



***Interfaz web para el monitoreo de la herramienta de réplica Slony***

*Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas*

***Autora:*** Nurisely Ramos López

***Tutores:*** Ing. Dalilys Martínez Gutiérrez  
Ing. Marcos Michel Martínez Pérez  
Ing. Rosnel Venero Acosta

***Universidad de las Ciencias Informáticas, Facultad 6***

***La Habana, junio de 2015***  
***Año 57 de la Revolución***



*"La revolución es algo que se lleva en el alma, no en la boca para vivir de ella"*

*Ernesto Che Guevara*

# DECLARACIÓN DE AUTORÍA

Por este medio declaro que soy la única autora de este trabajo y autorizo a la Universidad de las Ciencias Informáticas (UCI) a hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

---

Nurisely Ramos López

Autora

---

Ing. Dalilys Martínez Gutiérrez.

Tutora

---

Ing. Marcos Michel Martínez Pérez

Tutor

---

Ing. Rosnel Venero Acosta

Tutor

## DATOS DE CONTACTOS

Tutora:

Ing. Dalilys Martínez Gutiérrez

Universidad de las Ciencias Informáticas, La Habana, Cuba

Correo electrónico: [dmgutierrez@uci.cu](mailto:dmgutierrez@uci.cu)

Tutor:

Ing. Marcos Michel Martínez Pérez

Universidad de las Ciencias Informáticas, La Habana, Cuba

Correo electrónico: [mmartinezp@uci.cu](mailto:mmartinezp@uci.cu)

Tutor:

Ing. Rosnel Venero Acosta

Universidad de las Ciencias Informáticas, Habana, Cuba

e-mail: [rvacosta@uci.cu](mailto:rvacosta@uci.cu)

## AGRADECIMIENTOS

Agradezco a mi mamá Cecilia que aunque ya no esté entre nosotros sé que me estás viendo y que siempre estarás conmigo, por su amor, cariño, su confianza, por haberme convertido en la persona que hoy soy, por haberme guiado por el camino correcto. Le agradezco por todos los esfuerzos realizados para convertirme en lo que soy, estoy orgullosa de ser su hija. Te amo mamita.

A mi papa Bertho por ser uno de mis ídolos, por ser mama y papa, por ser mi consejero, por ser mi ejemplo, por apoyarme en cada decisión que tome en mi vida, por haberme sentido desde mi nacimiento hasta la mujer que hoy soy su niña especial, su Mumi. Te agradezco papa por nunca haberme dejado sola en los momentos más difíciles de mi vida, aunque a veces tuve q halarle las orejas, espero no hacerlo más. Te amo mucho papa.

Agradezco a Maritza mi madrastra por haber hecho a mi papa feliz de nuevo y por darle ese ser tan divino (mi hermanito) y Belsay mi hermanastra que espero que siga también mi ejemplo, que estudie y se supere.

A mis abuelos Bertho, Nury, Juan por su apoyo incondicional, por su amor y dedicación. Gracias por todo el cariño que me han dado. Los quiero mucho. En especial a mi abuela Cecilia por ser mi segunda madre por malcriarme, por hacerme esas comidas ricas y dulces ricos que siempre le pido cada vez que voy a Santiago. Te doy las gracias abue por estar aquí conmigo hoy en este día tan especial. Te amo abue.

A mi hermana Yilena que aunque no esté aquí conmigo le agradezco por sus consejos cuando se los pedí, por confiar en mí, por haber sido mi primera escuela de defensa, siempre nos fajábamos. Te quiero Yile.

Agradecer a mi novio Frank por su amor, cariño, comprensión. Gracias por estar siempre a mi lado, por ser mi compañero, consejero, amigo, gracias por ayudarme cuando más lo necesitaba, a pesar de todo. Agradezco mucho por soportarme y hacerme feliz en 1 año y 8 meses que cumplimos hoy Felicidades Amorsote. Te amo mucho.

Agradecer a mi familia, a mis tías Berzaida y Chucha, a mi tíos Leonardo y Carlos que ya no está entre nosotros, por haberme apoyado y quererme, a todos mis primos. Los quiero.

A una familia súper importante para mí, que me acogió cuando más lo necesitaba, aunque no estén aquí, le agradezco mucho por su apoyo total, su amor, su dedicación, por dejarme ser parte de sus vidas a Consuelo, Yuni, Efren y Lili.

Agradezco a mis tutores porque sin su ayuda no habría logrado lo que hoy ven aquí. A Dalilys que a pesar de que al principio no nos entendimos como tesista-tutora, hemos creado una buena amistad, le agradezco por guiarme en todos los aspectos técnicos del trabajo de diploma, por sus consejos, por transmitirme seguridad, por su apoyo, gracias por defenderme en toda la trayectoria, para ti mis más sinceros agradecimientos. A Marcos Michel que aunque no fue mi tutor desde el primer momento que fueron asignados los tutores ha estado apoyándome y defendiéndome desde entonces, gracias por la dedicación, por el esmero, sé que detrás de tanto carácter, hay un hombre sensible. A Rosnel mi tutor asignado a última hora, pero me demostró interés, dedicación, gracias por darme tus consejos y por preocuparte. Agradezco también al tribunal de tesis por haberme corregido y guiado para alcanzar mejores resultados en el documento y en la aplicación.

A los profesores que de una forma u otra han estado apoyándome a lo largo de mi carrera y de ellos quienes fueron para mí más que un ejemplo: Osmel, Yuya, Alleyne, Oscar, Jim y los que se me quedan a quienes respeto y aprecio mucho, no por ser la última es la menos importante, al contrario gracias a ella hoy por hoy estoy aquí, y es por eso que le agradezco a Yanelis Benítez, por ser mi amiga, por ser como una madre para mí, gracias de corazón, te debo mucho.

A mis compañeros de grupo de todos estos años de los cuales la vida alejó a algunos, aproximó a otros, e hizo permanente la presencia de los mejores. Menciono a quienes me hicieron más agradable estos últimos años y a los que nunca olvidaré: Sarahí la negra más linda de la UCI, Laritza que aunque nos empezamos a llevar en este último curso te considero una amiga más, a Clauditina que a pesar de haberla conocido en este último año para mí ha sido una amiga más, le agradezco también por ayudarme con lo de la tesis, por dejarme robarle un poco de su tiempo te quiero, a la Taimi, a Eric y Bidot con los que compartí muchas cosas, Jisel, Taimí, Morgado, Arletty, Ayme, Leo, el Castri. A Yunior al que le debo todo, gracias por ayudarme, por darme de tu tiempo, por aguantarme en este último curso has sido un buen amigo. Agradecerle a mi compañera de cuarto, Yisell que aunque la vida nos jugó una mala pasada, no me importa te sigo queriendo. En especial agradezco a mis más grande amiga Yezy que con el paso del tiempo se convirtió en una hermana para mí, gracias por preocuparte, por estar siempre presente, en todo momento sentí su apoyo y cariño y quiero que sepas que puedes seguir contando conmigo como el primer día. Gracias por los buenos momentos que pasamos juntas.

Agradezco a mis suegros Tanya y Leonardo y a mi cuñada y hermanita Leyani por hacerme parte de su familia y por todos los momentos y ayuda que me han brindado, muchas gracias.

Y dejo para el final y no por ser los menos importantes, al contrario los quiero más que a nada, a mi hermanito Brian David y mi sobrina Mónica quienes han sido motor impulsor. Brian David quiero que tenga el ejemplo de su hermana mayor y luche por superarse en un futuro; y Mónica quiero también que siga el ejemplo de su tía. Recuerden que todo este esfuerzo lo he hecho pensando en ustedes.

## RESUMEN

La Universidad de las Ciencias Informáticas promueve el desarrollo de la ciencia y la tecnología mediante la realización de proyectos productivos. Como parte del desarrollo y soporte a las tecnologías de bases de datos surge el Centro de Tecnologías de Gestión de Datos, en dicho centro se encuentra el departamento Desarrollo de Componentes el cual incluye en los servicios que brinda el de réplica de datos. En esta investigación se desarrolla una Interfaz web para el monitoreo de la herramienta de réplica Slony. La interfaz web obtenida permitirá a los administradores de base de datos del departamento verificar el funcionamiento de cualquier solución de réplica de datos que se ejecute mediante dicha herramienta. Para el desarrollo de la interfaz se realizó el análisis y selección de la metodología, tecnologías y herramientas. Se identificaron las funcionalidades que debe cumplir y se realizó el diseño a partir del cual se implementó la solución. Para comprobar el correcto funcionamiento se realizaron pruebas de aceptación arrojando resultados satisfactorios.

**Palabras clave:** datos, interfaz, réplica.

## **ABSTRACT**

University of Information Sciences promotes the development of science and technology by conducting productive projects. As part of the development and technology support database Technology Center Data Management arises in that center is the Component Development department which includes services provided by the data replication. This research develops a web interface to monitor Slony replication tool. The web interface allows administrators obtained database of the department verify the operation of any data replication solution that runs through this tool. Development of the interface analysis and choice of methodology, technologies and tools are made. Functionality to be met were identified and design from which the solution was implemented was conducted. To check the correct operation acceptance tests were performed yielding satisfactory results.

**Keywords:** data, interface, replica.

# ÍNDICE DE CONTENIDOS

INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	5
1.1. Réplica de datos.....	5
1.1.1. Tipos de réplicas .....	5
1.1.2. Entornos de Réplica .....	6
1.1.3. Modelo de distribución de datos .....	6
1.2. Monitoreo de las soluciones de réplica .....	7
1.3. La herramienta Slony y sus tareas de monitoreo .....	8
1.4. Tecnologías a emplear para el desarrollo de la solución.....	10
1.4.1. Metodologías de desarrollo.....	10
1.4.2. Lenguaje de Modelado .....	13
1.4.3. Herramienta de Modelado .....	13
1.4.4. Lenguajes de Programación.....	14
1.4.5. Entorno de Desarrollo Integrado (IDE).....	16
1.4.6. Plataforma de desarrollo.....	16
Conclusiones del capítulo.....	17
CAPÍTULO: 2 DISEÑO E IMPLEMENTACIÓN DE LA SOLUCIÓN.....	18
2.1. Modelo del Domino.....	18
2.2. Descripción de la solución .....	19
2.2.1. Historias de usuario.....	20
2.2.2. Lista de reserva del producto.....	30
2.2.3. Plan de iteraciones .....	31

2.2.4. Tarjetas CRC.....	32
2.3. Arquitectura base de la aplicación .....	34
2.3.1. Patrones de arquitectura.....	34
2.3.2. Patrones de diseño.....	35
2.4. Implementación de la aplicación .....	37
2.4.1. Tareas de ingeniería.....	37
2.4.2. Estándares de codificación.....	42
Conclusiones del capítulo.....	51
CAPÍTULO 3: VALIDACIÓN DE LA SOLUCIÓN.....	52
3.1. Proceso de pruebas.....	52
3.1.1. Pruebas de aceptación.....	52
3.2. Diseño de caso de prueba y registro de no conformidades.....	53
3.3. Análisis de los resultados de las pruebas .....	59
Conclusiones del capítulo.....	60
CONCLUSIONES GENERALES.....	61
RECOMENDACIONES.....	62
REFERENCIAS BIBLIOGRÁFICAS.....	63

## ÍNDICE DE TABLAS

Tabla 1: Tablas del clúster de Slony .....	9
Tabla 2: Descripción de los conceptos del diagrama de dominio .....	19
Tabla 3: Historia de Usuario Autenticar usuario. ....	20
Tabla 4: Cambiar contraseña .....	20
Tabla 5: Configurar la conexión .....	21
Tabla 6: Historia de Usuario Visualizar listado de las tablas. ....	22
Tabla 7: Historia de Usuario Visualizar los datos de cada tabla según se van seleccionando por el usuario. .....	22
Tabla 8: Historia de Usuario Visualizar listado de los nodos. ....	23
Tabla 9: Historia de Usuario Visualizar los sets de réplica. ....	23
Tabla 10: Historia de Usuario Visualizar los log de la herramienta Slony. ....	24
Tabla 11: Historia de Usuario Visualizar los logs del servidor PostgreSQL. ....	25
Tabla 12: Historia de Usuario Mostrar gráficamente el consumo de RAM del proceso Slon. ....	25
Tabla 13: Historia de Usuario Mostrar gráficamente el consumo de RAM del proceso PostgreSQL. ....	26
Tabla 14: Historia de Usuario Visualizar los logs por tipos según las categorías en las que se divide. ....	26
Tabla 15: Historia de Usuario Mostrar gráficamente el histórico de las operaciones realizadas dado un conjunto de parámetros. ....	27
Tabla 16: Historia de Usuario Mostrar las tablas con los resultados de la búsqueda en el histórico de las operaciones realizadas dado un conjunto de parámetros. ....	28
Tabla 17: Historia de Usuario Extraer de la tabla sl_path la información de los nodos. ....	28
Tabla 18: Historia de Usuario Visualizar versión de Slony. ....	29
Tabla 19: Lista de reserva del producto .....	30
Tabla20: Plan de iteraciones.....	32
Tabla 21: Tarjeta CRC Master .....	33
Tabla 22: Tarjeta CRC Conexión .....	33
Tabla 23: Tarea de Ingeniería Autenticar usuario. ....	38
Tabla 24: Tarea de Ingeniería Cambiar contraseña. ....	38
Tabla 25: Tarea de Ingeniería Configuración de conexión. ....	38
Tabla 26: Tarea de Ingeniería Visualizar listado de las tablas del clúster de Slony. ....	39

Tabla 27: Tarea de Ingeniería Visualizar los datos de cada tabla según se van seleccionando por el usuario. .....	39
Tabla 28: Tarea de Ingeniería Listar los nodos y mostrar la información de estos. ....	39
Tabla 29: Tarea de Ingeniería Visualizar los set de réplicas. ....	39
Tabla 30: Tarea de Ingeniería Visualizar los logs de la herramienta Slony. ....	40
Tabla 31: Tarea de Ingeniería Visualizar los logs del servidor PostgreSQL. ....	40
Tabla 32: Tarea de Ingeniería Mostrar gráficamente el consumo de RAM del proceso Slon. ....	40
Tabla 33: Tarea de Ingeniería Mostrar gráficamente el consumo de RAM del proceso PostgreSQL. ....	40
Tabla 34: Tarea de Ingeniería Visualizar los logs por tipos según las categorías en las que se divide. ....	41
Tabla 35: Tarea de Ingeniería Mostrar gráficamente el histórico de las operaciones realizadas dado un conjunto de parámetros. ....	41
Tabla 36: Tarea de Ingeniería Mostrar las tablas con los resultados de la búsqueda en el histórico de las operaciones realizadas dado un conjunto de parámetros. ....	41
Tabla 37: Tarea de Ingeniería Extraer de la tabla sl_path la información de los nodos. ....	42
Tabla 38: Tarea de Ingeniería Visualizar versión de Slony. ....	42
Tabla 39: Secciones para probar la historia de usuario Autenticar usuario ....	54
Tabla 40: Descripción de variables de la historia de usuario Autenticar usuario ....	55
Tabla 41: Listado de relaciones ....	55
Tabla 42: Matriz de datos de la historia de usuario Autenticar usuario específicamente de la Sección. ....	56
Tabla 43: No conformidades detectadas en la aplicación. ....	58

## ÍNDICE DE FIGURAS

Figura 1: Replicación a múltiples esclavos, siendo un esclavo a su vez maestro para otros esclavos. ....	8
Figura 2: Modelo de dominio.....	19
Figura 4: Diagrama de clases (Patrón MVC).....	35
Figura 5: Patrón Controlador.....	36
Figura 6: Patrón Experto.....	37
Figura 7: Interfaz de autenticación de usuario.....	45
Figura 8: Interfaz principal.....	45
Figura 9: Tablas del clúster de Slony. ....	46
Figura 10: Nodos de la réplica. ....	46
Figura 11: Logs del sistema. ....	47
Figura 12: Sets del sistema.....	48
Figura 13: Interfaz Réplica. ....	49
Figura 14: Interfaz Sistema. ....	49
Figura 15: Interfaz Cambiar Contraseña. ....	50
Figura 16: Interfaz Configuración de conexión. ....	50
Figura 17: Resultados de las pruebas por iteración. ....	60

## INTRODUCCIÓN

En la actualidad el gradual desarrollo alcanzado por las Tecnologías de la Información y las Comunicaciones (TIC) facilita la creación, distribución y manipulación de grandes volúmenes de información. El volumen de datos generado cada año crece de forma exponencial en todos los sectores. Debido a este crecimiento se ha hecho prácticamente imposible la gestión del mismo sin el uso de los sistemas informáticos.

Un sistema informático es un conjunto de hardware, software y recursos humanos que contiene funciones interrelacionadas, las cuales interactúan entre sí para lograr un objetivo. Estos cuentan con software específico, como son los Sistemas Gestores de Bases de Datos (SGBD) los que agilizan y optimizan la entrega de los recursos solicitados, organizan y gestionan la relación de los datos almacenados en la base de datos para su fácil manejo, haciendo más rápida la localización y acceso a estos, incluso desde lugares remotos.

Para aumentar la disponibilidad de los datos en los SGBD, mejorar los tiempos de respuestas e incrementar la tolerancia a fallos, se ha desarrollado la técnica de réplica de datos, esta se basa en mantener objetos de las bases de datos en múltiples ubicaciones. De esta forma se puede tener una réplica exacta o parcial de la base de datos que puede usarse para una restauración de emergencia o bien para realizar consultas sobre ella y aligerar la carga del servidor principal.

En la Universidad de las Ciencias Informáticas (UCI), se han creado un grupo de centros productivos destinados al desarrollo de aplicaciones informáticas, como es el Centro de Tecnologías de Gestión de Datos (DATEC), encargado de crear productos informáticos, desarrollar tecnologías y proveer servicios relacionados con la gestión de datos y el análisis de información, los sistemas de información y los sistemas de inteligencia de negocios para apoyar los procesos de dirección de las organizaciones.

