio en su Check In, ya sea porque se le adicionó otra persona a la habitación, toallas de playa, prórroga de estancia en el hotel, entre otras cosas, se realiza una Modificación de Habitación.

Regla 3. Tratamiento de Cargos Automáticos: Luego de un Tratamiento de Cargos se pueden realizar Cambio de Habitación, ya que en este caso se trasladan todos los cargos a la nueva habitación. Para Anular Depósitos Adelantados luego de que se haya hecho un cargo tiene que ser que el cliente se marche del hotel ese mismo día y se anule para devolver el dinero sobrante.

Regla 4. Anular Check In de Habitación: No es posible realizar esta actividad si la actividad anterior no es un Check In / Modificación de Habitaciones.

Regla 5. Corrección de Cargos: Esta actividad solo puede ser ejecutada siempre y cuando exista una explicación clara de los motivos de la corrección aprobada por la dirección del hotel y del departamento.

Regla 6. Traslado de Cargos: Esta actividad solo puede ser ejecutada siempre y cuando exista un Cambio de Habitaciones previo al traslado de cargos, ya que este es el único motivo por el que la actividad puede ser ejecutada.

2.5 Propuesta de solución

Se propone el desarrollo de un componente que permita extraer de las bases de datos de los hoteles que utilizan el sistema de información ZUN Suite, datos para conformar un registro de eventos en formato XES para su posterior análisis. Este análisis estará centrado en aplicar técnicas de minería de procesos que permitan encontrar deficiencias o anomalías en los procesos de gestión hotelera, además de brindar posibles soluciones a los problemas encontrados y lograr disminuir los tiempos de ejecución de estos procesos.

La solución informática debe de ser capaz de extraer el registro de eventos, centralizar los errores, las posibles soluciones, filtrar el registro de eventos de acuerdo a diferentes parámetros como son temporadas altas o bajas, nacionalidad, fecha de inicio y fecha fin, entre otras, además brindar información específica relacionada con los procesos de gestión hotelera.

Para generar el registro de eventos con los datos referentes al proceso Front Office o Carpeta, se analizaron las bases de datos del sistema ZUN Suite. El Sistema ZUN Suite se encuentra estructurado por módulos y cada módulo tiene asociado una base de datos. La base de datos *IHSecurity* se compone por 38 tablas, en las cuales se almacenan las trazas del proceso (Figura 2.2). La base de datos *Front* es utilizada igualmente para la obtención de los datos que conforman el registro de eventos, la misma contiene datos asociados al proceso de Carpeta y está compuesta por 300 tablas (Figura 2.3).

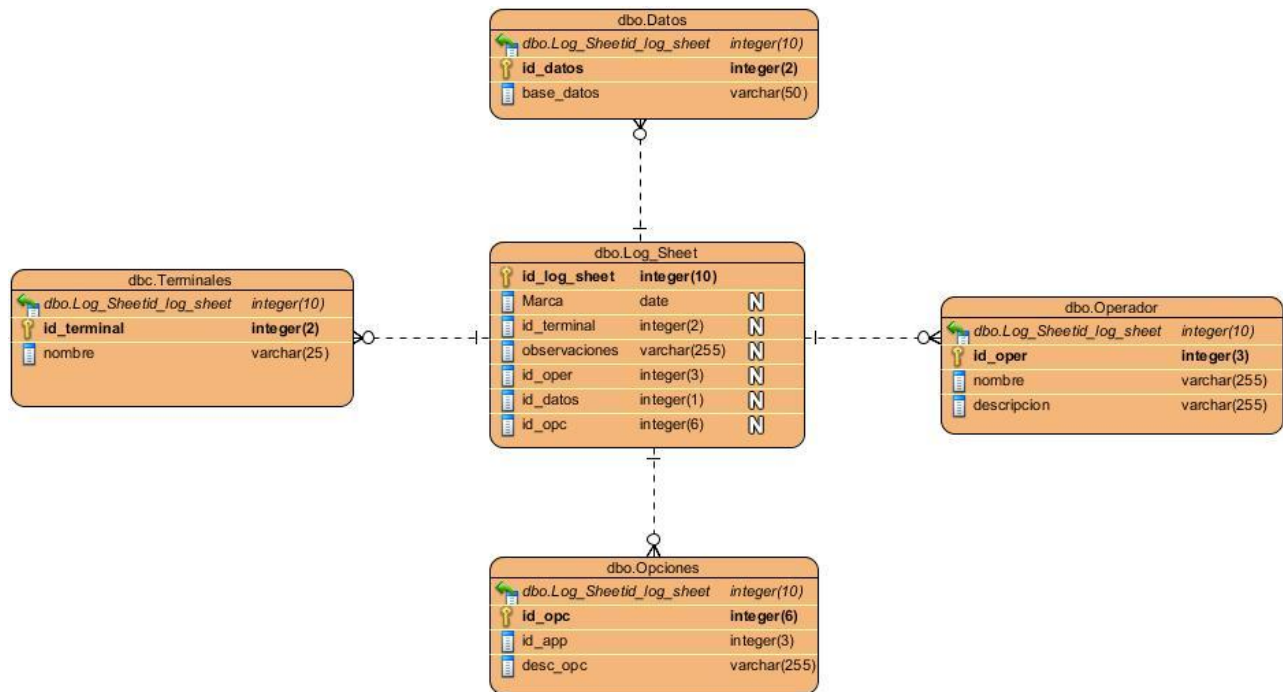


Figura 2.2: Modelo Entidad Relación de la base de datos IHSecurity.

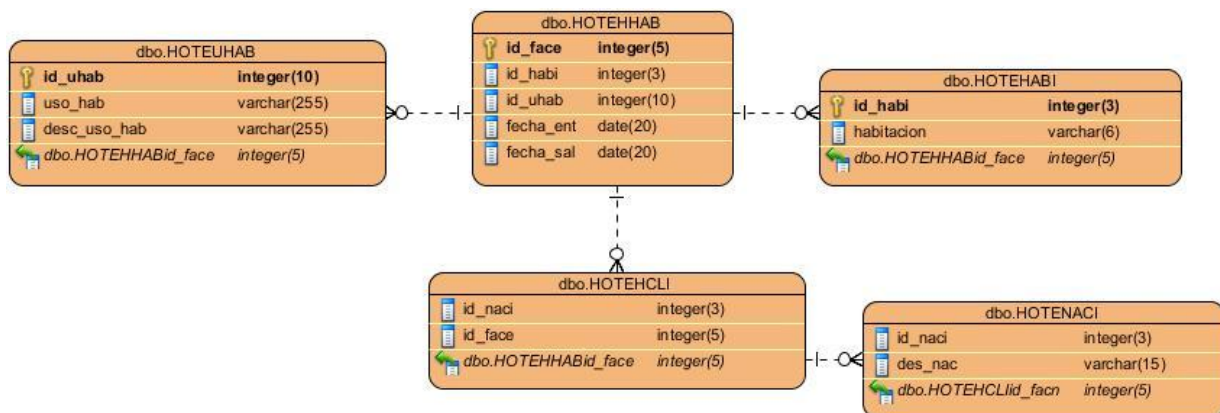


Figura 2.3: Modelo Entidad Relación de la base de datos Front.

En la base de datos *IHSecurity*, la tabla que almacena el registro de trazas del proceso es la tabla *dbo.Log_Sheet*. Las actividades a procesar se definieron en un análisis previo, quedando como actividades relevantes para conformar el registro de eventos las que tenían en la columna *id_opc* los siguientes valores (104031, 104104, 104032, 104020, 104021, 104108, 104102, 104101, 107004, 104100, 104103, 104105, 104027). Luego fue necesario establecer una relación con otros datos de importancia en la base de datos *Front*, para ello se identificaron aquellas tuplas de las seleccionadas anteriormente con valor 1 en la columna *id_datos*, esta columna es una llave foránea de la tabla *dbo.Datos* donde cada identificador representa un módulo del sistema.

Identificadas las actividades a procesar para la construcción del registro de eventos y establecidas las relaciones entre los datos, es posible la obtención de los mismos mediante consultas a la tabla *dbo.Log_Sheet* solicitando los valores correspondientes al identificador del evento, la marca de tiempo, observaciones, identificador de opción, identificador del operador, descripción de la opción y nombre del operador, todo esto a partir de relacionar las tablas mostradas en la Figura 2.2. Con el objetivo de facilitar los análisis posteriores se puede relacionar cada actividad con la habitación correspondiente, el valor de la habitación se obtiene del campo *observaciones* mediante transformaciones a la cadena de texto (parseo de los datos). Un ejemplo de estas transformaciones para las actividades *Cierre de Facturas* se presenta a continuación (Tabla 2.2):

Tabla 2.1: Ejemplo de transformaciones gramaticales.

Observación	Habitación
Hab. A2006 Chk.(2) 105985 CAJA 168.00	A2006

2.5.1 Diseño del algoritmo

A partir de los análisis previos para identificar y relacionar los datos de las bases de datos *IHSecurity* y *Front* se diseñó un algoritmo, el cual propone una serie de pasos a realizar para la obtención de los datos y la posterior extracción del registro de eventos. A continuación se definen una serie de parámetros empleados en el algoritmo:

Emb_N = Nombre de la base de datos embebida

Emb_P = Dirección de la base de datos embebida

FT = Tipo de filtrado

IHS = Conexión a la base de datos IHSecurity

EmbDb = Base de datos embebida

El algoritmo diseñado se divide en dos algoritmos principales para una mejor comprensión e interpretación de los pasos propuestos. El primer algoritmo denominado Filtrado de Datos, es el encargado de establecer la conexión al sistema ZUN Suite. Además realiza todo el procesamiento de los datos a partir de la relación entre las tablas y las bases de datos como se explicó anteriormente, este algoritmo culmina con la inserción de los datos en la base de datos embebida.

