



## Práctica de la asignatura ISG3

Autores: **Macarena Rodríguez Rodríguez**  
**Carlos Pompa Domínguez**



01	<i>¿Qué es SVL?</i>	3
02	<i>Especificaciones Técnicas</i>	7
03	<i>Herramientas Software Utilizadas</i>	7
04	<i>Subsistemas Identificados</i>	12
05	<i>Prototipo Web</i>	18
06	<i>Casos de Uso implementados</i>	12
	<i>1. Registro de Voluntarios</i>	12
	<i>Gestión de Alertas</i>	12
07	<i>Modelo Conceptual</i>	20
08	<i>Diseño de Clases</i>	20
09	<i>DAO</i>	20

## **C 1** ***¿Qué es SVL?***

## ¿Qué es SVL?

SVL (Servicio de Voluntariado Local) es una propuesta de portal web cuyo principal objetivo es facilitar el acceso a los servicios de tipo asistencial de aquellas personas que lo necesitan.

La idea principal es que dicho servicio sea de ámbito local, es decir, que dentro de una misma localidad pueda confluir la oferta y la demanda de dichos servicios.

Se trata básicamente de que todas las personas que quieran dedicar su tiempo y sus conocimientos a servir a otras personas de forma voluntaria, se registren en la web como voluntario, para que aquellos demandantes registrados a su vez, puedan encontrarlos.

Es decir, el fin último de la web es confluir oferta y demanda de servicios de voluntariado.

## **C 2** ***Especificaciones Técnicas***

## Especificaciones Técnicas

Las especificaciones técnicas del desarrollo del aplicativo son las siguientes:

- ❖ Servidor de Aplicaciones: **Tomcat 5.5.**
- ❖ Versión JAVA: **JDK 1.6.0**
- ❖ Base de datos: **MySQL Server 5.5**
- ❖ Interfaz de Usuario: **Páginas JSP y HTML con Javascript**
- ❖ Patrón del Controlador: **Front-Controller**
- ❖ Arquitectura en 3 capas **MVC.**



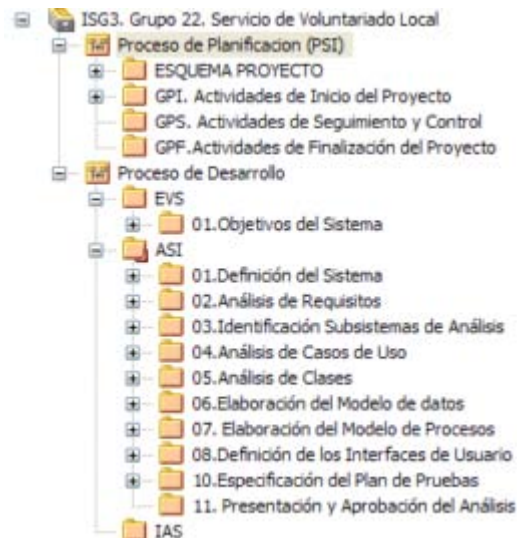
## ***Herramientas Software Utilizadas***

## Herramientas Software Utilizadas

Las herramientas Software utilizadas en el desarrollo del aplicativo son las siguientes:

❖ **IDE: Eclipse Helios Services Release 2**

❖ **Enterprise Architect:** Para el análisis y diseño del proyecto. Esta herramienta permite generar toda la documentación asociada a estas 2 fases.



❖ **Macromedia Dreamweaver:** Para el desarrollo del Prototipo Web.

❖ **Adobe Photoshop:** Para el diseño de la web y el tratamiento de imágenes asociadas.