Uno de los departamentos pertenecientes a dicho centro es el de Desarrollo de Componentes; entre los servicios más solicitados por los clientes al departamento se encuentra el de réplica de datos, en cuya soluciones se emplean varias herramientas, entre ellas Slony. Esta herramienta de replicación es muy utilizada dado que permite configurarse de “maestro a múltiples esclavos”, soporta la réplica en cascada y permite además a un esclavo ser a su vez maestro para otro servidor. Dicha herramienta presenta el inconveniente de que su administración y monitorización deben realizarse mediante la consola de comandos, resultando necesario un conocimiento avanzado de los administradores. Teniendo en cuenta lo antes plantado se define como **problema de investigación**:

## INTERFAZ WEB PARA EL MONITOREO DE LA HERRAMIENTA DE RÉPLICA SLONY

¿Cómo facilitar a los administradores de bases de datos, el monitoreo de las soluciones de réplica de datos realizadas con la herramienta Slony?

Para dar solución al problema planteado se define como **objeto de estudio**: el proceso de monitoreo de réplica de datos, enmarcado en el **campo de acción**: proceso de monitoreo de la réplica de datos con Slony.

Para dar solución al problema planteado se define como **objetivo general**: desarrollar una interfaz web para el monitoreo de la herramienta Slony que posibilite una monitorización adecuada de las soluciones de réplica desarrolladas con esta, derivándose los siguientes **objetivos específicos**:

- Realizar un estudio basado en los procesos de monitoreo de la herramienta Slony.
- Diseñar la interfaz web para el monitoreo de la herramienta Slony.
- Implementar la interfaz web para el monitoreo de la herramienta Slony.
- Realizar las pruebas a la interfaz web para el monitoreo de la herramienta Slony.

Para dar cumplimiento a los objetivos antes mencionados se definen las siguientes **tareas de investigación**:

- Revisión de los procesos de monitoreo de la herramienta Slony para identificar los requisitos funcionales de la aplicación.
- Selección de las herramientas y tecnologías a utilizar para el desarrollo de la solución.
- Definición de los requisitos funcionales y no funcionales de la solución.
- Implementación de los requisitos funcionales de la interfaz web para el monitoreo de la herramienta de réplica Slony.
- Diseño de los casos de prueba para los requisitos funcionales implementados en la solución.
- Validación de los requisitos funcionales implementados.

Según los elementos anteriormente expuestos se plantean como **preguntas científicas**:

- ¿Qué elementos se deben tener en cuenta para implementar una solución de réplica de datos?
- ¿Cómo efectúa las tareas de monitorización la herramienta de réplica de datos Slony para mantener controladas las soluciones de réplica de datos?
- ¿Existe alguna herramienta que permita la monitorización de Slony?

Los métodos de investigación que soportan el desarrollo del presente trabajo son la combinación dialéctica

de los **métodos Teóricos y Empíricos**. Entre los **Teóricos** se emplearán:

**Analítico-Sintético:** Se emplea para lograr extraer los elementos esenciales de las bibliografías consultadas con el objetivo de caracterizar los procesos de replicación y las soluciones de réplica de datos. Además de permitir realizar un análisis de soluciones semejantes a la que se desea implementar para obtener información que beneficie el proceso investigativo.

**Histórico-Lógico:** La investigación se realizará a partir del estudio del estado del arte de la problemática analizada, que permitirá conocer la trayectoria y desarrollo de aplicaciones para garantizar la sincronización de datos, teniendo en cuenta sus antecedentes históricos, las investigaciones realizadas y los resultados obtenidos por otros autores al efectuar dichas investigaciones, lo que dará la posibilidad de seleccionar aquellas herramientas, lenguajes y metodologías lógicas y adecuadas para el desarrollo de la aplicación.

### **Métodos Empíricos:**

**Observación:** Permitirá la ejecución correcta del proyecto, alertando sobre posibles atrasos y fallas en el cumplimiento de las tareas propuestas.

**Revisión de documentos:** Permitirá la elaboración del marco teórico conceptual y el desarrollo del diseño de la propuesta de la interfaz web.

El informe de la investigación está organizado en 3 capítulos:

- **Capítulo 1: Fundamentación teórica:** En este capítulo se incluyen los resultados del estudio del estado del arte de la investigación. Se sintetizan conceptos relacionados con la réplica de datos, se presentan la metodología, tecnologías y herramientas que se utilizarán para el desarrollo de la solución.
- **Capítulo 2: Diseño e implementación de la solución:** En este capítulo se presenta el diseño de la solución, como parte de este se muestra el modelo de dominio para explicar la interacción de los conceptos asociados al negocio; las historias de usuario, la lista de reserva del producto, el plan de iteraciones, las tarjetas CRC, el diagrama de clases, además para la implementación de la misma se presentan las tareas de ingeniería y los estándares de codificación a emplear.
- **Capítulo 3: Validación de la solución:** En este capítulo se presentan la estrategia de prueba definida por la metodología XP, que permiten comprobar el correcto funcionamiento de la solución,

## **INTERFAZ WEB PARA EL MONITOREO DE LA HERRAMIENTA DE RÉPLICA SLONY**

teniendo en cuenta los tipos de pruebas y las técnicas para ejecutarlas. Además se presenta el resultado de la aplicación de estas a las funcionalidades implementadas, comprobando el correcto funcionamiento de la solución.

## CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

En la actualidad son varias las empresas, instituciones y organizaciones que debido a su acelerado crecimiento por todo el mundo sienten la necesidad de establecer sedes o sucursales distribuidas en diferentes lugares geográficamente. Esta situación trae como consecuencia un gran flujo de información que debe ser analizada en cada una de las sedes, por tanto el uso de una base de datos centralizada podría ocasionar problemas de rendimiento al aumentar los tiempos de respuesta de la base de datos y además disminuirían los niveles de disponibilidad de la información. Como alternativa a estos problemas surge la réplica de datos.

En este capítulo se exponen conceptos y aspectos teóricos asociados a la réplica de datos. Se analiza el proceso de monitoreo de réplica de datos utilizando Slony. Además, se explican detalladamente los criterios de selección de las herramientas, tecnologías y metodología de desarrollo de software a utilizar para el desarrollo de una interfaz web para el monitoreo de la herramienta de réplica Slony.

### 1.1. Réplica de datos

La réplica de datos ha sido un tema ampliamente discutido por compañías desarrolladoras de gestores de bases de datos y software en general. Se entiende como réplica de datos al proceso de copiar y mantener objetos y datos entre dos o varios servidores de bases de datos, en un sistema de bases de datos de tipo distribuido. (1)

La replicación es un conjunto de tecnologías destinadas a la copia y distribución de datos desde una base de datos hacia otra o múltiples bases de datos, para luego sincronizarlas y mantener su coherencia. Además permite distribuir datos entre diferentes ubicaciones y entre usuarios remotos o móviles mediante redes locales y de área extensa, conexiones de acceso telefónico, conexiones inalámbricas e Internet. De forma general esta permite seleccionar de manera precisa los datos a replicar, enviar a cada destino solo los datos necesarios e incluso aplicar transformaciones en estos datos replicados. (2)

#### 1.1.1. Tipos de réplicas

Las réplicas se dividen en dos grupos dentro de los que se encuentran la:

- **Replicación Básica**

Con la replicación básica, la replicación de los datos proporciona el acceso a sólo lectura a los datos de las tablas que provienen de un sitio primario (maestro). Las aplicaciones pueden preguntar datos de las réplicas de datos locales para evitar el acceso a la red, sin importar su disponibilidad. Sin embargo, las aplicaciones a través del sistema deben tener acceso a los datos en el sitio primario cuando las actualizaciones sean necesarias. (3)

- **Replicación Avanzada (Simétrica)**

Las características avanzadas de replicación amplían las capacidades básicas de sólo lectura de la replicación, permitiendo que las aplicaciones hagan actualizaciones a las réplicas de las tablas, a través de un sistema replicado de la base de datos. Con la replicación avanzada, los datos pueden proveer lectura y acceso a actualizaciones a los datos de las tablas. (3)

### **1.1.2. Entornos de Réplica**

Los tipos de entornos de réplica son los siguientes:

- **Maestro- Esclavo:** También conocida como de solo lectura, permite a un solo maestro recibir consultas de lectura/escritura, mientras los esclavos solo pueden aceptar consultas de lectura. (4)
- **Multi-Maestro:** También llamada par-a-par o la réplica de camino de n, permite múltiples sitios, actuando como pares iguales. Cada sitio en un ambiente de réplica de multi-maestro es un sitio de maestro, y cada sitio se comunica con otros sitios maestros. Esta capacidad tiene también un severo impacto en el desempeño, debido a la necesidad de sincronizar los cambios entre los servidores. Este tipo de entorno puede ser usado para mantener sitios recuperables ante posibles desastres o caídas, así como para proveer sistemas con alta disponibilidad y para balancear la carga de consultas a través de las distintas ubicaciones. (4)

### **1.1.3. Modelo de distribución de datos**

Existen dos modelos de distribución de datos esencialmente aplicados a cada uno de los entornos antes vistos:

- **Asincrónica:** A menudo llamada almacena- y- reenvía, captura cualquier cambio local, los almacena en una cola, y, a intervalos regulares, propaga y aplica estos cambios en sitios remotos. Con esta forma de réplica, hay un período de tiempo antes de que todos los sitios alcancen la convergencia de datos. (4)
- **Sincrónica:** También conocida como la réplica en tiempo real, aplica cualquier cambio o ejecuta cualquier procedimiento reproducido en todos los sitios que participan en el ambiente de réplica como parte de una sola transacción. Si el procedimiento falla en cualquier sitio, entonces la transacción entera se anula. La réplica sincrónica asegura la consistencia de datos en todos los sitios en tiempo real. (4)

### 1.2. Monitoreo de las soluciones de réplica

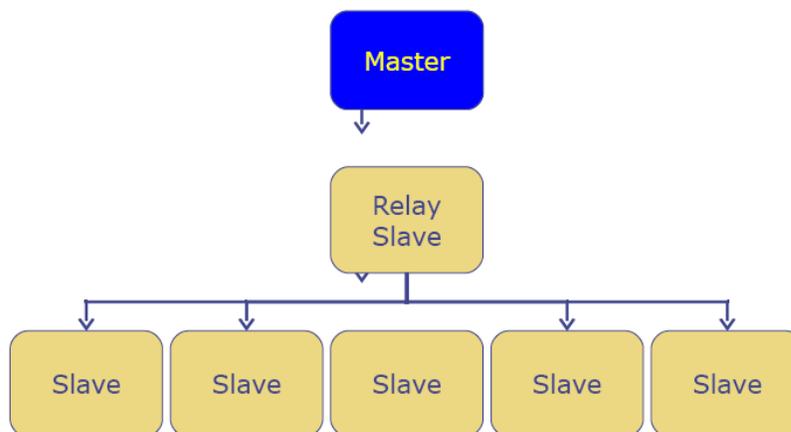
Al realizar una réplica de datos es vital que se efectúe un exhaustivo seguimiento y control en los procesos que se realizan para garantizar su correcto funcionamiento. La monitorización de las soluciones de réplica de datos consiste en realizar tareas de forma continua que ayuden a detectar y resolver problemas en el proceso, garantizando la efectividad de la réplica.

Para poder monitorear una solución de réplica de datos se deben considerar aspectos tales como (5):

- **Latencia:** el tiempo necesario para realizar los cambio entre nodos en una solución de réplica de datos.
- **Rendimiento:** la cantidad de actividad de replicación (medida en comandos suministrados durante un período de tiempo) que un sistema puede mantener en el tiempo.
- **Simultaneidad:** el número de procesos de replicación que pueden operar simultáneamente en un sistema.
- **Duración de sincronización:** el tiempo que tarda en finalizar una sincronización determinada.
- **Utilización de recursos:** recursos de hardware y de red utilizados como resultado del procesamiento de replicación.

## 1.3. La herramienta Slony y sus tareas de monitoreo

**Slony** es un sistema de replicación “maestro a múltiples esclavos”, implementado por la Comunidad de PostgreSQL. Dentro de sus características se encuentra el soporte a la réplica en cascada y permite a un esclavo ser a su vez maestro para otro servidor (Ver Figura 1: Replicación a múltiples esclavos, siendo un esclavo a su vez maestro para otros esclavos). Incluye los tipos de funcionalidades necesarias para replicar bases de datos grandes a un número razonablemente limitado de sistemas esclavos. , en este contexto razonable es un número no superior a 10 servidores. Si el número de servidores crece más allá de 10, el costo de las comunicaciones aumentará y no podrá estar al alcance de la mayoría de los clientes, y las ventajas incrementales de tener servidores múltiples fallarán en ese punto.



*Figura 1: Replicación a múltiples esclavos, siendo un esclavo a su vez maestro para otros esclavos.*

Slony es una excelente elección cuando se requiere realizar una replicación asincrónica de bases de datos, ya que la realiza usando disparadores para determinar las actualizaciones de las tablas, donde un solo origen (maestro) se puede replegar a los suscriptores múltiples (esclavos) incluyendo suscriptores conectados en cascada. Slony realiza una réplica de espejos, exactamente igual al origen de datos, no es posible actualizar los datos a medida que se produce algún cambio en ellos. (6)

Otras de las características de Slony es que las funciones principales de esta herramienta se encuentran en la base de datos implementadas mediante funciones SQL. Todos los datos referentes al clúster están

## INTERFAZ WEB PARA EL MONITOREO DE LA HERRAMIENTA DE RÉPLICA SLONY

almacenadas en 21 tablas que son creadas automáticamente por Slony la primera vez que se inicia el proceso. Para realizar el monitoreo de la misma es necesario consultar las tablas mencionadas para extraer los metadatos de la réplica y consultar los ficheros de logs del sistema donde Slony guarda las trazas de todas sus actividades.

En la Tabla 1: Tablas del clúster de Slony se dan a conocer las tablas del clúster de Slony que contienen información que puede ser particularmente interesante para el monitoreo de la réplica.

*Tabla 1: Tablas del clúster de Slony*

Tablas	Descripción
sl_confirm	Esta tabla contiene confirmación de eventos de replicación. Después de un período de tiempo, Slony elimina viejos eventos confirmados tanto de esta tabla, como la tabla sl_event.
sl_apply_stats	Esta tabla muestra los eventos del nodo local y comprueba la rapidez con que se están confirmando en otros nodos.
sl_event	Esta tabla contiene información sobre los eventos de replicación procesados en el nodo local.
sl_log_1 sl_log_2	Esta tabla contiene datos replicables. En un nodo de origen, esta es la "cola" de los datos que, necesariamente, no se han replicado en todas partes. Al examinar la tabla, puede examinar los detalles de los datos que son replicables.
sl_node	Esta tabla contiene la lista de los nodos del clúster.
sl_path	Esta tabla contiene información de conexión que indica cómo los procesos slon (los demonios slon son los programas que realmente realizan la replicación Slony), estos son para conectar a los nodos remotos, ya sea para acceder a los eventos, o para solicitar datos de replicación.
sl_listen	Esta tabla de configuración indica cómo los nodos escuchan eventos de otros nodos de la red Slony. Normalmente esto se completa automáticamente.
sl_registry	Esta tabla de configuración permite que se puede utilizar para almacenar datos de tiempo de ejecución diversos. Actualmente se utiliza sólo para gestionar el cambio entre las dos tablas de registro.
sl_seqlog	Esta tabla contiene el "último valor" de las secuencias duplicadas.
sl_set	Esta tabla contiene información de definición de conjuntos de replicación.
sl_setsync	Esta tabla contiene información sobre el estado de sincronización de cada conjunto de replicación, incluidos los datos de instantáneas de transacción.
sl_subscribe	Esta tabla indica las suscripciones están en efecto para cada conjunto de replicación.
sl_table	Esta tabla contiene la lista de tablas que se está replicando.

sl_archive_couter	Esta tabla se utiliza para generar el número de archivo de trasvase de registros.
sl_components	Esta tabla se utiliza para monitorizar los diversos componentes slon/Slonik que se están haciendo.
sl_config_lock	Esta tabla existe solamente para evitar la ejecución superposición de procedimientos de cambio de configuración y los posibles estancamientos resultantes.
sl_event_lok	Esta tabla existe solamente para evitar múltiples conexiones desde la creación de nuevos eventos al mismo tiempo y tal vez sacarlos de orden.
sl_log_script	Esta tabla captura consultas de script SQL que se propaguen a los nodos suscriptores.
sl_nodelock	Esta tabla se utiliza para prevenir múltiples instancias slon e identificar los backends para matar <code>enterminateNodeConnections()</code> .

Aun cuando Slony presenta todas las características antes mencionadas, las cuales hacen que sea una herramienta idónea para la replicación en bases de datos de PostgreSQL, presenta el inconveniente de que tanto la instalación, como la configuración y el monitoreo de la herramienta debe realizarse mediante una consola de comandos ya que esta no cuenta con una interfaz visual, esto dificulta sobre todo las tareas de monitoreo, fundamentales para el correcto funcionamiento de la réplica, ya que al realizarse en consola se requiere un conocimiento avanzado de los administradores. Debido a esto se decide desarrollar una interfaz web que permita a los administradores el monitoreo de la réplica sin necesidad de acudir a la consola de comandos.

### 1.4. Tecnologías a emplear para el desarrollo de la solución

Dentro del desarrollo de software y debido a la necesidad de que los proyectos lleguen al éxito y se obtenga un producto de gran valor, es necesario el empleo de metodologías de desarrollo de software, así como de herramientas específicas que satisfagan las necesidades de los clientes a quienes van dirigidos. En este epígrafe se refleja una breve descripción de la metodología y herramientas que serán utilizadas en todo el proceso de desarrollo de la solución.

#### 1.4.1. Metodologías de desarrollo

Las metodologías imponen un proceso disciplinado sobre el desarrollo de software con el fin de hacerlo más predecible y eficiente. Lo hacen desarrollando un proceso detallado con un fuerte énfasis en planificar, inspirado por otras disciplinas de la Ingeniería. A continuación se exponen las metodologías analizadas.

