



## **Facultad 4**

# Suite Administrativa para Servidores de Entornos Productivos: Módulos PostgresManager y MySQLManager

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

**Autor**

Marino Borrero Sánchez

**Tutor**

Ing. Michel Miranda Cairo

La Habana, junio 2012

## Declaración de Autoría

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmamos la presente a los \_\_\_\_ días del mes de Junio del 2012.

\_\_\_\_\_  
Marino Borrero Sánchez

\_\_\_\_\_  
Michel Miranda Cairo

# Dedicatoria

Dedico esta tesis a mi madre por su apoyo y amor incondicional.

# Agradecimientos

Deseo agradecerle en primer lugar a Dios, Él me dio la vida, el intelecto y todo lo que tengo.

A mi madre porque ha sido mi amiga, mi madre, mi padre y le debo todo lo que soy.

A toda mi familia por estar a mi lado, por sentirse orgullosos de mí.

A Silvita por su amor y sus oraciones. Por estar siempre al tanto de mí.

A Frenkel por su amistad y por su apoyo espiritual.

A todos mis amigos de la universidad que han estado a mi lado por cinco años, en especial a Randy, Pepo, Edilberto, Leo y Geiber.

A mi tutor Michel por toda su ayuda.

Y a todas aquellas personas que de una manera u otra han influenciado en mi formación durante mi carrera, mis más sinceros agradecimientos.

# Resumen

Dentro del vertiginoso desarrollo de las tecnologías digitales hoy en día, la gestión de la configuración juega un papel fundamental para la obtención de un software de calidad.

Los gestores de configuración de servidores de bases de datos de los proyectos productivos de la Universidad de las Ciencias Informáticas (UCI) no cuentan con una herramienta que les permita de forma simple realizar la configuración de los servidores remotos. Es por ello que deben hacer su trabajo manualmente editando archivos de configuración y escribiendo comandos sobre una consola. Lo que en muchas ocasiones provoca la introducción de errores y la ralentización del proceso de desarrollo.

El objetivo del presente trabajo consistió en el desarrollo de dos módulos que facilitaron la tarea de los gestores de configuración de servidores de bases de datos (Postgres y MySQL) de la UCI. Dichos módulos son integrables a la Suite Administrativa para Servidores de Entornos Productivos (SASEP).

El lenguaje de programación utilizado fue C++, con el framework Qt 4 y el Entorno de Desarrollo Integrado Qt Creator 2.2. La metodología de desarrollo escogida fue XP.

**Palabras claves:** gestión de la configuración, proceso de desarrollo de software, SASEP, servidores de bases de datos Postgres y MySQL.

# Contenido

|  |           |
|--|-----------|
| <b>Introducción</b> .....  | <b>1</b>  |
| <b>Estructura capitular</b> .....  | <b>5</b>  |
| <b>Capítulo 1: Fundamentación Teórica</b> .....                            | <b>6</b>  |
| <b>1.1 Introducción</b> .....  | <b>6</b>  |
| <b>1.2 Servidor de Base de Datos</b> .....                                 | <b>6</b>  |
| 1.2.1 Sistema Gestor de Bases de Datos (SGBD) .....                        | 7         |
| 1.2.2- Arquitectura cliente/servidor.....                                  | 8         |
| 1.2.3 Servidor PostgreSQL .....  | 9         |
| 1.2.4 Servidor MySQL .....   | 9         |
| <b>1.3 Uso del Protocolo SSH</b> .....                                     | <b>11</b> |
| <b>1.4 Principales Herramientas Existentes</b> .....                       | <b>11</b> |
| 1.4.1. Webmin .....  | 11        |
| 1.4.2 PhpPgAdmin.....  | 12        |
| 1.4.3 PgAdmin III.....   | 12        |
| 1.4.4 MySQL Administrator .....  | 13        |
| <b>1.5 Lenguaje de Programación y Framework</b> .....                      | <b>14</b> |
| 1.5.1 Lenguaje C++ .....   | 14        |
| 1.5.2 Framework Qt.....  | 15        |
| <b>1.6 Herramientas y metodologías</b> .....                               | <b>16</b> |
| 1.6.1 Entorno de Desarrollo Integrado (IDE).....                           | 16        |
| 1.6.2 Metodologías de Desarrollo .....                                     | 18        |
| <b>1.7 Conclusiones</b> .....  | <b>21</b> |
| <b>Capítulo 2: Propuesta de la solución</b> .....                          | <b>22</b> |
| <b>2.1 Introducción</b> .....  | <b>22</b> |
| <b>2.2 Descripción de los procesos vinculados al campo de acción</b> ..... | <b>22</b> |
| 2.2.1 Propuesta del sistema.....   | 22        |
| <b>2.3 Modelo del Sistema</b> .....  | <b>23</b> |
| 2.3.1 Funcionalidades Módulo PostgresManager.....                          | 23        |
| 2.3.2 Funcionalidades Módulo MySQLManager .....                            | 24        |
| 2.3.2. Características del sistema .....                                   | 24        |
| <b>2.4 Personal relacionado con la aplicación</b> .....                    | <b>25</b> |
| <b>2.5 Flujos de trabajo para el desarrollo de la aplicación</b> .....     | <b>25</b> |
| 2.5.1 Fase de Exploración.....   | 25        |
| 2.5.1.1 Historias de Usuario .....   | 26        |
| <b>2.6 Planificación de la entrega.</b> .....                              | <b>34</b> |
| 2.6.1 Estimación de esfuerzo por historia de usuario. ....                 | 34        |
| 2.6.2 Planificación de las Iteraciones .....                               | 36        |

|   |           |
|---|-----------|
| <b>2.7 Diseño .....</b>   | <b>37</b> |
| 2.7.1 Tarjetas CRC.....   | 38        |
| <b>2.7.2 Uso de patrones de diseño.....</b>                                   | <b>39</b> |
| <b>2.7.2.1 Patrones GRASP .....</b>   | <b>39</b> |
| <b>2.7.3 Estándares de codificación .....</b>                                 | <b>40</b> |
| 2.7.3.1 Identación, llaves de apertura, de cierre y tamaño de las líneas..... | 40        |
| 2.7.3.2 Conversión de nomenclatura .....                                      | 40        |
| 2.7.3.3 Estructuras de control .....  | 41        |
| <b>2.8 Conclusiones.....</b>  | <b>41</b> |
| <b>Capítulo 3: Implementación y Pruebas .....</b>                             | <b>42</b> |
| <b>3.1 Introducción .....</b>   | <b>42</b> |
| <b>3.2 Fase Iteraciones .....</b>   | <b>42</b> |
| 3.3.1 Iteración 1.....  | 42        |
| 3.3.2 Iteración 2.....  | 44        |
| 3.3.2 Iteración 3.....  | 47        |
| <b>3.5 Pruebas.....</b>   | <b>48</b> |
| 3.5.1 Pruebas de aceptación.....  | 49        |
| 3.5.2 Pruebas de aceptación para PostgresManager.....                         | 49        |
| 3.5.3 Pruebas de aceptación para MySQLManager.....                            | 60        |
| <b>3.6 Conclusiones.....</b>  | <b>68</b> |
| <b>Conclusiones Generales .....</b>   | <b>69</b> |
| <b>Recomendaciones .....</b>  | <b>70</b> |
| <b>Referencias Bibliográficas .....</b>                                       | <b>71</b> |
| <b>Anexos.....</b>  | <b>73</b> |

# Introducción

Con el desarrollo de las Tecnologías de la Información y las Comunicaciones (TICs), el mundo ha experimentado un período de avances tecnológicos denominado Revolución Informática. Esta abarca desde mediados del siglo XX a la actualidad. Hoy en día se estima que no hay ningún fenómeno en la historia económica que haya progresado de manera más rápida, y que haya tenido un mayor impacto que la Revolución Informática, dando lugar al proceso de informatización de la sociedad. Tomando un rol protagónico en dicho proceso se encuentra la elaboración de programas informáticos, que está regida por el proceso de desarrollo de software. El mismo es muy variable en dependencia del contexto donde se aplique y tiene como propósito la producción eficaz y eficiente de un producto software que reúna los requisitos del cliente.

Una variable influyente a la hora de poner en marcha el proceso de desarrollo de software es la gestión de la configuración. La cual tiene como objetivo principal asegurar la integridad de los productos y servicios desarrollados.

Entre los beneficios de la implementación del proceso de gestión de la configuración se encuentran:

- Asegurar la correcta configuración del software.
- Proporcionar la capacidad de controlar los cambios.
- Reducir los sobreesfuerzos causados por los problemas de integridad.
- Garantizar que todo el equipo trabaja sobre una misma línea base de productos.

Evidentemente la correcta gestión de la configuración es imprescindible para la obtención de un producto de calidad. Dentro de las tareas fundamentales de la gestión de la configuración se encuentra la implementación de un conjunto de servicios claves para el proceso de desarrollo. Algunos de estos servicios son el control de versiones, servidor web y base de datos. Siendo este último de los más importantes en entornos de desarrollo de software.

Los servidores de bases de datos, según Menéndez Barzanallana del Departamento de Informática y Sistemas de la Universidad de Murcia, surgen con motivo de la necesidad de las empresas de manejar grandes y complejos volúmenes de datos, y que además requieren compartir la información con un conjunto de clientes de una



manera segura. (1) El servidor de base de datos deberá ofrecer soluciones de forma fiable, rentable y de alto rendimiento.

Dentro de los servidores de bases de datos más usados existen dos mundialmente conocidos por la robustez, simplicidad y seguridad que implementan. El primero de ellos es MySQL con más de seis millones de instalaciones en todo el mundo según datos mostrados en su sitio oficial. (2) Y Postgres, muy conocido por su versatilidad y poder.

La explosión de nuevos servicios ha hecho que cada vez más aplicaciones dependan de estos servidores, delegando en los mismos la responsabilidad de la gestión y almacenamiento de la información. Para lograr esto se hace imprescindible contar con herramientas que simplifiquen la tarea de la configuración, seguridad, creación y gestión de bases de datos al tiempo que proporcionan mecanismos de integración con otros sistemas y políticas de copias de seguridad. Entre las más usadas mundialmente se encuentran Webmin, PhpPgAdmin y MySQLAdministrator. Las cuales presentan algunas limitaciones como por ejemplo:

- Se ejecutan constantemente sobre el servidor, lo que genera la sobrecarga del mismo (Webmin y PhpPgAdmin).
- Requieren configuración previa del servidor para ser usados.
- No permiten configurar elementos específicos como puerto de escucha, restricciones de seguridad y memoria.

Estos problemas obligan a los gestores de configuración a tener un conocimiento elevado de conexiones de redes, seguridad y sistemas operativos. Además las configuraciones deben hacerse en la mayoría de los casos de forma manual editando archivos de configuración y escribiendo comandos sobre una consola. Lo que propicia la introducción de errores, búsqueda exhaustiva en Internet y la ralentización del proceso de desarrollo de software.

En Cuba no existen empresas desarrolladoras de software que sean de gran impacto a nivel mundial. La más conocida es la Universidad de las Ciencias Informáticas (UCI) que además de la docencia se dedica a la producción de bienes y servicios informáticos. Esta universidad, a partir de su creación en el año 2002, ha influido notablemente en la posición de Cuba en cuanto a la producción de software respecta.

La UCI no está exenta de los problemas antes expuestos que afectan el proceso de desarrollo de software. A esto se le suma el hecho de que los gestores de

configuración de los proyectos productivos son muchas veces estudiantes o personas inexpertas. Esto trae como consecuencia que aumente el riesgo de una incorrecta configuración de los servidores de bases de datos.

El grupo de desarrollo del proyecto Implantación y Soporte del Centro de Tecnologías para la Formación (FORTES) ha creado la Suite Administrativa para Servidores de Entornos Productivos (SASEP). Esta herramienta posee una arquitectura modular basada en plugins. Lo que posibilita que la adición de nuevas funcionalidades sea casi ilimitada. Y tiene como objetivo facilitar la tarea de los gestores de configuración al brindar una interfaz intuitiva para administrar servidores de forma remota.

SASEP no cuenta con módulos para la configuración de los servidores de bases de datos Postgres y MySQL. Dichos módulos serían de gran utilidad ya que Postgres y MySQL son ampliamente usados en la UCI.

A partir de entrevistas realizadas a gestores de configuración del Centro FORTES (ver Anexo 1), se detectó que no utilizan ninguna herramienta para la realización de su trabajo. Esto se debe entre otras causas a la configuración manual que requieren inicialmente y al alto consumo de recursos.

Los gestores también exponen lo complejo del trabajo que realizan, al no contar con una herramienta que permita de forma visual y remota configurar servidores de bases de datos Postgres y MySQL.

De ahí que el **problema de la investigación** sea: ¿Cómo facilitar la configuración de los servidores de bases de datos Postgres y MySQL en entornos productivos de la UCI?

El **objeto de estudio** es el proceso de configuración de servidores de bases de datos.

Para darle solución al problema de la investigación se define como **objetivo general**: Desarrollar dos módulos de configuración de servidores de bases de datos MySQL y Postgres integrables a la Suite Administrativa para Servidores de Entornos Productivos.

El **campo de acción** está enmarcado en el proceso de configuración de servidores de bases de datos Postgres y MySQL en entornos productivos de la UCI.

Para darle cumplimiento al objetivo general se trazaron los siguientes **objetivos específicos**:

1. Analizar los aspectos teóricos-conceptuales de la disciplina objeto de estudio y demás teorías que permitan definir posición del investigador.
2. Desarrollar el análisis y diseño de dos módulos para SASEP que permitan configurar servidores Postgres y MySQL.
3. Implementar los módulos de configuración de servidores de bases de datos Postgres y MySQL.
4. Realizar pruebas a los módulos de configuración.

Para darle cumplimiento a los objetivos específicos se definen las siguientes **tareas de la investigación:**

- Estudio de bibliografías sobre la configuración de servidores.
- Estudio de las herramientas para la administración y configuración de servidores de bases de datos.
- Análisis de metodologías y herramientas para el desarrollo.
- Descripción del funcionamiento del servidor.
- Obtención de los artefactos generados durante el proceso de desarrollo.
- Implementación de la solución.
- Validación de la solución propuesta.

### **Métodos teóricos**

**Analítico-Sintético:** para el análisis de documentos y materiales sobre la configuración de servidores de bases de datos, de manera que se procese la información y se elaboren conclusiones.

**Histórico-Lógico:** para el estudio de la evolución y desarrollo de la configuración de servidores de bases de datos.

### **Métodos empíricos**

**Entrevistas:** para obtener información de los administradores y gestores sobre el trabajo que realizan y corroborar que realmente existe el problema.

# Estructura capitular

**Capítulo 1. Fundamentación Teórica:** Se observan los conceptos fundamentales referentes a la administración y configuración de servidores de bases de datos. Se aborda además un breve estudio de los lenguajes de programación más usados. Por último se describen las tecnologías, herramientas y metodologías a utilizar para dar solución al problema en cuestión.

**Capítulo 2. Propuesta de la solución:** Desarrollo de la propuesta de solución. Se definen las principales características que deben cumplir los módulos. Finalmente se enumeran las funcionalidades que deben implementar ambos módulos.

**Capítulo 3. Implementación y prueba:** Se realiza la descripción de la herramienta desde el enfoque de la programación. Además, se valida la propuesta a partir de pruebas realizadas al sistema.

# Capítulo 1: Fundamentación Teórica

## 1.1 Introducción

La configuración de servidores de bases de datos en entornos productivos de la UCI juega un papel fundamental en el proceso de desarrollo de software. Gracias a los servidores de bases de datos se facilita la manipulación de grandes cúmulos de información de forma fiable y coherente.