Algoritmo 1 Filtrado de Datos.

```
1: procedimiento realizarFiltrado(Emb_N, Emb_P, TF)
2:   IHS ← Establecer conexión a la base de datos IHSecurity.
3:   consulta ← Declarar consulta           > La consulta se realiza según TF.
4:   resultado ← ejecutarConsulta(consulta)
5:   mientras (resultado distinto de vacío) hacer > Se itera sobre las tuplas resultantes de
                                     la consulta.
6:     id ← obtenerIdEvento()
7:     marcaTiempo ← obtenerMarcaTiempo()
8:     observacion ← obtenerObservacion()
9:     id_opcion ← obtenerId_opcion()
10:    actividad ← obtenerActividad()
11:    recurso ← obtenerRecurso()
12:    habitación ← extraerHabitacion(observacion, id_opcion) > Se transforma el campo
                                                         observacion para obtener la habitacion
13:    nacionalidad ← obtenerNacionalidad(marcaTiempo, habitacion)
14:    id_traza ← obtenerId_Traza(marcaTiempo, habitacion)
15:    tipo_habitacion ← obtenerTipo_Habitacion(marcaTiempo, habitacion)
```


- 14: llaves ←obtenerLlavesDelMapa(mapa) > *Se guardan todas las llaves del mapa en una lista.*
- 15: **para cada** (llave en llaves)
- 16: traza ←crearTraza(llave) > *Se crea una traza con los valores asociados a esa llave en el mapa.*
- 17: xes ←escribirTraza(doc, traza) > *Se inserta la traza en el fichero xes*
- 18: **fin para cada**
- 19: retornar xes
- 20: **fin procedimiento**

2.6 Fase de planificación

La metodología XP define como fase inicial la planificación. “En esta fase, los clientes plantean a grandes rasgos las historias de usuario que son de interés para la primera entrega del producto. Al mismo tiempo el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto. Se prueba la tecnología y se exploran las posibilidades de la arquitectura del sistema” (Padanés, et al., 2007).

2.6.1 Historias de usuario

Las Historias de Usuario (HU) son la técnica utilizada en XP para especificar los requisitos del software. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el sistema debe poseer. El tratamiento de las historias de usuario es muy dinámico y flexible. En cualquier momento las HU pueden romperse, reemplazarse por otras más específicas o generales, añadirse nuevas o ser modificadas (Pressman, 2007). Cada HU es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en pocas semanas. Los requisitos funcionales del sistema se agruparon en cuatro HU de las que se muestran dos a continuación. El resto están contenidas en los anexos (ver **¡Error! No se encuentra el origen de la referencia.**).

Tabla 2.2: Historias de Usuario “Crear base de datos embebida”.

Historia de usuario	
Número: HU01	Nombre de la Historia de Usuario: Crear base de datos embebida

Modificación de la Historia de Usuario: Ninguna	
Usuario: Aliuska Batista Corella, Yordany Aguilera Martínez	Iteración asignada: 1
Prioridad en negocio: Media	Puntos estimados: 3 semanas
Riesgo en desarrollo: Media	Puntos reales: 3 semanas
Descripción: Permite crear una base de datos embebida para guardar los datos que se filtren para la extracción del registro de eventos. Además permite cargar una base de datos embebida existente.	
Observaciones:	

Tabla 2.3: Historias de Usuario “Filtrar datos del registro de eventos”.

Historia de usuario	
Número: HU03	Nombre de la Historia de Usuario: Filtrar datos del registro de eventos
Modificación de la Historia de Usuario: Ninguna	
Usuario: Aliuska Batista Corella, Yordany Aguilera Martínez	Iteración asignada: 2
Prioridad en negocio: Media	Puntos estimados: 3 semanas
Riesgo en desarrollo: Media	Puntos reales: 3 semanas
Descripción: Permite filtrar los datos del registro de eventos por: <ul style="list-style-type: none"> • Nacionalidad. • Temporadas (esta puede ser alta, baja o ambas). 	

<ul style="list-style-type: none"> • Ambas.
Observaciones:

2.6.2 Plan de iteraciones

Una vez finalizadas las HU se debe crear un plan de iteraciones, indicando cuáles se desarrollarán en cada iteración del programa. En la siguiente tabla (Tabla 2.4) se muestra cómo quedó definido el plan de iteraciones de la solución propuesta.

Tabla 2.4: Plan de duración de las iteraciones.

Iteraciones	Orden de las historias de usuario a implementar	Duración de las iteraciones (semanas)
Iteración 1	Crear base de datos embebida	3
Iteración 1	Configurar conexión al sistema ZUN Suite.	2
Iteración 2	Filtrar datos del registro de eventos	3
Iteración 2	Resumen de las configuraciones.	2
Total		10

2.7 Requisitos del sistema

De la palabra “requisito” existen numerosas definiciones, a continuación se presenta la que aparece en el glosario de la IEEE:

1. Una condición o necesidad de un usuario para resolver un problema o alcanzar un objetivo.
2. Una condición o capacidad que debe estar presente en un sistema o componentes del sistema para satisfacer un contrato, estándar, especificación u otro documento formal.
3. Una representación documentada de una condición o capacidad como en los puntos anteriores.

Según plantea (Pérez Huebe, 2005) existen dos tipos de requisitos, los funcionales y los no funcionales.

2.7.1 Requisitos funcionales del software

Los requisitos funcionales definen las funciones que el sistema será capaz de realizar. Describen las transformaciones que el sistema realiza sobre las entradas para producir salidas, especifican los servicios que debe proporcionar la aplicación atendiendo a lo expuesto por (Sommerville, 2005). En la Tabla (Tabla 2.5) se muestran los requisitos funcionales del sistema.

Tabla 2.5: Requisitos funcionales del software.

No.	Requisito funcional	No.	Requisito funcional
1.	Mostrar introducción.	8.	Filtrar datos del registro de eventos.
2.	Crear base de datos embebida.	9.	Filtrar por nacionalidad.
3.	Crear nueva base datos embebida.	10.	Filtrar por temporada.
4.	Cargar base de datos embebida existente.	11.	Filtrar por ambas.
5.	Configurar conexión al sistema ZUN Suite.	12.	Resumir configuraciones.
6.	Configurar conexión IHSecurity.	13.	Extraer registro de eventos.
7.	Configurar conexión Front.		

2.7.2 Métricas para la validación de requisitos

Estabilidad de los requisitos: El objetivo de esta métrica es medir la estabilidad de los requisitos y asegurar su adecuación antes de pasar al próximo flujo de trabajo. Se considera que los requisitos son estables cuando no existen adiciones o supresiones en ellos que impliquen modificaciones en las funcionalidades principales de la aplicación (Pressman, 2007). La estabilidad de los requisitos se calcula como: $ETR = [(RT - RM) / RT] * 100$.

ETR: valor de la estabilidad de los requisitos.

RT: total de requisitos definidos.

RM: número de requisitos modificados, que se obtienen como la sumatoria de los requisitos insertados, modificados y eliminados.

Esta métrica ofrece valores entre 0 y 100. El mejor valor de ETR es el más cercano a 100 porque mostrará que no se están realizando cambios sobre los requisitos, son estables y, por tanto, es confiable trabajar el análisis y diseño sobre ellos (Pressman, 2007).

Primera iteración: $ERT = [(13-0)/13] * 100 = 100$

Luego de realizada la primera iteración se determinó que los requisitos tenían un 100% de estabilidad en los requisitos funcionales del sistema.

2.7.3 Requisitos no funcionales del software

“Los requisitos no funcionales son restricciones de los servicios o funciones ofrecidos por el sistema. Incluyen restricciones de tiempo, sobre el proceso de desarrollo y estándares. Los requisitos no funcionales a menudo se aplican al sistema en su totalidad. Normalmente apenas se aplican a características o servicios individuales del sistema.” (Sommerville, 2005).

1. Usabilidad

RNU 1 El complemento debe presentar un acceso fácil y rápido, para facilitar su uso por usuarios con pocos conocimientos en el campo de la minería de procesos.

2. Diseño e implementación

RNDI 1 Se utilizará SQL Server 2005 como Sistema Gestor de Bases de Datos.

RNDI 2 Se utilizará Java como lenguaje de programación y ProM como marco de trabajo para la minería de procesos.

RNDI 3 Se utilizará Netbeans como IDE para el desarrollo del sistema.

RNDI 5 Se utilizará Visual Paradigm como herramienta de modelado.

RNDI 6 Se utilizará la base de datos objetual embebida Apache Derby.

3. Funcionamiento

Software:

RNFO 1 Para el funcionamiento en la PC cliente será necesario el marco de trabajo para la minería de procesos ProM en su versión 6.4.

Hardware:

RNFO 2 El componente a desarrollar deberá ejecutarse teniendo en cuenta las siguientes características mínimas:

- Procesador Intel Pentium Dual Core a 2.0 GHz, equivalente o superior.
- 2 GB de memoria RAM.
- Capacidad de 100 GB de disco duro.

4. Seguridad

RNS 1 El usuario y la contraseña de las conexiones a las bases de datos del hotel no será guardado, así como los parámetros de configuración.

5. Fiabilidad

RNF 1 Los datos serán almacenados en una base de datos embebida para evitar la modificación de los mismos sobre las bases de datos funcionales del sistema ZUN Suite.

6. Interfaz de usuario

RNIU 1 Las interfaces del módulo contendrán los datos de forma estructurada, permitiendo la correcta interpretación de la información.

RNIU 2 La entrada incorrecta de datos será mostrada al usuario claramente, detallando los campos donde se encuentra el error y mostrando como título el detalle del error.

RNIU 3 El diseño de la interfaz del sistema responderá a la ejecución de acciones de forma rápida, minimizando los pasos a dar en cada proceso.

7. Interconexión

RNI 1 La PC cliente se comunica con el servidor de base de datos mediante el puerto 1433.

RNI 2 Los datos requeridos para el análisis serán extraídos en el registro de eventos, partiendo de los datos existentes en la bases de datos del sistema ZUN Suite.

2.8 Conclusiones parciales

A partir del análisis previo del proceso y la descripción del mismo, así como las base de datos correspondientes al módulo ZUNpms, se identificó que los datos de los eventos asociados a las actividades del proceso, están ubicados en las tablas pertenecientes a las base de datos IHSecurity y Front las cuales deben ser combinadas para construir un registro de eventos.

El algoritmo diseñado permite la transformación de los datos identificados en la base de datos y su conversión hacia un registro de eventos. El complemento desarrollado utilizando la metodología XP, permite la ejecución del algoritmo en un entorno real a partir de la integración con las bases de datos del sistema ZUN Suite y la utilización de una base de datos embebidas.

Capítulo 3: Implementación y validación de la solución.

3.1 Introducción

En el presente capítulo se abordarán los aspectos fundamentales del proceso de desarrollo y pruebas al componente. Se describe el complemento que se integra a la plataforma ProM y sus principales funcionalidades. Se muestran sus vistas y los registros de eventos extraídos. Además se visualizan los resultados obtenidos a partir del análisis de los datos proporcionados por la base de datos del sistema ZUN Suite. Se valida la solución propuesta mediante la aplicación de las técnicas de minería de procesos aplicadas a los registros de eventos extraídos.