### Proceso Unificado de Rational (RUP)

RUP es una metodología pesada que define quien debe hacer qué, cuándo y cómo. Esta tiene tres características fundamentales: dirigido por casos de uso, centrado en la arquitectura, iterativo e incremental. Es dirigido por casos de uso porque estos se especifican, se diseñan, enlazan los flujos de trabajo y los casos de uso finales son la fuente a partir de la cual los ingenieros de prueba construyen sus casos de prueba. Centrado en la arquitectura por ser esta el eje central del desarrollo de un software y esta permite obtener los casos de uso, correctos y de manera económica. Es iterativo e incremental ya que se divide el producto en mini proyectos donde cada uno representa una iteración que resulta en un incremento.

RUP se compone de 4 fases, estas son:

- ✓ Inicio: durante la fase de inicio las iteraciones ponen mayor énfasis en las actividades de modelado del negocio y requisitos.
- ✓ Elaboración: se determina la arquitectura óptima para el producto.
- ✓ Construcción: se lleva a cabo la construcción del producto por medio de una serie de iteraciones.
- ✓ Transición: en la fase de transición se pretende garantizar que se tiene un producto preparado para su entrega a la comunidad de usuarios.

Además cuenta con nueve flujos de trabajo, de los cuales los seis primeros se conocen como flujos de ingeniería y los tres restantes de apoyo, estos son: Modelación del negocio, Requerimientos, Análisis y Diseño, Implementación, Prueba, Despliegue, Administración del proyecto, Ambiente, Administración de configuración y cambios (7) (8).

**SCRUM**, desarrollada por Ken Schwaber, Jeff Sutherland y Mike Beedle. Es una metodología de trabajo (no de análisis y diseño) usada para la planificación de los proyectos, basada en gestionar proyectos de software. (9) Esta es especialmente indicada para proyectos con un rápido cambio de requisitos. Tiene como característica el desarrollo de software, el cual se realiza mediante iteraciones, denominadas “corridas” o sprints. El resultado de cada sprint es un incremento ejecutable que se muestra al cliente. Exige reuniones a lo largo del proyecto para coordinación e integración. (10) Esta metodología durante su ciclo de vida pasa por determinadas etapas, estas son:

- **Planeamiento:** Se encarga de establecer la visión, definir expectativas y aseguramiento financiero.
- **Montaje:** Tiene como objetivo identificar otros requerimientos y priorizar las tareas para la primera iteración.
- **Desarrollo:** Encargada de implementar un sistema listo para entrega en una serie de iteraciones de treinta días (sprints).
- **Liberación:** El propósito es el despliegue operacional.

SCRUM es una metodología en la que se aplican de manera regular un conjunto de mejores prácticas lo que permite obtener el mejor resultado posible de un proyecto. Permite realizar entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto. (9)

**Programación Extrema o Extreme Programming (XP)** es una metodología de desarrollo ligero (o ágil) basada en una serie de valores y de buenas prácticas que persigue el objetivo de aumentar la productividad a la hora de desarrollar programas. Este modelo de programación se basa en una serie de metodologías de desarrollo de software en la que se da prioridad a los trabajos que dan un resultado directo. Una de sus características principales es que consiste en realizar una programación rápida en la que se tiene como requisito fundamental la participación e integración del usuario final con el equipo de trabajo. El objetivo que se perseguía en el momento de crear esta metodología era la búsqueda de un método que hiciera que los desarrollos fueran más sencillos aplicando el sentido común. (11)

Luego de haber estudiado estas metodologías se decide utilizar XP por:

- Es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software. Se basa en la realimentación continua entre el cliente y el equipo de desarrollo.
- Se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes: Es posible que durante la desarrollo del software se modifiquen los requisitos iniciales del mismo.
- Se utiliza para la realización de proyectos a corto plazo: Para dar solución al problema de la investigación antes planteado se cuenta con tan solo 5 meses.
- Fomenta el desarrollo de equipos pequeños: Para solucionar el problema de la investigación planteado solo se cuenta con una persona para todo el ciclo de desarrollo del software.

- Consiste en una programación rápida o extrema: esta metodología centra la atención del equipo en desarrollar las funcionalidades y no en una profunda documentación del proceso de desarrollo.

### **1.4.2. Lenguaje de Modelado**

UML es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el OMG, siglas de (Object Management Group) Grupo de Gestión de Objetos. Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema de software. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocios y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables. (12)

Con el objetivo de realizar el diseño de los diagramas de la solución y modelar su estructura y comportamiento se empleará este lenguaje de modelado durante el diseño de la misma. La decisión de utilizar UML como notación para el desarrollo del software se debe a que se ha convertido en un estándar que tiene las siguientes características:

- Permite modelar sistemas utilizando técnicas orientadas a objetos.
- Permite especificar todas las decisiones de análisis y diseño, construyéndose así modelos precisos, no ambiguos y completos.
- Puede conectarse con lenguajes de programación (Ingeniería directa e inversa).
- Permite documentar todos los artefactos de un proceso de desarrollo. (13)

### **1.4.3. Herramienta de Modelado**

Visual Paradigm – 8.0 es una herramienta que utiliza como lenguaje de modelado UML, el cual se empleará en el diseño de la solución. Esta herramienta contribuye a una rápida construcción de aplicaciones de mayor calidad y a un menor coste. Además permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. Soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. (14)

Se decide utilizar Visual Paradigm -8.0 porque brinda soporte al lenguaje de modelado UML. Además apoya el trabajo en equipo y como permite la reutilización de modelos de proyectos similares es más cómodo para

realizar el modelado. También reduce el tiempo en implementación y los errores en la codificación al incorporar entre sus ventajas la generación de código abierto.

### **1.4.4. Lenguajes de Programación**

Los lenguajes de programación permiten crear programas y software facilitando la tarea de programación, estos disponen de formas adecuadas que permiten ser leídas y escritas por personas que resultan independientes del modelo de computadora a utilizar. A continuación se exponen los que se estarán utilizando para la implementación de la interfaz web para el monitoreo de la herramienta Slony.

**HTML5 (HypertextMarkupLanguage 5)** es el nombre de la quinta revisión del lenguaje HTML. El cual tiene como característica especial, el resultado de agrupar las especificaciones relacionadas al desarrollo de páginas web: HTML 4, XHTML 1, DOM (DocumentObjectModel) nivel 2, e integrar algunos elementos de CSS nivel 2. (15)

La sintaxis de HTML 5 es compatible con HTML 4 y XHTML 1. Además de usar sintaxis de XML y MathML, el cual es un lenguaje usado para describir notaciones matemáticas, lo que facilita la integración de fórmulas matemáticas a Internet. HTML 5 agrega elementos nuevos al lenguaje, a algunos elementos existentes se le agregan atributos y otros elementos cambian. (15)

**CSS3 (Cascading Style Sheets)** es un lenguaje formal usado para definir la presentación de un documento estructurado escrito en HTML o XML. El W3C (World Wide Web Consortium) es el encargado de formular la especificación de las hojas de estilo que servirá de estándar para los agentes de usuario o navegadores. La idea que se encuentra detrás del desarrollo de CSS es separar la estructura de un documento de su presentación. (16)

**JavaScript** es un lenguaje de programación que se utiliza principalmente para crear páginas web dinámicas, que son aquellas que incorporan efectos como texto, animaciones, acciones que se activan al pulsar botones y ventanas con mensajes de aviso al usuario. Este lenguaje es interpretado, por lo que no es necesario compilar los programas para ejecutarlos. Se puede probar directamente en cualquier navegador sin necesidad de procesos intermedios. Hay que destacar que a pesar de su nombre, JavaScript no guarda ninguna relación directa con el lenguaje de programación Java. (17)

**AJAX** es una técnica que permite mejorar completamente la interacción del usuario con la aplicación, evitando las recargas constantes de la página, ya que el intercambio de información con el servidor se produce en un segundo plano. Ajax no es una tecnología en sí mismo, en realidad se trata de varias tecnologías independiente que se unen. (18)

Las tecnologías que forman AJAX son:

- XHTML y CSS, para crear una presentación basada en estándares.
- DOM, para la interacción y manipulación dinámica de la presentación.
- XML, XSLT y JSON, para el intercambio y la manipulación de información.
- XMLHttpRequest, para el intercambio asíncrono de información.
- JavaScript, para unir todas las demás tecnologías.

**PHP5 (Pre-procesador de hipertexto)** es un acrónimo de “Hypertext Preprocessor”, aunque los orígenes del nombre se refieren al término Personal Home Page. Es un lenguaje del lado del servidor, de código abierto, lo cual implica que existe un grupo de desarrolladores que contribuyen a la elaboración y fortalecimiento del lenguaje; su utilización no requiere pagos por derecho de autor. (19)

PHP dispone de una gran cantidad de características que lo convierten en la herramienta ideal para la creación de páginas web dinámicas. (19) Algunas de ellas:

- Soporte para una gran cantidad de bases de datos: MySQL, PostgreSQL, Oracle, MS SQL Server, SybaseSQL, Informix, entre otras.
- Integración con varias bibliotecas externas, permite generar documentos en PDF hasta analizar código XML.
- El código se pone al día continuamente con mejoras y extensiones de lenguaje para ampliar las capacidades de PHP.

Empleando los lenguajes de programación PHP, JavaScript, CSS y HTML, se proveerá una vista amigable y dinámica. Además el uso de JQuery hará a la interfaz compatible con distintos navegadores web, por su parte Ajax elimina la recarga constante de páginas mediante la creación de un elemento intermedio entre el usuario y el servidor.

### 1.4.5. *Entorno de Desarrollo Integrado (IDE)*

NetBeans IDE es una herramienta que utilizan los programadores para escribir, compilar, depurar y ejecutar programas. Escrito en Java aunque puede ser utilizado en cualquier otro lenguaje de programación. Posee un número importante de módulos para extenderlo lo que posibilita ampliar y facilitar su utilización. Es un producto libre y gratuito sin restricciones de uso. (20)

Principales características:

- Creación de Proyectos PHP.
- Integración con Symfony y ZenFramework.
- Editor de Código Fuente.
- Depuración de PHP.
- Integración con Sistemas de Control de Versiones.

Por lo tanto para el desarrollo de la interfaz web para el monitoreo de la herramienta de réplica Slony se utilizará NetBeans IDE por sus características y las habilidades que brinda el mismo para el trabajo con el lenguaje PHP. Además es un producto sin restricciones.

### 1.4.6. *Plataforma de desarrollo*

Una plataforma de desarrollo o marco de trabajo (framework) es un producto que se utiliza como base para la programación avanzada de aplicaciones que aporta una serie de funciones o códigos para realizar tareas habituales. Lo conforman un conjunto de librerías de código que contienen procesos o rutinas ya listos para usar. Los programadores lo utilizan para no tener que desarrollar manualmente las tareas básicas porque ya incorpora implementaciones que están probadas, funcionan y no se necesitan volver a programar. Además permiten desarrollar sistemas en menos tiempo, más robustos y con menos cantidad de errores.

**JQuery:** es un framework para el lenguaje JavaScript que implementa una serie de clases (de programación orientada a objetos) que facilitan la implementación del navegador que utilice el usuario, ya que funcionan de forma exacta en todas las plataformas más habituales. Este framework ofrece una infraestructura que permite crear aplicaciones complejas del lado del cliente con mayor facilidad. Por ejemplo, con jQuery se

pueden realizar proyectos que incluyan la creación de interfaces de usuario y efectos dinámicos. También incluye librerías que permiten al programador obtener productos en menos tiempo y libre de errores. Además es un framework desarrollado bajo las licencias de software libre y esta característica es vital para obtener la solución que propone este trabajo de diploma. (21)

JQuery es un producto con una aceptación muy buena por parte de los programadores y un grado de penetración en el mercado muy amplio, lo que hace suponer que es una de las mejores opciones. Además, es un producto serio, estable, bien documentado y con un gran equipo de desarrolladores a cargo de la mejora y actualización del framework. (21)

### **Conclusiones del capítulo**

En el capítulo se analizaron los tipos, entornos y modelos de distribución de las réplicas de datos. Se realizó también un estudio de la herramienta Slony, analizándose las tablas del clúster de este evidenciándose además la necesidad de desarrollar una interfaz web para el monitoreo de dicha herramienta. Con el fin de guiar el desarrollo de la investigación se decide emplear XP como metodología de desarrollo, usar UML como lenguaje de modelado y como herramienta CASE el Visual Paradigm v8.0. Los lenguajes de programación PHP5, HTML5, CSS3, Java Script ya estos permitirán brindarle al usuario una vista amigable y dinámica; la técnica Ajax que eliminará la recarga constante de páginas mediante la creación de un elemento intermedio entre el usuario y el servidor, como framework jQuery v1.3.4 que hará a la interfaz compatible con distintos navegadores web y el IDE de desarrollo NetBeans en su v 8.0.

## **CAPÍTULO: 2 DISEÑO E IMPLEMENTACIÓN DE LA SOLUCIÓN**

En el desarrollo o ciclo de vida de un sistema informático, el diseño del sistema constituye un elemento fundamental. En este se modela el sistema y se encuentra la forma para que soporte todos los requisitos, incluyendo los requisitos no funcionales, se asienta en el núcleo técnico del proceso de ingeniería del software y se aplica independientemente del paradigma de desarrollo utilizado. En la elaboración de la solución es preciso tener presente ciertos estándares, estilos y patrones que son muy útiles en la obtención de un diseño de software robusto. (22)

En este capítulo se presenta el diseño propuesto para la solución a desarrollar. Se presenta un modelo de dominio con el fin de proporcionar un mejor entendimiento de los principales conceptos del negocio, una descripción de las historias de usuario definidas, los requisitos tanto funcionales como no funcionales a tener en cuenta en su desarrollo, agrupados en la lista de reserva del producto, un plan de iteraciones que contempla el tiempo de implementación del sistema, las tarjetas CRC para describir las clases que se van a utilizar, así como la arquitectura base de la aplicación, dentro de esta los patrones arquitectónicos y de diseño. Se exponen las tareas de ingeniería definidas para cada historia de usuario con vistas a su implementación, además también se presenta el estándar de codificación utilizado.

### **2.1. Modelo del Domino**

A pesar de que la metodología de desarrollo de software XP no precisa una técnica específica para definir el negocio, con el fin de proporcionar un mejor entendimiento de los principales conceptos que se manejan en este, se decide realizar un modelo de dominio, el cual se describe mediante diagramas de UML, específicamente mediante diagramas de clases.

El modelo de dominio puede ser tomado como el punto de partida para el diseño del sistema. Este permitirá mostrar de manera visual los principales conceptos que se manejan, permitiendo a los usuarios y desarrolladores utilizar un vocabulario común para poder entender el contexto en que se desarrolla el sistema. Además contribuirá a identificar personas, eventos, transacciones y objetos involucrados en el negocio. (23)

El modelo de dominio que será tomado como punto de partida para el diseño del sistema se muestra en la Figura 2: Modelo de dominio.

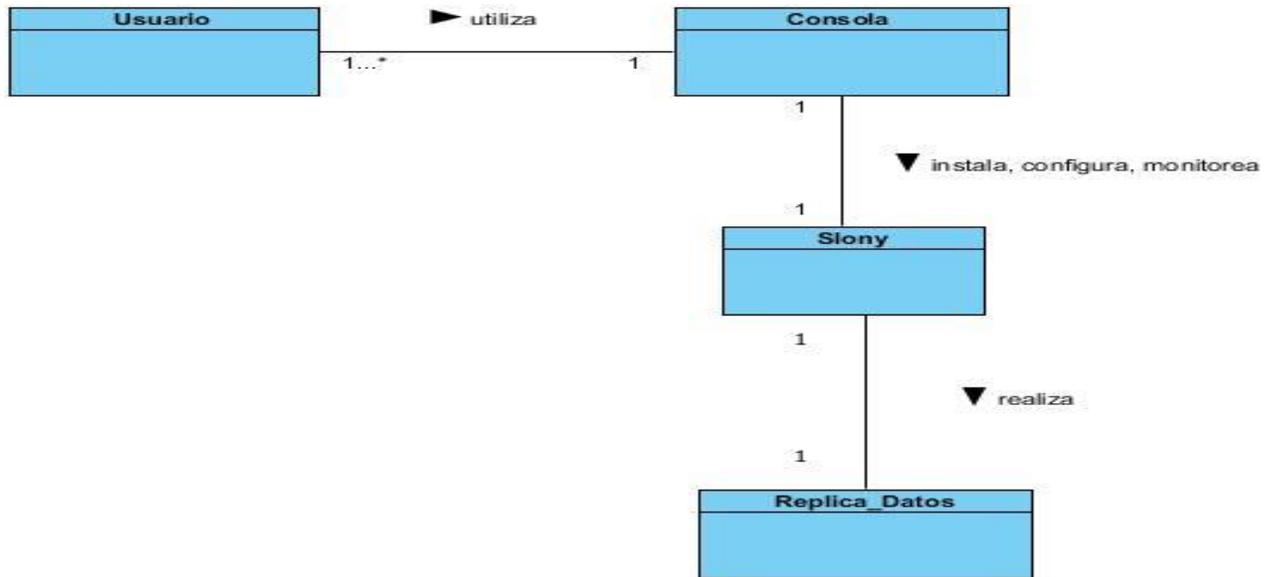


Figura 2: Modelo de dominio

Los conceptos involucrados en el dominio son los descritos en la siguiente tabla:

Tabla 2: Descripción de los conceptos del diagrama de dominio

Concepto	Descripción
Usuario	Persona encargada de realizar la monitorización a la solución de réplica de datos.
Consola	Sistema que permitirá la interacción del usuario con herramienta Slony para realizar la configuración de la solución de réplica de datos.
Réplica de Datos	Proceso de copiar y mantener objetos de las bases de datos en múltiples ubicaciones.
Slony	Herramienta de réplica de datos que permite la gestión de la solución de réplica de datos.