Para lograr un adecuado funcionamiento del servidor los gestores y administradores deben configurar los servidores editando y copiando archivos, escribiendo muchas líneas de comando además de usar varias herramientas. Esto se debe a que no se conoce una aplicación que permita de forma visual y remota realizar las configuraciones más importantes de los servidores de bases de datos. De las herramientas estudiadas ninguna por si sola satisface las necesidades de los gestores de configuración.

En el presente capítulo se realiza un estudio sobre los conceptos fundamentales de la administración y configuración de servidores de bases de datos. También se lleva a cabo un análisis del estado del arte y por último se presentan las herramientas, tecnologías y metodologías a utilizar.

## 1.2 Servidor de Base de Datos

Para el desarrollo de este trabajo se debe comprender el término servidor de base de datos como concepto. La Computer Audio Video Systems Integrator (CAVSI) plantea que *“un servidor de bases de datos relacionales es un sistema que utiliza la arquitectura cliente/servidor que proporciona servicios de gestión, administración y protección de la información a través de una red de computadoras, gobernadas por una serie de protocolos definidos y a la que pueden acceder los usuarios, de modo concurrente, a través de lo que se denomina aplicaciones clientes”*. (3)

Estos servidores tienen la capacidad de brindar solución a muchos de los problemas relacionados con el almacenamiento de la información en las empresas. Esto se debe a que pueden manejar grandes volúmenes de información de una manera estable, fiable, coherente y segura en un entorno heterogéneo de trabajo y de necesidades de información.

La información será almacenada en tablas que se componen de un conjunto de campos que describen su contenido, denominados columnas, y se relacionan entre sí a través de un conjunto definido de claves. El sistema de la base de datos, será el

responsable de mostrar la información requerida a través de conjuntos de datos planos o cursores, independizando las relaciones establecidas y la arquitectura de la base de datos de la necesidad de información del usuario. Mediante mecanismos de control de transacciones basados en reglas (disipadores) de definición del tipo de entrada de datos y reglas de validación de las entradas de datos, el sistema es capaz de proteger la información. A través de complejos sistemas de indexación, estos sistemas serán capaces de ordenar y acelerar las consultas a la información requerida. Cuanto mejor se indexen los datos, más rápidas se realizarán las consultas. Por último, y como un factor muy importante de cara al diseño de bases de datos, los sistemas deben proporcionar la posibilidad de automatizar operaciones de acceso, filtrado y control de los datos, a través de los procedimientos almacenados.

### 1.2.1 Sistema Gestor de Bases de Datos (SGBD)

Según CAVSI un SGBD se puede definir como una colección de programas destinados a servir de interfaz entre la base de datos, el usuario y las aplicaciones. En su composición se encuentran tres elementos fundamentales:

- un lenguaje de definición de datos
- un lenguaje de manipulación de datos
- un lenguaje de consulta

Todo ello permite definir los datos a distintos niveles de abstracción y manipular dichos datos, garantizando la seguridad e integridad de los mismos. Algunos ejemplos de SGBD son Oracle, DB2, PostgreSQL, MySQL, y MS SQL Server. (4)

Algunas características de los SGBD son:

- **Abstracción de la información:** Los SGBD no brindan detalle acerca del almacenamiento físico de los datos. No es relevante la cantidad de archivos que ocupa la base de datos, este hecho se hace transparente al usuario. De esta forma se pueden definir varios niveles de abstracción.
- **Independencia:** La independencia de los datos consiste en la capacidad de modificar el esquema (físico o lógico) de una base de datos sin tener que realizar cambios en las aplicaciones que se sirven de ella.
- **Consistencia:** En aquellos casos en los que no se ha logrado eliminar la redundancia, será necesario vigilar que aquella información que aparece repetida se actualice de forma coherente, es decir, que todos los datos repetidos se

actualicen de forma simultánea. Por otra parte, la base de datos representa una realidad determinada que tiene establecidas condiciones. En los SGBD existen herramientas que facilitan la programación de este tipo de condiciones.

- **Seguridad:** La información almacenada en una base de datos puede llegar a tener un gran valor. Los SGBD deben garantizar que esta información se encuentra segura frente a usuarios malintencionados, que intenten leer información privilegiada; frente a ataques que deseen manipular o destruir la información. Normalmente, los SGBD disponen de un complejo sistema de permisos a usuarios y grupos de usuarios, que permiten otorgar diversas categorías de permisos.
- **Integridad:** Se trata de adoptar las medidas necesarias para garantizar la validez de los datos almacenados. Es decir, se trata de proteger los datos ante fallos de hardware, datos introducidos por usuarios descuidados, o cualquier otra circunstancia capaz de corromper la información almacenada. Los SGBD proveen mecanismos para garantizar la recuperación de la base de datos hasta un estado consistente conocido en forma automática.
- **Respaldo:** Los SGBD deben proporcionar una forma eficiente de realizar copias de respaldo de la información almacenada en ellos, y de restaurar a partir de estas copias los datos que se hayan podido perder.
- **Control de la concurrencia:** En la mayoría de entornos lo más habitual es que sean muchas las personas que acceden a una base de datos, bien para recuperar información, bien para almacenarla. Y es también frecuente que dichos accesos se realicen de forma simultánea. Así pues, un SGBD debe controlar este acceso concurrente a la información, que podría derivar en inconsistencias.

Todo sistema gestor de base de datos utiliza la arquitectura cliente/servidor, a continuación se brinda una breve explicación de la misma.

### 1.2.2- Arquitectura cliente/servidor

Esta arquitectura es en esencia un cliente que realiza peticiones a otro programa (el servidor) que le da respuesta. Aunque esta idea se puede aplicar a programas que se ejecutan sobre una sola computadora es más ventajosa en un sistema operativo multiusuario distribuido a través de una red de computadoras. (5)

### **1.2.3 Servidor PostgreSQL**

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional basado en el proyecto Postgres, de la Universidad de Berkeley. El director de este proyecto fue el profesor Michael Stonebraker, y estuvo patrocinado por Defense Advanced Research Projects Agency (DARPA), el Army Research Office (ARO) y el National Science Foundation (NSF). (6)

Entre las principales características de este gestor de bases de datos se puede mencionar la implementación del estándar SQL92/SQL99. También soporta distintos tipos de datos como fecha, monetarios, elementos gráficos, datos sobre redes y cadenas de bits. Permite la creación de tipos propios, la declaración de funciones propias y la definición de disparadores. Incluye herencia entre tablas, lo que lo sitúa entre los gestores objeto-relacionales. Admite la gestión de usuarios, como también los permisos asignados a cada uno de ellos. Es un gestor estable, multiplataforma, potente y flexible. Permite la duplicación de bases de datos maestras en múltiples sitios de réplica. Cuenta con funciones de compatibilidad para ayudar en la transición desde otros sistemas menos compatibles con SQL. Permite el acceso encriptado vía SSL. Está disponible en sistemas basados en Unix y en Windows. Es un gestor flexible para hacer investigación y desarrollo sin costos de licenciamiento.

Tiene un mantenimiento y ajuste mucho menor que los productos de los proveedores comerciales, conservando todas las características, estabilidad y rendimiento. El código fuente está disponible para todos sin costo alguno. Posee herramientas gráficas de alta calidad para administrar las bases de datos y para hacer diseño de bases de datos. PostgreSQL ha agregado permisos a esquemas, funciones y otros objetos y nuevas opciones de configuración para aumentar la granularidad del control del administrador sobre la seguridad. (6)

### **1.2.4 Servidor MySQL**

MySQL es un sistema de gestión de bases de datos relacional, licenciado bajo la General Public License (GPL) de la GNU. Su diseño multihilo le permite soportar una gran carga de forma muy eficiente. MySQL fue creada por la empresa sueca MySQL AB, que mantiene el copyright del código fuente del servidor SQL, así como también de la marca. (2)

Este gestor de bases de datos es uno de los más usados el mundo del software libre, debido a su gran rapidez y facilidad de uso. (7) Esta gran aceptación es debida, en parte, a la existencia de gran cantidad de librerías y otras herramientas que permiten



su uso a través de diversos lenguajes de programación, además de su fácil instalación y configuración.

Entre sus principales características están:

- Aprovecha la potencia de sistemas multiprocesador, gracias a su implementación multihilo.
- Soporta varios tipos de datos para las columnas.
- Dispone de APIs (Application Programming Interfaces) en gran cantidad de lenguajes (C, C++, Java y PHP).
- Gran portabilidad entre sistemas.
- Soporta hasta 32 índices por tabla.
- Gestión de usuarios y contraseñas, manteniendo un buen nivel de seguridad en los datos.

Entre sus ventajas se puede mencionar que MySQL es libre. Significa que es permitido usar y modificar el software. Es posible el acceso al software MySQL desde internet y usarlo sin pagar nada en la versión gratuita y si se desea, se puede estudiar el código fuente y cambiarlo para adaptarlo a necesidades específicas.

MySQL se desarrolló originalmente para manejar grandes bases de datos mucho más rápido que las soluciones de gestores de bases de datos existentes y ha sido usado con éxito en entornos de producción de alto rendimiento durante varios años. (8) Trabaja en entornos cliente/servidor o incrustados; el software de bases de datos MySQL consiste en una aplicación cliente/servidor que trabaja con diferentes programas y bibliotecas, herramientas administrativas y diversas interfaces de programación para aplicaciones (APIs). También se proporciona el MySQL Server como biblioteca incrustada, que se puede incluir en una aplicación para obtener un producto más pequeño, rápido y fácil de administrar.

También MySQL tiene un buen control de acceso de usuarios y seguridad en los datos. Además de una Integración perfecta con el lenguaje PHP (Preprocesador de Hipertexto). Se admite la replicación de bases de datos trabajando con servidores MySQL maestros-esclavos. También se permite trabajar con el entorno MySQL Clúster para dar alta disponibilidad y rendimiento al sistema. Permite además el uso de MyODBC, que proporciona a MySQL soporte para programas clientes que usen conexiones ODBC (Open Database Connectivity). MySQL Soporta múltiples modos asignados para comportarse como otros gestores de bases de datos, definiendo la

validación o no de los valores erróneos o incorrectos y la forma en que se ingresan los datos. (9)

Una forma segura y rápida de conectarse a los servidores de bases de datos para realizar la administración y configuración de los mismos es a través del protocolo SSH.

### **1.3 Uso del Protocolo SSH**

SSH (Secure SHell) es un protocolo que facilita las comunicaciones seguras entre dos sistemas usando una arquitectura cliente/servidor y que permite a los usuarios conectarse a un host remotamente. SSH encripta la sesión de conexión, haciendo muy difícil que alguien pueda obtener contraseñas no encriptadas. (9)

El protocolo SSH encripta todo lo que envía y recibe, esto posibilita que se pueda utilizar para asegurar protocolos inseguros. Para realizar esta tarea se usa una técnica llamada reenvío por puerto, la misma incrementa la seguridad del sistema en general y de los datos.

El protocolo SCP (Secure Copy), creado con el objetivo de transferir archivos de forma segura en una red, proporciona la funcionalidad necesaria para la transmisión y manipulación de archivos sobre un flujo de datos fiable. Los datos viajan en paquetes cifrados mediante una conexión por SSH. Por lo tanto, la seguridad no la provee directamente el protocolo SCP, sino SSH.

### **1.4 Principales Herramientas Existentes**

#### **1.4.1. Webmin**

Webmin es una herramienta de administración web gráfica e intuitiva. Fue escrita por Jaime Cameron en lenguaje Perl. Es muy portable ya que soporta treinta y cinco distribuciones de UNIX y una gran cantidad de distribuciones de Linux. Es fácil ampliar y darle nuevas posibilidades ya que es abierto y está bien documentado. Provee una interfaz gráfica para prácticamente cualquier tarea que haya que hacer sobre un sistema basado en UNIX. Es importante aclarar que Webmin no modifica opciones desconocidas para él en los archivos de configuración y que no es necesario tener un entorno gráfico, ya que se puede utilizar con navegadores modo texto. (10) Las principales características de este entorno administrativo son:

- El cambio de contraseñas de usuarios.
- Planificación de copias de seguridad de directorios importantes.
- Asignación y administración de cuotas de disco.

- Configuración de los servidores Samba, POP, Apache, MySQL, Postgres y Servidor Proxy Squid.
- Análisis de históricos y monitorización del ancho de banda.

Sin embargo no posibilita configurar de forma remota en el servidor de bases de datos elementos como cambiar el puerto de escucha, correr un script sobre una base de datos, configuraciones de memoria ni reglas de conexión. Otras de sus desventajas es que sobrecarga el servidor al estar ejecutándose constantemente en el mismo.

#### **1.4.2 PhpPgAdmin**

PhpPgAdmin es una poderosa herramienta de administración basada en un interfaz Web para bases de datos PostgreSQL. Dispone de soporte para procedimientos almacenados, triggers y vistas. Las versiones actualizadas se desarrollan junto a PostgreSQL. PhpPgAdmin estuvo basado en phpMyAdmin, pero hoy día ya no comparte código con él e incluso provee las mismas funcionalidades.(11)

PhpPgAdmin tiene varias desventajas como por ejemplo que se ejecuta constantemente en el servidor provocando la sobrecarga del mismo, no permite realizar configuraciones de reglas de conexiones, memoria ni encriptación además de requerir una configuración manual del servidor remoto inicialmente.

#### **1.4.3 PgAdmin III**

PgAdmin III es una aplicación gráfica para gestionar el servidor de bases de datos PostgreSQL, siendo la más completa y popular con licencia Open Source. Está escrita en C++ usando la librería gráfica multiplataforma wxWidgets, lo que permite que se pueda usar en Linux, FreeBSD, Solaris, Mac OS y Windows. Es capaz de gestionar versiones a partir de la PostgreSQL 7.3 ejecutándose en cualquier plataforma, así como versiones comerciales de PostgreSQL como Pervasive Postgres, EnterpriseDB, Mammoth Replicator y SRA PowerGres. Además de estar diseñada para ejecutarse en la mayoría de los Sistemas Operativos. (12)

Esta aplicación también incluye un editor SQL con resaltado de sintaxis, un editor de código de la parte del servidor, un agente para lanzar scripts programados y soporte para el motor de replicación Slony-I. La conexión al servidor puede hacerse mediante conexión TCP/IP o Unix Domain Sockets, y puede encriptarse mediante SSL para mayor seguridad.

A pesar de sus múltiples ventajas esta herramienta posee algunas deficiencias que no se pueden obviar, ejemplo de ello es que no localiza automáticamente los archivos de

configuración, requiriendo además que se establezcan permisos especiales a los mismos. PgAdmin III tampoco visualiza el método de autenticación en la ventana de reglas de conexión sino que lo hace desde una interfaz diferente dificultando el trabajo. Otra de sus desventajas es que no se gestionan los errores al introducir una configuración incorrecta ni se permite volver a la configuración por defecto.

#### **1.4.4 MySQL Administrator**

MySQL Administrador es un software de administración de servidores de Bases de Datos de MySQL que ha creado MySQL AB. Se trata de un software multiplataforma, que por el momento se encuentra disponible para Linux y Microsoft Windows y que cuenta con un entorno gráfico de usuario muy intuitivo. (13)

Este producto suple las carencias que tiene MySQL Control Center en el área de Administración de servidores. MySQL Control Center ha quedado obsoleto y no sigue desarrollándose. Se ha sustituido por el conjunto de programas MySQL Administrator y MySQL Query Browser.

MySQL Administrador es una herramienta que permite realizar tareas administrativas sobre servidores de MySQL incluyendo:

- La configuración de las opciones de inicio de los servidores.
- Inicio y detención de servidores.
- Monitorización de conexiones al servidor.
- Administración de usuarios.
- Monitorización del estado del servidor, incluyendo estadísticas de uso.
- Visualización de los logs del servidor.
- Gestión de copias de seguridad y recuperaciones.
- Visualización de catálogos de datos.

