



Facultad de Ingeniería
Escuela de Ingeniería Civil en Informática

DESARROLLO DE UNA PLATAFORMA PARA LA SOLICITUD Y GESTIÓN DE REQUERIMIENTOS Y SCM

Por

Alejandro Alvarez Ahumada

Trabajo realizado para optar al Título de
INGENIERO CIVIL EN INFORMÁTICA
Prof. Guía: Carlos Becerra Castro
Prof. Co-Referente: Nombre Profesor Correferente
Junio 2012

Certifico que he leído este documento y que, en mi opinión, es adecuado en ámbito y calidad como trabajo para optar al título de Ingeniero Civil en Informática.

Carlos Becerra Castro Profesor Guía

Certifico que he leído este documento y que, en mi opinión, es adecuado en ámbito y calidad como trabajo para optar al título de Ingeniero Civil en Informática.

Nombre Profesor Correferente Profesor Co-Referente

Certifico que he leído este documento y que, en mi opinión, es adecuado en ámbito y calidad como trabajo para optar al título de Ingeniero Civil en Informática.

Nombre Profesor Informante 1 Profesor Informante

Aprobado por la Escuela de Ingeniería Civil en Informática, UNIVERSIDAD DE VALPARAÍSO.

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.

Agradecimientos

Aquí pueden colocar sus agradecimientos. Si han estudiado con becas es recomendable colocar los agradecimientos a las instituciones que les otorgaron las becas.

Índice general

Índice de tablas

Índice de figuras

Capítulo 1

Diseño

Este capítulo contiene toda la documentación generada correspondiente a la fase de diseño del sistema, en concreto aquí se encuentra:

- Diseño Arquitectónico.
- Diseño Lógico.
- Diseño de Datos.
- Diseño de Interfaces.
- Diseño de Pruebas.

1.1. Diseño Arquitectónico

1.1.1. Restricciones Arquitecturales

Antes de comenzar a definir la arquitectura del sistema es importante, identificar de forma explícita cuales son las restricciones arquitecturales que existen, las cuales se pueden deducir desde los requerimientos impuestos por el cliente, estas restricciones son:

- La aplicación debe ser desarrollada en lenguaje de programación Java.
- La aplicación debe ser desarrollada siguiendo los estándar Java EE, para mantener la compatibilidad con el servidor de aplicaciones del cliente (GlassFish).
- El acceso y guardado de los datos deben ser manejados con persistencia, específicamente haciendo uso del framework Hibernate.
- La autenticación de usuario debe realizarse a través de SSO, he integrase con el sistema actual de login que hace uso de esta tecnología.

1.1.2. Estructura del Sistema

Para satisfacer los requerimientos y restricciones del cliente, se utilizara la arquitectura Cliente-Servidor separada en Múltiples-Capas como lo plantea el estándar definido por Java EE 6[?].