3.2 Fase de desarrollo

En esta fase de la metodología se codifican las especificaciones recogidas por las HU de manera tal que se cumplan los requisitos de todo el sistema. La metodología de desarrollo XP plantea que el mejor artefacto que se puede generar es el código, su buena calidad influirá en gran medida en el producto final. La calidad del código es influenciada por los estándares de codificación los cuales serán tomados en cuenta en esta fase.

3.2.1 Estándares de codificación

Los estándares de codificación garantizan la comunicación fluida y directa entre los programadores, permitiendo el aumento de la reutilización y el mantenimiento de los sistemas. La metodología de desarrollo XP resalta dicha comunicación, por lo que es necesario que sigan ciertos estándares de programación para lograr que cualquier persona del equipo de desarrollo pueda modificar parte del código. Además tratando de que sea un código entendible por otros programadores que posteriormente puedan apoyarse en este trabajo y poder desarrollar otras soluciones. Los estándares que se utilizan en la implementación son los siguientes.

Máxima longitud de las líneas:

- Todas las líneas se limitan a un máximo de 120 caracteres.

Comentarios:

- Se utilizan comentarios de línea para hacer más entendible el código.

Comentarios de línea: comentario pequeño que solo abarca una línea y describe el código que le sigue.

// Comentario de línea

/* Comentario de línea */

Estilo de los nombres:

- Clases e Interfaces: los nombres de las clases presentan la primera letra en mayúscula, en caso de ser un nombre compuesto, la inicial de cada palabra se representa en mayúscula. Se utilizan nombres simples y de alguna manera que describan el contenido, se usan palabras completas, a no ser que la abreviatura sea muy conocida.
- Métodos y variables: los nombres de los métodos se representan en minúscula, en caso de ser un nombre compuesto, la inicial de la primera palabra se simboliza en minúscula, y la de las otras palabras que lo componen en mayúscula. Los nombres de las variables son cortos pero con significados lógicos, capaces de permitir a un observador identificar su función.

3.2.2 Diagrama de Componentes

Un componente es una unidad modular que puede reemplazarse en su propio entorno. Un sistema de software completo se puede concebir como un componente. Un diagrama de componentes muestra los elementos de diseño de un sistema de software. Representa el diseño con independencia del lenguaje o plataforma que el diseño utiliza. Permite visualizar de manera sencilla la estructura general del sistema y el comportamiento de los servicios que proporcionan (Pantaleón, Marta E. Zorrilla, 2013).

El diagrama de componentes representa la estructura física del sistema, su agrupación por paquetes y muestra las dependencias entre estos como se muestra en la figura (Figura 3.1).

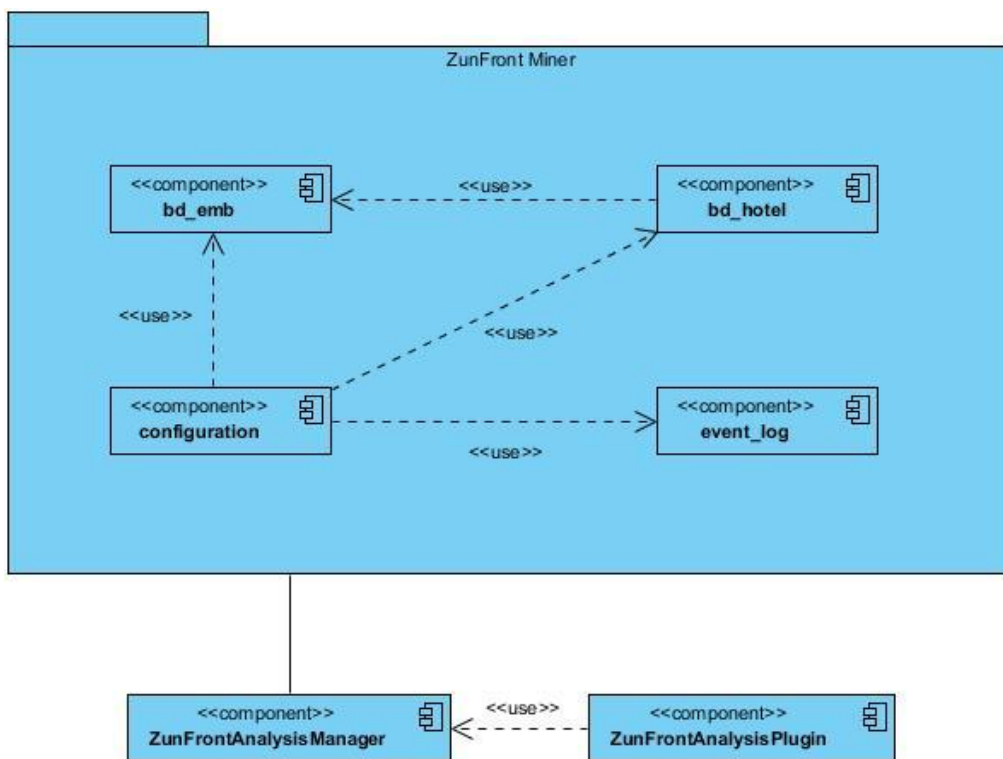


Figura 3.1: Diagrama de Componentes del sistema.

El complemento está conformado por seis componentes. El componente *bd_emb* es el encargado de gestionar el acceso a la base de datos embebida. El componente *bd_hotel* garantiza la conexión a las bases de datos del ZUN Suite. Por su parte el componente *configuration* se encarga de la visualización del componente implementado en la plataforma ProM. El componente *event_log* es el encargado de crear el registro de eventos. El *ZunFrontAnalysisManager* realiza la llamada al componente *configuration* y devuelve el registro de eventos. Por último el componente *ZunFrontAnalysisPlugin* es el encargado de establecer la configuración del componente para su integración al marco de trabajo ProM.

3.3 Fase de pruebas

El objetivo fundamental de las pruebas es determinar si el código generado funciona correctamente y realiza las operaciones deseadas. Además permiten comprobar que el producto cumple con los requisitos acordados con el cliente. Por último, verifican que el sistema sea capaz de funcionar normalmente en un ambiente similar o en el ambiente de despliegue (Pressman, 2007).

La metodología XP tiene entre sus artefactos las pruebas unitarias, las cuales permite confirmar que las HU han sido implementadas correctamente. Además están destinadas a evaluar si al final de una iteración se consiguió la funcionalidad requerida diseñada por el cliente final.

3.2.1 Pruebas de aceptación

“La validación del software se logra mediante una serie de pruebas que demuestren que se cumple con los requisitos (...). Al construir software personalizado para un cliente se aplica una serie de pruebas de aceptación que permiten al cliente validar todos los requisitos.” (Pressman, 2007). Las pruebas de aceptación en la metodología XP, también llamadas pruebas del cliente son especificadas por el mismo y se centran en las características y funcionalidades generales del sistema que son visibles y revisables por parte del cliente.

Los clientes son responsables de verificar que los resultados de estas pruebas sean correctos. Así mismo, en caso de que fallen varias pruebas, deben indicar el orden de prioridad de resolución. Una historia de usuario no se puede considerar terminada hasta tanto pase correctamente todas las pruebas de aceptación. Dado que la responsabilidad es grupal, es recomendable publicar los resultados de las pruebas de aceptación, de manera que todo el equipo esté al tanto de esta información (Joskowicz, 2008).

Tabla 3.1: Prueba de aceptación "Crear base de datos embebida".

Caso de prueba de aceptación	
Código: HU01_P1	Historia de usuario: HU01
Nombre: Crear base de datos embebida	
Descripción: Prueba para validar la funcionalidad de crear una base de datos embebida.	
Condiciones de ejecución: <ul style="list-style-type: none">• El usuario debe seleccionar el radio button de la opción de crear una nueva base de datos embebida o cargar una existente.• El usuario debe seleccionar la opción de crear la base de datos, indicando el nombre de la misma y la ubicación donde se guardará.• El sistema muestra el nombre de la nueva base de datos que ha sido creada y su ubicación, o sea, en que directorio se encuentra.	

<ul style="list-style-type: none"> • Si el usuario selecciona cargar una base de datos existente deberá ejecutar la opción de seleccionar la base de datos y su ubicación. • El sistema muestra el nombre de la nueva base de datos cargada y su ubicación. • Si el usuario escoge una carpeta que no se corresponde con una base de datos embebida, el sistema muestra un mensaje de error indicando que la carpeta escogida no es una base de datos. • El usuario puede seleccionar la opción de borrar la selección de la base de datos cargada con anterioridad. • El usuario debe seleccionar la opción Next.
<p>Resultados esperados: En caso de que se cumplan las condiciones de ejecución el sistema debe crear una nueva base de datos embebida o cargar una existente donde se guardarán los datos del registro de eventos a extraer.</p>
<p>Evaluación de la prueba: Prueba satisfactoria.</p>

Tabla 3.2: Prueba de aceptación: "Filtrar datos del registro de eventos".

Caso de prueba de aceptación	
Código: HU03_P1	Historia de usuario: HU03
Nombre: Filtrar datos del registro de eventos.	
Descripción: Prueba para validar la funcionalidad de filtrar los datos del registro de eventos.	
<p>Condiciones de ejecución:</p> <ul style="list-style-type: none"> • El usuario debe seleccionar las opciones de filtrado, por nacionalidad, por temporada o por ambas. • Si el usuario selecciona la opción de filtrar por nacionalidad, el sistema activa el panel de selección de los países. • El usuario debe escoger los países por los que desea filtrar. • El usuario debe seleccionar la opción Next. • Si el usuario selecciona la opción de filtrar por temporada, el sistema activa el panel de selección de los años y el combo box de las temporadas. • El usuario debe seleccionar los años por lo que desea filtrar y escoger el tipo de temporada, alta, baja o ambas. 	

- El usuario debe seleccionar la opción Next.
- Si el usuario selecciona la opción de filtrar por ambas, el sistema activa los paneles de selección de las nacionalidades y los países.
- El usuario debe seleccionar los países y los años por los que desea filtrar.
- El usuario debe seleccionar la opción Next.

Resultados esperados: En caso de que se cumplan las condiciones de ejecución el sistema debe guardar los parámetros de filtrado para extraer el registro de eventos.

Evaluación de la prueba: Prueba satisfactoria.