Esta herramienta resulta muy útil, sin embargo desde el punto de vista de configuración aún posee varias deficiencias como por ejemplo no permite realizar las configuraciones de forma remota automáticamente, no configura las distintas variables relacionadas con memoria, no permite definir el tamaño máximo de los logs de seguridad ni el tiempo de vida de los mismos. Tampoco permite definir el tiempo máximo para la autenticación de los usuarios entre otras opciones. Otra de sus deficiencias es su carácter privativo.

## 1.5 Lenguaje de Programación y Framework

SASEP es una herramienta que permite la administración remota y visual de servidores, desarrollada en el Departamento de Implantación y Soporte Técnico del Centro de Tecnologías para la Formación en la UCI. Está basada en una arquitectura modular extensible a través de plugins, los cuales posibilitan adicionarle nuevas funcionalidades para la administración de diferentes servicios.

Toda la comunicación de SASEP con el servidor es tramitada vía ssh, lo que la convierte en una conexión con un nivel excelente de seguridad. Así mismo proporciona una interfaz, como se muestra en la figura 1, que debe ser implementada por los plugins que se desarrollen.

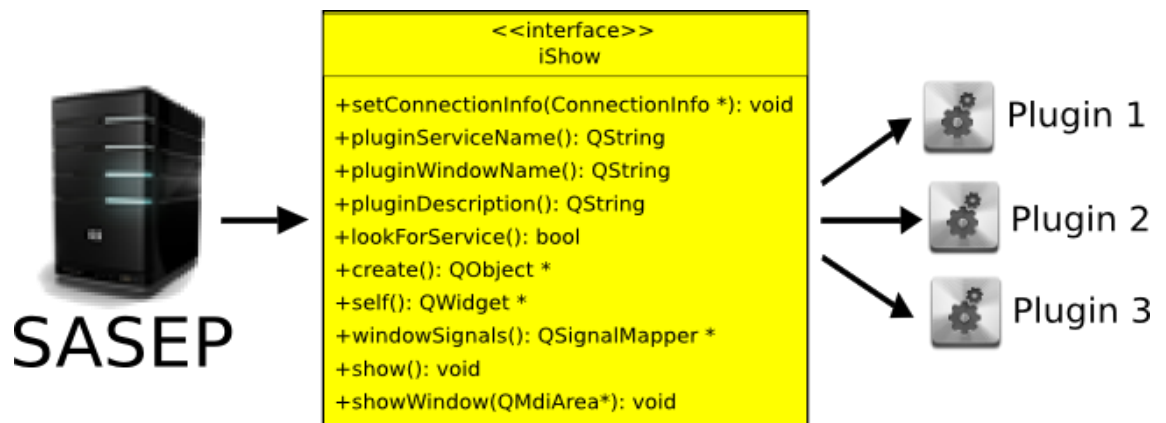


Figura 1. Interfaz para la implementación de nuevos plugins.

Al estar desarrollada en el lenguaje C++ y el framework Qt 4.7, es necesario para un correcto desempeño que los plugins sean implementados siguiendo estas pautas. Por tal motivo en esta investigación no se analizan otras tecnologías.

A continuación se detallan algunas de las características del lenguaje C++ y del framework Qt.

### 1.5.1 Lenguaje C++

C++ es un lenguaje de programación diseñado a mediados de 1980 por Bjarne Stroustrup. La intención de su creación fue extender al exitoso lenguaje de programación C con mecanismos para la manipulación de objetos. Este lenguaje es multiparadigma, o sea que permite la programación estructurada así como la orientada a objetos. Actualmente existe un estándar, denominado ISO C++, al que se han adherido la mayoría de los fabricantes de compiladores más modernos. Una

particularidad de C++ es la posibilidad de redefinir los operadores y de poder crear nuevos tipos que se comporten como tipos fundamentales.

Posee un conjunto muy completo de instrucciones de control, permite la agrupación de instrucciones, incluye el concepto de puntero y los argumentos de las funciones se transfieren por su valor. Además la Entrada/Salida se realiza a través de una biblioteca de funciones. También C++ tiene la ventaja de ocupar poco espacio en el disco y usa comandos breves. Debido a que se puede manejar actividades a bajo nivel, posibilita crear programas muy eficientes. Por otra parte este lenguaje posee una gran portabilidad, haciéndose adaptable a una gran variedad de computadoras y sistemas.

C++ es uno de los lenguajes más empleados en la actualidad, es por ello que posee un gran número de usuarios y existe abundante bibliografía del mismo. Además existen gran cantidad de herramientas, librerías, depuradores y compiladores para el mismo. (14)

#### **1.5.2 Framework Qt**

Qt es una biblioteca multiplataforma ampliamente usada en todo el mundo para el desarrollo de aplicaciones visuales, también se usa mucho para crear aplicaciones de consola para servidores. (15)

Qt utiliza el lenguaje de programación C++ de forma nativa, sin embargo se puede usar en varios lenguajes de programación a través de bindings. También es usada en sistemas para automoción, aeronavegación y aparatos domésticos.

Funciona en todas las principales plataformas, y tiene un amplio apoyo. Qt posee funciones para acceder a bases de datos mediante código SQL, así como uso de XML. Permite además la gestión de hilos, brinda soporte de red y tiene una API multiplataforma unificada. Esta posee funciones para la manipulación de archivos y ficheros, además de estructuras de datos tradicionales.

Qt usa la licencia GNU Lesser General Public License (LGPL) y es software libre y código abierto.

El framework QT es sin dudas uno de los frameworks más potentes de C++, es robusto y posee una gran cantidad de funciones predefinidas que apoyan la implementación.

## **1.6 Herramientas y metodologías**

### **1.6.1 Entorno de Desarrollo Integrado (IDE)**

Un entorno de desarrollo integrado o IDE (acrónimo en inglés de integrated development environment), es un programa informático compuesto por un conjunto de herramientas de programación. (16)

Un IDE consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (GUI). Los IDEs pueden ser aplicaciones por sí solas o pueden ser parte de aplicaciones existentes. Proveen un marco de trabajo amigable para la mayoría de los lenguajes de programación tales como C++, Python, Java, C#, Delphi y Visual Basic. En algunos lenguajes, un IDE puede funcionar como un sistema en tiempo de ejecución, en donde se permite utilizar el lenguaje de programación en forma interactiva, sin necesidad de trabajo orientado a archivos de texto.

Es posible que un mismo IDE pueda funcionar con varios lenguajes de programación. Este es el caso de Eclipse, al que mediante plugins se le puede añadir soporte de lenguajes adicionales.

#### **1.6.1.1 Eclipse**

Eclipse fue desarrollado originalmente por la compañía IBM. Es un IDE de código abierto y multiplataforma. Puede ser utilizado para una amplia gama de lenguajes gracias a su arquitectura basada en plugins. (17) Este mecanismo de módulos le permite a Eclipse extenderse usando otros lenguajes de programación como son C, C++ y Python. También le permite a Eclipse trabajar con lenguajes para procesado de texto como LaTeX, aplicaciones en red como Telnet y sistema de gestión de base de datos. Eclipse también posee herramientas para la refactorización y análisis del código. Otras de sus características son:

- Posee un editor de código con resaltado de sintaxis.
- Compilación en tiempo real.
- Posee herramientas para pruebas unitarias y para control de versiones.
- Asistentes para creación de proyectos y clases.

#### **1.6.1.2 Code::Blocks**

Code::Blocks es un entorno de desarrollo integrado libre y multiplataforma para el desarrollo de programas en lenguaje C y C++. Aunque está basado en la plataforma de interfaces gráficas WxWidgets, también puede usar la plataforma Qt a través de un plugin o complemento. (18)

Code::Blocks es un IDE construido como un núcleo altamente expansible mediante complementos (plugins). Hoy en día la mayor parte de la funcionalidad viene provista por los complementos predeterminados. No es un IDE autónomo que acepta complementos, sino que es un núcleo abstracto donde los complementos se convierten en una parte vital del sistema. Esto lo convierte en una plataforma muy dinámica y potente, no solo por la facilidad con que puede incluirse nueva funcionalidad, sino por la capacidad de poder usarla para construir otras herramientas de desarrollo tan solo añadiendo complementos.

Entre sus características más destacadas se pueden citar:

- Coloreo de sintaxis configurable.
- Tabulación inteligente de código.
- Autocompletado de código.
- Vista de argumentos de una función.
- Selección múltiple de funciones sobrecargadas.
- Generación de clases.
- Búsquedas avanzadas de cadenas dentro de archivos.
- Sistema de construcción rápido.
- Soporte para compilación en paralelo.
- Dependencias entre proyectos dentro del espacio de trabajo.

### 1.6.1.3 KDevelop

KDevelop es un entorno de desarrollo integrado para sistemas GNU/Linux y otros sistemas Unix, publicado bajo licencia GPL, orientado al uso bajo el entorno gráfico KDE, aunque también funciona con otros entornos como Gnome. Soporta distintos lenguajes de programación. Entre ellos se encuentran C, C++ y PHP.

Algunas de sus características son:

- Editor de código fuente con destacado de sintaxis e indentado automático
- Gestión de diferentes tipos de proyectos, como *CMake*, *Automake* y *qmake* (para proyectos basados en la biblioteca Qt )
- Navegador entre clases de la aplicación.
- Front-end para gcc, el conjunto de compiladores de GNU.
- Front-end para el depurador de GNU.
- Asistentes para generar y actualizar las definiciones de las clases y el framework de la aplicación.
- Completado automático del código en C y C++.
- Permite control de versiones.



#### **1.6.1.4 Qt Creator**

Qt Creator es un IDE multiplataforma, diseñado para hacer que el desarrollo en C++ de la aplicación Qt sea más rápido y fácil.

Qt Creator es distribuido bajo tres tipos de licencias:

- Qt Commercial Developer License
- Qt GNU LGPL v. 2.1
- Qt GNU GPL v. 3.0

Qt Creator no quiere ser un reemplazo de Eclipse ni Visual Studio, sino un IDE ligero pensado especialmente para el desarrollo en múltiples plataformas: Windows, Linux (desde la versión 2.6) y Mac OSX (desde 10.4 en adelante). (15)

Principales características de Qt Creator:

- Posee un avanzado editor de código C++.
- Soporta los lenguajes: C, C++, Ada, Pascal, Perl, PHP y Ruby.
- Posee también una GUI integrada y diseñador de formularios.
- Cuenta con herramientas para proyectos y administración.
- Ayuda sensible al contexto integrado.
- Depurador visual.
- Resaltado y auto-completado de código.
- Soporte para refactorización de código.

Luego de analizar los IDEs más conocidos que soportan la librería Qt, se decide usar QT Creator 2.2. El mismo fue creado específicamente para el trabajo con el framework Qt 4. También posee una amplia documentación y existe vasta experiencia en su uso además de las múltiples ventajas mencionadas anteriormente.

#### **1.6.2 Metodologías de Desarrollo**

Una metodología de desarrollo de software es un conjunto de pasos y procedimientos que deben seguirse para desarrollar software. Se puede definir dicha metodología como un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos software. Se debe destacar que el éxito del producto depende en gran parte de la metodología seleccionada por el equipo de desarrollo. En la actualidad existen dos tipos de metodologías de desarrollo: Las Metodologías Tradicionales o Pesadas y Las Metodologías Ágiles o Livianas. Cualquiera que sea la

metodología seleccionada, debe ser capaz de maximizar el potencial del equipo de desarrollo, además de aumentar la calidad del producto con los recursos y tiempos establecidos. (19)

### **1.6.2.1 Scrum**

Scrum es una metodología de desarrollo de software basada en un proceso iterativo e incremental utilizado comúnmente en entornos basados en el desarrollo ágil de software.

Es un modelo de referencia que precisa un conjunto de prácticas y roles, y que puede tomarse como punto de partida para definir el proceso de desarrollo que se ejecutará durante un proyecto. Los roles principales en Scrum son el ScrumMaster, que mantiene los procesos y trabaja de forma similar al director de proyecto, el ProductOwner, que representa a los stakeholders (clientes externos o internos), y el Team que incluye a los desarrolladores.

Durante cada sprint, un periodo entre 15 y 30 días, el equipo crea un incremento de software potencialmente entregable. El conjunto de características que forma parte de cada sprint viene del Product Backlog, que es un conjunto de requisitos de alto nivel priorizados que definen el trabajo a realizar. Los elementos del Product Backlog que forman parte del sprint se determinan durante la reunión de Sprint Planning. Durante esta reunión, el Product Owner identifica los elementos del Product Backlog que quiere ver completados y los hace del conocimiento del equipo. Entonces, el equipo determina la cantidad de ese trabajo que puede comprometerse a completar durante el siguiente sprint. Durante el sprint, nadie puede cambiar el Sprint Backlog, lo que significa que los requisitos están congelados durante el sprint. (20)

Scrum permite la creación de equipos organizados promoviendo la co-localización de todos los miembros del equipo, y la comunicación verbal entre todos los miembros y disciplinas involucrados en el proyecto.

Un principio fundamental de Scrum es el reconocimiento de que durante un proyecto los clientes pueden cambiar de idea sobre lo que quieren y necesitan (requirements churn), y que los desafíos impredecibles no pueden ser fácilmente enfrentados de una forma predictiva y planificada. Consecuentemente, Scrum adopta una aproximación pragmática, aceptando que el problema no puede ser completamente entendido o definido, y centrándose en maximizar la capacidad del equipo de entregar rápidamente y responder a requisitos emergentes.

Existen varias implementaciones de sistemas para gestionar el proceso de Scrum, que van desde notas amarillas y pizarras hasta paquetes de software. Una de las mayores ventajas de Scrum es que es muy fácil de aprender, y requiere muy poco esfuerzo para comenzarse a utilizar. (21)

#### **1.6.2.2 Programación Extrema (XP).**

La Programación Extrema es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. Se basa en la realimentación continua entre el cliente y el equipo de desarrollo.(22) XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico. Esta metodología se caracteriza por:

- Desarrollo iterativo e incremental que permite pequeñas mejoras.
- Pruebas unitarias continuas, frecuentemente repetidas y automatizadas, incluyendo pruebas de regresión.
- Programación por parejas permitiendo que el código sea revisado y discutido mientras se escribe proponiendo que la mayor calidad del código escrito de esta manera sea más importante que la posible pérdida de productividad inmediata. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento.
- Frecuente interacción del equipo de programación con el cliente o usuario y corrección de todos los errores antes de añadir nueva funcionalidad.
- Refactorización del código. Se basa en la reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio y así aumentar su legibilidad y mantenimiento pero sin modificar su comportamiento.
- Propiedad del código compartida, este método promueve el que todo el personal pueda corregir y extender cualquier parte del proyecto.
- Simplicidad en el código.
- Mayor comunicación, resulta más fácil identificar qué se debe y qué no se debe hacer. Mientras más simple es el sistema, menos tendrá que comunicar sobre

este, lo que lleva a una comunicación más completa, especialmente si se puede reducir el equipo de programadores.

Luego de haber analizado estas dos metodologías ágiles se decide seleccionar XP ya que se ajusta mejor al problema debido a las características y ventajas del mismo expuestas anteriormente.

### **1.7 Conclusiones**

En este capítulo se analizaron los conceptos claves en la configuración de servidores de bases de datos. Luego de haberse realizado un estudio de las principales herramientas existentes, se concluye afirmando que las mismas no satisfacen las necesidades de los gestores de configuración de la UCI. Esto se debe a que no permiten la configuración directa del servidor de forma remota, consumen, en algunos casos, recursos del servidor y no gestionan los errores introducidos por los usuarios en el proceso de configuración.

