



**UNIVERSIDAD TÉCNICA DEL NORTE**  
**FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS**  
**CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES**

**TEMA**

“Sistema de Gestión y Seguimiento de Proyectos de Software”

**Autor:** Edison Patricio Lomas Almeida

**Director:** Xavier Mauricio Rea Peñafiel

**Ibarra – Ecuador**

**2012**

## CERTIFICACIÓN

Certifico que la Tesis “**SISTEMA DE GESTIÓN Y SEGUIMIENTO DE PROYECTOS DE SOFTWARE**” ha sido realizada en su totalidad por el señor: Edison Patricio Lomas Almeida portador de la cédula de identidad número: 1002803896.



---

Xavier Mauricio Rea Peñafiel  
Director de la Tesis

# CERTIFICACIÓN

Quito, 17 de febrero del 2012

**Señores**

**UNIVERSIDAD TÉCNICA DEL NORTE**

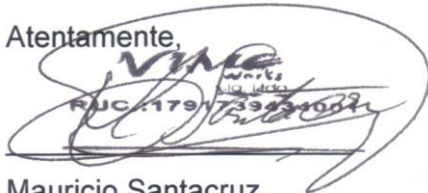
**Presente**

De mis consideraciones.-

Siendo auspiciantes del proyecto de tesis del Egresado EDISON PATRICIO LOMAS ALMEIDA con CI: 1002803896 quien desarrolló su trabajo con el tema “SISTEMA DE GESTIÓN Y SEGUIMIENTO DE PROYECTOS DE SOFTWARE”, me es grato informar que se han superado con satisfacción las pruebas técnicas y la revisión de cumplimiento de los requerimientos funcionales, por lo que se recibe el proyecto como culminado y realizado por parte del egresado EDISON PATRICIO LOMAS ALMEIDA. Una vez que hemos recibido la capacitación y documentación respectiva, nos comprometemos a continuar utilizando el mencionado aplicativo en beneficio de nuestra empresa.

El egresado EDISON PATRICIO LOMAS ALMEIDA puede hacer uso de este documento para los fines pertinentes en la Universidad Técnica del Norte.

Atentamente,



Mauricio Santacruz

**GERENTE GENERAL**

**VimeWorks Cía. Ltda.**



**UNIVERSIDAD TÉCNICA DEL NORTE**  
**CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO**  
**DE INVESTIGACIÓN**  
**A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL**  
**NORTE**

Yo, EDISON PATRICIO LOMAS ALMEIDA, con cedula de identidad Nro. 1002803896, manifiesto mi voluntad de ceder a la Universidad Técnica del Norte los derechos patrimoniales consagrados en la ley de propiedad intelectual del Ecuador, articulo 4, 5 y 6, en calidad de autor del trabajo de grado denominado: **“SISTEMA DE GESTIÓN Y SEGUIMIENTO DE PROYECTOS DE SOFTWARE”**, que ha sido desarrollada para optar por el título de Ingeniería en Sistemas Computacionales, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En mi condición de autor me reservo los derechos morales de la obra antes mencionada, aclarando que el trabajo aquí descrito es de mi autoría y que no ha sido previamente presentado para ningún grado o calificación profesional.

En concordancia suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la biblioteca de la Universidad Técnica del Norte

Firma

Nombre: EDISON PATRICIO LOMAS ALMEIDA

Cédula: 1002803896

Ibarra a los 24 días del mes de mayo del 2012



## UNIVERSIDAD TÉCNICA DEL NORTE

### BIBLIOTECA UNIVERSITARIA

## AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

### 1. IDENTIFICACIÓN DE LA OBRA

La UNIVERSIDAD TÉCNICA DEL NORTE dentro del proyecto Repositorio Digital Institucional determina la necesidad de disponer los textos completos de forma digital con la finalidad de apoyar los procesos de investigación, docencia y extensión de la universidad.

Por medio del presente documento dejo sentada mi voluntad de participar en este proyecto, para lo cual ponemos a disposición la siguiente investigación:

DATOS DE CONTACTO	
<b>CEDULA DE IDENTIDAD</b>	1002803896
<b>APELLIDOS Y NOMBRES</b>	EDISON PATRICIO LOMAS ALMEIDA
<b>DIRECCIÓN</b>	Av. San Vicente y 12 de Febrero
<b>EMAIL</b>	edisonlomas@gmail.com
<b>TELÉFONO FIJO</b>	NA
<b>TELÉFONO MOVIL</b>	090447763

DATOS DE LA OBRA	
<b>TITULO</b>	“SISTEMA DE GESTIÓN Y SEGUIMIENTO DE PROYECTOS DE SOFTWARE”
<b>AUTOR</b>	EDISON PATRICIO LOMAS ALMEIDA
<b>FECHA</b>	24 DE MAYO DEL 2012
<b>PROGRAMA</b>	PREGRADO
<b>TITULO POR EL QUE</b>	INGENIERÍA EN SISTEMAS COMPUTACIONALES
<b>DIRECTOR</b>	XAVIER MAURICIO REA PEÑAFIEL

### 2. AUTORIZACIÓN DE USO A FAVOR DE LA UNIVERSIDAD

Yo, EDISON PATRICIO LOMAS ALMEIDA, con cédula de identidad Nro. 1002803896, en calidad de autor y titular de los derechos patrimoniales de la obra o trabajo de grado descrito anteriormente, hago entrega del ejemplar respectivo en forma digital y autorizo a la Universidad Técnica del Norte, la publicación de la obra en el Repositorio Digital Institucional y el uso del archivo digital en la biblioteca de la universidad con fines

académicos, para ampliar la disponibilidad del material y como apoyo a la educación, investigación y extensión, en concordancia con la Ley de Educación Superior Artículo 143.



---

Firma

Nombre: EDISON PATRICIO LOMAS ALMEIDA

Cédula: 1002803896

Ibarra a los 24 días del mes de mayo del 2012

## DEDICATORIA

Mi tesis se la dedico a mis queridos padres quienes con su esfuerzo, cariño y apoyo incondicional me incentivaron a culminar una etapa más en mi vida.

Gracias.

Edison.

## **AGRADECIMIENTO**

Agradezco de manera muy especial a Dios, a mis padres y a la distinguida Universidad Técnica del Norte por brindarme la oportunidad de prepararme profesionalmente y desarrollarme como persona en sus aulas día a día.

Mi agradecimiento más sincero al Ingeniero Mauricio Rea director de tesis, por sus recomendaciones acertadas durante todo el proceso de desarrollo del proyecto que ahora concluyen en este documento.

Agradezco al Ingeniero Mauricio Santacruz por permitirme desarrollar el proyecto SIGESPro en su prestigiosa empresa.



## Contenido

CAPÍTULO 1. Introducción.....	3
1.1 Historia de la Empresa VimeWorks.....	3
1.2 Localización.....	3
1.3 Características de la empresa.....	3
1.4 Servicios que presta la empresa.....	3
1.5 Misión.....	4
1.6 Visión.....	4
1.7 Problemática.....	5
1.8 Objetivo General.....	5
1.9 Objetivos Específicos.....	5
CAPÍTULO 2. Diseño de la aplicación.....	7
2.1 Arquitectura de la aplicación.....	7
2.2 JBoss Seam.....	8
2.3 Jasig Cas.....	10
2.4 Exadel Flamingo.....	10
2.5 Seam Security.....	12
2.6 Adobe Flex.....	12
2.7 IBM ilog elixir.....	13
2.8 PostgreSQL.....	14
2.9 JBoss Application Server.....	15
CAPÍTULO 3. Implementación de la aplicación.....	17
3.1 Fase de Inicio.....	17
Lista de Riesgos.....	17
Documento de Visión.....	18
3.2 Fase de Elaboración.....	29
3.2.1 Descripción de Casos de Uso.....	29
3.3 Fase de construcción.....	107
3.3.1 Diagrama de Base de datos.....	107
3.3.2 Diccionario de Datos.....	109
3.3.3 Diagrama de Clases.....	129

3.4 Fase de transición.....	134
CAPÍTULO 4. Conclusiones y Recomendaciones .....	138
4.1 Conclusiones.....	138
4.2 Recomendaciones .....	139
Glosario.....	140
Bibliografía.....	146
Anexos .....	148
Script de creación de base de datos .....	148
Manual de Integración Jasig CAS con LDAP .....	157
Instalación y configuración de JDK.....	157
Instalación y configuración de apache maven.....	162
Configuración, compilación e instalación de CAS-SERVER-WEBAPP .....	169
Manual de Integración Aplicación JBoss SEAM con Adobe Flex .....	175

## **CAPÍTULO 1. Introducción.**

### **1.1 Historia de la Empresa VimeWorks.**

En septiembre del año 2000, David Meza y Carlos Vinueza, ambos estudiantes de la Escuela Politécnica del Ejército, tuvieron la idea de formar una empresa de desarrollo de software que incursionara en nuevas tecnologías, especialmente en desarrollo web. En aquellos años el desarrollo web estaba recién siendo considerado por el mercado. De sus apellidos nació el nombre Vi+Me (VimeWorks) y la concibieron de tal forma que todos los que la integraran tuvieran poder de decisión; entre las principales empresas con las que comenzó su trabajo fue TODO 1 y el Servicio de Rentas Internas; actualmente luego de 11 años en el mercado, se han realizado más de 50 proyectos en diferentes clientes del sector público y privado y continuará su trabajo en los años venideros.

### **1.2 Localización.**

La empresa VimeWorks Cía. Ltda. se encuentra localizada en la provincia de Pichincha, ciudad de Quito en la dirección Avenida Coruña N27-114 y Avenida Orellana.

### **1.3 Características de la empresa.**

La empresa se dedica al desarrollo de software a la medida, capacitación y asesoramiento en tecnología.

### **1.4 Servicios que presta la empresa.**

- Desarrollo de aplicaciones distribuidas sobre tecnología J2EE y .net
- J2EE: JSP, Servlets, XML, Oracle 9iAS, JBoss, Tomcat

- .net : Aplicaciones Windows Forms y ASP.NET
- Bases de datos: Oracle, SQLServer, MySQL, PostgreSQL.
- Consultoría, asesoría para la aplicación de mejores prácticas para el desarrollo de software.
- Uso de herramientas de colaboración y gestión de proyectos opensource y J2EE como CVS, Mailinglists, Ant, entre otros.
- Capacitación en tecnologías OSS, J2EE y .NET
- Administración de OC4J, Tomcat, JBoss
- XDoclets, Ant
- Desarrollo con EJBs, XDoclets, JavaBeans
- Desarrollo de aplicaciones con PHP
- Outsourcing

### 1.5 Misión.

Ser una empresa tecnológica reconocida en la ciudad de Quito, mediante el desarrollo de software a la medida de manera ágil y eficiente, conservando de esta manera la lealtad de nuestros clientes.

### 1.6 Visión.

Ser una organización dedicada a la innovación tecnológica, ofreciendo soluciones creativas y adecuadas para cada cliente, brindando asesoría ágil y adecuada, buscando la fidelidad de nuestros clientes, procurando obtener un margen de rentabilidad apropiado a través de la optimización en el uso de los recursos.

## 1.7 Problemática

La problemática actual de la empresa es no poder controlar de una forma adecuada el desarrollo de los proyectos de software, ni obtener información inmediata sobre el avance de cada proyecto. A raíz de esto surgen varios problemas como retraso en los proyectos, incumplimiento de tareas entre otros.

Por tal motivo existe la necesidad de implementar un sistema de gestión de proyectos, en el cual se pueda obtener información actualizada del avance de cada proyecto y tarea asignada además de mantener un registro de cada una de las tareas asignadas a cada desarrollador.

## 1.8 Objetivo General

Implementar una aplicación web de gestión de proyectos de desarrollo de software, tareas asignadas y requerimientos de usuarios dentro de un entorno de autorización centralizado, permitiendo su seguimiento y reportes de estado de los mismos.

## 1.9 Objetivos Específicos

- Diseñar una herramienta que permita supervisar y controlar el desarrollo de múltiples proyectos de la empresa.
- Integrar con un sistema de autenticación centralizada y proveer servicios de autorización y control de acceso para el sistema mediante el uso de un framework.
- Permitir la asignación de jefes de proyecto y desarrolladores a cada proyecto.
- Facilitar la asignación de tareas a los desarrolladores participantes en los proyectos.
- Permitir al desarrollador reportar avances sobre cada una de sus tareas.

- Permitir el reporte de errores de los proyectos y generar informes de estado de proyectos.

## CAPÍTULO 2. Diseño de la aplicación

### 2.1 Arquitectura de la aplicación.

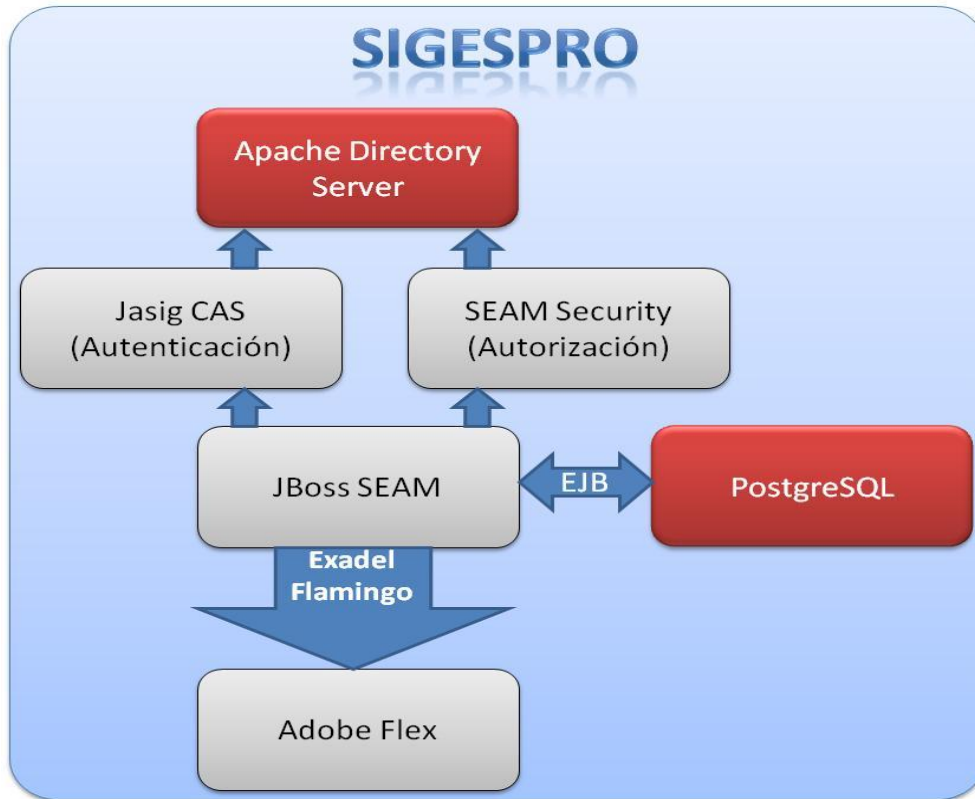
El Sistema de Gestión y Seguimiento de Proyectos de software, tiene como arquitectura la integración de varias herramientas de alto nivel como son: LDAP, J2EE, PostgreSQL, Adobe Flex.

El core de la aplicación está construido en la plataforma J2EE con el frameworks de JBoss Seam, para permitir la autenticación de usuarios utilizamos un servidos LDAP y un Single SignOn que en este caso está provisto por JASIG, con la finalidad de realizar un ahorro de tiempo a los usuarios que necesiten ser autenticados en diferentes aplicaciones.

La autorización de acceso a nuestro sistema está provisto por Seam-Security el cual cumple con la función de controlar el acceso únicamente a recursos asignados, esta parte es fundamental en el desarrollo de toda aplicación y se debe darle gran importancia, ya que en varios sistemas no se realiza este control y muchos desarrolladores lo pasan por alto.

Adicionalmente para la parte de las interfaces de usuario se realizan mediante Adobe Flex, el cual nos permite crear aplicaciones ricas en internet obviamente con un coste de ancho de banda un poco alto, adicionalmente para la interacción entre los módulos Adobe Flex y nuestro sistema utilizamos exadel flamingo para permitir tal conexión a los servicios de nuestro core.

Como motor de base de datos tenemos a una de las mejores plataformas de base de datos que además cuenta con una característica importante, la cual es que esta es opensource y con un costo cero de licenciamiento la cual es PostgreSQL, para el acceso a estos datos utilizamos JPA y EJB 3.0



## 2.2 JBoss Seam.

JBoss Seam<sup>1</sup> es un framework desarrollado por JBoss<sup>2</sup>, una división de Red Hat. Ha sido diseñado desde cero para eliminar la complejidad en la arquitectura y el nivel de la API. Permite a los desarrolladores ensamblar aplicaciones web complejas con simples anotaciones Plain Old Java Objects (POJOs), widgets en componentes de interfaz de usuario y XML.

Este es un framework poderoso para la construcción de aplicaciones Web 2.0, permite la integración con JavaScript y XML (AJAX), Java Server Faces (JSF), Enterprise Java Beans (EJB3), Java Portlets y Gestión de Procesos de Negocio (BPM).

<sup>1</sup> <http://seamframework.org/>

<sup>2</sup> <http://www.jboss.com/>



JBoss Seam combina dos frameworks: *Enterprise JavaBeans EJB3* y *Java Server Faces JSF*. Se puede acceder a cualquier componente EJB desde la capa de presentación refiriéndose a él mediante su nombre de componente Seam. JBoss Seam introduce el concepto de contextos. Cada componente de JBoss Seam existe dentro de un contexto. El contexto conversacional por ejemplo captura todas las acciones del usuario hasta que éste sale del sistema o cierra el navegador, inclusive puede llevar un control de múltiples pestañas y mantiene un comportamiento consistente cuando se usa el botón de regresar del navegador.



Mediante JBoss Seam se puede generar automáticamente una aplicación web con un CRUD (create-read-update-delete) básico a partir de una base de datos existente utilizando una herramienta de línea de comandos llamada *seam-gen* incluida con el framework. El desarrollo WYSIWYG (what you see is what you get) es facilitado a través del uso de un set de herramientas llamado JBoss Tools, que es un conjunto de plug-ins diseñados para el entorno integrado de desarrollo Eclipse. JBoss Seam puede ser integrado con las bibliotecas de componentes JSF JBoss RichFaces o con ICEFaces. Ambas bibliotecas poseen soporte para AJAX.

## 2.3 Jasig Cas.

JASIG<sup>3</sup> Cas fue desarrollado originalmente por la Universidad de Yale. Desde entonces se ha convertido en un proyecto JASIG.

Jasig Central Authentication Service<sup>4</sup> (Servicio de Autenticación Central JASIG) es un inicio de sesión único para la web (Single SingOn). Es un proceso de autenticación de la sesión de usuario que permite que el usuario proporcione sus credenciales (como nombre de usuario y contraseña) una sola vez para acceder a múltiples aplicaciones.

El inicio de sesión único en la autenticación del usuario le permite acceder a todas las aplicaciones que él o ella han sido autorizados a acceder. Elimina las futuras solicitudes de autenticación cuando el usuario cambia las solicitudes durante una sesión en particular. Single SingOn<sup>5</sup> funciona estrictamente con las aplicaciones que acceden mediante un navegador web. La solicitud para acceder a un recurso web es interceptada, ya sea por un componente en el servidor web o por la propia aplicación.

Los usuarios que no lograron ser autenticados se desvían a otro servicio y solo podrá regresar después de una autenticación exitosa.

## 2.4 Exadel Flamingo

Exadel Flamingo<sup>6</sup> es una solución de desarrollo de software de gran alcance para la integración de múltiples aplicaciones de internet enriquecidas (RIAs – Rich Internet Applications) construida sobre Adobe Flex, JavaFX, Java ME, Android y Java Widget Herramientas (Swing y SWT) con JBoss Seam o Spring Framework como la tecnología principal de código abierto del lado del servidor.

---

<sup>3</sup> <http://www.jasig.org/>

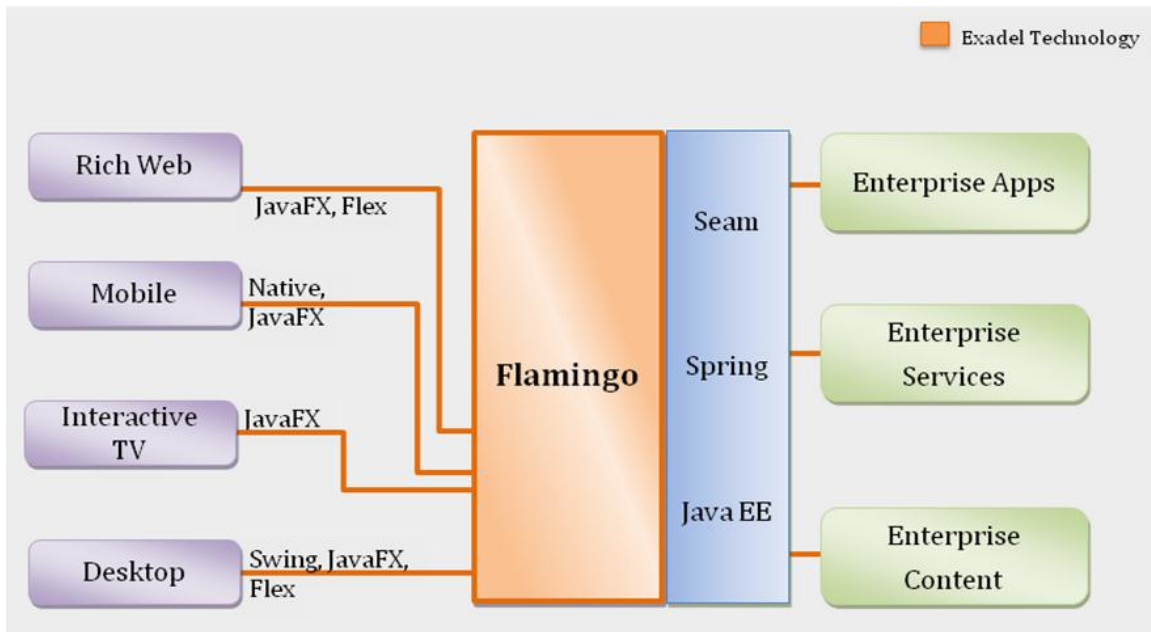
<sup>4</sup> [http://en.wikipedia.org/wiki/Central\\_Authentication\\_Service](http://en.wikipedia.org/wiki/Central_Authentication_Service)

<sup>5</sup> [http://en.wikipedia.org/wiki/Single\\_sign-on](http://en.wikipedia.org/wiki/Single_sign-on)

<sup>6</sup> <http://www.exadel.com>

Exadel Flamingo es una biblioteca de integración y el framework para las aplicaciones de internet que permite crear interfaces de Flex o JavaFX basados en el usuario para sus aplicaciones de JBoss Seam Framework / Spring. Flamingo es lo que permite que estas tecnologías diferentes trabajen juntas.

Flamingo provee una común capa de integración Seam / Spring para tecnologías RIA (aplicaciones de internet enriquecidas) con la que permite al desarrollador trabajar con objetos remotos Seam y servicios de Spring, así como entidades simples en el lado del servidor utilizando un enfoque nativo RIA en el lado del cliente. Esta comunicación entre el cliente y el servidor es transparente.



El desarrollador podrá generar gran parte del código automáticamente en patrones de código sensible. Esto permite a los desarrolladores que trabajan con proyectos de RIA escribir el código relacionado con la lógica de negocio de las aplicaciones en vez de gastar tiempo con la configuración del proyecto.

## 2.5 Seam Security.

Seam Security API es una característica opcional que Seam proporciona autenticación y autorización para la seguridad tanto de dominio y página de recursos dentro del proyecto seam.

La autenticación es una de las características proporcionadas por Seam Security se basan en JAAS (Java Authentication and Authorization Service), y como tal proporciona una API robusta y altamente configurable para la gestión de la autenticación de usuario. Sin embargo, para los requisitos de autenticación menos complejos Seam ofrece un método mucho más simple de autenticación que oculta la complejidad de JAAS.

La autorización, para ello existe una serie de funciones proporcionadas por la API de Seam de seguridad para garantizar el acceso a los componentes, los métodos de componentes, y las páginas.

## 2.6 Adobe Flex.

Flex<sup>7</sup> permite construir a través de una serie de controles de presentación de datos, un componente binario conteniendo la totalidad de la capa de presentación de la aplicación. Este componente está orientado a ser ejecutado en un navegador (Internet Explorer, Firefox, etc.).

Las tecnologías bajo Flex son principalmente, ActionScript y XML. La compilación de un código fuente de Flex nos da como resultado un objeto ejecutable Adobe Flash que se puede usar tanto en el escritorio cómo en la web. Tiene varios componentes y características que aportan funcionalidades tales como Servicios Web, objetos remotos, arrastrar y soltar, columnas ordenables, gráficas, efectos de animación y otras interacciones simples. El cliente solo carga la aplicación una vez, mejorando así el flujo de datos frente a aplicaciones basadas en HTML (PHP, ASP, JSP, CFMX), las cuales requieren de ejecutar plantillas en el

---

<sup>7</sup> <http://www.adobe.com/es/products/flex.html>

servidor para cada acción. El lenguaje y la estructura de archivos de Flex buscan el desacoplamiento de la lógica y el diseño.

Este componente fue inicialmente liberado como una aplicación de la J2EE o biblioteca de etiquetas JSP que compilaba el lenguaje de marcas Flex (MXML) y ejecutaba mediante ActionScript aplicaciones Flash (archivos SWF binarios). Versiones posteriores de Flex soportan la creación de archivos estáticos que son compilados, y que pueden ser distribuidos en línea sin la necesidad de tener una licencia de servidor.

## 2.7 IBM ilog elixir.

IBM ilog elixir amplía las plataformas Adobe Flex y Adobe AIR añadiendo controles de interfaz de usuario para obtener visualizaciones más intuitivas e interactivas.

Proporciona una amplia gama de visualizaciones gráficas avanzadas para las aplicaciones de líneas de negocio más exigentes. Los desarrolladores pueden aprovechar fácilmente las muchas pantallas de visualización Flex para crear una nueva categoría de aplicaciones que sean funcionalmente superiores y visualmente atractivas.

- Ofrece una amplia variedad de controles: gráficas 3D, diagramas radiales y diagramas de árbol, barras, indicadores y mini gráficos, diagramas organizativos , mapas y mapas de calor, escalas de tiempo y calendarios, visualizaciones de planificaciones orientadas a recursos y orientadas a tareas, y servicios avanzados para crear visualizaciones de diagramas intuitivas.
- Permite que los desarrolladores utilicen los componentes proporcionados o los personalicen para cumplir los requisitos exactos del usuario.
- Las mejoras se compilan con el último SDK (software development kit) de Flex, garantizando la conformidad con las características más nuevas de la plataforma base.

- Se integra a la perfección con los entornos de desarrollo integrado (IDE) de Adobe Flash Builder.
- Se ajusta con precisión al entorno de Flex, admitiendo totalmente el nuevo Flex 4 SDK, sus enlaces de datos y modelos de sucesos, y admitiendo también los efectos de la interacción de los usuarios estándar de Flex.
- Se despliega en la prácticamente omnipresente plataforma Flash, asegurando la compatibilidad en casi todos los navegadores.
- Su capacidad se puede ampliar para manejar grandes conjuntos de datos y dar soporte a aplicaciones a nivel de empresa.

## 2.8 PostgreSQL.

PostgreSQL<sup>8</sup> es un sistema de gestión de base de datos relacional orientada a objetos. Se ejecuta en todos los principales sistemas operativos, incluyendo Linux, UNIX (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64) y Windows.

Es totalmente compatible con ACID (atomicidad, coherencia, aislamiento, durabilidad), tiene soporte completo para claves foráneas, uniones, vistas, disparadores y procedimientos almacenados. Incluye la mayor parte de SQL: 2008 los tipos de datos, incluyendo INTEGER, NUMERIC, BOOLEAN, CHAR, VARCHAR, DATE, INTERVAL, y TIMESTAMP. También es compatible con el almacenamiento de objetos binarios grandes, como imágenes, sonidos o vídeo. Tiene interfaces nativas de programación C/ C++, Java, .Net, Perl, Python, Ruby, Tcl, ODBC, entre otros, y la documentación excepcional.

Una base de datos de clase empresarial, PostgreSQL cuenta con características avanzadas tales como Multi-Version Control de concurrencia (MVCC), punto en el tiempo de recuperación, tablespaces, replicación asincrónica, transacciones anidadas (puntos de retorno), backups en caliente, un optimizador de consultas sofisticado planificador, y escribir

---

<sup>8</sup> <http://en.wikipedia.org/wiki/PostgreSQL>

por delante de registro para la tolerancia a fallos. Es compatible con conjuntos de caracteres internacionales, las codificaciones de caracteres multibyte, unicode, y es la configuración regional para la ordenación, mayúscula y minúscula, y el formato. Es altamente escalable, tanto en la gran cantidad de datos que puede manejar y en el número de usuarios concurrentes que puede acomodar. Hay sistemas activos de PostgreSQL en entornos de producción que manejan más de 4 terabytes de datos.

## 2.9 JBoss Application Server.

JBoss<sup>9</sup> AS es el primer servidor de aplicaciones de código abierto, preparado para la producción y certificado J2EE, disponible en el mercado, ofreciendo una plataforma de alto rendimiento para aplicaciones de e-business.



<sup>9</sup> <http://www.jboss.org/>

Combinando una arquitectura orientada a servicios revolucionaria con una licencia de código abierto, JBoss AS puede ser descargado, utilizado, incrustado y distribuido sin restricciones por la licencia. Por este motivo es la plataforma más popular de middleware para desarrolladores, vendedores independientes de software y también para grandes empresas. Las características destacadas de JBoss incluyen:

- Producto de licencia de código abierto sin coste adicional.
- Cumple los estándares.
- Confiable a nivel de empresa
- Incrustarle orientado a arquitectura de servicios.
- Flexibilidad consistente
- Servicios del middleware para cualquier objeto de Java
- Ayuda profesional 24x7 de la fuente
- Soporte completo para JMX

Una distinción importante para esta clase de herramienta es que no sólo corresponde a un servidor que se ejecuta en Java, pero en realidad se aplica la parte de Java EE. Debido a que está basado en Java, el servidor de aplicaciones JBoss funciona en múltiples plataformas, utilizable en cualquier sistema operativo que soporte Java. JBoss AS fue desarrollado por JBoss, ahora una división de Red Hat.



## CAPÍTULO 3. Implementación de la aplicación

### 3.1 Fase de Inicio

#### Lista de Riesgos

##### 1. Propósito

El propósito de esta lista es identificar los riesgos que puedan ocurrir durante el transcurso del desarrollo del proyecto y desarrollar estrategias para contrarrestarlos.

Puntaje/ Magnitud	Descripción del riesgo e impacto	Estrategia de mitigación y/o plan de contingencia
10	Restablecer la arquitectura del sistema	Seleccionar la mejor herramienta de arquitectura que pueda existir.
9	Cambio en los procesos de desarrollo de software en la empresa	Tratar de generalizar el proceso de desarrollo para que la aplicación pueda adaptarse.
10	Sea requerido para un proyecto y no poder terminar con este.	Tratar de motivar continuamente a la empresa sobre las ventajas de este desarrollo.
9	Ya no exista el interés de la empresa para continuar con este proyecto	Tratar de motivar continuamente a la empresa sobre las ventajas de este desarrollo.

## Documento de Visión

### 1. Introducción

El objeto de esta sección es analizar y precisar las características y las necesidades para la implementación de un sistema que le permita gestionar los proyectos de software en desarrollo. Este documento agrupa las funcionalidades requeridas por el usuario final.

Las empresas que se dedican al desarrollo de software requieren de herramientas que le permitan realizar un seguimiento continuo a los dichos proyectos. La empresa VimeWorks dedicada a la implementación de software se halla en la necesidad de tener su propio sistema que le permita gestionar los procesos durante el desarrollo de proyectos. Mediante la necesidad de la empresa nace el sistema SIGESPro.

### 2. Alcance

SIGESPro – Sistema de Gestión de Proyectos de software - le permitirá a la empresa gestionar el control y supervisión de múltiples proyectos, adicionalmente permitirá a los jefes de proyecto asignar tareas específicas a cada uno de los desarrolladores.

El sistema generará un diagrama de Gantt en base a las tareas del proyecto y emitirá reportes del avance de los proyectos. Adicionalmente de los requerimientos y tareas permitirá el manejo de los costos es decir que al final del proyecto se podrá obtener cual fue el costo del desarrollo del proyecto. Por último contendrá un módulo de issues tracking (gestor de errores), con estados y prioridad. Se visualizará alertas de tareas asignadas y alertas de vencimiento de tiempo para realizar las tareas.

### 3. Posicionamiento

Para la empresa VimeWorks, ubicada en la ciudad de Quito, provincia Pichincha cuenta con sistemas que le permiten gestionar de forma separada los procesos durante el desarrollo de software, pero al manejarse de manera separada no le permiten obtener información del estado del proyecto. Por esta razón es importante el desarrollo de un sistema acorde a las necesidades de la empresa de esta forma le permitirá manejar la información desde un solo sistema, obtener indicadores y reportes con información al día.

#### 3.1 Declaración del problema

El problema de	La inexistencia en la empresa VimeWorks de una herramienta la cual permita, realizar un seguimiento adecuado a los proyectos de desarrollo de software y que permita almacenar información
afecta a	Jefes y gerentes de proyectos ya que no disponen de información actualizada del estado de los proyectos en desarrollo.
el impacto de este problema es	Retrasos en los plazos de tiempo establecidos para cada proyecto, no disponer de un historial de los problemas reportados.
una solución exitosa debería	Implementar un sistema de gestión de proyectos que se adapte a las necesidades de la empresa.  Permitir un mejor manejo de los procesos de desarrollo en un proyecto de software.

### 3.2 Declaración del posicionamiento del proyecto

Para	VimeWorks
Quien(es)	Elabora proyectos de software y no cuenta con un sistema que permita gestionarlos.
El (nombre del producto)	SIGESPro
Que	Es una aplicación web de gestión de proyectos de desarrollo de software, el que permite asignar tareas y requerimientos de usuarios dentro de un entorno de autorización centralizado, el cual permite realizar el seguimiento y reportes de estado de los mismos.
Debido a que	El software Microsoft Project Server es una herramienta la cual contiene una funcionalidad similar pero con la desventaja de su alto costo de licenciamiento.
Nuestro producto	SIGESPro, se ajusta a las necesidades puntuales de la empresa y no tendrá costo de licenciamiento.

#### 4. Descripción de los interesados en este proyecto

##### 4.1 Resumen de Stakeholders.

Nombre	Descripción	Responsabilidades
Ing. Mauricio Rea	Director de tesis	Supervisar el correcto desarrollo del sistema SIGESPro.
Sr. Edison Lomas	Egresado de la Carrera de Ingeniería en Sistemas Computacionales	Encargado de la implementación del sistema SIGESPro.
Ing. Mauricio Santacruz	Responsable a nivel directivo de VimeWorks Cía. Ltda.	Establecer las necesidades principales y requerimientos necesarios para el sistema de gestión de proyectos de desarrollo de software.