3.4 Descripción del análisis de los registros de eventos

El objetivo de este trabajo es disminuir el tiempo de los análisis de los procesos de la Gestión Hotelera, mediante la aplicación de técnicas de minería de proceso a los registros obtenidos a partir de las trazas de ejecución de dichos procesos. Por este motivo, se tiene en cuenta realizar el análisis con un enfoque basado en procesos, o sea, analizar el proceso en general, sobre el cumplimiento de las actividades representadas. Además, verificar cómo han sido ejecutadas las distintas instancias del proceso.

Como objeto de análisis se tiene al proceso Front Office o Carpeta, correspondiente al módulo ZUNpms del sistema ZUN Suite. El módulo ZUNpms se encarga de gestionar el flujo de información sobre la estancia de los clientes en el hotel y todas las operaciones que realiza en el mismo. Existen trece (13) actividades relacionadas con el proceso, como se mostró en el diagrama del proceso.

Para comenzar el análisis se deben extraer del sistema los datos correspondientes a los eventos, los modelos, los objetivos y las preguntas. Para ello, es necesario entender los datos almacenados para analizar cuáles pueden ser utilizados en el análisis y qué preguntas son de mayor importancia. En el capítulo 2 se explican las acciones que fueron necesarias para extraer los datos que conforman los registros de eventos.

Cuando se obtengan los registros de eventos a analizar, los cuales para una mayor organización se filtrarán por temporadas altas y bajas por cada año, para cada uno de ellos se crearán modelos de flujo de actividades y se les realizarán filtrados, aplicando las técnicas de minería de procesos descritas en el capítulo 1.

Obtenido el registro de eventos se procede a utilizar la herramienta ProM en su versión 6.4 para generar el modelo del proceso. Esta acción representa la primera etapa de la minería de procesos: el descubrimiento. En esta etapa se aplican los algoritmos de descubrimiento Heuristic Miner, Inductive Miner y Fuzzy Miner. Luego se aplicarán las técnicas de verificación de la conformidad como el Dotted Chart Analyzer y el LTL Checker.

Una vez obtenidos los modelos, se les puede aplicar otros procedimientos para convertirlos a una red de petri. En la red de petri aparecen actividades invisibles, las cuales se manifiestan cuando una o varias actividades de la instancia de proceso no se han registrado en las trazas. A esto se le denomina ausencia de información. Una actividad no queda reflejada en el registro de evento si:

- No fue informatizada.
- Sí fue informatizada, pero el sistema de información no deja constancia de su ocurrencia en el registro de eventos.
- Si el registro de evento fue afectado y presenta ruido.

Por último, se explicarán los resultados obtenidos en el análisis del registro de eventos y se expondrán las principales anomalías detectadas. Estos resultados se presentarán a un especialista en análisis de procesos de negocio con el fin de validar el objetivo de la investigación y contrastar el tiempo empleado en los análisis utilizando minería de procesos contra la utilización de técnicas tradicionales.

3.5 Extracción de los registros de eventos

Para la extracción del registro de eventos, fue necesario realizar un conjunto de consultas para obtener los datos que conformarían el registro de eventos. El mismo, está compuesto por nueve (9) campos, los cuales hacen referencia a las actividades del proceso Carpeta, que como se explicó anteriormente son trece (13). Los campos que componen las instancias de procesos son *id_log_sheet*, *marca*, *actividad*, *recurso*, *habitación*, *tipo_habitacion*, *id_face*, *nacionalidad* y *observacion*.

La naturaleza del registro de eventos en cuanto al rango de fechas de los mismos, se sitúa a partir del 1ro de mayo del 2012 al 31 de octubre del 2014, se tiene una totalidad de instancias de procesos de 53 838. Cuando se hace alusión a una instancia de proceso, se refiere a la estancia de un cliente en el hotel desde que llega hasta que se marcha del mismo en una habitación. Las instancias contienen un

total de 284 326 eventos, cada evento hace referencia a la ejecución de una actividad perteneciente a una instancia de proceso.

Para la extracción de los registros de eventos, fue necesario la creación de un componente sobre el marco de trabajo para la minería de procesos ProM, utilizando Java como lenguaje de programación, ya que el resto de los componentes que integran el ProM están desarrollados en este lenguaje. Como entorno de desarrollo se escogió el Netbeans, como herramienta Case para el modelado del proceso Visual Paradigm y como gestor de base de datos el SQL Server 2005 ya que sobre éste es donde se encuentran montadas las bases de datos del sistema ZUN Suite. El componente realizará las consultas pertinentes a las bases de datos IHSecurity y Front del sistema ZUN Suite. El mismo está compuesto por cinco (5) vistas en las que se realizan todos los pasos pertinentes para extraer el registro de eventos.

Para persistir los datos se utiliza una base de datos embebida en el complemento. La base de datos embebida permite almacenar la información que formará parte del registro de eventos a extraer. La misma se utiliza por cuestiones de integridad y seguridad de los datos almacenados en las bases de datos del sistema ZUN Suite, ya que a estas bases de datos no se le pueden hacer las transformaciones necesarias para conformar el registro de eventos. Para esto se utiliza Apache Derby, la cual es una librería que permite la creación de base de datos de forma embebida.

El componente permite en la siguiente vista (Figura 3.2), crear una base de datos embebida o cargar una existente para almacenar los datos que conformarán el registro de eventos. En esta pantalla se selecciona la ubicación y el nombre de dicha base de datos.

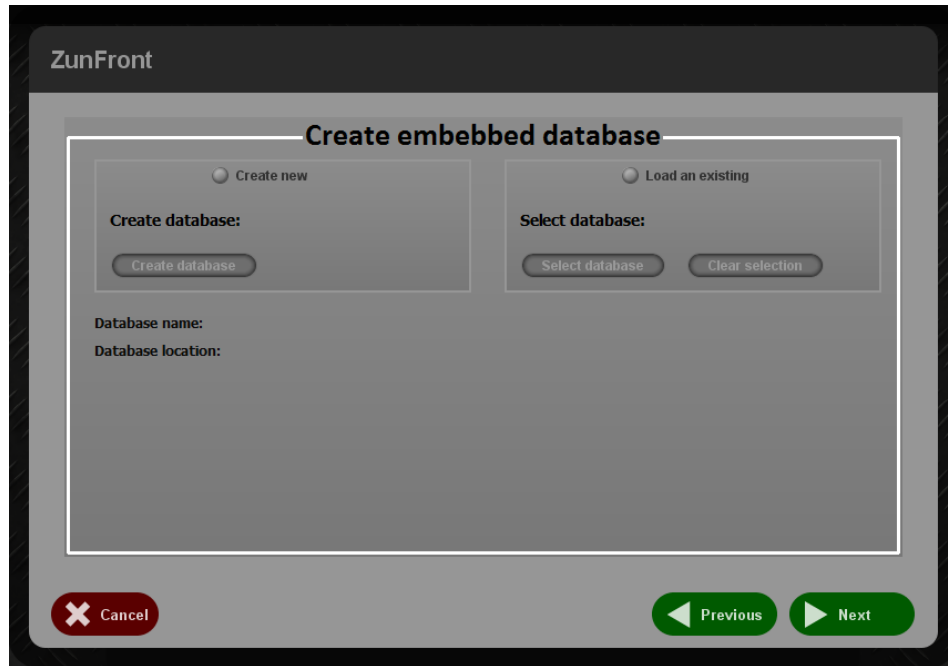


Figura 3.2: Crear base de datos embebida.

El componente permite realizar las configuraciones de conexión a las base de datos IHSecurity y Front, donde se almacenan los datos que conformarán el registro de eventos. La vista tiene dos tipos de conexiones, integrada y no integrada (Figura 3.3).

ZunFront

Configure ZUN Connection

IHSecurity configuration

Use integrated connection

IP address:
Ex:localhost

Username:
Session username

Password:
Session password

Test Connection

Front configuration

Use integrated connection

IP address:
Ex:localhost

Front Name:
Front database name

Username:
Session username

Password:
Session password

Test Connection

Cancel **Previous** **Next**

Figura 3.3: Configuración de las conexiones.

La siguiente vista del componente permite realizar los filtrados que corresponden a cómo y qué cantidad de instancias de procesos se almacenarán en la base de datos embebida para su posterior extracción. Los filtrados se pueden realizar por nacionalidad, temporadas o por ambas (Figura 3.4).

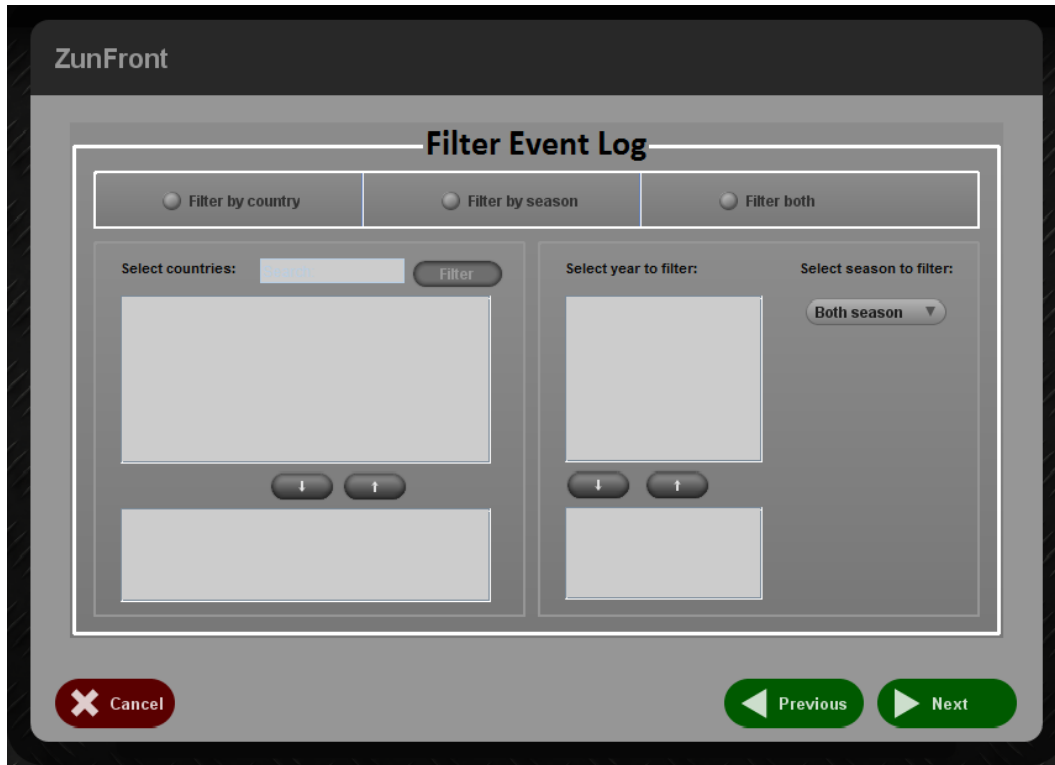


Figura 3.4: Filtrados del registro de eventos.