Luego del análisis de tecnologías, herramientas y metodologías de desarrollo, se decide el uso de:

- C++ como lenguaje de programación ya que es requerido por SASEP para un correcto funcionamiento del sistema.
- QT 4 como framework de desarrollo debido a que posee un conjunto amplio de funciones y librerías que apoyan la implementación. También es el framework usado por SASEP.
- QT Creator 2.2 como Entorno de Desarrollo Integrado, debido a sus amplias ventajas para programar aplicaciones para Linux y en específico aplicaciones para configurar servidores.
- XP como metodología de desarrollo debido a que se centra en el desarrollo del software y es muy versátil.

# Capítulo 2: Propuesta de la solución

## 2.1 Introducción

El presente capítulo está enmarcado en las dos primeras fases de XP (Exploración y Planificación). Se realiza el esbozo de la propuesta de solución al problema mediante el desarrollo de dos aplicaciones capaces de configurar los servidores de bases de datos Postgres y MySQL. Se realizará en primer lugar la descripción de los procesos vinculados al campo de acción, luego se presentarán las tareas de implementación y las características del sistema. Por último se presentarán los artefactos generados.

## 2.2 Descripción de los procesos vinculados al campo de acción

Según el presente estudio, no se tiene conocimiento de la existencia de una herramienta que permita la configuración de los servidores de bases de datos Postgres y MySQL de forma visual y remota. Existen herramientas que resuelven el problema pero no en su totalidad por lo que los gestores de configuración deben realizar su trabajo usando varias herramientas y haciendo muchas de las configuraciones de forma manual. Es por ello que se propone la creación de dos módulos de configuración de bases de datos para SASEP que permitan la configuración del servidor de bases de datos mediante una interfaz visual y que permitan además realizar dicha configuración de forma remota mediante el protocolo SSH.

### 2.2.1 Propuesta del sistema

Luego de analizar cómo se realiza el proceso de configuración de los servidores de bases de datos Postgres y MySQL en entornos productivos de la UCI, se ha concluido que se debe automatizar el mismo para facilitar el trabajo de los gestores y evitar la introducción de errores. Se propone desarrollar dos módulos de configuración para los servidores Postgres y MySQL integrables a SASEP.

Estos módulos posibilitarán configurar de forma visual las opciones más importantes en el servidor de base de datos. Este trabajo será posible porque la aplicación debe conectarse vía SSH al servidor y acceder a los ficheros de configuración y los modificará de acuerdo a las necesidades del gestor de configuración. La aplicación también utilizará comandos del propio servidor, pero lo hará de forma interna e invisible al usuario (gestor). Entre las opciones que manejará la aplicación se encuentra la seguridad, memoria, configuraciones de acceso y la ejecución de scripts.

Dentro del apartado de seguridad se podrá definir quienes se conectan al servidor, que tipo de conexión pueden usar, de qué forma pueden autenticarse, si se encripta las contraseñas o no, la realización de salvadas, entre otras opciones. El sistema también podrá configurar la cantidad de memoria destinada para el servidor, la memoria de trabajo (necesaria para ejecutar ciertas funciones), memoria de mantenimiento, y el tamaño de la caché efectiva (utilizada en la optimización de lectura de datos). Además existirá la posibilidad de cambiar el puerto de escucha, la cantidad máxima de conexiones entre otras opciones importantes en el servidor. También se podrá mostrar la configuración actual del servidor pues este es el punto de partida para que el gestor realice su trabajo de forma más sencilla.

### **2.3 Modelo del Sistema**

Las historias de usuario pueden estar compuestas por una o varias funcionalidades. Para el presente trabajo se propuso un total de cuarenta y cuatro funcionalidades que a continuación se enumeran.

#### **2.3.1 Funcionalidades Módulo PostgresManager**

1. Conectar a servidor remoto mediante SSH.
2. Configurar acceso al servidor.
3. Configurar el número máximo de clientes que pueden conectarse al servidor.
4. Configurar el tamaño de búfer de memoria del servidor.
5. Configurar el tamaño de la memoria de trabajo del servidor.
6. Configurar el tamaño de la memoria de mantenimiento del servidor.
7. Configurar el punto de chequeo.
8. Activar/desactivar conexión SSL.
9. Configurar el puerto de escucha del servidor.
10. Configurar el tiempo máximo para que un usuario se autentique.
11. Activar/desactivar encriptación en las contraseñas de los usuarios.
12. Configurar el tamaño máximo de logs.
13. Tiempo para la actualización de los logs de seguridad.
14. Mostrar la ubicación de los logs de seguridad.
15. Crear base de datos en el servidor.
16. Eliminar base de datos en el servidor.
17. Crear usuario en el servidor.
18. Eliminar usuario del servidor.
19. Correr script sobre el servidor.
20. Mostrar configuración actual del servidor.
21. Hacer una salva a las bases de datos del servidor seleccionadas.

22. Realizar restauración a las base de datos.
23. Activar/desactivar logs de seguridad.

### **2.3.2 Funcionalidades Módulo MySQLManager**

24. Conectar a servidor remoto mediante SSH.
25. Configurar acceso al servidor.
26. Configurar el número máximo de clientes que pueden conectarse al servidor.
27. Configurar el tamaño de búfer de memoria del servidor.
28. Configurar el tamaño de la caché de consulta.
29. Configurar el tamaño límite para caché de consulta.
30. Activar/desactivar conexión SSL.
31. Configurar el puerto de escucha del servidor.
32. Configurar el tiempo máximo para que un usuario se autentique.
33. Mostrar estado de encriptación de contraseñas.
34. Configurar el tamaño máximo de logs.
35. Tiempo para la actualización de los logs de seguridad.
36. Mostrar la ubicación de los logs de seguridad.
37. Crear base de datos en el servidor.
38. Eliminar base de datos en el servidor.
39. Crear usuario en el servidor.
40. Eliminar usuario del servidor.
41. Correr script sobre el servidor.
42. Mostrar configuración actual del servidor.
43. Realizar una salva a las bases de datos.
44. Realizar restauración de las bases de datos.

### **2.3.2. Características del sistema**

Las siguientes características son las que hacen al producto atractivo, usable, rápido y confiable.

- **Usabilidad:** En la interfaz se deben combinar diseños atractivos y refrescantes para el usuario y debe ser de fácil entendimiento.
- **Portabilidad:** La aplicación será ejecutada sobre Ubuntu (Linux).
- **Seguridad:** La conexión a los servidores será mediante SSH y las contraseñas no serán almacenadas. La transferencia de archivos será mediante SCP. La autenticación se realizará con un usuario con permisos administrativos.

- **Confidencialidad:** La información manejada por la herramienta estará protegida, utilizando el protocolo SSH para establecer conexiones seguras. Además del uso del complemento SCP para la transferencia segura de archivos.
- **Integridad:** La información manejada por la herramienta será objeto de cuidadosa protección contra la corrupción y estados inconsistentes, de la misma forma será considerada igual a la fuente o autoridad de los datos.
- **Rendimiento:** El sistema debe requerir un consumo mínimo de recursos.
- **Software:** El lenguaje de programación es C++. El IDE será QT Creator 2.2 y el framework Qt 4. El sistema operativo será Linux en específico la distribución de Ubuntu.
- **Hardware:** Memoria RAM de 256 Mb o superior. Espacio en disco de 100 Mb o superior.

## 2.4 Personal relacionado con la aplicación

Una persona relacionada con el sistema es aquella que está vinculada al desarrollo o al uso de la misma. En la aplicación en cuestión existirán dos roles:

- Desarrollador: Persona que implementará la aplicación.
- Gestor de Configuración: Persona que hará uso de la aplicación para configurar servidores de bases de datos.

## 2.5 Flujos de trabajo para el desarrollo de la aplicación

### 2.5.1 Fase de Exploración

Es la fase en la que se define el alcance general del proyecto. El cliente define lo que necesita mediante la redacción de sencillas historias de usuario. Los programadores estiman los tiempos de desarrollo en base a esta información. Debe quedar claro que las estimaciones realizadas en esta fase son primarias (ya que estarán basadas en datos de muy alto nivel), y podrían variar cuando se analicen más en detalle en cada iteración. Esta fase dura típicamente un par de semanas, y el resultado es una visión general del sistema, y un plazo total estimado. (19)



### 2.5.1.1 Historias de Usuario

Las Historias de usuarios sustituyen a los documentos de especificación funcional, y a los casos de uso. Estas historias son escritas por el cliente, en su propio lenguaje, como descripciones cortas de lo que el sistema debe realizar. (19)

A continuación se especifican las historias de usuario para cada módulo.

#### 2.5.1.1.1 Historias de Usuario módulo PostgresManager

Tabla 1: Historia de usuario Conexión a servidor mediante SSH.

|   |                             |
|---|-----------------------------|
| Historia de Usuario   | Nombre: Conexión a servidor |
| Número: 1   | Iteración: 1                |
| Usuario: Gestor de Configuración  |                             |
| Riesgo en desarrollo: Alto  |                             |
| Prioridad: Alta   |                             |
| Descripción: El gestor se conecta al servidor remoto ingresando su dirección ip y usuario. Esta funcionalidad permitirá la transferencia de ficheros al servidor. |                             |

Tabla 2: Historia de usuario Configurar acceso al servidor

|  |                                       |
|--|---------------------------------------|
| Historia de Usuario  | Nombre: Configurar acceso al servidor |
| Número: 2  | Iteración: 1                          |
| Usuario: Gestor de Configuración   |                                       |
| Riesgo en desarrollo: Alto   |                                       |
| Prioridad: Alta  |                                       |
| Descripción: El gestor podrá configurar quién se conecta al servidor y de qué forma puede hacerlo. |                                       |

Tabla 3: Historia de usuario Configurar memoria del servidor

|  |   |
|--|---|
| Historia de Usuario  | Nombre: Configurar memoria del servidor |
| Número: 3  | Iteración: 1                            |
| Usuario: Gestor de Configuración   |   |
| Riesgo en desarrollo: Alto   |   |
| Prioridad: Alta  |   |
| Descripción: El gestor podrá configurar aspectos relacionados con la memoria usada por el servidor, tales como el tamaño del búfer de memoria y la memoria de trabajo. |   |

Tabla 4: Historia de usuario Configuración de seguridad del servidor

|  |   |
|--|---|
| Historia de Usuario  | Nombre: Configuración de seguridad del servidor |
| Número: 4  | Iteración: 2                                    |
| Usuario: Gestor de Configuración   |   |
| Riesgo en desarrollo: Alto   |   |
| Prioridad: Alta  |   |
| Descripción: El gestor podrá configurar aspectos de seguridad como SSL, cantidad máxima de conexiones y logs de seguridad. |   |

Tabla 5: Historia de usuario Gestionar base de datos

|                     |                                 |
|---------------------|---------------------------------|
| Historia de Usuario | Nombre: Gestionar base de datos |
|---------------------|---------------------------------|

|  |              |
|--|--------------|
| Número: 5  | Iteración: 2 |
| Usuario: Gestor de Configuración   |              |
| Riesgo en desarrollo: Medio  |              |
| Prioridad: Alta  |              |
| Descripción: El gestor tiene la posibilidad de eliminar o crear bases de datos en el servidor. |              |

Tabla 6: Historia de usuario Gestionar usuario del servidor

|  |  |
|--|--|
| Historia de Usuario  | Nombre: Gestionar usuario del servidor |
| Número: 6  | Iteración: 2                           |
| Usuario: Gestor de Configuración   |  |
| Riesgo en desarrollo: Medio  |  |
| Prioridad: Alta  |  |
| Descripción: Se utilizará para crear o eliminar usuarios en el servidor. |  |

Tabla 7: Historia de usuario Correr Script sobre el servidor

|                                  |   |
|----------------------------------|---|
| Historia de Usuario              | Nombre: Correr Script sobre el servidor |
| Número: 7                        | Iteración: 3                            |
| Usuario: Gestor de Configuración |   |
| Riesgo en desarrollo: bajo       |   |
| Prioridad: media                 |   |

|  |  |
|--|--|
| Descripción: Le permitirá al gestor enviar al servidor consultas SQL almacenadas en archivos externos. |  |
|--|--|

Tabla 8: Historia de usuario Mostrar configuración actual del servidor

|  |   |
|--|---|
| Historia de Usuario  | Nombre: Mostrar configuración actual del servidor |
| Número: 8  | Iteración: 1                                      |
| Usuario: Gestor de Configuración   |   |
| Riesgo en desarrollo: Medio  |   |
| Prioridad: Alta  |   |
| Descripción: Le permitirá al gestor observar la configuración que posee el servidor actualmente. |   |

Tabla 9: Historia de usuario Realizar salva de las bases de datos del servidor

|   |   |
|---|---|
| Historia de Usuario   | Nombre: Realizar salva de las bases de datos del servidor |
| Número: 9   | Iteración: 3  |
| Usuario: Gestor de Configuración                                    |   |
| Riesgo en desarrollo: Medio   |   |
| Prioridad: Alta   |   |
| Descripción: Para la realización de salvadas de las bases de datos. |   |

Tabla 10: Historia de usuario Realizar restauración de las bases de datos del servidor

|   |  |
|---|--|
| Historia de Usuario   | Nombre: Realizar restauración de las bases de datos del servidor |
| Número: 10  | Iteración: 3   |
| Usuario: Gestor de Configuración  |  |
| Riesgo en desarrollo: Medio   |  |
| Prioridad: Alta   |  |
| Descripción: Para la realización de restauración de las bases de datos. |  |

#### 2.5.1.1.2 Historias de Usuario módulo MySQLManager

Tabla 11: Historia de usuario Conexión a servidor mediante SSH.

|   |                             |
|---|-----------------------------|
| Historia de Usuario   | Nombre: Conexión a servidor |
| Número: 11  | Iteración: 1                |
| Usuario: Gestor de Configuración  |                             |
| Riesgo en desarrollo: Alto  |                             |
| Prioridad: Alta   |                             |
| Descripción: El gestor se conecta al servidor remoto ingresando su dirección ip y usuario. Esta funcionalidad permitirá la transferencia de ficheros al servidor. |                             |

Tabla 12: Historia de usuario Configurar acceso al servidor

|                                  |                                       |
|----------------------------------|---------------------------------------|
| Historia de Usuario              | Nombre: Configurar acceso al servidor |
| Número: 12                       | Iteración: 1                          |
| Usuario: Gestor de Configuración |                                       |

|  |  |
|--|--|
| Riesgo en desarrollo: Alto   |  |
| Prioridad: Alta  |  |
| Descripción: El gestor podrá configurar quién se conecta al servidor y de qué forma puede hacerlo. |  |

Tabla 13: Historia de usuario Configurar memoria del servidor

|  |   |
|--|---|
| Historia de Usuario  | Nombre: Configurar memoria del servidor |
| Número: 13   | Iteración: 1                            |
| Usuario: Gestor de Configuración   |   |
| Riesgo en desarrollo: Alto   |   |
| Prioridad: Alta  |   |
| Descripción: El gestor podrá configurar los aspectos relacionados con la memoria usada por el servidor, tales como el tamaño del búfer de memoria y la memoria dedicada a consultas. |   |

Tabla 14: Historia de usuario Configuración de seguridad del servidor

|                                  |   |
|----------------------------------|---|
| Historia de Usuario              | Nombre: Configuración de seguridad del servidor |
| Número: 14                       | Iteración: 2                                    |
| Usuario: Gestor de Configuración |   |
| Riesgo en desarrollo: Alto       |   |
| Prioridad: Alta                  |   |

|  |  |
|--|--|
| Descripción: El gestor podrá configurar aspectos de seguridad como SSL, cantidad máxima de conexiones y logs de seguridad. |  |
|--|--|