##### 4.2 Resumen de usuarios.

Nombre	Descripción	Responsabilidades
Administrador	Administrador del sistema.	Creación de proyectos, creación de grupos de desarrollo.
Jefe de Proyecto	Administrador de proyectos asignados al mismo.	Crea tareas dentro del proyecto y las asigna al grupo de desarrolladores.

Desarrollador	Miembro del equipo de programadores	Reporta avances de las tareas asignadas.
Usuario	Usuario funcional del sistema a desarrollarse	Solicita nuevos requerimientos y/o reporta errores (issues).

### 4.3 Entorno del usuario

- Permitir supervisar y controlar el desarrollo de múltiples proyectos de la empresa.
- Integrarse con un sistema de autenticación centralizada para evitar la duplicidad de usuarios y contraseñas. Adicionalmente debe proveer servicios de autorización y control de acceso.
- Permitir la asignación de jefes de proyecto y equipo de desarrolladores a cada proyecto.
- Permitir el registro de requerimientos, asignarlos a los desarrolladores y reportar issues del proyecto.
- Generar informes de estado de proyectos.

## 5. Descripción general del sistema

### 5.1 Perspectiva del Sistema

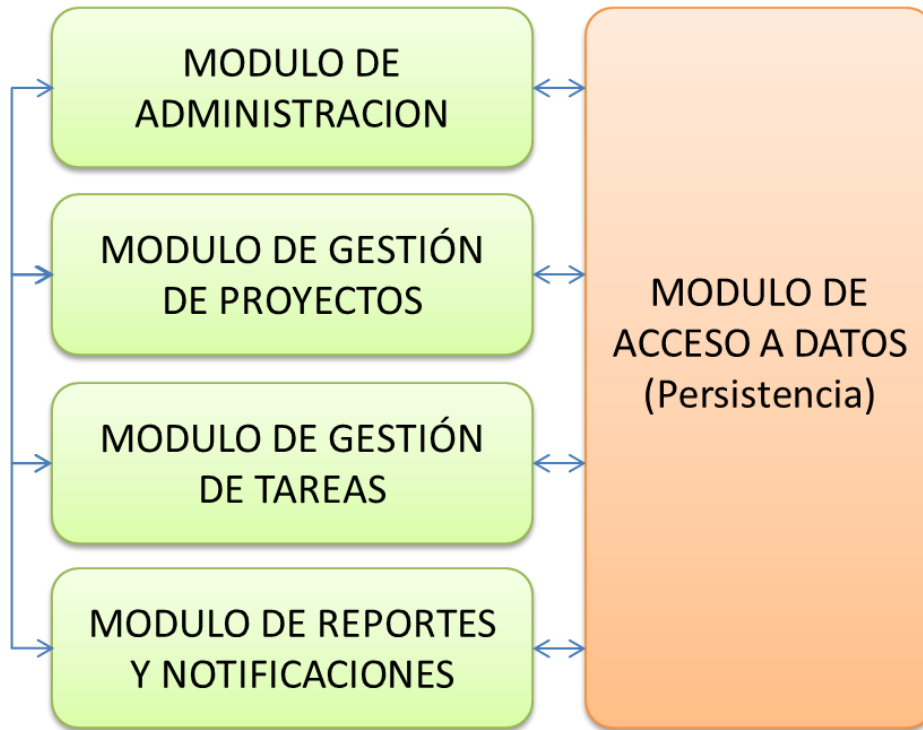
SIGESPro (Sistema de Gestión y Seguimiento de Proyectos) manejará cuatro tipos de roles: Administrador, Jefe de Proyecto, Desarrolladores y Usuarios.

El rol Administrador permitirá la creación de nuevos proyectos con sus respectivos parámetros y podrá asignar al jefe de proyecto y al equipo de desarrollo asignado al mismo. También estará en la capacidad de revisar un reporte general del estado de todos los proyectos, con indicadores sobre el cumplimiento o no de las tareas.

El rol Jefe de Proyecto estará en la capacidad de asignar tareas a los desarrolladores y revisar su avance, recibirá todas las peticiones de cambio de los usuarios y las direccionará a los desarrollos para su solución. Adicionalmente contará con un reporte en formato Gantt que le permitirá ver a detalle y en forma gráfica las tareas que sus desarrolladores tienen asignadas y un reporte estadístico con los indicadores de cumplimiento de cada uno de sus proyectos asignados.

El rol Desarrollador contará con la interfaz de manejo de los issues asignados a él y podrá registrar el inicio de sus actividades, avance de cumplimiento y novedades sobre la misma. También contará con un gráfico de Gantt que le permita visualizar sus propias tareas, adicionalmente tendrá un reporte estadístico de los indicadores de cumplimiento de sus actividades.

El rol Usuarios facilitará el reporte issues en el sistema para que sean atendidos por el equipo del proyecto y podrán consultar el estado de los mismos. Cuando un requerimiento reportado por un usuario sea completado se le enviará una notificación por correo de su solución.



## 5.2 Dependencias

SIGESPro depende de la forma en la que la empresa VimeWorks lleva a cabo el proceso de desarrollo de software y de la jerarquía de personal operativo.

## 5.3 Necesidades y Características

Necesidad del cliente	Prioridad	Solución Actual	Solución Propuesta
Asignar recursos a un proyecto	Alta	Asignan de forma verbal	Módulo de Gestión de Proyectos
Autenticación centralizada	Alta	No dispone.	Integrar al sistema con un servidor de autenticación centralizada.
Historial de incidencias	Alta	Se los reporta en un sistema	Almacenar de las tareas con su respectivo tipo en la base de



		independiente.	datos.
Indicadores de los proyectos	Alta	No dispone.	Realizar informes en base a los requerimientos del usuario.
Seguimiento a proyectos y equipo de desarrollo	Alta	Mediante reuniones del equipo desarrollador se comunican los avances del proyecto y las tareas.	Con alertas al inicio y finalización de cada tarea. Además del reporte de estados de cada una de las tareas.

<b>Beneficio</b>	<b>Características que lo apoyan</b>
Costo de licenciamiento bajo	Ya que el sistema no va a tener el costo de licenciamiento ya que la empresa ya cuenta con las licencias del software utilizado para el desarrollo de la aplicación.
Supervisión y control de proyectos	Mediante el servicio de notificaciones y los reportes.
Framework para servicios de autorización y control de acceso	Con la integración con CAS y el Seam-Security.
Issue tracking	Permite a los usuarios reportar errores presentados.
Reportes al día	Ya que la información va a estar disponible las 24

	horas los 365 días del año.
Seguridad en la información	Control de acceso en la asignación de roles a cada usuario.
Manejar tiempos rápidos de respuesta a los problemas empresariales.	Permitir de manera oportuna la toma de decisiones.

#### 5.4 Alternativas y sistemas competencia

Actualmente en el medio existen varias alternativas abiertas como Planner, Open Planner, Gantt Project, etc. para el manejo de gestión de proyectos. El sistema que más se adapta a los procesos de la empresa es el sistema Microsoft Project Server pero debido a sus altos costos de licenciamiento no la hacen una herramienta asequible para la empresa.

#### 6. Otros requerimientos del sistema

INFORMACIÓN GENERAL	
Número de usuarios actuales:	25
Número máximo de usuarios en 12 meses:	50
Número de transacciones actual esperado:	25
Número de transacciones esperado en 12 meses:	50
Disponibilidad de la aplicación:	24/7 todo el año

**Requerimientos estándares de Sistema:**

Software	Nombre	Versión	Observación
Sistema Operativo:	Red Hat Enterprise Linux Server	5	
Base de Datos:	PostgreSQL	8.4	
Servidor de aplicaciones:	JBoss Application Server	6	
Otros:	JDK	1.6	

HARDWARE	
Servidor:	Procesador XEON, 5 Gb. En RAM
Almacenamiento:	500 Mb
Otros:	

## Especificaciones de Rendimiento

Max Núm. de usuarios concurrentes LAN:	25
Max Núm. de usuarios concurrentes WAN:	25
Hora Pico:	09:00 – 18:00
Tiempo de respuesta esperado x usuario:	5 segundos
Tiempo de respuesta máx. x usuario:	10 segundos
Tiempo promedio de la transacción más pesada:	10 segundos
Tiempo máximo de la transacción más pesada:	20 segundos

## 3.2 Fase de Elaboración

### 3.2.1 Descripción de Casos de Uso

#### 3.2.1.1 Administración de Catálogos

##### Historial de Revisiones:

Fecha	Versión	Descripción	Autor
04-01-2011	1.0	Descripción inicial del caso de uso	Edison Lomas
06-01-2011	1.1	Modificación y actualización del caso de uso a petición del usuario funcional.	Edison Lomas

## Prototipo de las pantallas

Pantalla principal de la administración de catálogos

The screenshot shows a web application interface for catalog management. At the top, there is a search bar titled "Buscar Catalogo" with a dropdown menu for "Tipo Catalogo" (currently showing "Selecciones...") and a text input for "Nombre Catalogo". Below the search bar is a "Buscar" button. The main area is titled "Catalogos" and contains a table with the following columns: "Nombre Catalogo", "Descripción", "Tipo Catalogo", and "Opciones". The table lists several catalog entries, including "Error", "Privada", and "Requerimiento". At the bottom of the table is a pagination control showing "1" and "2". Below the table is a "Nuevo" button.

Nombre Catalogo	Descripción	Tipo Catalogo	Opciones
dfgdsfgd	gewrtetwert	Prioridad	
dsgdsgdg	weterterwert	Prioridad	
dsfgdfg	ewrtewyterwert	Prioridad	
dsfgsdf	gdsdgdgdsgfdg	Prioridad	
dsfgsdfgsd	gfsdgdgsgdfg	Prioridad	
Error		Tipo de tarea	
ertewtwer	tthfdhghfdgh	Prioridad	
fghdfghdfg	hdfhdhdfgh	Prioridad	
Privada	Este es un catalogo para agrupar las instituciones privadas	Instituciones	
Requerimiento		Tipo de tarea	

Pantalla de creación de un nuevo catálogo.

The screenshot shows a form titled "Nuevo Catalogo" for creating a new catalog. It includes a dropdown menu for "Tipo Catalogo" (showing "Selecciones..."), a text input for "Nombre Catalogo", and a larger text area for "Descripción". At the bottom, there is a legend for "Campos requeridos" (required fields) and two buttons: "Guardar" (Save) and "Cancelar" (Cancel).

Pantalla de edición de un catálogo.

Editar Catalogo

Tipo Catalogo: Prioridad

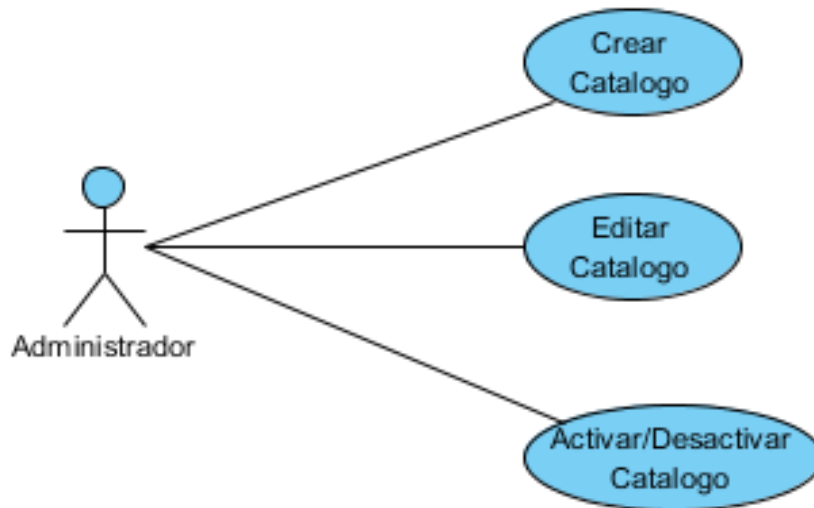
Nombre Catalogo: dfgdsfgd

Descripción: gewrtetwetwert

\* Campos requeridos

Aceptar Cancelar

**Diagrama de Caso de Uso:**



**Prerrequisitos:**

1. Usuario registrado y autorizado como administrador del sistema
2. Usuario debidamente autenticado en el sistema

## Flujo Principal – Crear Catalogo:

1. Se ingresar a la pantalla de administración de catálogos mediante el menú de la aplicación.
2. Se presenta un pantalla con los siguientes filtros de búsqueda:
  - Tipo de Catalogo
    - Editable en lista desplegable
    - Catalogo
  - Nombre Catalogo
    - Editable en caja de texto
    - Texto de máximo 50 caracteres.

Juntamente con un botón **Buscar**, una grilla de resultados debidamente paginada, y el botón **Nuevo**.

3. Al clic en el botón **Nuevo** se abre una ventana de registro de información con los siguiente campos:
  - Tipo de Catalogo
    - Editable en lista desplegable
    - Catalogo
    - Obligatorio
  - Nombre Catalogo
    - Editable en caja de texto
    - Texto de máximo 50 caracteres.
    - Obligatorio
  - Descripción
    - Editable en caja de texto
    - Texto de máximo 500 caracteres
    - No obligatorio

Juntamente con un botón **Guardar** y el botón **Cancelar**



4. Al dar clic en el botón **Guardar** se validan los datos, se guarda la información y se presenta un mensaje indicando que el catalogo se guardó satisfactoriamente. Si ocurriese algún error se presentara el mensaje correspondiente.
  
5. Al dar clic en **Cancelar** se cierra la ventana sin realizar ninguna acción.

### Flujo Alternativo 1 – Editar Catalogo:

1. Se ingresa a la pantalla de administración de catálogos mediante el menú de la aplicación.
  
2. Se presenta un pantalla con los siguientes filtros de búsqueda:
  - Tipo de Catalogo
    - Editable en lista desplegable
    - Catalogo
  - Nombre Catalogo
    - Editable en caja de texto
    - Texto de máximo 50 caracteres.

Juntamente con un botón **Buscar**, una grilla de resultados debidamente paginada, y el botón **Nuevo**.

3. Al clic en el botón **Buscar** se actualiza la grilla de resultado en base a los criterios de búsqueda. En la parte derecha de los registros de la grilla existen las opciones **Editar, Activar/Desactivar**.
  
4. Al dar clic en la opción **Editar** se abre una ventana de edición de información con los siguiente campos:
  - Tipo de Catalogo
    - Editable en lista desplegable
    - Catalogo

- Obligatorio
- Nombre Catalogo
  - Editable en caja de texto
  - Texto de máximo 50 caracteres.
  - Obligatorio
- Descripción
  - Editable en caja de texto
  - Texto de máximo 500 caracteres
  - No obligatorio

Juntamente con un botón **Aceptar** y el botón **Cancelar**.

6. Al dar clic en el botón **Aceptar** se validan los datos, se actualiza la información y se presenta un mensaje indicando que el catalogo se actualizo satisfactoriamente. Si ocurriese algún error se presentara el mensaje correspondiente.
5. Al dar clic en **Cancelar** se cierra la ventana sin realizar ninguna acción.

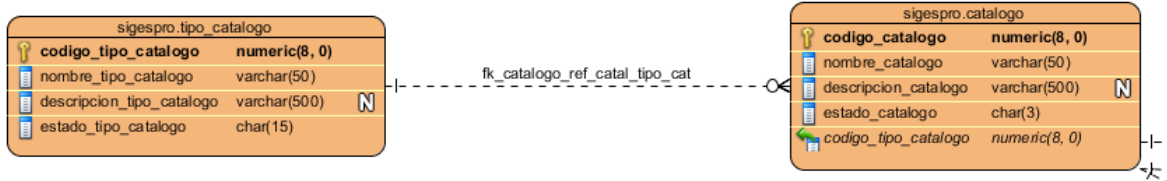
### **Flujo Alternativo 2 – Activar/Desactivar Catalogo:**

1. Se ingresa a la pantalla de administración de catálogos mediante el menú de la aplicación.
2. Se presenta una pantalla con los siguientes filtros de búsqueda:
  - Tipo de Catalogo
    - Editable en lista desplegable
    - Catalogo
  - Nombre Catalogo
    - Editable en caja de texto
    - Texto de máximo 50 caracteres.

Juntamente con un botón **Buscar**, una grilla de resultados debidamente paginada, y el botón **Nuevo**.

3. Al clic en el botón **Buscar** se actualiza la grilla de resultado en base a los criterios de búsqueda. En la parte derecha de los registros de la grilla existen las opciones **Editar** y **Activar/Desactivar** según corresponda.
4. Si el catalogo se encuentra activo se presenta la opción **Desactivar**, caso contrario se presenta la opción **Activar**.
5. Al dar clic en la opción correspondiente el catalogo se activa o se desactiva según corresponda y se presenta un mensaje indicando que el catalogo se actualizo correctamente. Si ocurriese algún error se presentara el mensaje correspondiente.

### Tablas que intervienen en el caso de uso.



### 3.2.1.2 Administración de Recursos

#### Historial de Revisiones:

Fecha	Versión	Descripción	Autor
05-01-2011	1.0	Descripción inicial del caso de uso	Edison Lomas
10-01-2011	1.1	Modificación y actualización del caso de uso a petición del usuario funcional.	Edison Lomas
12-01-2011	1.2	Modificación y actualización del caso de uso a petición del usuario funcional.	Edison Lomas

## Prototipo de las pantallas

Pantalla principal de la administración de recursos

Buscar Recurso

Tipo: Selecciones...  
Nombre:

Buscar

Recursos

Nombre	Descripción	URL	Opciones
Catálogos	Este es menú para el manejo de los catálogos.	/pages/administracion/catalogos.seam	
Instituciones	Este es menú para el manejo de las Instituciones.	/pages/administracion/instituciones.seam	
Proyectos	Este es menú para el manejo de los Proyectos.	/pages/administracion/proyectos.seam	
Usuarios	Página creada para la sincronizacion de los usuarios	/pages/administracion/usuarios.seam	
Proyecto			
Proyectos Asignados		/pages/proyecto/proyectosAsignados.seam	
Tareas Asignadas		/pages/proyecto/tareasAsignadas.seam	
Recursos		/pages/administracion/recursos.seam	
Administración	Este es el menú para la sección de Administración.		

Nuevo

Pantalla para la creación de un menú.

Nuevo Catalogo

Tipo:  Menú  Menú Item

Nombre:

Descripción:

\* Campos requeridos

Guardar Cancelar

Pantalla para la creación de nuevo menú ítem.

The screenshot shows a window titled "Nuevo Catalogo" with a close button in the top right corner. The form contains the following fields and controls:

- Tipo:** Radio buttons for "Menú" (unselected) and "Menú Item" (selected).
- Menú:** A dropdown menu currently showing "Selecciones...".
- Nombre:** An empty text input field.
- URL:** An empty text input field.
- Descripción:** A large empty text area with a vertical scrollbar.
- Campos requeridos:** A red asterisk icon followed by the text "Campos requeridos".
- Buttons:** Two blue buttons labeled "Guardar" and "Cancelar".

Pantalla para la actualización de un menú.

The screenshot shows a window titled "Editar Catalogo" with a close button in the top right corner. The form contains the following fields and controls:

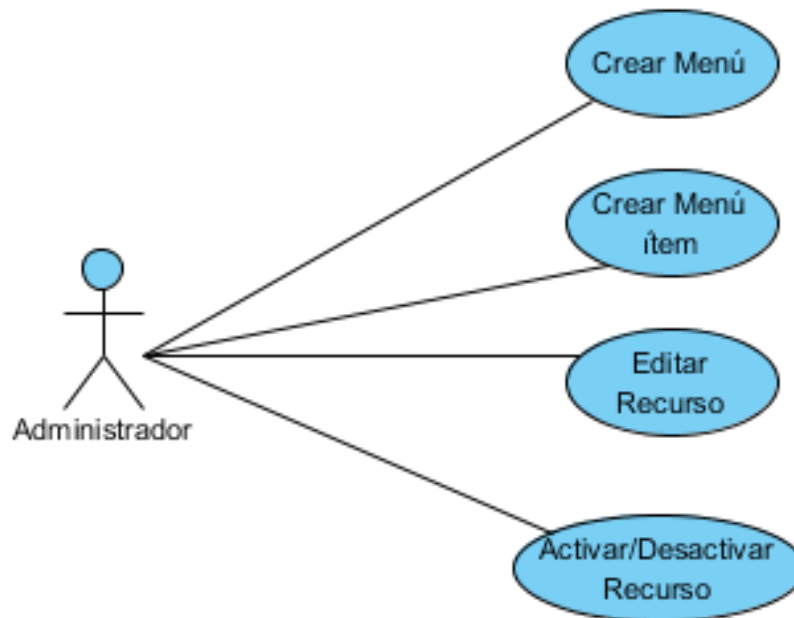
- Tipo:** Radio buttons for "Menú" (selected) and "Menú Item" (unselected).
- Nombre:** A text input field containing the word "Proyecto".
- Descripción:** A large empty text area with a vertical scrollbar.
- Campos requeridos:** A red asterisk icon followed by the text "Campos requeridos".
- Buttons:** Two blue buttons labeled "Aceptar" and "Cancelar".

Pantalla para la actualización de un menú ítem.

The screenshot shows a window titled "Editar Catalogo" with a close button in the top right corner. The form contains the following fields and controls:

- Tipo:** Radio buttons for "Menú" (unselected) and "Menú Item" (selected).
- Menú:** A dropdown menu currently showing "Administración".
- Nombre:** A text input field containing the word "Instituciones".
- URL:** A text input field containing the path "/pages/administracion/instituciones.seam".
- Descripción:** A text area containing the text "Este es menú para el manejo de las Instituciones.".
- Campos requeridos:** A red asterisk icon followed by the text "Campos requeridos".
- Buttons:** Two blue buttons labeled "Aceptar" and "Cancelar".

## Diagrama de Caso de Uso:



## Prerrequisitos:

1. Usuario registrado y autorizado como administrador del sistema
2. Usuario debidamente autenticado en el sistema.
3. Haber registrado instituciones en el sistema.

## Flujo Principal – Crear Menú:

1. Se ingresar a la pantalla de administración de recursos mediante el menú de la aplicación.
2. Se presenta un pantalla con los siguientes filtros de búsqueda:
  - Tipo
    - Editable en lista desplegable
    - Catalogo
    - Valores: Menú, Menú Ítem

- Menú
  - Editable en lista desplegable
  - Catalogo (Referencia a la tabla recurso los tipo menú)
  - Visible únicamente cuando el tipo sea igual a menú ítem
- Nombre
  - Editable en caja de texto
  - Texto de máximo 50 caracteres.

Juntamente con un botón **Buscar**, una grilla de resultados debidamente paginada, y el botón **Nuevo**.

3. Al dar clic en el botón **Nuevo** se abre una ventana de registro de información con los siguiente campos:

- Tipo
  - Editable en botones de radio
  - Catalogo
  - Obligatorio
- Menú
  - Editable en lista desplegable
  - Catalogo (Referencia a la tabla recurso los tipo menú)
  - Visible únicamente cuando el tipo sea igual a menú ítem
- Nombre
  - Editable en caja de texto
  - Texto de máximo 20 caracteres.
  - Obligatorio
- URL
  - Editable en caja de texto
  - Texto de máximo 100 caracteres
  - Obligatorio
- Descripción
  - Editable en caja de texto
  - Texto de máximo 100 caracteres
  - No obligatorio



Juntamente con un botón **Guardar** y el botón **Cancelar**

4. Al seleccionar en el campo **Tipo** Menú desaparecen los campos Menú y URL.
5. Al dar clic en el botón **Guardar** se validan los datos, se guarda la información y se presenta un mensaje indicando que el Recurso se guardó satisfactoriamente. Si ocurriese algún error se presentara el mensaje correspondiente.
6. Al dar clic en **Cancelar** se cierra la ventana sin realizar ninguna acción.

### Flujo Alternativo 1 – Crear Menú ítem:

1. Se ingresa a la pantalla de administración de recursos mediante el menú de la aplicación.
2. Se presenta una pantalla con los siguientes filtros de búsqueda:
  - Tipo
    - Editable en lista desplegable
    - Catalogo
    - Valores: Menú, Menú Ítem
  - Menú
    - Editable en lista desplegable
    - Catalogo (Referencia a la tabla recurso los tipo menú)
    - Visible únicamente cuando el tipo sea igual a menú ítem
  - Nombre
    - Editable en caja de texto
    - Texto de máximo 50 caracteres.

Juntamente con un botón **Buscar**, una grilla de resultados debidamente paginada, y el botón **Nuevo**.

3. Al clic en el botón **Nuevo** se abre una ventana de registro de información con los siguiente campos:

- Tipo
  - Editable en botones de radio
  - Catalogo
  - Obligatorio
- Nombre
  - Editable en caja de texto
  - Texto de máximo 20 caracteres.
  - Obligatorio
- Menú
  - Editable en lista desplegable
  - Catalogo (Referencia a la tabla recurso los tipo menú)
  - Obligatorio
  - Visible únicamente cuando el tipo sea igual a menú ítem
- URL
  - Editable en caja de texto
  - Texto de máximo 100 caracteres
  - Obligatorio
- Descripción
  - Editable en caja de texto
  - Texto de máximo 100 caracteres
  - No obligatorio

Juntamente con un botón **Guardar** y el botón **Cancelar**

4. Al dar clic en el botón **Guardar** se validan los datos, se guarda la información y se presenta un mensaje indicando que el Recurso se guardó satisfactoriamente. Si ocurriese algún error se presentara el mensaje correspondiente.

5. Al dar clic en **Cancelar** se cierra la ventana sin realizar ninguna acción.

## Flujo Alternativo 2 – Editar Recurso:

1. Se ingresa a la pantalla de administración de recursos mediante el menú de la aplicación.
2. Se presenta una pantalla con los siguientes filtros de búsqueda:
  - Tipo
    - Editable en lista desplegable
    - Catalogo
    - Valores: Menú, Menú Ítem
  - Menú
    - Editable en lista desplegable
    - Catalogo (Referencia a la tabla recurso los tipo menú)
    - Visible únicamente cuando el tipo sea igual a menú ítem
  - Nombre
    - Editable en caja de texto
    - Texto de máximo 50 caracteres.

Juntamente con un botón **Buscar**, una grilla de resultados debidamente paginada, y el botón **Nuevo**.

3. Al clic en el botón **Buscar** se actualiza la grilla de resultado en base a los criterios de búsqueda. En la parte derecha de los registros de la grilla existen las opciones **Editar** y **Activar/Desactivar**.
4. Al dar clic en la opción **Editar** se abre una ventana de edición de información con los siguientes campos:
  - Tipo
    - Editable en botones de radio
    - Catalogo
    - Obligatorio
  - Nombre
    - Editable en caja de texto
    - Texto de máximo 20 caracteres.

- Obligatorio
- Menú
  - Editable en lista desplegable
  - Catalogo (Referencia a la tabla recurso los tipo menú)
  - Obligatorio
  - Visible únicamente cuando el tipo sea igual a menú ítem
- URL
  - Editable en caja de texto
  - Texto de máximo 100 caracteres
  - Obligatorio
  - Visible únicamente cuando el tipo sea igual a menú ítem
- Descripción
  - Editable en caja de texto
  - Texto de máximo 100 caracteres
  - No obligatorio

Juntamente con un botón **Aceptar** y el botón **Cancelar**.

5. Al dar clic en el botón **Aceptar** se validan los datos, se actualiza la información y se presenta un mensaje indicando que el recurso se actualizo satisfactoriamente. Si ocurriese algún error se presentara el mensaje correspondiente.
6. Al dar clic en **Cancelar** se cierra la ventana sin realizar ninguna acción.

### Flujo Alternativo 3 – Activar/Desactivar Recurso:

1. Se ingresa a la pantalla de administración de recursos mediante menú de la aplicación.
2. Se presenta un pantalla con los siguientes filtros de búsqueda:
  - Tipo
    - Editable en lista desplegable
    - Catalogo



### 3.2.1.3 Administración de Usuarios

#### Historial de Revisiones:

Fecha	Versión	Descripción	Autor
11-01-2011	1.0	Descripción inicial del caso de uso	Edison Lomas

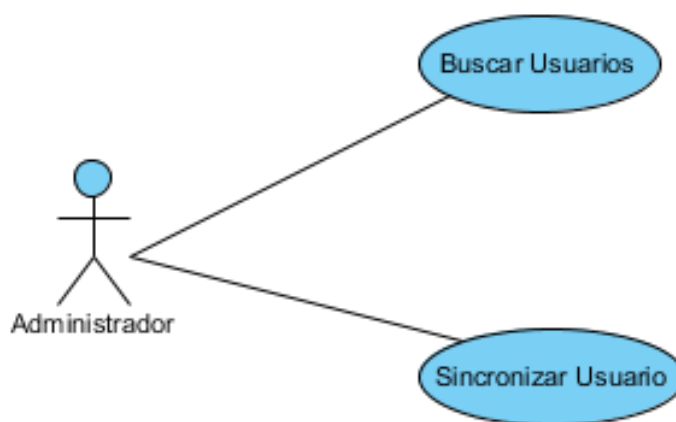
## Prototipo de las pantallas

Pantalla de administracion y sincronizacion de usuarios.

The screenshot displays a web interface for user management. At the top, there is a search section titled "Buscar Usuario" with a dropdown menu for "Rol" (currently showing "Selecciones...") and a text input field for "Nombre". Below these is a "Buscar" button. Underneath is a table titled "Usuarios" with columns for "Nombre", "Rol", and "Estado". The table contains four rows of user data. At the bottom of the interface is a "Sincronizar" button.

Nombre	Rol	Estado
Anibal Bravo	Desarrollador Junior	Activo
Darwin Argoti	Jefe de Proyecto	Inactivo
Darwin Argoti	Jefe de Proyecto	Activo
Edison Lomas	Desarrollador Senior	Activo

## Diagrama de Caso de Uso:



## Prerrequisitos:

1. Usuario registrado y autorizado como administrador del sistema
2. Usuario debidamente autenticado en el sistema.

3. Tener configurado correctamente el servidor LDAP.

### Flujo Principal – Buscar Usuarios:

1. Se ingresa a la pantalla de administración de usuarios mediante el menú de la aplicación.
2. Se presenta una pantalla con los siguientes filtros de búsqueda:
  - Rol
    - Editable en lista desplegable
    - Catalogo
  - Nombre
    - Editable en caja de texto
    - Texto

Juntamente con un botón **Buscar**, una grilla de resultados debidamente paginada, y el botón **Sincronizar**.

3. Al clic en el botón **Buscar** se actualiza la grilla con los registros que cumplan los criterios de búsqueda.

### Flujo Alternativo – Sincronizar Usuario:

1. Se ingresa a la pantalla de administración de catálogos mediante el menú de la aplicación.
2. Se presenta una pantalla con los siguientes filtros de búsqueda:
  - Tipo de Catalogo
    - Editable en lista desplegable
    - Catalogo
  - Nombre Catalogo

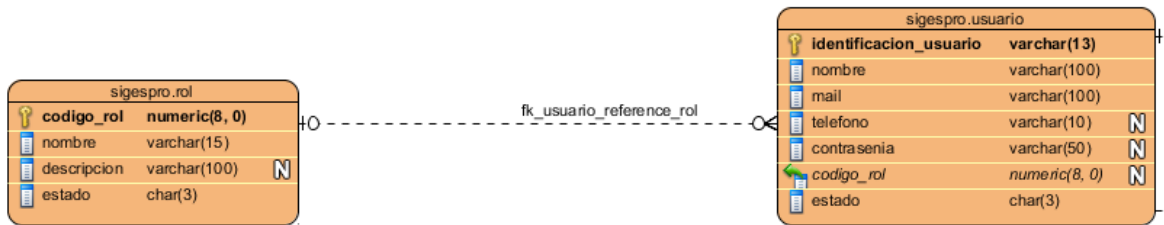


- Editable en caja de texto
- Texto de máximo 50 caracteres.

Juntamente con un botón **Buscar**, una grilla de resultados debidamente paginada, y el botón **Sincronizar**.

3. Al clic en el botón **Sincronizar**, se sincronizan los usuarios desde el servidor LDAP, actualizando a los existentes, creando los nuevos usuarios e inactivando a los usuarios ya no existentes. . Si ocurriese algún error se presentara el mensaje correspondiente.

### Tablas que intervienen en el caso de uso.



### 3.2.1.4 Administración de Instituciones

#### Historial de Revisiones:

Fecha	Versión	Descripción	Autor
17-01-2011	1.0	Descripción inicial del caso de uso	Edison Lomas
18-01-2011	1.1	Modificación y actualización del caso de uso a petición del usuario funcional.	Edison Lomas

## Prototipo de las pantallas

Pantalla principal de la administración de instituciones.

The screenshot shows a web interface for managing institutions. At the top, there is a search bar titled "Buscar Institución" with a dropdown menu for "Tipo de Institución" (currently showing "Seleccione...") and a text input for "Nombre Institución". Below the search bar is a "Buscar" button. Underneath is a table titled "Instituciones" with columns for "Nombre Institución", "Descripción", "Tipo de Institución", and "Opciones". The table contains one entry: "AyE Sistemas", "Empresa dedicada al desarrollo de sistemas informaticos", and "Privada". Below the table is a "Nuevo" button.

Nombre Institución	Descripción	Tipo de Institución	Opciones
AyE Sistemas	Empresa dedicada al desarrollo de sistemas informaticos	Privada	 

Pantalla de creación de una nueva institución.

The screenshot shows a form titled "Nueva Institución" for creating a new institution. It includes the following fields: "Tipo de Institución" (dropdown menu), "RUC Institución" (text input), "Nombre Institución" (text input), "Descripción" (text area), "Dirección" (text area), "Teléfono" (text input), "Teléfono Adicional" (text input), and another "Teléfono Adicional" (text input). At the bottom, there are "Guardar" and "Cancelar" buttons, and a note: "\* Campos requeridos".

Pantalla de edición de una institución.

The screenshot shows a web form titled "Editar Institución" with a close button in the top right corner. The form contains the following fields:

- Tipo de Institución: A dropdown menu with "Privada" selected.
- RUC Institución: A text input field containing "1002803898001".
- Nombre Institución: A text input field containing "AyE Sistemas".
- Descripción: A text area containing "Empresa dedicada al desarrollo de sistemas informaticos".
- Dirección: A text area containing "Av. San Vicente y 12 de Febrero".
- Teléfono: A text input field containing "090447763".
- Teléfono Adicional: Two empty text input fields.

At the bottom left, there is a red asterisk and the text "Campos requeridos". At the bottom center, there are two buttons: "Aceptar" and "Cancelar".