La última vista del componente permite visualizar un resumen de todas las configuraciones realizadas en las vistas anteriores. Además en esta vista se debe seleccionar la ubicación de donde se guardará el registro de evento una vez extraído (Figura 3.5).

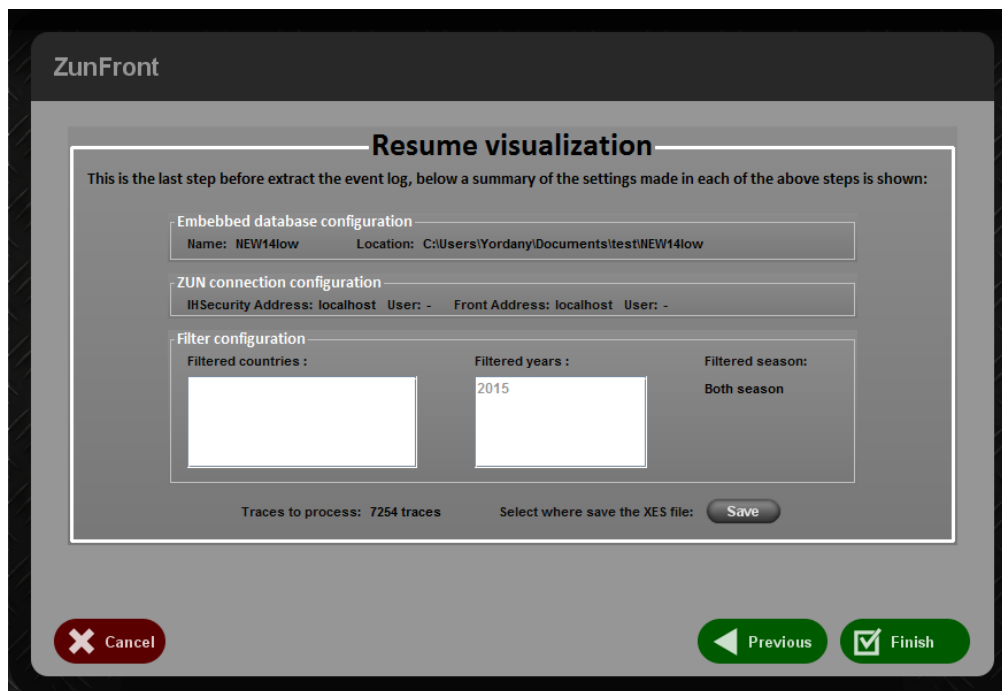


Figura 3.5: Visualización de todas las configuraciones.

3.6 Caso de estudio

Para validar el cumplimiento del objetivo planteado en la presente investigación, se realizó un caso de estudio basado en obtener registros de eventos con el componente desarrollado y aplicar a dichos registros de eventos técnicas de minería de procesos. Posteriormente se contrastarán los resultados obtenidos en el análisis además del tiempo empleado en la realización de los mismos contra las técnicas tradicionales de análisis de procesos de negocio.

Para verificar la utilidad de los registros de eventos extraídos utilizando el componente descrito con anterioridad en el análisis de procesos de negocio, se realiza a continuación el análisis de proceso Front Office o Carpeta aplicando las técnicas de minería de procesos que se describieron en el Capítulo 1. Primeramente se realiza el análisis por temporada, ya que los registros de eventos fueron extraídos de esta forma. Luego se eliminaron las instancias de procesos que tenía menos de tres eventos, ya que no aportaban ningún tipo de información relevante en el análisis. A continuación se muestran los mismos.

3.6.1 Análisis del registro de eventos correspondiente a la temporada baja 2012

El registro de eventos está compuesto por 50 809 eventos, agrupadas en 9 276 instancias de procesos de 12 actividades. El total de instancias de procesos que comienzan con Check In / Modificación de Habitaciones y terminan con Check Out de Habitaciones son 6 508 (70,15%) y el recurso que mayor participación tiene en este conjunto de instancias es Odaisis con 2 743 (14,55%). No cumplen con la ejecución del proceso 2 168 instancias (29,84%). De estas 856 (30,92%) comienzan con Check In / Modificación de Habitaciones, 1 664 (60,11%) terminan con Check Out de Habitaciones y 248 (8,95%) comienzan y terminan con otras actividades.

La actividad Corrección de cargos tiene un total de 129 (1,39%) instancias, de estas, en 69 (53,48%) instancias se realizan correcciones repetidas, lo cual representa una anomalía, ya que las Correcciones de Cargos se realizan por error del carpetero o por cargos hechos a habitaciones que no corresponden.

La actividad Anular Check In de Habitación tiene un total de 26 (0,28%) instancias de procesos, donde en 2 (7,69%) de ellas se realiza el Anular Check In de Habitaciones sin que antes existiera la actividad Check In / Modificación de Habitaciones. En un caso (3,84%) se ejecutan 2 Anular Depósitos Adelantados Seguidos, lo que significa que se tuvo que realizar una devolución a un cliente. Se muestra un ejemplo en la siguiente figura (Figura 3.6).

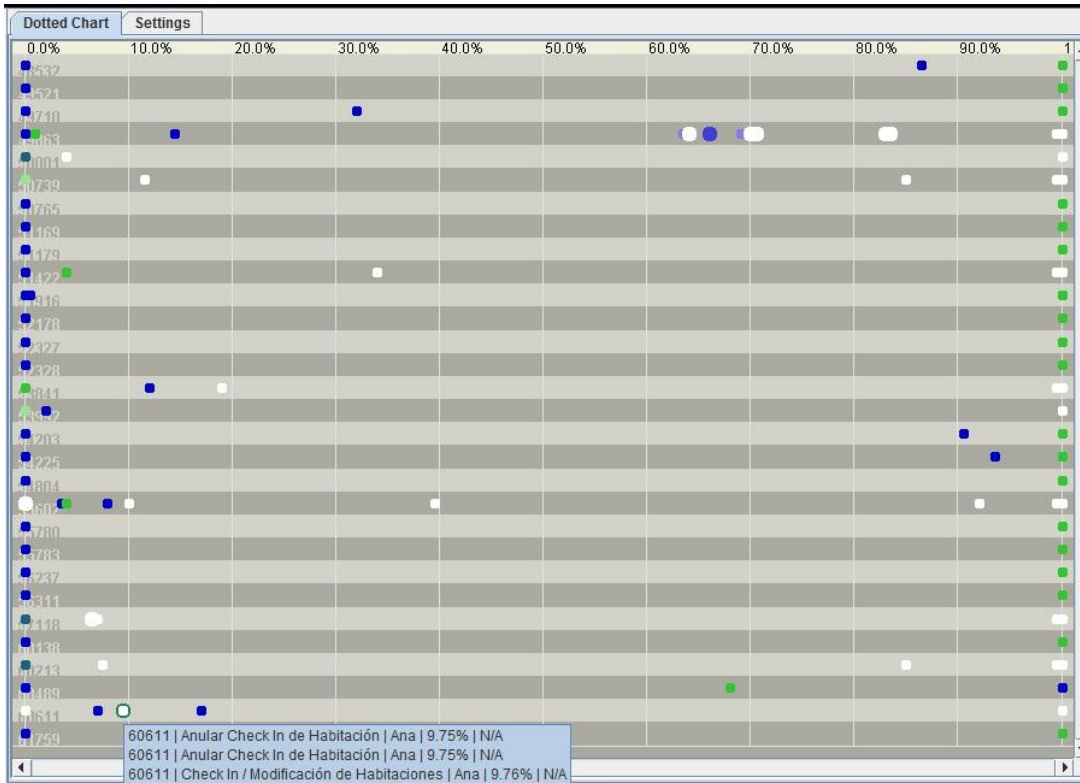


Figura 3.6: Anular Check In de Habitación repetido.

La actividad Cierre de Facturas es un de las más importantes dentro del proceso, ya que su función es cerrar todas las cuentas que se le han abierto al cliente (cargos). Esta actividad tiene un total de 4 476 (48,25%) instancias de procesos. A la actividad Realizar Cargos se les hace Cierre de Facturas a 1 891(42,24) instancias de 2 037 (21,95%). A la actividad Tratamiento de Cargos Automáticos se les hace Cierre de Facturas a 1 282 (28,64%) instancias de 3 115 (33,58%). Por último a la actividad Corrección/Cargo de Llamadas se les hace Cierre de Facturas a 117 (2,61%) instancias de 304 (3,27%). Se realizan 1 186 (26,49%) instancias correspondientes a Cierre de Facturas sin haber ningún cargo. De lo explicado anteriormente se muestra un ejemplo visual (Figura 3.7).

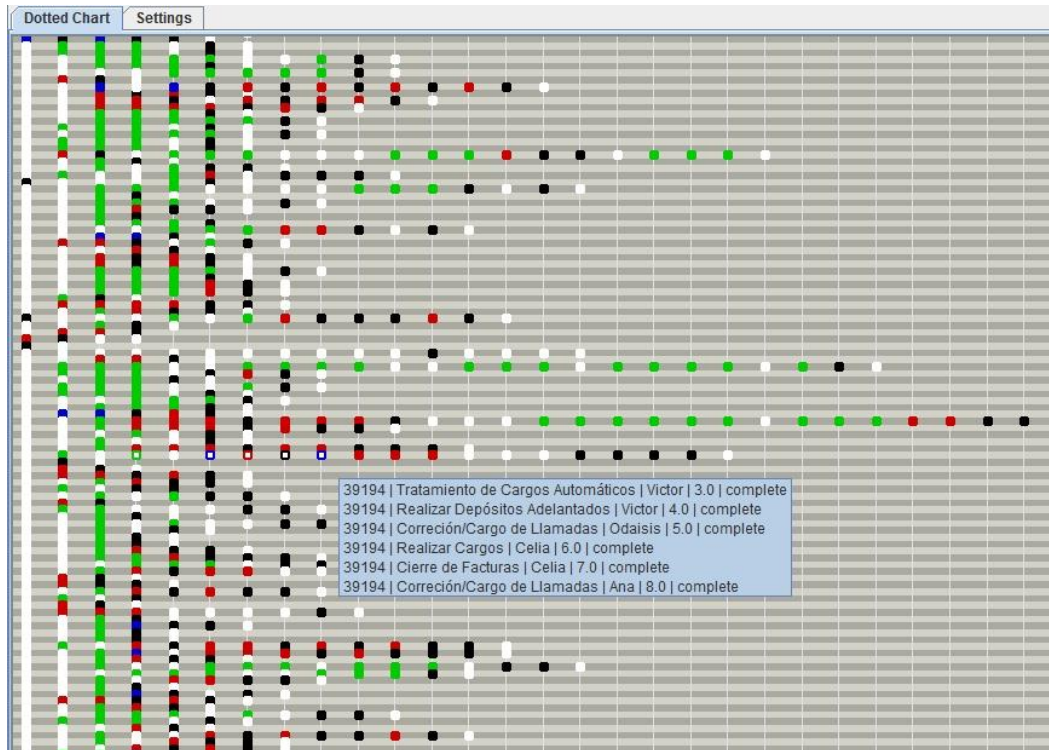


Figura 3.7: Cierre de Facturas a los cargos.

