



UNIVERSIDAD DE VALPARAÍSO  
Facultad de Ciencias  
Departamento de Computación  
Ingeniería Civil en Informática

**Desarrollo de una Plataforma para la Solicitud y  
Gestión de Requerimientos y SCM  
Propuesta de Trabajo de Título**

Alejandro Alvarez Ahumada  
Profesor Guía: Carlos Becerra Castro

14 de abril de 2012

## **Resumen**

La Dirección de Servicios de Información y Computación (DISICO) de la Universidad de Valparaíso durante los últimos años ha estado en constante crecimiento y en busca de mejoras que le permitan brindar un mejor servicio. Aunque en este poco tiempo son muchas las mejoras que se han hecho, aún quedan aspectos por mejorar, algunos de estos son los procesos relacionados a las solicitudes de requerimientos y solicitudes de cambios, para las cuales ya se han diseñado procedimientos y metodologías, sin embargo se carece de una herramienta que permita la automatización de estas. El propósito de este trabajo de título es dar solución a dicho problema mediante el desarrollo de una plataforma que permita automatizar los procedimientos actuales. Los principales resultados que se esperan son disminuir el tiempo y esfuerzo invertido en la aplicación de las metodologías que existen actualmente.

# Índice general

1.	Introducción . . . . .	3
2.	Definición del Problema . . . . .	4
2.1.	Problemas en las Solicitudes de Requerimientos . . . . .	4
2.2.	Problemas en la Gestión de Requerimientos en los Proyectos . . . . .	5
2.3.	Problemas en la Solicitudes de SCM . . . . .	5
2.4.	Solución Propuesta . . . . .	6
3.	Objetivos . . . . .	6
3.1.	Objetivos Generales . . . . .	6
3.2.	Objetivos Específicos . . . . .	6
4.	Metodología . . . . .	7
5.	Planificación . . . . .	10
6.	Recursos . . . . .	11
6.1.	Recursos de Software . . . . .	11
6.2.	Recursos de Hardware . . . . .	12

# Índice de figuras

1.	Esquema de la Fase de Análisis . . . . .	7
2.	Esquema de la Fase de Diseño . . . . .	8
3.	Esquema de la Fase de Implementación . . . . .	9
4.	Esquema de la Fase de Pruebas . . . . .	9

# 1. Introducción

La Dirección de Servicios de Información y Computación (DISICO) [9], con el objetivo de dar una mejor la calidad de servicio, actualmente esta dividida en 3 áreas: Área de Sistemas Financiero-Contables, Área de Desarrollo y Área de Redes, Comunicaciones y Soporte. Las funciones de las que DISICO es responsable se encuentran descritas en detalle en el Decreto 427 [6], siendo algunas de estas:

- Administrar todo el procesamiento de datos y documentación, que por medio de sistemas computacionales requiera la Universidad para su toma de decisiones.
- Establecer un catastro renovable en el tiempo de los requerimientos informáticos de los usuarios de las distintas unidades de la Universidad.
- Interrelacionar los sistemas con las otras áreas de desarrollo de la organización.
- Mantener en constante actualización los sistemas de información y propender a la creación y desarrollo de sistemas en ambientes corporativos.
- Establecer pautas para obtener una estandarización en los sistemas computacionales.

DISICO se encuentra en un proceso constante de crecimiento y mejora, para dar un mejor servicio a toda la comunidad de la Universidad de Valparaíso. En esta búsqueda constante de aspectos a mejorar, es que se han realizado mejoras como por ejemplo, el Desarrollo de Metodologías para Software Quality Assurance (SQA) y Software Configuration Management (SCM) [8]. Pero a pesar de esto se han detectado falencias tanto en los procesos de solicitudes de requerimientos, las cuales se realizan principalmente a través del correo institucional y en las solicitudes de cambios, las que cuentan con la metodología antes mencionada, pero aun no cuentan con una herramienta que permita su automatización.