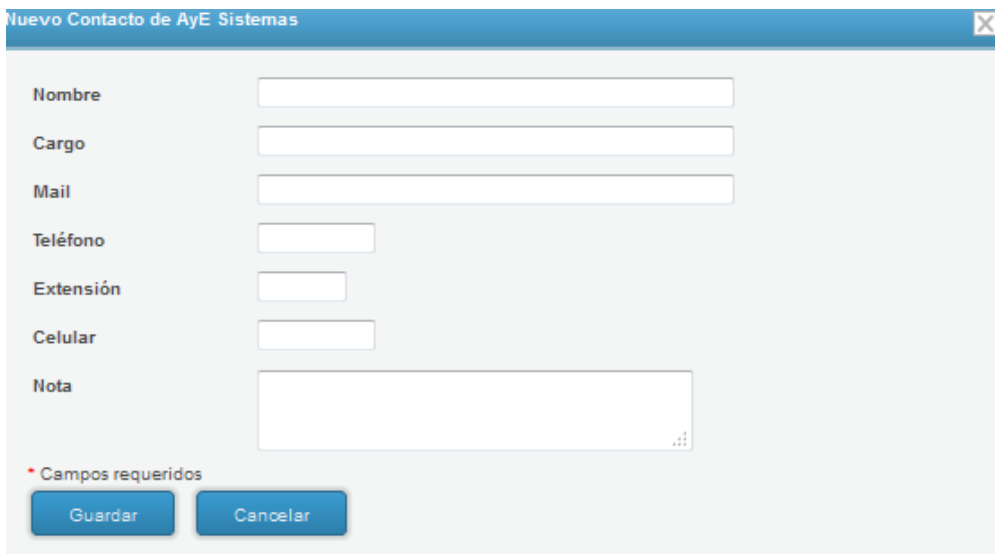
Pantalla principal de administración de contactos de la institución.

The screenshot shows a web interface for managing contacts. At the top, there is a blue header bar with the text "Buscar Contacto de AyE Sistemas" and a close button. Below this, there is a search form with two input fields: "Nombre" and "Cargo". A "Buscar" button is located below the search fields. Below the search form, there is a table titled "Contactos de AyE Sistemas". The table has three columns: "Nombre", "Cargo", and "Opciones". The table contains one row with the following data:

Nombre	Cargo	Opciones
Edison Lomas	Desarrollador	

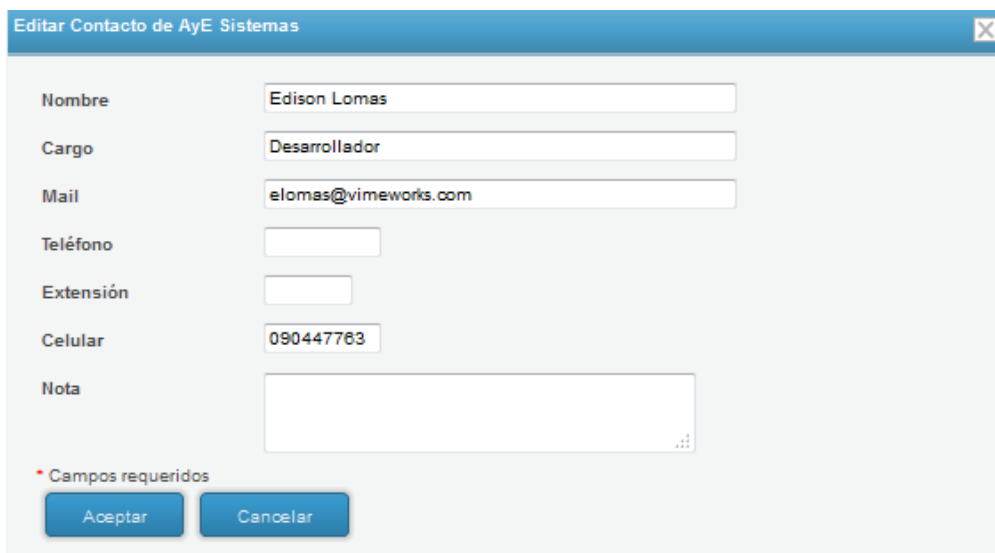
At the bottom of the interface, there are two buttons: "Nuevo" and "Regresar".

Pantalla de creación de un nuevo contacto.



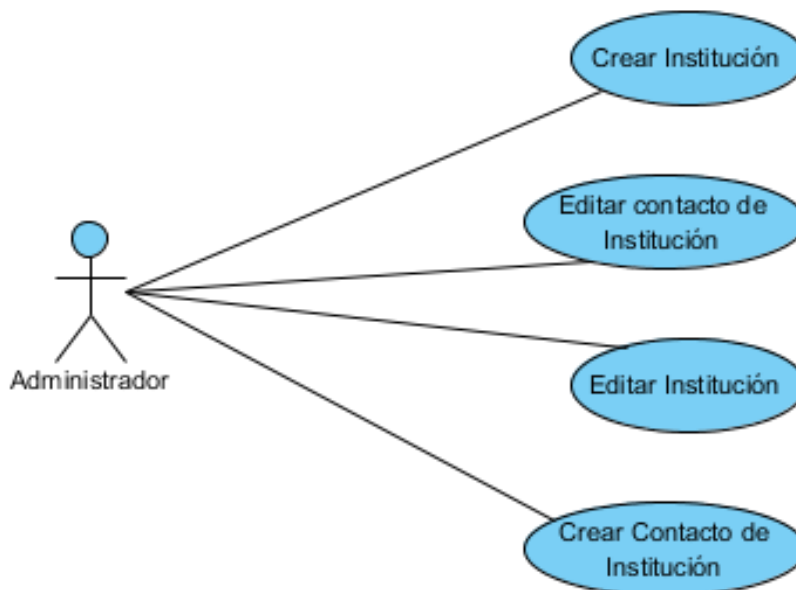
The screenshot shows a web form titled "Nuevo Contacto de AyE Sistemas". It contains several input fields for contact information: "Nombre", "Cargo", "Mail", "Teléfono", "Extensión", and "Celular". Below these is a larger text area for "Nota". At the bottom left, there is a red asterisk and the text "Campos requeridos". At the bottom center, there are two buttons: "Guardar" and "Cancelar".

Pantalla de edición de un contacto de una institución.



The screenshot shows a web form titled "Editar Contacto de AyE Sistemas". It contains several input fields with pre-filled data: "Nombre" (Edison Lomas), "Cargo" (Desarrollador), "Mail" (elomas@vimeworks.com), "Teléfono", "Extensión", and "Celular" (090447763). Below these is a larger text area for "Nota". At the bottom left, there is a red asterisk and the text "Campos requeridos". At the bottom center, there are two buttons: "Aceptar" and "Cancelar".

## Diagrama de Caso de Uso:



## Prerrequisitos:

1. Usuario registrado y autorizado como administrador del sistema
2. Usuario debidamente autenticado en el sistema.
3. Haber ingresado catálogos de tipo **Instituciones**.

## Flujo Principal – Crear Institución:

1. Se ingresará a la pantalla de administración de instituciones mediante el menú de la aplicación.
2. Se presenta una pantalla con los siguientes filtros de búsqueda:
  - Tipo de Institución
    - Editable en lista desplegable
    - Catalogo
  - Nombre Institución
    - Editable en caja de texto
    - Texto de máximo 50 caracteres

Juntamente con un botón **Buscar**, una grilla de resultados debidamente paginada, y el botón **Nuevo**.

3. Al dar clic en el botón **Nuevo** se abre una ventana de registro de información con los siguiente campos:

- Tipo de institución
  - Editable en lista desplegable
  - Catalogo
  - Obligatorio
- RUC Institución
  - Editable en caja de texto
  - Texto máximo 13 caracteres
  - Obligatorio
- Nombre Institución
  - Editable en caja de texto
  - Texto de máximo 50 caracteres.
  - Obligatorio
- Descripción
  - Editable en caja de texto
  - Texto de máximo 500 caracteres
  - No obligatorio
- Dirección
  - Editable en caja de texto
  - Texto máximo 250 caracteres
  - Obligatorio
- Teléfono
  - Editable en caja de texto
  - Máximo 9 caracteres
  - No obligatorio
- Teléfono adicional
  - Editable en caja de texto
  - Máximo 9 caracteres
  - No obligatorio
- Teléfono adicional

- Editable en caja de texto
- Máximo 9 caracteres
- No obligatorio

Juntamente con un botón **Guardar** y el botón **Cancelar**

4. Al dar clic en el botón **Guardar** se validan los datos, se guarda la información y se presenta un mensaje indicando que la institución se guardó satisfactoriamente. Si ocurriese algún error se presentara el mensaje correspondiente.
5. Al dar clic en **Cancelar** se cierra la ventana sin realizar ninguna acción.

### **Flujo Alternativo 1 – Editar Institución:**

1. Se ingresa a la pantalla de administración de instituciones mediante el menú de la aplicación.
2. Se presenta una pantalla con los siguientes filtros de búsqueda:
  - Tipo de institución
    - Editable en lista desplegable
    - Catalogo
  - Nombre institución
    - Editable en caja de texto
    - Texto de máximo 50 caracteres.

Juntamente con un botón **Buscar**, una grilla de resultados debidamente paginada, y el botón **Nuevo**.

3. Al dar clic en el botón **Buscar** se actualiza la grilla de resultado en base a los criterios de búsqueda. En la parte derecha de los registros de la grilla existen las opciones **Editar** y **Contactos**.



4. Al dar clic en la opción **Editar** se abre una ventana de edición de información con los siguiente campos:

- Tipo de institución
  - Editable en lista desplegable
  - Catalogo
  - Obligatorio
- RUC Institución
  - Editable en caja de texto
  - Texto máximo 13 caracteres
  - Obligatorio
- Nombre Institución
  - Editable en caja de texto
  - Texto de máximo 50 caracteres.
  - Obligatorio
- Descripción
  - Editable en caja de texto
  - Texto de máximo 500 caracteres
  - No obligatorio
- Dirección
  - Editable en caja de texto
  - Texto máximo 250 caracteres
  - Obligatorio
- Teléfono
  - Editable en caja de texto
  - Máximo 9 caracteres
  - No obligatorio
- Teléfono adicional
  - Editable en caja de texto
  - Máximo 9 caracteres
  - No obligatorio
- Teléfono adicional
  - Editable en caja de texto
  - Máximo 9 caracteres
  - No obligatorio

Juntamente con un botón **Aceptar** y el botón **Cancelar**.

5. Al dar clic en el botón **Aceptar** se validan los datos, se actualiza la información y se presenta un mensaje indicando que la institución se actualizo satisfactoriamente. Si ocurriese algún error se presentara el mensaje correspondiente.
6. Al dar clic en **Cancelar** se cierra la ventana sin realizar ninguna acción.

### Flujo Alternativo 2 – Crear Contacto de Institución:

1. Se ingresa a la pantalla de administración de instituciones mediante el menú de la aplicación.
2. Se presenta una pantalla con los siguientes filtros de búsqueda:
  - Tipo de institución
    - Editable en lista desplegable
    - Catalogo
  - Nombre institución
    - Editable en caja de texto
    - Texto de máximo 50 caracteres.

Juntamente con un botón **Buscar**, una grilla de resultados debidamente paginada, y el botón **Nuevo**.

3. Al dar clic en el botón **Buscar** se actualiza la grilla de resultado en base a los criterios de búsqueda. En la parte derecha de los registros de la grilla existen las opciones **Editar** y **Contactos**.
4. Al dar clic en **Contactos** se presenta una ventana con los siguientes filtros de búsqueda
  - Nombre
    - Editable en caja de texto

- Máximo 200 caracteres
- Cargo
  - Editable en caja de texto
  - Máximo 200 caracteres

Juntamente con un botón **Buscar**, una grilla de resultados debidamente paginada, el botón **Nuevo** y el botón **Regresar**.

5. Al dar clic en el botón **Nuevo** se presenta una ventana de registro de información con los siguientes campos:

- Nombre
  - Editable en caja de texto
  - Máximo 200 caracteres
  - Obligatorio
- Cargo
  - Editable en caja de texto
  - Máximo 200 caracteres
  - Obligatorio
- Mail
  - Editable en caja de texto
  - Máximo 100 caracteres
  - Obligatorio
- Teléfono
  - Editable en caja de texto
  - Máximo 9 caracteres
  - No obligatorio
- Extensión
  - Editable en caja de texto
  - Máximo 5 caracteres
  - No obligatorio
- Celular
  - Editable en caja de texto
  - Máximo 9 caracteres

- No obligatorio
- Nota
  - Editable en caja de texto
  - Máximo 500 caracteres
  - No obligatorio

Juntamente con un botón **Guardar** y el botón **Cancelar**

6. Al dar clic en el botón **Guardar** se validan los datos, se guarda la información y se presenta un mensaje indicando que el contacto de la institución se guardó satisfactoriamente. Si ocurriese algún error se presentara el mensaje correspondiente.
7. Al dar clic en Cancelar se cierra la ventana sin realizar ninguna acción.

### Flujo Alternativo 3 – Editar contacto de Institución:

1. Se ingresa a la pantalla de administración de instituciones mediante el menú de la aplicación.
2. Se presenta una pantalla con los siguientes filtros de búsqueda:
  - Tipo de institución
    - Editable en lista desplegable
    - Catalogo
  - Nombre institución
    - Editable en caja de texto
    - Texto de máximo 50 caracteres.

Juntamente con un botón **Buscar**, una grilla de resultados debidamente paginada, el botón **Nuevo** y el botón **Regresar**.

3. Al dar clic en el botón **Buscar** se actualiza la grilla de resultado en base a los criterios de búsqueda. En la parte derecha de los registros de la grilla existen las opciones **Editar** y **Contactos**.
4. Al dar clic en la opción **Contacto** se presenta una pantalla con los siguientes filtros de búsqueda.
  - Nombre
    - Editable en caja de texto
    - Texto de máximo 200 caracteres
  - Cargo
    - Editable en caja de texto
    - Texto de máximo 200 caracteres
5. Al dar clic en el botón **Buscar** se actualiza la grilla de resultado. En la parte derecha de los registros de la grilla existen las opción **Editar**.
6. Al dar clic en la opción **Editar** se presenta una pantalla de edición con los siguientes campos.
  - Nombre
    - Editable en caja de texto
    - Máximo 200 caracteres
    - Obligatorio
  - Cargo
    - Editable en caja de texto
    - Máximo 200 caracteres
    - Obligatorio
  - Mail
    - Editable en caja de texto
    - Máximo 100 caracteres
    - Obligatorio
  - Teléfono
    - Editable en caja de texto
    - Máximo 9 caracteres
    - No obligatorio

- Extensión
  - Editable en caja de texto
  - Máximo 5 caracteres
  - No obligatorio
- Celular
  - Editable en caja de texto
  - Máximo 9 caracteres
  - No obligatorio
- Nota
  - Editable en caja de texto
  - Máximo 500 caracteres
  - No obligatorio

Juntamente con un botón **Guardar** y el botón **Cancelar**

7. Al dar clic en el botón **Guardar** se validan los datos, se guarda la información y se presenta un mensaje indicando que el contacto de la institución se actualizado satisfactoriamente. Si ocurriese algún error se presentara el mensaje correspondiente.
8. Al dar clic en **Cancelar** se cierra la ventana sin realizar ninguna acción.

#### **Flujo Alternativo 4 – Regresar de contacto de Institución a Instituciones:**

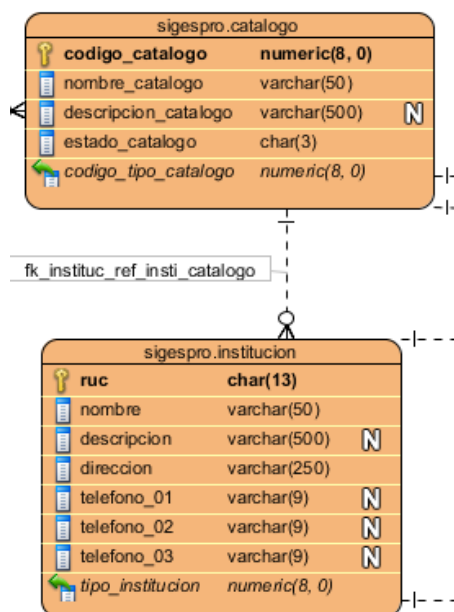
1. Se ingresa a la pantalla de administración de instituciones mediante el menú de la aplicación.
2. Se presenta una pantalla con los siguientes filtros de búsqueda:
  - Tipo de institución
    - Editable en lista desplegable
    - Catalogo
  - Nombre institución

- Editable en caja de texto
- Texto de máximo 50 caracteres.

Juntamente con un botón **Buscar**, una grilla de resultados debidamente paginada, el botón **Nuevo** y el botón **Regresar**.

3. Al dar clic en el botón **Regresar** se regresa a la pantalla de administración de instituciones.

### Tablas que intervienen en el caso de uso.



### 3.2.1.5 Administración de Proyectos

#### Historial de Revisiones:





Fecha	Versión	Descripción	Autor
24-01-2011	1.0	Descripción inicial del caso de uso	Edison Lomas
25-01-2011	1.1	Modificación y actualización del caso de uso a petición del usuario funcional.	Edison Lomas
26-01-2011	1.2	Modificación y actualización del caso de uso a petición del usuario funcional.	Edison Lomas
27-01-2011	1.3	Modificación y actualización del caso de uso a petición del usuario funcional.	Edison Lomas



## Prototipo de las pantallas

Pantalla principal de la administración de proyectos.

The screenshot shows a web application interface for project management. At the top, there is a search section titled "Buscar Proyecto" with a text input for "Nombre" and three dropdown menus for "Estado del Proyecto", "Institución", and "Jefe de Proyecto". A "Buscar" button is located below these fields. Below the search section is a table titled "Proyectos" with columns for "Nombre", "Descripción", "Jefe de Proyecto", "Institución", "Estado del Proyecto", and "Opciones". The table contains one row with the following data: "SIGESPro", "Sistema de Gestión y Seguimiento de Proyectos", "Darwin Argoti", "AyE Sistemas", "Por iniciar", and a set of icons for actions. A "Nuevo" button is located below the table.

Nombre	Descripción	Jefe de Proyecto	Institución	Estado del Proyecto	Opciones
SIGESPro	Sistema de Gestión y Seguimiento de Proyectos	Darwin Argoti	AyE Sistemas	Por iniciar	   

Pantalla de creación de un nuevo proyecto.

The screenshot shows a form titled "Nuevo Proyecto" for creating a new project. It includes the following fields: "Nombre" (text input), "Descripción" (text area), "Estado del Proyecto" (dropdown menu), "Institución" (dropdown menu), "Jefe de Proyecto" (dropdown menu), "Fecha de inicio" (calendar icon), and "Fecha de finalización" (calendar icon). A red asterisk indicates that the "Nombre", "Estado del Proyecto", "Institución", and "Jefe de Proyecto" fields are required. At the bottom, there are "Guardar" and "Cancelar" buttons.

Pantalla de edición de un proyecto.

The screenshot shows a window titled "Editar Proyecto" with a close button in the top right corner. The form contains the following fields:

- Nombre:** A text input field containing "SIGESPro".
- Descripción:** A text area containing "Sistema de Gestión y Seguimiento de Proyectos".
- Estado del Proyecto:** A dropdown menu with "Por iniciar" selected.
- Institución:** A dropdown menu with "AyE Sistemas" selected.
- Jefe de Proyecto:** A dropdown menu with "Darwin Argoti" selected.
- Fecha de inicio:** A date input field with "21/11/2011" and a calendar icon.
- Fecha de finalización:** A date input field with "22/11/2011" and a calendar icon.

Below the fields, there is a red asterisk and the text "Campos requeridos". At the bottom, there are two buttons: "Aceptar" and "Cancelar".

Pantalla principal de asignación de desarrolladores.

The screenshot shows a window titled "Buscar Usuario" with a close button in the top right corner. The form contains the following fields:

- Proyecto:** A text input field containing "SIGESPro".
- Rol:** A dropdown menu with "Selecciones..." selected.
- Nombre:** A text input field.

Below the fields, there is a "Buscar" button. Underneath, there is a section titled "Usuarios" containing a table with the following data:

Nombre	Rol	Estado	Opciones
Darwin Argoti	Jefe de Proyecto	Activo	
Edison Lomas	Desarrollador Senior	Activo	

At the bottom of the "Usuarios" section, there are two buttons: "Agregar" and "Regresar".

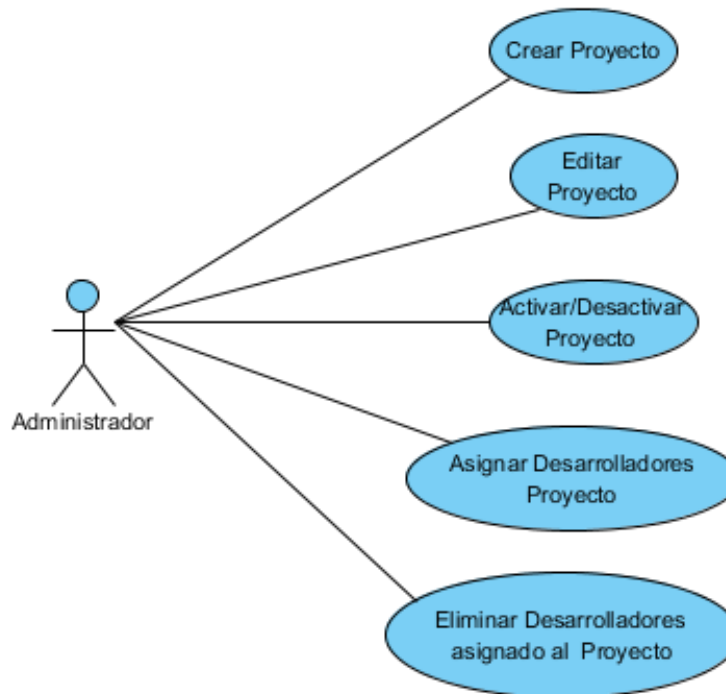
Pantalla para agregar un nuevo recurso.

The screenshot shows a window titled "Nuevo usuario para SIGESPro" with a close button in the top right corner. The form contains the following fields:

- Nombre:** A dropdown menu with "Selecciones..." selected.

Below the field, there is a red asterisk and the text "Campos requeridos". At the bottom, there are two buttons: "Guardar" and "Cancelar".

## Diagrama de Caso de Uso:



### Prerrequisitos:

1. Usuario registrado y autorizado como administrador del sistema
2. Usuario debidamente autenticado en el sistema.
3. Haber registrado instituciones en el sistema.

### Flujo Principal – Crear Proyecto:

1. Se ingresa a la pantalla de administración de proyectos mediante el menú de la aplicación.
2. Se presenta una pantalla con los siguientes filtros de búsqueda:
  - Nombre
    - Editable en caja de texto
    - Texto de máximo 50 caracteres
  - Estado de proyecto
    - Editable en lista desplegable

- Catalogo
- Institución
  - Editable en lista desplegable
  - Catalogo (Referencial a la tabla instituciones)
- Jefe de Proyecto
  - Editable en lista desplegable
  - Catalogo (Referencia a los usuarios con el rol Jefe de Proyecto)

Juntamente con un botón **Buscar**, una grilla de resultados debidamente paginada, y el botón **Nuevo**.

3. Al clic en el botón **Nuevo** se abre una ventana de registro de información con los siguiente campos:

- Nombre
  - Editable en caja de texto
  - Texto de máximo 50 caracteres.
  - Obligatorio
- Descripción
  - Editable en caja de texto
  - Texto de máximo 500 caracteres.
  - Obligatorio
- Estado de Proyecto
  - Editable en lista desplegable
  - Catalogo
  - Obligatorio
- Institución
  - Editable en lista desplegable
  - Catalogo (Referencia a la tabla de instituciones)
  - Obligatorio
- Jefe de proyecto
  - Editable en lista desplegable
  - Catalogo (Referencia a los usuarios con el rol Jefe de Proyecto)
  - No obligatorio
- Fecha de inicio

- Editable en componente de fecha
  - Fecha
  - No obligatorio
- Fecha de finalización
  - Editable en componente de fecha
  - Fecha
  - No obligatorio

Juntamente con un botón Guardar y el botón Cancelar

4. Al dar clic en el botón **Guardar** se validan los datos, se guarda la información y se presenta un mensaje indicando que el Proyecto se guardó satisfactoriamente. Si ocurriese algún error se presentara el mensaje correspondiente.
5. Al dar clic en **Cancelar** se cierra la ventana sin realizar ninguna acción.

### Flujo Alternativo 1 – Editar Proyecto:

1. Se ingresa a la pantalla de administración de Proyectos mediante el menú de la aplicación.
2. Se presenta un pantalla con los siguientes filtros de búsqueda:
  - Nombre
    - Editable en caja de texto
    - Texto de máximo 50 caracteres
  - Estado de proyecto
    - Editable en lista desplegable
    - Catalogo
  - Institución
    - Editable en lista desplegable
    - Catalogo (Referencial a la tabla instituciones)
  - Jefe de Proyecto

- Editable en lista desplegable
- Catalogo (Referencia a los usuarios con el rol Jefe de Proyecto)

Juntamente con un botón **Buscar**, una grilla de resultados debidamente paginada, y el botón **Nuevo**.

3. Al dar clic en el botón **Buscar** se actualiza la grilla de resultado en base a los criterios de búsqueda. En la parte derecha de los registros de la grilla existen las opciones **Editar, Activar/Desactivar y Desarrolladores**
4. Al dar clic en la opción **Editar** se abre una ventana de edición de información con los siguiente campos:
  - Nombre
    - Editable en caja de texto
    - Texto de máximo 50 caracteres.
    - Obligatorio
  - Descripción
    - Editable en caja de texto
    - Texto de máximo 500 caracteres.
    - Obligatorio
  - Estado de Proyecto
    - Editable en lista desplegable
    - Catalogo
    - Obligatorio
  - Institución
    - Editable en lista desplegable
    - Catalogo (Referencia a la tabla de instituciones)
    - Obligatorio
  - Jefe de proyecto
    - Editable en lista desplegable
    - Catalogo (Referencia a los usuarios con el rol Jefe de Proyecto)
    - No obligatorio
  - Fecha de inicio
    - Editable en componente de fecha

- Fecha
- No obligatorio
- Fecha de finalización
  - Editable en componente de fecha
  - Fecha
  - No obligatorio

Juntamente con un botón **Aceptar** y el botón **Cancelar**.

6. Al dar clic en el botón **Aceptar** se validan los datos, se actualiza la información y se presenta un mensaje indicando que el Proyecto se actualizo satisfactoriamente. Si ocurriese algún error se presentara el mensaje correspondiente.
5. Al dar clic en **Cancelar** se cierra la ventana sin realizar ninguna acción.

### **Flujo Alternativo 2 – Activar/Desactivar Proyecto:**

1. Se ingresa a la pantalla de administración de Proyectos mediante el menú de la aplicación.
2. Se presenta una pantalla con los siguientes filtros de búsqueda:
  - Nombre
    - Editable en caja de texto
    - Texto de máximo 50 caracteres
  - Estado de proyecto
    - Editable en lista desplegable
    - Catalogo
  - Institución
    - Editable en lista desplegable
    - Catalogo (Referencial a la tabla instituciones)
  - Jefe de Proyecto
    - Editable en lista desplegable

- Catálogo (Referencia a los usuarios con el rol Jefe de Proyecto)

Juntamente con un botón **Buscar**, una grilla de resultados debidamente paginada, y el botón nuevo.

3. Al dar clic en el botón **Buscar** se actualiza la grilla de resultado en base a los criterios de búsqueda. En la parte derecha de los registros de la grilla existen la opción **Editar, Activar/Desactivar y Desarrolladores**.
4. Si el proyecto se encuentra activo se presenta la opción **Desactivar**, caso contrario se presenta la opción **Activar**.
7. Al dar clic en la opción correspondiente el proyecto se activa o se desactiva según corresponda y se presenta un mensaje indicando que el proyecto se actualizo correctamente. Si ocurriese algún error se presentara el mensaje correspondiente.

### Flujo Alternativo 3 – Asignar Desarrolladores Proyecto:

1. Se ingresa a la pantalla de administración de Proyectos mediante el menú de la aplicación.
2. Se presenta un pantalla con los siguientes filtros de búsqueda:
  - Nombre
    - Editable en caja de texto
    - Texto de máximo 50 caracteres
  - Estado de proyecto
    - Editable en lista desplegable
    - Catálogo
  - Institución
    - Editable en lista desplegable
    - Catálogo (Referencial a la tabla instituciones)
  - Jefe de Proyecto



- Editable en lista desplegable
- Catalogo (Referencia a los usuarios con el rol Jefe de Proyecto)

Juntamente con un botón **Buscar**, una grilla de resultados debidamente paginada, y el botón nuevo.

3. Al dar clic en el botón **Buscar** se actualiza la grilla de resultado. En la parte derecha de los registros de la grilla existen las opciones **Editar**, **Activar/Desactivar** y **Desarrolladores**.
4. Al dar clic en la opción **Desarrolladores**, se presenta una página de administración de asignación de desarrolladores a proyectos con los siguientes filtros de búsqueda.
  - Proyecto
    - No editable
  - Rol
    - Editable en lista desplegable
    - Catalogo (Referencia a lo tabla Rol)

Juntamente con un botón **Buscar**, una grilla de resultados debidamente paginada, y el botón **Agregar** y el botón **Regresar**.

5. Al dar clic en **Agregar** se nos presenta una ventana con los siguientes campos.
  - Nombre
    - Editable en lista desplegable
    - Catalogo (Referencia a la tabla usuarios)
    - Obligatorio

Juntamente con el botón **Guardar** y **Cancelar**.

6. Al dar clic en el botón **Guardar** se validan los datos, se asigna el usuario y se presenta un mensaje indicando que el usuario fue asignado correctamente. Si ocurriese algún error se presentara el mensaje correspondiente.
7. Al dar clic en el botón **Cancelar** se cierra la venta sin realizar ninguna acción.

## Flujo Alternativo 4 – Eliminar Desarrolladores asignado al Proyecto:

1. Se ingresará a la pantalla de administración de Proyectos mediante el menú de la aplicación.
2. Se presenta una pantalla con los siguientes filtros de búsqueda:
  - Nombre
    - Editable en caja de texto
    - Texto de máximo 50 caracteres
  - Estado de proyecto
    - Editable en lista desplegable
    - Catalogo
  - Institución
    - Editable en lista desplegable
    - Catalogo (Referencial a la tabla instituciones)
  - Jefe de Proyecto
    - Editable en lista desplegable
    - Catalogo (Referencia a los usuarios con el rol Jefe de Proyecto)

Juntamente con un botón **Buscar**, una grilla de resultados debidamente paginada, y el botón **Agregar** y **Regresar**.

3. Al dar clic en el botón **Buscar** se actualiza la grilla de resultados en base a los criterios de búsqueda. En la parte derecha de los registros de la grilla existen las opciones **Editar**, **Activar/Desactivar** y **Desarrolladores**.
4. Al dar clic en la opción **Desarrolladores**, se presenta una página de administración de asignación de desarrolladores a proyectos con los siguientes filtros de búsqueda.
  - Proyecto
    - No editable
  - Rol
    - Editable en lista desplegable
    - Catalogo (Referencia a la tabla Rol)

Juntamente con un botón **Buscar**, una grilla de resultados debidamente paginada, y el botón **Agregar** y el botón **Regresar**.

5. Al dar clic en **Buscar** se actualiza la grilla de resultados en base a los criterios de búsqueda. En la parte derecha de los registros de la grilla existe la opción **Eliminar**.
6. Al dar clic en la opción **Eliminar** se presenta un cuadro de confirmación indicando que si se elimina un usuario asignado las tareas asignadas a dicho usuario se quedaran sin asignar, juntamente con el botón **Aceptar** y **Cancelar**
7. Al dar clic en **Aceptar** se remueve la asignación del usuario, se quita la asignación de las tareas y se presenta un cuadro de dialogo indicando que el usuario ha sido removido exitosamente. Si ocurriese algún error se presentara el mensaje correspondiente.
8. Al dar clic en el botón **Cancelar** se cierra la venta sin realizar ninguna acción.

### **Flujo Alternativo 5 – Regresar Desarrolladores asignado al Proyecto a Proyectos:**

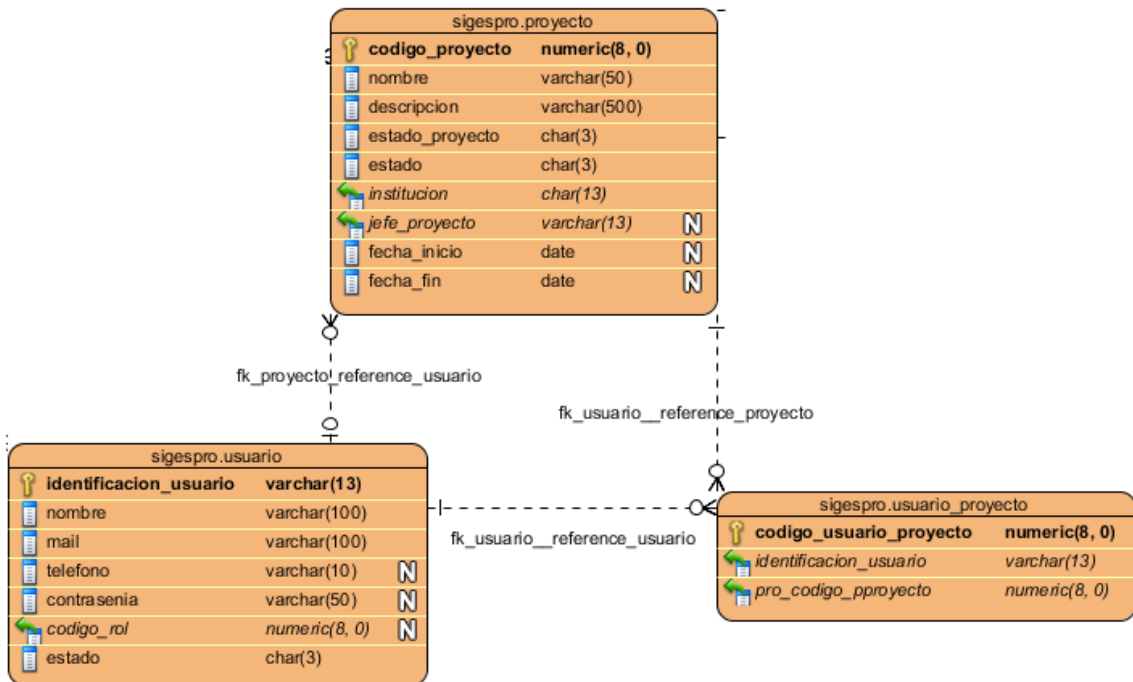
1. Se ingresar a la pantalla de administración de Proyectos mediante el menú de la aplicación.
2. Se presenta un pantalla con los siguientes filtros de búsqueda:
  - Nombre
    - Editable en caja de texto
    - Texto de máximo 50 caracteres
  - Estado de proyecto
    - Editable en lista desplegable
    - Catalogo
  - Institución
    - Editable en lista desplegable
    - Catalogo (Referencial a la tabla instituciones)

- Jefe de Proyecto
  - Editable en lista desplegable
  - Catalogo (Referencia a los usuarios con el rol Jefe de Proyecto)

Juntamente con un botón **Buscar**, una grilla de resultados debidamente paginada, y el botón **Agregar** y **Regresar**.