## 2.2. Descripción de la solución

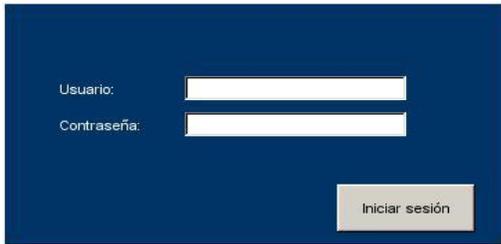
Se propone la implementación de una interfaz web para la monitorización de la herramienta de réplica Slony que permitirá al administrador una vez autenticado monitorear las tablas del clúster de Slony (Ver Tabla 1: Tablas del clúster de Slony), además los nodos, los sets y los logs de la réplica. El administrador podrá también realizar el análisis del comportamiento de la réplica mediante la consulta de tablas y gráficos que podrá elaborar definiendo un conjunto de parámetros según sus intereses.

## 2.2.1. Historias de usuario

Las historias de usuario representan una breve descripción del comportamiento del sistema empleando terminología del cliente sin lenguaje técnico. Se realiza una por cada característica principal del sistema, empleándose para hacer estimaciones de tiempo y para el plan de iteraciones. Con las historias de usuario se evita la creación de un gran documento de requisitos y las mismas presiden la creación de las pruebas de aceptación. (24)

En las tablas siguientes se detallan las dieciséis historias de usuario definidas para el desarrollo de la propuesta de solución de la investigación.

*Tabla 3: Historia de Usuario Autenticar usuario.*

Historia de Usuario	
<b>Número:</b> 1	<b>Nombre de la Historia de Usuario:</b> Autenticar usuario
<b>Cantidad de modificaciones a la Historia de Usuario:</b> Ninguna	
<b>Usuario:</b> Nurisely Ramos López	<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Muy Alta	<b>Puntos estimados:</b> 1 semana
<b>Riesgo en desarrollo:</b> Alto	<b>Puntos reales:</b> 1 semana
<b>Descripción:</b> El sistema debe permitir al usuario (administrador) autenticarse mediante un usuario y contraseña en la interfaz web para acceder luego a sus funcionalidades.	
<b>Observaciones:</b>	
<b>Prototipo de Interfaz:</b>	
	

*Tabla 4: Cambiar contraseña*

Historia de Usuario	
<b>Número:</b> 2	<b>Nombre de la Historia de Usuario:</b> Cambiar contraseña
<b>Cantidad de modificaciones a la Historia de Usuario:</b> Ninguna	
<b>Usuario:</b> Nurisely Ramos López	<b>Iteración asignada:</b> 1

## INTERFAZ WEB PARA EL MONITOREO DE LA HERRAMIENTA DE RÉPLICA SLONY

<b>Prioridad en negocio:</b> Muy Alta	<b>Puntos estimados:</b> 1 semana
<b>Riesgo en desarrollo:</b> Alto	<b>Puntos reales:</b> 1 semana
<b>Descripción:</b> El sistema debe permitir al usuario (administrador) luego de autenticarse en la interfaz cambiar su contraseña. El sistema debe actualizar los datos en el archivo de configuración de la aplicación.	
<b>Observaciones:</b>	
<b>Prototipo de Interfaz:</b>	



*Tabla 5: Configurar la conexión*

Historia de Usuario	
<b>Número:</b> 3	<b>Nombre de la Historia de Usuario:</b> Configurar la conexión
<b>Cantidad de modificaciones a la Historia de Usuario:</b> Ninguna	
<b>Usuario:</b> Nurisely Ramos López	<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Muy Alta	<b>Puntos estimados:</b> 1 semana
<b>Riesgo en desarrollo:</b> Alto	<b>Puntos reales:</b> 1 semana
<b>Descripción:</b> El sistema debe permitir al usuario (administrador) luego de autenticarse en la interfaz configurar la conexión a la base de datos, específicamente los parámetros ip y nombre de la base de datos, puerto, usuario, contraseña y clúster.	
<b>Observaciones:</b>	
<b>Prototipo de Interfaz:</b>	



# INTERFAZ WEB PARA EL MONITOREO DE LA HERRAMIENTA DE RÉPLICA SLONY

Tabla 6: Historia de Usuario Visualizar listado de las tablas.

Historia de Usuario	
<b>Número:</b> 4	<b>Nombre de la Historia de Usuario:</b> Visualizar listado de las tablas del clúster de Slony.
<b>Cantidad de modificaciones a la Historia de Usuario:</b> Ninguna	
<b>Usuario:</b> Nurisely Ramos López	<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Muy Alta	<b>Puntos estimados:</b> 1 semana
<b>Riesgo en desarrollo:</b> Alto	<b>Puntos reales:</b> 1 semana
<b>Descripción:</b> El sistema debe permitir al usuario (administrador) luego de autenticarse en la interfaz obtener las tablas del clúster de Slony.	
<b>Observaciones:</b>	
<b>Prototipo de Interfaz:</b>	



Tabla 7: Historia de Usuario Visualizar los datos de cada tabla según se van seleccionando por el usuario.

Historia de Usuario	
<b>Número:</b> 5	<b>Nombre de la Historia de Usuario:</b> Visualizar los datos de cada tabla según se van seleccionando por el usuario.
<b>Cantidad de modificaciones a la Historia de Usuario:</b> Ninguna	
<b>Usuario:</b> Nurisely Ramos López	<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Muy Alta	<b>Puntos estimados:</b> 1 semana
<b>Riesgo en desarrollo:</b> Alto	<b>Puntos reales:</b> 1 semana
<b>Descripción:</b> El sistema debe permitir al usuario (administrador) luego de autenticarse en la interfaz acceder a la información de las tablas del clúster de Slony.	
<b>Observaciones:</b>	
<b>Prototipo de Interfaz:</b>	

# INTERFAZ WEB PARA EL MONITOREO DE LA HERRAMIENTA DE RÉPLICA SLONY

Herramienta de Monitoreo Slony

Tablas | **Nodos** | Logs | Sets | Réplica | Sistema

No.	re origin	re target	re database	re sequence	re type	re status	re status	re status	re status	re status	re status	re status
1	1	500000001	2015-05-28 21:12:29.353 504.94	204009.2040 80	STANAL_PDO DC	1						
2	1	500000002	2015-05-28 21:12:31.133 504.94	204009.2040 80	STANAL_SET	1	1	setC				
3	1	500000003	2015-05-28 21:12:32.913 504.94	204009.2040 80	STANAL	1						
4	1	500000004	2015-05-28 21:12:34.693 504.94	204011.2040 80	SET_AEO_1 ABLE	1	1	public.catalog	catalogue_1	table	catalogue	
5	1	500000005	2015-05-28 21:12:36.473 504.94	204012.2040 80	STANAL	1						
6	1	500000006	2015-05-28 21:12:38.253 504.94	204013.2040 80	SET_AEO_1 ABLE	1	2	public.catalog	pk_primary	table	pk_1	

Tabla 8: Historia de Usuario Visualizar listado de los nodos.

Historia de Usuario	
<b>Número:</b> 6	<b>Nombre de la Historia de Usuario:</b> Visualizar los nodos.
<b>Cantidad de modificaciones a la Historia de Usuario:</b> Ninguna	
<b>Usuario:</b> Nurisely Ramos López	<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Muy Alta	<b>Puntos estimados:</b> 1 semana
<b>Riesgo en desarrollo:</b> Alto	<b>Puntos reales:</b> 1 semana
<b>Descripción:</b> El sistema debe permitir al usuario (administrador) luego de autenticarse en la interfaz visualizar los nodos que intervienen en la réplica.	
<b>Observaciones:</b>	
<b>Prototipo de Interfaz:</b>	

Herramienta de Monitoreo Slony

Tablas | **Nodos** | Logs | Sets | Réplica | Sistema

ID del Nodo	3	ID del Nodo	1
Acción:	id	Acción:	id
Nombre:	Nodo DelStore2	Nombre:	Nodo Maestro
Base de Datos:	DelStore2	Base de Datos:	maestro
IP del Nodo:	127.0.0.1	IP del Nodo:	127.0.0.1
Usuario de la Base de Datos:	postgres	Usuario de la Base de Datos:	postgres

Tabla 9: Historia de Usuario Visualizar los sets de réplica.

Historia de Usuario	
<b>Número:</b> 7	<b>Nombre de la Historia de Usuario:</b> Visualizar los sets de réplica.
<b>Cantidad de modificaciones a la Historia de Usuario:</b> Ninguna	
<b>Usuario:</b> Nurisely Ramos López	<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Muy Alta	<b>Puntos estimados:</b> 1 semana

## INTERFAZ WEB PARA EL MONITOREO DE LA HERRAMIENTA DE RÉPLICA SLONY

<b>Riesgo en desarrollo:</b> Alto	<b>Puntos reales:</b> 1 semana
<b>Descripción:</b> El sistema debe permitir al usuario (administrador) luego de autenticarse en la interfaz visualizar los set que intervienen en la réplica.	
<b>Observaciones:</b>	
<b>Prototipo de Interfaz:</b>	
	

*Tabla 10: Historia de Usuario Visualizar los log de la herramienta Slony.*

Historia de Usuario	
<b>Número:</b> 8	<b>Nombre de la Historia de Usuario:</b> Visualizar los logs de la herramienta Slony.
<b>Cantidad de modificaciones a la Historia de Usuario:</b> Ninguna	
<b>Usuario:</b> Nurisely Ramos López	<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Muy Alta	<b>Puntos estimados:</b> 1 semana
<b>Riesgo en desarrollo:</b> Alto	<b>Puntos reales:</b> 1 semana
<b>Descripción:</b> El sistema debe permitir al usuario (administrador) luego de autenticarse en la interfaz visualizar los logs de la herramienta Slony.	
<b>Observaciones:</b>	
<b>Prototipo de Interfaz:</b>	
	

## INTERFAZ WEB PARA EL MONITOREO DE LA HERRAMIENTA DE RÉPLICA SLONY

Tabla 11: Historia de Usuario Visualizar los logs del servidor PostgreSQL.

Historia de Usuario	
<b>Número:</b> 9	<b>Nombre de la Historia de Usuario:</b> Visualizar los logs del servidor PostgreSQL.
<b>Cantidad de modificaciones a la Historia de Usuario:</b> Ninguna	
<b>Usuario:</b> Nurisely Ramos López	<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Muy Alta	<b>Puntos estimados:</b> 1 semana
<b>Riesgo en desarrollo:</b> Alto	<b>Puntos reales:</b> 1 semana
<b>Descripción:</b> El sistema debe permitir al usuario (administrador) luego de autenticarse en la interfaz visualizar los logs del servidor de PostgreSQL.	
<b>Observaciones:</b>	
<b>Prototipo de Interfaz:</b>	
	

Tabla 12: Historia de Usuario Mostrar gráficamente el consumo de RAM del proceso Slon.

Historia de Usuario	
<b>Número:</b> 10	<b>Nombre de la Historia de Usuario:</b> Mostrar gráficamente el consumo de RAM del proceso Slon.
<b>Cantidad de modificaciones a la Historia de Usuario:</b> Ninguna	
<b>Usuario:</b> Nurisely Ramos López	<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Muy Alta	<b>Puntos estimados:</b> 1 semana
<b>Riesgo en desarrollo:</b> Alto	<b>Puntos reales:</b> 1 semana
<b>Descripción:</b> El sistema debe permitir al usuario (administrador) luego de autenticarse en la interfaz visualizar el consumo de RAM del proceso de Slon.	
<b>Observaciones:</b>	
<b>Prototipo de Interfaz:</b>	

# INTERFAZ WEB PARA EL MONITOREO DE LA HERRAMIENTA DE RÉPLICA SLONY



Tabla 13: Historia de Usuario Mostrar gráficamente el consumo de RAM del proceso PostgreSQL.

Historia de Usuario	
<b>Número:</b> 11	<b>Nombre de la Historia de Usuario:</b> Mostrar gráficamente el consumo de RAM del proceso PostgreSQL.
<b>Cantidad de modificaciones a la Historia de Usuario:</b> Ninguna	
<b>Usuario:</b> Nurisely Ramos López	<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Muy Alta	<b>Puntos estimados:</b> 1 semana
<b>Riesgo en desarrollo:</b> Alto	<b>Puntos reales:</b> 1 semana
<b>Descripción:</b> El sistema debe permitir al usuario (administrador) luego de autenticarse en la interfaz visualizar el consumo de RAM del proceso de PostgreSQL.	
<b>Observaciones:</b>	
<b>Prototipo de Interfaz:</b>	
<p>This is a duplicate of the screenshot shown in Table 13, illustrating the RAM usage monitoring interface for Slony and PostgreSQL.</p>	

Tabla 14: Historia de Usuario Visualizar los logs por tipos según las categorías en las que se divide.

Historia de Usuario	
<b>Número:</b> 12	<b>Nombre de la Historia de Usuario:</b> Visualizar los logs por tipos según las categorías en las que se divide.
<b>Cantidad de modificaciones a la Historia de Usuario:</b> Ninguna	

## INTERFAZ WEB PARA EL MONITOREO DE LA HERRAMIENTA DE RÉPLICA SLONY

<b>Usuario:</b> Nurisely Ramos López	<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Muy Alta	<b>Puntos estimados:</b> 1 semana
<b>Riesgo en desarrollo:</b> Alto	<b>Puntos reales:</b> 1 semana
<b>Descripción:</b> El sistema debe permitir al usuario (administrador) luego de autenticarse en la interfaz visualizar los logs por tipos: CDT LOG, CDT ERROR, CDT ESTATEMENT, CDT FATAL, CDT DETAIL, CTD CONFIG, CTD QUERY, CTD CONTEXT, CTD HINT, CTD INFO.	
<b>Observaciones:</b>	
<b>Prototipo de Interfaz:</b>	
	

*Tabla 15: Historia de Usuario Mostrar gráficamente el histórico de las operaciones realizadas dado un conjunto de parámetros.*

Historia de Usuario	
<b>Número:</b> 13	<b>Nombre de la Historia de Usuario:</b> Mostrar gráficamente el histórico de las operaciones realizadas dado un conjunto de parámetros.
<b>Cantidad de modificaciones a la Historia de Usuario:</b> Ninguna	
<b>Usuario:</b> Nurisely Ramos López	<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Muy Alta	<b>Puntos estimados:</b> 1 semana
<b>Riesgo en desarrollo:</b> Alto	<b>Puntos reales:</b> 1 semana
<b>Descripción:</b> El sistema debe permitir al usuario (administrador) luego de autenticarse en la interfaz visualizar gráficamente la cantidad de operaciones realizadas dado un parámetro (Actualizar, Insertar o Eliminar).	
<b>Observaciones:</b>	

**Prototipo de Interfaz:**



*Tabla 16: Historia de Usuario Mostrar las tablas con los resultados de la búsqueda en el histórico de las operaciones realizadas dado un conjunto de parámetros.*

### Historia de Usuario

**Número:** 14      **Nombre de la Historia de Usuario:** Mostrar las tablas con los resultados de la búsqueda en el histórico de las operaciones realizadas dado un conjunto de parámetros.

**Cantidad de modificaciones a la Historia de Usuario:** Ninguna

**Usuario:** Nurisely Ramos López      **Iteración asignada:** 1

**Prioridad en negocio:** Muy Alta      **Puntos estimados:** 1 semana

**Riesgo en desarrollo:** Alto      **Puntos reales:** 1 semana

**Descripción:** El sistema debe permitir al usuario (administrador) luego de autenticarse en la interfaz visualizar en una tabla la cantidad de operaciones realizadas dado parámetro (Actualizar, Insertar o Eliminar).

**Observaciones:**

**Prototipo de Interfaz:**



*Tabla 17: Historia de Usuario Extraer de la tabla sl\_path la información de los nodos.*

### Historia de Usuario

## INTERFAZ WEB PARA EL MONITOREO DE LA HERRAMIENTA DE RÉPLICA SLONY

<b>Número:</b> 15	<b>Nombre de la Historia de Usuario:</b> Extraer de la tabla sl_path la información de los nodos.
<b>Cantidad de modificaciones a la Historia de Usuario:</b> Ninguna	
<b>Usuario:</b> Nurisely Ramos López	<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Alta	<b>Puntos estimados:</b> 1 semana
<b>Riesgo en desarrollo:</b> Alto	<b>Puntos reales:</b> 1 semana
<b>Descripción:</b> El sistema debe permitir al usuario (administrador) luego de autenticarse en la interfaz extraer de la tabla sl_path todos los datos de los nodos.	
<b>Observaciones:</b>	
<b>Prototipo de Interfaz:</b>	

*Tabla 18: Historia de Usuario Visualizar versión de Slony.*

Historia de Usuario							
<b>Número:</b> 16	<b>Nombre de la Historia de Usuario:</b> Visualizar versión de Slony.						
<b>Cantidad de modificaciones a la Historia de Usuario:</b> Ninguna							
<b>Usuario:</b> Nurisely Ramos López	<b>Iteración asignada:</b> 1						
<b>Prioridad en negocio:</b> Media	<b>Puntos estimados:</b> 1 semana						
<b>Riesgo en desarrollo:</b> Bajo	<b>Puntos reales:</b> 1 semana						
<b>Descripción:</b> El sistema debe permitir al usuario (administrador) luego de autenticarse en la interfaz visualizar versión que se está utilizando.							
<b>Observaciones:</b>							
<b>Prototipo de Interfaz:</b>							
 <p>The screenshot shows a web application titled 'Herramienta de Monitoreo Slony'. It features a navigation menu with tabs for 'Inicio', 'Tablas', 'Nodos', 'Logs', 'Sets', 'Replica', and 'Sistema'. The 'Sistema' tab is currently selected. Below the menu, there is a table with three columns: 'Slony', 'Postgresql', and 'Ubuntu'. The table contains one row of data: '2.1.4', '9.3', and '14.04.2 LTS'.</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>Slony</th> <th>Postgresql</th> <th>Ubuntu</th> </tr> </thead> <tbody> <tr> <td>2.1.4</td> <td>9.3</td> <td>14.04.2 LTS</td> </tr> </tbody> </table>		Slony	Postgresql	Ubuntu	2.1.4	9.3	14.04.2 LTS
Slony	Postgresql	Ubuntu					
2.1.4	9.3	14.04.2 LTS					

**2.2.2. Lista de reserva del producto**

Una vez definidos los requisitos del software estos son agrupados en la lista de reserva del producto. Este artefacto permite una mayor organización logrando un mejor entendimiento por parte del equipo de desarrollo, ya que los requisitos funcionales son organizados por la prioridad que tenga cada uno y los no funcionales por su categoría. En la siguiente tabla se muestran los requisitos definidos para la implementación de la propuesta de solución, derivados de las historias de usuario acordadas con el cliente.