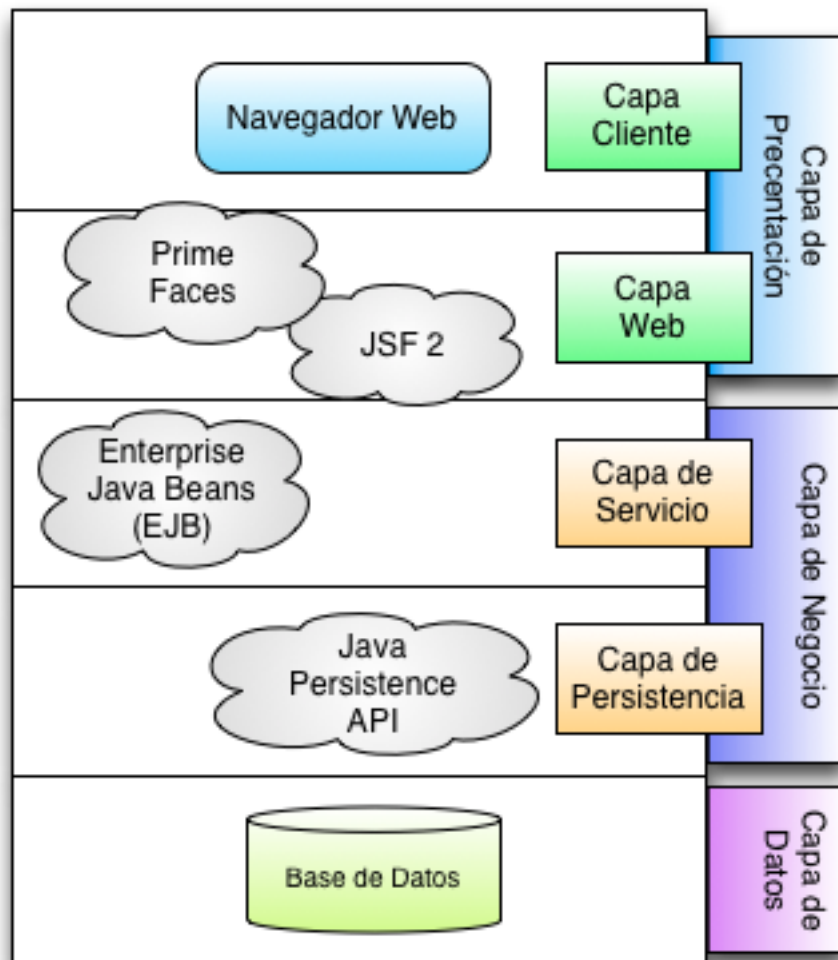


Figura 1.1: Diagrama de Arquitectura

A continuación se describe el propósito de cada una de las capas definidas en el diagrama:

1. **Capa de Presentación:** Es la capa destinada a la mostrar la parte gráfica de la aplicación, que se descompone en 2 sub-capas:
 - **Capa del Cliente:** Es la capa destinada a interpretar y mostrar la interfaz gráfica de usuario del lado del cliente. Esta capa se encuentra compuesta por:
 - Páginas web dinámicas que contienen varios tipos de lenguajes de marcas (HTML, XML u otros), las cuales son generadas por la capa web.
 - Un Navegador Web que interpreta las paginas enviadas por el servidor.
 - **Capa Web:** Representa los componentes web creados con la tecnología Java Server Faces, los cuales se ejecutan dentro del contenedor web del servidor de aplicaciones Java EE. Estos componentes generan paginas web dinámicas, las cuales son enviadas en respuestas a las peticiones HTTP del clientes e interpretadas por el navegador web del mismo.
2. **Capa de Negocio:** Es la capa intermedia, la cual comunica la capa de datos con la capa web, contiene toda la lógica particular del dominio del negocio y también envía y recupera información desde la capa de datos. Esta capa a su vez se compone por dos sub-capas, las cuales se almacenan y trabajan conjuntamente dentro del contenedor EJB del servidor de aplicaciones Java EE. Las dos sub-capas que componen esta capa son:
 - **Capa de Servicio:** Esta capa contiene toda la lógica de negocio y procesamiento de datos, y se comunica con la capa de persistencia para acceder a los datos.
 - **Capa de Persistencia:** Esta capa contiene toda la lógica para el manejo y uso de API de persistencia de Java (implementada con Hibernate), se encarga de mapear las tablas de la base de datos a entidades Java, maneja transacciones, inserta y recupera de datos desde esta.
3. **Capa de Datos:** Es un mecanismo de almacenamiento persistente, donde residen la información relevante para el sistema y es la encargada de acceder los mismos. En este caso esta formada por un gestor de base de datos relacional. Esta recibe solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

1.1.3. Estilo de Descomposición Modular

En esta sección se describe como se descompone el sistema en diferentes modulos, esto se realizara utilizando el enfoque Orientado a Objetos (OO), para este caso este enfoque es el mas adecuado, ya que el sistema debe ser implementado haciendo uso de un

lenguaje OO (particularmente java), así se mantendrá la correcta correspondencia entre los modelos y la implementación.