La actividad Realizar Depósitos Adelantados tiene un total de instancias de 2 972 (32,03%), en 100 (3,36%) instancias se realizan Depósitos Adelantados repetidos. La actividad Trasladar Cargos se debe realizar cuando ocurre un Cambio de Habitaciones, porque los cargos que tenía la antigua habitación son trasladados a la nueva. Esta actividad tiene un total de 220 (2,37%) instancias, de las cuales a 136 (61,86%) se les hace traslado sin haber Cambio de Habitaciones.

3.6.2 Análisis del registro de eventos correspondiente a la temporada alta 2012

El registro de eventos está compuesto por 13 847 instancias de procesos, agrupadas en 75 741 eventos de 12 actividades. La actividad Cambio de Habitaciones tiene un total de 3 076 (22,21%) instancias de procesos, el recurso Mayelin es el que mayor cantidad de ocasiones realiza esta actividad con un total de 317 (9,92%) eventos. A los turistas con nacionalidad canadiense es a los que más se le realiza la actividad Cambio de Habitaciones, con un total de 2 545 (82,73%) instancias, siendo también el recurso Mayelin el que más ejecuta esta actividad con un total de 266 (10,07%) eventos. Las habitaciones de tipo Tropical donde se alojan clientes con nacionalidad canadiense son a las que más Cambio de

Habitación se les realiza con un total de 1 593 eventos. Los turistas con nacionalidad cubana secundan a los de nacionalidad canadienses en cuanto a la cantidad de Cambio de Habitaciones realizadas con un total de 165 (5,36%) instancias siendo el recurso MariaE con 22 (13,01%) eventos el que más ejecuta esta actividad. De igual forma las habitaciones de tipo Tropical donde se alojan turistas con nacionalidad cubana son a las que más Cambio de Habitación se les realiza con un total de 113 (66,86%) eventos.

La actividad Corrección de Cargos tiene un total de 241 (1,74%) instancias de las cuales el recurso Marelis ejecuta la misma en 205 (27,18%) eventos, en 132 instancias (54,77%) se realizan correcciones repetidas lo cual es una muestra de que existen varios errores que conllevan a realizar estas correcciones. El recurso que ejecuta la mayor cantidad de estas correcciones es Marelis con 183 (28,50%) eventos.

La actividad Anular Check In de Habitación tiene un total de 30 (0,21%) de instancias donde el recurso que más ejecuta la misma es Odaisis, con 7 (9,18%) eventos. Del total de instancias existen 3 (10%) casos donde se anula el Check In de Habitación sin la existencia previa de un Check In / Modificación de Habitaciones lo que representa una anomalía en el proceso ya que para Anular Check In de Habitación debe existir anteriormente una actividad de Check In / Modificación de Habitaciones.

En el registro de eventos que se analiza, la actividad Cierre de Facturas cuenta con un total de 7 554 (54,55%) instancias de proceso, existen 1 470 (11,07%) eventos realizados por el recurso Alina lo que evidencia que es el recurso que en mayor cantidad de ocasiones realiza esta actividad. A la actividad Realizar Cargos se le realizan 3 589 Cierre de Facturas lo que representa un (47,51%) del total de instancias de la actividad Cierre de Facturas, siendo el recurso Alina el que más ejecuta estos Cierre de Facturas con un total de 980 (12,07%) eventos. La actividad Realizar Cargos cuenta con 3 945 (28,48%) instancias de las cuales a 356 (9,02%) no se le realiza Cierre de Facturas lo que evidencia anomalías en el proceso. Otro de los cargos es representado por la actividad Corrección/Cargo de Llamadas, a la cual se le realizan un total de 498 Cierres de Factura, en los que el recurso Victor es él que más ejecuta estas actividades con un total de 83 (9,58%) eventos. El total de instancias de procesos correspondientes a la actividad Corrección/Cargo de Llamadas es de 974 (7,03%) de las cuales a 258 (26,48%) no se les realiza Cierre de Facturas, lo que también representa una anomalía en la ejecución del proceso. El tercer tipo de cargo se denomina Tratamiento de Cargos Automáticos,

a los que se les Cierra Factura en 1 641 (21,72%) instancias de procesos, de los cuales la mayoría los realiza el recurso Alina con 178 (6,49%) eventos. La actividad Tratamiento de Cargos Automáticos está representada por 5 275 (38,09%) instancias de procesos, donde no se les realiza Cierre de Facturas a 3 478 (65,93%) de instancias, implicando anomalías en el proceso.

Existen casos en el hotel donde es necesario Anular un Depósito Adelantado, porque el cliente se marcha y todavía tiene dinero en su cuenta y se le debe entregar, o por error del carpetero al Realizar un Depósito a la habitación equivocada. Esta actividad tiene un total de instancias de 41 (0,29%), donde solamente en 1 (2,43%) caso se realiza el Anular Depósito Adelantado sin que este exista (Figura 3.8). Cabe destacar la importancia de encontrar esta anomalía aplicando técnicas de minería de procesos, ya que resulta casi imposible detectarla aplicando técnicas tradicionales de análisis de procesos.

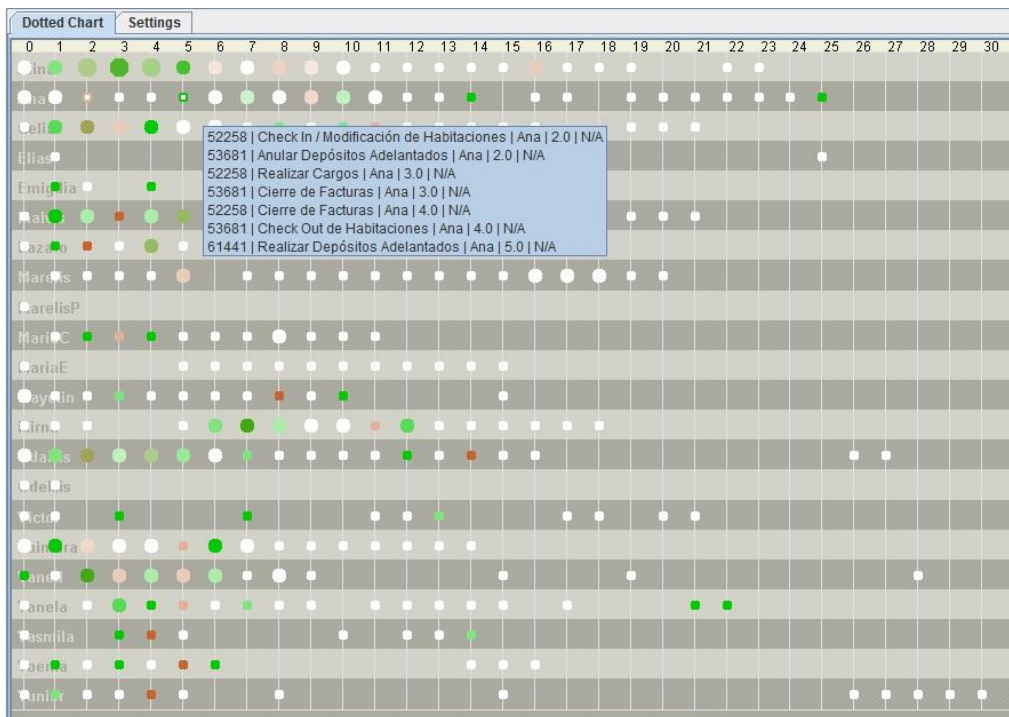


Figura 3.8: Anular Depósitos Adelantados.

3.6.3 Análisis del registro de eventos correspondiente a la temporada baja 2013

El registro de eventos está compuesto por 53 595 eventos, agrupadas en 11 177 instancias de procesos referentes a 12 actividades. La actividad Cambio de Habitación tiene un total de 1 323 (11,83%)

instancias de proceso, siendo el recurso Mayelin el que más realiza esta actividad con 152 (11,23%) eventos. Al igual que en el análisis anterior, a los turistas con nacionalidad canadiense son a los que más se les realiza esta actividad con 704 (53,21%) instancias, donde las habitaciones a las que más cambios se les realizan son las de tipo Tropical con un total de 456 (63,68%) instancias. Con la nacionalidad cubana se mantienen las estadísticas del análisis anterior y constituye el segundo país con mayor cantidad de Cambio de Habitación, con un total de 436 (32,95%) instancias. De forma similar se mantiene el tipo de habitación Tropical con 292 (65,61%) eventos de Cambio de Habitación.

Siempre que exista un Cambio de Habitación, si la habitación correspondiente contiene algún cargo asociado se le debe realizar un Traslado de Cargos a la nueva habitación. De 171 (1,52%) instancias referentes a la actividad Traslado de Cargos solamente en 6 (3,50%) casos se cumple con la condición anterior. En 72 (42,10%) casos se realiza el Cambio de Habitación, intermedio varias actividades y luego la actividad Traslado de Cargos. Las 93 (54,38%) instancias restantes presentan la actividad Traslado de Cargos sin que exista anteriormente un Cambio de Habitaciones, lo que representa una anomalía en la ejecución del proceso como se muestra en la siguiente figura (Figura 3.9).