## **C 4** ***Subsistemas Identificados***

## PROTOTIPO WEB

Los principales subsistemas identificados en el aplicativo son los siguientes:



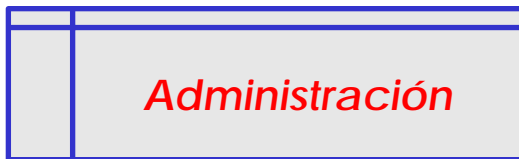
*Este subsistema permite la gestión de la información de los Voluntarios. Además se incluye un formulario de alta para que las personas que quieran ofrecer sus servicios puedan hacerlo de una forma rápida y sencilla.*



*Este subsistema permite la gestión de la información de los Demandantes. Se incluye un formulario de alta para que las personas que requieren de estos servicios puedan solicitarlo de una forma rápida y sencilla.*



*Este subsistema permite la gestión de las alertas sobre los diferentes servicios que se ofrecen. De este modo, se podrá avisar a los demandantes (vía sms o correo electrónico) de los servicios disponibles y que se adecúan a sus necesidades.*



*Este subsistema permite la gestión interna del aplicativo. Incluye todo lo referente a la gestión de perfiles y usuarios del sistema, así como la parametrización de las tablas maestras.*

## **C** **5** *Prototipo Web*

## PROTOTIPO WEB

El prototipo web es una versión “cartón piedra” y reducida de la aplicación hecha con **HTML** y **Javascript** en base a las especificaciones del aplicativo.