3. Al dar clic en el botón **Regresar**, se regresa a la pantalla de administración de proyectos

**Tablas que intervienen en el caso de uso.**



### 3.2.1.6 Autenticación de Usuarios

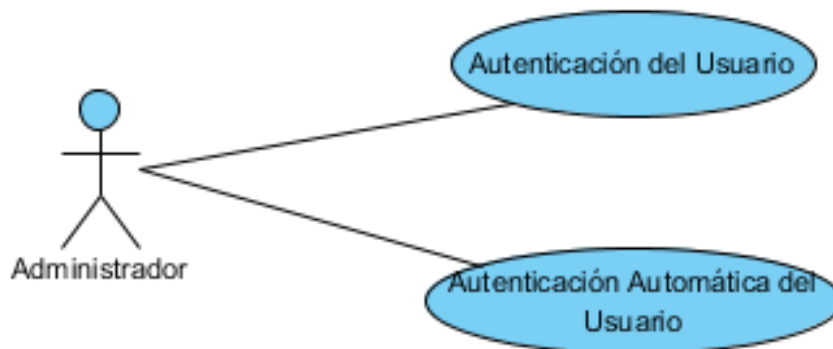
#### Historial de Revisiones:

Fecha	Versión	Descripción	Autor
07-01-2011	1.0	Descripción inicial del caso de uso	Edison Lomas

## Prototipo de pantallas

N/A

## Diagrama de Caso de Uso:



## Prerrequisitos:

1. Tener configurado correctamente el servidor LDAP.

## Flujo Principal – Autenticación del Usuario:

1. Se ingresa a la URL del sistema.
2. Se re direcciona al usuario al servidor central de autenticación.
3. Se presenta una pantalla con los siguientes campos:
  - Net ID
    - Editable en caja de texto
    - Texto
    - Obligatorio
  - Contraseña
    - Editable en caja de texto
    - Texto
    - Obligatorio

- Avisarme antes de abrir sesión en otros sitios.
  - Editable en control de selección
  - Booleano
  - N/A

Juntamente con el botón **Iniciar Sesión** y la opción **Limpiar**.

4. Al clic en el botón **Iniciar Sesión** se valida la información ingresada en el servidor LDAP. Si la información ingresada es correcta se redirige, a la pantalla desde la cual se solicitó el inicio de sesión.
5. Se la información ingresada no es correcta se presentara la información del error correspondiente.

### **Flujo Alternativo – Autenticación Automática del Usuario:**

1. Se ingresa a la URL del sistema.
2. Se re direcciona al usuario al servidor central de autenticación.
3. Se verifica que el usuario ya tiene un ticket de servicio activo, se re direcciona al usuario a la página que solicito el inicio de sesión.

### **Tablas que intervienen en el caso de uso.**

N/A

### 3.2.1.7 Proyectos Asignados

#### Historial de Revisiones:

Fecha	Versión	Descripción	Autor
07-02-2011	1.0	Descripción inicial del caso de uso	Edison Lomas
08-02-2011	1.1	Modificación y actualización del caso de uso a petición del usuario funcional.	Edison Lomas
09-02-2011	1.2	Modificación y actualización del caso de uso a petición del usuario funcional.	Edison Lomas
10-02-2011	1.3	Modificación y actualización del caso de uso a petición del usuario funcional.	Edison Lomas
13-02-2011	1.4	Modificación y actualización del caso de uso en base al proceso de actualización de estados del proyecto.	Edison Lomas
15-02-2011	1.5	Modificación y actualización del caso de uso a petición del usuario funcional.	Edison Lomas
16-02-2011	1.6	Modificación y actualización del caso de uso a petición del usuario funcional.	Edison Lomas



## Prototipo de pantallas

Página inicial de proyectos asignados.

Buscar Proyecto

Nombre

Estado del Proyecto Selecciones... ▾

Institución Selecciones... ▾

Buscar

Proyectos

Nombre	Descripción	Jefe de Proyecto	Institución	Estado del Proyecto	Opciones
SIGESPro	Sistema de Gestión y Seguimiento de Proyectos	Darwin Argoti	AyE Sistemas	Por iniciar	

Pantalla de cambio de estado de proyecto.

Nuevo Proyecto

Nombre SIGESPro

Descripción Sistema de Gestión y Seguimiento de Proyectos

Estado del Proyecto Por iniciar ▾

Institución AyE Sistemas

Jefe de Proyecto Darwin Argoti

Fecha de inicio 21/11/2011

Fecha de finalización 22/11/2011

\* Campos requeridos

Aceptar Cancelar

Pantalla de administración de tareas.

Buscar Tarea

Proyecto: SIGESPro

Nombre:

Asignada a: Selecciones... ▼

Tipo de Tarea: Selecciones... ▼

Prioridad: Selecciones... ▼

Buscar

Tareas

Nombre	Nombre	Descripción	Tiempo Propuesto(h)	Asignada a	Opciones
6	dfgfd	gdfgdfgdfg	19	Darwin Argoti	
8	fdsfds	sdfdsfs	0	Edison Lomas	
9	eeeeee	iuasd gsdfo fu asd s	0		
10	sdfsd	sdfsdfsdf	0	Darwin Argoti	
11	aaaaaaaaaaaa	ssssssssssss	0		
12	fff	dddd	0		
13	dsfas	fasfasfasdf	0		
14	yayyyyyyyyyyyyyy	vdfvdfvdfvdfv	0	Edison Lomas	
15	sdasdasdasd	asdasdasdasd	0		
16	aaaaaaaaaaaa	zcxzcxzcx	0	Darwin Argoti	

« « 1 2 » »

Nuevo Regresar

Pantalla de creación de tareas.

**Nueva Tarea**

Proyecto: SIGESPro Predecesoras

Nombre:

Descripción:

Fecha de inicio propuesta:

Tiempo Propuesto(h):

Fecha de finalización propuesta:

Asignada a:

Tipo de Tarea:

Prioridad:

\* Campos requeridos

Pantalla de tareas predecesoras.

**Predecesoras**

	Nombre	Descripción	Tiempo(h)	Asignada a
<input type="checkbox"/>	sdfsd	sdfsdfdsf	4	Darwin Argoti
<input type="checkbox"/>	aaaaaaaaaaaa	zozxozxozx	32	Darwin Argoti
<input type="checkbox"/>	AAAAAAAAAAAA	AAAAAAAAAAAA	4	
<input type="checkbox"/>	aaaaaaaaaaaa	ssssssssssss	3	
<input type="checkbox"/>	fff	dddd	4	
<input type="checkbox"/>	dsfas	fasfasdf	4	
<input type="checkbox"/>	sdasdasdasd	asdasdasdasd	23	
<input type="checkbox"/>	asdfsadf	sadfsadfsadfsadf	23	Darwin Argoti
<input type="checkbox"/>	sdfasd	fsafdsdafasdf	23	
<input type="checkbox"/>	dsfas	fasdfasdfasd	23	Darwin Argoti

«« « 1 2 » »»

\* Campos requeridos

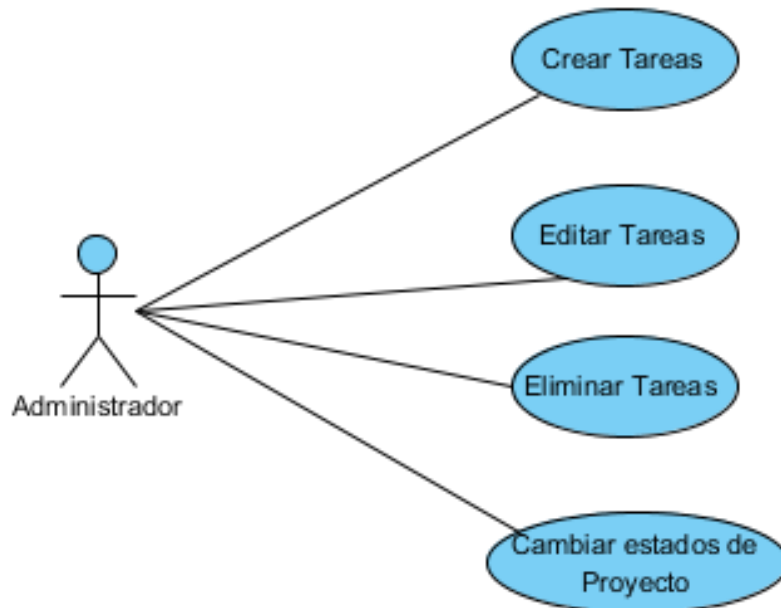
Pantalla de edición de tareas.

The screenshot shows a web form titled "Editar Tarea" with the following fields and values:

- Proyecto: SIGESPro
- Nombre: aaaaaaaaaaaaa
- Descripción: zcxzxczxczxc
- Fecha de inicio propuesta: 16/02/2012
- Tiempo Propuesto(h): 0
- Fecha de finalización propuesta: 16/02/2012
- Asignada a: Darwin Argoti
- Tipo de Tarea: Requerimiento
- Prioridad: dsfgdfg

Buttons: "Predecesoras", "Aceptar", "Cancelar".

Diagrama de Caso de Uso:



## Prerrequisitos:

1. Usuario registrado y autorizado como jefe de proyecto.
2. Usuario debidamente autenticado en el sistema.
3. Tener proyectos asignados.

## Flujo Principal – Crear Tareas:

1. Se ingresa a la pantalla de proyectos asignados mediante el menú de la aplicación.
2. Se presenta una pantalla con los siguientes filtros de búsqueda:
  - Nombre
    - Editable en caja de texto
    - Texto de máximo 50 caracteres
  - Estado de proyecto
    - Editable en lista desplegable
    - Catalogo
  - Institución
    - Editable en lista desplegable
    - Catalogo (Referencial a la tabla instituciones)

Juntamente con un botón **Buscar** y una grilla de resultados debidamente paginada.

3. Al clic en el botón **Buscar** se actualiza la grilla de resultados en base a los criterios de búsqueda. En la parte derecha de cada registro de la grilla existen las opciones **Editar y Tareas**.
4. Al dar clic en la opción **Tareas** se presenta una pantalla con los siguientes filtros de búsqueda:
  - Proyecto
    - Caja de texto de solo lectura
    - Texto
  - Nombre

- Editable en caja de texto
  - Texto de máximo 50 caracteres.
- Asignado a
  - Editable en lista desplegable
  - Catalogo (Referencia a la asignación de desarrolladores)
- Tipo de tarea
  - Editable en lista desplegable
  - Catalogo
- Prioridad
  - Editable en lista desplegable
  - Catalogo

Juntamente con un botón **Buscar**, una grilla debidamente paginada, el botón **Nuevo** y el botón **Regresar**

5. Al dar clic en el botón **Nuevo** se presenta una pantalla de ingreso de información con los siguientes campos:

- Proyecto
  - Caja de texto de solo lectura
  - Texto
- Nombre
  - Editable en caja de texto
  - Texto de 500 caracteres
  - Obligatorio
- Descripción
  - Editable en caja de texto
  - Texto de 5000 caracteres
  - Obligatorio
- Fecha de inicio propuesta
  - Editable en componente de fecha.
  - Fecha
  - Obligatorio
- Tiempo Propuesto(h)
  - Editable en caja de texto

- Entero de 3 caracteres
  - Obligatorio
- Fecha de finalización propuesta
  - Campo de texto de solo lectura
  - Fecha
  - Calculado
- Asignada a
  - Editable en lista desplegable
  - Catalogo (Referencia a la asignación de desarrolladores)
  - No obligatorio
- Tipo de tarea
  - Editable en lista desplegable
  - Catalogo
  - Obligatorio
- Prioridad
  - Editable en lista desplegable
  - Catalogo
  - Obligatorio

Juntamente con el botón **Predecesoras, Guardar y Cancelar**

6. Al colocar el tiempo propuesto automáticamente se calculara la fecha de finalización.
7. Al dar clic en el botón **Predecesoras** podemos registrar cuales son las tareas que preceden a nuestra tarea, para lo cual se abre una pantalla con el listado de las tareas para seleccionar las predecesoras juntamente con los botones **Aceptar** y **Cancelar**.
8. Al dar clic en el botón **Aceptar** se cierra las ventana y se guardan las tareas predecesoras.
9. Al dar clic en el botón **Cancelar** se cierra la ventana sin realizar ninguna acción.

10. Al dar clic en el botón **Guardar** se validan los datos, se guarda la información y se presenta un mensaje indicando que la Tarea se guardó satisfactoriamente. Si ocurriese algún error se presentara el mensaje correspondiente.
11. Al dar clic en **Cancelar** se cierra la ventana sin realizar ninguna acción.

### Flujo Alternativo 1 – Editar Tareas:

1. Se ingresa a la pantalla de proyectos asignados mediante el menú de la aplicación.
2. Se presenta una pantalla con los siguientes filtros de búsqueda:
  - Nombre
    - Editable en caja de texto
    - Texto de máximo 50 caracteres
  - Estado de proyecto
    - Editable en lista desplegable
    - Catalogo
  - Institución
    - Editable en lista desplegable
    - Catalogo (Referencial a la tabla instituciones)

Juntamente con un botón **Buscar** y una grilla de resultados debidamente paginada.

3. Al clic en el botón **Buscar** se actualiza la grilla de resultados en base a los criterios de búsqueda. En la parte derecha de cada registro de la grilla existen las opciones **Editar y Tareas**.
4. Al dar clic en la opción **Tareas** se presenta una pantalla con los siguientes filtros de búsqueda:
  - Proyecto
    - Caja de texto de solo lectura
    - Texto



- Nombre
  - Editable en caja de texto
  - Texto de máximo 50 caracteres.
- Asignado a
  - Editable en lista desplegable
  - Catalogo (Referencia a la asignación de desarrolladores)
- Tipo de tarea
  - Editable en lista desplegable
  - Catalogo
- Prioridad
  - Editable en lista desplegable
  - Catalogo

Juntamente con un botón **Buscar**, una grilla debidamente paginada, el botón **Nuevo** y el botón **Regresar**

5. Al dar clic en el botón **Buscar** se actualiza la grilla de resultados en base a los criterios de búsqueda. Junto a cada registro de la grilla se presentan las opciones **Editar** y **Eliminar**.
6. Al dar clic en la opción **Editar** se presenta una pantalla de edición de información con los siguientes campos:
  - Proyecto
    - Caja de texto de solo lectura
    - Texto
  - Nombre
    - Editable en caja de texto
    - Texto de 500 caracteres
    - Obligatorio
  - Descripción
    - Editable en caja de texto
    - Texto de 5000 caracteres
    - Obligatorio
  - Fecha de inicio propuesta

- Editable en componente de fecha.
  - Fecha
  - Obligatorio
- Tiempo Propuesto(h)
  - Editable en caja de texto
  - Entero de 3 caracteres
  - Obligatorio
- Fecha de finalización propuesta
  - Campo de texto de solo lectura
  - Fecha
  - Calculado
- Asignada a
  - Editable en lista desplegable
  - Catalogo (Referencia a la asignación de desarrolladores)
  - No obligatorio
- Tipo de tarea
  - Editable en lista desplegable
  - Catalogo
  - Obligatorio
- Prioridad
  - Editable en lista desplegable
  - Catalogo
  - Obligatorio

Juntamente con el botón **Predecesoras, Guardar y Cancelar**

7. Al dar clic en el botón **Predecesoras** podemos editar las tareas predecesoras de nuestra tarea, para lo cual se abre una pantalla con el listado de las tareas para seleccionar las predecesoras juntamente con los botones **Aceptar** y **Cancelar**.
8. Al dar clic en el botón **Aceptar** se cierra las ventana y se guardan las tareas predecesoras.
9. Al dar clic en el botón **Cancelar** se cierra la ventana sin realizar ninguna acción.

10. Al dar clic en el botón **Guardar** se validan los datos, se guarda la información y se presenta un mensaje indicando que la Tarea se actualizo satisfactoriamente. Si ocurriese algún error se presentara el mensaje correspondiente.
11. Al dar clic en **Cancelar** se cierra la ventana sin realizar ninguna acción.

### Flujo Alternativo 2 – Eliminar Tareas:

1. Se ingresa a la pantalla de proyectos asignados mediante el menú de la aplicación.
2. Se presenta un pantalla con los siguientes filtros de búsqueda:
  - Nombre
    - Editable en caja de texto
    - Texto de máximo 50 caracteres
  - Estado de proyecto
    - Editable en lista desplegable
    - Catalogo
  - Institución
    - Editable en lista desplegable
    - Catalogo (Referencial a la tabla instituciones)

Juntamente con un botón **Buscar** y una grilla de resultados debidamente paginada.

3. Al clic en el botón **Buscar** se actualiza la grilla de resultados en base a los criterios de búsqueda. En la parte derecha de cada registro de la grilla existen las opciones **Editar y Tareas**.
4. Al dar clic en la opción **Tareas** se presentan una pantalla con los siguientes filtros de búsqueda:
  - Proyecto
    - Caja de texto de solo lectura

- Texto
- Nombre
  - Editable en caja de texto
  - Texto de máximo 50 caracteres.
- Asignado a
  - Editable en lista desplegable
  - Catalogo (Referencia a la asignación de desarrolladores)
- Tipo de tarea
  - Editable en lista desplegable
  - Catalogo
- Prioridad
  - Editable en lista desplegable
  - Catalogo

Juntamente con un botón **Buscar**, una grilla debidamente paginada, el botón **Nuevo** y el botón **Regresar**

5. Al dar clic en el botón **Buscar** se actualiza la grilla de resultados en base a los criterios de búsqueda. Junto a cada registro de la grilla se presentan las opciones **Editar** y **Eliminar**.
6. Al dar clic en la opción **Eliminar** se presenta un cuadro de confirmación de eliminación con los botones **Aceptar** y **Cancelar**.
7. Al dar clic en el botón Aceptar se procede a eliminar la tarea.
8. Al dar clic en el botón Cancelar se cierra la ventana sin realizar ninguna acción.

### **Flujo Alternativo 3 – Cambiar estados de Proyecto:**

1. Se ingresa a la pantalla de proyectos asignados mediante el menú de la aplicación.

2. Se presenta un pantalla con los siguientes filtros de búsqueda:

- Nombre
  - Editable en caja de texto
  - Texto de máximo 50 caracteres
- Estado de proyecto
  - Editable en lista desplegable
  - Catalogo
- Institución
  - Editable en lista desplegable
  - Catalogo (Referencial a la tabla instituciones)

Juntamente con un botón **Buscar** y una grilla de resultados debidamente paginada.

3. Al clic en el botón **Buscar** se actualiza la grilla de resultados en base a los criterios de búsqueda. En la parte derecha de cada registro de la grilla existen las opciones **Editar** y **Tareas**.

4. Al dar clic en la opción **Editar** se abre una ventana de edición de información con los siguiente campos:

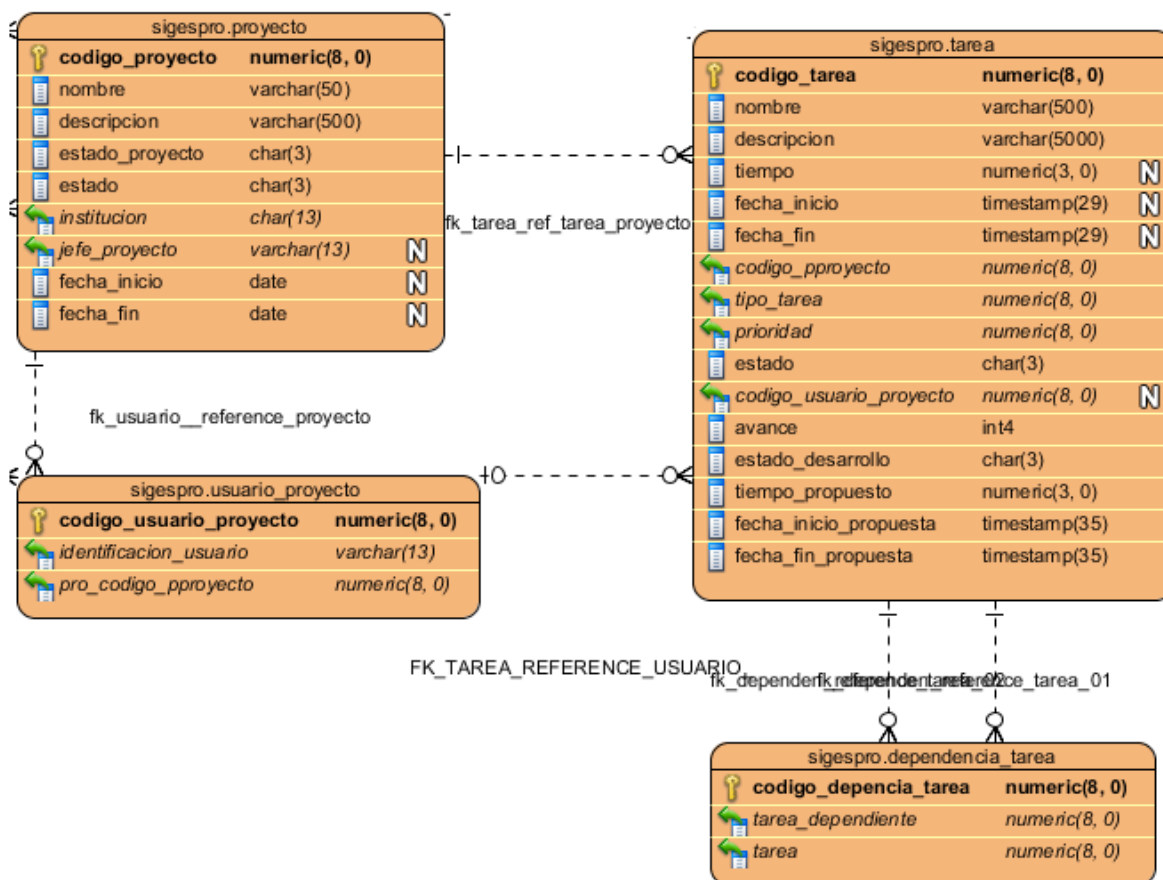
- Nombre
  - Caja de texto de solo lectura
- Descripción
  - Caja de texto de solo lectura
- Estado de Proyecto
  - Editable en lista desplegable
  - Catalogo
  - Obligatorio
- Institución
  - Caja de texto de solo lectura
- Jefe de proyecto
  - Caja de texto de solo lectura
- Fecha de inicio
  - Caja de texto de solo lectura
- Fecha de finalización

- Caja de texto de solo lectura.

Juntamente con un botón **Aceptar** y el botón **Cancelar**.

- Al dar clic en el botón **Aceptar** se validan los datos, se actualiza la información y se presenta un mensaje indicando que el Proyecto se actualizo satisfactoriamente. Si ocurriese algún error se presentara el mensaje correspondiente.
- Al dar clic en **Cancelar** se cierra la ventana sin realizar ninguna acción.

### Tablas que intervienen en el caso de uso.



### 3.2.1.8 Tareas Asignados

#### Historial de Revisiones:

Fecha	Versión	Descripción	Autor
28-02-2011	1.0	Descripción inicial del caso de uso	Edison Lomas
01-03-2011	1.1	Modificación y actualización del caso de uso a petición del usuario funcional.	Edison Lomas
02-03-2011	1.2	Modificación y actualización del caso de uso a petición del usuario funcional.	Edison Lomas
03-03-2011	1.3	Modificación y actualización del caso de uso a petición del usuario funcional.	Edison Lomas
04-03-2011	1.4	Modificación y actualización del caso de uso en base al manejo de las horas que tomo la tarea.	Edison Lomas
07-03-2011	1.5	Modificación y actualización del caso de uso a petición del usuario funcional.	Edison Lomas
08-03-2011	1.6	Modificación y actualización del caso de uso a petición del usuario funcional.	Edison Lomas

09-03-2011	1.7	Modificación y actualización del caso de uso a petición del usuario funcional.	Edison Lomas
------------	-----	--	--------------



## Prototipo de pantallas

Pantalla principal de tareas asignadas

**Buscar Tarea**

Proyecto: SIGESPro

Nombre:

Tipo de Tarea: Selecciones...

Prioridad: Selecciones...

Buscar

**Tareas**

Nombre	Nombre	Fecha de inicio propuesta	Tiempo Propuesto(h)	Tipo de Tarea	Prioridad	Avance	Opciones
8	fdsfds	16/02/2012	0	Requerimiento	fghdfghdfg	31%	
14	yayyyyyyyyyyyyyyy	16/02/2012	0	Requerimiento	fghdfghdfg	31%	
20	dsafdsaf	16/02/2012	0	Error	fghdfghdfg	0%	

Pantalla de reporte de avances de tareas

**Tarea**

Proyecto: SIGESPro

Nombre: fdsfds

Descripción: sdfsdfs

Tiempo Propuesto(h): 0

Fecha de inicio propuesta: 16/02/2012

Fecha de finalización propuesta: 15/02/2012

Fecha de inicio:

Tipo de Tarea: Requerimiento

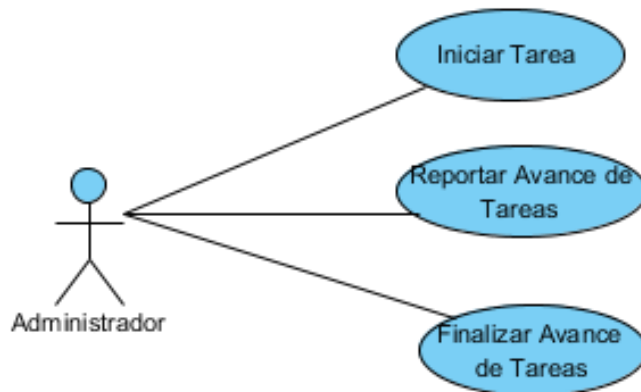
Prioridad: fghdfghdfg

Avance:  100 |  
31%

\* Campos requeridos

Aceptar Cancelar

## Diagrama de Caso de Uso:



## Prerrequisitos:

1. Usuario registrado y autorizado como desarrollador o jefe de proyecto.
2. Usuario debidamente autenticado en el sistema.
3. Tener tareas asignadas.

## Flujo Principal – Iniciar Tarea:

1. Se ingresará a la pantalla de tareas asignados mediante el menú de la aplicación.
2. Se presenta una pantalla con una lista desplegable con los proyectos que a los que hemos sido asignados.
3. Al seleccionar uno de los proyectos se nos presenta los siguientes filtros de búsqueda:
  - Nombre
    - Editable en caja de texto
    - Texto de máximo 50 caracteres
  - Tipo de tarea
    - Editable en lista desplegable

- Catalogo
- Prioridad
  - Editable en lista desplegable
  - Catalogo

Juntamente con un botón **Buscar** y una grilla de resultados debidamente paginada.

4. Al clic en el botón **Buscar** se actualiza la grilla de resultados en base a los criterios de búsqueda. En la parte derecha de cada registro de la grilla existen las opciones **Editar**.
5. Al dar clic en la opción **Editar** se presentan una pantalla con los siguientes campos:
  - Proyecto
    - Caja de texto de solo lectura
  - Nombre
    - Caja de texto de solo lectura
  - Descripción
    - Caja de texto de solo lectura
  - Tiempo Propuesto(h)
    - Caja de texto de solo lectura
  - Fecha de Inicio Propuesta
    - Caja de texto de solo lectura
  - Fecha de finalización Propuesta
    - Caja de texto de solo lectura
  - Tipo de tarea
    - Caja de texto de solo lectura
  - Prioridad
    - Caja de texto de solo lectura.

Juntamente con un botón **Iniciar Tarea** y el botón **Cancelar**

6. Al dar clic en el botón **Iniciar Tarea**, se coloca automáticamente la fecha de inicio de la tarea. Y se actualiza la pantalla con los siguientes campos:

- Proyecto
  - Caja de texto de solo lectura
- Nombre
  - Caja de texto de solo lectura
- Descripción
  - Caja de texto de solo lectura
- Tiempo Propuesto(h)
  - Caja de texto de solo lectura
- Fecha de Inicio Propuesta
  - Caja de texto de solo lectura
- Fecha de finalización Propuesta
  - Caja de texto de solo lectura
- Fecha de Inicio
  - Caja de texto de solo lectura
- Tipo de tarea
  - Caja de texto de solo lectura
- Prioridad
  - Caja de texto de solo lectura.
- Avance
  - Editable en componente slider.
  - Número entero
  - N/A

Juntamente con un botón **Aceptar** y el botón **Cancelar**, y se presenta un mensaje indicando que la tarea se actualizo correctamente. Si ocurriese algún error se presentara el mensaje respectivo.

7. Al dar clic en el botón Cancelar se cierra la ventana sin realizar ninguna acción.

## Flujo Alternativo 1 – Reportar Avance de Tareas

1. Se ingresara a la pantalla de tareas asignados mediante el menú de la aplicación.

2. Se presenta una pantalla con una lista desplegable con los proyectos que a los que hemos sido asignados.
3. Al seleccionar uno de los proyectos se nos presenta los siguientes filtros de búsqueda:
  - Nombre
    - Editable en caja de texto
    - Texto de máximo 50 caracteres
  - Tipo de tarea
    - Editable en lista desplegable
    - Catalogo
  - Prioridad
    - Editable en lista desplegable
    - Catalogo

Juntamente con un botón **Buscar** y una grilla de resultados debidamente paginada.

4. Al clic en el botón **Buscar** se actualiza la grilla de resultados en base a los criterios de búsqueda. En la parte derecha de cada registro de la grilla existen las opciones **Editar**.
5. Al dar clic en la opción **Editar** de una tarea ya iniciada se presentan una pantalla con los siguientes campos:
  - Proyecto
    - Caja de texto de solo lectura
  - Nombre
    - Caja de texto de solo lectura
  - Descripción
    - Caja de texto de solo lectura
  - Tiempo Propuesto(h)
    - Caja de texto de solo lectura
  - Fecha de Inicio Propuesta
    - Caja de texto de solo lectura
  - Fecha de finalización Propuesta

- Caja de texto de solo lectura
- Fecha de Inicio
  - Caja de texto de solo lectura
- Tipo de tarea
  - Caja de texto de solo lectura
- Prioridad
  - Caja de texto de solo lectura.
- Avance
  - Editable en componente slider.
  - Número entero
  - N/A

Juntamente con un botón **Aceptar** y el botón **Cancelar**

6. Colocamos el valor del avance de nuestra tarea y damos clic en **Aceptar**.
7. Al dar en el botón **Aceptar** se validan los datos, se guarda información y se presenta el mensaje indicando que la tarea se actualizo correctamente. Si ocurriese algún error damos se presenta el mensaje respectivo.
8. Al dar clic en **Cancelar** se cierra la ventana sin realizar ninguna acción.

## Flujo Alternativo 2 – Finalizar Avance de Tareas

1. Se ingresa a la pantalla de tareas asignados mediante el menú de la aplicación.
2. Se presenta una pantalla con una lista desplegable con los proyectos que a los que hemos sido asignados.
3. Al seleccionar uno de los proyectos se nos presenta los siguientes filtros de búsqueda:
  - Nombre

- Editable en caja de texto
- Texto de máximo 50 caracteres
- Tipo de tarea
  - Editable en lista desplegable
  - Catalogo
- Prioridad
  - Editable en lista desplegable
  - Catalogo

Juntamente con un botón **Buscar** y una grilla de resultados debidamente paginada.

4. Al clic en el botón **Buscar** se actualiza la grilla de resultados en base a los criterios de búsqueda. En la parte derecha de cada registro de la grilla existen las opciones **Editar**.
5. Al dar clic en la opción **Editar** de una tarea ya iniciada se presentan una pantalla con los siguientes campos:
  - Proyecto
    - Caja de texto de solo lectura
  - Nombre
    - Caja de texto de solo lectura
  - Descripción
    - Caja de texto de solo lectura
  - Tiempo Propuesto(h)
    - Caja de texto de solo lectura
  - Fecha de Inicio Propuesta
    - Caja de texto de solo lectura
  - Fecha de finalización Propuesta
    - Caja de texto de solo lectura
  - Fecha de Inicio
    - Caja de texto de solo lectura
  - Tipo de tarea
    - Caja de texto de solo lectura
  - Prioridad

- Caja de texto de solo lectura.
- Avance
  - Editable en componente slider.
  - Número entero
  - N/A

Juntamente con un botón **Aceptar** y el botón **Cancelar**

6. Colocamos el valor del avance de nuestra tarea en 100% la pantalla de modifica y nos presenta los siguientes campos.

- Proyecto
  - Caja de texto de solo lectura
- Nombre
  - Caja de texto de solo lectura
- Descripción
  - Caja de texto de solo lectura
- Tiempo Propuesto(h)
  - Caja de texto de solo lectura
- Fecha de Inicio Propuesta
  - Caja de texto de solo lectura
- Fecha de finalización Propuesta
  - Caja de texto de solo lectura
- Tiempo(h)
  - Editable en caja de texto
  - Número entero
  - Obligatorio
- Fecha de Inicio
  - Caja de texto de solo lectura
- Tipo de tarea
  - Caja de texto de solo lectura
- Prioridad
  - Caja de texto de solo lectura.
- Avance



- Editable en componente slider.
- Número entero
- N/A

Juntamente con el botón **Finalizar** y el botón **Cancelar**.

7. Al dar en el botón **Finalizar** se validan los datos, se guarda información y se presenta el mensaje indicando que la tarea se actualizo correctamente adicionalmente se coloca la fecha actual en la fecha de finalización de la tarea. Y nos modifica la pantalla de la siguiente forma:

- Proyecto
  - Caja de texto de solo lectura
- Nombre
  - Caja de texto de solo lectura
- Descripción
  - Caja de texto de solo lectura
- Tiempo Propuesto(h)
  - Caja de texto de solo lectura
- Fecha de Inicio Propuesta
  - Caja de texto de solo lectura
- Fecha de finalización Propuesta
  - Caja de texto de solo lectura
- Tiempo Propuesto
  - Caja de texto de solo lectura
- Fecha de Inicio
  - Caja de texto de solo lectura
- Fecha de Finalización
  - Caja de texto de solo lectura
- Tipo de tarea
  - Caja de texto de solo lectura
- Prioridad
  - Caja de texto de solo lectura.
- Avance
  - Caja de texto de solo lectura.

Sin ningún botón. Si ocurriese algún error damos se presenta el mensaje respectivo.