Por tanto se plantea dar solución a dichas falencias y los problemas que estas producen a través del desarrollo de una plataforma que le permita automatizar los procesos existentes, la cual permitirá un mejor control tanto del ciclo de vida de las tareas que se desarrollan en DISICO, como de quienes las realizan.

Este documento se estructurara de la siguiente forma:

- **En la sección 2** se analiza de forma mas detallada las falencias encontradas y los problemas que estas producen, junto con lo que se propone para dar solución a las mismas.
- **En la sección 3** encontrara los objetivos generales y específicos que se esperan cumplir con el desarrollo de este trabajo.
- **En la sección 4** se describe la Metodología que se utilizara para llevar a cabo este trabajo.
- **En la sección 5** se encuentra la Planificación.

- **En la sección 6** se describen los recursos necesarios para el desarrollo de este trabajo.

## 2. Definición del Problema

La Dirección de Servicios de Información y Computación (DISICO) de la Universidad de Valparaíso en su constante búsqueda por mejorar ha detectado algunos problemas, los cuales se nombran y detallan a continuación:

### 2.1. Problemas en las Solicitudes de Requerimientos

Las solicitudes de requerimientos que se reciben en DISICO, actualmente deben ser enviadas a través del correo electrónico institucional al Jefe de Área, quien revisa y evalúa cada solicitud para determinar quien realizar la tarea correspondiente, luego reenvía el correo original a dicha persona, en algunos casos adjuntando información adicional sobre como completar la tarea o en caso que la solicitud enviada en el correo este incompleta como para poder comprender o resolver completamente el requerimiento, se reenvía al funcionario el correo que ha enviado junto con las observaciones correspondientes

Aunque a primera vista parece simple, esto conlleva los siguientes problemas:

- El correo del Jefe de Área puede verse saturado con cientos de correos en un mismo día. Sobre todo en periodos críticos como son el inicio y termino de cada semestre.
- Los requerimientos quedan almacenados en emails y no se cuenta con ninguna estructura de datos que permita, mejores búsquedas, priorización de tareas, análisis de avance de las tareas, análisis posteriores acerca de velocidad de desarrollo o productividad, establecer dependencias entre tareas, complejidad de las tareas, etc.
- El Jefe de Área no tiene una visión clara de cuales ni cuantas tareas se han asignado a cada miembro del equipo.
- El Jefe de Área no tiene una visión clara sobre cuales ni cuantas tareas están aún en espera, en desarrollo o finalizadas.
- Los funcionarios que envían solicitudes de requerimientos desconocen en que estado se encuentran, cuanto se ha avanzado, ni la prioridad que esta tiene, y menos las complicaciones que pueden haber surgido, por lo que deben estar consultando por teléfono o esperando a recibir una respuesta.
- Entre tanto correo muchas veces se pierden los requerimientos debido a que el responsable leyó el correo y dejó la tarea para mas tarde debido a que en ese momento aún se encontraba realizando otras tareas.

## 2.2. Problemas en la Gestión de Requerimientos en los Proyectos

Además de los requerimientos que se reciben diariamente en DISICO, existen requerimientos que superan esta categoría, los cuales pasan a convertirse en proyectos, estos nuevos proyectos constan de un conjunto de requerimientos que son capturados por los responsables de dicho proyecto, quienes los agrupan y calendarizan mediante una Carta Gantt, la cual debe ser entregada al Jefe de Área.

Este sistema también presenta algunas falencias, las cuales son similares a las presentadas en el punto anterior y se detallan a continuación:

- Aunque la Carta Gantt es un instrumento importante, no le permite al Jefe de Área conocer cuales puntos ya han sido completados, en cuales se está trabajando actualmente y cual es el nivel de avance de estos.
- Tampoco tiene un registro de las causas que pueden estar demorando una tarea.
- Esto también implica que el Jefe de Área no tiene una visión clara de la productividad ni del avance de cada uno de los involucrados en un proyecto.