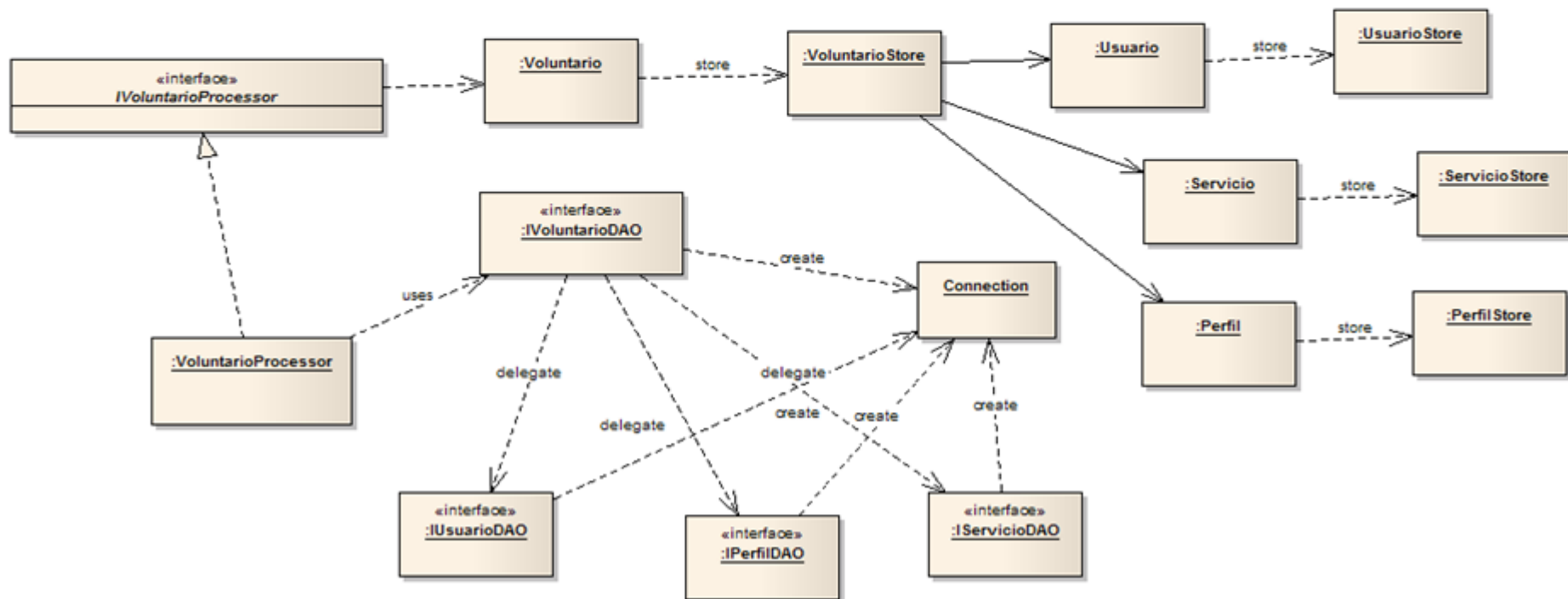


## **C 6**

### ***Casos de Uso implementados***

## Caso de Uso 1 – REGISTRO DE VOLUNTARIOS

<i>Business Object</i>	<i>Transfer Objects</i>	<i>Data Access Objects</i>	<i>Data Source(s)</i>
IVoluntarioProcessor/ Ivoluntario registrarVoluntario(Usuario): void	Voluntario	IVoluntarioDAO.registrarVoluntario(Usuario usuario) IUsuarioDAO.insertarUsuario(Usuario usuario) IPerfilDAO.insertarPerfil(int usua_id) IServicioDAO.insertarServicio(Servicio servicio);	Voluntario Usuario Perfil Servicio