Tabla 15: Historia de usuario Gestionar base de datos

|  |                                 |
|--|---------------------------------|
| Historia de Usuario  | Nombre: Gestionar base de datos |
| Número: 15   | Iteración: 2                    |
| Usuario: Gestor de Configuración   |                                 |
| Riesgo en desarrollo: Medio  |                                 |
| Prioridad: Alta  |                                 |
| Descripción: Mediante esta el gestor tiene la posibilidad de eliminar o crear bases de datos en el servidor. |                                 |

Tabla 16: Historia de usuario Gestionar usuario del servidor

|  |  |
|--|--|
| Historia de Usuario  | Nombre: Gestionar usuario del servidor |
| Número: 16   | Iteración: 2                           |
| Usuario: Gestor de Configuración   |  |
| Riesgo en desarrollo: Medio  |  |
| Prioridad: Alta  |  |
| Descripción: Se utilizará para crear o eliminar usuarios en el servidor. |  |

Tabla 17: Historia de usuario Correr Script sobre el servidor

|  |   |
|--|---|
| Historia de Usuario  | Nombre: Correr Script sobre el servidor |
| Número: 17   | Iteración: 3                            |
| Usuario: Gestor de Configuración   |   |
| Riesgo en desarrollo: bajo   |   |
| Prioridad: media   |   |
| Descripción: Le permitirá al gestor enviar al servidor consultas SQL almacenadas en archivos externos. |   |

Tabla 18: Historia de usuario Mostrar configuración actual del servidor

|  |   |
|--|---|
| Historia de Usuario  | Nombre: Mostrar configuración actual del servidor |
| Número: 18   | Iteración: 1                                      |
| Usuario: Gestor de Configuración   |   |
| Riesgo en desarrollo: Medio  |   |
| Prioridad: Alta  |   |
| Descripción: Le permitirá al gestor observar la configuración que posee el servidor actualmente. |   |

Tabla 19: Historia de usuario Realizar salva de las bases de datos del servidor

|                     |   |
|---------------------|---|
| Historia de Usuario | Nombre: Realizar salva de las bases de datos del servidor |
| Número: 19          | Iteración: 3  |



|   |  |
|---|--|
| Usuario: Gestor de Configuración                                    |  |
| Riesgo en desarrollo: Medio   |  |
| Prioridad: Alta   |  |
| Descripción: Para la realización de salvadas de las bases de datos. |  |

Tabla 20: Historia de usuario Realizar restauración de las bases de datos del servidor

|   |  |
|---|--|
| Historia de Usuario   | Nombre: Realizar restauración de las bases de datos del servidor |
| Número: 20  | Iteración: 3   |
| Usuario: Gestor de Configuración  |  |
| Riesgo en desarrollo: Medio   |  |
| Prioridad: Alta   |  |
| Descripción: Para la realización de restauración de las bases de datos. |  |

## 2.6 Planificación de la entrega.

En esta fase el cliente establece la prioridad de cada historia de usuario de acuerdo a la urgencia de sus necesidades. A partir de estas los desarrolladores realizan una estimación del esfuerzo necesario de cada una de ellas. Se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente. Una entrega debe obtenerse en no más de tres meses.

### 2.6.1 Estimación de esfuerzo por historia de usuario.

Las historias de usuario deben ser programadas en un tiempo entre una y tres semanas. Si la estimación es superior a tres semanas, debe ser dividida en dos o más historias. Si es menos de una semana, se debe combinar con otra historia. (19) Para ello se decide realizar la estimación de esfuerzo que arroja cada historia de usuario

con el objetivo de obtener un mejor desarrollo del sistema. La siguiente tabla muestra una estimación del tiempo medido en semanas para cada historia de usuario.

Tabla 21: Estimación de esfuerzo de historia de usuario módulo PostgresManager

| No | Historia de usuario                                      | Puntos de estimación (semanas) |
|----|--|--------------------------------|
| 1  | Conexión a servidor mediante SSH                         | 1                              |
| 2  | Configurar acceso al servidor                            | 2                              |
| 3  | Configurar memoria del servidor                          | 2                              |
| 4  | Configuración de seguridad del servidor                  | 2                              |
| 5  | Gestionar base de datos                                  | 1                              |
| 6  | Gestionar usuario del servidor                           | 1                              |
| 7  | Correr Script sobre el servidor                          | 1                              |
| 8  | Mostrar configuración actual del servidor                | 1                              |
| 9  | Realizar salva de las bases de datos del servidor        | 1                              |
| 10 | Realizar restauración de las bases de datos del servidor | 1                              |

Tabla 22: Estimación de esfuerzo de historia de usuario módulo MySQLManager

| No | Historia de usuario                     | Puntos de estimación (semanas) |
|----|---|--------------------------------|
| 1  | Conexión a servidor mediante SSH        | 1                              |
| 2  | Configurar acceso al servidor           | 2                              |
| 3  | Configurar memoria del servidor         | 2                              |
| 4  | Configuración de seguridad del servidor | 2                              |

|    |  |   |
|----|--|---|
| 5  | Gestionar base de datos                                  | 2 |
| 6  | Gestionar usuario del servidor                           | 1 |
| 7  | Correr Script sobre el servidor                          | 1 |
| 8  | Mostrar configuración actual del servidor                | 1 |
| 9  | Realizar salva de las bases de datos del servidor        | 1 |
| 10 | Realizar restauración de las bases de datos del servidor | 1 |

### 2.6.2 Planificación de las Iteraciones

Debido a que las historias de usuario no tienen suficiente detalle como para permitir su análisis y desarrollo, al principio de cada iteración se realizan las tareas necesarias de análisis, recabando con el cliente todos los datos que sean necesarios. El cliente, por lo tanto, también debe participar activamente durante esta fase del ciclo. Es recomendable priorizar las historias de usuario que sean más importantes para la arquitectura, sin embargo esto no siempre es posible ya que es el usuario quién decide que historias se implementará en cada iteración.

Después de definidas las historias de usuario y haber estimado el tiempo de cada una, se decide realizar 3 iteraciones. (Tablas 22 y 23)

Tabla 22: Planificación de iteraciones para el módulo PostgresManager

| Iteración | Historia de usuario                       | Duración (semanas) |
|-----------|---|--------------------|
| 1         | Conexión a servidor mediante SSH          | 6                  |
|           | Configurar acceso al servidor             |                    |
|           | Configurar memoria del servidor           |                    |
|           | Mostrar configuración actual del servidor |                    |
| 2         | Gestionar base de datos                   | 5                  |
|           | Gestionar usuario del servidor            |                    |

|   |  |   |
|---|--|---|
|   | Configuración de seguridad del servidor                  |   |
| 3 | Realizar restauración de las bases de datos del servidor | 3 |
|   | Realizar salva de las bases de datos del servidor        |   |
|   | Correr Script sobre el servidor                          |   |

Tabla 23: Planificación de iteraciones para el módulo MySQLManager

| Iteración | Historia de usuario                                      | Duración (semanas) |
|-----------|--|--------------------|
| 1         | Conexión a servidor mediante SSH                         | 6                  |
|           | Configurar acceso al servidor                            |                    |
|           | Configurar memoria del servidor                          |                    |
|           | Mostrar configuración actual del servidor                |                    |
| 2         | Gestionar base de datos                                  | 5                  |
|           | Gestionar usuario del servidor                           |                    |
|           | Configuración de seguridad del servidor                  |                    |
| 3         | Realizar restauración de las bases de datos del servidor | 3                  |
|           | Realizar salva de las bases de datos del servidor        |                    |
|           | Correr Script sobre el servidor                          |                    |

## 2.7 Diseño

Para el diseño la metodología XP no requiere de hacer representaciones de clases mediante diagramas, utiliza en su lugar las tarjetas de Contenido, Responsabilidad y Colaboración (CRC). La Metodología XP hace especial énfasis en diseños simples y

claros. Siempre velando que cumpla con los requerimientos de las historias de usuario.

### 2.7.1 Tarjetas CRC

Las tarjetas CRC utilizan la técnica de modelado basada en objetos. Donde una tarjeta CRC representa un objeto y sus responsabilidades. Seguidamente se muestran algunos ejemplos de las tarjetas CRC para cada uno de los módulos en cuestión. Las tarjetas están compuestas por el nombre de la clase, las funcionalidades de la misma (parte izquierda) y en la parte derecha las clases relacionadas con cada funcionalidad.

#### 2.7.1.1 Tarjetas CRC del Módulo PostgresManager

| <b>CMemoria</b>                     |                           |
|-------------------------------------|---------------------------|
| <b>Responsabilidades</b>            | <b>Clase relacionadas</b> |
| Configurar memoria compartida       | MainWindow                |
| Configurar memoria de trabajo       |                           |
| Configurar memoria de mantenimiento |                           |
| Configurar punto de chequeo         |                           |

| <b>CReglaConexion</b>    |                           |
|--------------------------|---------------------------|
| <b>Responsabilidades</b> | <b>Clase relacionadas</b> |
| Crear regla de conexión  | MainWindow                |
|                          | FReglaConexion            |

#### 2.7.1.2 Tarjetas CRC del Módulo MySQLManager

| <b>CConexion</b>                       |                           |
|--|---------------------------|
| <b>Responsabilidades</b>               | <b>Clase relacionadas</b> |
| Configurar número máximo de conexiones | MainWindow                |
| Configurar puerto de escucha           | User                      |
| Crear usuario                          | UserPass                  |
| Eliminar usuario                       | FReglaConexion            |

|  |              |
|--|--------------|
| Crear base de datos                    |              |
| Eliminar base de datos                 |              |
| Ejecutar script SQL                    | Pregunta     |
| Realizar salva a una base de datos     | SalidaScript |
| Configurar SSL                         |              |
| Configurar encriptación de contraseñas |              |
| Crear regla de conexión                |              |

| <b>CLogs</b>                      |                           |
|-----------------------------------|---------------------------|
| <b>Responsabilidades</b>          | <b>Clase relacionadas</b> |
| Configurar tamaño de logs         | MainWindow                |
| Configurar tiempo de vida de logs | FLogs                     |
| Mostrar directorio de logs        |                           |

## **2.7.2 Uso de patrones de diseño**

Un patrón de diseño es una solución a un problema de diseño. Tiene la característica principal de ser capaz de resolver problemas similares. Debe ser reutilizable, o sea aplicable a diferentes problemas de diseño en distintos contextos.

### **2.7.2.1 Patrones GRASP**

Los patrones GRASP (acrónimo en inglés General Responsibility Assignment Software Patterns) son patrones de diseño para la asignación de responsabilidades. Su uso es fundamental para un correcto diseño del software. Ayudan a la reutilización de diseño gráfico, identificando aspectos claves de la estructura de un diseño que puede ser aplicado en una gran cantidad de situaciones. Lo que supone una gran ventaja ya que disminuye los esfuerzos de desarrollo y mantenimiento y mejora la seguridad, eficiencia y consistencia del diseño. También ayudan a tener un software más flexible, modular y extensible. (23)

### 2.7.2.1.1 Patrón creador

Este patrón resulta muy útil ya que ayuda a definir quién debe ser el responsable de la creación de nuevas clases u objetos. La clase que instancie un objeto debe cumplir que:

- Tiene la información necesaria para realizar la creación del objeto.
- Usa directamente las instancias creadas del objeto.
- Almacena o maneja varias instancias de la clase.
- Contiene o agrega la clase.

Con el uso de este patrón se garantiza el bajo acoplamiento lo que facilita futuros trabajos de mantenimiento además de favorecer la reutilización.

### 2.7.2.1.2 Patrón experto

Indica que la responsabilidad de la creación de un objeto o la implementación de un método, debe recaer sobre la clase que conoce toda la información necesaria para crearlo. De esta forma se consigue un diseño con mayor cohesión y encapsulación en la información. Además los objetos utilizan su propia información para realizar sus responsabilidades. Lo que propicia un código más entendible y fácil de mantener.

### 2.7.2.1.3 Variaciones Protegidas

Es el umbral fundamental de protegerse del cambio, o sea que todo lo que se prediga en un análisis previo que es susceptible de modificaciones, se envuelve en una interfaz, utilizando el polimorfismo para crear varias implementaciones y posibilitar implementaciones futuras. Esto permite que quede lo menos ligado posible al sistema, de forma que cuando se produzca la variación, repercuta lo mínimo. (24)

## 2.7.3 Estándares de codificación

### 2.7.3.1 Identación, llaves de apertura, de cierre y tamaño de las líneas

La indentación debe ser sin tabulaciones, con un equivalente a cuatro espacios, el uso de las llaves “{}” será en una nueva línea. Cada línea de código contendrá hasta 80 caracteres para mejor legibilidad del código.

### 2.7.3.2 Conversión de nomenclatura

**Variables:** Siempre comienzan con minúscula y en caso de nombres compuestos la primera letra de cada palabra comienza con mayúscula.

**Constantes:** todas en mayúsculas, con caracteres de subrayado “\_” para separar palabras en caso de nombres compuestos.

**Clases:** siempre comienzan con mayúscula, en caso de nombre compuesto las palabras se separan con el carácter subrayado “\_” y el resto en minúscula.

**Funciones:** Siempre comienzan con minúscula y en caso de nombres compuestos la primera letra de cada palabra comienza con mayúscula. Los parámetros son separados por espacio luego de la coma que los separa.

### **2.7.3.3 Estructuras de control**

Se incluyen palabras reservadas como: if, for, foreach, while, switch, entre las estructuras de control y entre los paréntesis debe de existir un espacio. Se recomienda utilizar siempre llaves de apertura y cierre, incluso en situaciones en las que técnicamente son opcionales, esto aumenta la legibilidad y disminuye la probabilidad de errores lógicos. Si las condiciones son muy largas que sobrepasan el tamaño de la línea, estas se dividen en varias líneas. En el mejor de los casos cuando la condición es muy extensa, se puede dividir esta en variables y compararlas dentro de la estructura de control.

## **2.8 Conclusiones**

A partir del conocimiento del trabajo de los gestores de configuración de servidores de bases de datos en entornos productivos de la UCI, se ha creado la propuesta del sistema. Se ha determinado las personas relacionadas con la aplicación. También se presentaron las historias de usuarios y se definió su distribución en cada iteración además de la duración de las mismas medidas en semanas. Lo que permite el paso a la fase iteraciones.



## Capítulo 3: Implementación y Pruebas

### 3.1 Introducción

El presente capítulo está enmarcado en la fase iteraciones, fase principal de XP. Todas las funcionalidades del software serán desarrolladas en esta fase, generando al final de cada una un entregable funcional que implementa las historias de usuario asociadas a la iteración en cuestión. Al principio de cada iteración se realizan las tareas necesarias de análisis, recabando con el cliente todos los datos que sean necesarios. El cliente, por lo tanto, también debe participar activamente durante esta fase del ciclo. Finalmente se realizarán las pruebas de aceptación a cada funcionalidad del software.

### 3.2 Fase Iteraciones

Las historias de usuario se implementan en la iteración a la cual pertenecen. Es por ello que XP propone que se revise el plan de iteraciones y se modifique y actualice en caso de ser necesario. De igual modo sucede con las tareas de implementación asociadas a cada historia de usuario.

Para el presente trabajo se propone realizar tres iteraciones que a continuación se detallan (para ambos módulos).

#### 3.3.1 Iteración 1

En esta iteración se implementan las historias de usuarios más importantes para el correcto funcionamiento de la aplicación ya que contienen las funcionalidades imprescindibles. Las siguientes tablas muestran los detalles de cada tarea de implementación incluyendo su duración estimada en días.