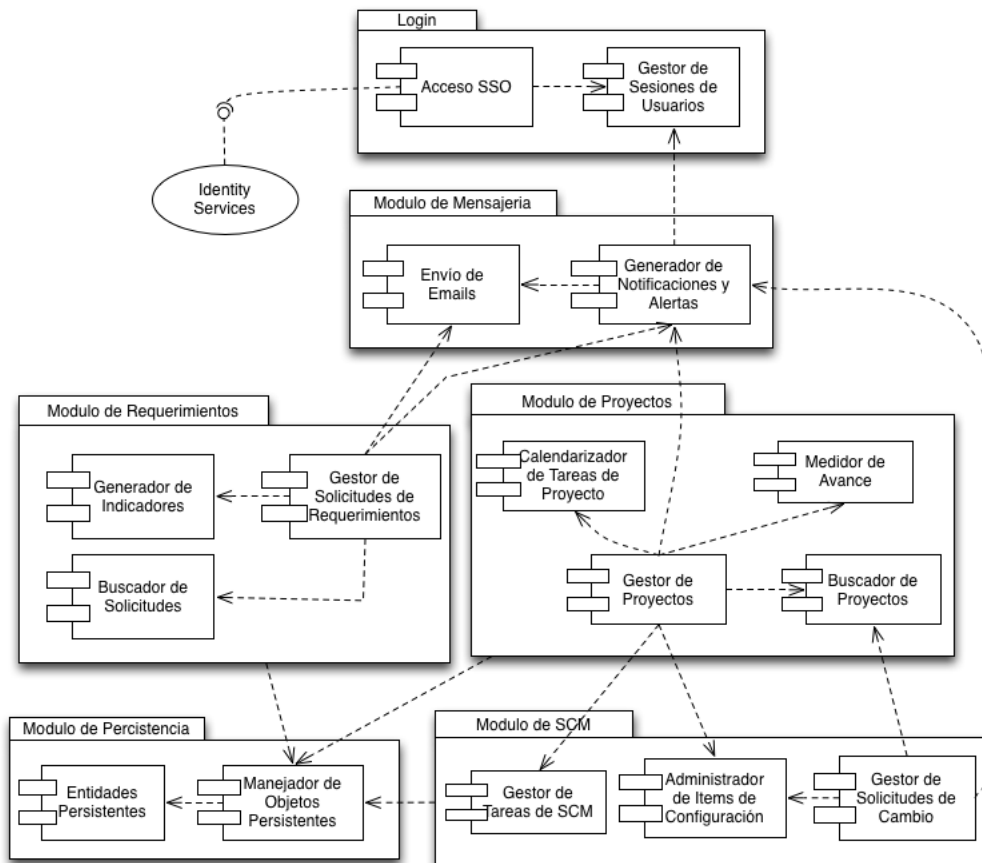


Figura 1.2: Diagrama de Descomposición Modular del Sistema

A continuación se describen las responsabilidades de cada modulo:

- **Identity Services:** Servicio web proporcionado por la Universidad de Valparaíso, que proporciona servicios de acceso a la información de todas las cuentas de los funcionarios de la Universidad, que se encuentran almacenadas en LDAP.
- **Login:**
 1. **Acceso SSO:** Componente que se conecta con el servicio *Identity Services* para recuperar la información de los usuarios, también se encarga de administrar las cookies generadas por el sistema de acceso de SSO de la Universidad.

2. **Gestor de Sesiones de Usuarios:** Mantiene el registro de todos los usuarios conectados a la aplicación, proporciona acceso a la información de los perfiles de usuarios y da la autorización para acceder a diferentes funcionalidades a los usuarios según sus roles.

■ **Modulo de Mensajeria:**

1. **Generador de Notificaciones y Alertas:** Se encarga de detectar los diferentes eventos ocurridos dentro del sistema que producen algún tipo de notificación o alerta y a la vez se preocupa de enviar dichas notificaciones y alertas a los usuarios que corresponde.
2. **Envío de Emails:** Proporciona funciones para la creación y envío de emails de manera manual o automática.

■ **Modulo de Requerimientos:**

1. **Gestor de Solicitudes de Requerimientos:** Permite administrar la creación, envío y respuesta de las solicitudes de requerimientos.
2. **Generador de Indicadores:** Este componente calcula los valores de los diferentes indicadores, y las medidas necesarias para generar los diferentes gráficos, que son necesarios en los resúmenes que ven los Jefes de Área y Departamento.
3. **Buscador de Solicitudes de Requerimientos:** Permite buscar solicitudes de requerimientos, según los diferentes criterios de búsqueda existentes.

■ **Modulo de Proyectos:**

1. **Gestor de Proyectos:** Proporciona acceso a las funciones de creación, eliminación y modificación de los proyectos, y la información de estos como sus tareas, responsables, avance.
2. **Medidor de Avance:** Este componente es el encargado de calcula las medidas de avance de cualquier proyecto.
3. **Buscador de Proyectos:** Permite la búsqueda de proyectos bajo diferentes criterios.
4. **Calendarizador de Tareas de Proyectos:** Este permite agregar, eliminar y editar tareas en un proyecto.

■ **Modulo de SCM:**

1. **Gestor de Tareas SCM:** Permite editar las información de las tareas de SCM y definir responsables para estas.