## 2.3. Problemas en la Solicitudes de SCM

Antiguamente DISICO no contaba con un procedimiento formal para los procesos de SCM, aunque se intentaba llevar a cabo un procedimiento similar al de asignación de tareas, pero con algunas pequeñas diferencias. Los problemas de este procedimiento fueron analizados y corregidos en el Trabajo de Título Desarrollo de Metodologías de SQA y SCM para la Dirección de Servicios de Información y Computación [8], en el cual se propuso una nueva metodología de SCM, en la cual ya fueron considerados la baja disponibilidad de tiempo del personal para aplicar estos procesos y la falta de recursos para las mismas.

En dicho trabajo para lograr un proceso de SCM de calidad se creó toda la documentación pertinente a los cambios que se realicen durante el desarrollo de un proyecto, pero siempre buscando que ésta se realice de forma rápida y precisa, apoyada de templates que exijan la información mínima para que la metodología cumpla sus objetivos.

Sin embargo aun con esta nueva metodología se tienen algunos problemas los cuales se detallan a continuación:

- La metodología propuesta no logra ser lo suficientemente ágil, debido a que creación de la documentación que exige, no se encuentra automatizada.
- La documentación se realiza en plantillas Word, por lo que la persona deberá descargar la plantilla si es que no la posee, abrirla con Word (o similar) para editarla, luego imprimirla y archivarla o si se desea compartirlas mantenerlas versionadas en SVN, de donde el resto de los involucrados en los cambios

deberán estar descargando la ultima versión para conocer los nuevos cambios, y actualizarlas si es necesario.

- El uso de estas plantillas Word no facilita la búsqueda rápida de información sobre cambios pasados, personas involucradas en los cambios, etc. Debido a que se debe descargar los últimos cambios de los documentos, buscar por los nombres de los documentos, cuales posiblemente están relacionados a la búsqueda en particular y abrirlos para buscar dentro de estos lo que se necesita.

Todas estas problemáticas han provocado que no se esté sacando el máximo provecho a la metodología diseñada y como podemos ver la causa principal es que no se posee una herramienta adecuada que le de soporte.

## **2.4. Solución Propuesta**

Como se observa la mayoría de los problemas identificados, se encuentran ampliamente relacionados, esto se debe a que tanto las solicitudes de requerimientos, los requerimientos de un proyecto y las solicitudes de cambios, derivan en tareas que debe realizar el personal de DISICO, y sobre las cuales existe escaso control y documentación actualmente. Dado lo anterior se propone desarrollar una plataforma de solicitud y gestión de requerimientos y SCM, la cual integrara estos 3 puntos (los requerimientos diarios, los proyectos y las peticiones de cambios) en una plataforma que ayudará tanto en la comunicación, documentación y control de estas actividades dentro del equipo, permitiendo el mapeo de estas, a tareas concretas que deben llevarse a cabo. Y que también, integre y brinde el soporte a la metodología de SCM diseñada para DISICO[8].

## **3. Objetivos**

### **3.1. Objetivos Generales**

El objetivo es el desarrollo de una aplicación para la solicitud y gestión de requerimientos y SCM para la Dirección de Servicios de Información y Computación de la Universidad de Valparaíso (DISICO).

### **3.2. Objetivos Específicos**

- Dar soporte mediante una aplicación web, a la Metodología de SCM diseñada para DISICO [8].
- Permitir monitorear el ciclo de vida completo de cada solicitud de requerimiento y de cambio que se recibe en DISICO.
- Mejorar el control que tiene el Jefe de Área sobre las labores que se encuentra desempeñando actualmente cada miembro del equipo.



- Mantener un historial de todas los requerimientos que se han completado, para permitir mejorar las estimaciones de tiempo y esfuerzo, para futuros requerimientos de similares características.
- Obtener mediciones cuantitativas de la productividad del equipo de desarrollo.