Tabla 3.1 Tarea conexión SSH

|  |                            |
|--|----------------------------|
| Número de la tarea: 1  | Número de HU:1             |
| Nombre: Conexión SSH   |                            |
| Tipo tarea: Implementación   | Puntos estimados (días): 7 |
| Descripción: Se implementa una funcionalidad que permita conectarse a un servidor usando el protocolo SSH para la transferencia de archivos. |                            |

Tabla 3.2 Tarea mostrar configuración actual del servidor

|   |                           |
|---|---------------------------|
| Número de la tarea: 2   | Número de HU:8            |
| Nombre: mostrar configuración actual del servidor   |                           |
| Tipo tarea: Implementación  | Puntos estimados (días):7 |
| Descripción: Se implementa una funcionalidad que permita observar la configuración actual del servidor. |                           |

Tabla 3.3 Tarea configuración de acceso al servidor

|  |                             |
|--|-----------------------------|
| Número de la tarea: 3  | Número de HU:2              |
| Nombre: configuración de acceso al servidor  |                             |
| Tipo tarea: Implementación   | Puntos estimados (días): 14 |
| Descripción: Se implementan funcionalidades que permitan la configuración de acceso al servidor. De forma que se pueda definir cómo y desde que lugar los usuarios se pueden conectar al servidor. |                             |

Tabla 3.4 Tarea configuración de memoria del servidor

|   |                            |
|---|----------------------------|
| Número de la tarea: 4                         | Número de HU:3             |
| Nombre: configuración de memoria del servidor |                            |
| Tipo tarea: Implementación                    | Puntos estimados (días): 5 |

Descripción: Se implementan funciones que permitan configurar la cantidad de memoria que usará el servidor de bases de datos.

Tabla 3.5 Tarea configurar memoria de trabajo y memoria de mantenimiento

|  |                     |
|--|---------------------|
| Número de la tarea: 5  | Número de HU:3      |
| Nombre: configurar memoria de trabajo y memoria de mantenimiento   |                     |
| Tipo tarea: Implementación   | Puntos estimados: 5 |
| Descripción: Se implementan funciones que permitan configurar la cantidad de memoria que usará el servidor para tareas de mantenimiento y para ejecutar ciertas consultas. |                     |

Tabla 3.6 Tarea configurar punto de chequeo

|   |                     |
|---|---------------------|
| Número de la tarea: 6   | Número de HU:3      |
| Nombre: configurar punto de chequeo   |                     |
| Tipo tarea: Implementación  | Puntos estimados: 4 |
| Descripción: En esta tarea se implementan funciones para configurar el punto de chequeo del servidor. |                     |

### 3.3.2 Iteración 2

En esta iteración se desarrollan las funcionalidades de importancia media. O sea, se implementan las historias de usuarios de riesgo relativamente bajo para el desarrollador.

**Tabla 3.7** Tarea activar/desactivar conexión SSL y encriptación de contraseñas

|  |                     |
|--|---------------------|
| Número de la tarea: 7  | Número de HU:4      |
| Nombre: activar/desactivar conexión SSL y encriptación de contraseñas  |                     |
| Tipo tarea: Implementación   | Puntos estimados: 4 |
| Descripción: Se implementa la forma de activar o desactivar las conexiones SSL y la encriptación de contraseñas. |                     |

**Tabla 3.8** Tarea tiempo espera para autenticación

|  |                     |
|--|---------------------|
| Número de la tarea: 8  | Número de HU:4      |
| Nombre: tiempo espera para autenticación   |                     |
| Tipo tarea: Implementación   | Puntos estimados: 2 |
| Descripción: Para configurar el tiempo de espera del servidor para la autenticación del usuario. |                     |

**Tabla 3.9** Tarea mostrar directorio de logs de seguridad

|   |                     |
|---|---------------------|
| Número de la tarea: 9                           | Número de HU:4      |
| Nombre: mostrar directorio de logs de seguridad |                     |
| Tipo tarea: Implementación                      | Puntos estimados: 2 |

Descripción: Mostrar el directorio actual en donde se guardan los logs de seguridad.

**Tabla 3.10** Tarea configurar tamaño máximo de logs de seguridad

|  |                     |
|--|---------------------|
| Número de la tarea: 10   | Número de HU:4      |
| Nombre: configurar tamaño máximo de logs de seguridad  |                     |
| Tipo tarea: Implementación   | Puntos estimados: 2 |
| Descripción: En esta tarea se implementan las funcionalidades que permiten configurar el tamaño máximo que tendrá un archivo de logs de seguridad. |                     |

**Tabla 3.11** Tarea configurar tiempo de vida de logs de seguridad

|   |                     |
|---|---------------------|
| Número de la tarea: 11  | Número de HU:4      |
| Nombre: configurar tiempo de vida de logs de seguridad  |                     |
| Tipo tarea: Implementación  | Puntos estimados: 2 |
| Descripción: En esta tarea se implementan las funcionalidades que permiten configurar el tiempo de vida de logs de seguridad. |                     |

### 3.3.2 Iteración 3

En esta iteración se implementan las historias de usuario de menor importancia para el cliente.

**Tabla 3.12** Tarea crear base de datos

|   |                     |
|---|---------------------|
| Número de la tarea: 12                          | Número de HU:5      |
| Nombre: crear base de datos                     |                     |
| Tipo tarea: Implementación                      | Puntos estimados: 3 |
| Descripción: Para la creación de bases de datos |                     |

**Tabla 3.13** Tarea eliminar base de datos

|   |                     |
|---|---------------------|
| Número de la tarea: 13  | Número de HU:5      |
| Nombre: eliminar base de datos  |                     |
| Tipo tarea: Implementación  | Puntos estimados: 4 |
| Descripción: Se implementan las funcionalidades que permiten eliminar una base de datos seleccionada. |                     |

**Tabla 3.14** Tarea crear usuario

|   |                     |
|---|---------------------|
| Número de la tarea: 14  | Número de HU:6      |
| Nombre: crear usuario   |                     |
| Tipo tarea: Implementación  | Puntos estimados: 3 |
| Descripción: Se implementan las funcionalidades que permiten crear un usuario en el |                     |

servidor.

**Tabla 3.15** Tarea eliminar usuario

|  |                     |
|--|---------------------|
| Número de la tarea: 15   | Número de HU:6      |
| Nombre: eliminar usuario   |                     |
| Tipo tarea: Implementación   | Puntos estimados: 4 |
| Descripción: Se implementan las funcionalidades que permiten eliminar un usuario en el servidor. |                     |

**Tabla 3.16** Tarea realizar salva y restauración de base de datos

|  |                      |
|--|----------------------|
| Número de la tarea: 16   | Número de HU:9       |
| Nombre: realizar salva de base de datos  |                      |
| Tipo tarea: Implementación   | Puntos estimados: 14 |
| Descripción: En esta tarea se implementan las funciones que permiten realizar una salva y una restauración a una base de datos seleccionada. |                      |

### 3.5 Pruebas

Las pruebas son un conjunto de actividades que se pueden planificar por adelantado y llevar a cabo sistemáticamente. Por esta razón se debe definir en el proceso de la ingeniería del software. (25) Todo esto contribuye a elevar la calidad de los productos desarrollados y a la seguridad de los programadores a la hora de introducir cambios o modificaciones.

La metodología XP divide las pruebas en dos grupos: pruebas unitarias, desarrolladas por los programadores, encargadas de verificar el código de forma automática y las

pruebas de aceptación, destinadas a evaluar si al final de una iteración se obtuvo la funcionalidad requerida, además de comprobar que dicha funcionalidad sea la esperada por el cliente. (26)

### 3.5.1 Pruebas de aceptación

Las pruebas de aceptación son pruebas de caja negra que se realizan a partir de las historias de usuario. (26) Durante las iteraciones las historias de usuario escogidas serán traducidas a prueba de aceptación. En ella se especifican, la perspectiva del cliente, y los escenarios para probar que la historia de usuario ha sido implementada correctamente. Una historia de usuario puede tener todas las pruebas de aceptación que desee para asegurar su funcionamiento. El objetivo específico de esta prueba es garantizar que los requerimientos han sido cumplidos y que el sistema ha sido aceptable. (27)

Seguidamente se muestran las pruebas de aceptación divididas en iteraciones.

### 3.5.2 Pruebas de aceptación para PostgresManager

#### 3.5.2.1 Iteración 1

**Tabla 3.17** Prueba de aceptación Historia de usuario 1 prueba 1

|  |                               |
|--|-------------------------------|
| <b>Caso de Prueba de aceptación</b>  |                               |
| <b>Código:</b> HU1_P1  | <b>Historia de usuario:</b> 1 |
| <b>Nombre:</b> Conexión a servidor mediante SSH.   |                               |
| <b>Descripción:</b> Prueba de la funcionalidad para conectarse vía SSH a un servidor con el objetivo de la transferencia de archivos.  |                               |
| <b>Condiciones de ejecución:</b> Se debe tener instalado OpenSSH en la máquina local y en el servidor.   |                               |
| <b>Pasos de ejecución:</b> El gestor se conecta al servidor de base de datos mediante SSH introduciendo nombre de usuario y contraseña del servidor, luego selecciona el fichero a transferir. |                               |
| <b>Resultados esperado:</b> Se realiza la conexión sin generar error y se transfiere el fichero.   |                               |



**Evaluación de la prueba:** Satisfactoria.

**Tabla 3.18** Prueba de aceptación Historia de usuario 2 prueba 1

|   |                               |
|---|-------------------------------|
| <b>Caso de Prueba de aceptación</b>   |                               |
| <b>Código:</b> HU2_P1   | <b>Historia de usuario:</b> 2 |
| <b>Nombre:</b> Configurar acceso al servidor.   |                               |
| <b>Descripción:</b> Prueba para la funcionalidad Configurar acceso al servidor  |                               |
| <b>Condiciones de ejecución:</b> El gestor de configuración debe haberse conectado al servidor de forma correcta.                                   |                               |
| <b>Pasos de ejecución:</b> El gestor determina la forma en que se conectan los usuarios al servidor de base de datos a través de reglas de conexión |                               |
| <b>Resultados esperado:</b> Se realiza la configuración de acceso sin errores   |                               |
| <b>Evaluación de la prueba:</b> Satisfactoria   |                               |

**Tabla 3.19** Prueba de aceptación Historia de usuario 3 prueba 1

|   |                               |
|---|-------------------------------|
| <b>Caso de Prueba de aceptación</b>   |                               |
| <b>Código:</b> HU3_P1   | <b>Historia de usuario:</b> 3 |
| <b>Nombre:</b> Configurar el tamaño de la memoria compartida.   |                               |
| <b>Descripción:</b> Prueba para la funcionalidad Configurar el tamaño de la memoria compartida.                   |                               |
| <b>Condiciones de ejecución:</b> El gestor de configuración debe haberse conectado al servidor de forma correcta. |                               |

|  |
|--|
| <b>Pasos de ejecución:</b> El gestor configura la cantidad de memoria RAM que usará el servidor. |
| <b>Resultados esperado:</b> Se realiza la configuración sin errores.                             |
| <b>Evaluación de la prueba:</b> insatisfactoria.   |

**Tabla 3.20** Prueba de aceptación Historia de usuario 3 prueba 2

|   |                               |
|---|-------------------------------|
| <b>Caso de Prueba de aceptación</b>   |                               |
| <b>Código:</b> HU3_P2   | <b>Historia de usuario:</b> 3 |
| <b>Nombre:</b> Configurar memoria de trabajo.   |                               |
| <b>Descripción:</b> Prueba para la funcionalidad Configurar memoria de trabajo.             |                               |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente.         |                               |
| <b>Pasos de ejecución:</b> El gestor intenta configurar el tamaño de la memoria de trabajo. |                               |
| <b>Resultados esperado:</b> Se realiza la configuración sin generar error.                  |                               |
| <b>Evaluación de la prueba:</b> Satisfactoria.  |                               |

**Tabla 3.21** Prueba de aceptación Historia de usuario 3 prueba 3

|   |                               |
|---|-------------------------------|
| <b>Caso de Prueba de aceptación</b>   |                               |
| <b>Código:</b> HU3_P3   | <b>Historia de usuario:</b> 3 |
| <b>Nombre:</b> Configurar memoria de mantenimiento.                                   |                               |
| <b>Descripción:</b> Prueba para la funcionalidad Configurar memoria de mantenimiento. |                               |

|   |
|---|
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente.               |
| <b>Pasos de ejecución:</b> El gestor intenta configurar el tamaño de la memoria de mantenimiento. |
| <b>Resultados esperado:</b> Se realiza la configuración sin generar error.                        |
| <b>Evaluación de la prueba:</b> Satisfactoria.  |

### 3.5.2.2 Iteración 2

**Tabla 3.22** Prueba de aceptación Historia de usuario 4 prueba 1

|  |                               |
|--|-------------------------------|
| <b>Caso de Prueba de aceptación</b>  |                               |
| <b>Código:</b> HU4_P1  | <b>Historia de usuario:</b> 4 |
| <b>Nombre:</b> Configurar cantidad máxima de conexiones.   |                               |
| <b>Descripción:</b> Prueba para la funcionalidad Configurar cantidad máxima de conexiones.       |                               |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente.              |                               |
| <b>Pasos de ejecución:</b> El gestor intenta configurar cuantas conexiones aceptará el servidor. |                               |
| <b>Resultados esperado:</b> Se realiza la configuración sin generar error.                       |                               |
| <b>Evaluación de la prueba:</b> insatisfactoria.   |                               |

**Tabla 3.23** Prueba de aceptación Historia de usuario 4 prueba 2

|                                     |                               |
|-------------------------------------|-------------------------------|
| <b>Caso de Prueba de aceptación</b> |                               |
| <b>Código:</b> HU4_P2               | <b>Historia de usuario:</b> 4 |

|   |
|---|
| <b>Nombre:</b> Activar/desactivar SSL.  |
| <b>Descripción:</b> Prueba para la funcionalidad Activar/desactivar SSL.            |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente. |
| <b>Pasos de ejecución:</b> El gestor intenta activar/desactivar SSL.                |
| <b>Resultados esperado:</b> Se realiza la configuración sin generar error.          |
| <b>Evaluación de la prueba:</b> Satisfactoria.                                      |

**Tabla 3.24** Prueba de aceptación Historia de usuario 4 prueba 3

|  |                               |
|--|-------------------------------|
| <b>Caso de Prueba de aceptación</b>  |                               |
| <b>Código:</b> HU4_P3  | <b>Historia de usuario:</b> 4 |
| <b>Nombre:</b> Activar/desactivar encriptación de contraseñas.                                   |                               |
| <b>Descripción:</b> Prueba para la funcionalidad Activar/desactivar encriptación de contraseñas. |                               |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente.              |                               |
| <b>Pasos de ejecución:</b> El gestor intenta activar/desactivar la encriptación de contraseñas.  |                               |
| <b>Resultados esperado:</b> Se realiza la configuración sin generar error                        |                               |
| <b>Evaluación de la prueba:</b> Satisfactoria.   |                               |

**Tabla 3.25** Prueba de aceptación Historia de usuario 4 prueba 4

|   |                               |
|---|-------------------------------|
| <b>Caso de Prueba de aceptación</b>                                 |                               |
| <b>Código:</b> HU4_P4   | <b>Historia de usuario:</b> 4 |
| <b>Nombre:</b> Configurar tamaño de archivos para log de seguridad. |                               |

|   |
|---|
| <b>Descripción:</b> Prueba para la funcionalidad Configurar tamaño de archivos para log de seguridad.           |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente.                             |
| <b>Pasos de ejecución:</b> El gestor intenta configurar el tamaño máximo que podrán tener los archivos de logs. |
| <b>Resultados esperado:</b> Se realiza la configuración sin generar error.                                      |
| <b>Evaluación de la prueba:</b> Satisfactoria.  |