A continuación se muestra la lista de reserva del producto en el producto perteneciente a la solución:

*Tabla 19: Lista de reserva del producto*

NO	Descripción	Estimación	Estimado por
<b>Prioridad: MUY ALTA</b>			
1	Autenticar usuario.	1	Analista
2	Cambiar contraseña.	1	Analista
3	Configurar la conexión.	1	Analista
4	Visualizar listado de las tablas del clúster de Slony.	1	Analista
5	Visualizar los datos de cada tabla según se van seleccionando por el usuario.	1	Analista
6	Visualizar los nodos.	1	Analista
7	Visualizar los set de réplicas.	1	Analista
8	Visualizar los logs de la herramienta Slony.	1	Analista
9	Visualizar los logs del servidor PostgreSQL.	1	Analista
10	Mostrar gráficamente el consumo de RAM del proceso Slon.	1	Analista
11	Mostrar gráficamente el consumo de RAM del proceso PostgreSQL.	1	Analista
12	Visualizar los logs por tipos según las categorías en las que se divide.	1	Analista
13	Mostrar gráficamente el histórico de las operaciones realizadas dado un	1	Analista

## INTERFAZ WEB PARA EL MONITOREO DE LA HERRAMIENTA DE RÉPLICA SLONY

	conjunto de parámetros.		
<b>14</b>	Mostrar las tablas con los resultados de la búsqueda en el histórico de las operaciones realizadas dado un conjunto de parámetros.	1	Analista
<b>Prioridad: ALTA</b>			
<b>15</b>	Extraer de la tabla sl_path la información de los nodos.	1	Analista
<b>Prioridad: MEDIA</b>			
<b>16</b>	Visualizar versión de Slony.	1	Analista
<b>Requisitos no funcionales</b>			
<b>Requisitos de Software</b>			
<b>1</b>	<p>Sistema Operativo: GNU/Linux preferentemente Ubuntu GNU/Linux 14.04.</p> <p>Se requiere tener instalado:</p> <ul style="list-style-type: none"> <li>- PHP5 v 5.5.</li> <li>- Apache v 2.4.</li> <li>- Mozilla Firefox v 17 o superior</li> </ul> <p>Los nombres de los paquetes coinciden con los pertenecientes a Ubuntu 14.04, estos pueden variar según el sistema operativo sobre el que se trabaje y la versión.</p>		
<b>Requisitos de Hardware:</b>			
<b>2</b>	<p>El ordenador debe tener como mínimo:</p> <ul style="list-style-type: none"> <li>- Microprocesador con una velocidad de 1.10 GHZ</li> <li>- Capacidad libre en disco duro de 100 MB</li> <li>- Memoria RAM <math>\geq</math> 256 MB</li> </ul>		
<b>Requisitos de Soporte</b>			
<b>3</b>	La interfaz contará antes de su puesta en marcha con un período de pruebas.		

### 2.2.3. Plan de iteraciones

Cuando se concluye el proceso de definir las Historias de usuario en la fase de planificación para guiar la etapa de implementación de la solución se procede a desarrollar el plan de iteraciones. Este plan se

## INTERFAZ WEB PARA EL MONITOREO DE LA HERRAMIENTA DE RÉPLICA SLONY

construye en la fase de Iteración y consiste en un documento que muestra la cantidad de Historias de Usuario definidas por el cliente en cada iteración y el tiempo estimado para completar las iteraciones, con el objetivo de planificar el período de implementación del sistema.

En el Plan de Iteraciones se puede observar las iteraciones acordadas entre el cliente y el desarrollador, las Historias de Usuario a desarrollar en cada una de ellas y el tiempo propuesto para la implementación de las mismas. El desarrollo de la interfaz tendrá una duración de 16 semanas en las cual se realizarán un total de 6 iteraciones.

Tabla20: Plan de iteraciones

Iteración	Descripción de la iteración	Orden de la HU a implementar	Duración total
1	Se desarrollan 3 Historias de Usuario de prioridad Muy Alta que resultan de vital importancia para el resto de la aplicación.	1, 2, 3	3 semanas
2	Se desarrollan 3 Historias de Usuario de prioridad Muy Alta que resultan de vital importancia para el resto de la aplicación.	4, 5, 6	3 semanas
3	Se desarrollan 3 Historias de Usuario de prioridad Muy Alta que resultan de vital importancia para el resto de la aplicación.	7, 8, 9	3 semanas
4	Se desarrollan 3 Historias de Usuario de prioridad Muy Alta que resultan de vital importancia para el resto de la aplicación.	10, 11, 12	3 semanas
5	Se desarrollan 2 Historias de Usuario de prioridad Muy Alta que resultan de vital importancia para el resto de la aplicación.	13, 14	2 semana
6	Se desarrolla 1 Historia de Usuario de prioridad Alta y 1 Historia de Usuario de prioridad Media.	15 y 16	2 semanas

### 2.2.4. Tarjetas CRC

Para la implementación de las 16 historias de usuario definidas para dar cumplimiento a los requisitos del cliente, se identificaron 2 tarjetas CRC, las que se muestran en las tablas siguientes donde en la parte

superior de la tarjeta se especifica la clase a la que pertenece cada objeto, en la columna izquierda las responsabilidades u objetivos que debe cumplir el objeto y a la derecha, las clases que colaboran.

*Tabla 21: Tarjeta CRC Master*

Tarjeta CRC	
<b>Clase:</b> Master	
Responsabilidades	Colaboraciones
<ul style="list-style-type: none"> <li>• Monitorear las tablas.</li> <li>• Mostrar la información de cada tabla.</li> <li>• Monitorear los datos de los sets.</li> <li>• Monitorear los logs de la herramienta Slony.</li> <li>• Monitorear los logs del servidor PostgreSQL.</li> <li>• Mostrar gráficamente el consumo de RAM del proceso Slon</li> <li>• Monitorear los datos de los nodos.</li> <li>• Monitorear los logs por tipos según las categorías en las que se divide.</li> <li>• Mostrar las tablas con los resultados de la búsqueda en el histórico de las operaciones realizadas dado un conjunto de parámetros.</li> <li>• </li> </ul>	<ul style="list-style-type: none"> <li>• Conexión_DB</li> </ul>

*Tabla 22: Tarjeta CRC Conexión*

Tarjeta CRC	
<b>Clase:</b> Conexión	
Responsabilidades	Colaboraciones
<ul style="list-style-type: none"> <li>• Conecta la base de datos.</li> <li>• Retornar todas las tablas de determinada base de datos.</li> <li>• Retorna los datos de determinada tabla seleccionada por el usuario.</li> <li>• Muestra la versión de Slony.</li> <li>• Desconecta la base de datos.</li> </ul>	

## 2.3. Arquitectura base de la aplicación

La arquitectura de Software es la forma de estructurar el sistema teniendo en cuenta un conjunto de patrones que proporcionan una guía a través del ciclo de desarrollo del software, permitiendo a analistas, diseñadores y programadores seguir una línea de trabajo en común para poder alcanzar los objetivos del sistema. De acuerdo con Roger S. Pressman, la arquitectura de software no es otra cosa que “(...) una descripción de los subsistemas y los componentes de un sistema informático y las relaciones entre ellos. De igual manera, la arquitectura de software de tres niveles, incluye todos estos aspectos, y además, brinda mejores opciones para proyectos informáticos de gran alcance y complejidad”.

### 2.3.1. Patrones de arquitectura

Para la creación del sistema se propone una arquitectura Modelo - Vista - Controlador (MVC) (25), para una mejor organización y manejo de las vistas. Este patrón separa los datos de una aplicación, la interfaz de usuario y la lógica de control en tres componentes distintos. Se puede observar frecuentemente en aplicaciones web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página; el modelo es la representación específica de la información con la cual el sistema opera y la lógica de negocio; y el controlador es el responsable de recibir los eventos de entrada desde la vista. Se fundamenta en separar cada uno de estos tres componentes en elementos independientes y claramente diferenciados. De esta manera, el diseño gana en cuanto a reutilización, ya que el desacople de elementos garantiza poder modificarlos con mayor facilidad y de forma independiente.

La Figura 4: Diagrama de clases (Patrón MVC) muestra el diagrama de clases correspondiente a la interfaz web:

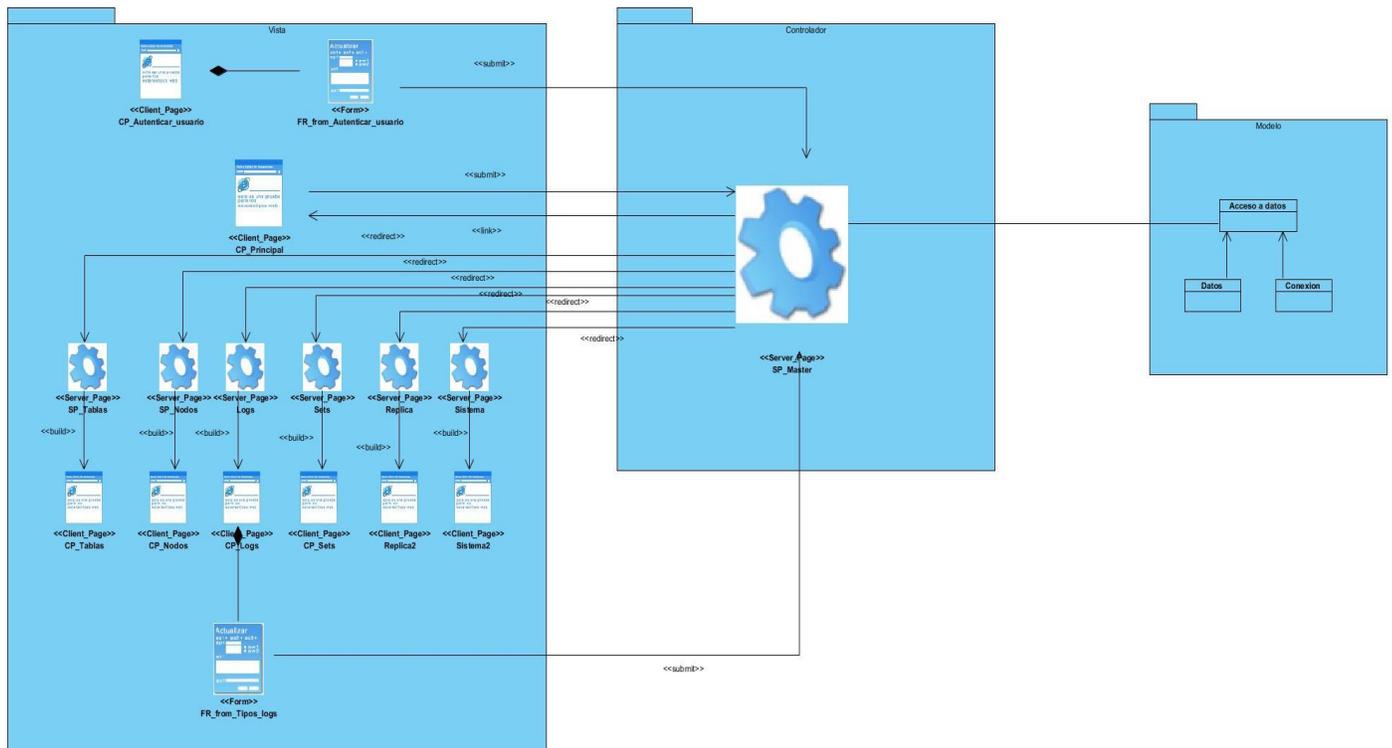


Figura 3: Diagrama de clases (Patrón MVC)

### 2.3.2. Patrones de diseño

Los patrones de diseño (26) constituyen el esqueleto de las soluciones a problemas comunes en el desarrollo de software y facilitan la reutilización de arquitecturas y diseños de sistemas exitosos. La aplicación de un patrón de diseño no tiene efectos sobre la estructura fundamental del sistema (arquitectura), pero puede tener una fuerte influencia sobre la arquitectura de un subsistema.

**Patrones GRASP:** Describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones. En la implementación de la solución se emplearon los siguientes patrones: Controlador, Creador y Experto.

- **Controlador:** El patrón controlador permite facilitar la centralización de actividades, delegar las actividades en otras clases con las que mantiene un modelo de alta cohesión. El uso de este patrón se pone de manifiesto en la utilización de diversos controladores en el sistema como por ejemplo

máster y conexión. Los controladores mencionados anteriormente son los encargados de interactuar con el modelo y enviar una respuesta a la vista. En la Figura 5: Patrón Controlador.

```
<?php
include("../configuracion/conexion.php");
class master {
    var $conexion_class;
    var $conecto;

    public function __construct() {
        $this->conexion_class = new Conexion_BD();
        $this->conecto = $this->conexion_class->conectar_bd();
    }
}
```

Figura 4: Patrón Controlador.

- Creador: La creación de objetos es una de las actividades más frecuentes en un sistema orientado a objetos. Guía la asignación de responsabilidades relacionadas con la creación de objetos. El propósito fundamental del patrón es encontrar un creador que debe conectar con el objeto producido en cualquier evento. En la Figura 5 se muestra la utilización del patrón en el sistema.

- Experto: Para el desarrollo de un sistema se pueden definir muchas clases de software, y una aplicación tal vez requiera el cumplimiento de varias responsabilidades. Si estas se asignan en forma adecuada, los sistemas tienden a ser más fáciles de entender, mantener y ampliar, y da la oportunidad de reutilizar los componentes en futuras aplicaciones. Este patrón se usa para asignar responsabilidades a clases que cuentan con la información necesaria para cumplirlas. La Figura 6: Patrón Experto muestra la utilización del patrón en el sistema:

```
<?php
class Conexion_BD {
    var $_ip;
    var $puertobd;
    var $nombrebd;
    var $usuariobd;
    var $contrasennabd;
    var $clusterbd;
    var $connect;

    public function __construct() {
        $configuracion = parse_ini_file("../configuracion/archivo_configuracion.conf");
        $this->_ip = $configuracion['_IP_BD'];
        $this->puertobd = $configuracion['_PUERTO_BD'];
        $this->nombrebd = $configuracion['_NOMBRE_BD'];
        $this->usuariobd = $configuracion['_USUARIO_BD'];
        $this->contrasennabd = $configuracion['_CONTRASENNA_BD'];
        $this->clusterbd = $configuracion['_CLUSTER_REP'];
        $this->connect = 0;
    }
}
```

Figura 5: Patrón Experto.

## 2.4. Implementación de la aplicación

En el presente epígrafe se exponen las tareas de ingeniería diseñadas con el objetivo de dividir las Historias de Usuario en tareas de implementación y realizar una mejor planificación del trabajo de los programadores y se sintetizan además el estándar de codificación que permitirá a los programadores un proceso de implementación más cómodo, rápido y con la menor cantidad de errores.

### 2.4.1. Tareas de ingeniería

Las tareas de ingeniería surgen de un proceso de revisión y análisis realizado por parte del equipo de desarrollo a las Historias de Usuario definidas con anterioridad, con el objetivo de dividir las funcionalidades

que engloba cada Historia de Usuario en tareas más pequeñas, tributando al correcto funcionamiento de la funcionalidad y una mejor implementación.

Cada una de las tareas está compuestas por su número, el número de Historia de Usuario de la que se deriva, el nombre de la tarea, tipo, la fecha de inicio y fecha de fin, el tiempo estimado, quién es el responsable de implementarla y una breve descripción. Para la implementación de la solución a desarrollar se identificaron un total de 16 tareas de ingenierías correspondientes a las Historias de Usuarios identificadas. Estas se definen en las tablas siguientes.

*Tabla 23: Tarea de Ingeniería Autenticar usuario.*

Tarea de Ingeniería	
<b>Número de Tarea:</b> 1	<b>Número Historia de Usuario:</b> 1
<b>Nombre Tarea:</b> Autenticar usuario.	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha Inicio:</b> 27/02/2015	<b>Fecha Fin:</b> 03/03/2015
<b>Programador responsable:</b> Nurisely Ramos López	
<b>Descripción:</b> Permite al usuario poder autenticarse en la aplicación.	

*Tabla 24: Tarea de Ingeniería Cambiar contraseña.*

Tarea de Ingeniería	
<b>Número de Tarea:</b> 2	<b>Número Historia de Usuario:</b> 2
<b>Nombre Tarea:</b> Cambiar contraseña.	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha Inicio:</b> 04/03/2015	<b>Fecha Fin:</b> 10/03/2015
<b>Programador responsable:</b> Nurisely Ramos López	
<b>Descripción:</b> Permite al usuario poder cambiar la contraseña.	

*Tabla 25: Tarea de Ingeniería Configuración de conexión.*

Tarea de Ingeniería	
<b>Número de Tarea:</b> 3	<b>Número Historia de Usuario:</b> 3
<b>Nombre Tarea:</b> Configuración de conexión.	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha Inicio:</b> 11/03/2015	<b>Fecha Fin:</b> 17/03/2015
<b>Programador responsable:</b> Nurisely Ramos López	

## INTERFAZ WEB PARA EL MONITOREO DE LA HERRAMIENTA DE RÉPLICA SLONY

**Descripción:** Permite al usuario poder cambiar la configuración de conexión.

*Tabla 26: Tarea de Ingeniería Visualizar listado de las tablas del clúster de Slony.*

Tarea de Ingeniería	
<b>Número de Tarea:</b> 4	<b>Número Historia de Usuario:</b> 4
<b>Nombre Tarea:</b> Visualizar listado de las tablas del clúster de Slony.	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha Inicio:</b> 18/03/2015	<b>Fecha Fin:</b> 24/03/2015
<b>Programador responsable:</b> Nurisely Ramos López	
<b>Descripción:</b> Permite al usuario poder visualizar todas las tablas del clúster de Slony.	