## 4. Metodología

Para el desarrollo de este proyecto, se ha optado por un modelo en cascada tradicional, dado que los entregables requeridos están fuertemente marcados por este modelo de cascada[7]. Además se mezcla con aspectos de la metodología iterativo e incremental, con lo que se espera solventar la rigidez del Modelo en Cascada y permitir la codificación desde fases tempranas del proyecto. Por lo que el proceso se estructurara de la siguiente forma:

- **Fase de Análisis:** Esta fase estará netamente enfocada a la investigación del marco conceptual, situación actual, búsqueda y análisis de herramientas y soluciones existentes, como también la definición de la mayor parte de los requerimientos. Al final de esta fase se debe tener un documento de definición de requerimientos y el Informe 2 del Seminario de Título 1.

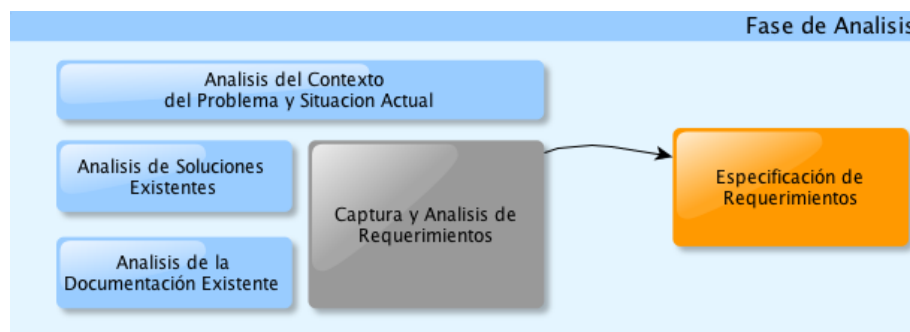


Figura 1: Esquema de la Fase de Análisis

- **Fase de Diseño:** En esta fase se trabajara principalmente en generar un diseño lógico, arquitectónico, de presentación, de datos y de pruebas para la aplicación. La forma en que se llevara a cabo esto será la siguiente:
  - Se genera una versión inicial de los modelos requeridos.
  - Se implementa un prototipo, que permitirá dar un primer vistazo al cliente, descubrir nuevos requerimientos y problemas en el diseño actual.

Luego se continuara iterando en esta fase de la siguiente forma:

- Se analizan y documentan los nuevos requerimientos descubiertos.
- Se refinan los modelos generados.

- Se refina el prototipo.
- Se revisa el nuevo prototipo con el cliente y se obtiene el feedback necesario.

Al final de esta fase se deben obtener los documentos de diseño necesarios y una estructura base de la cual partir a codificar el proyecto. Además del Tercer Informe para la Asignatura de Trabajo de Título 1.

Finalizada esta fase ya no se realizarán más cambios en los requerimientos.