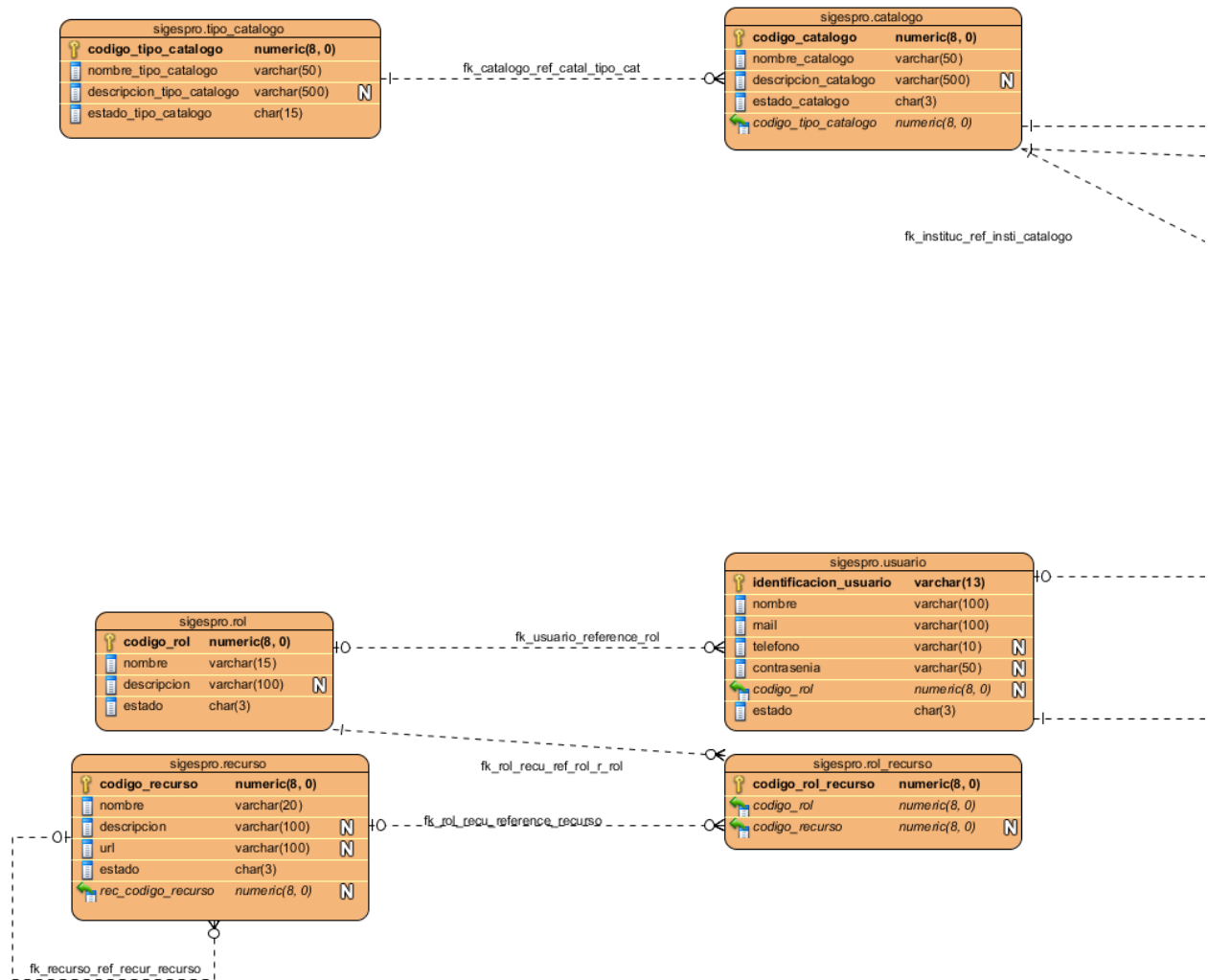
8. Al dar clic en **Cancelar** se cierra la ventana sin realizar ninguna acción.

**Tablas que intervienen en el caso de uso.**

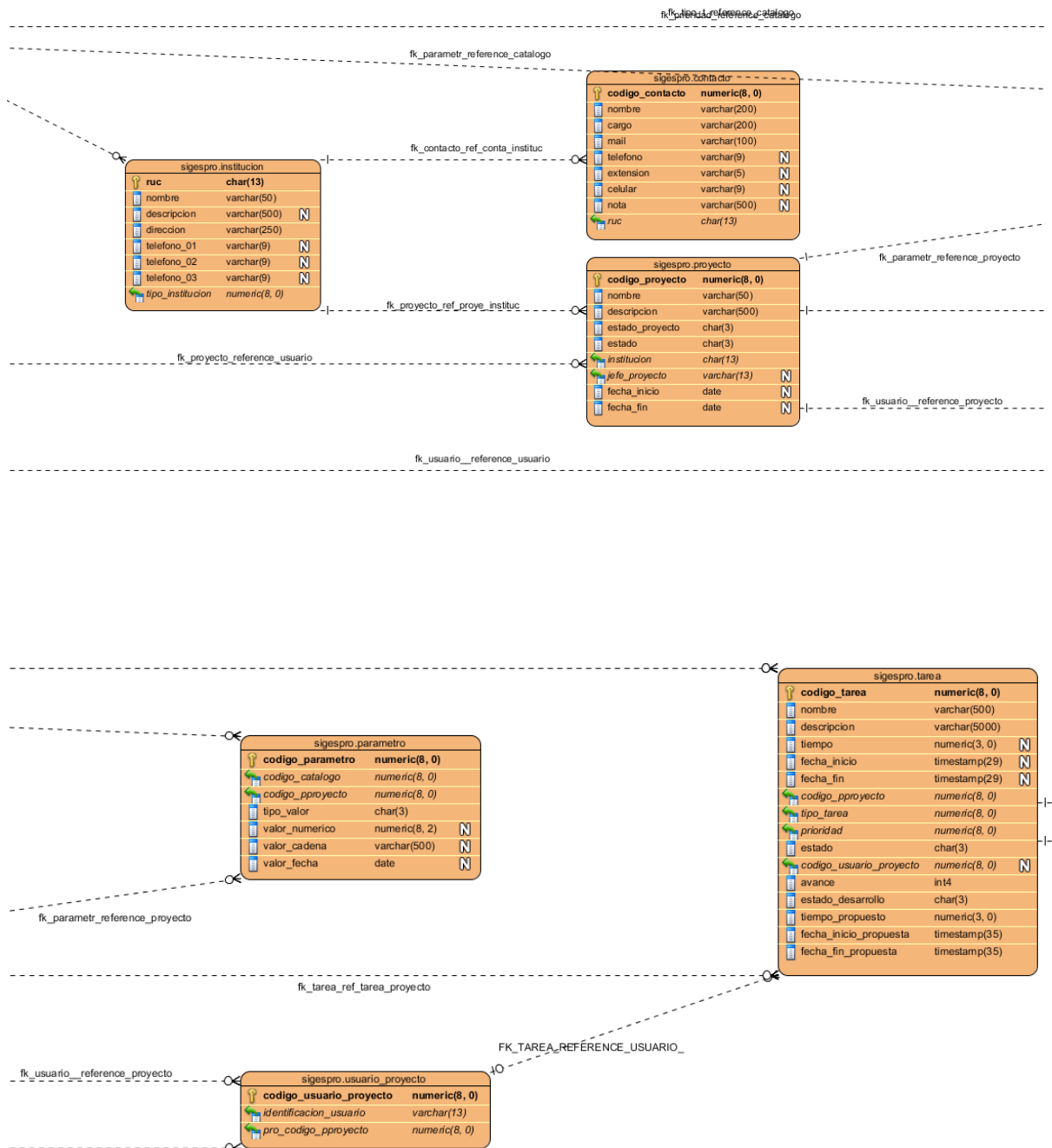


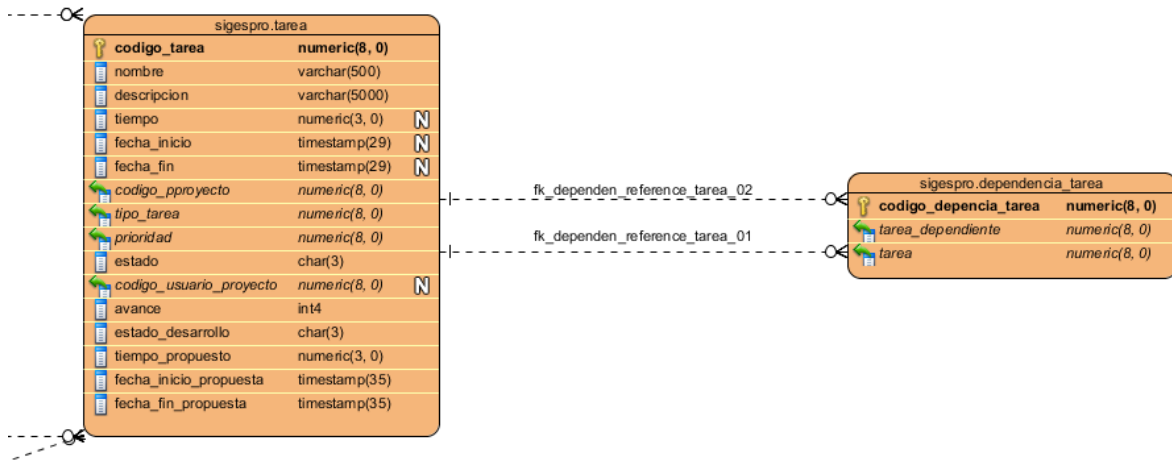
### 3.3 Fase de construcción

#### 3.3.1 Diagrama de Base de datos



# SIGESPro - Sistema de Gestión y Seguimiento de Proyectos de Software





### 3.3.2 Diccionario de Datos

#### TABLA CATALOGO

Esta entidad nos permite almacenar la información de cada uno de los catálogos que vamos a utilizar en el sistema, clasificados por grupos. Como por ejemplo los tipos de institución Pública o Privada.

Esta tabla contiene los siguientes campos:

- CODIGO\_CATALOGO.
  - Numeric(8,0)
  - Obligatorio
  - Clave primaria

Campo creado para almacenar el identificador del catálogo.

- NOMBRE\_CATALOGO
  - Varchar(50)
  - Obligatorio

Campo creado para almacenar el nombre del catálogo.

- DESCRIPCIÓN\_CATALOGO
  - Varchar(500)
  - No obligatorio

Campo creado para almacenar las descripciones del catálogo.

- ESTADO\_CATALOGO
  - Char(3)
  - Obligatorio

Campo creado para almacenar el estado del catálogo.

- CODIGO\_TIPO\_CATALOGO
  - Numeric(8,0)
  - Obligatorio

Campo creado para almacenar la referencia con la tabla tipo catálogo.

## **TABLA CONTACTO**

Esta entidad nos permite almacenar la información sobre los contacto de cada uno de los proyectos.

Esta tabla contiene los siguientes campos:

- CODIGO\_CONTACTO
  - Numeric(8,0)
  - Obligatorio
  - Clave primaria

Campo creado para almacenar el identificador de contactos.

- NOMBRE
  - Varchar(200)
  - Obligatorio

Campo creado para almacenar el nombre del contacto.

- CARGO
  - Varchar(200)
  - Obligatorio

Campo creado para almacenar el cargo del contacto.

- MAIL
  - Varchar(100)
  - Obligatorio

Campo creado para almacenar el mail del contacto.

- TELEFONO
  - Varchar(9)
  - No obligatorio

Campo creado para para almacenar el teléfono del contacto.

- EXTENSION
  - Varchar(5)
  - No obligatorio

Campo creado para almacenar la extensión del contacto.

- CELULAR
  - Varchar(9)

- No obligatorio

Campo creado para almacenar el teléfono celular del contacto.

- NOTA

- Varchar(500)
- No obligatorio

Campo creado para almacenar notas referentes al contacto.

- RUC

- Char(13)
- Obligatorio

Campo creado para almacenar la referencia hacia la institución.

### ***TABLA DEPENDENCIA\_TAREA***

Esta entidad nos permite tener una recursividad doble.

- CODIGO\_DEPENENCIA\_TAREA

- Numeric(8,0)
- Obligatorio
- Clave primaria

Campo creado para almacenar el identificador de la dependencia de tareas.

- TAREA\_DEPENDIENTE

- Numeric(8,0)
- Obligatorio

Campo creado para almacenar la referencia hacia la tarea.



- TAREA
  - Numeric(8,0)
  - Obligatorio

Campo creado para almacenar la referencia hacia la tarea.

### **TABLA INSTITUCION**

Esta entidad nos permite almacenar la información sobre las instituciones para las cuales se va a realizar el desarrollo del Sistema.

- RUC
  - Char(13)
  - Obligatorio
  - Clave primaria

Campo creado para almacenar el ruc de la institución.

- NOMBRE
  - Varchar(50)
  - Obligatorio

Campo creado para almacenar el nombre de la institución.

- DESCRIPCION
  - Varchar(500)
  - No obligatorio

Campo creado para almacenar la descripción de la institución.

- DIRECCION
  - Varchar(250)

- Obligatorio

Campo creado para almacenar la dirección de la institución.

- TELEFONO\_1
  - Varchar(9)
  - No obligatorio

Campo creado para almacenar el teléfono de la institución.

- TELEFONO\_2
  - Varchar(9)
  - No obligatorio

Campo creado para almacenar el teléfono de la institución.

- TELEFONO\_3
  - Varchar(9)
  - No obligatorio

Campo creado para almacenar el teléfono de la institución.

- TIPO\_INSTITUCION
  - Numeric(8,0)
  - Obligatorio

Campo creado para almacenar la referencia hacia el catálogo.

### **TABLA PARAMETRO**

Esta entidad nos permite almacenar los parámetros del sistema, como por ejemplo: Costo horas de trabajo.

- CODIGO\_PARAMETRO

- Numeric(8,0)
- Obligatorio
- Código primario

Campo creado para almacenar el identificador de parámetro.

- CODIGO\_CATALOGO

- Numeric(8,0)
- obligatorio

Campo creado para almacenar la referencia hacia el catálogo.

- CODIGO\_PROYECTO

- Numeric(8,0)
- obligatorio

Campo creado para almacenar la referencia hacia el proyecto

- TIPO\_VALOR

- Char(3)
- Obligatorio

Campo creado para almacenar el tipo de valor del parámetro.

- VALOR\_NUMERICO

- Numeric(8,2)
- No obligatorio

Campo creado para almacenar el valor numérico del parámetro.

- VALOR\_CADENA

- Varchar(500)
- No obligatorio

Campo creado para almacenar el valor de cadena del parámetro.

- VALOR\_FECHA
  - Date
  - No obligatorio

Campo creado para almacenar el valor de la fecha del parámetro.

### **TABLA PROYECTO**

Esta entidad nos permite almacenar la información básica de cada uno de los proyectos.

- CODIGO\_PROYECTO
  - Numeric(8,0)
  - Obligatorio
  - Clave primaria

Campo creado para almacenar el identificador de proyecto.

- NOMBRE
  - Varchar(50)
  - Obligatorio

Campo creado para almacenar el nombre del proyecto.

- DESCRIPCION
  - Varchar(500)
  - Obligatorio

Campo creado para almacenar la descripción del proyecto.

- ESTADO\_PROYECTO
  - Char(3)

- Obligatorio

Campo creado para almacenar el estado del proyecto.

- ESTADO

- Char(3)
- Obligatorio

Campo creado para almacenar los estados del proyecto para el manejo de los borrados lógicos.

- INSTITUCION

- Char(13)
- Obligatorio

Campo creado para almacenar la referencia a la institución.

- JEFE\_PROYECTO

- Varchar(13)
- No obligatorio

Campo creado para almacenar la referencia al usuario.

- FECHA\_INICIO

- Date
- No obligatorio

Campo creado para almacenar la fecha de inicio del proyecto.

- FECHA\_FIN

- Date
- No obligatorio

Campo creado para almacenar la fecha de culminación del proyecto.

## **TABLA RECURSO**

Esta entidad nos permite almacenar la información sobre los recursos disponibles del sistema.

- CODIGO\_RECURSO
  - Numeric(8,0)
  - Obligatorio
  - Clave primaria

Campo creado para almacenar el identificador del recurso.

- NOMBRE
  - Varchar(20)
  - Obligatorio

Campo creado para almacenar el nombre del recurso.

- DESCRIPCION
  - Varchar(100)
  - No obligatorio

Campo creado para almacenar la descripción del recurso.

- URL
  - Varchar(100)
  - No obligatorio

Campo creado para almacenar el URL del recurso.

- ESTADO
  - Char(3)

- Obligatorio

Campo creado para almacenar el estado del recurso.

- REC\_CODIGO\_RECURSO
  - Numeric(8,0)
  - No obligatorio

Campo creado para almacenar la referencia hacia el recurso.

## **TABLA ROL**

Esta entidad nos permite almacenar la información de los roles que va a manejar el sistema.

- CODIGO\_ROL
  - Numeric(8,0)
  - Obligatorio
  - Clave Primaria

Campo creado para almacenar el identificador del rol.

- NOMBRE
  - Varchar(15)
  - Obligatorio

Campo creado para almacenar el nombre del rol.

- DESCRIPCION
  - Varchar(100)
  - No obligatorio

Campo creado para almacenar la descripción del rol.

- ESTADO
  - Char(3)
  - Obligatorio

Campo creado para almacenar el estado del rol.

### **TABLA ROL\_RECURSO**

Esta entidad nos permite almacenar las relaciones entre rol y el recurso disponible para dicho rol.

- CODIGO\_ROL\_RECURSO
  - Numeric(8,0)
  - Obligatorio
  - Clave primaria

Campo creado para almacenar el identificador de rol recurso.

- CODIGO\_ROL
  - Numeric(8,0)
  - Obligatorio

Campo creado para almacenar la referencia hacia el rol.

- CODIGO\_RECURSO
  - Numeric(8,0)
  - No obligatorio

Campo creado para almacenar la referencia hacia el recurso.



## TABLA TAREA

- CODIGO\_TAREA
  - Numeric(8,0)
  - Obligatorio
  - Clave primaria

Campo creado para almacenar el identificador de la tarea.

- NOMBRE
  - Varchar(500)
  - Obligatorio

Campo creado para almacenar el nombre de la tarea.

- DESCRIPCION
  - Varchar(500)
  - Obligatorio

Campo creado para almacenar la descripción de la tarea.

- TIEMPO
  - Numeric(3,0)
  - No obligatorio

Campo creado para almacenar el tiempo que tomo realizar la tarea.

- FECHA\_INICIO
  - Timestamp
  - No obligatorio

Campo creado para almacenar la fecha de inicio de la tarea.

- FECHA\_FINAL

- Timestamp
- No obligatorio

Campo creado para almacenar la fecha de finalización de la tarea.

- CODIGO\_PROYECTO
  - Numeric(8,0)
  - Obligatorio

Campo creado para almacenar la referencia hacia proyecto.

- TIPO\_TAREA
  - Numeric(8,0)
  - Obligatorio

Campo creado para almacenar la referencia al catálogo.

- PRIORIDAD
  - Numeric(8,0)
  - Obligatorio

Campo creado para almacenar la referencia hacia el catálogo.

- ESTADO
  - Char(3)
  - Obligatorio

Campo creado para almacenar el estado en el que se encuentra la tarea,

- CODIGO\_USUARIO\_PROYECTO
  - Numeric(8,0)
  - No obligatorio

Campo creado para almacenar la referencia hacia la tabla usuario\_proyecto

- AVANCE
  - Integer
  - Obligatorio

Campo creado para almacenar el avance de la tarea.

- ESTADO\_DESARROLLO
  - Numeric(8,0)
  - No obligatorio

Campo creado para almacenar el estado de desarrollo de la tarea.

- TIEMPO\_PROPUESTO
  - Numeric(3,0)
  - Obligatorio

Campo creado para almacenar el tiempo propuesto para el desarrollo de la tarea.

- FECHA\_INICIO\_PROPUESTA
  - Timestamp
  - Obligatorio

Campo creado para almacenar la fecha de inicio propuesta para el inicio de la tarea.

- FECHA\_FIN\_PROPUESTA
  - Timestamp
  - Obligatorio

Campo creado para almacenar la fecha de finalización propuesta para la tarea.

## **TABLA USUARIO**

Esta entidad nos permite almacenar la información de los usuarios de cada sistema

- IDENTIFICACION\_USUARIO
  - Varchar(13)
  - Obligatorio
  - Clave primaria

Campo creado para almacenar el identificador del usuario.

- NOMBRE
  - Varchar(100)
  - Obligatorio

Campo creado para almacenar el nombre del usuario.

- MAIL
  - Varchar(100)
  - Obligatorio

Campo creado para almacenar el mail del usuario.

- TELEFONO
  - Varchar(10)
  - No obligatorio

Campo creado para almacenar el teléfono del usuario.

- CONTRASENIA
  - Varchar(50)
  - No obligatorio

Campo creado para almacenar la contraseña del usuario.

- CODIGO\_ROL
  - Numeric(8,0)
  - No obligatorio

Campo creado para almacenar la referencia hacia la tabla rol.

- ESTADO
  - Char(3)
  - Obligatorio

Campo creado para almacenar el estado del usuario.

### **TABLA TIPO\_CATALOGO**

Esta entidad nos permite almacenar los tipos de catálogos lo cual nos va a permitir filtrar cierta información y estandarizar un poco la misma como por ejemplo el tipo de institución.

- CODIGO\_TIPO\_CATALOGO
  - Numeric(8,0)
  - Obligatorio
  - Clave primaria

Campo creado para almacenar el identificador de tipo de catálogo.

- NOMBRE\_TIPO\_CATALOGO
  - Varchar(50)
  - Obligatorio

Campo creado para almacenar el nombre del tipo de catálogo.

- DESCRIPCION\_TIPO\_CATALOGO
  - Varchar(500)

- No obligatorio

Campo creado para almacenar la descripción del tipo de catálogo.

- ESTADO\_TIPO\_CATALOGO
  - Char(15)
  - Obligatorio

Campo creado para almacenar el estado del tipo de catálogo.

### ***TABLA USUARIO\_PROYECTO***

Esta entidad nos permite almacenar las relaciones entre usuarios, y proyectos ya que un usuario puede tener varios proyectos y viceversa

- CODIGO\_USUARIO\_PROYECTO
  - Numeric(8,0)
  - Obligatorio
  - Clave primaria

Campo creado para almacenar el identificador de usuario de proyecto.

- IDENTIFICACION\_USUARIO
  - Varchar(13)
  - Obligatorio

Campo creado para almacenar la referencia hacia el usuario.

- PRO\_CODIGO\_PROYECTO
  - Numeric(8,0)
  - Obligatorio

Campo creado para almacenar la referencia hacia proyecto.

### ***SECUENCIA SEQ\_CATALOGO***

Secuencia creada para generar el secuencial para el código de la tabla CATALOGO.

### ***SECUENCIA SEQ\_CONTACTO***

Secuencia creada para generar el secuencial para el código de la tabla CONTACTO.

### ***SECUENCIA SEQ\_DEPENDENCIA\_TAREA***

Secuencia creada para generar el secuencial para el código de la tabla DEPENDENCIA\_TAREA.

### ***SECUENCIA SEQ\_DESARROLLADOR\_PROYECTO***

Secuencia creada para generar el secuencial para el código de la tabla DESARROLLADOR\_PROYECTO.

### ***SECUENCIA SEQ\_PARAMETRO***

Secuencia creada para generar el secuencial para el código de la tabla PARAMETRO.

### ***SECUENCIA SEQ\_PROYECTO***

Secuencia creada para generar el secuencial para el código de la tabla PROYECTO.

### ***SECUENCIA SEQ\_RECURSO***

Secuencia creada para generar el secuencial para el código de la tabla RECURSO.

### ***SECUENCIA SEQ\_ROL***

Secuencia creada para generar el secuencial para el código de la tabla ROL.

### ***SECUENCIA SEQ\_ROL\_RECURSO***

Secuencia creada para generar el secuencial para el código de la tabla ROL\_RECURSO.

### ***SECUENCIA SEQ\_TAREA***

Secuencia creada para generar el secuencial para el código de la tabla TAREA.

### ***SECUENCIA SEQ\_TIPO\_CATALOGO***

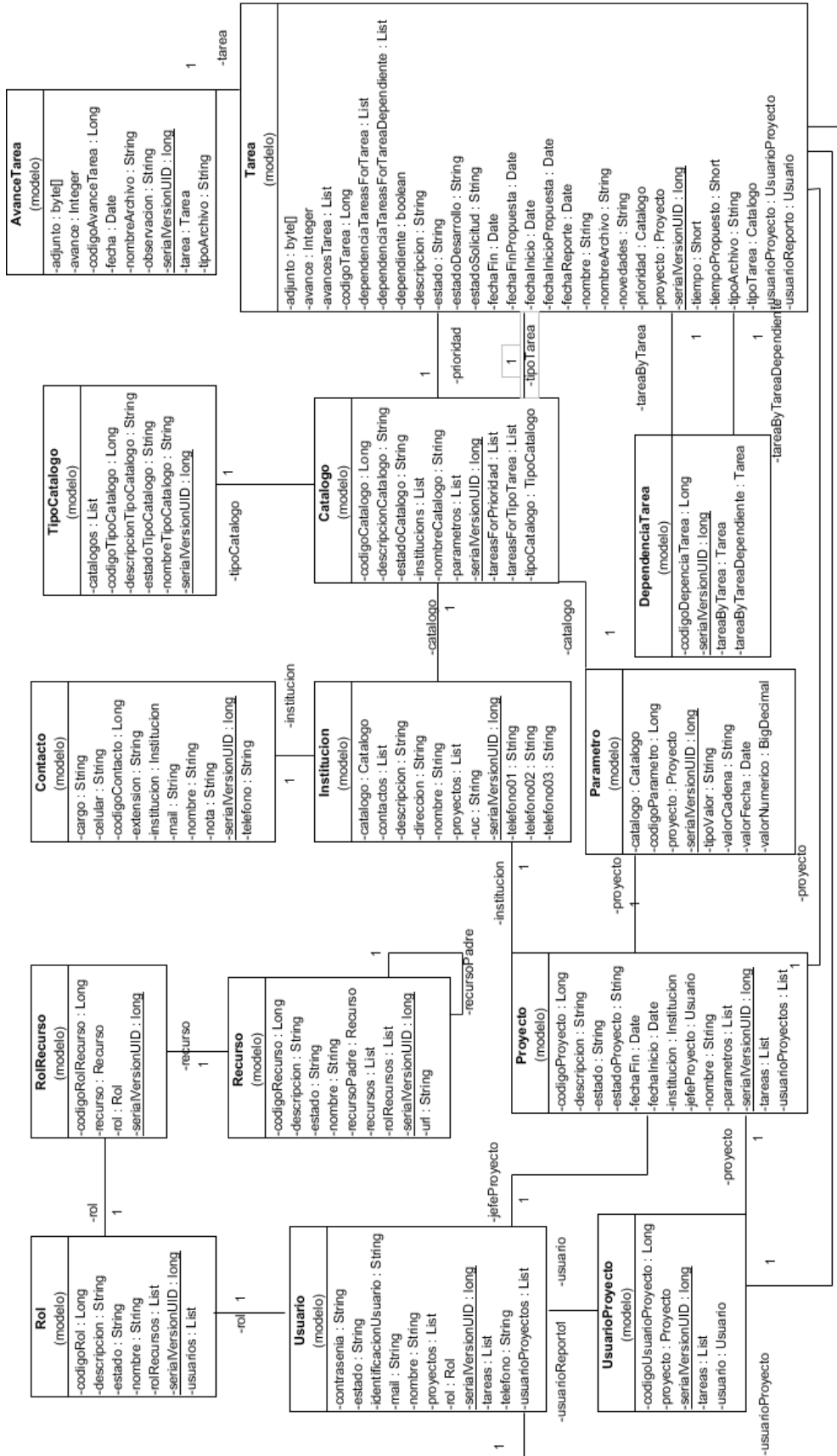
Secuencia creada para generar el secuencial para el código de la tabla TIPO\_CATALOGO.



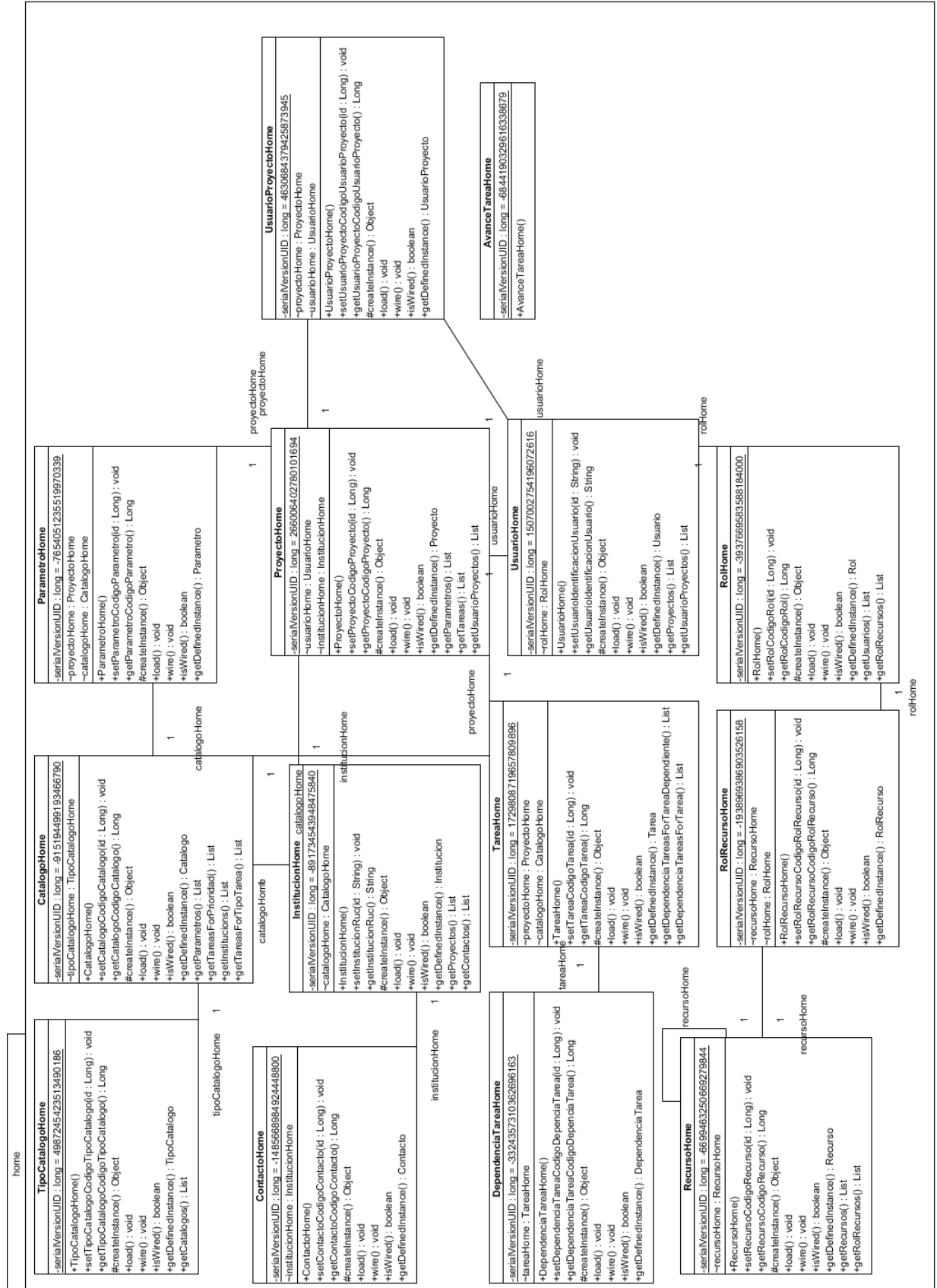
### ***SECUENCIA SEQ\_USUARIO\_PROYECTO***

Secuencia creada para generar el secuencial para el código de la tabla USUARIO\_PROYECTO.

### **3.3.3 Diagrama de Clases**









### 3.4 Fase de transición

## ACTA DE ENTREGA / RECEPCIÓN

**Proyecto:** SIGESPro

**Tema a tratar:** Entrega del Sistema de gestión y Seguimiento de Proyectos de Software

**Fecha:** 17 de Febrero del 2012

Luego de cumplidas las etapas contempladas en el plan de proyecto de desarrollo del sistema de gestión y seguimiento de proyectos de desarrollo de software SIGESPro, incluyendo la etapa de capacitación, validación funcional y validación de datos con el Ing. Mauricio Santacruz, gerente general de la empresa se acuerda lo siguiente:

- El Ing. Darwin Argoti, jefe de proyectos de la empresa, luego de haber recibido la capacitación técnica necesaria, será el encargado de realizar el mantenimiento técnico y brindar soporte a los usuarios, así como del desarrollo de nuevos requerimientos.
- El nuevo sistema de gestión de proyectos SIGESPro entrará en funcionamiento desde el día 21 de febrero del 2012.
- Resumen de las tablas generadas para el nuevo sistema.

Tablas actuales	Total Registros	Descripción
catalogo	0	Esta entidad nos va a permitir almacenar la información de la cada uno de los catálogos que vamos a utilizar en el sistema como Publica – Privadas.
contacto	0	Esta entidad almacena la información sobre el contacto de cada uno de los proyectos.
dependencia_tarea	0	Esta entidad permite tener una recursividad doble.
institucion	0	Esta entidad va a almacenar la información sobre las instituciones para las cuales se va a realizar el desarrollo del Sistema.
parametro	0	Esta entidad almacena los parámetros del sistema, en como día no laborables, costo de recursos, horas de trabajo.
proyecto	0	Esta entidad es la encargada de almacenar la información básica de cada uno de los proyectos.
recurso	0	Esta entidad va a almacenar la información sobre los recursos disponibles del sistema.
rol	0	Esta entidad es la encargada de almacenar la información de los roles que va a manejar el sistema.
rol_recurso	0	Esta entidad es la encargada de almacenar las relaciones entre rol y el recurso disponible para

		dicho rol.
tarea	0	Esta entidad es la encargada de almacenar cada una de las tareas que van a formar parte del proyecto.
tipo_catalogo	0	Esta entidad va a almacenar los tipos de catálogos lo cual nos va a permitir filtrar cierta información y estandarizar un poco la misma como por ejemplo el tipo de institución.
usuario	0	Esta entidad va a almacenar la información de los usuarios de cada sistema.
usuario_trabajo	0	Esta entidad va a almacenar las relaciones entre usuarios, y proyectos ya que un usuario puede tener varios proyectos y viceversa.

Para constancia y ratificando el contenido de esta acta firman:

---

Ing. Mauricio Santacruz  
Gerente General

---

Ing. Darwin Argoti  
Jefe de Proyectos

---

Edison Lomas  
Desarrollador del Sistema



## Observaciones:

Se entrega la siguiente documentación al Ing. Mauricio Santacruz.

- Carpeta de Proyecto con documentación impresa con el siguiente contenido:
  - Artefactos generados siguiendo la Metodología RUP para el Sistema de Gestión de Proyectos.
  - Documento de visión
  - Lista de Riesgos.
  - Documento de Arquitectura del Software.
  - Plan de Desarrollo de Software.
  - Glosario.
  - Casos de Uso.
  - Especificaciones de caso de uso.
  - Diagrama y diccionario de base de datos.
  