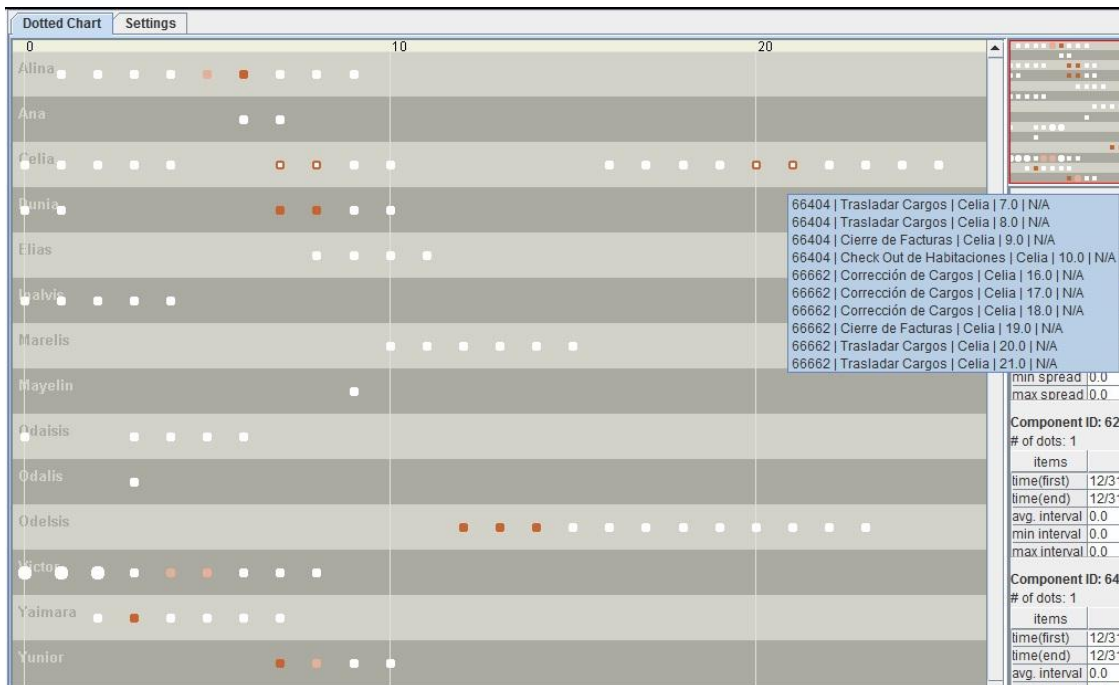


Figura 3.9: Traslado de Cargos sin Cambio de Habitaciones.

La cantidad de instancias que cumplen con una correcta ejecución del proceso al comenzar con la actividad Check In / Modificación de Habitaciones y terminar con la actividad Check Out de Habitaciones suman un total de 9 114 (81,54%). El resto de las instancias que no cumplen con esta condición son 2 063 (18,45%), de las cuales 574 (27,82%) comienzan con la actividad Check In / Modificación de Habitaciones; 1 390 (67,37%) terminan con la actividad Check Out de Habitaciones y 99 (4,79%) comienzan y terminan con otras actividades diferentes.

3.6.4 Análisis del registro de eventos correspondiente a la temporada alta 2013

El registro de eventos está compuesto por 79 366 eventos, agrupadas en 14 159 instancias de procesos referentes a 12 actividades. Comienzan con la actividad Check In / Modificación de Habitaciones y terminan con la actividad Check Out de Habitaciones 10 046 (70,95%) instancias de proceso, no cumplen con esta regla 4 113 (29,04%) instancias de proceso. Del total de instancias que no representan una correcta ejecución del proceso 1 153 (28,03%) comienzan con la actividad Check In / Modificación de Habitaciones; 2 695 (65,52%) terminan con la actividad Check Out de Habitaciones y 265 (6,44%) comienzan y terminan con otra actividad.

La actividad Anular Check In de Habitación está contenida en un total de 58 (0,40%) instancias de proceso, en 2 (3,44%) ocasiones se ejecuta esta actividad sin que exista previamente la actividad Check In / Modificación de Habitaciones, a su vez en uno de estos dos casos se ejecuta la actividad Anular Check In de Habitación 5 veces como se muestra en la figura (Figura 3.10) siguiente.

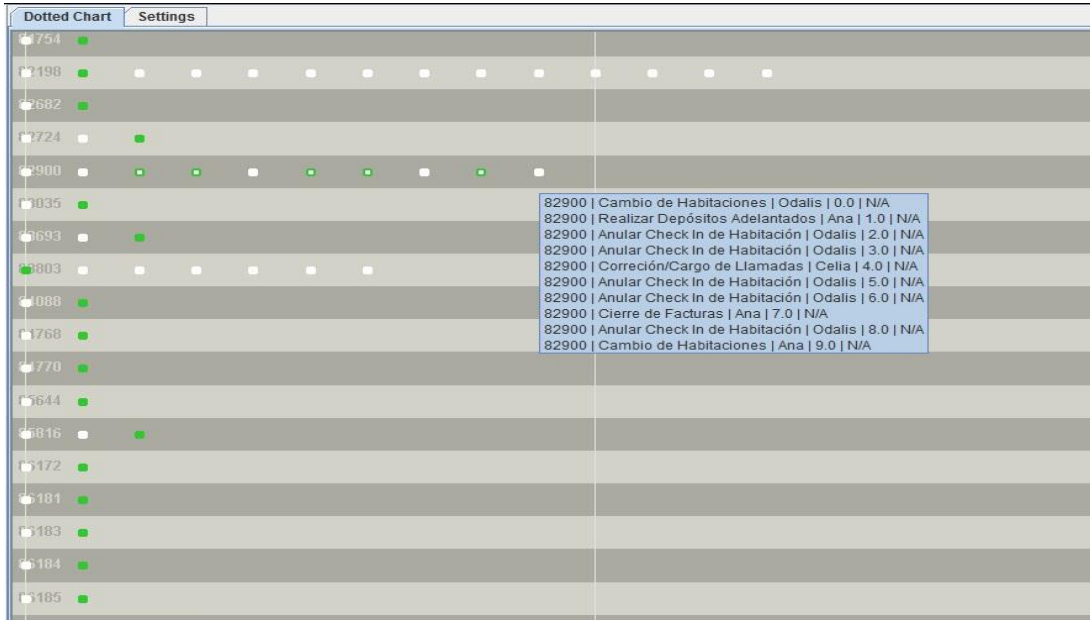


Figura 3.10: Anular Check In de Habitación repetido.

La actividad Trasladar Cargos se ejecuta en un total de 330 (2,33%) instancias. En 327 (99,09%) instancias se realiza la actividad Trasladar Cargos sin existir previamente la actividad Cambio de Habitaciones. Este dato es de gran relevancia para el análisis de proceso ya que el porcentaje que representa esta anomalía es significativo, siendo la entidad hotelera la principal afectada por la ocurrencia de estas anomalías.

Respecto a la ejecución correspondiente a la actividad Anular Depósitos Adelantados, la cual está presente en 13 (0,09%) instancias de proceso. En 6 (46,15%) casos se ejecuta el siguiente comportamiento Realizar Depósitos Adelantados – Anular Depósitos Adelantados – Realizar Depósitos Adelantados, lo que significa que el depósito fue anulado por error.

3.6.5 Análisis del registro de eventos correspondiente a la temporada baja 2014

El registro de eventos está compuesto por 24 815 eventos, agrupadas en 5 379 instancias de procesos referentes a 12 actividades. El correcto funcionamiento del proceso se evidencia en un total de 4 461 (82,63%) instancias del proceso, donde se comienza por la actividad Check In / Modificación de Habitaciones y se termina en la actividad Check Out de Habitaciones. En 918 (17,06%) casos se incumple esta regla donde 348 (37,90%) instancias comienzan con Check In / Modificación de Habitaciones; 523 (56,97%) instancias terminan con la actividad Check Out de Habitaciones y 47 (5,11%) comienzan y concluyen con otra actividad diferente a las antes mencionadas.

La actividad Cierre de Facturas se ejecuta en un total de 1 557 (28,94%) instancias, donde 710 (45,60%) se le realizan a la actividad Realizar Cargos; 432 (27,74%) a la actividad Tratamiento de Cargos Automáticos y 60 (3,85%) a la actividad Corrección/Cargo de Llamadas. Se realizan 355 (22,80%) Cierre de Facturas donde anteriormente no existe ningún cargo.

En el presente registro de eventos la actividad Trasladar Cargos se ejecuta en 62 (1,15%) instancias donde en solo 1 (1,61%) caso se realiza la actividad Trasladar Cargos luego de un Cambio de Habitaciones. También en 1 (1,61%) caso primero se realiza la actividad Trasladar Cargos y luego se ejecuta el Cambio de Habitaciones.

Tanto en este registro de eventos, como en los analizados anteriormente la actividad Check Out en Cierre de Facturas no se realiza, lo cual constituye una anomalía ya que esta actividad debe de ser ejecutada cuando algún cliente no tenga asociado ningún tipo de cargo a su habitación. Existen en los registros de eventos casos referentes a lo explicado anteriormente, por lo que se puede llegar a la conclusión de la relevancia que constituye el problema detectado.

Los análisis realizados con anterioridad tomaron un total de 10 horas aproximadamente. Estos análisis se presentaron al Lic. Fidel Pineda Bravo, profesor de la carrera de Turismo en la Facultad de Ciencias Económicas de la Universidad Central de Las Villas. El mismo se desempeñó por 2 años como profesor investigador de la Escuela de Hotelería y Turismo de Villa Clara Alberto Delgado Delgado, en esta institución realizó análisis de procesos de negocio en hoteles de ciudad y sol y playa de cadenas hoteleras como Horizontes, Isla Azul y Meliá; en todos estos hoteles se encuentra desplegado el sistema ZUN Suite. El especialista consultado realizó una estimación basada en su experiencia, de

cuánto tiempo tomaría realizar un análisis de procesos utilizando técnicas tradicionales que involucre esta cantidad de información y permita obtener resultados similares como los obtenidos utilizando técnicas de minería de procesos. El tiempo estimado para estos análisis es de 10 días como mínimo. De esta manera se comprueba que aplicando técnicas de minería de procesos es posible disminuir el tiempo requerido para el análisis de procesos de negocio, en entidades hoteleras donde se utiliza el sistema ZUN Suite.

3.7 Conclusiones parciales

El desarrollo de la técnica como un complemento del marco de trabajo ProM permite la utilización de sus resultados por parte de otros complementos del mismo. La misma permite a un analista obtener un registro de eventos para analizar los procesos que representa sin tener un conocimiento previo de cómo se encuentran estructurados los datos en la base de datos ni necesitar otras herramientas ajenas al ProM.