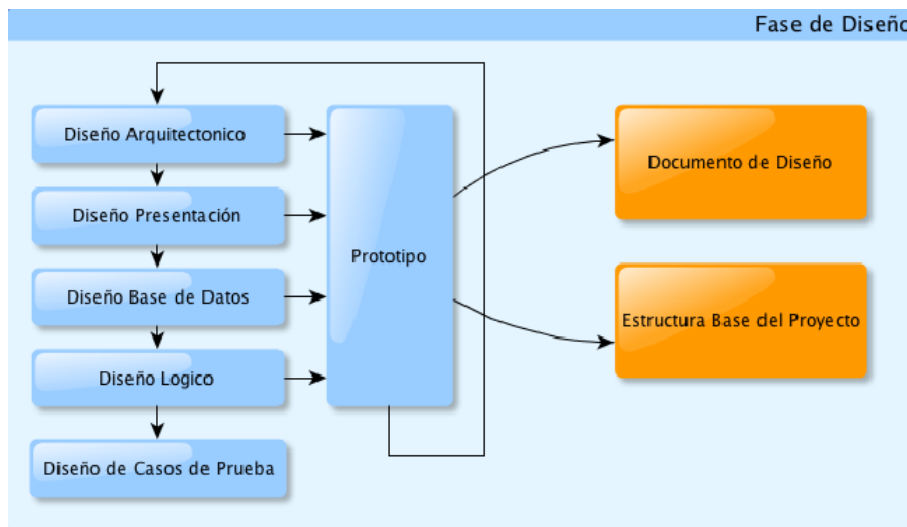


Figura 2: Esquema de la Fase de Diseño

- **Fase de Implementación:** En esta fase se comenzará a desarrollar todas las funcionalidades necesarias. La codificación se organizará de forma incremental, para lo cual se definen 3 incrementos los cuales son:
  - **Manejo de Tareas Diarias:** Consiste en implementar, todas las funcionalidades relacionados al manejo de las tareas diarias, tanto creación, envío y seguimiento.
  - **Manejo de Proyectos:** Consiste en incorporar el concepto de proyectos, junto con el manejo de tareas dentro de estos.
  - **Manejo de Solicitudes SCM:** Se incorporara las funcionalidades necesarias para el manejo de solicitudes SCM dentro de los Proyectos y la conversión de estas en tareas .

Al principio de cada incremento se definirá cuales son las funcionalidades necesarias a implementar para completar dicho incremento, el cliente revisara las funcionalidades implementadas y hará las correcciones que estime pertinentes, las que serán consideradas al inicio del nuevo incremento (en las correcciones solo se esperan correcciones pequeñas, no cambios en los requerimientos).

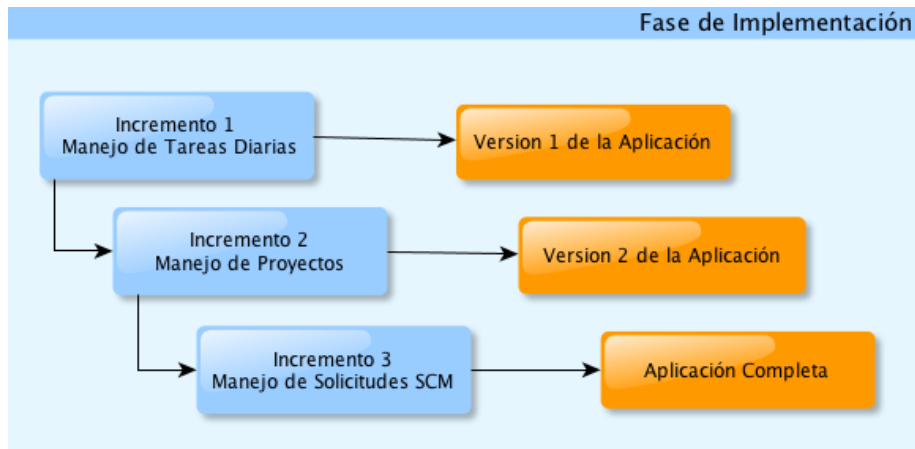


Figura 3: Esquema de la Fase de Implementación

- **Fase de Pruebas:** En esta ultima fase se concentrara netamente el desarrollo de todo tipo de pruebas, esto incluye la ejecución y automatización de estas, junto con la documentación, el análisis de los resultados, y la aplicación de las correcciones correspondientes en el caso de encontrar errores o bugs.