2. **Administrador de Items de Configuración:** Proporciona funciones para agregar, eliminar, editar y buscar Items de configuración de un proyecto.
3. **Gestor de Solicitudes de Cambio:** Permite crear, eliminar, responder solicitudes de cambio y gestiona el ciclo de vida de estas.

■ **Modulo de Persistencia:**

1. **Manejador de Objetos Persistentes:** Es una fachada a la que accede cualquiera de los componente de la capa de negocio, que proporciona acceso a las entidades persistentes utilizadas en el sistema.
2. **Entidades Persistentes:** Contiene todas las entidades persistentes del sistema y las cuales están mapeadas directamente a la base de datos a traves de hibernate, solo pueden ser accedidas a través del *Manejador de Objetos Persistentes*.

1.1.4. Modelo de Control

Para que el sistema trabaje como tal, se deben controlar los diferentes módulos o subsistemas de este de manera tal que sus servicios se entregue en el lugar correcto en el momento preciso. Para este caso el modelo de control seleccionado es el control Basado en Eventos de Broadcast, esta decisión se justifica principalmente en el echo que el sistema debe generar notificaciones o alertas en las pantallas de los diferentes usuarios en respuesta a algún evento de tiempo o generado por otro usuario, de esta manera mantener a los usuarios comunicados y con la información actualizada.

1.2. Diseño Lógico

Como se menciona anteriormente, se utilizara un paradigma Orientado a Objetos por lo que en esta sección, se presenta el diagrama de clases correspondiente al sistema a desarrollar.

1.2.1. Diseño de Clases

A continuación se presenta el diagrama de clases del sistema.