Las funcionalidades del componente desarrollado facilitan la aplicación de técnicas de minería de procesos en las instituciones hoteleras donde se utilice el sistema de información ZUN Suite como se evidencia en el caso de estudio. El análisis de los registros de eventos posibilita realizar un estudio de los datos registrados, facilitando una mejor comprensión de cómo realmente se ejecuta el proceso de Carpeta. El análisis de este proceso utilizando las técnicas de minería de procesos permite identificar anomalías en menor tiempo que utilizando técnicas tradicionales, que pueden afectar a la entidad hotelera y permite procesar mayor cantidad de información en la ejecución del proceso.

Conclusiones generales

Se realizó un análisis del estado del arte, sobre las diferentes técnicas de minería de procesos. El cumplimiento de los objetivos propuestos permite arribar a las siguientes conclusiones:

- A partir del análisis del proceso y de la estructura del formato XES se identificó que para realizar un análisis en entornos de la gestión hotelera es necesario considerar además de los datos requeridos por el formato XES la nacionalidad, el tipo de habitación y la habitación.
- El algoritmo diseñado permite la transformación de los datos identificados en la base de datos y su conversión hacia un registro de eventos en formato XES.
- El componente desarrollado para el marco de trabajo ProM de la minería de procesos, propicia la extracción de los registros de eventos de las base de datos del sistema ZUN Suite para su posterior análisis.
- La aplicación de técnicas de minería de procesos a los registros de eventos extraídos permite identificar anomalías y obtener una visión clara de cómo se ejecuta realmente el proceso analizado, perteneciente al módulo ZUNpms del sistema ZUN Suite.
- Los resultados obtenidos permiten comprobar que aplicando técnicas de minería de procesos, es posible disminuir el tiempo requerido para el análisis de procesos de negocio en entidades hoteleras donde se utiliza el sistema ZUN Suite.

Recomendaciones

- Se recomienda integrar en próximas versiones del sistema ZUN Suite, las técnicas utilizadas en el análisis del proceso Carpeta.
- Se recomienda mejorar en próximas versiones la trazabilidad del sistema ZUN Suite, para que se registre adecuadamente la relación entre los eventos que ocurren y las instancias de procesos a la que corresponden.
- Se recomienda incorporar al sistema de trazas del software de gestión ZUN Suite los datos identificados en la presente investigación como requeridos para analizar procesos en el área de la gestión hotelera.
- Se recomienda realizar este tipo de análisis sobre otros procesos en el área de la gestión hotelera.

Referencias bibliográficas

- AALST, Wil M. P. van der, 2011, Mining Additional Perspectives. In:Process Mining[online]. Springer Berlin Heidelberg. p.215–240. [Accessed 5 December 2013]. ISBN 978-3-642-19344-6, 978-3-642-19345-3. Available from: http://link.springer.com/chapter/10.1007/978-3-642-19345-3_8
- AALST, WM P van der, WEIJTERS, A J M M y MARUSTER, L, 2004. WorkflowMining:Discovering process models from event logs. En: IEEE Transactions on Knowledge and DataEngineering [enlínea].2004.Vol.16,no.9,pp.1128–1142.DOI 10.1109/TKDE.2004.47.Disponible desde:http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1316839.
- AALST, W.M.P. Van Der, 2012.Decomposing process mining problems using passages [enlínea]. Hamburg: s.n. 33rd International Conference on Application and Theory of PetriNetsandConcurrency,PETRINETS2012.ISBN03029743(ISSN);9783642311307 (ISBN).Disponible desde:<http://hinarigw.who.int/whalecomwww.scopus.com/whalecom0/inward/record.url?eid=2-s2.0-84862503604&partnerID=40&md5=dca5ca92267c6ff5800170e6769321fc>.
- AALST W. M. P, 2013, Mine your own business: Using process minig to turn big data into realvalue. Eindhoven. In: 2013.
- Abrahamsson, P., y otros. 2002. Agile software development methods Review and analysis. s.l. : VTT Publications, 2002.
- Beck, K. 1999. Extreme Programming Explained. Embrace Change. s.l. : Pearson Education, 1999.
- Echeverry Tobón, Luis Manuel y Delgado Carmona, Luz Elena. 2007. Caso práctico de la metodología ágil XP al desarrollo de software. Pereira : Universidad Tecnológica de Pereira, 2007.
- Figueroa, Roberth G., Solís, Camilo J. y Cabrera, Armando A. 2009. Metodologías Tradicionales Vs. Metodologías Ágiles. 2009.
- FUNDORA-RAMÍREZ OSIEL, 2013, Impacto de las características de un registro de evento enalgoritmos de descubrimiento de la minería de proceso. In: VIII Peña Tecnológica. VI Taller Científico provincial de jóvenes de la especialidad de tecnologías y sistemas. 2013.

- Gao, X. (2013). Towards the Next Generation Intelligent BPM. In the Era of Big Data. Department of Management Information System. China Mobile Communications Corporation, Beijing 100033, China.
- GÜNTHER C. W., Aalst W. M.P... 2007. Fuzzy Mining – Adaptive Process Simplification Based on Multiperspective Metrics. Brisbane, Australia : Springer, 2007. págs. 328-343.
- GÜNTHER C. W. (2009). Fuzzy Mining. In the Era of Big Data. Department of Management Information System. China Mobile Communications Corporation, Beijing 100033, China.
- H, YANG and DONGEN B.F, 2012, Estimating completeness of event logs. In: 2012.
- HERRERA, Raykenler Yzquierdo, CASTRO, Rogelio Silverio, CORTÉS, Manuel Lazo and GRAÑA, Adrian Torres, 2012, Minería de procesos como herramienta para la auditoria. Ciencias de la Información [online]. 2012. [Accessed 13 April 2012]. Available from: <http://intranet2.uci.cu/node/12558/postgrado/maestrias>
- IEEE Task Force On Process Mining, 2011. Manifiesto sobre Minería de Proceso [online]. S.l.:s.n. Disponible desde: <http://www.win.tue.nl/ieeetfpm/lib/exe/fetch.php?media=shared:pmm-spanish-v1.pdf>.
- J.C.A.M. Buijs, 2010. Mapping Data Sources to XES in a Generic Way [online]. Master Thesis. Eindhoven, The Netherlands: Technische Universiteit Eindhoven. [Accedido 16 marzo 2012]. Disponible desde: http://www.processmining.org/_media/xesame/xesma_thesis_final.pdf.
- Jeffries, R., Anderson, A. y Hendrickson, C. 2001. Extreme Programming Installed. s.l. : Addison-Wesley, 2001.
- Joskowicz, Jose. 2008. Reglas y Prácticas en eXtreme Programming. [En línea] 2008. [Citado el: 28 de Febrero de 2014.] <http://ie.fing.edu.uy/~josej/docs/XP%20-%20Jose%20Joskowicz.pdf>.
- Oracle Corporation And Its Affiliates, 2012. NetBeans IDE - Features. En: NetBeans IDE [online]. 2012. Disponible desde: <http://netbeans.org/features/index.html>.

- Padanés, María Carmen y Letelier, Patricio. 2007. Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP). Valencia : Laboratorio de Sistemas de Información. Departamento de Sistemas Informáticos y Computación., 2007.
- Pantaleón, Marta E. Zorrilla. Universidad de Cantabria. Universidad de Cantabria. [Online] [Cited: Marzo 21, 2013.] <http://personales.unican.es/zorrillm/PDFs/Docencia/Master/02%20-%20Modelos%20de%20datos%20ER-UML-relacional.pdf>.
- Pérez Huebe, Ma. de Lourdes. 2005. Ingeniería de Requerimientos. Pachuca : Universidad Autónoma del Estado de Hidalgo, 2005.
- Pressman, Roger. 2007. Ingeniería del Software, Un enfoque práctico. s.l. : Mc Graw Hill, 2007. roactivanet. [En línea] http://www.proactivanet.com/UserFiles/File/Noticias/Gesti%C3%B3n%20del%20Conocimiento_Inteli%281%29.pdf.
- Rozinat, A., Aalst, W.M.P. van der, 2008. Conformance checking of processes based on monitoring real behavior. *Inf. Syst.* 33, 64–95.
- Song, M. & van der Aalst, W.M.P. (2007). Supporting Process Mining by Showing Events at a Glance. Trabajo presentado en la 17th Annual Workshop on Information Technoregistroies and Systems, Montreal, Canadá.
- VAN DER AALST, W., ADRIANSYAH, A., DE MEDEIROS, A.K.A., ARCIERI, F., BAIER, T., BLICKLE, T., BOSE, J.C., VAN DEN BRAND, P., BRANDTJEN, R., BUIJS, J., BURATTIN, A., CARMONA, 2012, Process mining manifesto [online]. Clermont-Ferrand. 9th International Conference on Business Process Management, BPM 2011P. ISBN 18651348 (ISSN); 9783642281075 (ISBN). Available from: <http://hinarigw.who.int/whalecomwww.scopus.com/whalecom0/inward/record.url?eid=2-s2.0-84863011087&partnerID=40&md5=07d0f3bad2de1ffec274ccdb8c971b00> © 2012 Springer-Verlag.
- van der Aalst, W.M.P. (2011). *Process Mining. Discovery, Conformance and Enhancement of Business Processes*. London New York: Springer.
- van der Aalst, W.M.P., Adriansyah, A., Medeiros, A.K.A. et al. (2011). *Process Mining Manifesto*. IEEE.

- van der Aalst, W.M.P., ter Hofstede, Arthur, H. M. & Weske, M. (2003). Business Process Management: International Conference. Eindhoven University of Technoregistry, Netherlands.
- VISUAL PARADIMG, 2011. Visual modeling tool for building enterprise applications. En: Visual Paradim website [online]. 2011. Disponible desde: <http://www.visualparadigm.com/product/vpuml/provides/>.
- WEIJTERS A.J.M.M., AALST. W.M.P., 2003. Rediscovering Workflow Models from Event-Based Data using Little Thumb. 2003. págs. 151-162. Vol. 10
- WESKE, Mathias, 2007, *Business process management concepts, languages, architectures* [online]. Berlin; New York : Springer. [Accessed 3 November 2014]. ISBN 9783540735229 3540735224 3540735216 9783540735212. Available from: <http://site.ebrary.com/id/10230314>
- Yzquierdo, R. (2012). Modelo para la estimación de información ausente en las trazas usadas en la minería de proceso. Tesis de Doctorado, Universidad de las Ciencias Informáticas, Cuba.