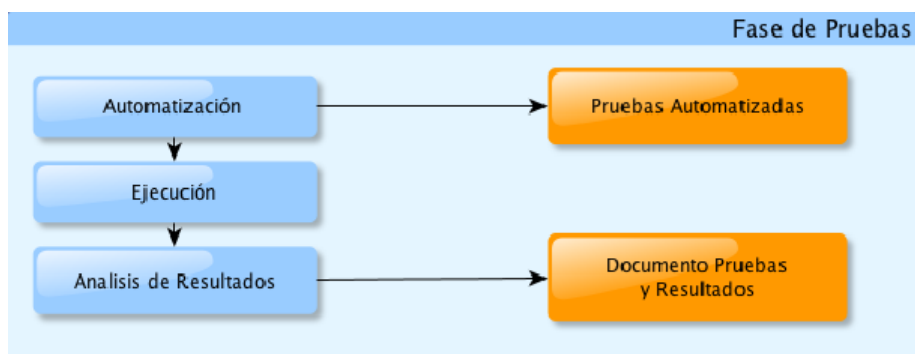


Figura 4: Esquema de la Fase de Pruebas

## 5. Planificación

A continuación se presenta la calendarización de las Actividades que se llevarán a cabo a lo largo del proyecto.

Nombre	F. Inicio	F. Termino
1.0 Propuesta	20-03-2012	05-03-2012
1.1 Introducción	20-03-2012	22-03-2012
1.2 Definición del Problema	23-03-2012	28-03-2012
1.3 Definición Metodología	29-03-2012	30-03-2012
1.4 Identificación de Recursos	01-04-2012	01-04-2012
1.5 Planificación	02-04-2012	05-04-2012
2.0 Fase de Análisis	05-04-2012	01-05-2012
2.1 Marco Conceptual	05-04-2012	20-04-2012
2.1.1 Definir Contexto del Problema	05-04-2012	06-04-2012
2.1.2 Definir Terminología	09-04-2012	10-04-2012
2.1.3 Búsqueda de Soluciones Actuales	11-04-2012	13-04-2012
2.1.4 Analisis de Soluciones Actuales	16-04-2012	20-04-2012
2.2 Captura de Requerimientos	11-04-2012	13-04-2012
2.2.1 Reuniones	11-04-2012	13-04-2012
2.2.2 Revision Documentación Existente	11-04-2012	13-04-2012
2.3 Análisis de Requerimientos	16-04-2012	20-04-2012
2.3.1 Identificación de Requerimientos Funcionales	16-04-2012	20-04-2012
2.3.2 Identificación de Requerimientos No Funcionales	18-04-2012	20-04-2012
2.4 Especificación de Requerimientos	23-04-2012	25-04-2012
2.4.1 Documentar Requerimientos	23-04-2012	25-04-2012
2.4.2 Priorizar los Requerimientos	25-04-2012	25-04-2012
2.5 Diseño de Casos de Uso	26-04-2012	27-04-2012
2.6 Diseño Modelo Conceptual	30-04-2012	01-05-2012
3.0 Fase de Diseño	01-05-2012	01-06-2012
3.1 Diseño Arquitectónico	01-05-2012	02-05-2012
3.2 Modelo de Datos	03-05-2012	07-05-2012
3.3 Diagrama de Clases	08-05-2012	09-05-2012
3.4 Diagrama de Paquetes	10-05-2012	10-05-2012
3.5 Diseño de Interfaces	10-05-2012	13-05-2012
3.6 Diseño Casos de Pruebas	13-05-2012	16-05-2012
3.7 Implementación Prototipo	08-05-2012	18-05-2012
3.8 Iteración 1 en el Diseño	21-05-2012	25-05-2012
3.9 Iteración 2 en el Diseño	28-05-2012	01-06-2012
4.0 Fase de Implementación	04-06-2012	06-07-2012
4.1 Incremento 1 - Tareas Diarias	04-06-2012	13-06-2012
4.2 Incremento 2 - Tareas de Proyectos	14-06-2012	25-06-2012
4.3 Incremento 3 - Procesos SCM	26-06-2012	06-07-2012