- Cd con información digital de:
  - Código fuente y librerías.
  - Documentación.
  - Diagramas Entidad Relación de Base de Datos.
  - Scripts de Base de Datos.
  - Diccionario de Datos.
  - Manual de Usuario.
  - Documentación para publicación de la aplicación en el servidor.

## CAPÍTULO 4. Conclusiones y Recomendaciones

### 4.1 Conclusiones

- Es factible el desarrollo de un sistema que permita supervisar y controlar el desarrollo de los múltiples proyectos de la empresa.
- La integración de una aplicación con un sistema de autenticación centralizada y con un proveedor de autorización es fácilmente configurable.
- Mediante la asignación de jefes de proyectos y desarrolladores a cada proyecto, informa a los integrantes del compromiso con dicho proyecto.
- Mediante la selección del desarrollador responsable de cada tarea se facilita la asignación de tareas por parte del jefe de proyecto.
- Cuando cada desarrollador reporta el avance de cada una de sus tareas se facilita el poder controlar el avance que tiene el proyecto.
- Al permitir a los usuarios reportar errores si permite que el usuario funcional vaya a tener un mayor compromiso y se involucra mucho más con el proyecto.
- El manejo de un proyecto de software es muy complicado ya que es muy complicado la estimación de tiempos, y la capacidad de los desarrolladores para cumplirlas es diferente.

## 4.2 Recomendaciones

- Las empresas de desarrollo de software deberían contar con un sistema que les facilite la administración y gestión de proyectos.
- Para mejorar la seguridad de las aplicaciones deberían integrarse con un sistema de autenticación centralizado.
- Es importante notificar a los desarrolladores cuando sean asignados para formar parte de un equipo de trabajo.
- El registro periódico de avance de cada tarea es importante para poder tener un reporte de avance del proyecto actualizado.
- De cada proyecto de desarrollo debemos llevar un muy buen registro de versionamiento.
- Definir adecuadamente los procesos para lograr que la implementación del sistema sea la más adecuada en base a los requerimientos del usuario.
- Utilizar las versiones y las últimas versiones finales de las herramientas que vamos a necesitar para nuestro desarrollo, y verificar que estas tengan el soporte adecuado.
- Codificar y documentar adecuadamente el código para facilitar el posterior mantenimiento de la misma aplicación.
- Utilizar los principios básicos de la programación orientada a objetos para mejorar la codificación y evitar la duplicidad de código.

## Glosario

- **A**

**Ant.** Herramienta usada en programación para la realización de tareas mecánicas y repetitivas, normalmente durante la fase de compilación y construcción (build). Es un software para procesos de automatización de compilación desarrollado en lenguaje Java y requiere la plataforma Java.

**Action script.** Lenguaje de programación de la Plataforma Adobe Flash. Originalmente desarrollado como una forma para que los desarrolladores programen de forma más interactiva. La programación con ActionScript permite mucha más eficiencia en las aplicaciones de la plataforma Flash para construir animaciones de todo tipo, desde simples a complejas, ricas en datos e interfaces interactivas.

- **C**

**Cas.** Central Authentication Service, servicio de autenticación central.

**Core.** Núcleo del proyecto.

- **E**

**Ejb.** Enterprise JavaBeans, son una de las API que forman parte del estándar de construcción de aplicaciones empresariales J2EE Oracle Corporation (inicialmente desarrollado por Sun Microsystems).

- **F**

**Framework.** Es el marco de la aplicación o conjunto de bibliotecas orientadas a la reutilización a muy gran escala de componentes software para el desarrollo rápido de aplicaciones.

- **G**

**Gantt.** Es una popular herramienta gráfica cuyo objetivo es mostrar el tiempo de dedicación previsto para diferentes tareas o actividades a lo largo de un tiempo total determinado.

- **H**

**Hibernate.** Framework para implementar la capa de persistencia en Java. Es un proyecto de código abierto que permite el mapeo de objetos a una base de datos relacional con su propio lenguaje de acceso a los datos HQL y con SQL.

- **I**

**Issue tracking.** Gestor de errores de proyectos.

- **J**

**JPA.** Java Persistence API, framework para implementar la capa de persistencia en Java. Es un proyecto de código abierto que permite el mapeo de objetos a una base de datos relacional.

- **L**

**LDAP.** Son las siglas de Lightweight Directory Access Protocol (en español Protocolo Ligero de Acceso a Directorios) que hacen referencia a un protocolo a nivel de aplicación el cual permite el acceso a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red. LDAP también es considerado una base de datos (aunque su sistema de almacenamiento puede ser diferente) a la que pueden realizarse consultas. Un directorio es un conjunto de objetos con atributos organizados en una manera lógica y jerárquica. El ejemplo más común es el directorio telefónico, que consiste en una serie de nombres (personas u organizaciones) que están ordenados alfabéticamente, con cada nombre teniendo una dirección y un número de teléfono adjuntos.

- **M**

**Mailinglist.** Listas de correo.

- **O**

**Outsourcing.** Es el proceso económico en el cual una empresa mueve o destina los recursos orientados a cumplir ciertas tareas hacia una empresa externa por medio de un contrato. Esto se da especialmente en el caso de la subcontratación de empresas especializadas. Para ello, pueden contratar solo al personal, caso en el cual los recursos los aportará el cliente (instalaciones, hardware y software), o contratar tanto el personal como los recursos.

- **P**

**POJOs.** Plain Old Java Object, término utilizado para enfatizar que un objeto Java no es más que un objeto bean de utilidad básica. A la práctica son un conjunto de APIS hechas por diferentes fabricantes que dan una alternativa a la implementación J2EE para EJB.

**Portlets.** Micro aplicaciones auto contenidas y desplegadas en un servidor compatible.

- **Q**

**Query.** Cadena de consulta, este término generalmente se utiliza para hacer referencia a una interacción con una base de datos.

- **R**

**RIA.** Rich Internet applications, o aplicaciones de Internet enriquecidas, son aplicaciones web que tienen la mayoría de las características de las aplicaciones de escritorio tradicionales. Estas aplicaciones utilizan un navegador web estandarizado

para ejecutarse y por medio de complementos o mediante una máquina virtual se agregan las características adicionales.

**Ruby.** Es un lenguaje de programación interpretado, reflexivo y orientado a objetos, creado por el programador japonés Yukihiro "Matz" Matsumoto, quien comenzó a trabajar en Ruby en 1993, y lo presentó públicamente en 1995. Combina una sintaxis inspirada en Python y Perl con características de programación orientada a objetos similares a Smalltalk. Comparte también funcionalidad con otros lenguajes de programación como Lisp, Lua, Dylan y CLU. Ruby es un lenguaje de programación interpretado en una sola pasada y su implementación oficial es distribuida bajo una licencia de software libre.

- **S**

**Seam.** Framework desarrollado por JBoss, introduce el concepto de contextos. Cada componente de Seam existe dentro de un contexto. El contexto conversacional por ejemplo captura todas las acciones del usuario hasta que éste sale del sistema o cierra el navegador - inclusive puede llevar un control de múltiples pestañas y mantiene un comportamiento consistente cuando se usa el botón de regresar del navegador.

**Servlet.** Pequeño programa que corre en un servidor. Por lo general son aplicaciones Java que corren en un entorno de servidor web. Esto es análogo a una aplicación Java que corre en un navegador.

**Single SingOn.** Es la autenticación única por parte del usuario para acceder a sus recursos de la aplicación. En el caso del Single SignOn, permite gestionar los passwords de acceso a diferentes aplicaciones mediante un único login y password de manera que el usuario sólo debe recordar una contraseña de acceso.

**Stakeholders.** Es el público interesado o el entorno interesado.

- **T**

**Tomcat.** Es un servidor web con soporte de servlets y JSPs. Tomcat no es un servidor de aplicaciones, como JBoss o JOnAS. Incluye el compilador Jasper, que compila JSPs convirtiéndolas en servlets.

- **U**

**Unicode.** Es un estándar de codificación de caracteres diseñado para facilitar el tratamiento informático, transmisión y visualización de textos de múltiples lenguajes y disciplinas técnicas además de textos clásicos de lenguas muertas. El término Unicode proviene de los tres objetivos perseguidos: universalidad, uniformidad y unicidad.

- **W**

**Widget.** Es una pequeña aplicación o programa, usualmente presentado en archivos o ficheros pequeños que son ejecutados por un motor de widgets o Widget Engine. Entre sus objetivos están dar fácil acceso a funciones frecuentemente usadas y proveer de información visual. Sin embargo, los widgets pueden hacer todo lo que la imaginación desee e interactuar con servicios e información distribuida en Internet; pueden ser vistosos relojes en pantalla, notas, calculadoras, calendarios, agendas, juegos, ventanas con información del tiempo en su ciudad, etc.

- **X**

**XDoclet.** Es un motor de Código abierto para el Lenguaje de programación Java, su función es la generación de código. Está asociado a la programación orientada a los atributos, es decir, usted puede lograr más funcionalidad agregándole metadata (atributos) a su código. Esto se lleva a cabo con tags JavaDoc. XDoclet permite integración continua en el componente de desarrollo orientado. Los desarrolladores deben concentrar su trabajo de edición en un solo fichero fuente Java por cada componente.



**XML.** Extensible Markup Language o lenguaje de marcado extensible, lenguaje basado en documentos de texto plano con etiquetas que delimitan los elementos. Fue desarrollado originalmente por W3C para separar en una plana web el contenido de la estructura, y con la finalidad de acabar sustituyendo el HTML (<http://www.w3.org/XML/>). En la actualidad se ha convertido en el formato estándar para intercambiar información entre aplicaciones.

## Bibliografía

### Libros.

[L01] FAIN, Yacov; RASPUTNIS, Victor; TARTAKOVSKY, Anatole. Enterprise Development with Flex. Primera Edición .O'Reilly Media / Adobe Dev Library. 2010.

[L02] NOBLE, Joshua; ANDERSON, Todd; LOTT, Joey; KAZOUN, Chafic; GEORGE, Abey. Flex 3 Cookbook: The Adobe Developer Library Guide for Rich Internet Application Developers. O'Reilly Media / Adobe Dev Library 2009

### Sitios Web.

[W01] <http://www.jboss.com/>

[W02] [http://es.wikipedia.org/wiki/JBoss\\_Seam](http://es.wikipedia.org/wiki/JBoss_Seam)

[W03] <http://www.jasig.org/>

[W04] [http://en.wikipedia.org/wiki/Central\\_Authentication\\_Service](http://en.wikipedia.org/wiki/Central_Authentication_Service)

[W05] [http://searchsecurity.techtarget.com/sDefinition/0,,sid14\\_gci340859,00.html](http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci340859,00.html)

[W06] [http://en.wikipedia.org/wiki/Single\\_sign-on](http://en.wikipedia.org/wiki/Single_sign-on)

[W07] <http://exadel.com/web/portal/opensource?redirected=/flamingo>

[W08] <http://code.google.com/p/exadel-flamingo/>

[W09] [http://www.exadel.com/flamingo/docs/guide/en/html\\_single/](http://www.exadel.com/flamingo/docs/guide/en/html_single/)

[W10] <http://static.springsource.org/spring-security/site/reference.html>

[W11] <http://carloscacique.blogspot.com/2009/07/que-es-spring-security.html>

[W12] <http://www.adobe.com/es/products/flex.html>

[W13] [http://es.wikipedia.org/wiki/Adobe\\_Flex](http://es.wikipedia.org/wiki/Adobe_Flex)

[W14] <http://www.joseduenas.com/archivos/%C2%BFque-es-adobe-flex>

[W15] <http://www-01.ibm.com/software/integration/visualization/elixir-standard/#>

[W16] <http://en.wikipedia.org/wiki/PostgreSQL>

[W17] <http://www.postgresql.org/about/>

[W18] <http://www.jboss.org/>

[W19] <http://exadel.org/flamingo>

## Anexos

### Script de creación de base de datos

```

CREATE ROLE "SIGESPro" LOGIN
    ENCRYPTED PASSWORD 'md532429c43ec000f52d662ab3a5accbcb4'
    SUPERUSER INHERIT CREATEDB CREATEROLE;

-- DROP DATABASE "SIGESPro";

CREATE DATABASE "SIGESPro"
    WITH OWNER = "SIGESPro"
        ENCODING = 'UTF8'
        LC_COLLATE = 'Spanish_Ecuador.1252'
        LC_CTYPE = 'Spanish_Ecuador.1252'
        CONNECTION LIMIT = -1;

/*=====*/
/* User: SIGESPRO */
/*=====*/
create schema SIGESPRO authorization "SIGESPro";

create sequence SIGESPRO.SEQ_CATALOGO
increment 1
minvalue 0
start 0;

create sequence SIGESPRO.SEQ_CONTACTO
increment 1
minvalue 0
start 0;

create sequence SIGESPRO.SEQ_DEPENDENCIA_TAREA
increment 1
minvalue 0
start 0;

create sequence SIGESPRO.SEQ_DESARROLLADOR
increment 1
minvalue 0
start 0;

create sequence SIGESPRO.SEQ_DESARROLLADOR_PROYECTO
increment 1
minvalue 0
start 0;

create sequence SIGESPRO.SEQ_PARAMETRO
increment 1
minvalue 0
start 0;

create sequence SIGESPRO.SEQ_PROYECTO

```

```

increment 1
minvalue 0
start 0;

create sequence SIGESPRO.SEQ_RECURSO
increment 1
minvalue 0
start 0;

create sequence SIGESPRO.SEQ_ROL
increment 1
minvalue 0
start 0;

create sequence SIGESPRO.SEQ_ROL_RECURSO
increment 1
minvalue 0
start 0;

create sequence SIGESPRO.SEQ_TAREA
increment 1
minvalue 0
start 0;

create sequence SIGESPRO.SEQ_TIPO_CATALOGO
increment 1
minvalue 0
start 0;

create sequence SIGESPRO.SEQ_USUARIO_PROYECTO
increment 1
minvalue 0
start 0;

/*=====*/
/* Table: CATALOGO */
/*=====*/
create table SIGESPRO.CATALOGO (
    CODIGO_CATALOGO      NUMERIC(8,0)      not null,
    NOMBRE_CATALOGO     VARCHAR(50)       not null,
    DESCRIPCION_CATALOGO VARCHAR(500)      null,
    ESTADO_CATALOGO     CHAR(3)           not null,
    CODIGO_TIPO_CATALOGO NUMERIC(8,0)      not null
);

comment on table SIGESPRO.CATALOGO is
'Esta entidad nos va a permitir almacenar la informacion de la cada uno de
los catalogos que vamos a utilizar en el sistema como Publica - Privadas';

alter table SIGESPRO.CATALOGO
    add constraint PK_CATALOGO primary key (CODIGO_CATALOGO);

/*=====*/
/* Table: CONTACTO */
/*=====*/

```

## SIGESPro - Sistema de Gestión y Seguimiento de Proyectos de Software

```
create table SIGESPRO.CONTACTO (
  CODIGO_CONTACTO      NUMERIC(8,0)      not null,
  NOMBRE                VARCHAR(200)     not null,
  CARGO                 VARCHAR(200)     not null,
  MAIL                  VARCHAR(100)     not null,
  TELEFONO              VARCHAR(9)       null,
  EXTENSION             VARCHAR(5)       null,
  CELULAR               VARCHAR(9)       null,
  NOTA                  VARCHAR(500)     null,
  RUC                   CHAR(13)         not null
);

comment on table SIGESPRO.CONTACTO is
'Esta entidad va a almacenar la informacion sobre el contacto de cada uno de
los proyectos';

alter table SIGESPRO.CONTACTO
  add constraint PK_CONTACTO primary key (CODIGO_CONTACTO);

/*=====*/
/* Table: DEPENDENCIA_TAREA */
/*=====*/
create table SIGESPRO.DEPENDENCIA_TAREA (
  CODIGO_DEPENCIATAREA NUMERIC(8,0)      not null,
  TAREA_DEPENDIENTE    NUMERIC(8,0)      not null,
  TAREA                 NUMERIC(8,0)      not null
);

comment on table SIGESPRO.DEPENDENCIA_TAREA is
'Esta entidad va a permitir tener una recursividad doble';

alter table SIGESPRO.DEPENDENCIA_TAREA
  add constraint PK_DEPENDENCIA_TAREA primary key (CODIGO_DEPENCIATAREA);

/*=====*/
/* Table: DESARROLLADOR */
/*=====*/
create table SIGESPRO.DESARROLLADOR (
  USERNAME          VARCHAR(25)      not null,
  IDENTIFICACION    VARCHAR(13)      not null,
  NOMBRE            VARCHAR(100)     not null,
  CODIGO_CATALOGO  NUMERIC(8,0)      null
);

comment on table SIGESPRO.DESARROLLADOR is
'Esta entidad va a permitir almacenar el nombre del usuario asignado a cada
proyecto, este usuario debe existir en el LDAP, por lo cual se debe
realizar la condici n por medio de aplicacion';

alter table SIGESPRO.DESARROLLADOR
  add constraint PK_DESARROLLADOR primary key (USERNAME);

/*=====*/
/* Table: DESARROLLADOR_PROYECTO */
/*=====*/
create table SIGESPRO.DESARROLLADOR_PROYECTO (
  CODIGO_DESARROLLADOR_PROYECTO NUMERIC(8,0)      not null,
```

## SIGESPro - Sistema de Gestión y Seguimiento de Proyectos de Software

```
        CODIGO_PROYECTO      NUMERIC(8,0)      not null,  
        USERNAME             VARCHAR(25)        not null  
);
```

```
comment on table SIGESPRO.DESARROLLADOR_PROYECTO is  
'Esta entidad va permite romper la relacion entre proyecto y  
desarrollador';
```

```
alter table SIGESPRO.DESARROLLADOR_PROYECTO  
    add constraint PK_DESARROLLADOR_PROYECTO primary key  
(CODIGO_DESARROLLADOR_PROYECTO);
```

```
/*=====*/  
/* Table: INSTITUCION */  
/*=====*/
```

```
create table SIGESPRO.INSTITUCION (  
    RUC                CHAR(13)          not null,  
    NOMBRE             VARCHAR(50)       not null,  
    DESCRIPCION        VARCHAR(500)      null,  
    DIRECCION          VARCHAR(250)      not null,  
    TELEFONO_01        VARCHAR(10)       null,  
    TELEFONO_02        VARCHAR(10)       null,  
    TELEFONO_03        VARCHAR(10)       null,  
    TIPO_INSTITUCION  NUMERIC(8)        not null  
);
```

```
comment on table SIGESPRO.INSTITUCION is  
'Esta entidad va a almacenar la informaci3n sobre las instituciones para las  
cuales se va a realizar el desarrollo del Sistema';
```

```
alter table SIGESPRO.INSTITUCION  
    add constraint PK_INSTITUCION primary key (RUC);
```

```
/*=====*/  
/* Table: PARAMETRO */  
/*=====*/
```

```
create table SIGESPRO.PARAMETRO (  
    CODIGO_PARAMETRO   NUMERIC(8,0)      not null,  
    CODIGO_CATALOGO    NUMERIC(8)        not null,  
    CODIGO_PROYECTO    NUMERIC(8,0)      not null  
);
```

```
comment on table SIGESPRO.PARAMETRO is  
'Esta entidad va a almacenar los parametros del sistema, en como dia no  
laborables, costo de recursos, horas de trabajo';
```

```
alter table SIGESPRO.PARAMETRO  
    add constraint PK_PARAMETRO primary key (CODIGO_PARAMETRO);
```

```
/*=====*/  
/* Table: PROYECTO */  
/*=====*/
```

```
create table SIGESPRO.PROYECTO (  
    CODIGO_PROYECTO    NUMERIC(8,0)      not null,  
    NOMBRE              VARCHAR(50)       not null,  
    DESCRIPCION         VARCHAR(500)      not null,  
    ESTADO_PROYECTO    CHAR(3)          not null,  
);
```

## SIGESPro - Sistema de Gestión y Seguimiento de Proyectos de Software

```
ESTADO          CHAR(3)          not null,  
INSTITUCION    CHAR(13)         not null  
);
```

```
comment on table SIGESPRO.PROYECTO is
```

```
'Esta entidad es la encargada de almacenar la informaci n de cada uno de  
los proyectos';
```

```
alter table SIGESPRO.PROYECTO  
  add constraint PK_PROYECTO primary key (CODIGO_PROYECTO);
```

```
/*=====*/  
/* Table: RECURSO */  
/*=====*/
```

```
create table SIGESPRO.RECURSO (  
  CODIGO_RECURSO    NUMERIC(8,0)    not null,  
  NOMBRE            VARCHAR(20)    not null,  
  DESCRIPCION       VARCHAR(100)   null,  
  URL               VARCHAR(100)   null,  
  ESTADO            CHAR(3)       not null,  
  REC_CODIGO_RECURSO NUMERIC(8,0)   null  
);
```

```
comment on table SIGESPRO.RECURSO is
```

```
'Esta entidad va a almacenar la informaci n sobre los recursos disponibles  
del sistema.';
```

```
alter table SIGESPRO.RECURSO  
  add constraint PK_RECURSO primary key (CODIGO_RECURSO);
```

```
/*=====*/  
/* Table: ROL */  
/*=====*/
```

```
create table SIGESPRO.ROL (  
  CODIGO_ROL        NUMERIC(8,0)    not null,  
  NOMBRE            VARCHAR(15)    not null,  
  DESCRIPCION       VARCHAR(100)   null,  
  ESTADO            CHAR(3)       not null  
);
```

```
comment on table SIGESPRO.ROL is
```

```
'Esta entidad es la encargada de almacenar la informaci n de los roles que va  
a manejar el sistema';
```

```
alter table SIGESPRO.ROL  
  add constraint PK_ROL primary key (CODIGO_ROL);
```

```
/*=====*/  
/* Table: ROL_RECURSO */  
/*=====*/
```

```
create table SIGESPRO.ROL_RECURSO (  
  CODIGO_ROL_RECURSO NUMERIC(8,0)    not null,  
  CODIGO_ROL         NUMERIC(8,0)    not null,  
  CODIGO_RECURSO     NUMERIC(8,0)    null  
);
```

```
comment on table SIGESPRO.ROL_RECURSO is
```



## SIGESPro - Sistema de Gestión y Seguimiento de Proyectos de Software

'Esta entidad es la encargada de alacernar las relaciones entre rol y el recurso disponible para dicho rol';

```
alter table SIGESPRO.ROL_RECURSO
  add constraint PK_ROL_RECURSO primary key (CODIGO_ROL_RECURSO);
```

```
/*=====*/
/* Table: TAREA */
/*=====*/
```

```
create table SIGESPRO.TAREA (
  CODIGO_TAREA          NUMERIC(8,0)          not null,
  NOMBRE                CHAR(50)             not null,
  DESCRIPCION           VARCHAR(500)         not null,
  TIEMPO                NUMERIC(3)           not null,
  FECHA_INICIO          TIMESTAMP            not null,
  FECHA_FIN             TIMESTAMP            not null,
  CODIGO_PROYECTO      NUMERIC(8,0)         not null,
  CODIGO_DESARROLLADOR_PROYECTO NUMERIC(8,0) null
);
```

```
comment on table SIGESPRO.TAREA is
```

'Esta entidad es la encargada de almacenar cada una de las tareas que van a formar parte del proyecto';

```
alter table SIGESPRO.TAREA
  add constraint PK_TAREA primary key (CODIGO_TAREA);
```

```
/*=====*/
/* Table: TIPO_CATALOGO */
/*=====*/
```

```
create table SIGESPRO.TIPO_CATALOGO (
  CODIGO_TIPO_CATALOGO NUMERIC(8,0)          not null,
  NOMBRE_TIPO_CATALOGO VARCHAR(50)           not null,
  DESCRIPCION_TIPO_CATALOGO VARCHAR(500)     null,
  ESTADO_TIPO_CATALOGO CHAR(15)             not null
);
```

```
comment on table SIGESPRO.TIPO_CATALOGO is
```

'Esta entidad va a almacenar los tipos de catalogos lo cual nos va a permitir filtrar cierta informacion y estandarizar un poco la misma como por ejemplo el tipo de institución'

```
alter table SIGESPRO.TIPO_CATALOGO
  add constraint PK_TIPO_CATALOGO primary key (CODIGO_TIPO_CATALOGO);
```

```
/*=====*/
/* Table: USUARIO */
/*=====*/
```

```
create table SIGESPRO.USUARIO (
  IDENTIFICACION_USUARIO VARCHAR(13)         not null,
  NOMBRE                  VARCHAR(100)        not null,
  MAIL                    VARCHAR(100)        not null
);
```

```
comment on table SIGESPRO.USUARIO is
```

'Esta entidad va a almacenar la informacion de los usuarios de cada sistema';

```

alter table SIGESPRO.USUARIO
  add constraint PK_USUARIO primary key (IDENTIFICACION_USUARIO);

/*=====*/
/* Table: USUARIO_PROYECTO */
/*=====*/
create table SIGESPRO.USUARIO_PROYECTO (
  CODIGO_USUARIO_PROYECTO NUMERIC(8,0)          not null,
  IDENTIFICACION_USUARIO VARCHAR(13)            not null,
  PRO_CODIGO_PROYECTO NUMERIC(8,0)              not null
);

comment on table SIGESPRO.USUARIO_PROYECTO is
'Esta entidad va a almacenar las relaciones entre usuarios, y proyectos ya
que un usuario puede tener varios proyectos y viceversa';

alter table SIGESPRO.USUARIO_PROYECTO
  add constraint PK_USUARIO_PROYECTO primary key
(CODIGO_USUARIO_PROYECTO);

alter table SIGESPRO.CATALOGO
  add constraint FK_CATALOGO_REF_CATAL_TIPO_CAT foreign key
(CODIGO_TIPO_CATALOGO)
  references SIGESPRO.TIPO_CATALOGO (CODIGO_TIPO_CATALOGO)
  on delete restrict on update restrict;

alter table SIGESPRO.CONTACTO
  add constraint FK_CONTACTO_REF_CONTA_INSTITUC foreign key (RUC)
  references SIGESPRO.INSTITUCION (RUC)
  on delete restrict on update restrict;

alter table SIGESPRO.DEPENDENCIA_TAREA
  add constraint FK_DEPENDEN_REFERENCE_TAREA_01 foreign key
(TAREA_DEPENDIENTE)
  references SIGESPRO.TAREA (CODIGO_TAREA)
  on delete restrict on update restrict;

alter table SIGESPRO.DEPENDENCIA_TAREA
  add constraint FK_DEPENDEN_REFERENCE_TAREA_02 foreign key (TAREA)
  references SIGESPRO.TAREA (CODIGO_TAREA)
  on delete restrict on update restrict;

alter table SIGESPRO.DESARROLLADOR
  add constraint FK_DESARROL_REFERENCE_CATALOGO foreign key
(CODIGO_CATALOGO)
  references SIGESPRO.CATALOGO (CODIGO_CATALOGO)
  on delete restrict on update restrict;

alter table SIGESPRO.DESARROLLADOR_PROYECTO
  add constraint FK_DESARROL_REFERENCE_PROYECTO foreign key
(CODIGO_PROYECTO)
  references SIGESPRO.PROYECTO (CODIGO_PROYECTO)
  on delete restrict on update restrict;

alter table SIGESPRO.DESARROLLADOR_PROYECTO
  add constraint FK_DESARROL_REFERENCE_DESARROL foreign key (USERNAME)

```

```
references SIGESPRO.DESARROLLADOR (USERNAME)
on delete restrict on update restrict;

alter table SIGESPRO.INSTITUCION
add constraint FK_INSTITUC_REF_INSTI_CATALOGO foreign key
(TIPO_INSTITUCION)
references SIGESPRO.CATALOGO (CODIGO_CATALOGO)
on delete restrict on update restrict;

alter table SIGESPRO.PARAMETRO
add constraint FK_PARAMETR_REFERENCE_CATALOGO foreign key
(CODIGO_CATALOGO)
references SIGESPRO.CATALOGO (CODIGO_CATALOGO)
on delete restrict on update restrict;

alter table SIGESPRO.PARAMETRO
add constraint FK_PARAMETR_REFERENCE_PROYECTO foreign key
(CODIGO_PROYECTO)
references SIGESPRO.PROYECTO (CODIGO_PROYECTO)
on delete restrict on update restrict;

alter table SIGESPRO.PROYECTO
add constraint FK_PROYECTO_REF_PROYE_INSTITUC foreign key (INSTITUCION)
references SIGESPRO.INSTITUCION (RUC)
on delete restrict on update restrict;

alter table SIGESPRO.RECURSO
add constraint FK_RECURSO_REF_RECUR_RECURSO foreign key
(REC_CODIGO_RECURSO)
references SIGESPRO.RECURSO (CODIGO_RECURSO)
on delete restrict on update restrict;

alter table SIGESPRO.ROL_RECURSO
add constraint FK_ROL_RECU_REFERENCE_RECURSO foreign key
(CODIGO_RECURSO)
references SIGESPRO.RECURSO (CODIGO_RECURSO)
on delete restrict on update restrict;

alter table SIGESPRO.ROL_RECURSO
add constraint FK_ROL_RECU_REF_ROL_R_ROL foreign key (CODIGO_ROL)
references SIGESPRO.ROL (CODIGO_ROL)
on delete restrict on update restrict;

alter table SIGESPRO.TAREA
add constraint FK_TAREA_REFERENCE_DESARROL foreign key
(CODIGO_DESARROLLADOR_PROYECTO)
references SIGESPRO.DESARROLLADOR_PROYECTO
(CODIGO_DESARROLLADOR_PROYECTO)
on delete restrict on update restrict;

alter table SIGESPRO.TAREA
add constraint FK_TAREA_REF_TAREA_PROYECTO foreign key (CODIGO_PROYECTO)
references SIGESPRO.PROYECTO (CODIGO_PROYECTO)
on delete restrict on update restrict;

alter table SIGESPRO.USUARIO_PROYECTO
```

```
    add constraint FK_USUARIO__REFERENCE_USUARIO foreign key
(IDENTIFICACION_USUARIO)
    references SIGESPRO.USUARIO (IDENTIFICACION_USUARIO)
    on delete restrict on update restrict;

alter table SIGESPRO.USUARIO_PROYECTO
    add constraint FK_USUARIO__REFERENCE_PROYECTO foreign key
(PRO_CODIGO_PROYECTO)
    references SIGESPRO.PROYECTO (CODIGO_PROYECTO)
    on delete restrict on update restrict;
```

## Manual de Integración Jasig CAS con LDAP

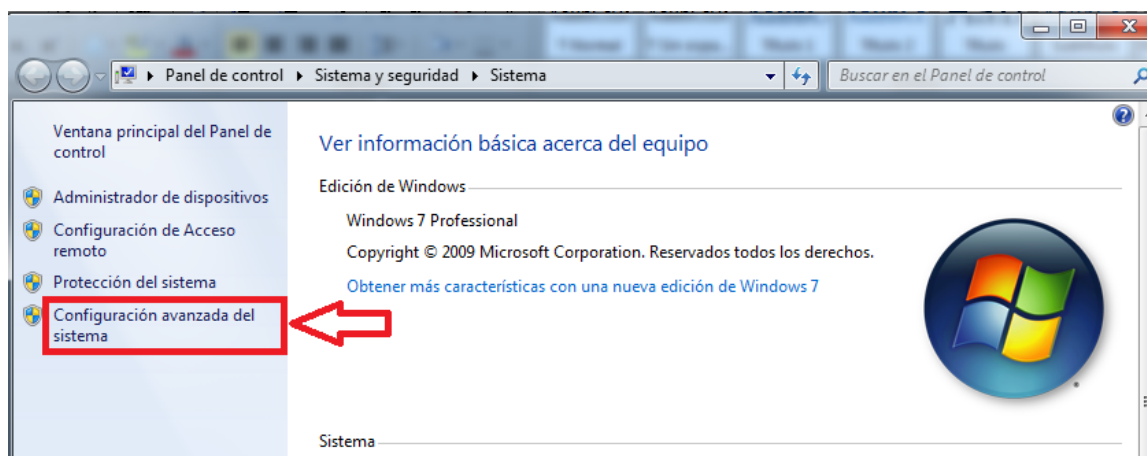
Para la integración de JASIG CAS con nuestro servidor LDAP es necesario

- Java JDK 1.6
- apache-maven-2.2.1
- cas-server-3.4.11

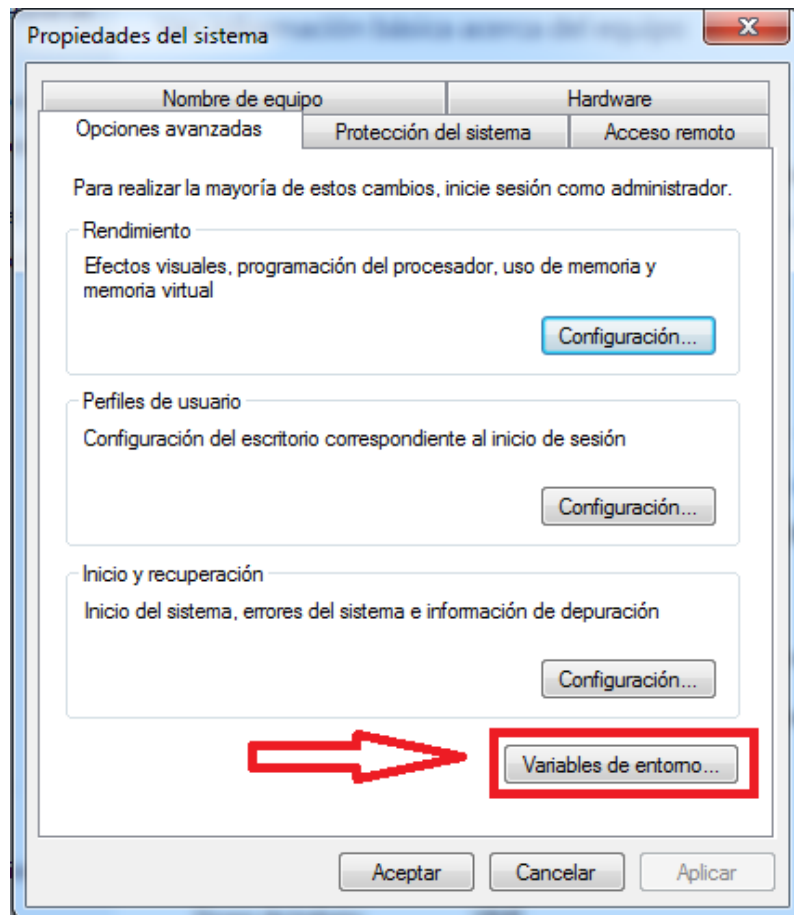
### Instalación y configuración de JDK

#### Windows

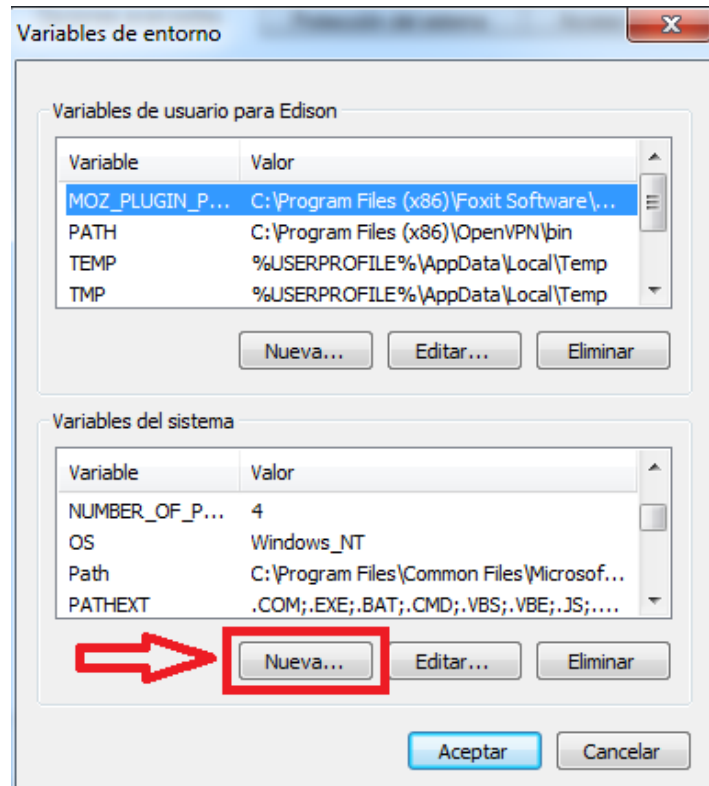
1. Descargamos la distribución de JDK correspondiente para nuestro sistema operativo que en este caso será jdk-6u18-windows-i586.exe
2. Instalamos el JDK descargado.
3. Para añadir la variable de entorno JAVA\_HOME, vamos a **Inicio ->Equipo**, damos clic derecho sobre equipo y damos clic en **Propiedades** si nos abre una ventana similar a la siguiente:



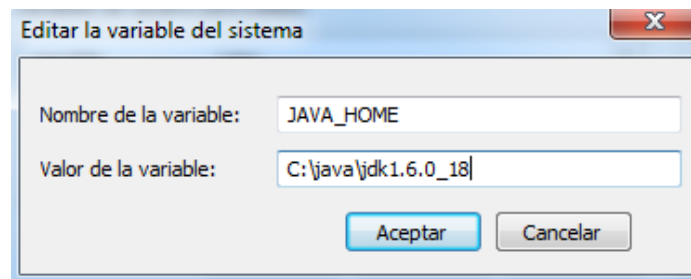
4. Damos clic en **Configuración Avanzada del Sistema**, y se abre la siguiente ventana.