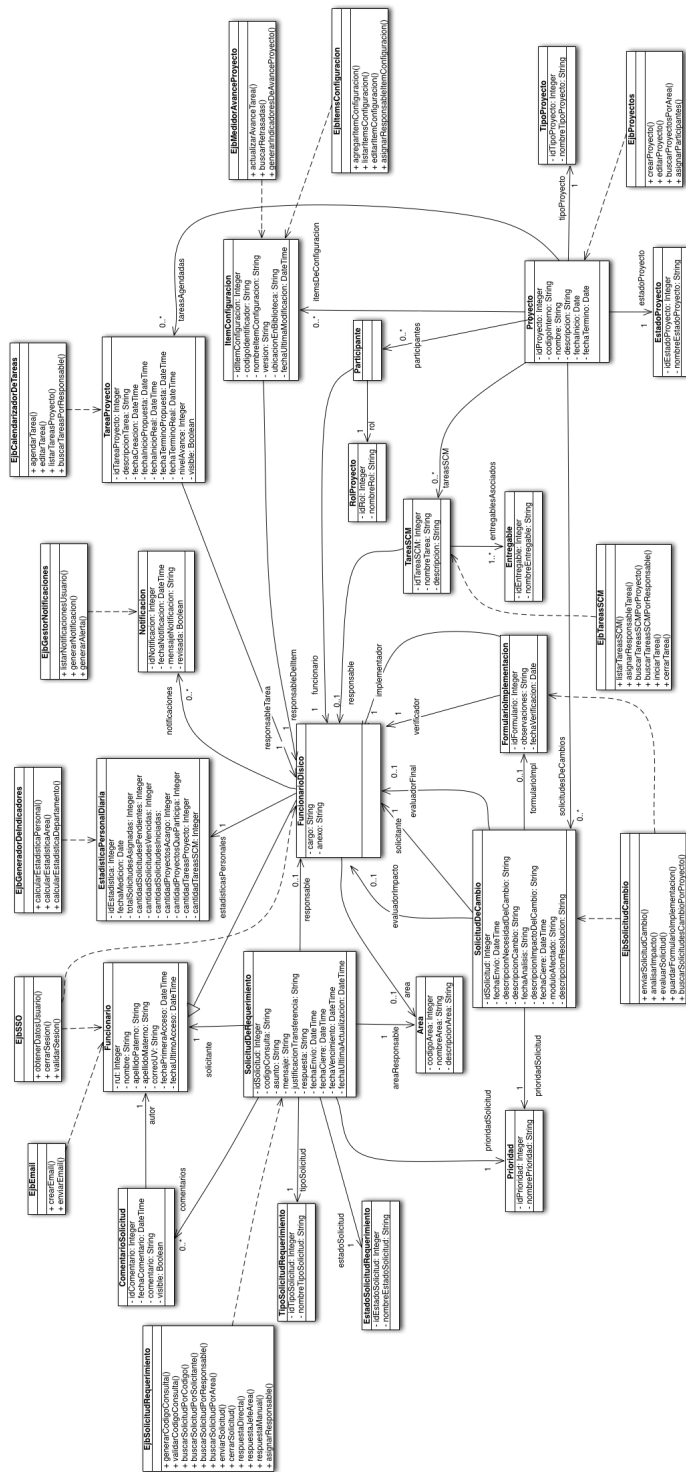


Figura 1.3: Diagrama de Clases

Para una mayor comprensión del diagrama anterior, es que se presenta una breve explicación de cada clase y los atributos y/o métodos que la componen.

1. Clase Funcionario:

Atributos de la Clase:

- rut:
- nombre:
- apellidoPaterno:
- apellidoMaterno:
- correoUV:
- fechaPrimerAcceso:
- fechaUltimoAcceso:

2. Clase FuncionarioDisico:

Atributos de la Clase:

- cargo:
- anexo:

3. Clase Area:

Atributos de la Clase:

- codigoArea:
- nombreArea:
- descripcionArea:

4. Clase SolicitudDeRequerimiento:

Atributos de la Clase:

- idSolicitud:
- codigoConsulta:
- asunto:
- mensaje:

- **justificacionTransferencia:**
- **respuesta:**
- **fechaEnvio:**
- **fechaCierre:**
- **fechaVencimiento:**
- **fechaUltimaActualizacion:**

5. Clase Notificacion:

Atributos de la Clase:

- **idNotificacion:**
- **fechaNotificacion:**
- **mensajeNotificacion:**
- **revisada:**

6. Clase EstadisticaPersonalDiaria: Atributos de la Clase:

- **idEstadistica:**
- **fechaMedicion:**
- **totalSolicitudesAsegnadas:**
- **cantidadSolicitudesPendientes:**
- **cantidadSolicitudesVencidas:**
- **cantidadSolicitudesIniciadas:**
- **cantidadProyectosAcargo:**
- **cantidadProyectosQueParticipa:**
- **cantidadTareasProyecto:**
- **cantidadTareasSCM:**

7. Clase ComentarioSolicitud: Atributos de la Clase:

- **idComentario:**
- **fechaComentario:**
- **comentario:**
- **visible:**

8. Clase TipoSolicitudRequerimiento: Atributos de la Clase:

- idTipoSolicitud:
- nombreTipoSolicitud:

9. Clase EstadoSolicitudRequerimiento: Atributos de la Clase:

- idEstadoSolicitud:
- nombreEstadoSolicitud:

10. Clase Prioridad: Atributos de la Clase:

- idPrioridad:
- nombrePrioridad:

11. Clase SolicitudDeCambio: Atributos de la Clase:

- idSolicitud:
- fechaEnvio:
- descripcionNecesidadDelCambio:
- descripcionCambio:
- fechaAnalisis:
- descripcionImpactoDelCambio:
- fechaCierre:
- moduloAfectado:
- descripcionResolucion:

12. Clase FormularioDeImplementacion: Atributos de la Clase:

- idFormulario:
- observaciones:
- fechaVerificacion:

13. Clase TareaSCM: Atributos de la Clase:

- idTareaSCM:
- nombreTarea:
- descripcion:

14. Clase Entregable: Atributos de la Clase:

- **idEntregable:**
- **nombreEntregable:**

15. Clase Proyecto: Atributos de la Clase:

- **idProyecto:**
- **codigoInterno:**
- **nombre:**
- **descripcion:**
- **fechaInicio:**
- **fechaTermino:**

16. Clase RolProyecto: Atributos de la Clase:

- **idRol:**
- **nombreRol:**

17. Clase Participante: Atributos de la Clase:

- **rol:**
- **funcionario:**

18. Clase EstadoProyecto: Atributos de la Clase:

- **idEstadoProyecto:**
- **nombreEstadoProyecto:**

19. Clase TipoProyecto: Atributos de la Clase:

- **idTipoProyecto:**
- **nombreTipoProyecto:**

20. Clase ItemConfiguracion: Atributos de la Clase:

- **idItemConfiguracion:**
- **codigoIdentificador:**
- **nombreItemConfiguracion:**
- **version:**