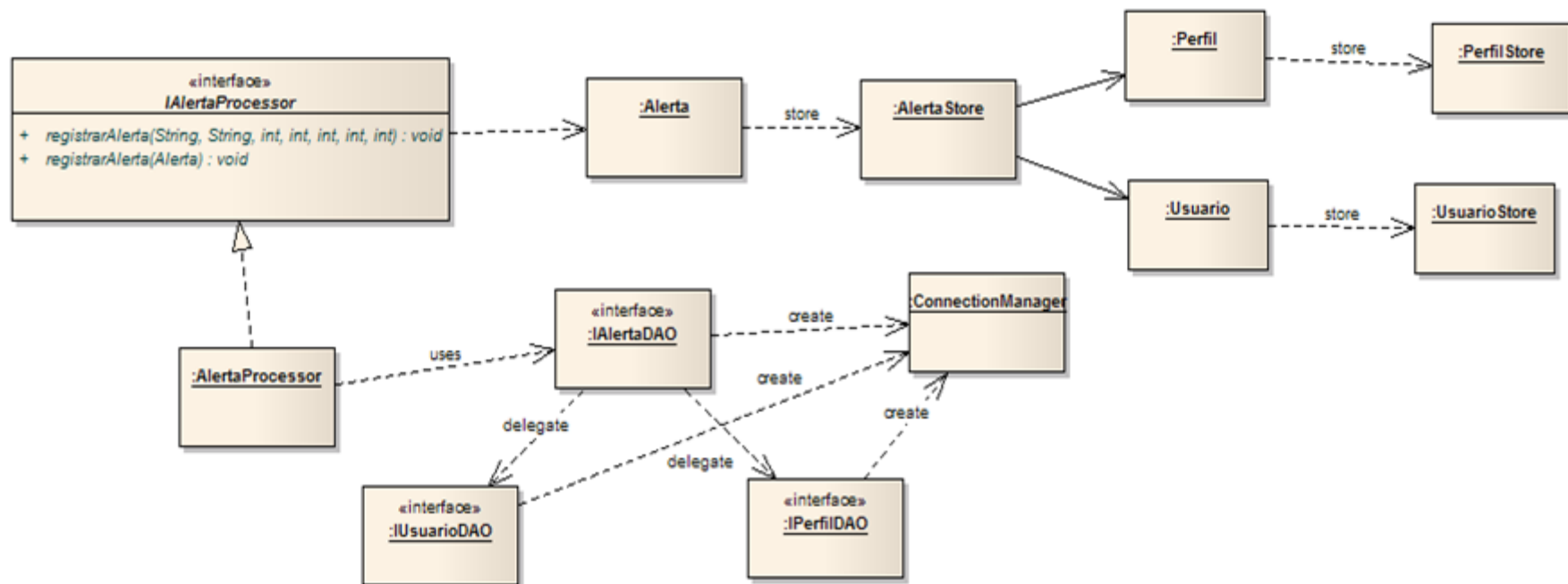


## Caso de Uso 1 – REGISTRO DE VOLUNTARIOS

<b>UC-0001</b>	<b>Registro de Voluntarios</b>	
<b>Versión</b>	1.0 (30-08-2011)	
<b>Autores</b>	Carlos Pompa Domínguez (Grupo 22)	
<b>Fuentes</b>		
<b>Objetivos asociados</b>		
<b>Requisitos asociados</b>	Registro de Voluntarios	
<b>Descripción</b>	Los usuarios anónimos se registran como voluntarios. Para ello tienen que dar de alta sus datos personales y después los servicios en los que se van a dar de alta como voluntario.	
<b>Precondición</b>	El usuario no está registrado previamente.	
<b>Secuencia normal</b>	<b>Paso</b>	<b>Acción</b>
	P1	El usuario solicita darse de alta
	P2	El sistema solicita los campos obligatorios para el registro
	P3	El usuario rellena los datos y los envía
	P4	El sistema registra los datos e informa de que los datos han sido registrados.
	P5	El sistema muestra la lista de servicios opcionales a registrar.
	P6	El usuario informa los servicios en los que se quiere dar de alta.
	P7	El sistema registra los datos e informa de que los datos han sido registrados.
<b>Post-condición</b>	<i>El usuario es registrado en el sistema como voluntario y han sido de alta los servicios.</i>	
<b>Excepciones</b>	<b>Paso</b>	<b>Acción</b>
	P1	Si el usuario no rellena los servicios sus datos si quedan registrados. Pudiendo editar luego los servicios.

## Caso de Uso 2 – GESTIÓN DE ALERTAS

<u>Business Object</u>	<u>Transfer Objects</u>	<u>Data Access Objects</u>	<u>Data Source(s)</u>
<b>IALertaProcessor/ IAlerta</b> <b>registrarAlerta(String</b> <b>aler_nombrecontacto,String</b> <b>aler_emailcontacto,int prov_id, int</b> <b>muni_id, int serv_id, int tibu_id,</b> <b>int tise_id);</b>	Alerta	<b>IVoluntarioDAO.addAlerta(String</b> <b>usua_usuario, String usua_email);</b> <b>IUsuarioDAO.insertarUsuario(Usuario</b> <b>usuario)</b> <b>IPerfilDAO.insertarPerfil(int usua_id)</b>	Alerta Usuario Perfil