5. Damos clic en el botón **Variables de entorno**, y se presenta la siguiente pantalla.



6. Damos clic en el botón **Nueva** y se abre una ventana, en la cual tenemos que ingresar la información de nuestra variable de entorno como se muestra a continuación.



Donde el campo *Nombre de la variable* siempre debe ser JAVA\_HOME, y el campo *Valor de variable* debe ser la ruta donde instalamos nuestro JDK.

7. Damos clic en **Aceptar** y listo tenemos configurada nuestra variable JAVA\_HOME.

## Linux (Ubuntu 11.10)

1. Descargamos la distribución de JDK correspondiente para nuestro sistema operativo que en este caso será *jdk-6u31-linux-i586.bin*.
2. Copiamos en el directorio que prefiramos.
3. Abrimos un terminal y mediante el comando `cd` nos dirigimos al directorio en el cual copiamos el archivo *jdk-6u31-linux-i586.bin*
4. Para instalar java JDK ejecutamos el siguiente comando.

```
sudo ./jdk-6u31-linux-i586.bin
```

5. Al terminar exitosamente la ejecución del comando creamos una carpeta para java con el siguiente comando.

```
sudo mkdir /usr/lib/jvm
```

6. Ahora movemos nuestro JDK a la carpeta que acabamos de crear, con el siguiente comando.

```
sudo mv jdk1.6.0_31/ /usr/lib/jvm/jdk1.6.0_31
```

7. Editamos el archivo ***environment***, para colocar nuestra variable de entorno y agregar a la variable de entorno ***PATH*** la ruta de nuestro JDK. Para lo cual ejecutamos el siguiente comando.

```
sudo vi /etc/environment
```

Y se nos abre el editor ***VI*** como se muestra en la siguiente pantalla.

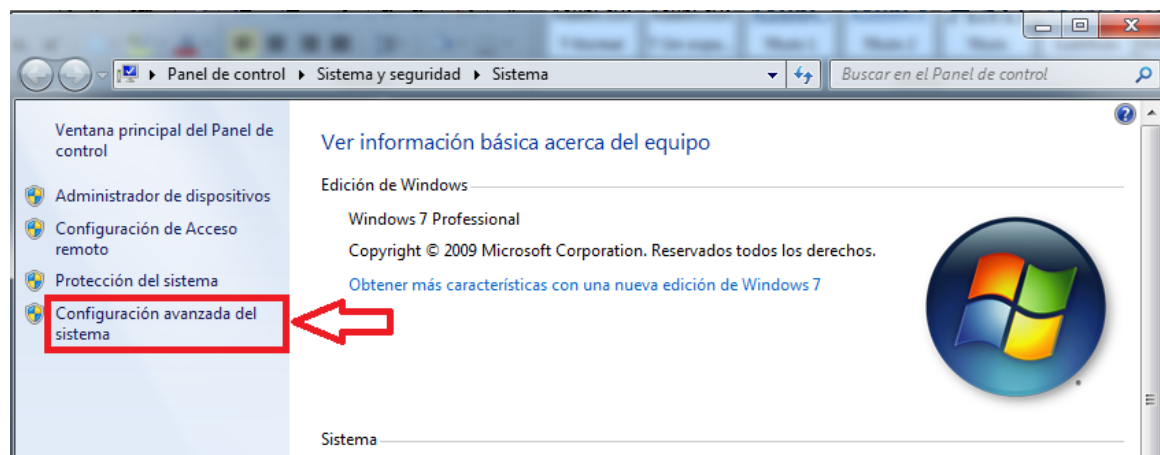




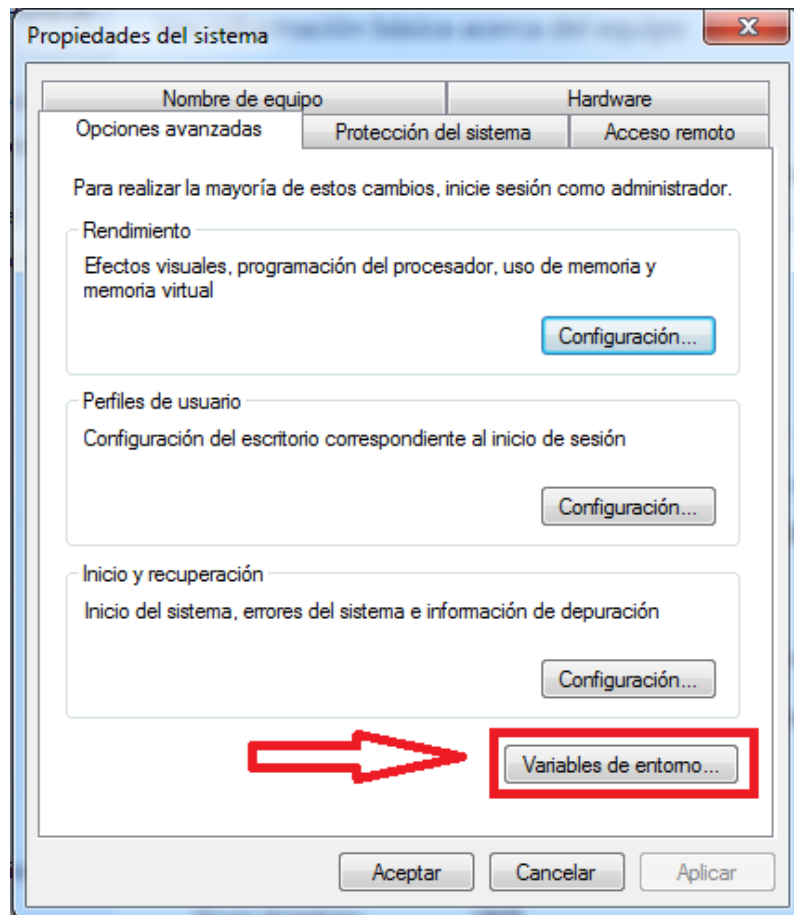
## Instalación y configuración de apache maven

### Windows

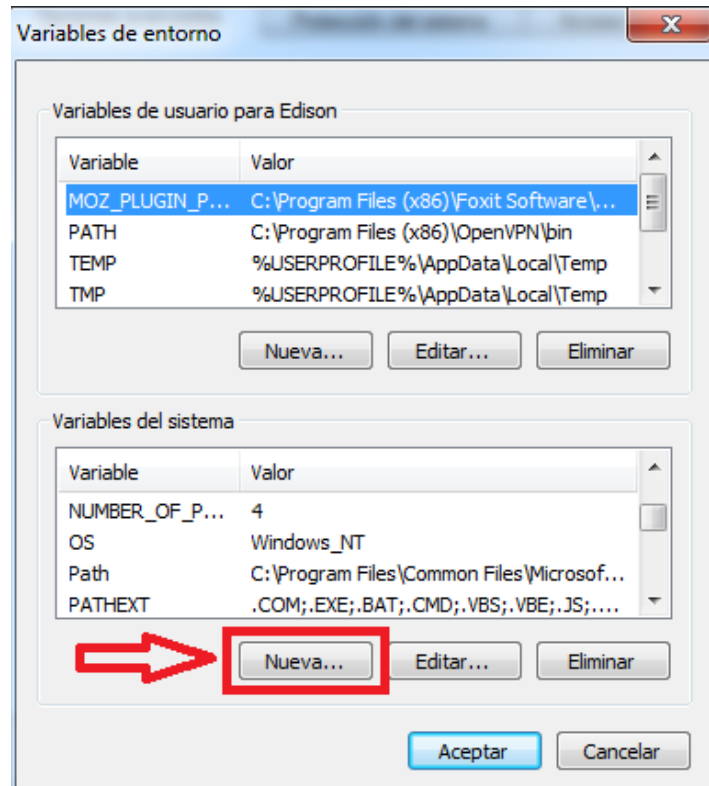
1. Descargamos la versión de maven que necesitamos desde la página de apache. La versión de maven que descarguemos es independiente del sistema operativo ya en el comprimido viene los archivos BAT para Windows y los SH para Linux. Para nuestro caso vamos a descargar la versión **apache-maven-2.2.1**.
2. Una vez descargado el archivo lo descomprimos en el directorio que sea de nuestro agrado.
3. Para añadir la variable de entorno MVN\_HOME, vamos a **Inicio ->Equipo**, damos clic derecho sobre equipo y damos clic en **Propiedades** si nos abre una ventana similar a la siguiente:



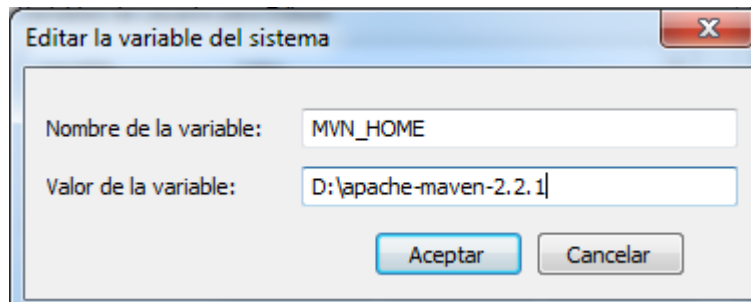
4. Damos clic en **Configuración Avanzada del Sistema**, y se abre la siguiente ventana.



5. Damos clic en el botón **Variables de entorno**, y se presenta la siguiente pantalla.

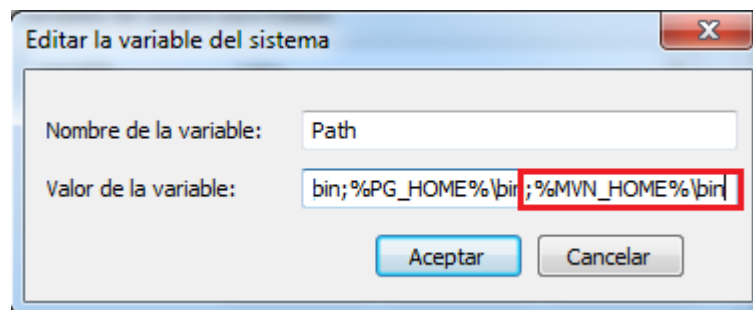


6. Damos clic en el botón **Nueva** y se abre una ventana, en la cual tenemos que ingresar la información de nuestra variable de entorno como se muestra a continuación.



Donde el campo *Nombre de la variable* no necesariamente debe ser siempre debe ser MVN\_HOME puede ser cualquier nombre que sea de nuestro agrado aunque por estándar es recomendable utilizar dicho nombre, y el campo *Valor de variable* debe ser la ruta donde descomprimos nuestra versión de maven.

7. Editamos la variable de entorno **PATH**, y al final añadimos la capeta bin de nuestra variable de entorno de la siguiente manera.



Lo que hicimos es añadir nuestra variable de entorno de maven a la variable de entorno PATH, con la finalidad de facilitar su ejecución. Para lo cual al final del valor de la variable añadimos ; % el nombre de nuestra variable que para este caso es MVN\_HOME % y la ruta del directorio BIN.

8. Damos clic en **Aceptar** y para probar si nuestra configuración ha sido exitosa abrimos un consola y colocamos en comando mvn. Y nos debe devolver una salida similar a la que se muestra en la siguiente gráfica.

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Versión 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.

C:\Users\Edison>mvn
[INFO] Scanning for projects...
[INFO] -----
[ERROR] BUILD FAILURE
[INFO] -----
[INFO]

You must specify at least one goal or lifecycle phase to perform build steps.
The following list illustrates some commonly used build commands:

    mvn clean
        Deletes any build output (e.g. class files or JARs).
    mvn test
        Runs the unit tests for the project.
    mvn install
        Copies the project artifacts into your local repository.
    mvn deploy
        Copies the project artifacts into the remote repository.
    mvn site
        Creates project documentation (e.g. reports or Javadoc).

Please see
http://maven.apache.org/guides/introduction/introduction-to-the-lifecycle.html
for a complete description of available lifecycle phases.

Use "mvn --help" to show general usage information about Maven's command line.

[INFO] -----
[INFO] For more information, run Maven with the -e switch
[INFO] -----
[INFO] Total time: 1 second
[INFO] Finished at: Sun Feb 19 20:15:49 COT 2012
[INFO] Final Memory: 1M/15M
[INFO] -----
C:\Users\Edison>
```

Recuerde que la consola se debe abrir después de haber configurado la variable, ya que si ejecutamos el comando en una consola que ya la teníamos abierta no va a funcionar.

### Linux (Ubuntu 11.10)

1. Descargamos la versión de maven que necesitamos desde la página de apache. La versión de maven que descarguemos es independiente del sistema operativo ya en el comprimido viene los archivos BAT para Windows y los SH para Linux. Para nuestro caso vamos a descargar la versión **apache-maven-2.2.1**.

2. Una vez descargado el archivo lo copiamos en cualquier directorio que prefiramos. Abrimos una terminal y mediante el comando `cd` nos ubicamos en el directorio que copiamos el archivo.

3. Y utilizamos el siguiente comando para descomprimir el archivo.

```
unzip apache-maven-2.2.1.zip
```

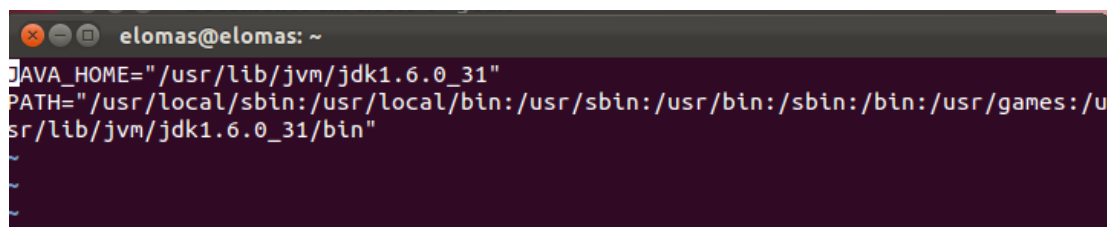
4. Movemos la carpeta que descomprimimos con el comando anterior a la carpeta `/usr/lib` con el siguiente comando.

```
sudo mv apache-maven-2.2.1 /usr/lib/
```

5. Editamos el archivo ***environment***, para colocar nuestra variable de entorno y agregar a la variable de entorno ***PATH*** la ruta de nuestro apache maven. Para lo cual ejecutamos el siguiente comando.

```
sudo vi /etc/environment
```

Y se nos abre el editor ***VI*** como se muestra en la siguiente pantalla.

A screenshot of a terminal window with a dark background. The prompt is 'elomas@elomas: ~'. The terminal displays the following environment variables:

```
JAVA_HOME="/usr/lib/jvm/jdk1.6.0_31"  
PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/lib/jvm/jdk1.6.0_31/bin"
```

Editamos este archivo y agregamos la variable `MVN_HOME` y la ruta del maven a la variable `PATH`. El archivo quedara de la siguiente manera.

```
elomas@elomas: ~/Escritorio
JAVA_HOME="/usr/lib/jvm/jdk1.6.0_31"
MVN_HOME="/usr/lib/apache-maven-2.2.1"
PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/u
sr/lib/jvm/jdk1.6.0_31/bin:/usr/lib/apache-maven-2.2.1/bin"
```

6. Guardamos el archivo y reiniciamos la sesión, y listo.
7. Para probar la configuración abrimos una terminal y ejecutamos el siguiente comando y nos debería mostrar una salida similar a la siguiente.

```
elomas@elomas: ~
elomas@elomas:~$ mvn
[INFO] Scanning for projects...
[INFO] -----
[ERROR] BUILD FAILURE
[INFO] -----
[INFO]
You must specify at least one goal or lifecycle phase to perform build steps.
The following list illustrates some commonly used build commands:

  mvn clean
    Deletes any build output (e.g. class files or JARs).
  mvn test
    Runs the unit tests for the project.
  mvn install
    Copies the project artifacts into your local repository.
  mvn deploy
    Copies the project artifacts into the remote repository.
  mvn site
    Creates project documentation (e.g. reports or Javadoc).

Please see
http://maven.apache.org/guides/introduction/introduction-to-the-lifecycle.html
for a complete description of available lifecycle phases.

Use "mvn --help" to show general usage information about Maven's command line.

[INFO] -----
[INFO] For more information, run Maven with the -e switch
[INFO] -----
[INFO] Total time: < 1 second
[INFO] Finished at: Sun Feb 19 23:29:16 ECT 2012
```



## Configuración, compilación e instalación de CAS-SERVER-WEBAPP

1. Descomprimos el archivo cas-server-3.4.11 que descargamos desde la página de Jasig en cualquier directorio que deseemos.
2. Abrimos el archivo cas-server-webapp\pom.xml y añadimos la siguiente dependencia:

```
<dependency>
```

```
<groupId>${project.groupId}</groupId>
```

```
<artifactId>cas-server-support-ldap</artifactId>
```

```
<version>${project.version}</version>
```

```
</dependency>
```

Como se muestra en la siguiente gráfica y guardamos el archivo.

```
<version>2.7.3</version>
<scope>runtime</scope>
</dependency>

<dependency>
  <groupId>org.hibernate</groupId>
  <artifactId>hibernate-validator</artifactId>
  <scope>runtime</scope>
</dependency>

<!--LDAP -->
<dependency>
  <groupId>${project.groupId}</groupId>
  <artifactId>cas-server-support-ldap</artifactId>
  <version>${project.version}</version>
</dependency>
</dependencies>

<build>
  <plugins>
```



3. Abrimos el archivo cas-server-3.4.11\cas-server-webapp\src\main\webapp\WEB-INF\deployerConfigContext.xml

4. Eliminamos o comentamos la siguiente línea ubicada en la línea número 92 aproximadamente.

```
<bean class="org.jasig.cas.authentication.handler.support.SimpleTestUsernamePasswordAuthenticationHandler"/>
```

Y añadimos las siguientes líneas.

```
<bean class="org.jasig.cas.adapters.ldap.BindLdapAuthenticationHandler">  
  <property name="filter" value="uid=%u" />  
  <property name="searchBase" value="ou=VimeworksLdap" />  
  <property name="contextSource" ref="contextSource" />  
</bean>
```

La propiedad **filter** es la propiedad mediante la cual se va a filtrar a los usuarios para su autenticación, en este caso lo haremos mediante el **uid** y el parámetro **%u** es el que va a contener el nombre del usuario, el cual va a ser insertado posteriormente mediante la aplicación web que va a permitir la autenticación.

La propiedad **searchBase** es el lugar en donde vamos a buscar para nuestro caso vamos a buscar en toda la unidad organizativa **VimeworksLdap**.

La propiedad **contextSource** es el contexto de nuestra conexión con el servidor LDAP, la cual la vamos a configurar a continuación.

Y agregamos un nuevo **bean** a nivel de la raíz aproximadamente en la línea 103 con la siguiente información:

```
<bean id="contextSource"  
  class="org.springframework.ldap.core.support.LdapContextSource">  
  <property name="pooled" value="false"/>  
  <property name="urls">  
    <list>  
      <value>ldap://ldap.vimeworks.com/</value>  
      <value>ldap://ldap02.vimeworks.com/</value>
```

```
        </list>
    </property>
    <property name="userDn" value="uid=admin,ou=system"/>
    <property name="password" value="ldapVimeworks"/>
    <property name="baseEnvironmentProperties">
        <map>
            <entry>
                <key>
                    <value>java.naming.security.authentication</value>
                </key>
                <value>simple</value>
            </entry>
        </map>
    </property>
</bean>
```

En este **bean** lo que hacemos es configurar la conexión a los servidores, para esta configuración en específico se realizó para un pool de servidores.

Además configuramos el nombre del usuario y la contraseña del usuario que nos va a permitir realizar las consultas en nuestro servidor LDAP.

Nos quedaría algo de la siguiente manera:

```

<!-- <bean
class="org.jasig.cas.authentication.handler.support.SimpleTestUsernamePasswordAuthenticationHandler" /> -->
<bean class="org.jasig.cas.adaptors.ldap.BindLdapAuthenticationHandler" >
  <property name="filter" value="uid=%u" />
  <property name="searchBase" value="ou=VimeworksLdap" />
  <property name="contextSource" ref="contextSource" />
</bean>
</list>
</property>
</bean>

<bean id="contextSource" class="org.springframework.ldap.core.support.LdapContextSource">
  <property name="pooled" value="false"/>
  <property name="urls">
    <list>
      <value>ldap://ldap.vimeworks.com/</value>
    </list>
  </property>
  <property name="userDn" value="uid=admin,ou=system"/>
  <property name="password" value="ldapVimeworks"/>
  <property name="baseEnvironmentProperties">
    <map>
      <entry>
        <key>
          <value>java.naming.security.authentication</value>
        </key>
        <value>simple</value>
      </entry>
    </map>
  </property>
</bean>

```

5. Abrimos una consola y nos dirigimos hasta la ruta donde descomprimos el cas-server que descargamos.
6. Colocamos los comando `cd cas-server-webapp`
7. El comando `mvn clean install package`. Una vez terminada la ejecución del comendo tenemos listo el `cas-server-webapp` para ser deployado en el servidor.
8. Verificamos en el directorio `cas-server-3.4.11\cas-server-webapp\target` se nos crea un archivo llamado **cas.war**. Copiamos este archivo en el directorio `apache-tomcat-6.0.20\webapps` y corremos el nuestro servidor.
9. Verificamos si se levantó nuestro cas webapp mediante el navegador. Y verificamos su funcionalidad.

Introduzca su NetID y Contraseña.

NetID:  
elomas

Contraseña:  
●●●●●●

Avisarme antes de abrir sesión en otros sitios.

INICIAR SESIÓN | limpiar

Por razones de seguridad, por favor cierre su sesión y su navegador web cuando haya terminado de acceder a los servicios que requieren autenticación.

Languages:  
[English](#) | [Spanish](#) | [French](#) | [Russian](#) | [Nederlands](#) | [Svenskt](#) | [Italiano](#) | [Urdu](#) | [Chinese \(Simplified\)](#) | [Deutsch](#) | [Japanese](#) | [Croatian](#) | [Czech](#) | [Slovenian](#) | [Catalan](#) | [Macedonian](#) | [Polish](#)

Copyright © 2005 - 2010 Jasig, Inc. All rights reserved.  
Powered by [Jasig Central Authentication Service 3.4.11](#)

Si ocurre un error de autenticación debería presentarnos un error similar al siguiente.

Introduzca su NetID y Contraseña.

NetID:  
elomas

Contraseña:

Avisarme antes de abrir sesión en otros sitios.

INICIAR SESIÓN | limpiar

Por razones de seguridad, por favor cierre su sesión y su navegador web cuando haya terminado de acceder a los servicios que requieren autenticación.

Languages:  
[English](#) | [Spanish](#) | [French](#) | [Russian](#) | [Nederlands](#) | [Svenskt](#) | [Italiano](#) | [Urdu](#) | [Chinese \(Simplified\)](#) | [Deutsch](#) | [Japanese](#) | [Croatian](#) | [Czech](#) | [Slovenian](#) | [Catalan](#) | [Macedonian](#) | [Polish](#)

No se puede determinar que las credenciales proporcionadas sean auténticas.

Copyright © 2005 - 2010 Jasig, Inc. All rights reserved.  
Powered by [Jasig Central Authentication Service 3.4.11](#)

Si las credenciales son correctas se presenta un mensaje similar al siguiente:



Si se presentara un mensaje diferente a los indicados debemos revisar el log de la aplicación y revisar la configuración.

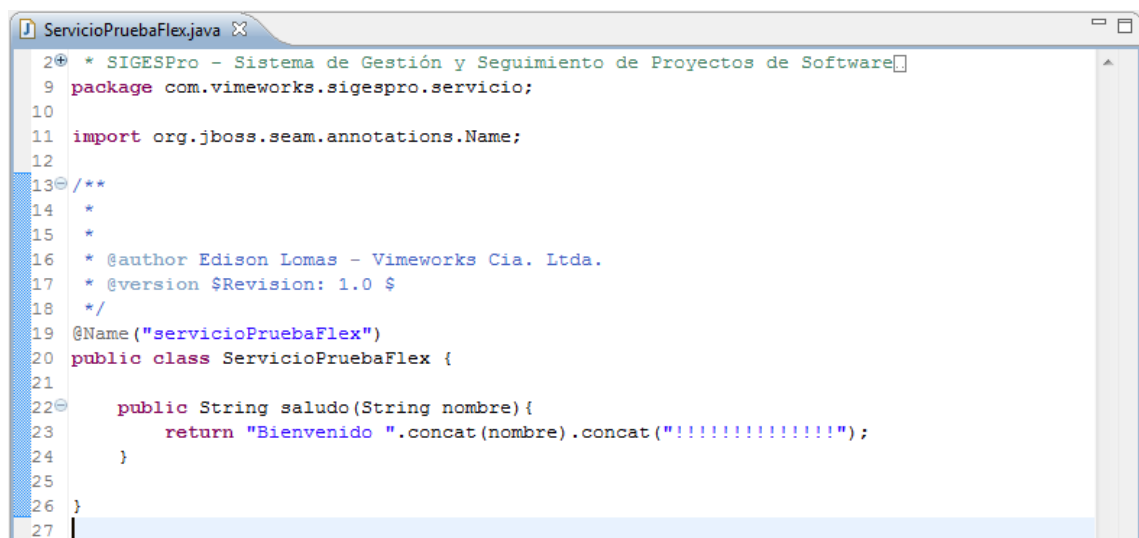
## Manual de Integración Aplicación JBoss SEAM con Adobe Flex

Para la integración de una aplicación desarrollada con el framework de JBoss Seam con Adobe Flex, necesitamos lo siguiente:

- Adobe Flex 3.2
- Flex Builder 3
- Flamingo 2.1
- amf-serializer-2.1
- scannotation-1.0.2
- Proyecto JBoss Seam configurado

### Creación de un Proyecto Flex

1. Abrimos la carpeta WEB-INF de nuestra aplicación.
2. Creamos un servicio de prueba.



```
2+ * SIGESPro - Sistema de Gestión y Seguimiento de Proyectos de Software
9 package com.vimeworks.sigepro.servicio;
10
11 import org.jboss.seam.annotations.Name;
12
13 /**
14 *
15 *
16 * @author Edison Lomas - Vimeworks Cia. Ltda.
17 * @version $Revision: 1.0 $
18 */
19 @Name("servicioPruebaFlex")
20 public class ServicioPruebaFlex {
21
22     public String saludo(String nombre) {
23         return "Bienvenido ".concat(nombre).concat("!!!!!!!!!!!!!!!!");
24     }
25
26 }
27
```

3. Creamos una carpeta llamada flex, y dentro de esta carpeta creamos un archivo XML llamado services-config.xml, con el siguiente contenido:

```

<?xmlversion="1.0"encoding="UTF-8"?>

<services-config>
  <services>
    <service
      id="servicio-seam"
      class="flex.messaging.services.RemotingService"
      messageTypes="flex.messaging.messages.RemotingMessage">
      <destinationid="servicioPruebaFlex"/>
      <default-channels>
      <channel ref="seam-amf"/>
      </default-channels>
    </service>
  </services>
  <channels>
    <channel-definition id="seam-amf"
      class="mx.messaging.channels.AMFChannel">
      <endpoint
        uri="http://{server.name}:{server.port} /SIGESPro/seam/resource/amf"
        class="flex.messaging.endpoints.AMFEndpoint"/>
    </channel-definition>
  </channels>
</services-config>

```

La estructura anterior es la que debe tener nuestro archivo de configuración de servicios.

Dentro de la etiqueta **service** debemos colocar todos los servicios que deseemos exponer.

El id del servicio puede ser cualquier nombre que deseemos y dentro de esta etiqueta tenemos una etiqueta llamada **destination** en la cual vamos a colocar en la propiedad **id** el nombre de nuestro recurso ya sea este un bean o un servicio. Por cada servicio que deseemos exponer para flex debemos crear una etiqueta **destination**.



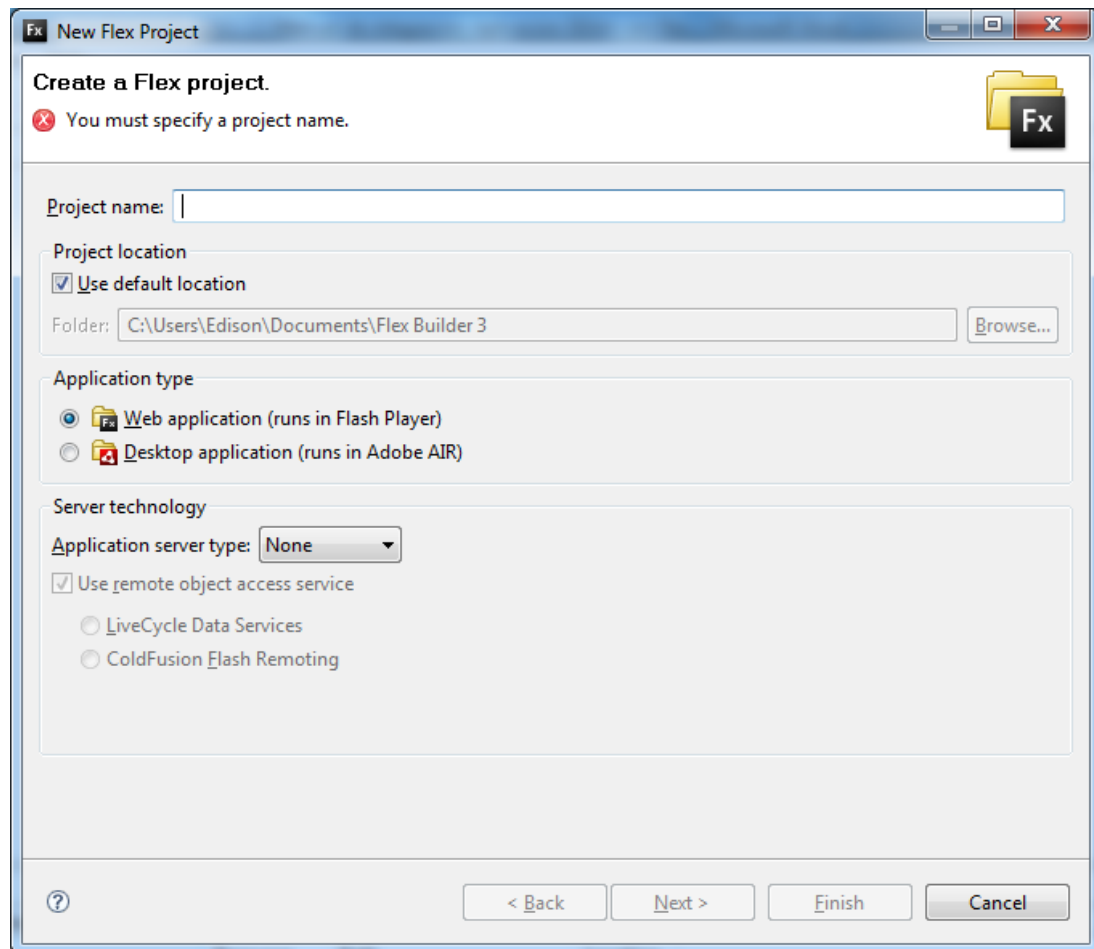
Y dentro de la etiqueta **chanel**s creamos el canal de conexión de nuestros servicios. En la etiqueta **endpoint** en la propiedad Uri colocamos la URL de nuestra aplicación y añadimos al final **seam/resource/amf** ya que por esta dirección van a exponerse nuestros servicios.

4. Agregamos las librerías necesarias en la carpeta **lib** que se encuentra en el interior de nuestra carpeta **WEB-INF** las librerías necesarias son las siguientes:

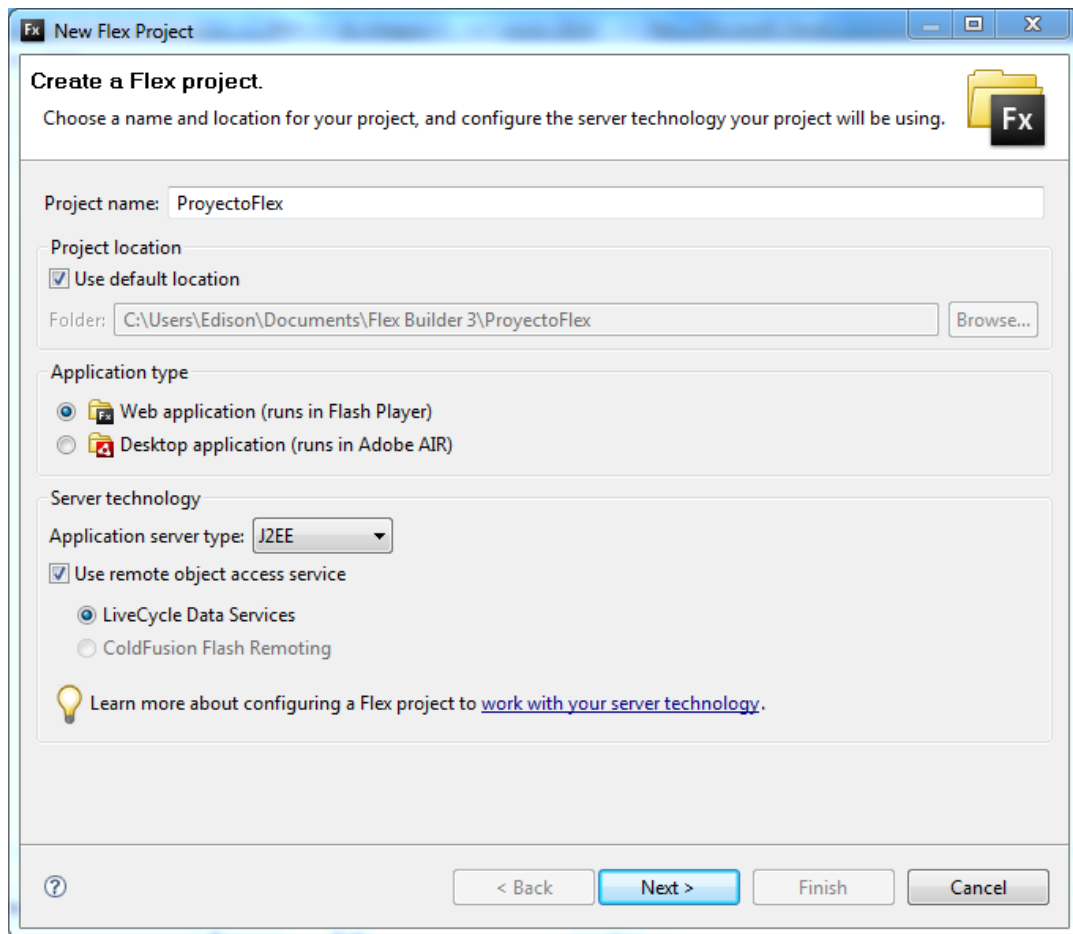
- flamingo-service-2.1.jar
- flamingo-services-common-2.1.jar
- amf-serializer-2.1.jar
- scannotation-1.0.2.jar

Y listo nuestra aplicación J2EE está lista para presentar los servicios para que sean consumidos por flex.

5. Creamos nuestro proyecto flex, para lo cual abrimos Flex Builder, damos clic en **File ->New ->Flex Project** y nos presenta la siguiente pantalla.

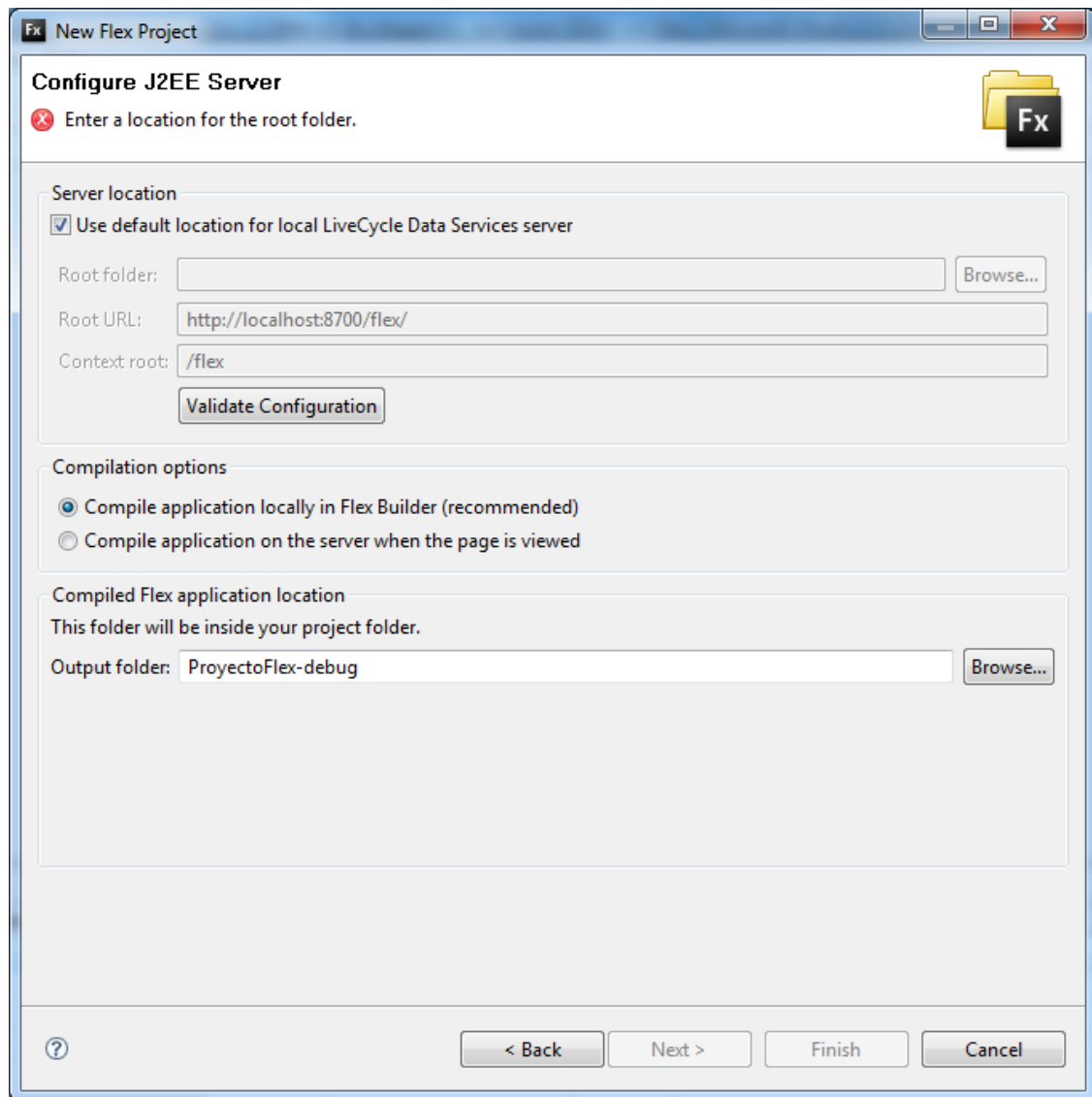


6. En la ventana ponemos el nombre del proyecto, que para este ejemplo será **Proyecto Flex**. En la opción **Application Type** seleccionamos la opción **Web Application**, y en la sección **Server technology** seleccionamos **J2EE** en la opción **Application Server type**, y seleccionamos la opción **Use remote object Access service** de la siguiente manera.



Y damos clic en **Next>**.

7. Al dar clic en **Next>** nos presenta la siguiente pantalla.



Deseleccionamos la opción **Use default location for local LiveCycle Data Service server**, para colocar manualmente la configuración del servidor.

En el campo **Root Folder** seleccionamos la carpeta raíz del contenido web de nuestro proyecto.

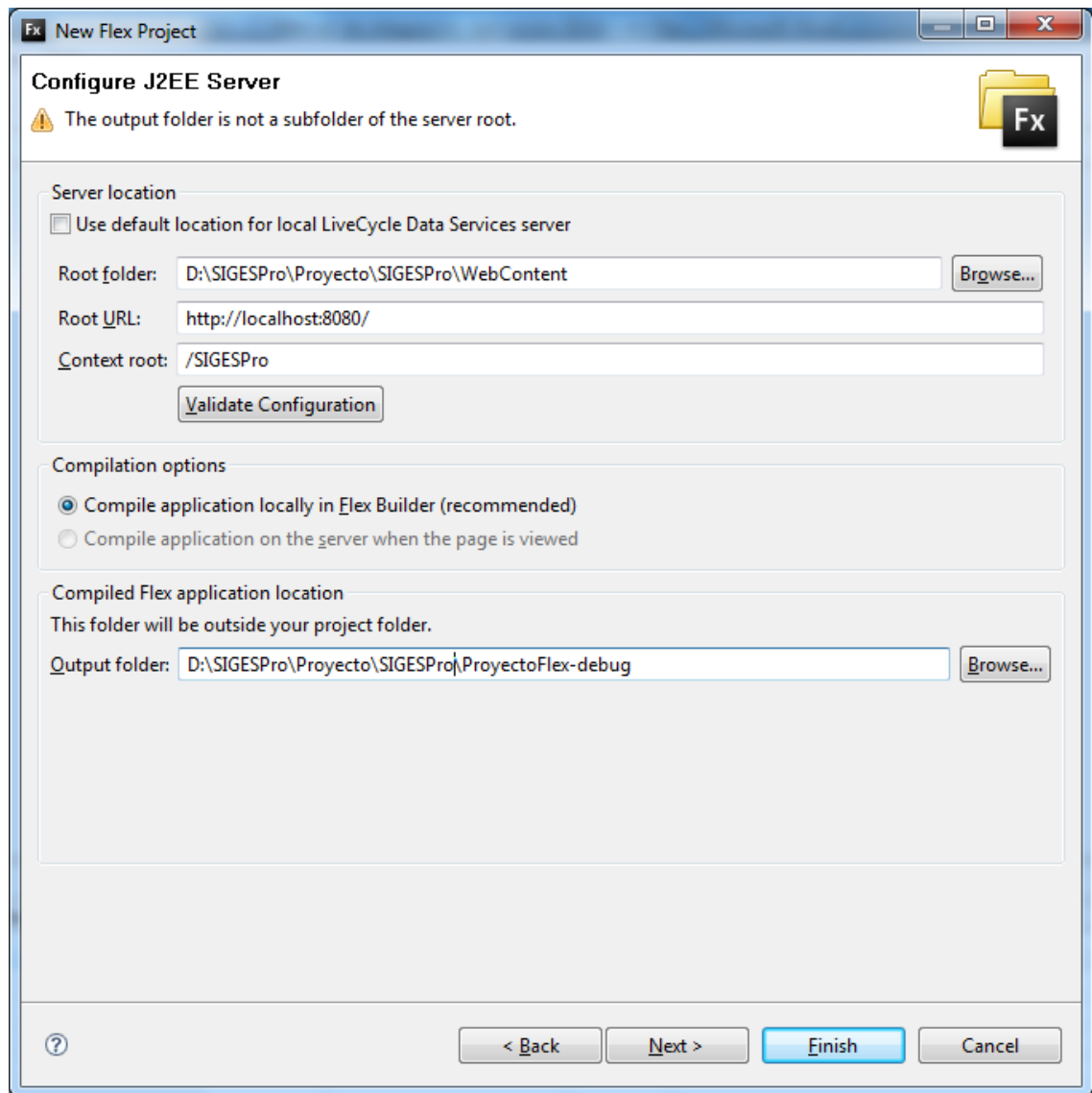
En el campo **Root Url** colocamos el URL de acceso a nuestro servidor.

En el campo **Context root** colocamos el context root de nuestra aplicación. Damos clic en el botón **Validate Configuration** para validar la configuración, si da algún error debemos revisar que exista el archivo de configuración de servicios flex services-config.xml dentro de la carpeta flex en la carpeta WEB-INF.

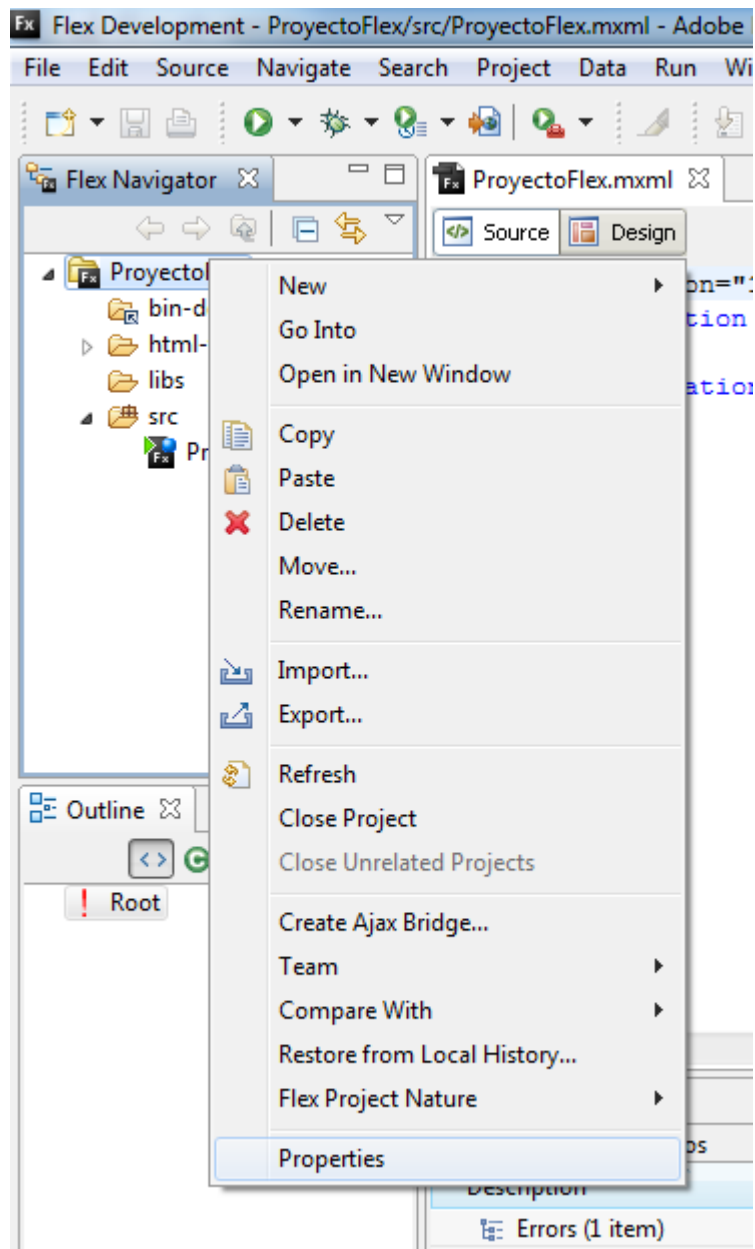
En la sección **Compilation option** seleccionamos la opción **Compile application locally in Flex Builder**.

En la sección **Compiled Flex application location** dejamos el directorio que nos indica por default solo eliminamos el WebContent o si deseamos la cambiamos no hay ningún problema.

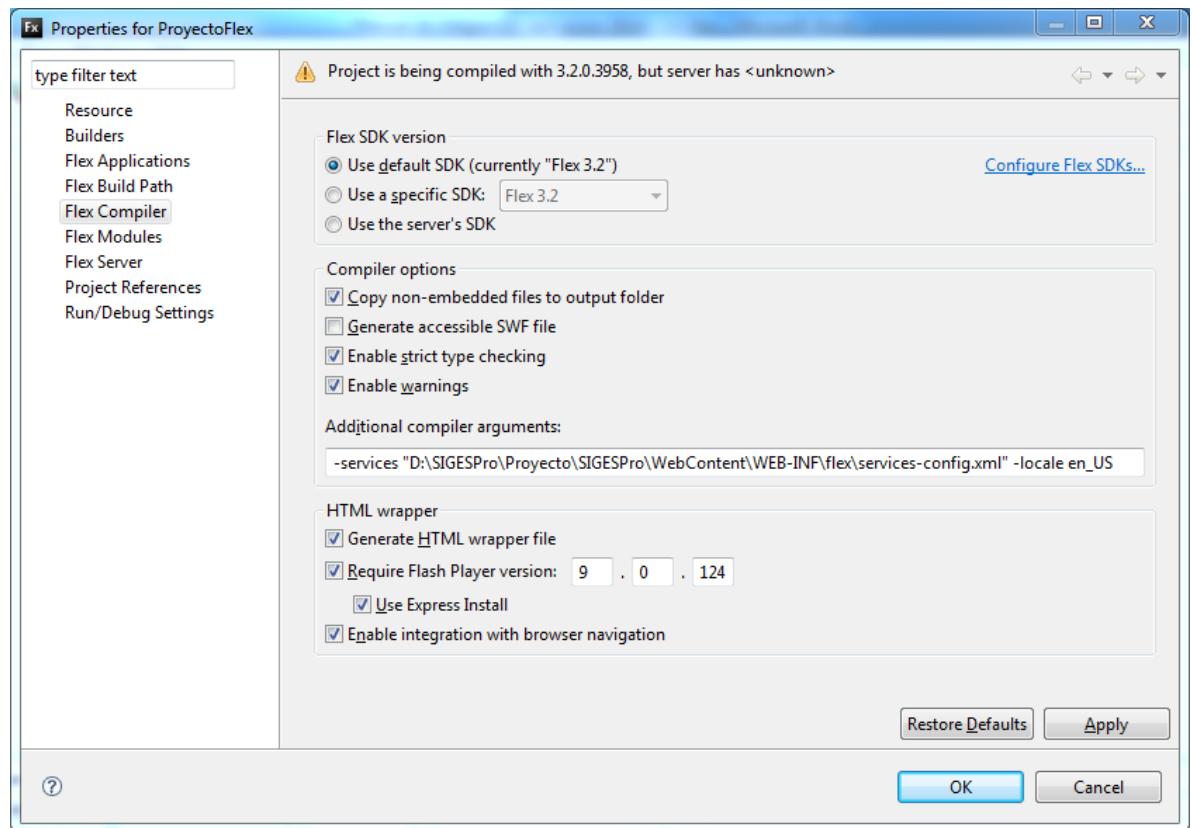
La ventana nos quedara de la siguiente manera.



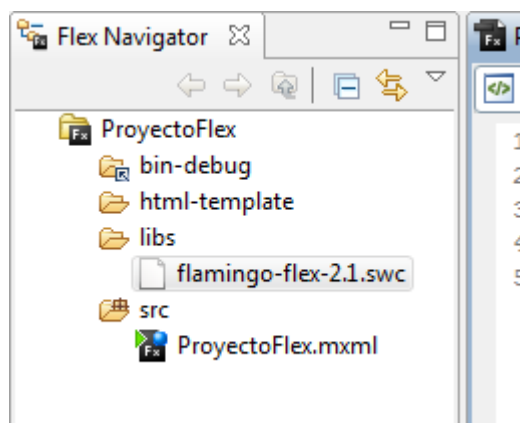
8. Al dar clic en **Finish** y tenemos creado nuestro proyecto.
9. Damos clic derecho sobre el proyecto y damos clic en propiedades.



10. Al dar clic en propiedades se nos presenta la pantalla de propiedades y nos dirigimos a la sección **Flex Compiler** y seleccionamos la opción **Use default SDK** de la sección **Flex SDK version** y damos clic en el botón **Apply** y luego **OK**.



11. Agregamos la librería de flamingo para lo cual copiamos el archivo **flamingo-flex-2.1.swc** en la carpeta **libs** de nuestro proyecto.





12. Para probar nuestra configuración lo primero que hacemos es agregar las librerías de flamingo `xmlns:flamingo="com.exadel.flamingo.flex.components.flamingo.*"` a nuestro componente que para este caso es nuestro `mx:Application`, de la siguiente manera.

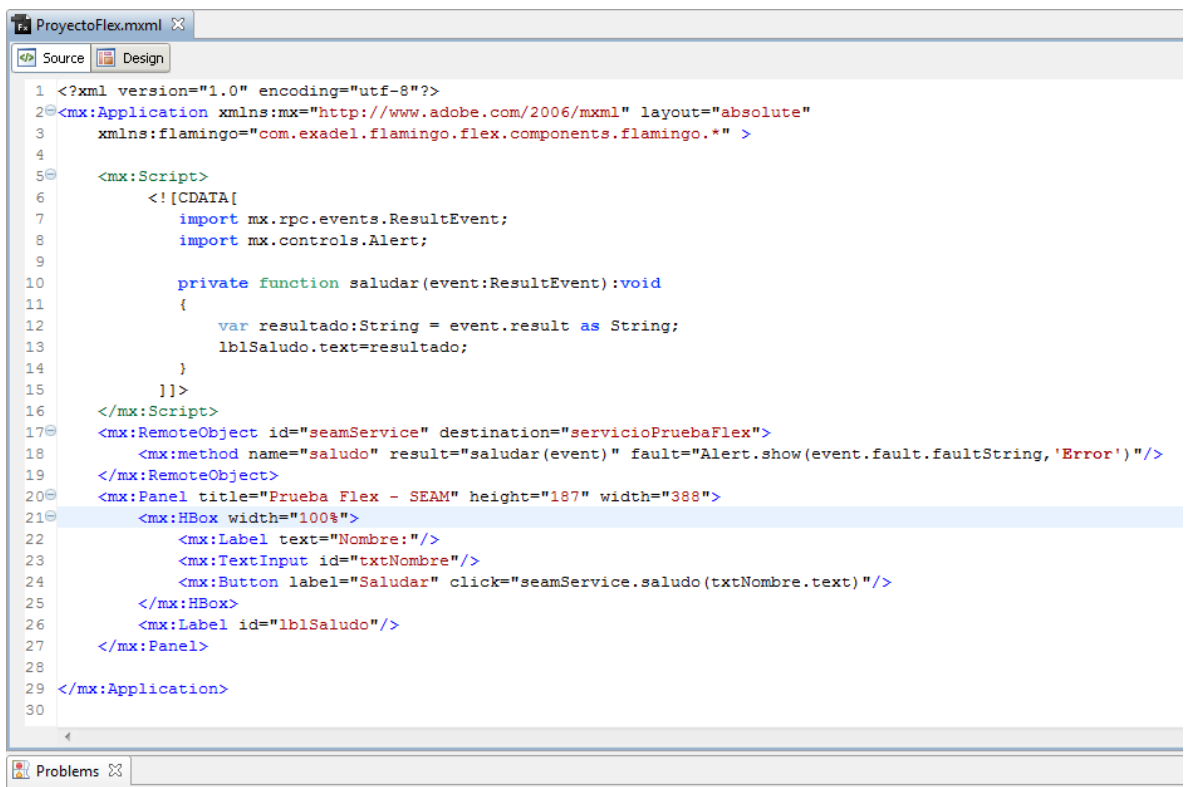


```

1 <?xml version="1.0" encoding="utf-8"?>
2 <mx:Application xmlns:mx="http://www.adobe.com/2006/mxml" layout="absolute"
3   xmlns:flamingo="com.exadel.flamingo.flex.components.flamingo.*" >
4
5 </mx:Application>
6

```

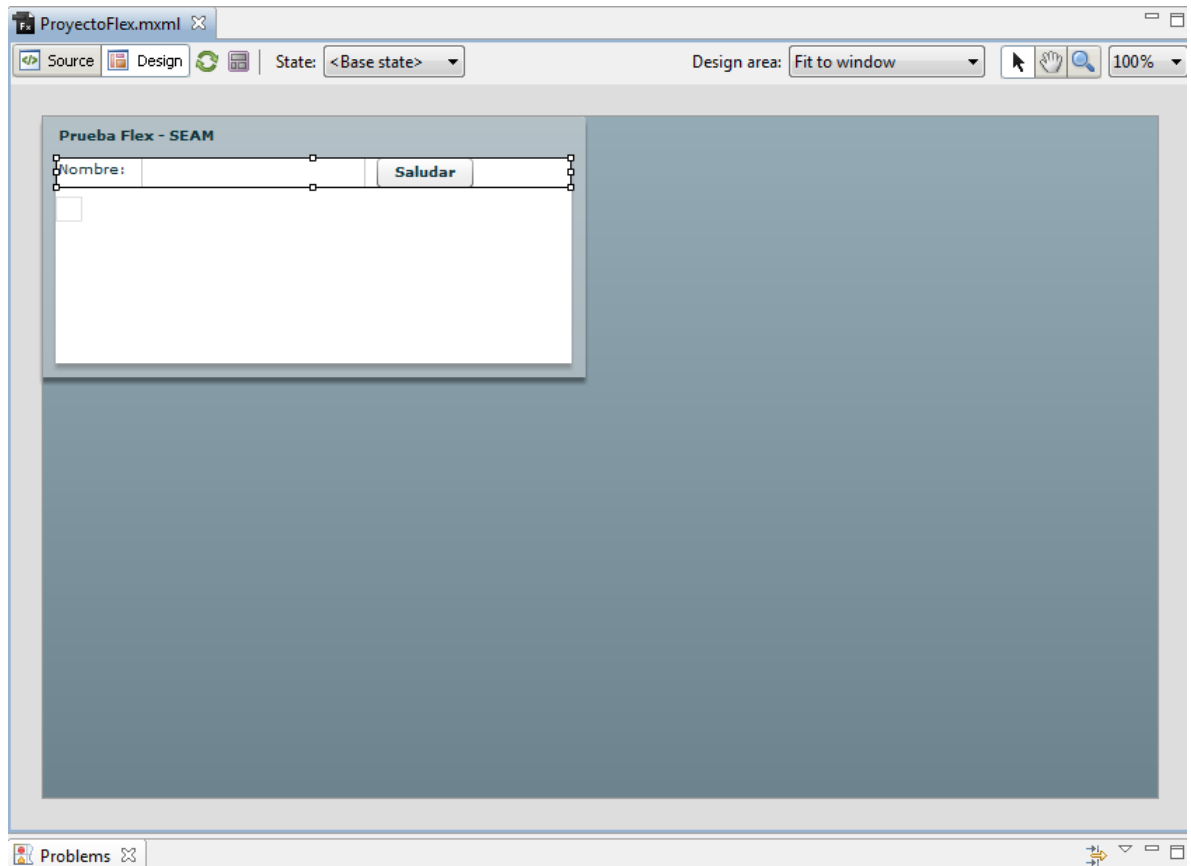
13. Y agregamos el siguiente código para tener una pantalla de pruebas.



```

1 <?xml version="1.0" encoding="utf-8"?>
2 <mx:Application xmlns:mx="http://www.adobe.com/2006/mxml" layout="absolute"
3   xmlns:flamingo="com.exadel.flamingo.flex.components.flamingo.*" >
4
5   <mx:Script>
6     <![CDATA[
7       import mx.rpc.events.ResultEvent;
8       import mx.controls.Alert;
9
10      private function saludar(event:ResultEvent):void
11      {
12        var resultado:String = event.result as String;
13        lblSaludo.text=resultado;
14      }
15    ]]>
16  </mx:Script>
17  <mx:RemoteObject id="seamService" destination="servicioPruebaFlex">
18    <mx:method name="saludo" result="saludar(event)" fault="Alert.show(event.fault.faultString, 'Error')"/>
19  </mx:RemoteObject>
20  <mx:Panel title="Prueba Flex - SEAM" height="187" width="388">
21    <mx:HBox width="100%">
22      <mx:Label text="Nombre:"/>
23      <mx:TextInput id="txtNombre"/>
24      <mx:Button label="Saludar" click="seamService.saludo(txtNombre.text)"/>
25    </mx:HBox>
26    <mx:Label id="lblSaludo"/>
27  </mx:Panel>
28
29 </mx:Application>
30

```



Añadimos una sección de script para añadir nuestro método que se ejecutara cuando el servidor devuelva la respuesta.

```

<mx:Script>
    <![CDATA[
        import mx.rpc.events.ResultEvent;
        import mx.controls.Alert;

        privatefunction saludar(event:ResultEvent):void
        {
            var resultado:String = event.result as String;
            lblSaludo.text=resultado;
        }
    ]>
</mx:Script>
    
```

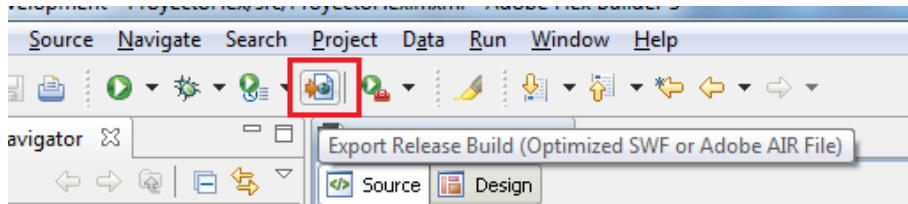
Añadimos el elemento **RemoteObject** que va a realizar el vínculo con nuestro servicio SEAM y con su método además indicamos los métodos a ejecutarse cuando envié la respuesta el servidor y cuando se presente un error.

```
<mx:RemoteObject id="seamService" destination="servicioPruebaFlex">
    <mx:method name="saludo" result="saludar(event)"
        fault="Alert.show(event.fault.faultString,'Error')"/>
</mx:RemoteObject>
```

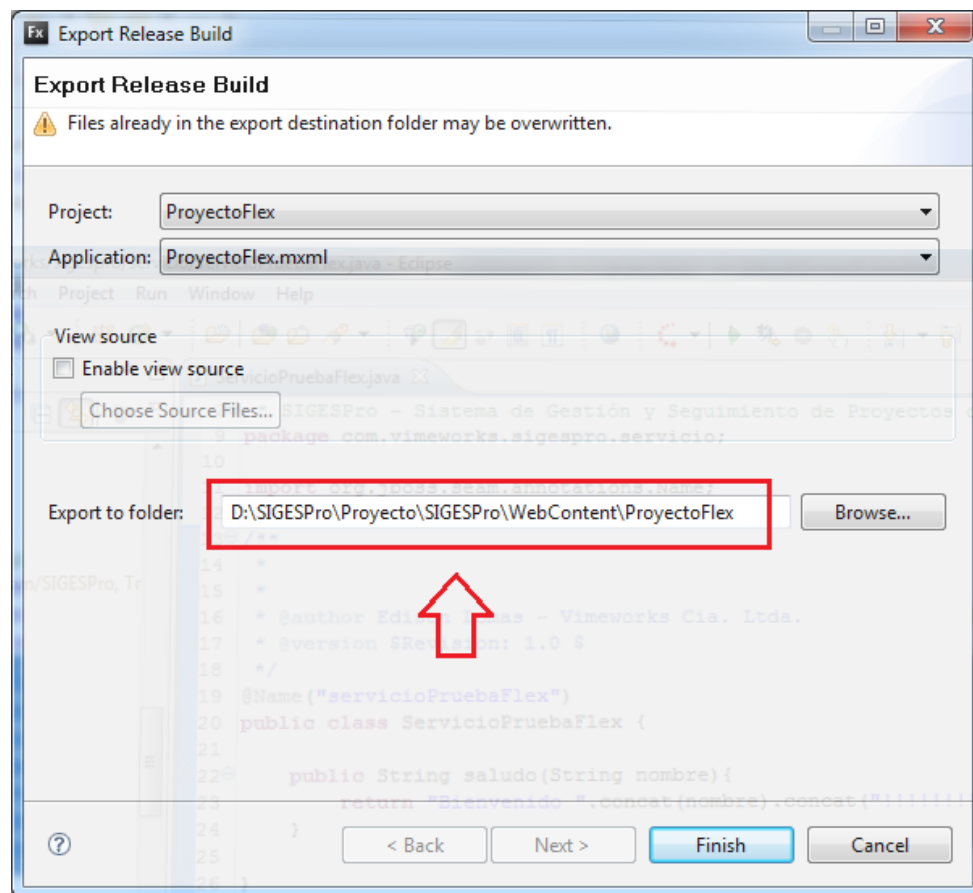
Y por el ultimo el código necesario para crear nuestra ventana y la llamada al método que va a ejecutarse al momento de dar clic en el botón correspondiente.

```
<mx:Panel title="Prueba Flex - SEAM" height="187" width="388">
    <mx:HBox width="100%">
        <mx:Label text="Nombre:"/>
        <mx:TextInput id="txtNombre"/>
        <mx:Button label="Saludar"
            click="seamService.saludo(txtNombre.text)"/>
    </mx:HBox>
    <mx:Label id="lblSaludo"/>
</mx:Panel>
```

14. Damos clic en el botón **Exportar Release Build** de la barra de tarea para exportar nuestro **SWF**.

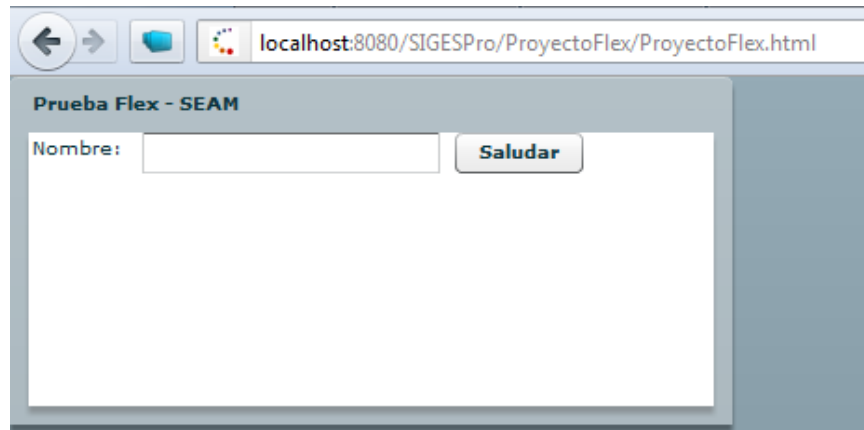


Y nos presenta la siguiente pantalla.

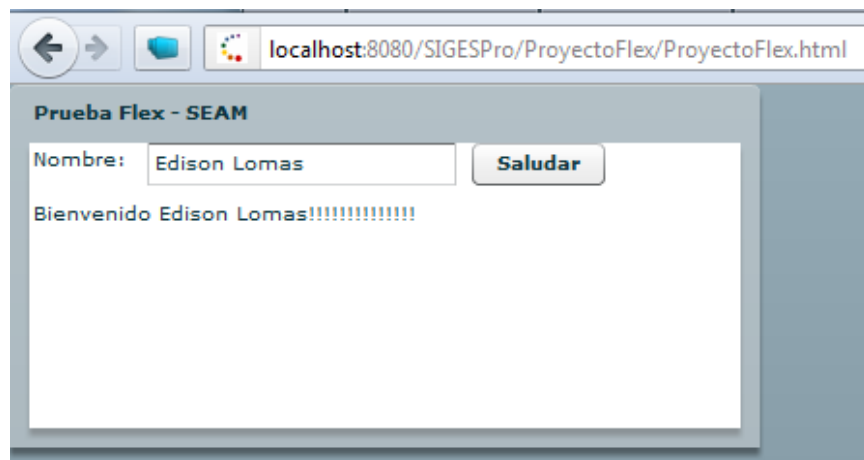


No indica el path a donde se va a exportar, tomar en cuenta que debe ser dentro del WebContet de nuestra aplicación. Y damos clic en **Finish**

15. Levantamos nuestra aplicación y probamos.



Colocamos el nombre y damos clic en el botón **Saludar** y debería presentarnos el mensaje que indicamos en nuestro mensaje indicado en el servicio.



Adicionar el script de creación de la base de datos como anexo, y ejemplos de código fuente de varias partes del sistema (en los casos de uso puede poner algo así: como referencia técnica, observar el anexo de código fuente número XX)