**Tabla 3.26** Prueba de aceptación Historia de usuario 4 prueba 5

|  |                               |
|--|-------------------------------|
| <b>Caso de Prueba de aceptación</b>  |                               |
| <b>Código:</b> HU4_P5  | <b>Historia de usuario:</b> 4 |
| <b>Nombre:</b> Configurar tiempo de vida de los logs de seguridad.   |                               |
| <b>Descripción:</b> Prueba para la funcionalidad Configurar tiempo de vida de los logs de seguridad.                     |                               |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente.                                      |                               |
| <b>Pasos de ejecución:</b> El gestor intenta configurar el tiempo que permanecerán los logs de seguridad en el servidor. |                               |
| <b>Resultados esperado:</b> Se realiza la configuración sin generar error.   |                               |
| <b>Evaluación de la prueba:</b> Satisfactoria.   |                               |

**Tabla 3.27** Prueba de aceptación Historia de usuario 4 prueba 6

|                                     |                               |
|-------------------------------------|-------------------------------|
| <b>Caso de Prueba de aceptación</b> |                               |
| <b>Código:</b> HU4_P6               | <b>Historia de usuario:</b> 4 |

|   |
|---|
| <b>Nombre:</b> Mostrar directorio de logs de seguridad.   |
| <b>Descripción:</b> Prueba para la funcionalidad Mostrar directorio de logs de seguridad.                   |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente.                         |
| <b>Pasos de ejecución:</b> El gestor intenta observar el directorio en donde radican los logs de seguridad. |
| <b>Resultados esperado:</b> Se realiza la configuración sin generar error.                                  |
| <b>Evaluación de la prueba:</b> Satisfactoria.  |

**Tabla 3.28** Prueba de aceptación Historia de usuario 4 prueba 7

|  |                               |
|--|-------------------------------|
| <b>Caso de Prueba de aceptación</b>  |                               |
| <b>Código:</b> HU4_P7  | <b>Historia de usuario:</b> 4 |
| <b>Nombre:</b> Activar/desactivar logs de seguridad.                                     |                               |
| <b>Descripción:</b> Prueba para la funcionalidad Activar/desactivar logs de seguridad.   |                               |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente.      |                               |
| <b>Pasos de ejecución:</b> El gestor intenta activar o desactivar los logs de seguridad. |                               |
| <b>Resultados esperado:</b> Se realiza la configuración sin generar error.               |                               |
| <b>Evaluación de la prueba:</b> Satisfactoria.   |                               |

**Tabla 3.29** Prueba de aceptación Historia de usuario 5 prueba 1

|                                     |
|-------------------------------------|
| <b>Caso de Prueba de aceptación</b> |
|-------------------------------------|

|   |                               |
|---|-------------------------------|
| <b>Código:</b> HU5_P1   | <b>Historia de usuario:</b> 5 |
| <b>Nombre:</b> Adicionar base de datos.   |                               |
| <b>Descripción:</b> Prueba para la funcionalidad Adicionar base de datos.   |                               |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente y el mismo no debe contener errores de configuración. Se debe introducir un nombre para la base de datos distinto de la cadena vacía y que no exista ninguna base de datos con dicho nombre. |                               |
| <b>Pasos de ejecución:</b> El gestor intenta adicionar una nueva base de datos.   |                               |
| <b>Resultados esperado:</b> Se crea una nueva base de datos en el servidor con el nombre proporcionado.   |                               |
| <b>Evaluación de la prueba:</b> Satisfactoria.  |                               |

**Tabla 3.30** Prueba de aceptación Historia de usuario 5 prueba 2

|   |                               |
|---|-------------------------------|
| <b>Caso de Prueba de aceptación</b>   |                               |
| <b>Código:</b> HU5_P2   | <b>Historia de usuario:</b> 5 |
| <b>Nombre:</b> Eliminar base de datos.  |                               |
| <b>Descripción:</b> Prueba para la funcionalidad Eliminar base de datos.  |                               |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente y el mismo no debe contener errores de configuración. Se debe seleccionar un nombre de una de las bases de datos existentes. |                               |
| <b>Pasos de ejecución:</b> El gestor intenta eliminar una base de datos.  |                               |
| <b>Resultados esperado:</b> Se elimina la base de datos cuyo nombre coincide con el elegido.  |                               |

**Evaluación de la prueba:** Satisfactoria.

**Tabla 3.31** Prueba de aceptación Historia de usuario 6 prueba 1

| <b>Caso de Prueba de aceptación</b>   |                               |
|---|-------------------------------|
| <b>Código:</b> HU6_P1   | <b>Historia de usuario:</b> 6 |
| <b>Nombre:</b> Adicionar usuario.   |                               |
| <b>Descripción:</b> Prueba para la funcionalidad Adicionar usuario.   |                               |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente y el mismo no debe contener errores de configuración. Se debe proporcionar un nombre de usuario distinto de la cadena vacía y que no exista otro usuario con dicho nombre. |                               |
| <b>Pasos de ejecución:</b> El gestor intenta crear un usuario nuevo en el servidor.   |                               |
| <b>Resultados esperado:</b> Se crea un nuevo usuario en el servidor con el nombre y la contraseña proporcionados.   |                               |
| <b>Evaluación de la prueba:</b> Satisfactoria.  |                               |

**Tabla 3.32** Prueba de aceptación Historia de usuario 6 prueba 2

| <b>Caso de Prueba de aceptación</b>   |                               |
|---|-------------------------------|
| <b>Código:</b> HU6_P2   | <b>Historia de usuario:</b> 6 |
| <b>Nombre:</b> Eliminar usuario.  |                               |
| <b>Descripción:</b> Prueba para la funcionalidad Eliminar usuario.  |                               |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente y el mismo no debe contener errores de configuración. Se debe seleccionar un nombre de |                               |



|   |
|---|
| usuario existente.  |
| <b>Pasos de ejecución:</b> El gestor intenta eliminar un usuario del servidor.              |
| <b>Resultados esperado:</b> Se elimina el usuario cuyo nombre coincide con el seleccionado. |
| <b>Evaluación de la prueba:</b> Satisfactoria.  |

**Tabla 3.33** Prueba de aceptación Historia de usuario 7 prueba 1

|  |                               |
|--|-------------------------------|
| <b>Caso de Prueba de aceptación</b>  |                               |
| <b>Código:</b> HU7_P1  | <b>Historia de usuario:</b> 7 |
| <b>Nombre:</b> Correr Script sobre el servidor.  |                               |
| <b>Descripción:</b> Prueba para la funcionalidad Correr Script sobre el servidor.  |                               |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente y el mismo no debe contener errores de configuración. Se debe seleccionar un archivo que contenga código SQL sin errores. Se debe autenticar en el servidor nuevamente para poder enviar el archivo SQL al mismo. |                               |
| <b>Pasos de ejecución:</b> El gestor intenta correr un script SQL sobre el servidor seleccionando un archivo con el código SQL.  |                               |
| <b>Resultados esperado:</b> El servidor ejecuta el código SQL y se muestran los datos en caso de ser solicitado por el usuario.  |                               |
| <b>Evaluación de la prueba:</b> Satisfactoria.   |                               |

### 3.5.2.3 Iteración 3

**Tabla 3.34** Prueba de aceptación Historia de usuario 8 prueba 1

|   |                               |
|---|-------------------------------|
| <b>Caso de Prueba de aceptación</b>   |                               |
| <b>Código:</b> HU8_P1   | <b>Historia de usuario:</b> 8 |
| <b>Nombre:</b> Mostrar configuración actual del servidor.                                   |                               |
| <b>Descripción:</b> Prueba para la funcionalidad Mostrar configuración actual del servidor. |                               |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente.         |                               |
| <b>Pasos de ejecución:</b> El gestor intenta observar la configuración actual del servidor. |                               |
| <b>Resultados esperado:</b> Se muestra la configuración actual del servidor.                |                               |
| <b>Evaluación de la prueba:</b> Satisfactoria.  |                               |

**Tabla 3.35** Prueba de aceptación Historia de usuario 9 prueba 1

|   |                               |
|---|-------------------------------|
| <b>Caso de Prueba de aceptación</b>   |                               |
| <b>Código:</b> HU9_P1   | <b>Historia de usuario:</b> 9 |
| <b>Nombre:</b> Realizar salva de base de datos.   |                               |
| <b>Descripción:</b> Prueba para la funcionalidad Realizar salva de base de datos.   |                               |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente. Y no deben existir errores de configuración. Se debe seleccionar una base de datos existente. |                               |
| <b>Pasos de ejecución:</b> El gestor intenta realizar la salva a una base de datos del servidor.  |                               |
| <b>Resultados esperado:</b> Se realiza una salva de la base de datos seleccionada.  |                               |

**Evaluación de la prueba:** Satisfactoria.

**Tabla 3.36** Prueba de aceptación Historia de usuario 10 prueba 1

|   |                                |
|---|--------------------------------|
| <b>Caso de Prueba de aceptación</b>   |                                |
| <b>Código:</b> HU10_P1  | <b>Historia de usuario:</b> 10 |
| <b>Nombre:</b> Realizar restauración de base de datos.  |                                |
| <b>Descripción:</b> Prueba para la funcionalidad Realizar restauración de base de datos.  |                                |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente. Y no deben existir errores de configuración. Se debe seleccionar un archivo de salva existente y con una correcta sintaxis. |                                |
| <b>Pasos de ejecución:</b> El gestor intenta realizar la restauración a una base de datos del servidor.   |                                |
| <b>Resultados esperado:</b> Se realiza la restauración de la base de datos seleccionada.  |                                |
| <b>Evaluación de la prueba:</b> Satisfactoria.  |                                |

### 3.5.3 Pruebas de aceptación para MySQLManager

#### 3.5.3.1 Iteración 1

**Tabla 3.37** Prueba de aceptación Historia de usuario 10 prueba 1

|   |                                |
|---|--------------------------------|
| <b>Caso de Prueba de aceptación</b>   |                                |
| <b>Código:</b> HU10_P1  | <b>Historia de usuario:</b> 10 |
| <b>Nombre:</b> Conexión a servidor mediante SSH.  |                                |
| <b>Descripción:</b> Prueba para la funcionalidad para conectarse vía SSH a un servidor para la transferencia de archivos. |                                |

|  |
|--|
| <b>Condiciones de ejecución:</b> Se debe tener instalado OpenSSH en la máquina local y en el servidor.   |
| <b>Pasos de ejecución:</b> El gestor se conecta al servidor de base de datos mediante SSH introduciendo nombre de usuario y contraseña del servidor, luego selecciona el fichero a transferir. |
| <b>Resultados esperado:</b> Se realiza la conexión sin generar error y se transfiere el fichero.   |
| <b>Evaluación de la prueba:</b> Satisfactoria.   |

**Tabla 3.38** Prueba de aceptación Historia de usuario 11 prueba 1

|  |                                |
|--|--------------------------------|
| <b>Caso de Prueba de aceptación</b>  |                                |
| <b>Código:</b> HU11_P1   | <b>Historia de usuario:</b> 11 |
| <b>Nombre:</b> Configurar acceso al servidor.  |                                |
| <b>Descripción:</b> Prueba para la funcionalidad Configurar acceso al servidor.  |                                |
| <b>Condiciones de ejecución:</b> El gestor de configuración debe haberse conectado al servidor de forma correcta.                                    |                                |
| <b>Pasos de ejecución:</b> El gestor determina la forma en que se conectan los usuarios al servidor de base de datos a través de reglas de conexión. |                                |
| <b>Resultados esperado:</b> Se realiza la configuración de acceso sin errores.   |                                |
| <b>Evaluación de la prueba:</b> Satisfactoria.   |                                |

**Tabla 3.39** Prueba de aceptación Historia de usuario 12 prueba 1

|                                     |                                |
|-------------------------------------|--------------------------------|
| <b>Caso de Prueba de aceptación</b> |                                |
| <b>Código:</b> HU12_P1              | <b>Historia de usuario:</b> 12 |

|   |
|---|
| <b>Nombre:</b> Configurar el tamaño de la memoria compartida.   |
| <b>Descripción:</b> Prueba para la funcionalidad Configurar el tamaño de la memoria compartida.                   |
| <b>Condiciones de ejecución:</b> El gestor de configuración debe haberse conectado al servidor de forma correcta. |
| <b>Pasos de ejecución:</b> El gestor configura la cantidad de memoria RAM que usará el servidor.                  |
| <b>Resultados esperado:</b> Se realiza la configuración sin errores.  |
| <b>Evaluación de la prueba:</b> Satisfactoria.  |

**Tabla 3.40** Prueba de aceptación Historia de usuario 12 prueba 2

|  |                                |
|--|--------------------------------|
| <b>Caso de Prueba de aceptación</b>  |                                |
| <b>Código:</b> HU12_P2   | <b>Historia de usuario:</b> 12 |
| <b>Nombre:</b> Configurar caché de consulta.   |                                |
| <b>Descripción:</b> Prueba para la funcionalidad Configurar caché de consulta.                       |                                |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente.                  |                                |
| <b>Pasos de ejecución:</b> El gestor intenta configurar la cantidad de memoria dedicada a consultas. |                                |
| <b>Resultados esperado:</b> Se realiza la configuración sin generar error.                           |                                |
| <b>Evaluación de la prueba:</b> insatisfactoria.   |                                |

**Tabla 3.41** Prueba de aceptación Historia de usuario 12 prueba 3

|   |                                |
|---|--------------------------------|
| <b>Caso de Prueba de aceptación</b>   |                                |
| <b>Código:</b> HU12_P3  | <b>Historia de usuario:</b> 12 |
| <b>Nombre:</b> Configurar límite de caché de consulta.  |                                |
| <b>Descripción:</b> Prueba para la funcionalidad Configurar límite de caché de consulta.          |                                |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente.               |                                |
| <b>Pasos de ejecución:</b> El gestor intenta configurar el tamaño límite de la caché de consulta. |                                |
| <b>Resultados esperado:</b> Se realiza la configuración sin generar error.                        |                                |
| <b>Evaluación de la prueba:</b> Satisfactoria.  |                                |

### 3.5.3.2 Iteración 2

**Tabla 3.42** Prueba de aceptación Historia de usuario 13 prueba 1

|  |                                |
|--|--------------------------------|
| <b>Caso de Prueba de aceptación</b>  |                                |
| <b>Código:</b> HU13_P1   | <b>Historia de usuario:</b> 13 |
| <b>Nombre:</b> Configurar cantidad máxima de conexiones.   |                                |
| <b>Descripción:</b> Prueba para la funcionalidad Configurar cantidad máxima de conexiones.       |                                |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente.              |                                |
| <b>Pasos de ejecución:</b> El gestor intenta configurar cuantas conexiones aceptará el servidor. |                                |
| <b>Resultados esperado:</b> Se realiza la configuración sin generar error.                       |                                |

**Evaluación de la prueba:** Satisfactoria.

**Tabla 3.43** Prueba de aceptación Historia de usuario 13 prueba 2

|   |                                |
|---|--------------------------------|
| <b>Caso de Prueba de aceptación</b>   |                                |
| <b>Código:</b> HU13_P2  | <b>Historia de usuario:</b> 13 |
| <b>Nombre:</b> Activar/desactivar SSL.  |                                |
| <b>Descripción:</b> Prueba para la funcionalidad Activar/desactivar SSL.            |                                |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente. |                                |
| <b>Pasos de ejecución:</b> El gestor intenta activar/desactivar SSL.                |                                |
| <b>Resultados esperado:</b> Se realiza la configuración sin generar error.          |                                |
| <b>Evaluación de la prueba:</b> Satisfactoria.                                      |                                |