*Tabla 27: Tarea de Ingeniería Visualizar los datos de cada tabla según se van seleccionando por el usuario.*

Tarea de Ingeniería	
<b>Número de Tarea:</b> 5	<b>Número Historia de Usuario:</b> 5
<b>Nombre Tarea:</b> Visualizar los datos de cada tabla según se van seleccionando por el usuario.	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha Inicio:</b> 25/03/2015	<b>Fecha Fin:</b> 31/03/2015
<b>Programador responsable:</b> Nurisely Ramos López	
<b>Descripción:</b> Permite al usuario poder visualizar todos los datos e información de las tablas del clúster de Slony.	

*Tabla 28: Tarea de Ingeniería Listar los nodos y mostrar la información de estos.*

Tarea de Ingeniería	
<b>Número de Tarea:</b> 6	<b>Número Historia de Usuario:</b> 6
<b>Nombre Tarea:</b> Visualizar los nodos.	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha Inicio:</b> 01/04/2015	<b>Fecha Fin:</b> 07/04/2015
<b>Programador responsable:</b> Nurisely Ramos López	
<b>Descripción:</b> Permite al usuario visualizar los nodos que están participando en la réplica así como la información perteneciente a cada uno de ellos.	

*Tabla 29: Tarea de Ingeniería Visualizar los set de réplicas.*

Tarea de Ingeniería	
<b>Número de Tarea:</b> 7	<b>Número Historia de Usuario:</b> 7
<b>Nombre Tarea:</b> Visualizar los set de réplicas	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1

## INTERFAZ WEB PARA EL MONITOREO DE LA HERRAMIENTA DE RÉPLICA SLONY

<b>Fecha Inicio:</b> 08/04/2015	<b>Fecha Fin:</b> 14/04/2015
<b>Programador responsable:</b> Nurisely Ramos López	
<b>Descripción:</b> Permite al usuario visualizar los sets de réplica que están participando en la réplica.	

*Tabla 30: Tarea de Ingeniería Visualizar los logs de la herramienta Slony.*

Tarea de Ingeniería	
<b>Número de Tarea:</b> 8	<b>Número Historia de Usuario:</b> 8
<b>Nombre Tarea:</b> Visualizar los logs de la herramienta Slony.	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha Inicio:</b> 15/04/2015	<b>Fecha Fin:</b> 21/04/2015
<b>Programador responsable:</b> Nurisely Ramos López	
<b>Descripción:</b> Permite al usuario visualizar los logs de la herramienta Slony según vayan siendo generados por la misma.	

*Tabla 31: Tarea de Ingeniería Visualizar los logs del servidor PostgreSQL.*

Tarea de Ingeniería	
<b>Número de Tarea:</b> 9	<b>Número Historia de Usuario:</b> 9
<b>Nombre Tarea:</b> Visualizar los logs del servidor PostgreSQL.	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha Inicio:</b> 15/04/2015	<b>Fecha Fin:</b> 21/04/2015
<b>Programador responsable:</b> Nurisely Ramos López	
<b>Descripción:</b> Permite al usuario visualizar los logs del servidor de PostgreSQL según vayan siendo generados por el mismo.	

*Tabla 32: Tarea de Ingeniería Mostrar gráficamente el consumo de RAM del proceso Slon.*

Tarea de Ingeniería	
<b>Número de Tarea:</b> 10	<b>Número Historia de Usuario:</b> 10
<b>Nombre Tarea:</b> Mostrar gráficamente el consumo de RAM del proceso Slon.	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha Inicio:</b> 22/04/2015	<b>Fecha Fin:</b> 28/04/2015
<b>Programador responsable:</b> Nurisely Ramos López	
<b>Descripción:</b> Permite al usuario visualizar el consumo de RAM del proceso Slon.	

*Tabla 33: Tarea de Ingeniería Mostrar gráficamente el consumo de RAM del proceso PostgreSQL.*

Tarea de Ingeniería
---------------------

## INTERFAZ WEB PARA EL MONITOREO DE LA HERRAMIENTA DE RÉPLICA SLONY

<b>Número de Tarea:</b> 11	<b>Número Historia de Usuario:</b> 11
<b>Nombre Tarea:</b> Mostrar gráficamente el consumo de RAM del proceso PostgreSQL.	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha Inicio:</b> 29/04/2015	<b>Fecha Fin:</b> 05/05/2015
<b>Programador responsable:</b> Nurisely Ramos López	
<b>Descripción:</b> Permite al usuario visualizar el consumo de RAM del proceso PostgreSQL.	

*Tabla 34: Tarea de Ingeniería Visualizar los logs por tipos según las categorías en las que se divide.*

Tarea de Ingeniería	
<b>Número de Tarea:</b> 12	<b>Número Historia de Usuario:</b> 12
<b>Nombre Tarea:</b> Visualizar los logs por tipos según las categorías en las que se divide.	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.5
<b>Fecha Inicio:</b> 06/05/2015	<b>Fecha Fin:</b> 12/05/2015
<b>Programador responsable:</b> Nurisely Ramos López	
<b>Descripción:</b> Permite al usuario visualizar los logs por los tipos según las categorías en las que se dividen: CDT LOG, CDT ERROR, CDT ESTATEMENT, CDT FATAL, CDT DETAIL, CTD CONFIG, CTD QUERY, CTD CONTEXT, CTD HINT, CTD INFO.	

*Tabla 35: Tarea de Ingeniería Mostrar gráficamente el histórico de las operaciones realizadas dado un conjunto de parámetros.*

Tarea de Ingeniería	
<b>Número de Tarea:</b> 13	<b>Número Historia de Usuario:</b> 13
<b>Nombre Tarea:</b> Mostrar gráficamente el histórico de las operaciones realizadas dado un conjunto de parámetros.	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.5
<b>Fecha Inicio:</b> 13/05/2015	<b>Fecha Fin:</b> 19/06/2015
<b>Programador responsable:</b> Nurisely Ramos López	
<b>Descripción:</b> Permite al usuario visualizar en una gráfica el porcentaje de las operaciones dado un conjunto de parámetros: Actualizar, Insertar o Eliminar.	

*Tabla 36: Tarea de Ingeniería Mostrar las tablas con los resultados de la búsqueda en el histórico de las operaciones realizadas dado un conjunto de parámetros.*

Tarea de Ingeniería	
<b>Número de Tarea:</b> 14	<b>Número Historia de Usuario:</b> 14
<b>Nombre Tarea:</b> Mostrar las tablas con los resultados de la búsqueda en el histórico de las operaciones realizadas dado un conjunto de parámetros.	

<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha Inicio:</b> 20/05/2015	<b>Fecha Fin:</b> 26/05/2015
<b>Programador responsable:</b> Nurisely Ramos López	
<b>Descripción:</b> Permite al usuario visualizar en una tabla los resultados de dichas operaciones.	

*Tabla 37: Tarea de Ingeniería Extraer de la tabla sl\_path la información de los nodos.*

Tarea de Ingeniería	
<b>Número de Tarea:</b> 15	<b>Número Historia de Usuario:</b> 15
<b>Nombre Tarea:</b> Extraer de la tabla sl_path la información de los nodos.	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha Inicio:</b> 27/05/2015	<b>Fecha Fin:</b> 02/06/2015
<b>Programador responsable:</b> Nurisely Ramos López	
<b>Descripción:</b> Permite al usuario concatenarla tabla sl_path con la tabla sl_node para obtener y mostrar la información de los nodos.	

*Tabla 38: Tarea de Ingeniería Visualizar versión de Slony.*

Tarea de Ingeniería	
<b>Número de Tarea:</b> 16	<b>Número Historia de Usuario:</b> 16
<b>Nombre Tarea:</b> Visualizar versión de Slony.	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha Inicio:</b> 03/06/2015	<b>Fecha Fin:</b> 09/06/2015
<b>Programador responsable:</b> Nurisely Ramos López	
<b>Descripción:</b> Permite al usuario visualizar la versión de Slony.	

#### **2.4.2. Estándares de codificación**

El propósito fundamental de los estándares de codificación (27) es establecer una arquitectura y un estilo consistente, independiente del programador, para que el sistema resulte fácil de entender y por supuesto fácil de mantener. El mayor problema que trata de enfrentar esta práctica es tratar de entender el formato y el estilo utilizado en el código escrito por otros desarrolladores.

Los estándares de codificación son un complemento a la programación por pares y es importante tener un buen estándar de codificación que permita clarificar el código más que confundir, promover la necesidad del

código, permitir que los programas se acerquen lo mejor posible al lenguaje natural, incorporar las mejores prácticas de la codificación para un mejor entendimiento entre los programadores.

Para la implementación de la solución se empleará el estándar de codificación definido para PHP, disponible en <http://www.php-fig.org/psr/psr-1/es/>, el cual se presenta a continuación: (28)

### 1. Archivos

#### Etiquetas PHP

- ✓ El código PHP debe utilizar las etiquetas largas `<?php ?>` o las etiquetas cortas para imprimir salida de información `<?= ?>`; no debe emplear otras variantes.

#### Codificación de caracteres

- ✓ El código PHP debe utilizar codificación UTF-8 sin BOM.

#### Efectos secundarios

- ✓ Un archivo debería declarar estructuras (clases, funciones, constantes, etcétera) y no causar efectos secundarios, o debería ejecutar partes de la lógica de negocio, pero no debería hacer las dos cosas.
- ✓ La frase "efectos secundarios" significa: que la ejecución de la lógica de negocio no está directamente relacionado con declarar clases, funciones, constantes, etcétera, simplemente la de incluir el archivo.
- ✓ "Efectos secundarios" incluyen, pero no se limitan a: generar salidas, uso explícito de requiere o include, conexiones a servicios externos, modificación de configuraciones iniciales, enviar errores o excepciones, modificar variables globales o estáticas, leer o escribir un archivo, etcétera.

### 2. Espacios de nombres y nombres de las Clases

- ✓ Los espacios de nombres y las clases deben seguir el estándar PSR-0.
- ✓ Esto significa que cada clase estará en un fichero independiente y está dentro de un espacio de nombres en al menos un nivel: un nombre de proveedor de nivel superior.
- ✓ El código escrito para PHP 5.3 o superior debe hacer un uso formal de los espacios de nombres.

- ✓ El código escrito para PHP 5.2.x o inferior debería emplear una convención de pseudo-espacios de nombres con prefijos en los nombres de las clases con el formato por ejemplo, Proveedor\_.

### 3. Constantes de Clases, Propiedades y Métodos

El término "clases" hace referencia a todas las clases e interfaces.

#### Constantes

- ✓ Las constantes de las clases deben declararse siempre en mayúsculas y separadas por guiones bajos.

#### Propiedades

- ✓ Esta guía evita intencionadamente cualquier recomendación respecto al uso de las notaciones `$StudlyCaps`<sup>1</sup>, `$camelCase`<sup>2</sup>, o `$guion_bajo` en los nombres de las propiedades.
- ✓ Cualquiera que sea la convención en nomenclatura, debería ser utilizada de forma coherente con un alcance razonable. Este alcance puede ser a nivel de proveedor, a nivel de paquete, a nivel de clase o a nivel de método.

#### Métodos

- ✓ Los nombres de los métodos deben declararse en notación camelCase ().

#### 2.4.3. Interfaces principales de la aplicación

En este epígrafe se muestran ejemplos de las interfaces del sistema desarrollado. La Figura 7 Interfaz de autenticación de usuario corresponde a la interfaz de autenticación, en esta el usuario (administrador) podrá autenticarse mediante un usuario y contraseña en la interfaz web para acceder luego a sus funcionalidades. Luego de introducir los datos se debe seleccionar la opción Iniciar sesión.

---

<sup>1</sup> `$StudlyCaps`, es una forma de notación de texto que sigue el patrón de palabras en minúscula sin espacios y con la primera letra de cada palabra en mayúscula.

<sup>2</sup> `$camelCase`, es una forma de notación de texto que sigue el patrón de palabras en minúscula sin espacios y con la primera letra de cada palabra en mayúsculas exceptuando la primera palabra.

## INTERFAZ WEB PARA EL MONITOREO DE LA HERRAMIENTA DE RÉPLICA SLONY

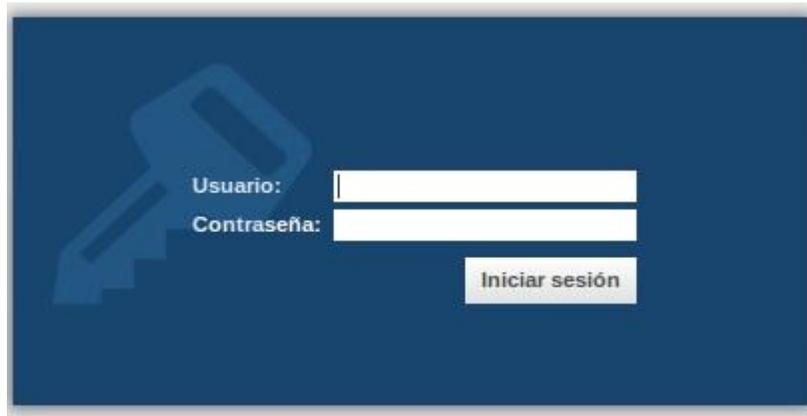


Figura 6: Interfaz de autenticación de usuario

En la Figura 8 Interfaz principal el usuario (administrador) luego de autenticarse en la interfaz podrá visualizar el consumo de RAM del proceso de Slon y de PostgreSQL, el usuario podrá también acceder desde esta interfaz a las tablas, nodos, logs, sets. de la réplica.

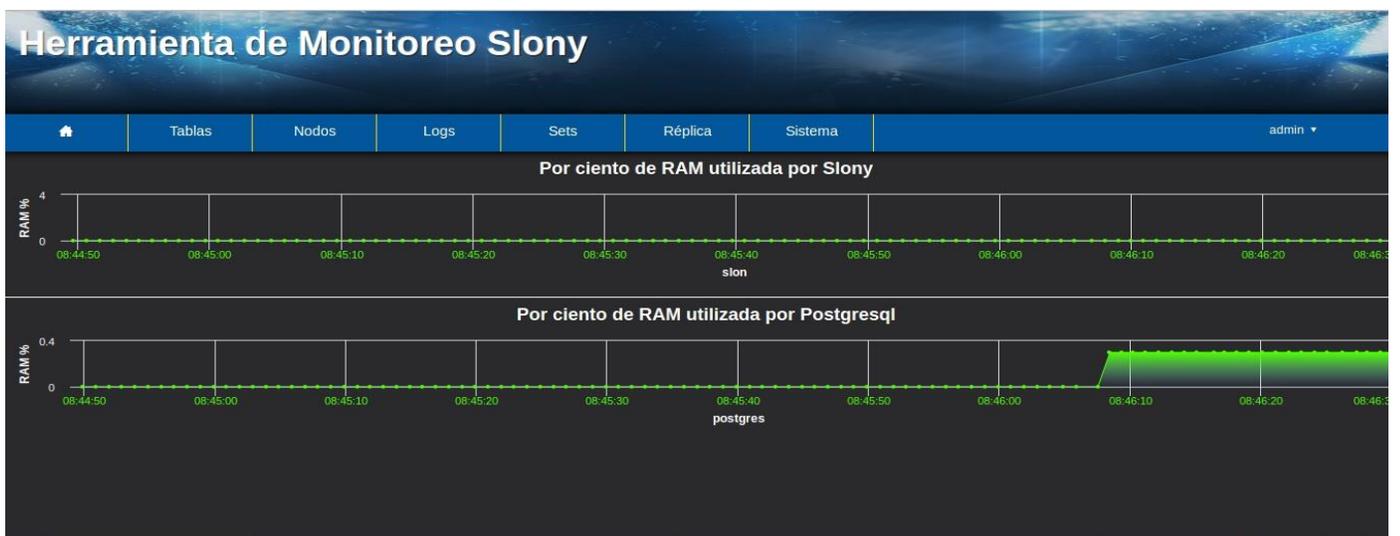


Figura 7: Interfaz principal.

La Figura 9 Tablas del clúster de Slony representa a las tablas del clúster de Slony. Se puede acceder a las tablas mediante la opción *Tablas* en la parte superior izquierda al lado del menú Inicio. Esta permitirá al usuario visualizar los datos de esta y la información de cada una.

sl_event													
No	ev_origin	ev_seqno	ev_timesta mp	ev_snapsho t	ev_type	ev_data1	ev_data2	ev_data3	ev_data4	ev_data5	ev_data6	ev_data7	ev_data8
1	1	5000000001	2015-06-18 23:12:29.781 634-04	264088:2640 88:	ENABLE_NO DE	1							
2	1	5000000002	2015-06-18 23:12:31.135 573-04	264089:2640 89:	STORE_SET	1	1	set1					
3	1	5000000003	2015-06-18 23:12:31.151 189-04	264090:2640 90:	SYNC								
4	1	5000000004	2015-06-18 23:12:31.162 22-04	264091:2640 91:	SET_ADD_T ABLE	1	1	public.catego ries	categories_p key	tabla categories			
5	1	5000000005	2015-06-18 23:12:31.262 175-04	264092:2640 92:	SYNC								
6	1	5000000006	2015-06-18 23:12:31.273 504-04	264093:2640 93:	SET_ADD_T ABLE	1	2	public.cust_h ist	pk_primary	tabla st_hist			

Figura 8: Tablas del clúster de Slony.

En la Figura 10 Nodos de la réplica se muestra los nodos insertados. Se puede acceder a los nodos mediante la opción *Nodos* en la parte superior izquierda al lado de la opción *Tablas*. Esta permitirá al usuario visualizar todos los nodos que están insertados.