- **ubicacionEnBiblioteca:**
- **fechaUltimaModificacion:**

21. Clase TareaProyecto: Atributos de la Clase:

- **idTareaProyecto:**
- **descripcionTarea:**
- **fechaCreacion:**
- **fechaInicioPropuesta:**
- **fechaInicioReal:**
- **fechaTerminoPropuesta:**
- **fechaTerminoReal:**
- **nivelAvance:**
- **visible:**

22. Clase EjbSolicitudRequerimeinto: Métodos de la Clase:

- **generarCodigoConsulta:**
- **validarCodigoConsulta:**
- **buscarSolicitudPorCodigo:**
- **buscarSolicitudPorSolicitante:**
- **buscarSolicitudPorResponsable:**
- **buscarSolicitudPorArea:**
- **enviarSolicitud:**
- **cerrarSolicitud:**
- **respuestaDirecta:**
- **respuestaJefeArea:**
- **respuestaManual:**
- **asignarResponsable:**

23. Clase EjbEmail: Métodos de la Clase:

- **crearEmail:**
- **enviarEmail:**

24. Clase EjbSSO: Métodos de la Clase:

- obtenerDatosUsuario:
- cerrarSesion:
- validarSesion:

25. Clase EjbGeneradorDeIndicadores: Métodos de la Clase:

- :

26. Clase EjbGestorNotificaciones: Métodos de la Clase:

- calcularEstadisticaPersonal:
- calcularEstadisticaArea:
- calcularEstadisticaDepartamento:

27. Clase EjbCalendarizadorDeTareas: Métodos de la Clase:

- agendarTarea:
- editarTarea:
- listarTareasProyecto:
- buscarTareasPorResponsable:

28. Clase EjbMedidorAvanceProyecto: Métodos de la Clase:

- actualizarAvanceTarea:
- buscarTareasRetrasadas:
- generarIndicadoresDeAvanceProyecto:

29. Clase EjbItemsConfiguracion: Métodos de la Clase:

- :

30. Clase EjbProyectos: Métodos de la Clase:

- :

31. Clase EjbTareasSCM: Métodos de la Clase:

- :

32. Clase EjbSolicitudDeCambio: Métodos de la Clase:

- :

1.3. Diseño de Datos

Dado que los datos serán almacenados dentro de una base de datos relacional, se presenta en esta sección el diagrama de entidad relación que se implementará.