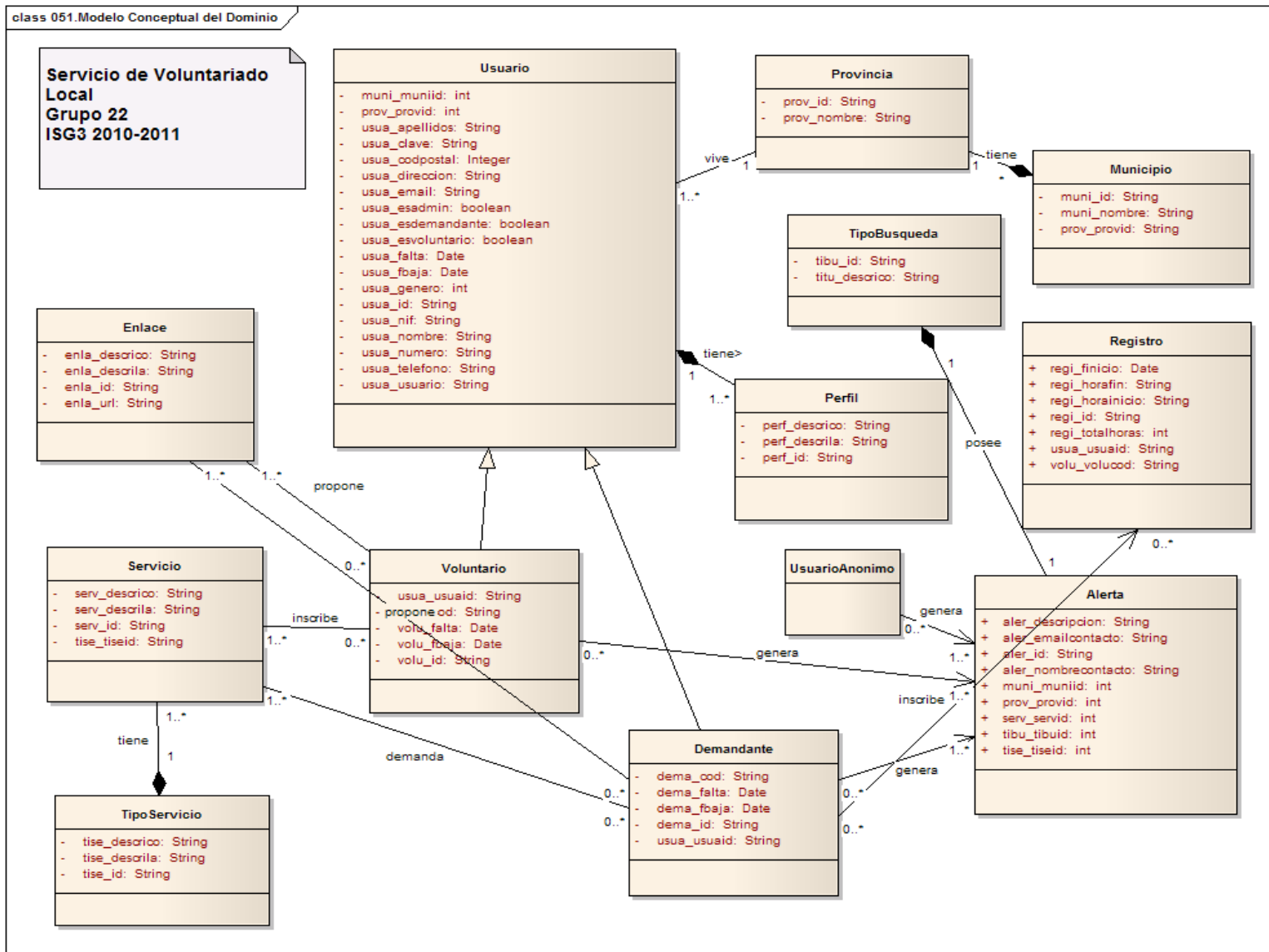
## Caso de Uso 2 – GESTIÓN DE ALERTAS

<b>UC-0002</b>	<b>Gestión de Alertas</b>	
<b>Versión</b>	1.0 (30-08-2011)	
<b>Autores</b>	Macarena Rodríguez Rodríguez (Grupo 22)	
<b>Fuentes</b>		
<b>Objetivos asociados</b>		
<b>Requisitos asociados</b>	Registro de Voluntarios y Servicios	
<b>Descripción</b>	Los usuarios anónimos solicitan dar de alta una alerta en el sistema. Para ello tienen que dar de alta un usuario y un email. Al dar de alta la alerta se registra con los datos mínimos y se da de alta con el perfil no registrado. No se dan de alta servicios.	
<b>Precondición</b>	El usuario no está registrado previamente.	
<b>Secuencia normal</b>	<b>Paso</b>	<b>Acción</b>
	P1	El usuario solicita dar de alta una alerta
	P2	El sistema solicita los campos obligatorios para la creación de una alerta.
	P3	El usuario rellena los datos y los envía
	P4	El sistema registra los datos e informa de que los datos han sido registrados.
	P5	El sistema registra como clave el nombre del usuario.
	P6	El sistema registra los datos e informa de que los datos han sido registrados.
<b>Post-condición</b>	<i>La alerta es dada de alta en el sistema.</i>	
<b>Mejoras</b>	<b>Paso</b>	<b>Acción</b>
	P1	Mandar un correo al usuario con la alerta dada de alta.