ID del Nodo:	3	ID del Nodo:	1
Activo:	si	Activo:	si
Nombre:	Nodo DellStore2	Nombre:	Nodo Maestro
Base de Datos:	DellStore2	Base de Datos:	master
IP del Nodo:	127.0.0.1	IP del Nodo:	127.0.0.1
Usuario de la Base de Datos:	postgres	Usuario de la Base de Datos:	postgres

Figura 9: Nodos de la réplica.

En la Figura 11: Logs del sistema se muestra los logs de Slony y PostgreSQL. Se puede acceder a los logs mediante la opción *Logs* en la parte superior izquierda al lado de la opción *Nodos*. Esta permitirá al usuario visualizar todos los logs del sistema.



Figura 10: Logs del sistema.

En la Figura 12: Sets del sistema se muestra los sets. Se puede acceder a los sets mediante la opción *Sets* en la parte superior izquierda al lado de la opción *Logs*. Esta permitirá al usuario visualizar todos los sets del sistema.

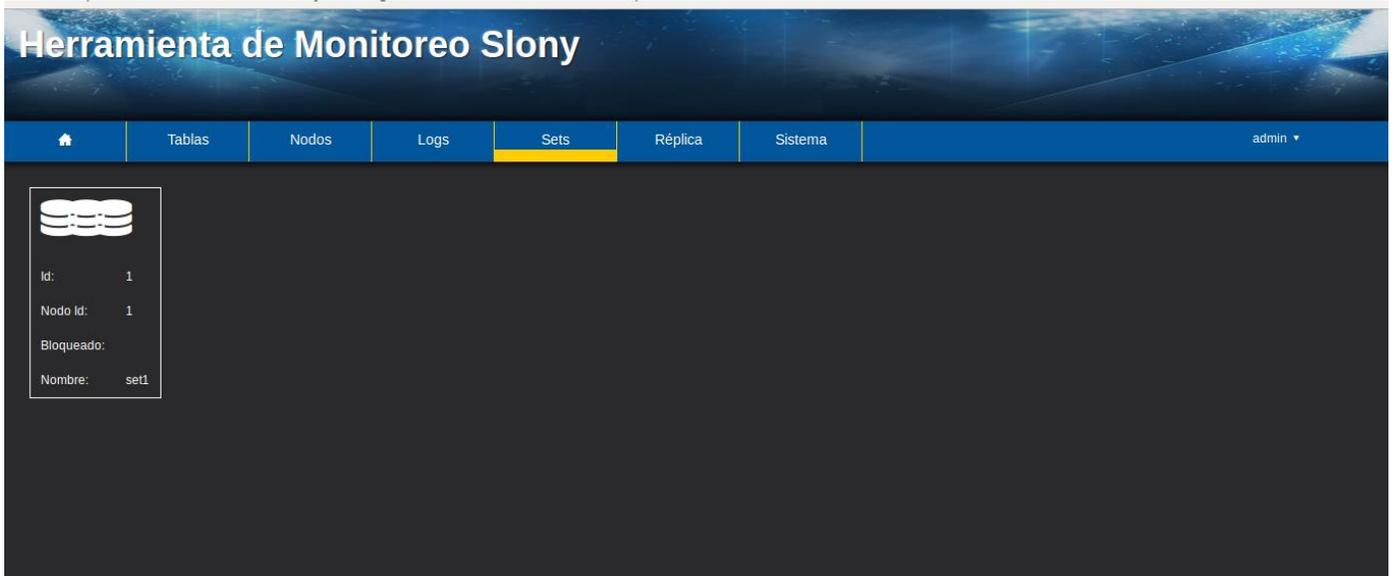


Figura 11: Sets del sistema.

En la Figura 13: Interfaz Réplica se muestra las gráficas y las tablas de las operaciones realizadas según los parámetros dados. Se puede acceder a estas mediante la opción *Réplica* en la parte superior izquierda al lado de la opción *Sets*.



## INTERFAZ WEB PARA EL MONITOREO DE LA HERRAMIENTA DE RÉPLICA SLONY

Figura 12: Interfaz Réplica.

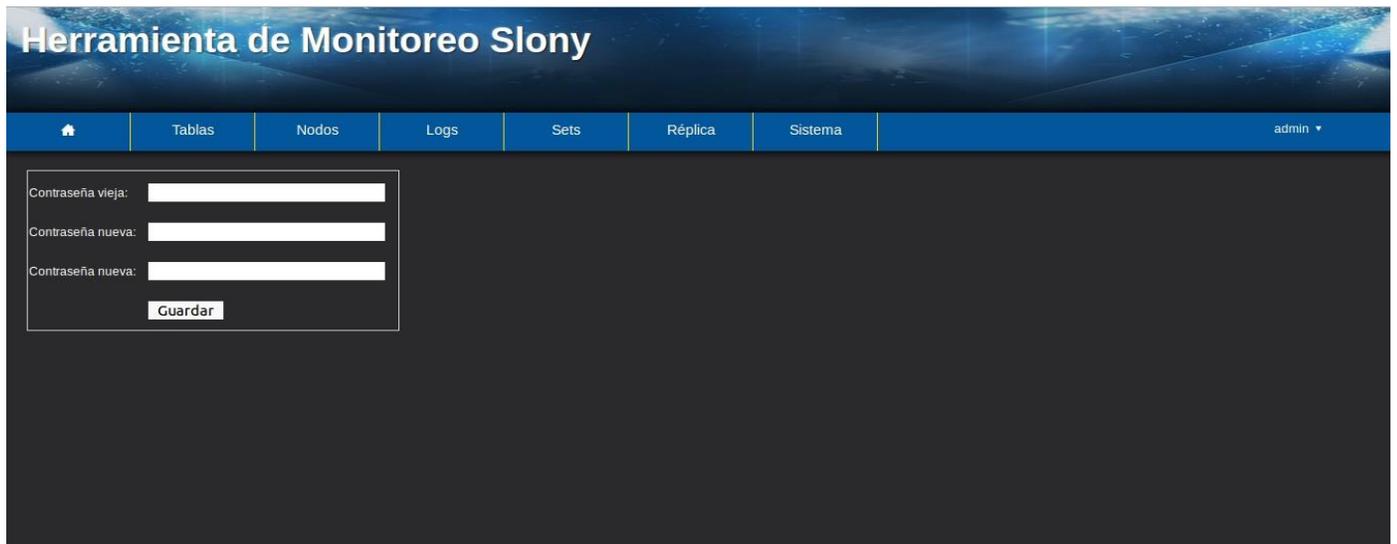
En la Figura 14: Interfaz Sistema se muestran las versiones de las tecnologías utilizadas para la realización de esta interfaz web. Se pueden visualizar mediante la opción *Sistema* en la parte superior izquierda al lado de la opción *Réplica*.



Slony	Postgresql	Ubuntu
2.1.4	9.3	14.04.2 LTS

Figura 13: Interfaz Sistema.

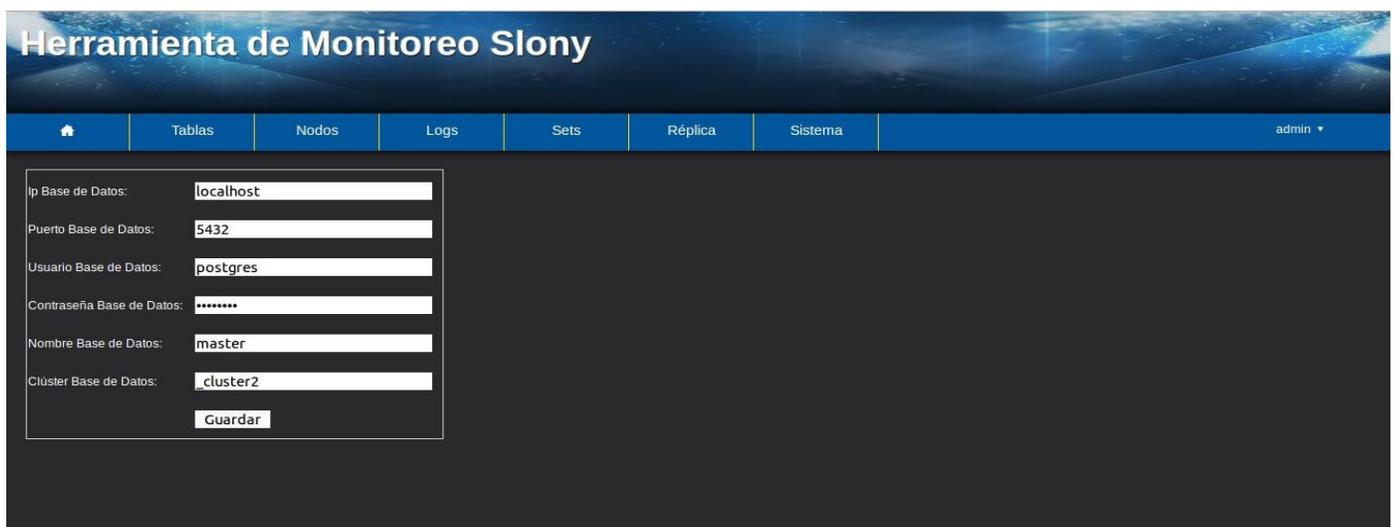
En la Figura 15: Interfaz Cambiar Contraseña se muestra dónde es que se va a cambiar la contraseña. Se puede observar en la parte superior derecha al lado del administrador registrado, donde se encuentra una pequeña flecha.



The screenshot shows the 'Herramienta de Monitoreo Slony' web interface. The top navigation bar includes a home icon, 'Tablas', 'Nodos', 'Logs', 'Sets', 'Réplica', 'Sistema', and a user dropdown menu showing 'admin'. The main content area is a dark grey panel with a white-bordered form for changing a password. The form contains three input fields: 'Contraseña vieja:', 'Contraseña nueva:', and 'Contraseña nueva:'. Below the fields is a 'Guardar' button.

Figura 14: Interfaz Cambiar Contraseña.

En la Figura 16: Interfaz Configuración de conexión se muestra los parámetros de la configuración de conexión. Se puede observar en la parte superior derecha al lado del administrador registrado, donde se encuentra una pequeña flecha.



The screenshot shows the 'Herramienta de Monitoreo Slony' web interface for connection configuration. The top navigation bar is identical to Figure 14. The main content area is a dark grey panel with a white-bordered form for configuring connection parameters. The form includes six input fields: 'Ip Base de Datos:' (localhost), 'Puerto Base de Datos:' (5432), 'Usuario Base de Datos:' (postgres), 'Contraseña Base de Datos:' (masked with dots), 'Nombre Base de Datos:' (master), and 'Cluster Base de Datos:' (cluster2). A 'Guardar' button is located at the bottom of the form.

Figura 15: Interfaz Configuración de conexión.

### Conclusiones del capítulo

En este capítulo para facilitar el entendimiento del proceso de desarrollo de la solución se confeccionó el modelo de dominio para comprender el funcionamiento del negocio, el diagrama de clases para representar las clases de la interfaz web implementadas. Fueron identificadas 16 historias de usuario desglosadas en 16 requisitos funcionales. Se identificaron también 6 requisitos no funcionales todos estos agrupados en la lista de reserva del producto. Se confeccionaron 2 tarjetas CRC para la implementación de las historias de usuario definidas. Se elaboró además un plan de iteraciones, con el que se concluye que la implementación de la extensión tendrá una duración de 16 semanas. Se empleó Modelo-Vista-Controlador como patrón arquitectónico, y entre los patrones de diseño se utilizaron los patrones GRASP: Experto, Creador y Controlador. Se diseñaron un total de 16 tareas de ingeniería para desarrollar las Historias de Usuarios y cumplir con la implementación de cada una de las funcionalidades que debe cumplir el sistema desarrollado para satisfacer las necesidades del cliente. La implementación fue guiada por el estándar de codificación para el lenguaje PHP lo que permitió realizar una entrega funcional al terminar cada iteración planificada en la etapa de análisis y cumplir con la fecha de entrega definido por el cliente.

### CAPÍTULO 3: VALIDACIÓN DE LA SOLUCIÓN

Después de haber implementado y desarrollado, se dispone de los detalles suficientes para proceder a la construcción del sistema, y una vez concluido este, se procede a la verificación del cumplimiento de los requisitos funcionales mediante las pruebas de software. En el presente capítulo se determinan los tipos de pruebas a realizar y los casos de prueba que serán aplicados al sistema.

#### 3.1. Proceso de pruebas

Las pruebas son fundamentales en XP, estas constituyen una etapa dentro del desarrollo del software que permiten comprobar y revelar la calidad de un producto final. Son utilizadas para identificar fallos en la implementación o usabilidad del programa. Implican la participación directa del usuario en el desarrollo de las mismas y en la validación. (29)

Sommerville en el texto Ingeniería del software. Vol. 7 plantea tres etapas de pruebas: pruebas de componentes, pruebas de sistema y pruebas de aceptación. Para la presente investigación por las características y funcionalidades de la interfaz web desarrollada se aplicarán las pruebas de aceptación.

##### 3.1.1. Pruebas de aceptación

Estas pruebas resultan las más importantes dado que las mismas verifican y controlan el grado de aceptación del cliente con el producto que se ha desarrollado, marcando así el final de una iteración y el inicio de la próxima. El cliente final del software es quien debe desarrollar estas pruebas para corroborar que se cumplan correctamente las funcionalidades exigidas por él.

Las pruebas de aceptación son creadas a partir de la realización de las historias de usuario. Durante cada iteración, la historia de usuario escogida en la planificación de iteraciones se convertirá en una o varias pruebas de aceptación; y el usuario especifica los aspectos a probar cuando una historia de usuario ha sido correctamente implementada.

Durante las pruebas de aceptación se aplicará el método de caja negra, el cual se encarga de examinar los aspectos funcionales del sistema sin tener en cuenta su estructura lógica interna, de ahí que se centre en los requisitos funcionales del software sin tener en cuenta la estructura interna del mismo. Estas pruebas

permiten obtener un conjunto de condiciones de entrada que ejerciten completamente todas las funcionalidades del programa. Se trata de un enfoque que intenta descubrir diferentes tipos de errores tales como: (22)

- Funciones incorrectas o ausentes
- Errores en la interfaz
- Errores en estructuras de datos o en accesos a bases de datos externas
- Errores de rendimiento
- Errores de inicialización y de terminación

A pesar de que la metodología XP no especifica la utilización de casos de prueba como una técnica para probar las funcionalidades del sistema, se utiliza la misma con el fin de demostrar que las funciones del software desarrollado son operativas, que la entrada se acepta de forma adecuada y que se produce una salida correcta, así como que la integridad de la información externa se mantiene.

Para confeccionar los casos de prueba de caja negra se empleará la Técnica de la Partición de Equivalencia: encargada de dividir el campo de entrada en clases de datos que tienden a ejercitar determinadas funciones del software. Esta técnica se utiliza para representar, a través de escenarios, una descripción de la acción del usuario, la respuesta que se espera y el resultado obtenido, lo que permite comprobar el buen funcionamiento del sistema. Se dirige a una definición de casos de prueba que descubran clases de errores, reduciendo así el número total de casos de prueba que hay que desarrollar.

### **3.2. Diseño de caso de prueba y registro de no conformidades.**

Es muy importante asegurar que la realización de las pruebas se efectúe con buena calidad y para esto se hace necesario, que la construcción de los casos de prueba que se ejecutan para comprobar el correcto funcionamiento del sistema se realice de forma precisa y eficiente. Cada planilla de caso de prueba se diseña según las funcionalidades descritas en las historias de usuario, recogiendo la especificación de cada una, dividiéndola en secciones y escenarios, detallando las funcionalidades descritas en estas y describiendo cada variable que recoge la historia de usuario en cuestión. De esta manera quedan plasmadas las revisiones realizadas al caso de prueba, así como un registro de todo aquello que no se corresponde con la calidad del producto.

En el proceso de pruebas en cuestión se hace uso de las pruebas de aceptación, como técnica de prueba, la pruebas de caja negra y el método utilizado partición de equivalencia. La Tabla 39 Secciones para probar la historia de usuario Autenticar usuario se utiliza para desglosar las historias de usuario en secciones y a su vez en escenarios, haciendo más fructífera la ejecución de las pruebas. Esta contiene los siguientes campos:

- Nombre de la sección: se especifica el nombre de la sección [SC1: Nombre de la sección].
- Escenarios de la sección: se especifican los escenarios de cada sección [EC1.1: Nombre del Escenario].
- Descripción de la funcionalidad: se describe brevemente la funcionalidad del escenario.

El siguiente es un ejemplo donde se detalla la historia de usuario Autenticar Usuario.

*Tabla 39: Secciones para probar la historia de usuario Autenticar usuario*

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad
SC1: Autenticar usuario	EC1.1: Autenticar usuario con datos correctos	El administrador ingresa el usuario y la contraseña de forma correcta accediendo a la página principal de la interfaz.
	EC1.2: Autenticar usuario con datos incorrectos	El administrador ingresa alguno de los datos incorrectos, ya sea el usuario o la contraseña. El sistema muestra un mensaje indicando que ha ocurrido un error.
	EC2.2: Autenticar usuario con datos vacíos	El administrador al ingresa los datos deja un campo vacío, ya sea el usuario o la contraseña. El sistema muestra un mensaje indicando que falta campo por completar.

A partir de esta descripción se detallan las variables asociadas a la historia de usuario. Esta descripción está compuesta por los siguientes campos:

- No: se enumeraron todos los campos o variables, descritos en el caso de prueba.
- Nombre del campo: se especificó el nombre del campo de entrada.
- Clasificación: se especificó la clasificación según el componente de diseño utilizado, ejemplo: texto o consulta.

- Nulo: se especificó si el campo puede ser nulo o no, para ello solo se utilizó Sí o No en cada caso respectivamente.
- Descripción: se describieron brevemente los datos que debían introducirse.

En la Tabla 40: Descripción de variables de la historia de usuario Autenticar usuario se muestra un ejemplo de la descripción de variables de la historia de usuario Autenticar usuario.