1.3.1. Modelo Entidad Relación

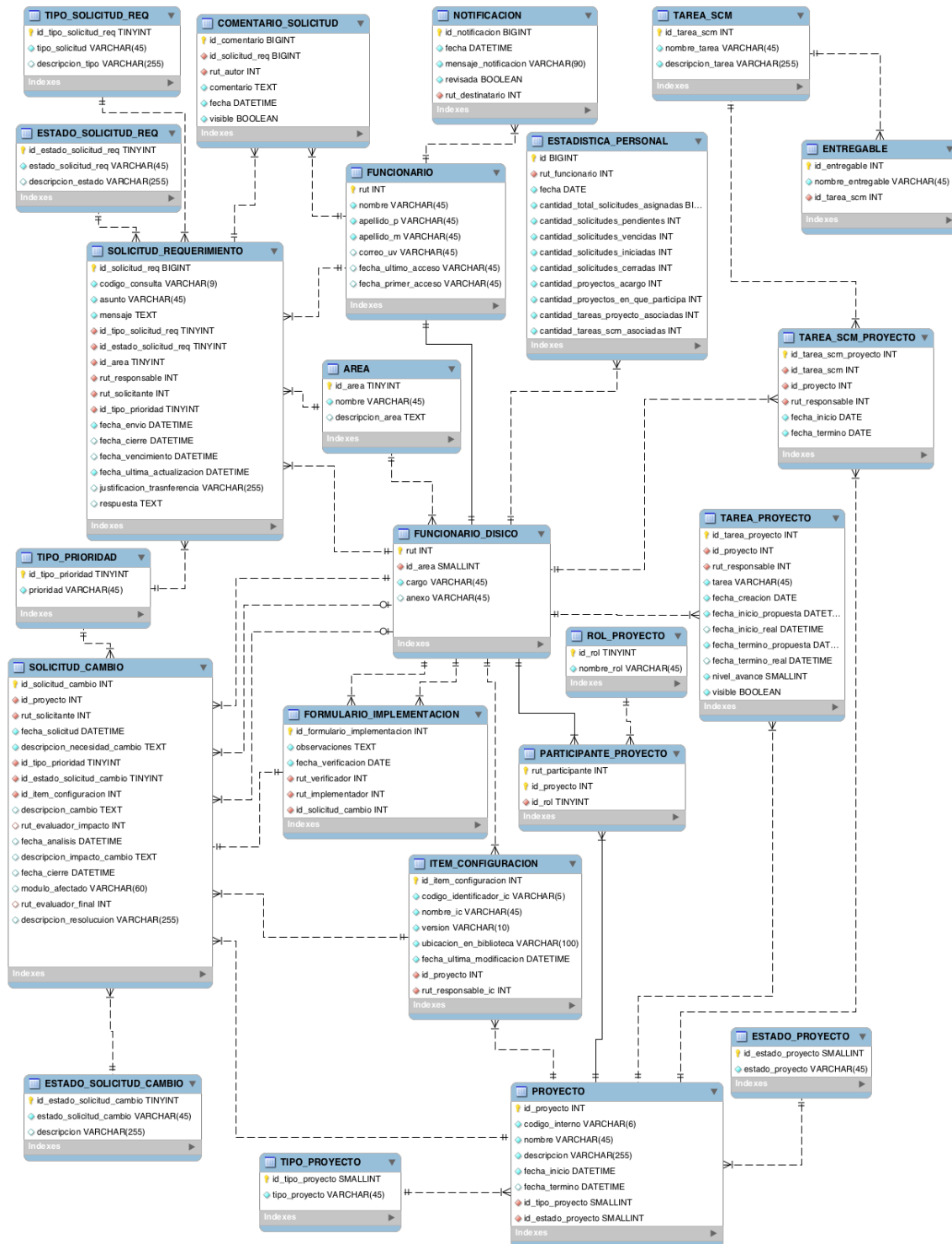


Figura 1.4: Modelo Físico de la Base de Datos

1.3.2. Diccionario de Datos

Para dar un mayor nivel de detalle a continuación se presenta el diccionario de datos de la base de datos, en la cual se describe cada tabla con sus atributos.

Bibliografía

- [1] Universidad de Valparaíso, “Dirección de servicios de información y computación.” Último acceso: 4 Abril 2012, <http://www.disico.uv.cl>.
- [2] Rectoría de la Universidad de Valparaíso, “Decreto universitario nº427.” Último acceso: 4 Abril 2012, <http://disico.uv.cl/images/docs/decreto427.pdf>.
- [3] P. Méndez, “Desarrollo de Metodologías de SQA y SCM para la Dirección de Servicios de Información y Computación,” Trabajo de Título, Universidad de Valparaíso, 2011.
- [4] The Institute of Electrical and Eletronics Engineers, “IEEE Standard Glossary of Software Engineering Terminology.” IEEE Standard, September 1990.
- [5] Coasin Group, “Mesa de Ayuda TI.” Último acceso: 25 Abril 2012, <http://www.coasin.cl/index.php?id=246>.
- [6] Osiatis, “ITIL V3 Gestión de Servicios TI.” Último acceso: 1 Mayo 2012, <http://itilv3.osiatis.es/>.
- [7] Hesk, “Hesk Software Help.” Último acceso: 25 Abril 2012, <http://www.hesk.com>.
- [8] osTicket, “osTicket Support Ticket System.” Último acceso: 25 Abril 2012, <http://osticket.com/>.
- [9] OTRS Group, “OTRS Help Desk.” Último acceso: 25 Abril 2012, <http://www.otrs.com/es>.
- [10] SysAid Technologies, “SysAid IT.” Último acceso: 30 Abril 2012, <http://www.sysaid.com>.
- [11] Sun View, “ChangeGear - Service Desk.” Último acceso: 25 Abril 2012, <http://www.sunviewsoftware.com/products/servicedesk.aspx>.

- [12] International Organization for Standardization, *ISO 10007: 2003 – Quality management systems – Guidelines for configuration management*, 2003.
- [13] IEEE, *IEEE Standard for Software Configuration Management Plans*. The Institute of Electrical and Electronics Engineers, Inc., 2005.
- [14] IEEE, *IEEE Standard for Configuration Management in Systems and Software Engineering*. The Institute of Electrical and Electronics Engineers, Inc., 2012.
- [15] Oracle, “The java ee 6 tutorial.” Último acceso: 1 Junio 2012, <http://docs.oracle.com/javasee/6/tutorial/doc/>.