## ***Modelo Conceptual***

## Modelo Conceptual

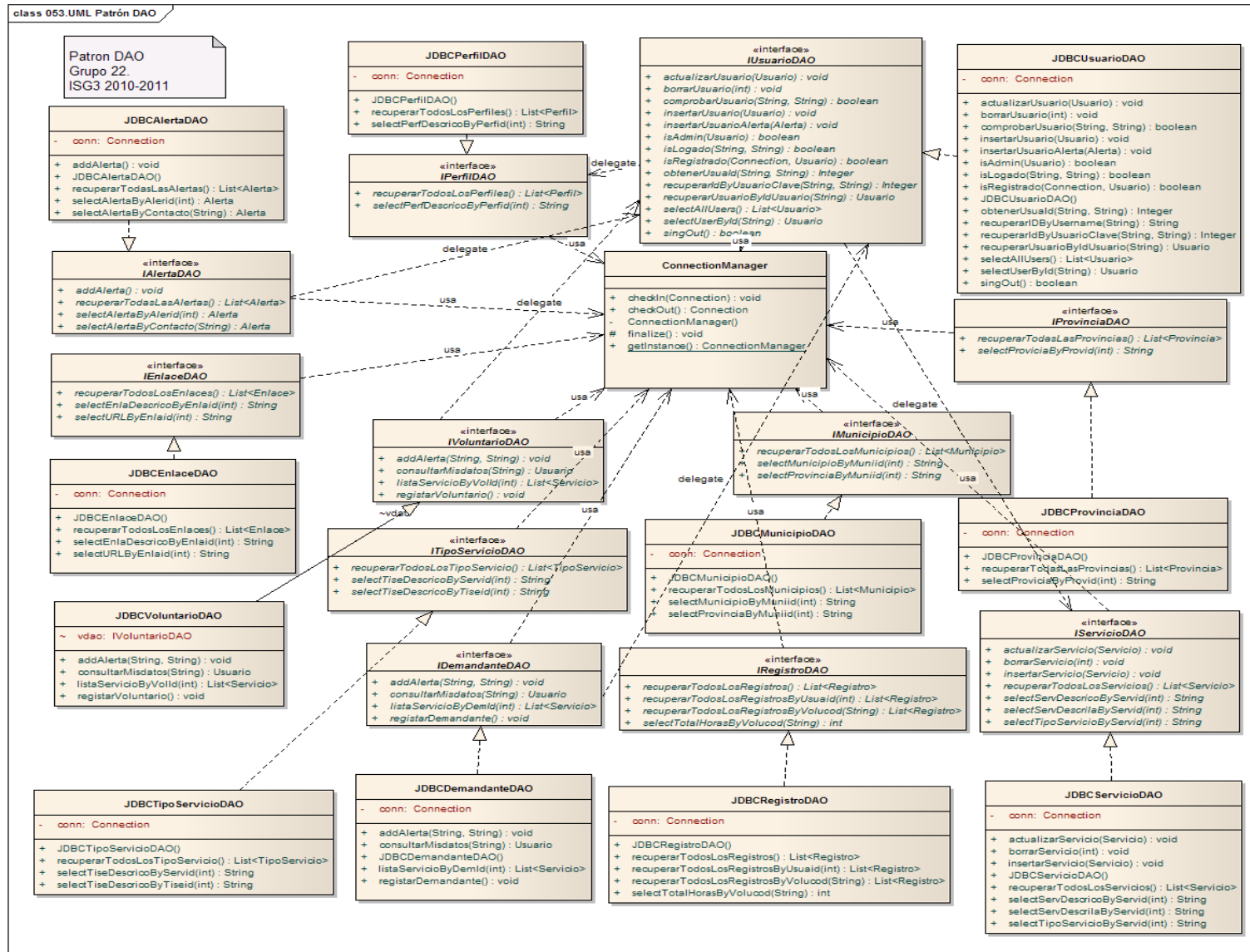


## **C 8** ***Diseño de Clases***

[illegible]

## **C 9** ***DAO***

## DAO







**Fin de la Presentación**

Autores: *Macarena Rodríguez Rodríguez*  
*Carlos Pompa Domínguez*