*Tabla 40: Descripción de variables de la historia de usuario Autenticar usuario*

No.	Nombre del campo	Clasificación	Descripción
1	usuario	texto	Los usuarios deben comenzar con letras, seguido de combinaciones de letra, números y guion bajo.
2	contraseña	texto	Las contraseñas deben ser construida con una mezcla de caracteres alfabéticos (donde se combinen las mayúsculas y las minúsculas), dígitos e incluso caracteres especiales (@, !, &).

Para la ejecución de los casos de prueba se empleó la base de datos de tamaño normal de *Dell Store*<sup>3</sup>, disponible en el proyecto “Colección de bases de datos de ejemplos para PostgreSQL”<sup>4</sup>, diseñada para la versión 8.1 o superior del gestor; la cual tiene la estructura mostrada en la Tabla 41 Listado de relaciones.

*Tabla 41: Listado de relaciones*

Esquema	Nombre	Tipo	Total de registros
<i>public</i>	<i>categories</i>	tabla	16
<i>public</i>	<i>cust_hist</i>	tabla	60350
<i>public</i>	<i>customers</i>	tabla	20000
<i>public</i>	<i>inventory</i>	tabla	10000
<i>public</i>	<i>orderlines</i>	tabla	60350
<i>public</i>	<i>orders</i>	tabla	12000
<i>public</i>	<i>products</i>	tabla	10000
<i>public</i>	<i>reorder</i>	tabla	0

<sup>3</sup>Disponible en <http://linux.dell.com/dvdstore/>

<sup>4</sup>Disponible en <http://pgfoundry.org/projects/dbsamples/>

La descripción de las variables de Tabla 40: Descripción de variables de la historia de usuario Autenticar usuario permitió que se realizara una matriz de datos, donde se evaluó y probó la validez de cada uno de los datos introducidos en la extensión, específicamente en la sección que se estuvo probando. Utilizando un juego de datos válidos e inválidos se identificó el empleo de la técnica de partición de equivalencia. Esta matriz de datos contiene los siguientes aspectos:

- Id del escenario: se especifica el id del escenario [EC1.1,..., ECm.n]
- Escenario: se especifica el nombre del escenario.
- Variables [1, 2,..., n]: se especifica el nombre de la variable o el número, según la tabla de descripción de variables, y en su celda correspondiente se indicó el valor del dato [V (Válido), I (Inválido), N/A (No Aplica)].
- Respuesta del sistema: se escribe el resultado que se esperaba al realizar la prueba.
- Resultado de la prueba: se escribe el resultado que se obtuvo al realizar la prueba.
- Flujo central: se expone el flujo central de la historia de usuario al que se le estaba diseñando el caso de prueba.

En la Tabla 42: Matriz de datos de la historia de usuario Autenticar usuario específicamente de la Sección se muestra un ejemplo de la matriz de datos para la historia de usuario Autentica, de la SC1 Autentica usuario, con los juegos de datos tomados de la base de datos de prueba seleccionada.

*Tabla 42: Matriz de datos de la historia de usuario Autenticar usuario específicamente de la Sección*

ID del escenario	Escenario	Variables		Respuesta	Flujo
		1	2		
EC1.1	Autenticar usuario con datos correctos	V (admin)	V nury	El administrador se autentica correctamente.	1. El administrador introduce en el sistema su usuario y contraseña correctamente. 2. Se muestra la interfaz principal del sistema
EC1.2	Autenticar usuario con datos incorrectos	F	V	El sistema muestra un mensaje indicando que ha ocurrido un error.	1. El administrador introduce en el sistema su usuario y contraseña con datos incorrectos.
		adm	nury		
		V	F		
		admin	12nur		

## INTERFAZ WEB PARA EL MONITOREO DE LA HERRAMIENTA DE RÉPLICA SLONY

					2. El sistema muestra un mensaje indicando que ha ocurrido un error.
EC1.3	Autenticar usuario con datos vacío con datos incorrectos	I	V	El sistema muestra un mensaje indicando que falta campo por completar.	1. El administrador accede a la interfaz con su usuario y contraseña correctamente. 2. El sistema muestra un mensaje indicando que falta campo por completar.
		-	nury		
		V	I		
		admin	-		

Después de realizadas todas las pruebas, los resultados que no fueron satisfactorios pasaron a ser no conformidades y se emitieron en el registro de defectos y dificultades detectados que se encuentra en la parte final de cada diseño de caso de prueba. Se muestran en la Tabla 43: No conformidades detectadas en la aplicación todas estas no conformidades que se descubrieron en la interfaz web, donde se especifican los siguientes campos:

- Elemento: se especifica el nombre del elemento.
- No.: se especifica el número de la no conformidad.
- NC: se describe la no conformidad.
- Aspecto asociado: se especifica el aspecto correspondiente a la no conformidad.
- Etapa de detección: se especifica la etapa de detección del error.
- Clasificación: se pone una (S), en caso de que la no conformidad estuviera clasificada como significativa, (NS), en caso de que la no conformidad estuviera clasificada como no significativa y una (R), en caso de que la no conformidad solo fuera una recomendación
- Estado NC: se coloca el estado de la no conformidad y la fecha, cada vez que se revisa se deja el estado anterior y se coloca el nuevo con la fecha en que se revisó [RA: Resuelta, PD: Pendiente, NP: No Procede].
- Respuesta: se comienza a llenar a partir de la segunda iteración, y es responsabilidad del equipo de desarrollo, quien especifica si se ha resuelto o no la no conformidad y en caso de no proceder se explica la causa.

## INTERFAZ WEB PARA EL MONITOREO DE LA HERRAMIENTA DE RÉPLICA SLONY

*Tabla 43: No conformidades detectadas en la aplicación.*

Elemento	No.	NC	Aspecto asociado	Etapa de detección	Clasificación	Estado NC	Respuesta
Aplicación	1	El sistema al cambiar la contraseña no muestra un mensaje indicando el resultado de la operación.	Cambiar contraseña	Prueba	S	30/05/2015 PD 30/05/2015 RA	Corregido
Aplicación	2	Al seleccionar la opción <i>Tablas</i> se muestra la lista de tablas del clúster de Slony, pero no los datos que estas contienen	Visualizar los datos de cada tabla según se van seleccionando por el usuario.	Prueba	S	30/05/2015 PD 30/05/2015 RA	Corregido
Aplicación	3	Al seleccionar la opción para visualizar los logs del sistema se deben listar los tipos: CDT LOG, CDT ERROR, CDT STATEMENT, CDT FATAL, CDT DETAIL, CTD CONFIG, sin embargo no muestra el último CTD CONFIG.	Visualizar los logs de la herramienta Slony.	Prueba	S	30/05/2015 PD 30/05/2015 RA	Corregido
Aplicación	4	No se corresponde el % que se visualiza en la gráfica con el histórico de las operaciones.	Mostrar gráficamente el histórico de las operaciones realizadas dado un conjunto de parámetros.	Prueba	S	30/05/2015 PD 30/05/2015 RA	Corregido
Aplicación	5	Cuando se visualizan los datos de los	Visualizar los nodos	Prueba	S	30/05/2015 PD	Corregido

		nodos en la opción <i>Nodos</i> no se visualizan todos los datos (no se muestra la información id del nodo)				30/05/2015 RA	
Aplicación	6	Al seleccionar la opción <i>Sistema</i> esta no .se muestra la información asociada a la opción.	Visualizar versión de Slony	Prueba	NS	30/05/2015 PD 30/05/2015 RA	Corregido

### 3.3. Análisis de los resultados de las pruebas

En esta investigación se diseñaron un total de 16 casos de prueba, se realizó una primera iteración de pruebas en la cual fueron probados los casos de prueba correspondientes a las historias de usuario de prioridad Muy Alta, detectándose un total de 5 no conformidades significativas y 1 recomendación. En una segunda iteración fueron probados los casos de prueba de prioridad Alta, donde no se detectaron no conformidades y no se hicieron recomendaciones. En la tercera iteración fueron probados los casos de prueba de prioridad Media, donde se detectaron 1 no conformidad no significativa y 1 recomendación. Se realizó una cuarta iteración donde fueron probados nuevamente el total de casos de prueba diseñados para asegurar la calidad del producto, en la cual no se detectaron no conformidades. La siguiente gráfica muestra el resultado de la aplicación de las pruebas.

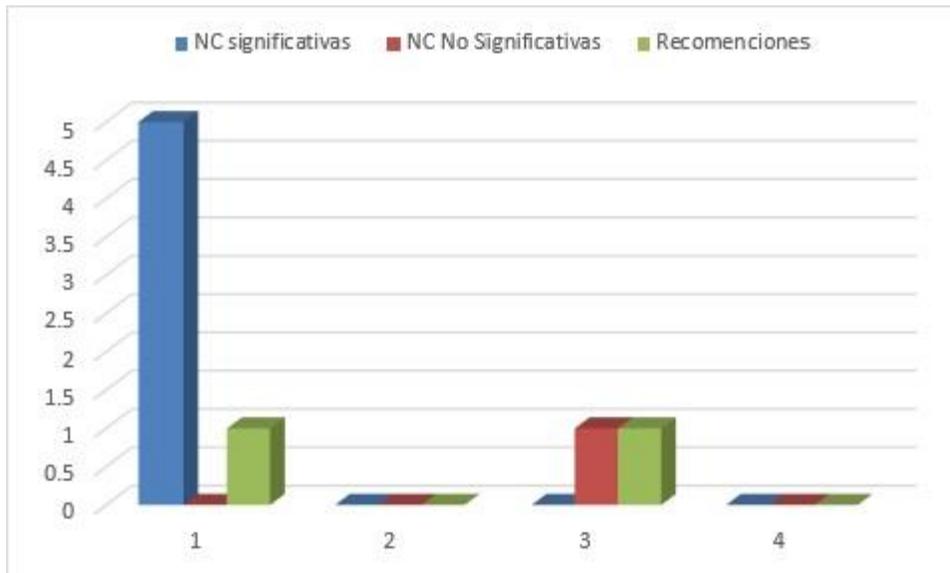


Figura 16: Resultados de las pruebas por iteración.

### Conclusiones del capítulo

Con el desarrollo del proceso de pruebas se pudo identificar y resolver los errores en la implementación aumentando la calidad del producto final obtenido. Para ello se realizaron cuatro iteraciones de pruebas en las cuales se probaron un total de 16 casos de prueba, detectándose 5 no conformidades significativas, 1 no significativa y 2 recomendaciones, las cuales fueron resueltas demostrando que la interfaz desarrollada cumple con las expectativas del cliente.

### CONCLUSIONES GENERALES

El desarrollo de la reciente investigación y los resultados generados por la misma, han permitido arribar a las siguientes conclusiones:

- Se realizó la selección de la metodología, herramientas y tecnologías para el desarrollo de la solución entre las que se encuentran: como metodología de desarrollo de software XP, UML como lenguaje de modelado y como herramienta CASE el Visual Paradigm 8.0. Se utilizó HTML5, CSS3, JavaScript, PHP5 como lenguajes de programación, jQuery 1.3.4 como framework, la técnica Ajax y el IDE de desarrollo NetBeans IDE 8.0.
- Se obtuvo la interfaz web para el monitoreo de la herramienta de réplica Slony, la cual implementa todas las funcionalidades identificadas para dar solución a la problemática planteada.
- La interfaz web permite al administrador monitorizar las tablas del clúster de Slony, además los nodos, los sets, los logs de la réplica. El administrador podrá también realizar análisis del comportamiento de la réplica mediante la consulta de tablas y gráficos que podrá elaborar definiendo un conjunto de parámetros según sus intereses.
- Se realizaron pruebas de aceptación utilizando el método de caja negra mediante la técnica de partición de equivalencia lo que permitió validar el correcto funcionamiento de la solución.

De manera general se le dio cumplimiento al objetivo planteado y actualmente la herramienta se encuentra a disposición de la comunidad de desarrollo de la Universidad de las Ciencias Informáticas.

### RECOMENDACIONES

Los objetivos de este trabajo se cumplieron, pero a lo largo de su desarrollo han surgido ideas que pudieran implementarse en un futuro, por lo que se recomienda:

- ✓ Probar el funcionamiento de la aplicación en otras distribuciones de Linux y utilizando versiones superiores de Slony.
- ✓ Agregar a la aplicación el soporte necesario para también monitorear el uso y almacenamiento utilizado por el servidor de bases de datos.
- ✓ Agregar a la aplicación un módulo que permita administrar la herramienta Slony.

## REFERENCIAS BIBLIOGRÁFICAS

1. Carrazana Carbonell, Luis E, y otros. Réplica entre Servidores Oracle. *Alternativas de solución*. Ciudad Habana : s.n., : s.n., 2006.
2. Moreno, G.R.M.,. *Replicación en PostgreSQL 9.0*, in *Facultad de Ciencias Exactas Base de Datos II*. . 2012, Universidad Nacional de Salta.
3. ITESCAM (Instituto Tecnológico Superior de Calkini). [En línea] [En línea]. [Citado el: 22 de 02 de 2015.] <http://www.itescam.edu.mx/principal/sylabus/fpdb/recursos/r11294.PDF>..
4. Vega, Norge Fajardo. *Sistema de réplica para bases de*. Ciudadde La Habana : s.n., 2007.
5. Microsoft. *Prácticas recomendadas para la Administración de replicación*. [En línea] 2014. <http://technet.microsoft.com>..
6. Slony - I. [En línea] 2007. [Citado el: 23 de 02 de 2015.] <http://www.slony.info>.
7. Yadima Morales Hernández, Joel Tamayo Vega. *Desarrollo del módulo General de la colección El Navegante*. Habana : s.n., : s.n., 2011.
8. *Departamento de Sistemas Informáticos y Computación Universidad Politécnica de Valencia. Rational Unified Process*. Valencia : s.n. : s.n.
9. Propel ORM. [En línea] En línea: 2010. [Citado el: 15/ de 01/ de 2011.] Disponible en: <http://www.propelorm.org/>..
10. Python vs. Php - UltimoCombate.com - Batallas resueltas en la Red. [En línea] 21 de enero de 2011. [Citado el: 21 de 01 de 2011.] [http://www.ultimocombate.com/python-vs-php\\_id512.html](http://www.ultimocombate.com/python-vs-php_id512.html)..
11. Procesos de Software. [En línea] 2013. [Citado el: 05 de Noviembre de 2013];.] <http://procesosdesoftware.wikispaces.com/METODOLOGIA+XP>.
12. Group, O.M. UML® Resource Page. [En línea] 1997-2013. <http://www.uml.org/>.
13. Miguel Ángel Conde González, J.C.-b.G., Rosa María Martín Moren, Inmaculada González Pérez & Francisco José García Peñalvo. *ARQUITECTURA PARA UN LMS BASADA EN PORTLETS: CLAYNET 2.0*. s.l. Vol. Vol. 7. Nº2. s.l. : Universidad DSalamanca., 2006.
14. ALFARO. *Herramientas Case*. 1999.
15. CASTRO, LUIS. ¿Qué es HTML 5? About.com Internet básico. [En línea] 2012. [Citado el: 12/ de 02/ de 2015.] Disponible en: <http://aprenderinternet.about.com/od/Glosario/g/Que-Es-Html-5.htm>Qué es HTML 5 – Conoce qué es HTML 5, cómo se diferencia de HTML 4 y cuáles son sus características principales.

16. CodeBox: Glosario. [online]. [En línea] [Citado el: 12/ de 02/ de 2015.] Disponible en: <http://www.codebox.es/glosario>.
17. Capítulo 1. Introducción | Introducción a JavaScript | LibrosWeb.es. [online]. [En línea] [Citado el: 12 de 02/ de 2012.] Disponible en: <http://www.librosweb.es/javascript/capitulo1.html>.
18. Eguiluz, Javier. *Introducción a Ajax*. s.l. : <https://librosweb.es/libro/ajax/>.
19. HINOSTROZA, Raul Rodas. Características de PHP. [En línea] 22/ de 02/ de 2007. [Citado el: 22/ de 02/ de 2015.] Disponible en: <http://www.linuxcentro.net/linux/staticpages/index.php?page=CaracteristicasPHP>.
20. Bienvenido a NetBeans y [www.netbeans.org](http://www.netbeans.org),. *Portal del IDE Java de Código Abierto*. [En línea] 2012. [Citado el: 22/ de 02/ de 2015.] Disponible en: [http://netbeans.org/index\\_es.html](http://netbeans.org/index_es.html).
21. Alvarez, M.A.,. *Manual de jQuery*.
22. Pressman, R.S.,. *Ingeniería de Software: Un enfoque práctico, 2006*.
23. Tecnología y Synergix. [En línea] 22 de 02 de 2014. <http://synergix.wordpress.com/2008/07/10/modelo-de-dominio/>.
24. Salazar, L. Escribiendo Historias de Usuario Altamente Efectivas. [En línea] 12 de 08 de 2013. [Citado el: 22 de 02 de 2015.] [http://www.gazafatonarioit.com/2013/08/escribiendo-historias-de-usuario\\_12.html](http://www.gazafatonarioit.com/2013/08/escribiendo-historias-de-usuario_12.html).
25. Patron MVC. [En línea] 29 de 3 de 2015. <http://www.e-continua.com.mx/index.php/15->.
26. Patrones de Diseño. [En línea] <http://arlethparedes.wordpress.com/2012/08/27/patrones-de-arquitectura-vs-patrones-de-diseno/>.
27. Guía para Implementar Estándares de Codificación v 1.0. [En línea] 30 de 3 de 2015. [http://www.inf.utfsm.cl/~visconti/xp/Guia\\_>Estandares\\_Codificacion\\_2.doc](http://www.inf.utfsm.cl/~visconti/xp/Guia_>Estandares_Codificacion_2.doc).
28. PSR-1 —Codificación estándar básica. [En línea] [Citado el: 26 de marzo de 2015.] <http://www.php-fig.org/psr/psr-1/es/>.
29. Sommerville, I., *Ingeniería del software. Vol. 7*.