Nombre	F. Inicio	F. Termino
5.0 Fase de Pruebas	23-07-2012	01-10-2012
5.1 Automatización de las Pruebas	23-07-2012	20-08-2012
5.2 Ejecución de las Pruebas	21-08-2012	10-09-2012
5.3 Análisis de Resultados	11-09-2012	20-09-2012
5.4 Documento Final de Pruebas	21-09-2012	01-10-2012
6.0 Implantación	04-10-2012	18-10-2012
7.0 Desarrollo Manual de Usuario	19-10-2012	28-10-2012
8.0 Documento Final y Conclusiones	30-10-2012	25-11-2012

## 6. Recursos

### 6.1. Recursos de Software

Los principales recursos de software necesarios para realizar este trabajo son los siguientes:

- **NetBeans 7.1.1:** Es un entorno de desarrollo integrado, está diseñado principalmente para el lenguaje de programación Java - pero puede servir para cualquier otro lenguaje de programación. Es gratuito y sin restricciones de uso.[4]
- **Hibernate 4.1:** Es una herramienta de Mapeo objeto-relacional (ORM) para la plataforma Java.[1]
- **PrimeFaces 3.2:** PrimeFaces es una librería de componentes visuales open source desarrollada y mantenida por Prime Technology, una compañía Turca de IT especializada en consultoría ágil, JSF, Java EE y Outsourcing.[5]
- **GlassFish 3.1.1:** GlassFish es un servidor de aplicaciones de software libre desarrollado por Sun Microsystems, compañía adquirida por Oracle Corporation, que implementa las tecnologías definidas en la plataforma Java EE y permite ejecutar aplicaciones que siguen esta especificación. Es gratuito y de código libre, se distribuye bajo un licenciamiento dual a través de la licencia CDDL y la GNU GPL.[3]
- **SQL Server 2008 R2:** Es un sistema gestor de Bases de Datos relacionales, desarrollado por Microsoft, que además incluye un conjunto de herramientas para su administración. [2]

## 6.2. Recursos de Hardware

Los recursos de hardware necesarios para el desarrollo de la aplicación, estan dados principalmente por NetBeans que es el con mayores requisitos de hardware, estos son:

- **Procesador:** Dual-Core Intel (32 o 64-bit)
- **Memoria RAM:** 1 GB
- **Espacio en disco:** 650 MB

Los recursos de hardware de los que se disponen son:

- **PC Lenovo:** Procesador Intel Core 2 Duo, con 4 GB de RAM y 300 GB de Disco duro.
- **MacBook Pro:** Procesador Intel Core i5, con 4 GB de RAM y 500 GB de Disco duro.
- **Servidor Dell Inc. PowerEdge R710:** Servidor Dell con 2 procesadores i5, 48 GB de RAM y 450GB de Disco duro.

# Bibliografía

- [1] JBoss Community. Hibernate. Último acceso: 6 Abril 2012, <http://www.hibernate.org/>.
- [2] Microsoft. Sql server. Último acceso: 9 Abril 2012, <http://www.microsoft.com/es-es/sqlserver/default.aspx>.
- [3] Oracle Corporation. Glassfish. Último acceso: 4 Abril 2012, <http://glassfish.java.net/es/>.
- [4] Oracle Corporation. Netbeans. Último acceso: 4 Abril 2012, <http://netbeans.org/>.
- [5] Prime Technology. Primefaces. Último acceso: 7 Abril 2012, <http://primefaces.org/>.
- [6] Rectoría de la Universidad de Valparaíso. Decreto universitario n°427. Último acceso: 4 Abril 2012, <http://disico.uv.cl/images/docs/decreto427.pdf>.
- [7] Ian Sommerville. *Ingeniería de Software*. Pearson Addison Wesley, 2005. 7ma Edición.
- [8] Paloma Méndez Soto. Desarrollo de Metodologías de SQA y SCM para la Dirección de Servicios de Información y Computación. Trabajo de título, Universidad de Valparaíso, 2011.
- [9] Universidad de Valparaíso. Dirección de servicios de información y computación. Último acceso: 4 Abril 2012, <http://www.disico.uv.cl>.