**Tabla 3.44** Prueba de aceptación Historia de usuario 13 prueba 3

|   |                                |
|---|--------------------------------|
| <b>Caso de Prueba de aceptación</b>   |                                |
| <b>Código:</b> HU13_P3  | <b>Historia de usuario:</b> 13 |
| <b>Nombre:</b> Mostrar estado de la encriptación de contraseñas.  |                                |
| <b>Descripción:</b> Prueba para la funcionalidad Mostrar estado de la encriptación de contraseñas.              |                                |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente.                             |                                |
| <b>Pasos de ejecución:</b> El gestor intenta observar la configuración actual de la encriptación de contraseña. |                                |

|  |
|--|
| <b>Resultados esperado:</b> Se realiza la configuración sin generar error. |
| <b>Evaluación de la prueba:</b> Satisfactoria.                             |

**Tabla 3.45** Prueba de aceptación Historia de usuario 13 prueba 4

|   |                                |
|---|--------------------------------|
| <b>Caso de Prueba de aceptación</b>   |                                |
| <b>Código:</b> HU13_P4  | <b>Historia de usuario:</b> 13 |
| <b>Nombre:</b> Configurar tamaño de archivos para log de seguridad.   |                                |
| <b>Descripción:</b> Prueba para la funcionalidad Configurar tamaño de archivos para log de seguridad.           |                                |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente.                             |                                |
| <b>Pasos de ejecución:</b> El gestor intenta configurar el tamaño máximo que podrán tener los archivos de logs. |                                |
| <b>Resultados esperado:</b> Se realiza la configuración sin generar error.                                      |                                |
| <b>Evaluación de la prueba:</b> Satisfactoria.  |                                |

**Tabla 3.46** Prueba de aceptación Historia de usuario 13 prueba 5

|  |                                |
|--|--------------------------------|
| <b>Caso de Prueba de aceptación</b>  |                                |
| <b>Código:</b> HU13_P5   | <b>Historia de usuario:</b> 13 |
| <b>Nombre:</b> Configurar tiempo de vida de los logs de seguridad.                                   |                                |
| <b>Descripción:</b> Prueba para la funcionalidad Configurar tiempo de vida de los logs de seguridad. |                                |



|  |
|--|
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente.                                      |
| <b>Pasos de ejecución:</b> El gestor intenta configurar el tiempo que permanecerán los logs de seguridad en el servidor. |
| <b>Resultados esperado:</b> Se realiza la configuración sin generar error.   |
| <b>Evaluación de la prueba:</b> Satisfactoria.   |

**Tabla 3.47** Prueba de aceptación Historia de usuario 13 prueba 6

|   |                                |
|---|--------------------------------|
| <b>Caso de Prueba de aceptación</b>   |                                |
| <b>Código:</b> HU13_P6  | <b>Historia de usuario:</b> 13 |
| <b>Nombre:</b> Mostrar directorio de logs de seguridad.   |                                |
| <b>Descripción:</b> Prueba para la funcionalidad Mostrar directorio de logs de seguridad.                         |                                |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente.                               |                                |
| <b>Pasos de ejecución:</b> El gestor intenta observar el directorio en donde se encuentran los logs de seguridad. |                                |
| <b>Resultados esperado:</b> Se realiza la configuración sin generar error.  |                                |
| <b>Evaluación de la prueba:</b> Satisfactoria.  |                                |

**Tabla 3.48** Prueba de aceptación Historia de usuario 14 prueba 1

|   |                                |
|---|--------------------------------|
| <b>Caso de Prueba de aceptación</b>     |                                |
| <b>Código:</b> HU14_P1                  | <b>Historia de usuario:</b> 14 |
| <b>Nombre:</b> Adicionar base de datos. |                                |

|   |
|---|
| <b>Descripción:</b> Prueba para la funcionalidad Adicionar base de datos.   |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente y el mismo no debe contener errores de configuración. Se debe introducir un nombre para la base de datos distinto de la cadena vacía y que no exista ninguna base de datos con dicho nombre. |
| <b>Pasos de ejecución:</b> El gestor intenta adicionar una nueva base de datos.   |
| <b>Resultados esperado:</b> Se crea una nueva base de datos en el servidor con el nombre proporcionado.   |
| <b>Evaluación de la prueba:</b> Satisfactoria.  |

**Tabla 3.49** Prueba de aceptación Historia de usuario 14 prueba 2

|   |                                |
|---|--------------------------------|
| <b>Caso de Prueba de aceptación</b>   |                                |
| <b>Código:</b> HU14_P2  | <b>Historia de usuario:</b> 14 |
| <b>Nombre:</b> Eliminar base de datos.  |                                |
| <b>Descripción:</b> Prueba para la funcionalidad Eliminar base de datos.  |                                |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente y el mismo no debe contener errores de configuración. Se debe seleccionar un nombre de una de las bases de datos existentes. |                                |
| <b>Pasos de ejecución:</b> El gestor intenta eliminar una base de datos.  |                                |
| <b>Resultados esperado:</b> Se elimina la base de datos cuyo nombre coincide con el elegido.  |                                |
| <b>Evaluación de la prueba:</b> Satisfactoria.  |                                |

**Tabla 3.50** Prueba de aceptación Historia de usuario 15 prueba 1

|   |                                |
|---|--------------------------------|
| <b>Caso de Prueba de aceptación</b>   |                                |
| <b>Código:</b> HU15_P1  | <b>Historia de usuario:</b> 15 |
| <b>Nombre:</b> Adicionar usuario.   |                                |
| <b>Descripción:</b> Prueba para la funcionalidad Adicionar usuario.   |                                |
| <b>Condiciones de ejecución:</b> Se debe estar conectado al servidor correctamente y el mismo no debe contener errores de configuración. Se debe proporcionar un nombre de usuario distinto de la cadena vacía y que no exista otro usuario con dicho nombre. |                                |
| <b>Pasos de ejecución:</b> El gestor intenta crear un usuario nuevo en el servidor.   |                                |
| <b>Resultados esperado:</b> Se crea un nuevo usuario en el servidor con el nombre y la contraseña proporcionados.   |                                |
| <b>Evaluación de la prueba:</b> Satisfactoria.  |                                |

### 3.6 Conclusiones

En el presente capítulo se realizaron los modelos necesarios para llevar a cabo el proceso de implementación del sistema. Se mostraron de forma detallada las tareas seleccionadas por iteración para un mejor entendimiento. Por último se realizaron las pruebas de aceptación de cada una de las funcionalidades del software resultando satisfactorias con excepción de tres funcionalidades. Los problemas presentados fueron solventados inmediatamente procurando que errores del mismo tipo no se volvieran a repetir como lo estipula la metodología XP. Por lo que se concluye afirmando que se realizó el desarrollo exitoso de dos módulos para SASEP que permiten la configuración de servidores Postgres y MySQL de forma remota y a través de una interfaz gráfica. Con esto se soluciona la problemática planteada en este trabajo.

## Conclusiones Generales

Con el desarrollo del presente trabajo y la implementación de los módulos PostgresManager y MySQLManager integrables a la Suite Administrativa para Servidores de Entornos Productivos, se concluye afirmando que:

- Se logró el desarrollo de dos módulos de configuración de servidores de bases de datos Postgres y MySQL integrables a SASEP. Dichos módulos lograron facilitar el proceso de configuración de los servidores de bases de datos en la UCI.
- Se elaboró el diseño metodológico de la investigación, dando paso al análisis de las principales herramientas utilizadas a nivel mundial para la administración y configuración de servidores Postgres y MySQL. Demostrando que dichas herramientas no solventan los problemas existentes en la UCI en cuanto a la configuración de servidores respecta.
- Se determinó un total de cuarenta y cuatro funcionalidades y las características del sistema necesarias para el adecuado funcionamiento de los módulos de modo que se brinde una solución más factible al problema planteado.
- Se confeccionaron los diferentes artefactos surgidos en las fases de desarrollo de la metodología XP. Lo que permitió el paso a la fase de Iteraciones en donde se implementaron las funcionalidades del software.
- Se realizaron las pruebas de aceptación a cada funcionalidad de ambos módulos, arrojando resultados satisfactorios.
- Los gestores del Centro FORTES aceptaron las aplicaciones con entusiasmo, alegando lo útil de las mismas al permitir de forma sencilla realizar las configuraciones más importantes de los servidores.

## Recomendaciones

Los módulos PostgresManager y MySQLManager fueron aceptados con entusiasmo por los gestores de configuración del Centro FORTES. Sin embargo existen algunas recomendaciones que se exponen a continuación:

- Implementar una funcionalidad que permita observar el consumo de recursos del servidor.
- Añadir nuevas funcionalidades que permitan configurar otras variables del servidor.
- Agregar a los módulos un editor de código SQL.

# Referencias Bibliográficas

1. **Menéndez Barzanallana, Rafael.** <http://www.um.es/docencia>. [En línea] 2007. <http://www.um.es/docencia>.
2. **MySQL AB.** [mysql.com](http://mysql.com). [En línea] 2012. <http://mysql.com>.
3. **Computer Audio Systems Integrator.** [cavsi.com](http://cavsi.com). [En línea] 2006. <http://www.cavsi.com/preguntasrespuestas/que-es-un-sistema-gestor-de-bases-de-datos-o-sgbd/>.
4. **Computer Audio Video Systems Integrator.** [cavsi.com](http://cavsi.com). [En línea] 2006. <http://www.cavsi.com/preguntasrespuestas/que-es-un-sistema-gestor-de-bases-de-datos-o-sgbd/>.
5. **ECURED.** <http://www.ecured.cu>. [En línea] 2012. [http://www.ecured.cu/index.php/Arquitectura\\_Cliente\\_Servidor](http://www.ecured.cu/index.php/Arquitectura_Cliente_Servidor).
6. **Pecos, Daniel.** [En línea] 07 de 2002. [http://danielpecos.com/docs/mysql\\_postgres/x15.html#AEN17](http://danielpecos.com/docs/mysql_postgres/x15.html#AEN17).
7. **MySQL.** [En línea] 06 de junio de 2012. [Citado el: 06 de junio de 2012.] <http://www.mysql.com/documentation/index.html>.
8. **Daniel Pecos.** [En línea] 06 de junio de 2012. [Citado el: 06 de junio de 2012.] [http://www.danielpecos.com/docs/mysql\\_postgres/index.html](http://www.danielpecos.com/docs/mysql_postgres/index.html).
9. **OpenBSD.** OpenBSD. [En línea] 2004. <http://www.openssh.org/es/index.html>.
10. **Webmin.** [En línea] 03 de 04 de 2012. <http://www.webmin.com/>.
11. **PhpPgAdmin.** phpPgAdmin. [En línea] 2012. <http://phppgadmin.sourceforge.net/doku.php?id=start>.
12. **PgAdmin.** pgAdmin. [En línea] 2011. <http://www.pgadmin.org/>.
13. **MySQL AB.** [dev.mysql.com](http://dev.mysql.com). [En línea] 2011. [dev.mysql.com/doc/administrator/](http://dev.mysql.com/doc/administrator/).
14. **Stroustrup, Bjarne.** *The C++ programming language*. 1997.
15. **Nokia.** QT. [En línea] 2012. [Citado el: 05 de 05 de 2012.] <http://qt.nokia.com/products/developer-tools>.
16. **ECURED.** ECURED. [En línea] 06 de 05 de 2012. [Citado el: 13 de 05 de 2012.] [www.ecured.cu/index.php/IDE\\_de\\_](http://www.ecured.cu/index.php/IDE_de_).
17. **The Eclipse Foundation.** [eclipse.org](http://eclipse.org). [En línea] 2012. [Citado el: 02 de junio de 2012.] <http://eclipse.org>.
18. **Jennic Wireless Microcontrollers .** Jennic Wireless Microcontrollers . [En línea] 2012. [Citado el: 02 de junio de 2012.] [http://www.jennic.com/support/user\\_guides/jn-ug-3028\\_codeblocks\\_ide](http://www.jennic.com/support/user_guides/jn-ug-3028_codeblocks_ide).
19. **Joskowicz, José.** *Reglas y prácticas en XP*. Vigo, España : s.n., 2008.
20. **Schwaber, K.** *Advanced Development Methods. SCRUM Development Process Retrieved*. 2006.
21. **Schwaber, K.** *Agile software development with Scrum*. 2001.
22. **Chin, Gary.** *Agile Project Management: How to Succeed in the Face of Changing Project Requirements*. 2004.
23. **ECURED.** ECURED. [En línea] [http://www.ecured.cu/index.php/Patr%C3%B3n\\_de\\_dise%C3%B1o\\_de\\_software](http://www.ecured.cu/index.php/Patr%C3%B3n_de_dise%C3%B1o_de_software).

24. **Astudillo, Marcello Visconti y Hernán.** *Fundamentos de Ingeniería de Software.* 2008.
25. **Pressman, R. S.** *Ingeniería del software.* Ciudad Habana : Félix Varela, 2005.
26. **.R, Allende.** *Desarrollo de Portales y Extranet con Plone.* 2006.
27. **BECK, F. M. Y. K.** *Planeando en Programación Extrema.* 2000.
28. <http://doc.qtsoftware.com/qtcreator-1.1/creator-quick-tour.html>. [En línea]
29. [http://www.ecured.cu/index.php/Qt\\_Creator](http://www.ecured.cu/index.php/Qt_Creator). [En línea] ECURED.  
[http://www.ecured.cu/index.php/Qt\\_Creator](http://www.ecured.cu/index.php/Qt_Creator).
30. **Saavedra Gutierrez, Jorge A.** [En línea] 05 de 2007.  
[http://jorgesaavedra.wordpress.com/2007/05/05/lenguajes-de-programacion/..](http://jorgesaavedra.wordpress.com/2007/05/05/lenguajes-de-programacion/)
31. **Microsoft.** <http://microsoft.com>. [En línea] 2012. [Citado el: 13 de 05 de 2012.]  
<http://microsoft.com>.
32. **Presman, R. S.** *Ingeniería del Software, un enfoque práctico.* 2002.
33. **WxWidgets.** WxWidgets. [En línea] [Citado el: 05 de 14 de 2012.]  
<http://www.wxwidgets.org/>.
34. **Perl.** Perl. [En línea] 2012. [Citado el: 13 de 05 de 2012.] [www.perl.org/](http://www.perl.org/).
35. **C++ Builder.** [En línea] 2012. [Citado el: 14 de 05 de 2012.]  
[www.embarcadero.com/products/cbuilder](http://www.embarcadero.com/products/cbuilder)En caché - Similares .
36. **Alvarez, Miguel Angel.** [En línea] 2003.  
<http://www.desarrolloweb.com/articulos/1325.php..>

# Anexos

## **Anexo 1.** Entrevista realizada a gestores de configuración del Centro FORTES.

Nombre del entrevistado: \_\_\_\_\_

Área a la que pertenece: \_\_\_\_\_

Buenos días, como parte de mi investigación estoy realizando una entrevista con el objetivo de obtener una serie de datos que posteriormente sirvan de apoyo en el desarrollo de dos módulos integrables a SASEP, que faciliten la configuración de servidores Postgres y MySQL.

1. ¿Qué tarea realiza dentro de la gestión de la configuración?
2. ¿Qué servidor de base de datos utiliza?
3. ¿Se apoya en alguna herramienta para realizar las configuraciones del servidor remoto?
4. De usar alguna herramienta, decir si presenta inconvenientes.
5. De no usar ninguna herramienta, decir las causas.